



PENGEMBANGAN SISTEM INFORMASI PENCATATAN DAN PELAPORAN GIZI BALITA

SKRIPSI

Disusun oleh:
Timothy Julian
NIM: 165150400111057



PROGRAM STUDI SISTEM INFORMASI
JURUSAN SISTEM INFORMASI
FAKULTAS ILMU KOMPUTER
UNIVERSITAS BRAWIJAYA
MALANG
2020



PENGESAHAN

PENGEMBANGAN SISTEM INFORMASI PENCATATAN DAN PELAPORAN GIZI BALITA

SKRIPSI

Untuk memenuhi sebagian persyaratan memperoleh gelar Sarjana Komputer

Disusun oleh:

Timothy Julian

NIM: 165150400111057

Skrripsi ini telah diuji dan dinyatakan lulus pada

23 April 2020

Telah diperiksa dan disetujui oleh:

Pembimbing I

Fajar Pradana, S.ST., M.Eng.
NIP: 198711212015041004

Pembimbing II

Dr. Eng. Fitra Abdurrachman Bachtiar, S.T., M.Eng.
NIP: 198406282019031006

Mengetahui

Ketua Jurusan Sistem Informasi



Dr. Eng. Herman Tolle, S.T., M.T.
NIP: 197408232000121001



PERNYATAAN ORISINALITAS

Saya menyatakan dengan sebenar-benarnya bahwa sepanjang pengetahuan saya, di dalam naskah skripsi ini tidak terdapat karya ilmiah yang pernah diajukan oleh orang lain untuk memperoleh gelar akademik di suatu perguruan tinggi, dan tidak terdapat karya atau pendapat yang pernah ditulis atau diterbitkan oleh orang lain, kecuali yang secara tertulis disitasi dalam naskah ini dan disebutkan dalam daftar pustaka.

Apabila ternyata didalam naskah skripsi ini dapat dibuktikan terdapat unsur-unsur plagiasi, saya bersedia skripsi ini digugurkan dan gelar akademik yang telah saya peroleh (sarjana) dibatalkan, serta diproses sesuai dengan peraturan perundang-undangan yang berlaku (UU No. 20 Tahun 2003, Pasal 25 ayat 2 dan Pasal 70).

Malang, 23 April 2020



Timothy Julian

NIM: 165150400111057



ABSTRAK

Timothy Julian, Pengembangan Sistem Informasi Pencatatan dan Pelaporan Gizi Balita

Pembimbing: Fajar Pradana S.ST., M.Eng., dan Dr. Eng. Fitra Abdurrahman Bachtiar S.T., M.Eng.

Posyandu merupakan tempat pelayanan kesehatan yang memiliki tujuan untuk memberikan kemudahan kepada masyarakat sekitar untuk mendapatkan pelayanan kesehatan, khususnya untuk ibu, bayi, dan anak balita. Posyandu sendiri memiliki beberapa jenis layanan seperti pelayanan kesehatan ibu dan anak, imunisasi, penyuluhan gizi, dan pelayanan lainnya. Setelah menyelesaikan proses pelayanan, Posyandu sendiri wajib untuk mencatat hasil pelayanan kesehatan yang telah dilakukan, dan diserahkan kepada pihak bidan untuk dibuatkan laporan untuk Puskesmas. Namun berdasarkan hasil wawancara dengan salah satu dosen gizi dari Fakultas Kedokteran Universitas Brawijaya, bidan Desa Kribet Kabupaten Malang, serta hasil tinjauan langsung pada salah satu Posyandu di Desa Kribet, Kabupaten Malang, didapati bahwa adanya beberapa permasalahan yang dihadapi di Posyandu dan pada pihak bidan. Dikarenakan proses pencatatan hasil pelayanan kesehatan pada Posyandu yang masih dilakukan secara manual, dan menggunakan banyak dokumen yang berbeda-beda dapat membuat proses pencatatan maupun pembuatan laporan gizi oleh bidan akan rentan mengalami kesalahan dan tidak terorganisir dengan baik, yang memungkinkan untuk terjadinya kesalahan indikasi kesehatan anak di wilayah tersebut. Sehingga untuk menyelesaikan permasalahan tersebut, dikembangkanlah sebuah sistem informasi berbasis *website* yang dibuat menggunakan metode SDLC *waterfall* dengan pendekatan berbasis objek, yang dapat mengakomodir dan membantu proses pencatatan hasil pelayanan kesehatan pada Posyandu, khususnya untuk pelayanan kesehatan balita, dan pembuatan laporan gizi balita oleh bidan. Proses analisis kebutuhan menghasilkan 4 aktor, 33 kebutuhan fungsional, dan 1 kebutuhan non fungsional. Dari hasil analisis kebutuhan tersebut dilakukan perancangan dan implementasi sistem menggunakan pendekatan berbasis objek. Pengujian terhadap sistem informasi yang telah dibangun ini menggunakan *validation testing* mendapatkan hasil 100 % *valid*, serta dengan menggunakan *user acceptance testing* (UAT) kepada 5 orang pengguna mendapatkan tingkat penerimaan sebesar 97.1 % yang artinya para pengguna rata-rata sangat setuju terhadap sistem yang telah dibuat.

Kata kunci: Sistem Informasi, Posyandu, Gizi, Balita

ABSTRACT

Timothy Julian, Pengembangan Sistem Informasi Pencatatan dan Pelaporan Gizi Balita

Supervisors: Fajar Pradana S.ST., M.Eng., and Dr. Eng. Fitra Abdurrachman Bachtiar S.T., M.Eng.

Posyandu is a health service facility that has the objective to provide health services for the local society, especially for mothers, babies, and toddler. Posyandu itself has several types of services, such as maternal and child health services, immunization, nutrition counseling, and other type of services. After its service, Posyandu also have to record the results of health services that has been given, and give it to the midwife in charge, so the midwife can make a nutrition report for the Puskesmas. However, based on the interview with one of the nutrition lecturers from Faculty of Medicine Brawijaya University, also with one of the midwife at Krehet Village, Malang Regency, and from the result of the observation on one of the Posyandu at Krehet Village, Malang Regency, it was found that there were several problems faced at the Posyandu and on the midwife also. Because of the process of recording of the health services at the Posyandu is still done manually, and still using many different documents, can make the process of recording and making nutrition reports by the midwife vulnerable to mistakes and disorganized, and can lead to even more mistakes of children's health indication in that area. To solve this problem, a web-based information system created using waterfall SDLC and object oriented approach was developed, that could accommodate and help the process of recording of the health services at Posyandu and making nutrition reports for the midwife. On the requirements analysis phase, produced 4 actors, 33 functional requirements, and 1 non-functional requirements. From the results the requirements analysis phase, the design and implementation process using object-oriented approach of the system carried out. The testing phase conducted using validation testing and got the results of 100% valid, as well as the user acceptance test (UAT) to 5 users got the results of 97.1% acceptance rate, which means that the users strongly agrees to the system that has been made.

Keywords: Information System, Posyandu, Nutrition, Toddler



DAFTAR ISI

PENGESAHAN	ii
PERNYATAAN ORISINALITAS	iii
PRAKATA	iv
ABSTRAK	vi
ABSTRACT	vii
DAFTAR ISI	viii
DAFTAR TABEL	xii
DAFTAR GAMBAR	xvii
DAFTAR LAMPIRAN	xix
BAB 1 PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	3
1.3 Tujuan	3
1.4 Manfaat	3
1.5 Batasan Masalah	4
1.6 Sistematika Pembahasan	4
BAB 2 LANDASAN KEPUSTAKAAN	6
2.1 Kajian Pustaka	6
2.2 Sistem Informasi	8
2.3 Pengembangan Sistem Informasi	9
2.3.1 <i>Waterfall</i>	9
2.4 Pendekatan Berorientasi Objek	10
2.5 Fase Pendefinisian Kebutuhan	11
2.5.1 Pemodelan Proses Bisnis	12
2.5.2 <i>Use Case Diagram</i>	13
2.5.3 Kebutuhan Fungsional	15
2.5.4 Spesifikasi Kebutuhan Fungsional	15
2.5.5 Kebutuhan Non Fungsional	16
2.6 Fase Perancangan Sistem	16
2.6.1 <i>Sequence Diagram</i>	17
2.6.2 <i>Class Diagram</i>	18



2.6.3 Physical Data Model.....	20
2.6.4 Pseudocode.....	21
2.6.5 Wireframe.....	21
2.7 Fase Implementasi Sistem.....	21
2.7.1 Model View Controller (MVC).....	22
2.7.2 Codeigniter (CI).....	22
2.7.3 Bootstrap.....	23
2.8 Fase Pengujian Sistem.....	23
2.8.1 Pengujian Black Box.....	23
2.8.2 User Acceptance Testing.....	24
2.9 Pos Pelayanan Terpadu (Posyandu).....	26
2.10 Kartu Menuju Sehat (KMS).....	26
2.11 Status Gizi.....	28
BAB 3 METODOLOGI.....	30
3.1 Identifikasi Masalah.....	31
3.2 Studi Literatur.....	31
3.3 Analisis Kebutuhan Sistem.....	31
3.4 Perancangan Sistem.....	32
3.5 Implementasi Sistem.....	32
3.6 Pengujian Sistem.....	32
3.7 Kesimpulan dan Saran.....	32
BAB 4 ANALISIS KEBUTUHAN.....	33
4.1 Gambaran Umum Sistem.....	33
4.2 Analisis Proses Bisnis.....	34
4.2.1 Proses Bisnis Penjadwalan Kegiatan Posyandu Saat Ini (<i>as-is</i>).....	34
4.2.2 Proses Bisnis Pelayanan Posyandu Saat Ini (<i>as-is</i>).....	35
4.2.3 Analisis Proses Bisnis Pelaporan Gizi Saat Ini (<i>as-is</i>).....	38
4.2.4 Evaluasi Proses Bisnis Saat Ini (<i>as-is</i>).....	39
4.2.5 Rekomendasi Proses Bisnis Penjadwalan Kegiatan Posyandu (<i>to-be</i>).....	40
4.2.6 Rekomendasi Proses Bisnis Pelayanan Posyandu (<i>to-be</i>).....	41



4.2.7 Rekomendasi Proses Bisnis Pembuatan Laporan Gizi (to- be).....	43
4.2.8 Analisis Hasil Rekomendasi Proses Bisnis	43
4.3 Identifikasi Aktor.....	44
4.4 Kebutuhan Fungsional.....	45
4.5 Spesifikasi Kebutuhan Fungsional.....	47
4.6 Kebutuhan Non Fungsional.....	55
4.7 Pemodelan Kebutuhan	56
4.7.1 <i>Use Case Diagram</i>	56
4.7.2 <i>Use Case Scenario</i>	58
BAB 5 PERANCANGAN DAN IMPLEMENTASI	80
5.1 Perancangan Arsitektur	80
5.1.1 <i>Sequence Diagram</i> Pencatatan Kehadiran dan Hasil Pelayanan Kesehatan Balita	80
5.1.2 <i>Sequence Diagram</i> Menampilkan KMS Balita.....	81
5.1.3 <i>Sequence Diagram</i> Menampilkan Laporan Gizi.....	82
5.1.4 <i>Class Diagram Controller</i>	83
5.1.5 <i>Class Diagram Model</i>	84
5.1.6 Unifikasi <i>Class Diagram</i>	86
5.2 Perancangan Data.....	88
5.3 Perancangan Komponen.....	89
5.3.1 Perancangan Komponen Fungsi Pencatatan Kesehatan Balita	89
5.3.2 Perancangan Komponen Fungsi Menampilkan Kartu Menuju Sehat (KMS) Balita	90
5.3.3 Perancangan Komponen Menampilkan Laporan Gizi.....	92
5.4 Perancangan Antarmuka	96
5.4.1 Perancangan Antarmuka <i>Dashboard User</i>	96
5.4.2 Perancangan Antarmuka <i>Form</i> Pencatatan Kesehatan Balita	97
5.4.3 Perancangan Antarmuka Profil Balita	98
5.4.4 Perancangan Antarmuka Kartu Menuju Sehat (KMS) Balita.....	99
5.4.5 Perancangan Antarmuka Laporan Gizi.....	100



5.5 Implementasi Kode Program	102
5.5.1 Kode Program Pencatatan Kesehatan Balita	102
5.5.2 Kode Program Menampilkan KMS Balita	103
5.5.3 Kode Program Menampilkan Laporan Gizi	107
5.6 Implementasi Basis Data	110
5.7 Implementasi Antarmuka	113
5.7.1 Implementasi Antarmuka <i>Dashboard User</i> Posyandu	113
5.7.2 Implementasi Antarmuka Pencatatan Kehadiran dan Pencatatan Hasil Pelayanan Kesehatan Balita	114
5.7.3 Implementasi Antarmuka Profil Balita	114
5.7.4 Implementasi Antarmuka Halaman Kartu Menuju Sehat (KMS) Balita	115
5.7.5 Implementasi Antarmuka <i>Dashboard User</i> Bidan	116
5.7.6 Implementasi Antarmuka Laporan Gizi	117
BAB 6 PENGUJIAN SISTEM	118
6.1 Pengujian Validasi	118
6.2 <i>User Acceptance Testing</i> (UAT)	147
6.3 Analisis Hasil Pengujian	151
BAB 7 KESIMPULAN DAN SARAN	152
7.1 Kesimpulan	152
7.2 Saran	153
DAFTAR REFERENSI	154
LAMPIRAN A HASIL WAWANCARA	157
LAMPIRAN B STANDAR ANTROPOMETRI PENILAIAN STATUS GIZI PADA ANAK BALITA	161
LAMPIRAN C BERITA ACARA	165
LAMPIRAN D VALIDASI KEBUTUHAN	168
LAMPIRAN E DATA PENDUKUNG	171
LAMPIRAN F HASIL PENGUJIAN USER ACCEPTANCE TESTING (UAT)	173



DAFTAR TABEL

Tabel 2.1 Notasi BPMN	12
Tabel 2.2 Notasi <i>use case diagram</i>	14
Tabel 2.3 Notasi <i>sequence diagram</i>	17
Tabel 2.4 Notasi <i>class diagram</i>	19
Tabel 2.5 Notasi <i>physical data model</i>	20
Tabel 2.6 <i>Pseudocode</i> menentukan angka positif atau negatif	21
Tabel 2.7 Bobot nilai jawaban	25
Tabel 2.8 Rentang presentase nilai kuesioner	25
Tabel 2.9 Kategori status gizi balita	29
Tabel 4.1 Evaluasi proses bisnis <i>as-is</i>	39
Tabel 4.2 Aktivitas proses bisnis	44
Tabel 4.3 Identifikasi aktor	44
Tabel 4.4 Kebutuhan fungsional sistem	45
Tabel 4.5 Spesifikasi kebutuhan fungsional	48
Tabel 4.6 <i>Use case scenario login</i>	58
Tabel 4.7 <i>Use case scenario</i> registrasi bidan	59
Tabel 4.8 <i>Use case scenario</i> menampilkan data bidan aktif	60
Tabel 4.9 <i>Use case scenario</i> memperbaharui data bidan aktif	60
Tabel 4.10 <i>Use case scenario</i> menghapus data bidan aktif	61
Tabel 4.11 <i>Use case scenario</i> menampilkan data bidan yang perlu divalidasi	62
Tabel 4.12 <i>Use case scenario</i> memvalidasi data bidan	62
Tabel 4.13 <i>Use case scenario</i> menjadwalkan hari Posyandu	63
Tabel 4.14 <i>Use case scenario</i> mendaftarkan akun orang tua balita	64
Tabel 4.15 <i>Use case scenario</i> menampilkan data orang tua balita yang terdaftar	65
Tabel 4.16 <i>Use case scenario</i> memperbaharui data orang tua balita yang terdaftar	65
Tabel 4.17 <i>Use case scenario</i> menghapus data orang tua balita yang terdaftar	66
Tabel 4.18 <i>Use case scenario</i> mendaftarkan data balita baru	66
Tabel 4.19 <i>Use case scenario</i> menampilkan data balita yang terdaftar	67
Tabel 4.20 <i>Use case scenario</i> memperbaharui data balita yang terdaftar	67



Tabel 4.21 <i>Use case scenario</i> menghapus data balita yang terdaftar	68
Tabel 4.22 <i>Use case scenario</i> mendata kehadiran kader Posyandu	69
Tabel 4.23 <i>Use case scenario</i> mencatat kehadiran dan hasil pelayanan kesehatan balita.....	69
Tabel 4.24 <i>Use case scenario</i> mendaftarkan Posyandu baru	70
Tabel 4.25 <i>Use case scenario</i> mendaftarkan kader Posyandu.....	71
Tabel 4.26 <i>Use case scenario</i> menampilkan data Posyandu dan kader Posyandu yang terdaftar	72
Tabel 4.27 <i>Use case scenario</i> memperbaharui data Posyandu yang terdaftar	72
Tabel 4.28 <i>Use case scenario</i> menghapus data Posyandu yang terdaftar	73
Tabel 4.29 <i>Use case scenario</i> memperbaharui data kader Posyandu yang terdaftar	73
Tabel 4.30 <i>Use case scenario</i> menghapus data kader Posyandu yang terdaftar .	74
Tabel 4.31 <i>Use case scenario</i> menampilkan laporan gizi dan hasil pelayanan Posyandu	75
Tabel 4.32 <i>Use case scenario</i> mengunduh laporan gizi dan hasil pelayanan Posyandu	75
Tabel 4.33 <i>Use case scenario</i> menampilkan profil balita dan riwayat hasil pemeriksaan	76
Tabel 4.34 <i>Use case scenario</i> memperbaharui data hasil pemeriksaan.....	76
Tabel 4.35 <i>Use case scenario</i> menghapus data hasil pemeriksaan	77
Tabel 4.36 <i>Use case scenario</i> menampilkan KMS Balita	78
Tabel 4.37 <i>Use case scenario</i> memberikan rekomendasi gizi	78
Tabel 4.38 <i>Use case scenario Logout</i>	79
Tabel 5.1 <i>Pseudocode</i> fungsi catatKesehatanBalita()	89
Tabel 5.2 <i>Pseudocode</i> fungsi lihatKMS(ID_Balita)	90
Tabel 5.3 <i>Pseudocode</i> fungsi lihatLaporan()	92
Tabel 5.4 Penjelasan rancangan antarmuka <i>dashboard user</i>	96
Tabel 5.5 Penjelasan rancangan antarmuka <i>form</i> pencatatan kesehatan balita .	97
Tabel 5.6 Penjelasan rancangan antarmuka profil balita	98
Tabel 5.7 Penjelasan rancangan antarmuka Kartu Menuju Sehat (KMS) balita ...	99
Tabel 5.8 Penjelasan rancangan antarmuka laporan gizi	101
Tabel 5.9 Kode program fungsi catatKesehatanBalita()	102
Tabel 5.10 Kode program fungsi LihatKMS(\$ID_balita).....	104



Tabel 5.11 Kode program fungsi linatLaporan()	107
Tabel 5.12 Hasil implementasi basis data	111
Tabel 6.1 Pengujian validasi <i>login</i>	118
Tabel 6.2 Pengujian validasi <i>login</i> alternatif	119
Tabel 6.3 Pengujian validasi registrasi badan	119
Tabel 6.4 Pengujian validasi registrasi badan alternatif	120
Tabel 6.5 Pengujian validasi menampilkan data badan aktif	120
Tabel 6.6 Pengujian validasi memperbaharui data badan aktif	121
Tabel 6.7 Pengujian validasi memperbaharui data badan aktif alternatif	121
Tabel 6.8 Pengujian validasi menghapus data badan aktif	122
Tabel 6.9 Pengujian validasi menghapus data badan aktif alternatif	122
Tabel 6.10 Pengujian validasi menampilkan data badan yang perlu divalidasi	123
Tabel 6.11 Pengujian validasi memvalidasi data badan	123
Tabel 6.12 Pengujian validasi menjadwalkan hari Posyandu	124
Tabel 6.13 Pengujian validasi menjadwalkan hari Posyandu alternatif	124
Tabel 6.14 Pengujian validasi mendaftarkan akun orang tua balita	125
Tabel 6.15 Pengujian validasi mendaftarkan akun orang tua balita alternatif	125
Tabel 6.16 Pengujian validasi menampilkan data orang tua yang terdaftar	126
Tabel 6.17 Pengujian validasi memperbaharui data orang tua balita yang terdaftar	126
Tabel 6.18 Pengujian validasi memperbaharui data orang tua balita yang terdaftar alternatif	127
Tabel 6.19 Pengujian validasi menghapus data orang tua balita yang terdaftar	127
Tabel 6.20 Pengujian validasi menghapus data orang tua balita yang terdaftar alternatif	128
Tabel 6.21 Pengujian validasi mendaftarkan data balita baru	128
Tabel 6.22 Pengujian validasi mendaftarkan data balita baru alternatif	129
Tabel 6.23 Pengujian validasi menampilkan data balita yang terdaftar	129
Tabel 6.24 Pengujian validasi memperbaharui data balita yang terdaftar	130
Tabel 6.25 Pengujian validasi memperbaharui data balita yang terdaftar alternatif	130
Tabel 6.26 Pengujian validasi menghapus data balita yang terdaftar	131
Tabel 6.27 Pengujian validasi menghapus data balita yang terdaftar alternatif	131



Tabel 6.28 Pengujian validasi mendata kehadiran kader Posyandu.....	132
Tabel 6.29 Pengujian validasi mendata kehadiran kader Posyandu alternatif...	132
Tabel 6.30 Pengujian validasi mencatat kehadiran dan hasil pelayanan kesehatan balita.....	133
Tabel 6.31 Pengujian validasi mencatat kehadiran dan hasil pelayanan kesehatan balita alternatif.....	133
Tabel 6.32 Pengujian validasi mendaftarkan Posyandu baru.....	134
Tabel 6.33 Pengujian validasi pendaftaran Posyandu baru alternatif.....	134
Tabel 6.34 Pengujian validasi mendaftarkan kader Posyandu	135
Tabel 6.35 Pengujian validasi pendaftaran kader Posyandu baru alternatif.....	135
Tabel 6.36 Pengujian validasi menampilkan data Posyandu dan kader Posyandu yang terdaftar.....	136
Tabel 6.37 Pengujian validasi memperbaharui data Posyandu yang terdaftar..	136
Tabel 6.38 Pengujian validasi memperbaharui data balita yang terdaftar alternatif.....	137
Tabel 6.39 Pengujian validasi menghapus data Posyandu yang terdaftar.....	137
Tabel 6.40 Pengujian validasi menghapus data balita yang terdaftar alternatif	138
Tabel 6.41 Pengujian validasi memperbaharui data kader Posyandu yang terdaftar.....	138
Tabel 6.42 Pengujian validasi memperbaharui data balita yang terdaftar alternatif.....	139
Tabel 6.43 Pengujian validasi menghapus data kader Posyandu yang terdaftar	139
Tabel 6.44 Pengujian validasi menghapus data kader Posyandu yang terdaftar alternatif.....	140
Tabel 6.45 Pengujian validasi menampilkan laporan gizi dan hasil pelayanan Posyandu.....	140
Tabel 6.46 Pengujian validasi menampilkan laporan gizi dan hasil pelayanan Posyandu alternatif.....	141
Tabel 6.47 Pengujian validasi mengunduh laporan gizi dan hasil pelayanan Posyandu.....	141
Tabel 6.48 Pengujian validasi mengunduh laporan gizi dan hasil pelayanan Posyandu alternatif.....	142
Tabel 6.49 Pengujian validasi menampilkan profil balita dan riwayat hasil pemeriksaan.....	142
Tabel 6.50 Pengujian validasi memperbaharui data hasil pemeriksaan.....	143



Tabel 6.51 Pengujian validasi memperbaharui data hasil pemeriksaan alternatif	143
Tabel 6.52 Pengujian validasi menghapus data hasil pemeriksaan	144
Tabel 6.53 Pengujian validasi menghapus data hasil pemeriksaan alternatif.....	144
Tabel 6.54 Pengujian validasi menampilkan KMS balita	145
Tabel 6.55 Pengujian validasi memberikan rekomendasi gizi	145
Tabel 6.56 Pengujian validasi memberikan rekomendasi gizi alternatif	146
Tabel 6.57 Pengujian validasi <i>logout</i>	146
Tabel 6.58 Pengujian validasi <i>logout</i> alternatif	147
Tabel 6.59 Hasil <i>User Acceptance Testing</i> (UAT) aktor <i>admin</i>	148
Tabel 6.60 Perhitungan nilai kuesioner <i>User Acceptance Testing</i> (UAT) aktor <i>admin</i>	148
Tabel 6.61 Hasil <i>User Acceptance Tesing</i> (UAT) aktor <i>bidan</i>	149
Tabel 6.62 Perhitungan nilai kuesioner <i>User Acceptance Testing</i> (UAT) aktor <i>bidan</i>	149
Tabel 6.63 Hasil <i>User Acceptance Testing</i> (UAT) aktor <i>Posyandu</i>	150
Tabel 6.64 Perhitungan nilai kuesioner <i>User Acceptance Tesing</i> (UAT) aktor <i>Posyandu</i>	150
Tabel 6.65 Hasil <i>User Acceptance Testing</i> (UAT)	151

DAFTAR GAMBAR

Gambar 2.1 Proses <i>waterfall</i>	9
Gambar 2.2 Ilustrasi konsep <i>Model – View – Controller</i>	22
Gambar 2.3 Ilustrasi Struktur Organisasi Posyandu	26
Gambar 2.4 Bagian 1 dan 2 dari KMS.....	27
Gambar 2.5 Bagian 3,4 dan 5 dari KMS	27
Gambar 3.1 Diagram metodologi penelitian	30
Gambar 4.1 Gambaran Umum Sistem	33
Gambar 4.2 Pemodelan proses bisnis penjadwalan kegiatan Posyandu saat ini (<i>as-is</i>)	35
Gambar 4.3 Pemodelan proses bisnis pelayanan Posyandu saat ini (<i>as-is</i>).....	37
Gambar 4.4 Pemodelan proses bisnis pembuatan laporan gizi balita saat ini (<i>as-is</i>).....	38
Gambar 4.5 Rekomendasi proses bisnis penjadwalan kegiatan Posyandu (<i>to-be</i>)	40
Gambar 4.6 Rekomendasi proses bisnis pelayanan Posyandu (<i>to-be</i>).....	42
Gambar 4.7 Rekomendasi proses bisnis pelaporan gizi (<i>to-be</i>).....	43
Gambar 4.8 <i>Use case diagram</i>	57
Gambar 5.1 <i>Sequence diagram</i> pencatatan kehadiran dan hasil pelayanan kesehatan balita.....	81
Gambar 5.2 <i>Sequence diagram</i> menampilkan KMS balita.....	82
Gambar 5.3 <i>Sequence diagram</i> menampilkan laporan gizi	83
Gambar 5.4 <i>Class diagram controller</i>	84
Gambar 5.5 <i>Class diagram model</i>	85
Gambar 5.6 Unifikasi <i>class diagram</i>	87
Gambar 5.7 <i>Physical data model</i>	88
Gambar 5.8 Rancangan antarmuka <i>dashboard user</i>	96
Gambar 5.9 Rancangan antarmuka <i>form</i> pencatatan kesehatan balita.....	97
Gambar 5.10 Rancangan antarmuka profil balita.....	98
Gambar 5.11 Rancangan antarmuka Kartu Menuju Sehat (KMS) balita.....	99
Gambar 5.12 Rancangan antarmuka laporan gizi.....	100
Gambar 5.13 Implementasi antarmuka <i>dashboard user</i> Posyandu	113



Gambar 5.14 Implementasi antarmuka *modal* untuk pencatatan kehadiran dan hasil pelayanan kesehatan balita 114

Gambar 5.15 Implementasi antarmuka profil balita 115

Gambar 5.16 Implementasi antarmuka Kartu Menuju Sehat (KMS) balita 116

Gambar 5.17 Implementasi antarmuka *dashboard user* bidan 116

Gambar 5.18 Implementasi antarmuka laporan gizi 117



DAFTAR LAMPIRAN

LAMPIRAN A HASIL WAWANCARA.....	Error! Bookmark not defined.
A.1 Hasil Wawancara Dengan Intan Yusuf Habibie S.Gz., M.Sc. (Dosen Program Studi Ilmu Gizi FK UB).....	157
A.2 Hasil Wawancara Dengan Komariah Tusokilah A.Md (Bidan Desa Krebet, Kabupaten Malang).....	159
LAMPIRAN B STANDAR ANTROPOMETRI PENILAIAN STATUS GIZI PADA ANAK BALITA.....	Error! Bookmark not defined.
B.1 Standar Berat Badan menurut Umur (BB/U) Balita Laki-Laki Umur 0 – 10 bulan.....	161
B.2 Standar Berat Badan menurut Umur (BB/U) Balita Perempuan Umur 0 – 10 bulan.....	161
B.3 Standar Panjang Badan menurut Umur (PB/U) Balita Laki-Laki Umur 0 – 10 bulan.....	161
B.4 Standar Panjang Badan menurut Umur (PB/U) Balita Perempuan Umur 0 – 10 bulan.....	162
B.5 Standar Tinggi Badan menurut Umur (TB/U) Balita Laki-Laki Umur 24 – 35 bulan.....	162
B.6 Standar Tinggi Badan menurut Umur (TB/U) Balita Perempuan Umur 24 – 35 bulan.....	162
B.7 Standar Berat Badan menurut Panjang Badan (BB/PB) Balita Laki-Laki 45 – 50 cm (untuk balita umur 0 - 24 bulan).....	163
B.8 Standar Berat Badan menurut Panjang Badan (BB/PB) Balita Perempuan 45 – 50 cm (untuk balita umur 0 - 24 bulan).....	163
B.9 Standar Berat Badan menurut Tinggi Badan (BB/TB) Balita Laki- Laki 65 – 70 cm (untuk balita umur > 24 bulan).....	163
B.10 Standar Berat Badan menurut Tinggi Badan (BB/TB) Balita Perempuan 65 – 70 cm (untuk balita umur > 24 bulan).....	164
LAMPIRAN C BERITA ACARA.....	Error! Bookmark not defined.
C.1 Berita Acara Observasi Posyandu.....	165
C.2 Berita Acara Observasi Posyandu.....	166
C.3 Berita Acara Pengujian UAT Kepada Pihak Posyandu.....	167
LAMPIRAN D VALIDASI KEBUTUHAN.....	Error! Bookmark not defined.
D.1 Diskusi dan validasi kebutuhan dengan Bapak Intan Yusuf Habibie S.Gz., M.Sc.....	168
D.2 Contoh <i>Prototype</i> Untuk Validasi Kebutuhan.....	168



D.3 Dokumen Kebutuhan Sistem..... 169

LAMPIRAN E DATA PENDUKUNG **Error! Bookmark not defined.**

E.1 Contoh Laporan Pengukuran Balita Menggunakan Microsoft Excel..... 171

E.2 Contoh Laporan Kehadiran Kader..... 171

E.3 Contoh Undangan Kegiatan Posyandu..... 171

E.4 Contoh Surat Registrasi Bidan 172

E.5 Contoh Register Penimbangan Balita..... 172

LAMPIRAN F HASIL PENGUJIAN USER ACCEPTANCE TESTING (UAT)..... **Error! Bookmark not defined.**

F.1 Hasil Pengujian UAT Pada Admin..... 173

F.2 Hasil Pengujian UAT Pada Bidan..... 174

F.3 Hasil Pengujian UAT Pada Kader Posyandu..... 175



BAB 1 PENDAHULUAN

1.1 Latar Belakang

Indonesia merupakan negara dengan jumlah penduduk terbanyak keempat di dunia dengan sekitar 265 juta jiwa dan berdasarkan data dari Kementerian Kesehatan Republik Indonesia, pada tahun 2018 Indonesia memiliki sekitar 23 juta anak balita dan tentu jumlah generasi penerus bangsa yang sebanyak itu, Indonesia memiliki tanggung jawab yang besar juga dalam memberikan pelayanan kesehatan yang baik sejak masa kecil mereka, khususnya dalam nal gizi. Gizi merupakan suatu proses dari sebuah makhluk hidup untuk menggunakan makanan yang dikonsumsi untuk bertahan hidup, bertumbuh maupun menjalankan fungsi tubuh lainnya secara normal (Pekik, 2006). Status gizi seorang anak juga bisa diklasifikasikan menjadi beberapa kategori, antara lain seperti gizi berlebih, baik, kurang, maupun gizi buruk. Cara untuk menentukan status gizi seorang anak balita dapat dilihat dari jenis kelamin serta 3 indikator, yaitu melalui berat badan menurut umur (BB/U), tinggi badan menurut umur (TB/U), serta berat badan menurut tinggi badan (BB/TB), dimana dari ketiga nilai indikator status gizi tersebut, dibandingkan dengan nilai baku pertumbuhan dari WHO untuk bisa ditentukan status gizi seorang anak (Direktorat Gizi Masyarakat, 2017).

Faktor penyebab gizi buruk pada anak dapat diklasifikasikan menjadi dua penyebab yaitu penyebab secara langsung maupun tidak langsung. Penyebab secara langsung adalah pola konsumsi makan anak tersebut baik dari segi jumlah maupun kualitas pangan, serta penyakit infeksi yang mungkin dideritanya. Sedangkan untuk penyebab tidak langsungnya adalah ketersediaan pangan di rumah tangga, tingkat kemiskinan, maupun pola pikir dan asuh orang tua kepada anak (Hidayat & Jahari, 2012). Kekurangan gizi maupun gizi buruk yang diderita oleh anak balita jika tidak ditangani tentu akan menjadi permasalahan yang serius bagi anak, seperti anak akan lebih rentan terhadap penyakit infeksi seperti kwashiorkor, defisiensi vitamin, marasmus, dan penyakit lainnya yang bisa berdampak buruk untuk proses tumbuh kembang dan kecerdasan seorang anak (Mulliah, et al., 2017).

Data dari kegiatan Pemantauan Status Gizi (PSG) yang dilakukan oleh Direktorat Gizi Masyarakat pada tahun 2017 menunjukkan bahwa masih terdapat 3,8% balita mengalami gizi buruk dan 14% balita Indonesia mengalami gizi kurang (Direktorat Gizi Masyarakat, 2017). Tentu masalah ini bisa dikurangi dan dicegah dengan cara dilakukan pemeriksaan terhadap kesehatan anak secara berkala di Pos Pelayanan Terpadu atau Posyandu, karena terbukti melalui penelitian yang mempelajari mengenai pemanfaatan pelayanan kesehatan di Posyandu dan hubungannya dengan status gizi dan morbiditas balita, menunjukkan bahwa balita yang lebih sering memanfaatkan pelayanan kesehatan di Posyandu oleh orang tuanya memiliki status gizi yang lebih baik dan jarang terkena penyakit (morbiditas) dibandingkan dengan balita yang jarang memanfaatkan pelayanan kesehatan di Posyandu (Hidayat & Jahari, 2012).



Pelayanan kesehatan pada Posyandu mencakup lima kegiatan utama yaitu pelayanan kesehatan ibu dan anak, keluarga berencana, pemberian imunisasi, penyuluhan gizi, serta pencegahan dan penanggulangan penyakit diare (Kementerian Kesehatan RI Pusat Promosi Kesehatan, 2012). Untuk pelayanan anak, khususnya anak balita, pihak Posyandu juga menyediakan Kartu Menuju Sehat (KMS) yang diberikan kepada orang tua sang anak yang ada di dalam buku KIA. Kartu tersebut memiliki fungsi utama sebagai alat untuk memantau pertumbuhan anak melalui pencatatan yang mengacu pada grafik pertumbuhan normal anak, lalu sebagai catatan riwayat pelayanan kesehatan dasar anak seperti pemberian vitamin, ASI, maupun imunisasi, serta yang terakhir adalah alat untuk mengedukasi dari pihak Posyandu kepada pemegang KMS seperti cara perawatan dan pemberian makan anak yang baik (Kementerian Kesehatan Republik Indonesia, 2010).

Menurut wawancara yang telah dilakukan kepada salah satu dosen gizi dari Fakultas Kedokteran Universitas Brawijaya (Lampiran A.1) guna mendapatkan gambaran awal mengenai kondisi permasalahan yang ada di lapangan, yang ditindaklanjuti dengan melakukan wawancara kepada bidan yang bertugas untuk mengawasi Posyandu-Posyandu yang ada di Desa Krebet, Kabupaten Malang (Lampiran A.2), serta melakukan tinjauan langsung pada salah satu Posyandu di desa tersebut (Lampiran C.1) yaitu Posyandu Balita Srikaya, untuk mendapatkan gambaran yang lebih mendalam mengenai kondisi yang ada di Posyandu-Posyandu di desa tersebut, khususnya terkait proses pencatatan dan pelaporan gizi balita. Dari proses tersebut, ditemukan beberapa permasalahan terkait pelayanan kesehatan anak balita di Posyandu, yaitu diantaranya adalah belum adanya sistem yang bisa membantu untuk para kader melakukan proses pencatatan kesehatan anak, sehingga masih dilakukan secara manual menggunakan buku catatan milik Posyandu, dimana nantinya catatan tersebut baru akan direkap oleh bidan yang bertanggungjawab untuk dibuatkan laporannya kepada pihak Puskesmas di daerah tersebut (Tusokilah, 2019). Tentu hal tersebut akan membuat proses pencatatan, maupun pelaporan data kesehatan di tingkat Posyandu akan rentan mengalami kesalahan dan tidak terorganisir, karena bidan harus merekap secara manual data kesehatan balita dari banyak Posyandu yang bisa mengakibatkan adanya kemungkinan untuk terjadi kesalahan indikasi kesehatan anak di wilayah tersebut, seperti dimana seharusnya terindikasi sehat, namun karena adanya kesalahan pada proses pencatatan atau perekapan, anak tersebut terindikasi mengalami gizi buruk maupun sebaliknya. Hal tersebut bisa berujung pada kesalahan pemberian penanganan dan pelaporan mengenai kondisi gizi balita di daerah tersebut oleh pihak Puskesmas (Habibie, 2019).



Pada saat ini, sudah terdapat penelitian yang telah melakukan pengembangan suatu sistem informasi yang menangani kegiatan Posyandu secara keseluruhan seperti pada penelitian oleh Abu Khoiri pada tahun 2008 (Khoiri, 2008). Melalui penelitian ini, akan dibuat sebuah sistem informasi yang serupa dengan penelitian tersebut, dimana dapat mengakomodir proses pelayanan pada Posyandu, namun pada penelitian ini akan berfokus pada pencatatan dan pelaporan hasil pelayanan gizi untuk balita saja, dimana nantinya pihak Posyandu dapat melakukan pencatatan hasil pelayanan gizi dari seorang balita ke dalam sistem, dan informasi tersebut dapat langsung dilihat oleh pihak bidan sehingga tidak perlu merekap kembali hasil pencatatan tersebut secara manual, serta dapat dilihat oleh pihak orang tua balita melalui aplikasi perangkat bergerak berbasis Android pada *smartphone* mereka yang terintegrasi dengan sistem berbasis *web* ini.

1.2 Rumusan Masalah

Berdasarkan permasalahan pada latar belakang yang telah diuraikan, pokok permasalahan dapat dirumuskan sebagai berikut:

1. Bagaimana hasil analisis kebutuhan dari sistem informasi pencatatan dan pelaporan gizi balita yang akan dibangun?
2. Bagaimana hasil rancangan dan implementasi sistem informasi pencatatan dan pelaporan gizi balita?
3. Bagaimana tingkat penerimaan pengguna dan hasil pengujian terhadap sistem informasi pencatatan dan pelaporan gizi balita?

1.3 Tujuan

1. Mengetahui kebutuhan dari sistem informasi pencatatan dan pelaporan gizi balita ini.
2. Merancang dan mengimplementasikan sistem informasi pencatatan dan pelaporan gizi balita.
3. Menguji sistem serta mengetahui tingkat penerimaan pengguna daripada sistem informasi pencatatan dan pelaporan gizi balita ini.

1.4 Manfaat

Manfaat dari penelitian ini yaitu:

1. Membantu pihak Posyandu untuk melakukan pencatatan hasil pelayanan kesehatan pada Posyandu yang lebih terorganisir.
2. Mengurangi kesalahan pada proses pencatatan maupun pembuatan laporan gizi.
3. Memberikan laporan yang valid terkait kondisi kesehatan anak balita dari tingkat Posyandu.



1.5 Batasan Masalah

Batasan masalah pada penelitian ini adalah:

1. Posyandu yang dijadikan dasar acuan penelitian ini adalah Posyandu Balita Srikaya Desa Krebet, Kabupaten Malang.
2. Pengembangan sistem ini menggunakan pendekatan berbasis objek dengan menggunakan *framework* Codeigniter (CI).
3. Sistem yang akan dikembangkan adalah sistem informasi berbasis web.
4. Metode pengembangan yang akan digunakan adalah metode *waterfall*.
5. Sistem informasi yang akan dikembangkan akan berfokus pada pencatatan dan pelaporan hasil pelayanan kesehatan dan gizi balita yang ada di Posyandu kepada pihak bidan.

1.6 Sistematika Pembahasan

Sistematika pembahasan pada skripsi ini adalah sebagai berikut:

BAB I PENDAHULUAN

Pada bab 1 berisi tentang pembahasan mengenai latar belakang masalah akan mengapa dan urgensi akan dibutuhkannya sebuah sistem informasi untuk pencatatan dan pelaporan gizi balita, rumusan masalah yang berisi daftar masalah yang akan dipecahkan dalam penelitian ini, tujuan dan manfaat berisi apa saja yang dapat dihasilkan dan diberikan kepada para *stakeholders* yang ada, batasan masalah berisi mengenai hal-hal yang menjadi pembatas dalam melakukan penelitian ini, dan yang terakhir adalah pembahasan mengenai sistematika penulisan penelitian ini yang berisi urutan penelitian yang akan dilakukan.

BAB II LANDASAN PUSTAKA

Pada bab 2 ini berisi mengenai kajian pustaka yang dijadikan sebagai bahan acuan dalam melakukan penelitian serta penjelasan teori-teori yang mendasari penelitian ini seperti sistem informasi, metode *waterfall*, maupun setiap tahapan dalam proses pengembangan sistem informasi baik dari proses analisis kebutuhan, perancangan, implementasi, hingga pengujian sistem informasi dan juga penjelasan mengenai Posyandu, Kartu Menuju Sehat (KMS), hingga status gizi.

BAB III METODOLOGI PENELITIAN

Pada bab 3, berisi mengenai penjelasan akan metodologi yang akan digunakan dalam penelitian ini, dimana metodologi yang akan digunakan terdiri dari 7 tahap yaitu dimulai dari identifikasi masalah, studi literatur, analisis kebutuhan sistem, perancangan sistem, implementasi sistem, pengujian sistem, serta kesimpulan dan saran.



BAB IV ANALISIS KEBUTUHAN

Pada bab 4 ini akan berisi mengenai hasil analisis kebutuhan yang telah dilakukan melalui proses diskusi, wawancara, maupun peninjauan langsung ke lapangan ke dalam bentuk penggambaran umum sistem untuk menggambarkan sistem yang akan dibangun dalam penelitian ini, analisis proses bisnis yang terdiri dari, analisis proses bisnis saat ini (*as is*), identifikasi permasalahan yang ada pada proses bisnis saat ini, analisis rekomendasi proses bisnis (*to-be*), dan analisis hasil rekomendasi proses bisnis. Berikutnya akan dijelaskan mengenai aktor-aktor yang akan berperan, kebutuhan fungsional beserta spesifikasinya dan kebutuhan non fungsional yang ada dalam sistem, dan terakhir adalah pemodelan kebutuhan ke dalam *use case diagram* dan *use case scenario* untuk setiap kebutuhan fungsional tersebut.

BAB V PERANCANGAN DAN IMPLEMENTASI SISTEM

Pada bab 5 ini berisi mengenai hasil proses perancangan dan hasil implementasi sistem yang telah dilakukan berdasarkan hasil analisis kebutuhan yang telah dijabarkan pada bagian sebelumnya, yang terdiri dari perancangan arsitektur menggunakan *sequence diagram* dan *class diagram*, perancangan data menggunakan *physical data diagram*, perancangan komponen menggunakan *pseudocode*, dan perancangan antarmuka menggunakan *wireframe*. Lalu akan dijabarkan hasil implementasi ke dalam bahasa pemrograman dari hasil rancangan sistem tersebut yang terdiri dari implementasi kode program, basis data, serta implementasi antarmuka.

BAB VI PENGUJIAN SISTEM

Pada bab 6 ini membahas mengenai proses pengujian sistem dengan melakukan pengujian validasi yang dilakukan secara *blackbox* untuk menguji validitas fungsionalitas sistem, serta pengujian menggunakan *user acceptance testing* (UAT) yang diberikan kepada pengguna menggunakan serangkaian pertanyaan berdasarkan beberapa kriteria penerimaan untuk mengetahui tingkat penerimaan sistem oleh para pengguna.

BAB VII KESIMPULAN DAN SARAN

Pada bab 7 ini merupakan hasil evaluasi secara keseluruhan dari proses penelitian yang telah dilakukan ke dalam bentuk kesimpulan dan saran untuk penelitian selanjutnya.



BAB 2 LANDASAN KEPUSTAKAAN

Bab ini memuat teori-teori dasar yang berhubungan dengan penelitian ini yang terdiri dari tinjauan pustaka terhadap penelitian-penelitian relevan yang telah dilakukan sebelumnya untuk dijadikan sebagai perbandingan dalam menjalankan penelitian ini, hingga dasar-dasar teori yang bisa digunakan untuk mendukung proses pengembangan sistem informasi pencatatan dan pelaporan gizi balita ini.

2.1 Kajian Pustaka

Dalam melakukan penelitian ini, perlu untuk melakukan pendalaman mengenai penelitian-penelitian yang telah dilakukan sebelumnya yang berkaitan dengan topik pada penelitian ini dan dapat digunakan sebagai dasar acuan maupun perbandingan dalam melakukan penelitian ini.

Penelitian pertama merupakan penelitian yang dilakukan oleh Faisal Anwar, Ali Khomsan, Dadang Sukandar, Hadi Riyadi dan Eddy S. Mudjajanto pada tahun 2010 dari Institut Pertanian Bogor dengan judul "*High participation in the Posyandu Nutrition Program Improved Children Nutritional Status*" pada jurnal *Nutrition Research and Practice*, dimana pada penelitian dilakukan analisis terhadap status gizi dan konsumsi makanan pada 300 anak balita di daerah Cianjur, Jawa Barat dimana dari 300 anak balita tersebut terbagi menjadi dua kategori, yaitu yang memiliki tingkat partisipasi tinggi serta yang memiliki tingkat partisipasi rendah dalam mengikuti program gizi oleh Posyandu. Dari penelitian ini, didapatkan hasil bahwa program gizi yang diselenggarakan oleh Posyandu memiliki dampak yang positif terhadap status gizi balita, jika diukur berdasarkan berat badan / umur (BB/U) dan berat badan / tinggi (BB/TB), dan semakin tinggi frekuensi anak balita berkunjung ke Posyandu, semakin baik status gizi anak tersebut (Anwar, et al., 2010).

Penelitian berikutnya adalah penelitian yang dilakukan oleh Tjetjep Syarif Hidayat dan Abas Basuni Jahari pada tahun 2012 dengan judul "Perilaku Pemanfaatan Posyandu Hubungannya Dengan Status Gizi Dan Morbiditas Balita" pada jurnal Buletin Penelitian Kesehatan, yang merupakan studi deskriptif analitik dengan menggunakan data sekunder dari Riskesdas (Riset Kesehatan Dasar) Departemen Kesehatan Republik Indonesia pada tahun 2007/2008 mengenai hubungan pemanfaatan pelayanan kesehatan di Posyandu dengan status gizi dan morbiditas balita dengan jumlah sampe sebanyak 70.210 rumah tangga. Dari penelitian ini, didapatkan kesimpulan bahwa balita yang lebih sering memanfaatkan pelayanan kesehatan di Posyandu oleh orang tuanya, memiliki status gizi yang lebih baik dan jarang terkena penyakit (morbiditas) dibandingkan dengan balita yang jarang memanfaatkan pelayanan kesehatan di Posyandu (Hidayat & Jahari, 2012).



Penelitian berikutnya adalah penelitian dari Endah Sudarmilah, Hasyim Asyari, Geri Gebyar Giwangkoro pada tahun 2011 dengan judul "*Web Based Information System for Motoric and Mental Development in Early Childhood*" yang diambil dari *proceeding* pada *1st International Conference on Information System for Business Competitiveness (ICISPC)* yang diadakan oleh Universitas Diponegoro. Pada penelitian ini para peneliti membuat suatu sistem pendukung keputusan berbasis web dengan menggunakan metode pengembangan sistem *Waterfall* yang terdiri dari 6 tahap pengembangan sistem yaitu pendefinisian kebutuhan, analisis kebutuhan, desain dan perancangan sistem, koding sistem, pengujian dan integrasi sistem, serta implementasi dan pemeliharaan sistem. Untuk metode pengambilan keputusan yang digunakan dalam menentukan perkembangan status gizi anak oleh sistem adalah *Analytical Hierarchy Process (AHP)* yang merupakan metode pengambilan keputusan dengan menggunakan beberapa kriteria dasar. Dari penelitian ini, didapatkan hasil yaitu sebuah sistem pendukung keputusan berbasis web yang dapat digunakan dan membantu para petugas Posyandu, dan orang tua anak balita, untuk bisa memantau mengambil keputusan yang lebih baik terkait perkembangan status gizi anak tersebut berdasarkan informasi yang didapatkan dari sistem (Sudarmilah, et al., 2011).

Penelitian terakhir adalah penelitian yang dilakukan oleh Abu Khoiri pada tesisnya yang berjudul "*Pengembangan Sistem Informasi Posyandu Guna Mendukung Surveilans Kesehatan Ibu & Anak Berbasis Masyarakat Pada Desa Siaga (Studi Kasus di Kelurahan Manisrejo Kecamatan Taman Kota Madiun Provinsi Jawa Timur)*" pada tahun 2008. Pada penelitian ini terdapat beberapa permasalahan yang dialami oleh Posyandu di kelurahan Manisrejo, seperti kader mengalami kesulitan dalam mengisi laporan, belum jelasnya informasi yang diberikan oleh laporan hasil kegiatan Posyandu dan permasalahan lainnya. Sehingga dilakukan pengembangan Sistem Informasi Posyandu (SIP) yang bisa mendukung proses surveilans kesehatan ibu dan anak dan memberikan kualitas informasi dari laporan Posyandu secara keseluruhan yang lebih baik (Khoiri, 2008).

Keterkaitan penelitian pertama dan kedua yang berjudul "*High Participation in The Posyandu Nutrition Program Improved Children Nutritional Status*" dan "*Perilaku Pemanfaatan Posyandu Hubungannya Dengan Status Gizi Dan Morbiditas Balita*" dengan penelitian ini adalah dimana penelitian tersebut dapat digunakan sebagai dasar acuan mengenai pentingnya melakukan pemeriksaan kesehatan dan mengikuti program gizi yang diselenggarakan oleh Posyandu, terhadap status gizi dan kesehatan secara keseluruhan seorang anak balita. Sedangkan keterkaitan pada penelitian ketiga yang berjudul "*Web Based Information System for Motoric and Mental Development in Early Childhood*" dengan penelitian ini adalah ada pada metode pengembangan sistem yang digunakan yaitu *waterfall* dan sama-sama menggunakan *website* sebagai dasar *platform* pengembangan sistem yang digunakan.



Untuk keterkaitan penelitian terakhir yang berjudul "Pengembangan Sistem Informasi Posyandu Guna Mendukung Surveilans Kesehatan Ibu & Anak Berbasis Masyarakat Pada Desa Siaga (Studi Kasus di Kelurahan Manisrejo Kecamatan Taman Kota Madiun Provinsi Jawa Timur)" dengan penelitian ini adalah ada pada topik penelitian yang dilakukan, yaitu sama-sama meneliti mengenai pengembangan sistem informasi untuk Posyandu, letak perbedaannya ada pada cakupan penelitian, dimana pada penelitian ini berfokus pada pencatatan dan pelaporan gizi serta hasil pelayanan untuk balita, yang telah dilakukan oleh pihak Posyandu kepada pihak bidan pada wilayah tersebut dan informasi tersebut diakses oleh pihak orang tua balita melalui aplikasi perangkat bergerak berbasis android melalui *smartphone* mereka. Selain itu, sistem yang akan dikembangkan dalam penelitian ini juga tidak terbatas penggunaannya hanya pada satu Posyandu saja, melainkan dapat digunakan oleh beberapa Posyandu yang berbeda melalui jaringan internet.

2.2 Sistem Informasi

Sistem informasi disini merupakan suatu sistem yang berada di dalam suatu organisasi yang memiliki peran untuk menerima *input*, melakukan proses, serta menghasilkan *output* yang berupa informasi yang bermanfaat bagi penggunanya untuk membuat pelayanan kegiatan operasional organisasi yang lebih efektif dan efisien (Hutahean, 2015). Sistem informasi merupakan salah satu hasil dari proses rekayasa perangkat lunak, atau sering disebut dengan *software engineering*. Rekayasa perangkat lunak sendiri merupakan suatu disiplin ilmu yang memperhatikan semua aspek dalam proses membuat suatu perangkat lunak, mulai dari proses analisis spesifikasi sistem hingga proses pemeliharaan setelah sistem sudah selesai dibuat (Sommerville, 2011). Dalam proses rekayasa perangkat lunak, terdapat suatu pendekatan sistematis yang disebut dengan proses perangkat lunak, dimana terdapat empat aktivitas yang mendasari proses ini yaitu,

1. Spesifikasi perangkat lunak, pada aktivitas ini pihak pengguna dan pihak pengembang sama-sama mendefinisikan perangkat lunak seperti apa yang akan dikembangkan dan batasan-batasan daripada perangkat lunak tersebut.
2. Pengembangan perangkat lunak, tahap dimana dilakukan proses perancangan dan pemrograman perangkat lunak.
3. Validasi perangkat lunak tahap untuk memastikan bahwa perangkat lunak yang telah dibuat sudah sesuai dengan kebutuhan pengguna.
4. Evolusi perangkat lunak, tahap dimana perangkat lunak dikembangkan lebih lanjut berdasarkan kebutuhan pengguna maupun pasar.

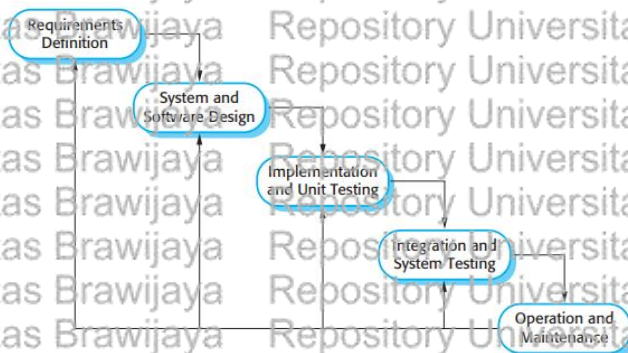


2.3 Pengembangan Sistem Informasi

Dalam proses pengembangan sistem informasi, akan dibutuhkan serangkaian pendekatan sistematis yang secara garis besar dapat merepresentasikan apa saja yang akan dilakukan dalam pengembangan sistem informasi tersebut dalam menyelesaikan suatu permasalahan. Pendekatan tersebut disebut dengan siklus pengembangan perangkat lunak atau dikenal dengan *Software Development Life Cycle* (SDLC). Sommerville menjelaskan bahwa terdapat tiga model pengembangan sistem yang umum digunakan, yaitu model *waterfall*, *incremental development*, serta *reuse-oriented software engineering* (Sommerville, 2011).

2.3.1 Waterfall

Waterfall merupakan salah satu model *Software Development Life Cycle* (SDLC) yang berfokus pada perencanaan matang di awal untuk semua proses dan aktivitas sebelum dimulai aktivitas tersebut. Metode *waterfall* sendiri merupakan metode yang linear sekuensial dimana setiap tahap dijalankan secara berurutan dan fase berikutnya sebaiknya tidak dimulai sebelum fase sebelumnya terselesaikan. Meskipun metode *waterfall* ini merupakan metode rekayasa perangkat lunak yang menekankan pada pengerjaan secara bertahap, pada praktiknya, setiap fase dapat memberikan umpan balik kepada fase sebelumnya maupun fase berikutnya, sehingga metode ini tidak sepenuhnya linear sekuensial, melainkan bisa dilakukan modifikasi terhadap hasil dari fase sebelumnya jika diperlukan (Sommerville, 2011).



Gambar 2.1 Proses *waterfall*

Sumber: Sommerville (2011)

Dalam Gambar 2.1 tersebut dijabarkan terkait tahapan pada metode *waterfall* yang terdiri dari 5 fase yaitu pendefinisian kebutuhan, desain sistem, implementasi dan pengujian unit, integrasi dan pengujian sistem, serta pengoperasian dan pemeliharaan sistem.

1. Pendefinisian Kebutuhan

Pada tahap ini, dilakukan proses penggalian kebutuhan kepada pengguna sistem terkait fungsi, batasan-batasan, maupun tujuan dari sistem yang akan dibuat, dimana hasil atau keluaran dari tahap ini adalah dokumentasi kebutuhan sistem.



2. Desain sistem

Tahap berikutnya setelah mendapatkan kebutuhan sistem yang jelas adalah mendesain secara keseluruhan sistem yang akan dibangun melalui diagram-diagram pendukung.

3. Implementasi dan pengujian unit

Setelah desain arsitektur sistem telah selesai, maka sistem dapat diimplementasikan menggunakan bahasa pemrograman yang digunakan untuk membangun sistem tersebut ke dalam bentuk unit-unit sistem, serta dilakukannya pengujian unit untuk memastikan bahwa setiap unit yang dibuat telah sesuai dengan spesifikasi sistem.

4. Integrasi dan pengujian sistem

Tahap berikutnya adalah mengintegrasikan dan menguji unit-unit sistem tersebut menjadi sebuah sistem yang utuh untuk memastikan bahwa secara keseluruhan sistem telah sesuai dengan kebutuhan yang telah ditentukan. Setelah sistem telah sesuai, maka sistem dapat digunakan oleh pengguna.

5. Pengoperasian dan pemeliharaan sistem

Tahap terakhir ini adalah fase pemeliharaan sistem yang telah digunakan oleh pengguna, dimana selama fase ini dapat dilakukan proses perbaikan maupun pengembangan lanjut sistem jika dibutuhkan oleh pengguna.

2.4 Pendekatan Berorientasi Objek

Pendekatan berorientasi objek merupakan salah satu paradigma dalam pembangunan sistem yang memodelkan bahwa sistem tersusun dari sekumpulan objek yang saling berinteraksi satu dengan yang lainnya, dimana setiap objek memiliki fungsi dan datanya masing-masing. Pendekatan berorientasi objek sendiri mencakup tiga proses yaitu *object oriented analysis* (OOA), *object oriented design* (OOD), serta *object oriented implementation* (OOI). Proses *object oriented analysis* merupakan tahap dimana akan dilakukan proses analisis kebutuhan kepada para *stakeholders* yang berhubungan dengan sistem yang akan dibangun terkait kebutuhan sistem dalam bentuk model *use case*. Tahap berikutnya pada *object oriented design* merupakan tahap untuk melakukan perancangan dari sistem yang dibangun menggunakan diagram-diagram UML pada perancangan berorientasi objek seperti *class diagram*, dan *sequence diagram*. Serta yang terakhir adalah mengimplementasikan hasil analisis dan perancangan tersebut ke dalam bahasa pemrograman yang digunakan dengan menggunakan konsep pendekatan berorientasi objek juga (Glasser, 2009).



Dalam mengembangkan sistem dengan menggunakan pendekatan berbasis objek memiliki kelebihanannya diantara lain:

1. Tingkat *reusability* kode program yang lebih baik.
2. Lebih mudah untuk memahami sistem karena pemodelan yang didasarkan pada entitas di dunia nyata.
3. Proses *maintenance* dan perubahan terhadap sistem yang lebih cepat dan mudah.

2.5 Fase Pendefinisian Kebutuhan

Proses pendefinisian kebutuhan ini merupakan tahap untuk mengetahui dan mendefinisikan fungsi apa saja yang dapat dilakukan oleh sistem dan batasan-batasannya. Tujuan dari dilakukannya proses ini adalah untuk menghasilkan sebuah dokumen kebutuhan yang di dalamnya terdapat spesifikasi sistem yang memenuhi kebutuhan para *stakeholder* dan disetujui oleh pihak-pihak terkait. Terdapat 4 aktivitas pada proses ini, yang pertama adalah *feasibility study* yaitu aktivitas untuk menentukan apakah sistem yang akan dibangun layak untuk dilanjutkan atau tidak. Lalu berikutnya adalah *requirements elicitation and analysis* yaitu tahap untuk menentukan kebutuhan dari sistem berdasarkan pengamatan dan diskusi kepada para *stakeholders*. Berikutnya adalah *requirements specification* yaitu aktivitas untuk menspesifikasikan kebutuhan yang telah dianalisis sebelumnya ke dalam suatu dokumen, dimana di dalamnya terdapat kebutuhan sistem, dan spesifikasi kebutuhan sistem tersebut. Serta yang terakhir adalah *requirements validation* yaitu tahap untuk memastikan bahwa semua kebutuhan yang telah didefinisikan sudah sesuai dan lengkap (Sommerville, 2011).






Untuk sistem yang memiliki kebutuhan yang sederhana dan jelas, proses analisis kebutuhan juga dapat dilakukan secara sederhana seperti diskusi melalui telepon dan mencatat poin-poin penting terkait kebutuhan pengguna tersebut, lalu kemudian memberikan pernyataan kebutuhan untuk diulas dan diterima oleh pengguna (Pressman, 2010). Pada penelitian ini akan dilakukan proses analisis kebutuhan dengan melakukan proses diskusi, wawancara, serta observasi kepada pihak-pihak yang terkait dengan sistem ini seperti badan, kader Posyandu, maupun dengan salah satu dosen gizi FK UB, untuk mendapatkan gambaran kebutuhan sistem serta proses bisnis pelayanan kesehatan gizi di Posyandu, dan untuk memvalidasi hasil analisis tersebut dari sistem yang akan dibangun guna menghasilkan sistem yang tepat guna.



2.5.1 Pemodelan Proses Bisnis

Proses bisnis merupakan serangkaian aktivitas bisnis yang merepresentasikan langkah-langkah yang dibutuhkan untuk mencapai suatu tujuan bisnis. Di dalam proses bisnis juga terdapat aliran informasi dan sumber daya yang digunakan dari satu aktivitas ke aktivitas lainnya. Suatu proses bisnis bisa dilakukan dalam sebuah organisasi saja, namun bisa juga berinteraksi dengan proses bisnis yang dilakukan oleh organisasi lain. Dalam memodelkan sebuah proses bisnis, dapat menggunakan notasi-notasi yang sudah terstandarisasi yaitu *Business Process Modelling Notation* (BPMN). BPMN digunakan untuk memodelkan proses bisnis sebuah organisasi ke dalam notasi-notasi sehingga dapat dimengerti oleh para *stakeholders* akan bagaimana alur proses bisnis dari sebuah organisasi maupun interaksi yang dilakukan organisasi tersebut dengan organisasi lain dalam mencapai tujuan bisnis (Weske, 2007). Pada Tabel 2.1 dijabarkan mengenai notasi-notasi yang terdapat pada BPMN, seperti adanya elemen *event*, *activity*, *gateway*, *sequence flow*, *message flow*, *association*, *pool*, *lane*, *data object*, *message*, *group*, serta *text annotation* yang dapat digunakan untuk memodelkan proses bisnis dari suatu organisasi (Object Management Group, 2011).

Tabel 2.1 Notasi BPMN

Notasi	Keterangan
	Event adalah notasi yang menggambarkan sebuah kejadian yang akan berjalan selama proses bisnis. <i>Event</i> bisa mempengaruhi, menjadi pemicu, atau sebuah hasil dari proses bisnis yang berjalan. Terdapat 3 jenis <i>event</i> pada BPMN, yaitu <i>start</i> , <i>intermediate</i> , dan <i>end</i> .
	Activity merupakan representasi dari kegiatan yang dilakukan pada proses bisnis tersebut. Sebuah <i>activity</i> bisa terdiri dari aktivitas aktivitas lain didalamnya (<i>non-atomic</i>) atau hanya berdiri sendiri (<i>atomic</i>). Dalam sebuah <i>activity</i> yang <i>atomic</i> , atau sebuah <i>task</i> , terdapat beberapa jenis yaitu seperti <i>service task</i> , <i>send task</i> , <i>receive task</i> , <i>user task</i> , <i>manual task</i> , <i>business rule task</i> , dan <i>script task</i> .
	Gateway merupakan notasi yang merepresentasikan sebuah percabangan maupun penggabungan dari sebuah aliran aktivitas yang ada dalam proses bisnis.
	Sequence Flow merupakan notasi yang menjelaskan urutan aktivitas yang akan dijalankan dalam satu proses.
	Message Flow merupakan notasi yang menjelaskan pertukaran pesan yang terjadi antara entitas yang berbeda.

Sumber: Object Management Group (2011)



Tabel 2.1 Notasi BPMN (lanjutan)

Notasi	Keterangan
	Association merupakan sebuah notasi yang dapat digunakan untuk menghubungkan elemen BPMN dengan elemen informasi dan artefak
	Pool merupakan notasi yang menggambarkan sebuah entitas dalam suatu bisnis. Pool dapat mewakili sebuah organisasi atau instansi.
	Lane merupakan sebuah notasi yang menjelaskan tentang partisi dalam suatu <i>pool</i> . Lane dapat digunakan untuk merepresentasikan sebuah entitas lain yang masih terdapat di dalam sebuah organisasi.
	Data Object merupakan sebuah notasi yang menjabarkan tentang data yang dibutuhkan atau data yang menjadi hasil dalam sebuah aktivitas.
	Message digunakan untuk menggambarkan komunikasi antar beberapa entitas atau komunikasi yang dilakukan dalam sebuah entitas.
	Group merupakan sebuah notasi yang dapat digunakan untuk mengelompokkan atau mengkategorikan sebuah aktivitas tertentu yang memiliki kesamaan.
	Text Annotation merupakan notasi yang dapat digunakan untuk menjelaskan sebuah elemen dalam BPMN.

Sumber: Object Management Group (2011)

2.5.2 Use Case Diagram

Diagram *use case* ini merupakan suatu diagram yang memiliki fungsi utama untuk memodelkan aktor-aktor yang akan berperan di dalam sistem dan apa saja interaksi yang bisa aktor tersebut lakukan dengan sistem. Diagram ini utamanya digunakan sebagai alat dalam proses penggalan kebutuhan sistem (Sommerville, 2011). Pada pemodelan *use case diagram* terdapat dua elemen utama yang harus ada, yaitu aktor dan *use case*, dimana aktor merupakan segala sesuatu yang dapat berinteraksi secara langsung dengan sistem, yang diberi nama sesuai dengan perannya. Sedangkan *use case* sendiri merupakan suatu notasi berbentuk elips dengan menggunakan kata kerja aktif di dalam elips tersebut yang menyatakan



aktivitas dari perspektif aktor. Setiap *use case* yang dibuat harus bisa merepresentasikan sebuah tujuan spesifik yang ingin dicapai oleh aktor tersebut, sehingga dalam pembuatan *use case* harus dihindari pembuatan *fat use case* atau *use case* yang masih abstrak seperti “Kelola data pengguna” dikarenakan *use case* tersebut tidak memiliki tujuan yang spesifik dan jelas (Kurniawan, 2018).


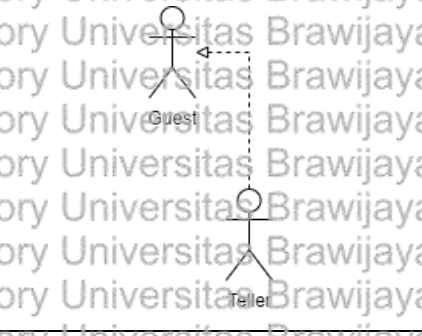
Dalam membuat *use case* diagram, diperlukan juga *use case scenario* yang merupakan penjelasan dari alur kegiatan dari perspektif pengguna dalam menggunakan sistem. *Use case scenario* ini terdiri dari beberapa bagian yaitu *aktor* atau pengguna pada *use case* tersebut, *objective* atau tujuan akhir setelah *use case* tersebut berhasil dilakukan, *pre-condition* atau kondisi yang harus terpenuhi sebelum sebelum sebuah *use case* bisa dimisiasi atau dieksekusi oleh aktor, *main flow* merupakan alur utama dari *use case* yang mengarahkan pada skenario yang berhasil, *alternative flow* merupakan jalur alternatif interaksi aktor dengan sistem yang menggambarkan percabangan atau pilihan yang bisa diambil oleh aktor maupun skenario yang gagal, dan terakhir terdapat *post-condition* yang merupakan kondisi spesifik yang terjadi setelah *use case* berhasil dijalankan (Kurniawan, 2018). Pada Tabel 2.2 dijelaskan mengenai notasi-notasi yang terdapat pada *use case* diagram, seperti adanya notasi *use case*, *actors*, *association*, *include relationship*, *extend relationship*, dan *generalize relationship*.

Tabel 2.2 Notasi *use case diagram*

Notasi	Keterangan
	Use case dapat digambarkan menggunakan bentuk elips yang didalamnya terdapat nama dari <i>use case</i> tersebut, dimana <i>use case</i> berfungsi untuk menentukan apa yang bisa dilakukan, dan yang dapat menghasilkan hasil yang dapat diobservasi bagi aktor tersebut.
	Actor merupakan entitas yang merepresentasikan sebuah peran yang bisa berinteraksi dengan <i>use case</i> . Aktor sendiri dapat merepresentasikan berbagai peran seperti pengguna manusia, perangkat keras, maupun sistem lain.
	Association merupakan sebuah garis yang dapat menggambarkan hubungan antara aktor dengan <i>use case</i> .
	Include relationship atau disebut juga <i>uses relationship</i> menggambarkan perilaku atau kegiatan dalam suatu <i>use case</i> akan termasuk dalam suatu <i>use case</i> lainnya. <i>Include relationship</i> ini dapat digunakan untuk menyederhanakan <i>use case</i> menjadi beberapa <i>use case</i> .

Sumber: Object Management Group (2017)

Tabel 2.2 Notasi *use case diagram* (lanjutan)

Notasi	Keterangan
	<p>Extend relationship, bertujuan untuk menggambarkan situasi dimana sebuah <i>use case</i> memiliki hubungan dengan <i>use case</i> tambahan lain namun, <i>use case</i> utama ini dapat berdiri sendiri, dan <i>use case</i> tambahan tersebut berupa <i>use case</i> yang opsional untuk dilakukan.</p>
	<p>Generalizes relationship menunjukkan suatu hubungan antara dua entitas, dimana satu entitas akan lebih umum (<i>superclass</i>) dan satu kelas akan lebih spesifik (<i>subclass</i>). Hubungan ini bisa terjadi antara hubungan antar aktor maupun antar <i>use case</i>.</p>

Sumber: Object Management Group (2017)

2.5.3 Kebutuhan Fungsional

Kebutuhan sistem sendiri dapat dibagi menjadi dua jenis yaitu kebutuhan fungsional dan kebutuhan non fungsional sistem. Kebutuhan fungsional sistem disini merupakan pernyataan-pernyataan yang menjelaskan apa saja yang bisa sistem lakukan, merespon dari *input* tertentu, maupun bagaimana sistem akan berfungsi secara keseluruhan. Dalam menentukan kebutuhan fungsional sebuah sistem, diusahakan untuk bisa menghasilkan kebutuhan yang lengkap dan konsisten. Arti lengkap disini berarti semua kebutuhan sistem telah terdefiniskan dengan baik. Sedangkan arti konsisten berarti semua kebutuhan yang terdefiniskan tersebut tidak memiliki arti yang saling bertolak belakang satu dengan yang lain (Sommerville, 2011).

2.5.4 Spesifikasi Kebutuhan Fungsional

Setelah kebutuhan fungsional telah terdefiniskan, diperlukan untuk menspesifikasikan atau mendeskripsikan lebih lanjut daripada kebutuhan fungsional tersebut ke dalam spesifikasi kebutuhan fungsional. Pada spesifikasi kebutuhan fungsional ini, akan mendeskripsikan dengan lebih rinci kebutuhan fungsional yang telah didefinisikan tersebut seperti apabila salah satu kebutuhan fungsional dari sistem adalah untuk memproses suatu data, maka pada spesifikasi kebutuhan fungsional tersebut akan mendeskripsikan data apa saja yang akan diperlukan untuk memenuhi kebutuhan fungsional tersebut (Sommerville, 2011).



2.5.5 Kebutuhan Non Fungsional

Kebutuhan non fungsional merupakan kebutuhan yang lebih menitikberatkan pada perilaku maupun batasan-batasan yang harus dipenuhi dari sebuah sistem yang akan dibangun. Kebutuhan non fungsional ini umumnya meliputi kebutuhan akan keandalan sistem, keamanan sistem, kecepatan respon sistem, hingga *usability* sistem yang baik juga bisa termasuk ke dalam kebutuhan non fungsional. Kebutuhan non fungsional juga bisa termasuk kebutuhan yang kritis atau penting dari sebuah sistem, karena meskipun saat sistem telah memenuhi semua kebutuhan fungsional yang telah terdefiniskan dengan baik, namun gagal untuk memenuhi kebutuhan non fungsional, akan membuat pengguna sulit atau bahkan tidak menggunakan sistem tersebut. Sebuah kebutuhan non fungsional juga dapat menghasilkan beberapa kebutuhan fungsional yang terkait dalam memenuhi kebutuhan non fungsional tersebut. Sehingga, kedua jenis kebutuhan ini saling berhubungan dan sama-sama penting untuk bisa dipenuhi dalam sebuah sistem dengan baik (Sommerville, 2011).

2.5.5.1 Usability

Usability merupakan suatu atribut kualitas untuk menilai sebuah seberapa mudah sebuah tampilan sistem digunakan oleh pengguna nya. Terdapat 5 komponen dari *usability* sendiri yaitu (1) *learnability* yang memiliki arti seberapa mudah dalam pengguna mempelajari dan menyelesaikan tugas tertentu pada program saat pertama kali menggunakannya. (2) Efisiensi memiliki arti seberapa cepat pengguna dapat menyelesaikan suatu tugas setelah sebelumnya pernah mempelajari tampilan program tersebut. (3) *Memorability* memiliki arti seberapa mudah pengguna menggunakan program tersebut dan menyelesaikan tugas tertentu setelah sekian waktu tidak menggunakan program tersebut. (4) *Error*: Seberapa banyak *error* atau kesalahan yang dialami oleh pengguna dan bagaimana sistem dapat membantu pengguna untuk memperbaiki kesalahan tersebut. (5) *User Satisfaction* memiliki arti seberapa nyaman dan puas pengguna terhadap program tersebut. (Nielsen, 2012).

2.6 Fase Perancangan Sistem

Perancangan sistem merupakan suatu proses analisis dan pembuatan model-model abstrak dari sebuah sistem, dimana dari model tersebut dapat merepresentasikan perspektif tertentu dari sistem yang akan dibangun. Perancangan model dari sebuah sistem biasanya menggunakan notasi grafis yang didasarkan pada *Unified Modelling Language* (UML) (Sommerville, 2011). Proses perancangan sendiri terdiri dari dua jenis pendekatan yaitu perancangan dengan pendekatan terstruktur, atau berbasis objek. Perancangan terstruktur merupakan perancangan yang berorientasi pada pemodelan aliran data, *business rule*, hingga *software tools*. Sedangkan perancangan berbasis objek, merupakan metode yang menekankan pada pemodelan sistem ke dalam kumpulan objek yang mempunyai atribut dan fungsi yang saling berhubungan (Mulyani, 2016).









Dalam penelitian ini, fase perancangan sistem akan dilakukan dengan membuat diagram-diagram UML dengan pendekatan berorientasi objek atau *object oriented design* seperti, *class diagram*, dan *sequence diagram* untuk memvisualisasikan dan merancang cara kerja sistem, *physical data diagram* untuk rancangan basis data yang akan digunakan, *pseudocode* untuk memberikan gambaran cara kerja suatu komponen dari sistem, serta *wireframe* untuk memberikan rancangan antar muka sistem.

2.6.1 Sequence Diagram

Sequence diagram merupakan diagram UML yang digunakan untuk memodelkan interaksi antara aktor dan objek-objek yang berada di dalam sistem, maupun interaksi antara objek dengan objek-objek lainnya. Diagram ini digunakan untuk memodelkan proses interaksi atau skenario yang terjadi di dalam sebuah use case. Yang perlu diketahui dari membuat suatu *sequence diagram* adalah objek-objek yang terlibat, metode-metode yang dimiliki oleh kelas yang diinstansiasi menjadi objek tersebut, serta skenario yang akan terjadi pada use case tersebut (Sommerville, 2011). Pada Tabel 2.3, dijabarkan beberapa notasi yang digunakan pada *sequence diagram* seperti adanya notasi aktor, *boundary*, *control*, *entity*, *lifeline*, *activation box*, *call*, *asynchronous message*, *reply*, dan *combined fragment*.

Tabel 2.3 Notasi *sequence diagram*

Notasi	Keterangan
	Aktor merupakan orang, proses ataupun sistem lain yang bisa berinteraksi dengan sistem, diluar dari sistem itu sendiri.
	Boundary merupakan tipe objek yang merepresentasikan objek yang dapat mengatur interaksi antara sistem dengan aktor.
	Control merupakan tipe objek berfungsi untuk mengatur interaksi antara objek <i>boundary</i> dengan objek <i>entity</i> .
	Entity merupakan objek yang berfungsi untuk merepresentasikan objek yang tidak berubah-ubah, dan biasanya digunakan dalam basis data.
	Lifeline merupakan sebuah notasi yang terdiri dari sebuah kotak dan garis vertikal putus-putus yang merepresentasikan waktu hidup dari objek atau partisipan tersebut.
	Activation Box merupakan notasi yang menyatakan bahwa objek tersebut sedang aktif dan berinteraksi dengan elemen lain

Sumber: Object Management Group (2017)

Tabel 2.3 Notasi *sequence diagram* (lanjutan)

Notasi	Keterangan
	<p>Call merupakan notasi yang menyatakan bahwa suatu objek memanggil suatu fungsi yang ada pada objek lain atau pada dirinya sendiri, dan arah panah mengarah pada objek yang memiliki metode tersebut.</p>
	<p>Asynchronous message merupakan notasi yang menyatakan bahwa suatu objek sedang mengirim data/masukan pada objek lain, dimana arah panah mengarah pada objek yang menerima masukan tersebut.</p>
	<p>Reply merupakan notasi yang menyatakan bahwa suatu objek yang telah menjalankan suatu operasi atau metode menghasilkan suatu kembalian ke objek tertentu, dan arah panah mengarah pada objek yang akan menerima kembalian tersebut.</p>
	<p>Combined Fragment merupakan sebuah persegi panjang yang dapat digunakan untuk merepresentasikan sebuah kondisi alternatif yang dapat terjadi dalam suatu <i>sequence diagram</i>.</p>

Sumber: Object Management Group (2017)

2.6.2 Class Diagram

Class diagram merupakan sebuah diagram yang digunakan dalam mengembangkan sebuah sistem berbasis objek, yang bertujuan untuk menggambarkan kelas-kelas yang berperan dalam suatu sistem dan hubungan antar objek kelas tersebut. Notasi utama dalam sebuah *class diagram* merupakan notasi kelas, dimana terdapat nama kelas tersebut di bagian atas, pada bagian tengah terdapat atribut yang menggambarkan karakteristik dari kelas tersebut, serta operasi atau fungsi-fungsi yang bisa dilakukan kelas tersebut di bagian bawah. Dalam membuat hubungan antar kelas dalam *class diagram*, terdapat beberapa jenis hubungan yang bisa terjadi, diantaranya terdapat hubungan asosiasi, hubungan generalisasi, hubungan agregasi, serta hubungan komposisi. Pada Tabel 2.4, dijabarkan mengenai beberapa notasi yang digunakan pada *class diagram*, seperti notasi kelas, hubungan asosiasi, generalisasi, agregasi, maupun komposisi (Lunn, 2003).



Tabel 2.4 Notasi class diagram

Notasi	Keterangan
	<p>Kelas merupakan suatu entitas yang merepresentasikan objek yang akan berperan dalam sistem tersebut, dan biasanya memiliki atribut yang menggambarkan karakteristik dari kelas tersebut, serta fungsi-fungsi yang dapat dilakukan oleh kelas tersebut.</p>
	<p>Asosiasi merupakan notasi yang menggambarkan adanya relasi antar kelas dengan makna umum.</p>
	<p>Generalisasi merupakan notasi yang digunakan untuk menghubungkan kelas yang memiliki makna umum (<i>superclass</i>) dengan kelas yang lebih spesifik (<i>subclass</i>). Generalisasi juga dapat disebut dengan hubungan "is a", seperti yang ada pada ilustrasi bahwa objek <i>sports car</i> dan <i>saloon car</i> keduanya merupakan sebuah <i>car</i>. Hubungan generalisasi dinotasikan dengan garis dengan ujung berbentuk panah putih yang mengarah ke <i>superclass</i>.</p>
	<p>Agregasi merupakan notasi yang menggambarkan hubungan antar kelas dimana satu kelas (<i>whole</i>) bisa terdiri dari beberapa kelas lainnya (<i>part</i>). Seperti yang ada pada ilustrasi hubungan objek <i>car</i> dengan <i>engine</i>, dimana <i>engine</i> merupakan bagian dari <i>car</i>, namun <i>car</i> bukan merupakan bagian dari <i>engine</i>. Hubungan agregasi dinotasikan dengan garis dengan ujung berbentuk wajik putih yang mengarah ke kelas <i>whole</i>.</p>
	<p>Komposisi merupakan notasi yang menggambarkan hubungan antar kelas dimana satu kelas merupakan bagian dari kelas lain dan kelas tersebut tidak bisa berdiri sendiri. Seperti yang ada pada ilustrasi hubungan antara objek <i>car</i> dengan <i>body</i>, dimana <i>body</i> merupakan bagian dari <i>car</i>, dan tidak bisa berdiri sendiri, sehingga saat objek <i>car</i> hilang, maka objek <i>body</i> pun akan hilang. Hubungan komposisi dinotasikan dengan garis dengan ujung berbentuk wajik berwarna yang mengarah ke kelas utama.</p>

Sumber: Lunn (2003)



2.6.3 Physical Data Model

Physical data model atau model relasional merupakan salah satu cara dalam perancangan data yang dapat merepresentasikan bagaimana data disusun dan dihubungkan dalam sebuah DBMS, sehingga dalam menentukan hubungan satu tabel dengan tabel lain, maupun tipe-tipe data pada kolom yang dimiliki oleh setiap tabel. Dengan menggunakan PDM, gambaran dari penyimpanan basis data suatu sistem dapat dijabarkan dengan detail dan jelas (Visual Paradigm, 2018). Pada Tabel 2.5, dijabarkan mengenai notasi yang terdapat pada *physical data model* seperti adanya notasi tabel, dan relasi.

Tabel 2.5 Notasi *physical data model*

Notasi	Keterangan
	<p>Tabel merupakan notasi yang merepresentasikan tabel pada basis data yang memiliki kolom-kolom yang disebut dengan atribut untuk menyimpan data yang dimiliki tabel tersebut. Atribut-atribut tersebut juga bisa berupa sebuah <i>primary key</i> maupun <i>foreign key</i>.</p>
	<p>Relasi merupakan notasi yang menghubungkan satu tabel dengan tabel lain berdasarkan persamaan antara <i>primary key</i> di satu tabel dengan <i>foreign key</i> di tabel yang dituju.</p> <p>One to One Relationship merupakan salah satu jenis kardinalitas yang menyatakan bahwa satu entitas memiliki hubungan satu dan hanya satu dengan entitas lain.</p>
	<p>One to Many Relationship merupakan jenis kardinalitas yang menyatakan bahwa satu entitas dapat memiliki hubungan dengan banyak entitas lain.</p>
	<p>Many to Many Relationship merupakan jenis kardinalitas yang menyatakan bahwa satu entitas dapat berhubungan dengan banyak entitas lain, dan sebaliknya.</p>

Sumber: Visual Paradigm (2018)



2.6.4 Pseudocode

Pseudocode merupakan serangkaian perintah yang dituliskan dalam bahasa yang natural atau umum yang bertujuan untuk menjelaskan suatu proses tanpa menggunakan sintaks atau semantik dari bahasa pemrograman tertentu. Pada umumnya, sintaks yang digunakan oleh pengembang dalam membuat *pseudocode* bisa berbeda satu pengembang dengan yang lainnya. Dikarenakan dalam membuat *pseudocode*, tidak ada aturan yang baku, dan bergantung pada pengetahuan dan bahasa pemrograman yang paling dipahami oleh pengembang tersebut (Blackedget, 2006). Pada Tabel 2.6, merupakan contoh *pseudocode* untuk menentukan angka yang dimasukkan merupakan angka positif atau negatif.

Tabel 2.6 Pseudocode menentukan angka positif atau negatif

1	output("Input positive or negative numbers")
2	input (number)
3	Begin:
4	IF number > 0
5	THEN
6	output("Number is positive")
7	ENDIF
8	IF number < 0
9	THEN
10	output("Number is negative")
11	ENDIF
12	END

Sumber: Blackedget (2006)

2.6.5 Wireframe

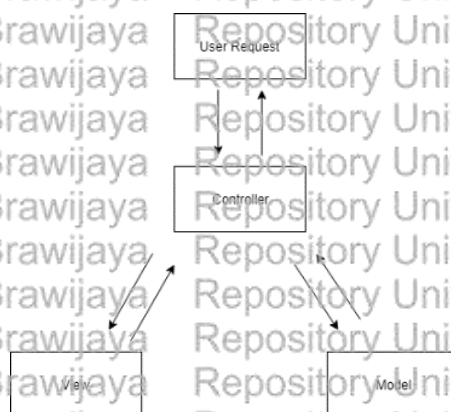
Wireframe merupakan suatu rancangan tampilan atau *prototype* dari sebuah sistem secara detail, yang dapat dibuat menggunakan perangkat lunak pengolah citra digital atau perangkat lunak khusus untuk desain aplikasi. Wireframe ini berfungsi untuk menampilkan ilustrasi menu atau konten apa saja yang akan ditampilkan, agar pihak-pihak yang terlibat dari sistem dapat mendapatkan gambaran secara jelas mengenai sistem yang nantinya akan dibangun (Tolle, et al, 2017).

2.7 Fase Implementasi Sistem

Implementasi sistem merupakan fase selanjutnya setelah menyelesaikan proses perancangan sistem dengan cara mengubah hasil rancangan tersebut menjadi serangkaian unit program yang saling berhubungan untuk nantinya dapat menjadi sebuah sistem yang jadi seutuhnya (Sommerville, 2011). Pada fase ini akan dilakukan proses implementasi berdasarkan hasil analisis dan perancangan yang telah dilakukan pada fase sebelumnya ke dalam bentuk sistem informasi berbasis web dengan menggunakan pendekatan berbasis objek serta teknologi lainnya yang mendukung



2.7.1 Model View Controller (MVC)



Gambar 2.2 Ilustrasi konsep Model – View - Controller

Sumber: Basuki (2011)

Model-View-Controller merupakan konsep dalam pengembangan sistem yang memisahkan antara *layer application-logic* dengan *layer presentation*, sehingga seorang *programmer* bisa fokus pada pengembangan *core-system* nya, sedangkan *designer* dapat fokus pada tampilan web nya dan bisa mengerjakan secara paralel. Dalam Gambar 2.2, terdapat ilustrasi dari cara kerja konsep *Model-View-Controller* ini, dimana saat adanya *user request* pada sistem, maka permintaan tersebut akan ditangani oleh *controller* untuk nantinya akan memanggil *view* ataupun melakukan query pada *model*, dimana nantinya hasil akan dikembalikan kepada *controller* untuk ditampilkan ke *browser* untuk bisa dilihat oleh pengguna. *Model* sendiri bertugas untuk mengelola semua fungsi yang terkait dengan manipulasi basis data sistem, *view* berfungsi untuk menampilkan halaman web ke pengguna, sedangkan *controller* bertugas untuk mengontrol aliran aplikasi (Basuki, 2011).

2.7.2 Codeigniter (CI)

Codeigniter merupakan salah satu *framework* PHP yang menggunakan konsep *Model-View-Controller* (MVC) yang dibuat pertama kali oleh Ellislab, Inc dan dirilis pada tahun 2006. Codeigniter dapat membantu dan mempercepat pengembang dalam membuat suatu aplikasi berbasis web jika dibandingkan harus menulis semua kode program dari awal, karena dengan menggunakan *framework* codeigniter ini, sudah terdapat banyak fungsi-fungsi yang umumnya ada pada suatu aplikasi berbasis web yang hanya perlu dipanggil, sehingga seorang pengembang bisa fokus pada pembuatan fitur daripada sistem itu sendiri (Basuki, 2011).



2.7.3 Bootstrap

Framework Bootstrap merupakan salah satu *framework* CSS yang banyak digunakan dalam pengembangan *website*. Bootstrap sendiri dikembangkan oleh *Bootstrap Core Team* yang dibuat pertama kali oleh Mark Otto, dan Jacob Thronton pada tahun 2011. Bootstrap dapat membantu proses desain tampilan suatu *website* menjadi lebih mudah dan cepat karena Bootstrap sudah menyediakan fitur-fitur *built-in* yang hanya perlu dipanggil oleh pengembang untuk mengatur tampilan sesuai kebutuhan. Bootstrap juga dapat membuat *website* yang dikembangkan menjadi responsif sehingga dapat membuat tampilan menyesuaikan diri di semua perangkat, baik pada *desktop* maupun *mobile* (Abdulloh, 2018).

2.8 Fase Pengujian Sistem

Pengujian suatu sistem merupakan tahap yang dilakukan dalam proses pengembangan sistem yang memiliki dua tujuan utama yaitu untuk menunjukkan apakah sistem sudah bisa melakukan apa yang telah direncanakan sebelumnya baik kepada pihak pengguna maupun kepada pengembang lain, dan tujuan lainnya adalah untuk memastikan bahwa sistem sudah bebas dari kesalahan sebelum pada akhirnya digunakan oleh pengguna. Proses pengujian sistem ini juga termasuk dalam proses verifikasi dan validasi, dimana verifikasi artinya adalah memastikan bahwa proses pembuatan sistem/produk tersebut sudah tepat (*Are we building the product right?*), sedangkan validasi adalah memastikan bahwa produk kita sudah tepat atau sesuai dengan kebutuhan atau keinginan pengguna (*Are we building the right product?*) (Sommerville, 2011). Pada fase ini akan dilakukan proses pengujian terhadap sistem yang telah dibangun dengan menggunakan dua pengujian, yaitu pengujian validasi yang dilakukan secara *black box* untuk memvalidasi bahwa semua kebutuhan fungsional telah terpenuhi pada sistem yang dibangun. Serta menggunakan *user acceptance testing* (UAT) dengan menggunakan skala likert untuk menguji tingkat penerimaan pengguna terhadap sistem yang telah dibangun.

2.8.1 Pengujian *Black Box*

Pengujian *black box* atau *behavioural testing* merupakan pengujian yang berfokus pada memvalidasi kebutuhan fungsional dari suatu perangkat lunak. Pengujian *black box* ini dilakukan dengan membuat serangkaian kondisi masukan yang bisa diberikan kepada sistem untuk bisa memastikan bahwa semua kebutuhan fungsional dari sistem sudah terpenuhi. Pengujian *black box* dapat dilakukan pada akhir fase pengembangan sistem, karena lebih berfokus pada memastikan bahwa sistem sudah sesuai dengan kebutuhan yang telah ditentukan, bukan untuk menguji struktur kontrol daripada sistem yang dibangun. Salah satu jenis pengujian *black box* adalah pengujian validasi atau *validation testing* yaitu pengujian untuk memvalidasi suatu perangkat lunak sudah sesuai dengan kebutuhan yang telah terdefiniskan (Pressman, 2010).



2.8.2 User Acceptance Testing

Pengujian penerimaan atau *acceptance testing* merupakan pengujian sistem yang dilakukan setelah sistem telah melewati dan memenuhi kriteria yang telah ditentukan pada fase pengujian teknis sistem. *User acceptance testing* ini dilakukan kepada pengguna untuk memastikan bahwa sistem telah memenuhi kriteria penerimaan yang telah ditentukan sebelumnya. Pengujian ini juga berfungsi untuk membantu pengguna untuk menentukan apakah akan menerima atau menolak sistem yang telah dibuat tersebut. Pada penelitian ini akan dilakukan pengujian dengan menggunakan UAT berdasarkan dua kriteria penerimaan pengguna yaitu kriteria *functional correctness and completeness* serta *usability*, dimana kriteria *functional correctness and completeness* sendiri merupakan kriteria yang bertujuan untuk memastikan bahwa semua fungsi yang telah dijabarkan pada spesifikasi kebutuhan telah terimplementasi dengan benar pada sistem tersebut, sehingga bisa diajukan pertanyaan “*Does the system do what we want it to do?*”. Untuk kriteria *usability* sendiri merupakan kriteria yang bertujuan untuk memastikan bahwa sistem tersebut mudah untuk dipelajari dan digunakan oleh *user*, sehingga bisa diajukan pertanyaan “*How easy it is to use the system?*” dan “*How easy it is to learn?*”. Pertanyaan-pertanyaan tersebut kemudian dapat diberikan kepada target pengguna, untuk menentukan sistem yang telah dibuat sudah sesuai dengan kebutuhan dan dapat diterima oleh pengguna (Naik & Tripathy, 2008).

2.8.2.1 Purposive Sampling

Purposive sampling atau *sampling* pertimbangan merupakan proses pengambilan sampel secara tidak acak (*non-random sampling*) yang pada pengambilan sampel nya ditentukan berdasarkan pertimbangan daripada peneliti tersebut. *Purposive sampling* ini dapat digunakan untuk penelitian yang memang memerlukan kriteria khusus dari sampel yang ingin diteliti, sehingga menghasilkan sampel yang lebih representatif terkait tujuan daripada penelitian yang sedang dilakukan (Hasan, 2012). Penggunaan *purposive sampling* pada penelitian ini ada pada proses identifikasi permasalahan dan analisis kebutuhan, serta proses pengujian sistem khususnya pada *user acceptance testing*. Dimana pada kedua proses tersebut, dilakukan hanya kepada para responden yang ada pada lingkup penelitian, serta yang nantinya akan berperan dalam menggunakan sistem ini, yaitu kepada Bapak Intan Yusuf Habibie S.Gz., M.Sc., selaku dosen gizi Fakultas Kedokteran UB, yang akan berperan sebagai aktor *admin*. Ibu Komariah Tusokilah A.Md., selaku bidan yang bertugas untuk mengawasi Posyandu Posyandu yang ada di Desa Kreet, Kabupaten Malang. Serta petugas atau kader Posyandu pada salah satu Posyandu yang dijadikan acuan pada penelitian ini yaitu Posyandu Balita Srikaya, Desa Kreet, Kabupaten Malang. *Purposive sampling* ini dilakukan pada penelitian ini, khususnya pada pengambilan responden kader Posyandu yang hanya berasal dari satu Posyandu saja, adalah dikarenakan proses bisnis pelayanan Posyandu yang berada di Desa Kreet memiliki kesamaan satu dengan yang lainnya, sehingga permasalahan yang ada pada satu Posyandu, bisa juga serupa terjadi di Posyandu lainnya.



2.8.2.2 Skala Likert

Skala likert merupakan skala psikometrik yang banyak digunakan dalam proses penelitian, khususnya yang menggunakan survei atau angket yang diberikan kepada responden, dimana skala likert disini berfungsi untuk mengukur persepsi, sikap maupun pendapat seseorang atau kelompok mengenai suatu kasus atau kejadian yang sedang diteliti. Terdapat dua bentuk pertanyaan yang dapat digunakan dalam skala likert, yaitu pertanyaan positif dan negatif, dimana untuk pertanyaan positif digunakan untuk mengukur skala positif yang memiliki skor 5,4,3,2,1, sedangkan pertanyaan negatif digunakan untuk mengukur skala negatif yang memiliki skor 1,2,3,4,5 atau -2,-1,0,1,2. Proses perhitungan dengan menggunakan skala likert ini diawali dengan memberikan bobot nilai pada setiap respon yang diberikan, dimana pada Tabel 2.7 dijabarkan mengenai pembobotan nilai berdasarkan hasil respon dari setiap pertanyaan yang diberikan. (Choizes, 2019).

Tabel 2.7 Bobot nilai jawaban

Jawaban	Bobot Nilai
Sangat Setuju	5
Setuju	4
Netral	3
Tidak Setuju	2
Sangat Tidak Setuju	1

Sumber: Choizes (2019)

Setelah bobot nilai telah ditentukan, maka berikutnya untuk menghitung nilai kuesioner secara keseluruhan, adalah dengan menghitung nilai total dari jawaban pada pertanyaan, dikalikan dengan bobot nilai berdasarkan pilihan jawaban seperti pada Tabel 2.7, lalu dibagi dengan nilai maksimum yang memungkinkan untuk didapatkan dari kuesioner tersebut. Perhitungan nilai kuesioner dijabarkan pada Persamaan 2.1.

$$\text{Nilai Kuesioner} = \frac{\text{total skor}}{\text{nilai maksimum}} \times 100\% \quad (2.1)$$

Setelah didapatkan nilai kuesioner secara keseluruhan, maka nilai tersebut dapat diinterpretasikan artinya berdasarkan rentang nilai yang dijabarkan pada Tabel 2.8 berikut.

Tabel 2.8 Rentang presentase nilai kuesioner

Rentang Presentase Nilai	Hasil
80% - 100%	Sangat Setuju
60% - 79,99%	Setuju
40% - 59,99%	Netral
20% - 39,99%	Tidak Setuju
0% - 19,99%	Sangat Tidak Setuju

Sumber: Choizes (2019)



2.9 Pos Pelayanan Terpadu (Posyandu)

Pos Pelayanan Terpadu atau sering disingkat dengan kata Posyandu merupakan bentuk Upaya Kesehatan Bersumberdaya Masyarakat (UKBM) yang dilaksanakan dari oleh dan untuk masyarakat, dimana Posyandu memiliki tujuan untuk bisa memberikan kemudahan serta memberdayakan masyarakat dalam hal mendapatkan pelayanan kesehatan, khususnya bagi ibu, bayi serta anak balita. Pelayanan utama di Posyandu mencakup lima kegiatan yaitu pelayanan kesehatan ibu dan anak, keluarga berencana, pemberian imunisasi, pelayanan gizi, serta pencegahan dan penanggulangan penyakit diare (Kementerian Kesehatan RI Pusat Promosi Kesehatan, 2012).

Pelayanan gizi dalam Posyandu akan dilakukan oleh para kader Posyandu yang akan melakukan kegiatan seperti penimbangan berat, konseling gizi, pemberian makanan tambahan (PMT), pemberian vitamin A, maupun mendeteksi dini gangguan pertumbuhan atau malnutrisi agar bisa dirujuk ke Puskesmas. Setelah Posyandu melakukan kegiatan pelayanan, maka wajib untuk melakukan proses pencatatan hasil kegiatan tersebut sesuai dengan format yang sudah diberikan dari pihak Puskesmas. Hasil pencatatan tersebut nantinya akan dilaporkan kepada pihak Puskesmas melalui seorang bidan yang bertanggung jawab pada beberapa Posyandu di suatu desa untuk membuat laporan hasil pencatatan tersebut kepada pihak Puskesmas (Kementerian Kesehatan RI, 2011). Ilustrasi struktur organisasi secara umum untuk proses pencatatan dan pelaporan hasil kegiatan Posyandu dapat dilihat pada Gambar 2.3.



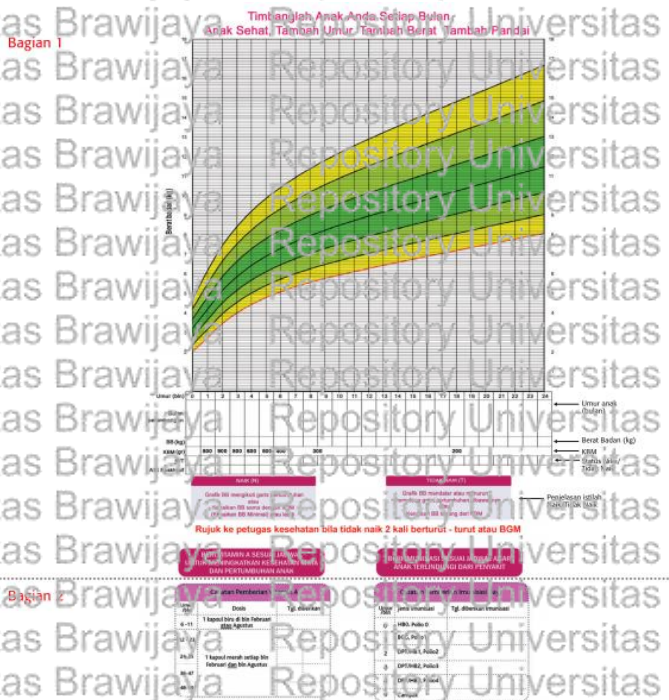
Gambar 2.3 Ilustrasi Struktur Organisasi Posyandu

2.10 Kartu Menuju Sehat (KMS)

Kartu Menuju Sehat (KMS) merupakan kartu yang terdiri dari 5 bagian utama yang berisi kurva pertumbuhan normal anak berdasarkan indeks antropometri berat badan menurut umur (BB/U). Dimana dengan adanya pencatatan melalui KMS ini diharapkan gangguan pertumbuhan atau resiko kelebihan gizi dapat terdeteksi sejak dini sehingga dapat dilakukan tindakan yang cepat dan tepat. Fungsi utama KMS sendiri yaitu sebagai alat untuk memantau pertumbuhan anak melalui pencatatan yang mengacu pada grafik pertumbuhan normal anak, lalu sebagai catatan riwayat pelayanan kesehatan dasar anak seperti pemberian vitamin, ASI, maupun imunisasi, serta yang terakhir adalah alat untuk

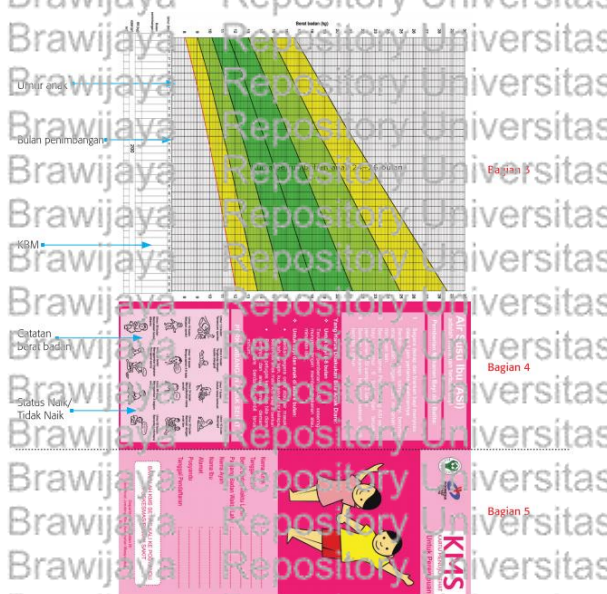


mengedukasi dari pihak Posyandu kepada pemegang KMS seperti cara perawatan dan pemberian makan anak yang baik. KMS sendiri dibedakan untuk anak laki-laki dan perempuan, dimana untuk anak laki-laki berwarna dasar biru, sedangkan untuk anak perempuan berwarna dasar merah muda (Kementerian Kesehatan Republik Indonesia, 2010). Dalam Gambar 2.4 dan Gambar 2.5 merupakan contoh isi dari KMS untuk anak perempuan yang terdiri dari lima bagian.



Gambar 2.4 Bagian 1 dan 2 dari KMS

Sumber: Kementerian Kesehatan Republik Indonesia (2010)



Gambar 2.5 Bagian 3, 4 dan 5 dari KMS

Sumber: Kementerian Kesehatan Republik Indonesia (2010)



Dalam contoh KMS yang ditunjukkan dalam Gambar 2.4 dan Gambar 2.5, terdapat beberapa bagian utama KMS yang dapat digunakan oleh pihak Posyandu untuk mencatat riwayat pertumbuhan anak balita tersebut. Pada bagian atas dalam Gambar 2.4 bagian 1, terdapat kurva yang dapat digunakan oleh pihak Posyandu untuk mencatat pertumbuhan berat badan balita, dimana hasil pencatatan tersebut dapat dibandingkan dengan kurva yang sudah terdapat pada KMS tersebut yang merupakan standar pertumbuhan berat badan balita menurut standar antropometri WHO 2005. Pada bagian bawah dalam Gambar 2.4 bagian 1, terdapat tabel yang berisikan umur balita, bulan penimbangan, berat badan, kenaikan berat badan minimal (KBM), indikator naik / turun, serta kolom pemberian ASI eksklusif. Pada indikator naik atau turun tersebut, pihak Posyandu akan memberikan keterangan naik apabila kenaikan berat badan balita sama dengan atau lebih dari KBM yang telah ditentukan pada umur balita tersebut, sedangkan sebaliknya akan diberikan keterangan turun apabila berat badan balita pada bulan berikutnya kurang dari KBM (Kementerian Kesehatan Republik Indonesia, 2010).

2.11 Status Gizi

Gizi merupakan suatu proses dari sebuah makhluk hidup untuk menggunakan makanan yang dikonsumsinya untuk bertahan hidup, bertumbuh maupun menjalankan fungsi tubuh lainnya secara normal (Pekik, 2006). Baik buruknya kondisi gizi seseorang dapat ditentukan melalui status gizi orang tersebut, dimana status gizi merupakan status kesehatan tubuh seseorang yang dapat dilihat dari keseimbangan antara kebutuhan dan asupan gizi tubuhnya. Status gizi seseorang dapat digolongkan menjadi dua kategori, yaitu saat kebutuhan gizi tubuh dengan asupan gizi tubuh ada pada proporsi yang baik dan penggunaan gizi tubuh juga normal, maka tubuh memiliki status gizi normal, sedangkan apabila saat kebutuhan gizi tubuh dengan asupan gizi yang didapatkan tidak seimbang, serta penggunaan gizi pada tubuh ada ketidaknormalan, maka kondisi tersebut bisa disebut sebagai malnutrisi. Keadaan malnutrisi sendiri terdapat dua jenis kondisi yaitu kurang gizi (*undernutrition*) dan gizi berlebih (*overnutrition*), dimana kurang gizi adalah kondisi saat tubuh kekurangan satu atau lebih nutrisi yang dibutuhkan, sedangkan jika tubuh mendapatkan asupan nutrisi yang terlalu berlebih daripada yang dibutuhkan, maka kondisi tersebut disebut dengan gizi berlebih (National Institute of Open Schooling, 2014).

Dalam menentukan status gizi seorang anak balita, diperlukan beberapa data dari balita tersebut diantaranya jenis kelamin, umur balita, berat badan, tinggi badan, serta cara pengukuran. Dalam pengukuran tinggi seorang anak balita dalam menentukan status gizinya, apabila balita tersebut masih berumur 0 – 24 bulan dan diukur secara telentang, maka akan menggunakan panjang badannya sebagai pengukuran tinggi, sedangkan apabila berumur 0 – 24 bulan namun diukur secara berdiri, maka hasil pengukurannya akan dikoreksi dengan menambahkan 0,7 cm. Lalu apabila balita tersebut sudah berumur > 24 bulan, dan diukur secara berdiri, maka akan menggunakan tinggi badannya sebagai pengukuran tinggi. Namun apabila sudah berumur > 24 bulan, namun pengukuran dilakukan secara



telentang, maka hasil pengukuran akan dikoreksi dengan dikurangi 0.7 cm (Kemenkes RI, 2011). Dari data-data tersebut, maka dapat ditentukan status gizi anak balita tersebut melalui tiga indikator yaitu melalui berat badan menurut umur (BB/U), panjang badan menurut umur (PB/U) atau tinggi badan menurut umur (TB/U), serta berat badan menurut panjang badan (BB/PB) atau berat badan menurut tinggi badan (BB/TB). Untuk masing-masing indikator status gizi tersebut, diperlukan untuk menghitung angka Z-Score anak tersebut terlebih dahulu yaitu dengan menggunakan rumus pada Persamaan 2.2 (Direktorat Gizi Masyarakat, 2017).

$$Z - \text{Score} = \frac{\text{nilai individu subjek} - \text{nilai median baku rujukan}}{\text{nilai simpang baku rujukan}} \quad (2.2)$$

“Nilai individu subjek” disini merupakan nilai hasil pengukuran baik berat badan atau panjang/tinggi badan. “Nilai median baku rujukan” disini merupakan nilai median yang berada pada standar antropometri yang contohnya dapat dilihat dalam Lampiran B. Sedangkan “nilai simpang baku rujukan” merupakan hasil pengurangan nilai median dengan -1 SD apabila berat badan atau panjang/tinggi badan balita kurang dari median, namun apabila berat badan atau panjang/tinggi badan balita lebih dari median, maka “nilai simpang baku rujukan” dihasilkan dari pengurangan nilai $+1$ SD dengan nilai median nya (Kemenkes RI, 2011). Setelah dilakukan perhitungan Z-Score, maka dapat ditentukan status gizi dari anak balita tersebut yang merujuk pada standar penentuan status gizi yang dikeluarkan oleh Kementerian Kesehatan Republik Indonesia. Pada Tabel 2.9 berikut dijabarkan mengenai batasan-batasan untuk mengetahui kategori status gizi menurut ketiga indikator tersebut.

Tabel 2.9 Kategori status gizi balita

Indikator	Status Gizi	Z-Score
Berat Badan Menurut Umur (BB/U)	Gizi Buruk	$< -3,0$ SD
	Gizi Kurang	$-3,0$ SD s/d $-2,0$ SD
	Gizi Baik	$-2,0$ SD s/d $2,0$ SD
	Gizi Lebih	$\geq 2,0$ SD
Panjang Badan Menurut Umur (PB/U) atau Tinggi Badan Menurut Umur (TB/U)	Sangat Pendek	$< -3,0$ SD
	Pendek	$-3,0$ SD s/d $-2,0$ SD
	Normal	$\geq -2,0$ SD
Berat Badan Menurut Panjang Badan (BB/PB) atau Berat Badan Menurut Tinggi Badan (BB/TB)	Sangat Kurus	$< -3,0$ SD
	Kurus	$-3,0$ SD s/d $-2,0$ SD
	Normal	$-2,0$ SD s/d $2,0$ SD
	Gemuk	$> 2,0$ SD

Sumber: Kemenkes RI (2011)

BAB 3 METODOLOGI

Pada bab ini dibahas mengenai metodologi yang akan digunakan dalam melakukan proses penelitian ini. Metodologi penelitian untuk pengembangan Sistem Informasi Pencatatan dan Pelaporan Gizi ini didasarkan dari metode *waterfall* yang bersifat linear-sekuensial dengan perencanaan yang matang di awal untuk setiap fase nya serta adanya umpan balik dari setiap fase, ke fase sebelum maupun sesudahnya (Sommerville, 2011). Metode ini dipilih karena lingkup penelitian yang tidak terlalu luas serta pendefinisian kebutuhan yang tidak terlalu kompleks sehingga memungkinkan untuk dapat dilakukan dengan lengkap di awal proses pengembangan sehingga tidak memerlukan adanya proses iterasi.

Tahapan metodologi penelitian yang digunakan pada penelitian ini terdiri dari identifikasi masalah, studi literatur, analisis kebutuhan, perancangan sistem, implementasi sistem, pengujian sistem, serta pengambilan kesimpulan dan saran. Adapun diagram alir metodologi tersebut dijelaskan dalam Gambar 3.1



Gambar 3.1 Diagram metodologi penelitian



3.1 Identifikasi Masalah

Dalam melakukan penelitian ini, diperlukan proses identifikasi permasalahan dan ruang lingkupnya yang akan diselesaikan melalui sistem yang akan dibangun. Proses identifikasi masalah ini dilakukan dengan melakukan wawancara dengan Bapak Intan Yusuf Habiebie S.Gz., M.Sc., selaku dosen gizi dari FK UB untuk mendapatkan gambaran permasalahan yang dihadapi di lapangan terlebih dahulu. Setelah itu, dilanjutkan dengan melakukan wawancara dengan Ibu Komariah Tusokilah A.Md., selaku bidan desa yang bertugas, dan dengan kegiatan observasi di salah satu Posyandu di Desa Kreet, Kabupaten Malang yaitu Posyandu Balita Srikaya, untuk mendapatkan gambaran secara langsung mengenai bagaimana alur proses pencatatan dan pelaporan hasil pelayanan kesehatan kepada balita dilakukan, serta permasalahan yang dihadapi di lapangan.

3.2 Studi Literatur

Pada tahap ini dilakukan studi pustaka dan literatur mengenai permasalahan yang diangkat yaitu terkait Posyandu, kartu menuju sehat, status gizi, serta dasar-dasar pengembangan suatu sistem informasi berbasis *web* dengan pendekatan berbasis objek yang meliputi aspek-aspek pada fase pendefinisian kebutuhan, perancangan sistem, implementasi, hingga pengujian sistem. Tujuan dilakukannya studi literatur ini adalah untuk memperkaya pengetahuan terhadap permasalahan, serta menjadi dasar acuan teori dalam melakukan penelitian ini. Literatur-literatur yang digunakan diperoleh dari buku, *paper*, penelitian terdahulu, jurnal nasional maupun internasional maupun dokumentasi internet.

3.3 Analisis Kebutuhan Sistem

Tahapan analisis kebutuhan sistem merupakan suatu proses untuk mendapatkan kebutuhan maupun batasan daripada sistem yang akan dibangun. Proses analisis kebutuhan melanjutkan dari kegiatan identifikasi permasalahan dengan cara melakukan wawancara dan observasi lanjutan kepada pihak bidan maupun kader dari Posyandu untuk mendapatkan permasalahan serta kebutuhan yang lebih detail dari masing-masing pihak. Dimana proses analisis kebutuhan ini hanya dilakukan kepada satu orang bidan yang bertugas untuk mengawasi Posyandu di Desa Kreet, Kabupaten Malang, serta kepada kader Posyandu di satu Posyandu di Desa Kreet, Kabupaten Malang. Hal ini dilakukan karena proses bisnis pelayanan Posyandu tersebut, hampir serupa juga di Posyandu lain, sehingga dapat dilakukan proses analisis pada satu Posyandu saja. Setelah mendapatkan permasalahan dan kebutuhan dari sistem yang akan dibangun, maka berikutnya dilakukan diskusi lanjutan dengan Bapak Intan Yusuf Habiebie S.Gz., M.Sc., guna memvalidasi kebutuhan fungsional dan non fungsional yang telah didefinisikan. Setelah mendapatkan informasi yang dibutuhkan maka akan dilakukan dijabarkan hasil analisis proses bisnis, identifikasi aktor yang akan terlibat dalam sistem, kebutuhan fungsional, spesifikasi kebutuhan fungsional, kebutuhan non fungsional dari sistem yang akan dikembangkan, serta penjelasan *use case* dalam bentuk *use case diagram* dan *use case scenario*.



3.4 Perancangan Sistem

Setelah melakukan proses analisis kebutuhan dan telah mengidentifikasi gambaran kebutuhan maupun aktor-aktor yang akan menggunakan sistem ini, maka tahap berikutnya adalah melakukan perancangan sistem dimana gambaran cara kerja dan tampilan sistem divisualisasikan melalui diagram-diagram perancangan sistem. Pada proses perancangan sistem ini, akan dilakukan perancangan arsitektur menggunakan *sequence diagram*, dan *class diagram*, perancangan data menggunakan *physical data diagram*, perancangan komponen menggunakan *pseudocode*, dan perancangan antarmuka menggunakan *wireframe*.

3.5 Implementasi Sistem

Setelah proses perancangan sistem telah dilakukan, maka berikutnya adalah mengimplementasikan hasil rancangan tersebut ke dalam bahasa pemrograman yang sesuai dalam membuat suatu sistem informasi berbasis *web* dengan pendekatan berbasis objek menggunakan *framework* codeigniter. Dalam mengimplementasikan antarmuka dari sistem informasi ini, akan digunakan HTML, CSS, dan Javascript, dengan menggunakan *framework* Bootstrap. Untuk mengimplementasikan logika sistem, akan menggunakan bahasa PHP5, dan terakhir untuk mengimplementasikan struktur basis data dan melakukan manipulasi data didalamnya akan menggunakan MySQL.

3.6 Pengujian Sistem

Jika tahap implementasi kebutuhan baik fungsional maupun non fungsional sistem telah selesai, maka tahap berikutnya adalah tahap pengujian sistem menggunakan metode pengujian *black box* untuk menguji validitas fungsi sistem sudah sesuai dengan kebutuhan dan rancangan yang telah dibuat sebelumnya, serta melakukan *user acceptance testing* kepada para pengguna dengan cara meminta pengguna untuk menggunakan sistem tersebut sebelum diberikan pertanyaan-pertanyaan terkait kriteria penerimaan dari sistem yaitu *functional correctness and completeness* serta *usability* menggunakan skala likert, untuk menilai tingkat penerimaan sistem yang telah dibuat. Pelaksanaan UAT ini akan dilakukan kepada 5 orang pengguna yang akan mewakili masing-masing jenis aktor yang akan berperan dalam sistem ini.

3.7 Kesimpulan dan Saran

Penarikan kesimpulan dan saran untuk penelitian selanjutnya dilakukan setelah semua tahapan dari metodologi penelitian telah diselesaikan. Proses penulisan kesimpulan maupun saran dilakukan untuk menyimpulkan hasil dari penelitian ini serta memberikan masukan maupun perbaikan terhadap penelitian ini untuk kedepannya.



BAB 4 ANALISIS KEBUTUHAN

Pada bab 4 ini akan dilakukan proses analisis kebutuhan sistem yang akan dibangun berdasarkan permasalahan yang telah dipaparkan pada bab sebelumnya. Pada bab ini akan dijabarkan kebutuhan sistem dalam beberapa bagian, yang pertama adalah penggambaran umum sistem yang akan dibangun dalam penelitian ini, analisis proses bisnis saat ini (*as-is*), analisis permasalahan yang berada pada proses bisnis saat ini, serta rancangan proses bisnis yang direkomendasikan (*to-be*) setelah diimplementasikannya sistem informasi ini, identifikasi aktor, penjelasan mengenai kebutuhan fungsional, spesifikasi kebutuhan fungsional, dan kebutuhan non fungsional sistem, lalu akan dilakukan pemodelan kebutuhan *use case* menggunakan *use case* diagram, serta penjelasan *use case* sistem menggunakan *use case specification*.

4.1 Gambaran Umum Sistem

Sistem informasi pencatatan dan pelaporan gizi balita yang akan dikembangkan dalam penelitian ini bertujuan untuk memfasilitasi pihak Posyandu untuk melakukan pencatatan hasil pelayanan gizi balita yang dilakukan secara rutin setiap bulannya, dan juga yang berkaitan dengan data administratif orang tua dan balita yang terdaftar pada Posyandu tersebut. Dimana data administratif dan hasil pencatatan tersebut dapat digunakan oleh pihak bidan untuk memantau hasil pencatatan setiap Posyandunya serta membuat laporan gizi untuk dilaporkan kepada pihak puskesmas. Selain itu, data hasil pencatatan tersebut juga dapat dilihat oleh pihak orang tua balita melalui aplikasi perangkat bergerak berbasis Android yang dapat menampilkan data anak balita beserta hasil pengukuran dari Posyandu. Sehingga dengan adanya aplikasi perangkat bergerak yang melengkapi sistem informasi ini, diharapkan orang tua balita juga dapat ikut memantau pertumbuhan anak balita, serta mendapatkan informasi terbaru dari pihak Posyandu secara langsung pada perangkat bergerak nya. Dalam Gambar 4.1 berikut, terdapat gambaran umum sistem yang akan dikembangkan dalam penelitian ini.



Gambar 4.1 Gambaran Umum Sistem



4.2 Analisis Proses Bisnis

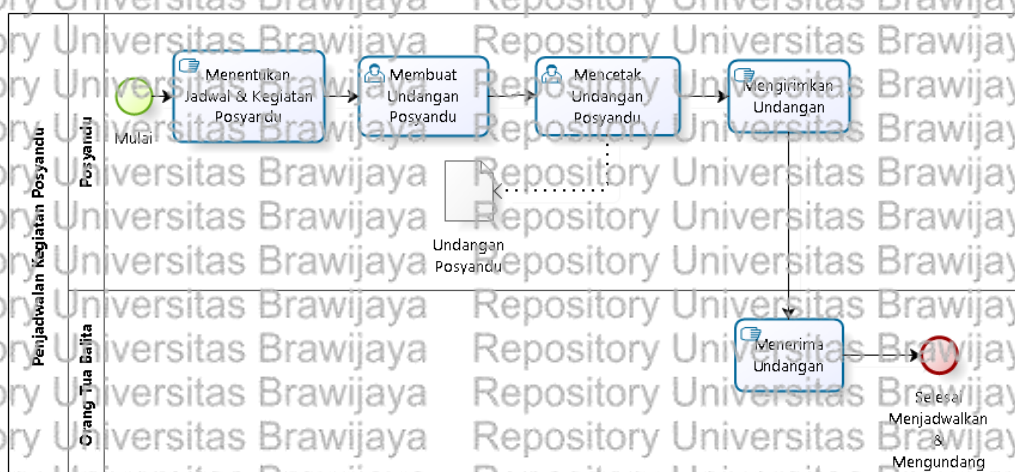
Pada bagian ini akan dilakukan proses analisis proses bisnis dengan menggunakan notasi-notasi pemodelan BPMN dari proses pelayanan pada Posyandu yang didapatkan dari diskusi dan observasi yang telah dilakukan pada pihak badan maupun kader Posyandu di Posyandu Balita Srikaya Desa Kreet, Kabupaten Malang (Lampiran C.1 dan Lampiran C.2), dimana pelayanan Posyandu dimulai dari penjadwalan kegiatan Posyandu, lalu pada hari Posyandu dilaksanakan akan dilakukan proses pelayanan kesehatan kepada balita yang hadir, dan akhirnya setelah proses pelayanan selesai, akan dibuatkan laporan gizi dan hasil pelayanan yang telah dilakukan. Analisis proses bisnis ini dilakukan untuk mendapatkan gambaran alur proses pelayanan Posyandu secara *as-is* atau kondisi saat ini, analisis permasalahan dari proses bisnis saat ini, maupun analisis rekomendasi proses bisnis *to-be* atau analisis proses setelah diimplementasikan sistem pada Posyandu.

4.2.1 Proses Bisnis Penjadwalan Kegiatan Posyandu Saat Ini (*as-is*)

Proses bisnis untuk menjadwalkan kegiatan Posyandu merupakan kegiatan yang dilakukan oleh pihak Posyandu setiap bulannya sebelum melakukan pelayanan kesehatan. Tujuan dari proses bisnis ini adalah menentukan tanggal, waktu, lokasi, serta kegiatan apa saja yang akan dilakukan pada Posyandu tersebut, hingga akhirnya mengundang para warga atau orang tua yang memiliki balita di wilayah pelayanan Posyandu tersebut untuk hadir pada waktu dan tempat yang telah ditentukan. Bentuk undangan yang digunakan oleh pihak Posyandu untuk diberikan kepada warga dapat dilihat dalam Lampiran E.3. Untuk menjadwalkan kegiatan Posyandu saat ini (*as-is*) membutuhkan waktu sekitar 1 jam 30 menit hingga 2 jam tergantung dari banyaknya warga yang harus diberikan undangan secara langsung dan terdiri dari beberapa aktivitas diantaranya.

1. Menentukan jadwal & kegiatan Posyandu, seperti lokasi, waktu, maupun kegiatan yang akan dilakukan pada Posyandu tersebut.
2. Membuat undangan kegiatan Posyandu.
3. Mencetak undangan kegiatan Posyandu untuk didistribusikan
4. Mengundang dan mengirimkan undangan kepada para orang tua balita satu per satu.
5. Orang tua balita menerima undangan kegiatan Posyandu dari pihak Posyandu.

Dalam Gambar 4.2, merupakan hasil pemodelan proses bisnis untuk penjadwalan kegiatan Posyandu saat ini (*as-is*).



Gambar 4.2 Pemodelan proses bisnis penjadwalan kegiatan Posyandu saat ini (*as-is*)

4.2.2 Proses Bisnis Pelayanan Posyandu Saat Ini (*as-is*)

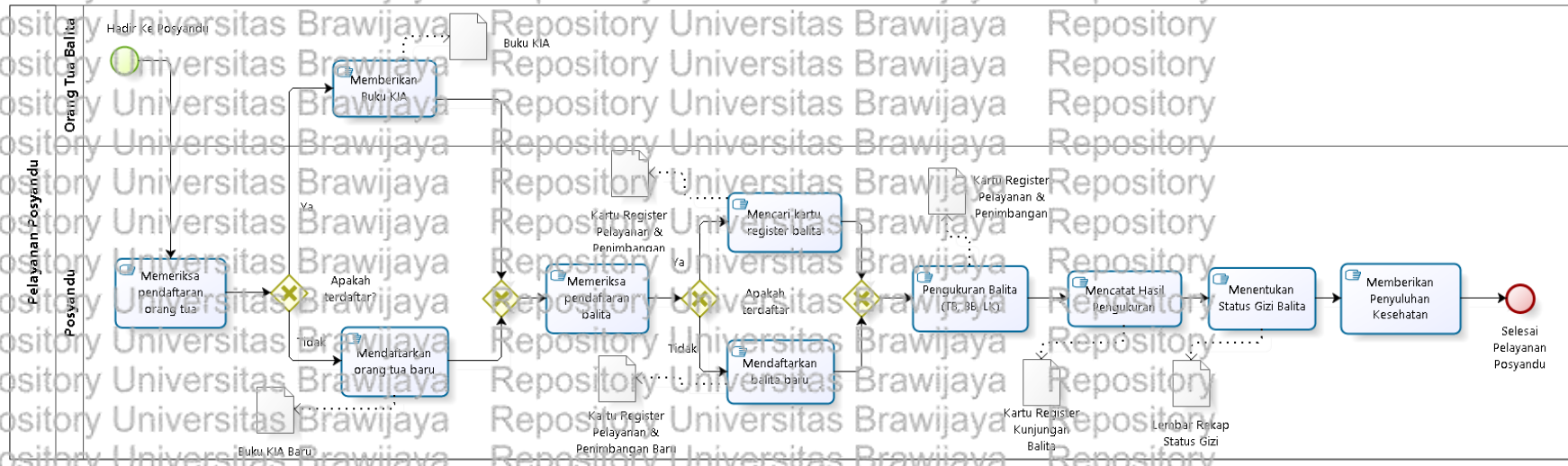
Setelah dilakukannya penjadwalan dan mengirimkan undangan kepada para orang tua, maka berikutnya adalah melakukan pelayanan kesehatan yang akan diberikan oleh para kader Posyandu kepada balita yang hadir pada Posyandu tersebut. Tujuan dari proses bisnis pelayanan Posyandu, khususnya untuk Posyandu balita, adalah untuk melakukan pengukuran dan pencatatan hasil pengukuran tersebut ke dalam catatan milik Posyandu, atau yang disebut dengan lembar register, agar baik dari pihak Posyandu maupun dari pihak orang tua dapat mengetahui dan memantau kemajuan pertumbuhan dari anak balita tersebut, agar apabila terjadi pertumbuhan yang tidak normal atau malnutrisi, maka pihak Posyandu dapat memberikan penanganan yang tepat kepada anak balita maupun memberikan saran kepada orang tua balita tersebut. Untuk dapat menyelesaikan proses pelayanan Posyandu saat ini (*as-is*) akan membutuhkan waktu rata-rata sekitar 15 menit untuk selesai melayani satu anak balita. Proses pelayanan Posyandu terdiri dari beberapa aktivitas diantaranya.

1. Saat orang tua balita hadir di Posyandu, maka pihak Posyandu akan memeriksa apakah orang tua tersebut sudah terdaftar di Posyandu tersebut.
2. Jika orang tua Posyandu tersebut sudah terdaftar, maka orang tua balita akan memberikan buku KIA kepada pihak Posyandu
3. Jika belum terdaftar, maka pihak Posyandu akan mendaftarkan terlebih dahulu orang tua balita tersebut dan akan memberikan buku KIA baru.
4. Pihak Posyandu kemudian memeriksa apakah balita dari orang tua tersebut sudah terdaftar pada Posyandu
5. Jika sudah terdaftar, maka pihak Posyandu akan langsung mencari kartu register pelayanan dan penimbangan milik balita tersebut



6. Jika belum terdaftar, maka pihak Posyandu akan mendaftarkan balita tersebut dan membuatkan kartu register pelayanan dan penimbangan yang baru.
7. Setelah itu, akan dilakukan penimbangan dan pengukuran balita yang mencakup tinggi / panjang badan, berat badan, dan lingkar kepala, dimana hasil pengukuran tersebut akan sementara dicatat pada kartu register pelayanan dan penimbangan milik balita tersebut.
8. Hasil pencatatan sementara pada kartu tersebut akan diberikan kepada kader Posyandu lain yang bertugas untuk mencatat hasil pengukuran tersebut pada kartu register kunjungan yang berisi data hasil pengukuran semua balita pada hari tersebut.
9. Dari data tersebut, akan dilakukan perhitungan status gizi dari balita tersebut, dan dicatat pada lembar rekap status gizi.
10. Setelah proses pencatatan selesai, maka pihak Posyandu dapat memberikan penyuluhan kesehatan kepada orang tua balita.

Dalam Gambar 4.3, merupakan hasil pemodelan proses bisnis untuk pelayanan Posyandu saat ini (*as-is*).



Gambar 4.3 Pemodelan proses bisnis pelayanan Posyandu saat ini (as-is)

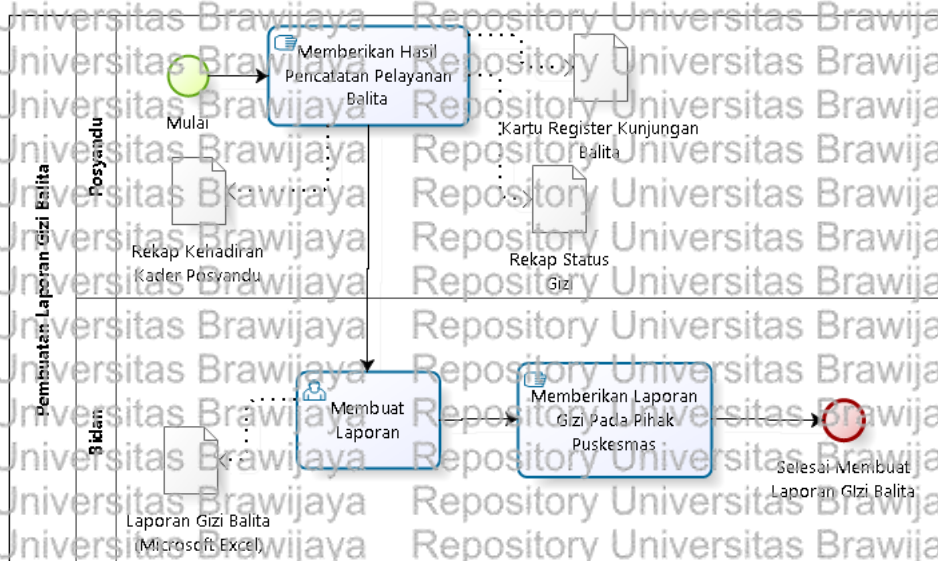


4.2.3 Analisis Proses Bisnis Pelaporan Gizi Saat Ini (*as-is*)

Setelah dilakukannya proses pelayanan kesehatan kepada balita pada Posyandu, maka proses berikutnya adalah pembuatan laporan gizi yang dilakukan oleh bidan yang bertanggung jawab. 1 orang bidan dapat bertanggung jawab terhadap beberapa Posyandu sekaligus dalam suatu wilayah, sehingga setelah proses pelayanan kesehatan pada Posyandu telah selesai dilakukan, maka hasil pencatatan masing-masing Posyandu akan diberikan kepada bidan untuk dibuatkan laporan gizi nya untuk nantinya dapat diteruskan kepada pihak Puskesmas. Untuk contoh laporan yang dibuat oleh bidan, dapat dilihat dalam Lampiran E.1 dan Lampiran E.2. Proses pembuatan laporan gizi ini membutuhkan waktu sekitar 2 jam, dan terdiri dari beberapa aktivitas diantaranya:

1. Pihak masing-masing Posyandu akan memberikan hasil pencatatan pelayanan balita yang berupa kartu register kunjungan balita, rekap status gizi, maupun rekap kehadiran kader Posyandu kepada bidan.
2. Bidan membuat laporan dengan menggunakan Microsoft Excel.
3. Bidan memberikan laporan tersebut kepada pihak Puskesmas.

Dalam Gambar 4.4, merupakan hasil pemodelan proses bisnis untuk penjadwalan kegiatan Posyandu saat ini (*as-is*).



Gambar 4.4 Pemodelan proses bisnis pembuatan laporan gizi balita saat ini (*as-is*)



4.2.4 Evaluasi Proses Bisnis Saat Ini (*as-is*)

Dari hasil pemodelan proses bisnis saat ini (*as-is*), serta melalui hasil observasi dan diskusi dengan pihak-pihak terkait (Lampiran A.1, Lampiran C.1, Lampiran C.2), maka dapat dilakukan proses evaluasi proses bisnis saat ini. Pada Tabel 4.1 dijabarkan mengenai hasil evaluasi proses bisnis saat ini yang terdiri dari kendala atau permasalahan, dampak, serta solusi yang ditawarkan.

Tabel 4.1 Evaluasi proses bisnis *as-is*

<p>Kendala / Permasalahan</p>	<ol style="list-style-type: none"> 1. Proses distribusi undangan Posyandu yang masih dilakukan secara manual oleh pihak Posyandu. 2. Proses pencatatan hasil pelayanan kesehatan anak balita di Posyandu, maupun yang terkait dengan administratif seperti pendaftaran balita yang masih dilakukan secara manual dan dilakukan berkali-kali pada dokumen yang berbeda akan rentan untuk mengalami kesalahan. 3. Pihak bidan harus membuat laporan hasil pelayanan secara manual dengan cara mencatat ulang hasil pencatatan pada buku catatan Posyandu ke sebuah laporan berupa file Microsoft Excel satu per satu untuk setiap Posyandu yang dipegang oleh bidan tersebut.
<p>Dampak</p>	<ol style="list-style-type: none"> 1. Tidak semua warga mendapatkan undangannya karena mungkin sedang tidak berada di rumah saat undangan didistribusikan, yang mengakibatkan tingkat partisipasi kegiatan Posyandu yang tidak optimal. 2. Proses pencatatan yang bersifat manual dan dilakukan berkali-kali pada dokumen yang berbeda tersebut masih rentan mengalami kesalahan sehingga dapat mengurangi validitas data hasil pengukuran dan pelayanan kesehatan balita yang dicatat. 3. Proses pembuatan laporan hasil pelayanan yang dilakukan oleh bidan dengan cara memindahkan data dari buku catatan Posyandu ke dalam laporan pada Microsoft Excel, juga rentan mengalami kesalahan, dan memerlukan waktu yang lama karena bidan harus merekap data anak satu per satu untuk membuat laporan bulanan.
<p>Solusi</p>	<ol style="list-style-type: none"> 1. Menyediakan sistem informasi yang dapat digunakan oleh pihak Posyandu untuk membuat dan mengirimkan undangan kegiatan Posyandu kepada para warga yang terdaftar di Posyandu tersebut 2. Menyediakan sistem informasi yang dapat digunakan oleh pihak Posyandu untuk mencatat hasil pelayanan kesehatan maupun administrasi pendaftaran orang tua dan balita yang berkunjung ke Posyandu selama proses pelayanan kesehatan berlangsung 3. Menyediakan sistem informasi yang dapat digunakan oleh bidan untuk membuat laporan gizi dari hasil pencatatan proses pelayanan kesehatan oleh masing-masing Posyandu.



4.2.5 Rekomendasi Proses Bisnis Penjadwalan Kegiatan Posyandu (to-be)

Pada proses bisnis rekomendasi untuk penjadwalan kegiatan Posyandu ini akan mencoba untuk menyelesaikan permasalahan terkait proses distribusi undangan kegiatan Posyandu kepada pihak orang tua balita, dimana pada proses bisnis yang direkomendasikan ini, akan menggunakan sistem informasi yang dapat membantu pihak Posyandu untuk membuat dan mengirimkan undangan Posyandu langsung kepada *smartphone* milik orang tua balita yang terdaftar pada sistem. Sehingga dengan adanya sistem informasi ini, pihak Posyandu tidak perlu untuk mencetak dan mendistribusikan secara manual satu per satu undangan untuk menghadiri kegiatan Posyandu kepada orang tua balita. Dengan adanya sistem informasi ini, proses penjadwalan kegiatan Posyandu dapat diselesaikan dalam waktu sekitar 15 menit, dan terdiri dari beberapa aktivitas yang direkomendasikan, diantaranya.

1. Menentukan jadwal & kegiatan Posyandu, seperti lokasi, waktu, maupun kegiatan yang akan dilakukan pada Posyandu tersebut.
2. Membuat undangan Posyandu pada form yang disediakan pada sistem.
3. Mengirimkan undangan tersebut kepada pihak orang tua balita.
4. Orang tua balita menerima undangan kegiatan Posyandu pada aplikasi Android mereka.

Dalam Gambar 4.5, merupakan hasil pemodelan rekomendasi proses bisnis penjadwalan kegiatan Posyandu (to-be).



Gambar 4.5 Rekomendasi proses bisnis penjadwalan kegiatan Posyandu (to-be)

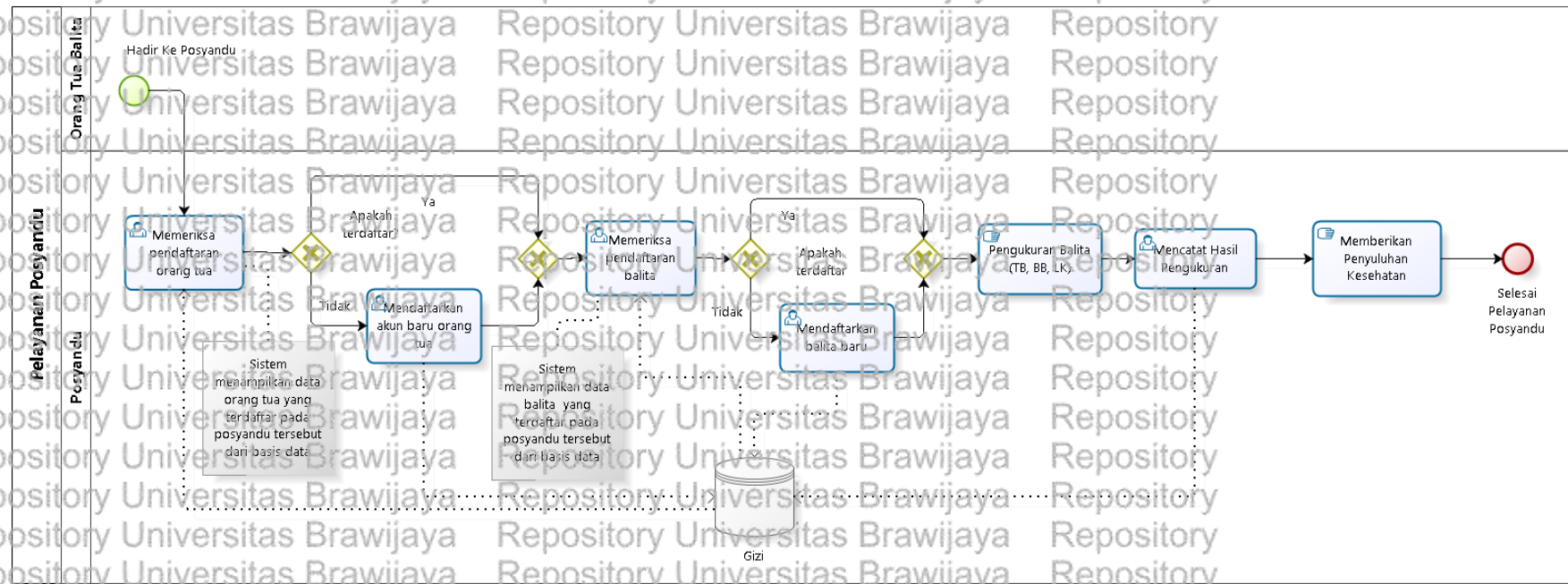


4.2.6 Rekomendasi Proses Bisnis Pelayanan Posyandu (*to-be*)

Pada proses bisnis rekomendasi untuk pelayanan Posyandu ini, akan mencoba untuk menyelesaikan permasalahan terkait proses pelayanan Posyandu yang dalam proses pencatatannya, baik untuk administratif maupun untuk pencatatan hasil pelayanan balitanya masih dilakukan secara manual menggunakan banyak dokumen yang berbeda. Pada proses bisnis yang direkomendasikan ini, akan menggunakan sistem informasi yang dapat membantu pihak Posyandu dalam melakukan pencarian, pendaftaran, maupun pencatatan hasil pelayanan Posyandu, dimana dengan adanya sistem informasi ini, dapat menggantikan pencatatan yang dilakukan secara berulang kali pada dokumen yang berbeda, sehingga diharapkan dapat meningkatkan validitas data yang dicatat dan meningkatkan efisiensi pelayanan Posyandu. Dengan adanya sistem informasi ini, proses pelayanan Posyandu untuk satu anak balita dan orang tuanya yang hadir dapat diselesaikan dalam kurun waktu sekitar 14 menit, dan terdiri dari beberapa aktivitas yang direkomendasikan, diantaranya.

1. Saat orang tua balita hadir di Posyandu, maka pihak Posyandu akan memeriksa apakah orang tua tersebut sudah terdaftar di Posyandu tersebut dengan melakukan pencarian pada data orang tua terdaftar yang ada pada sistem.
2. Jika belum terdaftar, maka pihak Posyandu akan mendaftarkan terlebih dahulu orang tua balita tersebut ke dalam sistem.
3. Pihak Posyandu kemudian memeriksa apakah balita dari orang tua tersebut sudah terdaftar pada Posyandu dengan melakukan pencarian data balita yang terdaftar pada sistem.
4. Jika belum terdaftar, maka pihak Posyandu akan mendaftarkan balita tersebut ke dalam sistem.
5. Setelah itu, pihak Posyandu akan melakukan penimbangan dan pengukuran balita yang mencakup tinggi / panjang badan, berat badan, dan lingkar kepala.
6. Pihak Posyandu kemudian mencatat hasil penimbangan dan pengukuran tersebut ke dalam sistem.
7. Setelah proses pencatatan selesai, maka pihak Posyandu dapat memberikan penyuluhan kesehatan kepada orang tua balita.

Dalam Gambar 4.6, merupakan hasil pemodelan rekomendasi proses bisnis pelayanan Posyandu (*to-be*).



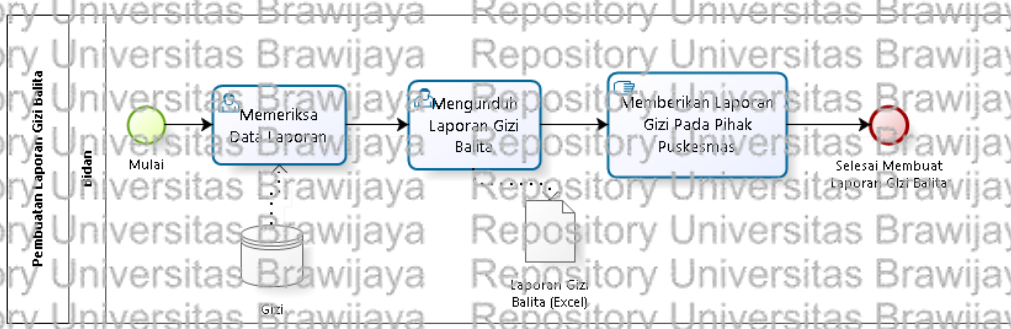
Gambar 4.6 Rekomendasi proses bisnis pelayanan Posyandu (to-be)



4.2.7 Rekomendasi Proses Bisnis Pembuatan Laporan Gizi (*to-be*)

Pada rekomendasi proses bisnis untuk pembuatan laporan gizi ini akan mencoba untuk menyelesaikan permasalahan yang dihadapi oleh pihak badan dalam membuat laporan gizi dari hasil pencatatan setiap Posyandu yang badan tersebut pegang. Pada proses bisnis yang direkomendasikan ini, badan hanya perlu memilih periode waktu laporan gizi yang dibutuhkan pada *filter* yang disediakan oleh sistem lalu mengunduh laporan gizi tersebut ke dalam format Microsoft Excel untuk nantinya dapat diteruskan kepada pihak Puskesmas. Proses pembuatan laporan gizi oleh badan hanya akan membutuhkan waktu sekitar 20 menit, dan terdiri dari beberapa aktivitas diantaranya.

1. Memeriksa data laporan yang ada pada sistem.
2. Mengunduh laporan gizi ke dalam bentuk Microsoft Excel.
3. Memberikan laporan gizi kepada pihak Puskesmas.



Gambar 4.7 Rekomendasi Proses bisnis pelaporan gizi (*to-be*)

4.2.8 Analisis Hasil Rekomendasi Proses Bisnis

Setelah dilakukannya rekomendasi proses bisnis untuk penjadwalan kegiatan Posyandu, pelayanan Posyandu, serta pembuatan laporan gizi, dimana didalamnya terdapat sistem informasi yang dapat memfasilitasi beberapa aktivitas yang dilakukan oleh aktor-aktor dalam menyelesaikan proses bisnis tersebut. Aturan penomoran kode aktivitas akan menggunakan aturan A-GIZI-X, dimana A merupakan singkatan dari aktivitas, GIZI merupakan representasi dari sistem yang akan dibuat, dan X merupakan nomor urut dari aktivitas. Pada Tabel 4.2 akan dijabarkan mengenai aktivitas-aktivitas dalam proses bisnis yang akan difasilitasi oleh sistem informasi yang akan dibangun.



Tabel 4.2 Aktivitas proses bisnis

Proses Bisnis	Kode Aktivitas	Aktivitas
Penjadwalan Kegiatan Posyandu	A-GIZI-001	Membuat dan mengirimkan undangan kegiatan Posyandu.
	A-GIZI-002	Mendaftarkan dan mengelola data orang tua balita di Posyandu.
Pelayanan Posyandu	A-GIZI-003	Mendaftarkan dan mengelola data balita di Posyandu.
	A-GIZI-004	Pencatatan hasil pelayanan dan pengukuran balita.
Pembuatan laporan gizi	A-GIZI-005	Membuat dan mengunggah laporan gizi.

4.3 Identifikasi Aktor

Berdasarkan hasil analisis permasalahan yang didapatkan melalui proses wawancara kepada Bapak Intan Yusuf Habibie S.Gz., M.Sc selaku dosen program studi ilmu gizi Fakultas Kedokteran UB (Lampiran A.1), dan wawancara kepada Ibu Komariah Tusokilah A.Md., selaku bidan pada Desa Krebet yang bertugas (Lampiran A.2), yang kemudian dilanjutkan dengan kegiatan observasi dan diskusi dengan pihak kader Posyandu pada Posyandu Balita Srikaya di Desa Krebet, Kabupaten Malang (Lampiran C.1 & Lampiran C.2), maka didapatkan aktor-aktor yang akan berperan dalam sistem ini. Pada identifikasi aktor ini menggunakan aturan penomoran ACT-GIZI-X, dimana ACT merupakan singkatan dari *actor*, GIZI merupakan representasi dari sistem yang akan dibuat, dan X merupakan nomor urut dari aktor yang didefinisikan. Terdapat 4 aktor utama yang dapat menggunakan sistem ini yaitu aktor *Guest*, *Admin*, Posyandu, serta Bidan yang akan dijabarkan pada Tabel 4.3.

Tabel 4.3 Identifikasi aktor

Kode Aktor	Aktor	Deskripsi
ACT-GIZI-001	<i>Guest</i>	<i>Guest</i> merupakan pengguna yang bisa melakukan registrasi sebagai bidan dan melakukan autentikasi ke dalam sistem.
ACT-GIZI-002	<i>Admin</i>	<i>Admin</i> merupakan pengguna yang memiliki akses untuk melihat dan memvalidasi aktor Bidan yang mendaftar pada sistem.
ACT-GIZI-003	Posyandu	Posyandu merupakan pengguna yang memiliki hak akses untuk melakukan proses pendaftaran dan pengelolaan data orang tua dan pencatatan hasil pelayanan kesehatan balita.
ACT-GIZI-004	Bidan	Bidan merupakan pengguna yang memiliki hak akses dalam melihat laporan gizi dan hasil pelayanan kesehatan Posyandu serta mengelola Posyandu dan kader yang terdaftar pada sistem.



4.4 Kebutuhan Fungsional

Berdasarkan hasil diskusi lanjutan untuk memvalidasi kebutuhan dengan menggunakan *prototype* sistem kepada Bapak Intan Yusuf Habibie S.Gz., M.Sc selaku dosen program studi ilmu gizi Fakultas Kedokteran UB (Lampiran D), serta data-data pendukung yang didapatkan melalui observasi (Lampiran E), maka dapat dilakukan proses penjabaran kebutuhan fungsional untuk aktor-aktor yang akan berperan dalam sistem. Untuk aturan penomoran untuk kebutuhan fungsional akan menggunakan kode KF-GIZI-X. KF sendiri merupakan singkatan dari kebutuhan fungsional, GIZI merupakan representasi dari sistem yang akan dibangun, dan X adalah urutan penomoran dari pendefinisian kebutuhan fungsional. Pada Tabel 4.4 akan dijabarkan kebutuhan fungsional dari sistem informasi yang akan dibangun.

Tabel 4.4 Kebutuhan fungsional sistem

Aktor	Kode Kebutuhan	Dasar Kebutuhan	Kebutuhan
Guest (ACT-GIZI-001)	KF-GIZI-001	Lampiran D.3	Sistem harus bisa melakukan autentikasi pada Admin, Bidan atau Posyandu yang sudah terdaftar sebelumnya untuk mengakses fungsionalitas yang tersedia dalam sistem.
	KF-GIZI-002	Lampiran D.3	Sistem harus bisa melakukan registrasi bidan baru.
	KF-GIZI-003	Lampiran D.3	Sistem harus bisa menampilkan data bidan yang aktif dan sudah tervalidasi sebelumnya.
Admin (ACT-GIZI-002)	KF-GIZI-004	Lampiran D.3	Sistem harus bisa memperbaharui data bidan yang aktif dan sudah tervalidasi sebelumnya.
	KF-GIZI-005	Lampiran D.3	Sistem harus bisa menghapus data bidan yang aktif dan sudah tervalidasi sebelumnya.
	KF-GIZI-006	Lampiran D.3	Sistem harus bisa menampilkan data bidan yang perlu untuk divalidasi.
	KF-GIZI-007	Lampiran D.3	Sistem harus bisa memvalidasi data bidan yang mendaftar ke dalam sistem.
Posyandu (ACT-GIZI-003)	KF-GIZI-008	A-GIZI-001, Lampiran D.3	Sistem harus bisa menjadwalkan kegiatan Posyandu.
	KF-GIZI-009	A-GIZI-002, Lampiran D.3	Sistem harus bisa mendaftarkan dan membuat akun baru untuk orang tua balita.
	KF-GIZI-010	A-GIZI-002, Lampiran D.3	Sistem harus bisa menampilkan data orang tua balita yang sudah terdaftar dalam bentuk tabel.



Tabel 4.4 Kebutuhan fungsional sistem (lanjutan)

Aktor	Kode Kebutuhan	Dasar Kebutuhan	Kebutuhan
Posyandu (ACT-GIZI-003)	KF-GIZI-011	A-GIZI-002, Lampiran D.3	Sistem harus bisa memperbaharui data orang tua balita yang sudah terdaftar.
	KF-GIZI-012	A-GIZI-002, Lampiran D.3	Sistem harus bisa menghapus data orang tua balita yang sudah terdaftar.
	KF-GIZI-013	A-GIZI-003, Lampiran D.3	Sistem harus bisa mendaftarkan balita baru di Posyandu.
	KF-GIZI-014	A-GIZI-003, Lampiran D.3	Sistem harus bisa menampilkan data balita yang sudah terdaftar di Posyandu tersebut dalam bentuk tabel.
	KF-GIZI-015	A-GIZI-003, Lampiran D.3	Sistem harus bisa memperbaharui data balita yang sudah terdaftar di Posyandu tersebut.
	KF-GIZI-016	A-GIZI-003, Lampiran D.3	Sistem harus bisa menghapus data balita yang sudah terdaftar di Posyandu tersebut.
	KF-GIZI-017	Lampiran D.3	Sistem harus bisa mendaftarkan kader di Posyandu.
	KF-GIZI-018	A-GIZI-004, Lampiran D.3	Sistem harus bisa melakukan pencatatan kehadiran dan hasil pelayanan kesehatan pada anak balita.
	KF-GIZI-019	Lampiran D.3	Sistem harus bisa mendaftarkan Posyandu baru.
	KF-GIZI-020	Lampiran D.3	Sistem harus bisa mendaftarkan kader pada salah satu Posyandu yang sudah terdaftar.
Bidan (ACT-GIZI-004)	KF-GIZI-021	Lampiran D.3	Sistem harus bisa menampilkan data Posyandu dan kader Posyandu yang sudah terdaftar.
	KF-GIZI-022	Lampiran D.3	Sistem harus bisa memperbaharui data Posyandu yang sudah terdaftar.
	KF-GIZI-023	Lampiran D.3	Sistem harus bisa menghapus data Posyandu yang sudah terdaftar.
	KF-GIZI-024	Lampiran D.3	Sistem harus bisa memperbaharui data kader Posyandu yang sudah terdaftar.
	KF-GIZI-025	Lampiran D.3	Sistem harus bisa menghapus data kader Posyandu yang sudah terdaftar.
	KF-GIZI-026	A-GIZI-005, Lampiran D.3	Sistem harus bisa menampilkan laporan gizi dan hasil pelayanan Posyandu.



Tabel 4.4 Kebutuhan fungsional sistem (lanjutan)

Aktor	Kode Kebutuhan	Dasar Kebutuhan	Kebutuhan
Bidan (ACT-GIZI-004)	KE-GIZI-027	A-GIZI-005, Lampiran D.3	Sistem harus bisa mengundun laporan gizi dan hasil pelayanan Posyandu ke dalam bentuk Microsoft Excel.
	KE-GIZI-028	A-GIZI-003, Lampiran D.3	Sistem harus bisa menampilkan profil balita dan riwayat hasil pemeriksaan kesehatan balita setiap bulan dalam bentuk tabel.
Posyandu (ACT-GIZI-003), Bidan (ACT-GIZI-004)	KE-GIZI-029	A-GIZI-003, Lampiran D.3	Sistem harus bisa memperbaharui data hasil pemeriksaan kesehatan balita.
	KE-GIZI-030	A-GIZI-003, Lampiran D.3	Sistem harus bisa menghapus data hasil pemeriksaan kesehatan balita.
	KE-GIZI-031	A-GIZI-003, Lampiran D.3	Sistem harus bisa menampilkan pertumbuhan berat badan anak balita dalam bentuk Kartu Menuju Sehat.
	KE-GIZI-032	Lampiran D.3	Sistem harus bisa memberikan saran/rekomendasi asupan gizi balita.
Admin (ACT-GIZI-002), Posyandu (ACT-GIZI-003), Bidan (ACT-GIZI-004)	KE-GIZI-033	Lampiran D.3	Sistem harus bisa menghentikan sesi dan keluar menuju ke halaman <i>login</i> untuk melakukan autentikasi ulang.

4.5 Spesifikasi Kebutuhan Fungsional

Dari hasil analisis kebutuhan fungsional yang telah dijabarkan pada bagian sebelumnya, berikutnya akan dilakukan penjabaran mengenai spesifikasi dari kebutuhan fungsional tersebut, untuk mendapatkan gambaran yang lebih detail mengenai apa saja yang perlu diimplementasikan dari sistem untuk memenuhi kebutuhan fungsional tersebut. Untuk aturan penomoran pada spesifikasi kebutuhan menggunakan akan mengikut aturan penomoran pada pendefinisian kebutuhan fungsional, namun ditambahkan satu penomoran lagi dibagian belakang sehingga menjadi KE-GIZI-X-Y, dimana Y menunjukkan nomor urut spesifikasi dari definisi kebutuhan utama. Pada Tabel 4.5 merupakan penjabaran dari spesifikasi kebutuhan fungsional dari sistem informasi yang akan dibangun.



Tabel 4.5 Spesifikasi kebutuhan fungsional

Kode Kebutuhan	Kode SRS	Deskripsi
KF-GIZI-001	KF-GIZI-001-1	Sistem harus bisa menampilkan <i>form</i> yang berisi kolom untuk memasukkan <i>username</i> dan <i>password</i> .
	KF-GIZI-001-2	Sistem harus bisa menampilkan peringatan apabila pengguna tidak mengisi <i>form</i> secara lengkap
	KF-GIZI-001-3	Sistem harus bisa menampilkan peringatan apabila <i>username</i> atau <i>password</i> yang dimasukkan salah.
	KF-GIZI-001-4	Sistem terdapat tombol " <i>login</i> " untuk melakukan autentikasi pengguna.
KF-GIZI-002	KF-GIZI-002-1	Sistem harus bisa menampilkan <i>form</i> yang berisi kolom untuk memasukkan nomor registrasi, nama, tempat tanggal lahir, <i>username</i> , <i>password</i> , dan konfirmasi <i>password</i> untuk mendaftarkan badan baru ke dalam sistem
	KF-GIZI-002-2	Sistem terdapat tombol untuk mengunggah berkas surat registrasi dan foto diri badan.
	KF-GIZI-002-3	Sistem harus bisa menampilkan peringatan apabila pengguna tidak mengisi <i>form</i> secara lengkap.
	KF-GIZI-002-4	Sistem terdapat tombol " <i>registrasi</i> " untuk mengirimkan dan menyimpan data sesuai dengan masukan pengguna.
	KF-GIZI-002-5	Sistem harus bisa menampilkan <i>pop-up</i> peringatan apabila pendaftaran berhasil maupun gagal.
KF-GIZI-003	KF-GIZI-003-1	Sistem harus bisa menampilkan data nomor registrasi, nama, <i>username</i> , dan status dari badan yang sudah tervalidasi dan aktif dalam tabel.
	KF-GIZI-003-2	Sistem terdapat kolom untuk memasukkan kata kunci yang dapat digunakan untuk melakukan pencarian terhadap data badan yang sudah tervalidasi dan aktif yang ada pada tabel.
	KF-GIZI-003-3	Sistem terdapat tombol detail pada setiap data badan pada tabel yang dapat menampilkan detail data nomor registrasi, nama, tempat dan tanggal lahir, <i>username</i> , foto diri, serta surat registrasi badan yang sudah tervalidasi dan aktif dalam bentuk <i>pop up</i> .
KF-GIZI-004	KF-GIZI-004-1	Sistem terdapat tombol <i>edit</i> pada tabel data badan yang sudah tervalidasi dan aktif yang dapat menampilkan <i>form</i> untuk memperbaharui data badan yang dipilih tersebut.
	KF-GIZI-004-2	Sistem terdapat tombol " <i>simpan</i> " untuk menyimpan perubahan data badan yang telah dilakukan.
	KF-GIZI-004-3	Sistem harus bisa menampilkan pesan perubahan data badan berhasil / gagal setelah tombol <i>simpan</i> ditekan.



Tabel 4.5 Spesifikasi kebutuhan fungsional (lanjutan)

Kode Kebutuhan	Kode SRS	Deskripsi
KF-GIZI-005	KF-GIZI-005-1	Sistem terdapat tombol hapus pada tabel data bidang untuk menonaktifkan bidang tersebut.
	KF-GIZI-005-2	Sistem harus bisa menampilkan pesan konfirmasi setelah menekan tombol hapus.
	KF-GIZI-005-3	Sistem harus bisa menampilkan pesan penonaktifan data bidang berhasil / gagal.
KF-GIZI-006	KF-GIZI-006-1	Sistem harus bisa menampilkan data nomor registrasi, nama, <i>username</i> dan status dari bidang yang perlu untuk divalidasi dalam tabel.
	KF-GIZI-006-2	Sistem terdapat kolom untuk memasukkan kata kunci yang dapat digunakan untuk melakukan pencarian terhadap data bidang yang perlu divalidasi ada pada tabel.
	KF-GIZI-006-3	Sistem terdapat tombol detail pada setiap data bidang di tabel yang dapat menampilkan detail data nomor registrasi, nama, tempat dan tanggal lahir, <i>username</i> , foto diri, serta surat registrasi bidang yang sudah tervalidasi dalam bentuk <i>pop up</i> .
KF-GIZI-007	KF-GIZI-007-1	Sistem terdapat tombol "Validasi" pada <i>pop up</i> data bidang yang dapat menerima dan mengaktifkan bidang yang mendaftar tersebut.
KF-GIZI-008	KF-GIZI-008-1	Sistem harus bisa menampilkan <i>form</i> yang berisi kolom untuk memasukkan tanggal, waktu, tempat, dan deskripsi kegiatan Posyandu tersebut.
	KF-GIZI-008-2	Sistem terdapat tombol "jadwalkan" untuk menyimpan dan mengirimkan jadwal hari Posyandu ke akun orang tua balita yang terdaftar di Posyandu tersebut.
	KF-GIZI-008-3	Sistem harus bisa menampilkan pesan penjadwalan berhasil / gagal setelah tombol penjadwalan ditekan.
	KF-GIZI-008-4	Sistem harus bisa menampilkan hari Posyandu berikutnya yang telah terjadwalkan pada halaman <i>dashboard</i> Posyandu.
KF-GIZI-009	KF-GIZI-009-1	Sistem harus bisa menampilkan <i>form</i> yang berisi kolom untuk memasukkan nama ayah dan ibu, NIK ayah dan ibu, nomor kartu keluarga, alamat, nomor <i>handphone</i> , <i>username</i> , <i>password</i> serta konfirmasi <i>password</i> .
	KF-GIZI-009-2	Sistem terdapat tombol "daftar" untuk menyimpan data akun orang tua balita pada sistem.
	KF-GIZI-009-3	Sistem harus bisa menampilkan pesan pendaftaran berhasil / gagal setelah tombol "daftar" ditekan.



Tabel 4.5 Spesifikasi kebutuhan fungsional (lanjutan)

Kode Kebutuhan	Kode SRS	Deskripsi
KF-GIZI-010	KF-GIZI-010-1	Sistem harus bisa menampilkan nama ayah dan ibu, alamat, <i>username</i> dari data orang tua balita yang telah terdaftar pada tabel.
	KF-GIZI-010-2	Sistem terdapat kolom untuk memasukkan kata kunci yang dapat digunakan untuk melakukan pencarian terhadap data orang tua balita yang telah terdaftar yang ada pada tabel.
KF-GIZI-011	KF-GIZI-011-1	Sistem terdapat tombol <i>edit</i> pada tabel data orang tua yang dapat menampilkan <i>form</i> untuk memperbaharui data orang tua balita tersebut.
	KF-GIZI-011-2	Sistem terdapat tombol "simpan" untuk menyimpan perubahan data orang tua balita yang dilakukan.
	KF-GIZI-011-3	Sistem harus bisa menampilkan pesan perubahan data orang tua balita berhasil / gagal setelah tombol "simpan" ditekan.
KF-GIZI-012	KF-GIZI-012-1	Sistem terdapat tombol hapus pada tabel data orang tua balita untuk menonaktifkan data orang tua balita tersebut.
	KF-GIZI-012-2	Sistem harus bisa menampilkan pesan konfirmasi setelah menekan tombol hapus.
	KF-GIZI-012-3	Sistem harus bisa menampilkan pesan peronaktifan data orang tua balita berhasil / gagal.
KF-GIZI-013	KF-GIZI-013-1	Sistem harus bisa menampilkan <i>form</i> yang berisi kolom untuk memasukkan <i>username</i> akun orang tua, nama balita, jenis kelamin, tanggal lahir, berat badan lahir, NIK, dan nomor BPJS anak balita.
	KF-GIZI-013-2	Sistem harus bisa melakukan pencarian <i>username</i> akun orang tua balita yang sudah terdaftar pada sistem pada kolom <i>username</i> akun orang tua balita.
	KF-GIZI-013-3	Sistem terdapat tombol "daftar" untuk menyimpan data dari balita baru pada sistem.
KF-GIZI-014	KF-GIZI-013-4	Sistem harus bisa menampilkan pesan pendaftaran balita baru tersebut berhasil / gagal setelah tombol daftar ditekan.
	KF-GIZI-014-1	Sistem harus bisa menampilkan nama ayah dan ibu, nama anak, jenis kelamin, umur (bulan) dari anak balita yang telah terdaftar pada tabel.
	KF-GIZI-014-2	Sistem terdapat kolom untuk memasukkan kata kunci yang dapat digunakan untuk melakukan pencarian terhadap data anak balita yang telah terdaftar yang ada pada tabel.



Tabel 4.5 Spesifikasi kebutuhan fungsional (lanjutan)

Kode Kebutuhan	Kode SRS	Deskripsi
KF-GIZI-014	KF-GIZI-014-3	Sistem terdapat tombol detail pada tabel untuk menampilkan halaman profil dan riwayat pemeriksaan anak balita tersebut.
	KF-GIZI-015-1	Sistem terdapat tombol <i>edit</i> pada tabel data balita yang dapat menampilkan <i>form</i> untuk memperbaharui data balita tersebut.
KF-GIZI-015	KF-GIZI-015-2	Sistem terdapat tombol "simpan" untuk menyimpan perubahan data balita yang dilakukan.
	KF-GIZI-015-3	Sistem harus bisa menampilkan pesan perubahan data balita berhasil / gagal setelah tombol "simpan" ditekan.
	KF-GIZI-016-1	Sistem terdapat tombol hapus pada tabel data balita untuk menonaktifkan balita tersebut.
KF-GIZI-016	KF-GIZI-016-2	Sistem harus bisa menampilkan pesan konfirmasi setelah menekan tombol hapus.
	KF-GIZI-016-3	Sistem harus bisa menampilkan pesan penonaktifan balita berhasil / gagal.
	KF-GIZI-017-1	Sistem harus bisa menampilkan <i>form</i> yang berisi kolom untuk memasukkan tanggal Posyandu, dan nama kader.
KF-GIZI-017	KF-GIZI-017-2	Sistem harus bisa melakukan pencarian nama kader yang terdaftar pada Posyandu tersebut pada kolom nama kader.
	KF-GIZI-017-3	Sistem terdapat tombol "simpan" untuk mendaftarkan kehadiran kader Posyandu pada form yang ditampilkan.
	KF-GIZI-017-4	Sistem harus bisa menampilkan pesan pendaftaran kehadiran kader tersebut berhasil / gagal setelah tombol "simpan" ditekan.
KF-GIZI-018	KF-GIZI-018-1	Sistem harus bisa menampilkan <i>form</i> yang berisi kolom untuk memasukkan tanggal Posyandu, nama balita, hasil pengukuran dan pemeriksaan kesehatan balita yang meliputi berat badan, tinggi badan, lingkaran kepala, cara pengukuran, imunisasi, pemberian Vit. A, ASI eksklusif, PMT, keluhan maupun hasil diagnosa.
	KF-GIZI-018-2	Sistem harus bisa melakukan pencarian nama balita yang sudah terdaftar pada Posyandu tersebut pada kolom nama balita.
	KF-GIZI-018-3	Sistem terdapat tombol "simpan" untuk menyimpan data kehadiran dan hasil pelayanan kesehatan balita di Posyandu pada <i>form</i> yang ditampilkan.
	KF-GIZI-018-4	Sistem harus bisa menampilkan pesan penyimpanan hasil pencatatan kehadiran dan pelayanan kesehatan balita tersebut berhasil / gagal setelah tombol "simpan" ditekan.



Tabel 4.5 Spesifikasi kebutuhan fungsional (lanjutan)

Kode Kebutuhan	Kode SRS	Deskripsi
KF-GIZI-019	KF-GIZI-019-1	Sistem harus bisa menampilkan <i>form</i> yang berisi kolom untuk memasukkan nama Posyandu, <i>username</i> , alamat, RT, RW, kelurahan / desa, <i>password</i> dan konfirmasi <i>password</i> .
	KF-GIZI-019-2	Sistem terdapat tombol untuk mengunggah foto profil dari Posyandu yang ingin didaftarkan.
	KF-GIZI-019-3	Sistem terdapat tombol "daftar" untuk menyimpan data dan membuat akun Posyandu baru pada sistem.
	KF-GIZI-019-4	Sistem harus bisa menampilkan pesan pendaftaran Posyandu baru berhasil / gagal setelah tombol "daftar" ditekan.
KF-GIZI-020	KF-GIZI-020-1	Sistem harus bisa menampilkan <i>form</i> yang berisi kolom untuk memasukkan nama kader, NIK, nama Posyandu, jenis kelamin, dan alamat kader.
	KF-GIZI-020-2	Sistem harus bisa melakukan pencarian nama Posyandu yang sudah terdaftar pada sistem pada kolom nama Posyandu.
	KF-GIZI-020-3	Sistem terdapat tombol untuk mengunggah foto KTP kader.
	KF-GIZI-020-4	Sistem terdapat tombol "daftar" untuk menyimpan data kader Posyandu pada sistem.
	KF-GIZI-020-5	Sistem harus bisa menampilkan pesan pendaftaran Posyandu berhasil / gagal setelah tombol "daftar" ditekan.
KF-GIZI-021	KF-GIZI-021-1	Sistem harus bisa menampilkan nama, alamat, RT, RW, kelurahan / desa, dan <i>username</i> dari Posyandu pada tabel.
	KF-GIZI-021-2	Sistem terdapat kolom untuk memasukkan kata kunci yang dapat digunakan untuk melakukan pencarian terhadap data Posyandu yang sudah terdaftar yang ada pada tabel.
KF-GIZI-022	KF-GIZI-022-1	Sistem terdapat tombol <i>edit</i> pada tabel data Posyandu yang dapat menampilkan <i>form</i> untuk memperbaharui data Posyandu tersebut.
	KF-GIZI-022-2	Sistem terdapat tombol "simpan" untuk menyimpan perubahan data Posyandu yang dilakukan.
	KF-GIZI-022-3	Sistem harus bisa menampilkan pesan perubahan data Posyandu berhasil / gagal setelah tombol "simpan" ditekan.



Tabel 4.5 Spesifikasi kebutuhan fungsional (lanjutan)

Kode Kebutuhan	Kode SRS	Deskripsi
KF-GIZI-023	KF-GIZI-023-1	Sistem terdapat tombol hapus pada tabel data Posyandu untuk menonaktifkan Posyandu tersebut.
	KF-GIZI-023-2	Sistem harus bisa menampilkan pesan konfirmasi setelah menekan tombol "hapus".
	KF-GIZI-023-3	Sistem harus bisa menampilkan pesan penonaktifan data Posyandu berhasil / gagal.
KF-GIZI-024	KF-GIZI-024-1	Sistem terdapat tombol <i>edit</i> pada tabel data kader Posyandu yang dapat menampilkan <i>form</i> untuk memperbaharui data kader Posyandu tersebut.
	KF-GIZI-24-2	Sistem terdapat tombol "simpan" untuk menyimpan perubahan data kader Posyandu yang dilakukan.
	KF-GIZI-024-3	Sistem harus bisa menampilkan pesan perubahan data kader Posyandu berhasil / gagal setelah tombol simpan ditekan.
KF-GIZI-025	KF-GIZI-025-1	Sistem terdapat tombol "hapus" pada tabel data kader Posyandu untuk menonaktifkan kader Posyandu tersebut.
	KF-GIZI-025-2	Sistem menampilkan pesan konfirmasi setelah menekan tombol "hapus".
	KF-GIZI-025-3	Sistem harus bisa menampilkan pesan penonaktifan kader Posyandu berhasil / gagal.
KF-GIZI-026	KF-GIZI-026-1	Sistem harus bisa melakukan seleksi data hasil pelayanan Posyandu berdasarkan nama Posyandu untuk menjadi sumber data laporan gizi dan hasil pelayanan Posyandu.
	KF-GIZI-026-2	Sistem harus bisa melakukan seleksi data hasil laporan gizi dan pelayanan Posyandu berdasarkan bulan maupun tahun tertentu.
	KF-GIZI-026-3	Sistem harus bisa menampilkan laporan anak balita pertumbuhan normal yang meliputi data nama balita, jenis kelamin, umur (bulan) berat badan, tinggi badan, lingkaran kepala, dan status gizi dalam bentuk BB/U, TB/U, dan BB/TB pada tabel.
	KF-GIZI-026-4	Sistem harus bisa menampilkan laporan anak balita malnutrisi yang meliputi data nama balita, jenis kelamin, umur (bulan) berat badan, tinggi badan, lingkaran kepala, dan status gizi dalam bentuk BB/U, TB/U, dan BB/TB pada tabel.
	KF-GIZI-026-5	Sistem terdapat tombol detail pada tabel laporan anak balita pertumbuhan normal dan laporan anak balita malnutrisi untuk menampilkan halaman profil dan riwayat pemeriksaan anak balita tersebut.



Tabel 4.5 Spesifikasi kebutuhan fungsional (lanjutan)

Kode Kebutuhan	Kode SRS	Deskripsi
	KF-GIZI-026-6	Sistem terdapat kolom untuk memasukkan kata kunci yang dapat digunakan untuk melakukan pencarian terhadap data laporan anak balita pertumbuhan normal dan anak balita malnutrisi yang ada pada tabel.
KF-GIZI-026	KF-GIZI-026-7	Sistem harus bisa menampilkan laporan kehadiran balita yang meliputi data angka kehadiran balita dan data angka balita yang terdaftar pada suatu Posyandu pada periode waktu tersebut dalam bentuk <i>bar chart</i> .
	KF-GIZI-026-8	Sistem harus bisa menampilkan laporan kehadiran kader yang meliputi data angka kehadiran kader dan data angka kader yang terdaftar pada suatu Posyandu pada periode waktu tersebut dalam bentuk <i>bar chart</i> .
KF-GIZI-027	KF-GIZI-027-1	Sistem terdapat tombol unduh untuk mengunduh laporan balita pertumbuhan normal atau laporan balita malnutrisi ke dalam bentuk Microsoft Excel.
	KF-GIZI-028-1	Sistem harus bisa menampilkan data balita seperti nama, jenis kelamin, tanggal lahir, umur (bulan), berat badan lahir, NIK, No. BPJS, alamat tempat tinggal, nama ayah dan ibu, nomor <i>handphone</i> , status gizi balita dalam bentuk BB/U, TB/U, dan BB/TB.
KF-GIZI-028	KF-GIZI-028-2	Sistem harus bisa menampilkan data hasil pemeriksaan kesehatan balita seperti tanggal periksa, umur (bulan), tinggi badan, berat badan, lingkar kepala, pemberian imunisasi, vitamin A, ASI, serta PMT pada tabel.
	KF-GIZI-029-1	Sistem terdapat tombol "edit" pada tabel hasil riwayat pemeriksaan kesehatan balita yang dapat menampilkan <i>form</i> untuk memperbaiki data hasil pemeriksaan kesehatan tersebut.
KF-GIZI-029	KF-GIZI-029-2	Sistem terdapat tombol "simpan" untuk menyimpan perubahan data hasil pemeriksaan kesehatan balita yang dilakukan.
	KF-GIZI-029-3	Sistem harus bisa menampilkan pesan perubahan hasil pemeriksaan kesehatan balita tersebut berhasil / gagal setelah tombol "simpan" ditekan.
	KF-GIZI-030-1	Sistem terdapat tombol "hapus" pada tabel riwayat hasil pemeriksaan kesehatan balita untuk menghapus data hasil pemeriksaan kesehatan balita tersebut.
KF-GIZI-030	KF-GIZI-030-2	Sistem menampilkan pesan konfirmasi penghapusan data hasil pemeriksaan kesehatan balita setelah menekan tombol "hapus".
	KF-GIZI-030-3	Sistem harus bisa menampilkan pesan penghapusan data hasil pemeriksaan kesehatan balita berhasil / gagal.



Tabel 4.5 Spesifikasi kebutuhan fungsional (lanjutan)

Kode Kebutuhan	Kode SRS	Deskripsi
KF-GIZI-031	KF-GIZI-031-1	Sistem terdapat tombol lihat KMS pada profil balita untuk harus bisa menampilkan Kartu Menuju Sehat balita tersebut.
	KF-GIZI-031-2	Sistem harus bisa menampilkan kurva pertumbuhan berat badan anak balita.
	KF-GIZI-031-3	Sistem harus bisa menampilkan standar kurva pertumbuhan berat badan balita mulai dari -3SD, -2 SD, -1 SD, median, + 1 SD, + 2 SD, dan + 3 SD sesuai dengan standar antropometri berat badan anak balita.
	KF-GIZI-031-4	Sistem harus bisa menampilkan informasi hasil pengukuran setiap bulannya yang berisi umur (bulan), bulan penimbangan, berat badan, angka Kenaikan Berat Badan Minimal (KBM), indikator naik/turunnya berat badan balita, dan pemberian ASI eksklusif.
KF-GIZI-032	KF-GIZI-032-1	Sistem terdapat tombol beri saran gizi pada profil balita untuk menampilkan <i>form</i> pemberian saran gizi kepada balita tersebut.
	KF-GIZI-032-2	Sistem harus bisa menampilkan <i>form</i> yang berisi kolom nama balita dan kolom untuk memasukkan saran gizi kepada balita tersebut.
	KF-GIZI-032-3	Sistem terdapat tombol "kirim" untuk mengirimkan saran gizi kepada akun orang tua balita tersebut.
	KF-GIZI-032-4	Sistem harus bisa menampilkan pesan pemberian saran gizi kepada anak balita tersebut berhasil / gagal setelah tombol "kirim" ditekan.
KF-GIZI-033	KF-GIZI-033-1	Sistem terdapat tombol "logout" pada bagian <i>navbar</i> yang dapat mengeluarkan pengguna dari sistem.
	KF-GIZI-033-2	Sistem harus bisa menampilkan pesan konfirmasi setelah tombol "logout" ditekan.

4.6 Kebutuhan Non Fungsional

Pada bagian ini akan dijelaskan mengenai kebutuhan non fungsional dari sistem informasi yang akan dibangun. Aturan penomoran untuk kebutuhan non fungsional adalah menggunakan NF-GIZI-X, dimana NF merupakan singkatan dari *non functional*, GIZI merupakan representasi dari sistem yang akan dibangun, sedangkan X merupakan penomoran dari kebutuhan non fungsional yang akan didefinisikan. Kebutuhan non fungsional yang ada pada sistem informasi ini merupakan *usability* dengan kode kebutuhan NF-GIZI-001. Aspek *usability* disini berarti sistem yang akan dibangun adalah sistem yang memiliki tampilan sederhana dan jelas sehingga penggunaanya dapat dengan mudah mempelajari dan menggunakan sistem ini.

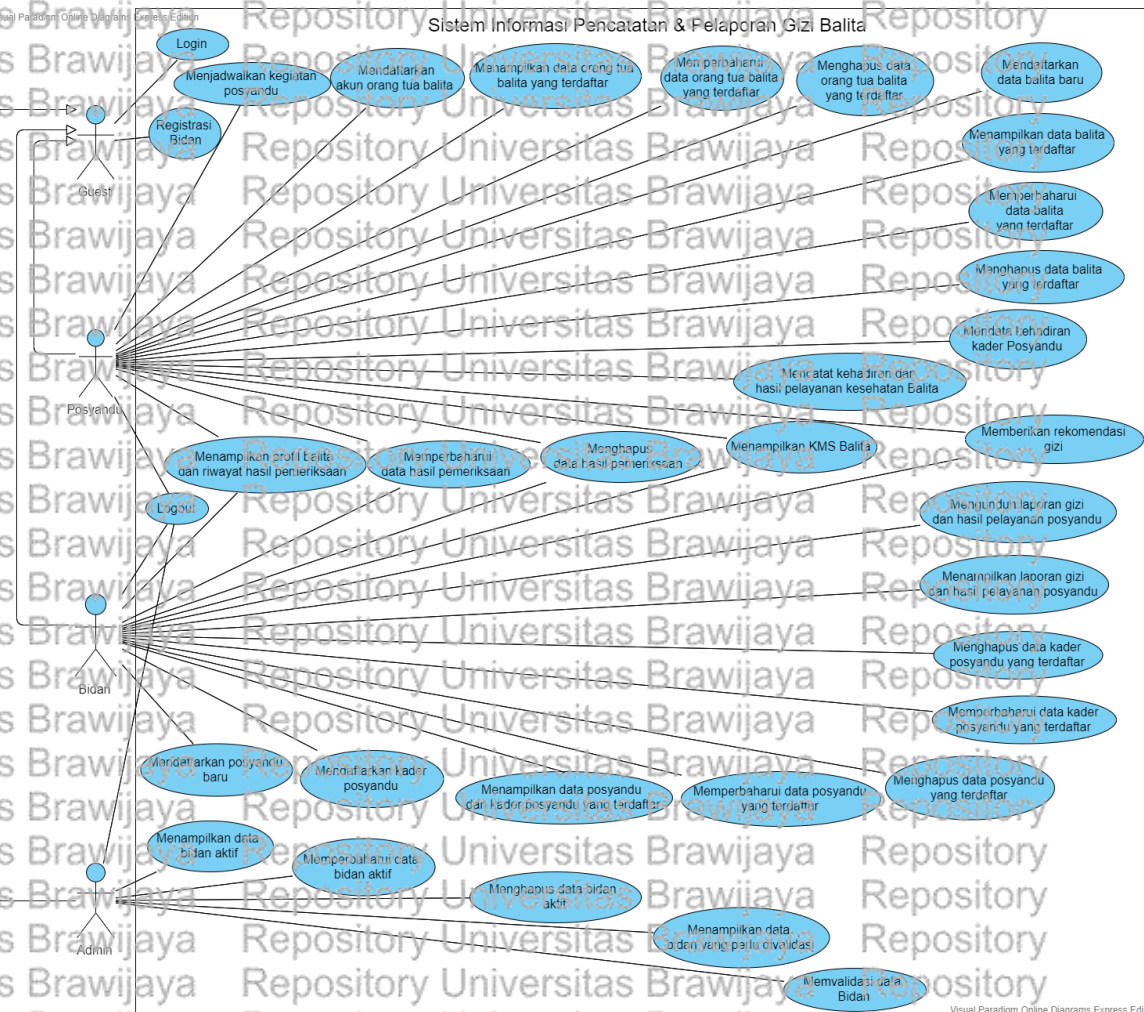


4.7 Pemodelan Kebutuhan

Pada bagian ini akan dijabarkan mengenai hasil pemodelan kebutuhan ke dalam bentuk *use case diagram* dan dilakukan penjelasan penggunaan setiap *use case* yang berada pada sistem ini dalam bentuk *use case scenario*.

4.7.1 Use Case Diagram

Use case diagram berfungsi untuk menggambarkan bagaimana pengguna akan berinteraksi dengan sistem. *Use case diagram* ini dibuat berdasarkan kebutuhan fungsional yang dibuat, dan terdapat 33 *use case* yang berasal dari 4 aktor pada sistem ini. Dalam Gambar 4.8 merupakan hasil pemodelan kebutuhan dalam bentuk *use case diagram*.



Gambar 4.8 Use case diagram



4.7.2 Use Case Scenario

Use case scenario merupakan penjelasan dari alur kegiatan dari perspektif pengguna dalam menggunakan sistem. *Use case scenario* ini terdiri dari beberapa bagian antara lain kode kebutuhan, *use case name*, aktor, *objective*, *pre-condition*, *main flow*, *alternative flow*, serta *post-condition* (Kurniawan, 2018). Pada Tabel 4.6 hingga Tabel 4.38 merupakan penjabaran 33 *use case scenario* yang ada pada sistem.

Tabel 4.6 Use case scenario login

Kode Kebutuhan	KE-GIZI-001
<i>Use Case Name</i>	<i>Login</i>
<i>Actor</i>	<i>Guest</i>
<i>Objective</i>	Aktor dapat masuk ke halaman dashboard <i>Admin</i> , <i>Bidan</i> , atau <i>Posyandu</i> sesuai dengan hak akses yang diberikan.
<i>Pre-condition</i>	Halaman <i>login</i> ditampilkan.
<i>Main flow</i>	<ol style="list-style-type: none"> 1. Aktor mengisi <i>username</i> dan <i>password</i> dengan lengkap. 2. Aktor menekan tombol "<i>login</i>". 3. Sistem mengautentikasi <i>username</i> dan <i>password</i>. 4. Sistem menampilkan halaman <i>dashboard</i> sesuai hak akses yang sesuai.
<i>Alternative flow</i>	<p>(i) Kesalahan pada pengisian <i>username</i> dan <i>password</i>:</p> <ol style="list-style-type: none"> 1. Jika aktor tidak mengisi kolom <i>username</i> atau <i>password</i>, maka sistem akan menampilkan peringatan untuk mengisi kolom secara lengkap. <p>(ii) Autentikasi Gagal:</p> <ol style="list-style-type: none"> 1. Jika autentikasi <i>username</i> dan <i>password</i> aktor salah, maka sistem akan menampilkan pesan bahwa <i>username</i> atau <i>password</i> salah. <p>(iii) Akun bidan belum diaktivasi:</p> <ol style="list-style-type: none"> 1. Jika akun bidan belum diaktivasi oleh <i>admin</i>, maka akan ditampilkan pesan bahwa akun perlu diaktivasi terlebih dahulu.
<i>Post-condition</i>	Halaman <i>dashboard</i> ditampilkan sesuai dengan hak akses yang diberikan kepada aktor.



Tabel 4.7 Use case scenario registrasi bidan

Kode Kebutuhan	KE-GIZI-002
Use Case Name	Registrasi Bidan
Actor	Guest
Objective	Aktor dapat membuat akun bidan baru pada sistem.
Pre-condition	Halaman registrasi bidan ditampilkan.
Main flow	<ol style="list-style-type: none"> 1. Aktor mengisi kolom nomor registrasi, nama, jenis kelamin, tempat lahir, tanggal lahir, <i>username</i>, <i>password</i>, dan konfirmasi <i>password</i> dengan lengkap. 2. Aktor menekan tombol unggah surat registrasi bidan. 3. Aktor memilih berkas surat registrasi bidan. 4. Aktor menekan tombol unggah foto diri bidan. 5. Aktor memilih foto diri bidan. 6. Aktor menekan tombol "registrasi". 7. Sistem menampilkan <i>pop-up</i> bahwa proses registrasi bidan telah berhasil dilakukan. 8. Sistem menampilkan halaman <i>login</i>.
Alternative flow	<p>(I) Kesalahan pada pengisian kolom pendaftaran bidan:</p> <ol style="list-style-type: none"> 1. Jika aktor tidak mengisi kolom isian registrasi bidan secara lengkap, maka sistem akan menampilkan peringatan untuk mengisi semua kolom. <p>(II) Kesalahan pada pengisian kolom <i>password</i> dan konfirmasi <i>password</i>:</p> <ol style="list-style-type: none"> 1. Jika isian kolom <i>password</i> dan konfirmasi <i>password</i> yang diisi oleh aktor tidak sesuai, maka sistem akan menampilkan pesan "Password tidak sama". <p>(III) Mengunggah <i>file</i> surat registrasi atau foto bidan dengan ukuran melebihi batas</p> <ol style="list-style-type: none"> 1. Jika <i>file</i> yang dipilih melebihi batas, maka sistem akan menampilkan notifikasi "File terlalu besar". <p>(IV) <i>Username</i> atau nomor registrasi ganda</p> <ol style="list-style-type: none"> 1. Jika <i>username</i> yang dimasukkan oleh aktor sudah ada pada basis data, maka sistem akan menampilkan pesan "Pendaftaran gagal, nomor registrasi / <i>username</i> sudah terdaftar".
Post-condition	Data akun bidan terdaftar dalam sistem.

Tabel 4.8 *Use case scenario* menampilkan data badan aktif

Kode Kebutuhan	KF-GIZI-003
<i>Use Case Name</i>	Menampilkan data badan aktif
<i>Actor</i>	<i>Admin</i>
<i>Objective</i>	Aktor dapat melihat data badan aktif.
<i>Pre-condition</i>	Halaman <i>dashboard</i> admin ditampilkan dan aktor terautentikasi sebagai <i>admin</i>
<i>Main flow</i>	<ol style="list-style-type: none"> Aktor memilih menu 'lihat data badan aktif' pada menu <i>dashboard admin</i>. Sistem menampilkan data badan yang aktif dalam bentuk tabel yang terdiri dari data nomor registrasi badan, nama badan, <i>username</i> badan, status badan, serta tombol detail data badan, <i>edit</i>, dan hapus.
<i>Alternative flow</i>	-
<i>Post-condition</i>	Sistem menampilkan data badan aktif.

Tabel 4.9 *Use case scenario* memperbaharui data badan aktif

Kode Kebutuhan	KF-GIZI-004
<i>Use Case Name</i>	Memperbaharui data badan aktif.
<i>Actor</i>	<i>Admin</i>
<i>Objective</i>	Aktor dapat memperbaharui data badan yang sudah aktif.
<i>Pre-condition</i>	Halaman data badan aktif ditampilkan dan aktor terautentikasi sebagai <i>admin</i> .
<i>Main flow</i>	<ol style="list-style-type: none"> Aktor menekan tombol "<i>edit</i>" pada salah satu data badan aktif yang mau diperbaharui datanya. Sistem menampilkan <i>form "edit"</i> data badan aktif yang terdapat kolom. Aktor melakukan perubahan data yang diperlukan Aktor menekan tombol "simpan". Sistem menyimpan hasil pembaharuan data badan aktif tersebut. Sistem menampilkan <i>pop-up</i> bahwa data badan aktif tersebut sudah berhasil diperbaharui.
<i>Alternative flow</i>	<p>(I) Kesalahan pada pengisian kolom pendaftaran badan:</p> <ol style="list-style-type: none"> Jika aktor mengosongkan salah satu kolom pada <i>form</i> maka sistem akan menampilkan peringatan untuk mengisi kolom tersebut. <p>(II) Kesalahan pada pengisian kolom <i>password</i> dan konfirmasi <i>password</i>:</p> <ol style="list-style-type: none"> Jika aktor ingin mengubah <i>password</i> akun badan dan <i>password</i> dan konfirmasi <i>password</i> yang diisi oleh aktor tidak sesuai, maka sistem akan menampilkan pesan "<i>Password tidak sama</i>".



	(III) Mengunggah <i>file</i> surat registrasi atau foto badan dengan ukuran melebihi batas: 1. Jika <i>file</i> yang dipilih melebihi batas, maka sistem akan menampilkan notifikasi " <i>File terlalu besar</i> ".
	(IV) <i>Username</i> atau nomor registrasi ganda: 1. Jika aktor melakukan pembaharuan data <i>username</i> atau nomor registrasi badan dengan data yang sudah ada pada basis data, maka sistem akan menampilkan pesan bahwa <i>edit</i> data gagal.
<i>Post-condition</i>	Data badan aktif yang dipilih diperbaharui.

Tabel 4.10 Use case scenario menghapus data badan aktif

Kode Kebutuhan	KF-GIZI-005
<i>Use Case Name</i>	Menghapus data badan aktif.
<i>Actor</i>	Admin
<i>Objective</i>	Aktor dapat menonaktifkan data badan yang sudah aktif.
<i>Pre-condition</i>	Halaman data badan aktif ditampilkan dan aktor terautentikasi sebagai <i>admin</i> .
<i>Main flow</i>	<ol style="list-style-type: none"> 1. Aktor menekan tombol "<i>hapus</i>" pada salah satu data badan aktif yang mau dinonaktifkan datanya. 2. Sistem menampilkan <i>pop-up</i> konfirmasi penonaktifan data badan. 3. Aktor mengonfirmasi penonaktifan data badan. 4. Sistem mengubah status badan tersebut menjadi "<i>Tidak Aktif</i>". 5. Sistem menampilkan <i>pop-up</i> penonaktifan badan berhasil.
<i>Alternative flow</i>	(I) Aktor membatalkan penghapusan data badan: <ol style="list-style-type: none"> 1. Aktor menekan tombol batal untuk membatalkan penonaktifan data badan. 2. Sistem menampilkan kembali halaman data badan aktif.
<i>Post-condition</i>	Status dari badan yang dipilih, diubah menjadi " <i>Tidak Aktif</i> ".



Tabel 4.11 Use case scenario menampilkan data bidang yang perlu divalidasi

Kode Kebutuhan	KF-GIZI-006
<i>Use Case Name</i>	Menampilkan data bidang yang perlu divalidasi.
<i>Actor</i>	<i>Admin</i>
<i>Objective</i>	Aktor dapat melihat data bidang yang perlu divalidasi.
<i>Pre-condition</i>	Halaman <i>dashboard admin</i> ditampilkan dan aktor terautentikasi sebagai <i>admin</i> .
<i>Main flow</i>	<ol style="list-style-type: none"> 1. Aktor memilih menu lihat data bidang yang perlu divalidasi pada menu <i>dashboard admin</i>. 2. Sistem menampilkan data bidang yang perlu divalidasi dalam bentuk tabel yang terdiri dari data nomor registrasi bidang, nama bidang, <i>username</i> bidang, status bidang, serta tombol detail data bidang.
<i>Alternative flow</i>	-
<i>Post-condition</i>	Data bidang yang perlu divalidasi ditampilkan.

Tabel 4.12 Use case scenario memvalidasi data bidang

Kode Kebutuhan	KF-GIZI-007
<i>Use Case Name</i>	Memvalidasi data bidang.
<i>Actor</i>	<i>Admin</i>
<i>Objective</i>	Aktor dapat memvalidasi data bidang.
<i>Pre-condition</i>	Halaman data bidang yang perlu divalidasi ditampilkan dan aktor terautentikasi sebagai <i>admin</i> .
<i>Main flow</i>	<ol style="list-style-type: none"> 1. Aktor menekan tombol detail pada salah satu data bidang yang perlu divalidasi. 2. Sistem menampilkan data bidang dalam bentuk <i>pop-up</i> beserta <i>button</i> untuk memvalidasi atau menolak data bidang tersebut. 3. Aktor menekan tombol "validasi" untuk memvalidasi dan mengaktifkan data bidang tersebut. 4. Sistem mengubah status bidang tersebut menjadi "Aktif". 5. Sistem menampilkan pesan data bidang berhasil divalidasi
<i>Alternative flow</i>	-
<i>Post-condition</i>	Data bidang tervalidasi dan status bidang berubah menjadi aktif.



Tabel 4.13 Use case scenario menjadwalkan hari Posyandu

Kode Kebutuhan	KE-GIZI-008
Use Case Name	Menjadwalkan hari Posyandu.
Actor	Posyandu
Objective	Aktor dapat menjadwalkan hari Posyandu.
Pre-condition	Halaman <i>dashboard</i> Posyandu ditampilkan dan aktor terautentikasi sebagai Posyandu.
Main flow	<ol style="list-style-type: none"> 1. Aktor memilih menu penjadwalan hari Posyandu pada menu <i>dashboard</i> Posyandu. 2. Sistem menampilkan <i>form</i> untuk melakukan penjadwalan hari Posyandu. 3. Aktor memasukkan hari, tanggal, waktu, tempat, dan deskripsi kegiatan Posyandu pada <i>form</i> yang ditampilkan. 4. Aktor menekan tombol "jadwalkan". 5. Sistem menyimpan hasil penjadwalan hari Posyandu ke dalam sistem dan mengirimkannya kepada akun orang tua balita yang terdaftar pada Posyandu tersebut. 6. Sistem menampilkan <i>pop-up</i> bahwa penjadwalan hari Posyandu berhasil dilakukan.
Alternative flow	<p>(I) Kesalahan pada pengisian <i>form</i> penjadwalan</p> <ol style="list-style-type: none"> 1. Jika aktor mengosongkan salah satu kolom pada <i>form</i> penjadwalan hari Posyandu, maka sistem akan menampilkan peringatan untuk mengisi kolom tersebut. <p>(II) Tanggal Posyandu ada di masa lampau</p> <ol style="list-style-type: none"> 1. Jika aktor mengisi tanggal Posyandu pada <i>form</i> dengan tanggal yang berada pada waktu sebelum tanggal penjadwalan hari Posyandu dilakukan, maka sistem akan menampilkan <i>pop up</i> bahwa penjadwalan gagal.
Post-condition	Jadwal hari Posyandu berikutnya tersimpan dan terkirim.



Tabel 4.14 Use case scenario mendaftarkan akun orang tua balita

Kode Kebutuhan	KE-GIZI-009
Use Case Name	Mendaftarkan akun orang tua balita.
Actor	Posyandu
Objective	Aktor dapat mendaftarkan akun baru untuk orang tua balita.
Pre-condition	Halaman <i>dashboard</i> Posyandu ditampilkan dan aktor terautentikasi sebagai Posyandu.
Main flow	<ol style="list-style-type: none"> Aktor memilih menu pendaftaran akun orang tua balita pada menu <i>dashboard</i> Posyandu. Sistem menampilkan <i>form</i> untuk melakukan pendaftaran akun orang tua balita. Aktor memasukkan nama ayah dan ibu, NIK ayah dan ibu, nomor kartu keluarga, alamat, nomor <i>handphone</i> <i>username</i>, <i>password</i>, dan konfirmasi <i>password</i>. Aktor menekan tombol daftar. Sistem menyimpan data pendaftaran akun orang tua balita. Sistem menampilkan <i>pop-up</i> bahwa pendaftaran akun orang tua balita berhasil dilakukan.
Alternative flow	<p>(I) Data <i>username</i>, NIK, No. KK, atau No. HP ganda:</p> <ol style="list-style-type: none"> Jika terdapat data yang dimasukkan ganda dengan yang ada pada basis data, maupun pada masukkan pada <i>form</i>, maka sistem akan menampilkan pesan pendaftaran orang tua balita gagal. <p>(II) Kesalahan pada pengisian kolom <i>password</i> dan konfirmasi <i>password</i>:</p> <ol style="list-style-type: none"> Jika isian kolom <i>password</i> dan konfirmasi <i>password</i> yang diisi oleh aktor tidak sesuai, maka sistem akan menampilkan pesan "Password tidak sama". <p>(III) Kesalahan pada pengisian <i>form</i> pendaftaran</p> <ol style="list-style-type: none"> Jika aktor mengosongkan salah satu kolom pendaftaran, maka sistem akan menampilkan pesan "Please fill in this field" pada kolom tersebut.
Post-condition	Akun orang tua balita tersimpan dalam sistem.

Tabel 4.15 *Use case scenario* menampilkan data orang tua balita yang terdaftar

Kode Kebutuhan	KF-GIZI-010
<i>Use Case Name</i>	Menampilkan data orang tua balita yang terdaftar
<i>Actor</i>	Posyandu
<i>Objective</i>	Aktor dapat melihat data orang tua balita yang terdaftar
<i>Pre-condition</i>	Halaman <i>dashboard</i> Posyandu ditampilkan dan aktor terautentikasi sebagai Posyandu.
<i>Main flow</i>	<ol style="list-style-type: none"> Aktor memilih menu lihat data orang tua balita pada menu <i>dashboard</i> Posyandu. Sistem menampilkan data orang tua balita yang telah terdaftar pada sistem dalam bentuk tabel yang terdiri dari data nama ayah, nama ibu, <i>username</i> orang tua, nomor HP, alamat, serta tombol <i>edit</i>, dan hapus data orang tua balita
<i>Alternative flow</i>	-
<i>Post-condition</i>	Data orang tua balita yang telah terdaftar pada sistem ditampilkan.

Tabel 4.16 *Use case scenario* memperbaharui data orang tua balita yang terdaftar

Kode Kebutuhan	KF-GIZI-011
<i>Use Case Name</i>	Memperbaharui data orang tua balita yang terdaftar.
<i>Actor</i>	Posyandu
<i>Objective</i>	Aktor dapat memperbaharui data orang tua balita yang terdaftar
<i>Pre-condition</i>	Halaman data orang tua balita ditampilkan dan aktor terautentikasi sebagai Posyandu.
<i>Main flow</i>	<ol style="list-style-type: none"> Aktor menekan tombol "<i>edit</i>" pada salah satu data orang tua balita yang mau diperbaharui datanya. Sistem menampilkan <i>form edit</i> data orang tua balita. Aktor melakukan perubahan data yang diperlukan. Aktor menekan tombol simpan. Sistem memperbaharui data orang tua balita tersebut. Sistem menampilkan <i>pop-up</i> bahwa data orang tua balita tersebut sudah berhasil diperbaharui.
<i>Alternative flow</i>	<p>(i) Data <i>username</i>, NIK, No. KK, atau No. HP ganda:</p> <ol style="list-style-type: none"> Jika aktor melakukan pembaharuan data terhadap data <i>username</i>, NIK, No. KK, atau No. HP dari orang tua balita dengan data yang sudah ada pada basis data, maka sistem akan menampilkan pesan bahwa <i>edit</i> data orang tua gagal. <p>(ii) Kesalahan pada pengisian <i>form</i> edit data orang tua.</p>



	<p>1. Jika aktor mengosongkan salah satu kolom pada saat memperbaharu data akun orang tua, maka sistem akan menampilkan pesan “Please fill out this field” pada kolom tersebut.</p> <p>(III) Kesalahan pada pengisian kolom <i>password</i> dan konfirmasi <i>password</i>:</p> <p>1. Jika aktor ingin mengubah <i>password</i> akun orang tua dan <i>password</i> dan konfirmasi <i>password</i> yang diisi oleh aktor tidak sesuai, maka sistem akan menampilkan pesan “Password tidak sama”.</p>
<i>Post-condition</i>	Data orang tua balita yang dipilih diperbaharui.

Tabel 4.17 Use case scenario menghapus data orang tua balita yang terdaftar

Kode Kebutuhan	KF-GIZI-012
<i>Use Case Name</i>	Menghapus data orang tua balita yang terdaftar.
<i>Actor</i>	Posyandu
<i>Objective</i>	Aktor dapat menonaktifkan orang tua balita yang terdaftar.
<i>Pre-condition</i>	Halaman data orang tua balita ditampilkan dan aktor terautentikasi sebagai Posyandu.
<i>Main flow</i>	<ol style="list-style-type: none"> Aktor menekan tombol “hapus” pada salah satu data orang tua balita yang mau dinonaktifkan datanya. Sistem menampilkan <i>pop-up</i> konfirmasi penonaktifan data orang tua balita. Aktor mengonfirmasi penonaktifan data orang tua balita. Sistem mengubah status orang tua balita tersebut menjadi “Tidak Aktif”. Sistem menampilkan <i>pop-up</i> konfirmasi penonaktifan data berhasil.
<i>Alternative flow</i>	<p>(I) Aktor membatalkan penghapusan data orang tua balita:</p> <ol style="list-style-type: none"> Aktor menekan tombol batal untuk membatalkan penonaktifan data orang tua balita. Sistem menampilkan kembali halaman data orang tua balita.
<i>Post-condition</i>	Orang tua balita yang dipilih diubah statusnya menjadi “Tidak Aktif”.

Tabel 4.18 Use case scenario mendaftarkan data balita baru

Kode Kebutuhan	KF-GIZI-013
<i>Use Case Name</i>	Mendaftarkan data balita baru.
<i>Actor</i>	Posyandu
<i>Objective</i>	Aktor dapat mendaftarkan data balita baru di Posyandu tersebut.
<i>Pre-condition</i>	Halaman <i>dashboard</i> Posyandu ditampilkan dan aktor terautentikasi sebagai Posyandu.
<i>Main flow</i>	<ol style="list-style-type: none"> Aktor memilih menu pendaftaran balita baru pada menu <i>dashboard</i> Posyandu.



	<ol style="list-style-type: none"> Sistem menampilkan <i>form</i> untuk melakukan pendaftaran data balita baru. Aktor memasukkan <i>username</i> akun orang tua, nama balita, jenis kelamin, tanggal lahir, berat badan lahir, NIK, dan nomor BPJS anak balita pada <i>form</i> yang ditampilkan. Aktor menekan tombol “daftar”. Sistem menyimpan data pendaftaran balita baru. Sistem menampilkan <i>pop-up</i> bahwa pendaftaran balita baru telah berhasil dilakukan.
<i>Alternative flow</i>	(l) Umur balita tidak sesuai: <ol style="list-style-type: none"> Jika umur anak balita yang didapatkan dari tanggal lahir yang dimasukkan pada <i>form</i> tidak sesuai (< 0 bulan atau > 60 bulan) maka sistem akan menampilkan pesan pendaftaran balita gagal.
<i>Post-condition</i>	Data balita baru tersimpan dalam sistem.

Tabel 4.19 Use case scenario menampilkan data balita yang terdaftar

Kode Kebutuhan	KF-GIZI-014
<i>Use Case Name</i>	Menampilkan data balita yang terdaftar.
<i>Actor</i>	Posyandu
<i>Objective</i>	Aktor dapat melihat data balita yang terdaftar di Posyandu tersebut.
<i>Pre-condition</i>	Halaman <i>dashboard</i> Posyandu ditampilkan dan aktor terautentikasi sebagai Posyandu.
<i>Main flow</i>	<ol style="list-style-type: none"> Aktor memilih menu lihat data balita pada menu <i>dashboard</i> Posyandu. Sistem menampilkan data balita yang telah terdaftar di Posyandu tersebut dalam bentuk tabel yang meliputi data nama anak, nama ayah, nama ibu, jenis kelamin, umur (bulan), serta tombol untuk melihat profil balita, <i>edit</i>, serta hapus data balita.
<i>Alternative flow</i>	-
<i>Post-condition</i>	Data balita yang telah terdaftar di Posyandu tersebut ditampilkan.

Tabel 4.20 Use case scenario memperbaharui data balita yang terdaftar

Kode Kebutuhan	KF-GIZI-015
<i>Use Case Name</i>	Memperbaharui data balita yang terdaftar
<i>Actor</i>	Posyandu
<i>Objective</i>	Aktor dapat memperbaharui data balita yang terdaftar
<i>Pre-condition</i>	Halaman data balita ditampilkan dan aktor terautentikasi sebagai Posyandu.
<i>Main flow</i>	<ol style="list-style-type: none"> Aktor menekan tombol “<i>edit</i>” pada salah satu data balita yang mau diperbaharui datanya.



	<ol style="list-style-type: none"> 2. Sistem menampilkan <i>form</i> "edit" data balita dengan kolom-kolom data balita yang sudah terisi dengan data balita yang dipilih. 3. Aktor melakukan perubahan data yang diperlukan. 4. Aktor menekan tombol "simpan". 5. Sistem memperbaharui data balita tersebut. 6. Sistem menampilkan <i>pop-up</i> bahwa data balita tersebut sudah berhasil diperbaharui.
<i>Alternative flow</i>	<p>(f) Umur balita tidak sesuai.</p> <ol style="list-style-type: none"> 1. Jika umur anak balita yang didapatkan dari tanggal lahir yang dimasukkan pada <i>form</i> tidak sesuai (< 0 bulan atau > 60 bulan) maka sistem akan menampilkan pesan <i>edit</i> data balita gagal.
<i>Post-condition</i>	Data balita yang dipilih diperbaharui.

Tabel 4.21 Use case scenario menghapus data balita yang terdaftar

Kode Kebutuhan	KF-GIZI-015
<i>Use Case Name</i>	Menghapus data balita yang terdaftar.
<i>Actor</i>	Posyandu
<i>Objective</i>	Aktor dapat menonaktifkan data balita yang terdaftar.
<i>Pre-condition</i>	Halaman data balita ditampilkan dan aktor terautentikasi sebagai Posyandu.
<i>Main flow</i>	<ol style="list-style-type: none"> 1. Aktor menekan tombol "hapus" pada salah satu data balita yang mau dinonaktifkan datanya. 2. Sistem menampilkan <i>pop-up</i> konfirmasi penonaktifan data balita. 3. Aktor mengonfirmasi penonaktifan data balita. 4. Sistem mengubah status dari balita tersebut menjadi "Tidak Aktif". 5. Sistem menampilkan <i>pop-up</i> konfirmasi penonaktifan data berhasil.
<i>Alternative flow</i>	<p>(f) Aktor membatalkan penghapusan data balita:</p> <ol style="list-style-type: none"> 1. Aktor menekan tombol batal untuk membatalkan penonaktifan data balita. 2. Sistem menampilkan kembali halaman data balita.
<i>Post-condition</i>	Data balita yang dipilih diubah statusnya menjadi "Tidak Aktif"

Tabel 4.22 *Use case scenario* mendata kehadiran kader Posyandu

Kode Kebutuhan	KE-GIZI-017
<i>Use Case Name</i>	Mendata kehadiran kader Posyandu.
<i>Actor</i>	Posyandu
<i>Objective</i>	Aktor dapat mendata kehadiran kader di Posyandu tersebut.
<i>Pre-condition</i>	Halaman <i>dashboard</i> Posyandu ditampilkan dan aktor terautentikasi sebagai Posyandu.
<i>Main flow</i>	<ol style="list-style-type: none"> Aktor memilih menu pendataan kehadiran kader Posyandu pada menu <i>dashboard</i> Posyandu. Aktor memasukkan tanggal dilaksanakannya Posyandu, dan nama kader yang hadir. Aktor menekan tombol simpan. Sistem menyimpan data kehadiran kader Posyandu tersebut dalam basis data. Sistem menampilkan <i>pop-up</i> bahwa pendataan kehadiran kader Posyandu telah berhasil dilakukan.
<i>Alternative flow</i>	<p>(I) Data kehadiran kader ganda:</p> <ol style="list-style-type: none"> Apabila kehadiran kader didata lebih dari satu kali untuk periode bulan dan tahun yang sama, maka sistem akan menampilkan pesan pendataan kehadiran kader gagal.
<i>Post-condition</i>	Kehadiran kader tersimpan dalam sistem.

Tabel 4.23 *Use case scenario* mencatat kehadiran dan hasil pelayanan kesehatan balita

Kode Kebutuhan	KE-GIZI-018
<i>Use Case Name</i>	Mencatat kehadiran dan hasil pelayanan kesehatan balita.
<i>Actor</i>	Posyandu
<i>Objective</i>	Aktor dapat mencatat kehadiran dan hasil pelayanan kesehatan balita di Posyandu tersebut.
<i>Pre-condition</i>	Halaman <i>dashboard</i> Posyandu ditampilkan dan aktor terautentikasi sebagai Posyandu.
<i>Main flow</i>	<ol style="list-style-type: none"> Aktor memilih menu pencatatan kehadiran dan hasil pelayanan kesehatan balita pada menu <i>dashboard</i> Posyandu. Sistem menampilkan <i>form</i> untuk melakukan pencatatan kehadiran dan hasil pelayanan kesehatan balita. Aktor mengisi data tanggal Posyandu, nama balita, hasil pengukuran dan pemeriksaan kesehatan balita yang meliputi berat badan, tinggi badan, lingkaran kepala, cara pengukuran, imunisasi, pemberian Vit. A, ASI eksklusif, PMT, keluhan maupun hasil diagnosa pada <i>form</i> tersebut. Aktor menekan tombol "Simpan".



	<p>5. Sistem menyimpan data kehadiran dan hasil pelayanan kesehatan balita di Posyandu tersebut dalam basis data.</p> <p>6. Sistem menampilkan <i>pop-up</i> bahwa pencatatan kehadiran dan hasil pelayanan kesehatan balita di Posyandu tersebut telah berhasil.</p>
<i>Alternative flow</i>	<p>(I) Data umur balita saat dilakukan pelayanan kesehatan tidak sesuai:</p> <p>1. Jika data umur saat dilakukan pemeriksaan kesehatan tidak sesuai (< 0 bulan, atau > 60 bulan) maka sistem akan menampilkan pesan pencatatan kesehatan balita gagal.</p> <p>(II) Data umur balita saat dilakukan pelayanan kesehatan ganda:</p> <p>1. Jika data umur saat dilakukan pemeriksaan kesehatan ganda dengan data yang ada pada basis data, maka sistem akan menampilkan pesan pencatatan kesehatan balita gagal.</p>
<i>Post-condition</i>	Data kehadiran dan hasil pelayanan kesehatan balita tersimpan dalam sistem.

Tabel 4.24 Use case scenario mendaftarkan Posyandu baru

Kode Kebutuhan	KF-GIZI-019
<i>Use Case Name</i>	Mendaftarkan Posyandu baru.
<i>Actor</i>	Bidan
<i>Objective</i>	Aktor dapat mendaftarkan Posyandu baru pada sistem.
<i>Pre-condition</i>	Halaman <i>dashboard</i> Bidan ditampilkan dan aktor terautentikasi sebagai Bidan.
<i>Main flow</i>	<ol style="list-style-type: none"> Aktor memilih menu pendaftaran Posyandu baru. Sistem menampilkan <i>form</i> untuk melakukan pendaftaran Posyandu baru. Aktor memasukkan nama, alamat, RT, RW, kelurahan / desa, <i>username</i>, <i>password</i> dan konfirmasi <i>password</i> Posyandu baru pada <i>form</i> yang ditampilkan. Aktor menekan tombol "unggah foto profil". Aktor memilih foto profil Posyandu tersebut. Aktor menekan tombol "daftar". Sistem menyimpan data Posyandu baru. Sistem menampilkan <i>pop-up</i> bahwa pendaftaran Posyandu baru telah berhasil dilakukan.
<i>Alternative flow</i>	<p>(I) Data <i>username</i> ganda:</p> <ol style="list-style-type: none"> Jika data <i>username</i> dari Posyandu yang didaftarkan ganda dengan ada yang pada basis data, maka sistem akan menampilkan pesan bahwa pendaftaran Posyandu baru gagal.
<i>Post-condition</i>	Data Posyandu baru tersimpan dalam sistem.

Tabel 4.25 *Use case scenario* mendaftarkan kader Posyandu

Kode Kebutuhan	KE-GIZI-020
<i>Use Case Name</i>	Mendaftarkan kader Posyandu.
<i>Actor</i>	Bidan
<i>Objective</i>	Aktor dapat mendaftarkan kader Posyandu baru pada sistem
<i>Pre-condition</i>	Halaman <i>dashboard</i> Bidan ditampilkan dan aktor terautentikasi sebagai Bidan.
<i>Main flow</i>	<ol style="list-style-type: none"> 1. Aktor memilih menu pendaftaran kader Posyandu. 2. Sistem menampilkan <i>form</i> untuk melakukan pendaftaran kader Posyandu 3. Aktor memasukkan nama kader, NIK, nama Posyandu, jenis kelamin, dan alamat kader. 4. Aktor menekan tombol unggah foto KTP kader Posyandu 5. Aktor memilih dan mengunggah foto KTP kader Posyandu. 6. Aktor menekan tombol daftar. 7. Sistem menyimpan data kader Posyandu tersebut. 8. Sistem menampilkan <i>pop-up</i> bahwa pendaftaran kader Posyandu telah berhasil dilakukan.
<i>Alternative flow</i>	<p>(I) Data NIK ganda</p> <ol style="list-style-type: none"> 1. Jika data NIK kader yang dimasukkan dalam <i>form</i> ganda dengan yang ada pada basis data, maka sistem akan menampilkan pesan pendaftaran kader baru gagal.
<i>Post-condition</i>	Data kader Posyandu baru tersimpan dalam sistem.



Tabel 4.26 Use case scenario menampilkan data Posyandu dan kader Posyandu yang terdaftar

Kode Kebutuhan	KF-GIZI-021
Use Case Name	Menampilkan data Posyandu dan kader Posyandu yang terdaftar.
Actor	Bidan
Objective	Aktor dapat melihat data Posyandu dan kader Posyandu yang telah terdaftar.
Pre-condition	Halaman <i>dashboard</i> bidan ditampilkan dan aktor terautentikasi sebagai bidan.
Main flow	<ol style="list-style-type: none"> 1. Aktor memilih menu kelola data Posyandu dan kader pada menu <i>dashboard</i> bidan. 2. Sistem menampilkan data Posyandu yang telah terdaftar sebelumnya dalam bentuk tabel.
Alternative flow	-
Post-condition	Data Posyandu dan kader Posyandu yang telah terdaftar ditampilkan.

Tabel 4.27 Use case scenario memperbaharui data Posyandu yang terdaftar

Kode Kebutuhan	KE-GIZI-022
Use Case Name	Memperbaharui data Posyandu yang terdaftar.
Actor	Bidan
Objective	Aktor dapat memperbaharui data Posyandu yang telah terdaftar.
Pre-condition	Halaman kelola data Posyandu dan kader ditampilkan dan aktor terautentikasi sebagai Bidan.
Main flow	<ol style="list-style-type: none"> 1. Aktor menekan tombol "<i>edit</i>" pada salah satu data Posyandu yang mau diperbaharui datanya. 2. Sistem menampilkan <i>form "edit"</i> data Posyandu dengan kolom-kolom data Posyandu yang sudah terisi dengan data Posyandu yang dipilih. 3. Aktor melakukan perubahan data yang diperlukan. 4. Aktor menekan tombol "<i>simpan</i>". 5. Sistem memperbaharui data Posyandu tersebut. 6. Sistem menampilkan <i>pop-up</i> bahwa data Posyandu tersebut sudah berhasil diperbaharui.
Alternative flow	<p>(I) Data <i>username</i> ganda.</p> <ol style="list-style-type: none"> 1. Jika aktor memperbaharui data <i>username</i> Posyandu dengan data yang sudah ada pada basis data, maka sistem akan menampilkan pesan bahwa pembaharuan data Posyandu gagal.
Post-condition	Data Posyandu yang dipilih diperbaharui.



Tabel 4.28 Use case scenario menghapus data Posyandu yang terdaftar

Kode Kebutuhan	KF-GIZI-023
Use Case Name	Menghapus data Posyandu yang terdaftar
Actor	Bidan
Objective	Aktor dapat menonaktifkan data Posyandu yang terdaftar.
Pre-condition	Halaman kelola data Posyandu dan kader ditampilkan dan aktor terautentikasi sebagai Bidan.
Main flow	<ol style="list-style-type: none"> 1. Aktor menekan tombol hapus pada salah satu data Posyandu yang mau dinonaktifkan datanya. 2. Sistem menampilkan <i>pop-up</i> konfirmasi penonaktifan data Posyandu. 3. Aktor mengonfirmasi penonaktifan data Posyandu. 4. Sistem mengubah status dari Posyandu tersebut menjadi "Tidak Aktif". 5. Sistem menampilkan <i>pop-up</i> konfirmasi penonaktifan data berhasil.
Alternative flow	<p>(I) Aktor membatalkan penghapusan data Posyandu:</p> <ol style="list-style-type: none"> 1. Aktor menekan tombol batal untuk membatalkan penonaktifan data Posyandu. 2. Sistem menampilkan kembali halaman kelola data Posyandu dan kader.
Post-condition	Posyandu yang dipilih diubah statusnya menjadi "Tidak Aktif".

Tabel 4.29 Use case scenario memperbaharui data kader Posyandu yang terdaftar

Kode Kebutuhan	KF-GIZI-024
Use Case Name	Memperbaharui data kader Posyandu yang terdaftar
Actor	Bidan
Objective	Aktor dapat memperbaharui data kader Posyandu yang telah terdaftar.
Pre-condition	Halaman kelola data Posyandu dan kader ditampilkan dan aktor terautentikasi sebagai bidan.
Main flow	<ol style="list-style-type: none"> 1. Aktor menekan tombol <i>edit</i> pada salah satu data kader Posyandu yang mau diperbaharui datanya. 2. Sistem menampilkan <i>form edit</i> data kader Posyandu dengan kolom-kolom data kader Posyandu yang sudah terisi dengan data kader Posyandu yang dipilih. 3. Aktor melakukan perubahan data yang diperlukan. 4. Aktor menekan tombol simpan. 5. Sistem memperbaharui data kader Posyandu tersebut.



	6. Sistem menampilkan <i>pop-up</i> bahwa data kader Posyandu tersebut sudah berhasil diperbaharui.
<i>Alternative flow</i>	(I) Data NIK ganda 1. Jika aktor memperbaharui data NIK kader dengan data NIK yang sudah ada pada basis data, maka sistem akan menampilkan pesan bahwa <i>edit</i> data kader gagal.
<i>Post-condition</i>	Data kader Posyandu yang dipilih diperbaharui

Tabel 4.30 Use case scenario menghapus data kader Posyandu yang terdaftar

Kode Kebutuhan	KF-GIZI-025
<i>Use Case Name</i>	Menghapus data kader Posyandu yang terdaftar.
<i>Actor</i>	Bidan
<i>Objective</i>	Aktor dapat menonaktifkan data kader Posyandu yang terdaftar.
<i>Pre-condition</i>	Halaman kelola data Posyandu dan kader ditampilkan dan aktor terautentikasi sebagai bidan.
<i>Main flow</i>	1. Aktor menekan tombol hapus pada salah satu data kader Posyandu yang mau dinonaktifkan datanya. 2. Sistem menampilkan <i>pop-up</i> konfirmasi penonaktifan data kader Posyandu. 3. Aktor mengonfirmasi penonaktifan data kader Posyandu. 4. Sistem mengubah status dari kader Posyandu tersebut menjadi "Tidak Aktif". 5. Sistem menampilkan <i>pop-up</i> konfirmasi penghapusan data berhasil.
<i>Alternative flow</i>	(I) Aktor membatalkan penghapusan data kader Posyandu. 1. Aktor menekan tombol batal untuk membatalkan penonaktifan data kader Posyandu. 2. Sistem menampilkan kembali halaman data kelola data Posyandu dan kader.
<i>Post-condition</i>	Kader Posyandu yang dipilih diubah statusnya menjadi "Tidak Aktif".



Tabel 4.31 Use case scenario menampilkan laporan gizi dan hasil pelayanan Posyandu

Kode Kebutuhan	KF-GIZI-026
Use Case Name	Menampilkan laporan gizi dan hasil pelayanan Posyandu.
Actor	Bidan
Objective	Aktor dapat melihat laporan gizi dan hasil pelayanan Posyandu.
Pre-condition	Halaman <i>dashboard</i> Bidan ditampilkan dan aktor terautentikasi sebagai Bidan.
Main flow	<ol style="list-style-type: none"> Aktor memilih menu lihat laporan gizi dan hasil pelayanan Posyandu. Sistem menampilkan laporan balita pertumbuhan normal, balita malnutrisi, kehadiran balita di Posyandu, serta kehadiran kader Posyandu, untuk semua Posyandu yang terdaftar dibawah bidan tersebut pada periode bulan dan tahun saat itu.
Alternative flow	<p>(I) Aktor menyeleksi data laporan gizi dan hasil pelayanan Posyandu:</p> <ol style="list-style-type: none"> Aktor melakukan seleksi data laporan gizi dan hasil pelayanan Posyandu berdasarkan nama Posyandu dan periode waktu laporan melalui <i>dropdown</i> nama Posyandu dan periode waktu laporan. Sistem menampilkan laporan balita pertumbuhan normal, balita malnutrisi, kehadiran balita di Posyandu, serta kehadiran kader Posyandu, untuk Posyandu dan periode waktu yang telah dipilih.
Post-condition	Laporan gizi dan hasil pelayanan Posyandu ditampilkan.

Tabel 4.32 Use case scenario mengunduh laporan gizi dan hasil pelayanan Posyandu

Kode Kebutuhan	KF-GIZI-027
Use Case Name	Mengunduh laporan gizi dan hasil pelayanan Posyandu.
Actor	Bidan
Objective	Aktor dapat mengunduh laporan gizi dan hasil pelayanan Posyandu ke dalam format Microsoft Excel.
Pre-condition	Halaman laporan gizi dan hasil pelayanan Posyandu ditampilkan dan aktor terautentikasi sebagai Bidan.
Main flow	<ol style="list-style-type: none"> Aktor menekan tombol unduh laporan normal Sistem mengunduh laporan pada tabel laporan balita pertumbuhan normal ke dalam format Microsoft Excel.
Alternative flow	<p>(I) Mengunduh laporan balita malnutrisi:</p> <ol style="list-style-type: none"> Aktor menekan tombol unduh laporan malnutrisi Sistem mengunduh laporan pada tabel laporan balita malnutrisi ke dalam format Microsoft Excel.
Post-condition	Laporan balita pertumbuhan normal atau balita malnutrisi dapat diunduh ke dalam format Microsoft Excel.



Tabel 4.33 Use case scenario menampilkan profil balita dan riwayat hasil pemeriksaan

Kode Kebutuhan	KF-GIZI-028
Use Case Name	Menampilkan profil balita dan riwayat hasil pemeriksaan.
Actor	Posyandu, Bidan
Objective	Aktor dapat melihat profil balita dan riwayat hasil pemeriksaan kesehatan dari balita yang dipilih.
Pre-condition	Halaman data balita ditampilkan atau halaman laporan gizi ditampilkan dan aktor terautentikasi sebagai Posyandu atau bidan.
Main flow	<ol style="list-style-type: none"> 1. Aktor menekan tombol "detail" pada salah satu data balita yang ingin dilihat profil dan riwayat hasil pemeriksaannya. 2. Sistem menampilkan data profil balita dan riwayat hasil pemeriksaan kesehatan yang pernah dilakukan oleh balita yang dipilih tersebut.
Alternative flow	-
Post-condition	Profil balita dan riwayat hasil pemeriksaan kesehatan balita ditampilkan.

Tabel 4.34 Use case scenario memperbaharui data hasil pemeriksaan

Kode Kebutuhan	KF-GIZI-029
Use Case Name	Memperbaharui data hasil pemeriksaan.
Actor	Posyandu, Bidan
Objective	Aktor dapat memperbaharui data hasil pemeriksaan kesehatan balita
Pre-condition	Halaman profil balita dan riwayat hasil pemeriksaan kesehatan balita ditampilkan dan aktor terautentikasi sebagai Posyandu atau bidan.
Main flow	<ol style="list-style-type: none"> 1. Aktor menekan tombol "edit" pada salah satu data hasil pemeriksaan kesehatan balita. 2. Sistem menampilkan form "edit" untuk memperbaharui data hasil pemeriksaan kesehatan balita. 3. Aktor melakukan perubahan data hasil pemeriksaan kesehatan balita yang diperlukan. 4. Aktor menekan tombol simpan. 5. Sistem memperbaharui data hasil pemeriksaan kesehatan balita tersebut. 6. Sistem menampilkan <i>pop-up</i> bahwa data hasil pemeriksaan kesehatan balita sudah berhasil diperbaharui.
Alternative flow	(I) Mengosongkan kolom berat badan / tinggi badan / lingkaran kepala / cara pengukuran
Post-condition	1. Jika aktor mengosongkan salah satu kolom tersebut saat melakukan pembaharuan data hasil pemeriksaan, maka sistem akan menampilkan peringatan untuk mengisi kolom tersebut. Data hasil pemeriksaan balita tersebut diperbaharui.

Tabel 4.35 *Use case scenario* menghapus data hasil pemeriksaan

Kode Kebutuhan	KE-GIZI-030
<i>Use Case Name</i>	Menghapus data hasil pemeriksaan.
<i>Actor</i>	Posyandu, Bidan
<i>Objective</i>	Aktor dapat menghapus data hasil pemeriksaan kesehatan balita.
<i>Pre-condition</i>	Halaman profil balita dan riwayat hasil pemeriksaan kesehatan balita ditampilkan dan aktor terautentikasi sebagai Posyandu atau bidan.
<i>Main flow</i>	<ol style="list-style-type: none"> 1. Aktor menekan tombol "hapus" pada salah satu data hasil pemeriksaan kesehatan balita yang mau dihapus datanya. 2. Sistem menampilkan <i>pop-up</i> konfirmasi penghapusan data hasil pemeriksaan kesehatan balita. 3. Aktor mengonfirmasi penghapusan data hasil pemeriksaan kesehatan balita pada bulan tersebut. 4. Sistem menghapus data hasil pemeriksaan kesehatan balita pada bulan tersebut dari basis data. 5. Sistem menampilkan <i>pop-up</i> konfirmasi penghapusan data berhasil.
<i>Alternative flow</i>	<p>(I) Aktor membatalkan penghapusan data hasil pemeriksaan kesehatan balita:</p> <ol style="list-style-type: none"> 1. Aktor menekan tombol batal untuk membatalkan penghapusan data hasil pemeriksaan kesehatan balita. 2. Sistem menampilkan kembali halaman profil dan riwayat hasil pemeriksaan kesehatan balita.
<i>Post-condition</i>	Data hasil pemeriksaan kesehatan balita pada bulan tersebut terhapus dari sistem.



Tabel 4.36 Use case scenario menampilkan KMS Balita

Kode Kebutuhan	KF-GIZI-031
Use Case Name	Menampilkan KMS balita
Actor	Posyandu, Bidan
Objective	Aktor dapat melihat KMS Balita.
Pre-condition	Halaman profil balita dan riwayat hasil pemeriksaan kesehatan balita ditampilkan dan aktor terautentikasi sebagai Posyandu atau bidan.
Main flow	<ol style="list-style-type: none"> 1. Aktor menekan tombol lihat KMS yang berada pada profil balita yang ditampilkan. 2. Sistem menampilkan KMS balita tersebut yang terdiri dari kurva pertumbuhan berat badan balita, garis indikator standar pertumbuhan berat badan balita, serta informasi hasil pengukuran setiap bulan dalam bentuk tabel.
Alternative flow	-
Post-condition	KMS Balita tersebut ditampilkan.

Tabel 4.37 Use case scenario memberikan rekomendasi gizi

Kode Kebutuhan	KF-GIZI-032
Use Case Name	Memberikan rekomendasi gizi.
Actor	Posyandu, Bidan
Objective	Aktor dapat memberikan rekomendasi atau saran gizi kepada orang tua balita untuk anak balitanya.
Pre-condition	Halaman profil balita dan riwayat hasil pemeriksaan kesehatan balita ditampilkan dan aktor terautentikasi sebagai Posyandu atau bidan.
Main flow	<ol style="list-style-type: none"> 1. Aktor menekan tombol beri saran gizi yang berada pada profil balita yang ditampilkan. 2. Sistem menampilkan form untuk memberikan rekomendasi atau saran gizi kepada orang tua balita. 3. Aktor mengisi rekomendasi atau saran gizi pada form yang ditampilkan. 4. Aktor menekan tombol "kirim". 5. Sistem menyimpan rekomendasi atau saran gizi tersebut pada basis data. 6. Sistem menampilkan pesan pemberian rekomendasi atau saran gizi berhasil.
Alternative flow	<p>(I) kolom saran gizi kosong:</p> <ol style="list-style-type: none"> 1. Jika aktor mengosongkan kolom saran gizi dan menekan tombol "Kirim", maka sistem akan menampilkan pesan peringatan untuk mengisi kolom saran gizi tersebut.
Post-condition	Rekomendasi atau saran gizi disimpan dan terkirim pada akun orang tua balita tersebut.



Tabel 4.38 Use case scenario Logout

Kode Kebutuhan	KE-GIZI-033
Use Case Name	Logout
Actor	Admin, Posyandu, Bidan
Objective	Aktor dapat keluar dari sistem
Pre-condition	Aktor sudah terautentikasi dan berada di dalam sistem.
Main flow	<ol style="list-style-type: none"> 1. Aktor menekan <i>dropdown</i> yang berada pada bagian <i>navbar</i> sistem dengan tulisan "Hai, (nama user)" 2. Aktor menekan tombol "Keluar". 3. Sistem menampilkan <i>pop-up</i> konfirmasi untuk melakukan <i>logout</i>. 4. Aktor menekan tombol "Keluar". 5. Sistem menghentikan sesi dan menampilkan halaman <i>login</i>.
Alternative flow	<p>(I) Membatalkan <i>logout</i>:</p> <ol style="list-style-type: none"> 1. Aktor menekan tombol batal untuk membatalkan <i>logout</i> dari sistem. 2. Sistem menampilkan kembali halaman saat aktor menekan tombol <i>logout</i> tersebut.
Post-condition	Sistem akan menghentikan sesi dan menampilkan halaman <i>login</i> .



BAB 5 PERANCANGAN DAN IMPLEMENTASI

Pada bab ini, akan dibahas mengenai hasil perancangan dan implementasi hasil rancangan tersebut ke dalam bahasa pemrograman untuk pembuatan sistem informasi berbasis *web* ini. Pada proses perancangan sistem ini terdiri dari beberapa bagian, yaitu perancangan arsitektur, perancangan data, perancangan komponen, serta perancangan antarmuka, sedangkan pada bagian implementasi sistem, terdapat implementasi kode program, implementasi basis data, dan implementasi antarmuka.

5.1 Perancangan Arsitektur

Perancangan arsitektur merupakan proses perancangan yang didalamnya termasuk merancang struktur secara keseluruhan dari sebuah sistem untuk memberikan gambaran konseptual dari komponen-komponen atau modul sistem yang akan dibangun dan bagaimana interaksi antar komponen tersebut (Pressman, 2010). Pada perancangan arsitektur sistem informasi yang akan dikembangkan ini, akan menggunakan *sequence diagram* dan *class diagram*.

5.1.1 *Sequence Diagram* Pencatatan Kehadiran dan Hasil Pelayanan Kesehatan Balita

Dalam Gambar 5.1 merupakan *sequence diagram* untuk proses pencatatan kehadiran dan hasil pelayanan kesehatan balita. Pada *sequence diagram* ini, terdapat beberapa objek yang berperan, antara lain aktor "Posyandu", *boundary* "Halaman *Dashboard*" yang berperan untuk aktor "Posyandu" dapat memilih menu pencatatan kehadiran dan hasil pelayanan kesehatan balita, *boundary* "Modal Pencatatan Kesehatan Balita" yang berperan untuk menerima masukan dari aktor terkait data hasil pemeriksaan kesehatan balita, dan *boundary* "Modal Notifikasi" yang berperan untuk memberikan informasi kepada aktor apakah proses pencatatan telah berhasil dilakukan atau tidak.

Untuk objek *controller* terdapat *controller* "*Dashboard*" yang berperan untuk melakukan *redirect* pada halaman *dashboard* aktor "Posyandu", dan *controller* "Balita" yang berperan untuk mengeksekusi fungsi "*catatKesehatanBalita()*". Untuk objek *model* terdapat "Balita_Model" yang berperan untuk melakukan *query* terkait data balita, dan *model* "Laporan_Kesehatan_Model" yang berperan untuk melakukan *query* untuk menentukan apakah data hasil pemeriksaan kesehatan yang dimasukkan oleh *user* tidak ganda dengan yang ada pada basis data.

Pada *sequence diagram* ini, terdapat kondisi alternatif yang bisa terjadi, dimana setelah melakukan pemanggilan fungsi "*cekUmurGanda(ID_balita, umurPeriksa)*" pada laporan kesehatan model, apabila fungsi tersebut mengembalikan nilai 0, maka proses pencatatan hasil pemeriksaan kesehatan balita akan dilanjutkan dengan melakukan perhitungan status gizi dan melakukan *insert* data pada basis data, lalu melakukan *redirect* pada fungsi "*index*" pada *controller dashboard*



dengan membawa pesan sukses dan menampilkan modal notifikasi bahwa proses pencatatan kesehatan balita telah berhasil. Sedangkan apabila fungsi tersebut mengembalikan nilai 1, atau artinya terdapat data umur periksa yang ganda pada basis data, maka sistem akan langsung melakukan *redirect* pada fungsi "index" pada *controller* "*Dashboard*" dengan membawa pesan gagal dan menampilkan modal notifikasi bahwa proses pencatatan kesehatan balita telah gagal.



Gambar 5.1 Sequence diagram pencatatan kehadiran dan hasil pelayanan kesehatan balita

5.1.2 Sequence Diagram Menampilkan KMS Balita

Dalam Gambar 5.2 merupakan *sequence diagram* untuk menampilkan KMS balita. Pada *sequence diagram* ini terdapat beberapa objek yang berperan, diantaranya terdapat aktor "Posyandu", lalu untuk objek *boundary* terdapat *boundary* "Halaman Profil Balita" yang berperan untuk menampilkan profil balita yang dipilih, dan menyediakan *button* untuk aktor dapat melihat KMS dari balita tersebut, serta *boundary* "Halaman KMS Balita" yang berperan untuk menampilkan grafik pertumbuhan berat badan balita dan riwayat pemeriksaan balita tersebut.

Untuk objek *controller*, terdapat *controller* "Balita" yang berperan untuk mengeksekusi fungsi "lihatKMS(ID_balita)". Untuk objek *model* terdapat "Laporan_Kesehatan_Model" yang berperan untuk melakukan *query* terkait laporan gizi dari balita yang dipilih, serta "Balita_Model" yang berperan untuk melakukan *query* terkait data profil daripada balita tersebut.



Gambar 5.2 Sequence diagram menampilkan KMS balita

5.1.3 Sequence Diagram Menampilkan Laporan Gizi

Dalam Gambar 5.3 merupakan *sequence diagram* untuk menampilkan laporan gizi. Pada *sequence diagram* ini terdapat beberapa objek yang berperan, diantaranya terdapat aktor "Bidan", lalu untuk objek *boundary* terdapat "Halaman Dashboard" yang berperan untuk aktor dapat memilih menu untuk melihat laporan gizi, serta *boundary* "Halaman Laporan Gizi" yang berperan untuk menampilkan data laporan kepada aktor. Untuk objek *controller* sendiri terdapat *controller* "Dashboard" yang berperan untuk mengeksekusi fungsi "lihatLaporan()". Untuk objek *model* sendiri, terdapat *model* "Laporan_Kesehatan_Model" yang berperan untuk melakukan *query* untuk mengambil data laporan balita dengan pertumbuhan normal, laporan balita malnutrisi, jumlah balita yang melakukan pemeriksaan, serta tanggal laporan yang tersedia pada basis data. Untuk "Kader_Model", berperan untuk melakukan *query* untuk mengambil data jumlah kader yang terdaftar. Untuk "Laporan_Kehadiran_Kader_Model", berperan untuk melakukan *query* untuk mendapatkan data jumlah kader yang hadir pada saat Posyandu dilaksanakan. Lalu untuk "Balita_Model", berperan untuk melakukan *query* untuk mendapatkan data jumlah balita yang terdaftar pada Posyandu. Serta yang terakhir adalah "Posyandu_Model" yang berperan untuk melakukan *query* untuk mendapatkan data Posyandu.

Pada *sequence diagram* ini, terdapat kondisi alternatif yang bisa dilakukan oleh aktor yaitu melakukan seleksi sumber data laporan berdasarkan Posyandu, bulan, dan tahun laporan. Saat pertama kali mengakses fungsi "lihatLaporan()", maka sumber data laporan akan secara *default* menampilkan data laporan untuk semua Posyandu yang berada dibawah naungan aktor bidan tersebut pada bulan dan tahun saat fungsi tersebut diakses. Sedangkan apabila aktor melakukan seleksi sumber data laporan, maka sistem akan memanggil fungsi "lihatLaporan()" kembali yang berada pada *controller* Dashboard, namun kali ini sumber data pada *query* nya akan menyesuaikan dengan kondisi seleksi sumber data laporan yang diberikan oleh aktor.



Gambar 5.3 Sequence diagram menampilkan laporan gizi

Pada bagian ini akan dijabarkan mengenai hasil perancangan *class diagram controller*, *class diagram model*, dan unifikasi atau penggabungan *class diagram controller* dan *model*.

5.1.4 Class Diagram Controller

Dalam Gambar 5.4 merupakan *class diagram controller* yang ada pada sistem informasi yang akan dibuat, dimana terdapat 7 *controller* yang terdiri dari *class* "Dashboard", "Ortu", "User", "Bidan", "Posyandu", "Kader", serta "Balita". Semua *class controller* tersebut memiliki hubungan generalisasi dengan kelas *CI_Controller*.



Visual Paradigm Online Diagrams Express Edition

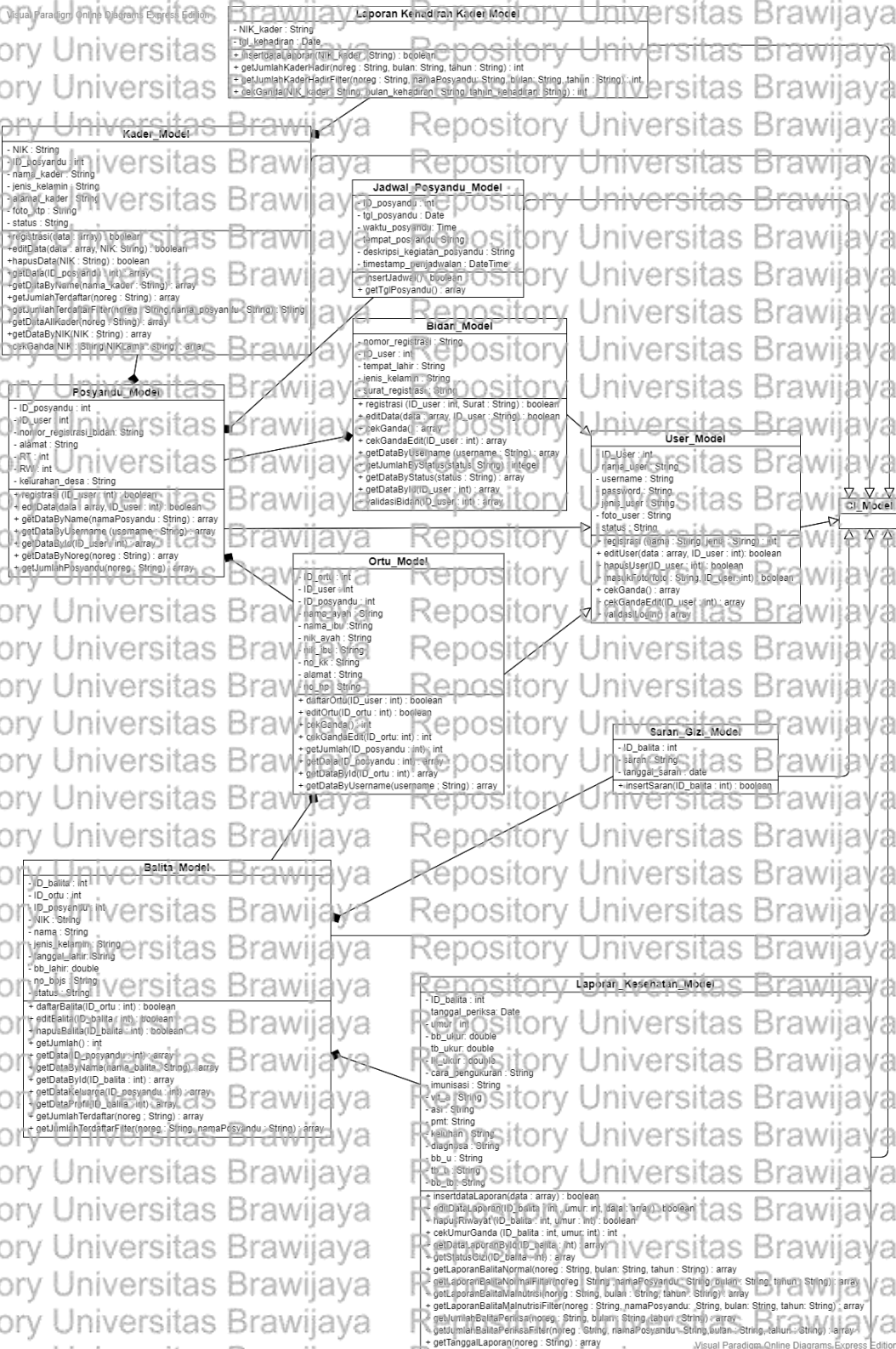


Visual Paradigm Online Diagrams Express Edition

Gambar 5.4 Class diagram controller

5.1.5 Class Diagram Model

Dalam Gambar 5.5 merupakan *class diagram model* yang ada pada sistem informasi yang akan dibuat, dimana terdapat 10 kelas *model*. Masing-masing dari kelas yang ada akan berperan untuk mengambil, memasukkan, maupun melakukan manipulasi data yang berkaitan sesuai dengan nama masing-masing *class* tersebut, seperti *User_Model* merupakan kelas yang berperan untuk mengambil maupun memanipulasi data yang berkaitan dengan data user. Untuk hubungan antar kelas, terdapat hubungan generalisasi antara kelas *CI_Model* dengan semua kelas yang ada kecuali untuk kelas *Posyandu_model*, *Bidan_Model*, dan *Ortu_Model*, dimana kelas tersebut memiliki hubungan generalisasi dengan kelas *User_Model*. Lalu terdapat hubungan komposisi diantara *Bidan_Model* dengan *Posyandu_Model*, *Posyandu_Model* dengan *Ortu_Model*, *Jadwal_Posyandu_Model*, serta *Kader_Model*, lalu diantara *Kader_Model* dengan *Laporan_Kehadiran_Kader_Model*, lalu diantara *Ortu_Model* dengan *Balita_Model*, dan terakhir diantara *Balita_Model* dengan *Saran_Gizi_Model* dan *Laporan_Kesehatan_Model*.



Gambar 5.5 Class diagram model

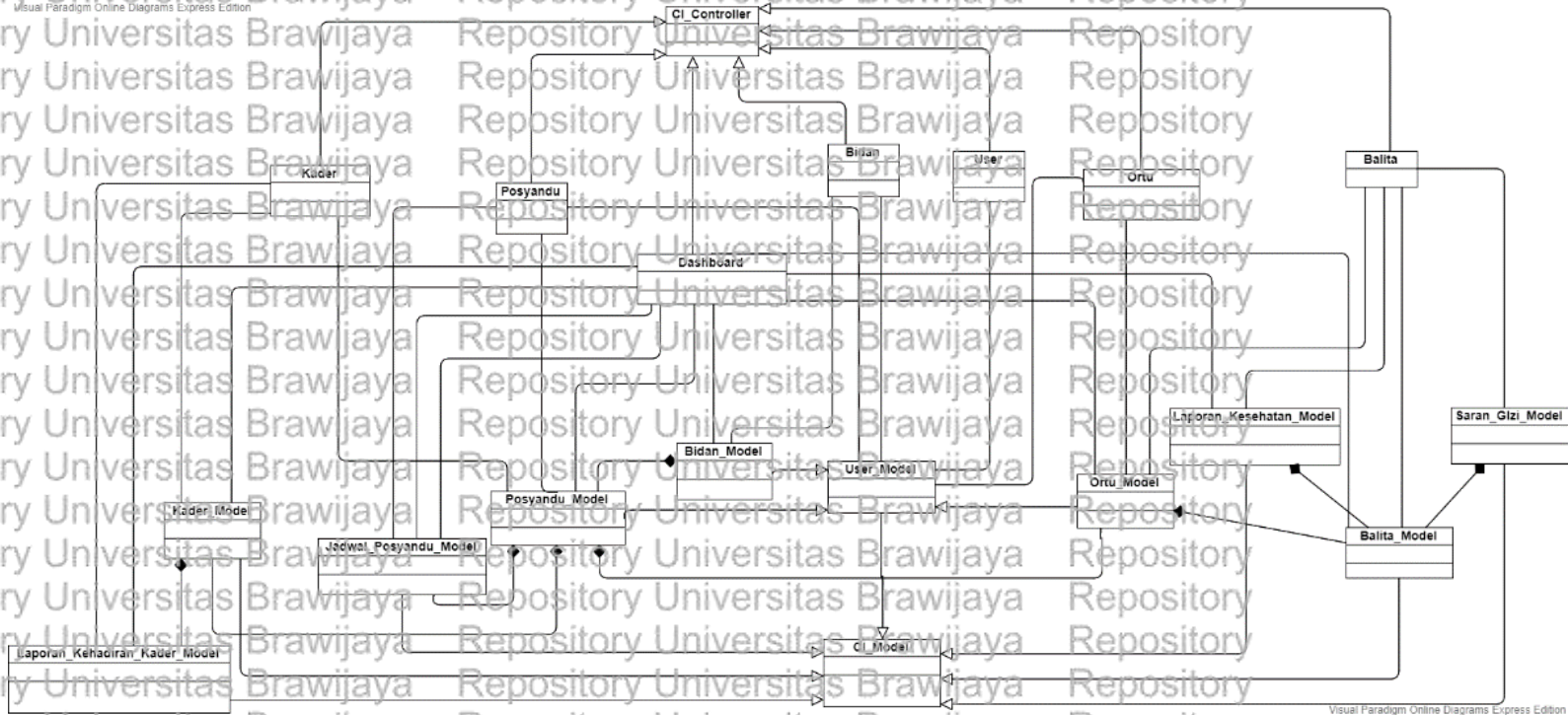


5.1.6 Unifikasi *Class Diagram*

Dalam Gambar 5.6 merupakan hasil unifikasi atau penggabungan antara *class diagram controller* dengan *class diagram model* yang berada dalam Gambar 5.4 dan Gambar 5.5. Pada unifikasi *class diagram* ini, setiap kelas dalam *class controller* dapat berhubungan atau mengakses beberapa *class model* yang ada pada sistem seperti yang ada pada kelas *controller* Posyandu yang memiliki hubungan dengan kelas *Posyandu_Model*, *Jadwal_Posyandu_Model*, serta *User_Model*.



Visual Paradigm Online Diagrams Express Edition



Visual Paradigm Online Diagrams Express Edition

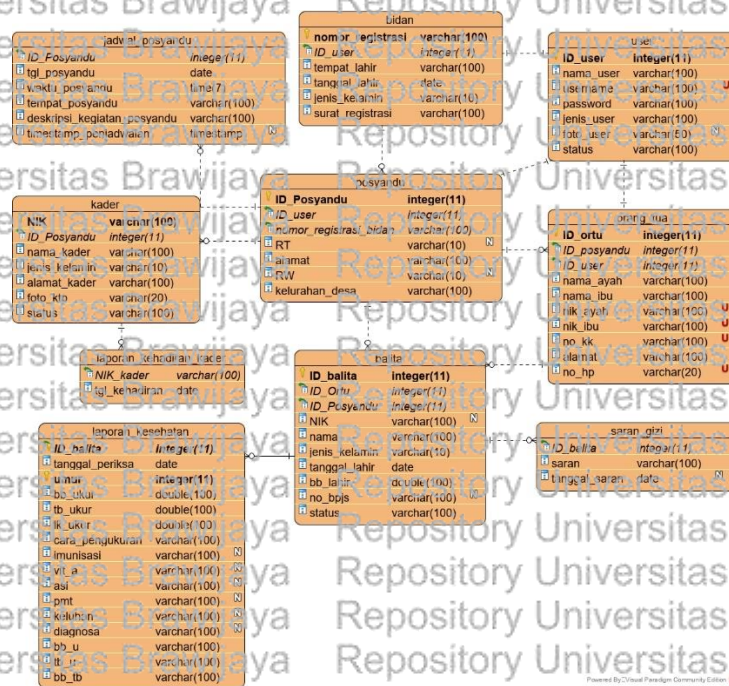
Gambar 5.6 Unifikasi *class diagram*



5.2 Perancangan Data

Untuk perancangan data akan menggunakan *physical data model* untuk mendefinisikan struktur tabel seperti nama kolom, tipe data, *primary key*, dan *foreign key*, maupun relasi antar tabel pada basis data yang akan diimplementasikan dalam sistem informasi ini. *Physical data model* untuk sistem informasi ini terdiri dari 10 tabel yang terdiri dari tabel “user”, yang berperan untuk menyimpan data yang berkaitan dengan user dari sistem informasi ini, tabel “bidan” yang berperan untuk menyimpan data yang berkaitan dengan bidan yang terdaftar. Tabel “orang_tua” yang berperan untuk menyimpan data yang berkaitan dengan orang tua balita yang terdaftar dalam sistem. Tabel “Posyandu” yang berperan untuk menyimpan data yang berkaitan dengan suatu Posyandu yang terdaftar dalam sistem.

Tabel “kader” yang berperan untuk menyimpan data yang berkaitan dengan kader Posyandu yang terdaftar dalam sistem. Tabel “laporan_kehadiran_kader” yang berperan untuk menyimpan data yang berkaitan mengenai laporan kehadiran seorang kader Posyandu. Tabel “jadwal_Posyandu” yang berperan untuk menyimpan data yang berkaitan dengan penjadwalan Posyandu. Tabel “balita” yang berperan untuk menyimpan data yang berkaitan dengan balita yang terdaftar dalam sistem. Tabel “laporan_kesehatan” yang berperan untuk menyimpan data yang berkaitan dengan hasil pencatatan kesehatan balita, dan tabel “saran_gizi” yang berperan untuk menyimpan data yang berkaitan dengan saran gizi yang diberikan kepada seorang balita. Rancangan *physical data model* dapat dilihat dalam Gambar 5.7.



Gambar 5.7 Physical data model



5.3 Perancangan Komponen

Sebuah komponen merupakan sebuah elemen terkecil dari sebuah sistem yang didalamnya terdapat alur logika maupun struktur data yang dibutuhkan untuk mengimplementasikan alur logika tersebut (Pressman, 2010). Perancangan komponen ini dilakukan untuk dapat mendefinisikan alur logika dari komponen-komponen utama penyusun sistem informasi yang akan dibangun ini. Untuk bagian perancangan komponen ini, akan dijabarkan *pseudocode* dari tiga komponen utama sistem yaitu fungsi Pencatatan Kesehatan Balita, Menampilkan Kartu Menuju Sehat (KMS) Balita, dan Menampilkan Laporan Gizi.

5.3.1 Perancangan Komponen Fungsi Pencatatan Kesehatan Balita

Pada Tabel 5.1 merupakan hasil perancangan komponen untuk melakukan Pencatatan Kesehatan Balita, dengan nama fungsi `catatKesehatanBalita()`

Tabel 5.1 Pseudocode fungsi `catatKesehatanBalita()`

Fungsi <code>catatKesehatanBalita()</code>	
1	START
2	
3	Inisialisasi variabel <code>post</code> = hasil semua input POST
4	Inisialisasi variabel <code>nama_balita</code> = nilai variabel POST pada index
5	' <code>nama_balita_kesehatan</code> '
6	Inisialisasi variabel <code>dataBalita</code> = hasil kembalian dari
7	pemanggilan <code>method getDataByName (nama_balita)</code>
8	Inisialisasi variabel <code>tgl_periksa</code> = nilai variabel POST pada index
9	' <code>tgl_pemeriksaan_kesehatan</code> '
10	Inisialisasi variabel <code>umurPeriksa</code> = pembulatan nilai dari variabel
11	(<code>tgl_periksa - dataBalita</code> pada index ke 0 dan index
12	' <code>tanggal_lahir</code> ') / 2629746
13	Inisialisasi variabel <code>cara_pengukuran</code> = nilai variabel POST pada
14	index ' <code>cara_pengukuran</code> '
15	Inisialisasi variabel <code>ID_balita</code> = nilai variabel <code>dataBalita</code> pada
16	index ' <code>ID_balita</code> '
17	Inisialisasi variabel <code>cekUmur</code> = hasil kembalian dari pemanggilan
18	<code>method cekUmurGanda (ID_balita, umurPeriksa)</code>
19	
20	IF (<code>umurPeriksa >= 0 AND umurPeriksa <= 60 AND cekUmur == 0</code>)
21	THEN
22	Inisialisasi variabel <code>BB_ukur</code> = nilai variabel POST
23	pada index ' <code>bb_ukur</code> '
24	Inisialisasi variabel <code>tb_ukur</code> = nilai variabel POST
25	pada index ' <code>tb_ukur</code> '
26	Inisialisasi variabel <code>jk</code> = nilai variabel POST pada
27	index ' <code>jenis_kelamin</code> '
28	IF (<code>umurPeriksa <= 24 AND caraPengukuran == "Berdiri"</code>)
29	THEN
30	variabel <code>tb_ukur</code> = <code>tb_ukur + 0.7</code>
31	ENDIF
32	IF (<code>umurPeriksa > 24 AND caraPengukuran == "Telentang"</code>)
33	THEN
34	variabel <code>tb_ukur</code> = <code>tb_ukur - 0.7</code>
35	ENDIF
36	Inisialisasi variabel <code>statusGizi</code> = hasil kembalian
37	dari pemanggilan <code>method statusGizi (jk, umurPeriksa, bb_ukur,</code>
38	<code>tb_ukur, cara_pengukuran)</code>
39	Inisialisasi variabel <code>data</code> bertipe <code>data array</code>



```

30      nilai index 'ID_balita' = nilai variabel ID_balita
31      nilai index 'umur' = nilai variabel umurPeriksa
32      nilai index 'tb_ukur' = nilai variabel tb_ukur
33      nilai index 'bb_u' = nilai variabel statusGizi pada
      index 'bb_u'
34      nilai index 'tb_u' = nilai variabel statusGizi pada
      index 'tb_u'
35      nilai index 'bb_tb' = nilai variabel statusGizi pada
      index 'bb_tb'
36      Inisialisasi variabel status = hasil kembalian dari
      pemanggilan method insertDataLaporan
37
38      IF (status = TRUE)
39      THEN
40          redirect menuju halaman Dashboard
      dengan query string variabel status = berhasilCatatKesehatan
41      ELSE
42      THEN
43          redirect menuju halaman Dashboard
      dengan query string variabel status = gagalCatatKesehatan
44      ENDIF
45
46      ELSE
47      THEN
48          redirect menuju halaman Dashboard dengan query
      string variabel status = gagalCatatKesehatan
49      ENDIF
50      END
  
```

5.3.2 Perancangan Komponen Fungsi Menampilkan Kartu Menuju Sehat (KMS) Balita

Pada Tabel 5.2 merupakan hasil perancangan komponen untuk fungsi Menampilkan Kartu Menuju Sehat (KMS) Balita, dengan nama fungsi lihatKMS(ID_balita).

Tabel 5.2 Pseudocode fungsi lihatKMS(ID_Balita)

```

Fungsi lihatKMS(ID_balita)
1  START
2  Inisialisasi variabel dataLaporan = hasil kembalian dari
      pemanggilan method getDataLaporanById(ID_balita)
3  Inisialisasi variabel dataBalita = hasil kembalian dari
      pemanggilan method getDataById(ID_balita)
4  Inisialisasi variabel jk = nilai variabel dataBalita pada index
      jenis_kelamin
5  Inisialisasi variabel
      bulan, minus3SD, minus2SD, minus1SD, medianSD, plus1SD, plus2SD, plus3SD,
      dan data_bb dengan nilai ''
6  Inisialisasi variabel kbm bertipe data array =
      0,0.8,0.9,0.8,0.6,0.5,0.4,0.3,0.2
7  Inisialisasi variabel bb_timbang bertipe data array = nilai 0
      sebanyak 60
8
9
10 IF (jk = "Laki-Laki")
11 THEN
12     inisialisasi variabel minusTigaSD bertipe data array
      = nilai standar - 3 SD berat badan anak balita laki-laki
13     inisialisasi variabel minusDuaSD bertipe data array
      = nilai standar - 2 SD berat badan anak balita laki-laki
  
```



```

14      Inisialisasi variabel minusSatuSD bertipe data array
      = nilai standar - 1 SD berat badan anak balita laki-laki
15      Inisialisasi variabel median bertipe data array =
      nilai standar median berat badan anak balita laki-laki
16      Inisialisasi variabel plusSatuSD bertipe data array
      = nilai standar + 1 SD berat badan anak balita laki-laki
17      Inisialisasi variabel plusDuaSD bertipe data array
      = nilai standar + 2 SD berat badan anak balita laki-laki
18      Inisialisasi variabel plusTigaSD bertipe data array
      = nilai standar + 3 SD berat badan anak balita laki-laki
19
20      ELSE
21      THEN
22      Inisialisasi variabel minusTigaSD bertipe data array
23      = nilai standar - 3 SD berat badan anak balita perempuan
24      Inisialisasi variabel minusDuaSD bertipe data array
      = nilai standar - 2 SD berat badan anak balita perempuan
25      Inisialisasi variabel minusSatuSD bertipe data array
      = nilai standar - 1 SD berat badan anak balita perempuan
26      Inisialisasi variabel median bertipe data array =
      nilai standar median berat badan anak balita perempuan
27      Inisialisasi variabel plusSatuSD bertipe data array
      = nilai standar + 1 SD berat badan anak balita perempuan
28      Inisialisasi variabel plusDuaSD bertipe data array =
      nilai standar + 2 SD berat badan anak balita perempuan
29      Inisialisasi variabel plusTigaSD bertipe data array
      = nilai standar + 3 SD berat badan anak balita perempuan
30      ENDIF
31      Inisialisasi variabel umurMax = 0
32      Inisialisasi variabel a = 0
33      DOWHILE a <= 60
34      IF (terdapat nilai pada variabel dataLaporan pada index
35      ke a pada index 'umur'
36      THEN
37      variabel umurMax = dataLaporan pada index ke a pada
38      index 'umur'
39      ENDDOWHILE
40      Inisialisasi variabel i = 0
41      DOWHILE i <= 60
42      variabel bulan = nilai variabel bulan + ',' + nilai
43      variabel i + ' '
44      variabel minus3SD = nilai variabel minus3SD + ',' +
      nilai variabel minus3SD pada index ke i + ' '
45      variabel minus2SD = nilai variabel minus2SD + ',' +
      nilai variabel minus2SD pada index ke i + ' '
46      variabel minus1SD = nilai variabel minus1SD + ',' +
      nilai variabel minus1SD pada index ke i + ' '
      variabel medianSD = nilai variabel medianSD + ',' +
      nilai variabel medianSD pada index ke i + ' '

```



```

47 variabel plus1SD = nilai variabel plus1SD + '"' +
nilai variabel plus1SD pada index ke i + '"'
48 variabel plus2SD = nilai variabel plus2SD + '"' +
nilai variabel plus2SD pada index ke i + '"'
49 variabel plus3SD = nilai variabel plus1SD + '"' +
nilai variabel plus3SD pada index ke i + '"'
50 IF (terdapat nilai pada variabel dataLaporan
pada index ke i pada index 'umur')
51 THEN
52 inisialisasi variabel umur = nilai
variabel dataLaporan pada index ke i pada index 'umur'
53 nilai variabel bb_timbang pada index
'nilai' variabel umur = nilai variabel dataLaporan pada index ke i
pada index 'bb ukur'
54 ENDIF
55 variabel data_bb = nilai variabel data bb + '"' +
nilai variabel bb_timbang pada index ke i + '"'
56 i = i + 1
57 ENDDOWHILE
58
59 Inisialisasi variabel data pada index 'bulan' = nilai variabel
bulan
60 Inisialisasi variabel data pada index 'minus3' = nilai variabel
minus3SD
61 Inisialisasi variabel data pada index 'minus2' = nilai variabel
minus2SD
62 Inisialisasi variabel data pada index 'minus1' = nilai variabel
minus1SD
63 Inisialisasi variabel data pada index 'median' = nilai variabel
median
64 Inisialisasi variabel data pada index 'plus1' = nilai variabel
plus1SD
65 Inisialisasi variabel data pada index 'plus2' = nilai variabel
plus2SD
66 Inisialisasi variabel data pada index 'plus3' = nilai variabel
plus3SD
67 Inisialisasi variabel data pada index 'data_bb' = nilai variabel
data_bb
68 Inisialisasi variabel data pada index 'kbm' = nilai variabel kbm
69 Inisialisasi variabel data pada index 'dataBalita' = nilai
variabel dataBalita
70 Inisialisasi variabel data pada index 'dataLaporan' = nilai
variabel dataLaporan
71
72 redirect menuju halaman dashboard_lihat_kms dengan membawa
variabel data
73
74 END

```

5.3.3 Perancangan Komponen Menampilkan Laporan Gizi

Pada Tabel 5.3 merupakan hasil perancangan komponen untuk fungsi Menampilkan Laporan Gizi, dengan nama fungsi lihatLaporan().

Tabel 5.3 Pseudocode fungsi lihatLaporan()

Fungsi lihatLaporan()	
1	START
2	
3	IF (terdapat nilai pada variabel SESSION pada index 'noreg')
4	THEN
5	inisialisasi variabel noreg = nilai variabel SESSION pada
6	index 'noreg'
	inisialisasi variabel jumlahBalitaTerdaftar = 0



```

7      IF (terdapat nilai pada variabel POST pada index
8      'submit')
9          THEN
10         Inisialisasi variabel Posyandu = nilai POST
11         dengan index 'Posyandu'
12         Inisialisasi variabel tanggalFilter = 1
13         Inisialisasi variabel bulanFilter = nilai POST
14         dengan index 'bulan'
15         Inisialisasi variabel tahunFilter = nilai POST
16         dengan index 'tahun'
17         IF (nilai variabel Posyandu != Semua
18         Posyandu)
19             THEN
20                 Inisialisasi variabel data pada
21                 index 'laporanBalitaNormal' = hasil kembalian dari pemanggilan
22                 method getLaporanBalitaNormalFilter
23                 (noreg, Posyandu, bulanFilter, tahunFilter)
24                 Inisialisasi variabel data pada
25                 index 'laporanBalitaMalnutrisi' = hasil kembalian dari
26                 pemanggilan method getLaporanBalitaMalnutrisiFilter
27                 (noreg, Posyandu, bulanFilter, tahunFilter)
28                 Inisialisasi variabel data pada
29                 index 'kaderTerdaftar' = hasil kembalian dari pemanggilan method
30                 getJumlahKaderTerdaftarFilter (noreg, Posyandu)
31                 Inisialisasi variabel data pada
32                 index 'kaderHadir' = hasil kembalian dari pemanggilan method
33                 getJumlahKaderHadirFilter (noreg, Posyandu, bulanFilter, tahunFilter)
34                 Inisialisasi variabel data pada
35                 index 'balitaHadir' = hasil kembalian dari pemanggilan method
36                 getJumlahBalitaPeriksaFilter
37                 (noreg, Posyandu, bulanFilter, tahunFilter)
38                 Inisialisasi variabel balita =
39                 hasil kembalian dari pemanggilan method getJumlahTerdaftarFilter
40                 (noreg, Posyandu, bulanFilter, tahunFilter)
41                 Inisialisasi variabel parameter =
42                 nilai dari variabel tahunFilter-bulanFilter-tanggalFilter dalam
43                 bentuk waktu
44                 Inisialisasi variabel i = 0
45                 DOWHILE i < jumlah variabel
46                 balita
47                 Inisialisasi variabel umur
48                 = hasil nilai pembulatan (variabel parameter - nilai variabel
49                 balita pada index ke i dan index tanggal lahir) / 2629746
50                 IF (umur > 60)
51                     THEN
52                         CONTINUE
53                     ELSE
54                         variabel
55                         jumlahBalitaTerdaftar = jumlahBalitaTerdaftar + 1
56                     ENDF
57                 variabel i = i + 1
58                 ENDDOWHILE
59                 Inisialisasi variabel data pada
60                 index 'balitaTerdaftar' = nilai variabel jumlahBalitaTerdaftar
61                 ELSE
62                     Inisialisasi variabel data pada
63                     index 'laporanBalitaNormal' = hasil kembalian dari pemanggilan
64                     method getLaporanBalitaNormal (noreg, bulanFilter, tahunFilter)
65                     Inisialisasi variabel data pada
66                     index 'laporanBalitaMalnutrisi' = hasil kembalian dari
67                     pemanggilan method getLaporanBalitaMalnutrisi
68                     (noreg, bulanFilter, tahunFilter)

```



```

38     Inisialisasi variabel data pada
index 'kaderTerdaftar' = hasil kembalian dari pemanggilan method
getJumlahTerdaftar (noreg, Posyandu)
39     Inisialisasi variabel data pada
index 'kaderHadir' = hasil kembalian dari pemanggilan method
getJumlahKaderHadirFilter (noreg, bulanFilter, tahunFilter)
40     Inisialisasi variabel data pada
index 'balitaHadir' = hasil kembalian dari pemanggilan method
getJumlahBalitaPeriksa (noreg, bulanFilter, tahunFilter)
41     Inisialisasi variabel balita =
hasil kembalian dari pemanggilan method getJumlahTerdaftar
(noreg, bulanFilter, tahunFilter)
42     Inisialisasi variabel parameter =
nilai dari variabel tahunFilter-bulanFilter-tanggalFilter dalam
bentuk waktu
43     Inisialisasi variabel i = 0
44     DOWHILE i < jumlah variabel
balita
45     Inisialisasi variabel umur
hasil nilai pembulatan (variabel parameter nilai variabel
balita pada index ke i dan index 'tanggal lahir') / 2629746
46     IF (umur > 60)
47     THEN
48     CONTINUE
49     ELSE
49     variabel
50     jumlahBalitaTerdaftar = jumlahBalitaTerdaftar + 1
51     ENDFILE
52     variabel i = i + 1
52     ENDDOWHILE
53     Inisialisasi variabel data pada
54     index 'balitaTerdaftar' = nilai variabel jumlahBalitaTerdaftar
55     ENDFILE
56     ELSE
57     Inisialisasi variabel data pada index
57     'laporanBalitaNormal' = hasil kembalian dari pemanggilan method
58     getLaporanBalitaNormal (noreg, bulanFilter, tahunFilter)
59     Inisialisasi variabel data pada index
59     'laporanBalitaMalnutrisi' = hasil kembalian dari pemanggilan
method getLaporanBalitaMalnutrisi (noreg, bulanFilter, tahunFilter)
60     Inisialisasi variabel data pada index
60     'kaderTerdaftar' = hasil kembalian dari pemanggilan method
getJumlahTerdaftar (noreg, Posyandu)
61     Inisialisasi variabel data pada index
61     'kaderHadir' = hasil kembalian dari pemanggilan method
getJumlahKaderHadirFilter (noreg, bulanFilter, tahunFilter)
62     Inisialisasi variabel data pada index
62     'balitaHadir' = hasil kembalian dari pemanggilan method
getJumlahBalitaPeriksa (noreg, bulanFilter, tahunFilter)
63     Inisialisasi variabel balita = hasil kembalian
dari pemanggilan method getJumlahTerdaftar
(noreg, bulanFilter, tahunFilter)
64     Inisialisasi variabel parameter = nilai dari
variabel tahunFilter-bulanFilter-tanggalFilter dalam bentuk waktu
64     Inisialisasi variabel i = 0
65     DOWHILE i < jumlah variabel balita
65     Inisialisasi variabel umur = hasil
66     nilai pembulatan (variabel parameter - nilai variabel balita pada
66     index ke i dan index 'tanggal lahir') / 2629746
67     IF (umur > 60)
68     THEN
69     CONTINUE
70     ELSE
70     variabel jumlahBalitaTerdaftar =
71     jumlahBalitaTerdaftar + 1

```




```

72      ENDIF
73      variabel i = i + 1
74      ENDDOWHILE
75      Inisialisasi variabel data pada index
76      'balitaTerdaftar' = nilai variabel jumlahBalitaTerdaftar
77      ENDIF
78      Inisialisasi variabel data pada index 'Posyandu' = hasil
79      kembalian dari pemanggilan method getDataByNoreg(noreg)
80      Inisialisasi variabel getTanggal = hasil kembalian dari
81      pemanggilan method getTanggalLaporan(noreg)
82      Inisialisasi variabel cekBulan dan cekTahun = ''
83      Inisialisasi variabel bulan bertipe data array =
84      nilai index '1' = 'Januari'
85      nilai index '2' = 'Februari'
86      nilai index '3' = 'Maret'
87      nilai index '4' = 'April'
88      nilai index '5' = 'Mei'
89      nilai index '6' = 'Juni'
90      nilai index '7' = 'Juli'
91      nilai index '8' = 'Agustus'
92      nilai index '9' = 'September'
93      nilai index '10' = 'Oktober'
94      nilai index '11' = 'November'
95      nilai index '12' = 'Desember'
96      Inisialisasi variabel tahun bertipe data array
97      Inisialisasi variabel i=0
98      DOWHILE i < jumlah variabel getTanggal
99      Inisialisasi variabel ambilTahun = nilai tahun pada
100     variabel getTanggal pada index ke i dan index 'tanggal_periksa'
101     IF nilai variabel ambilTahun == variabel cekTahun
102     THEN
103     CONTINUE
104     ELSE
105     Inisialisasi variabel cekTahun = nilai tahun
106     pada variabel getTanggal pada index ke i dan index
107     'tanggal_periksa'
108     Memasukkan nilai variabel cekTahun ke array
109     tahun
110     ENDIF
111     Inisialisasi variabel data pada index 'bulan' = nilai
112     variabel bulan
113     Inisialisasi variabel data pada index 'tahun' = nilai
114     variabel tahun
115     Inisialisasi variabel data pada index 'namaPosyandu' =
116     nilai variabel namaPosyandu
117     Inisialisasi variabel data pada index 'bulanFilter' = nilai
118     variabel bulanFilter
119     Inisialisasi variabel data pada index 'tahunFilter' = nilai
120     variabel tahunFilter
121     redirect halaman dashboard_bidan_laporan dengan membawa
122     variabel data
123
124     ELSE
125     redirect halaman login
126
127     ENDIF
128     END

```



5.4 Perancangan Antarmuka

Perancangan antarmuka dibuat untuk memberikan gambaran dari tampilan sistem informasi yang akan dibuat. Pada bagian ini, akan dijabarkan mengenai hasil perancangan antarmuka untuk lima antarmuka utama yang berada pada sistem informasi ini, yaitu perancangan antarmuka untuk *dashboard user*, perancangan antarmuka untuk *form* pencatatan kesehatan balita, perancangan antarmuka untuk profil balita, perancangan antarmuka untuk Kartu Menuju Sehat (KMS) balita, dan terakhir perancangan antarmuka untuk laporan gizi.

5.4.1 Perancangan Antarmuka *Dashboard User*

Dalam Gambar 5.8 merupakan hasil rancangan antarmuka untuk *dashboard user*.



Gambar 5.8 Rancangan antarmuka *dashboard user*

Pada Tabel 5.4 merupakan penjelasan terkait komponen-komponen yang terdapat pada rancangan antarmuka *dashboard user*.

Tabel 5.4 Penjelasan rancangan antarmuka *dashboard user*

No.	Nama Komponen	Deskripsi
1.	Menu <i>Dashboard</i>	Navigasi utama untuk fungsionalitas utama yang dapat <i>user</i> akses berbentuk <i>card</i> persegi panjang.
2.	Nama menu	Nama dari menu <i>dashboard</i> yang dapat <i>user</i> akses.
3.	<i>Icon</i> menu	<i>Icon</i> yang merepresentasikan menu tersebut
4.	Informasi Tambahan	Teks yang dapat menampilkan informasi tambahan terkait menu yang akan ditampilkan seperti "Jumlah Balita Terdaftar: 10"
5.	<i>Side Menu</i>	Menu yang berada pada bagian sebelah kiri dari halaman utama yang dapat digunakan untuk melakukan navigasi antar menu
6.	Nama dan Logo Sistem	Teks yang menampilkan sebuah logo / <i>icon</i> dan nama dari sistem informasi ini yaitu "Sistem Informasi Gizi Balita"
7.	<i>Dropdown Logout</i>	Teks yang menampilkan tulisan "Hai," diikuti dengan nama dari <i>user</i> yang sedang mengakses sistem, dan dapat ditekan untuk menampilkan sebuah <i>dropdown</i> untuk melakukan <i>logout</i> dari sistem.



5.4.2 Perancangan Antarmuka *Form* Pencatatan Kesehatan Balita

Dalam Gambar 5.9 merupakan hasil rancangan antarmuka untuk *form* pencatatan kesehatan anak.



Gambar 5.9 Rancangan antarmuka *form* pencatatan kesehatan balita

Pada Tabel 5.5, merupakan penjelasan terkait komponen-komponen yang terdapat pada rancangan antarmuka *form* pencatatan kesehatan balita.

Tabel 5.5 Penjelasan rancangan antarmuka *form* pencatatan kesehatan balita

No.	Nama Komponen	Deskripsi
1.	Label <i>Input</i>	Label untuk kolom-kolom <i>input</i> pada <i>form</i> .
2.	Kolom <i>Input</i>	Kolom <i>input</i> yang berada pada <i>form</i> yang dapat <i>user</i> gunakan untuk memasukkan data terkait hasil pemeriksaan kesehatan balita seperti tanggal Posyandu, nama balita, berat badan, tinggi badan, lingkaran kepala, cara pengukuran, imunisasi, vitamin A, ASI, pemberian makanan tambahan, keluhan, maupun diagnosa balita.
3.	<i>Submit Button</i>	<i>Button</i> yang dapat ditekan untuk mengirimkan data yang telah dimasukkan pada kolom <i>input</i> pada sistem.



5.4.3 Perancangan Antarmuka Profil Balita

Dalam Gambar 5.10 merupakan hasil perancangan antarmuka profil balita



Gambar 5.10 Rancangan antarmuka profil balita

Pada Tabel 5.6 merupakan penjelasan komponen-komponen yang berada pada rancangan antarmuka profil balita.

Tabel 5.6 Penjelasan rancangan antarmuka profil balita

No.	Nama Komponen	Deskripsi
1.	<i>Breadcrumbs</i>	<i>Breadcrumbs</i> yang menyediakan navigasi untuk pengguna dapat berpindah ke halaman sebelumnya
2.	Teks Nama Balita	Teks yang berisikan nama balita yang sedang ditampilkan profilnya.
3.	<i>Button</i> Lihat KMS	<i>Button</i> yang terdapat pada bagian profil balita untuk menampilkan Kartu Menuju Sehat milik balita tersebut
4.	<i>Button</i> Saran Gizi	<i>Button</i> yang terdapat pada bagian profil balita untuk mengirimkan saran gizi untuk balita tersebut.
5.	Data Balita	Teks yang menampilkan data mengenai balita yang sedang ditampilkan profil nya, seperti nama, jenis kelamin, tanggal lahir, umur, berat badan lahir, NIK, No. BPJS, Alamat, Nama Ayah dan Ibu, No. HP orang tua, maupun status gizi dalam bentuk BB/U, TB/U, dan BB/TB
6.	Tabel Riwayat Pemeriksaan Balita	Tabel yang menampilkan data riwayat pemeriksaan balita setiap bulannya di Posyandu, dan pada tabel ini juga <i>user</i> dapat melakukan modifikasi data riwayat pemeriksaan balita maupun menghapus data tersebut



5.4.4 Perancangan Antarmuka Kartu Menuju Sehat (KMS) Balita

Dalam Gambar 5.11, merupakan hasil rancangan antarmuka Kartu Menuju Sehat (KMS) balita.



Gambar 5.11 Rancangan antarmuka Kartu Menuju Sehat (KMS) balita

Pada Tabel 5.7, merupakan penjelasan terkait komponen-komponen yang terdapat pada rancangan antarmuka Kartu Menuju Sehat (KMS) balita.

Tabel 5.7 Penjelasan rancangan antarmuka Kartu Menuju Sehat (KMS) balita

No.	Nama Komponen	Deskripsi
1.	Breadcrumbs	<i>Breadcrumbs</i> yang menyediakan navigasi untuk pengguna dapat berpindah ke halaman sebelumnya
2.	Container KMS Balita	<i>Container</i> yang terdapat teks “KMS Balita” diikuti dengan nama balita yang KMS nya sedang ditampilkan, serta berfungsi untuk menampilkan grafik pertumbuhan berat badan balita dan tabel riwayat pemeriksaan kesehatan balita yang dapat berubah warnanya sesuai dengan jenis kelamin balita yang sedang ditampilkan.
3.	Grafik pertumbuhan berat badan balita	Grafik yang dapat menampilkan pertumbuhan berat badan balita berdasarkan hasil pencatatan oleh Posyandu ke dalam sistem dalam bentuk garis, serta perbandingan berat badan dengan standar antropometri mulai dari -3 SD hingga $+3$ SD.
4.	Tabel Riwayat Pemeriksaan Balita	Tabel yang menampilkan data riwayat pemeriksaan balita setiap bulannya di Posyandu.



5.4.5 Perancangan Antarmuka Laporan Gizi

Dalam Gambar 5.12, merupakan hasil rancangan antarmuka untuk laporan gizi



Gambar 5.12 Rancangan antarmuka laporan gizi

Pada Tabel 5.8, merupakan penjelasan terkait komponen-komponen yang terdapat pada rancangan antarmuka laporan gizi.



Tabel 5.8 Penjelasan rancangan antar muka laporan gizi

No.	Nama Komponen	Deskripsi
1.	<i>Breadcrumbs</i>	<i>Breadcrumbs</i> yang menyediakan navigasi untuk pengguna dapat berpindah ke halaman sebelumnya
2.	<i>Filter</i> Posyandu	Label dengan teks “Posyandu” dan <i>dropdown</i> yang menampilkan pilihan Posyandu yang dapat dipilih untuk menampilkan data laporan gizi pada Posyandu tersebut.
3.	<i>Filter</i> Bulan	Label dengan teks “Bulan” dan <i>dropdown</i> yang menampilkan pilihan bulan dari Januari hingga Desember yang dapat dipilih untuk menampilkan data laporan gizi pada bulan tersebut.
4.	<i>Filter</i> Tahun	Label dengan teks “Tahun” dan <i>dropdown</i> yang menampilkan pilihan tahun yang dapat dipilih untuk menampilkan data laporan gizi pada tahun tersebut.
5.	<i>Button</i> Lihat	<i>Button</i> yang berfungsi untuk melakukan seleksi data (<i>Filter</i>) laporan gizi pada Posyandu, bulan, dan tahun yang telah dipilih oleh <i>user</i> .
6.	Teks judul halaman laporan gizi	Teks judul halaman laporan gizi dengan tulisan “Laporan Gizi Balita dan Pelayanan Posyandu”
7.	Teks sumber data	Teks yang bertuliskan sumber data dari laporan gizi yang sedang ditampilkan, seperti “Data Posyandu Srikaya Desember 2019”
8.	<i>Button</i> unduh laporan balita pertumbuhan normal	<i>Button</i> yang berfungsi untuk melakukan unduh data ke dalam format Microsoft Excel untuk laporan balita pertumbuhan normal
9.	Tabel balita pertumbuhan normal	Tabel yang menampilkan data-data balita dengan pertumbuhan normal.
10.	<i>Button</i> unduh laporan balita malnutrisi	<i>Button</i> yang berfungsi untuk melakukan unduh data ke dalam format Microsoft Excel untuk laporan balita malnutrisi
11.	Tabel balita pertumbuhan normal	Tabel yang menampilkan data-data balita yang mengalami malnutrisi.
12.	Grafik laporan kehadiran balita	Grafik yang menampilkan jumlah perbandingan antara jumlah balita yang hadir dengan jumlah balita yang terdaftar pada Posyandu tersebut pada periode tersebut dalam bentuk <i>bar chart</i> .
13.	Grafik laporan kehadiran kader	Grafik yang menampilkan jumlah perbandingan antara jumlah kader yang hadir dengan jumlah kader yang terdaftar pada Posyandu dalam bentuk <i>bar chart</i> .



5.5 Implementasi Kode Program

Pada bagian ini akan dijabarkan mengenai hasil implementasi kode program pada beberapa fungsi utama yang berada pada sistem ini. Terdapat penjabaran tiga kode program fungsi utama pada bagian ini, yaitu kode program untuk fungsi pencatatan kesehatan balita, fungsi untuk menampilkan KMS balita, serta fungsi untuk menampilkan laporan gizi.

5.5.1 Kode Program Pencatatan Kesehatan Balita

Pada Tabel 5.9 merupakan hasil implementasi kode program untuk fungsi pencatatan kesehatan balita dengan nama fungsi `catatKesehatanBalita()` yang berada pada kelas *controller* Balita. Fungsi utama dari kode ini adalah untuk menerima hasil pengukuran dan pemeriksaan balita yang telah dilakukan di Posyandu, melakukan perhitungan status gizi berdasarkan kondisi balita tersebut, dan menyimpannya di dalam sistem.

Tabel 5.9 Kode program fungsi `catatKesehatanBalita()`

```

1 function catatKesehatanBalita() {
2     $post = $this->input->post();
3     $nama_balita = $post['nama_balita_kesehatan'];
4     $dataBalita = $this->Balita_model-
5     >getDataByName($nama_balita);
6     $tgl_periksa = $post['tgl_pemeriksaan_kesehatan'];
7     $umurPeriksa = floor((strtotime($tgl_periksa) -
8     strtotime($dataBalita[0]['tanggal_lahir'])) / 2629746);
9     $cara_pengukuran = $post['cara_pengukuran'];
10    $ID_balita = $dataBalita[0]['ID_balita'];
11
12    $cekUmur = $this->Laporan_kesehatan_model-
13    >cekUmurGanda($ID_balita,$umurPeriksa);
14
15    if($umurPeriksa >= 0 && $umurPeriksa <= 60 && $cekUmur == 0) {
16
17        $bb_ukur = $post['bb_ukur'];
18        $tb_ukur = $post['tb_ukur'];
19
20        $jk = $dataBalita[0]['jenis_kelamin'];
21
22        if($umurPeriksa <= 24 && $cara_pengukuran == "Berdiri") {
23            $tb_ukur += 0.7;
24        }
25        if($umurPeriksa > 24 && $cara_pengukuran == "Telentang") {
26            $tb_ukur -= 0.7;
27        }
28        $statusGizi = $this-
29        >statusGizi($jk,$umurPeriksa,$bb_ukur,$tb_ukur,$cara_pengukuran);
30
31        $data = array(
32            'ID_balita' => $ID_balita,
33            'umur' => $umurPeriksa,
34            'tb_ukur' => $tb_ukur,
35            'bb_u' => $statusGizi['bb_u'],
36            'tb_u' => $statusGizi['tb_u'],
37            'bb_tb' => $statusGizi['bb_tb']
38        );
39    }

```




```

40     $status = $this->Laporan_kesehatan_model->
41     >insertDataLaporan($data);
42
43
44     if($status){
45         redirect('Dashboard/?status=berhasilCatatKesehatan');
46     }
47     else{
48         redirect('Dashboard/?status=gagalCatatKesehatan');
49     }
50 }
51 else{
52     redirect('Dashboard/?status=gagalCatatKesehatan');
53 }
54 }

```

Pada baris 2 – 10 pada fungsi `catatKesehatanBalita` berfungsi untuk menerima input yang dilakukan oleh pengguna serta mengambil data balita dari basis data berdasarkan nama balita yang diinputkan oleh pengguna, untuk menghitung umur saat balita tersebut diperiksa dan mengambil ID dari balita tersebut. Pada baris 12 – 15 berfungsi untuk melakukan pengecekan umur dari balita tersebut saat dilakukan pemeriksaan, dimana apabila umur balita saat diperiksa sudah terdapat pada basis data atau tidak sesuai dengan kriteria umur seorang balita yaitu diatas atau sama dengan 0 dan dibawah atau sama dengan 60 bulan, maka sistem akan menolak pencatatan tersebut, sedangkan jika memenuhi kriteria umur tersebut, maka akan dilanjutkan proses pencatatannya. Pada baris 17 – 27 akan dilakukan proses penyesuaian tinggi badan balita, dimana apabila umur periksa balita tersebut dibawah atau sama dengan 24 bulan dan cara pengukurannya berdiri, maka tinggi badan balita tersebut akan ditambahkan 0.7 cm, sedangkan apabila umur periksa balita tersebut diatas 24 bulan dan cara pengukurannya secara telentang, maka tinggi badan balita tersebut akan dikurangi dengan 0.7 cm.

Lalu pada baris 28-29 akan dilakukan pemanggilan fungsi `statusGizi($jk,$umurPeriksa,$bb_ukur,$tb_ukur,$cara_pengukuran)` yang akan mengembalikan tiga nilai status gizi yaitu status gizi berat badan berdasarkan umur, tinggi badan berdasarkan umur, serta berat badan dibandingkan dengan tinggi badan. Pada baris 31 – 40 dilakukan proses penyimpanan data hasil pemeriksaan balita tersebut pada basis data, dan pada baris 43-52 adalah untuk melakukan *redirect* pada halaman Dashboard berdasarkan kondisi yang terjadi pada fungsi tersebut.

5.5.2 Kode Program Menampilkan KMS Balita

Pada Tabel 5.10 merupakan hasil implementasi kode program untuk menampilkan KMS balita yang bisa diakses oleh aktor Bidan maupun aktor Posyandu dengan nama fungsi `lihatKMS($ID_balita)` yang berada pada kelas *controller* Balita. Fungsi utama dari kode ini adalah untuk menyajikan hasil pencatatan yang telah dilakukan oleh pihak Posyandu untuk setiap anak balita yang melakukan pemeriksaan kesehatan dalam bentuk grafik pertumbuhan berat badan anak serta tabel riwayat hasil pengukuran berat badan anak tersebut.

Tabel 5.10 Kode program fungsi LihatKMS(\$ID_balita)

```

1 function lihatKMS($ID_balita){
2   $dataLaporan = $this->Laporan_kesehatan_model-
3   >getDataLaporanById($ID_balita);
4   $dataBalita = $this->Balita_model->getDataById($ID_balita);
5   $jks = $dataBalita[0]['jenis_kelamin'];
6   $bulan = '';
7   $minus3SD = '';
8   $minus2SD = '';
9   $minus1SD = '';
10  $medianSD = '';
11  $plus1SD = '';
12  $plus2SD = '';
13  $plus3SD = '';
14  $data_bk = '';
15  $kkm = array(0,0.8,0.9,0.8,0.6,0.5,0.4,0.3,0.2);
16  $bk_timbang = array(0,0,0,0,0,0,0,0,0,
17                    0,0,0,0,0,0,0,0,0,
18                    0,0,0,0,0,0,0,0,0,
19                    0,0,0,0,0,0,0,0,0,
20                    0,0,0,0,0,0,0,0,0,
21                    0,0,0,0,0,0,0,0,0);
22
23  if ($jks == 'Laki-Laki') {
24    $minusTigaSD = array(2.1, 2.9, 3.8, 4.4, 4.9, 5.3, 5.7,
25    5.9, 6.2, 6.4, 6.6, 6.8, 6.9, 7.1, 7.2, 7.4, 7.5, 7.7, 7.8, 8.0,
26    8.1, 8.2, 8.4, 8.5, 8.6, 8.8, 8.9, 9.0, 9.1, 9.2, 9.4, 9.5, 9.6,
27    9.7, 9.8, 9.9, 10.0, 10.1, 10.2, 10.3, 10.4, 10.5, 10.6, 10.7,
28    10.8, 10.9, 11.0, 11.1, 11.2, 11.3, 11.4, 11.5, 11.6, 11.7, 11.8,
29    11.9, 12.0, 12.1, 12.2, 12.3, 12.4);
30
31    $minusDuaSD = array(2.5, 3.4, 4.3, 5.0, 5.6, 6.0, 6.4,
32    6.7, 6.9, 7.1, 7.4, 7.6, 7.7, 7.9, 8.1, 8.3, 8.4, 8.6, 8.8, 8.9,
33    9.1, 9.2, 9.4, 9.5, 9.7, 9.8, 10.0, 10.1, 10.2, 10.4, 10.5, 10.7,
34    10.8, 10.9, 11.0, 11.2, 11.3, 11.4, 10.5, 11.6, 11.8, 11.9, 12.0,
35    12.1, 12.2, 12.4, 12.5, 12.6, 12.7, 12.8, 12.9, 13.1, 13.2, 13.3,
36    13.4, 13.5, 13.6, 13.7, 13.8, 14.0, 14.1);
37
38    $minusSatuSD = array(2.9, 3.9, 4.9, 5.7, 6.2, 6.7, 7.1,
39    7.4, 7.7, 8.0, 8.2, 8.4, 8.6, 8.8, 9.0, 9.2, 9.4, 9.6, 9.8, 10.0,
40    10.1, 10.3, 10.5, 10.7, 10.8, 11.0, 11.2, 11.3, 11.5, 11.7, 11.8,
41    12.0, 12.1, 12.3, 12.4, 12.6, 12.7, 12.9, 13.0, 13.1, 13.3, 13.4,
42    13.6, 13.7, 13.8, 14.0, 14.1, 14.3, 14.4, 14.5, 14.6, 14.8, 15.0,
43    15.1, 15.2, 15.4, 15.5, 15.6, 15.8, 15.9, 16.0);
44
45    $median = array(3.3, 4.5, 5.6, 6.4, 7.0, 7.5, 7.9, 8.3,
46    8.6, 8.9, 9.2, 9.4, 9.6, 9.9, 10.1, 10.3, 10.5, 10.7, 10.9, 11.1,
47    11.3, 11.5, 11.8, 12.0, 12.2, 12.4, 12.5, 12.7, 12.9, 13.1, 13.3,
48    13.5, 13.7, 13.8, 14.0, 14.2, 14.3, 14.5, 14.7, 14.8, 15.0, 15.2,
49    15.3, 15.5, 15.7, 15.8, 16.0, 16.2, 16.3, 16.5, 16.7, 16.8, 17.0,
50    17.2, 17.3, 17.5, 17.7, 17.8, 18.0, 18.2, 18.3);
51
52    $plusSatuSD =
53    array(3.9, 5.1, 6.3, 7.2, 7.8, 8.4, 8.8, 9.2, 9.6, 9.9,
54    10.2, 10.5, 10.8, 11.0, 11.3, 11.5, 11.7, 12.0, 12.2, 12.5,
55    12.7, 12.9, 13.2, 13.4, 13.6, 13.9, 14.1, 14.3, 14.5, 14.8, 15.0,
56    15.2, 15.4, 15.6, 15.8, 16.0, 16.2, 16.4, 16.6, 16.8, 17.0, 17.2,
57    17.4, 17.6, 17.8, 18.0, 18.2, 18.4, 18.6, 18.8, 19.0, 19.2, 19.4,
58    19.6, 19.8, 20.0, 20.2, 20.4, 20.6, 20.8, 21.0);
59
60    $plusDuaSD =
61

```





```

62 array (4.4, 5.8, 7.1, 8.0, 8.7, 9.3, 9.8, 10.3, 10.7);
63 11.0, 11.4, 10.7, 12.0, 12.3, 12.6, 12.8, 13.1, 13.4, 13.7, 13.9,
64 14.2, 14.5, 14.7, 15.0, 15.3, 15.5, 15.8, 16.1, 16.3, 16.6,
65 16.9, 17.1, 17.4, 17.6, 17.8, 18.1, 18.3, 18.6, 18.8, 19.0,
66 19.3, 19.5, 19.7, 20.0, 20.2, 20.5, 20.7, 20.9, 21.2, 21.4,
67 21.7, 21.9, 22.2, 22.4, 22.7, 22.9, 23.2, 23.4, 23.7, 23.9, 24.2);
68
69 $plusTigaSD =
70 array (5.0, 6.6, 8.0, 9.0, 9.7, 10.4, 10.9, 11.4, 11.9,
71 12.3, 12.7, 13.0, 13.3, 13.7, 14.0, 14.3, 14.6, 14.9, 15.3, 15.6,
72 15.9, 16.2, 16.5, 16.8, 17.1, 17.5, 17.8, 18.1, 18.4, 18.7, 19.0,
73 19.3, 19.6, 19.9, 20.2, 20.4, 20.7, 21.0, 21.3, 21.6, 21.9, 22.1,
74 22.4, 22.7, 23.0, 23.3, 23.6, 23.9, 24.2, 24.5, 24.8, 25.1, 25.4,
75 25.7, 26.0, 26.3, 26.6, 26.9, 27.2, 27.6, 27.9);
76
77 }
78
79 $minusTigaSD = array (2.0, 2.7, 3.4, 4.0, 4.4, 4.8, 5.1, 5.3,
80 5.6, 5.8, 5.9, 6.1, 6.3, 6.4, 6.6, 6.7, 6.9, 7.0, 7.2, 7.3, 7.5,
81 7.6, 7.8, 7.9, 8.1, 8.2, 8.4, 8.5, 8.6, 8.8, 8.9, 9.0, 9.1, 9.3,
82 9.4, 9.5, 9.6, 9.7, 9.8, 9.9, 10.1, 10.2, 10.3, 10.4, 10.5, 10.6,
83 10.7, 10.8, 10.9, 11.0, 11.1, 11.2, 10.3, 11.4, 11.5, 11.6, 11.7,
84 11.8, 11.9, 12.0, 12.1);
85
86 $minusDuaSD = array (2.4, 3.2, 3.9, 4.5, 5.0, 5.4, 5.7,
87 6.0, 6.3, 6.5, 6.7, 6.9, 7.0, 7.2, 7.4, 7.6, 7.7, 7.9, 8.1, 8.2,
88 8.4, 8.6, 8.7, 8.9, 9.0, 9.2, 9.4, 9.5, 9.7, 9.8, 10.0, 10.1,
89 10.3, 10.4, 10.5, 10.7, 10.8, 10.9, 11.1, 11.2, 11.3, 11.5, 11.6,
90 11.7, 11.8, 12.0, 12.1, 12.2, 12.3, 12.4, 12.6, 12.7, 12.8, 12.9,
91 13.0, 13.2, 13.3, 13.4, 13.5, 13.6, 13.7);
92
93 $minusSatuSD = array (2.8, 3.6, 4.5, 5.2, 5.7, 6.1, 6.5,
94 6.8, 7.0, 7.3, 7.5, 7.7, 7.9, 8.1, 8.3, 8.5, 8.7, 8.9, 9.1, 9.2,
95 9.4, 9.6, 9.8, 10.0, 10.2, 10.3, 10.5, 10.7, 10.9, 11.1, 11.2,
96 11.4, 11.6, 11.7, 11.9, 12.0, 12.2, 12.4, 12.5, 12.7, 12.8, 13.0,
97 13.1, 13.3, 13.4, 13.6, 13.7, 13.9, 14.0, 14.2, 14.3, 14.5, 14.6,
98 14.8, 14.9, 15.1, 15.2, 15.3, 15.5, 15.6, 15.8);
99
100 $median = array (3.2, 4.2, 5.1, 5.8, 6.4, 6.9, 7.3, 7.6,
101 7.9, 8.2, 8.5, 8.7, 8.9, 9.2, 9.4, 9.6, 9.8, 10.0, 10.2, 10.4,
102 10.6, 10.9, 11.1, 11.3, 11.5, 11.7, 11.9, 12.1, 12.3, 12.5, 12.7,
103 12.9, 13.1, 13.3, 13.5, 13.7, 13.9, 14.0, 14.2, 14.4, 14.6,
104 14.8, 15.0, 15.2, 15.3, 15.5, 15.7, 15.9, 16.1, 16.3, 16.4, 16.6,
105 16.8, 17.0, 17.2, 17.3, 17.5, 17.7, 17.9, 18.0, 18.2);
106
107 $plusSatuSD = array (3.7, 4.8, 5.8, 6.6, 7.3, 7.8, 8.2,
108 8.6, 9.0, 9.3, 9.6, 9.9, 10.1, 10.4, 10.6, 10.9, 11.1, 11.4, 11.6,
109 11.8, 12.1, 12.3, 12.5, 12.8, 13.0, 13.3, 13.5, 13.7, 14.0, 14.2,
110 14.4, 14.7, 14.9, 15.1, 15.4, 15.6, 15.8, 16.0, 16.3, 16.5, 16.7,
111 16.9, 17.2, 17.4, 17.6, 17.8, 18.1, 18.3, 18.5, 18.8, 19.0, 19.2,
112 19.4, 19.7, 19.9, 20.1, 20.3, 20.6, 20.8, 21.0, 21.2);
113
114 $plusDuaSD = array (4.2, 5.5, 6.6, 7.5, 8.2, 8.8, 9.3,
115 9.8, 10.2, 10.5, 10.9, 11.2, 11.5, 11.8, 12.1, 12.4, 12.6, 12.9,
116 13.2, 13.5, 13.7, 14.0, 14.3, 14.6, 14.8, 15.1, 15.4, 15.7, 16.0,
117 16.2, 16.5, 16.8, 17.1, 17.3, 17.6, 17.9, 18.1, 18.4, 18.7, 19.0,
118 19.2, 19.5, 19.8, 20.1, 20.4, 20.7, 20.9, 21.2, 21.5, 21.8, 22.1,
119 22.4, 22.6, 22.9, 23.2, 23.5, 23.8, 24.1, 24.4, 24.6, 24.9);
120
121 $plusTigaSD = array (4.8, 6.2, 7.5, 8.5, 9.3, 10.0, 10.6,
122 11.1, 11.6, 12.0, 12.4, 12.8, 13.1, 13.5, 13.8, 14.1, 14.5, 14.8,
123 15.1, 15.4, 15.7, 16.0, 16.4, 16.7, 17.0, 17.3, 17.7, 18.0, 18.3,
124 18.7, 19.0, 19.3, 19.6, 20.0, 20.3, 20.6, 20.9, 21.3, 21.6, 22.0,
125 22.3, 22.7, 23.0, 23.4, 23.7, 24.0, 24.5, 24.8, 25.2, 25.5, 25.9,
126 26.3, 26.6, 27.0, 27.4, 27.7, 28.1, 28.5, 28.8, 29.2, 29.5);

```



```

127 }
128     $umurMax = 0;
129     for($a=0; $a <= 60; $a++) {
130         if (isset($dataLaporan[$a]['umur'])) {
131             $umurMax = $dataLaporan[$a]['umur'];
132         }
133     }
134     for ($i=0; $i <= $umurMax; $i++) {
135         $bulan = $bulan . "'." . $i . "'";
136         $minus3SD = $minus3SD . "'." . $minusTigaSD[$i] . "'";
137         $minus2SD = $minus2SD . "'." . $minusDuaSD[$i] . "'";
138         $minus1SD = $minus1SD . "'." . $minusSatuSD[$i] . "'";
139         $medianSD = $medianSD . "'." . $median[$i] . "'";
140         $plus1SD = $plus1SD . "'." . $plusSatuSD[$i] . "'";
141         $plus2SD = $plus2SD . "'." . $plusDuaSD[$i] . "'";
142         $plus3SD = $plus3SD . "'." . $plusTigaSD[$i] . "'";
143         if (isset($dataLaporan[$i]['umur'])) {
144             $umur = $dataLaporan[$i]['umur'];
145             $bb_timbang[$umur] = $dataLaporan[$i]['bb_ukur'];
146         }
147         $data bb = $data bb . "'." . $bb_timbang[$i] . "'";
148     }
149     $data['bulan'] = trim($bulan, "'");
150     $data['minus3'] = trim($minus3SD, "'");
151     $data['minus2'] = trim($minus2SD, "'");
152     $data['minus1'] = trim($minus1SD, "'");
153     $data['median'] = trim($medianSD, "'");
154     $data['plus1'] = trim($plus1SD, "'");
155     $data['plus2'] = trim($plus2SD, "'");
156     $data['plus3'] = trim($plus3SD, "'");
157     $data['data_bb'] = trim($data bb, "'");
158     $data['kbm'] = $kbm;
159     $data['dataBalita'] = $dataBalita;
160     $data['dataKesehatan'] = $dataLaporan;
161
162     $this->load->view('dashboard_lihat_kms', $data);
163 }

```

Pada baris kode 2-21 berfungsi untuk melakukan inisialisasi variabel-variabel yang akan digunakan dalam menampilkan data yang dibutuhkan dalam KMS balita. Lalu pada baris 23 - 77 berfungsi untuk memberikan nilai untuk variabel \$minus3SD, \$minus2SD, \$minus1SD, \$medianSD, \$plus1SD, \$plus2SD, dan \$plus3SD apabila nilai variabel jk atau jenis kelamin dari balita tersebut adalah laki-laki, sedangkan pada baris 78 - 127 untuk anak perempuan, dimana angka-angka tersebut merupakan angka standar pertumbuhan berat badan anak balita berdasarkan standar antropometri penilaian status gizi anak yang dikeluarkan oleh Kementerian Kesehatan Republik Indonesia melalui Keputusan Menteri Kesehatan Republik Indonesia pada tahun 2011 (Kemenkes RI, 2011), dan contohnya dapat dilihat dalam Lampiran B.

Pada baris 128 - 148 memiliki fungsi untuk memasukkan nilai-nilai pada variabel baru dalam bentuk *string* menggunakan nilai daripada variabel-variabel yang telah diberikan nilai sebelumnya, serta data berat badan anak hasil pencatatan Posyandu yang telah tersimpan pada basis data. Pada baris 149 - 162 adalah untuk menyimpan nilai dari variabel-variabel yang telah diberikan nilai sebelumnya ke dalam variabel data yang berbentuk *array*, dimana nantinya variabel data tersebut akan dibawa pada tampilan dashboard_lihat_kms untuk ditampilkan.



5.5.3 Kode Program Menampilkan Laporan Gizi

Pada Tabel 5.11 merupakan hasil implementasi kode program untuk menampilkan laporan gizi yang bisa diakses oleh aktor Bidan dengan nama fungsi `lihatLaporan()` yang berada pada kelas *controller* Dashboard. Fungsi utama dari kode ini adalah untuk menyajikan dalam bentuk tabel serta grafik mengenai hasil pencatatan kesehatan balita yang telah dilakukan oleh masing-masing Posyandu yang dibawah Bidan tersebut. Pada laporan gizi ini, terdapat 4 macam jenis laporan, yaitu laporan untuk anak balita yang memiliki pertumbuhan normal, laporan untuk anak balita yang mengalami malnutrisi, laporan jumlah kehadiran balita, serta laporan jumlah kehadiran kader Posyandu.

Tabel 5.11 Kode program fungsi `lihatLaporan()`

```

Fungsi lihatLaporan()
1  public function lihatLaporan() {
2      if(isset($_SESSION['noreg'])) {
3          $noreg = $_SESSION['noreg'];
4          $jumlahBalitaTerdaftar = 0;
5          //JIKA USER MELAKUKAN FILTER
6          if(isset($_POST['submit'])) {
7              $Posyandu = $this->input->post('Posyandu');
8              $tanggalFilter = 1;
9              $bulanFilter = $this->input->post('bulan');
10             $tahunFilter = $this->input->post('tahun');
11
12             if($Posyandu != 'Semua Posyandu') {
13                 //MENAMPILKAN DATA SPESIFIK POSYANDU
14                 $data['laporanBalitaNormal'] =
15                 $this->Laporan_kesehatan_model->
16                 >getLaporanBalitaNormalFilter($noreg, $Posyandu, $bulanFilter, $tahunFilter);
17
18                 $data['laporanBalitaMalnutrisi'] =
19                 $this->Laporan_kesehatan_model->
20                 >getLaporanBalitaMalnutrisiFilter($noreg, $Posyandu, $bulanFilter, $tahunFilter);
21
22                 $data['kaderTerdaftar'] =
23                 $this->Kader_model->
24                 >getJumlahTerdaftarFilter($noreg, $Posyandu);
25
26                 $data['kaderHadir'] =
27                 $this->Laporan_kehadiran_kader_model->
28                 >getJumlahKaderHadirFilter($noreg, $Posyandu, $bulanFilter, $tahunFilter);
29
30                 $data['balitaHadir'] =
31                 $this->Laporan_kesehatan_model->
32                 >getJumlahBalitaPeriksaFilter($noreg, $Posyandu, $bulanFilter, $tahunFilter);
33
34                 $balita =
35                 $this->Balita_model->
36                 >getJumlahTerdaftarFilter($noreg, $Posyandu);
37
38                 $parameter = strtotime("$tahunFilter-$bulanFilter-$tanggalFilter");
39
40                 for($i=0; $i < sizeof($balita); $i++) {
41                     $sumur = floor(($parameter
42                     strtotime($balita[$i]['tanggal_lahir'])) / 2629746);
43                     if($sumur > 60) {
44                         continue;
45                     }

```



```

46     }
47     else{
48         $jumlahBalitaTerdaftar++;
49     }
50     }
51     $data['balitaTerdaftar'] =
52     $jumlahBalitaTerdaftar;
53
54
55     }
56     else{
57         //MENAMBAH SEMUA POSYANDU
58
59         $data['laporanBalitaNormal'] =
60         $this->laporan_kesehatan_model-
61         >getLaporanBalitaNormal($noreg,$bulanFilter,$tahunFilter);
62         $data['laporanBalitaMalnutrisi'] =
63         $this->laporan_kesehatan_model-
64         >getLaporanBalitaMalnutrisi($noreg,$bulanFilter,$tahunFilter);
65
66         $data['kaderTerdaftar'] =
67         $this->kader_model-
68         >getJumlahTerdaftar($noreg);
69         $data['kaderHadir'] =
70         $this->laporan_kehadiran_kader_model-
71         >getJumlahKaderHadir($noreg,$bulanFilter,$tahunFilter);
72
73         $data['balitaHadir'] =
74         $this->laporan_kesehatan_model-
75         >getJumlahBalitaPeriksa($noreg,$bulanFilter,$tahunFilter);
76
77         $balita = $this->Balita_model-
78         >getJumlahTerdaftar($noreg);
79         $parameter = strtotime("$tahunFilter-
80         $bulanFilter-$tanggalFilter");
81
82         for($i=0; $i < sizeof($balita); $i++){
83             $umur = floor(($parameter
84             strtotime($balita[$i]['tanggal_lahir']) / 2629746);
85             if($umur > 60){
86                 continue;
87             }
88             else{
89                 $jumlahBalitaTerdaftar++;
90             }
91         }
92         $data['balitaTerdaftar'] =
93         $jumlahBalitaTerdaftar;
94     }
95 }
96 }
97 }
98 else{
99     //DEFAULT NYA SAAT BUKA PERTAMA KALI
100     $Posyandu = 'Semua Posyandu';
101     $tanggalFilter = 1;
102     $bulanFilter = date('m');
103     $tahunFilter = date('Y');
104
105     $data['laporanBalitaNormal'] =
106     $this->laporan_kesehatan_model-
107     >getLaporanBalitaNormal($noreg,$bulanFilter,$tahunFilter);
108     $data['laporanBalitaMalnutrisi'] =
109     $this->laporan_kesehatan_model-
110     >getLaporanBalitaMalnutrisi($noreg,$bulanFilter,$tahunFilter);

```



```

111
112 $data['balitaHadir'] =
113 $this->Laporan_kesehatan_model-
114 >getJumlahBalitaPeriksa($noreg,$bulanFilter,$tahunFilter);
115
116 $data['kaderTerdaftar'] =
117 $this->Kader_model->getJumlahTerdaftar($noreg);
118 $data['kaderHadir'] =
119 $this->Laporan_kehadiran_kader_model-
120 >getJumlahKaderHadir($noreg,$bulanFilter,$tahunFilter);
121
122
123
124 $balita = $this->Balita_model-
125 >getJumlahTerdaftar($noreg);
126 $parameter = strtotime("$tahunFilter-$bulanFilter-
127 $tanggalFilter");
128
129 for($i=0; $i < sizeof($balita); $i++){
130     $umur = floor($parameter -
131     strtotime($balita[$i]['tanggal_lahir'])) / 2629746;
132     if($umur > 60){
133         continue;
134     }
135     else{
136         $jumlahBalitaTerdaftar++;
137     }
138 }
139 $data['balitaTerdaftar'] = $jumlahBalitaTerdaftar;
140
141 }
142 $data['Posyandu'] = $this->Posyandu_model-
143 >getDataByNoreg($noreg);
144
145 $getTanggal = $this->Laporan_kesehatan_model-
146 >getTanggalLaporan($noreg);
147 $cekBulan = '';
148 $cekTahun = '';
149 $bulan = array('1' => 'Januari',
150               '2' => 'Februari',
151               '3' => 'Maret',
152               '4' => 'April',
153               '5' => 'Mei',
154               '6' => 'Juni',
155               '7' => 'Juli',
156               '8' => 'Agustus',
157               '9' => 'September',
158               '10' => 'Oktober',
159               '11' => 'November',
160               '12' => 'Desember');
161 $tahun = array();
162
163 for ($i=0; $i < sizeof($getTanggal); $i++) {
164     $ambilTahun = date('Y',
165     strtotime($getTanggal[$i]['tanggal_periksa']));
166     if ($ambilTahun == $cekTahun) {
167         continue;
168     }
169     else {
170         $cekTahun = date('Y',
171         strtotime($getTanggal[$i]['tanggal_periksa']));
172         array_push($tahun, $cekTahun);
173     }
174 }
175

```



```

176 }
177 }
178 $data['bulan'] = $bulan;
179 $data['tahun'] = $tahun;
180 $data['namaPosyandu'] = $Posyandu;
181 $namaBulan= DateTime::createFromFormat('!m',
182 $bulanFilter);
183 $data['bulanFilter'] = $namaBulan->format('F');
184 $data['tahunFilter'] = $tahunFilter;
185
186 $this->load->view('dashboard_bidan_laporan',$data);
187
188 else{
189     redirect(base_url());
190 }
191 }
192

```

Pada baris kode 2-4 berfungsi untuk menginisialisasi variabel \$noreg dan \$jumlahBalitaTerdaftar. Lalu pada baris 6 – 10 adalah untuk memeriksa, apakah pengguna melakukan proses seleksi sumber data laporan yang akan ditampilkan, jika iya, maka akan dilakukan inialisasi variabel untuk \$Posyandu, \$tanggalFilter, \$bulanFilter, dan \$tahunFilter. Pada baris 12 – 55 akan dieksekusi apabila pengguna melakukan seleksi sumber data untuk menampilkan data laporan dari salah satu Posyandu yang tersedia, sedangkan pada baris 56 – 97 akan dieksekusi apabila pengguna melakukan seleksi sumber data laporan untuk menampilkan data laporan dari semua Posyandu yang ada, dan pada baris 98 – 141 akan dieksekusi pada saat pengguna pertama kali membuka halaman laporan gizi yang akan menampilkan data laporan untuk semua Posyandu pada bulan dan tahun saat ini.

Untuk ketiga kondisi tersebut, sistem akan melakukan pemanggilan ke beberapa fungsi ke kelas model untuk mendapatkan data-data yang diperlukan seperti data laporan balita pertumbuhan normal, data balita malnutrisi, jumlah balita yang hadir, jumlah balita yang terdaftar, jumlah kader yang hadir, maupun jumlah kader yang terdaftar. Lalu pada baris 142 – 177 merupakan proses untuk mempersiapkan data-data yang bisa digunakan oleh pengguna untuk proses seleksi sumber data laporan seperti data Posyandu yang dibawah oleh Bidan tersebut, data bulan serta data tahun berdasarkan data laporan yang tersedia pada basis data. Pada baris 178 – 186 dilakukan proses penyimpanan nilai pada variabel data yang berbentuk array, dimana nantinya variabel data tersebut akan dibawa pada tampilan dashboard_bidan_laporan untuk ditampilkan.

5.6 Implementasi Basis Data

Berdasarkan hasil perancangan *physical data model* yang telah dilakukan pada tahap perancangan, maka dari hasil perancangan tersebut akan diimplementasikan ke dalam sebuah basis data yang hasil implementasinya dapat dilihat pada Tabel 5.12.



Tabel 5.12 Hasil implementasi basis data

```

1 CREATE TABLE user (
2 ID_user INTEGER NOT NULL AUTO INCREMENT,
3 nama_user varchar(100) NOT NULL,
4 username varchar(50) NOT NULL,
5 password varchar(100) NOT NULL,
6 jenis_user varchar(50) NOT NULL,
7 foto_user varchar(100) DEFAULT NULL,
8 status varchar(100) NOT NULL,
9 PRIMARY KEY (ID_user),
10 CONSTRAINT username UNIQUE (username)
11 ) ENGINE=InnoDB DEFAULT CHARSET=latin1;
12
13 CREATE TABLE bidan (
14 nomor_registrasi varchar(100) NOT NULL,
15 ID_user INTEGER NOT NULL,
16 tempat_lahir varchar(100) NOT NULL,
17 tanggal_lahir date NOT NULL,
18 jenis_kelamin varchar(10) NOT NULL,
19 surat_registrasi varchar(100) NOT NULL,
20 PRIMARY KEY (nomor_registrasi),
21 CONSTRAINT user_bidan FOREIGN KEY (ID_user) REFERENCES user
22 (ID_user) ON DELETE CASCADE
23 ) ENGINE=InnoDB DEFAULT CHARSET=latin1;
24
25 CREATE TABLE Posyandu (
26 ID_Posyandu INTEGER NOT NULL AUTO INCREMENT
27 ID_user INTEGER NOT NULL,
28 nomor_registrasi_bidan varchar(100) NOT NULL,
29 alamat varchar(100) NOT NULL,
30 RT varchar(10) DEFAULT NULL,
31 RW varchar(10) DEFAULT NULL,
32 kelurahan_desa varchar(100) NOT NULL,
33 PRIMARY KEY (ID_Posyandu),
34 CONSTRAINT Posyandu_bidan FOREIGN KEY (nomor_registrasi_bidan)
35 REFERENCES bidan (nomor_registrasi) ON DELETE CASCADE ON UPDATE
36 CASCADE
37 CONSTRAINT user_Posyandu FOREIGN KEY (ID_user) REFERENCES user
38 (ID_user) ON DELETE CASCADE
39 ) ENGINE=InnoDB DEFAULT CHARSET=latin1;
40
41 CREATE TABLE orang_tua (
42 ID_ortu INTEGER NOT NULL AUTO INCREMENT,
43 ID_Posyandu INTEGER NOT NULL,
44 ID_user INTEGER NOT NULL,
45 nama_ayah varchar(100) NOT NULL,
46 nama_ibu varchar(100) NOT NULL,
47 nik_ayah varchar(100) NOT NULL,
48 nik_ibu varchar(100) NOT NULL,
49 no_kk varchar(100) NOT NULL,
50 alamat varchar(100) NOT NULL,
51 no_hp varchar(20) NOT NULL,
52 PRIMARY KEY (ID_ortu),
53 CONSTRAINT nik_ayah UNIQUE (nik_ayah),
54 CONSTRAINT nik_ibu UNIQUE (nik_ibu),
55 CONSTRAINT no_kk UNIQUE (no_kk),
56 CONSTRAINT no_hp UNIQUE (no_hp),
57 CONSTRAINT Posyandu_has_ortu FOREIGN KEY (ID_Posyandu)
58 REFERENCES Posyandu (ID_Posyandu) ON DELETE CASCADE ON UPDATE NO
59 ACTION,
60 CONSTRAINT user_ortu FOREIGN KEY (ID_user) REFERENCES user
61 (ID_user) ON DELETE CASCADE
62 ) ENGINE=InnoDB DEFAULT CHARSET=latin1;
63

```



```

64 CREATE TABLE balita
65     ID_balita INTEGER NOT NULL AUTO INCREMENT,
66     ID_ortu INTEGER NOT NULL,
67     ID_Posyandu INTEGER NOT NULL,
68     NIK varchar(100) DEFAULT NULL,
69     nama varchar(100) NOT NULL,
70     jenis_kelamin varchar(10) NOT NULL,
71     tanggal_lahir date NOT NULL,
72     bb_lahir double NOT NULL,
73     no_bpjs varchar(100) DEFAULT NULL,
74     status varchar(100) NOT NULL,
75     PRIMARY KEY (ID_balita),
76     CONSTRAINT anak_ortu FOREIGN KEY (ID_ortu) REFERENCES
77     orang_tua (ID_ortu) ON DELETE CASCADE ON UPDATE NO ACTION,
78     CONSTRAINT anak_Posyandu FOREIGN KEY (ID_Posyandu) REFERENCES
79     Posyandu (ID_Posyandu) ON DELETE CASCADE ON UPDATE NO ACTION
80 ) ENGINE=InnoDB DEFAULT CHARSET=latin1;
81
82 CREATE TABLE laporan_kesehatan (
83     ID_balita INTEGER NOT NULL,
84     tanggal_periksa date NOT NULL,
85     umur INTEGER NOT NULL,
86     bb_ukur double NOT NULL,
87     tb_ukur double NOT NULL,
88     lk_ukur double NOT NULL,
89     cara_pengukuran varchar(100) NOT NULL,
90     imunisasi varchar(100) DEFAULT NULL,
91     vit_a varchar(100) DEFAULT NULL,
92     asi varchar(100) DEFAULT NULL,
93     pnt varchar(100) DEFAULT NULL,
94     keluhan varchar(100) DEFAULT NULL,
95     diagnosa varchar(100) DEFAULT NULL,
96     bb_u varchar(100) NOT NULL,
97     tb_u varchar(100) NOT NULL,
98     bb_tb varchar(100) NOT NULL,
99     PRIMARY KEY (ID_balita, umur),
100     KEY ID_balita (ID_balita),
101     CONSTRAINT laporan_kesehatan_balita FOREIGN KEY (ID_balita)
102     REFERENCES balita (ID_balita) ON DELETE CASCADE ON UPDATE CASCADE
103 ) ENGINE=InnoDB DEFAULT CHARSET=latin1;
104
105 CREATE TABLE saran_gizi (
106     ID_balita INTEGER NOT NULL,
107     saran varchar(255) NOT NULL,
108     tanggal_saran date NOT NULL,
109     KEY ID_balita (ID_balita),
110     CONSTRAINT saran_gizi_balita FOREIGN KEY (ID_balita) REFERENCES
111     balita (ID_balita) ON DELETE CASCADE
112 ) ENGINE=InnoDB DEFAULT CHARSET=latin1;
113
114 CREATE TABLE jadwal_Posyandu (
115     ID_posyandy INTEGER NOT NULL,
116     tgl_Posyandu date NOT NULL,
117     waktu_Posyandu time NOT NULL,
118     tempat_Posyandu varchar(100) NOT NULL,
119     deskripsi_kegiatan_Posyandu varchar(100) NOT NULL,
120     timestamp_penjadwalan timestamp NOT NULL DEFAULT
121     CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP,
122     KEY ID_Posyandu (ID_Posyandu),
123     CONSTRAINT jadwal_Posyandu FOREIGN KEY (ID_Posyandu)
124     REFERENCES Posyandu (ID_Posyandu) ON DELETE CASCADE ON UPDATE NO
125     ACTION
126 ) ENGINE=InnoDB DEFAULT CHARSET=latin1;
127
128 CREATE TABLE kader (

```



```

129 NIK varchar(100) NOT NULL,
130 ID_Posyandu INTEGER NOT NULL,
131 nama_kader varchar(100) NOT NULL,
132 jenis_kelamin varchar(10) NOT NULL,
133 alamat_kader varchar(100) NOT NULL,
134 foto_ktp varchar(20) NOT NULL,
135 status varchar(100) NOT NULL,
136 PRIMARY KEY (NIK),
137 CONSTRAINT Posyandu_has_kader FOREIGN KEY (ID_Posyandu)
138 REFERENCES Posyandu (ID_Posyandu) ON DELETE CASCADE ON UPDATE NO
139 ACTION
140 ) ENGINE=InnoDB DEFAULT CHARSET=latin1;
141
142 CREATE TABLE laporan_kehadiran_kader (
143 NIK_kader varchar(100) NOT NULL,
144 tgl_kehadiran date NOT NULL,
145 KEY_NIK_kader (NIK_kader),
146 CONSTRAINT laporan_kader FOREIGN KEY (NIK_kader) REFERENCES
147 kader (NIK) ON DELETE CASCADE ON UPDATE CASCADE
148 ) ENGINE=InnoDB DEFAULT CHARSET=latin1;

```

5.7 Implementasi Antarmuka

Pada bagian ini, akan dijabarkan mengenai hasil implementasi antarmuka berdasarkan hasil perancangan dan implementasi kode program yang telah dilakukan.

5.7.1 Implementasi Antarmuka *Dashboard User Posyandu*

Dalam Gambar 5.13 merupakan hasil implementasi antarmuka *dashboard* untuk *user* Posyandu, dimana terdapat 7 menu utama yang bisa diakses atau digunakan oleh *user* Posyandu, yaitu menu untuk menjadwalkan hari Posyandu, mencatat kehadiran dan hasil pelayanan kesehatan balita, menampilkan data akun orang tua, menampilkan data balita terdaftar, mendaftarkan akun baru orang tua, mendaftarkan balita baru, serta mendaftarkan kehadiran kader Posyandu.



Gambar 5.13 Implementasi antarmuka *dashboard user Posyandu*



5.7.2 Implementasi Antarmuka Pencatatan Kehadiran dan Pencatatan Hasil Pelayanan Kesehatan Balita

Dalam Gambar 5.14 merupakan hasil implementasi antarmuka untuk melakukan pencatatan kehadiran dan pencatatan hasil pelayanan kesehatan balita pada Posyandu dalam bentuk *modal pop-up*. *Modal* ini dapat diakses dengan menekan menu pencatatan kehadiran dan hasil pelayanan kesehatan balita yang berada pada *dashboard* utama user Posyandu, dimana pada *modal* ini terdapat kolom-kolom dalam sebuah *form* yang berkaitan dengan pencatatan hasil pelayanan kesehatan balita seperti berat badan, tinggi badan, lingkaran kepala, cara pengukuran, imunisasi, pemberian vitamin A, pemberian ASI, pemberian makanan tambahan, keluhan, maupun diagnosa penyakit balita tersebut.

The screenshot shows a web-based form titled "Pencatatan Kehadiran dan Hasil Pelayanan Kesehatan Balita". The form is organized into several sections:

- Header:** "SISTEM INFORMASI POSYANDU" and "Pencatatan Kehadiran dan Hasil Pelayanan Kesehatan Balita".
- Form Fields:**
 - Tanggal Pemeriksaan Kesehatan: 03/06/2020
 - Nama Balita: (input field)
 - Berat Badan (kg): (input field)
 - Tinggi Badan (cm): (input field)
 - Lingkar Kepala (cm): (input field)
 - Cara Pengukuran: (dropdown menu)
 - Pemberian Imunisasi: (dropdown menu)
 - Pemberian Vit. A: (dropdown menu)
 - Pemberian ASI: (dropdown menu)
 - Pemberian Makanan Tambahan: (dropdown menu)
 - Keluhan Kesehatan Balita: (input field)
 - Diagnosa: (input field)
- Buttons:** "SIMPAN" (Save) at the bottom right.

Gambar 5.14 Implementasi antarmuka *modal* untuk pencatatan kehadiran dan hasil pelayanan kesehatan balita

5.7.3 Implementasi Antarmuka Profil Balita

Dalam Gambar 5.15 merupakan hasil implementasi antarmuka untuk profil balita, dimana pada halaman ini akan ditampilkan dua informasi utama, pada bagian atas terdapat informasi mengenai balita tersebut seperti nama balita, jenis kelamin tanggal lahir, umur, maupun informasi mengenai status gizi balita tersebut berdasarkan hasil terakhir pemeriksaan kesehatan. Pada bagian ini, user Posyandu juga dapat melihat KMS atau Kartu Menuju Sehat untuk balita tersebut, maupun mengirimkan saran gizi kepada akun orang tua balita tersebut. Pada bagian bawah, terdapat informasi mengenai riwayat hasil pemeriksaan kesehatan balita tersebut yang disajikan dalam bentuk tabel dimana terdapat juga opsi untuk memodifikasi dan menghapus data riwayat hasil pemeriksaan tersebut.



Gambar 5.15 Implementasi antarmuka profil balita

5.7.4 Implementasi Antarmuka Halaman Kartu Menuju Sehat (KMS) Balita

Dalam Gambar 5.16, merupakan hasil implementasi antarmuka untuk KMS balita, dimana halaman ini ditampilkan juga dua informasi utama yang berada pada bagian atas terdapat grafik pertumbuhan berat badan balita yang dapat menampilkan hasil pengukuran berat badan balita setiap bulannya, serta standar antropometri berat badan balita mulai dari -3 SD, hingga $+3$ SD. Pada bagian bawah juga akan ditampilkan riwayat hasil pengukuran berat badan balita setiap bulannya yang disertai juga informasi mengenai KBM atau Kenaikan Berat Badan Minimal (dalam gram) yang harus dicapai oleh balita tersebut, serta indikator apakah berat badan balita tersebut naik atau tidak berdasarkan standar KBM tersebut.



Gambar 5.16 Implementasi antarmuka Kartu Menuju Sehat (KMS) balita

5.7.5 Implementasi Antarmuka Dashboard User Bidan

Dalam Gambar 5.17, berikut merupakan hasil implementasi antarmuka untuk *dashboard user* bidan, dimana terdapat 4 menu utama yang dapat digunakan oleh *user* bidan, yaitu melihat laporan Posyandu, mengelola data Posyandu dan kader yang telah terdaftar, mendaftarkan Posyandu baru, serta mendaftarkan kader baru.



Gambar 5.17 Implementasi antarmuka dashboard user bidan



5.7.6 Implementasi Antarmuka Laporan Gizi

Dalam Gambar 5.18 merupakan hasil implementasi antarmuka untuk laporan gizi yang bisa diakses oleh *user* bidan setelah menekan menu lihat laporan Posyandu pada *dashboard* utama. Pada halaman ini, terdapat pada bagian atas merupakan bagian *user* bisa melakukan seleksi sumber data laporan yang ingin ditampilkan, lalu terdapat 4 jenis laporan yang terdapat pada halaman ini, yaitu laporan balita pertumbuhan normal, yang akan menampilkan semua balita yang melakukan pemeriksaan pada bulan tersebut yang memiliki status gizi normal atau baik. Pada laporan kedua yaitu laporan balita malnutrisi, akan menampilkan data balita yang melakukan pemeriksaan pada bulan tersebut dan memiliki status gizi tidak normal atau mengalami malnutrisi. Serta pada bagian bawah terdapat dua laporan, yang pertama adalah laporan mengenai jumlah kehadiran balita yang melakukan pemeriksaan pada bulan tersebut dan jumlah balita yang terdaftar, serta terdapat laporan jumlah kehadiran kader pada bulan tersebut dan jumlah kader yang terdaftar.

The screenshot displays the 'Dashboard Bidan' interface for the 'SISTEM INFORMASI GIZI BALITA'. The main content area is titled 'Laporan Gizi Balita dan Pelayanan Posyandu' and includes a date filter for 'Desember 2019'. Below this, there are four distinct report sections:

- Laporan Balita Pertumbuhan Normal:** A table listing children with columns for Name, Gender, Age, Weight (BB), Height (TB), Length (LK), and BMI (BB/TB). It includes a search bar and an 'Export Excel Laporan Malnutrisi' button.
- Laporan Balita Malnutrisi:** A table listing children with columns for Name, Gender, Age, Weight (BB), Height (TB), Length (LK), and BMI (BB/TB). It includes a search bar and an 'Export Excel Laporan Malnutrisi' button.
- Laporan Jumlah Kehadiran Balita ke Posyandu:** A bar chart comparing 'Jumlah Balita Terdaftar' (Registered Children) and 'Jumlah Balita Terhadir' (Present Children).
- Laporan Kehadiran Kader Posyandu:** A bar chart comparing 'Kader Hadir' (Present Staff) and 'Jumlah Kader Terdaftar' (Registered Staff).

Gambar 5.18 Implementasi antarmuka laporan gizi



BAB 6 PENGUJIAN SISTEM

Setelah dilakukan proses perancangan dan implementasi, maka pada bagian terakhir ini akan dilakukan proses pengujian sistem yang bertujuan untuk memastikan bahwa sistem yang telah dibuat sudah memenuhi semua kebutuhan yang telah didefinisikan, serta dapat diterima dengan baik oleh pengguna / *stakeholders*. Proses pengujian sistem ini akan menggunakan pengujian validasi dan *user acceptance testing* (UAT).

6.1 Pengujian Validasi

Pengujian validasi ini akan dilakukan dengan teknik pengujian *black box* yaitu dengan memberikan serangkaian masukan pada sistem dan mengamati hasil keluarannya untuk memastikan bahwa fungsionalitas sistem yang telah terdefiniskan, sudah sesuai dan bekerja dengan semestinya pada sistem yang telah diimplementasikan. Proses pengujian validasi ini akan dilakukan berdasarkan *use case scenario*, dan akan dinyatakan berhasil atau *valid* apabila *output* atau hasil yang diberikan oleh sistem dalam suatu kasus uji sudah sesuai dengan ekspektasi dari fungsionalitas tersebut. Pada Tabel 6.1 merupakan hasil pengujian untuk kasus uji *login* dengan skenario tes *main flow*. Hasil pengujian kasus uji *login* ini mendapatkan hasil *valid*.

Tabel 6.1 Pengujian validasi *login*

Kode Kebutuhan	KF-GIZI-001
Nama Kasus Uji	<i>Login</i>
Skenario Tes	Skenario <i>Main Flow</i>
Prosedur Uji	<ol style="list-style-type: none"> 1. Membuka halaman login. 2. Memasukkan <i>username</i> dan <i>password</i> yang sudah ada pada basis data. 3. Menekan tombol "MASUK".
Hasil Yang Diharapkan	Sistem berhasil mengautentikasi pengguna dan menampilkan halaman <i>dashboard</i> sesuai dengan jenis <i>user</i> .
Hasil	Sistem berhasil mengautentikasi pengguna dan menampilkan halaman <i>dashboard</i> sesuai dengan jenis <i>user</i> .
Status	Valid

Pada Tabel 6.2 merupakan hasil pengujian validasi untuk kasus uji *login* dengan skenario tes *alternative flow* (II). Hasil pengujian kasus uji *login* alternatif ini mendapatkan hasil *valid*.

Tabel 6.2 Pengujian validasi *login* alternatif

Kode Kebutuhan	KF-GIZI-001
Nama Kasus Uji	Login
Skenario Tes	Skenario <i>Alternative Flow</i> (II) Autentikasi gagal
Prosedur Uji	<ol style="list-style-type: none"> 1. Membuka halaman login. 2. Memasukkan <i>username</i> dan <i>password</i> yang belum ada pada basis data. 3. Menekan tombol "MASUK".
Hasil Yang Diharapkan	Sistem menampilkan kembali halaman login dengan pesan " <i>username</i> atau <i>password</i> salah!".
Hasil	Sistem menampilkan kembali halaman login dengan pesan " <i>username</i> atau <i>password</i> salah!".
Status	Valid

Pada Tabel 6.3 merupakan hasil pengujian validasi untuk kasus uji registrasi bidan dengan skenario tes *main flow*. Hasil pengujian pada kasus uji registrasi bidan ini mendapatkan hasil *valid*.

Tabel 6.3 Pengujian validasi registrasi bidan

Kode Kebutuhan	KF-GIZI-002
Nama Kasus Uji	Registrasi Bidan
Skenario Tes	Skenario <i>Main Flow</i>
Prosedur Uji	<ol style="list-style-type: none"> 1. Mengisi semua kolom pada <i>form</i> pendaftaran bidan dengan benar. 2. Menekan tombol "Choose file" untuk mengunggah surat registrasi bidan dengan format PDF. 3. Menekan tombol "Choose file" untuk mengunggah foto profil dengan format JPG atau PNG. 4. Menekan tombol "REGISTRASI".
Hasil Yang Diharapkan	Sistem mengarahkan kembali pada halaman <i>login</i> dengan menampilkan <i>pop-up</i> "Pendaftaran berhasil, data berhasil disimpan".
Hasil	Sistem mengarahkan kembali pada halaman <i>login</i> dengan menampilkan <i>pop-up</i> "Pendaftaran berhasil, data berhasil disimpan".
Status	Valid

Pada Tabel 6.4 merupakan hasil pengujian validasi untuk kasus uji registrasi bidan dengan skenario tes *alternative flow* (IV). Hasil pengujian pada kasus uji registrasi bidan alternatif ini mendapatkan hasil *valid*.



Tabel 6.4 Pengujian validasi registrasi badan alternatif

Kode Kebutuhan	KF-GIZI-002
Nama Kasus Uji	Registrasi Badan
Skenario Tes	Skenario <i>Alternative flow</i> (IV) <i>Username</i> atau nomor registrasi ganda.
Prosedur Uji	<ol style="list-style-type: none"> 1. Mengisi semua kolom pada <i>form</i> pendaftaran badan dengan data yang sudah ada pada basis data. 2. Menekan tombol “Choose file” untuk mengunggah surat registrasi badan dengan format PDF. 3. Menekan tombol “Choose file” untuk mengunggah foto profil dengan format JPG atau PNG. 4. Menekan tombol “REGISTRASI”.
Hasil Yang Diharapkan	Sistem menampilkan <i>pop-up</i> dengan pesan “Pendaftaran gagal, nomor registrasi / <i>username</i> sudah terdaftar”
Hasil	Sistem menampilkan <i>pop-up</i> dengan pesan “Pendaftaran gagal, nomor registrasi / <i>username</i> sudah terdaftar”
Status	Valid

Pada Tabel 6.5 merupakan hasil pengujian validasi untuk kasus uji menampilkan data badan aktif dengan skenario tes *main flow*. Hasil pengujian pada kasus uji menampilkan data badan aktif ini mendapatkan hasil *valid*.

Tabel 6.5 Pengujian validasi menampilkan data badan aktif

Kode Kebutuhan	KF-GIZI-003
Nama Kasus Uji	Menampilkan data badan aktif
Skenario Tes	Skenario <i>main flow</i>
Prosedur Uji	<ol style="list-style-type: none"> 1. Memilih menu “Lihat badan aktif”
Hasil Yang Diharapkan	Sistem menampilkan data badan aktif dalam bentuk tabel
Hasil	Sistem menampilkan data badan aktif dalam bentuk tabel
Status	Valid

Pada Tabel 6.6 merupakan hasil pengujian validasi untuk kasus uji memperbaharui data badan aktif dengan skenario tes *main flow*. Hasil pengujian pada kasus uji memperbaharui data badan aktif ini mendapatkan hasil *valid*.



Tabel 6.6 Pengujian validasi memperbaharui data badan aktif

Kode Kebutuhan	KF-GIZI-004
Nama Kasus Uji	Memperbaharui data badan aktif
Skenario Tes	Skenario <i>main flow</i>
Prosedur Uji	<ol style="list-style-type: none"> 1. Menekan tombol "edit" pada salah satu data badan aktif. 2. Melakukan perubahan data pada kolom nama badan. 3. Menekan tombol "Simpan".
Hasil Yang Diharapkan	Sistem menyimpan perubahan data badan aktif dan menampilkan <i>pop up</i> "Edit data berhasil".
Hasil	Sistem menyimpan perubahan data badan aktif dan menampilkan <i>pop up</i> "Edit data berhasil".
Status	Valid

Pada Tabel 6.7 merupakan hasil pengujian validasi untuk kasus uji memperbaharui data badan aktif dengan skenario tes *alternative flow (I)*. Hasil pengujian pada kasus uji memperbaharui data badan aktif alternatif ini mendapatkan hasil *valid*.

Tabel 6.7 Pengujian validasi memperbaharui data badan aktif alternatif

Kode Kebutuhan	KE-GIZI-004
Nama Kasus Uji	Memperbaharui data badan aktif
Skenario Tes	<p>Skenario <i>alternative flow (I)</i></p> <p>Mengosongkan salah satu kolom</p> <ol style="list-style-type: none"> 1. Menekan tombol "edit" pada salah satu data badan aktif 2. Mengosongkan kolom nama badan 3. Menekan tombol "Simpan"
Hasil Yang Diharapkan	Sistem menampilkan pesan "Please fill out this field" pada kolom nama badan.
Hasil	Sistem menampilkan pesan "Please fill out this field" pada kolom nama badan.
Status	Valid

Pada Tabel 6.8 merupakan hasil pengujian validasi untuk kasus uji menghapus data badan aktif dengan skenario tes *main flow*. Hasil pengujian pada kasus uji menghapus data badan aktif ini mendapatkan hasil *valid*.



Tabel 6.8 Pengujian validasi menghapus data badan aktif

Kode Kebutuhan	KF-GIZI-005
Nama Kasus Uji	Menghapus data badan aktif
Skenario Tes	Skenario <i>main flow</i>
Prosedur Uji	<ol style="list-style-type: none"> 1. Menekan tombol “hapus” pada salah satu data badan aktif. 2. Mengonfirmasi penonaktifan data dengan menekan tombol “hapus” pada <i>pop up</i> konfirmasi yang ditampilkan.
Hasil Yang Diharapkan	Sistem mengubah status badan aktif tersebut menjadi “Tidak Aktif” dan menampilkan <i>pop up</i> “Penonaktifan data berhasil”
Hasil	Sistem mengubah status badan aktif tersebut menjadi “Tidak Aktif” dan menampilkan <i>pop up</i> “Penonaktifan data berhasil”
Status	Valid

Pada Tabel 6.9 merupakan hasil pengujian validasi untuk kasus uji menghapus data badan aktif dengan skenario tes *alternative flow* (I). Hasil pengujian pada kasus uji menghapus data badan aktif alternatif ini mendapatkan hasil *valid*.

Tabel 6.9 Pengujian validasi menghapus data badan aktif alternatif

Kode Kebutuhan	KF-GIZI-005
Nama Kasus Uji	Menghapus data badan aktif
Skenario Tes	<p>Skenario <i>alternative flow</i> (I)</p> <p>Membatalkan penghapusan data</p> <ol style="list-style-type: none"> 1. Menekan tombol “hapus” pada salah satu data badan aktif. 2. Membatalkan penonaktifan data dengan menekan tombol “batalkan” pada <i>pop up</i> konfirmasi yang ditampilkan.
Hasil Yang Diharapkan	Sistem menampilkan kembali halaman data badan aktif.
Hasil	Sistem menampilkan kembali halaman data badan aktif.
Status	Valid

Pada Tabel 6.10 merupakan hasil pengujian validasi untuk kasus uji menampilkan data badan yang perlu divalidasi dengan skenario tes *main flow*. Hasil pengujian pada kasus uji menampilkan data badan yang perlu divalidasi ini mendapatkan hasil *valid*.



Tabel 6.10 Pengujian validasi menampilkan data bidang yang perlu divalidasi

Kode Kebutuhan	KF-GIZI-006
Nama Kasus Uji	Menampilkan data bidang yang perlu divalidasi
Skenario Tes	Skenario <i>main flow</i>
Prosedur Uji	1. Memilih menu "Validasi pendaftaran bidang"
Hasil Yang Diharapkan	Sistem menampilkan data bidang yang perlu divalidasi dalam bentuk tabel.
Hasil	Sistem menampilkan data bidang yang perlu divalidasi dalam bentuk tabel.
Status	Valid

Pada

Tabel 6.11, merupakan hasil pengujian validasi untuk kasus uji memvalidasi data bidang dengan skenario tes *main flow*. Hasil pengujian pada kasus uji memvalidasi data bidang ini mendapatkan hasil *valid*.

Tabel 6.11 Pengujian validasi memvalidasi data bidang

Kode Kebutuhan	KF-GIZI-007
Nama Kasus Uji	Memvalidasi data bidang
Skenario Tes	Skenario <i>main flow</i>
Prosedur Uji	1. Menekan tombol "detail" pada salah satu data bidang yang ingin divalidasi 2. Menekan tombol "validasi" pada <i>pop up</i> detail data bidang yang ditampilkan oleh sistem
Hasil Yang Diharapkan	Sistem mengubah status bidang tersebut menjadi "aktif" dan menampilkan <i>pop up</i> "Validasi data berhasil".
Hasil	Sistem mengubah status bidang tersebut menjadi "aktif" dan menampilkan <i>pop up</i> "Validasi data berhasil".
Status	Valid

Pada Tabel 6.12, merupakan hasil pengujian validasi untuk kasus uji menjadwalkan hari Posyandu dengan skenario tes *main flow*. Hasil pengujian pada kasus uji menjadwalkan hari Posyandu ini mendapatkan hasil *valid*.



Tabel 6.12 Pengujian validasi menjadwalkan hari Posyandu

Kode Kebutuhan	KF-GIZI-008
Nama Kasus Uji	Menjadwalkan hari Posyandu
Skenario Tes	Skenario <i>main flow</i>
Prosedur Uji	<ol style="list-style-type: none"> Memilih menu "Penjadwalan hari Posyandu" pada <i>dashboard</i>. Mengisi <i>form</i> penjadwalan hari Posyandu dengan benar. Menekan tombol "jadwalkan".
Hasil Yang Diharapkan	Sistem menyimpan dan mengirimkan hasil penjadwalan hari Posyandu ke akun orang tua yang terdaftar pada Posyandu tersebut, serta menampilkan <i>pop up</i> "Penjadwalan Berhasil".
Hasil	Sistem menyimpan dan mengirimkan hasil penjadwalan hari Posyandu ke akun orang tua yang terdaftar pada Posyandu tersebut, serta menampilkan <i>pop up</i> "Penjadwalan Berhasil".
Status	Valid

Pada Tabel 6.13 merupakan hasil pengujian validasi untuk kasus uji menjadwalkan hari Posyandu dengan skenario tes *alternative flow* (II). Hasil pengujian pada kasus uji menjadwalkan hari Posyandu alternatif ini mendapatkan hasil *valid*.

Tabel 6.13 Pengujian validasi menjadwalkan hari Posyandu alternatif

Kode Kebutuhan	KF-GIZI-008
Nama Kasus Uji	Menjadwalkan hari Posyandu
Skenario Tes	Skenario <i>alternative flow</i> (II) Tanggal Posyandu ada di masa lampau
Prosedur Uji	<ol style="list-style-type: none"> Memilih menu "Penjadwalan hari Posyandu" pada <i>dashboard</i>. Mengisi <i>form</i> penjadwalan hari Posyandu dengan tanggal yang berada di masa lampau (tanggal sebelum dilakukannya penjadwalan hari Posyandu). Menekan tombol "jadwalkan".
Hasil Yang Diharapkan	Sistem menampilkan <i>pop up</i> "Penjadwalan gagal".
Hasil	Sistem menampilkan <i>pop up</i> "Penjadwalan gagal".
Status	Valid

Pada Tabel 6.14 merupakan hasil pengujian validasi untuk kasus uji mendaftarkan akun orang tua balita dengan skenario tes *main flow*. Hasil pengujian pada kasus uji mendaftarkan akun orang tua balita ini mendapatkan hasil *valid*.



Tabel 6.14 Pengujian validasi mendaftarkan akun orang tua balita

Kode Kebutuhan	KF-GIZI-009
Nama Kasus Uji	Mendaftarkan akun orang tua balita
Skenario Tes	Skenario <i>main flow</i>
Prosedur Uji	<ol style="list-style-type: none"> Memilih menu “Pendaftaran Akun Baru Orang Tua” pada <i>dashboard</i>. Mengisi <i>form</i> pendaftaran akun orang tua dengan data pendaftaran yang belum ada pada basis data. Menekan tombol “Daftar”.
Hasil Yang Diharapkan	Sistem menyimpan data akun orang tua balita tersebut dan menampilkan <i>pop up</i> “Daftar akun orang tua berhasil”.
Hasil	Sistem menyimpan data akun orang tua balita tersebut dan menampilkan <i>pop up</i> “Daftar akun orang tua berhasil”.
Status	Valid

Pada Tabel 6.15 merupakan hasil pengujian validasi untuk kasus uji mendaftarkan akun orang tua balita dengan skenario tes *alternative flow* (I). Hasil pengujian pada kasus uji mendaftarkan akun orang tua balita alternatif ini mendapatkan hasil *valid*.

Tabel 6.15 Pengujian validasi mendaftarkan akun orang tua balita alternatif

Kode Kebutuhan	KF-GIZI-009
Nama Kasus Uji	Mendaftarkan akun orang tua balita
Skenario Tes	Skenario <i>alternative flow</i> (I) Data <i>username</i> , NIK, No. KK, atau No. HP ganda
Prosedur Uji	<ol style="list-style-type: none"> Memilih menu “Pendaftaran Akun Baru Orang Tua” pada <i>dashboard</i>. Mengisi <i>form</i> pendaftaran dengan mengisi pada kolom <i>username</i> dengan data yang sudah ada pada basis data Menekan tombol “Daftar”
Hasil Yang Diharapkan	Sistem menampilkan <i>pop up</i> “Pendaftaran gagal, data sudah terdaftar”.
Hasil	Sistem menampilkan <i>pop up</i> “Pendaftaran gagal, data sudah terdaftar”.
Status	Valid

Pada Tabel 6.16 merupakan hasil pengujian validasi untuk kasus uji menampilkan data orang tua balita yang terdaftar dengan skenario tes *main flow*. Hasil pengujian pada kasus uji menampilkan data orang tua yang terdaftar ini mendapatkan hasil *valid*.



Tabel 6.16 Pengujian validasi menampilkan data orang tua yang terdaftar

Kode Kebutuhan	KF-GIZI-010
Nama Kasus Uji	Menampilkan data orang tua yang terdaftar
Skenario Tes	Skenario <i>main flow</i>
Prosedur Uji	1. Memilih menu "Lihat data akun orang tua"
Hasil Yang Diharapkan	Sistem menampilkan data orang tua yang terdaftar dalam bentuk tabel.
Hasil	Sistem menampilkan data orang tua yang terdaftar dalam bentuk tabel.
Status	Valid

Pada Tabel 6.17 merupakan hasil pengujian validasi untuk kasus uji memperbaharui data orang tua balita yang terdaftar dengan skenario tes *main flow*. Hasil pengujian pada kasus uji memperbaharui data orang tua balita yang terdaftar ini mendapatkan hasil *valid*.

Tabel 6.17 Pengujian validasi memperbaharui data orang tua balita yang terdaftar

Kode Kebutuhan	KF-GIZI-011
Nama Kasus Uji	Memperbaharui data orang tua balita yang terdaftar
Skenario Tes	Skenario <i>main flow</i>
Prosedur Uji	1. Menekan tombol "edit" pada salah satu data orang tua balita yang terdaftar 2. Melakukan perubahan data pada nama ayah 3. Menekan tombol "Simpan".
Hasil Yang Diharapkan	Sistem menyimpan perubahan data orang tua balita dan menampilkan <i>pop up</i> "Edit data orang tua berhasil"
Hasil	Sistem menyimpan perubahan data orang tua balita dan menampilkan <i>pop up</i> "Edit data orang tua berhasil"
Status	Valid

Pada Tabel 6.18 merupakan hasil pengujian validasi untuk kasus uji memperbaharui data orang tua balita yang terdaftar dengan skenario tes *alternative flow (I)*. Hasil pengujian pada kasus uji memperbaharui data orang tua balita yang terdaftar alternatif ini mendapatkan hasil *valid*.



Tabel 6.18 Pengujian validasi memperbaharui data orang tua balita yang terdaftar alternatif

Kode Kebutuhan	KF-GIZI-011
Nama Kasus Uji	Memperbaharui data orang tua balita yang terdaftar
Skenario Tes	Skenario <i>alternative flow</i> (I) Mengosongkan salah satu kolom
Prosedur Uji	<ol style="list-style-type: none"> 1. Menekan tombol “edit” pada salah satu data orang tua balita yang terdaftar pada tabel. 2. Mengosongkan kolom alamat rumah. 3. Menekan tombol “Simpan”
Hasil Yang Diharapkan	Sistem menampilkan pesan “Please fill out this field” pada kolom alamat rumah.
Hasil	Sistem menampilkan pesan “Please fill out this field” pada kolom alamat rumah.
Status	Valid

Pada Tabel 6.19 merupakan hasil pengujian validasi untuk kasus uji menghapus data orang tua balita yang terdaftar dengan skenario tes *main flow*. Hasil pengujian pada kasus uji menghapus data orang tua balita yang terdaftar ini mendapatkan hasil *valid*.

Tabel 6.19 Pengujian validasi menghapus data orang tua balita yang terdaftar

Kode Kebutuhan	KF-GIZI-012
Nama Kasus Uji	Menghapus data orang tua balita yang terdaftar
Skenario Tes	Skenario <i>main flow</i>
Prosedur Uji	<ol style="list-style-type: none"> 1. Menekan tombol “hapus” pada salah satu data orang tua balita 2. Mengonfirmasi penonaktifan data dengan menekan tombol “hapus” pada <i>pop up</i> konfirmasi yang ditampilkan.
Hasil Yang Diharapkan	Sistem menonaktifkan orang tua balita tersebut dan menampilkan <i>pop up</i> “Hapus data orang tua berhasil”
Hasil	Sistem menonaktifkan orang tua balita tersebut dan menampilkan <i>pop up</i> “Hapus data orang tua berhasil”
Status	Valid

Pada Tabel 6.20 merupakan hasil pengujian validasi untuk kasus uji menghapus data orang tua balita yang terdaftar dengan skenario tes *alternative flow* (I). Hasil pengujian pada kasus uji menghapus data orang tua balita yang terdaftar alternatif ini mendapatkan hasil *valid*.



Tabel 6.20 Pengujian validasi menghapus data orang tua balita yang terdaftar alternatif

Kode Kebutuhan	KF-GIZI-012
Nama Kasus Uji	Menghapus data orang tua balita yang terdaftar
Skenario Tes	Skenario <i>alternative flow</i> (I) Membatalkan penghapusan data
Prosedur Uji	1. Menekan tombol “hapus” pada salah satu data orang tua balita. 2. Membatalkan peronaktifan data dengan menekan tombol “batal” pada <i>pop up</i> konfirmasi yang ditampilkan.
Hasil Yang Diharapkan	Sistem menampilkan kembali halaman data orang tua balita yang terdaftar.
Hasil	Sistem menampilkan kembali halaman data orang tua balita yang terdaftar.
Status	Valid

Pada Tabel 6.21 merupakan hasil pengujian validasi untuk kasus uji mendaftarkan data balita baru dengan skenario tes *main flow*. Hasil pengujian pada kasus mendaftarkan data balita baru ini mendapatkan hasil *valid*.

Tabel 6.21 Pengujian validasi mendaftarkan data balita baru

Kode Kebutuhan	KF-GIZI-013
Nama Kasus Uji	Mendaftarkan data balita baru
Skenario Tes	Skenario <i>main flow</i> 1. Memilih menu “Pendaftaran Balita Baru” pada <i>dashboard</i> Posyandu.
Prosedur Uji	2. Mengisi form pendaftaran balita baru menggunakan data balita yang belum ada pada basis data dengan lengkap dan benar 3. Menekan tombol “Daftar”
Hasil Yang Diharapkan	Sistem menyimpan data balita tersebut dan menampilkan <i>pop up</i> “Pendaftaran Balita Berhasil”
Hasil	Sistem menyimpan data balita tersebut dan menampilkan <i>pop up</i> “Pendaftaran Balita Berhasil”
Status	Valid

Pada Tabel 6.22 merupakan hasil pengujian validasi untuk kasus uji mendaftarkan data balita baru dengan skenario tes *alternative flow* (I). Hasil pengujian pada kasus uji mendaftarkan data balita alternatif ini mendapatkan hasil *valid*.



Tabel 6.22 Pengujian validasi mendaftarkan data balita baru alternatif

Kode Kebutuhan	KF-GIZI-013
Nama Kasus Uji	Mendaftarkan data balita baru
Skenario Tes	Skenario <i>alternative flow</i> (!) Umur balita tidak sesuai
Prosedur Uji	<ol style="list-style-type: none"> Memilih menu "Pendaftaran Balita Baru" pada <i>dashboard</i> Posyandu. Mengisi form pendaftaran balita baru dengan data tanggal lahir yang melebihi 60 bulan dari tanggal dilakukannya pendaftaran balita tersebut. Menekan tombol "Daftar".
Hasil Yang Diharapkan	Sistem menampilkan pesan "Pendaftaran balita gagal, periksa kembali data pendaftaran".
Hasil	Sistem menampilkan pesan "Pendaftaran balita gagal, periksa kembali data pendaftaran".
Status	Valid

Pada Tabel 6.23 merupakan hasil pengujian validasi untuk kasus uji menampilkan data balita yang terdaftar dengan skenario tes *main flow*. Hasil pengujian pada kasus uji menampilkan data balita yang terdaftar ini mendapatkan hasil *valid*.

Tabel 6.23 Pengujian validasi menampilkan data balita yang terdaftar

Kode Kebutuhan	KF-GIZI-014
Nama Kasus Uji	Menampilkan data balita yang terdaftar
Skenario Tes	Skenario <i>main flow</i>
Prosedur Uji	<ol style="list-style-type: none"> Memilih menu "Lihat data balita terdaftar"
Hasil Yang Diharapkan	Sistem menampilkan data balita yang terdaftar dalam bentuk tabel.
Hasil	Sistem menampilkan data balita yang terdaftar dalam bentuk tabel.
Status	Valid

Pada Tabel 6.24 merupakan hasil pengujian validasi untuk kasus uji memperbaharui data balita yang terdaftar dengan skenario tes *main flow*. Hasil pengujian pada kasus uji memperbaharui data balita yang terdaftar ini mendapatkan hasil *valid*.



Tabel 6.24 Pengujian validasi memperbaharui data balita yang terdaftar

Kode Kebutuhan	KF-GIZI-015
Nama Kasus Uji	Memperbaharui data balita yang terdaftar.
Skenario Tes	Skenario <i>main flow</i>
Prosedur Uji	<ol style="list-style-type: none"> 1. Menekan tombol “edit” pada salah satu data balita yang terdaftar. 2. Melakukan perubahan pada data NIK balita. 3. Menekan tombol “Simpan”.
Hasil Yang Diharapkan	Sistem menyimpan perubahan data balita dan menampilkan <i>pop up</i> “Edit data balita berhasil”.
Hasil	Sistem menyimpan perubahan data balita dan menampilkan <i>pop up</i> “Edit data balita berhasil”.
Status	Valid

Pada Tabel 6.25 merupakan hasil pengujian validasi untuk kasus uji memperbaharui data balita yang terdaftar dengan skenario tes *alternative flow* (I). Hasil pengujian pada kasus uji memperbaharui data balita yang terdaftar alternatif ini mendapatkan hasil *valid*.

Tabel 6.25 Pengujian validasi memperbaharui data balita yang terdaftar alternatif

Kode Kebutuhan	KF-GIZI-011
Nama Kasus Uji	Memperbaharui data balita yang terdaftar
Skenario Tes	<p>Skenario <i>alternative flow</i> (I)</p> <p>Umur balita tidak sesuai</p>
Prosedur Uji	<ol style="list-style-type: none"> 1. Menekan tombol “edit” pada salah satu data balita yang terdaftar pada tabel. 2. Mengubah tanggal lahir balita dengan tanggal lahir melebihi 60 bulan dari tanggal dilakukannya perubahan data balita. 3. Menekan tombol “Simpan”
Hasil Yang Diharapkan	Sistem menampilkan pesan “Edit data balita gagal”.
Hasil	Sistem menampilkan pesan “Edit data balita gagal”.
Status	Valid

Pada Tabel 6.26 merupakan hasil pengujian validasi untuk kasus uji menghapus data balita yang terdaftar dengan skenario tes *main flow*. Hasil pengujian pada kasus uji menghapus data balita yang terdaftar ini mendapatkan hasil *valid*.



Tabel 6.26 Pengujian validasi menghapus data balita yang terdaftar

Kode Kebutuhan	KF-GIZI-016
Nama Kasus Uji	Menghapus data balita yang terdaftar
Skenario Tes	Skenario <i>main flow</i>
Prosedur Uji	<ol style="list-style-type: none"> 1. Menekan tombol “hapus” pada salah satu data balita pada tabel 2. Mengonfirmasi penonaktifan data dengan menekan tombol “hapus” pada <i>pop up</i> konfirmasi yang ditampilkan.
Hasil Yang Diharapkan	Sistem menonaktifkan balita tersebut dan menampilkan <i>pop up</i> “Penonaktifan data balita berhasil”.
Hasil	Sistem menonaktifkan data balita tersebut dan menampilkan <i>pop up</i> “Penonaktifan data balita berhasil”.
Status	Valid

Pada Tabel 6.27 merupakan hasil pengujian validasi untuk kasus uji menghapus data balita yang terdaftar dengan skenario tes *alternative flow* (I). Hasil pengujian pada kasus uji menghapus data balita yang terdaftar alternatif ini mendapatkan hasil *valid*.

Tabel 6.27 Pengujian validasi menghapus data balita yang terdaftar alternatif

Kode Kebutuhan	KF-GIZI-016
Nama Kasus Uji	Menghapus data balita yang terdaftar
Skenario Tes	<p>Skenario <i>alternative flow</i> (I)</p> <p>Membatalkan penghapusan data</p> <ol style="list-style-type: none"> 1. Menekan tombol “hapus” pada salah satu data balita pada tabel. 2. Membatalkan penonaktifan data dengan menekan tombol “batal” pada <i>pop up</i> konfirmasi yang ditampilkan.
Hasil Yang Diharapkan	Sistem menampilkan kembali halaman data balita yang terdaftar
Hasil	Sistem menampilkan kembali halaman data balita yang terdaftar
Status	Valid

Pada Tabel 6.28 merupakan hasil pengujian validasi untuk kasus uji mendata kehadiran kader Posyandu dengan skenario tes *main flow*. Hasil pengujian pada kasus uji mendata kehadiran kader Posyandu ini mendapatkan hasil *valid*.



Tabel 6.28 Pengujian validasi mendata kehadiran kader Posyandu

Kode Kebutuhan	KF-GIZI-017
Nama Kasus Uji	Mendata kehadiran kader Posyandu
Skenario Tes	Skenario <i>main flow</i>
Prosedur Uji	<ol style="list-style-type: none"> Memilih menu "Pendataan kehadiran kader" Mengisi form pendataan dengan data tanggal hari Posyandu dan nama kader yang ingin didata kehadirannya. Menekan tombol "Simpan"
Hasil Yang Diharapkan	Sistem menampilkan pesan "Pendataan kehadiran kader berhasil"
Hasil	Sistem menampilkan pesan "Pendataan kehadiran kader berhasil"
Status	Valid

Pada Tabel 6.29 merupakan hasil pengujian validasi untuk kasus uji mendata kehadiran kader Posyandu dengan skenario tes *alternative flow* (I). Hasil pengujian pada kasus uji mendata kehadiran kader Posyandu alternatif ini mendapatkan hasil *valid*.

Tabel 6.29 Pengujian validasi mendata kehadiran kader Posyandu alternatif

Kode Kebutuhan	KF-GIZI-017
Nama Kasus Uji	Mendata kehadiran kader Posyandu
Skenario Tes	Skenario <i>alternative flow</i> (I) Data kehadiran kader ganda
Prosedur Uji	<ol style="list-style-type: none"> Memilih menu "Pendataan kehadiran kader" Mengisi form pendataan kehadiran kader dengan data tanggal dan nama kader yang sudah didata pada bulan dan tahun yang sama sebelumnya. Menekan tombol "Simpan".
Hasil Yang Diharapkan	Sistem menampilkan pesan "Pendataan kehadiran kader gagal".
Hasil	Sistem menampilkan pesan "Pendataan kehadiran kader gagal".
Status	Valid

Pada Tabel 6.30 merupakan hasil pengujian validasi untuk kasus uji mencatat kehadiran dan hasil pelayanan kesehatan balita dengan skenario tes *main flow*. Hasil pengujian pada kasus uji mencatat kehadiran dan hasil pelayanan kesehatan balita ini mendapatkan hasil *valid*.



Tabel 6.30 Pengujian validasi mencatat kehadiran dan hasil pelayanan kesehatan balita

Kode Kebutuhan	KF-GIZI-018
Nama Kasus Uji	Mendata kehadiran kader Posyandu
Skenario Tes	Skenario <i>main flow</i>
Prosedur Uji	<ol style="list-style-type: none"> Memilih menu "Pencatatan kehadiran dan hasil pelayanan kesehatan balita". Mengisi form pencatatan kehadiran dan hasil pelayanan kesehatan balita dengan lengkap dan benar. Menekan tombol "Simpan".
Hasil Yang Diharapkan	Sistem menampilkan pesan "Pencatatan hasil pemeriksaan kesehatan balita berhasil".
Hasil	Sistem menampilkan pesan "Pencatatan hasil pemeriksaan kesehatan balita berhasil".
Status	Valid

Pada Tabel 6.31 merupakan hasil pengujian validasi untuk kasus uji mencatat kehadiran dan hasil pelayanan kesehatan balita dengan skenario tes *alternative flow* (II). Hasil pengujian pada kasus uji mencatat kehadiran dan hasil pelayanan kesehatan balita alternatif ini mendapatkan hasil *valid*.

Tabel 6.31 Pengujian validasi mencatat kehadiran dan hasil pelayanan kesehatan balita alternatif

Kode Kebutuhan	KF-GIZI-018
Nama Kasus Uji	Mendata kehadiran kader Posyandu
Skenario Tes	<p>Skenario <i>alternative flow</i> (II)</p> <p>Data umur balita saat dilakukan pelayanan kesehatan ganda.</p> <ol style="list-style-type: none"> Memilih menu "Pencatatan kehadiran dan hasil pelayanan kesehatan balita". Mengisi form pencatatan kehadiran dan hasil pelayanan kesehatan balita dengan lengkap namun menggunakan data tanggal pemeriksaan dan nama balita yang sama dengan sebelumnya. Menekan tombol "Simpan".
Hasil Yang Diharapkan	Sistem menampilkan pesan "Pencatatan kesehatan balita gagal, periksa kembali data".
Hasil	Sistem menampilkan pesan "Pencatatan kesehatan balita gagal, periksa kembali data".
Status	Valid



Pada Tabel 6.32 merupakan hasil pengujian validasi untuk kasus uji mendaftarkan Posyandu baru skenario tes *main flow*. Hasil pengujian pada kasus uji mendaftarkan Posyandu baru ini mendapatkan hasil *valid*.

Tabel 6.32 Pengujian validasi mendaftarkan Posyandu baru

Kode Kebutuhan	KF-GIZI-019
Nama Kasus Uji	Mendaftarkan Posyandu baru
Skenario Tes	Skenario <i>main flow</i>
Prosedur Uji	<ol style="list-style-type: none"> Memilih menu "Pendaftaran Posyandu baru". Mengisi form pendaftaran Posyandu baru dengan data yang lengkap dan benar. Menekan tombol "Daftar".
Hasil Yang Diharapkan	Sistem menampilkan pesan "Pendaftaran Posyandu berhasil".
Hasil	Sistem menampilkan pesan "Pendaftaran Posyandu berhasil".
Status	Valid

Pada Tabel 6.33 merupakan hasil pengujian validasi untuk kasus uji pendaftaran Posyandubarur dengan skenario tes *alternative flow (I)*. Hasil pengujian pada kasus uji pendaftaran Posyandu baru alternatif ini mendapatkan hasil *valid*.

Tabel 6.33 Pengujian validasi pendaftaran Posyandu baru alternatif

Kode Kebutuhan	KF-GIZI-019
Nama Kasus Uji	Pendaftaran Posyandu baru
Skenario Tes	Skenario <i>alternative flow (I)</i> Data <i>username</i> ganda
Prosedur Uji	<ol style="list-style-type: none"> Memilih menu "Pendaftaran Posyandu baru". Mengisi form pendaftaran Posyandu baru dengan lengkap namun menggunakan data <i>username</i> yang sudah terdaftar sebelumnya pada basis data. Menekan tombol "Daftar".
Hasil Yang Diharapkan	Sistem menampilkan pesan "Pendaftaran Posyandu gagal".
Hasil	Sistem menampilkan pesan "Pendaftaran Posyandu gagal".
Status	Valid

Pada Tabel 6.34 merupakan hasil pengujian validasi untuk kasus uji mendaftarkan kader Posyandu skenario tes *main flow*. Hasil pengujian pada kasus uji mendaftarkan kader Posyandu ini mendapatkan hasil *valid*.



Tabel 6.34 Pengujian validasi mendaftarkan kader Posyandu

Kode Kebutuhan	KF-GIZI-020
Nama Kasus Uji	Mendaftarkan kader baru
Skenario Tes	Skenario <i>main flow</i>
Prosedur Uji	<ol style="list-style-type: none"> Memilih menu "Pendaftaran kader Posyandu". Mengisi form pendaftaran kader Posyandu dengan data yang lengkap dan benar Menekan tombol "Daftar"
Hasil Yang Diharapkan	Sistem menampilkan pesan "Pendaftaran kader baru berhasil"
Hasil	Sistem menampilkan pesan "Pendaftaran kader baru berhasil"
Status	Valid

Pada Tabel 6.35 merupakan hasil pengujian validasi untuk kasus uji pendaftaran kader Posyandu baru dengan skenario tes *alternative flow* (I). Hasil pengujian pada kasus uji pendaftaran Posyandu baru alternatif ini mendapatkan hasil *valid*.

Tabel 6.35 Pengujian validasi pendaftaran kader Posyandu baru alternatif

Kode Kebutuhan	KF-GIZI-020
Nama Kasus Uji	Pendaftaran kader Posyandu baru
Skenario Tes	Skenario <i>alternative flow</i> (I) Data NIK ganda
Prosedur Uji	<ol style="list-style-type: none"> Memilih menu "Pendaftaran kader Posyandu" Mengisi form pendaftaran kader Posyandu dengan lengkap namun menggunakan data NIK yang sudah terdaftar sebelumnya pada basis data. Menekan tombol "Daftar".
Hasil Yang Diharapkan	Sistem menampilkan pesan "Pendaftaran kader baru gagal"
Hasil	Sistem menampilkan pesan "Pendaftaran kader baru gagal"
Status	Valid

Pada Tabel 6.36 merupakan hasil pengujian validasi untuk kasus uji menampilkan data Posyandu dan kader Posyandu yang terdaftar dengan skenario tes *main flow*. Hasil pengujian pada kasus uji menampilkan data Posyandu dan kader Posyandu yang terdaftar ini mendapatkan hasil *valid*.



Tabel 6.36 Pengujian validasi menampilkan data Posyandu dan kader Posyandu yang terdaftar

Kode Kebutuhan	KF-GIZI-021
Nama Kasus Uji	Menampilkan data Posyandu dan kader Posyandu yang terdaftar
Skenario Tes	Skenario <i>main flow</i>
Prosedur Uji	1. Memilih menu “Kelola data Posyandu dan kader”
Hasil Yang Diharapkan	Sistem menampilkan data Posyandu dan kader Posyandu yang terdaftar dalam bentuk tabel.
Hasil	Sistem menampilkan data Posyandu dan kader Posyandu yang terdaftar dalam bentuk tabel.
Status	Valid

Pada Tabel 6.37 merupakan hasil pengujian validasi untuk kasus uji Memperbaharui data Posyandu yang terdaftar dengan skenario tes *main flow*. Hasil pengujian pada kasus uji memperbaharui data Posyandu yang terdaftar ini mendapatkan hasil *valid*.

Tabel 6.37 Pengujian validasi memperbaharui data Posyandu yang terdaftar

Kode Kebutuhan	KF-GIZI-022
Nama Kasus Uji	Memperbaharui data Posyandu yang terdaftar.
Skenario Tes	Skenario <i>main flow</i>
Prosedur Uji	1. Menekan tombol “edit” pada salah satu data Posyandu yang terdaftar. 2. Melakukan perubahan pada data alamat Posyandu. 3. Menekan tombol “Simpan”.
Hasil Yang Diharapkan	Sistem menyimpan perubahan data balita dan menampilkan <i>pop up</i> “Edit data Posyandu berhasil”
Hasil	Sistem menyimpan perubahan data balita dan menampilkan <i>pop up</i> “Edit data Posyandu berhasil”
Status	Valid

Pada Tabel 6.38 merupakan hasil pengujian validasi untuk kasus uji memperbaharui data Posyandu yang terdaftar dengan skenario tes *alternative flow (I)*. Hasil pengujian pada kasus uji memperbaharui data balita yang terdaftar alternatif ini mendapatkan hasil *valid*.



Tabel 6.38 Pengujian validasi memperbaharui data balita yang terdaftar alternatif

Kode Kebutuhan	KF-GIZI-022
Nama Kasus Uji	Memperbaharui data Posyandu yang terdaftar.
Skenario Tes	Skenario <i>alternative flow</i> (I) Data <i>username</i> ganda.
Prosedur Uji	<ol style="list-style-type: none"> 1. Menekan tombol “edit” pada salah satu data Posyandu yang terdaftar pada tabel. 2. Mengubah data <i>username</i> Posyandu dengan data <i>username</i> yang sudah ada pada basis data. 3. Menekan tombol “Simpan”.
Hasil Yang Diharapkan	Sistem menampilkan pesan “Edit data Posyandu gagal”
Hasil	Sistem menampilkan pesan “Edit data Posyandu gagal”
Status	Valid

Pada Tabel 6.39 merupakan hasil pengujian validasi untuk kasus uji menghapus data Posyandu yang terdaftar dengan skenario tes *main flow*. Hasil pengujian pada kasus uji menghapus data Posyandu yang terdaftar ini mendapatkan hasil *valid*.

Tabel 6.39 Pengujian validasi menghapus data Posyandu yang terdaftar

Kode Kebutuhan	KE-GIZI-023
Nama Kasus Uji	Menghapus data Posyandu yang terdaftar.
Skenario Tes	Skenario <i>main flow</i> .
Prosedur Uji	<ol style="list-style-type: none"> 1. Menekan tombol “hapus” pada salah satu data Posyandu pada tabel. 2. Mengonfirmasi penonaktifan data dengan menekan tombol “hapus” pada <i>pop up</i> konfirmasi yang ditampilkan.
Hasil Yang Diharapkan	Sistem menonaktifkan Posyandu tersebut dan menampilkan <i>pop up</i> “Penonaktifan Posyandu berhasil”.
Hasil	Sistem menonaktifkan Posyandu tersebut dan menampilkan <i>pop up</i> “Penonaktifan Posyandu berhasil”.
Status	Valid

Pada Tabel 6.40 merupakan hasil pengujian validasi untuk kasus uji menghapus data Posyandu yang terdaftar dengan skenario tes *alternative flow* (I). Hasil pengujian pada kasus uji menghapus data balita yang terdaftar alternatif ini mendapatkan hasil *valid*.



Tabel 6.40 Pengujian validasi menghapus data balita yang terdaftar alternatif

Kode Kebutuhan	KF-GIZI-023
Nama Kasus Uji	Menghapus data Posyandu yang terdaftar.
Skenario Tes	Skenario <i>alternative flow</i> (I) Membatalkan penghapusan data.
Prosedur Uji	1. Menekan tombol “hapus” pada salah satu data Posyandu pada tabel. 2. Membatalkan penonaktifan data dengan menekan tombol “batal” pada <i>pop up</i> konfirmasi yang ditampilkan.
Hasil Yang Diharapkan	Sistem menampilkan kembali halaman data Posyandu dan kader Posyandu yang terdaftar.
Hasil	Sistem menampilkan kembali halaman data Posyandu dan kader Posyandu yang terdaftar.
Status	Valid

Pada Tabel 6.41 merupakan hasil pengujian validasi untuk kasus uji memperbaharui data kader Posyandu yang terdaftar dengan skenario tes *main flow*. Hasil pengujian pada kasus uji memperbaharui data kader Posyandu yang terdaftar ini mendapatkan hasil *valid*.

Tabel 6.41 Pengujian validasi memperbaharui data kader Posyandu yang terdaftar

Kode Kebutuhan	KF-GIZI-024
Nama Kasus Uji	Memperbaharui data kader Posyandu yang terdaftar.
Skenario Tes	Skenario <i>main flow</i>
Prosedur Uji	1. Menekan tombol “edit” pada salah satu data kader Posyandu yang terdaftar. 2. Melakukan perubahan pada data nama kader Posyandu. 3. Menekan tombol “Simpan”.
Hasil Yang Diharapkan	Sistem menyimpan perubahan data Posyandu dan menampilkan <i>pop up</i> “Edit data kader berhasil”
Hasil	Sistem menyimpan perubahan data kader Posyandu dan menampilkan <i>pop up</i> “Edit data kader berhasil”.
Status	Valid

Pada Tabel 6.42 merupakan hasil pengujian validasi untuk kasus uji memperbaharui data kader Posyandu yang terdaftar dengan skenario tes *alternative flow* (I). Hasil pengujian pada kasus uji memperbaharui data kader Posyandu yang terdaftar alternatif ini mendapatkan hasil *valid*.



Tabel 6.42 Pengujian validasi memperbaharui data balita yang terdaftar alternatif

Kode Kebutuhan	KF-GIZI-024
Nama Kasus Uji	Memperbaharui data kader Posyandu yang terdaftar
Skenario Tes	Skenario <i>alternative flow</i> (I) Data NIK ganda.
Prosedur Uji	<ol style="list-style-type: none"> 1. Menekan tombol “edit” pada salah satu data balita yang terdaftar pada tabel. 2. Mengubah data NIK kader Posyandu dengan data NIK kader lain yang sudah terdapat pada basis data. 3. Menekan tombol “Simpan”.
Hasil Yang Diharapkan	Sistem menampilkan pesan “Edit data kader gagal”
Hasil	Sistem menampilkan pesan “Edit data kader gagal”
Status	Valid

Pada Tabel 6.43 merupakan hasil pengujian validasi untuk kasus uji menghapus data kader Posyandu yang terdaftar dengan skenario tes *main flow*. Hasil pengujian pada kasus uji menghapus data kader Posyandu yang terdaftar ini mendapatkan hasil *valid*.

Tabel 6.43 Pengujian validasi menghapus data kader Posyandu yang terdaftar

Kode Kebutuhan	KE-GIZI-025
Nama Kasus Uji	Menghapus data kader Posyandu yang terdaftar
Skenario Tes	Skenario <i>main flow</i>
Prosedur Uji	<ol style="list-style-type: none"> 1. Menekan tombol “hapus” pada salah satu data kader Posyandu pada tabel 2. Mengonfirmasi penghapusan data dengan menekan tombol “hapus” pada <i>pop up</i> konfirmasi yang ditampilkan.
Hasil Yang Diharapkan	Sistem menghapus data kader Posyandu tersebut dan menampilkan <i>pop up</i> “Hapus data kader berhasil”.
Hasil	Sistem menghapus data kader Posyandu tersebut dan menampilkan <i>pop up</i> “Hapus data kader berhasil”.
Status	Valid

Pada Tabel 6.44 merupakan hasil pengujian validasi untuk kasus uji menghapus data kader Posyandu yang terdaftar dengan skenario tes *alternative flow* (I). Hasil pengujian pada kasus uji menghapus data kader Posyandu yang terdaftar alternatif ini mendapatkan hasil *valid*.



Tabel 6.44 Pengujian validasi menghapus data kader Posyandu yang terdaftar alternatif

Kode Kebutuhan	KF-GIZI-025
Nama Kasus Uji	Menghapus data kader Posyandu yang terdaftar
Skenario Tes	Skenario <i>alternative flow</i> (I) Membatalkan penghapusan data.
Prosedur Uji	1. Menekan tombol “hapus” pada salah satu data kader Posyandu pada tabel. 2. Membatalkan penghapusan data dengan menekan tombol “batal” pada <i>pop up</i> konfirmasi yang ditampilkan.
Hasil Yang Diharapkan	Sistem menampilkan kembali halaman data Posyandu dan kader Posyandu yang terdaftar.
Hasil	Sistem menampilkan kembali halaman data Posyandu dan kader Posyandu yang terdaftar.
Status	Valid

Pada Tabel 6.45 merupakan hasil pengujian validasi untuk kasus uji menampilkan laporan gizi dan hasil pelayanan Posyandu dengan skenario tes *main flow*. Hasil pengujian pada kasus uji menampilkan laporan gizi dan hasil pelayanan Posyandu ini mendapatkan hasil *valid*.

Tabel 6.45 Pengujian validasi menampilkan laporan gizi dan hasil pelayanan Posyandu

Kode Kebutuhan	KF-GIZI-026
Nama Kasus Uji	Menampilkan laporan gizi dan hasil pelayanan Posyandu.
Skenario Tes	Skenario <i>main flow</i> .
Prosedur Uji	1. Memilih menu “Lihat laporan gizi”.
Hasil Yang Diharapkan	Sistem menampilkan laporan gizi dan hasil pelayanan Posyandu.
Hasil	Sistem menampilkan laporan gizi dan hasil pelayanan Posyandu.
Status	Valid

Pada Tabel 6.46 merupakan hasil pengujian validasi untuk kasus uji menampilkan laporan gizi dan hasil pelayanan Posyandu dengan skenario tes *alternative flow* (I). Hasil pengujian pada kasus uji menampilkan laporan gizi dan hasil pelayanan Posyandu alternatif ini mendapatkan hasil *valid*.



Tabel 6.46 Pengujian validasi menampilkan laporan gizi dan hasil pelayanan Posyandu alternatif

Kode Kebutuhan	KF-GIZI-026
Nama Kasus Uji	Menampilkan laporan gizi dan hasil pelayanan Posyandu.
Skenario Tes	Skenario <i>alternative flow</i> (I) Seleksi data laporan gizi dan hasil pelayanan Posyandu
Prosedur Uji	<ol style="list-style-type: none"> Memilih menu "Lihat laporan gizi". Memilih sumber data laporan gizi dan hasil pelayanan Posyandu dari <i>dropdown</i> pilihan Posyandu serta periode waktu laporan yang ingin ditampilkan. Menekan tombol "Lihat".
Hasil Yang Diharapkan	Sistem menampilkan laporan gizi dan hasil pelayanan Posyandu dengan sumber data pilihan Posyandu dan periode waktu yang dipilih oleh aktor.
Hasil	Sistem menampilkan laporan gizi dan hasil pelayanan Posyandu dengan sumber data pilihan Posyandu dan periode waktu yang dipilih oleh aktor.
Status	Valid

Pada Tabel 6.47 merupakan hasil pengujian validasi untuk kasus uji mengunduh laporan gizi dan hasil pelayanan Posyandu dengan skenario tes *main flow*. Hasil pengujian pada kasus uji mengunduh laporan gizi dan hasil pelayanan Posyandu ini mendapatkan hasil *valid*.

Tabel 6.47 Pengujian validasi mengunduh laporan gizi dan hasil pelayanan Posyandu

Kode Kebutuhan	KF-GIZI-027
Nama Kasus Uji	Mengunduh laporan gizi dan hasil pelayanan Posyandu
Skenario Tes	Skenario <i>main flow</i> .
Prosedur Uji	<ol style="list-style-type: none"> Menekan tombol "Unduh Excel Laporan Normal".
Hasil Yang Diharapkan	Sistem mengunduh laporan gizi dan hasil pelayanan Posyandu untuk balita pertumbuhan normal dalam format Microsoft Excel.
Hasil	Sistem mengunduh laporan gizi dan hasil pelayanan Posyandu untuk balita pertumbuhan normal dalam format Microsoft Excel.
Status	Valid

Pada Tabel 6.48 merupakan hasil pengujian validasi untuk kasus uji mengunduh laporan gizi dan hasil pelayanan Posyandu dengan skenario tes *alternative flow* (I). Hasil pengujian pada kasus uji mengunduh laporan gizi dan hasil pelayanan Posyandu alternatif ini mendapatkan hasil *valid*.



Tabel 6.48 Pengujian validasi mengunduh laporan gizi dan hasil pelayanan Posyandu alternatif

Kode Kebutuhan	KF-GIZI-027
Nama Kasus Uji	Mengunduh laporan gizi dan hasil pelayanan Posyandu
Skenario Tes	Skenario <i>alternative flow</i> (I) Mengunduh laporan balita malnutrisi.
Prosedur Uji	1. Menekan tombol "Unduh Excel Laporan Malnutrisi".
Hasil Yang Diharapkan	Sistem mengunduh laporan gizi dan hasil pelayanan Posyandu untuk balita malnutrisi dalam format Microsoft Excel.
Hasil	Sistem mengunduh laporan gizi dan hasil pelayanan Posyandu untuk balita malnutrisi dalam format Microsoft Excel.
Status	Valid

Pada Tabel 6.49 merupakan hasil pengujian validasi untuk kasus uji menampilkan profil balita dan riwayat hasil pemeriksaan skenario tes *main flow*. Hasil pengujian pada kasus uji menampilkan profil balita dan riwayat hasil pemeriksaan ini mendapatkan hasil *valid*.

Tabel 6.49 Pengujian validasi menampilkan profil balita dan riwayat hasil pemeriksaan

Kode Kebutuhan	KF-GIZI-028
Nama Kasus Uji	Menampilkan profil balita dan riwayat hasil pemeriksaan
Skenario Tes	Skenario <i>main flow</i>
Prosedur Uji	1. Memilih menu "Lihat data balita". 2. Menekan tombol "Detail" pada salah satu data balita yang ditampilkan pada tabel.
Hasil Yang Diharapkan	Sistem menampilkan profil balita dan riwayat hasil pemeriksaan balita tersebut.
Hasil	Sistem menampilkan profil balita dan riwayat hasil pemeriksaan balita tersebut.
Status	Valid

Pada Tabel 6.50 merupakan hasil pengujian validasi untuk kasus uji memperbaharui data hasil pemeriksaan dengan skenario tes *main flow*. Hasil pengujian pada kasus uji memperbaharui data hasil pemeriksaan ini mendapatkan hasil *valid*.



Tabel 6.50 Pengujian validasi memperbaharui data hasil pemeriksaan

Kode Kebutuhan	KF-GIZI-029
Nama Kasus Uji	Memperbaharui data hasil pemeriksaan.
Skenario Tes	Skenario <i>main flow</i>
Prosedur Uji	<ol style="list-style-type: none"> 1. Menekan tombol “edit” pada salah satu data hasil pemeriksaan pada tabel. 2. Melakukan perubahan pada data berat badan balita. 3. Menekan tombol “Simpan”.
Hasil Yang Diharapkan	Sistem menyimpan perubahan data hasil pemeriksaan dan menampilkan <i>pop up</i> “Edit riwayat kesehatan balita berhasil”.
Hasil	Sistem menyimpan perubahan data hasil pemeriksaan dan menampilkan <i>pop up</i> “Edit riwayat kesehatan balita berhasil”.
Status	Valid

Pada Tabel 6.51 merupakan hasil pengujian validasi untuk kasus uji memperbaharui data hasil pemeriksaan dengan skenario tes *alternative flow* (I). Hasil pengujian pada kasus uji memperbaharui data hasil pemeriksaan alternatif ini mendapatkan hasil *valid*.

Tabel 6.51 Pengujian validasi memperbaharui data hasil pemeriksaan alternatif

Kode Kebutuhan	KF-GIZI-029
Nama Kasus Uji	Memperbaharui data kader Posyandu yang terdaftar
Skenario Tes	Skenario <i>alternative flow</i> (I) Kolom berat badan / tinggi badan / lingkaran kepala / cara pengukuran kosong.
Prosedur Uji	<ol style="list-style-type: none"> 1. Menekan tombol “edit” pada salah satu data balita yang terdaftar pada tabel. 2. Mengosongkan kolom berat badan. 3. Menekan tombol “Simpan”.
Hasil Yang Diharapkan	Sistem menampilkan peringatan pada kolom berat badan dengan pesan “Please fill out this field”.
Hasil	Sistem menampilkan peringatan pada kolom berat badan dengan pesan “Please fill out this field”.
Status	Valid

Pada Tabel 6.52 merupakan hasil pengujian validasi untuk kasus uji menghapus data hasil pemeriksaan dengan skenario tes *main flow*. Hasil pengujian pada kasus uji menghapus data hasil pemeriksaan ini mendapatkan hasil *valid*.



Tabel 6.52 Pengujian validasi menghapus data hasil pemeriksaan

Kode Kebutuhan	KF-GIZI-030
Nama Kasus Uji	Menghapus data hasil pemeriksaan.
Skenario Tes	Skenario <i>main flow</i>
Prosedur Uji	<ol style="list-style-type: none"> 1. Menekan tombol “hapus” pada salah satu data hasil pemeriksaan balita tersebut yang berada pada tabel. 2. Mengonfirmasi penghapusan data dengan menekan tombol “hapus” pada <i>pop up</i> konfirmasi yang ditampilkan.
Hasil Yang Diharapkan	Sistem menghapus data hasil pemeriksaan tersebut dan menampilkan <i>pop up</i> “Hapus data riwayat kesehatan balita berhasil”.
Hasil	Sistem menghapus data hasil pemeriksaan tersebut dan menampilkan <i>pop up</i> “Hapus data riwayat kesehatan balita berhasil”.
Status	Valid

Pada Tabel 6.53 merupakan hasil pengujian validasi untuk kasus uji menghapus data hasil pemeriksaan dengan skenario tes *alternative flow* (I). Hasil pengujian pada kasus uji menghapus data hasil pemeriksaan alternatif ini mendapatkan hasil *valid*.

Tabel 6.53 Pengujian validasi menghapus data hasil pemeriksaan alternatif

Kode Kebutuhan	KF-GIZI-030
Nama Kasus Uji	Menghapus data hasil pemeriksaan
Skenario Tes	<p>Skenario <i>alternative flow</i> (I)</p> <p>Membatalkan penghapusan data.</p>
Prosedur Uji	<ol style="list-style-type: none"> 1. Menekan tombol “hapus” pada salah satu data hasil pemeriksaan balita yang berada pada tabel. 2. Membatalkan penghapusan data dengan menekan tombol “batal” pada <i>pop up</i> konfirmasi yang ditampilkan.
Hasil Yang Diharapkan	Sistem menampilkan kembali halaman profil balita dan riwayat hasil pemeriksaan.
Hasil	Sistem menampilkan kembali halaman profil balita dan riwayat hasil pemeriksaan.
Status	Valid

Pada Tabel 6.54 merupakan hasil pengujian validasi untuk kasus uji menampilkan KMS balita dengan skenario tes *main flow*. Hasil pengujian pada kasus uji menampilkan KMS balita ini mendapatkan hasil *valid*.



Tabel 6.54 Pengujian validasi menampilkan KMS balita

Kode Kebutuhan	KF-GIZI-031
Nama Kasus Uji	Menampilkan KMS balita
Skenario Tes	Skenario <i>main flow</i>
Prosedur Uji	1. Menekan tombol “Lihat KMS” pada halaman profil balita dan riwayat hasil pemeriksaan.
Hasil Yang Diharapkan	Sistem menampilkan halaman KMS balita
Hasil	Sistem menampilkan halaman KMS balita.
Status	Valid

Pada Tabel 6.55 merupakan hasil pengujian validasi untuk kasus uji memberikan rekomendasi gizi dengan skenario tes *main flow*. Hasil pengujian pada kasus uji memberikan rekomendasi gizi mendapatkan hasil *valid*.

Tabel 6.55 Pengujian validasi memberikan rekomendasi gizi

Kode Kebutuhan	KF-GIZI-032
Nama Kasus Uji	Memberikan rekomendasi gizi.
Skenario Tes	Skenario <i>main flow</i> .
Prosedur Uji	<ol style="list-style-type: none"> 1. Menekan tombol “Beri saran gizi” pada halaman profil balita dan riwayat hasil pemeriksaan. 2. Mengisi kolom saran gizi pada <i>pop up</i> form yang ditampilkan. 3. Menekan tombol “Kirim”.
Hasil Yang Diharapkan	Sistem menyimpan hasil saran gizi dan menampilkan pesan “Saran gizi berhasil dikirim”.
Hasil	Sistem menyimpan hasil saran gizi dan menampilkan pesan “Saran gizi berhasil dikirim”.
Status	Valid

Pada Tabel 6.56 merupakan hasil pengujian validasi untuk kasus uji memberikan rekomendasi gizi dengan skenario tes *alternative flow* (I). Hasil pengujian pada kasus uji memberikan rekomendasi gizi alternatif ini mendapatkan hasil *valid*.



Tabel 6.56 Pengujian validasi memberikan rekomendasi gizi alternatif

Kode Kebutuhan	KF-GIZI-032
Nama Kasus Uji	Memberikan rekomendasi gizi.
Skenario Tes	Skenario <i>alternative flow</i> (I): Mengosongkan kolom saran gizi
Prosedur Uji	1. Menekan tombol “Beri saran gizi” pada halaman profil balita dan riwayat hasil pemeriksaan. 2. Mengosongkan kolom saran gizi pada <i>pop up form</i> yang ditampilkan. 3. Menekan tombol “Kirim”
Hasil Yang Diharapkan	Sistem menampilkan peringatan pada kolom saran gizi dengan pesan “Please fill out this field”.
Hasil	Sistem menampilkan peringatan pada kolom saran gizi dengan pesan “Please fill out this field”.
Status	Valid

Pada Tabel 6.57 merupakan hasil pengujian validasi untuk kasus uji *logout* dengan skenario tes *main flow*. Hasil pengujian pada kasus uji *logout* mendapatkan hasil *valid*.

Tabel 6.57 Pengujian validasi *logout*

Kode Kebutuhan	KF-GIZI-033
Nama Kasus Uji	<i>Logout</i>
Skenario Tes	Skenario <i>main flow</i>
Prosedur Uji	1. Menekan tombol <i>dropdown</i> yang berada pada kanan atas bagian <i>navbar</i> sistem. 2. Menekan tombol “Keluar” 3. Sistem menampilkan <i>pop up</i> konfirmasi untuk <i>logout</i> 4. Menekan tombol “Keluar”.
Hasil Yang Diharapkan	Sistem akan menghentikan sesi dari aktor tersebut dan menampilkan halaman <i>login</i> .
Hasil	Sistem akan menghentikan sesi dari aktor tersebut dan menampilkan halaman <i>login</i> .
Status	Valid

Pada Tabel 6.58 merupakan hasil pengujian validasi untuk kasus uji *logout* dengan skenario tes *alternative flow* (I). Hasil pengujian pada kasus uji *logout* alternatif mendapatkan hasil *valid*.

Tabel 6.58 Pengujian validasi *logout* alternatif

Kode Kebutuhan	KF-GIZI-033
Nama Kasus Uji	<i>Logout</i>
Skenario Tes	Skenario <i>alternative flow</i> (!): Membatalkan <i>logout</i>
Prosedur Uji	1. Menekan tombol <i>dropdown</i> yang berada pada kanan atas bagian <i>navbar</i> sistem. 2. Menekan tombol "Keluar" 3. Sistem menampilkan <i>pop-up</i> konfirmasi untuk <i>logout</i> . 4. Menekan tombol "Batal"
Hasil Yang Diharapkan	Sistem menampilkan kembali halaman saat aktor menekan tombol <i>logout</i> .
Hasil	Sistem menampilkan kembali halaman saat aktor menekan tombol <i>logout</i> .
Status	Valid

6.2 User Acceptance Testing (UAT)

Setelah dilakukannya pengujian secara *black box* untuk sistem yang telah dibuat, maka nya adalah menguji sistem dengan menggunakan *User Acceptance Testing* (UAT) kepada para pengguna untuk memastikan bahwa sistem dapat diterima dengan baik oleh para pengguna. UAT ini akan dilakukan dengan kepada 5 pengguna yang akan mewakili jenis aktor yang berperan dalam sistem, yaitu 1 orang *admin*, 1 orang bidan, serta 3 orang kader Posyandu. UAT akan dilakukan dengan melakukan demonstrasi cara kerja sistem secara umum lalu meminta setiap responden untuk menggunakan fungsi-fungsi utama sistem, sebelum akhirnya diberikan kuesioner tingkat penerimaan pengguna yang terdiri dari beberapa pertanyaan dan dapat dijawab dengan menggunakan skala likert dan untuk hasil pengujian dapat dilihat dalam Lampiran F. Pada Tabel 6.59 merupakan hasil jawaban kuesioner yang diberikan kepada responden untuk aktor *admin*.



Tabel 6.59 Hasil User Acceptance Testing (UAT) aktor admin

No.	Pertanyaan	Jawaban				
		Sangat Setuju	Setuju	Netral	Tidak Setuju	Sangat Tidak Setuju
<i>Functional correctness and completeness</i>						
1.	Apakah sistem informasi ini dapat membantu anda dalam mengelola bidang yang terdaftar?	✓				
2.	Apakah sistem informasi ini dapat membantu anda memvalidasi data bidang yang mendaftar?	✓				
<i>Usability</i>						
3.	Apakah sistem informasi ini mudah untuk digunakan?	✓				
4.	Apakah sistem informasi ini mudah untuk dipelajari?	✓				
Total		4				

Dari hasil pengujian UAT kepada 1 responden yang berperan sebagai aktor admin pada Tabel 6.59, maka dapat dilakukan perhitungan total nilai kuesioner yang ada pada Tabel 6.60.

Tabel 6.60 Perhitungan nilai kuesioner User Acceptance Testing (UAT) aktor admin

Jawaban	Bobot	Jumlah Respon	Nilai
Sangat Setuju	5	4	20
Setuju	4	0	0
Netral	3	0	0
Tidak Setuju	2	0	0
Sangat Tidak Setuju	1	0	0
Total nilai			20

Dari total nilai tersebut, dapat dilakukan perhitungan untuk mendapatkan hasil akhir tingkat penerimaan pengguna untuk aktor admin dengan perhitungan sebagai berikut:

$$\text{Nilai akhir} = \frac{\text{total nilai}}{\text{nilai maksimum}} \times 100\% = \frac{20}{20} \times 100\% = 100\% \text{ (Sangat Setuju)}$$

Berikutnya pada Tabel 6.61 merupakan hasil pengujian UAT untuk aktor bidang



Tabel 6.61 Hasil User Acceptance Tesing (UAT) aktor bidan

No.	Pertanyaan	Jawaban				
		Sangat Setuju	Setuju	Netral	Tidak Setuju	Sangat Tidak Setuju
<i>Functional correctness and completeness</i>						
1.	Apakah sistem informasi ini dapat membantu anda dalam membuat laporan gizi balita dan hasil pelayanan Posyandu?	✓				
2.	Apakah sistem informasi ini dapat membantu anda dalam mengelola Posyandu dan kader Posyandu?	✓				
<i>Usability</i>						
3.	Apakah sistem informasi ini mudah untuk digunakan?		✓			
4.	Apakah sistem informasi ini mudah untuk dipelajari?		✓			
Total		3	1			

Dari hasil pengujian UAT kepada 1 responden yang berperan sebagai aktor bidan pada Tabel 6.61, maka dapat dilakukan perhitungan total nilai kuesioner yang berada pada Tabel 6.62.

Tabel 6.62 Perhitungan nilai kuesioner User Acceptance Testing (UAT) aktor bidan

Jawaban	Bobot	Jumlah Respon	Nilai
Sangat Setuju	5	3	15
Setuju	4	1	4
Netral	3	0	0
Tidak Setuju	2	0	0
Sangat Tidak Setuju	1	0	0
Total nilai			19

Dari total nilai tersebut, dapat dilakukan perhitungan untuk mendapatkan hasil akhir tingkat penerimaan pengguna untuk aktor bidan dengan perhitungan berikut.

$$\text{Nilai akhir} = \frac{\text{total nilai}}{\text{nilai maksimum}} \times 100\% = \frac{19}{20} \times 100\% = 95\% \text{ (Sangat Setuju)}$$

Berikutnya pada Tabel 6.63 merupakan hasil pengujian UAT untuk aktor Posyandu.



Tabel 6.63 Hasil User Acceptance Testing (UAT) aktor Posyandu

No.	Pertanyaan	Jawaban				
		Sangat Setuju	Setuju	Netral	Tidak Setuju	Sangat Tidak Setuju
<i>Functional correctness and completeness</i>						
1.	Apakah sistem informasi ini dapat membantu anda dalam menjadwalkan dan mengirimkan undangan kegiatan Posyandu?	✓✓	✓			
2.	Apakah sistem informasi ini dapat membantu anda dalam melakukan pencatatan hasil pengukuran dan pemeriksaan balita?	✓✓				
3.	Apakah sistem informasi ini dapat membantu anda dalam mengelola data orang tua balita?	✓✓				
4.	Apakah sistem informasi ini dapat membantu anda dalam mengelola data balita yang ada pada Posyandu?	✓✓				
5.	Apakah sistem informasi ini dapat membantu anda dalam mendata kehadiran kader dalam kegiatan hari Posyandu?	✓✓	✓			
<i>Usability</i>						
6.	Apakah sistem informasi ini mudah untuk digunakan?	✓✓	✓			
7.	Apakah Sistem informasi ini mudah untuk dipelajari?	✓✓	✓			
Total		17	4			

Dari hasil pengujian UAT kepada 3 responden yang berperan sebagai aktor Posyandu pada Tabel 6.63, maka dapat dilakukan perhitungan total nilai kuesioner pada Tabel 6.64.

Tabel 6.64 Perhitungan nilai kuesioner User Acceptance Tesing (UAT) aktor Posyandu

Jawaban	Bobot	Jumlah Respon	Nilai
Sangat Setuju	5	17	85
Setuju	4	4	16
Netral	3	0	0
Tidak Setuju	2	0	0
Sangat Tidak Setuju	1	0	0
Total nilai			101



Dari total nilai tersebut, dapat dilakukan perhitungan untuk mendapatkan hasil akhir tingkat penerimaan pengguna untuk aktor Posyandu dengan perhitungan berikut:

$$\text{Nilai akhir} = \frac{\text{total nilai}}{\text{nilai maksimum}} \times 100\% = \frac{101}{105} \times 100\% = 96.2\% \text{ (Sangat Setuju)}$$

Dari ketiga nilai akhir tersebut, dapat dilakukan perhitungan untuk mendapatkan nilai rata-rata tingkat penerimaan sistem yang ada pada Tabel 6.65.

Tabel 6.65 Hasil User Acceptance Testing (UAT)

Aktor	Nilai Akhir
Admin	100 % (Sangat Setuju)
Bidan	95 % (Sangat Setuju)
Posyandu	96.2 % (Sangat Setuju)
Rata-Rata	97.1 % (Sangat Setuju)

6.3 Analisis Hasil Pengujian

Dari pengujian yang telah dilakukan yaitu dengan menggunakan pengujian secara *black box* untuk memvalidasi kebutuhan fungsional, serta menggunakan *User Acceptance Testing (UAT)* dengan skala likert untuk mengukur tingkat penerimaan pengguna terhadap sistem yang telah dibangun, maka dapat ditarik kesimpulan sebagai berikut. Pengujian *black box* yang dilakukan terhadap 33 kebutuhan fungsional sistem telah dilakukan dimana skenario uji didasarkan pada *use case scenario* dengan menjalankan *main flow* serta *alternative flow* pada setiap kebutuhan fungsional. Hasil pengujian *black box* ini mendapatkan kesimpulan bahwa sistem yang telah dibangun ini telah 100% *valid* berhasil mengimplementasikan semua kebutuhan fungsional yang telah didefinisikan dan telah berfungsi sesuai dengan ekspektasi.

Untuk hasil pengujian menggunakan *User Acceptance Testing (UAT)* menggunakan skala likert kepada 5 pengguna, didapatkan bahwa tingkat penerimaan pengguna untuk aktor *admin* adalah 100 % (Sangat setuju), untuk aktor *bidan* adalah 95 % (Sangat setuju), serta aktor *Posyandu* adalah 96.2 % (sangat setuju). Sehingga jika dirata-rata dari tingkat penerimaan ketiga aktor tersebut, dapat disimpulkan bahwa sistem yang telah dibangun mendapatkan tingkat penerimaan pengguna sebesar 97.1 % dimana artinya para pengguna rata-rata sangat setuju terhadap sistem yang telah dibuat.



BAB 7 KESIMPULAN DAN SARAN

7.1 Kesimpulan

Berdasarkan hasil penelitian yang telah dilakukan, maka dapat diperoleh beberapa kesimpulan sebagai berikut.

1. Berdasarkan hasil analisis kebutuhan dari sistem informasi pencatatan dan pelaporan gizi balita ini diperoleh bahwa terdapat 4 jenis aktor yang akan terlibat dalam sistem yaitu *user*, *admin*, Posyandu, dan bidan, 33 kebutuhan fungsional serta 1 kebutuhan non fungsional. Hasil analisis kebutuhan ini didapatkan melalui wawancara dan diskusi permasalahan dengan Bapak Intan Yusuf Habibie S.Gz., M.Sc., selaku dosen gizi Fakultas Kedokteran Universitas Brawijaya serta bidan desa yang bertugas yaitu Ibu Komariah Tusokilian A.Md., yang dilanjutkan dengan tinjauan langsung pada Posyandu Balita Srikaya Desa Krebet, Kabupaten Malang, dan pada akhirnya dilakukan validasi kebutuhan dengan Bapak Intan Yusuf Habibie S.Gz., M.Sc.
2. Pada proses perancangan sistem informasi pencatatan dan pelaporan gizi balita ini dilakukan berdasarkan kebutuhan yang telah didefinisikan pada tahap analisis kebutuhan, yang terdiri dari perancangan arsitektur menggunakan *sequence diagram* dan *class diagram*, perancangan data menggunakan *physical data diagram*, perancangan komponen menggunakan *pseudocode*, serta perancangan antarmuka sistem menggunakan *wireframe*. Setelah dilakukannya perancangan sistem, maka hasil perancangan tersebut akan diimplementasikan ke dalam bahasa pemrograman *web* dengan menggunakan pendekatan berbasis objek menggunakan *framework* Codeigniter dan *framework* tampilan Bootstrap. Proses implementasi ini menghasilkan implementasi kode program, implementasi basis data serta implementasi antarmuka sistem.
3. Pengujian sistem dilakukan pada tahap terakhir setelah proses perancangan dan implementasi sistem telah selesai dilakukan. Proses pengujian dilakukan dengan menggunakan pengujian validasi atau *validation testing* yaitu menguji sistem secara *blackbox* untuk memvalidasi bahwa sistem sudah memenuhi semua kebutuhan dengan benar atau *valid*. Setelah pengujian validasi, akan dilakukan pengujian *User Acceptance Testing (UAT)* kepada pihak-pihak terkait yaitu kepada 1 orang aktor admin, 1 orang aktor bidan, serta 3 orang kader Posyandu dengan memberikan serangkaian pertanyaan kepada para responden dan dapat dijawab dengan menggunakan skala likert yang sudah disediakan. Hasil dari pengujian validasi ini mendapatkan nilai 100% *valid* yang berarti bahwa sistem yang telah diimplementasikan sudah memenuhi semua kebutuhan dengan baik. Untuk hasil pengujian UAT kepada 5 orang tersebut mendapatkan hasil rata-rata sebesar 97,1% yang artinya pengguna rata-rata sangat setuju terhadap sistem yang telah dibuat.



7.2 Saran

Berdasarkan hasil penelitian yang telah dilakukan dalam mengembangkan sistem informasi pencatatan dan pelaporan gizi balita ini adalah sebagai berikut:

1. Mengembangkan cakupan dan fungsionalitas sistem agar bisa mengakomodasi pelayanan kesehatan yang ada pada tingkat Puskesmas.
2. Mengembangkan fungsionalitas sistem untuk mengakomodasi pelayanan kesehatan Posyandu selain balita.
3. Memperkaya fungsionalitas sistem yang berhubungan dengan aplikasi perangkat bergerak berbasis Android agar bisa memberikan lebih banyak manfaat bagi pengguna aplikasi tersebut.



DAFTAR REFERENSI

- Abdulloh, R., 2018. *7 in 1 Pemrograman Web untuk Pemula*. Jakarta: PT Elex Media Komputindo.
- Anwar, F. et al., 2010. High Participation in The Posyandu Nutrition program Improved Child Nutritional Status. *Nutrition Research and Practice*, 1(1), pp. 208-214.
- Basuki, A. P., 2011. *Membangun Web Berbasis PHP Dengan Framwework Codeigniter*. 3 penyunt. Yogyakarta: Lokomedia.
- Blackedget, J. M., 2006. Principal of Software Engineering Dalam: *Digital Signal Processing (Second Edition)*. Loughbourough : Horwood Publishing Limited, pp. 299-363.
- Choizes, 2019. *Pengertian Skala Likert dan Contoh Cara Hitung Kuesionernya*. [Online]. Available at: <https://www.diedit.com/skala-likert/> [Diakses 28 September 2019].
- Direktorat Gizi Masyarakat, 2017. *Buku Status Pemantauan Status Gizi Tahun 2017*. Jakarta: Kementerian Kesehatan.
- Glasser, M., 2009. *Open Verification Methodology Cookbook*. 1st penyunt. New York: Springer Dordrecht Heidelberg.
- Habibie, I. Y., 2019. *Deskripsi Kondisi Proses Pencatatan dan Pelaporan Gizi Balita di Posyandu Desa Krebbe, Kabupaten Malang*. [Wawancara] (23 Agustus 2019).
- Hasan, M. I., 2012. *Pokok-Pokok Materi Statistik (Statistik Inferensif)*. 2nd penyunt. Jakarta: PT Bumi ksara.
- Hidayat, T. S. & Jahari, A. B., 2012. Perilaku Pemanfaatan Posyandu Hubungannya Dengan Status Gizi dan Morbiditas Balita. *Buletin Penelitian Kesehatan*, 40(1), pp. 1-10.
- Hutahean, J., 2015. *Konsep Sistem Informasi*. Yogyakarta: Deepublish.
- Kemendes RI, 2011. *Keputusan Menteri Kesehatan Republik Indonesia Tentang Standar Antropometri Penilaian Status Gizi Anak*, Jakarta: Kementerian Kesehatan Republik Indonesia.
- Kementerian Kesehatan Republik Indonesia, 2010. *Peraturan Menteri Kesehatan Republik Indonesia Tentang Penggunaan KMS Bagi Balita*, Jakarta: Kementerian Kesehatan.
- Kementerian Kesehatan RI Pusat Promosi Kesehatan, 2012. *Buku Pegangan Kader Posyandu*. Jakarta: Kementerian Kesehatan.
- Kementerian Kesehatan RI, 2011. *Pedoman Umum Pengelolaan Posyandu*. 1st penyunt. Jakarta: Kementerian Kesehatan RI.



Khoiri, A., 2008. *Pengembangan Sistem Informasi Posyandu Guna Mendukung Surveilans Kesehatan Ibu & Anak Berbasis Masyarakat Pada Desa Siaga (Studi Kasus Di Kelurahan Manisrejo Kecamatan Taman)*, Semarang: Universitas Diponegoro.

Kurniawan, T. A., 2018. Permodelan Use Case (UML): Evaluasi Terhadap Beberapa Kesalahan Dalam Praktik. *Jurnal Teknologi Informasi dan Ilmu Komputer (JTIIK)*, V(1), pp. 77-86.

Lunn, K., 2003. *Software Development With UML*. 1 penyunt. New York: Palgrave Macmillan.

Muliah, N., Wardoyo, A. S. & Mahmudiono, T., 2017. Hubungan Frekuensi Penimbangan, Penggunaan Garam Beryodium dan Pemberian Vitamin A Dengan Kejadian Underweight Pada Balita di Provinsi Jawa Timur. *Jurnal Media Gizi Indonesia*, pp. 40-46.

Mulyani, S., 2016. *Metode Analisis dan Perancangan Sistem*. Bandung: Abdi Sistemika.

Naik, K. & Tripathy, P., 2008. *Software Testing and Quality Assurance Theory and Practice*. 1st penyunt. Hoboken: John Wiley & Sons, Inc..

National Institute of Open Schooling, 2014. Nutritional Status. Dalam: *Food and Nutrition*. India: National Institute of Open Schooling, pp. 91-110.

Nielsen, J., 2012. *Usability 101: Introduction to Usability*. [Online] Available at: <https://www.nngroup.com/articles/usability-101-introduction-to-usability/> [Diakses 18 September 2019].

Object Management Group, 2011. *Business Process Model and Notation (BPMN) Version 2.0*. [Online] Available at: http://www.oatsolutions.com.br/artigos/SpecBPMN_v2.pdf [Diakses 9 September 2019].

Object Management Group, 2017. *Unified Modelling Language*. [Online] Available at: <https://www.omg.org/spec/UML> [Diakses 8 November 2019].

Pekik, D., 2006. *Panduan Gizi Lengkap*, Yogyakarta: Andi.

Pressman, R. S., 2010. *Software Engineering A Practitioner's Approach Seventh Edition*. 7th penyunt. New York: McGraw-Hill Companies, Inc..

Sommerville, I., 2011. *Software Engineering Ninth Edition*. Boston: Pearson.

Sudarmilah, E., Asy'ari, H. & Giwangkoro, G. G., 2011. *Web Based Information System for Motoric and Mental Development in Early Childhood*. Semarang, Universitas Diponegoro.

Tolle, H., Pinandito, A., Kharisma, A. P. & Dewi, R. K., 2017. *Pengembangan Aplikasi Perangkat Bergerak (Konsep dan Implementasi)*. Malang: UB Press.



Tusokilah, K., 2019. *Penjelasan Proses Pelayanan Posyandu Serta Permasalahan yang Dihadapi Dalam Proses Pencatatan dan Pelaporan Gizi Balita Pada Posyandu Desa Krebbe, Kabupaten Malang* [Wawancara] (6 September 2019).

Visual Paradigm, 2018. *Conceptual, Logical and Physical Data Model*. [Online] Available at: <https://www.visual-paradigm.com/support/documents/youerguide/3563/3564/85378/conceptual.html> [Diakses 9 November 2019].

Weske, M., 2007. *Business Process Management Concepts, Languages, Architectures*. Postdam: Springer-Verlag Berlin Heidelberg.



LAMPIRAN A HASIL WAWANCARA

A.1 Hasil Wawancara Dengan Intan Yusuf Habibie S.Gz., M.Sc. (Dosen Program Studi Ilmu Gizi FK UB)

HASIL WAWANCARA

Hari & Tanggal: Jumat, 23 Agustus 2019

Waktu: 14.00 WIB

Tempat: Fakultas Ilmu Komputer

Pewawancara: Timothy Julian (Mahasiswa Sistem Informasi Fakultas Ilmu Komputer UB)

Narasumber: Intan Yusuf Habibie S.Gz., M.Sc (Dosen program studi ilmu gizi Fakultas Kedokteran UB)

Pewawancara: Selamat siang pak Yusuf, perkenalkan saya Timothy Julian dari FILKOM mau melakukan wawancara dengan bapak terkait penelitian yang sedang saya lakukan untuk membuat sistem informasi terkait gizi. Untuk pertama-tama boleh saya tau mengenai kondisi pelayanan kesehatan anak balita di Posyandu?

Narasumber: Siang mas Julian, jadi dalam urusan kesehatan anak balita itu, kita mulai dari level nasional ada yang namanya Kementerian Kesehatan, lalu ada Dinas Kesehatan Provinsi, Dinas Kesehatan Kota/Kabupaten, dimana Dinas Kesehatan tingkat kota ini yang merancang program-program kesehatan, sedangkan yang melaksanakannya ada pada Unit Pelaksana Teknis di tingkat kota yaitu Puskesmas dan dibawahnya lagi ada Posyandu. Salah satu program kesehatannya itu adalah adanya proses pencatatan dan pelaporan status gizi anak dan itu dilakukan menggunakan pendekatan *bottom-up* atau dari level paling bawah yaitu pada Posyandu-Posyandu untuk melakukan proses pencatatan status gizi, dan melaporkannya ke tingkat atasnya yaitu Puskesmas dan seterusnya hingga level nasional. Letak kompleksitas atau sering terjadi kendala nya adalah ada pada tingkat Posyandu dan Puskesmas ini, khususnya untuk anak yang berada di bawah 5 tahun atau anak balita.

Pewawancara: Permasalahan seperti apa ya itu pak?

Narasumber: Jadi kan di Posyandu ini ada yang namanya KMS, atau Kartu Menuju Sehat nah di KMS itu akan dilakukan proses pencatatan berdasarkan hasil pengukuran yang dilakukan kepada anak tersebut untuk melihat *progress* pertumbuhan anak seperti tinggi maupun berat badan. Nah proses pencatatan itu dilakukan oleh ibu-ibu kader yang bertugas di Posyandu tersebut dan diawasi oleh ibu bidan desa atau kelurahan. Biasanya, di Posyandu itu proses pencatatan tersebut masih dilakukan secara manual, seperti misalnya saya melakukan pengukuran anak saya di Posyandu, lalu setelah diukur oleh kader, dia akan memberi tahu ke kader lain yang berada di meja pencatatan mengenai hasil pengukurannya, baru akan dicatat di sebuah buku register balita bahwa anak tersebut memiliki berat sekian kilogram. Disini kita bicara bahwa dalam proses pencatatan data kesehatan anak balita tersebut validitas datanya masih rendah.

Pewawancara: Maksudnya validitas data itu bagaimana pak, dan apa dampaknya jika memang terjadinya kesalahan pencatatan data kesehatan di Posyandu tersebut?



Narasumber: Jadi, jika kita lihat secara skala kecil memang mungkin dirasa tidak jadi masalah hal tersebut terjadi sekali-kali, namun saat ini ada isu nasional yaitu kasus *stunting* atau balita pendek, dan angka balita *stunting* di Indonesia masih cukup tinggi pada tahun 2010 berkisar 35% dan sekarang berkisar di angka 23%. Nah angka tersebut selain melalui survey, juga bisa didapatkan dari proses yang tadi saya jelaskan, yang menggunakan proses pencatatan dan pelaporan secara *bottom-up* dari tingkat Posyandu di kelurahan yang melakukan pemeriksaan kesehatan anak dan pencatatan dan melaporkannya ke tingkat atasnya. Nah, jika validitas data proses pencatatan di Posyandu saja masih rendah, bisa saja pihak Posyandu salah memberikan saran gizi maupun obat-obatan kepada anak tersebut yang terindikasi gizi buruk karena adanya kesalahan proses pencatatan, maupun berimbas pada hasil pelaporan di tingkat atas dan seterusnya yang menjadi tidak valid juga, dan bisa mengakibatkan kesalahan interpretasi tingkat kesehatan balita di wilayah tersebut. Jadi, dari permasalahan tersebut, kita perlu nih kontribusi dari sisi teknologinya yang bisa memudahkan para *user* nya atau dalam hal ini para kader dan bidan untuk melakukan dokumentasi yang lebih baik dan meningkatkan tingkat validitas data kesehatan anak balita.

Pewawancara: Baik pak terima kasih untuk waktunya serta pemaparan mengenai kondisi pelayanan kesehatan anak balita, saya akhiri wawancaranya disini, selamat siang.

Narasumber: Baik mas sama-sama, selamat siang.

Malang, 23 Agustus 2019

Intan Yusuf Habibie S.Gz, M.Sc



A.2 Hasil Wawancara Dengan Komariah Tusokilah A.Md (Bidan Desa Krebet, Kabupaten Malang)

HASIL WAWANCARA

Hari & Tanggal: Jumat, 6 September 2019.

Waktu: 11:00 WIB

Tempat: Posyandu Balita Srikaya, Desa Krebet, Kabupaten Malang

Pewawancara: Timothy Julian (Mahasiswa Sistem Informasi Fakultas Ilmu Komputer UB)

Narasumber: Komariah Tusokilah A.Md (Bidan Desa Krebet, Kabupaten Malang)

Pewawancara: Selamat siang bu komariah, perkenalkan saya Timothy Julian, mahasiswa Fakultas Ilmu Komputer Universitas Brawijaya Malang.

Narasumber: Siang mas, ada yang bisa saya bantu?

Pewawancara: Jadi begini bu, saya ingin melakukan wawancara dengan ibu, untuk mengetahui lebih dalam proses pelayanan di posyandu itu seperti apa, khususnya dalam hal proses pencatatan dan pelaporannya hasil pelayanan posyandu bu, dan mungkin apabila ada permasalahan yang mungkin ibu atau pihak kader posyandu hadapi selama ini, bisa dijelaskan juga..

Narasumber: Baik mas, jadi untuk di posyandu sendiri, proses pelayanannya dimulai dari penjadwalan serta pembagian tugas kader Posyandu terlebih dahulu, jadi ditentukan siapa yang akan mencatat, siapa yang akan menimbang, dan juga menentukan tanggal Posyandu pada bulan ini akan diadakan kapan. Setelah sudah ditentukan, maka para kader Posyandu dapat langsung membuat dan mencetak selebaran undangan kegiatan Posyandu dan mulai mendistribusikannya ke rumah-rumah di wilayah Posyandu tersebut. Pada hari H Posyandu, itu di Posyandu ada beberapa meja, ada meja pendaftaran, meja pengukuran, meja pencatatan, dan meja penyuluhan. Jadi saat orang tua balita dan anaknya hadir, maka akan langsung diarahkan pada meja pendaftaran, untuk diperiksa apakah sudah terdaftar atau belum orang tua nya. Jika sudah, maka orang tua akan memberikan buku KIA nya, namun jika belum atau warga baru, maka orang tua dapat mendaftarkan diri di posyandu tersebut. Lalu setelah itu, diperiksa juga, apakah anak balita nya juga sudah terdaftar atau belum di Posyandu tersebut, jika belum maka akan didaftarkan terlebih dahulu oleh pihak Posyandu pada lembar register balita nya, jika sudah maka akan dicarikan lembar registernya.

Jika proses pendaftaran sudah selesai, maka berikutnya akan dilakukan proses pengukuran, untuk tinggi badan, berat badan, serta lingkar kepala balita tersebut oleh kader Posyandu, dan hasil pengukuran tersebut akan diberikan kepada kader yang lain untuk dicatat pada lembar register pelayanan dan penimbangan milik balita tersebut maupun pada buku KIA khususnya pada KMS atau Kartu Menuju Sehat dan



hasil pencatatan tersebut kemudian diberikan lagi ke kader Posyandu berikutnya untuk dicatat di lembar rekap lainnya yang diberikan oleh saya berdasarkan format dari Puskesmas agar memudahkan saya untuk nantinya merekap dan membuatkan laporannya. Terakhir adalah melakukan penulisan kesehatan kepada orang tua balita terkait program kesehatan dari pemerintah, maupun konsultasi apabila ada permasalahan terkait kesehatan atau pertumbuhan anak balitanya.

Pewawancara: Baik bu terima kasih untuk perjabarannya, lalu setelah kegiatan Posyandu selesai, hasil pencatatan tadi proses merekap dan pembuatan laporannya bagaimana ya bu?

Narasumber: Nah jadi kan semisal sudah selesai proses pelayanannya di semua Posyandu yang saya pegang di bulan itu, maka semua hasil pencatatannya akan diberikan ke saya oleh masing-masing Posyandu, nantinya akan saya pindahkan satu per satu dari lembar rekap tersebut pada komputer saya di aplikasi Microsoft Excel untuk nantinya menghasilkan laporan gizi di wilayah tersebut pada bulan itu. Nantinya jika sudah saya buat kan, saya setor setiap bulannya ke pinak Puskesmas melalui *flash disk*.


Pewawancara: Selama ini untuk proses pencatatan dan pelaporannya itu, ada permasalahan yang mungkin di Posyandu atau ibu sendiri rasakan tidak ya?

Narasumber: Oh ya ada mas, saya sebagai bidan kan bertanggung jawab untuk mengawasi posyandu-posyandu di desa ini dan juga membuatkan laporan hasil pelayanan gizi setiap bulannya ke pihak Puskesmas, nah jadi karena semua proses pencatatan pelayanan posyandu ini masih dilakukan menggunakan kertas ini, saya setiap bulan harus secara manual satu per satu menindahkan dari lembar rekap tersebut ke komputer saya, yang tentunya sangat memakan waktu dan rentan untuk saya bisa melakukan kesalahan karena banyaknya data yang harus dipindahkan ke dalam komputer mas.

Pewawancara: Baik bu, terima kasih banyak untuk pemaparan dan penjelasannya, saya akhiri wawancara nya disini ya bu, selamat siang.

Narasumber: Oke mas, sama-sama.

Kred: 6 September 2019


Komarah Jusipiah
(Bidan Desa Kred)



LAMPIRAN B STANDAR ANTROPOMETRI PENILAIAN STATUS GIZI PADA ANAK BALITA

B.1 Standar Berat Badan menurut Umur (BB/U) Balita Laki-Laki Umur 0 – 10 bulan

Umur (Bulan)	-3 SD	-2 SD	-1 SD	Median	1 SD	2 SD	3 SD
0	2.9	3.0	3.1	3.2	3.3	3.4	3.5
1	3.9	4.1	4.3	4.5	4.7	4.9	5.1
2	4.8	5.1	5.4	5.6	5.9	6.2	6.6
3	5.5	5.9	6.3	6.7	7.1	7.5	8.0
4	6.2	6.7	7.2	7.6	8.1	8.6	9.1
5	6.7	7.3	7.9	8.4	8.9	9.4	9.9
6	7.2	7.9	8.5	9.1	9.7	10.3	10.9
7	7.4	8.1	8.8	9.4	10.0	10.6	11.2
8	7.5	8.2	8.9	9.5	10.1	10.7	11.3
9	7.6	8.3	9.0	9.6	10.2	10.8	11.4
10	7.7	8.4	9.1	9.7	10.3	10.9	11.5

B.2 Standar Berat Badan menurut Umur (BB/U) Balita Perempuan Umur 0 – 10 bulan

Umur (Bulan)	-3 SD	-2 SD	-1 SD	Median	1 SD	2 SD	3 SD
0	2.8	2.8	2.8	2.8	2.8	2.8	2.8
1	3.6	3.7	3.8	3.9	4.0	4.1	4.2
2	4.5	4.6	4.7	4.8	4.9	5.0	5.1
3	5.3	5.4	5.5	5.6	5.7	5.8	5.9
4	6.1	6.2	6.3	6.4	6.5	6.6	6.7
5	6.8	6.9	7.0	7.1	7.2	7.3	7.4
6	7.5	7.6	7.7	7.8	7.9	8.0	8.1
7	8.2	8.3	8.4	8.5	8.6	8.7	8.8
8	8.9	9.0	9.1	9.2	9.3	9.4	9.5
9	9.6	9.7	9.8	9.9	10.0	10.1	10.2
10	10.3	10.4	10.5	10.6	10.7	10.8	10.9

B.3 Standar Panjang Badan menurut Umur (PB/U) Balita Laki-Laki Umur 0 – 10 bulan

Umur (Bulan)	-3 SD	-2 SD	-1 SD	Median	1 SD	2 SD	3 SD
0	44.2	44.3	44.4	44.5	44.6	44.7	44.8
1	52.5	52.6	52.7	52.8	52.9	53.0	53.1
2	59.2	59.3	59.4	59.5	59.6	59.7	59.8
3	65.3	65.4	65.5	65.6	65.7	65.8	65.9
4	71.0	71.1	71.2	71.3	71.4	71.5	71.6
5	76.8	76.9	77.0	77.1	77.2	77.3	77.4
6	82.5	82.6	82.7	82.8	82.9	83.0	83.1
7	88.0	88.1	88.2	88.3	88.4	88.5	88.6
8	93.4	93.5	93.6	93.7	93.8	93.9	94.0
9	98.7	98.8	98.9	99.0	99.1	99.2	99.3
10	104.0	104.1	104.2	104.3	104.4	104.5	104.6



B.4 Standar Panjang Badan menurut Umur (PB/U) Balita Perempuan Umur 0 – 10 bulan

Umur (Bulan)	Panjang Badan (cm)						
	-3 SD	-2 SD	-1 SD	Median	1 SD	2 SD	3 SD
0	43.6	45.4	47.3	49.1	51.0	52.9	54.7
1	46.4	48.8	51.7	53.7	55.6	57.5	59.9
2	51.0	53.0	55.0	57.1	59.1	61.1	63.2
3	53.5	55.6	57.7	59.8	61.9	64.0	66.1
4	55.9	58.2	60.4	62.7	64.9	67.2	69.5
5	57.4	59.6	61.8	64.0	66.2	68.5	70.7
6	58.9	61.1	63.5	65.7	68.0	70.3	72.6
7	60.3	62.7	65.0	67.3	69.6	71.9	74.2
8	61.7	64.1	66.4	68.7	71.1	73.4	75.8
9	62.9	65.3	67.7	70.1	72.6	74.9	77.4
10	64.1	66.5	69.0	71.5	73.9	76.4	78.9

B.5 Standar Tinggi Badan menurut Umur (TB/U) Balita Laki-Laki Umur 24 – 35 bulan

Umur (Bulan)	Tinggi Badan (cm)						
	-3 SD	-2 SD	-1 SD	Median	1 SD	2 SD	3 SD
24	76.0	77.3	78.4	79.1	80.2	81.2	82.3
25	76.8	78.1	79.1	80.1	81.1	82.1	83.3
26	77.3	78.5	79.5	80.5	81.5	82.5	83.3
27	77.5	78.7	79.7	80.7	81.7	82.7	83.3
28	78.0	79.2	80.1	81.1	82.1	83.1	84.0
29	78.5	79.7	80.6	81.6	82.6	83.6	84.5
30	78.7	79.9	80.8	81.8	82.8	83.8	84.7
31	78.9	80.1	81.0	82.0	83.0	84.0	85.0
32	79.1	80.3	81.2	82.2	83.2	84.2	85.2
33	79.3	80.5	81.4	82.4	83.4	84.4	85.4
34	79.5	80.7	81.6	82.6	83.6	84.6	85.6
35	79.7	80.9	81.8	82.8	83.8	84.8	85.8

B.6 Standar Tinggi Badan menurut Umur (TB/U) Balita Perempuan Umur 24 – 35 bulan

Umur (Bulan)	Tinggi Badan (cm)						
	-3 SD	-2 SD	-1 SD	Median	1 SD	2 SD	3 SD
24	76.0	77.3	78.4	79.1	80.2	81.2	82.3
25	76.8	78.1	79.1	80.1	81.1	82.1	83.3
26	77.5	78.8	79.7	80.7	81.7	82.7	83.3
27	78.1	79.4	80.3	81.3	82.3	83.3	84.0
28	78.5	79.8	80.7	81.7	82.7	83.7	84.5
29	79.0	80.3	81.2	82.2	83.2	84.2	85.0
30	79.5	80.8	81.7	82.7	83.7	84.7	85.5
31	80.0	81.3	82.2	83.2	84.2	85.2	86.0
32	80.5	81.8	82.7	83.7	84.7	85.7	86.5
33	81.0	82.3	83.2	84.2	85.2	86.2	87.0
34	81.5	82.8	83.7	84.7	85.7	86.7	87.5
35	82.0	83.3	84.2	85.2	86.2	87.2	88.0



B.7 Standar Berat Badan menurut Panjang Badan (BB/PB) Balita Laki-Laki 45 – 50 cm (untuk balita umur 0 - 24 bulan)

Panjang Badan (cm)	Berat Badan (kg)						
	-3 SD	-2 SD	-1 SD	Median	1 SD	2 SD	3 SD
45.0	1.9	2.0	2.2	2.4	2.7	3.0	3.3
45.5	1.9	2.1	2.3	2.5	2.8	3.1	3.4
46.0	2.0	2.2	2.4	2.6	2.9	3.2	3.5
46.5	2.1	2.3	2.5	2.7	3.0	3.2	3.6
47.0	2.2	2.4	2.6	2.8	3.0	3.3	3.7
47.5	2.3	2.5	2.7	2.9	3.1	3.4	3.8
48.0	2.3	2.6	2.7	2.9	3.2	3.5	3.9
48.5	2.4	2.7	2.9	3.1	3.3	3.6	4.0
49.0	2.4	2.8	2.9	3.1	3.4	3.6	4.2
49.5	2.5	2.9	3.0	3.2	3.5	3.8	4.3
50.0	2.6	3.0	3.0	3.3	3.6	3.9	4.4

B.8 Standar Berat Badan menurut Panjang Badan (BB/PB) Balita Perempuan 45 – 50 cm (untuk balita umur 0 - 24 bulan)

Panjang Badan (cm)	Berat Badan (kg)						
	-3 SD	-2 SD	-1 SD	Median	1 SD	2 SD	3 SD
45.0	1.9	2.0	2.2	2.4	2.7	3.0	3.3
45.5	2.0	2.1	2.3	2.5	2.8	3.1	3.4
46.0	2.0	2.2	2.4	2.6	2.9	3.2	3.5
46.5	2.1	2.3	2.5	2.7	3.0	3.3	3.6
47.0	2.2	2.4	2.6	2.8	3.0	3.3	3.7
47.5	2.2	2.5	2.6	2.9	3.2	3.5	3.8
48.0	2.3	2.5	2.7	3.0	3.3	3.6	4.0
48.5	2.4	2.6	2.8	3.1	3.4	3.7	4.1
49.0	2.4	2.8	2.9	3.2	3.5	3.8	4.2
49.5	2.5	2.9	3.0	3.3	3.6	3.9	4.3
50.0	2.6	3.0	3.1	3.4	3.7	4.0	4.5

B.9 Standar Berat Badan menurut Tinggi Badan (BB/TB) Balita Laki-Laki 65 – 70 cm (untuk balita umur > 24 bulan)

Tinggi Badan (cm)	Berat Badan (kg)						
	-3 SD	-2 SD	-1 SD	Median	1 SD	2 SD	3 SD
65.0	6.0	6.3	6.7	7.1	8.1	9.3	9.6
65.5	6.0	6.4	7.0	7.6	8.2	9.4	9.9
66.0	6.1	6.5	7.1	7.7	8.5	9.7	10.1
66.5	6.1	6.6	7.2	7.8	8.5	9.2	10.1
67.0	6.2	6.7	7.3	8.0	8.8	9.5	10.4
67.5	6.3	6.8	7.4	8.0	8.7	9.5	10.4
68.0	6.4	6.9	7.5	8.1	8.8	9.6	10.5
68.5	6.4	7.0	7.6	8.2	9.0	9.7	10.6
69.0	6.6	7.1	7.7	8.4	9.1	9.9	10.8
69.5	6.6	7.2	7.8	8.5	9.2	10.0	10.9
70.0	6.8	7.3	7.9	8.6	9.3	10.2	11.1



B.10 Standar Berat Badan menurut Tinggi Badan (BB/TB) Balita Perempuan 65 – 70 cm (untuk balita umur > 24 bulan)

Tinggi Badan (cm)	Berat Badan (kg)						
	-3 SD	-2 SD	-1 SD	Median	1 SD	2 SD	3 SD
65.0	5.6	6.1	6.6	7.2	7.9	8.7	9.7
65.5	5.7	6.2	6.7	7.4	8.1	8.9	9.8
66.0	5.8	6.3	6.8	7.5	8.2	9.0	10.0
66.5	5.8	6.4	6.9	7.6	8.3	9.1	10.1
67.0	5.9	6.4	7.0	7.7	8.4	9.2	10.2
67.5	6.0	6.5	7.1	7.8	8.5	9.3	10.3
68.0	6.1	6.6	7.2	7.9	8.7	9.5	10.5
68.5	6.2	6.7	7.3	8.0	8.8	9.6	10.6
69.0	6.3	6.8	7.4	8.1	8.9	9.8	10.8
69.5	6.3	6.9	7.5	8.2	9.0	9.9	10.9
70.0	6.4	7.0	7.6	8.3	9.1	10.0	11.0



LAMPIRAN C BERITA ACARA

C.1 Berita Acara Observasi Posyandu

BERITA ACARA OBSERVASI

Pada hari Jumat, tanggal 6 bulan September tahun 2019 telah dilakukan kegiatan observasi penelitian skripsi yang dilakukan di Posyandu Ballita Pos IB "Srikaya" Dusun Krajan Desa Krebet, Kecamatan Bululawang, Kabupaten Malang, Jawa Timur, oleh

Nama : Timothy Julian

NIM : 105150400111057

Asal Instansi : Fakultas Ilmu Komputer, Universitas Brawijaya, Malang

Kegiatan : Observasi dan diskusi mengenai proses pelayanan kesehatan ballita di Posyandu Srikaya, Desa Krebet

Foto Dokumentasi:



Krebet, 6 September 2019

(Handwritten signature)

Koordinator Posyandu

Bidan Desa Krebet



C.2 Berita Acara Observasi Posyandu

BERITA ACARA OBSERVASI

Pada hari Sabtu, tanggal 5 bulan Oktober, tahun 2019, telah dilakukan kegiatan observasi penelitian skripsi yang dilakukan di Posyandu Balita Pos IB "Srikaya" Dusun Krajan, Desa Kerebet, Kecamatan Bululawang, Kabupaten Malang, Jawa Timur, oleh

Nama : Timothy Julian

NIM : 165150400111057

Asal Instansi : Fakultas Ilmu Komputer, Universitas Brawijaya, Malang

Kegiatan : observasi dan diskusi dengan bidan desa mengenai

Proses Pelayanan dan Pelaporan gizi anak balita

Foto Dokumentasi:



Kerebet, 5 Oktober 2019

[Signature]
Kemandi Jusofitas
Bidan Desa Kerebet



C.3 Berita Acara Pengujian UAT Kepada Pihak Posyandu

BERITA ACARA OBSERVASI

Pada hari Sabtu, tanggal 8 bulan Februari tahun 2020 telah dilakukan kegiatan observasi penelitian skripsi yang dilakukan di Posyandu Balita Pos IB "Srikaya" Dusun Krajan, Desa Krebbe, Kecamatan Bululawang, Kabupaten Malang, Jawa Timur, oleh

Nama : Timothy Julian

NIM : 165150400111057

Asal Instansi : Fakultas Ilmu Komputer, Universitas Brawijaya, Malang

Kegiatan : Pengujian Sistem, demonstrasi Sistem, serta user acceptance
Testing kepada badan Siterak Posyandu

Foto Dokumentasi:



Krebbe, 8 Februari 2020

(Amir Huda Sari)



LAMPIRAN D VALIDASI KEBUTUHAN

D.1 Diskusi dan validasi kebutuhan dengan Bapak Intan Yusuf Habiebie S.Gz., M.Sc.



D.2 Contoh *Prototype* Untuk Validasi Kebutuhan





D.3 Dokumen Kebutuhan Sistem

DOKUMEN KEBUTUHAN SISTEM

User Sistem:

User	Deskripsi
Guest	Guest merupakan pengguna yang bisa melakukan registrasi sebagai bidan dan melakukan autentikasi ke dalam sistem.
Admin	Admin merupakan pengguna yang memiliki akses untuk melihat dan memvalidasi bidan yang mendaftar pada sistem.
Posyandu	Posyandu merupakan pengguna yang memiliki hak akses untuk melakukan proses pendaftaran dan pengelolaan data orang tua serta balita, dan melakukan pengelolaan pencatatan hasil pelayanan kesehatan balita.
Bidan	Bidan merupakan pengguna yang memiliki hak akses dalam melihat laporan gizi dan hasil pelayanan kesehatan posyandu serta mengelola posyandu dan kader yang terdaftar pada sistem.

Kebutuhan fungsional sistem:

User	Kebutuhan
Guest	Login.
	Registrasi bidan baru.
	Menampilkan data bidan aktif.
Admin	Memperbaharui data bidan aktif.
	Menghapus data bidan aktif.
	Menampilkan data bidan yang perlu divalidasi.
Posyandu	Memvalidasi data bidan yang mendaftar.
	Meniadakan hari & kegiatan posyandu.
	Mendaftarkan akun baru untuk orang tua balita.
	Menampilkan data orang tua balita yang telah terdaftar.
	Memperbaharui data orang tua balita.
	Menghapus data orang tua balita.
	Mendaftarkan balita baru.



	Menampilkan data balita yang telah terdaftar.
	Memperbaharui data balita.
	Menghapus data balita.
	Mendata kehadiran kader posyandu.
	Mencatat kehadiran dan hasil pelayanan kesehatan balita.
	Mendaftarkan posyandu baru.
	Mendaftarkan kader posyandu baru.
	Menampilkan data posyandu dan kader posyandu yang telah terdaftar.
Bidan	Memperbaharui data posyandu.
	Menghapus data posyandu.
	Memperbaharui data kader posyandu.
	Menghapus data kader posyandu.
	Membuat laporan gizi dan hasil pelayanan posyandu.
	Mengunduh laporan gizi dan hasil pelayanan posyandu dalam format Microsoft Excel.
	Menampilkan profil balita dan riwayat pemeriksaan kesehatan.
Posyandu & Bidan	Memperbaharui data riwayat pemeriksaan kesehatan.
	Menghapus data riwayat pemeriksaan kesehatan.
	Menampilkan pertumbuhan berat badan anak balita dalam bentuk kartu Menuju Sehat (KMS).
	Memberikan saran/rekomendasi gizi.
Admin, Posyandu, Bidan	Logout

Kebutuhan non fungsional sistem:

Kebutuhan Non Fungsional	Deskripsi
Usability	Sistem yang memiliki tampilan sederhana dan secara jelas menggambarkan fungsi yang dapat dilakukan, sehingga pengguna dapat dengan mudah mempelajari dan menggunakan sistem ini.

Saya yang bertanda tangan dibawah ini, dengan ini menyetujui semua isi yang dijabarkan pada dokumen kebutuhan sistem ini.

Malang, 20 Oktober 2019

Intan Yusuf Habibia



LAMPIRAN E DATA PENDUKUNG

E.1 Contoh Laporan Pengukuran Balita Menggunakan Microsoft Excel

DATA HASIL PENGUKURAN BALITA		KEMBALI																
DESA	: KREBET	Bakim = Keluarga miskin																
TAHUN	: 2019	G. No. 100 Orang Tua																
No	Nama Balita	Jenis Kelamin	Lahir	NIK	ANAK	SPES	IBU	AWA	NO	Tg	Um	BB	HT	NTD	W	NTD	W	
17	AFRIDA FAUZIL	1	2	1-2018	5	50071500140702100001	1	Tri Wj Laili Sufitau	5070190	500714400100100000	Krebet RT 12/11	11	13	5	45	161	107	N
18	Alfa Nur	1	1	1-2018	5	50071500140702100001	1	Tri Wj Laili Sufitau	5070190	500714400100100000	Krebet RT 12/11	11	13	5	45	161	107	N
19	SAVILLI SYTA ALIA	1	21	1-2018	5	50071500140702100001	3	noel Waluyo H Alsan	5007184	500714010010010000	Krebet RT 12/10	10	13	5	45	154	60	N
20	RIZKA DEVA SUGITA	1	24	1-2018	5	50071500140702100001	2	M. Yusuf Novia Re	5007140	5008134411970000	Krebet RT 13/13	13	13	5	44	164	60	N
21	INDRA PRADANA W	1	21	1-2018	5	50071500140702100001	1	noel Waluyo H Alsan	5007184	500714010010010000	Krebet RT 12/10	10	13	5	45	154	60	N
22	INDRA PRADANA W	1	21	1-2018	5	50071500140702100001	1	noel Waluyo H Alsan	5007184	500714010010010000	Krebet RT 12/10	10	13	5	45	154	60	N
23	W. Nikka Rahayu Anaskey	1	20	1-2018	5	50071500140702100001	1	Bagus Say Nina Ima	5073002	5007145009020000	Krebet RT 15/14	14	13	4	42	141	111	N
24	W. NURAS	1	18	1-2018	5	50071500140702100001	1	Bagus Say Nina Ima	5073002	5007145009020000	Krebet RT 15/14	14	13	4	42	141	111	N
25	RENDAWAN R	1	1	1-2018	5	50071500140702100001	1	Andi AAN Iko Koe	5007102	500714400100100000	Krebet RT 11/12	12	13	5	40	160	60	N
26	Andi Nurmay	1	1	1-2018	5	50071500140702100001	1	Andi AAN Iko Koe	5007102	500714400100100000	Krebet RT 11/12	12	13	5	40	160	60	N
27	Andi Nurmay	1	1	1-2018	5	50071500140702100001	1	Andi AAN Iko Koe	5007102	500714400100100000	Krebet RT 11/12	12	13	5	40	160	60	N
28	Andi Nurmay	1	1	1-2018	5	50071500140702100001	1	Andi AAN Iko Koe	5007102	500714400100100000	Krebet RT 11/12	12	13	5	40	160	60	N
29	ANAN ESAR W	1	1	1-2018	5	50071500140702100001	1	Andi AAN Iko Koe	5007102	500714400100100000	Krebet RT 11/12	12	13	5	40	160	60	N
30	ANAN ESAR W	1	1	1-2018	5	50071500140702100001	1	Andi AAN Iko Koe	5007102	500714400100100000	Krebet RT 11/12	12	13	5	40	160	60	N
31	ANAN ESAR W	1	1	1-2018	5	50071500140702100001	1	Andi AAN Iko Koe	5007102	500714400100100000	Krebet RT 11/12	12	13	5	40	160	60	N
32	ANAN ESAR W	1	1	1-2018	5	50071500140702100001	1	Andi AAN Iko Koe	5007102	500714400100100000	Krebet RT 11/12	12	13	5	40	160	60	N
33	ANAN ESAR W	1	1	1-2018	5	50071500140702100001	1	Andi AAN Iko Koe	5007102	500714400100100000	Krebet RT 11/12	12	13	5	40	160	60	N
34	ANAN ESAR W	1	1	1-2018	5	50071500140702100001	1	Andi AAN Iko Koe	5007102	500714400100100000	Krebet RT 11/12	12	13	5	40	160	60	N

E.2 Contoh Laporan Kehadiran Kader

DATA JUMLAH KADER DAN KADER AKTIF PADA MASING-MASING POSYANDU														
NO	NAMA POSYANDU	JUMLAH KADER	JUMLAH KADER AKTIF											
			AGUSTUS	SEPTEMBER	OKTUBER	NOVEMBER	DESEMBER	JANUARI	FEBRUARI	MARET	APRIL	MAY	JUNI	JULI
1	ANGGUR	5	5	5	5	5	5	5	5	5	5	5	5	5
2	SRIKAYA	5	5	5	5	5	5	5	5	5	5	5	5	5
3	MANGGA	5	5	5	5	5	5	5	5	5	5	5	5	5
4	DELIMA	5	5	5	5	5	5	5	5	5	5	5	5	5
5	SALAK	5	5	5	5	5	5	5	5	5	5	5	5	5
6	PISANG	5	5	5	5	5	5	5	5	5	5	5	5	5
7	MANGGIS	5	5	5	5	5	5	5	5	5	5	5	5	5

E.3 Contoh Undangan Kegiatan Posyandu

UNDANGAN

Kepada Yth.
Ibu Hamil, Ibu Menyusui Balita
Pos Srikaya 1b Desa Krebet

.....
Di Tempat

Assalamualaikum Wr. Wb.

Dalam rangka untuk meningkatkan kesehatan warga dilingkungan POS SRIKAYA umumnya, kami selaku pengurus Posyandu SRIKAYA mengundang Ibu Hamil, Ibu Menyusui Balita untuk berkenan hadir di Posyandu Pada:

Hari/ Tanggal : Sabtu, 5 Oktober 2019
Jam : 09.00 WIB
Tempat : Rumah Ibu Khoiriyah (RT 17)
JL. Tugu Hitam III Desa Krebet

Acara : Penimbangan, Imunisasi
Pemeriksaan Kesehatan
Balita dan Ibu Hamil,
Ibu Menyusui, Pemberian
Makanan Tambahan Balita.

Demikian undangan ini kami sampaikan, besar harapan kami agar Ibu/Saudari dapat hadir tepat waktu.

Atas perhatian dan kehadirannya kami ucapkan terima kasih.

Wassalamu'alaikum Wr. Wb.

Tembusan
1. PKK RT/RW Desa Krebet.
2. PKK Desa Krebet.
3. TIM Kesehatan Desa Krebet.



E.4 Contoh Surat Registrasi Bidan



E.5 Contoh Register Penimbangan Balita

KARTI PENIMBANGAN KREBET - BULULAWANG
KREBET - BULULAWANG

REGISTER PELAYANAN KESEHATAN
POS SRIKAYA - 13 DESA KREBET KEC. BULULAWANG

No	Tgl	Posyandu	TB	BB	LK	YA	TDK	FEB	AGST	TTD PETUGAS	No	TGL	KELUHAN & HASIL PEMERIKSAAN			PERAPA
													BB	TENSI	DIAGNOSA	
1	12-01-2019	127	17	57	28,5											
2	12-01-2019															



LAMPIRAN F HASIL PENGUJIAN USER ACCEPTANCE TESTING (UAT)

F.1 Hasil Pengujian UAT Pada Admin

Kuesioner Pengujian User Acceptance Testing

Nama: Intan Susil Hanibie

Jabatan/Posisi: Dosen Jurusan Gezi UB

Beri tanda centang (✓) pada kolom penilaian untuk setiap pertanyaan berikut

No.	Pertanyaan	Jawaban				
		Sangat Setuju	Setuju	Netral	Tidak Setuju	Sangat Tidak Setuju
1.	Apakah sistem informasi ini dapat membantu anda dalam mengelola bidang yang terdaftar?		✓			
2.	Apakah sistem informasi ini dapat membantu anda memvalidasi data bidang yang mendaftarkan?		✓			
3.	Apakah sistem informasi ini mudah untuk digunakan?		✓			
4.	Apakah sistem informasi ini mudah untuk dipelajari?	✓				

Makassar, 15 Februari 2020


(Intan Susil Hanibie)



F.2 Hasil Pengujian UAT Pada Bidang

Kuesioner Pengujian User Acceptance Testing

Nama: Ani Indah Sari

Jabatan/Posisi: Perawat Desa Krebet (Asstn. Bidan)

Beri tanda centang (✓) pada kolom penilaian untuk setiap pertanyaan berikut

No.	Pertanyaan	Jawaban				
		Sangat Setuju	Setuju	Netral	Tidak Setuju	Sangat Tidak Setuju
1.	Apakah sistem informasi ini dapat membantu anda dalam membuat laporan gizi balita dan hasil pelayanan posyandu?	✓				
2.	Apakah sistem informasi ini dapat membantu anda dalam mengelola posyandu dan kader?	✓				
3.	Apakah sistem informasi ini mudah untuk digunakan?		✓			
4.	Apakah sistem informasi ini mudah untuk dipelajari?	✓				

Krebet, 08.02.2020

(Ani Indah Sari)



F.3 Hasil Pengujian UAT Pada Kader Posyandu

Kuesioner Pengujian User Acceptance Testing

Nama: Mari Kusuma Wahyuning Pratiwi
 Jabatan/Posisi: Kader Konselor (Sikop) IB

Beri tanda centang (✓) pada kolom penilaian untuk setiap pertanyaan berikut

No.	Pertanyaan	Jawaban				
		Sangat Setuju	Setuju	Netral	Tidak Setuju	Sangat Tidak Setuju
1.	Apakah sistem informasi ini dapat membantu anda dalam menjadwalkan dan mengirimkan undangan kegiatan posyandu?	✓				
2.	Apakah sistem informasi ini dapat membantu anda dalam melakukan pencatatan hasil pengukuran dan pemeriksaan balita?	✓				
3.	Apakah sistem informasi ini dapat membantu anda dalam mengelola data orang tua balita?	✓				
4.	Apakah sistem informasi ini dapat membantu anda dalam mengelola data balita yang ada pada posyandu?	✓				
5.	Apakah sistem informasi ini dapat membantu anda dalam mendata kehadiran kader posyandu dalam kegiatan hari posyandu?	✓				
6.	Apakah sistem informasi ini mudah untuk digunakan?	✓				
7.	Apakah sistem informasi ini mudah untuk dipelajari?	✓				

Kred, 8 Feb 2020

(M. Wahyuning Pratiwi)



Kuesioner Pengujian User Acceptance Testing

Nama:

B.H. Khoiriyah

Jabatan/Posisi:

Kader Balita (Stekaya B)

Beri tanda centang (✓) pada kolom penilaian untuk setiap pertanyaan berikut.

No.	Pertanyaan	Jawaban				
		Sangat Setuju	Setuju	Netral	Tidak Setuju	Sangat Tidak Setuju
1.	Apakah sistem informasi ini dapat membantu anda dalam menjadwalkan dan mengirimkan undangan kegiatan posyandu?	✓				
2.	Apakah sistem informasi ini dapat membantu anda dalam melakukan pencatatan hasil pengukuran dan pemeriksaan balita?	✓				
3.	Apakah sistem informasi ini dapat membantu anda dalam mengelola data orang tua balita?	✓				
4.	Apakah sistem informasi ini dapat membantu anda dalam mengelola data balita yang ada pada posyandu?	✓				
5.	Apakah sistem informasi ini dapat membantu anda dalam mendata kehadiran kader posyandu dalam kegiatan hari posyandu?	✓				
6.	Apakah sistem informasi ini mudah untuk digunakan?	✓				
7.	Apakah sistem informasi ini mudah untuk dipelajari?	✓				

Kredip, 8 Feb 2020


(B.H. Khoiriyah)



Kuesioner Pengujian User Acceptance Testing

Nama: KO SADAH

Jabatan/Posisi: KADER POSYANDU

Beri tanda centang (✓) pada kolom penilaian untuk setiap pertanyaan berikut

No.	Pertanyaan	Jawaban				
		Sangat Setuju	Setuju	Netral	Tidak Setuju	Sangat Tidak Setuju
1.	Apakah sistem informasi ini dapat membantu anda dalam menjadwalkan dan mengirimkan undangan kegiatan posyandu?					
2.	Apakah sistem informasi ini dapat membantu anda dalam melakukan pencatatan hasil pengukuran dan pemeriksaan balita?	✓				
3.	Apakah sistem informasi ini dapat membantu anda dalam mengelola data orang tua balita?	✓	✓			
4.	Apakah sistem informasi ini dapat membantu anda dalam mengelola data balita yang ada pada posyandu?	✓				
5.	Apakah sistem informasi ini dapat membantu anda dalam mendata kehadiran kader posyandu dalam kegiatan hari posyandu?		✓			
6.	Apakah sistem informasi ini mudah untuk digunakan?					
7.	Apakah sistem informasi ini mudah untuk dipelajari?		✓			

KKeTel.108-02-2020

(Signature)
(.....K.O.SADAH.....)