repository.ub.ac.id

PENGEMBANGAN SISTEM INFORMASI AUTOPSI VERBAL (STUDI KASUS : DINAS KESEHATAN KABUPATEN MALANG)

SKRIPSI

Untuk memenuhi sebagian persyaratan
memperoleh gelar Sarjana Komputer

Disusun oleh:
Muhammad Ridho
NIM: 155150401111100

PROGRAM STUDI SISTEM INFORMASI
JURUSAN SISTEM INFORMASI
FAKULTAS ILMU KOMPUTER
UNIVERSITAS BRAWIJAYA
MALANG
2019

UNIVERSITAS BRAWIJAYA

PENGESAHAN

PENGEMBANGAN SISTEM INFORMASI AUTOPSI VERBAL
(STUDI KASUS : DINAS KESEHATAN KABUPATEN MALANG)

SKRIPSI

Diajukan untuk memenuhi sebagian persyaratan
memperoleh gelar Sarjana Komputer

Disusun Oleh :
Muhammad Ridho
NIM: 155150401111100

Skripsi ini telah diuji dan dinyatakan lulus pada
24 Juli 2019
Telah diperiksa dan disetujui oleh:

Pembimbing I	Pembimbing II
Ismiarta Aknuranda, S.T., M.Sc., Ph.D.	Lutfi Fanani, S.Kom., M.T., M.Sc.
NIK: 201006 740719 1 001	NIK: 201607 890217 1 001

Mengetahui,
Ketua Jurusan Sistem Informasi

Dr. Eng. Herman Tolle, S.T., M.T.
NIP: 19740823 200012 1 001

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 referensi.

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, 26 Juli 2019

Muhammad Ridho

NIM: 155150401111100

PRAKATA

Puji syukur kehadirat Allah SWT yang telah melimpahkan rahmat, taufik dan hidayah-Nya sehingga memberikan kemudahan dan kelancaran dalam menyelesaikan skripsi yang berjudul ““Pengembangan Sistem Informasi Autopsi Verbal (Studi Kasus : Dinas Kesehatan Kabupaten Malang)”. Sebagai syarat untuk menyelesaiakan Program Sarjana (S1) pada Program Sarjana Fakultas Ilmu Komputer Jurusan Sistem Informasi Universitas Brawijaya.

Penulis menyadari bahwa penulisan ini tidak dapat terselesaikan tanpa dukungan dari berbagai pihak baik moril maupun materil. Oleh karena itu, penulis ingin menyampaikan rasa hormat dan ucapan terima kasih kepada semua pihak yang telah membantu dalam penyusunan skripsi ini terutama kepada:

1. Allah SWT yang dengan hidayah dan kuasa-Nya, penulis tidak akan mampu menyelesaikan skripsi,
2. Kedua Orang tua penulis yang telah memberikan dukungan baik moril maupun materil serta doa yang tiada henti-hentinya kepada penulis,
3. Bapak Wayan Firdaus Mahmudy, S.Si, M.T, Ph.D selaku Dekan Fakultas Ilmu Komputer Universitas Brawijaya,
4. Bapak Herman Tolle, Dr Eng., S.T, M.T. selaku Ketua Jurusan Sistem Informasi Fakultas Ilmu Komputer Universitas Brawijaya,
5. Bapak Yusi Tyroni Mursityo, S.Kom., M.AB, selaku Ketua Program Studi Sistem Informasi Fakultas Ilmu Komputer Universitas Brawijaya,
6. Bapak Ismiarta Aknuranda, S.T, M.Sc, Ph.D. selaku dosen pembimbing satu yang telah dengan sabar membimbing, memberikan masukan dan mengarahkan penulis sehingga dapat menyelesaikan skripsi ini,
7. Bapak Lutfi Fanani, S.Kom., M.T., M.Sc. selaku dosen pembimbing dua yang telah dengan sabar membimbing, memberikan masukan dan mengarahkan penulis sehingga dapat menyelesaikan skripsi ini,
8. Temen banget Dwi Christina Fauziyah, Rizky Wahyu Setiawan, Ridwan Fajar Widodo, Reinardus Dimas Bagus Kurniawan selaku sahabat penulis yang selalu memberikan semangat, ledekan, kadang traktiran, bantuan serta dukungan baik penuh maupun parsial kepada penulis,
9. Temen dimulai dari Abid, Adi, Agatha, Ahyar, Ammar, Anderson, Anin, Anvel, Arif, Aswin, Atikah, Azizah, Azkia, Bagoes, Bernard, Bhanio, Bintang, Chandra, Chia, Citra, Devita, Dila, Dinda, Edo, Erni, Erwina, Fajar, Faqih, Fauzi, Febrian, Felix, Fian, Filza, Ghazi, Hana, Handhika, Imam, Inggit, Irvan, Jeje, Karita, Kiko, Labina, Lili, Malik, Marisa, Meirina, Nadhiyah, Nafiani, Nilna, Nina, Nofry, Nyoman, Opi, Pandu, Putri, Radea, Rani, Ratna, Rayhan, Restivanny, Reza, Rinda, Rinov, Rohayat, Ryan, Sandi, Sarah, Satrio, Shella, Shiel, Stefanus,

Surya, Vallian, Venzy, Vira, Vita, Wibi, Weni, Widhi, Willy, Wulan, Yauma, Beni, Ulul,Arda, Alfan, Idris, Bibin, Sofi, Mansyur, Gita, Asa, Firman,

10. Kepada keluarga BIG IO yang tidak kenal lelah telah menemani dan menyemangati penulis serta menemani penulis semasa perkuliahan hingga akhir perkuliahan,
11. Seluruh civitas akademika Sistem Informasi Universitas Brawijaya yang telah banyak memberi bantuan dan dukungan selama penyelesaian skripsi ini.

Penulis menyadari bahwa skripsi ini masih jauh dari sempurna dikarenakan terbatasnya pengalaman dan pengetahuan yang dimiliki penulis. Oleh karena itu, penulis mengharapkan segala bentuk saran serta masukan bahkan kritik yang membangun dari berbagai pihak. Semoga skripsi ini dapat bermanfaat bagi para pembaca dan semua pihak khususnya dalam bidang teknologi informasi dan kesehatan.

Malang, 26 Juli 2019

Penulis

ridhomuhmd@gmail.com

repository.ub.ac.id

ABSTRAK

Muhammad Ridho, Pengembangan Sistem Informasi Autopsi Verbal (Studi Kasus : Dinas Kesehatan Kabupaten Malang)

Pembimbing: Ismiarta Aknuranda, S.T, M.Sc, Ph.D dan

Lutfi Fanani, S.Kom., M.T., M.Sc

Dinas Kesehatan Kabupaten Malang memiliki *pilot project* untuk mencatatkan data penyebab kematian di wilayah Gondanglegi dan Kepanjen. Salah satu cara yang bisa dilakukan adalah autopsi verbal yaitu mewawancarai anggota keluarga mengenai tanda dan gejala yang muncul sebelum seseorang meninggal. Proses autopsi verbal yang sedang berjalan masih menggunakan pencatatan secara manual dengan menggunakan kertas sehingga data yang dikumpulkan rawan mengalami kehilangan maupun kerusakan. Selain itu belum terdapat sistem untuk mengolah data kematian yang telah dikumpulkan. Berdasarkan masalah yang diuraikan, maka perlu dibuat sistem informasi untuk mempermudah pencatatan, penyimpanan dan pengolahan data kematian hasil autopsi verbal. Pengembangan sistem informasi menggunakan metode *prototyping* dengan memberikan gambaran sistem secara umum kepada klien untuk mendapatkan umpan balik berupa saran maupun kritik. Pengembangan dimulai dari tahapan pemodelan proses bisnis dan analisis persyaratan, perancangan prototipe, evaluasi prototipe, perancangan lanjutan, implementasi dan pengujian. Implementasi menghasilkan sistem informasi autopsi verbal berbasis web. Sistem ini digunakan untuk mencatat hasil autopsi verbal ke dalam sistem, memverifikasi data hasil autopsi verbal, mengelola data hasil autopsi verbal dan melihat informasi secara umum Selanjutnya hasil implementasi diuji dengan pengujian validasi yang menunjukkan hasil pengujian valid sesuai dengan persyaratan yang telah diidentifikasi, pengujian kompatibilitas peramban menunjukkan terdapat 9% *error issues* dan sistem dapat dijalankan pada beberapa peramban tanpa ditemukan kendala.

Kata kunci: *autopsi verbal, data kematian, prototyping, sistem informasi.*

UNIVERSITAS BRAWIJAYA

repository.ub.ac.id

ABSTRACT

Muhammad Ridho, Development of a Verbal Autopsy Information System (Case Study: Healt Service Office of Malang District)

Supervisors: Ismiarta Aknuranda, S.T, M.Sc, Ph.D and

Lutfi Fanani, S.Kom., M.T., M.Sc

Malang District's health service offices own a pilot project whose objective is to record in the form of causes of death in the region Gondanglegi and Kepanjen. One of the ways to record such data is through verbal autopsy which is done by interviewing family members of the deceased on visible signs or symptoms of a certain individual before death. The ongoing process of the verbal autopsy is still recorded manually using paper, which when gathered is vulnerable to being lost or damaged. Otherwise, there was no system to record the data that has been gathered. Based on the above-mentioned problem, an information system is needed to ease the recording, storing, and processing of data that contains verbal autopsy. The development of the information system uses a prototyping method which presents a picture of the system in general toward the client in order to gain feedback in the form of advice or critique. Development is started with the modeling of the business process and requirements of the analysis, prototype designing, prototype evaluation, continued designing, which is finished with implementation and testing. The implementation results in a web-based verbal autopsy information system. The system is utilized to record, verify, manage, and manage verbal autopsy data and to present general information. The next step is to test the implementation results with validation testing which shows valid test results which are in accord with the identified requirements, tests towards the compatibility of the browser show that the error issues were at 9% and the system was run on a variety of browsers without any problems whatsoever.

Keywords: *verbal autopsy, death data, prototyping, information system.*

UNIVERSITAS BRAWIJAYA

DAFTAR ISI

PERSETUJUAN ii
PERNYATAAN ORISINALITAS iii
PRAKATA iv
ABSTRAK vi
ABSTRACT vii
DAFTAR ISI viii
DAFTAR TABEL xii
DAFTAR GAMBAR xiv
DAFTAR LAMPIRAN xvi
BAB 1 PENDAHULUAN 1
1.1 Latar belakang 1
1.2 Rumusan masalah 3
1.3 Tujuan 3
1.4 Manfaat 4
1.5 Batasan masalah 4
1.6 Sistematika pembahasan 4
BAB 2 LANDASAN KEPUSTAKAAN 6
2.1 Penelitian Sebelumnya 6
2.2 Autopsi Verbal 7
2.3 Sistem Informasi 8
2.4 Rekayasa Perangkat Lunak (*Software Engineering*) 10
2.5 *Prototyping* 10
2.6 Proses Bisnis (*Business Process*) 11
2.7 *Business Process Model and Notation* (BPMN) 12
2.7.1 Flow Objects 13
2.7.2 Connecting Objects 15
2.7.3 Swimlanes 15
2.7.4 Artifacts 16
2.8 Pemodelan *Use Case* 17
2.8.1 Identifikasi Tipe Pemangku Kepentingan 17

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

2.8.2 Analisis Masalah 18
2.8.3 Identifikasi Kebutuhan Pemangku Kepentingan dan Pengguna 19
2.8.4 Identifikasi Fitur 19
2.8.5 Identifikasi Persyaratan Fungsional dan Non-Fungsional 20
2.8.6 Pemodelan *Use Case* 20
2.9 *Unified Modelling Language* (UML) 22
2.9.1 Use Case Diagram 22
2.9.2 Sequence Diagram 23
2.9.3 Class Diagram 24
2.10 Pengujian 26
2.10.1 Pengujian Validasi 27
2.10.2 Pengujian Kompatibilitas 27
BAB 3 METODOLOGI 29
3.1 Pemodelan Proses Bisnis dan Analisis Persyaratan 30
3.2 Perancangan Prototipe 30
3.3 Evaluasi Prototipe 30
3.4 Perancangan Lanjutan 30
3.5 Implementasi 31
3.6 Pengujian 31
3.6.1 Pengujian Validasi (*Validation Testing)* 31
3.6.2 Pengujian Kompatibilitas (*Compatibility Testing*) 31
BAB 4 PEMODELAN PROSES BISNIS DAN ANALISIS PERSYARATAN 32
4.1 Pemodelan Proses Bisnis 32
4.1.1 Analisis Proses Bisnis *As-Is* 32
4.1.2 Analisis Proses Bisnis *To-Be* 34
4.2 Analisis Persyaratan 35
4.2.1 Analisis Permasalahan 36
4.2.2 Identifikasi Tipe Pemangku Kepentingan 36
4.2.3 Identifikasi Pengguna 37
4.2.4 Identifikasi Fitur 38
4.2.5 Persyaratan Fungsional 40

4.2.6 Persyaratan Nonfungsional 42
4.3 Pemodelan *Use Case* 43
4.3.1 Use Case Diagram 44
4.3.2 Deskripsi Aktor 46
4.3.3 Spesifikasi *Use Case* 47
4.4 Perancangan dan Evaluasi Prototipe 58
4.4.1 Perancangan Prototipe 59
4.4.1.1 Prototipe *Low Fidelity* 60
4.4.1.2 Prototype *High Fidelity* 63
4.4.2 Evaluasi Prototipe 66
BAB 5 PERANCANGAN LANJUT 69
5.1 Perancangan Arsitektur Sistem 69
5.2 Pemodelan Kelas 70
5.2.1 Diagram Kelas Analisis 70
5.2.2 Diagram Kelas Perancangan 72
5.3 Perancangan Interaksi Antar Objek 74
5.3.1 Memasukan Data Hasil autopsi Verbal 74
5.3.2 Memverifikasi Data Hasil Autopsi Verbal 75
5.3.3 Menghapus Data Pengguna 76
5.4 Perancangan Basis Data 77
BAB 6 IMPLEMENTASI 79
6.1 Spesifikasi Lingkungan Implementasi 79
6.2 Implementasi Algoritme 80
6.2.1 Memasukan Data Hasil Autopsi Verbal 80
6.2.2 Mengelola Data Hasil Autopsi Verbal 83
6.2.3 Mengelola Data Pengguna 84
6.2.4 Memonitor Informasi Autopsi Verbal 87
6.3 Implementasi Antarmuka Pengguna 88
6.3.1 Antarmuka Dasbor 89
6.3.2 Antarmuka Formulir 89
6.3.3 Antarmuka Verifikasi Data hasil autopsi Verbal 90
6.3.4 Antarmuka Mengelola Data hasil autopsi Verbal 90

6.3.5 Antarmuka Mengelola Data Pengguna 91
BAB 7 PENGUJIAN 92
7.1 Pengujian Validasi 92
7.1.1 Pengujian Validasi Memasukan Data Hasil Autopsi Verbal..... 92
7.1.2 Pengujian Validasi Memverifikasi Data Hasil Autopsi Verbal..... 93
7.1.3 Pengujian Validasi Melihat Detail Data hasil autopsi Verbal..... 93
7.1.4 Pengujian Validasi Memonitor Hasil Autopsi Verbal..... 94
7.2 Pengujian Kompatibilitas Peramban.....95
7.3 Analisis Hasil Pengujian.....96
BAB 8 PENUTUP 97
8.1 Kesimpulan.....97
8.2 Saran 98
DAFTAR REFRENSI 99
LAMPIRAN A TRANSKIP WAWANCARA..... 101
LAMPIRAN B PENGUJIAN VALIDASI..... 102

DAFTAR TABEL

Tabel 2.1 Notasi Event 13
Tabel 2.2 Notasi *Activity* 14
Tabel 2.3 Notasi *Gateway* 14
Tabel 2.4 Notasi *Connecting Objects* 15
Tabel 2.5 Notasi *Swimlanes* 16
Tabel 2.6 Notasi *Artefacts* 16
Tabel 2.7 Analisis Tipe Pemangku Kepentingan 18
Tabel 2.8 Kerangka Dokumentasi Pernyataan Masalah 18
Tabel 2.9 Fitur Sistem 19
Tabel 2.10 Format Dokumentasi Spesifikasi *Use Case* 20
Tabel 2.11 Notasi pada *Sequence Diagram* 23
Tabel 2.12 Notasi pada Class *Diagram* 24
Tabel 4.1 Pernyataan Masalah 36
Tabel 4.2 Jenis Pemangku Kepentingan 37
Tabel 4.3 Hasil Identifikasi Pengguna 38
Tabel 4.4 Hasil Identifikasi Fitur 39
Tabel 4.5 Persyaratan Fungsional 40
Tabel 4.6 Persyaratan Non-Fungsional 43
Tabel 4.7 Hubungan antara *Use Case* dengan Aktivitas Proses Bisnis *To-Be* 45
Tabel 4.8 Hubungan *Use Case* dengan Pemangku Kepentingan 45
Tabel 4.9 Hubungan *Use Case* dengan Fitur 46
Tabel 4.10 Deskripsi Aktor 46
Tabel 4.11 Spesifikasi *Use Case Login* 47
Tabel 4.12 Spesifikasi *Use Case Logout* 49
Tabel 4.13 Spesifikasi *Use Case* Memasukan Data Hasil autopsi Verbal 50
Tabel 4.14 Spesifikasi *Use Case* Melakukan Verifikasi Data Hasil Autopsi Verbal 52
Tabel 4.15 Spesifikasi *Use Case* Mengelola Data Hasil autopsi Verbal 53
Tabel 4.16 Spesifikasi *Use Case* Mengelola Data Pengguna 55
Tabel 4.17 Spesifikasi *Use Case* Memonitor Informasi Autopsi Verbal 58
Tabel 4.18 Hubungan *Use Case* dengan Fitur 59

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

Tabel 6.1 Spesifikasi Perangkat Keras 79
Tabel 6.2 Spesifikasi Perangkat Lunak 79
Tabel 6.3 Implementasi Algoritme Penambahan Data hasil autopsi Verbal 80
Tabel 6.4 Implementasi Algoritme Mengelola Data Hasil Autopsi Verbal..... 84
Tabel 6.5 Implementasi Algoritme Mengelola Data Pengguna 85
Tabel 6.6 Implementasi Memonitor Informasi Autopsi Verbal 87
Tabel 7.1 Pengujian dan Kasus Uji Mencatat Data Hasil Autopsi Verbal..... 92
Tabel 7.2 Pengujian dan Kasus Uji Memverifikasi Data Hasil Autopsi Verbal 93
Tabel 7.3 Pengujian dan Kasus Uji Melihat Detail Data hasil autopsi Verbal 94
Tabel 7.4 Pengujian dan Kasus Uji Melihat Detail Data Autopsi Verbal 94
Tabel 7.5 Jenis Peramban yang Digunakan pada Pengujian kompatibilitas peramban 95

DAFTAR GAMBAR

Gambar 2.1 Skema Alur Terstrukur sebuah Sistem Informasi 8
Gambar 2.2 Komponen yang Ada pada *Computer-Based Information System* 10
Gambar 2.3 *Prototyping Model* 10
Gambar 2.4 Contoh Template *RUP Use Case Specification* 21
Gambar 2.5 Contoh *use case diagram* 23
Gambar 2.6 Contoh Hubungan Antar Kelas 25
Gambar 2.7 Contoh Diagram Kelas pada Tingkat Analisis 26
Gambar 2.8 Contoh hasil pengujian kompatibilitas menggunakan *Sortsite* 28
Gambar 3.1 Diagram Alur Metode Penelitian 29
Gambar 4.1 Proses Bisnis *As-Is* Pengumpulan Data hasil autopsi Verbal 33
Gambar 4.2 Proses Bisnis *To-Be* Pengumpulan Data hasil autopsi Verbal 34
Gambar 4.3 Proses Bisnis *To-Be* Pemantauan Data Hasil Autopsi Verbal 35
Gambar 4.4 Kodifikasi Fitur 39
Gambar 4.5 Kodifikasi Persyaratan Fungsional 40
Gambar 4.6 Kodifikasi Persyaratan Nonfungsional 43
Gambar 4.7 *Use Case Diagram* Autopsi Verbal 44
Gambar 4.8 Prototipe *Low Fidelity* Dasbor 60
Gambar 4.9 Prototipe *Low Fidelity* Data Kematian 61
Gambar 4.10 Prototipe *Low Fidelity* Data Pengguna 62
Gambar 4.11 Prototipe *Low Fidelity* Formulir 63
Gambar 4.12 Prototipe Halaman Dasbor 64
Gambar 4.13 Prototipe Halaman Data Pengguna 64
Gambar 4.14 Prototipe Halaman Data kematian 65
Gambar 4.15 Prototipe Halaman Formulir 65
Gambar 4.16 Evaluasi Prototipe Halaman Dasbor 66
Gambar 4.17 Evaluasi Prototipe Halaman Data Pengguna 67
Gambar 4.18 Evaluasi Prototipe Halaman Data kematian 67
Gambar 4.19 Evaluasi Prototipe Halaman Formulir 68
Gambar 4.20 Evaluasi Prototipe Halaman Verifikator 68
Gambar 5.1 Arsitektur Diagram Sistem Informasi Autopsi Verbal 69

Gambar 5.2 Diagram Kelas Analisis 70
Gambar 5.3 Atribut Kelas Analisis 71
Gambar 5.4 Diagram Kelas Perancangan untuk *Controller* 72
Gambar 5.5 Diagram Kelas Perancangan untuk *Model* 73
Gambar 5.6 Diagram Kelas Perancangan untuk *View* 73
Gambar 5.7 *Sequence Diagram* Menambah Hasil autopsi Verbal 75
Gambar 5.8 *Sequence Diagram* Memverifikasi Data Hasil autopsi Verbal 76
Gambar 5.9 *Sequence* Diagram Menghapus Data Pengguna 77
Gambar 5.10 *Physical Data Model* Sistem Informasi Autopsi Verbal 78
Gambar 6.1 Implementasi Antarmuka *Dashboard* 89
Gambar 6.2 Implementasi Antarmuka Formulir Autopsi Verbal 90
Gambar 6.3 Implementasi Antarmuka Data Kematian 90
Gambar 6.4 Implementasi Antarmuka Data Kematian 91
Gambar 6.5 Implementasi Antarmuka Data Pengguna 91
Gambar 7.1 Kategori *Compatibility Issues* pada SortSite 95
Gambar 7.2 Hasil Pengujian kompatibilitas peramban 96

DAFTAR LAMPIRAN

LAMPIRAN A TRANSKIP WAWANCARA 101
LAMPIRAN B PENGUJIAN VALIDASI 102

BAB 1 PENDAHULUAN

1.1 Latar belakang

Pelaporan dan pengurusan akta kematian di wilayah Malang masih rendah. Eny Hari Sutiarny, Kepala Dispendukcapil Kota Malang mengatakan bahwa masyarakat masih kurang aktif dalam pengurusan surat kematian (Radar Malang, 2018). Padahal pengurusan akta kematian tersebut penting bagi data kependudukan. Data kematian yang terdapat pada Dinas Kependudukan dan Pencatatan Sipil dapat hanya dapat menunjukan data secara kuantitatif. Seharusnya, juga terdapat data kualitatif mengenai kematian dan riwayat kesehatan. Data kuantitatif dapat digunakan untuk melihat tinggi dan rendahnya jumlah kematian suatu wilayah. Data kualitatif dapat digunakan untuk mengetahui penyebab kematian, perilaku almarhum/almarhumah maupun kondisi pada suatu daerah. Akan tetapi, informasi penyebab kematian tidak terekam pada Dinas Kependudukan dan Pencatatan Sipil. Data penyebab kematian dapat memberikan informasi dan rekomendasi bagi pemerintah maupun masyarakat umum untuk mengantisipasi wabah/penyakit yang umumnya menjangkit wilayah tertentu. Data penyebab kematian tersebut dapat dijadikan rekomendasi bagi kebijakan selanjutnya. Laporan yang lengkap dan tercatat dapat digunakan untuk kemudian ditangani secara medis agar penyebab kematian dengan kasus yang sama dapat berkurang.

Salah satu cara untuk mengetahui penyebab kematian adalah autopsi. Menurut Kamus Besar Bahasa Indonesia (2019) autopsi adalah pemeriksaan tubuh mayat dengan cara melakukan pembedahan untuk mengetahui penyebab terjadinya kematian, penyakit, dan sebagainya. Autopsi dilakukan untuk mengetahui penyebab kematian sesuai ketentuan yang diizinkan. Akan tetapi, autopsi memiliki beberapa kelemahan untuk diterapkan di Indonesia. Biaya autopsi pada tahun 2011 di unit pelayanan autopsi di Instalasi Kedokteran Forensik dan Medikolegal Rumah Sakit Umum Daerah Dr. Soetomo Surabaya sebesar Rp1.369.587,00 (Sulistyorini, *et al., 2011*) dan RSUD Mataram Kota menerapkan biaya sebesar Rp5.000.000,- pada tahun 2017. Biaya yang dikeluarkan tergolong tinggi bagi mayoritas masyarakat Indonesia. Selain itu, menurut fatwa Majelis Ulama Indonesia (MUI) nomor 6 tahun 2009 bahwa autopsi dapat dilakukan dengan syarat hal itu menjadi jalan keluar satu-satunya untuk mengetahui penyebab kematian dalam penyelidikan hukum maupun penelitian kedokteran. Jenazah yang dijadikan objek autopsi harus memiliki izin baik melalui wasiat, izin ahli waris maupun izin pemerintah yang ditetapkan sesuai perundang-undangan. Hal ini membuat sulitnya autopsi dilakukan mengingat persyaratan yang ada. Menurut Ketua Umum Perhimpunan Dokter Forensik Indonesia, Ade Firmansyah Sugiharto minimnya jumlah autopsi yang dilakukan untuk kepentingan penegakan hukum, disebabkan oleh pihak keluarga yang tidak memberi izin pembedahan (Sugiharto, 2017). Rumah Sakit Cipto Mangunkusumo dalam satu tahun hanya dapat mengautopsi sekitar 400 mayat

dari 4.000 atau hanya sekitar 10%. Hal ini sangat tidak efektif jika harus dilakukan di daerah yang belum memiliki fasilitas lengkap serta dokter yang mumpuni mengelola proses autopsi.

Salah satu solusi yang dapat diterapkan selain proses autopsi secara bedah forensik adalah autopsi verbal. Autopsi verbal adalah suatu metode yang digunakan dalam proses mengetahui penyebab kematian dengan cara melakukan wawancara dengan anggota keluarga maupun kerabat terdekat mengenai tanda dan gejala yang muncul sebelum seseorang meninggal. Proses autopsi verbal menggunakan kuisioner yang telah terstandar. Tujuan utama autopsi verbal adalah mengidentifikasi jumlah dan penyebab sebelum kematian. Mengingat pada lingkungan masyarakat kurangnya pencatatan angka kematian dan juga data penyebab kematian. Proses autopsi verbal dilakukan untuk melakukan pencatatan terhadap semua penyebab kematian, tidak hanya untuk penegakan hukum. Autopsi verbal yang terdapat di Indonesia memiliki landasan hukum berupa Peraturan Bersama Menteri Dalam Negeri dan Menteri Kesehatan Nomor 15 Tahun 2010, Nomor 162 /MENKES/PB/I/2010 pasal 6. Autopsi verbal dapat digunakan mengingat mahalnya biaya autopsi serta presepsi yang beredar di masyarakat akan stigma negatif melakukan autopsi.

Autopsi verbal digunakan untuk mendapatkan angka dan penyebab kematian (mortalitas) di Indonesia yang terus mengalami peningkatan setiap tahunnya. Kondisi mortalitas di Indonesia dapat digambarkan melalui angka kematian kasar atau *Crude Death Rate* (CDR). CDR mencerminkan banyaknya kematian yang terjadi pada tahun tertentu per 1000 penduduk pada pertengahan tahun. CDR memberikan informasi mengenai keadaan kematian suatu negara tanpa ada pertimbangan umur, jenis kelamin, dan kriteria-kriteria tertentu dan memperhatikan keadaan kematian secara keseluruhan. Tinggi dan rendahnya tingkat mortalitas pada suatu daerah tidak hanya mempengaruhi tingkat pertumbuhan penduduk, melainkan menjadi barometer tinggi dan rendahnya tingkat kesehatan masyarakat di daerah tersebut (Bagus, 2008).

Dinas Kesehatan Kabupaten Malang memiliki *pilot project* untuk mengembangkan pencatatan kematian melalui autopsi verbal. Pengembangan *pilot project* diawali dari dua daerah yaitu Gondanglegi dan Kepanjen. Dinas Kesehatan Kabupaten Malang sudah memulai mengumpulkan data catatan kematian yang ada dengan cara autopsi verbal. Dipilihnya autopsi verbal dikarenakan biaya yang dikeluarkan lebih sedikit dibandingkan autopsi bedah. Autopsi verbal juga dapat mengurangi stigma negatif yang selama ini beredar di masyarakat mengenai autopsi yang mengacu pada hal negatif. Proses yang sedang berjalan saat ini adalah tahap pengumpulan data yang dilakukan oleh kader pewawancara. Data yang ada di wilayah penelitian dikumpulkan secara manual melalui pengisian kuisioner *smarthealth*. Daftar pertanyaan yang ditanyakan merupakan pertanyaan yang berkaitan dengan data penyebab kematian pasien. Dinas Kesehatan Kabupaten Malang belum memiliki sistem untuk mengolah data yang terkumpul menjadi informasi yang dapat

dimanfaatkan pemangku kepentingan. Pengumpulan data secara manual dengan menggunakan kertas memiliki risiko berupa kehilangan data maupun kerusakan data hasil kematian yang telah dikumpulkan kader pewawancara. Pencarian data menjadi sulit dilakukan seiring bertambahnya data seiring berjalannya waktu.

Sistem informasi autopsi verbal diharapkan dapat membantu Dinas Kesehatan Kabupaten Malang untuk mengumpulkan dan mengolah data kematian. Pengumpulan, penyimpanan dan pengolahan data dengan sistem informasi dapat memberikan informasi kasus kematian yang ada di Indonesia kedepannya. Tanpa adanya sistem informasi yang terintegrasi, data yang ada tidak bisa memenuhi kebutuhan pemangku kepentingan. Sistem informasi autopsi verbal akan membantu Dinas Kesehatan Kabupaten Malang untuk mengumpulkan data melalui sistem, menyimpannya dalam basis data serta mengolah data menjadi informasi yang berguna. Informasi yang ada dapat digunakan baik pemerintah maupun masyarakat umum untuk mendapatkan informasi kesehatan yang sesuai dengan kasus yang terjadi. Hal ini dapat digunakan dalam pengambilan kebijakan yang tepat guna mengurangi jumlah kematian akibat penyebab tertentu.

1.2 Rumusan masalah

Berdasarkan pada latar belakang yang dijelaskan diatas, muncul beberapa pertanyaan rumusan masalah pada penelitian ini,

1. Bagaimana hasil analisis persyaratan sistem untuk mengembangkan sistem informasi autopsi verbal pada Dinas Kesehatan Kabupaten Malang?
2. Bagaimana rancangan sistem informasi autopsi verbal pada Dinas Kesehatan Kabupaten Malang?
3. Bagaimana hasil implementasi sistem informasi autopsi verbal pada Dinas Kesehatan Kabupaten Malang?
4. Bagaimana hasil pengujian sistem sehingga dapat digunakan untuk proses autopsi verbal?

1.3 Tujuan

Berikut merupakan beberapa tujuan dilakukannya penelitian ini,

1. Untuk menganalisis persyaratan sistem informasi autopsi verbal pada Dinas Kesehatan Kabupaten Malang.
2. Untuk merancang sistem informasi autopsi verbal pada Dinas Kesehatan Kabupaten Malang.
3. Untuk mengimplementasikan sistem informasi autopsi verbal pada Dinas Kesehatan Kabupaten Malang.
4. Untuk menguji sistem sehingga dapat digunakan dalam proses autopsi verbal

1.4 Manfaat

Hasil penelitian ini dalam jangka panjang dapat bermanfaat untuk membantu visualisasi pencatatan kematian beserta gejala yang ada sebelumnya. Aplikasi dapat melakukan input terhadap data yang dibutuhkan untuk kemudian diolah menjadi informasi yang berguna bagi instansi terkait. Informasi disajikan dalam bentuk dasbor yang dapat digunakan untuk menyusun kebijakan secara tepat dalam menangani gejala yang ada di masyarakat. Bagi peneliti dapat bermanfaat sebagai wadah dan sarana dalam meningkatkan keilmuan di bidang teknologi informasi dan ilmu komputer. Diharapkan juga, penelitian mengenai aplikasi autopsi verbal dapat menjadi referensi, rujukan maupun pengembangan yang lebih lanjut untuk penelitian berikutnya.

1.5 Batasan masalah

Batasan masalah dalam penelitian ini memiliki tujuan agar permasalahan yang diangkat dapat diselesaikan secara terfokus dan tidak melebar pada substansi lainnya. Adapun batasan masalah pada penelitian ini adalah :

1. Autopsi verbal dilakukan pada daerah Kepanjen, Kabupaten Malang.
2. Formulir autopsi verbal yang diolah berisi pertanyaan mengenai kematian almarhum/almarhumah.
3. Sistem Informasi yang dibuat merupakan sistem informasi yang berbasis web.

1.6 Sistematika pembahasan

Berikut adalah struktur maupun sistematika pembahasan skripsi pada penelitian ini,

BAB I Pendahuluan

Bab ini menjelaskan mengenai latar belakang dibuatnya sistem informasi autopsi verbal, rumusan masalah yang terdapat pada penelitian, tujuan dilakukannya penelitian mengenai autopsi verbal, manfaat yang didapatkan, batasan masalah yang diangkat dan sistematika penulisan laporan penelitian.

BAB II Landasan Kepustakaan

Bab ini menerangkan tinjauan pustaka, penelitian sebelumnya yang dijadikan daftar refrensi, teori yang berhubungan dan konsep dari literatur ilmiah yang sesuai. Pada bab ini terdapat beberapa penelitian sebelumnya yang memiliki kesamaan tema pengembangan sistem kesehatan maupun autopsi verbal sehingga dapat menjadi rujukan bagi peneliti.

BAB III Metodologi

Bab ini membahas tahapan dan langkah yang harus dilakukan peneliti dalam menyelesaikan rumusan masalah untuk mendapatkan jawaban penelitian. Bab ini juga membahas mengenai penggunaan metode yang sesuai dalam membangun sistem informasi autopsi verbal.

BAB IV Pemodelan Proses Bisnis dan Analisis Persyaratan

Bab ini menjelaskan pemodelan proses bisnis dan analisis persyaratan yang dibutuhkan untuk membuat sistem informasi. Pemodelan proses bisnis memberikan visualisasi alur proses bisnis yang berjalan pada proses autopsi verbal. Analisis persyaratan dilakukan dengan melakukan analisis permasalahan yang ada, identifikasi pemangku kepentingan, identifikasi pengguna, identifikasi fitur baik secara fungsional maupun non-fungsional serta pemodelan *use case* untuk menggambarkan sistem. Pembuatan prototipe dilakukan untuk memberikan gambaran sistem informasi kepada pemangku kepentingan terkait.

BAB V Perancangan Lanjut

Perancangan lanjut berisi mengenai rancangan sistem informasi autopsi verbal yang dibuat berdasarkan analisis persyaratan yang dilakukan sebelumnya. Perancangan dilakukan dengan melakukan pemodelan kelas, perancangan interaksi antar objek, perancangan basis data.

BAB VI Implementasi

Implementasi berisikan hasil implementasi sistem informasi autopsi verbal yang dibuat menggunakan kerangka kerja *CodeIgniter*. Pembuatan dilakukan berdasarkan perancangan sistem informasi yang dibuat sebelumnya sehingga terbentuk sistem informasi autopsi verbal.

BAB VII Pengujian

Bab ini menjelaskan dan menjabarkan hasil pengujian terhadap sistem informasi yang telah dibangun. Pengujian yang dilakukan menggunakan menggunakan metode pengujian validasi dan kompatibilitas.

BAB VIII PENUTUP

Bab ini berisi kesimpulan berdasarkan hasil penelitian beserta saran yang diperoleh untuk pengembangan sistem informasi kedepannya.

BAB 2 LANDASAN KEPUSTAKAAN

2.1 Penelitian Sebelumnya

Kajian pustaka dilakukan dengan mengkaji penelitian yang pernah dilakukan oleh peneliti lain sebelumnya. Topik penelitian yang diangkat sebelumnya merupakan topik yang memiliki kemiripan dengan topik pembahasan yang diangkat oleh peneliti.

Penelitian yang menjadi refrensi pada penelitian ini dilakukan oleh Nasir (2008) menjelaskan bahwa dalam melakukan evaluasi progroram KIA pemerintah kota Lamongan sangat sulit untuk bisa mengevaluasi apakah program tersebut telah sesuai dengan rencana atau tidak, dikarenakan keterbatasan dalam memperoleh data. Data yang dipakai masih dalam bentuk manual sehingga data tersebut sedikit diragukan dalam proses pencatatannya. Oleh karena itu peneliti ingin membantu pemerintah dalam melakukan pencatatan, agar data yang disimpan dapat digunakan dan memiliki tingkat keakurasian yang tinggi. Penelitian yang dilakukan oleh Nasir (2008) akan dijadikan refrensi dalam mengembangkan sistem informasi pada bidang kesehatan.

Penelitian lain yang menjadi referensi adalah penelitian yang telah dilakukan saudari Febriyanisa (2018) terkait proses pencatatan dan pelaporan pada PT. PLN unit Transmisi Jawa Bagian Timur dan Bali unit pemeliharaan proteksi, meter dan otomasi. Pembuatan sistem informasi pencatatan dilakukan agar proses yang biasa dilakukan secara manual dapat dibuat lebih efektif dan efisien dengan adanya sistem informasi. Metode dalam pengembangan sistem informasi yang dilakukan adalah metode *waterfall* yang dimulai dari tahap analisis kebutuhan, perancangan, implementasi dan pengujian. Hasil implementasi menghasilkan sebuah sistem informasi berbasis web. Sistem yang dibuat memungkinkan petugas melakukan kegiatan menambah data pengujian, mencetak hasil pengujian, verifikasi yang dilakukan oleh suvervisor serta pihak yang lain dapat memeriksa secara langsung hasil pengujian. Penelitian yang dilakukan Febriyanisa (2018) akan dijadikan refrensi dalam proses analisis persyaratan dan proses perancangan sistem informasi lanjutan yang akan dibuat.

Penelitian lain yang menjadi refrensi adalah penelitian yang dilakukan oleh Vuksanovic & Sudarevic (2011) membahas perbandingan antara pengembangan web yang dilakukan dari awal dan pengembangan web dengan menggunakan kerang kerja. Hasil penelitian menunjukkan bahwa pengembangan web menggunakan kerangka kerja akan memberi lebih banyak keuntungan yaitu pengembangan web bisa dilakakan lebih cepat karena tidak perlu membangun dari awal, telah terdapat standarisasi dan konsistensi dalam awal pengembangan sehingga dapat memprediksi hasil yang akan didapatkan, kode dapat digunakan kembali baik kode yang terdapat pada library, class, dan function. Kerangka kerja dapat digunakan untuk membantu pengembang tidak banyak melakukan tugas yang berhubungan dengan pemrograman secara umum. Namun, pengembangan

aplikasi yang lebih kecil biasanya dapat dilakukan tanpa menggunakan kerangka kerja. Penelitian yang telah dilakukan Vuksanovic & Sudarevic (2011) akan menjadi referensi penelitian ini dalam hal mengembangan sistem informasi menggunakan kerangka kerja yang tersedia.

Penelitian lain yang menjadi referensi adalah penelitian yang dilakukan oleh Sadabadi & Tabatabaei (2009) membahas mengenai siklus pengembangan perangkat lunak dengan menggunakan prototipe. Selain itu, terdapat langkah dan tahapan yang perlu dilakukan dalam melakukan proses *protoyping* serta elemen yang membantu terciptanya *prototyping* yang berhasil. Metode *Prototyping* memiliki dua pendekatan yaitu *throw-away prototyping* dan *evolutionary prototyping.* Pendekatan *prototyping* dengan berdasarkan *throw-away prototyping* dibuat untuk membantu klien yang belum memiliki kemampuan cukup dalam pembuatan sistem, mendefinisikan kebutuhan yang diinginkan. Pendekatan ini dapat diselesaikan secara cepat dengan alat sederhana setelah wawancara atau diskusi selesai dilakukan dengan klien. Setelah pengembang dan klien telah sepakat dalam definisi kebutuhan dan berhasil melakukan validasi kebutuhan, fungsi, dan prosedur dari sistem secara baik dan telah dicapai kesepakatan, prototipe yang ada dapat dibuang karena tujuan yang ingin digapai adalah mendefinisikan kebutuhan sistem secara visual. Pendekatan *Evolutionary prototyping* menekankan pada pembuatan prototipe akan dibuat setelah semua kebutuhan dan persyaratan yang diinginkan klien telah didefinisikan dengan baik. Prototipe yang dibuat harus bersifat terstruktur dan harus dikembangkan secara berkelanjutan hingga menjadi sistem final yang akan digunakan. Penelitian yang dilakukan Sadabadi & Tabatabaei (2009) akan dijadikan refrensi dalam pemilihan metode *prototyping* dan melakukan proses *prototyping* dalam melakukan penelitian.

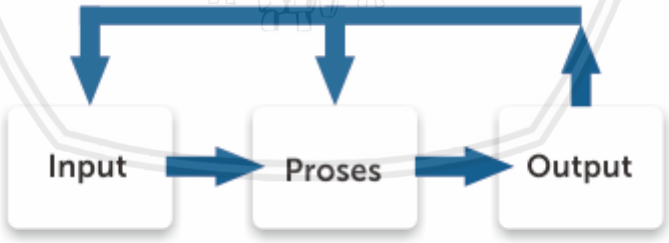
2.2 Autopsi Verbal

Autopsi Verbal merupakan penelusuran suatu rangkaian peristiwa atau keadaan atau gejala maupun tanda penyakit yang mengarah pada kematian melalui wawancara dengan pihak keluarga maupun pihak lain yang mengetahui kondisi almarhum/almarhumah sebelum meninggal (KEMENKES NOMOR 162/MENKES/PB/I, 2010). Menurut (Institute for Health Metrics and Evaluation, 2018) menjelaskan bahwa *Verbal Autopsy* (VA) adalah metode untuk menentukan penyebab kematian serta penyebab spesifik kematian yang tidak memiliki sistem registrasi vital yang lengkap. Autopsi verbal terdiri dari pewawancara terlatih yang menggunakan kuesioner untuk mengumpulkan informasi tentang tanda-tanda, gejala, dan karakteristik demografis dari orang yang baru meninggal. Pengumpulan data dilakukan dari seseorang yang akrab dengan orang yang meninggal sehingga dapat mengetahui data lebih lengkap. Instrumen Autopsi Verbal standar digunakan dan dipasangkan dengan metode analitik yang mudah diterapkan. Hal ini diharapkan dapat membantu menjembatani kesenjangan yang signifikan akan kurangnya informasi tentang penyebab kematian.

2.3 Sistem Informasi

Sistem informasi merupakan sekumpulan komponen yang memiliki fungsi untuk melakukan pengumpulan, penyimpanan, dan pemrosesan data yang disajikan dalam bentuk informasi pengetahuan dan dapat diambil manfaatnya serta produk digital (Zwass, 2017). Menurut Zwass (2017), perusahaan dan supplier perlu bersaing dengan menggunakan sistem informasi dalam kegiatan penting, kegiatan operasional harian dan kegiatan yang berhubungan dengan interaksi pada pelanggan mereka. Menurut Vojtech Democ (2015), sistem informasi adalah suatu perangkat yang menjadi inti penting perusahaan besar dalam mengelola perusahaan maupun organisasi didalamnya. Sistem informasi secara umum dapat digunakan untuk melakukan proses otamatisasi aktivitas rutin yang berjalan setiap harinya pada bidang administrasi dan manajemen. Menurut Stair & Reynolds (2014), sistem informasi adalah sekumpulan komponen yang saling berelasi satu sama lain. Komponen tersebut dapat melakukan pengumpulan data, manipulasi data, menyimpan data dan menyebarkannya pada komponen yang berelasi. Sistem informasi akan memberi informasi yang dibutuhkan oleh pemangku kepentingan serta menyediakan umpan balik guna mencapai tujuan yang ingin digapai.

Sistem informasi memiliki alur terstrukut yang harus dilewati. Alur yang dimiliki oleh sistem informasi harus dimulai dari *input* (masukan), melakukan proses, keluaran yang dihasilkan (*output*) dan umpan balik (*feedback*) pengguna. Setiap alur dalam sistem informasi akan menentukan kualitas informasi yang akan didapatkan. Proses masukan data sebisa mungkin berasal dari data yang baik dan benar sehingga dapat memberikan keluaran yang bermanfaat bagi organisasi. Alur sistem informasi seperti yang terlihat pada gambar 2.1.



Gambar 2.1 Skema Alur Terstrukur sebuah Sistem Informasi

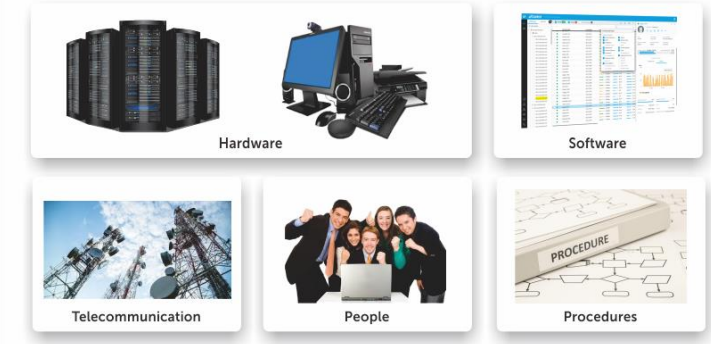
Alur yang terstrukur dalam sistem informasi memiliki tahapan yang perlu dilewati karena setiap tahapan didalamnya memiliki keterkaitan satu sama lain. Berikut merupakan uraian pada tiap tahap sistem informasi tersebut.

a. ***Input*** merupakan aktifitas yang digunakan dalam pengumpulan data mentah untuk selanjutnya diolah. Data mentah yang dikumpulkan ahrus data yang berkualitas dan memiliki nilai. Data yang baik akan menghasilkan keluaran informasi yang baik pula. Namun, jika data yang dikumpulkan buruk, maka hasil keluaran informasi nya juga akan bernilai

buruk. Oleh karena itu, proses input dapat menentukan kualitas *output* informasi.

b. **Proses** merupakan aktifitas yang digunakan dalam mengolah data mentah yang telah dikumpukan menjadi informasi yang bermanfaat. Proses juga bertugas menyalurkan informasi agar bisa dimanfaatkan. Data yang diolah harus memiliki nilai untuk disampaikan dan membantu organisasi menuju tujuan yang ingin digapai. Tahapan proses dapat berupa proses perhitungan data, membandingkan antar data yang berhasil didapatkan, pengambilan keputusan berdasarkan alternatif yang dapat diberikan hingga penyimpanan data yang dapat digunakan untuk keperluan masa mendatang.
c. ***Output*** merupakan aktifitas yang berfungsi menghasilkan informasi berharga. Informasi yang disajikan kepada pengguna dapat berupa laporan maupun dokumen. *Output* yang dikeluarkan juga dapat digunakan menjadi *input* bagi sistem lainnya. *Output* yang dikeluarkan juga harus bernilai dan bermanfaat agar tidak menjadi data yang buruk untuk sistem informasi berikutnya.
d. **Umpan balik** merupakan aktifitas yang berfungsi memberikan perubahan *input* atau aktivitas proses lainnya karena hasil *output* yang masih bisa ditingkatkan. Adanya umpan balik dapat membantu membuat data, proses maupun informasi ang salah menjadi benar dan dapat memiliki nilai lebih dibandingkan sebelumnya.
(Stair & Reynolds, 2014).

Sistem informasi dapat dilakukan maupun terjadi secara manual maupun memakai sistem yang terhubung secara sistematis dengan perangkat komputer. Sistem informasi yang menggunakan komputer dalam pengolahan data menjadi informasi disebut dengan *Computer-Based Information System* (CBIS) yaitu Sistem Informasi berbasis Komputerisasi. CBIS terdiri dari perangkat keras (*hardware),* perangkat lunak *(software)*, basis data (*databases),* telekomunikasi, organisasi, dan prosedur (SOP) seperti pada gambar 2.2 berikut.



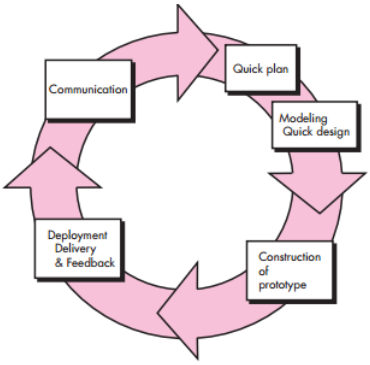
Gambar 2.2 Komponen yang Ada pada *Computer-Based Information System*

2.4 Rekayasa Perangkat Lunak (*Software Engineering*)

Perangkat lunak (*software*) merupakan program komputer beserta dokumentasi yang terdapat didalamnya. Rekayasa perangkat lunak (*Software Engineering*) adalah disiplin ilmu yang berfokus pada segala aspek yang berkaitan dengan produksi perangkat lunak. Rekayasa perangkat lunak dimulai dari tahap spesifikasi sistem hingga sistem selesai dibangun dan siap digunakan klien. *Software Engineering* merupakan bagian dari rekayasa sistem. Aspek yang terdapat dalam produksi perangkat lunak bukan hanya berfokus pada proses pengembangan perangkat lunak, namun termasuk aktivitas manajemen proyek dan *tools* dalam pengembangannya, metode yang digunakan dan teori yang mendukung pengembangan perangkat lunak (Sommerville, 2011).

2.5 *Prototyping*

Prototyping model merupakan sebuah model pengembangan perangkat lunak yang dimulai dengan pembuatan analisis persyratan dari pengguna dilanjutkan dengan perancangan prototipe dan evaluasi prototipe dari pihak pengguna. Proses pengembangan sistem menggunakan metode *prototyping* dapat dilihat pada Gambar 2.1 berikut.



Gambar 2.3 *Prototyping Model*

Sumber: Pressman (2010)

Gambar 2.3 merupakan siklus dalam proses pembuatan prototipe yang terjadi secara umum, lima tahapan yang dapat digunakan adalah:

1. *Communication*. Fase ini merupakan tahap komunikasi dengan pihak pengguna atau klien pertama kalinya. Hal ini dilakukan dengan melakukan wawancara untuk mendapatkan ide dari sistem yang akan dikembangkan kedepannya. Pada fase ini bertujuan untuk mendefinisikan persyaratan yang akan dibangun pada sistem.
2. *Quick Plan.* Fase ini memiliki fungsi untuk menggapai tujuan yang akan diambil berdasarkan tahap komunikasi sebelumnya. Langkah yang perlu dilakukan selanjutnya yaitu menjadwalkan pembuatan rancangan prototipe yang akan diajukan serta pembuatan jadwal laporan kepada para pemangku kepentingan untuk dilakukan evaluasi.
3. *Construction of prototype*. Fase ini bertujuan untuk membuat konstruksi rancangan atau prototipe *low fidelity* dari sistem yang akan dibangun nantinya. Pembuatan prototipe *low fidelity* akan dilanjutkan dalam tahap selanjutnya.
4. *Development delivery and feedback*. Prototipe diperkenalkan kepada klien atau pemangku kepentingan, kemudian dilakukan percobaan terhadap prototipe yang dikembangkan. Uji coba yang dilakukan bertujuan agar klien dapat memberikan pendapatnya terkait prototipe. Jika klien tidak setuju dan memiliki saran terhadap prototipe, maka akan dilakukan tahap 1,2 dan 3. Jika klien setuju dengan prototipe yang dikembangkan maka prototipe dapat dilanjutkan menjadi pembuatan sistem.

Metode prototyping memiliki dua pendekatan yang dapat dipilih sesuai kebutuhan yang diinginkan, yaitu *throw-away prototyping* dan *evolutionary prototyping*. T*hrow-away prototyping,* melakukan pengembangan sistem tanpa terlebih dahulu mengetahui spesifikasi yang akan dibangun pada awal pengembangan sehingga persyratan sistem masih bersifat abstrak dan belum diketahui sepenuhnya. *Evolutionary prototyping,* meakukan proses persyaratan spesifikasi sistem terlebih dahulu sehingga pada awal proses spesifikasi dan persyaratan sistem sudah tergambar dengan jelas. Hal ini dapat membantu sehingga mengurangi resiko persyaratan menjadi tidak terpenuhi (Pressman, 2010). Penelitian ini, menggunakan pendekatan *evolutionary prototyping* karena spesifikasi sistem dapat diketahui dari awal.

2.6 Proses Bisnis (*Business Process*)

Proses bisnis merupakan kumpulan aktivitas yang terdiri dari satu maupun lebih dari satu input sehingga dapat dimanfaatkan sebagai output yang memiliki nilai bagi pengguna. Kumpulan aktivitas yang ada dalam dalam proses bisnis dilakukan secara terkoordinasi untuk mewujudkan tujuan yang ingin digapai dalam bisnis. Proses bisnis dapat diberlakukan oleh internal dalam suatu organisasi, namun dapat juga digunakan untuk memungkinkan proses bisnis

dilakukan dengan kolaborasi bersama organisasi lain dan berinteraksi antar organisasi (Weske, 2007). Berikut ini adalah jenis aktivitas menurut Weske (2017) dalam proses bisnis:

1. *Manual Activity* (Aktivitas manual) merupakan kegiatan maupun aktivitas yang tidak didukung oleh sistem secara manual maupun komputer. Contoh dari aktifitas manual adalah mengirimkan dokumen kantor untuk ditunjukan kepada teman kantor melalui perantara kurir.
2. *User Interaction Activity* (Aktivitas interaksi pengguna) merupakan aktivitas yang dilakukan seseorang dengan tujuan agar pekerjaan menjadi lebih efektif melalui bantuan sistem informasi. Pekerjaan yang efektif memerlukan adanya antarmuka sistem yang baik dan sesuai dengan kebutuhan.
3. System Activity (Aktivitas system) merupakan aktivitas yang tidak melibatkan pengguna dalam menjalankan aksinya. Aktivitas ini dilakukan oleh sistem secara keseluruhan. Contoh dari aktivitas ini ialah menghitung jumlah total saldo rekening nasabah.

Penggambaran proses bisnis dapat dilakukan dengan menggunakan bantuan diagram proses bisnis. Diagram proses bisnis dapat mempresentasikan dan memberikan gambaran aktivitas dan pekerjaan. Diagram proses bisnis merepresentasikan aktivitas yang bisa dilakukan dari suatu proses bisnis dan model hubungan keduanya. Diagram proses bisnis memiliki notasi grafik terstandar yang disebut UML atau bisa disebut Diagram Aktivitas dan Business Process Model and Notation (BPMN) (Weske, 2007).

2.7 *Business Process Model and Notation* (BPMN)

Permodelan proses bisnis digunakan untuk menginformasikan proses bisnis kepada khalayak umum agar dapat dimengerti dengan satu standar yang sama . BPMN merupakan suatu aplikasi yang dirancang untuk melakukan berbagai jenis pemodelan untuk menginformasikan proses bisnis secara *end-to-end.* BPMN dapat memberikan notasi sederhana yang mudah dimengerti oleh semua pengguna yang terlibat. Analisis bisnis yang diberikan oleh pengguna dapat memulai perancangan dari awal proses hingga tahap teknis sehingga bertanggung jawab terhadap implementasi teknologi melalui penggambaran BPMN. BPMN akan mengelola dan memantau suatu proses bisnis yang terlibat pada pihak tertentu. BPMN diharapkan dapat mengurangi ketidaksesuaian antara desain dan proses bisnis dan juga proses implementasi yang akan dilakukan. Standarisasi juga dapat dilakukan oleh BPMN dengan cara melakukan notasi pemodelan dengan sudut pandang yang berbeda dari setiap pihaknya. Adanya BPMN menjadi sarana dalam menginformasikan proses bisnis kepada pengguna bisnis lain, pelaksana proses, pelanggan dan pemasok (Object Management Group (OMG), 2011).

BPMN mempunyai elemen kunci yang dimana elemen tersebut merupakan pilihan dari beberapa bentuk ikon yang digunakan dalam elemen grafis. Elemen

grafis tersebut diidentifikasikan sesuai kebutuhan. Tujuan yang ingin dicapai adalah membuat suatu bahas visual yang terstandar agar semua proses bisa dimodelkan sehingga mudah dikenali dan dipahami. Ada empat kategori elemen BPMN yang dapat menunjang dalam membuat pemodelan bisnis, yaitu *flow objects, data, connecting objects, swimlanes,* dan *artifacts* (Object Management Group (OMG), 2011).

2.7.1 Flow Objects

Flow objects merupakan elemen pokok yang dapat menunjukan perilaku mulai dari awal hingga akhir proses bisnis. Elemen *flow objects* mempunyai tiga elemen yang dapat digunakan oleh pengguna diantaranya, yaitu:

1. *Events*

   Event merupakan notasi yang dapat menunjukkan suatu kejadian selama proses pemodelan berlangsung. Selama proses pemodelan dapat dipengaruhi *event* yang dapat menimbulkan dampak kedepannya. Secara umum *event* memerlukan reaksi ataupun juga dapat memunculkan reaksi. Didalam *Event* terdapat tiga tipe utama yaitu *start events, end events,* dan *intermediate events* (Object Management Group (OMG), 2011). Berikut merupakan penjelasan mengenai tiap tipe notasi pada *eevent* yang ditunjukkan pada Tabel 2.1.

Tabel 2.1 Notasi Event

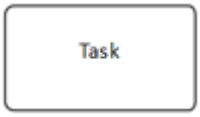
Notasi	Nama	Deskripsi
	Start events	Proses dimulainya event.
	Intermediate events	Suatu kendala yang terjadi pada saat awal atau akhir pemodelan yang dapat mempengaruhi proses bisnis.
	End events	Proses berakhirnya *event*.

Sumber: Object Management Group (2011)

2. *Activity*

   Activity merupakan notasi yang menunjukkan aktivitas yang terjadi saat proses bisnis dilakukan. Setiap *activity* mewakili poin dalam setiap pkerjaan yang dilakukan. *Activity* merupakan elemen yang dapat dikerjakan dari proses BPMN (Object Management Group (OMG), 2011) . *Activity* memiliki dua bagian yakni *task* dan *sub-process*. Penjelasan mengenai notasi activity ditunjukkan pada tabel 2.2.

Tabel 2.2 Notasi *Activity*

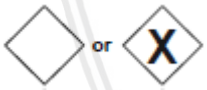
Notasi	Nama	Deskripsi
	*Task*	Menggambarkan suatu aktivitas yang terjadi dalam proses bisnis, akitivitas ini tidak dapat dipecahkan ketingkat yang lebih kecil.
	Sub-process	Menggambarkan suatu aktivitas yang terjadi dalam proses bisnis, dimana aktivitas tersebut dapat dijelaskan lebih rinci pada satuan proses.

Sumber: Object Management Group (2011)

3. *Gateways*

Gateways adalah salah satu notasi yang dapat mengendalikan alur proses bisnis. Apabila alur tersebut tidak dipelukan adanya kontrol maupun proses yang perlu dikendalikan, maka *gateways* dapat dihilangkan (Object Management Group (OMG), 2011). Berikut penjelasan mengenai notasi *activity* yang ditujukkan pada tabel 2.3.

Tabel 2.3 Notasi *Gateway*

Notasi	Nama	Deskripsi
	*Exclusive*	Proses divergen berfungsi untuk membuat satu atau lebih jalur alternatif dalam aliran proses bisnis. Proses dovergen juga berfungsi sebagai titik pengalihan jalan untuk suatu proses. Sedangkan untuk proses konvergen digunakan untuk menggabungkan jalur alternatif.
	Inclusive	Proses divergen digunakan untuk membuat jalur alternatif dan juga jalur parallel. Proses konvergen digunakan untuk menggabungkan kombinasi antara jalur alternatif dan juga jalur paralel.
	Parallel	Digunakan untuk menggabungkan dan menciptakan aliran paralel. Pada aliran *parallel* menunjukkan

Notasi	Nama	Deskripsi
		proses yang dapat dijalankan bersama-sama.
	Complex	Digunakan untuk memodelkan alur yang kompleks
	Event-Based	Menggambarkan terjadinya Titik percabangan dalam proses dimana jalur alternatif mengikuti *gateway* yang didasarkan pada *event* yang terjadi.

Sumber: Object Management Group (2011)

2.7.2 Connecting Objects

Dalam menggabungkan *flow objects* satu dengan lainnya memiliki empat cara, yaitu dengan *sequence flows, message flows, associations* dan *data associations* (Object Management Group (OMG), 2011). Penjelasan tersebut dapat ditujukkan pada tabel 2.4.

Tabel 2.4 Notasi *Connecting Objects*

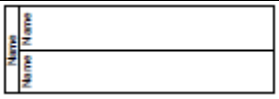
Notasi	Nama	Deskripsi
	Sequence flow	Menggambarkan aliran kontrol dalam proses bisnis yang dapat menunjukkan urutan aktivitas yang terjadi dalam proses dan koreografi.
	Message flow	Menggambarkan aliran *message* antara pengirim dan penerima di sebuah aktivitas.
	Association	Menggambarkan penghubung anatara informasi dan artefak dengan elemen grafis BPMN.
	Data association	menggambarkan hubungan aktivitas BPMN dengan notasi data.

Sumber: Object Management Group (2011)

2.7.3 Swimlanes

Swimlanes digunakan untuk mempartisi satu set aktivitas dari aktivitas lain. Terdapat dua tipe *swimlanes* yang berbeda pada BPMN, yaitu *pool* dan *lane* (Object Management Group (OMG), 2011). Penjelasan mengenai *swimlanes* dapat ditujukkan pada tabel 2.5.

Tabel 2.5 Notasi *Swimlanes*

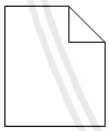
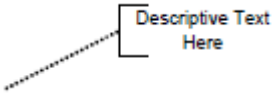
Notasi	Nama	Deskripsi
	Pool	Digunakan untuk mempartisi proses dari *pool* atau peserta lainnya, dimana *pool* berisi proses dan *sequence flow* yang menghubungkan aktivitas.
	*Lane*	Digunakan untuk mempartisi, mengatur dan mengkategorikan aktivitas dalam *pool.*

Sumber: Object Management Group (2011)

2.7.4 Artifacts

Artifacts merupakan suatu informasi pendukung yang terdapat dalam proses. Namun artifacts tidak secara langsung mempengaruhi aliran proses (Object Management Group (OMG), 2011). Penjelasan mengenai hal tersebut dapat ditujukkan pada tabel 2.6.

Tabel 2.6 Notasi *Artefacts*

Notasi	Nama	Deskripsi
	*Data Object*	Menggambarkan informasi aktivitas yang dilakukan. Data *object* mewakili *object* tunggal maupun kumpulan *object.*
	Data Store	Menggambarkan lokasi penyimpanan data, pada proses ini dapat membaca atau menulis data.
	*Text Annotation*	Merupakan *modeler* yang menyediakan informasi teks tambahan bagi pembaca diagram BPMN.
	Group	Pengelompokkan yang berada dalam kategori yang sama pada proses bisnis. Pengelompokkan tidak dipengaruhi oleh urutan alur dalam *group.*

Sumber: Object Management Group (2011)

2.8 Pemodelan *Use Case*

Pemodelan *use case* dapat menggambarkan suatu aktor dalam menggunakan sistem guna melakukan sesuatu yang dianggap penting. Penggunaan *use case* dapat membantu memahami nilai yang harus disediakan sistem bagi pemangku kepentingan. Fungsi pemodelan use case adalah untuk mencapai tujuan dari pembuatan sistem, pemodel harus berfokus pada siapa pengguna dan apa yang harus dilakukan sistem agar pengguna dapat melakukan sesuatu yang berguna (Bittner & Spence, 2002). Untuk mendapatkan pemodelan *use case* yang baik dan benar maka diperlukan beberapa hal diantaraya adalah identifikasi pemangku kepentingan, analisis masalah, idetifikasi kebutuhan pengguna, identifikasi fitur, persyaratan fungsional dan non-fungsional.

2.8.1 Identifikasi Tipe Pemangku Kepentingan

Langkah pertama sebelum membuat pemodelan adalah memahami pemangku kepentingan. Pemangku kepentingan merupakan seorang individu yang secara material dipengaruhi oleh hasil dari sistem atau proyek yang menghasilkan sistem. Apabila pengguna tidak terpengaruhi secara material oleh hasil dari sistem, mereka tidak akan menggunakan sistem dan sistem akan gagal. Untuk mengembangkan *use case* yang efektif pemahaman pemangku kepentingan sangat dibutuhkan. Salah satu cara yang paling efektif adalah dengan melibatkan langsung perwakilan dari pemangku kepentingan dalam mengembangkan dan meninjau *use case* itu sendiri (Bittner & Spence, 2002).

Hal yang paling utama untuk memahami jenis pemangku kepentingan adalah mengidentifikasi jenis dari pemangku kepentingan. Yang dmaksud dengan tipe pemangku kepentingan adalah klasifikasi sekelompok pemangku kepentingan dengan cara berbagi karakteristik yang sama dan memiliki hubungan dengan sistem. Tidak terbatasnya tipe pemangku kepentingan, bisa saja dikembangkan mengikuti kebutuhan dari sistem (Bittner & Spence, 2002). Beberapa kategori yang masuk dalam tipe pemangku kepentingan sebagai berikut:

1. Pengguna, adalah tipe pemangku kepentingan yang akan menggambil peran dengan ditentukan oleh para aktor dalam model *use case.*
2. Sponsor, merupakan tipe pemangku kepentingan yang berinvestasi saat memproduksi sistem tersebut. Yang termasuk kedalam tipe pemangku kepentingan ini diantaranya adalah para manajer bisnis, pemodal, pemegang saham, kepala departemen, penjual, pemasar, anggota komite pengarah, dll.
3. Pengembang, merupakan tipe pemangku kepentingan yang terlibat dalam produksi serta dukungan sistem. Yang termasuk kedalam tipe pemangku kepentingan ini diantaranya adalah manajer proyek, pengelola sistem penguji, staf pendukung, perancang, pembuat kode, penulis teknis, staf produksi, dll.
4. Pihak berwenang, merupakan tipe pemangku kepentingan yang ahli dalam aspek tertentu dari suatu masalah atau solusi domain. Yang termasuk kedalam

tipe pemangku kepentingan ini diantaranya adalah otoritas legislatif, standar organisasi, pakar teknologi, dll.

5. Pelanggan, merupakan tipe pemangku kepentingan yang sebenarnya akan membeli sistem di akhir. Yang termasuk kedalam tipe pemangku kepentingan ini diantaranya adalah pembeli, evaluator, angkutan dan agen yang bertindak atas nama organisasi pembelian.

Pada tabel 2.7 akan dijelaskan bagaimana cara dalam melakukan analisis analisis tipe pemangku kepentingan.

Tabel 2.7 Analisis Tipe Pemangku Kepentingan

Tipe Pemangku Kepentingan	Deskripsi	Pemangku Kepentingan
Menunjukkan tipe pemangku kepentingan	Menjelaskan tipe pemangku kepentingan	Menyebutkan siapa saja pemangku kepentingan yang termasuk kedalam tipe pemangku kepentingan.

Sumber: Bittner & Spence (2002)

2.8.2 Analisis Masalah

Masalah akan terjadi pada saat terjadi perbedaan anatara hal yang dirasakan berbeda dengan hal yang diinginkan. Untuk memenuhi kebutuhan pelanggan, maka harus memahami masalah apa saja yang ingin diselesaikan atau dipenuhi. Dalam menyelesaikan suatu masalah, pengembang harus benar-benar fokus pada masalah sebenarnya dan fokus pada masalah yang benar-benar harus dipecahkan. Dalam memahami kebutuhan pengembang harus terlebih dahulu memahami masalah tersebut. Pernyataan masalah merupakan acar terbaik dalam menangkap suatu masalah. Pernyataan masalah merupakan hasil dari ringkasan solusi dan pemahaman bersama para pemangku kepenttingan yang harus dipecahkan yang terdiri dari penilaian dampak suatu masalah dan manfaat apa saja yang akan diperoleh dari solusi tersebut. Dengan adanya analisis masalah ini bertujuan untuk memastikan bahwa semua pemangku kepentingan yang terlibat menyetujui permasalahan apa saja yang harus diselesaikan (Bittner & Spence, 2002). Pada tabel 2.8 menunjukkan kerangka dalam mendokumentasikan analisis masalah.

Tabel 2.8 Kerangka Dokumentasi Pernyataan Masalah

Masalah	Mendeskripsikan masalah
Mempengaruhi	Tipe pemangku kepentingan yang dipengaruhi oleh sutau masalah.
Dampak	Dampak yang akan terjadi pada pemangku kepentingan yang akan mempengaruhi kegiatan bisnis serta organisasi dari pemangku kepentingan

	tersebut.
Solusi	Merupakan ringkasan dari semua pemangku kepentingan yang bermanfaat guna menyelesaikan suatu masalah.

Sumber: Bittner & Spence (2002)

2.8.3 Identifikasi Kebutuhan Pemangku Kepentingan dan Pengguna

Pemenuhan kebutuhan dari setiap pemangku kepentingan dapat membantu pengembang aplikasi dalam menyelesaikan suatu masalah. Menurut Leffingwell & Widrig (1999) menyatakan bahwa kebutuhan pemangku kepentingan merupakan refleksi dari masalah bisnis, pribadi atau operasional (atau peluang) yang harus diatasi untuk membenarkan pertimbangan, pembelian atau penggunaan sistem baru. Melakukan analisis masalah memungkinkan untuk memahami bagaimana dan sejauh mana aspek-aspek yang berbeda dari masalah tersebut, sehingga dapat mempengaruhi berbagai jenis pemangku kepentingan yang ada didalamnya. Latar belakang serta justifikasi dari setiap pemangku kepentingan juga sangat diperlukan dalam analisis masalah. Setiap deskripsi dari analisis masalah tersebut haruslah mencakup alasan yang menunjukkan secara jelas mengenai tipe pemangku kepentingan mana saja yang dapat terkena dampak dari solusi yang ditawarkan tersebut(Bittner & Spence, 2002).

2.8.4 Identifikasi Fitur

Fitur merupakan layanan atau kualitas dari sistem yang diperlukan guna memberikan manfaat kepada pengguna dan membantu memenuhi kebutuhan pemangku kepentingan serta penggunanya. Terdapat dua fitur dalam melakukan analisis masalah, yakni *functional* atau *nonfunctional.* Fitur juga digunakan untuk merangkum kemampuan dan kualitas dari suatu produk yang akan dibagun, oleh karena itulah fitur harus dapat diakses oleh semua anggota tim proyek dan semua pemangku kepentingan. Setiap fitur akan dimanifestasikan secara lebih detail dalam model *use-case*. Untuk membuat mekanisme yang kuat maka diperlukan kombinasai dari fitur dan *use case* guna mengelola *scope* dari sistem, kombinasi tersebut akan menjaga semua pemangku kepentingan yang terlibat serta dapat menginformasikan tentang kemajuan sistem dalam memenuhi kebutuhan spesifikasi yang lengkap serta bentuk yang mudah diakses dan dikelola (Bittner & Spence, 2002). Pada tabel 2.9 menunjukkan bagaimana cara mendokumentasikan beberapa fitur sistem.

Tabel 2.9 Fitur Sistem

Kode Fitur	Deskripsi
Kode dari suatu fitur	Merupakan deskripsi dari fitur yang ada dalam sistem

Sumber: Bittner & Spence (2002)

2.8.5 Identifikasi Persyaratan Fungsional dan Non-Fungsional

Seringkali kebutuhan perangkat lunak diklasifikasikan sebagai persyaratan fungsional atau non-fungsional. Persyaratan fungsional merupakan suatu fitur yang harus disediakan sistem, baik itu dari segi rekasi terhadap input tertentu, dan bagaimana sistem harus berperilaku dalam situasi tertentu. Persyaratan fungsional sistem menggambarkan apa yang harus dilakukan oleh sistem (Sommerville, 2007). Persyaratan non-fungsional merupakan kebutuhan yang terikat dengan layanan spesifik yang disampaikan oleh sistem kepada penggunanya. Beberapa sifat sistem yag akan muncul setelah mendefinisikan batasan pada implementasi sistem seperti *reliability,* waktu respon, keamanan, kompatibilitas peramban (Sommerville, 2007).

2.8.6 Pemodelan *Use Case*

Kumpulan dari semua *use case* adalah model *use case*. Model *use case* terdiri dari aktor dan hubungan antara *use case* dan aktor yang menggambarkan sistem tertentu. Ada beberapa komponen yang ada pada model *use case*, yaitu aktor dan *use case.* Aktor mewakili manusia, perangkat keras, perangkat lunak, sistem lain atau hal-hal yang berinteraksi dengan sistem. Aktor juga bisa disebut seseorang atau sesuatu yang berada diluar sistem dan berinteraksi dengan sistem (IBM, 2007). Ciri yang dimiliki aktor adalah memiliki nama dan deskripsi singkat dan mereka terkatit dengan *use case* untuk berinteraksi. Sedangkan ciri yang dimiliki oleh *use case* adalah merepresentasikan seperangkat tindakan yang harus dilakukan oleh sistem untuk para aktornya (IBM, 2007). *Use case* dimulai oleh aktor, dapat melibatkan lebih dari satu aktor, disediakan oleh sistem, bukan fungsi atau fitur, dan tidak dapat diuraikan. *Use case* memiliki nama dan deskripsi rinci yang pada dasarnya merupakan cerita bagaimana para aktor menggunakan sistem untuk mencapai tujuan tertentu dan apa yang sistem lakukan untuk memenuhi tujuan tersebut (Bittner & Spence, 2002).

Pada model *use case* ini tidak menyediakan penjelasan atau deskripsi yang lengkap terkait pemodelan sistem, maka dari itu harus didukung persyaratan lainnya. Dalam model *use case* terdapat spesifikasi *use case.* Spesifikasi *use case* menggambarkan detail dari setiap *use case* dan memuat penjelasan mengenai bagaimana aktor dan sistem bekerja sama (IBM, 2007). Tabel 2.10 menunjukkan format dokumentasi spesifikasi *use case.*

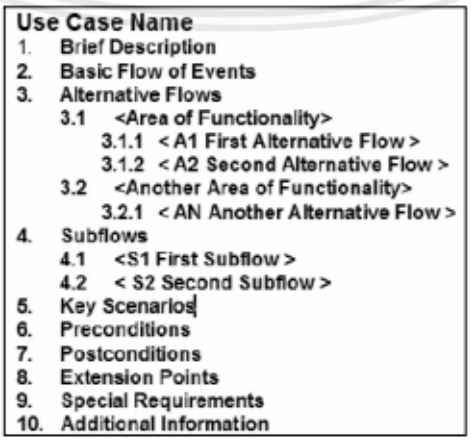
Tabel 2.10 Format Dokumentasi Spesifikasi *Use Case*

Name	Nama dari *use case.* Dimana setiap *use case* memiliki nama untuk menunjukkan hal apa saja yang akan dicapai oleh interaksi tersebut.
Brief Description	Berisi deskripsi singkat beserta tujuan dari *use case*.
Actor	Menggambarkan interaksi pengguna dengan *use case.*
Pre-Condition	Berisi deskripsi tektual yang mendefinisikan kondisi yang

	harus dipenuhi sebelum *use case* dijalankan.
Post-Condition	Deskripsi tekstual yang mendefinisikan bagaimana kondisi yang terjadi ketika *use case* berakhir.
Extension	Merupakan informasi penggunaan *use case* lain yang dikembangkan dan bisa bersifat opsional/pilihan yang diikutsertakan dalam *use case.*
Basic Flow	Alur yang dilalui ketika *use case* sedang berjalan dengan normal.
Alternative Flow	Alur opsional yang akan dilalui saat *use case* tidak berjalan normal.
Sub Flow	Berisi beberapa kelompok alur yang merupakan penyederhanaan alur normal *use case* yang terlalu terperinci.

Sumber: Diadapatasi dari (Bittner & Spence, 2002)

Use case merupakan teks terstruktur yang didalamnnya terdapat *use case style. Use case* yang disusun mengacu pada *use case style*. Terdapat banyak *style* atau gaya dalam menyusun *use case,* gaya yang dipilih harus membuat *use case* mudah dipahami oleh semua pemangku kepentingan. *Use case style* yang akan digunakan pada penelitian ini adalah *RUP style.* Contoh *use case* dengan *RUP style* dapat disesuaikan sesuai dengan kebutuhan sistem. Pada *RUP style* terdapat *basic flow* yang terdiri dari langkah bernomor dan bernama, pada RUP *styke* tidak mengarahkan ke *alternative flow* dan menunjukkan keadaaan sukses. Terdapat pula *alternative flow* yang memiliki nama beserta dengan langkah-langkahnya. Dalam *alternative flow* ini *use case* dimulai. Pada *flow* tersebut memuat apa yang menyebabkan *use case* dimulai, yang terjadi setelah mulai, dan menyatakan dimana *use case* berlanjut atau berakhir (IBM, 2007). Berikut pada Gambar 2.4 merupakan contoh dari Template *RUP Use Case Specification.*


Use Case Name
1. Brief Description
2. Basic Flow of Events
3. Alternative Flows
   3.1 <Area of Functionality>
      3.1.1 < A1 First Alternative Flow >
      3.1.2 < A2 Second Alternative Flow >
   3.2 <Another Area of Functionality>
      3.2.1 < AN Another Alternative Flow >
4. Subflows
   4.1 <S1 First Subflow >
   4.2 < S2 Second Subflow >
5. Key Scenarios
6. Preconditions
7. Postconditions
8. Extension Points
9. Special Requirements
10. Additional Information

Gambar 2.4 Contoh Template *RUP Use Case Specification*

Sumber : IBM (2007)

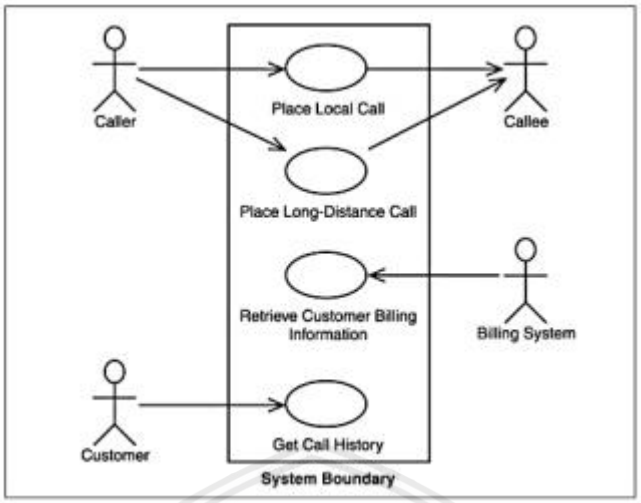
2.9 *Unified Modelling Language* (UML)

Model dapat digunakan untuk menyederhanakan realitas yang terjadi kemudian memberi gambaran lengkap suatu sistem dari suatu perspektif yang diangkat. Pemodelan dapat membantu pihak pengembang dalam memberikan visualisasi, menentukan, mendokumentasikan struktur serta perilaku dari suatu sistem. Bahasa standar yang dapat digunakan dalam pemodelan salah satunya adalah *Unified Modelling Language* (UML) (IBM, 2004).

Unified Modelling Language (UML) merupakan bahasa standar yang dapat digunakan pengembangan untuk membuat *blue print* dari perangkat lunak yang dikembangkan. UML atau bisa disebut bahasa pemodelan yang terpadu berisi notasi pengembangan dan pemodelan sistem berorientasi objek. UML dapat berguna untuk memberikan visualisasi, penentuan kasus, dan mendokumentasikan sistem perangkat lunak. UML membantu membangun perangkat lunak serta mempermudah pemahaman pengembang perangkat lunak kepada tim maupun orang lain yang terlibat. UML memiliki beberapa diagram yang dapat digunakan dalam proses membangun sistem. Pihak pengembang dapat memilih sesuai kebutuhan sistem yang dibangun dengan pemangku kepentingan (Pressman, 2010). Berikut beberapa diagram UML yang digunakan dalam mengembangkan sistem pada penelitian ini.

2.9.1 Use Case Diagram

Sebuah u*se case* menggambarkan interaksi yang terjadi antara pengguna dengan sistem yang digunakan. Hal ini dapat membantu menentukan langkah-langkah yang perlu dilakukan untuk menggapai tujuan. *Use case* memiliki variasi urutan langkah dalam penggambaran berbagai skenario. *Use case* merupakan template yang dapat menggambarkan kepada pengaembang fungsi sistem dari sudut pandang seorang pengguna (Pressman, 2010). *Use case* biasanya dapat mendefinisikan sesuatu yang harus maupun bisa dilakukan oleh sistem. *Use case* memiliko konsep pemodelan antara aktor, *use case* dan subjek. Aktor merupakan pengguna ataupun sistem lain yang berinteraksi pada sistem namun disimbolkan sebagai entitas yang berada diluar dari sistem. Subjek adalah sistem yang mempertimbangkan berlakunya *use case.* Hubungan antara *use case,* aktor dan sistem dapat divisualisasikan menggunakan diagram terstandar yaitu *use case diagram* (Object Management Group (OMG), 2005). *Use case diagram* merupakan pemodelan hubungan aktor dengan *use case* yang menjabarkan fungsi sistem (Pressman, 2010). Gambar 2.5 menunjukkan contoh *use case diagram.*



Gambar 2.5 Contoh *use case diagram*

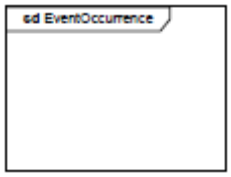
Sumber : Bittner & Spence (2002)

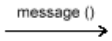
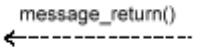
Paga Gambar 2.5 merupakan contoh *use case diagram* pada sistem telepon. Ketika memvisualisasikan interpretasi *use case diagram,* panah seharusnya bukan merupakan arus arah data. Setiap sistem hampir memiliki pertukaran informasi yang terjadi dua arah antara aktor dan *use case*. Beberapa kasus pada *use case diagram* ditampilkan tanpa memberikan arah panah, hal tersebut dapat diterima dalam standar pemodelan UML, panah dalam hubungan asosiasi berguna dalam menunjukkan inisator yang memulai komunikasi dan bisa memberikan urutan penggunaan *use case* (Bittner & Spence, 2002).

2.9.2 Sequence Diagram

Sequence diagram merupakan notasi yang digunakan untuk memberi gambaran interaksi antar objek dengan menggunakan pesan pada sekuensi *use case* yang ditampilan secara model grafis. Hal tersebut memberi urutan pertukaran penerimaan dan pengiriman pesan antar objek untuk menyelesaikan tugas. Diagram menunjukkan interaksi satu *use case* dalam satu skenario (Pressman, 2010). Keuntungan dalam menggunakan *sequence diagram* adalah mudahnya membaca penyampaian pesan pada model *sequence diagram* (Booch, et al., 2007). *Sequence diagram* memiliki notasi yang dijabarkan pada tabel 2.11.

Tabel 2.11 Notasi pada *Sequence Diagram*

Notasi	Nama	Deskripsi
	*Frame*	Memberikan gambaran mengenai serangkaian objek yang bertukar pesan antara satu dan lainnya sebagai bagian dari sistem.

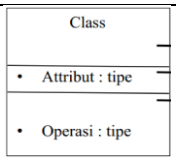
Notasi	Nama	Deskripsi
	Entity	Memberikan gambaran entitas dengan atribut yang dapat melakukan perekaman data.
	Boundary	Merupakan penghubung antara aktor dengan sistem
	Control	Mengontrol aktifitas maupun fungsi yang terjadi pada *event.*
	*Message*	Memberi gambaran pengiriman pesan.
	*Return Values*	Memberi gambaran pengembalian hasil pengiriman pesan .
	Lifeline	Memberi gambaran pada aktor dan sistem dan merupakan indikasi urutan kehidupan.
	Bar	Menunjukkan periode waktu yang berada didalam *lifeline* .

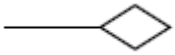
Sumber : Object Management Group (2005)

2.9.3 Class Diagram

Class diagram memiliki sifat statis dan dapat digunakan untuk melakukan pemodelan kelas, termasuk atributnya, operasi dan hubungan serta asosiasi mereka dengan kelas lain. *Class diagram* dapat menunjukkan hubungan antar kelas (Pressman, 2010). *Class diagram* berguna dalam menunjukkan keberadaan suatu kelas dan hubungan logisnya dari suatu sistem. Dalam kelas analisis, *class diagram* dapat berguna menunjukan peran dan juga tanggung jawab entitas secara umum. Dalam kelas desain, *class diagram* dapat berguna menangkap struktur kelas untuk kemudian terbentuk arsitektur sistem. Terdapat dua elemen yang penting terdapat dalam pemodelan *class diagram,* yaitu kelas dan hubungan dasarnya (Booch, et al., 2007) Dalam *class diagram* terdapat beberapa notasi yang ditunjukkan pada tabel 2.12.

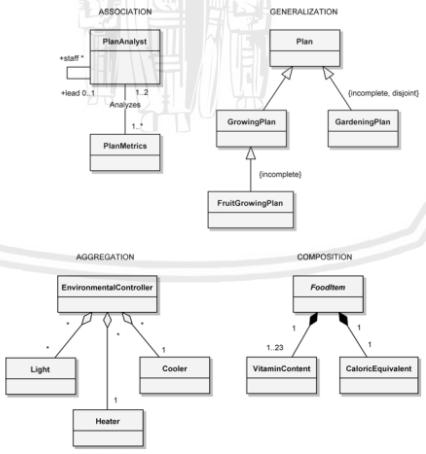
Tabel 2.12 Notasi pada Class *Diagram*

Notasi	Nama	Deskripsi
	*Class*	*Template* kelas yang memberikan penjelasan terkait objek yang dibuat dan membantu mendefinisikan isi kelas berupa nama kelas, atribut dan operasi yang terdapat pada kelas tersebut.

Notasi	Nama	Deskripsi
	Association	Merupakan gambaran yang menunjukan adanya hubungan antara *class.*
	Direct Association	Merupakan simbol yang menjelaskan bahwa satu *class* dapat digunakan oleh *class* yang lainnya.
	Generalization	Merupakan simbol pewarisan sifat antara kelas orang tua/*class* utama (*parent)* pada kelas anakny/ *class* anak (*child*).
	*Aggregation*	Merupakan simbol yang memberikan gambaran sebuah elemen dapat terdiri dari susunan komponen yang lebih kecil.
	Dependency	Merupakan simbol yang memberikan gambaran sebuah *class* memiliki ketergantungan dengan *class* yang lainnya

Sumber: Microsoft Developer Networks (2017)

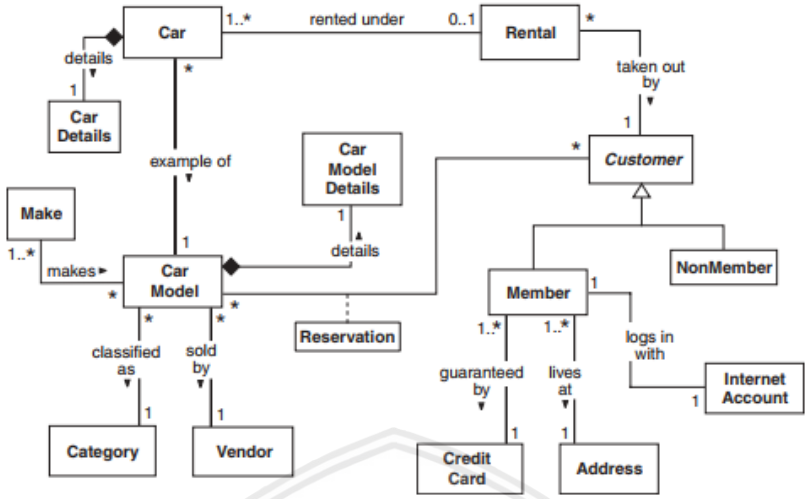
Sebuah kelas dalam *class diagram* memiliki kemungkinan untuk berdiri sendiri yang hampir mustahil. Hubungan antar kelas disimbolkan sebagai relasi pada *class diagram*. Hubungan antar kelas pada *class diagram* dapat dibagi menjadi generalisasi, asosiasi, agregasi dan depedensi yang dijabarkan pada Gambar 2.6 berikut.



Gambar 2.6 Contoh Hubungan Antar Kelas

Sumber: (Booch, et al., 2007)

Diagram kelas dapat memberikan petunjuk kelas yang terdapat pada sistem dan hubungan objek dari setiap kelas dapat berhubungan bersama. Diagram kelas yang memberikan analisis direpresentasikan sebagai kotak berlabel dan hubungan dengan kotak berlabel lainnya. Berikut pada Gambar 2.7 menjabarkan diagram kelas yang memberikan analisis.



Gambar 2.7 Contoh Diagram Kelas pada Tingkat Analisis

Sumber: Docherty (2005)

2.10 Pengujian

Pengujian merupakan tahap untuk menunjukkan program, aplikasi maupun sistem informasi yang dibuat dapat melakukan tugas yang ditujukan serta menemukan kesalahan, kekurangan bahkan cacat pada perangkat lunak sebelum digunakan. Pengujian dapat menunjukan ketika terdapat kesalahan atau kekurangan pada perangkat lunak, namun tidak dapat memastikan bersih dari kesalahan dan kekurangan. Pengujian merupakan salah satu fase penting dalam pengembangan sistem informasi. Hal ini karena pengujian dapat memastikan bahwa perangkat lunak telah memenuhi persyaratan pengguna baik secara fungsional maupun nonfungsional serta mendeteksi kekurangan sistem (Sommerville, 2007). Menurut Sommerville dalam proses pengujian memiliki dua tujuan berbeda, yaitu:

1. Menunjukkan kepada pihak pengembang dan pelanggan bahwa perangkat lunak, program maupun sistem informasi yang dikembangkan telah memenuhi persyaratan fungsional maupun persyaratan secara nonfungsional.
2. Menemukan kondisi saat perangkat lunak bertindak secara tidak benar, tidak sesuai spesifikasi dan tidak diinginkan. Hal ini merupakan konsekuensi dari adanya kekurangan maupun cacat pada perangkat lunak. Cacat pengujian muncul ketika terdapat perilaku perangkat lunak yang tidak seharusnya terjadi seperti sistem tidak berjalan, interaksi dengan sistem lain yang tidak diinginkan, salah perhitungan maupun data yang rusak.

2.10.1 Pengujian Validasi

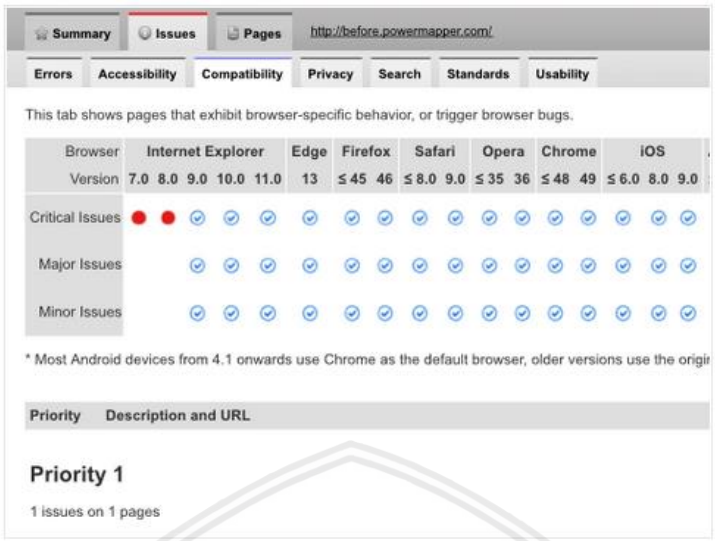
Pengujian validasi merupakan pengujian yang berfungsi menguji kesesuaian sistem yang dikembangkan telah sesuai dengan persyaratan pengguna. Pengujian validasi menggunakan persyaratan pengguna yang telah diidentifikasi pada tahap selanjutnya (Sommerville, 2007). Validasi dilakukan dengan serangkaian uji tes yang dapat menunjukan kesesuain dengan persyaratan perangkat lunak yang diinginkan. (Pressman, 2010). Pressman menjelaskan bahwa terdapat dua kondisi yang mungkin terjadi setelah fase pengujian, yaitu:

1. Fungsi perangkat lunak sesuai dengan persyaratan dan spesifikasi awal sehingga dapat diterima
2. Muncul penyimpangan persyaratan dan spesifikasi perangkat lunak sehingga akan menjadi catatan yang perlu diperbaiki. Penguji dapat mengatasi kesalahan penyimpangan yang ada dengan mengacu pada *use case* yang dibuat pada tahap analisis persyaratan.

2.10.2 Pengujian Kompatibilitas

Pengujian kompatibilitas peramban merupakan pengujian yang berfokus untuk menguji antarmuka pengguna pada aplikasi berbasis web atau *WebApp* pada lingkungan yang berbeda-beda. Pengujian kompatibilitas peramban dapat digunakan untuk menyingkap kesalahan maupun masalah yang dapat ditelusuri dengan cara melakukan serangkaian tes uji pada peramban yang berbeda (misalnya, *browser*). Pengujian ini dilakukan untuk memastikan perangkat lunak yang dikembangkan mampu bekerja persis ketika aplikasi dijalankan pada tampilan perangkat, sistem operasi, *browser*, perbedaan kecepatan koneksi serta *platform* yang berbeda. Hal yang perlu dilakukan saat menguji kompatibilitas peramban adalah menentukan serangkaian konfigurasi pengujian dan menguji perangkat lunak yang telah dikembangkan pada konfigurasi yang telah dibuat. Konfigurasi yang dapat dilakukan pada pengujian kompatibilitas diantaranyaperamban *browser* yang berbeda, tampilan perangkat yang berbeda, sistem operasi yang berbeda serta kecepatan jaringan internet yang berbeda (Pressman, 2010).

Salah satu pernagkat lunak pihak ketiga yang dapat membantu pengujian kompatibilitas adalah *Sortsite. Sortsite* merupakan dapat menguji pada komponen *tag* HTML, *script,* CSS, kesalahan halaman yang akan dipanggil, teknologi tidak didukung pada *browser*, dll (PowerMapper, 2018) Pada gambar 2.8 merupakan cotoh pengujian kompatibilitas yang dilakukan dengan perangkat lunak *Sortsite.*

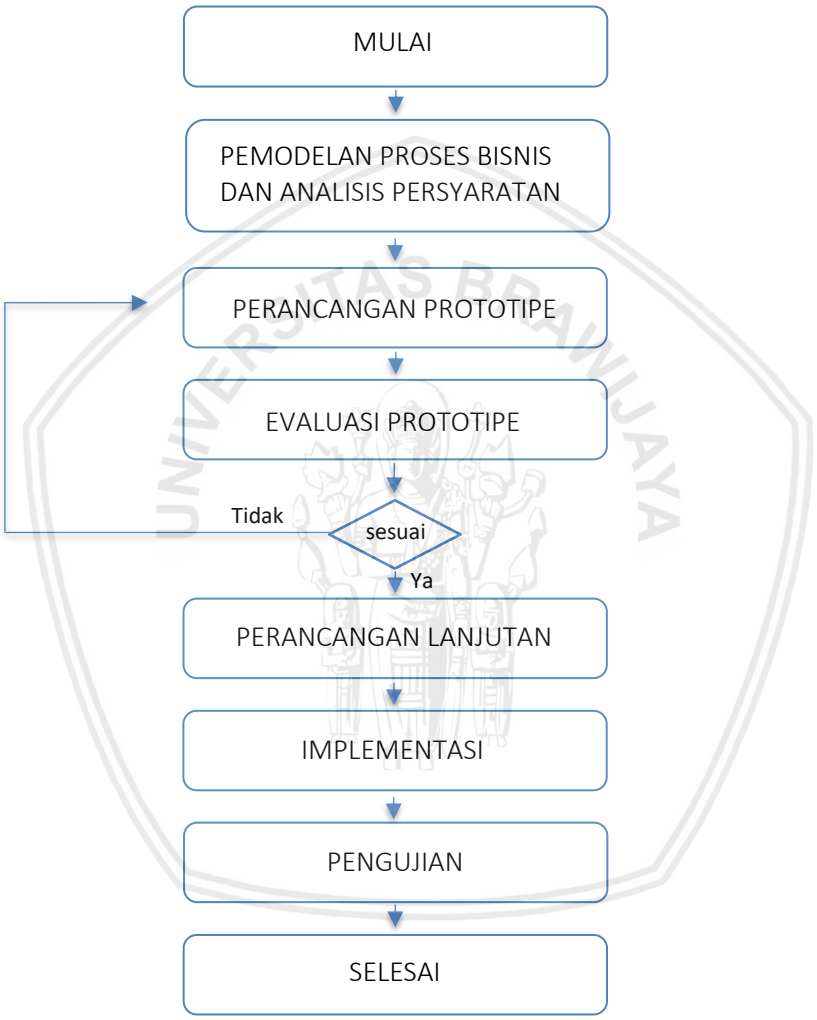


Gambar 2.8 Contoh hasil pengujian kompatibilitas menggunakan *Sortsite*

Sumber: PowerMapper (2018)

BAB 3 METODOLOGI

Penelitian yang dikerjakan akan menggunakan diagram alur metodologi sebagai gambaran kerangka kerja penelitian. Hal ini bertujuan agar penelitian yang dilakukan peneliti sesuai dengan metode yang diajukan, yaitu sebagai berikut :



Gambar 3.1 Diagram Alur Metode Penelitian

3.1 Pemodelan Proses Bisnis dan Analisis Persyaratan

Pemodelan proses bisnis dan analisis persyaratan dilakukan pada tahap-tahap menggali persyaratan bagi pemangku kepentingan yang terkait dengan sistem. Penggalian persyaratan dilakukan melalui wawancara serta observasi. Wawancara dilakukan terhadap Dinas Kesehatan Kabupaten Malang yang memiliki *pilot project* aplikasi autopsi verbal. Observasi dilakukan terhadap proses yang ada pada kondisi nyata dilapangan. Adanya observasi diharapkan dapat menjadikan improvisasi yang lebih baik kedepannya. Analisis persyaratan akan menghasilkan alur proses bisnis serta kebutuhan fungsional dan kebutuhan non-fungsional. Selain itu, terdapat *use case diagram* yang menggambarkan interaksi sistem dengan pengguna.

3.2 Perancangan Prototipe

Perancangan prototipe dilakukan dengan membuat rancangan antar muka pengguna dari sistem yang akan dibuat berdasarkan kebutuhan pemangku kepentingan. Prototipe dapat digunakan untuk menggambarkan rangkaian prosedur dan alur kerja sistem yang akan dibuat. Hal ini bertujuan untuk memberikan gambaran sistem secara utuh kepada pemangku kepentingan. Setelah perancangan prototipe selesai dilakukan, rancangan antar muka pengguna ditunjukan kepada pemangku kepentingan. Pemangku kepentingan dapat melihat dan memberikan umpan balik berupa saran maupun kritik terhadap prototipe.

3.3 Evaluasi Prototipe

Pada tahap evaluasi, para pemangku kepentingan dapat memberikan umpan balik berupa komentar, kritik maupun saran terhadap prototipe yang telah dibuat. Pemangku kepentingan dapat memastikan sistem telah memenuhi syarat yang diinginkan sebelumnya. Jika terdapat persyaratan yang belum terpenuhi, maka akan dilakukan iterasi pada tahap pengembangan prototipe sesuai kebutuhan yang terlewati. Saat prototipe telah memenuhi kebutuhan yang dibutuhkan pemangku kepentingan dan telah terdefinisi dengan baik sesuai yang dibutuhkan, maka pengembangan dapat dilakukan ke tahap selanjutnya.

3.4 Perancangan Lanjutan

Perancangan lanjutan dibuat menggunakan diagram yang telah terstandar untuk memberi penjelasan visual terhadap analisis persyaratan yang telah dilakukan sebelumnya. Terdapat beberapa diagram yang dapat dilakukan aitu *class diagram, sequence diagram,* dan *physical data model diagram (PDM)*. Seluruh diagram tersebut dibuat berdasarkan pada *use case diagram*, kebutuhan fungsional dan kebutuhan non fungsional yang telah dibuat pada tahap analisis persyaratan.

3.5 Implementasi

Tahap implementasi sistem merupakan tahap melakukan implementasi rancangan sistem menjadi bahasa pemrograman. Hasil implementasi berupa sistem informasi autopsi verbal berbasis web yang dapat digunakan pengguna. Pengembangan sistem menggunakan CodeIgniter sebagai kerangka kerjanya dengan bahasa pemrograman HTML, PHP dan Java Script serta basis data menggunakan MySql.

3.6 Pengujian

Implementasi yang telah selesai dilakukan oleh tim akan dievaluasi dalam fase pengujian. Pengujian dibuat untuk melakukan pengecekan terhadap kesesuaian fitur dengan dokumen yang telah dibuat pada tahap analisis persyaratan. Pengujian yang dilakukan dijelaskan pada sub bab berikut.

3.6.1 Pengujian Validasi (*Validation Testing)*

Pengujian validasi dilakukan dengan mencocokan antara analisis persyaratan dengan hasil implementasi sistem informasi. Pengujian yang dilakukan merupakan proses untuk mengetahui sistem yang telah dibuat memenuhi kebutuhan. Terdapat dua tahap yang dilakukan yaitu pengecekan hasil implementasi dibandingkan kesesuaiannya dengan daftar kebutuhan serta analisis hasil uji.

3.6.2 Pengujian Kompatibilitas (*Compatibility Testing*)

Pengujian kompatibilitas adalah pengujian yang membandingkan penggunaan antarmuka pengguna suatu ekosistem telah sesuai dengan yang diinginkan. Pengujian kompatibilitas dapat digunakan untuk menguji adanya masalah pada sistem pada saat digunakan pada peramban website yang berbeda. Pada pengujian ini, dilakukan menggunakan perangkat lunak Sortsite untuk membantu pengujian.

BAB 4 PEMODELAN PROSES BISNIS DAN ANALISIS PERSYARATAN

4.1 Pemodelan Proses Bisnis

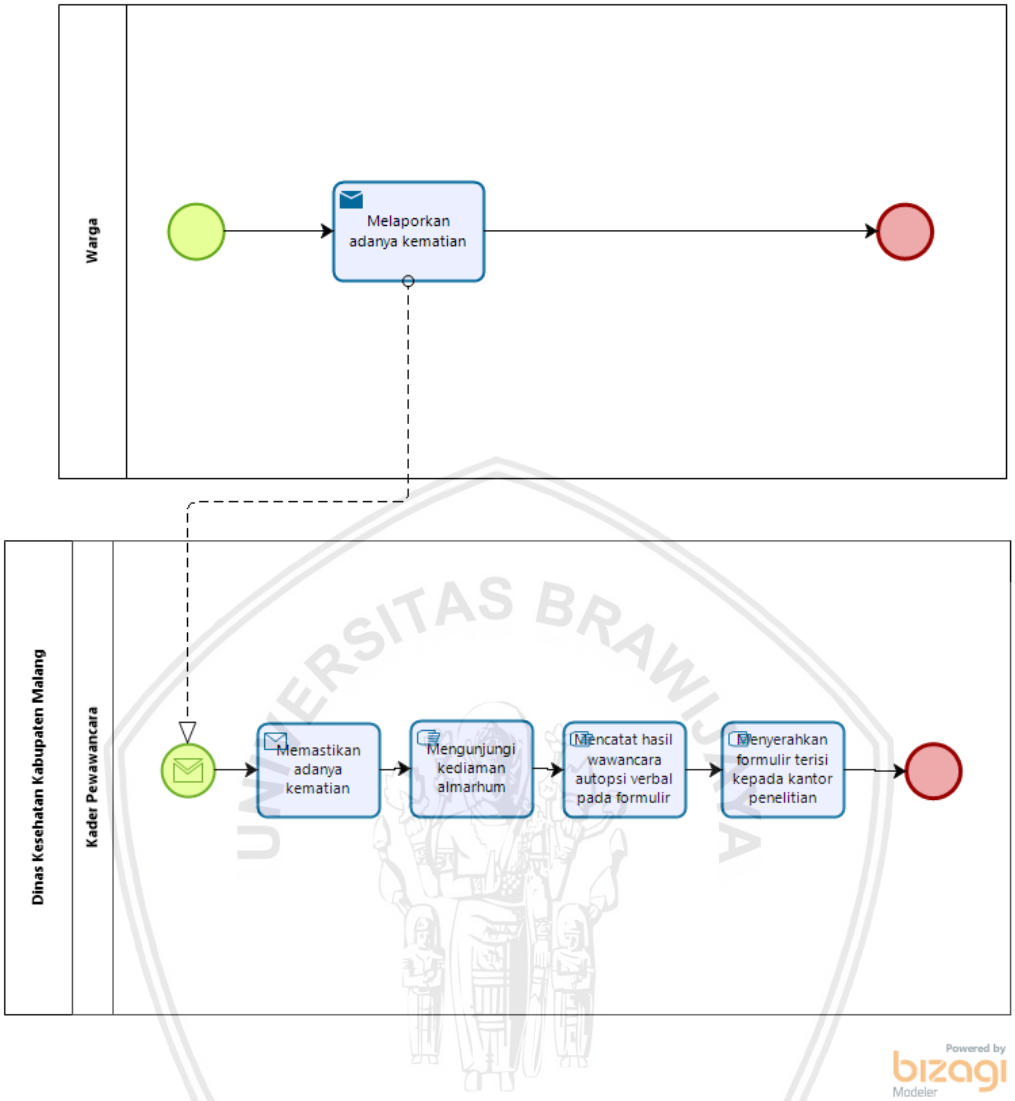
Pemodelan proses bisnis merupakan proses untuk menggambarkan aktivitas bisnis yang dilakukan pada proses autopsi verbal di wilayah Kepanjen, Kabupaten Malang. Informasi proses bisnis didapatkan melalui observasi dan juga wawancara. Observasi kegiatan autopsi verbal dilakukan di desa Dilem, salah satu desa di wilayah kecamatan Kepanjen. Wawancara dilakukan dengan Bapak Sujarwo sekaligus kepala penelitian dalam kegiatan autopsi verbal. Pemodelan proses bisnis pada penelitian dilakukan untuk memberikan pemodelan proses bisnis yang berjalan saat ini (*as-is*) dan pemodelan proses bisnis usulan (*to-be*) pada kegiatan autopsi verbal di kabupaten malang.

4.1.1 Analisis Proses Bisnis *As-Is*

Proses bisnis *as-is* merupakan proses yang sedang berjalan saat ini pada proses autopsi verbal dinas kesehatan kabupaten malang. Analisis proses bisnis *as-is* bertujuan untuk mengidentifikasi proses bisnis dan analisis masalah yang terjadi. Identifikasi proses autopsi verbal yang dilakukan terkait pengumpulan data dan juga visualisasi informasi yang dibutuhkan pemangku kepentingan. Proses bisnis digambarkan melalui model diagram BPMN untuk membantu standarisasi pemahaman analisis proses bisnis saat ini (*as-is*).

4.1.1.1 Proses Bisnis *As-Is* Pengumpulan Data hasil autopsi Verbal

Proses bisnis saat ini terkait dengan pengumpulan data hasil autopsi verbal yang dilakukan oleh kader desa atau kader pewawancara telah terlatih. Proses pengumpulan data kematian dimulai ketika kader pewawancara mengetahui informasi kematian di sekitarnya. Kader pewawancara akan mendatangi keluarga atau kerabat terdekat almarhum/almarhumah untuk melakukan autopsi verbal. Kader pewawancara membawa formulir "survei kematian smarthealth *extend* Indonesia Kabupaten Malang" dalam bentuk *print out.* Kader pewawancara akan melakukan wawancara terkait penyebab kematian berdasarkan poin yang terkandung pada formulir. Hasil wawancara kemudian disimpan untuk diserahkan pada Dinas Kesehatan Kabupaten Malang. Proses bisnis tersebut digambarkan dengan notasi BPMN pada Gambar 4.1.



Gambar 4.1 Proses Bisnis *As-Is* Pengumpulan Data hasil autopsi Verbal

4.1.1.2 Proses Bisnis *As-Is* Pemantauan Data Hasil Autopsi Verbal

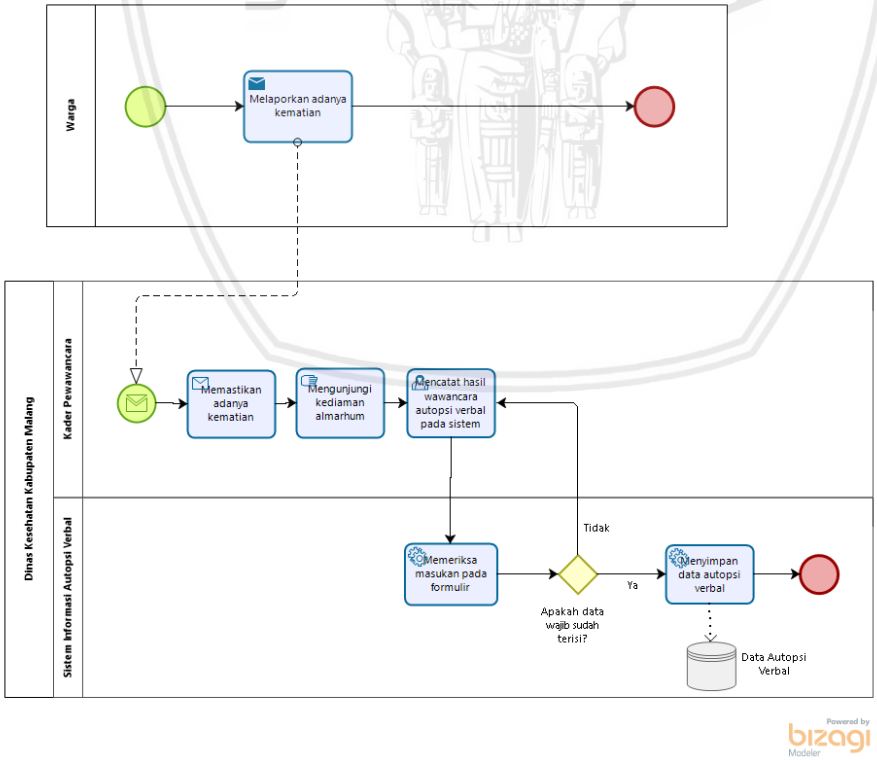
Proses bisnis saat ini terkait dengan pelaporan data hasil autopsi verbal. Data hasil autopsi verbal yang telah dilakukan kader pewawancara kemudian dikumpulkan pada kantor penelitian. Desa Dilem telah memiliki 300 data catatan kematian yang dikumpulkan kader pewawancara. Data tersebut berupa formulir yang sudah terisi dan siap diolah menjadi informasi yang dibutuhkan. Terdapat permasalahan yaitu belum tersedianya media untuk melaporkan pengumpulan data menjadi informasi yang dapat dilihat semua pihak terkait. Data yang dibutuhkan akan dicari satu persatu maupun sepengetahuan kader mengenai data terkait.

4.1.2 Analisis Proses Bisnis *To-Be*

Proses bisnis *to-be* merupakan proses bisnis yang usulannya ditawarkan pada penelitian ini terkait proses autopsi verbal pada dinas kesehatan kabupaten malang. Analisis proses bisnis *to-be* bertujuan untuk melakukan identifikasi terhadap proses bisnis yang dapat ditingkatkan maupun dieliminasi. Identifikasi proses autopsi verbal yang dilakukan terkait pengumpulan data dan juga visualisasi informasi yang dibutuhkan *pemangku kepentingan* terkait. Proses bisnis akan digambarkan dalam model diagram BPMN untuk membantu standarisasi pemahaman analisis proses bisnis yang akan diusulkan (*to-be*).

4.1.2.1 Proses Bisnis *To-Be* Pengumpulan Data hasil autopsi Verbal

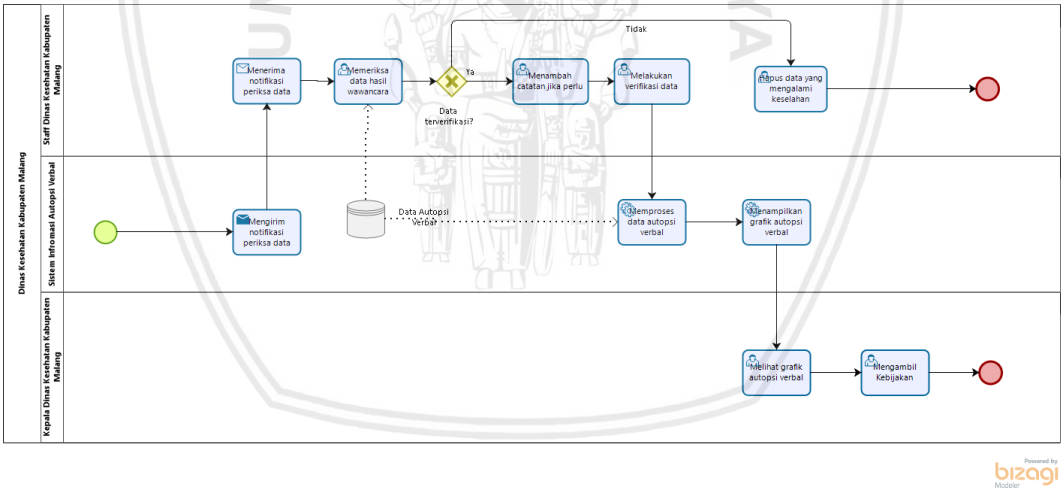
Proses bisnis usulan pada pengumpulan data hasil autopsi verbal melibatkan aktor terkait yaitu kader pewawancara dan sistem untuk melakukan pencatatan. Proses bisnis usulan dapat memanfaatkan sistem informasi untuk menyimpan data secara terpusat pada satu basis data. Kader pewawancara dapat memasukan hasil wawancara secara langsung ke dalam sistem tanpa perlu lagi membawa formulir *offline.* Data yang diinputkan akan dimasukan ke dalam satu basis data. Adanya basis data dapat membantu kader wawancara untuk lebih mudah mencatat serta sistem dapat menyimpan data tanpa tercecer ataupun terjadi kehilangan data. Proses bisnis tersebut digambarkan dengan notasi BPMN pada Gambar 4.2.



Gambar 4.2 Proses Bisnis *To-Be* Pengumpulan Data hasil autopsi Verbal

4.1.2.2 Proses Bisnis *To-Be* Pemantauan Data Hasil Autopsi Verbal

Proses bisnis usulan pada pelaporan data hasil autopsi verbal melibatkan aktor terkait yaitu kepala dinas kesehatan kabupaten malang sebagai pemangku kebijakan, staf dinas kesehatan kabupaten malang dan kader pewawancara. Pemantauan dimulai dengan mendapatkan data hasil autopsi verbal. Data yang ada dikumpulkan dalam satu basis data untuk mempermudah pengolahan data. Kader pewawancara dapat memasukan *form input* pada sistem agar dapat terekam pada basis data. Selanjutnya data akan disimpan dalam basis data yang dibuat. Staf dinas kesehatan kabupaten malang yang merupakan staf verifikator memiliki fungsi untuk melakukan verifikasi terhadap data yang diinputkan kader pewawancara. Data yang telah terverifikasi mengalami kematian bisa diverifikasi oleh staf verifikator dinas kesehatan kabupaten malang. Catatan khusus dapat ditambahkan oleh verifikator jika terdapat data yang dianggap tidak sesuai agar dapat diingat. Data yang telah di verifikasi oleh staf dinas kesehatan kabupaten malang telah disimpan pada status siap digunakan untuk kemudian diolah. Jika terdapat data yang tidak sesuai, staf verifikator dapat melakukan penghapusan data baik yang belum terverifikasi maupun sudah terverifikasi. Data yang telah tersimpan akan ditampilkan dalam bentuk visualisasi dasbor yang membantu *pemangku kepentingan* melihat informasi didalamnya. Proses bisnis ini digambarkan dengan notasi BPMN yang ditunjukan pada Gambar 4.3.



Gambar 4.3 Proses Bisnis *To-Be* Pemantauan Data Hasil Autopsi Verbal

4.2 Analisis Persyaratan

Analisis persyaratan menjelaskan analisis permasalahan yang dihadapi berdasarkan kondisi yang ada di lapangan beserta analisis persyaratan yang dibutuhkan untuk pengembangan sistem.

4.2.1 Analisis Permasalahan

Analisis permasalahan dilakukan dalam memahami permasalahan yang sedang dihadapi pemangku kepentingan beserta solusi atas permasalahan yang dihadapi. Data untuk membuat analisis permasalahan didapatkan dari observasi dan wawancara yang dilakukan dengan Bapak Sujarwo sebagai kepala penelitian untuk proses autopsi verbal. Observasi dilakukan pada desa Dilem, salah satu desa di wilayah kecamatan Kepanjen. Berikut hasil analisis masalah ditunjukan pada Tabel 4.1.

Tabel 4.1 Pernyataan Masalah

Masalah	● Pencatatan hasil autopsi verbal dilakukan dengan cara manual yaitu mengisi cetakan formulir autopsi verbal. ● Belum terdapat sistem untuk menampilkan data hasil autopsi verbal menjadi informasi bagi pemangku kepentingan.
Mempengaruhi	Kader pewawancara, staf dinas kesehatan kabupaten malang, kepala dinas kesehatan kabupaten malang.
Dampak	● Risiko tinggi terjadi kerusakan data atau kehilangan formulir hasil autopsi verbal. ● Data belum disimpan dalam satu pusat. ● Pihak pemangku kepentingan tidak bisa memeriksa dan melihat visualisasi data hasil wawancara.
Solusi	Mengembangkan sistem yang dapat membantu kader pewawancara melakukan pencatatan hasil wawancara secara langsung di tempat wawancara maupun di kantor penelitian dinas kesehatan, data hasil wawancara bisa terpusat pada satu tempat, dan pemangku kepentingan bisa dengan mudah memeriksa hasil wawancara secara keseluruhan.

4.2.2 Identifikasi Tipe Pemangku Kepentingan

Identifikasi tipe pemangku kepentingan (*stakeholder*) dilakukan dengan tujuan untuk mengklasifikasikan dan mengetahui pengguna yang sesuai serta memberikan hak akses yang bisa dilakukan dengan sistem. Sumber data didapatkan dari observasi dan wawancara yang dilakukan dengan Bapak Sujarwo sebagai kepela penelitian untuk proses autopsi verbal. Observasi dilakukan pada desa Dilem, salah satu desa di wilayah kecamatan Kepanjen. Tabel 4.2 menunjukan jenis pemangku kepentingan yang terdapat pada pada penelitian ini.

Tabel 4.2 Jenis Pemangku Kepentingan

Jenis Pemangku Kepentingan	Deskripsi	Pemangku Kepentingan
Pengguna	Orang yang terlibat dalam proses autopsi verbal dan akan menggunakan sistem secara langsung	- Kader Pewawancara - Staf Verifikator Dinas Kesehatan Kabupaten Malang - Staf Administrator Dinas Kesehatan Kabupaten Malang - Kepala Dinas Kesehatan Kabupaten Malang
Pengembang	Orang yang terkait dengan pengembangan sistem informasi	Peneliti
Pihak yang berwenang	Organisasi maupun seseorang yang terlibat dan turut memberikan andil dalam pengembangan sistem informasi autopsi verbal dengan cara membantu menjabarkan proses pada wawancara autopsi verbal sesuai aturan yang berlaku.	Dinas Kesehatan Kabupaten Malang

4.2.3 Identifikasi Pengguna

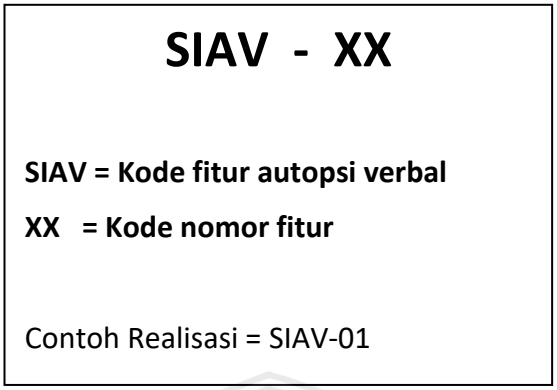
Identifikasi pengguna dilakukan untuk mengetahui aktor yang langsung berhubungan dengan sistem yang dibangun. Identifikasi pengguna dilakukan dengan cara mengolah data jenis pemangku kepentingan yang telah dijelaskan. Identifikasi pengguna merupakan tahap untuk mengidentifikasi aktor yang akan dimodelkan dalam diagram *use case.* Tabel 4.3 menjelaskan hasil identifikasi pengguna.

Tabel 4.3 Hasil Identifikasi Pengguna

Tipe Pemangku Kepentingan	Tipe Pengguna	Deskripsi
Pengguna	Kader Pewawancara	Seseorang yang telah terlatih untuk melakukan wawancara autopsi verbal dengan keluarga maupun kerabat almarhum/almarhumah.
	Staf Administrator Dinas Kesehatan Kabupaten Malang	Seseorang yang bekerja pada Dinas Kesehatan Kabupaten Malang yang memiliki autoritas untuk mengolah data pengguna dan memantau proses autopsi verbal di wilayah Kabupaten Malang.
	Staf Verifikator Dinas Kesehatan Kabupaten Malang	Seseorang yang bekerja pada Dinas Kesehatan Kabupaten Malang yang memiliki autoritas untuk melakukan verifikasi data hasil autopsi verbal dan memantau proses autopsi verbal di wilayah Kabupaten Malang.
	Kepala Dinas Kesehatan Kabupaten Malang	Seseorang yang memiliki kedudukan tertinggi pada Dinas Kesehatan Kabupaten Malang dan memiliki kewenangan untuk memberikan kebijakan berdasar hasil autopsi verbal yang telah dilakukan.

4.2.4 Identifikasi Fitur

Kodifikasi fitur dilakukan untuk memberikan gambaran besar penamaan kode fitur yang dilakukan dalam penelitian. Kode fitur ditunjukan dengan SIAV dan dilanjutkan dengan nomor fitur. Kodifikasi fitur dapat dilihat pada gambar 4.4



Gambar 4.4 Kodifikasi Fitur

Identifikasi fitur dilakukan untuk memberikan solusi yang ditawarkan sistem kepada pengguna. Pembuatan identifikasi fitur dilakukan sebagai informasi agar tidak terjadi ambiguitas dalam penulisan fitur secara fungsional maupun nonfungsional pada sistem informasi yang dikembangkan. Kode yang dibuat akan melekat sebagai identitas pengenal dari setiap fitur. Keterangan kode fitur secara lebih jelas dapat dilihat pada Gambar 4.4 dan dijabarkan secara detail pada Tabel 4.4.

Tabel 4.4 Hasil Identifikasi Fitur

Kode Fitur	Deskripsi
SIAV-01	Sistem dapat digunakan pada proses mencatat dan menyimpan hasil autopsi verbal yang berisi data catatan kematian almarhum/almarhumah.
SIAV-02	Sistem dapat digunakan pada proses mengelola data hasil autopsi verbal yang berisi data catatan kematian almarhum/almarhumah.
SIAV-03	Sistem dapat digunakan pada proses mengelola seluruh data pengguna dalam sistem.
SIAV-04	Sistem dapat menampilkan informasi hasil autopsi verbal untuk melihat statistik berdasarkan catatan kematian
SIAV-05	Sistem dapat digunakan untuk mengenali identitas pengguna dan memberikan hak akses sesuai dengan jabatan dan fungsi yang dimiliki.

.

4.2.5 Persyaratan Fungsional

Kode persyaratan fungsional dilakukan untuk memberikan gambaran besar penamaan kode fitur yang dilakukan dalam penelitian. Kode fitur ditunjukan dengan "SIAV" yang menandakan kode Sistem Informasi Autopsi Verbal, kode "F" yang menunjukan kodifikasi fungsional dan dilanjutkan dengan nomor fitur fungsional. Kodifikasi fitur fungsional dapat dilihat pada gambar 4.5

Gambar 4.5 Kodifikasi Persyaratan Fungsional

Persyaratan fungsional merupakan fitur yang dibutuhkan sistem dapat berjalan secara baik dan dapat memenuhi persyaratan pengguna. Persyaratan fungsional memiliki identitas kode untuk menamai persyaratan fungsional, keterangan kode persyaratan fungsional terdapat pada hasil identifikasi fitur Tabel 4.5.

Tabel 4.5 Persyaratan Fungsional

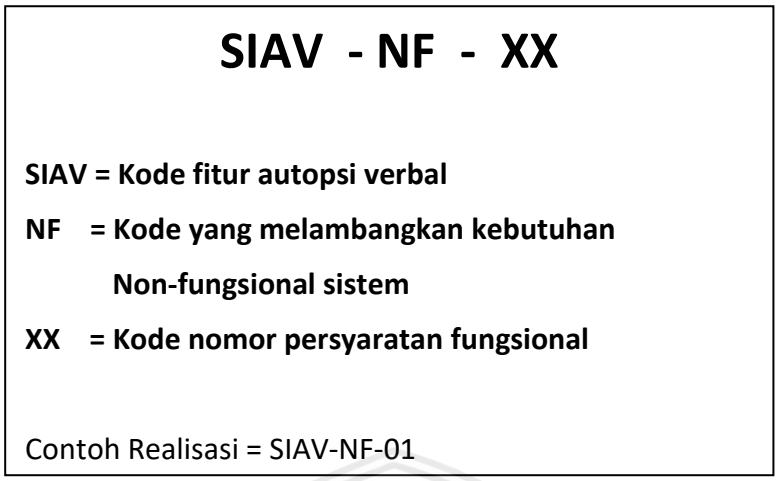
No.	Kode Fitur	Kode Persyaratan Fungsional	Deskripsi
1	SIAV-01	SIAV-F-01	Sistem harus menampilkan formulir wawancara autopsi verbal yang berisi daftar pertanyaan wawancara.
2		SIAV-F-02	Sistem harus dapat diisikan dengan data hasil autopsi verbal.
3		SIAV-F-03	Sistem harus dapat mencegah data yang penting belum terisi oleh kader pewawancara.
4		SIAV-F-04	Sistem harus mengirimkan data hasil autopsi verbal

No.	Kode Fitur	Kode Persyaratan Fungsional	Deskripsi
			kedalam basis data yang telah disediakan.
5		SIAV-F-05	Sistem harus menyimpan data hasil autopsi verbal pada basis data yang telah disediakan sistem.
6	SIAV-02	SIAV-F-06	Sistem harus dapat menampilkan ringkasan data hasil autopsi verbal yang belum terverifikasi.
7		SIAV-F-07	Sistem harus dapat menampilkan detail data hasil autopsi verbal yang belum terverifikasi.
8		SIAV-F-08	Sistem harus dapat melakukan verifikasi hasil autopsi verbal.
9		SIAV-F-09	Sistem dapat melampirkan catatan saat akan melakukan verifikasi
10		SIAV-F-10	Sistem harus dapat menghapus data hasil autopsi verbal yang belum terverifikasi.
11		SIAV-F-11	Sistem dapat melakukan sortir data hasil autopsi verbal yang belum terverifikasi.
12		SIAV-F-12	Sistem dapat melakukan pencarian data hasil autopsi verbal yang belum terverifikasi
13		SIAV-F-13	Sistem harus dapat menampilkan detail data hasil autopsi verbal yang terverifikasi.
14		SIAV-F-14	Sistem harus dapat menampilkan ringkasan data hasil autopsi verbal yang terverifikasi.
15		SIAV-F-15	Sistem harus dapat menghapus data hasil autopsi verbal yang terverifikasi.
16		SIAV-F-16	Sistem dapat melakukan sortir data hasil autopsi verbal yang terverifikasi
17		SIAV-F-17	Sistem dapat melakukan pencarian data hasil autopsi verbal yang terverifikasi
18		SIAV-F-18	Sistem dapat melakukan download data hasil autopsi verbal.
19	SIAV-03	SIAV-F-19	Sistem harus dapat menampilkan formulir penambahan data pengguna.
20		SIAV-F-20	Sistem harus menyimpan data pengguna pada basis data yang telah disediakan.

No.	Kode Fitur	Kode Persyaratan Fungsional	Deskripsi
21		SIAV-F-21	Sistem harus menampilkan ringkasan data pengguna dalam bentuk tabel.
22		SIAV-F-22	Sistem harus dapat mengubah data pengguna .
23		SIAV-F-23	Sistem harus dapat memperbarui data pengguna.
24		SIAV-F-24	Sistem harus dapat menampilkan detail data pengguna yang tercatat pada sistem.
25		SIAV-F-25	Sistem harus dapat menghapus data pengguna yang tercatat pada sistem.
26		SIAV-F-26	Sistem dapat melakukan sortir data pengguna
27		SIAV-F-27	Sistem dapat melakukan pencarian data pengguna
28	SIAV-04	SIAV-F-28	Sistem harus menampilkan informasi hasil autopsi verbal.
29		SIAV-F-29	Sistem harus menampilkan grafik hasil autopsi verbal.
30	SIAV-05	SIAV-F-30	Sistem harus menampilkan otentikasi pengguna.
31		SIAF-F-31	Sistem harus melakukan otentifikasi pengguna.
32		SIAF-F-32	Sistem harus mengganti *session* berdasarkan pengguna yang sedang aktif pada sistem.
33		SIAF-F-33	Sistem dapat memberhentikan *session* pengguna yang aktif dalam sistem.
34		SIAF-F-34	Sistem dapat menghentikan aktifitas pengguna yang telah logout.

4.2.6 Persyaratan Nonfungsional

Kode persyaratan nonfungsional dilakukan untuk memberikan gambaran besar penamaan kode fitur yang dilakukan dalam penelitian. Kode fitur ditunjukan dengan “SIAV” yang menandakan kode Sistem Informasi Autopsi Verbal, kode “NF” yang menunjukan kodifikasi fungsional dan dilanjutkan dengan nomor fitur fungsional. Kodifikasi fitur fungsional dapat dilihat pada gambar 4.6



Gambar 4.6 Kodifikasi Persyaratan Nonfungsional

Persyaratan nonfungsional merupakan kondisi yang harus dipenuhi agar sistem dapat berjalan dengan baik. Persyaratan nonfungsional memiliki identitas kode untuk menamai persyaratan nonfungsional, keterangan kode persyaratan nonfungsional terdapat pada hasil identifikasi fitur Tabel 4.6.

Tabel 4.6 Persyaratan Non-Fungsional

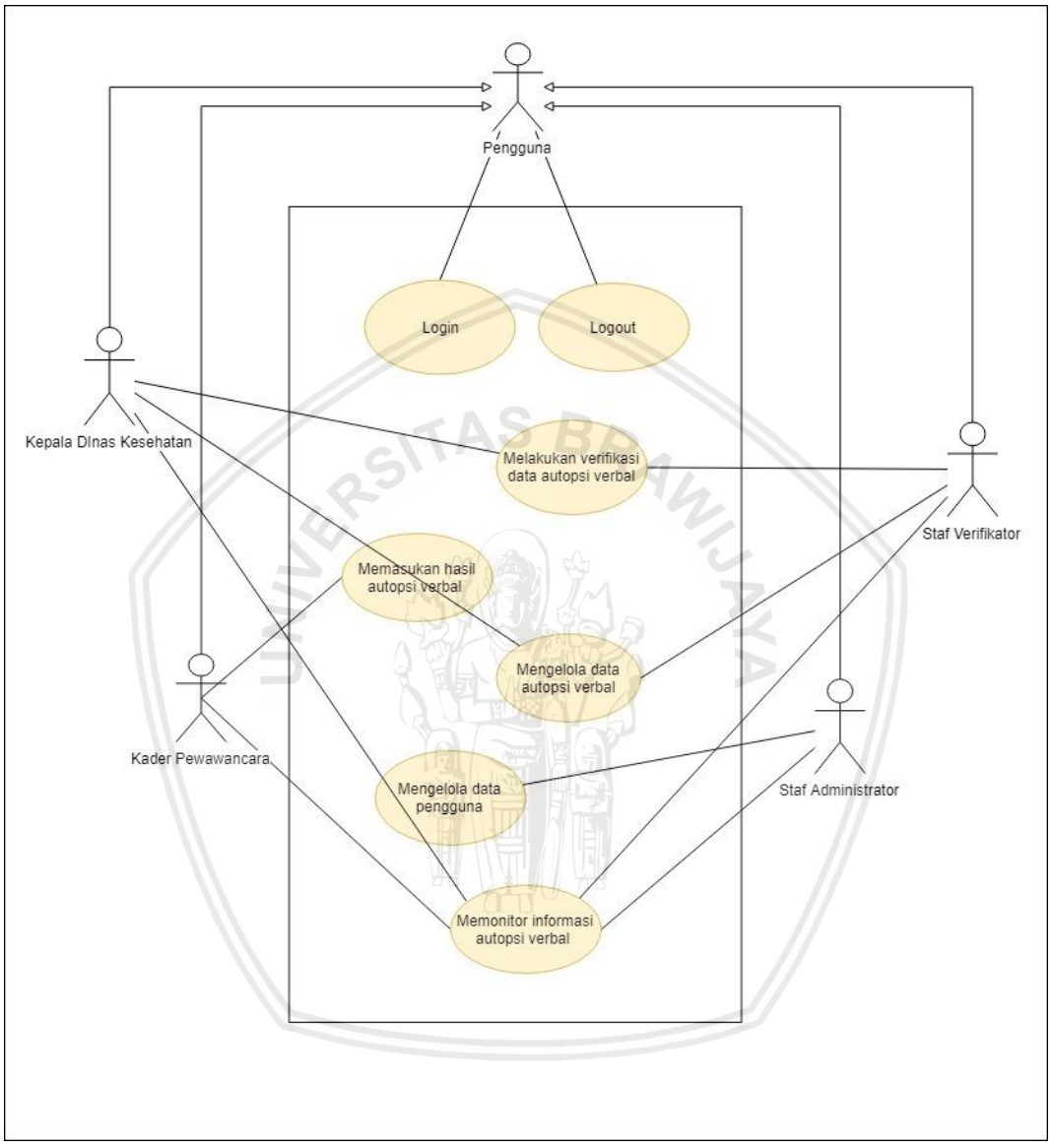
No	Kode Fitur	Kode Persyaratan Non-Fungsional	Deskripsi
1	SIAV-01	SIAV-NF-01	Sistem dapat berjalan versi aplikasi peramban seperti Internet Explorer, Microsoft Edge, Firefox, Safari. Opera, Chrome, iOS dan Android tanpa mengalami kendala tampilan.
2	SIAV-02		
3	SIAV-03		
4	SIAV-04		
5	SIAV-05		

4.3 Pemodelan *Use Case*

Pemodelan *use case* adalah pemodelan yang digunakan sebagai standarisasi untuk menunjukan hubungan aktor dalam menggunakan sistem. Aktor dalam sistem mampu melakukan sesuatu yang dianggap penting dan sistem memberikan umpan balik untuk memenuhi kebutuhan tersebut. Pemodelan *use case* melakukan identifikasi terhadap aktor, identifikasi *use case* yang dibuat secara detail bagi setiap setiap *use case.* Identifikasi aktor dilakukan dengan cara mengelompokkan pengguna sistem dilihat berdasarkan karakteristik yang dimiliki.

4.3.1 Use Case Diagram

Use case menunjukkan hubungan antara aktor dan sistem serta tujuan yang ingin digapai. *Use case diagram* memberikan visualisasi antara hubungan aktor dan *use case. Use case diagram* autopsi verbal terdapat dalam Gambar 4.7.



Gambar 4.7 *Use Case Diagram* Autopsi Verbal

Use case yang teridentifikasi akan terhubung antara solusi yang ditawarkan dalam proses bisnis usulan (*to-be)*, pemangku kepentingan, dan fitur yang telah diidentifikasi sebelumnya. Hubungan dilakukan untuk menunjukan korelasi antara pemodelan proses bisnis dan *use case* yang dibuat. Hubungan *use case* dengan proses bisnis *to-be* ditunjukkan pada Tabel 4.7.

Tabel 4.7 Hubungan antara *Use Case* dengan Aktivitas Proses Bisnis *To-Be*

No	Proses Bisnis To-Be	Aktivitas	*Use Case*
1.	Pengumpulan Data hasil Autopsi Verbal.	Mengisi hasil wawancara pada formulir yang telah dicetak.	Menyimpan data hasil autopsi verbal pada basis data yang telah disediakan sistem
2.	Pelaporan Data Hasil Autopsi Verbal.		Memonitor hasil autopsi verbal dalam bentuk ringkasan umum.

Selanjutnya perlu dihubungkan antara *use case* dengan pemangku kepentingan pada sistem. Hubungan *use case* dengan pemangku kepentingan ditunjukkan pada Tabel 4.8.

Tabel 4.8 Hubungan *Use Case* dengan Pemangku Kepentingan

No	*Use case*	Pengguna
1.	Login, logout	Kader Pewawancara, Staf Administrator Dinas Kesehatan Kabupaten Malang, Staf Verifikator Dinas Kesehatan Kabupaten Malang, Kepala Dinas Kesehatan Kabupaten Malang
2.	Memasukan data hasil autopsi verbal	Kader Pewawancara
3.	Melakukan verifikasi data hasil autopsi verbal	Staf Verifikator Dinas Kesehatan Kabupaten Malang, Kepala Dinas Kesehatan Kabupaten Malang
4.	Mengelola data hasil autopsi verbal	Staf Verifikator Dinas Kesehatan Kabupaten Malang, Kepala Dinas Kesehatan Kabupaten Malang
5.	Mengelola data pengguna	Staf Administrator Dinas Kesehatan Kabupaten Malang
6.	Memonitor informasi autopsi verbal	Kader Pewawancara, Staf Administrator Dinas Kesehatan Kabupaten Malang, Staf Verifikator Dinas Kesehatan Kabupaten Malang, Kepala Dinas Kesehatan Kabupaten Malang

Identifikasi selanjutnya dilakukan antara hubungan *use case* dengan fitur. Hal ini memberikan penjelasan dan perbandingan antara kode fitur dengan *use case* yang telah dianalisis sebelumnya. Hubungan *use case* dengan fitur ditunjukkan pada Tabel 4.9.

Tabel 4.9 Hubungan *Use Case* dengan Fitur

No	*Use Case*	Fitur
1.	Melakukan *login.*	SIAV-05
2.	Melakukan *logout.*	
3.	Memasukan data hasil autopsi verbal.	SIAV-01
4.	Melakukan verifikasi data hasil autopsi verbal.	SIAV-02
5.	Mengelola data hasil autopsi verbal.	
6.	Mengelola data pengguna dalam sistem.	SIAV-03
7.	Memonitor informasi autopsi verbal.	SIAV-04

4.3.2 Deskripsi Aktor

Deskripsi aktor memberikan penjelasan tanggung jawab dan tugas setiap aktor dan tujuan aktor dalam menggunakan sistem. Deskripsi aktor dapat dilihat pada Tabel 4.10.

Tabel 4.10 Deskripsi Aktor

No	Nama Aktor	Deskripsi Aktor
1.	Pengguna	Aktor diperankan oleh individu terkait yang memiliki hubungan langsung dengan sistem. Pengguna dapat berinteraksi dengan sistem sesuai hak akses yang dimiliki pengguna bersangkutan.
2.	Kader Pewawancara	Aktor kader pewawancara diperankan oleh individu yang terkait dengan proses wawancara autopsi verbal. Kader pewawancara dapat menginputkan data hasil wawancara yang telah dilakukan serta melihat dan memonitor informasi autopsi verbal.
3.	Staf Administrator Dinas Kesehatan Kabupaten	Aktor Staf Administrator Dinas Kesehatan Kabupaten Malang diperankan oleh staf yang ada pada Dinas Kesehatan Kabupaten Malang yang terkait dengan autopsi verbal. Aktor Staf Dinas Kesehatan Kabupaten Malang dapat melihat informasi autopsi verbal,

No	Nama Aktor	Deskripsi Aktor
	Malang	mengelola data pengguna dan memberikan hak akses yang diperlukan sesuai dengan jabatan yang diemban.
4.	Staf Verifikator Dinas Kesehatan Kabupaten Malang	Aktor Staf Dinas Kesehatan Kabupaten Malang diperankan oleh staf yang ada pada Dinas Kesehatan Kabupaten Malang yang terkait dengan autopsi verbal. Aktor Staf Dinas Kesehatan Kabupaten Malang dapat melihat dan memonitor informasi autopsi verbal, melakukan verifikasi terhadap data yang dimasukan, mengelola data hasil autopsi verbal.
5.	Kepala Dinas Kesehatan Kabupaten Malang	Aktor Kepala Dinas Kesehatan Kabupaten Malang diperankan oleh Kepala Dinas Kesehatan Kabupaten Malang yang memiliki kewenangan dan menentukan kebijakan terkait autopsi verbal. Aktor Staf Dinas Kesehatan Kabupaten Malang dapat melihat dan memonitor informasi autopsi verbal dan mengelola data hasil autopsi verbal dan mendownload data secara detail.

4.3.3 Spesifikasi *Use Case*

Spesifikasi *use case* memberikan penjelasan terkait hubungan kerja yang terjadi antara aktor dengan setiap *use case* yang teridentifikasi dalam sistem. Spesifikasi *use case* terdiri dari deskripsi dan tujuan yang digapai *use case*, aktor yang menjalankan peran dalam fungsi terkait, syarat kondisi terpenuhi sebelum *use case* bisa dijalankan, hasil yang diharapkan setelah *use case* dijalankan, alur normal yang seharusnya terjadi pada *use case*, pilihan alternatif saat menjalankan *use case* tidak sesuai alur normal.

4.3.3.1 Spesifikasi *Use Case* Login

Spesifikasi *use case login* merupakan penjelasan terkait aktor yang menggunakan sistem dengan akses sesuai ketentuan dan jabatan yang dimiliki pengguna bersangkutan. Spesifikasi *use case Login* tertera pada Tabel 4.11

Tabel 4.11 Spesifikasi *Use Case Login*

Brief Description	*Use case login* menjabarkan langkah yang harus dilakukan pengguna untuk dapat masuk ke dalam ekosistem sesuai hak akses yang sudah dimiliki sebelumnya.
Actor	Kader Pewawancara, Staf Administrator Dinas Kesehatan

	Kabupaten Malang, Staf Verifikator Dinas Kesehatan Kabupaten Malang, Kepala Dinas Kesehatan Kabupaten Malang
Pre-condition	- Perangkat memiliki koneksi untuk terhubung dengan internet. - Sistem dapat terhubung dengan server autopsi verbal. - Data mengenai jabatan dan hak akses aktor tersimpan dan dapat dikenali sistem.
Post-condition	- Aktor dapat masuk pada sistem informasi berbasis web autopsi verbal. - Sistem dapat mendeteksi peran aktor dan hak akses dalam sistem. - Sistem memberikan hak akses yang dimiliki oleh aktor sesuai jabatan yang diemban.
Basic Flow	**{*use case* mulai dilakukan}** 1. Aktor kader pewawancara, staf administrator dinas kesehatan kabupaten malang, staf verifikator dinas kesehatan kabupaten malang, dan kepala dinas kesehatan memilih untuk masuk ke dalam sistem. **{isi data}** 2. Kader pewawancara, staf administrator dinas kesehatan kabupaten malang, staf verifikator dinas kesehatan kabupaten malang, dan kepala dinas kesehatan mengisi *username* dan *password* yang benar. 3. Kader pewawancara, staf administrator dinas kesehatan kabupaten malang, staf verifikator dinas kesehatan kabupaten malang, dan kepala dinas kesehatan kabupaten malang melakukan. 4. Sistem mengautentifikasi pengguna dengan memeriksa email yang dimiliki/*username* dan *password.* **{mengautentifikasi pengguna}** 5. Sistem menyimpan informasi mengenai identitas dan hak akses pengguna. 6. Status *login* berhasil dilakukan sistem dan pengguna masuk ke dalam sistem. **{*use case* telah selesai dilakukan}** 7. *Use case* telah selesai.

Alternative Flow	**A1. Menangani *email/Username* atau *Password* Kosong** Pada **{mengautentifikasi pengguna}** apabila kader pewawancara, staf dinas kesehatan, dan kepala dinas kesehatan tidak mengisi salah satu email/*username* maupun *password*, sistem akan mendeteksi dan muncul berupa pesan eror yang memberitahukan pengguna bahwa email/*username* dan *password* kosong serta wajib diisikan oleh pengguna, kemudian kembali ke **{isi data}**. **A2. Menangani *email/Username* atau *Password* Salah** Apabila *email*/*username* dan *password* yang dimasukan oleh pengguna mengalami kesalahan maupun tidak teridentifikasi, maka sistem akan menampilkan pesan berupa masukan salah.
Sub Flow	

4.3.3.2 Spesifikasi *Use Case* Logout

Spesifikasi *use case logout* menjelaskan aktor yang akan keluar dari sistem yang telah digunakan. Spesifikasi *use case logout* memberi penjelasan urutan kegiatan yang harus dilakukan aktor saat selesai berinteraksi dengan sistem dan ingin meninggalkan sistem. Spesifikasi *use case logout* terdapat pada Tabel 4.12.

Tabel 4.12 Spesifikasi *Use Case Logout*

Brief Description	*Use case* logout memberi penjelasan tentang bagaimana aktor keluar dari sistem ketika telah selesai menggunakan sistem.
Actor	Kader Pewawancara, Staf Administrator Dinas Kesehatan Kabupaten Malang, Staf Verifikator Dinas Kesehatan Kabupaten Malang, Kepala Dinas Kesehatan Kabupaten Malang
Pre-condition	- Perangkat memiliki koneksi untuk terhubung dengan internet. - Sistem dapat terhubung dengan server autopsi verbal. - Data mengenai jabatan dan hak akses aktor tersimpan dan dapat dikenali sistem.
Post-condition	Aktor berhasil keluar dari lingkungan sistem.
Basic Flow	**{*use case* mulai dilakukan}** 1. Aktor memilih untuk keluar dari sistem dengan menggunakan fungsi *logout* pada sistem. 2. Sistem menghapus hak akses yang telah diberikan kepada

	user. **{*use case* telah selesai dilakukan}** *3. Use case* telah selesai.
Alternative Flow	-
Sub Flow	

4.3.3.3 Spesifikasi *Use Case* Memasukan Data Hasil autopsi Verbal

Spesifikasi *use case* memasukan data hasil autopsi verbal menjelaskan aktor yang akan menggunakan sistem untuk memasukan data hasil wawancara verbal yang sebelumnya dilakukan. Spesifikasi *use case* memasukan data hasil autopsi verbal memberi penjelasan urutan kegiatan yang harus dilakukan aktor saat memasukan hasil data wawancara ke dalam sistem. Spesifikasi *use case* memasukan data hasil autopsi verbal terdapat pada Tabel 4.13

Tabel 4.13 Spesifikasi *Use Case* Memasukan Data Hasil autopsi Verbal

Brief Description	*Use case* memasukan data hasil autopsi verbal menjelaskan bagaimana aktor kader pewawancara menggunakan sistem untuk menampilkan formulir wawancara autopsi verbal, mengisikan formulir dengan data hasil autopsi verbal, mengirimkan data hasil autopsi verbal menuju basis data dan menyimpan data hasil wawancara pada basis data yang disediakan.
Actor	Kader Pewawancara
Pre-condition	- Perangkat memiliki koneksi untuk terhubung dengan internet. - Sistem dapat terhubung dengan server autopsi verbal. - Data mengenai jabatan dan hak akses aktor tersimpan dan dapat dikenali sistem.
Post-condition	Sistem berhasil menyimpan data hasil autopsi verbal
Basic Flow	**{*use case* mulai dilakukan}** 1. Kader pewawancara memilih menambahkan data hasil wawancara untuk dimasukan ke dalam sistem. 2. Sistem memberikan tampilan formulir wawancara autopsi verbal kepada kader pewawancara. **{Pengisian data autopsi verbal}** 3. Kader pewawancara memasukkan data yang terdapat pada

	formulir autopsi verbal. Data yang dimasukkan terdiri dari data identitas almarhum, data narasumber dan nama kader, data kecelakaan almarhum, kondisi kronis yang pernah dialami almarhum, daftar keluhan almarhum, konsumsi almarhum terhadap tembakau serta rekaman kesehatan almarhum. 4. Kader pewawancara melakukan penyimpanan hasil pencatatan data wawancara autopsi verbal. **{menyimpan data hasil autopsi verbal}** 5. Sistem memproses data wawancara autopsi verbal untuk disimpan dalam basis data. **{sistem melakukan penyimpanan terhadap data hasil autopsi verbal}** 6. Sistem memberi pemberitahuan data berhasil tersimpan **{*use case* telah selesai dilakukan}** 7. *Use case* telah selesai.
Alternative Flow	**A1. Format Tidak Sesuai Ketentuan dan Isian yang akan dimasukan Kosong** Pada saat **{sistem melakukan penyimpanan data hasil autopsi verbal}** apabila pada formulir yang wajib diiisi ternyata masih kosong, maka sistem akan menampilkan pesan kepada kader pewawancara berupa isian wajib diisi. **{kembali menuju {isi data}}.** **A2. Penyimpanan Data Hasil autopsi Verbal Gagal** Pada **{sistem melakukan penyimpanan data hasil autopsi verbal }** sistem akan menampilkan pesan eror saat data gagal dimasukan kedalam sistem serta menampilkan kembali formulir kosong autopsi verbal. **{kembali ke formulir}**
Sub Flow	-

4.3.3.4 Spesifikasi *Use Case* Melakukan Verifikasi Data hasil autopsi Verbal

Spesifikasi *use case* melakukan verifikasi data hasil autopsi verbal menjelaskan aktor saat melakukan verifikasi data yang telah dikumpulkan dari hasil autopsi verbal. Aktor dapat melakukan proses verifikasi data hasil autopsi verbal, memberikan catatan tambahan maupun melakukan penghapusan data hasil autopsi verbal. Spesifikasi *use case* melakukan vverifikasi data hasil autopsi verbal terdapat pada Tabel 4.14

Tabel 4.14 Spesifikasi *Use Case* Melakukan Verifikasi Data Hasil Autopsi Verbal

Brief Description	*Use case* memverifikasi data hasil autopsi verbal menjelaskan bagaimana aktor staf verifikator dinas kesehatan dan kepala dinas kesehatan kabupaten malang dapat menggunakan sistem untuk memverifikasi data hasil autopsi verbal. Pengguna dapat menampilkan ringkasan data dalam bentuk tabel, melakukan verifikasi data, memberikan catatan tambahan dan melakukan penghapusan data.
Actor	Staf verifikator dinas kesehatan kabupaten malang atau Kepala Dinas Kesehatan Kabupaten Malang.
Pre-condition	- Perangkat memiliki koneksi untuk terhubung dengan internet. - Sistem dapat terhubung dengan server autopsi verbal. - Data mengenai jabatan dan hak akses aktor tersimpan dan dapat dikenali sistem.
Post-condition	- Staf verifikator dinas kesehatan kabupaten malang dan Kepala Dinas Kesehatan Kabupaten Malang dapat melihat data hasil autopsi verbal yang belum terverifikasi di dalam sistem. - Staf verifikator dinas kesehatan kabupaten malang dan Kepala Dinas Kesehatan Kabupaten Malang dapat melakukan verifikasi data hasil autopsi verbal.
Basic Flow	**{*use case* mulai dilakukan}** 1. *Use case* dimulai ketika aktor staf verifikator dinas kesehatan kabupaten malang dan Kepala Dinas Kesehatan Kabupaten Malang memilih untuk melihat ringkasan data hasil autopsi verbal yang belum terverifikasi. 2. Sistem menampilkan tabel data hasil autopsi verbal yang belum terverifikasi. **{menampilkan data hasil autopsi verbal}** 3. Staf verifikator dinas kesehatan kabupaten malang memilih untuk melihat detail catatan kematian setiap individu. **{menampilkan detail data hasil autopsi verbal}** 4. Staf verifikator dinas kesehatan kabupaten malang memilih untuk melakukan verifikasi data hasil autopsi verbal. **{menampilkan pilihan verifikasi data}** 5. Sistem menampilkan pesan verifikasi data serta pilihan untuk menambahkan catatan berupa pesan dari verifikator jika

	diperlukan . **{sistem melakukan verifikasi data hasil autopsi verbal }** **{*use case* telah selesai dilakukan}** 6. *Use case* telah selesai dilakukan.
Alternative Flow	**A1. Menghapus Data hasil autopsi Verbal yang Belum Terverifikasi.** Alternatif *flow* merupakan alternatif dalam melakukan penghapusan data hasil autopsi verbal yang tidak sesuai maupun mengalami kesalahan. 1. *Alternatif flow* ini dijalankan ketika staf verifikator dinas kesehatan kabupaten malang dan kepala Dinas Kesehatan Kabupaten Malang memilih untuk menghapus data hasil autopsi verbal. 2. Sistem menampilkan formulir hasil autopsi verbal beserta isian yang telah diisikan sebelumnya oleh pengguna. **{tampil hapus data}**. 3. Staf verifikator dinas kesehatan kabupaten malang dan kepala Dinas Kesehatan Kabupaten Malang menghapus data. **{menghapus data}** 4. Sistem menghapus data hasil autopsi verbal. **{sistem menghapus data}** 5. *Use case* telah selesai dilakukan.
Sub Flow	-

4.3.3.5 Spesifikasi *Use Case* Mengelola Data Hasil autopsi Verbal

Spesifikasi *use case* mengelola data hasil autopsi verbal memberi penjabaran aktor yang akan menggunakan sistem pada proses pengelolaan data yang telah dikumpulkan dari hasil autopsi verbal. Aktor dapat melakukan proses untuk melihat data, menghapus dan juga mendownload data dalam format excel. Spesifikasi *use case* mengelola data hasil autopsi verbal terdapat pada Tabel 4.15

Tabel 4.15 Spesifikasi *Use Case* Mengelola Data Hasil autopsi Verbal

Brief Description	*Use case* mengelola data hasil autopsi verbal menjelaskan bagaimana aktor staf verifikator dinas kesehatan dan kepala dinas kesehatan kabupaten malang dapat menggunakan sistem dalam proses pengelolaan data hasil autopsi verbal. Pengguna dapat menampilkan ringkasan data dalam bentuk tabel, melihat

	detail data, menghapus data yang ada dan melakukan download data hasil autopsi verbal dalam bentuk excel.
Actor	Staf verifikator dinas kesehatan kabupaten malang dan dan kepala dinas kesehatan kabupaten malang
Pre-condition	- Perangkat memiliki koneksi untuk terhubung dengan internet. - Sistem dapat terhubung dengan server autopsi verbal. - Data mengenai jabatan dan hak akses aktor tersimpan dan dapat dikenali sistem.
Post-condition	- Staf verifikator dinas kesehatan kabupaten malang dan kepala dinas kesehatan kabupaten malang memiliki akses untuk melihat data hasil autopsi verbal yang berhasil tersimpan di dalam sistem. - Staf verifikator dinas kesehatan kabupaten malang dan kepala dinas kesehatan kabupaten malang dapat mengunduh data catatan kematian dalam format excel.
Basic Flow	**{*use case* mulai dilakukan}** 1. Staf verifikator dinas kesehatan kabupaten malang dan kepala dinas kesehatan kabupaten malang memilih untuk melihat ringkasan data hasil autopsi verbal dalam bentuk tabel. 2. Sistem menampilkan tabel hasil autopsi verbal yang tersimpan pada basis data dalam bantuk tabel. **{menampilkan data hasil autopsi verbal}** 3. Staf verifikator dinas kesehatan kabupaten malang dan kepala dinas kesehatan kabupaten malang memilih untuk melihat detail catatan kematian setiap individu. **{menampilkan detail data hasil autopsi verbal}** 4. Staf verifikator dinas kesehatan kabupaten malang dan kepala dinas kesehatan kabupaten malang memilih untuk mengunduh data hasil autopsi verbal. **{menampilkan pilihan mengunduh data}** 5. Sistem melakukan proses unduh data dalam bentuk excel kepadapengguna. **{sistem memproses unduhan data hasil autopsi verbal }** 6. *sub flow* **Menghapus Data Hasil autopsi Verbal** **{*use case* telah selesai}** 7. *Use case* telah selesai dilakukan.

Alternative Flow	**A1. Menghapus Data Hasil autopsi Verbal.** Alternatif *flow* ini digunakan untuk melakukan penghapusan data hasil autopsi verbal yang tidak sesuai maupun mengalami kesalahan. 1. *Alternatif flow* ini dijalankan ketika staf verifikator dinas kesehatan kabupaten malang dan kepala dinas kesehatan kabupaten malang memilih untuk menghapus data hasil autopsi verbal. 2. Sistem menampilkan formulir hasil autopsi verbal beserta isian yang telah diisikan sebelumnya oleh pengguna. **{tampil hapus data}**. 3. Staf verifikator dinas kesehatan kabupaten malang menghapus data. **{menghapus data}** 4. Sistem menghapus data hasil autopsi verbal. **{sistem menghapus data}** 5. *Use case* telah selesai dilakukan.
Sub Flow	-

4.3.3.6 Spesifikasi *Use Case* Mengelola Data Pengguna

Spesifikasi *use case* mengelola data pengguna memberikan penjabaran yang dilakukan staf administrator dinas kesehatan kabupaten malang untuk menggunakan sistem sesuai tujuan dan peran yang diemban setiap pengguna. Proses mengelola data pengguna terdapat pada Tabel 4.16.

Tabel 4.16 Spesifikasi *Use Case* Mengelola Data Pengguna

Brief Description	*Use case* mengelola data pengguna memberi penjabaran staf administrator dinas kesehatan kabupaten malang melakukan penambahan data pengguna, melihat data pengguna, melakukan pengubahan data pengguna dan peran yang dimiliki serta menghapus data pengguna.
Actor	Staf administrator dinas kesehatan kabupaten malang.
Pre-condition	- Perangkat memiliki koneksi untuk terhubung dengan internet. - Sistem dapat terhubung dengan server autopsi verbal. - Data mengenai jabatan dan hak akses aktor tersimpan dan dapat dikenali sistem.

Post-condition	- Sistem dapat menyimpan data pengguna sistem informasi autopsi verbal dengan sukses.
Basic Flow	**{*use case* mulai dilakukan}** 1. Staf administrator dinas kesehatan kabupaten malang memilih untuk melakukan proses penambahan data pengguna. 2. Sistem memberikan tampilan formulir tambah pengguna. **{isi data pengguna}** 3. Staf administrator dinas kesehatan kabupaten malang memasukkan data pengguna berupa *username, password,* nama dan juga jabatan serta peran dalam sistem. 4. Staf administrator dinas kesehatan kabupaten malang memilih untuk melakukan penyimpanan data pengguna. **{melakukan penyimpanan data pengguna}** 5. Sistem melakukan proses penyimpanan data pengguna. **{sistem menyimpan data pengguna}** 6. Sistem memberikan tampilan data pengguna yang telah tersimpan dalambentuk tabel. **{menampilkan data pengguna}** **{*use case* selesai dilakukan}** 7. *Use case* telah selesai dilakukan.
Alternative Flow	**A1. Melihat Data Pengguna** Pada **{menampilkan data pengguna}** pada *basic flow,* apabila staf administrator dinas kesehatan kabupaten malang memilih untuk melihat data pengguna, maka: 1. Staf administrator dinas kesehatan kabupaten malang memilih menu data pengguna. 2. Sistem memuat daftar data pengguna setiap individu. **{menampilkan data pengguna}** 3. Sistemmemberikan tampilan detail data setiap pengguna. 4. *Use case* selesai dilakukan. **A2. Melakukan Pengubahan Data Pengguna** Pada **{menampilkan data pengguna},** staf administrator dinas kesehatan kabupaten malang memilih untuk melakukan pengubahan data, maka : 1. Sistem menampilkan formulir pengguna yang dapat

	diubah datanya. **{tampil mengubah data}** **{isi data}** 2. Staf dinas administrator kesehatan kabupaten malang melakukan pengubahan data. 3. Staf dinas administrator kesehatan kabupaten malang menyimpan data. **{menyimpan data pengguna*}*** 4. Sistem mengubah data pengguna. **{sistem mengubah data}** 5. Sistem memberikan tampilan data pengguna dalam tabel yang telah diperbarui. 6. *Use case* selesai. **A3. Melakukan Penghapusan Data Pengguna** Pada **{menampilkan data pengguna}**, apabila staf administrator dinas kesehatan kabupaten malang memilih untuk melakukan penghapusan data, maka : 1. Sistem menampilkan konfirmasi berupa pesan untuk menghapus data pengguna. **{konfirmasi hapus}** 2. Staf administrator dinas kesehatan kabupaten malang melakukan pengghapusan data pengguna 3. Sistem menghapus data pengguna pada basis data. **{sistem menghapus data pengguna}** 4. Sistem menampilkan data pengguna dalam tabel yang telah diperbarui. *Use case* selesai dilakukan.
Sub Flow	-

4.3.3.7 Spesifikasi *Use Case* Memonitor Informasi Autopsi Verbal

Spesifikasi *use case* memonitor informasi autopsi verbal menjelaskan tujuan dan cara yang dilakukan aktor pengguna kader pewawancara, staf dinas kesehatan kabupaten malang, kepala dinas kesehatan kabupaten malang memonitor informasi hasil autopsi verbal. Aktor dapat melakukan monitor informasi autopsi verbal. Spesifikasi *use case* memonitor informasi autopsi verbal terdapat pada Tabel 4.17

Tabel 4.17 Spesifikasi *Use Case* Memonitor Informasi Autopsi Verbal

Brief Description	Aktor kader pewawancara, staf administrator dinas kesehatan kabupaten malang, staf verifikator dinas kesehatan kabupaten malang, dan kepala dinas kesehatan kabupaten malang memonitor atau melihat grafik hasil autopsi verbal yang telah dilakukan.
Actor	Kader Pewawancara, Staf Administrator Dinas Kesehatan Kabupaten Malang, Staf Verifikator Dinas Kesehatan Kabupaten Malang, Kepala Dinas Kesehatan Kabupaten Malang
Pre-condition	- Perangkat memiliki koneksi untuk terhubung dengan internet. - Sistem dapat terhubung dengan server autopsi verbal. - Data mengenai jabatan dan hak akses aktor tersimpan dan dapat dikenali sistem.
Post-condition	kader pewawancara, staf administrator dinas kesehatan kabupaten malang, staf verifikator dinas kesehatan kabupaten malang, dan kepala dinas kesehatan kabupaten malang dapat melihat informasi umum dari hasil autopsi verbal.
Basic Flow	**{*use case* mulai dilakukan}** 1. *Use case* dimulai ketika kader pewawancara, staf administrator dinas kesehatan kabupaten malang, staf verifikator dinas kesehatan kabupaten malang, dan kepala dinas kesehatan kabupaten malang memilih fungsi yang dibutuhkan. 2. Sistem akan menampilkan informasi hasil autopsi verbal untuk setiap periode. **{menampilkan informasi autopsi verbal}** **{*use case* selesai dilakukan}** 3. *Use case* telah selesai dilakukan
Alternative Flow	-

4.4 Perancangan dan Evaluasi Prototipe

Perancangan awal prototipe digunakan agar pihak Dinas Kesehatan Kabupaten Malang mendapatkan gambaran awal sistem informasi yang akan dibuat. Pengembangan prototipe dilakukan dengan mempertimbangkan analisis kebutuhan yang telah dilakukan sebelumnya. Prototipe yang telah selesai dibuat kemudian ditunjukan kepada pihak Dinas Kesehatan Kabupaten Malang.

Kesesuain antara kebutuhan dan keinginan pengguna menjadi bahan pertimbangan untuk melanjutkan ke tahap selanjutnya. Prototipe yang telah disetujui kemudian dilanjutkan menuju tahap perancangan dan implementasi sistem.

4.4.1 Perancangan Prototipe

Perancangan prototipe dilakukan dengan menggambarkan prototipe *low fidelity* yang merupakan sketsa awal sistem informasi. Sketsa yang ada akan dikembangkan menjadi prototipe low fidelity dengan bantuan perangkat lunak desain yang dikeluarkan Adobe.Inc yaitu Adobe XD untuk menggambarkan rancangan desain antarmuka pengguna. Berdasarkan identifikasi fitur yang telah dilakukan sebelumnya, terdapat empat fitur utama sistem informasi yang dikonversi menjadi empat halaman antar muka pengguna yang dijadikan prototipe untuk ditunjukan kepada Dinas Kesehatan Kabupaten Malang. Fitur yang dibuat prototipenya berasal dari analisis *use case* mencatat data hasil autopsi verbal, mengelola data hasil autopsi verbal, mengelola data pengguna dan memonitor informasi autopsi verbal. Terdapat halaman antarmuka dasbor, manajamen data, manajemen pengguna dan formulir. Hubungan antara *use case* dengan perancangan awal prototipe dapat dilihat pada tabel 4.18 berikut.

Tabel 4.18 Hubungan *Use Case* dengan Fitur

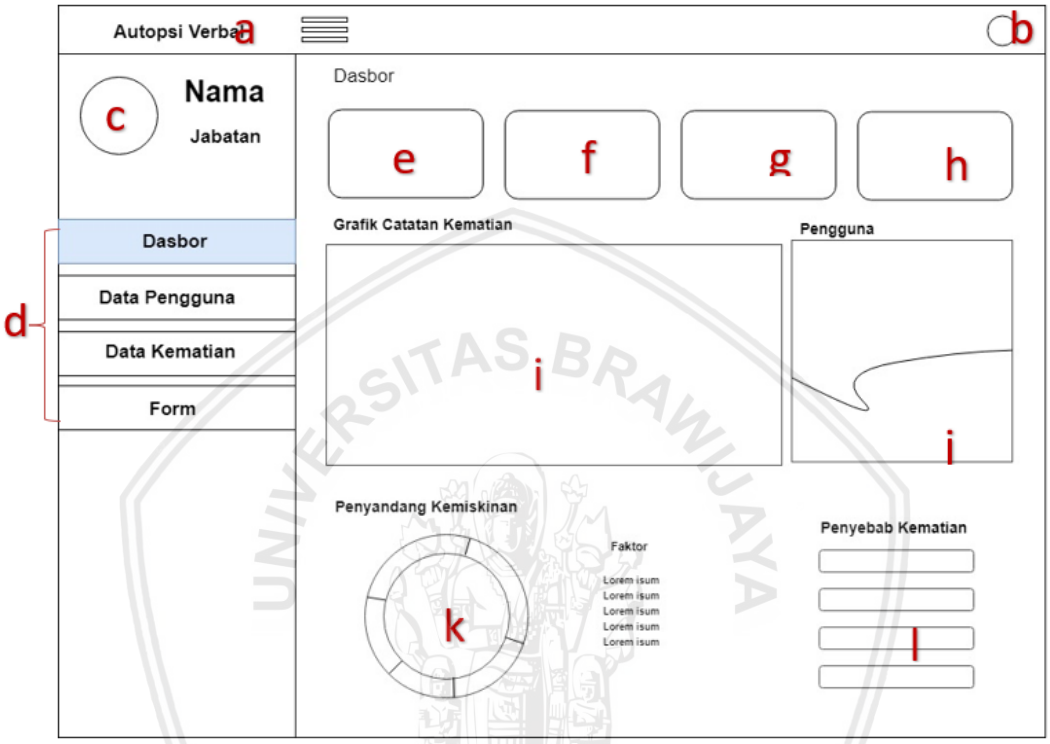
No	*Use Case*	Prototipe
1.	Mencatat hasil autopsi verbal.	Formulir
2.	Melakukan verifikasi data hasil autopsi verbal	Verifikasi
3.	Mengelola data hasil autopsi verbal.	Data Kematian
4.	Mengelola data pengguna dalam sistem.	Data Pengguna
5.	Memonitor informasi autopsi verbal	Dasbor

Prototipe awal merupakan rancangan antar muka pengguna *low fidelity* yang memberikan gambaran awal sistem informasi. Prototipe ini berisi sketsa tampilan antarmuka pengguna sistem informasi autopsi verbal. Sketsa yang dibuat antara lain sketsa antarmuka dasbor, data kematian, data pengguna dan formulir.

4.4.1.1 Prototipe *Low Fidelity*

1. Dasbor

Pada halaman dashboard berisi mengenai ringkasan terkait autopsi verbal yang terjadi di kota Malang. Perancangan antarmuka tersebut ditunjukkan pada Gambar 4.8.



Gambar 4.8 Prototipe *Low Fidelity* Dasbor

Keterangan:

a. Judul sistem
b. Menu logout
c. Profil user
d. Menu utama
e. Data jumlah kematian
f. Data jumlah petugas survey
g. Jumlah kematian bulan Maret
h. Jumlah pengguna aktif aplikasi dashboard
i. Grafik catatan kematian

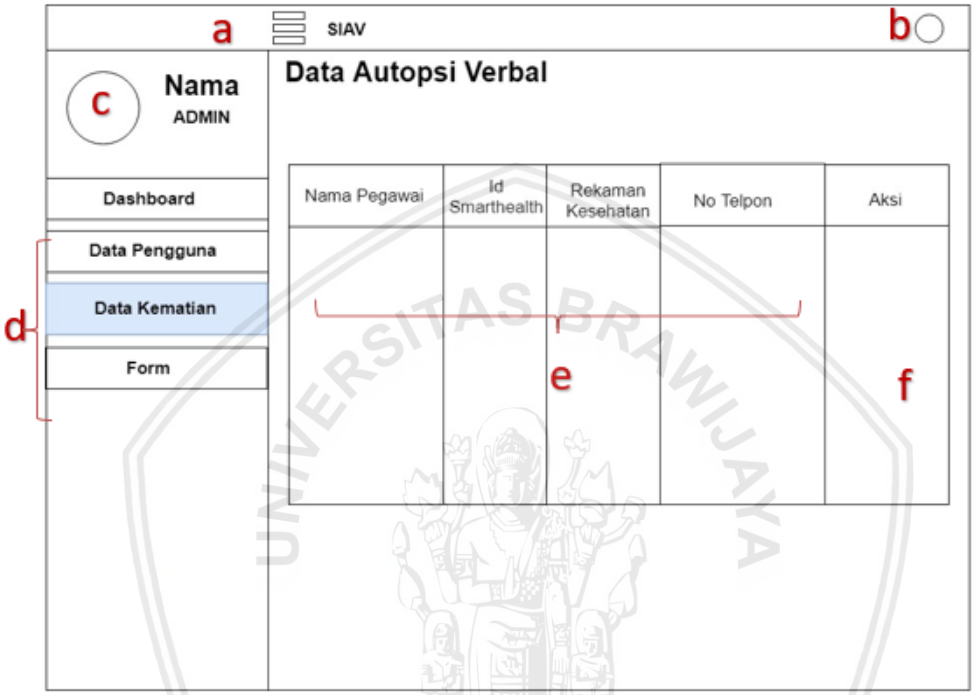
j. Grafik pengguna aktif

k. *Donut chart* dari data grafik penyakit

l. Grafik penyebab kematian

2. Data Kematian

Pada halaman ini berisi mengenai data mengenai autopsi verbal terkait dengan nama, riwayat penyakit, alamat dan aksi.



Gambar 4.9 Prototipe *Low Fidelity* Data Kematian

Keterangan:

a. Judul sistem

b. Menu logout

c. Profil user

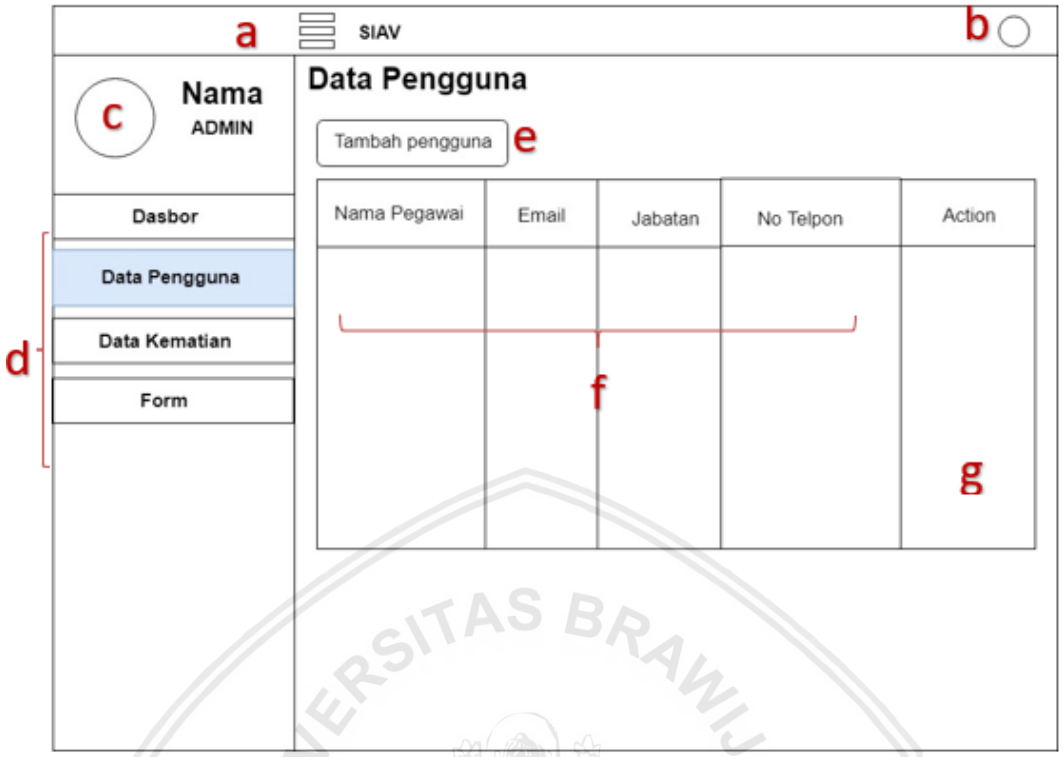
d. Menu utama

e. Informasi pengguna

f. Kolom Aksi

3. Data Pengguna

Manajemen pengguna berisi mengenai pengguna mana saja yang dapat mengakses data tersebut, yang berisi tambah pengguna, email, username, password dan aksi.



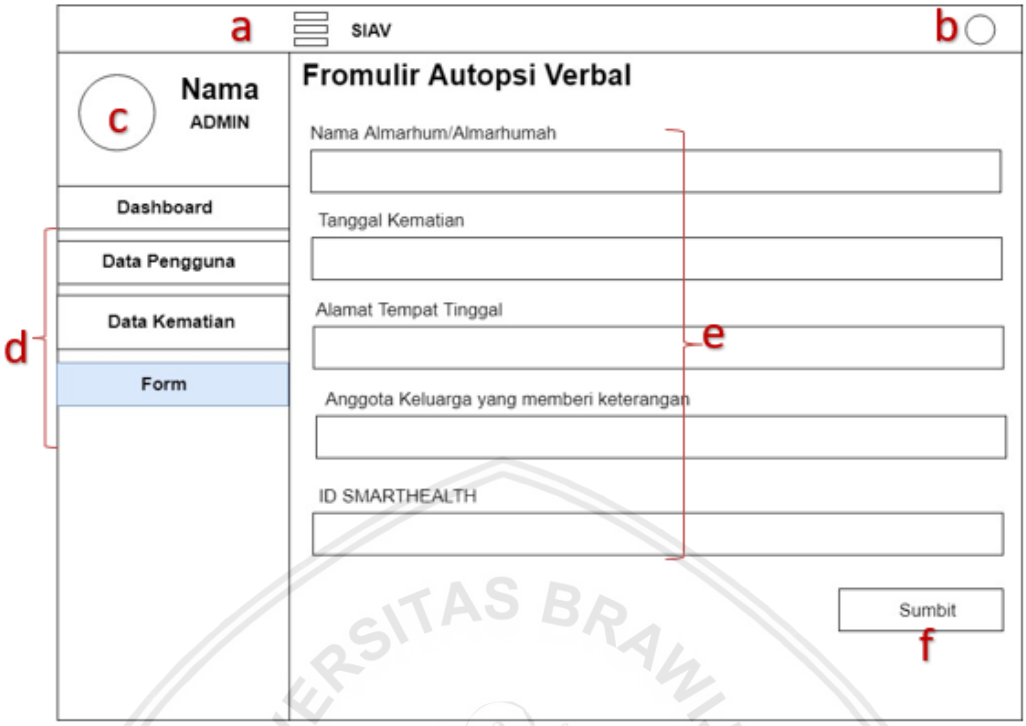
Gambar 4.10 Prototipe *Low Fidelity* Data Pengguna

Keterangan:

a. Judul sistem
b. Menu logout
c. Profil user
d. Menu utama
e. Tambah Data
f. Informasi pengguna
g. Kolom action

4. Formulir

Pada halaman formulir, pengguna dapat menambahkan data terkait autopsy verbal yang terjadi di Kota Malang, mulai dari nama, riwayat penyakit, alamat, verifikasi dan kembali ke kader.



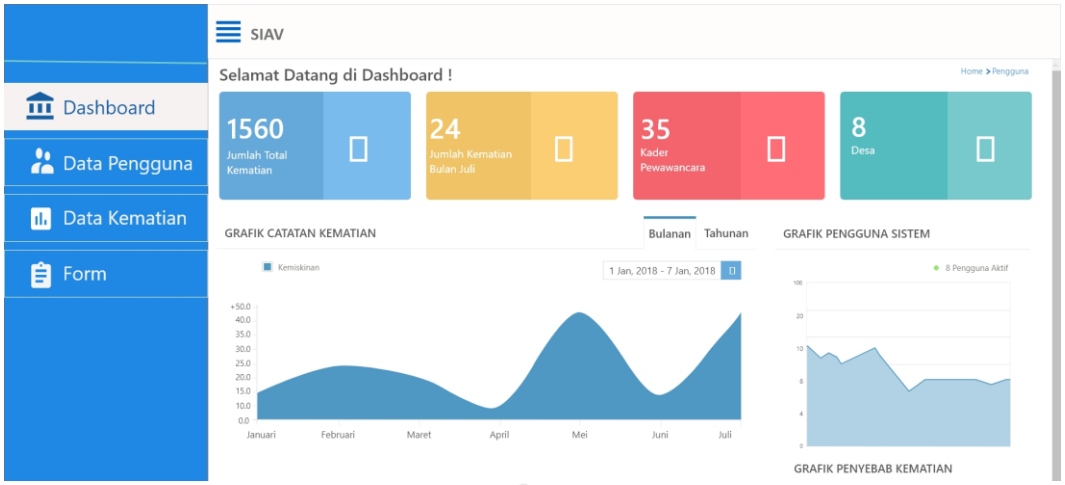
Gambar 4.11 Prototipe *Low Fidelity* Formulir

Keterangan :

a. Judul sistem
b. Menu logout
c. Profil user
d. Menu utama
e. Tambah data
f. Tombol Submit

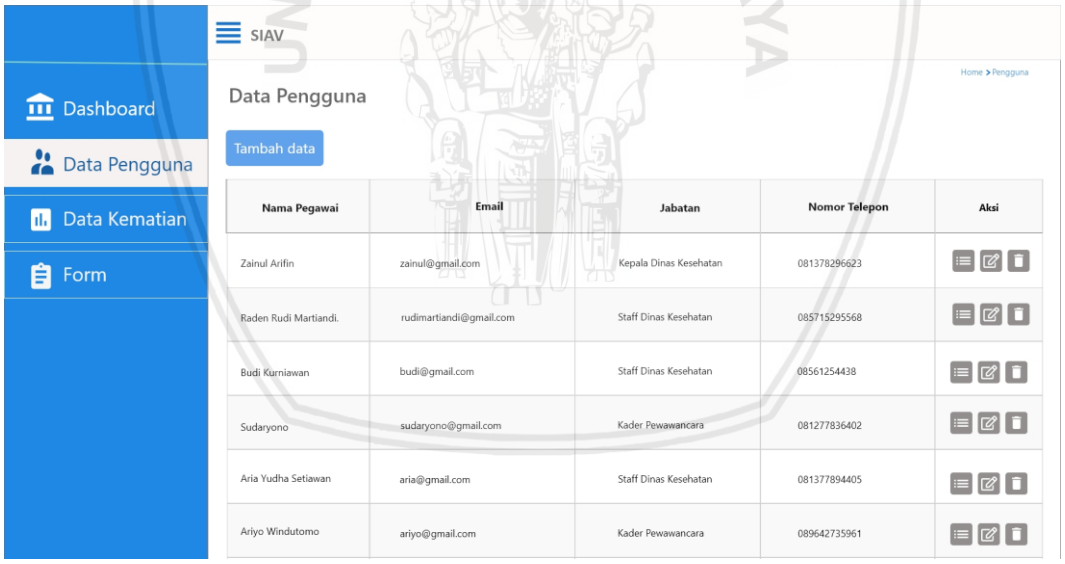
4.4.1.2 Prototype *High Fidelity*

Pada halaman dasbor terdapat empat kotak yang menggambarkan data aplikasi secara umum. Prototipe halaman dasbor dapat dilihat pada gambar 4.12 berikut.



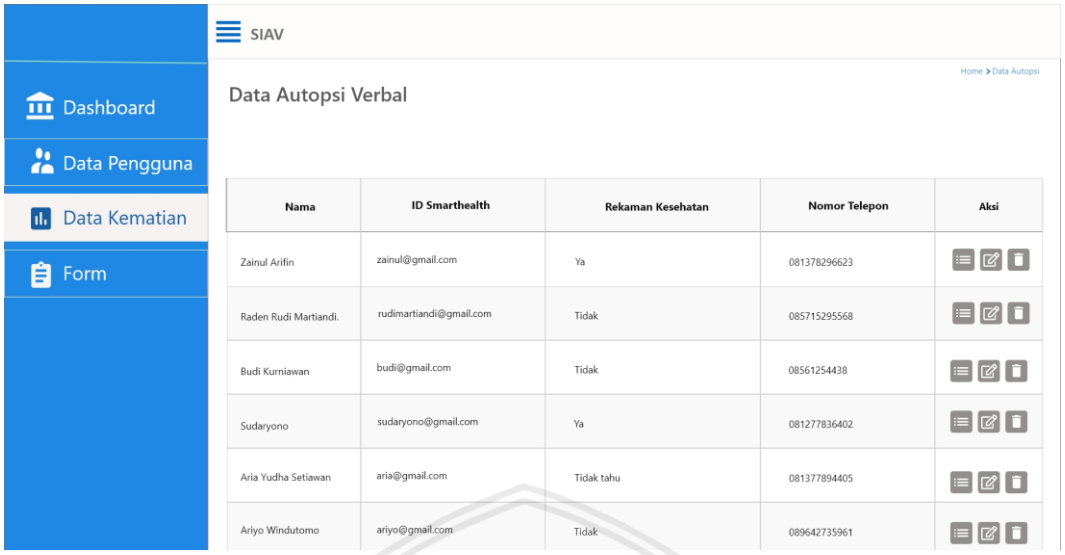
Gambar 4.12 Prototipe Halaman Dasbor

Prototipe data pengguna merupakan visualisasi dari *use case* mengelola data pengguna yang telah dianalisis sebelumnya. Pengguna dapat melakukan pengolahan data pengguna dengan adanya tambah data pengguna, melihat detail pengguna, melakukan edit pengguna dan juga menghapus pengguna. Prototipe halaman data pengguna dapat dilihat pada gambar 4.13 berikut.



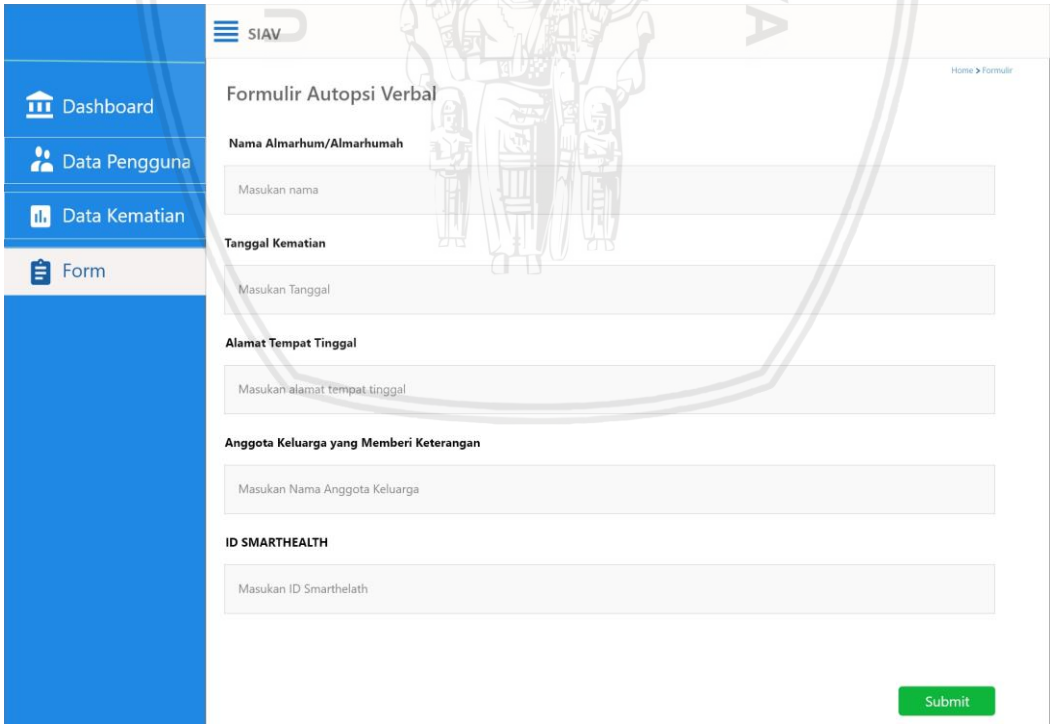
Gambar 4.13 Prototipe Halaman Data Pengguna

Prototipe data kematian merupakan visualisasi dari *use case* mengelola data kematian yang telah dianalisis sebelumnya. Pengguna dapat melakukan pengolahan data kematian dengan melihat detail data kematian, melakukan edit data dan juga menghapus data. Prototipe halaman data kematian dapat dilihat pada gambar 4.14 berikut.



Gambar 4.14 Prototipe Halaman Data kematian

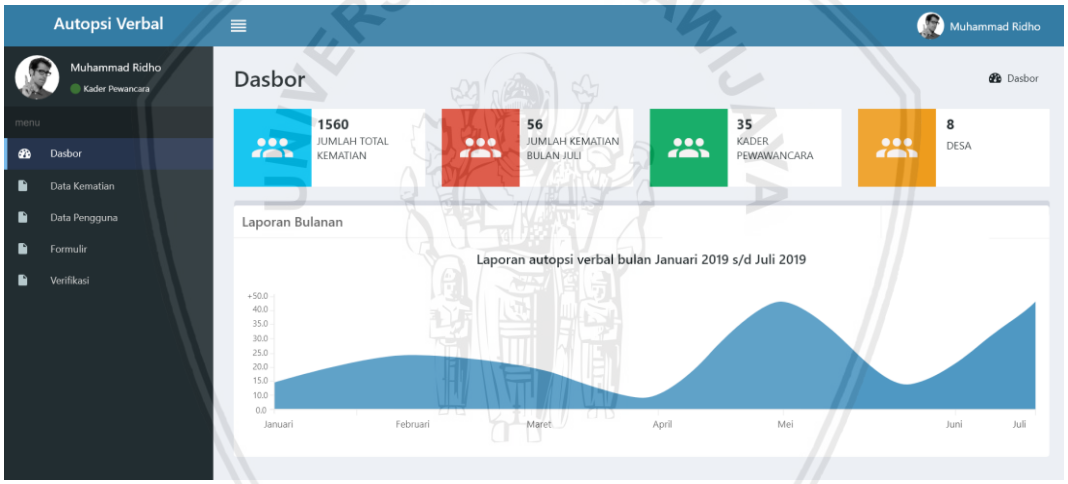
Prototipe formulir merupakan visualisasi dari *use case* memasukan hasil wawancara yang telah dianalisis sebelumnya. Pengguna dapat melakukan *input* formulir pada halaman formulir. Prototipe halaman formulir dapat dilihat pada gambar 4.15 berikut.



Gambar 4.15 Prototipe Halaman Formulir

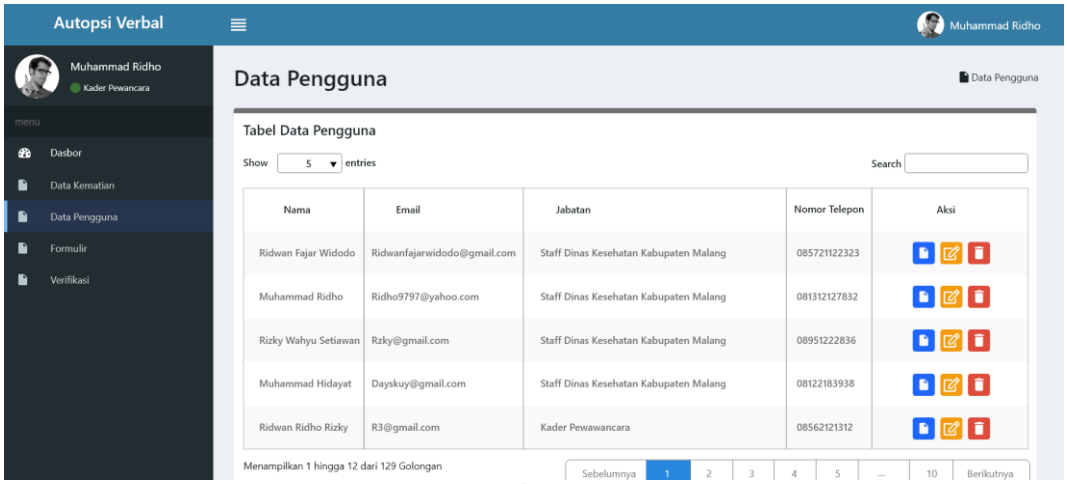
4.4.2 Evaluasi Prototipe

Evaluasi prototipe dilakukan setelah pihak Dinas Kesehatan Kabupaten Malang memberikan umpan balik berupa komentar maupun saran terhadap prototipe awal sistem informasi autopsi verbal. Terdapat beberapa saran yang diberikan pada prototipe awal. Saran perbaikan dilakukan terhadap data verifikasi yang dibutuhkan secara terpisah sehingga dibuat halaman baru yaitu halaman verifikasi. Saran selanjutnya adalah aksi penghapusan pada data kematian dihilangkan sehingga data kematian yang ada hanya bisa dilihat maupun diedit. Perbaikan dilakukan agar pihak Dinas Kesehatan dapat menggunakan sistem informasi sesuai kebutuhan yang diinginkan. Halaman antarmuka pengguna memiliki total evaluasi sebanyak 3 kali. Berikut merupakan tahap final dari perbaikan yang telah dilakukan. Perbaikan dilakukan terhadap halaman antarmuka dasbor, manajamen data, manajemen pengguna dan formulir. Perbaikan prototipe halaman dasbor dapat dilihat pada gambar 4.16 berikut.



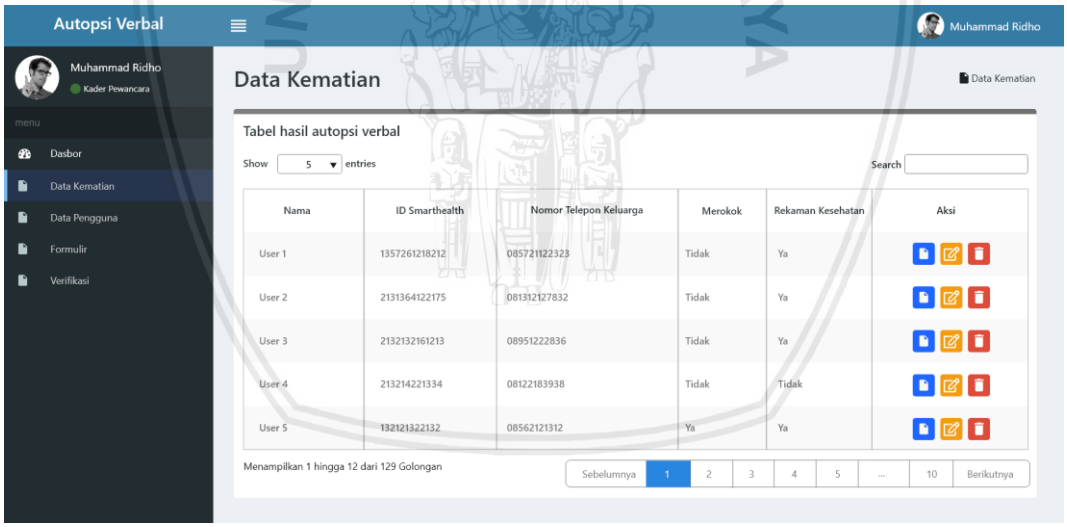
Gambar 4.16 Evaluasi Prototipe Halaman Dasbor

Prototipe data pengguna merupakan visualisasi dari *use case* mengelola data pengguna yang telah dianalisis sebelumnya. Perbaikan terhadap prototipe data pengguna terdapat pada tema sistem secara keseluruhan dan pemilihan aksi yang lebih berwarna agar mudah dibedakan. Evaluasi Prototipe halaman data pengguna dapat dilihat pada gambar 4.17 berikut.



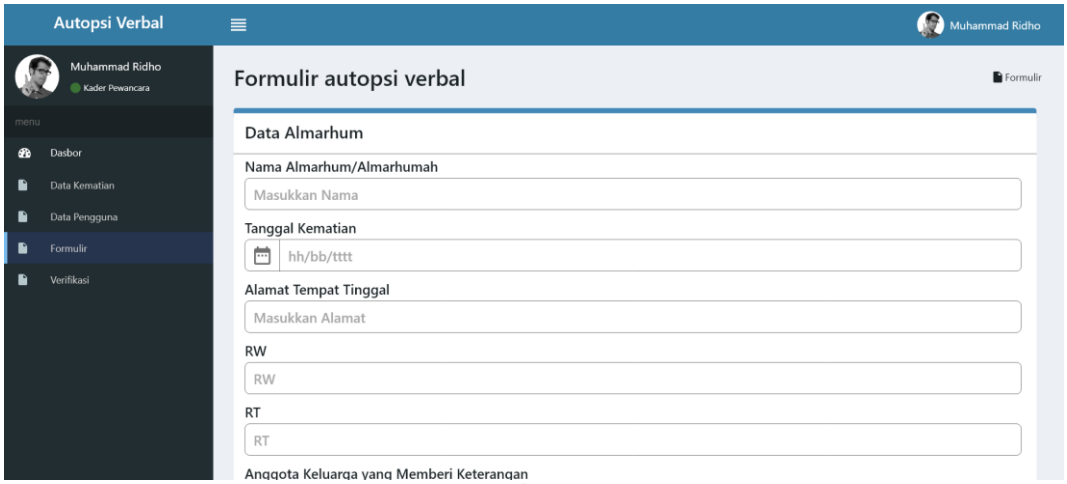
Gambar 4.17 Evaluasi Prototipe Halaman Data Pengguna

Prototipe data kematian merupakan visualisasi dari *use case* mengelola data kematian yang telah dianalisis sebelumnya. Perbaikan terhadap prototipe data kematian terletak pada aksi edit yang dihapuskan dan pemilihan aksi yang lebih berwarna agar mudah dibedakan. Evaluasi prototipe halaman data kematian dapat dilihat pada gambar 4.18 berikut.



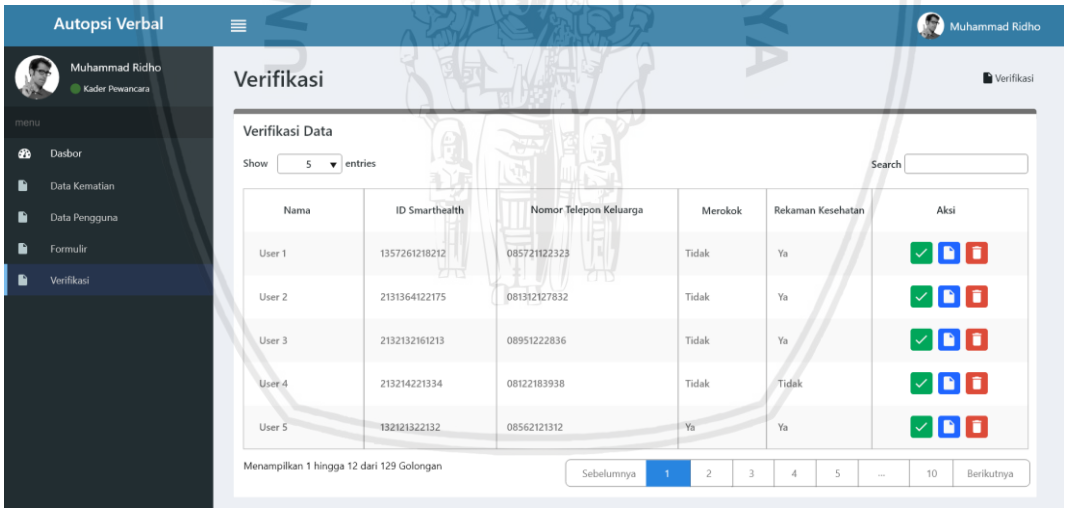
Gambar 4.18 Evaluasi Prototipe Halaman Data kematian

Prototipe formulir merupakan visualisasi dari *use case* memasukan hasil wawancara yang telah dianalisis sebelumnya. Evaluasi dilakukan agar pengguna dapat melakukan *input* formulir pada halaman formulir per kategori dan tidak merasa terlalu banyak pada tahap *input* data. Evaluasi prototipe halaman formulir dapat dilihat pada gambar 4.19 berikut.



Gambar 4.19 Evaluasi Prototipe Halaman Formulir

Prototipe verifikasi merupakan visualisasi dari *use case* melakukan verifikasi data hasil autopsi verbal yang telah dianalisis sebelumnya. Evaluasi dilakukan agar pengguna dapat melakukan verifikasi terhadap data, melihat detail data hasil autopsi verbal dan menghapus data yang tidak sesuai. Evaluasi prototipe halaman verifikasi dapat dilihat pada gambar 4.20 berikut.



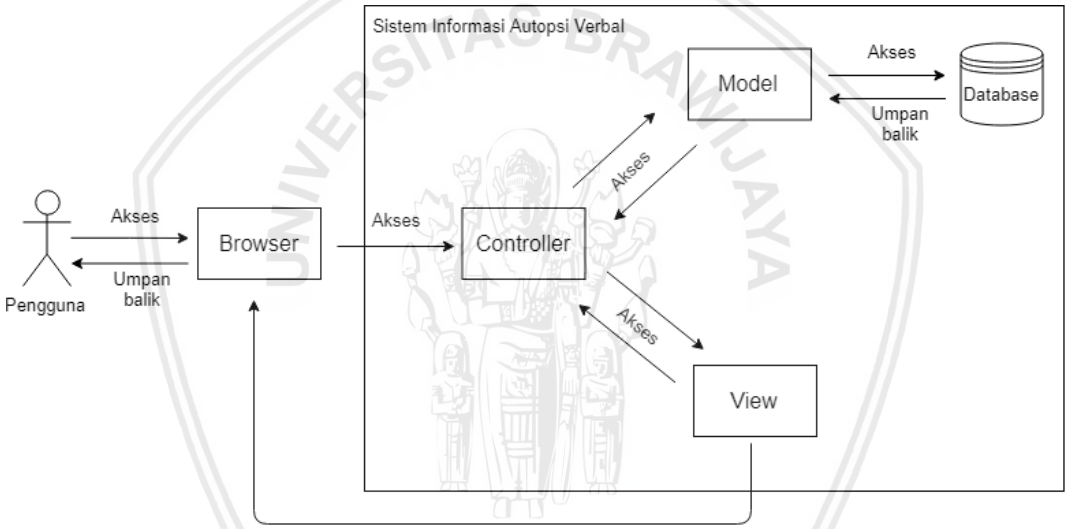
Gambar 4.20 Evaluasi Prototipe Halaman Verifikator

BAB 5 PERANCANGAN LANJUT

Pada bab ini membahas mengenai perancangan lanjutan sistem informasi yang akan dilakukan pada Sistem Informasi Autopsi Verbal Dinas Kesehatan Kabupaten Malang studi kasus wilayah Kabupaten Malang. Sistem informasi yang akan dirancang terdiri dari perancangan arsitektur sistem, pemodelan diagram kelas, perancangan *sequence diagram* dan perancangan basis data.

5.1 Perancangan Arsitektur Sistem

Perancangan arsitektur sistem dilakukan dengan menggunakan arsitektur diagram. Perancangan arsitektur sistem dilakukan untuk memberikan gambaran interaksi antar tiap komponen pada sistem informasi autopsi verbal. Perancangan arsitektur sistem informasi autopsi verbal dapat dilihat pada Gambar 5.1 sebagai berikut.



Gambar 5.1 Arsitektur Diagram Sistem Informasi Autopsi Verbal

Gambar 5.1 menunjukkan arsitektur diagram sistem informasi autopsi verbal yang menggunakan prinsip *Model-View-Controller* (MVC) yang membagi tugas menjadi tiga sesuai fungsinya masing-masing. Sistem yang akan dibangun dikembangkan dengan kerangka kerja CodeIgniter dan bahasa pemrograman PHP. Penggunaan CodeIgniter dipilih dikarenakan sudah menerapkan prinsip *Model-View-Controller* serta merupakan aplikasi sumber terbuka sehingga banyak digunakan oleh khalayak secara umum. Sistem akan bekerja sesuai dengan *trigger* yang diberikan pengguna. Pengguna yang terdiri dari kader pewawancara, staf administrator dinas kesehatan kabupaten malang, staf verifikator dinas kesehatan kabupaten malang, dan kepala dinas kesehatan kabupaten malang mengakses sistem melalui aplikasi browser yang telah terhubung dengan internet. Browser akan melakukan akses kepada *controller* kemudian memberikan perintah untuk melakukan aksi. *Controller* yang telah

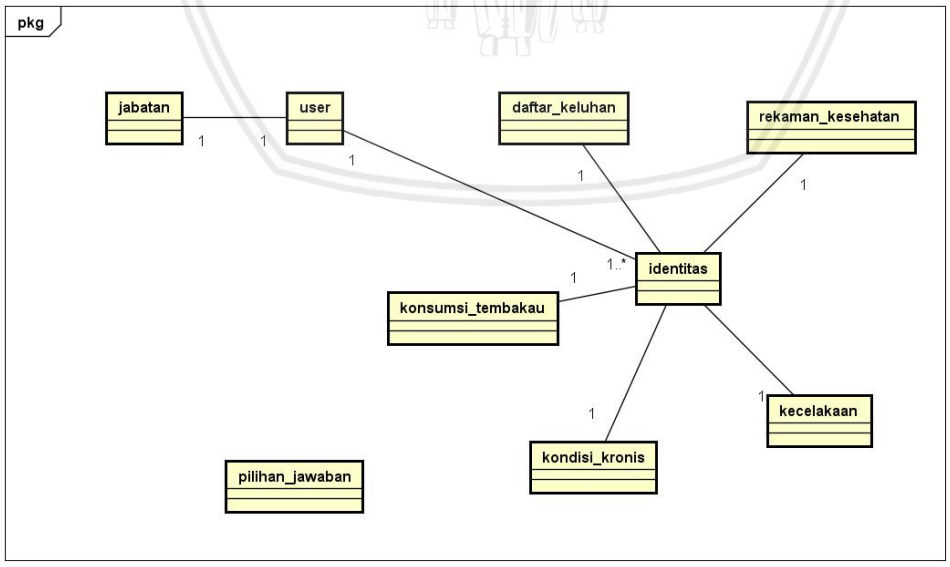
mendapat permintaan dari *browser* akan melakukan aksi yang dibutuhkan. Jika *browser* meminta data untuk ditampilkan, maka *Controller* akan memanggil fungsi yang ada pada *model* untuk mengeksekusi perintah pengambilan data dari *database. Controller* tidak dapat langsung mengakses untuk mengambil data dari *database* dan melainkan harus melalui model. Data yang telah diambil melalui *model* kemudian ditampilkan oleh *View* dengan perantara dari *Controller. View* akan memberikan tampilan yang dibutuhkan pengguna melalui browser.

5.2 Pemodelan Kelas

Pemodelan kelas adalah sekumpulan dokumentasi yang menggambarkan bagian kelas sistem informasi autopsi verbal yang dikembangkan. Pemodelan kelas digambarkan dalam satu model ke dalam diagram kelas. Diagram kelas terbagi menjadi dua diagram kelas, yaitu diagram kelas yang berisi analisis dan diagram kelas yang berisi perancangan. Diagram kelas analisis menjadi acuan untuk membuat diagram kelas objek dan perancangan basis data. Diagram kelas perancangan memberi gambaran kelas pada sistem dan menjelaskan hubungan antar kelas yang ada dalam mengembangkan sistem.

5.2.1 Diagram Kelas Analisis

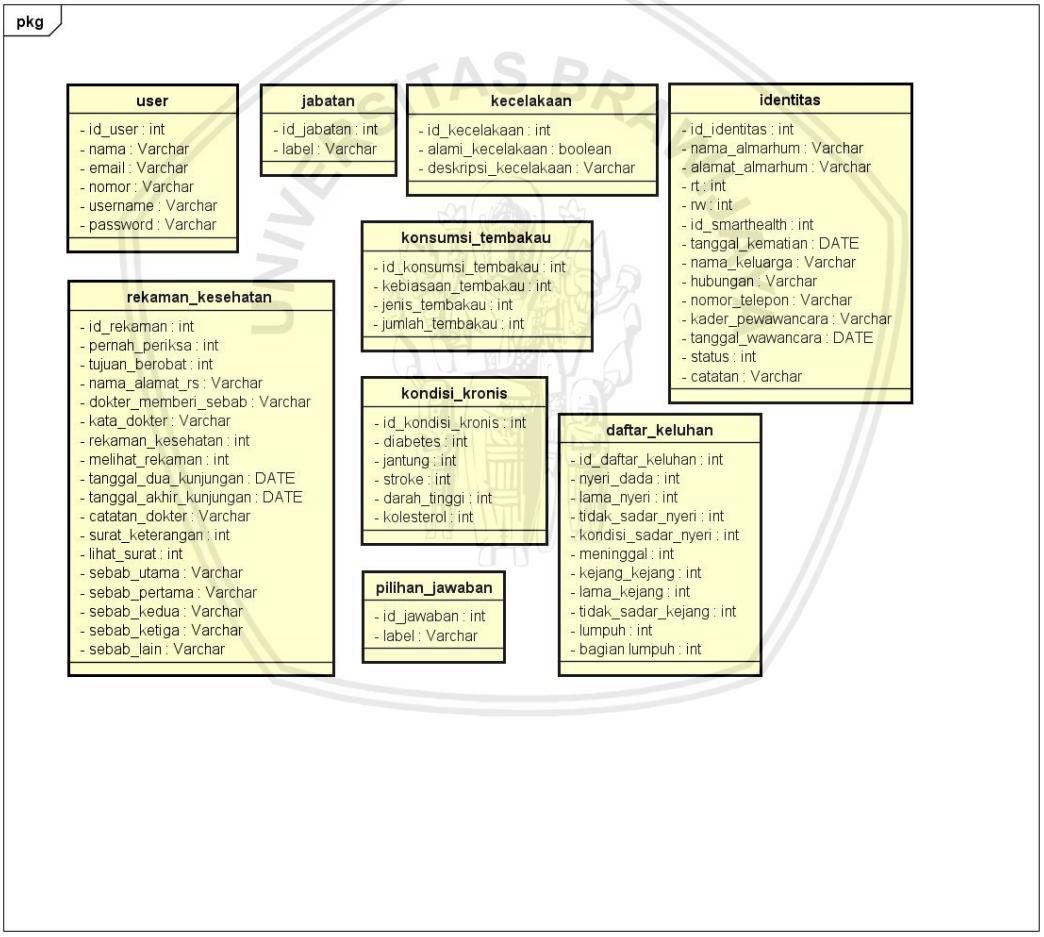
Diagram kelas analisis adalah hubungan antar setiap kelas yang merepresentasikan objek yang menjadi komponen penyusun sistem. Setiap objek terhubung yang memberikan gambaran interaksi antar data yang dibangun. Kelas ini dihasilkan dari *use case diagram* yang kemudian dilanjutkan hasil analisisnya. Diagram kelas analisis menjadi panduan dalam pemodelan perancangan basis data yang akan dibuat. Berikut pada Gambar 5.2 menggambarkan diagram kelas analisis sistem informasi autopsi verbal.



Gambar 5.2 Diagram Kelas Analisis

Diagram kelas analisis menunjukkan terdapat 9 kelas yang saling berhubungan. Diagram kelas analisis didapatkan dari penggalian pada *use case* diagram. Pengguna yang terdiri dari kader pewawancara melakukan wawancara untuk mendapatkan data hasil autopsi verbal dalam kelas diagram formulir. Staf verifikator dinas kesehatan kabupaten malang juga harus melakukan verifikasi status wawancara yang telah dilakukan. Formulir terdiri dari identitas, rekaman kesehatan, kecelakaan, kondisi_kronis, tembakau dan daftar_keluhan yang tergambar dalam satu formulir autopsi verbal.

Kelas diagram memiliki atribut yang menunjukan data pada setiap kelas. Diagram kelas atribut dibuat untuk menggambarkan kebutuhan atribut dari setiap kelas. Atribut merupakan isi dari suatu elemen dari objek. Atribut memiliki tipe data yang berbeda untuk setiap atribut. Berikut pada Gambar 5.3 Diagram Kelas Atribut Sistem Informasi Autopsi Verbal.

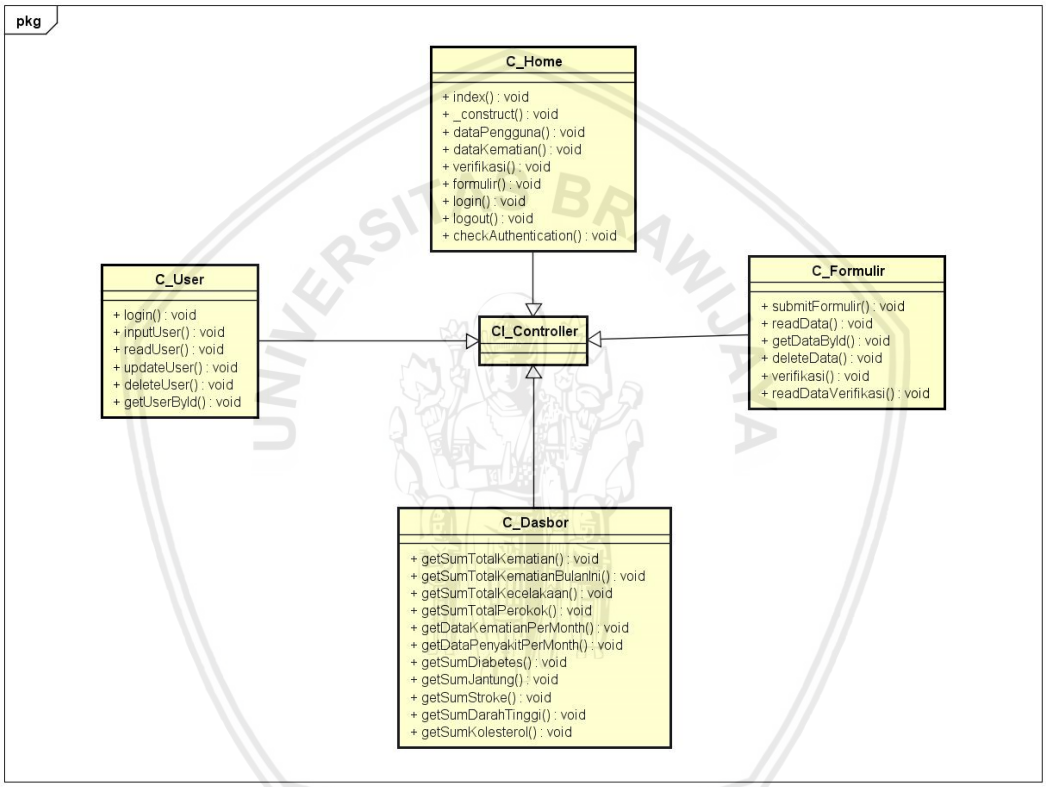


Gambar 5.3 Atribut Kelas Analisis

Pada Gambar 5.3 menunjukan terdapat 9 kelas atribut yang masing-masing kelas memiliki atributnya sendiri Diagram kelas atribut dapat dijadikan panduan untuk pembuatan rancangan basis data.

5.2.2 Diagram Kelas Perancangan

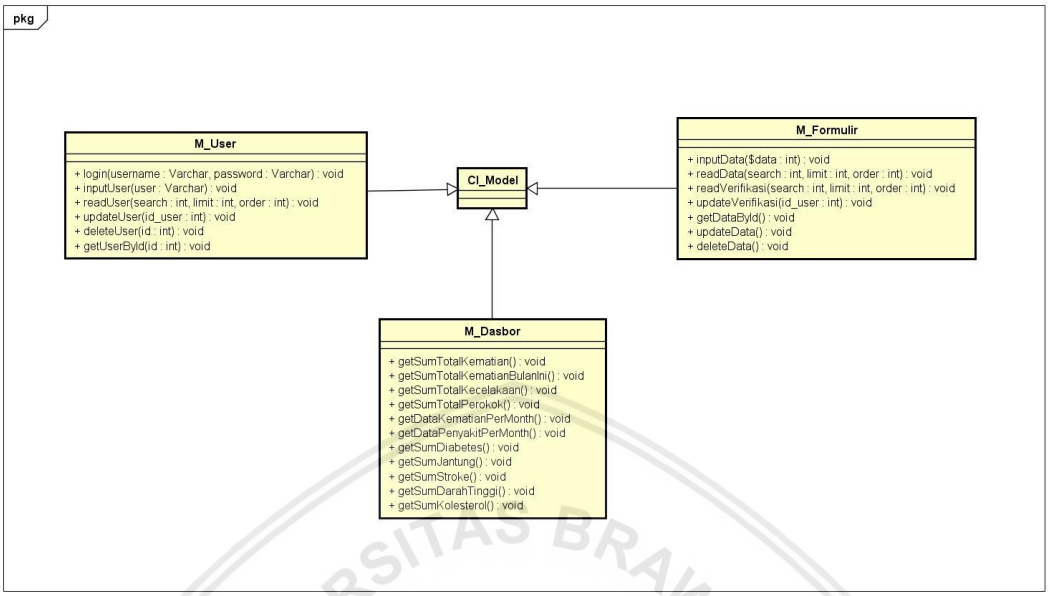
Diagram kelas perancangan merupakan pengembangan lebih lanjut dari diagram kelas analisis sebelumnya yang ditambahkan operation dalam setiap kelas. Diagram kelas perancangan dibuat sesuai dengan identifikasi yang dilakukan pada diagram kelas analisis. Diagram kelas perancangan menggambarkan kelas CI_Model dan CI_Controller serta sub yang terdapat didalamnya. Diagram kelas perancangan juga memiliki relasi secara keseluruhan antara kelas dalam sistem. Diagram kelas yang menunjukan adanya relasi antar kelas *controller* ditunjukkan pada Gambar 5.4. Diagram kelas yang menunjukan adanya relasi antar kelas *model* ditunjukkan pada Gambar 5.5.



Gambar 5.4 Diagram Kelas Perancangan untuk *Controller*

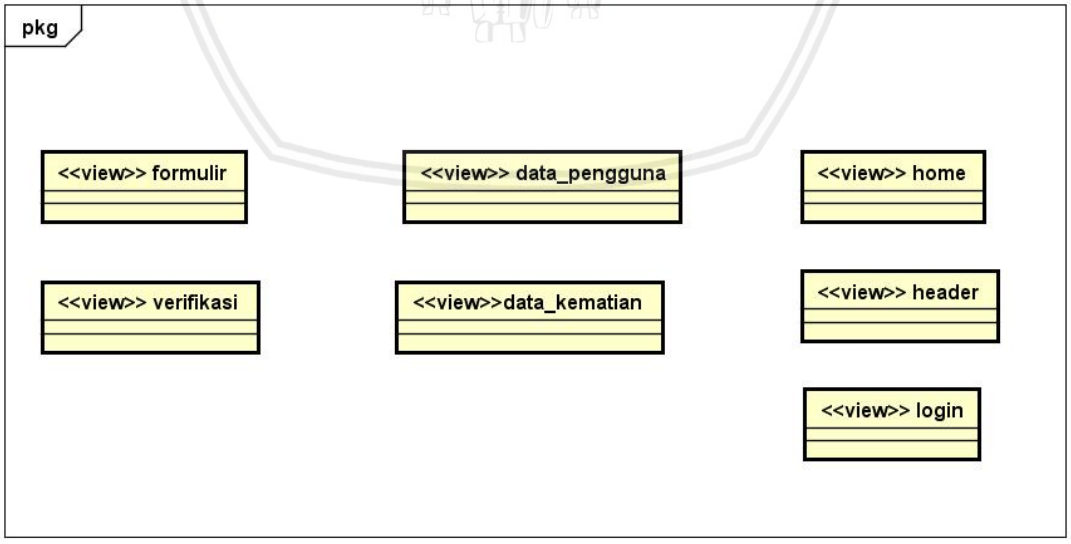
Diagram kelas perancangan *controller* memiliki 4 kelas controller yang meng *extend* pada kelas yang lebih tinggi yaitu kelas CI_Controller. CI_Controller ini memiliki 4 kelas turunan yang setiap kelasnya terdapat fungsi maupun *behavior.* Fungsi yang terdapat pada kelas perancangan controller didapatkan dari identifikasi persyaratan fungsional pada tahap analisis persyaratan yang telah diidentifikasi sebelumnya. *Controller* memberikan fungsi-fungsi terkait logika pemrograman untuk memberikan kemampuan sistem memenuhi kebutuhan pengguna. Fungsi sistem untuk melakukan pengelolaan data pengguna ditangani oleh C_User dan untuk dasbor ditangani oleh C_Dasbor. Fungsi sistem untuk melakukan pengelolaan formulir ditangani oleh C_Formulir dan untuk menangani pemrosesan halaman menggunakan C_Home. Terdapat diagram kelas

perancangan yang menggambarkan rancangan *model* seperti pada Gambar 5.5 sebagai berikut.



Gambar 5.5 Diagram Kelas Perancangan untuk *Model*

Diagram kelas *model* memiliki 3 kelas model yang meng *extend* kelas yang lebih tinggi yaitu kelas CI_Model. CI_Model ini memiliki 3 kelas turunan yang setiap kelasnya memiliki model masing-masing. *Model* menyediakan fungsi untuk melakukan akses data yang tersimpan dalam basis data. Fungsi akses data formulir autopsi verbal dilakukan oleh M_Formulir, akses data pengguna ditangani oleh M_User, dan untuk akses data dasbor ditangani oleh M_Dasbor. Diagram kelas perancangan *view* ditunjukkan pada Gambar 5.6 sebagai berikut.



Gambar 5.6 Diagram Kelas Perancangan untuk *View*

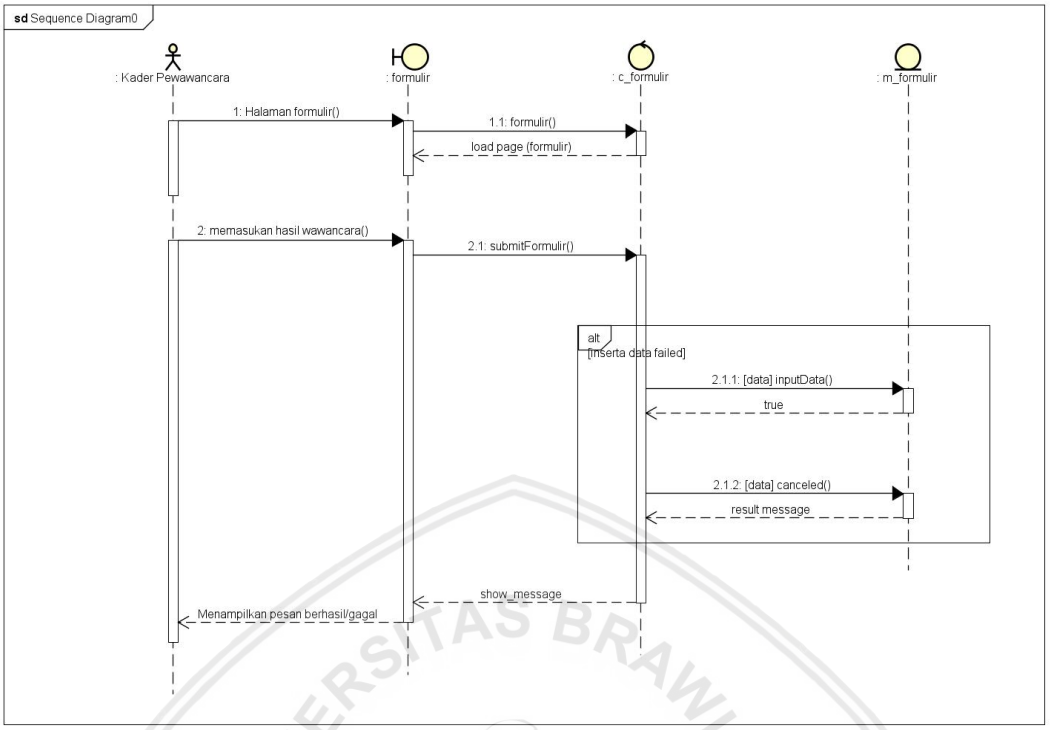
Diagram kelas perancangan *view* menggambarkan 7 *view.* Dalam diagram kelas *view* tidak mempunyai kelas turunan yang dapat diturunkan. Perancangan diagram kelas *view* ini akan dijadikan sebagai acuan untuk membuat *view.* Penamaan file akan didasarkan pada diagram perancangan view dalam mengembangkan sistem informasi menggunakan *framework* CodeIgniter dan konsep MVC. *View* merupakan tampilan antar muka yang berhubungan langsung antara pengguna dengan sistem untuk mengakses fungsi-fungsi yang tersedia. Pada *view* home dan header merepresentasikan tampilan untuk tampilan awal yaitu dasbor autopsi verbal. Pada *view login* merepresentasikan tampilan untuk masuk ke dalam sistem dengan memasukan *username* dan *password* yang sesuai. Pada *view* formulir merepresentasikan tampilan untuk mengisikan formulir id-smarthealth autopsi verbal yang akan diisikan kader pewawancara. Pada *view* verifikasi merepresentasikan tampilan untuk melakukan verifikasi terhadap data yang telah diinputkan. Pada *view* data_kematian merepresentasikan tampilan untuk mengelola data hasil autopsi verbal. Pada *view* data_pengguna merepresentasikan tampilan untuk mengelola data pengguna yang tersedia dalam sistem.

5.3 Perancangan Interaksi Antar Objek

Perancangan interaksi objek merupakan gambaran dari adanya pertukaran pesan pada sistem. Baik itu terjadi antara entitas aktor, *boundary* maupun *control,* dan objek *model*. Perancangan objek ini didapat dari interaksi yang terjadi pada *use case* yang telah diidentifikasi pada tahap analisis persyaratan. Perancangan akan dimodelkan pada diagram terstandar yaitu *sequence diagram. Sequence diagram* yang dimodelkan pada bagian ini merupakan penggambaran dari interaksi pada *use case* memasukan data hasil autopsi verbal.

5.3.1 Memasukan Data Hasil autopsi Verbal

Perancangan interaksi antar objek memasukan data autopsi verbal memberikan gambaran interaksi objek dari *use case* memasukan data hasil autopsi verbal*. Sequence diagram* memasukan data hasil autopsi verbal ditunjukan pada Gambar 5.7 berikut.

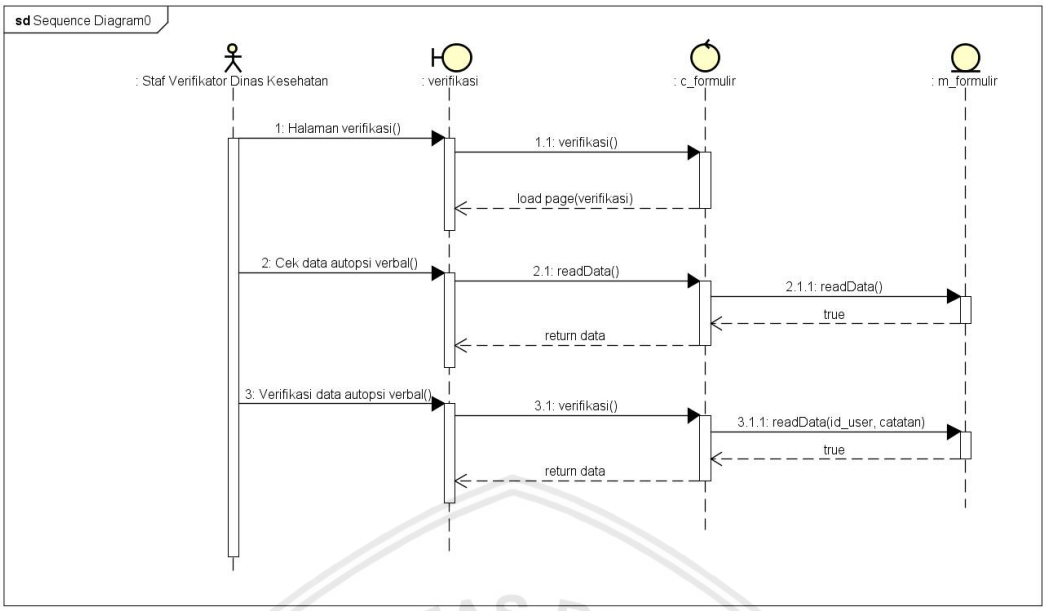


Gambar 5.7 *Sequence Diagram* Menambah Hasil autopsi Verbal

S*equence diagram* menambah data wawancara autopsi verbal dimulai ketika kader pewawancara menambah data hasil wawancara yang telah dilakukan. Proses dimulai dari kader pewawancara mengisi tampilan formulir penambahan data hasil autopsi verbal. Data yang telah diisi kemudian dikirimkan menuju c_formulir dan memanggil fungsi submitFormulir. Data yang ada pada *controller* harus dikirimkan menuju *model*. Fungsi inputData pada model akan menyimpan data pada basis data autopsi verbal yang tersedia.

5.3.2 Memverifikasi Data Hasil Autopsi Verbal

Perancangan interaksi objek melakukan verifikasi data hasil autopsi verbal memberikan gambaran interaksi objek dari *use case* melakukan verifikasi data wawancara autopsi verbal. Berikut pada Gambar 5.8 ditunjukkan *sequence diagram* verifikasi data hasil autopsi verbal.

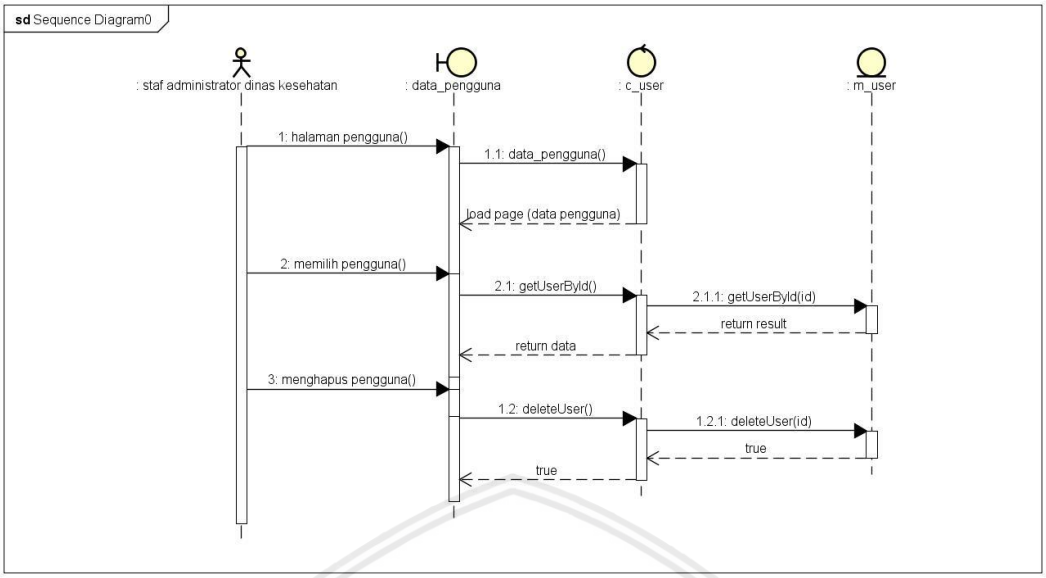


Gambar 5.8 *Sequence Diagram* Memverifikasi Data Hasil autopsi Verbal

Sequence diagram melakukan verifikasi data hasil autopsi verbal dimulai ketika staf administrator dinas kesehatan kabupaten malang melihat data autopsi verbal yagn telah dimasukan kader pewawancara. Fungsi readData pada *controller* berguna untuk menjalankan fungsi readData yang terdapat pada m_formulir. Selanjutnya data yang berasal dari *database* berhasil diambil oleh fungsi readData dan mengembalikan nilai true. Staf dinas kesehatan kabupaten malang memilih fungsi verifikasi, *controller* verifikasi akan memanggil fungsi updateVerifikasi yang terdapat pada m_formulir. Selanjutnya data verifikasi akan mengupdate data pada database untuk memverifikasi data hasil autopsi verbal.

5.3.3 Menghapus Data Pengguna

Perancangan interaksi objek menghapus data pengguna merupakan gambaran interaksi antar objek dari *use case* menghapus data pengguna. Berikut pada Gambar 5.9 ditunjukkan *sequence diagram* menghapus data pengguna.

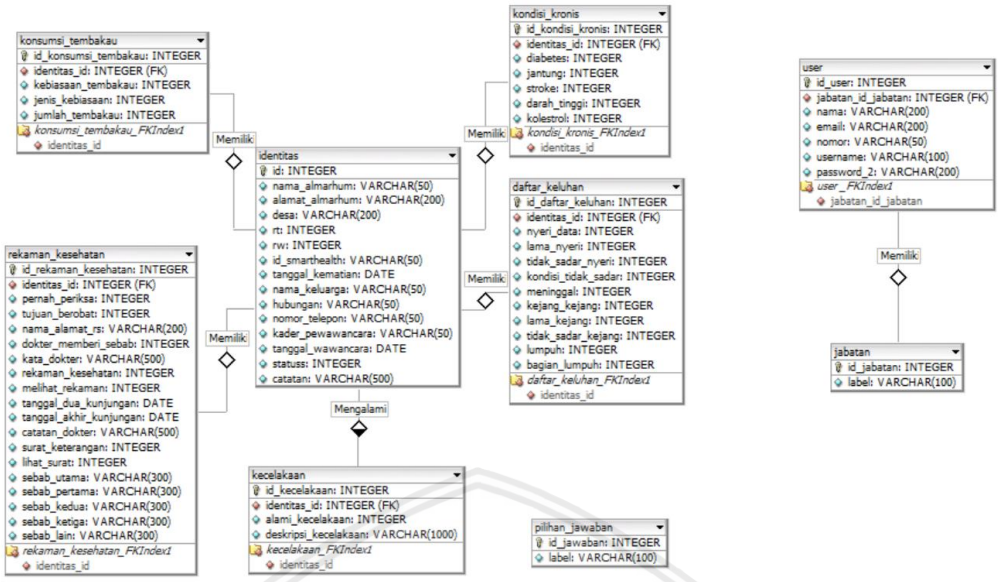


Gambar 5.9 *Sequence* Diagram Menghapus Data Pengguna

Sequence diagram menghapus data pengguna dilakukan saat staf administrator dinas kesehatan kabupaten malang ditugaskan untuk menghapus data pengguna. Pengguna membuka halaman data_pengguna untuk memuat semua data pengguna yang terdapat pada sistem. Memilih pengguna akan memanggil *controller* dengan fungsi getDataById untuk mendapatkan data salah satu pengguna sistem. Fungsi getDataById yang terdapat pada *controller* meminta data dari database melalui fungsi getDataById pada model m_user. Kemudian data yang dibutuhkan dikirimkan untuk ditampilkan melalui *contoller*. Pengguna yang dihapus akan memanggil fungsi deleteUser pada *controller*. Fungsi deleteUser kemudian dikirimkan menuju deleteUser pada m_user. deleteUser akan mengakses database untuk menjalankan hapus user sesuai id yang ingin dihapus. *Model* m_user memberikan respon eksekusi penghapusan berhasil dijalankan.

5.4 Perancangan Basis Data

Basis data yang akan dikembangkan pada sistem informasi autopsi verbal menggunakan basis data MySql. Basis data yang dirancang memiliki tujuan agar dalam pengembangan kedepannya dapat dilakukan *maintenance* dengan lebih mudah karena memiliki dokumentasi. Perancangan basis data mengacu pada perancangan kelas diagram analisis sebelumnya. *Physical Data Model* yang ada digambarkan dalam setiap relasi antar tabel yang ada di dalamnya. Berikut ini adalah perancangan basis data yang digambarkan pada *Physical Data Model* ditunjukkan pada Gambar 5.10.



Gambar 5.10 *Physical Data Model* Sistem Informasi Autopsi Verbal

BAB 6 IMPLEMENTASI

Pada bab ini akan membahas mengenai spesifikasi untuk melakukan implementasi serta hasil dari implementasi sistem informasi autopsi verbal yang dibangun. Terdapat spesifikasi lingkungan implementasi dan implementasi berupa sebagian fitur kode program, dan implementasi antarmuka pengguna yang berasal dari hasil implementasi. Implementasi dilakukan berdasarkan perancangan sistem pada tahap perancangan sehingga dapat dilakukan pelacakan terhadap fitur yang diimplementasikan.

6.1 Spesifikasi Lingkungan Implementasi

Spesifikasi lingkungan implementasi memberikan gambaran spesifikasi baik perangkat keras (*hardware*) dan perangkat lunak (*software)* yang digunakan dalam proses pembuatan sistem informasi autopsi verbal. Perangkat keras menggunakan perangkat laptop dengan spesifikasi seperti yang tertera pada Tabel 6.1. Spesifikasi perangkat lunak tertera pada Tabel 6.2.

Tabel 6.1 Spesifikasi Perangkat Keras

Unit Laptop	Acer Swift SF314-54G
CPU	Intel® Core™ i3 7020U 2,3 GHz
Kapasitas RAM yang dimiliki	8 GB
Kapasitas ROM atau Penyimpanan	1000 GB
Kartu Grafis	Nvidia GeForce MX150
Resolusi Perangkat Layar	1920 x 1080 pixels

Tabel 6.2 Spesifikasi Perangkat Lunak

Sistem Operasi	Windows 10 Home Single Language
Web Server	Apache 2.4.27
DBMS	MariaDB 10.1.26
Bahasa Pemrograman	HTML, PHP 7.1.9, CSS, Javascriptt, SQL
Editor Kode Program	Sublime Text 3
Perangkat Lunak Pendukung	XAMPP Astah UML Bizagi Modeler Microsoft Word 2016 Sortsite

6.2 Implementasi Algoritme

Implentasi algoritme terdiri dari beberapa kode program yang diimplementasikan berdasarkan pada rancangan yang dimiliki. Implementasi algoritme yang dilakukan antara lain berkaitan dengan proses melakukan memasukan data hasil autopsi verbal, mengelola data hasil autopsi verbal, mengelola data pengguna dan memonitor informasi autopsi verbal.

6.2.1 Memasukan Data Hasil Autopsi Verbal

Implementasi algoritme menambah data hasil autopsi verbal dibuat berdasarkan rancangan algoritme memasukan data hasil autopsi verbal. Implementasi algoritme ini digunakan untuk menambah data autopsi verbal. Berikut hasil implementasi algoritme memasukan data hasil autopsi verbal yang berupa kode program pada Tabel 6.3.

Tabel 6.3 Implementasi Algoritme Penambahan Data hasil autopsi Verbal

No.	Kode Program
1	`  public function submitFormulir(){`
2	`    $this->load->model("m_data");`
	`    //identitas`
3	`    $nama_almarhum = $this->input->post('nama_almarhum');`
4	`    $tanggal_kematian = $this->input->post('tanggal_kematian');`
5	`    $alamat_almarhum = $this->input->post('alamat_almarhum');`
6	`    $rw = $this->input->post('rw');`
7	`    $rt = $this->input->post('rt');`
8	`    $nama_keluarga = $this->input->post('nama_keluarga');`
9	`    $hubungan = $this->input->post('hubungan');`
10	`    $nomor_telepon = $this->input->post('nomor_telepon');`
11	`    $kader_pewawancara = $this->input->post('kader_pewawancara');`
12	`    $id_smarthealth = $this->input->post('id_smarthealth');`
13	`    $tanggal_wawancara = $this->input->post('tanggal_wawancara');`
	`    //kecelakaan`
14	`    $alami_kecelakaan = $this->input->post('alami_kecelakaan');`
15	`    $deskripsi_kecelakaan = $this->input->post('deskripsi_kecelakaan');`

```
    //kondisi kronis
    $diabetes = $this->input->post('diabetes');
16  $jantung = $this->input->post('jantung');
17  $stroke = $this->input->post('stroke');
18  $darah_tinggi = $this->input-
19  >post('darah_tinggi');
    $kolesterol = $this->input->post('kolesterol');
20  //daftar keluhan
    $nyeri_data = $this->input->post('nyeri_data');
21  $lama_nyeri = $this->input->post('lama_nyeri');
22  $tidak_sadar_nyeri = $this->input-
23  >post('tidak_sadar_nyeri');
    $kondisi_tidak_sadar = $this->input-
24  >post('kondisi_tidak_sadar');
    $meninggal = $this->input->post('meninggal');
25  $kejang_kejang = $this->input-
26  >post('kejang_kejang');
    $lama_kejang = $this->input->post('lama_kejang');
27  $tidak_sadar_kejang = $this->input-
28  >post('tidak_sadar_kejang');
    $lumpuh = $this->input->post('lumpuh');
29  $bagian_lumpuh = $this->input-
30  >post('bagian_lumpuh');
    //tembakau
    $kebiasaan_tembakau = $this->input-
31  >post('kebiasaan_tembakau');
    $jenis_tembakau = $this->input-
32  >post('jenis_tembakau');
    $jumlah_tembakau = $this->input-
33  >post('jumlah_tembakau');
    //rekaman kesehatan
    $pernah_periksa = $this->input-
34  >post('pernah_periksa');
    $tujuan_berobat = $this->input-
35  >post('tujuan_berobat');
    $nama_alamat_rs = $this->input-
36  >post('nama_alamat_rs');
    $dokter_memberi_sebab = $this->input-
37  >post('dokter_memberi_sebab');
    $kata_dokter = $this->input->post('kata_dokter');
    $rekaman_kesehatan = $this->input-
38  >post('rekaman_kesehatan');
39  $melihat_rekaman = $this->input-
```

```
>post('melihat_rekaman');
40      $tanggal_dua_kunjungan       =       $this->input-
>post('tanggal_dua_kunjungan');
41      $tanggal_akhir_kunjungan      =      $this->input-
>post('tanggal_akhir_kunjungan');
42      $catatan_dokter           =           $this->input-
>post('catatan_dokter');
43      $surat_keterangan          =          $this->input-
>post('surat_keterangan');
44      $lihat_surat = $this->input->post('lihat_surat');
        $sebab_utama = $this->input->post('sebab_utama');
45      $sebab_pertama            =           $this->input-
46  >post('sebab_pertama');
47      $sebab_kedua = $this->input->post('sebab_kedua');
        $sebab_ketiga             =           $this->input-
48  >post('sebab_ketiga');
49      $sebab_lain = $this->input->post('sebab_lain');


50
        $data = array(
          'nama_almarhum' => $nama_almarhum,
          'tanggal_kematian' => $tanggal_kematian,
51
          'alamat_almarhum' => $alamat_almarhum,
52
          'rw' => $rw,
53
          'rt' => $rt,
54
          'nama_keluarga' => $nama_keluarga,
55
          'hubungan' => $hubungan,
56
          'nomor_telepon' => $nomor_telepon,
57
          'kader_pewawancara' => $kader_pewawancara,
58
          'id_smarthealth' => $id_smarthealth,
59
          'alami_kecelakaan' => $alami_kecelakaan,
60
          'deskripsi_kecelakaan' => $deskripsi_kecelakaan,
61
          'tanggal_wawancara' => $tanggal_wawancara,
62
          'diabetes' => $diabetes,
63
          'jantung' => $jantung,
          'stroke' => $stroke,
64
          'darah_tinggi' => $darah_tinggi,
65
          'kolesterol' => $kolesterol,
66
          'nyeri_data' => $nyeri_data,
67
          'lama_nyeri' => $lama_nyeri,
68
          'tidak_sadar_nyeri' => $tidak_sadar_nyeri,
69
          'kondisi_tidak_sadar' => $kondisi_tidak_sadar,
70
          'meninggal' => $meninggal,
```


repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

```
71          'kejang_kejang' => $kejang_kejang,
72          'lama_kejang' => $lama_kejang,
73          'tidak_sadar_kejang' => $tidak_sadar_kejang,
74          'lumpuh' => $lumpuh,
75          'bagian_lumpuh' => $bagian_lumpuh,
76          'kebiasaan_tembakau' => $kebiasaan_tembakau,
77          'jenis_tembakau' => $jenis_tembakau,
78          'jumlah_tembakau' => $jumlah_tembakau,
79          'pernah_periksa' => $pernah_periksa,
80          'tujuan_berobat' => $tujuan_berobat,
81          'nama_alamat_rs' => $nama_alamat_rs,
82          'dokter_memberi_sebab' => $dokter_memberi_sebab,
83          'kata_dokter' => $kata_dokter,
84          'rekaman_kesehatan' => $rekaman_kesehatan,
85          'melihat_rekaman' => $melihat_rekaman,
86          'tanggal_dua_kunjungan'                          =>
87  $tanggal_dua_kunjungan,
88          'tanggal_akhir_kunjungan'                        =>
89  $tanggal_akhir_kunjungan,
            'catatan_dokter' => $catatan_dokter,
90          'surat_keterangan' => $surat_keterangan,
            'lihat_surat' => $lihat_surat,
91          'sebab_utama' => $sebab_utama,
92          'sebab_pertama' => $sebab_pertama,
93          'sebab_kedua' => $sebab_kedua,
94          'sebab_ketiga' => $sebab_ketiga,
95          'sebab_lain' => $sebab_lain
96      );
97
98      $this->m_data->input_data($data);
99      redirect("/formulir");
      }
100
```

6.2.2 Mengelola Data Hasil Autopsi Verbal

Implementasi algoritme mengelola data hasil autopsi verbal yang dibuat berdasar pada rancangan algoritme mengelola data hasil autopsi verbal. Implementasi ini digunakan oleh sistem dalam mengelola data hasil autopsi verbal berupa baca (*read),* dan hapus (*delete)*. Berikut hasil implementasi algoritme mengelola data hasil autopsi verbal yang berupa kode program pada Tabel 6.4.

Tabel 6.4 Implementasi Algoritme Mengelola Data Hasil Autopsi Verbal

No.	Kode Program
1	`Public function readData(){`
2	`$this->load->model("m_data");`
3	`$search = $_POST['search']['value'];`
4	`$limit = $_POST['length'];`
5	`$start = $_POST['start'];`
6	`$order_index = $_POST['order'][0]['column'];`
7	`$order_field= $_POST['columns'][$order_index]['data'];`
8	`$order_ascdesc = $_POST['order'][0]['dir'];`
9	`$sql_data = $this->m_data->read_data($search, $limit, $start, $order_field, $order_ascdesc);`
10	`$sql_total = $this->m_data->count_all();`
11	`$sql_filter = $this->m_data-> count_filter ($search);`
12	`$callback = array(`
13	`'draw'=>$_POST['draw'],`
14	`'recordsTotal'=>$sql_total,`
15	`'recordsFiltered'=>$sql_filter,`
16	`'data'=>$sql_data`
17	`);`
18	`header('Content-Type: application/json');`
19	`echo json_encode($callback); // Convert array $callback ke json`
20	`}`
21	`public function deleteData(){`
22	`$this->load->model("m_data");`
23	`$id = $_POST['delete_id_data'];`
24	`$this->m_data->deleteData($id);`
25	`redirect("/data-kematian");`
26	`}`
27	
28	

6.2.3 Mengelola Data Pengguna

Implementasi algoritme mengelola data pengguna mengacu pada rancangan algoritme mengelola data pengguna. Implementasi algoritma ini digunakan untuk mengelola data pengguna berupa buat (*create*), baca (*read),* ubah (*edit*), hapus (*delete)*. Berikut hasil implementasi algoritme mengelola data pengguna yang berupa kode program pada Tabel 6.5.

Tabel 6.5 Implementasi Algoritme Mengelola Data Pengguna

No.	Kode Program
1	`public function inputUser(){`
2	`$this->load->model("m_data");`
3	`$nama_user = $this->input->post('nama_user');`
4	`$email = $this->input->post('email');`
5	`$nomor_user = $this->input->post('nomor_user');`
6	`$username = $this->input->post('username');`
7	`$password = $this->input->post('password');`
8	`$jabatan = $this->input->post('jabatan');`
9	`$user = array(`
10	`'nama_user' => $nama_user,`
11	`'email' => $email,`
12	`'nomor_user' => $nomor_user,`
13	`'username' => $username,`
14	`'password' => $password,`
15	`'jabatan' => $jabatan`
16	`);`
17	`$this->m_data->inputUser($user);`
18	`redirect("/data-pengguna");`
19	`}`
20	`public function readUser(){`
21	`$this->load->model("m_data");`
22	`$search = $_POST['search']['value'];`
23	`$limit = $_POST['length'];`
24	`$start = $_POST['start']; // Ambil data start`
25	`$order_index = $_POST['order'][0]['column'];`
26	`$order_field =`
27	`$_POST['columns'][$order_index]['data'];`
28	`$order_ascdesc = $_POST['order'][0]['dir']; // penentukan order by ascending atau descending`
29	
30	`$sql_total = $this->m_data->count_all(); // Panggil fungsi count_all pada SiswaModel`
31	`$sql_data = $this->m_data->read_user($search, $limit, $start, $order_field, $order_ascdesc); // Panggil fungsi filter pada SiswaModel`

```
32      $sql_filter = $this->m_data->count_filter_user($search);    //    Panggil    fungsi
33  count_filter pada SiswaModel
34      // $sql_filter = 0;
35      $callback = array(
36          'draw'=>$_POST['draw'],    //    Ini    dari
    datatablenya
            'recordsTotal'=>$sql_total,
37          'recordsFiltered'=>$sql_filter,
            'data'=>$sql_data
        );
38
      header('Content-Type: application/json');
39       echo json_encode($callback); // Convert array
40  $callback ke json
41
42    }

43    public function update_user(){
44      $this->load->model("m_data");
      $id_user = $_POST['edit_id_user'];
45      $this->m_data->update_user($id_user); // Panggil
46  fungsi edit()
47      redirect("/data-pengguna");
48    }
49
50    public function get_pengguna_by_id(){
51      $this->load->model("m_data");
52      $id = $_POST['id'];
53      $result = $this->m_data->get_pengguna_by_id($id);
54      echo json_encode($result);
55    }
56    public function deleteUser(){
57      $this->load->model("m_data");
58      $id_user = $_POST['delete_id_user'];
59      $this->m_data->delete_user($id_user); // Panggil
    fungsi edit()
60      redirect("/data-pengguna");
      }
```

6.2.4 Memonitor Informasi Autopsi Verbal

Implementasi algoritme memonitor informasi autopsi verbal dibuat berdasarkan rancangan algoritme memonitor informasi autopsi verbal. Implementasi algoritme ini merupakan fungsi sistem untuk melihat informasi hasil autopsi verbal. Berikut hasil implementasi algoritme memonitor informasi autopsi verbal pada Tabel 6.7.

Tabel 6.6 Implementasi Memonitor Informasi Autopsi Verbal

No.	Kode Program
1	`public function getSumTotalKematian(){`
2	`$this->load->model("m_data");`
3	`$result = $this->m_data->getSumTotalKematian();`
4	`echo json_encode($result);`
5	`}`
6	`public function getSumTotalKematianBulanIni(){`
7	`$this->load->model("m_data");`
8	`$result = $this->m_data->getSumTotalKematianBulanIni();`
9	`echo json_encode($result);`
10	`}`
11	`public function getSumTotalKecelakaan(){`
12	`$this->load->model("m_data");`
13	`$result = $this->m_data->getSumTotalKecelakaan();`
14	`echo json_encode($result);`
15	`}`
16	`public function getSumTotalPerokok(){`
17	`$this->load->model("m_data");`
18	`$result = $this->m_data->getSumTotalPerokok();`
19	`echo json_encode($result);`
20	`}`
21	`public function getDataKematianPerMonth(){`
22	`$this->load->model('m_data');`
23	`$result = $this->m_data->getDataKematianPerMonth();`
24	`echo json_encode($result);`
25	`}`

```
26    public function getDataPenyakitPerMonth() {
27          $this->load->model('m_data');
28          $result            =            $this->m_data-
>getDataPenyakitPerMonth();
29          echo json_encode($result);
30    }

31    public function getSumDiabetes(){
32        $this->load->model("m_data");
33        $result = $this->m_data->getSumDiabetes();
34        echo json_encode($result);
35    }

36    public function getSumJantung(){
37        $this->load->model("m_data");
38        $result = $this->m_data->getSumJantung();
39        echo json_encode($result);
40    }

41    public function getSumStroke(){
          $this->load->model("m_data");
42        $result = $this->m_data->getSumStroke();
          echo json_encode($result);
43    }

44    public function getSumDarahTinggi(){
45        $this->load->model("m_data");
46        $result = $this->m_data->getSumDarahTinggi();
47        echo json_encode($result);
48    }

49    public function getSumKolesterol(){
50          $this->load->model('m_data');
51          $result = $this->m_data->getSumKolesterol();
52          echo json_encode($result);
53    }
```

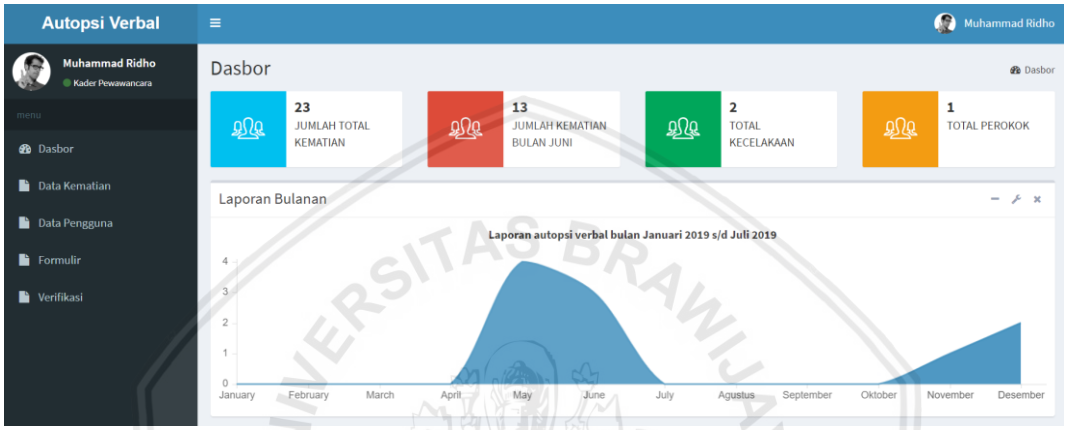
6.3 Implementasi Antarmuka Pengguna

Implementasi antarmuka pengguna merupakan hasil dari implementasi yang dibuat sebelumnya. Sistem informasi autopsi verbal menghasilkan tampilan

antarmuka pengguna dasbor, formulir, verifikasi, data pengguna dan data kematian.

6.3.1 Antarmuka Dasbor

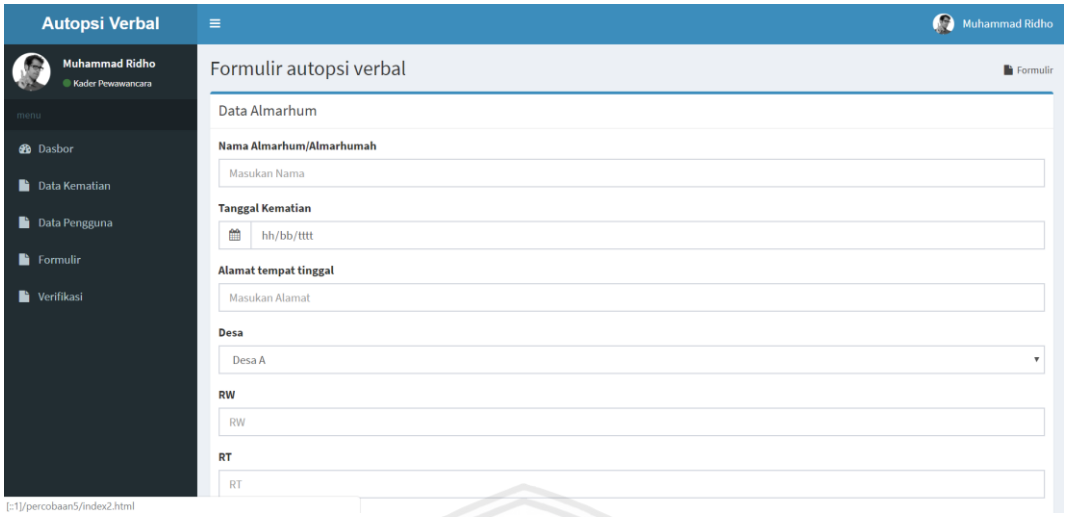
Antarmuka dasbor merupakan tampilan pawal ketika pengguna berhasil masuk ke dalam sistem. Menu *sidebar* dan pada bagian konten terdiri dari data kematian, data pengguna, formulir, verifikasi. Berikut pada Gambar 6.1 hasil antarmuka pengguna dasbor.



Gambar 6.1 Implementasi Antarmuka *Dashboard*

6.3.2 Antarmuka Formulir

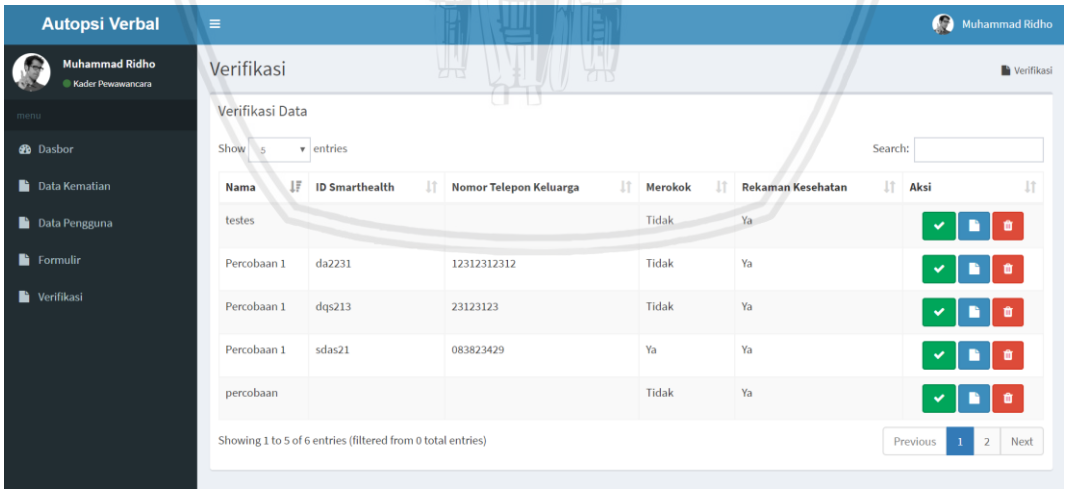
Antarmuka formulir merupakan tampilan yang dilihat kader pewawancara saat akan mengisi data hasil autopsi verbal ke dalam sistem. Terdapat daftar pertanyaan yang termasuk formulir standar id smarthealth yang dapat diiskan kader pewawancara. Kategori pertanyaan dibagi dalam 5 kategori yaitu kategori identitas, kecelakaan, kondisi kronis, daftar keluhan, konsumsi tembakau dan kondisi kronis. Terdapat tombol next untuk melanjutkan pertanyaan pada kategori selanjutnya dan tombol prev untuk memundurkan formulir pada pertanyaan sebelumnya serta tombol submit untuk memasukan data yang telah diisi ke dalam database. Berikut pada Gambar 6.2 hasil antarmuka pengguna formulir.



Gambar 6.2 Implementasi Antarmuka Formulir Autopsi Verbal

6.3.3 Antarmuka Verifikasi Data hasil autopsi Verbal

Antarmuka verifikasi data hasil autopsi verbal merupakan tampilan yang dilihat pengguna ketika memilih untuk melakukan verifikasi data kematian. Pengguna yang dapat melihat data ini adalah staf verifikator dinas kesehatan kabupaten malang. Terdapat tabel yang menampilkan nama, id smarthealth, nomor tekepon keluarga, keterangan merokok dan rekaman kesehatan. Terdapat juga tombol aksi berwarna hijau untuk verifikasi data hasil autopsi verbal, tombol berwarna biru untuk melihat detail data, serta tombol hapus berwarna merah untuk megnhapus data. Berikut pada Gambar 6.3 hasil antarmuka pengguna verifikasi.

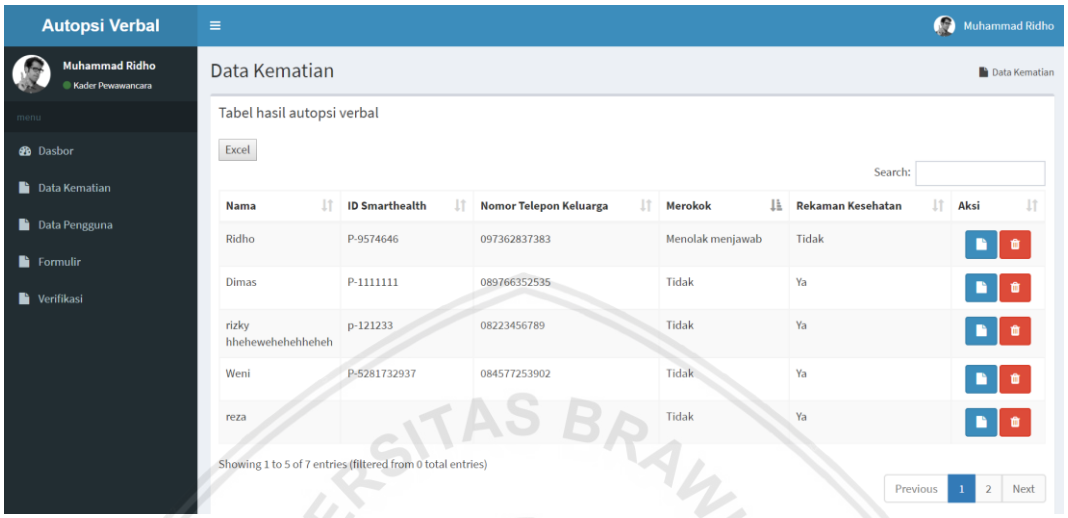


Gambar 6.3 Implementasi Antarmuka Data Kematian

6.3.4 Antarmuka Mengelola Data hasil autopsi Verbal

Antarmuka data kematian merupakan tampilan yang dilihat pengguna ketika memilih data kematian. Pengguna yang dapat melihat data ini adalah staf dinas

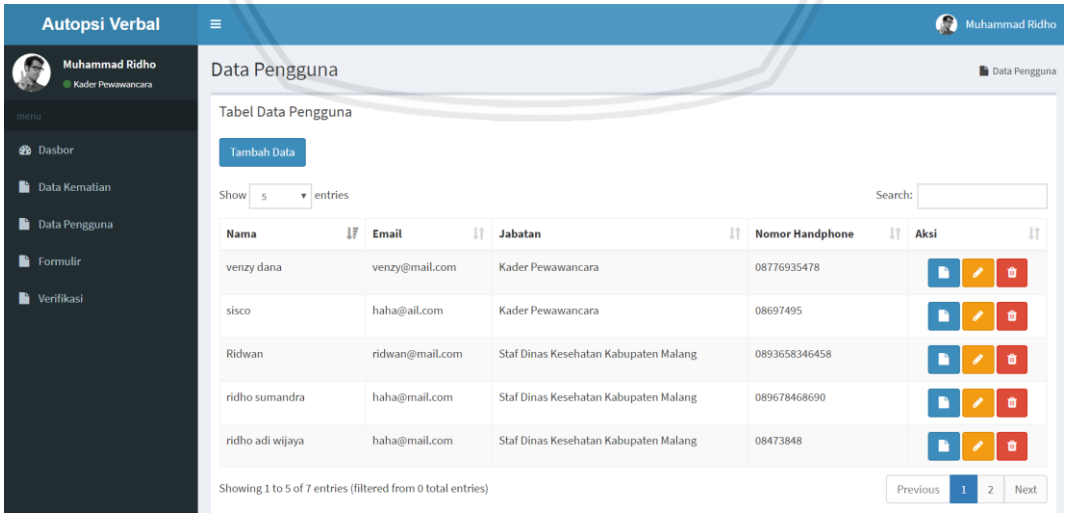
kesehatan kabupaten malang dan kader pewawancara. Terdapat tabel yang menampilkan nama, id smarthealth, nomor tekepon keluarga, keterangan merokok dan rekaman kesehatan. Terdapat juga tombol aksi berwarna biru untuk melihat detail data, serta tombol hapus berwarna merah untuk megnhapus data. Berikut pada Gambar 6.4 hasil antarmuka data kematian.



Gambar 6.4 Implementasi Antarmuka Data Kematian

6.3.5 Antarmuka Mengelola Data Pengguna

Antarmuka data pengguna merupakan tampilan yang dilihat pengguna ketika memilih data pengguna. Pengguna yang dapat melihat data ini adalah staf dinas kesehatan kabupaten malang. Terdapat tabel yang menampilkan nama, email jabatan, nomor handphone. Terdapat juga tombol aksi berwarna biru untuk melihat detail data, tombol edit berwarna oranye untuk melakukan pengguna, serta tombol hapus berwarna merah untuk menghapus data. Berikut pada Gambar 6.5 hasil antarmuka data pengguna.



Gambar 6.5 Implementasi Antarmuka Data Pengguna

BAB 7 PENGUJIAN

Pada bab ini akan membahas tentang proses pengujian yang dilakukan pada sistem informasi autopsi verbal. Pengujian bertujuan untuk mengetahui sistem informasi dapat berjalan dengan baik dan sesuai dengan persyaratan yang telah didefinisikan. Terdapat dua jenis pengujian yaitu pengujian validasi dan pengujian kompatibilitas sistem.

7.1 Pengujian Validasi

Pengujian validasi merupakan pengujian yang dilakukan untuk memastikan sistem yang dibangun dapat berjalan sesuai dengan analisis persyaratan yang sebelumnya telah didefinisikan. Pengujian yang dilakukan berisi hasil pengujian validasi terhadap fitur yang akan digunakan pengguna nantinya. Pengujian validasi yang diujikan akan berdasar pada skenario *use case* mencatat data hasil autopsi verbal, mengelola data hasil autopsi verbal dan mengelola data pengguna.

7.1.1 Pengujian Validasi Memasukan Data Hasil Autopsi Verbal

Pengujian dan uji validasi digunakan untuk memastikan *use case* pengguna dapat menggunakan sistem untuk mencatat data hasil autopsi verbal dengan kode persyaratan SIAV-F-05. Rencana pengujian dan kasus uji yang akan dilakukan dapat dilihat pada Tabel 7.1

Tabel 7.1 Pengujian dan Kasus Uji Mencatat Data Hasil Autopsi Verbal

Kode Pengujian	VT-SIAV-01
Nomor Skenario	Skenario 1
Kode Persyaratan	SIAV-F-05
Tujuan Pengujian	Pengujian dapat dilakukan penguji dengan memasukan data hasil autopsi verbal ke dalam sistem dan tersimpan dalam *database.*
Prosedur Uji	1. Penguji memiliki identifikasi sebagai kader pewawancara. 2. Penguji memilih mengakses menu formulir. 3. Penguji melakukan pengisian formulir autopsi verbal. 4. Penguji memilih tombol submit
Hasil yang diharapkan	Sistem memberikan notifikasi pada pengguna bahwa data berhasil tersimpan dan data yang dimasukan tersimpan dalam *database*.
Hasil Pengujian	Valid

Masukkan dan saran	Data berhasil dimasukan dan tersimpan dalam *database*.

7.1.2 Pengujian Validasi Memverifikasi Data Hasil Autopsi Verbal

Pengujian dan uji validasi digunakan untuk memastikan *use case* pengguna dapat menggunakan sistem untuk melakukan verifikasi hasil autopsi verbal dengan kode persyaratan SIAV-F-08. Rencana pengujian dan kasus uji yang akan dilakukan dapat dilihat pada Tabel 7.2

Tabel 7.2 Pengujian dan Kasus Uji Memverifikasi Data Hasil Autopsi Verbal

Kode Pengujian	VT-SIAV-02
Nomor Skenario	Skenario 2
Kode Persyaratan	SIAV-F-08
Tujuan Pengujian	Pengujian dapat dilakukan penguji untuk melakukan verifikasi data hasil autopsi verbal sehingga berubah menjadi data yang telah terverifikasi.
Prosedur Uji	1. Penguji memilki identifikasi sebagai staf verifikator dinas kesehatan kabupaten malang. 2. Penguji memilih mengakses menu verifikasi. 3. Penguji melihat data hasil autopsi verbal yang ingin diverifikasi. 4. Penguji melihat detail data hasil autopsi verbal 5. Penguji melakukan klik ikon verifikasi pada kolom aksi 6. Penguji melakukan klik tombol verifikasi
Hasil yang diharapkan	Data tidak lagi muncul pada tabel verifikasi dan telah berpindah menjadi data kematian.
Hasil Pengujian	Valid
Masukkan dan saran	Data berhasil diverifikasi oleh verifikator.

7.1.3 Pengujian Validasi Melihat Detail Data hasil autopsi Verbal

Pengujian dan uji validasi digunakan untuk memastikan *use case* pengguna dapat menggunakan sistem untuk menampilkan detail data hasil autopsi verbal yang terverifikasi dengan kode persyaratan SIAV-F-13. Rencana pengujian dan kasus uji yang akan dilakukan dapat dilihat pada Tabel 7.3

Tabel 7.3 Pengujian dan Kasus Uji Melihat Detail Data hasil autopsi Verbal

Kode Pengujian	VT-SIAV-03
Nomor Skenario	Skenario 3
Kode Persyaratan	SIAV-F-13
Tujuan Pengujian	Pengujian dapat dilakukan untuk memastikan penguji bisa melihat detail data hasil autopsi verbal setiap almarhum/almarhumah.
Prosedur Uji	1. Penguji memiliki identifikasi sebagai staf verifikator dinas kesehatan kabupaten malang. 2. Penguji memilih menu data kematian. 3. Penguji memilih data hasil autopsi verbal yang ingin dilihat detailnya. 4. Penguji melakukan klik ikon detail pada kolom aksi. 5. Penguji melakukan klik tombol detail.
Hasil yang diharapkan	Muncul detail data hasil autopsi verbal secara lengkap.
Hasil Pengujian	Valid
Masukkan dan saran	Dapat melihat detail data.

7.1.4 Pengujian Validasi Memonitor Hasil Autopsi Verbal

Pengujian dan uji validasi digunakan untuk memastikan *use case* pengguna dapat melihat tampilan informasi hasil autopsi verbal dengan kode persyaratan SIAV-F-28. Rencana pengujian dan kasus uji yang akan dilakukan dapat dilihat pada Tabel 7.4

Tabel 7.4 Pengujian dan Kasus Uji Melihat Detail Data Autopsi Verbal

Kode Pengujian	VT-SIAV-04
Nomor Skenario	Skenario 4
Kode Persyaratan	SIAV-F-28
Tujuan Pengujian	Pengujian dapat dilakukan untuk memastikan penguji dapat melihat informasi umum hasil autopsi verbal.
Prosedur Uji	1. Penguji teridentifikasi telah terdaftar dalam sistem. 2. Penguji memilih menu dasbor. 3. Penguji melihat informasi autopsi verbal secara umum.
Hasil yang diharapkan	Muncul informasi dan grafik hasil autopsi verbal.

Hasil Pengujian	Valid
Masukkan dan saran	Dashbor menampilkan data kematian.

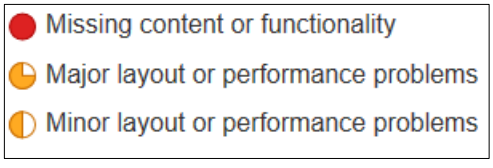
7.2 Pengujian Kompatibilitas Peramban

Pengujian kompatibilitas peramban dilakukan untuk mengetahui antarmuka pengguna pada perangkat maupun ekosistem yang berbeda. Pengujian dilakukan untuk memenuhi analisis persyaratan nonfungsional dengan kode SIAF-NF-01. Pengujian dilakukan dengan menggunakan bantuan perangkat lunak Sortsite dilakukan untuk mengetahui peforma sistem apabila sistem dijalankan pada peramban yang berbeda. Pengujian ini dilakukan untuk memenuhi persyaratan non fungsional PNFSIP-01. Untuk melakukan pengujian ini, digunakan perangkat lunak Sortsite versi 5.33.842.0. Peramban yang digunakan beserta versinya pada pengujian kompatibilitas peramban ini ditunjukkan pada Tabel 7.5 sebagai berikut.

Tabel 7.5 Jenis Peramban yang Digunakan pada Pengujian kompatibilitas peramban

Nama Peramban
Internet Explorer versi 11
Microsoft Edge versi 18
Firefox versi 66
Safari versi 12
Opera versi 60
Chrome versi 74
iOS versi <10 11 12
Android <3 4

Pengujian kompatibilitas dengan menggunakan Sortsite memiliki 3 kategori masalah yang terdapat pada sistem yaitu criticall issues berupa hilangnya konten maupun fungsi, *major issues* berupa hilangnya garis besar *layout* dan masalah performa serta *minor issues* yaitu hilangnya *layout* secara minor dan masalah performa. seperti pada Gambar 7.1 sebagai berikut.



Gambar 7.1 Kategori *Compatibility Issues* pada SortSite

Pengujian kompatibilitas ditunjukan pada gambar 7.2 yang menggambarkan pengujian sistem informasi autopsi verbal pada saat digunakan pada beberapa peramban yang berbeda.

Gambar 7.2 Hasil Pengujian kompatibilitas peramban

Hasil pengujian kompatibilitas peramban dengan perangkat lunak Sortsite menghasilkan 9% *error issues*. *Error issues* ini dikarenakan sistem yang dibangun memiliki masalah *critical issues* pada peramban *Firefox* versi 66 dan peramban *Chrome* versi 74. Hal ini terjadi lantaran pada halaman *login* memberikan peringatan keamanan karena tidak menggunakan HTTPS untuk keamanan data sandi. Sistem informasi yang dikembangkan masih menggunakan *Hypertext Transfer Protocol* (HTTP) yang belum memiliki keamanan (*security)* berupa HTTPS. Terdapat *major issues* pada jenis peramban *Internet Explorer (IE)* versi 11 dan jenis peramban *Safari* versi 12 dikarenakan terdapat CSS filter yang tidak lagi didukung oleh peramban tersebut. Selain itu, terdapat *minor issues* pada peramban Internet Explorer versi 11. Hal ini disebabkan karena properti yang dimiliki CSS tidak memiliki dukungan pada peramban diatas sehingga menimbulkan *minor issues* berupa hilangnya sebagian kecil tampilan yang didukung CSS. Jenis peramban *Microsoft Edge, Opera, iOS* dan *Android* tidak mengalami kendala dalam menjalankan sistem informasi autopsi verbal.

7.3 Analisis Hasil Pengujian

Pengujian validasi yang telah dilakukan perlu dihitung presentase keberhasilannya. Persentase validitas pengujian dapat dihitung menggunakan cara berikut:

Persentase = (Jumlah Uji Kasus Valid / Jumlah Uji Kasus Diujikan) x 100%

Persentase = (4 / 4) x 100%

Persentase = 100% (Valid)

Kesimpulan analisis yagn didapatkan berdasarkan hasil pengujian berupa tingkat keberhasilan validasi dan kompatibiliti. Pengujian validasi menghasilkan 100% valid terhadap analisis persyaratan dan pengujian Kompatibilitas menghasilkan *error issues* sebesar 9% dan dapat diterima tanpa kendala pada jenis peramban Microsoft Edge, Opera, iOS dan Android.

BAB 8 PENUTUP

8.1 Kesimpulan

Berdasarkan penelitian yang dilakukan, didapatkan kesimpulan dalam pengembangan Sistem Informasi Autopsi Verbal dengan hasil sebagai berikut:

1. Analisis persyaratan menghasilkan analisis permasalahan yang dihadapi pengguna, identifikasi terhadap tipe pemangku kepentingan, identifikasi pengguna yang menghasilkan 4 pengguna, identifikasi fitur yang menghasilkan 5 fitur sistem, identifikasi persyaratan fungsional sistem yang menghasilkan 34 persyaratan fungsional, identifikasi persyaratan nonfungsional sistem yang menghasilkan 1 persyaratan. Analisis persyaratan sistem mengenai tujuan sistem menghasilkan 7 *use case diagram. Use case diagram* dijelaskan secara terperinci pada spesifikasi *use case.* Terdapat 5 halaman prototipe tahap akhir yang menjadi gambaran dalam pembuatan sistem.
2. Perancangan lanjutan sistem menghasilkan interaksi antar komponen dan didefinisikan dalam arsitektur sistem. Pola perancangan yang digunakan adalah *Model-View-Controller (MVC)*, pemodelan kelas divisualisasikan melalui diagram terstandar yaitu diagram kelas. Diagram kelas perancangan menghasilkan 4 diagram kelas *controller* dan menghasilkan sebanyak 3 diagram kelas *model.* Diagram kelas *view* menghasilkan sebanyak 7 kelas sebagai *view* yang menjadi rancangan halaman sistem kedepannya. Perancangan lanjutan sistem menghasilkan perancangan basis data ke dalam *Physical Data Model* (PDM) dan terdapat 9 tabel pada sistem informasi autopsi verbal. Perancangan arsitektur sistem akan digunakan sebagai panduan dan acuan dalam melakukan implementasi sistem.
3. Implementasi sistem menghasilkan sistem informasi autopsi verbal berbasis web. Sistem informasi dikembangkan dengan menggunakan kerangka kerja Bootstrap dan CodeIgniter. Bahasa pemrograman yang digunakan adalah HTML, CSS, PHP dan Javascript serta database MySQL. Sistem informasi autopsi verbal memiliki fitur *login*, *logout*, mencatat hasil autopsi verbal, mengelola data hasil autopsi verbal, mengelola data pengguna sistem dan memonitor informasi autopsi verbal.
4. Hasil pengujian validasi sistem informasi autopsi verbal berdasarkan skenario uji yang dibuat mencatatkan hasil 100% valid dan sistem dapat digunakan sesuai dengan analisis persyaratan. Hasil pengujian kompatibilitas peramban dengan perangkat lunak Sortsite menghasilkan 9% error issues. Error issues ini disebabkan karena sistem memiliki masalah critical issues pada peramban Firefox versi 66 dan peramban Chrome versi 74 pada halaman login berdasarkan Sortsite. Terdapat *major issues* pada jenis peramban *Internet Explorer (IE)* versi 11 dan jenis peramban *Safari* versi 12 dikarenakan terdapat

CSS filter yang tidak lagi didukung oleh peramban tersebut. Selain itu, terdapat *minor issues* pada peramban Internet Explorer versi 11. Jenis peramban *Microsoft Edge, Opera, iOS* dan *Android* tidak mengalami kendala dalam menjalankan sistem informasi autopsi verbal.

8.2 Saran

Saran yang diberikan dapat digunakan sebagai bahan pertimbangan untuk melakukan pengembangan lebih lanjut mengenai Sistem Informasi Autopsi Verbal, diantaranya sebagai berikut:

1. Perlu dilakukan penerapan Sistem Informasi Autopsi Verbal secara langsung di lapangan oleh Dinas Kesehatan Kabupaten Malang sebagai kegiatan evaluasi agar penerapan sistem kedepannya dapat lebih baik lagi.
2. Perlu dilakukan pengembangan Sistem Informasi Autopsi Verbal lebih lanjut sebagai proses integrasi dengan sistem kesehatan lain yang ada pada Dinas Kesehatan Kabupaten Malang. Hal ini perlu dilakukan mengingat data kematian dapat dijadikan patokan untuk memberikan rekomendasi kesehatan bagi masyarakat sekitarnya.

DAFTAR REFRENSI

Bittner, K. & Spence, I., 2002. *Use Case Modeling.* 1st penyunt. Boston: Addison Wesley.

Booch, G., Rumbaugh, J. E. & Jacobson, I., 2007. *The Unified Modeling Language User Guide.* 3rd penyunt. Boston: Addison-Wesley.

Docherty, M. O., 2005. *Object-Oriented Analysis and Design : Understanding System Development with UML 2.0.* s.l., s.n.

Febriyanisa, S. R., 2018. Pengembangan Sistem Informasi Pencatatan Pengujian Relay Proteksi Pada Sub-Unit Basecamp Pt. Pln (Persero) Transmisi Jawa Bagian Timur Dan Bali Menggunakan Metode Waterfall. [Jurnal JPTIK] Tersedia di: <http://j-ptiik.ub.ac.id/index.php/j-ptiik/article/view/4672> [Diakses 2 Maret 2019]

IBM, 2004. *DEV475 Mastering Object-Oriented Analysis and Design with UML 2.0.* s.l.:IBM Rational University.

IBM, 2007. *Writing Good Use Cases. Student Manual.* s.l.:IBM Rational University.

Institute for Health Metrics and Evaluation, 2018. *Verbal Autopsy (VA).* s.l.:healthdata.

Kamus Besar Bahasa Indonesia, 2019. *Autopsi.* s.l., s.n.

Kemenkes Nomor 162/Menkes/Pb/I, 2010. *Peraturan Bersama Mentri Dalam Negeri Dan Mentri Kesehatan.* Jakarta: Kementrian Dalam Negeri. < Http://Pdk3mi.Org/?P=Download> [Diakses 28 Februari 2019]

Leffingwell , D. & Widrig, D., 1999. *Managing Software Requirements: A Unified Approach (Addison-wesley Object Technology Series).* 1st penyunt. Boston: Addison Wesley.

Microsoft Developer Networks , 2017. *UML Activity Diagrams.* s.l.:Microsoft. [Online] <https://docs.microsoft.com/en-us/visualstudio/modeling/uml-activity-diagrams-reference?view=vs-2015> [Diakses 03 Februari 2019]

Nasir, M., 2008. Pengembangan Sistem Informasi Pelayanan Kesehatan Ibu dan Bayi Untuk Mendukung Evaluasi Program Kesehatan Ibu dan Anak (KIA) Di Puskesmas Kabupaten Lamongan.

Object Management Group (OMG), 2005. *Unified Modeling Language: Superstructure.* s.l.:s.n. [Online] <https://www.omg.org/spec/UML/2.0/Superstructure/PDF/> [Diakses 18 Februari 2019]

Object Management Group (OMG), 2011. *Business Process Model and Notation (BPMN).* s.l.:s.n. [Online] <http://www.omg.org/spec/BPMN/2.0/PDF > [Diakses 10 Februari 2019]

PowerMapper, 2018. *SortSite - Browser Compatibility Tests..* s.l., s.n.[Oniline]< https://www.powermapper.com/products/sortsite/checks/browsercompatibility/>[Diakses 03 Maret 2019]

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

Sadabadi, A. T. & Tabatabaei, N. M., 2009. Rapid prototyping for software projects with user interfaces.

Sommerville, I., 2007. *Software Engineering.* 8 penyunt. Boston: Addison-Wesley.

Sommerville, I., 2011. *Software Engineering.* 9th penyunt. London: s.n. [Online] <https://www.softwaretestinghelp.com/software-compatibility-testing/> [Diakses 14 Maret 2019]

Stair, R. & Reynolds, G., 2014. Fundamentals of Information Systems. *Cengage Learning,* Volume 8.

Vuksanovic, I. P. & Sudarevic, B., 2011. *Use of Web Application Frameworks in the Development of Small Applications.* Zagreb, National and University Library.

Weske, M., 2007. *Business Process Management.* New York: Springer-Verlag Berlin Heidelberg.

Zwass, V., 2017. *Information system,* s.l.: s.n. [Online] <https://www.britannica.com/topic/information-system> [Diakses 20 Maret 2019]

LAMPIRAN A TRANSKIP WAWANCARA

Transkip Wawancara dengan Dr. SUJARWOTO, S.IP, M.Si, MPA

Berikut adalah transkrip hasil wawancara bersama Bapak Dr. Sujarwoto, S.IP, M.Si, MPA, selaku Ketua Peneliti Autopsi Verbal. Wawancara dilakukan pada:

Waktu : 23 Febuari 2019

Tempat : Gedung Fakultas Ilmu Administrasi Administrasi Universitas Brawijaya

Narasumber : Dr. Sujarwoto, S.IP, M.Si, MPA

Hasil Wawancara

Keterangan: P = Pewawancara N = Narasumber

P : Bagaimana penerapan proses autopsi verbal yang ada saat ini?

N : Ada beberapa tahapan dan juga aktor yang berperan saat ini adalah kader pewawancara. Terdapat informasi kematian lalu, kader pewawancara melakukan wawancara dengan keluarga terdekat almarhum/almarhumah untuk mendapatkan data lebih lengkap mengenai data kematian. Data lengkap kematian dapat memberikan gambaran pada kondisi wilayah penelitian yang dilakukan. Untuk saat ini autopsi verbal sedang dalam tahap pengumpulan data yang berada di wilayah kepanjen dan gondanglegi.

P : Apa terdapat sistem informasi atau aplikasi pada proses autopsi verbal?

N : Belum ada sistem informasi secara keseluruhan. Proses yang ada saat ini menggunakan formulir yang dapat dimasukan kedalam Open Data Kit (ODK) lalu pengelolaan data belum dilakukan dan masih dalam tahap pengumpulan.

P : Kira-kira apa yang bapak harapkan dari sistem informasi untuk membantu proses pengelolaan autopsi verbal?

N : Saya mengharapkan adanya sistem informasi yang dapat diinputkan data autopsi verbal. Data autopsi verbal smarthealth yang ada kemudian dapat diolah untuk membantu proses pengelolaan data sehingga dapat diambil kesimpulan mengenai informasi kesehatan pada wilayah bersangkutan dan membantu pemangku kepentingan mengambil kebijakan berdasarkan informasi kematian.

Malang, 23 Febuari 2019

Mengetahui

Dr. SUJARWOTO, S.IP, M.Si, MPA

LAMPIRAN B PENGUJIAN VALIDASI

Pengujian Validasi Mencatat Data Hasil Autopsi Verbal

Nama : *Dr. SUJARWOTO, S.IP, M.Si, MPA*

Jabatan : Ketua Peneliti Autopsi Verbal

Waktu Pengujian : 3 Juli 2019

Kode Pengujian	VT-SIAV-01
Nomor Skenario	Skenario 1
Kode Persyaratan	SIAV-F-05
Tujuan Pengujian	Pengujian dilakukan untuk memastikan penguji dapat *mencatat data hasil autopsi verbal ke dalam sistem dan* tersimpan dalam *database.*
Prosedur Uji	1. Penguji teridentifikasi sebagai kader pewawancara. 2. Penguji memilih menu formulir. 3. Penguji melakukan pengisian formulir autopsi verbal. 4. Penguji memilih tombol submit
Hasil yang diharapkan	Sistem memberikan pesan data berhasil tersimpan dan data yang dimasukan tersimpan dalam *database*.
Hasil Pengujian	Valid
Masukkan dan saran	Data berhasil di entry dan tersimpan di database

Malang,3 Juli 2019

(Dr. SUJARWOTO, S.IP, M.Si, MPA)

Pengujian Validasi Melihat Detail Data Autopsi Verbal

Nama : Dr. SUJARWOTO, S.IP, M.Si, MPA

Jabatan : Ketua Peneliti Autopsi Verbal

Waktu Pengujian : 3 Juli 2019

Kode Pengujian	VT-SIAV-03
Nomor Skenario	Skenario 3
Kode Persyaratan	SIAV-F-13
Tujuan Pengujian	Pengujian dilakukan untuk memastikan penguji dapat melihat detail data hasil autopsi verbal setiap almarhum/almarhumah.
Prosedur Uji	1. Penguji teridentifikasi sebagai staf verifikator dinas kesehatan kabupaten malang. 2. Penguji memilih menu data kematian. 3. Penguji memilih data hasil autopsi verbal yang ingin dilihat detailnya. 4. Penguji melakukan klik ikon detail pada kolom aksi. 5. Penguji melakukan klik tombol detail.
Hasil yang diharapkan	Muncul detail data hasil autopsi verbal secara lengkap.
Hasil Pengujian	Valid
Masukkan dan saran	Data berhasil di entry dan tersimpan di database

Malang,3 Juli 2019

(Dr. SUJARWOTO, S.IP, M.Si, MPA)

Pengujian Validasi Memverifikasi Data Hasil Autopsi Verbal

Nama : Dr. SUJARWOTO, S.IP, M.Si, MPA

Jabatan : Ketua Peneliti Autopsi Verbal

Waktu Pengujian : 3 Juli 2019

Kode Pengujian	VT-SIAV-02
Nomor Skenario	Skenario 2
Kode Persyaratan	SIAV-F-08
Tujuan Pengujian	Pengujian dilakukan untuk memastikan penguji dapat melakukan verifikasi data hasil autopsi verbal sehingga berubah menjadi data yang telah terverifikasi.
Prosedur Uji	1. Penguji teridentifikasi sebagai staf verifikator dinas kesehatan kabupaten malang. 2. Penguji memilih menu verifikasi. 3. Penguji memilih data hasil autopsi verbal yang ingin diverifikasi. 4. Penguji melihat detail data hasil autopsi verbal 5. Penguji melakukan klik ikon verifikasi pada kolom aksi 6. Penguji melakukan klik tombol verifikasi
Hasil yang diharapkan	Data tidak lagi muncul pada tabel verifikasi dan telah berpindah menjadi data kematian.
Hasil Pengujian	Valid
Masukkan dan saran	Data berhasil diverifikasi oleh petugas

Malang, 3 Juli 2019

(Dr. SUJARWOTO, S.IP, M.Si, MPA)

Pengujian Validasi Memonitor Hasil Autopsi Verbal

Nama : Dr. SUJARWOTO, S.IP, M.Si, MPA

Jabatan : Ketua Peneliti Autopsi Verbal

Waktu Pengujian : 3 Juli 2019

Kode Pengujian	VT-SIAV-04
Nomor Skenario	Skenario 4
Kode Persyaratan	SIAV-F-28
Tujuan Pengujian	Pengujian dilakukan untuk memastikan penguji dapat melihat informasi umum hasil autopsi verbal.
Prosedur Uji	1. Penguji teridentifikasi telah terdaftar dalam sistem. 2. Penguji memilih menu dasbor. 3. Penguji melihat informasi autopsi verbal secara umum.
Hasil yang diharapkan	Muncul informasi dan grafik hasil autopsi verbal.
Hasil Pengujian	Valid
Masukkan dan saran	Dashboard menampilkan data kematian.

Malang, 3 Juli 2019

(Dr. SUJARWOTO, S.IP, M.Si, MPA)