PENGEMBANGAN APLIKASI INVESTASI SYARIAH PADA PT MASLAHAT AMANAH UMAT BERBASIS ANDROID

SKRIPSI

Untuk memenuhi sebagian persyaratan
memperoleh gelar Sarjana Komputer

Disusun oleh:
Jeva Prahayusma Putra Emanda
NIM: 155150207111098

PROGRAM STUDI TEKNIK INFORMATIKA
JURUSAN TEKNIK INFORMATIKA
FAKULTAS ILMU KOMPUTER
UNIVERSITAS BRAWIJAYA
MALANG
2019

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

PENGESAHAN

PENGEMBANGAN APLIKASI INVESTASI SYARIAH PADA PT MASLAHAT AMANAH UMAT BERBASIS ANDROID

SKRIPSI

Diajukan untuk memenuhi sebagian persyaratan
memperoleh gelar Sarjana Komputer

Disusun Oleh :
Jeva Prahayusma Putra Emanda
NIM: 155150207111098

Skripsi ini telah diuji dan dinyatakan lulus pada
4 April 2019
Telah diperiksa dan disetujui oleh:

Dosen Pembimbing I	Dosen Pembimbing 2
Nurudin Santoso, S.T., M.T	Eriq Muhammad A. J., S.T, M.Kom
NIP: 197409162000121001	NIP: 198504102012121001

Mengetahui
Ketua Jurusan Teknik Informatika

Tri Astoto Kurniawan, S.T, M.T, Ph.D
NIP: 197105182003121001

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

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, 2 Maret 2019

METERAI TEMPEL
6000
ENAM RIBU RUPIAH

Jeva Prahayusma Putra Emanda

NIM: 155150207111098

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

PRAKATA

Segala puji dan syukur bagi Allah SWT atas berkah, limpahan rahmat yang diberikan kepada penulis, sehingga dapat menyelesaikan skripsi berjudul "PENGEMBANGAN APLIKASI INVESTASI SYARIAH PADA PT MASLAHAT AMANAH UMAT BERBASIS MOBILE".

Dalam penyusunan dan penulisan naskah skripsi ini penulis mendapatkan banyak hambatan serta rintangan, namun pada akhirnya penulis dapat melaluinya berkat adanya bimbingan dan bantuan serta dukungan dari berbagai pihak baik secara moral maupun spiritual. Dalam kesempatan ini penulis ingin mengucapkan banyak terima kasih kepada yang telah membantu dalam penyelesaian skripsi ini diantaranya :

1. Bapak Nurudin Santoso, S.T., M.T selaku Dosen pembimbing pertama yang telah bersedia meluangkan waktu untuk memberikan arahan selama penyusunan skripsi, memberikan banyak solusi dalam pengerjaan skripsi, serta menjadi pembimbing yang sabar, solutif, dan memberikan contoh sebagai pribadi yang profesional dalam memberikan bimbingan kepada mahasiswa. Sehingga ilmu yang telah diajarkan dapat bermanfaat dikemudian hari.
2. Bapak Eriq Muhammad Adams Jonemaro, S.T, M.Kom selaku Dosen pembimbing kedua yang telah telah bersedia meluangkan waktu untuk memberikan arahan selama penyusunan skripsi dan mengajarkan kedisiplinan dalam menyelesaikan kewajiban. Memberikan contoh sebagai pribadi yang sabar, solutif, dan profesional dalam memberikan bimbingan kepada mahasiswa. Sehingga ilmu yang telah diajarkan dapat bermanfaat dikemudian hari.
3. Bapak Wayan Firdaus Mahmudy, S.Si, M.T, Ph.D selaku Dekan Fakultas Ilmu Komputer Universitas Brawijaya.
4. Bapak Tri Astoto kurniawan, S.T, M.T dan Agus Wahyu Widodo, S.T, M.Cs., selaku Ketua Jurusan Teknik Informatika Universitas Brawijaya dan Ketua Program Studi Teknik Informatika Universitas Brawijaya.
5. Ayah dan Ibu tercinta yang selalu memberikan dukungan do'a, moril, material, motivasi, kedisiplinan, dan nasihat untuk menjadi pribadi yang sabar dan pekerja keras. Serta adik-adik penulis yang tersayang.
6. Bapak Zulham Annas. Selaku pendiri dari PT Maslahat Amanah Umat yang telah memberikan saya izin dan motivasi untuk melakukan penelitian ini.
7. Bapak Syaifudin, Drs., M.hum. Selaku petugas ruang baca yang telah memberi kesempatan dan kemudahan dalam melakukan penelitian sehingga sangat membantu dalam penyelesaian skripsi ini.

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

8. Fithri Annisa, Muhammad Abdan, dan Dimas Aryo yang telah memberikan semangat, motivasi, dan menemani dalam penyelesaian skripsi ini.
9. Serta seluruh warga Himpunan Mahasiswa Informatika (HMIF) dan Eksekutif Mahasiswa Informatika (EMIF) yang tidak dapat saya sebutkan namanya satu per satu, yang telah memberikan dukungan dan motivasi kepada penulis, menjadi teman dan sahabat yang baik dan menyenangkan selama masa perkuliahan.
10. Sahabat-sahabat saya dari asisten dosen dan base campe yang telah menyemangati saya dalam masa perkuliahan mapaun dalam penyelesaian skripsi saya.
11. Semua pihak yang telah banyak membantu untuk berbagi ilmu yang tidak bisa penulis sebutkan satu per satu.

Semoga Allah SWT melimpahkan rahmat berlipat ganda kepada seluruh pihak yang membantu penulis dalam menyelesaikan skripsi ini. Semoga apa yang telah ditulis dalam skripsi ini dapat bermanfaat bagi semua pihak. Amin ya Rabbal a'lamin.

Malang, Maret 2019

Penulis
jevaprahayusma@student.ub.ac.id

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

repository.ub.ac.id

ABSTRAK

Perkembangan *fintech* saat ini begitu cepat dan memudahkan manusia. Dimana pertumbuhannya di Indonesiapun cukup banyak untuk penyedia jasa layanan *fintech*. Oleh karena itu PT Maslahat Amanah Umat sebagai perusahaan yang berbasis syariah mengembangkan sebuah *web site* untuk mempermudah jalannya sebuah investasi. Akan tetapi kurangnya cakupan pengguna dan tidak semua orang suka membuka sebuah alamat *web site* melalu telfon genggamnya. Karena pasar usaha yang ditawarkan merupakan golongan menengah kebawah. Sehingga dikembangkannya aplikasi investasi syariah berbasis Android Maslahat Kita yang akan menjadi pilihan lain untuk melakukan kegiatan *finance* pada pengguna telfon genggam dengan sistem operasi Android.

Aplikasi Maslahat Kita pada sistem operasi android ini dibangun dengan tujuan mempermudah pengguna untuk melakukan transaksi atau akad yang akan dilakukan ketika ingin mengajukan usahanya atau mengajukan investasinya pada usaha yang di inginkan. Pada aplikasi ini juga dilakukan secara *real time* menggunakan *firebase.* Sehingga komunikasi antar pengguna dengan hak akses tertentu bisa menggunakan perangkat lain untuk memakasi aplikasi ini. Serta penggunaan grafik untuk mempermudah monitoring usaha yang dilakukan mitra pada aplikasi ini. Untuk itu aplikasi Maslahat Kita menggunakan beberapa pengujian diantaranya pengujian unit, pengujian integrasi, pengujian validasi dan pengujian *compatibility*. Dimana pengujian unit dan pengujian integrase menunjukan valid untuk setiap program yang di jalankan. Sedangkan untuk pengujian validasi memaparkan hasil yang valid pada fungsionalitas yang telah dibuat. Dan juga pada pengujian *compatibility* untuk setiap versi yang telah diuji. Menghasilkan hasil yang valid karena tidak adanya perubahan secara tampilan aplikasi atau kendala *error* pada setiap fungsioanalitasnya ketika diuji dengan versi Android yang berbeda.

Kata kunci : *fintech*, *Android, firebase, finance, real time*.

UNIVERSITAS BRAWIJAYA

repository.ub.ac.id

ABSTRACT

Day by day, the development of fintech is getting faster and make human's life easier. Even in Indonesia, fintech provider has massively proliferated. Under that reason, PT. Maslahat Ummah whose company working under the basis of sharia developed a website to make investation easier. But then, the lack of coverage for the user, and how people less prefer accessing website from their mobile phone. Because of business market that offered were mid to low class businesses. Therefore, developed an android based sharia investation application named Maslahat Kita to act as an alternative to do some finances activity through their android based mobile phone.

The android based Maslahat Kita apps was created with a goal to make user feel a lot easier when it comes to transactions or contractual agreements in their businesses or investations they desire. This application also integrated real time with firebase. Therefore communication between users with specific access right is possible to be done with various devices under this application. And then, the usage of charts going to make monitoring easier for our partners. So that is why The Maslahat Kita application has gone through several testing process such as compatibility testing, integrity testing, validity testing and unit testing. Where unit testing and integrity testing used as valid indicator for the program as a whole. But in validity testing, to measure whether agreed functionality is working or not. Last but not least compatibility testing to test every versions created. We are proud to declare that the testing result should be categorized as valid, because there's no significant changes to the system's design and there is no errors at every functionality agreed, when it is being tested on different Android version.

Keyword : fintech, Android, firebase, finance, real time.

UNIVERSITAS BRAWIJAYA

DAFTAR ISI

PENGESAHAN ii
PERNYATAAN ORISINALITAS iii
PRAKATA iv
ABSTRAK vi
ABSTRACT vii
DAFTAR ISI viii
DAFTAR TABEL xii
DAFTAR GAMBAR xv
BAB 1 PENDAHULUAN 1
1.1 Latar Belakang 1
1.2 Rumusan Masalah 2
1.3 Tujuan 2
1.4 Manfaat 2
1.5 Batasan Masalah 2
1.6 Sistematika Pembahasan 3
BAB 2 LANDASAN KEPUSTAKAAN 4
2.1 Kajian Pustaka 4
2.2 *Fintech* Syariah 5
2.2.1 Peminjaman Dan Investasi Syariah 6
2.3 Android 7
2.4 *Firebase* 8
2.5 *Waterfall Model* 9
2.6 Pemodelan Proses Bisnis 11
2.7 *Use Case Diagram* 13
2.8 *Activity Diagram* 14
2.9 *Class Diagram* 15
2.10 Pengujian *White Box* 17
2.10.1 Pengujian Unit 18
2.10.2 Pengujian Integrasi 18

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

2.11 Pengujian *Black Box* 18
2.11.1 Pengujian Validasi 18
2.11.2 Pengujian *Compatibility* 19
BAB 3 METODOLOGI 20
3.1 STUDI LITERATUR 20
3.2 PENGUMPULAN DATA 21
3.3 ANALISIS KEBUTUHAN 21
3.4 PERANCANGAN 21
3.5 IMPLEMENTASI 21
3.6 PENGUJIAN 21
3.7 PENGAMBILAN KESIMPULAN 22
BAB 4 Rekayasa kebutuhan 23
4.1 Gambaran Umum Sistem 23
4.2 Proses Bisnis 23
4.3 Analisis Kebutuhan 32
4.3.1 Elisitasi Kebutuhan 32
4.3.2 Identifikasi Aktor 32
4.4 Batasan Sistem 33
4.5 Lingkungan Operasi 33
4.6 Daftar Kebutuhan 33
4.6.1 Kebutuhan Fungsional 34
4.6.1 Kebutuhan Non Fungsional 36
4.7 Pemodelan Kebutuhan 36
4.7.1 Use Case Diagram 36
4.7.2 Use Case Scenario 37
BAB 5 PERANCANGAN DAN IMPLEMENTASI 48
5.1 Perancangan 48
5.1.1 Perancangan *Activity Diagram* 48
5.1.2 Perancangan *Class Diagram* 50
5.1.3 Perancangan Komponen 52
5.1.4 Perancangan Basis Data 54
5.1.5 Perancangan Antarmuka 59

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

5.1.5.1 Perancangan *Screenflow* 59
5.1.5.1 Perancangan *wireframe* 61
5.2 Implementasi Sistem 75
5.2.1 Spesifikasi Piranti Pendukung 75
5.2.2 Implementasi Kode Program 75
5.2.3 Implementasi Basis Data 80
5.2.4 Implementasi Antarmuka 84
BAB 6 PENGUJIAN 102
6.1 Pengujian Unit 102
6.1.1 Pengujian Unit Kelas PengajuanUsahaActivity Pada Operasi onClick 102
6.1.2 Pengujian Unit Kelas VerifikasiInvestasiFragment untuk operasi onDataChange 104
6.1.3 Pengujian Unit Kelas PelaporanUsahaActivity Untuk Operasi onClick 108
6.2 Pengujuan Integrasi 112
6.3 Pengujian Validasi 116
6.2.1 Pengujian Validasi Pangajuan Sebagai Mitra Usaha 116
6.2.2 Pengujian Validasi Verifikasi Pangajuan Sebagai Mitra Usaha . 117
6.2.3 Pengujian Validasi Pangajuan Sebagai Investor 117
6.2.4 Pengujian Validasi Verifikasi Pangajuan Sebagai Investor 118
6.2.5 Pengujian Validasi Pangajuan Pembiayaan Usaha 118
6.2.6 Pengujian Validasi Verifikasi Pangajuan Pembiayaan Usaha 119
6.2.7 Pengajuan Investasi 120
6.2.8 Pengujian Verifikasi Pengajuan Investasi 120
6.2.9 Pengujian Validasi Pencairan Dana Usaha 121
6.2.10 Pengujian Validasi Pengangsuran 121
6.2.11 Pengujian Validasi Verifikasi Pengangsuran 122
6.2.12 Pengujian Validasi Pelaporan Pengangsuran 122
6.2.13 Pengujian Validasi Pencairan Dana Hasil Investasi 123
6.4 Pengujian *Compatibility* 123
6.5 Analisis Hasil Pengujian 125
BAB 7 KESIMPULAN DAN SARAN 127

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

7.1 Kesimpulan..... 127
7.2 Saran 127
DAFTAR REFRENSI 128

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

DAFTAR TABEL

Tabel 2.1 Pertumbuhan Perbankan Syariah Di Indonesia 6
Tabel 2.2 Pertumbuhan Perbankan Syariah Di Indonesia 11
Tabel 2.3 Pertumbuhan Perbankan Syariah Di Indonesia (lanjutan)..... 12
Tabel 2.4 Elemen *Use Case Diagram* 13
Tabel 2.5 Simbol-simbol Pada *Activity* Diagram (Sugiarti, 2013)..... 14
Tabel 2.6 Simbol-simbol Pada *Activity* Diagram (Sugiarti, 2013) (lanjutan)..... 15
Tabel 2.7 Simbol-simbol Pada *Class* Diagram (Sugiarti, 2013)..... 16
Tabel 4.1 Daftar Proses Bisnis 31
Tabel 4.2 Daftar Proses Bisnis (lanjutan) 32
Tabel 4.3 Daftar Identifikasi Aktor 33
Tabel 4.4 Daftar Kebutuhan Fungsional..... 34
Tabel 4.5 Daftar Kebutuhan Fungsional (lanjutan) 35
Tabel 4.6 Daftar Kebutuhan Fungsional (lanjutan) 36
Tabel 4.7 Daftar Kebutuhan Non Fungsional 36
Tabel 4.8 *Use Case Scenario* Pengajuan Sebagai Mitra Usaha 38
Tabel 4.9 *Use Case Scenario* Verifikasi Pengajuan Sebagai Mitra Usaha..... 38
Tabel 4.10 *Use Case Scenario* Verifikasi Pengajuan Sebagai Mitra Usaha (lanjutan) 39
Tabel 4.11 *Use Case Scenario* Pengajuan Sebagai Investor 39
Tabel 4.12 *Use Case Scenario* Pengajuan Sebagai Investor (lanjutan) 40
Tabel 4.13 *Use Case Scenario* Verifikasi Pengajuan Sebagai Mitra Usaha..... 40
Tabel 4.14 *Use Case Scenario* Verifikasi Pengajuan Sebagai Mitra Usaha (lanjutan) 41
Tabel 4.15 *Use Case Scenario* Pengajuan Pembiayaan Usaha 41
Tabel 4.16 *Use Case Scenario* Verifikasi Pengajuan Sebagai Mitra Usaha..... 42
Tabel 4.17 *Use Case Scenario* Pengajuan Investasi..... 43
Tabel 4.18 *Use Case Scenario* Verifikasi Pengajuan Investasi..... 43
Tabel 4.19 *Use Case Scenario* Verifikasi Pengajuan Investasi (lanjutan) 44
Tabel 4.20 *Use Case Scenario* Pencairan Dana Usaha 44
Tabel 4.21 *Use Case Scenario* Pencairan Dana Usaha (lanjutan)..... 45

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

Tabel 4.22 *Use Case Scenario* Pengangsuran 45
Tabel 4.23 *Use Case Scenario* Verifikasi Pengangsuran 46
Tabel 4.24 *Use Case Scenario* Pelaporan Angsuran 47
Tabel 4.25 *Use Case Scenario* Pencairan Dana Hasil Investasi 47
Tabel 5.1 *Pseudocode Method* onClick 52
Tabel 5.2 *Pseudocode Method* onDataChance 53
Tabel 5.3 *Pseudocode Method* onClick 54
Tabel 5.4 Struktur Tabel UserManagement 56
Tabel 5.5 Struktur Tabel VirtualAccount 56
Tabel 5.6 Struktur Tabel VerifikasiInvestasi 57
Tabel 5.7 Struktur Tabel MitraManagement 57
Tabel 5.8 Struktur Tabel InvestorManagement 57
Tabel 5.9 Struktur Tabel InvestorManagement (lanjutan) 58
Tabel 5.10 Struktur Tabel PengajuanUsaha 58
Tabel 5.11 Struktur Tabel MonitoringUsaha 59
Tabel 5.12 Spesifikasi Perangkat Keras 75
Tabel 5.13 Spesifikasi Perangkat Lunak 75
Tabel 6.1 Hasil Pengujian Unit *Method onClick* 104
Tabel 6.2 Hasil Pengujian Unit *Method onDataChange* 107
Tabel 6.3 Hasil Pengujian Unit *Method onDataChange* (lanjutan) 108
Tabel 6.4 Hasil Pengujian Unit *Method onClick* 111
Tabel 6.5 Hasil Pengujian Unit *Method onClick* (lanjutan) 112
Tabel 6.6 Hasil Pengujian Integrasi *Method onDataChange* 115
Tabel 6.7 Hasil Pengujian Integrasi *Method onDataChange* (lanjutan) 116
Tabel 6.8 Pengujian Validasi Pengajuan Sebagai Mitra Usaha 116
Tabel 6.9 Pengujian Validasi Pengajuan Sebagai Mitra Usaha (lanjutan) 117
Tabel 6.10 Pengujian Validasi Verifikasi Pengajuan Sebagai Mitra Usaha 117
Tabel 6.11 Pengujian Validasi Pengajuan Sebagai Investor 118
Tabel 6.12 Pengujian Validasi Verifikasi Pangajuan Sebagai Investor 118
Tabel 6.13 Pengujian Validasi Pangajuan Pembiayaan Usaha 119
Tabel 6.14 Pengujian Validasi Verifikasi Pangajuan Pembiayaan Usaha 119
Tabel 6.15 Pengujian Validasi Pengajuan Investasi 120

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

Tabel 6.16 Pengujian Validasi Pengajuan Investasi 120
Tabel 6.17 Pengujian Validasi Pengajuan Investasi (lanjutan)..... 121
Tabel 6.18 Pengujian Validasi Pencairan Dana Usaha 121
Tabel 6.19 Pengujian Validasi Pengangsuran..... 121
Tabel 6.20 Pengujian Validasi Verifikasi Pengangsuran..... 122
Tabel 6.21 Pengujian Validasi Pelaporan Pengangsuran 122
Tabel 6.22 Pengujian Validasi Pencairan Dana Hasil Investasi 123
Tabel 6.23 Hasil Pengujian *Compatibility*..... 125

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

DAFTAR GAMBAR

Gambar 2.1 Langkah-langkah *Waterfall Model* (Sommervile,2011) 10

Gambar 2.2 Contoh Proses Bisnis Produksi dan Pemasaran (Heryanto,2018).... 12

Gambar 2.3 Contoh Use Case Sistem Informasi Monitoring Penjualan dan Stok Barang (Hendini, 2016) 14

Gambar 2.4 Contoh Activity Diagram Login Admin/Karyawan Gudang (Hendini, 2016) 15

Gambar 2.5 Contoh *Class Diagram Login* Pemesanan Dan Penjualan Pada Dewi Florist (Setiay, 2013)..... 17

Gambar 3.1 Alur Metodelogi 20

Gambar 4.1 Proses Bisnis Pendafataran Mitra 24

Gambar 4.2 Proses Bisnis Pendafataran Investor 25

Gambar 4.3 Proses Bisnis Pengajuan Usaha 26

Gambar 4.4 Proses Bisnis *Market Matching* 27

Gambar 4.5 Proses Bisnis Realisasi Dana 29

Gambar 4.6 Proses Bisnis Pengangsuran Dana Usaha 30

Gambar 4.7 *Use Case Diagram* 37

Gambar 5.1 *Activity* Diagram Pengajuan Usaha 48

Gambar 5.2 *Activity Diagram* Verifikasi Investasi 49

Gambar 5.3 *Activity Diagram* Pelaporan Usaha 50

Gambar 5.4 *Class Diagram* Aplikasi MaslahatKita 51

Gambar 5.5 Perancangan Basis Data 55

Gambar 5.6 Perancangan *Screenflow* 60

Gambar 5.7 Perancangan Antarmuka *Login* 61

Gambar 5.8 Perancangan Antarmuka *Guest* 62

Gambar 5.9 Perancangan Antarmuka Mitra 62

Gambar 5.10 Perancangan Antarmuka Pengajuan Usaha 63

Gambar 5.11 Perancangan Antarmuka *Monitoring* Usaha Mitra 63

Gambar 5.12 Perancangan Antarmuka Pelaporan Usaha 64

Gambar 5.13 Perancangan Antarmuka *List* Pengajuan Usaha Baru 64

Gambar 5.14 Perancangan Antarmuka Detail Pengajuan Usaha Baru 65

Gambar 5.15 Perancangan Antarmuka List Pengajuan Usaha Ditolak 66

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

Gambar 5.16 Perancangan Antarmuka Detail Pengajuan Usaha Ditolak 66
Gambar 5.17 Perancangan Antarmuka *List* Pengajuan Diterima 67
Gambar 5.18 Perancangan Antarmuka Detail Pengajuan Usaha Diterima 67
Gambar 5.19 Perancangan Antarmuka Admin 68
Gambar 5.20 Perancangan Antarmuka *List* Verifikasi Pengajuan Usaha..... 69
Gambar 5.21 Perancangan Antarmuka Detail Verifikasi Pengajuan Usaha..... 69
Gambar 5.22 Perancangan Antarmuka *Form* Penolakan..... 70
Gambar 5.23 Perancangan Antarmuka *List* Verifikasi Investasi 71
Gambar 5.24 Perancangan Antarmuka Investor 72
Gambar 5.25 Perancangan Antarmuka *Market Place* 72
Gambar 5.26 Perancangan Antarmuka Detail Usaha Mitra 73
Gambar 5.27 Perancangan Antarmuka *Form* Saran 74
Gambar 5.28 Perancangan Antarmuka *Monitoring* Usaha Untuk Investor..... 74
Gambar 5.29 Implementasi Basis Data UserManagement..... 80
Gambar 5.30 Implementasi Basis Data VirtualAccount..... 81
Gambar 5.31 Implementasi Basis Data VerifikasiInvestasi..... 81
Gambar 5.32 Implementasi Basis Data MitraManagement 82
Gambar 5.33 Implementasi Basis Data InvestorManagement..... 82
Gambar 5.34 Implementasi Basis Data PengajuanUsaha 83
Gambar 5.35 Implementasi Basis Data MonitoringUsaha..... 83
Gambar 5.36 Implementasi Antarmuka *Login*..... 84
Gambar 5.37 Implementasi Antarmuka *Guest* 85
Gambar 5.38 Implementasi Antarmuka Mitra..... 86
Gambar 5.39 Implementasi Antarmuka Pengajuan Usaha..... 86
Gambar 5.40 Implementasi Antarmuka *Monitoring* Usaha Mitra..... 87
Gambar 5.41 Implementasi Antarmuka Pelaporan Usaha 88
Gambar 5.42 Implementasi Antarmuka *List* Pengajuan Usaha Baru..... 88
Gambar 5.43 Implementasi Antarmuka Detail Pengajuan Usaha Baru..... 89
Gambar 5.44 Implementasi Antarmuka List Pengajuan Usaha Ditolak..... 90
Gambar 5.45 Implementasi Antarmuka Detail Pengajuan Usaha Ditolak..... 91
Gambar 5.46 Implementasi Antarmuka *List* Pengajuan Diterima 91
Gambar 5.47 Implementasi Antarmuka Detail Pengajuan Usaha Diterima 92

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

Gambar 5.48 Implementasi Antarmuka Admin 93
Gambar 5.49 Implementasi Antarmuka *List* Verifikasi Pengajuan Usaha 94
Gambar 5.50 Implementasi Antarmuka Detail Verifikasi Pengajuan Usaha 95
Gambar 5.51 Implementasi Antarmuka *Form* Penolakan 96
Gambar 5.52 Implementasi Antarmuka *List* Verifikasi Investasi 97
Gambar 5.53 Implementasi Antarmuka Investor 98
Gambar 5.54 ImplementasiAntarmuka *Market Place* 98
Gambar 5.55 Implementasi Antarmuka Detail Usaha Mitra 99
Gambar 5.56 Implementasi Antarmuka *Form* Saran 100
Gambar 5.57 Implementasi Antarmuka *Monitoring* Usaha Untuk Investor 101
Gambar 6.1 *Flow Graph Method onClick* 103
Gambar 6.2 *Flow Graph Method onDataChange* 106
Gambar 6.3 *Flow Graph Method onDataChange* 114
Gambar 6.4 Tampilan Aplikasi Pada Android Versi 4.4 (*Kitkat*)..... 124
Gambar 6.5 Tampilan Aplikasi Pada Android Versi 7.0 (*Nougat*) 124
Gambar 6.6 Tampilan Aplikasi Pada Android Versi 8.0 (*Oreo*) 125

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

BAB 1 PENDAHULUAN

1.1 Latar Belakang

Pada saat ini banyak masyarakat membutuhkan sebuah peminjaman untuk melakukan sebuah bisnis yang akan di jalankan. Akan tetapi terkendala dalam hal *financial*. Yang mana kebanyak proses tersebut membutuhkan sebuah pihak yang akan menbantu berjalannya sebuah bisnis seseorang. Dimana zaman yang sudah maju ini berkembang akan teknologi yang memepermudah masyarat untuk dapat melakukan proses peminjaman. Yang mana teknologi tersebut di namakan *fintech* kepanjangan dari financial technologi bergerak pada bidang financial yang menggunakan technology sebagai wadahnya. Selain dalam hal tersebut masyarakat terutama umat muslim dalam hal peminjaman juga kawatir akan namanya *riba* bila melakukkan peminjaman dengan perusahaan konvensional. Olah karena itu dibuatlah *fintech* yang berbasis syariah dalam sistemnya nanti yang akan di kembangkan lebih lanjut dari PT Maslahat Amanah Umat dalam bentuk sebuah aplikasi *mobile* berbasis Android yang dinamakan aplikasi Maslahat Kita. Dimana nanti akan terdapat tiga fitur utama yaitu pengajuan usaha, investasi usaha, dan monitoring usaha.

Yang mana bisnis *fintech* saat ini cukup menjamur. Yang mana terdapat data *fintech lending* terdapat 29 perusahaan dan dalam waktu dekat akan mencapai 96 perusahaan yang mana pasar bisnis ini cukup luas. Berdasarkan data OJK, perkembangan *fintech lending* sepanjang tahun lalu tumbuh sangat signifikan. Peminat pengguna jasa P2P Fintech Lending, baik dari pemberi pinjaman maupun pencari pinjaman sangat besar. Hingga akhir 2017, total jumlah pemberi pinjaman mencapai 100.940 orang, naik 603 persen dari Desember 2016 sebanyak 14.364 orang. Dari total tersebut, sekitar 75 persen atau 75.769 orang disumbang dari Pulau Jawa. Sementara itu, pemberi pinjaman dari luar Pulau Jawa menyumbang sekitar 24 persen atau sebanyak 24.028 orang. Pemberi pinjaman juga ternyata ada yang berasal dari luar negeri, yakni sekitar 1 persen atau sebanyak 1.143 orang. Jumlah pencari pinjaman juga tumbuh sangat signifikan. OJK mencatat total jumlah pencari pinjaman mencapai 259.635 orang per Desember 2017, naik 581 persen dari Desember 2016 sebanyak 38.105 orang. Dari total jumlah pencari pinjaman itu, sebanyak 237.159 orang, atau 91 persen berasal dari Pulau Jawa. Sementara sisanya, yakni 9 persen atau sebanyak 22.316 orang berasal dari luar Pulau Jawa. Seiring dengan jumlah pemberi dan pencari pinjaman yang meningkat, nilai pinjaman yang dikucurkan pun tumbuh signifikan, yakni mencapai Rp2,56 triliun atau melesat 800 persen dari Desember 2016 sebesar Rp284,15 miliar (Ringkang, 2018).

Oleh karena data yang saat ini bisnis *fintech* mulai banyak. Akan tetapi binis serupa yang berbasis syariah saat ini masih kurang. Yang mana untuk malakukan pendaftaran usaha harus mengisi banyak form yang di sediakan pada *website*nya. Dan *platform* yang di gunakan juga masih terbatas. Maka dari itu di butuhkan

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

sebuah media yang mempermudah sistem investasi tersebut. Yang mana nantinya dampak dari pembuatan sistem *fintech* p2p yang berbasis syariah ini dapat membantu orang lain. Yang mana sistem ini akan mengatur proses akad maupun proses peminjaman yang nanti akan di ajalankan.

1.2 Rumusan Masalah

Berdasarkan uraian pada bagian latar belakang, maka masalah yang ada dapat dirumuskan sebagai berikut:

1. Bagaimana jalannya proses bisnis yang di lakukan pada aplikasi ini?
2. Bagaimana perancangan dan implementasi dalam pembuatan sistem *fintech* berbasis syariah tersebut?
3. Bagaimana *compatibility* dan fungsionalitas dari aplikasi *fintech* berbasis syariah ini?

1.3 Tujuan

Berdasarkan rumusan pada masalah yang ada di atas, maka dapat diketahui yaitu sebagai berikut:

1. Dapat merancang aplikasi *fintech p2p lending* syariah ini menjadi aplikasi berbasis mobile.
2. Dapat menjalankan proses bisnis *fintech p2p lending* berbasis syariah yang di lakukan pada aplikasi tersebut.
3. Untuk mengatahui *compatibility* dan fungsionalitas dari aplikasi *fintech p2p lending* berbasis syariah.

1.4 Manfaat

Manfaat yang akan di peroleh dari penelitian ini yaitu:

1. Mempermudah dalam hal untuk melakukan peminjaman dana usaha.
2. Mempermudah dalam proses akad nantinya.
3. Lebih transparansi akan dana yang masuk maupun keluar
4. Menambah efesiansi waktu yang di butuhkan dalam proses peminjaman atau berinvestasi.

1.5 Batasan Masalah

Agar permasalahan yang dibuat pada aplikasi ini dapat terfokus, maka pada penelitian ini akan dibatasi dalam hal:

1. Pembuatan aplikasi pada platform android yang nantinya akan dibuat
2. Tidak membahas tentang penggunaan firebase real time.
3. Tidak adanya pembahasan pada sistem ke amanan pada aplikasi tersebut.
4. Aplikasi ini berjalan 24 jam dalam platform yang sudah di sediakan.
5. Menggunakan sistem akad murabahah untuk proses bisnis yang dilakukan.

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

1.6 Sistematika Pembahasan

Sistematika penulisan dalam tugas akhir ini nantinya akan dilakukan sebagai berikut:

BAB 1 PENDAHULUAN

Di dalam pendahulan memuat latar belakang, rumusan masalah, tujuan, manfaat, batasan masalah, dan sistematika penulisan pada penelitan ini.

BAB 2 LANDASAN KEPUSTAKAAN

Menguraikan tentang dasar teori dan refrensi yang akan mendasari pada pembuatan palikasi pemesanan pencetakan yang berbasis Android, dengan analisis perancangan serta pengujiannya nanti akan seperti apa.

BAB 3 METODE PENELITIAN

Membahas metode yang digunakan dalam penelitian ini yang terdiri dari studi literatur, perancangan sistem perangkat lunak, implementasi sistem perangkat lunak, pengujian dan analisis, serta pengambilan keputusan untuk pembuatan aplikasi pemesenan pencetakan ini.

BAB 4 REKAYASA KEBUTUHAN

Membahas tentang daftar terstruktur dari kebutuhan perangkat lunak, secara fungsional, data, maupun non fungsional. Serta memuat akan pemangku kepentingan maupun aktor pada sistem yang nanti akan di kut sertakan. Dan juga nanti akan berupa diagram-diagram yang di butuhkan pada analisis kebutuhan pada sistem *fintech p2p lending* berbasis syariah yang akan di implementasikan.

BAB 5 PERANCANGAN DAN IMPLEMENTASI

Membahas tentang implementasi dari pembuatan aplikasi Maslahat Kita. Yang mana pada bab ini akan membahans tentang rancangan arsitektur yang akan dibuat dalam hal struktur maupunn setiap komponen utama yang akan di implementasikan. Maupun perancangan data dalam model data maupun basis data yang annti akan di representasikan. Serta implementasi dari fungsi-fungsi utama yang telah di tentukan maupaun implementasi antar muka yang telah dibuat.

BAB 6 PENGUJIAN

Memuat tentang berbagai data pengujian serta kasus-kasus yang akan di paparkan nantinya. Dalam hal hal ini juga akan di lakukan tentang pengujian unit, pengujian integrasi, pengujian validasi dan pengujian kompatibilitas yang diujikan pada aplikasi tersebut.

BAB 7 PENUTUP

Memuat kesimpulan yang diperoleh dari pembuatan dan pengujian perangkat lunak yang dikembangkan dalam tugas akhir ini ini serta saran-saran untuk pengembangan lebih lanjut lagi dari pembuatan aplikasi MaslahtKita ini.

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

BAB 2 LANDASAN KEPUSTAKAAN

2.1 Kajian Pustaka

Penelitian yang dilakukan oleh Imanuel Adhitya Wulanata Chrismastianto (2017) dengan judul *"ANALISIS SWOT IMPLEMENTASI TEKNOLOGI FINANSIAL TERHADAP KUALITAS LAYANAN PERBANKAN DI INDONESIA"*. Teknologi Finansial atau *Financial Technology* (*FinTech*) pada dunia perbankan. Menurut definisi yang talah dipaparkan oleh *National Digital Research Centre* (NDRC), teknologi finansial merupakan istilah yang digunakan untuk menyebutkan pada suatu inovasi di bidang jasa finansial, yang mana istilah tersebut berasal dari kata "*financial*" dan "technology" (*FinTech*) yang merujuk pada inovasi finansial deengan sentuhan teknologi modern (Sukma 2016). Konsep FinTech itu mengadaptasi perkembangan teknologi dipadukan dengan bidang finansial pada lembaga perbankan, sehingga diharapkan dapat memfasilitasi proses transaksi keuangan yang lebih mudah, aman serta modern, meliputi layanan keuangan berbasis digital yang saat ini telah berkembang di Indonesia, yaitu *payment channel system*, *digital banking, online digital insurance*, *Peer to Peer* (P2P) *Lending*, serta *crowd funding*. Konsep di atas senada dengan apa yang telah disampaikan oleh Deputi Direktur Direktorat Perizinan dan Informasi Perbankan OJK, Tris Yulianta, mengungkapkan bahwa lembaga perbankan perlu memanfaatkan penerapan pada teknologi finansial untuk meningkatkan efisiensi kegiatan operasional dan mutu pelayanan bank kepada nasabahnya, sebab pemanfaatan teknologi finansial tersebut sejalan dengan bertambah berkembangnya kebutuhan masyarakat akan layanan keuangan berbasis online dan penggunaan media internet untuk akses data digital.

Penelitian yang dilakukan oleh Irman Muzdalifa, inayah Auliah Rahma, Bella Gita Novalia (2018) dengan judul "*PERAN FINTECH DALAM MENINGKATKAN KEUANGAN INKLUSIF PADA UMKM DI INDONESIA (PENDEKATAN KEUANGAN SYARIAH)"*.Diamana pada sistem keuangan syariah merupakan sistem keuangan yang menghubungkan antara pihak yang membutuhkan dana dengan pihak yang memiliki dana lebih melalui produk dan jasa keuangan yang sesuai dengan prinsip-prinsip syariah.2 Seluruh transaksi yang terjadi dalam kegiatan keuangan syariah harus dilaksanakan berdasarkan prinsip-prinsip syariah. Prinsip syariah adalah prinsip yang berdasarkan pada ajaran Al-Qur'an dan Sunnah. Dalam konteks Indonesia, prinsip syariah adalah prinsip hukum Islam dalam kegiatan perbankan dan keuangan berdasarkan fatwa yang dikeluarkan oleh lembaga yang memiliki kewenangan dalam penetapan fatwa di bidang syariah. Sistem keuangan syariah didasari oleh dua prinsip utama, yaitu prinsip syar'i dan prinsip tabi'i.

Di antara prinsip-prinsip syariah dalam sistem keuangan yaitu :

1. Kebebasan bertransaksi, namun harus didasari prinsip suka sama suka dan tidak ada pihak yang di zalimi dengan didasari oleh akad yang sah.
2. Bebas dari maghrib (maysir, gharar, haram, riba).

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

3. Bebas dari upaya mengendalikan, merekayasa dan memanipulasi harga.
4. Semua orang berhak mendapatkan informasi yang berimbang, memadai, dan akurat agar bebas dari ketidaktahuan dalam bertransaksi.
5. Pihak-pihak yang bertransaksi harus mempertimbangkan kepentingan pihak ketiga yang mungkin dapat terganggu, oleh karenannya pihak ketiga diberikan hak atau pilihan.
6. Transaksi didasarkan pada kerja sama yang saling menguntungkan dan solidaritas.
7. Setiap transaksi dilaksanakan dalam rangka mewujudkan kemaslahatan manusia.
8. Mengimplementasikan zakat.

Sedangkan prinsip-prinsip tabi'i adalah prinsip-prinsip yang dihasilkan melalui interpretasi akal dan ilmu pengetahuan dalam menjalankan bisnis seperti manajemen permodalan, dasar dan analisis teknis, manajemen cash flow, manajemen resiko dan lainya. Dengan demikian, sistem keuangan syariah diformulasikan dari kombinasi dua kekuatan sekaligus, pertama prinsip-prinsip syar'i yang diambil dari AlQur'an dan Sunnah dan kedua prinsip-prinsip tabi'i yang merupakan hasil interpretasi akal manusia dalam menghadapi masalah-masalah ekonomi seperti manajemen, keuangan, bisnis dan prinsip-prinsip ekonomi lainya yang relevan. Diamana pertubuhan pada bisnis sector UMKM ini cukup pesar dan meningkatkan kinerja perbankkan syariah.

2.2 *Fintech* Syariah

Fintech berasal dari istilah *Fintech* berasal dari istilah *financial technology* atau teknologi finansial. Menurut *The NationalDigital Research Centre* (NDRC), di Dublin, Irlandia, mendefinisikan *fintech* sebagai " *innovation infinancial services*" atau "inovasi dalam layanan keuangan fintech" yang merupakan suatu inovasi padasektor finansial yang mendapatsentuhan teknologi modern.Transaksi keuangan melalui fintech ini meliputipembayaran,investasi, peminjaman uang, transfer,rencana keuangan dan pembanding produk keuangan. Fintech dengan layanan keuangan seperti *crowdfunding*, *mobile payments*, dan jasa transfer uang menyebabkan revolusi dalam bisnis *startup*. Dengan crowdfunding, bisa memperoleh dana dari seluruh dunia dengan mudah, bahkan dari orang yang belum pernah ditemui sekalipun *Fintech* juga memungkinkan transfer uang secara global atau internasional. Jasa pembayaran seperti PayPal otomatis mengubah kurs mata uang, sehingga yang berada di Amerika bisa membeli barang dari Indonesia dengan mudahnya, *Fintech* juga memiliki peran penting dalam mengubah perilaku dan ekspektasi konsumen diantaranya :

a. Dapat mengakses data dan informasi kapan saja dan dimana saja
b. Menyamaratakan bisnis besar dan kecil sehingga cenderung untuk memiliki ekspektasi tinggi meski terhadap bisnis kecil yang baru dibangun

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

Fintech secara *global* menunjukkan secara pesat *fintech* berkembang di berbagai sektor, mulai dari *startup* pembayaran, peminjaman (*lending*), perencanaan keuangan (*personal finance*), investasi ritel, pembiayaan (*crowdfunding*), remitansi, riset keuangan, dan lain-lain. Pelaku *fintech* saat ini di Indonesia masih dominan berbisnis *payment* (43%), pinjaman (17%), dan sisanya berbentuk agregator, *crowdfunding* dan lain-lain. Sering fintech saat ini juga di imbangi dengan perkembangan perbankang syariah. Yangan mana Bank Syariah merupakan bank yang menjalankan kegiatan usahannya berdasarkan prinsip syariah dan menurut jenisnya terdiri atas Bank Umum Syariah dan Bank Pembiayaan Rakyat Syariah. Saat ini, pertumbuhan Bank Syariah sangat signifikan. Hal itu berdasarkan pada laporan Statistik perbankan syariah yang ditunjukkan dalam Tabel 2.1.

Tabel 2.1 Pertumbuhan Perbankan Syariah Di Indonesia

Indikator	Tahun						
	2011	2012	2013	2014	2015	2016	2017
Total Aset Bank Umum Syariah	116.930	147.581	180.360	204.961	213.423	254.184	288.027
Total Aset Unit Usaha Syariah	28.536	47.437	61.916	67.383	82.839	102.320	136.154

Sumber: Otoritas Jasa Keuangan (OJK)

Dari data yang ada diatas, dapat diketahui bahwa total aset Bank Umum Syariah dan Unit Usaha Syariah menunjukkan pertumbuhan yang signifikan. Pada tahun 2011 total aset Bank Umum Syariah sebesar 116.930, angka tersebut mengalami peningkatan yang signifikan pada setiap tahun dan mencapai 288.027 pada tahun 2017. Itu berarti dalam hal syariah pun ini tetap akan menjadi sebuah prospek bisnis yang akan maju ke depannya. Dimana *fintech* syariah yang akan di buat ini akan menargetkan pasar syariah yang transparan dan tentunya dalam tuntunan syariat Islam. (Irma Muzdalifa, 2018)

2.2.1 Peminjaman Dan Investasi Syariah

Peminjaman dalam hal syariah ini akan berasaskan dalam sistem bagi hasil. Yang mana akan menggunakan akad sebuah syarat dalam melakukan peminjaman maupaun berinvestasi. Dimana akan keterdiri dari 3 jenis dalam akad yaitu:

a. Akad *Murabahah* merupakan akad jual beli suatu barang dengan menegaskan harga belinya kepada pembeli dan pembeli membayamya dengan harga yang lebih sebagai laba.

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

b. *Musyarakah* ialah akad kerjasama antara dua pihak atau lebih untuk suatu usaha tertentu, dimana masing-masing pihak memberikan kontribusi dana dengan ketentuan bahwa keuntungan dibagi berdasarkan kesepakatan sedangkan kerugian berdasarkan porsi kontribusi dana berupa kas maupun asset non- kas yang diperkenankan oleh syariah

c. *Mudharabah* adalah bentuk kerja sama antara dua atau lebih pihak di mana pemilik modal (*shahibul amal*) mempercayakan sejumlah modal kepada pengelola (*mudharib*) dengan suatu perjanjian di awal. Bentuk ini menegaskan kerja sama dengan kontribusi seratus persen modal dari pemilik modal dan keahlian dari pengelola.

Dari akad-akad yang ada pada syariat islam tersebut yang mana akan menjadi pedoman dalam pembuatan *fintech* berbasis syariah nantinya. (Maslahat, 2018)

2.3 Android

Android merupakan sistem operasi open source yang telah dikembangkan oleh Google untuk perangkat mobile. Aplikasi Android dapat dikembangkan melalui Android SDK (Android *Standart Development Kit*) dengan menggunakan bahasa java dan Google menyediakan banyak library yang dapat membantu *developer* dalam mengembangkan program pada Android. Perangkat Android sendiri terdiri dari Sistem Operasi, *middleware,* dan *key application*. Pengambangan software Android menggunakan bahasa pemprograman jva dan aplikasi Android tidak berjalan langsung di atas kernel sistem operasu namun aplikasi tersebut berjalan diatas dalvik *virtual machine* yang khusus di optimasikan untuk perangkat mobile.

Adapun arsitektur Android terdiri dari 5 Layer, yaitu [SP-12]:

1. *Layer Application*, layer dimana pada aplikasi tersebut berjalan seperti email client, kalender, peta, dan lain-lain.
2. *Layer Framework*, layer yang berisikan framework API yang digunakan oleh *developer* dalam membangun aplikasiuntuk *platform* Android.
3. *Layer Runtime*, layer dimana library yang digunakan oleh berbagai komponen dalam sistem Android tersimpan.
4. *Layer Runtime*, layer yang berisi berbagai core librabries yang menyediakan sebagian besar fungsionalitas yang serupa dengan API pada pemprograman Java.
5. Pada OS Android terdapat tiga sensor dasar, yaitu:
6. Sensor gerak, merupakan sensor yang mengukur setiap pergerakan perpindahan maupun perputaran yang bertumpu pada tiga sumbu (x, y, z). Beberapa yang termasuk sensor *accelerometer*, sensor gravitasi, sensor perputaran vector.
7. Sensor *environmental*, merupakan sensor yang mendeteksi tentang hal-hal menyangkut kondisi disekitar seperti suhu udara, penerangan, dan

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

kelembabn. Termasuk dalam kategori ini adalah terminator, barometer, dan photometer.

8. Sensor posisi, yaitu sensor yang berfungsi mengukur posisi secara fisik dari perangkat sendiri. Termasuk dalam sensor orientasi dan *magnetometer*.

Terdapat beberapa hardware yang didukung oleh Android yang dapat digunakan pada sebagian besar perangkat dengan OS Android. Diantaranya adalah [AND-15]:

1. Prosesor yang mendukung adalah ARM Architecture (ARMv7 atau setelahnya bahkan android 5.0 mendukung ARMv8-A) baik 32bit maupun 64bit. Selain itu sejak tahun 2012 Android juga mendukung intel untuk digunakan sebagai prosesor dari perangkat yang digunakan sebagai prosesor dari perangkat yang digunakan.
2. RAM (*Random Access Memory*) yang mendukung bebagai jenis kapasitas. Namun pada tahun 2013 mualai dari Android vesi 4.4 (Kitkat) RAM minimal yang digunakan adalah 512 MB.
3. GPU (*Grapich Processing Unit*) yang digunakan pada perangka-perangkat Android adalah OpenGL ES 1.1, 2.0, 3.0, dan 3.1.
4. Kamera dengan berbagai jenis kemampuan pengambilan gambar maupun fitur yang telah tersedia pada sebuah perangkat android.
5. Layar sentuh dengan berbagaiukuran dan resolusi dan dapat me-render tampilan 2D dan 3D.
6. Berbagai jenis perangkat sensor tambahan yang dapat meningkatkan perforna dan fungsi dari perangkat android.

Beberapa keunggulan Android adalah:

1. *Open Source*, sehingga siapa saja bebas untuk melakukan pengembangan aplikasi menggunakan platform Android tanpa harus membayar licence untuk developer.
2. Android dikembangkan oleh Google yang merupakan salah satu perusahaan paling besar, paling kaya, dan paling inovatif di dunia teknologi saat ini, sehingga mempunyai banyak sponsor yang akan mendukung para pengembangan aplikasi Android.
3. Aplikasi Android sangat mudah untuk dipasarkan dari pada aplikasi pada *platform* lain. Hal ini yang paling penting bagi pada pengembang aplikasi *mobile* baik pemula ataupun professional.

2.4 *Firebase*

Firebase adalah platform terpadu Google yang menyatukan berbagai fitur canggih untuk aplikasi, termasuk *backend* seluler, analisis, dan fitur pertumbuhan serta monetisasi. Yang mana implementasi pada *firebase* ini akan banyak di aplikasikan dalam beberpaa lini *platform* yang akan di buat nantinya. sehingga memudahkan pengembang dalam membuat sebuah aplikasi. (android, 2018)

2.4.1 *Firebase Realtime Database*

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

Firebase Realtime Database merupakan *cloud database*. Data disimpan dalam format JSON dan akan disinkronkan secara realtime pada setiap klien yang terhubung. Ketika membangun aplikasi hybrid lintas platform, seperti Android dan iOS maka semua klien berbagi satu *instance realtime database* dan secara otomatis menerima pembaruan dengan data tertentu. *Firebase Realtime Database* adalah basis data NoSQL dan karena itu memiliki optimalisasi dan fungsionalitas yang berbeda dibandingkan dengan basis data relasional. Membuat database Firebase bisa melalui import file JSON ke konsol Firebase, atau dapat juga dibuat langsung melalui halaman konsol *Realtime Database* secara manual. . (Mirza Ilhami, 2017)

2.4.2 *Firebase Storage*

Firebase Storage digunakan untuk menyimpan data seperti gambar, audio dan video. Sebagian aplikasi pasti menggunakan *file* gambar, audio atau video. Adanya *Firebase Storage* sangat memudahkan proses unggahan dan unduhan untuk aplikasi. Beberapa kelebihan utama dari Firebase Storage adalah sebagai berikut:

a. *Strong*

   Firebase Storage dapat melakukan unggahan dan unduhan *file* di semua kualitas jaringan internet. Ia dapat berhenti atau melanjutkan, sehingga menghemat waktu dan *badwidth* pengguna.

b. *Secure*

   Firebase Storage terintegrasi dengan *Firebase Authentication* untuk menyediakan autentikasi yang mudah dan intuitif untuk pengembang. Model keamanan dapat diatur berdasarkan nama *file*, ukuran, tipe konten dan metadata lainnya.

c. *Scalable*

   Firebase Storage didukung oleh *Google Cloud Storage* hingga skala petabyte. Ini sangat penting, ketika aplikasi yang dari prototipe kemudian berkembang viral dengan jutaan pengguna.

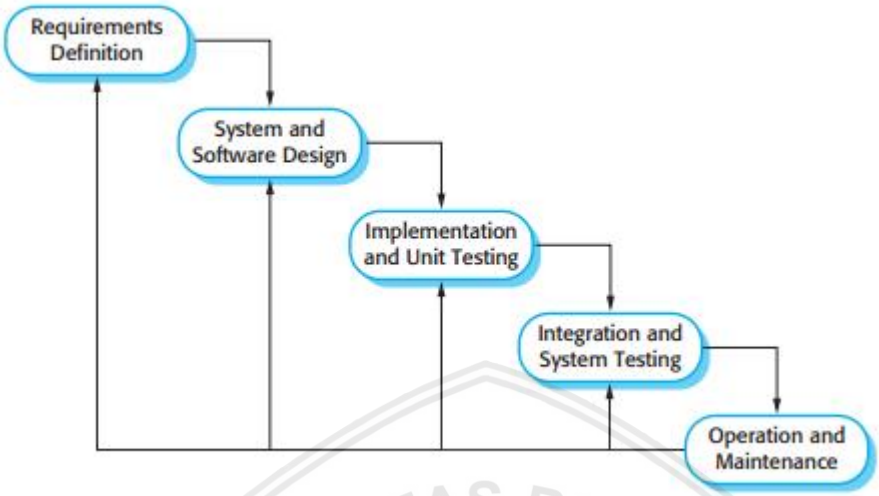
Proses unggahan ke *storage* dapat dilakukan melalui aplikasi pengguna menggunakan AngularJS. Ini lebih mudah diimplementasikan karena tidak perlu menggunakan pemrograman berbasis server seperti PHP atau ASP. Ketika file telah berhasil diunggah ke *storage*, maka akan mengembalikan link unduhan (*download link*) yang dapat disimpan ke *Firebase Database*. (Mirza Ilhami, 2017)

2.5 *Waterfall Model*

Tahapan yang dilakukan pada penggunan utama dari *waterfall model* langsung mencerminkan aktivitas pengembangan dasar. Terdapat 5 tahapan pada *waterfall model,* yaitu *requirement, analysis and definition, system and software*

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

*design, implementation and unit testing, integration and system testing,*dan *operation and maintenance*(Sommervile,2011).



Gambar 2.1 Langkah-langkah *Waterfall Model* (Sommervile,2011)

Berikut ini penjelasan tahapan-tahapan dari *waterfall model*:

1. *Requirement Analysis and Definition*

Merupakan sebuah tahapan penentuan fitur, analisa kebutuhan sistem, kendala pembuatan dan tujuan sistem yanag akan melalui konsultasi dengan pengguna sistem.

2. *System and Software Design*

Merupakan tahapan pembentukan arsitektur sistem,perancangan algortima,serta perancangan antarmuka sistem berdasarkan kebutuhan yang telah ditetapkan pada tahapan sebelumnya.

3. *Implementation and Unit Testing*

Merupakan tahapan hasil yang dilakukan dari rancangan perangkat lunak untuk direalisasikan sebagai satu set program atau unit program.

4. *Integration and System Testing*

Merupakan tahapan untuk melakukan integrasi pada setiap unit program satu sama lain dan diuji sebagai satu sistem yang lengkap untuk memastikan sistem sudah memenuhi persyaratan yang sudah ditetapkan.

5. *Operation and Maintenance*

Merupakan tahapan untuk instalasi dan penerapan sistem. Untuk tahap ini juga dilakukan pengujian pada saat sistem dijalankan. Yang mana

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

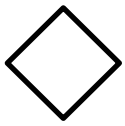
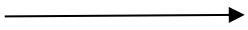
diperuntukkan untuk menemukan dan memperbaiki kesalahan yang tidak ditemukan pada tahap pembuatan.

2.6 Pemodelan Proses Bisnis

Proses bisnis merupakan sebuah kumpulan aktivitas yang dijalankan secara koordinasi pada lingkungan organisasional dan lingkungan teknis. Aktivitas-aktivitas itu akan bersama-sama mencapai tujuan pada sebuah bisnis. Setiap proses bisnis ditetapkan oleh satu organisasi (bagian), namun dapat berinteraksi dengan proses yang dijalankan pada organisasi (bagian) lainnya. (Weske, 2007).

BPMN bertujuan untuk memberikan semua pengguna bisnis dengan notasi yang mudah dipahami, dimulai dari analis bisnis dalam pembuatan awal konsep proses, pengembangteknologi yangbertanggung jawab untuk menerapkan prosesyang ada, dan proses pengelolaan serta memonitornya. Oleh sebab itu BPMN berfungsi sebagai jembatan antara perancangan proses bisnis dengan implementasi proses bisnis (Weske, 2007).

Tabel 2.2 Pertumbuhan Perbankan Syariah Di Indonesia

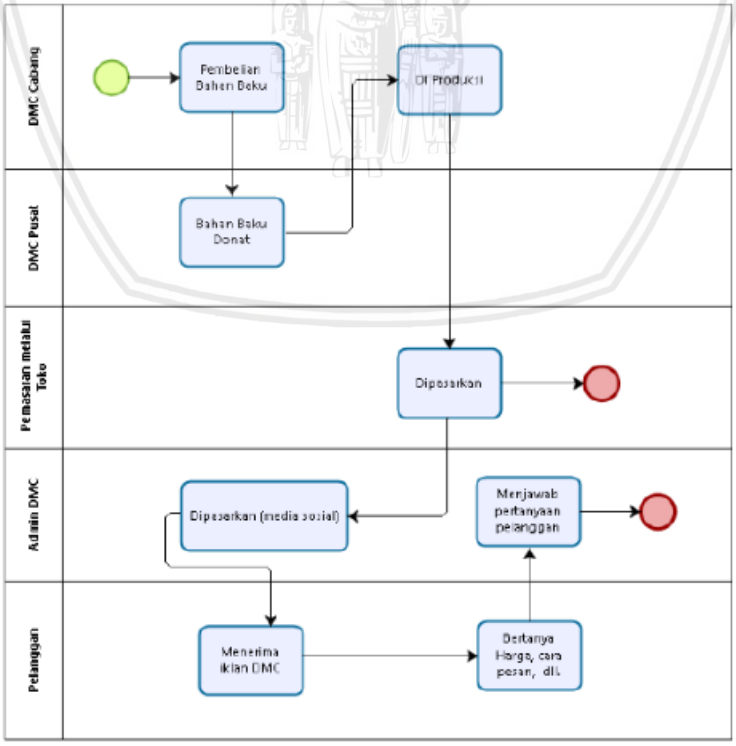
Elemen	Deskripsi	Notasi
Event	Merupakan kejadian yang akan berjalan ada proses bisnis. Dan akan mempengaruhi aliran pada proses bisnis dan memiliki sebuah hasil.	
Activity	Merupakan sebuah notasi yang dapat menjelaskan pekerjaan apa saja yang harus dilakukan pada sebuah proses bisnis pada sebuah perusahaan	
Gateway	Merupakan sebuah notasi untuk memper-sentasikan sebuah percabangan serta pengabungan dari sebuah path yang ada pada sebuah proses bisnis	
Sequence Flow	Merupukan sebuah notasi untuk menje-laskan urutan aktivitas yang akan di jalankan pada sebuah entitas	

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

Tabel 2.3 Pertumbuhan Perbankan Syariah Di Indonesia (lanjutan)

Elemen	Deskripsi	Notasi
Massage Flow	Merupakan sebuah notasi untuk menjelaskan pertukaran pesan yang akan terjadi pada entitas yang berbeda	
Pool	Merupakan sebuah notasi untuk merepresentasikan peserta dalam proses. Juga sebagai batasan dari kegiatan pool lainnya.	
Lane	Merupakan sebuah notasi untuk partisi sub-dalam swimlane dan akan memperpanjang seluruh panjang pool, baik secara vertikal atau horizontal, jalur digunakan untuk mengatur dan mengkategorikan kegiatan.	

Sumber: *Bussiness* Process *Model and Notation* (BPMN) *version* 2.0, 2011



Gambar 2.2 Contoh Proses Bisnis Produksi dan Pemasaran (Heryanto,2018)

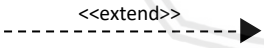
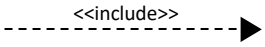
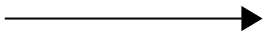
2.7 *Use Case Diagram*

Use case menggambarkan sebuah sistem dan berapa aktor yang akan menggunakan, terdapat beberapa *use case*, serta interaksi antara sistem dengan aktor, dan bagaimana hubungan antar keduanya. *Use case* yaitu permodelan untuk kelakuan (*behavior*) sistem informasi yang akan dibuat nantinya. (Network 2016).

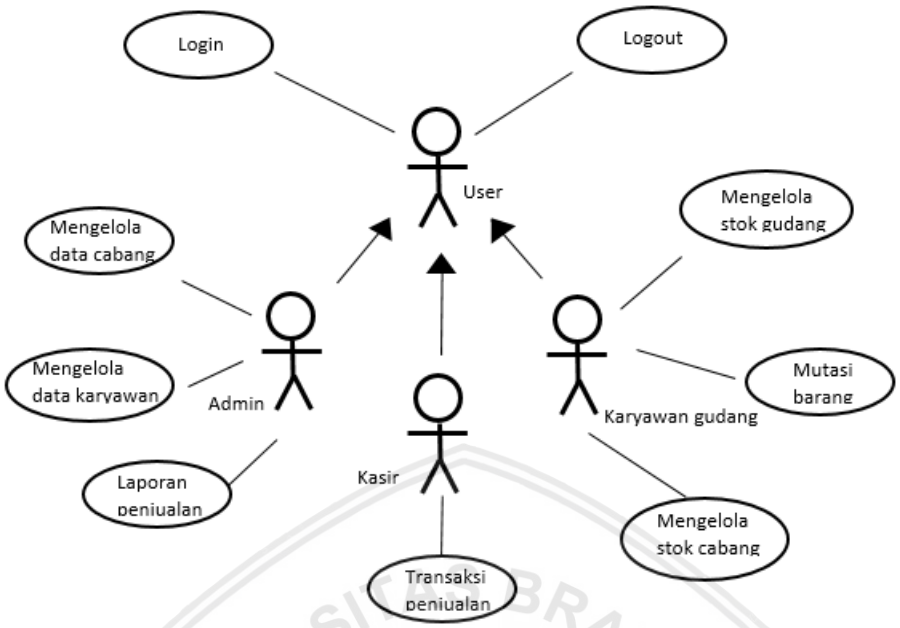
Tujuan utama dari *use case* adalah membantu tim pengembang sistem untuk memvisualisasikan kebutuhan fungsional sistem, termasuk hubungan antara aktor yang berinteraksi dengan sistem serta hubungan antar *use case* berbeda. (Bell, 2013)

Dengan kata lain *use case diagram* digunakan untuk menjelaskan bagaimana sebuah sistem dapat bekerja beserta interaksinya dengan dunia di luar sistem. (Schmuller, 2004)

Tabel 2.4 Elemen *Use Case Diagram*

No	Simbol Elemen	Nama Elemen	Deskripsi Elemen
1		*Actor*	Merepresentasikan seseorang atau sesuatuyang berinteraksi dengan sistem.
2		*Use case*	Gambaran fungsionalitas dari suatu sistem, agar pengguna paham mengenai kegunaan sistem yang dibangun.
3		*Association*	Menghubungkan link antar elemen.
4		*Extends*	Menunjukkan bahwa suatu use case merupakan tambahan fungsional dari use case lainnya jika suatu kondisi terpenuhi.
5		*Include*	Menunjukkan bahwa suatu use case seluruhnya merupakan fungsionalitas dari use case lainnya.
6		*Generalized*	Menunjukkan spesialisasi actor untuk dapat berpatisipasi dengan use case.

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA



Gambar 2.3 Contoh Use Case Sistem Informasi Monitoring Penjualan dan Stok Barang (Hendini, 2016)

2.8 *Activity Diagram*

Acitivity diagram merupakan diagram yang menggambarkan pada alur dari suatu proses bisnis atau alur pada sistem, dari awal hingga akhir (Network, 2016). *Activity diagram* menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis atau daftar yang ada pada perangkat lunak (Sukamto dan Shalahuddin 2013:161). *Activity diagram* juga banyak digunakan untuk mendefinisikan hal-hal berikut :

- Rancangan proses bisnis dimana setiap urutan aktivitas yang digambarkan merupakan proses bisnis sistem yang dideifnisikan.
- Urutan atau pengelompokan tampilan dan sistem / *user interface*, dimana setiap aktivitas dianggap memiliki sebuah rancangan antarmuka tampilan..
- Rancangan menu yang ditampilkan pada perangkat lunak.

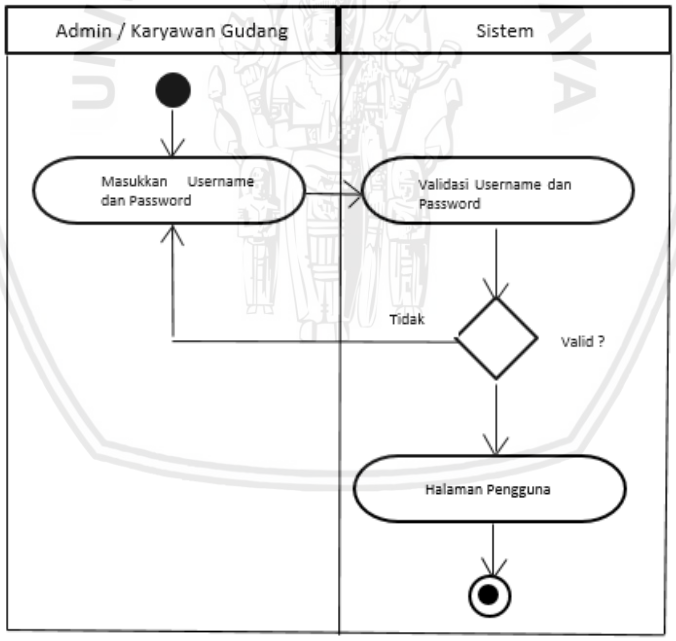
Tabel 2.5 Simbol-simbol Pada *Activity* Diagram (Sugiarti, 2013)

Nama	Simbol	Deskripsi
Status awal	●	Status awal aktivitas sistem.
Aktivitas	aktivitas	Aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja.

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

Tabel 2.6 Simbol-simbol Pada *Activity* Diagram (Sugiarti, 2013) (lanjutan)

Nama	Simbol	Deskripsi
Percabangan (*Decision*)		Asosiasi percabangan dimana jika ada pilihan aktivitas lebih dari satu.
Penggabungan (*join*)		Asosiasi penggabungan dimana lebih dari suatu aktivitas digabungkan menjadi satu.
Status akhir		Status akhir yang dilakukan sistem.
Swimlane	nama swimlame	Memisahkan organisasi bisnis yang bertanggung jawab terhadap aktivitas yang terjadi.



Gambar 2.4 Contoh Activity Diagram Login Admin/Karyawan Gudang (Hendini, 2016)

2.9 *Class Diagram*

Class diagram menggambarkan struktur sistem dari segi pendefinisan kelas-kelas yang akan dibuat untuk membangun sistem. Kelas memiliki apa yang disebut atribut dan metode atau operasi. *Class diagram* mendefiniskan jenis-jenis objek dalam sistem dan berbagai hubungan statis yang terdapat diantara mereka

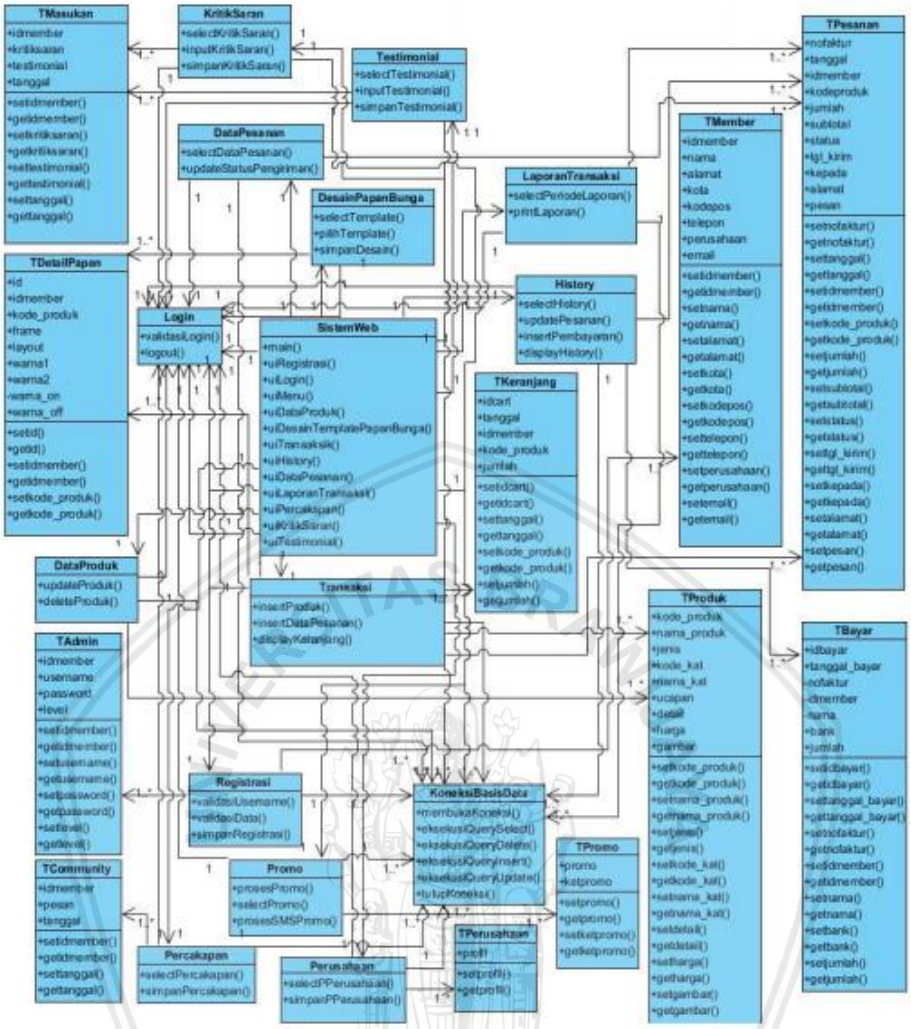
repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

(Sugiarti, 2013). Tujuan digambarkannya *class diagram* adalah supaya pengembang perangkat lunak dapat membuat program sesuai dengan batase yang terdapat pada class diagram. Berikut adalah simbol-simbol pada class diagrampada Tabel 2.7.

Tabel 2.7 Simbol-simbol Pada *Class* Diagram (Sugiarti, 2013)

No	Gambar	Nama	Keterangan
1		*Generalisation*	Hubungan dimana objek anak (*descendent*) berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk (*ancestor*).
2		*Nary Association*	Upaya untuk menghindari asosiasi dengan lebih dari 2 objek.
3		*Class*	Himpunan dari objek-objek yang berbagi atribut serta operasi yang sama
4		*Collaboration*	Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu *actor*.
5		*Realization*	Operasi yang benar-benar dilakukan oleh suatu objek.
6		*Dependency*	Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri (*independent*) akan mempengaruhi elemen yang bergantung padanya.
7		*Association*	Apa yang menghubungankan antara objek lainnya.

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA



Gambar 2.5 Contoh *Class Diagram Login* Pemesanan Dan Penjualan Pada Dewi Florist (Setiay, 2013)

2.10 Pengujian *White Box*

Pengujian *white box* merupakan sebuah pengujian yang didasarkan pada pengecekan terhadap rincian pada perancangan, menggunakan struktur kontrol dari desain program secara prosedural untuk membagi pengujian ke dalam beberapa kasus pengujian .Pengujian *white box* berfokus pada struktur control program. Test case dilakukan untuk memastikan bahwa semua statemen pada program sudah dieksekusi paling tidak satu kali selama pengujian dan bahwa semua kondisi logis telah diuji. Pengujian basic path, tehnik pengujian *white box*, menggunakan grafik (matriks grafiks) untuk melakukan serangkaian pengujian yang *independent* secara linear yang akan memastikan cakupan. *White box testing* disebut juga pengujian *glass box*. (Presman, 2010).

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

2.10.1 Pengujian Unit

Pengujian unit yaitu proses pengujian secara invidual terhadap *subprogram, subroutine* atau prosedur pada sebuah program (J.Myers). Pengujian ini berfokus kepada bagian terkecil sebuah program. Tujuan dari pengujian ini adalah memastikan bahwa fungsi telah dibuat dan akan di lakukan integrase dengan fungsi lainnya dapat berjalan baik dalam arti fungsi tersebut menghasilkan keluaran *(output)* yang benar. Selain itu untuk pengembang dari aplikasi juga dapat mengidentifikasi kesalahan sejak dini.

2.10.2 Pengujian Integrasi

Pengujian Integrasi merupakan teknik pengujian yang difokuskan pada gabungan unit-unit atau modul-modul yang akan membentuk satu kesatuan fungsional. Pengujian ini didasarkan pada informasidari deskripsi perancangan awal perangkat lunak. Pengujian ini gunakan untuk menemukan kesalahan antarmuka antar modul. Pada pengujian ini umumnya dilakukan oleh pengembang itu sendiri atau dilakukan antar pengembang lainnya. Dimana pengujian ini biasanya dilakukan dengan cara *white-box* dan *black-box*. Pada pengujian integrasi dikenal juga istilah integrasi *non-incremental* dan integrasi *incremental*. Integrasi *non-incremental* yaitu proses integrasi yang menggunakan cara penggabungan langsung pada modul-modul yang akan terlibat. Kemudian program akan di uji secara keseluruhan. Hal ini dapat menyebabkan terjadinya kesalahan yang cukup kompleks dikarekan pada perkembangan program yang besar. Sedangkan pada integrasi *incremental* yaitu integrasi yang dilakukan dengan bertahap. Pengujian ini juga dilakukan persegmen sehingga kesalahan didapatkan akan mudah diisolasi dan diperbaiki. (Wibisono, 2002)

2.11 Pengujian *Black Box*

Black box testing merupakan pengujian perangkat lunak yang berfokus pada spesifikasi fungsional dari perangkat lunak. Diamana kasus uji dapat mendefinisikan kumpulan kondisi untuk masukan dan melakukan pengetesan pada spesifikasi fungsional dari sebuah program. *Black box testing* bukanlah sebuah solusi alternatif dari *white box testing* akan tetapi lebih merupakan pelengkap untuk menguji hal-hal yang tidak dapat dicakup oleh *white box testing*. (Mustqbal, 2015)

Selain itu pegujian *black box* juga merupakan suatu pendekatan untuk melakukan pengujian pada perangkat lunak dari segi spesifikasi fungsional tanpda ada menguji desai dank ode program yang telah dibuat. Pengujian tersebut juga memiluki tujuan untuk mengetahui semua fungsi-fungsi, masukan, maupun keluaran dari sebuah perangkat lunak sudah sesuai dengan psesifikasi yang telah dibutuhkan. (S & Shalahudin, 2016)

2.11.1 Pengujian Validasi

Pada pengujian validasi merupakan sebuah pengujian yang mana termasuk ke dalam metode *black box.* Dimana pengujian ini menunjuk kepada sebuah

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

kumpulan berbeda dari aktivitas yang memastikan bahwaperangkat lunak yang telah dibangun dapat ditelusuri terhadap kebutuhan yang telah di definisikan.

Pengujain validasi memiliki maksud dalam hal untuk memasikan seluruh kebutuhan yang telah dipaparkan sebelumnya telah diimplementasikan pada sebuah perangkat lunak yang telah dibangun. Pengujian validasi dikatakan berhasil bila semua fungsionalitas yang sudah paparkan dapat berjalan sesua dengan yang diharapkan. (Pressman, 2015)

2.11.2 Pengujian *Compatibility*

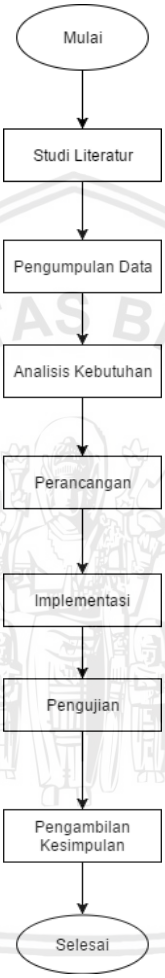
Pada pengujian *compatibility* adalah sebuah pengujain yang bertujuan untuk mengetahui sistem yang telah dibangun atau dikembangkan berjalan dengan baik jika dijalankan pada sistem operasi, perangkat, *browser*, dan lain-lain secara berbeda. (Pressman, 2015)

Pada pengujian *compatibility* yang dilakukan pada penelitian kali ini akan membandingkan pada sistem operasi Android dengan versi yang berbeda satu sama lain. Dimana terkadang pada setiap versi yang berbeda sebuah tampilan aplikasi bisa saja berbuah atau terdapat masalah. Sehingga aplikasi tidak dapat dijalankan pada versi tertentu. Yang mana pengujian *compatibility* ini dibutuhkan untuk melihat performa aplikasi yang telah dibangun dapat dijalankan pada beberapa versi sistem Android yang berbeda.

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

BAB 3 METODOLOGI

Dalam penelitian ini, terdapat beberapa tahapan yang harus dilakukan seperti studi literature, pengumpulan data, analisis kebutuhan, perancangan, implementasi, pengujian, pengambilan kesimpulan. Yang akan di jabarkan pada sub-bab sub-bab mengenai metodologi yang digunakan. Untuk alur metodelogi yang digunakan dapat dilihat pada Gambar 3.1.



Gambar 3.1 Alur Metodelogi

3.1 STUDI LITERATUR

Studi literature merupakan cara yang digunakan dalam pengumpulan informasi yang diperlukan dengan sumber yang digunakan berupa data-data yang terkait. Dalam studi literature dikumpulkan data-data dari sumber yang terpercaya yang menjadikan sumber tersebut menjadi dasar dari teori yang akan diterapkan lebih lanjut pada sistem yang dibuat. Studi literature yang digunakan :

1. Proses bisnis peminjaman syariah
2. Perancangan aplikasi android
3. Penggunaan *firebase realtime database*

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

4. Rekayasa Perangkat Lunak

3.2 PENGUMPULAN DATA

Pada tahap ini membutuhkan data-data untuk proses implementasi ke dalam sistem. Pengumpulan data didapatkan dari CSR yang diberikan pada PT Maslahat Amanah Umat, data dari beberapa buku tentang mekanisme sebuah bisnis *fintech*, dan juga pengetahuan umum tentang bisnis *fintech* investasi yang divalidasi terlebih dahulu oleh pakar. Yang nantinya hasil dari pengumpulan data tersebut dapat di lakukan beberapa proses yang akan di lakukan dalam penelitian ini.

3.3 ANALISIS KEBUTUHAN

Pada tahap ini merupakan proses mendapatkan kebutuhan yang dilakukan untuk merancang perangkat lunak yang akan di buat. Kebutuhan yang didapatkan akan dikembangkan menjadi *use case.* Yang nantinya akan dijelaskan detailnya ke dalam *use case scenario.* Di mana akan mengulas dari setiap *use case* yang telah dibuat

Analisis kebutuhan sistem akan dilakukan dengan melihat bisnis proses yang sudah dilakukan pada PT Maslahat Amanah Umat. Sehingga nantinya akan di baut pada proses bisnis model. Yang dapat dilihat dengan jelas proses bisnis yang terjadi pada nantinya akan seperti apa.

3.4 PERANCANGAN

Tahap ini merupakan lanjutan dari analisis, dimana mulai merancang sistem menggunakan pemodelan *Unified Modelling Language (UML)*. Perancangan sistem ini meliputi beberapa tahapan yaitu, dengan pembuatan *activity diagram ,class diagram,* perancangan basis data *,* perancangan antarmuka dan perancangan aplikasi.

3.5 IMPLEMENTASI

Tahap ini dimana data yang sudah diperoleh dari pakar dan beberapa sumber lain ke dalam sistem yang akan dibuat. Implementasi dilakukan dengan menggunakan bahasa pemrograman *java,* dan menggunakan *firebase* untuk basis datanya, penyimpanan, ataupun *authentication*. Serta dugunakan bahas XML untuk membuat antarmuka.

3.6 PENGUJIAN

Tahap ini bertujuan untuk menunjukkan sistem aplikasi yang telah dibuat dapat b ekerja sesuai dengan fungsinya. Pengujian bertujuan untuk mengetahui kesalahan pada sistem. Pengujian yang dilakukan yaitu pengujian unit, pengujian integrasi, pengujian validasi.

Pengujian unit menguji setiap bagian terkecil dari perangkat lunak. Pengujian unit dilakukan dengan menggunakan metode *white box testing* dan diuji

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

menggunakan jenis pengujian unit. Pengujian integrasi dilakukan untuk menguji hubungan antar modul dalam sistem dan kesesuain spesifikasi kebutuhan dengan sistem yang telah dibuat. Pengujian integrasi menggunakan metode *white box testing.* Pengujian validasi dilakukan apakah semua fungsi yang ada pada sistem sudah berjalan sesuai. Pengujian validasi menggunakan metode *black box testing.* Serta pengujian *compatibility* untuk melihat performa dari aplikasi dari lingkungan yang berbeda dengan menggunakan metode *black box.*

3.7 PENGAMBILAN KESIMPULAN

Pada tahap ini, pengambilan kesimpulan didasarkan pada hasil analisa pada tahap perancangan. penerapan dan pengujian sistem berhasil dilakukan. Ditahap ini, rumusan masalah mengenai pembuatan sebuah aplikasi percetakan dengan pada platform andorid, analisis perancangan sistem dan implementasi data pakar dan pengetahuan ke dalam sistemi akan terjawab. Selain itu untuk menjawab rumusan masalah, tahap ini juga bertujuan untuk mengetahui kekurangan dalam penelitian agar dapat menjadi referensi bagi penelitian selanjutnya dalam mengembangkan penelitian ini lebih lanjut.

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

BAB 4 REKAYASA KEBUTUHAN

Pada bab ini akan membahas tentang proses rekayasa kebutuhan yang digunakan dalam penelitian pengembangan aplikasi berbasis android untuk menghubungkan mitra usaha dan investor yang berbasis syariah pada pt maslahat amanah umat .Pada rekayasa kebutuhan perangkat lunak ini digunakan untuk memenuhi kebutuhan yang didapat dari pengguna maupun perusahaan tersebut. Dalam rekayasa kebutuhan dibagi menjadi proses rekayasa kebutuhan dan pemodelan kebutuhan. Proses rekayasa kebutuhan terdiri dari gambaran umum sistem, proses bisnis, identifikasi aktor,batasan sistem, lingkungan operasi dan spesifikasi kebutuhan. Dalam pemodelan yang dilakukan terdiri dari pemodelan *use case diagram* dan *use case scenario*.

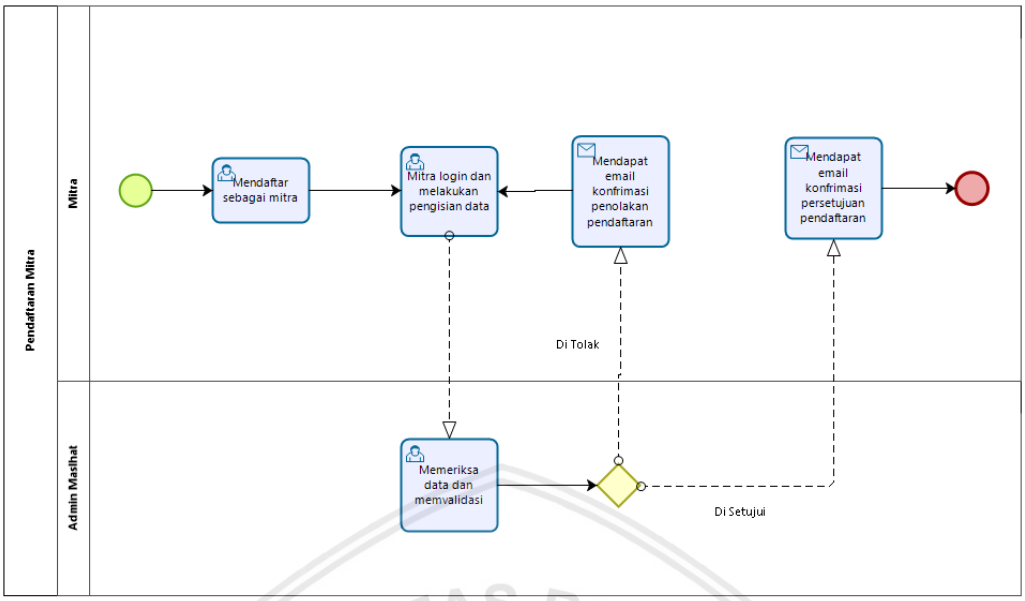
4.1 Gambaran Umum Sistem

Sistem yang dibangun dalam penelitian ini yaitu merupakan sebuah perangkat lunak dengan judul "MaslahatKita". Yang mana pada perangkat lunak ini merupakan pengembangan yang dilakukan pada PT Maslahat Amanah Umat. Pada sistem ini bertujuan untuk mempermudah hubungan antara investor dengan mitra memalui aplikasi berbasis android ini. Sehingga pada aplikasi ini digunakan *p2p lending* untuk mempertemukan mitra dan investor dengan menggunakan aplikasi MaslahatKita. Yang mana mitra dapat mengajukan usahanya dengan mudah dan juga investor dapat mencari mitra usaha untuk di danai nantinya dengan mudah. Pada proses perjanjian investasi dengan aplikasi ini digunakannya akad murabahah pada syariat islam. Pada sistem ini digunakannya bahasa Java dalam implementasinya dan juga menggunakan *realtime database* serta *authentifikasi* dengan *firebase*. Penggunaan *firebase* ini ditunjukan juga agar pengguna nantinya dapat menggunakan aplikasi ini dimanapun dengan konsep *client-server.* Yang mana pengguna harus tersambung dengan internet agar dapat menggunakan aplikasi tersebut.

4.2 Proses Bisnis

Pada aplikasi ini menggunakan proses bisnis dengan berbagai macam proses bisnis yang di jalankan. Dimana proses bisnis tersebut akan memuat konsep *p2p lending.* Diantarnya yaitu:

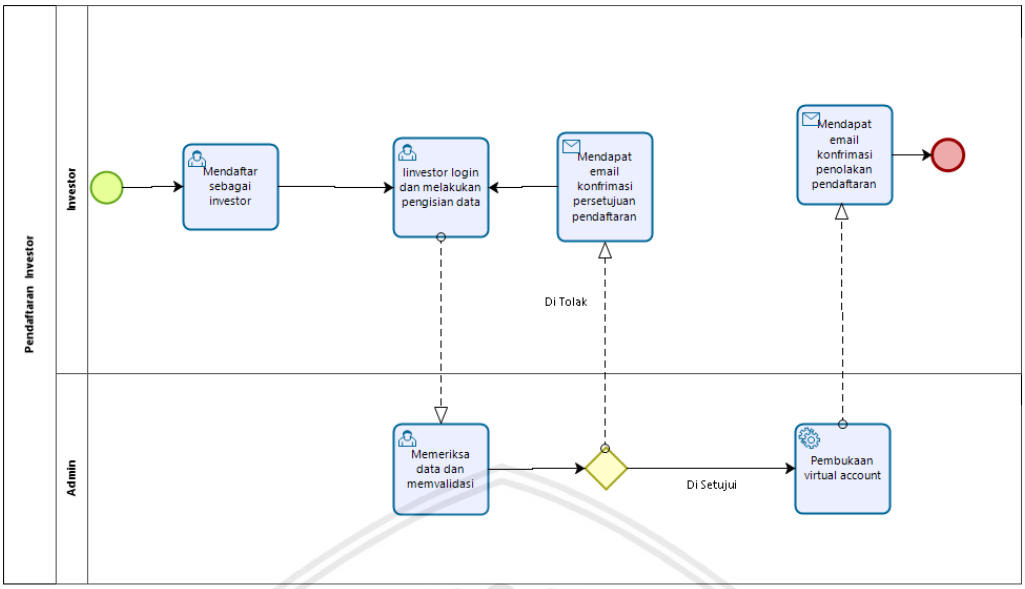
repository.ub.ac.id
UNIVERSITAS BRAWIJAYA



Gambar 4.1 Proses Bisnis Pendafataran Mitra

Pada proses bisnis gambar 4.1 merupakan proses binis Pendafataran Mitra. Dimana pada proses bisnis tersebut akan di lakukkan oleh pengguna ketika ingin mendaftar sebagai mitra usaha pada PT Maslahat Amanah Umat. Untuk itu alur proses bisnis yang di lakukkan yaitu:

1. Pengguna melakukan pendaftaran dengan memilih menjadi mitra
2. Setelah itu pengguna akan *login* dan mengisi data-data yang ada untuk mengajukan sebagai mitra usaha yang *valid*.
3. Kemudian admin akan memeriksa data yang ada untuk memvalidasinya.
4. Jika data yang masuk terdapat kesalahan maka admin akan melakukan penolakan dan admin akan mengirim *email* kepada pengguna.
5. Setelah pengguna menerima *email* maka pengguna dapat memperbaiki keselahan yang ada.
6. Jika data yang masuk sudah benar maka sistem dapat mengirim *email* persetujuan.
7. Kemudian mitra usaha akan menerima *email*.

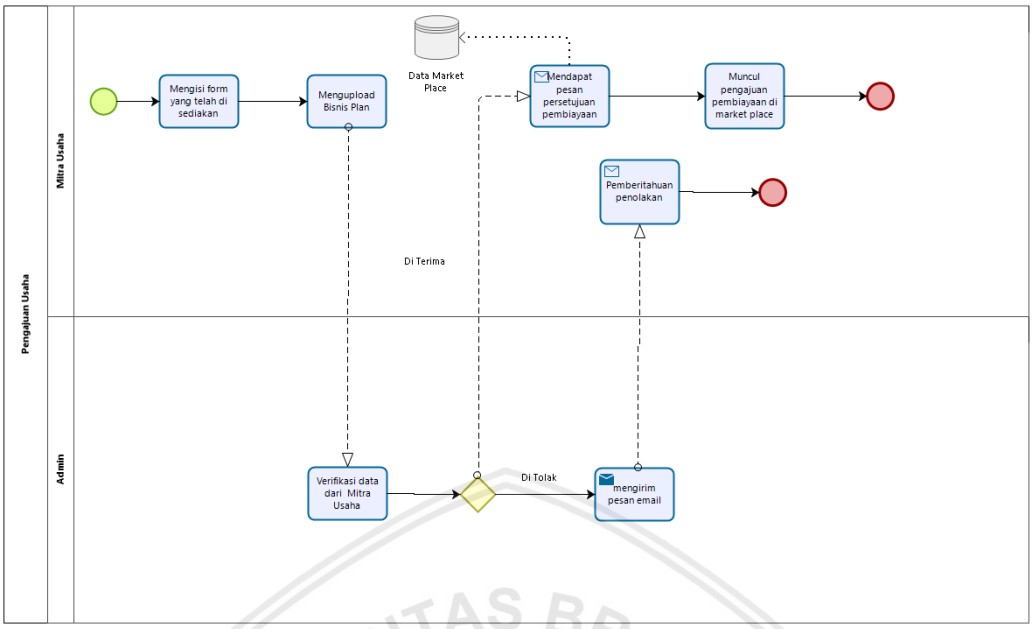


Gambar 4.2 Proses Bisnis Pendafataran Investor

Pada proses bisnis gambar 4.2 merupakan proses binis Pendafataran Investor. Dimana pada proses bisnis tersebut akan di lakukkan oleh pengguna ketika ingin mendaftar sebagai investor pada PT Maslahat Amanah Umat. Untuk itu alur proses bisnis yang di lakukkan yaitu:

1. Pengguna melakukan pendaftaran dengan memilih menjadi investor
2. Setelah itu pengguna akan *login* dan mengisi data-data yang ada untuk mengajukan sebagai investor yang *valid*.
3. Kemudian admin akan memeriksa data yang ada untuk memvalidasinya.
4. Jika data yang masuk terdapat kesalahan maka admin akan melakukan penolakan dan admin akan mengirim *email* kepada pengguna.
5. Setelah pengguna menerima *email* maka pengguna dapat memperbaiki keselahan yang ada.
6. Jika data yang masuk sudah benar maka sistem akan membuatkan investor sebuah virtual *account* dan sistem juga akna mengirim *email* persetujuan kepada investor
7. Kemudian investor akan menerima *email*.

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

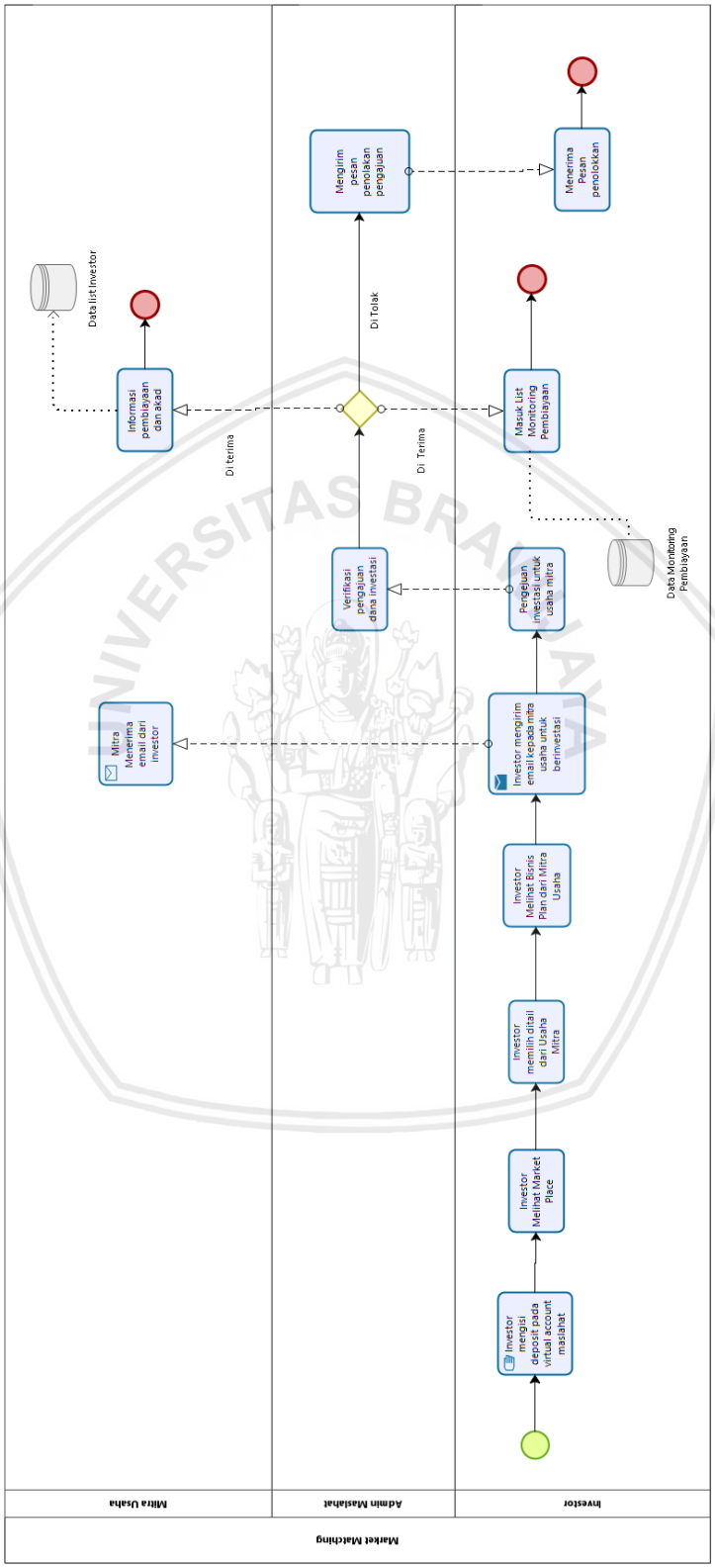


Gambar 4.3 Proses Bisnis Pengajuan Usaha

Pada gambar 4.3 merupakan proses bisnis untuk melakukan pengajuan usaha yang akan di lakukan mitra. Dimana pada proses tersebut akan digunakan untuk melakukan pengajuan dana untuk usaha yang dilakukan mitra. Untuk itu alur proses bisnis yang di lakukkan yaitu:

1. Dimana pada proses tersebut mitra harus mengisi form yang telah tersedia untuk melakukan pengajuan usaha.
2. Kemudian mitra harus mengupload rencana bisnis mereka. Di karenakan untuk memperlihatkan kepada investor. Agar investor lebih percaya akan usaha yang dilakukan mitra dan investor akan memperoleh ke untungan pada jangka waktu yang ditentukan nantinya.
3. Dimana setelah memalukan hal tersebut maka semua data yang telah dikirim akan di perikas kelayakannya oleh admin dari PT Maslahat Amanah Umat.
4. Kemudan bila disetujui oleh admin maka mitra akan menerima email bahwa pengajuan usaha yang dikirimkan telah di setujui
5. Admin maka pengajuan usaha tersebut akan di munculkan di market place.
6. Akan tetapi bila pengajuan tersebut di tolak maka mitra akan menerima pesan adanya penolakan melalui email. Yang berisi tentang

apa saja yang kurang atau kesalahan yang di lakukan oleh mitra dalam mengajukan usaha.



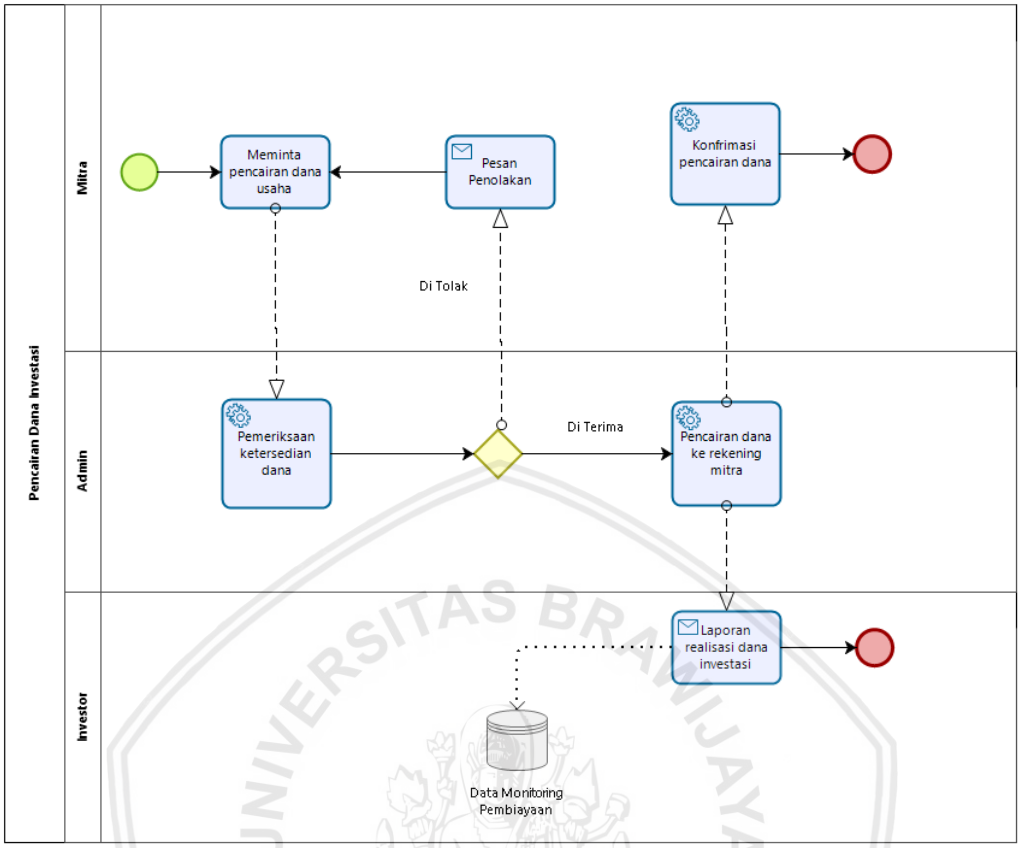
Gambar 4.4 Proses Bisnis *Market Matching*

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

Untuk proses bisnis pada gambar 4.4 merukapan *market matching* yang berarti proses untuk melakukan investasi pada usaha mitra. Pada proses bisnis tersebut melalui beberapa tahapan agar investor dapat melakukan pendanaan untuk mitra. Untuk itu alur proses bisnis yang di lakukkan yaitu:

1. Yang mana proses awal dilakukan oleh investor ialah investor haruslah mengisi deposit ke PT Maslahat Amanah Umat dengan mengirim dana pada bank tertentu.
2. Setelah deposit pada aplikasi telah terisi. Maka investor dapat memilih usaha pada mitra yang ingin melakukkan pendanaan atau berinvestasi pada usaha yang di miliki mitra.
3. Sebelum melakukan pendanaan maka investor dapat melihat detail usaha dari mitra melalui aplikasi. Dan juga dapat melihat proposal atau *business plan* yang telah mitra usaha *upload* pada aplikasi MaslahatKita.
4. Jika investor tertarik pada salah satu usaha makan investor dapat mengirim email ke pada mitra sebelum investor mengajukan investasinya kepada usaha tersebut.
5. Kemudian investor akan mengajukan pendanaan kepada usaha yang telah di pilihnya kemuda akan di verifikasi oleh admin MaslahatKita.
6. Admin MaslahatKita akan memeriksa pengajuan pembiayaan yang dilakukan oleh investor.
7. Jika admin menolak akan pendanaan yang di ajukan pada investor, maka akan mengirim pesan penolakkan kepada investor.
8. Apabila pengajuan pendanaan tersebut di setujui, maka akan masuk pada list monitoring usaha. Dan juga untuk mitra akan mendapatkan informasi pembiayaan.

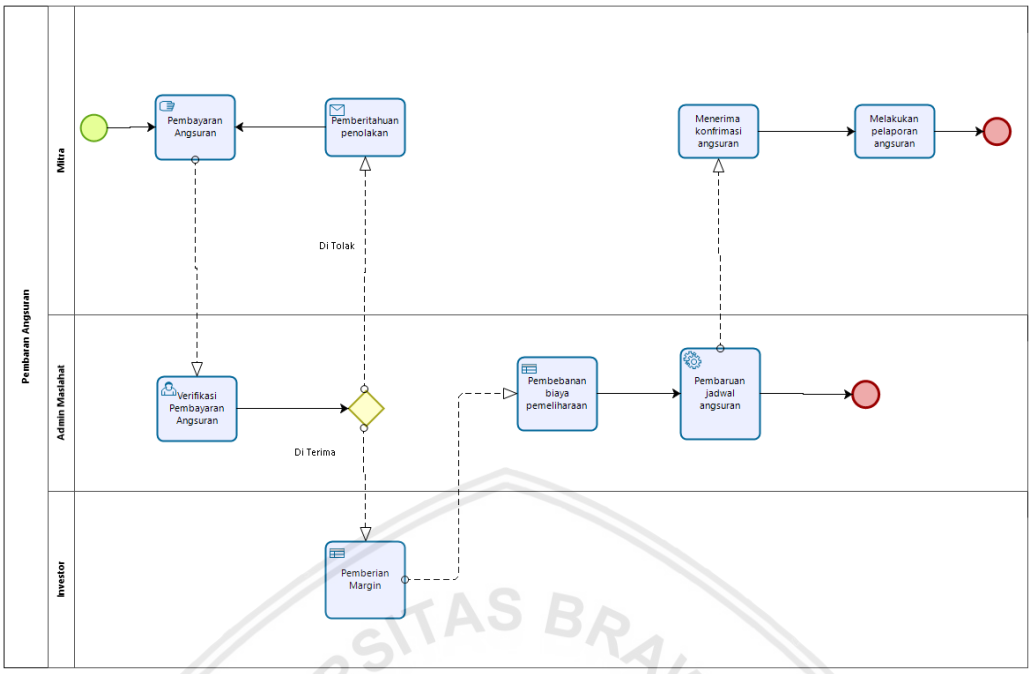
repository.ub.ac.id
UNIVERSITAS BRAWIJAYA



Gambar 4.5 Proses Bisnis Realisasi Dana

Pada proses bisnis gambar 4.5 merupakan proses binis realisasi dana. Proses bisnis tersebut merupakan sebuah proses binis untuk mitra dapat merealisasikan dana yang telah investor investasikan kepada usaha yang dilakukan pada mitra tersebut. Untuk itu alur proses bisnis yang di lakukkan yaitu:

1. Dimana mitra nantinya akan melaukkan pengambilan dana usaha yang akan di periksa ketersediaannya oleh sistem.
2. Bila dana tidak tersedia maka mitra akan menerima pesan penolakan untuk proses pencairan dana usaha tersebut.
3. Akan tetapi bila di setejui untuk pencairan dana maka sistem akan mengirimkan dana usaha tersebut dari deposit MaslahatKita kepada mitra yang meminta pencairan dana.
4. Nantinya mitra akan menerima konfrimasi dana yang sudah di cairkan ke rekening mitra.
5. Sedangkan investor juga akan menerima pesan realisasi dana yang sudah di lakukan oleh mitra. Dan kemudian akan masuk pada database untuk nantinya dapat di monitoring oleh investor.



Gambar 4.6 Proses Bisnis Pengangsuran Dana Usaha

Pada proses bisnis gambar 4.6 merupakan proses binis pengansuran dana. Dimana pada proses bisnis tersebut akan di lakukkan oleh mitra dalam kewajibannya sebagai mitra usaha yang melakukan pengangsuran sesuai jadwal waktu yang telah di tentukan. Yang di lakukan pada pengangsuran ini menggunakan akad murabahah. Dimana akad tersebut merupakan akad bagi hasil yang telah di sepakati. Untuk itu alur proses bisnis yang di lakukkan yaitu:

1. Nantinya mitra akan melakukan transfer dana pengangsuran dana yang telah di sepakati serta memberikan margin yang telah di sepakati. Bila mitra tidak dapat membayar pada tanggal yang telah di sepakati maka akan ditunggakan pada bulan berikutnya.
2. Setelah itu dana pengangsuran akan di verifikasi oleh admin MaslahatKita.
3. Bila pengangsuran tersebut di tolak maka mitra akan penerimaan pesan penolakan dan harus membayarkan yang sudah tercantum pada akad yang sudah di sepakati.
4. Jika pengangsuran tersebut di setujui maka investor nantinya akan menerima dana pengembalian beserta margin bila ada pada bulan tersebut sebuah keuntungan yang di hasilkan oleh mitra.
5. Dari margin yang telah di sepakati juga akan di bagi kepada admin, sebagai biaya pemeliharan sistem.

6. Dan nantinya sistem akan melakukan pembaruan pada jadwal angsuran untuk bulan depan.

Pada setiap proses bisnis yang ada maka akan di berikan kode PBM-XX, di mana PBM merupakan singkatan Proses Binis MaslahatKita, XX menunjukan penomeran untuk setiap proses bisnis. Dan akan di jabarkan pada tabel 4.1 setaip proses bisnis yang ada.

Tabel 4.1 Daftar Proses Bisnis

No.	Kode Proses Bisnis	Nama	Deskripsi
1	PBM-01	Proses Bisnis Pendaftaran Mitra	Pada proses bisnis tersebut dilakukan oleh mitra, dan admin. Akan tetapi proses ini akan dipusatkan pada proses pendaftaran sebagai mitra usaha yang dilakukan oleh mitra.
2	PBM-02	Proses Bisnis Pendaftaran Investor	Pada proses bisnis tersebut dilakukan oleh investor, dan admin. Akan tetapi proses ini akan dipusatkan pada proses pendaftaran sebagai investor yang dilakukan oleh investor.
3	PBM-03	Proses Bisnis Pengajuan Usaha	Pada proses bisnis tersebut dilakukan oleh mitra, dan admin. Akan tetapi proses ini akan dipusatkan pada proses pengajuan dana usaha yang dilakukan oleh mitra.
4	PBM-04	Proses Bisnis Market Matching	Pada proses bisnis tersebut dilakukan oleh mitra, admin, serta investor. Akan tetapi proses ini akan dipusatkan pada proses pengajuan investasi yang dilakukan oleh investor.
5	PBM-05	Proses Bisnis Realisasi Dana	Pada proses bisnis tersebut dilakukan oleh mitra, admin, serta investor. Akan tetapi proses ini akan dipusatkan pada proses pencairan dana yang dilakukan oleh mitra dengan beberapa tahapan secara *online* ataupun *offline*.

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

Tabel 4.2 Daftar Proses Bisnis (lanjutan)

No.	Kode Proses Bisnis	Nama	Deskripsi
6	PBM-06	Proses Bisnis Pengangsuran Dana Usaha	Pada proses bisnis tersebut dilakukan oleh mitra, admin, serta investor. Akan tetapi proses ini akan dipusatkan pada proses pencairan dana yang dilakukan oleh mitra dengan beberapa tahapan secara *online* ataupun *offline.*

4.3 Analisis Kebutuhan

Analisis kebutuhan adalah tahapan awal dalam melakukan proses pengembangan perangkat lunak. Pada proses ini akan dilakukan elisitasi kebutuhan, identifiaksi actor apa saja yang terlibat, mendefinisasikan kebutuhan fungsional dan non-fungsional beserta spesifikasinya. Nantinya kebutuhan-kebutuhan yang telah dipaparkan akan dimodelkan dengan pemodelan kebutuhan menggunakan *Unified Modeling Language* (UML) berupa use case diagram.

4.3.1 Elisitasi Kebutuhan

Elisitasi kebutuhan merupakan sekumpulan aktivitas yang ditujukan untuk menemukan kebutuhan suatu sistem dan menyesuaikan domain permsalahan pada sistem. Pada elisitas kebutuhan ini digunakan untuk lebih memahami dari permasalahan dalam menentukan nilai kompleksitas yang ada pada sebauh perangkat lunak yang sedang di kembangkan. Nantinya, pada permasalahan tersebut akan didapatka solusi-solusi untuk membantu penyelesaian permasalahan yang ada.

Teknik yang di gunakan pada penelitian ini digunakan wawancara langsung pada direktur IT dan perencanaan bisnis PT Masalahat Amanah Umat. Agar dapat secara langsung mengetahui proses bisnis dan permasalahan yang ada pada PT Maslahat Amanah Umat.

4.3.2 Identifikasi Aktor

Identifikasi aktor adalah sebauh tahapan untuk melakukan identifikasi terhadap actor apa saja yang terlibat dalam sistem MasalahatKita. Tahap identifikasi aktor dilakukan dengan cara mendefinisikan aktor beserta Karakteristik Aktor. Tabel 4.2 menunjukkan aktor yang terdapat dalam sistem MaslahatKita beserta deskripsi pada setiap aktornya.

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

Tabel 4.3 Daftar Identifikasi Aktor

No	Identifikasi Aktor	Karakteristik Aktor
1	Mitra	Mitra merupakan aktor yang dapat mengajukan pendanaan pada usahanya, dapat mencarikan daan usaha, serta melakukan pengasuran dengan tenor yang telah disepakati.
2	Admin MaslahatKita	Admin MaslahatKita merupakan aktor yang memverifikasi beberapa kebutuhan yang ada pada mitra maupun investor yaitu: verifikasi sebagai mitra usaha, verifikasi pengajuan usaha, verifikasi pengangsuran dana, verifikasi investor, dan juga vefikasi investasi.
3	Investor	Investor merupakan aktor dapa melakukan sebuah investasi pada usaha yang di jalankan pada mitra, juga dapat pencariran dana dari hasil investasi, dan monitoring investasi.

4.4 Batasan Sistem

Batasan-batasan yang digunakan pada pengembangan sistem ini adalah sebagai berikut.

a. Mitra maupun investor yang mendaftar harus menunggu verifikasi dari admin MaslahatKita.
b. Aturan keuntungan yang di sediakan sudah terdapat pada sistem sebanyak 15%.

4.5 Lingkungan Operasi

Agar dapat berfungsi dengan baik, sistem ini dapat diaskses menggunanakan peramban web dan dengan spesifikasi perangkat lunak sebagai berikut.

a. DBMS : *Realtime Database* FireBase
b. Bahasa : Java
c. *Operating System* : Minimal Android 4.4 (*Kitkat*)

4.6 Daftar Kebutuhan

Daftar Kebutuhan adalah suatu kebutuhan yang harus disediakan oleh sistem. Kebutuhan tersebut dibedakan menjadi dua jenis yaitu kebutuhan fungsional maupun kebutuhan non fungsional. Setiap kebutuhan akan diberikan kode MK-X-YYY-ZZ, di mana MK adalah singkatan MaslahatKita , untuk X menunjukkan nomor definisi kebutuhan fungsional atau non fungsional, YYY menunjukkan nomor urutan kebutuhan dan ZZ menunjukkan nomor urutan spesifikasi kebutuhan.

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

4.6.1 Kebutuhan Fungsional

Tabel 4.4 Daftar Kebutuhan Fungsional

No.	Kode Proses bisnis	Kode Kebutuhan	Deskripsi	Nama Use Case
1	PBM-01	MK-1-001	Sistem harus dapat menambahkan pengguna baru dengan mendaftar sebagai mitra.	Pengajuan Sebagai Mitra Usaha
		MK-1-002	Sistem harus dapat memverifikasi data pengajuan sebagai mitra yang diajukan pengguna.	Verifikasi Mitra Usaha
2	PBM-02	MK-1-003	Sistem harus dapat menambahkan penguna baru dengan mendaftar sebagai investor.	Pengajuan Sebagai Investor
		MK-1-004	Sistem harus dapat memverifikasi data pengajuan sebagai investor yang diajukan pengguna.	Verifikasi Investor
3	PBM-03	MK-1-005	Sistem harus dapat mengajukan pembiayaan pada usaha mitra.	Pengajuan Pembiayaan Usaha
		MK-1-005-01	Sistem harus dapat memasukkan proposal pengajuan	
		MK-1-006	Sistem harus dapat memverifikasi data pengajuan pembiayaan usaha yang di lakukan oleh mitra.	Verifikasi Pengajuan Pembiayaan Usaha
4	PBM-04	MK-1-007	Sistem harus dapat melakukan investasi yang dilakukan oleh investor.	Pengajuan Investasi
		MK-1-007-01	Sistem harus dapat melihat proposal pengajuan dana yang di *upload* oleh mitra	

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

Tabel 4.5 Daftar Kebutuhan Fungsional (lanjutan)

No.	Kode Proses bisnis	Kode Kebutuhan	Deskripsi	Nama Use Case
4	PBM-04	MK-1-007-02	Sistem harus dapat mengi-rimkan *email* ke-pada mitra yang ingin diberikan pendanaan	Pengajuan Investasi
		MK-1-008	Sistem harus dapat mem-verifikasi data pengajuan investasi usaha yang di lakukan oleh investor.	Verifikasi Pengajuan Investasi
5	PBM-05	MK-1-009	Sistem harus dapat mela-kukan proses pencairan dana usaha yang telah tersedia dilakukan oleh mitra	Pencairan Dana Usaha
		MK-1-009-01	Sistem harus dapat mem-berikan pelaporan rea-lisasi dana kepada inves-tor	
6	PBM-06	MK-1-010	Sistem harus dapat me-lakukan pengangsuran yang dilakukan oleh mitra	Pengangsuran
		MK-1-011	Sistem harus dapat mem-verifikasi angsuran yang diberikan oleh mitra	Verifikasi Pengangsuran
		MK-1-011-01	Sistem harus dapat mem-bagi margin kepada in-vestor maupun perusa-haan sesuai peraturan yang ada	
		MK-1-011-02	Sistem harus dapat me-lakukan pembaruan pada jadwal angsuran	

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

Tabel 4.6 Daftar Kebutuhan Fungsional (lanjutan)

No.	Kode Proses bisnis	Kode Kebutuhan	Deskripsi	Nama Use Case
6	PBM-06	MK-1-012	Sistem harus dapat menyimpan data pelaporan pengangsuran yang dilakukan oleh mitra	Pelaporan Angsuran
7	-	MK-1-013	Sistem harus dapat melakukan pencairan dana hasil investasi yang dilakukan oleh investor	Pencairan Dana Hasil Investasi

4.6.1 Kebutuhan Non Fungsional

Kebutuhan non fungsional berisi daftar kebutuhan yang dapat dijadikan acuan dalam menentukan kualitas secara keseluruhan dari sistem. Dan juga untuk melihat batasan sistem baik pada lingkungan pengembangan serta oprasional pada sistem tersebut. Daftar kebutuhan non fungsional ditunjukkan pada Tabel 4.7.

Tabel 4.7 Daftar Kebutuhan Non Fungsional

No	Kode Kebutuhan	Kebutuhan Sistem	Parameter
1	MK-2-001	Sistem dapat dijalankan pada Sistem Operasi Android 4.4(*KitKat*), Android 7.0(*Nougat*), dan Android 8.0(*Oreo*).	*Compatibility*

4.7 Pemodelan Kebutuhan

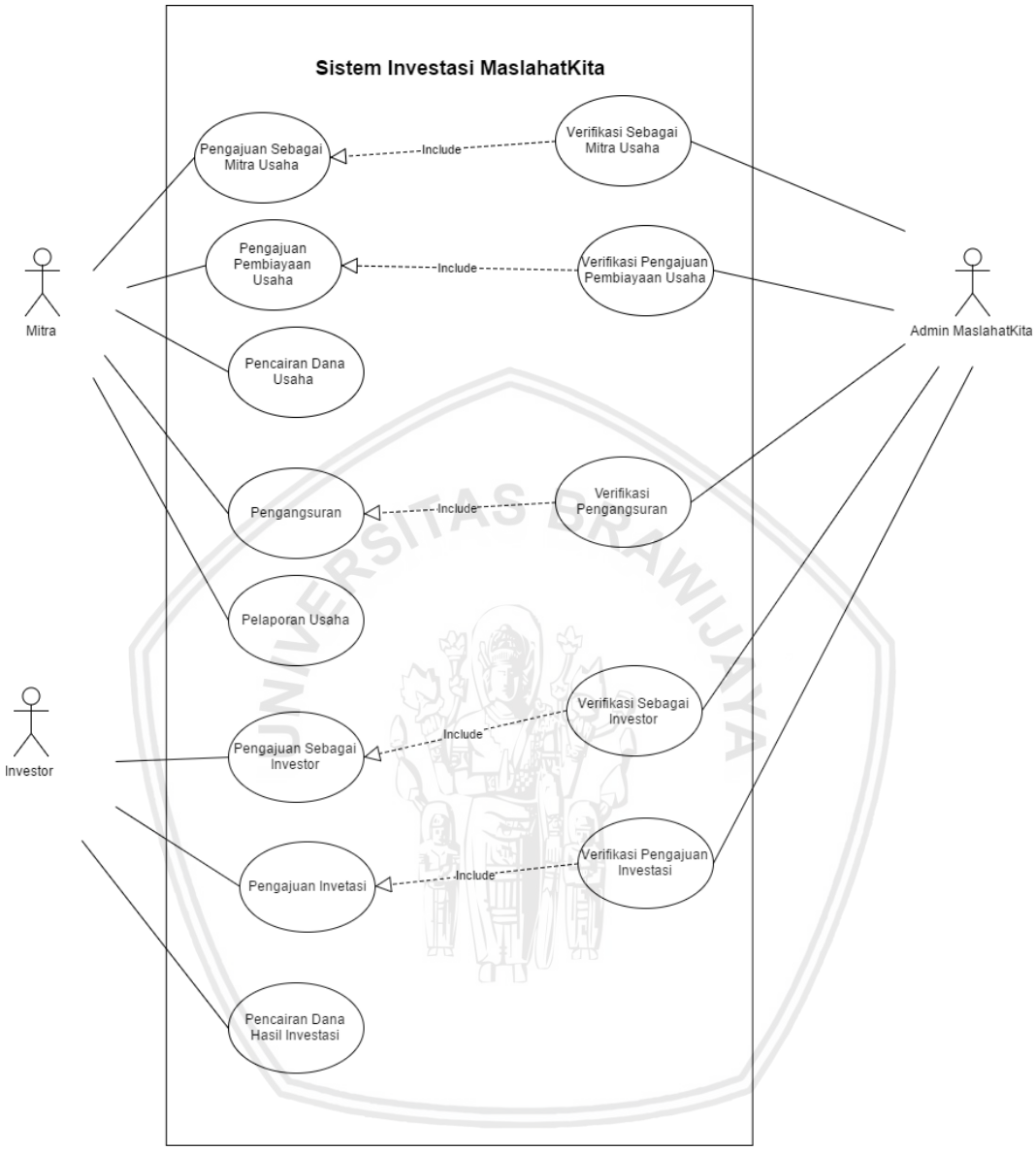
Setelah dilakukan proses rekayasa kebutuhan maka perlu dilakukan pemodelan untuk mempermudah pembacaan kebutuhan dalam tahap perancangan sistem. Proses pemodelan kebutuhan ini akan menggunakan *Unified Modelling Language* (UML) yaitu *use case diagram* yang kemudian akan dijelaskan pada alur sistem yang lebih rinci menggunakan *use case scenario*.

4.7.1 Use Case Diagram

Use case diagram digunakan untuk mengetahui fungsi-fungsi yang dapat dilakukan oleh sistem berdasarkan sudut pandang aktor. Aktor disini adalah pengguna sebagai aktor tunggal dari system. Aktor-aktor tersebut memiliki 13 fitur

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

(*use case*) utama yang dapat dilakukan oleh pengguna kepada sistem. *Use Case Diagram* dapat dilihat pada Gambar 4.7.



Gambar 4.7 *Use Case Diagram*

4.7.2 Use Case Scenario

Use case scenario adalah sebuah uraian dari alur kegiatan yang terjadi saat menggunakan sistem dan dikategorikan menurut masing-masing *use case* yang telah dibuat. *Use case scenario* dari sistem ditunjukkan pada Tabel 4.8 -Tabel 4.25.

1. Pengajuan Sebagai Mitra Usaha

   Use case scenario pengajuan sebagai mitra usaha merupakan penjelasan lebih rinci dari *use case* pengajuan sebagai mitra usaha. Tabel 4.8 menunjukkan *use case scenario* untuk pengajuan sebagai mitra usaha.

Tabel 4.8 *Use Case Scenario* Pengajuan Sebagai Mitra Usaha

Kode Kebutuhan	MK-1-001
Objective	Aktor dapat mendaftarkan sebagai mitra usaha dari PT Masalahat Amanah Umat
Actors	Mitra
Pre-condition	Pengguna hanya sebagai *guest* pada aplikasi MaslahatKita
Main flow	1. Pengguna menekan tombol daftar 2. Sistem menambilkan form daftar sebagai mitra 3. Pengguna mengisi form yang telah tersedia 4. Sistem menyimpan data semetara dari pengguna 5. Pengguna melakukan *login* pada aplikasi 6. Pengguna mengisi form sebagai mitra aktif 7. Pengguna menekan tombol daftar pada form pengejuan untuk melakukan pengajuan 8. Sistem akan meberikan semua data tersebut kepada admin MaslahatKita untuk di verifikasi
Alternative flows	1. Jika form tidak terisi lengkap maka sistem akan minta untuk mengisi keseluruhannya. 2. Jika pengguna menekan tombol *cancel* maka tidak akan data tidak akan di masukkan ke dalam sistem
Post-condition	Data pengajuan sebagai mitra berhasil dikirim oleh mitra

2. Verifikasi Pengajuan Sebagai Mitra Usaha

Use case scenario verifikasi pengajuan sebagai mitra usaha merupakan penjelasan lebih rinci dari *use case* verifikasi pengajuan sebagai mitra usaha. Tabel 4.9 sampai 4.10 menunjukkan *use case scenario* untuk verifikasi pengajuan sebagai mitra usaha.

Tabel 4.9 *Use Case Scenario* Verifikasi Pengajuan Sebagai Mitra Usaha

Kode Kebutuhan	MK-1-002
Objective	Aktor sudah mendaftarkan sebagai mitra usaha dari PT Masalahat Amanah Umat
Actors	Admin MaslahatKita

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

Tabel 4.10 *Use Case Scenario* Verifikasi Pengajuan Sebagai Mitra Usaha (lanjutan)

Pre-condition	Admin belum melakukan verifikasi pada mitra yang membelum mendapatkan status aktif
Main flow	1. Admin akan menekan tombol verifikasi 2. Admin memeriksa data pengajuan sebagai mitra 3. Jika admin menyetujui pengajuan tersebut maka admin akan menekan tombol *approve* 4. Maka sistem akan mengirim *email* kepada mitra bahwa akun yang telah diajukan sudah terdaftar
Alternative flows	1. Jika admin menolak untuk pengajuan tersebut maka admin akan menekan tombol *reject.* 2. Kemudian sistem akan menampilkan form untuk mengisi pesan yang akan dikirimkan kepada pengguna yang mengajukan 3. Setelah itu admin menekan tombol *Okay* bila ingin mengirim pesan tersebut serta menolak pengajuan tersebut 4. Jika pengguna menekan tombol *cancel* maka tidak akan di lakukan verifikasi pada pengguna tersebut
Post-condition	Data pengajuan sebagai mitra berhasil diverifikasi oleh admin

3. Pengajuan Sebagai Investor

Use case scenario pengajuan sebagai investor merupakan penjelasan lebih rinci dari *use case* pengajuan sebagai investor. Tabel 4.11 sampai 4.12 menunjukkan *use case scenario* untuk pengajuan sebagai mitra.

Tabel 4.11 *Use Case Scenario* Pengajuan Sebagai Investor

Kode Kebutuhan	MK-1-003
Objective	Aktor dapat mendaftarkan sebagai mitra usaha dari PT Masalahat Amanah Umat
Actors	Investor
Pre-condition	Pengguna hanya sebagai *guest* pada aplikasi MaslahatKita
Main flow	1. Pengguna menekan tombol daftar 2. Sistem menambilkan form daftar sebagai investor

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

Tabel 4.12 *Use Case Scenario* Pengajuan Sebagai Investor (lanjutan)

Main flow	3. Pengguna mengisi form yang telah tersedia 4. Sistem menyimpan data semetara dari pengguna 5. Pengguna melakukan login pada aplikasi 6. Pengguna mengisi form sebagai investor aktif 7. Pengguna menekan tombol daftar pada form pengejuan untuk melakukan pengajuan 8. Sistem akan meberikan semua data tersebut kepada admin MaslahatKita untuk di verifikasi
Alternative flows	1. Jika form tidak terisi lengkap maka sistem akan minta untuk mengisi keseluruhannya. 2. Jika pengguna menekan tombol cancel maka tidak akan data tidak akan di masukkan ke dalam sistem
Post-condition	Data pengajuan sebagai mitra berhasil dikirim oleh investor

4. Verifikasi Pengajuan Sebagai Investor

Use case scenario verifikasi pengajuan sebagai investor merupakan penjelasan lebih rinci dari *use case* verifikasi pengajuan sebagai investor. Tabel 4.13 sampai 4.14 menunjukkan *use case scenario* untuk verifikasi pengajuan sebagai investor.

Tabel 4.13 *Use Case Scenario* Verifikasi Pengajuan Sebagai Mitra Usaha

Kode Kebutuhan	MK-1-004
Objective	Pengguna berhasil terverifikasi sebagai Investor usaha dari PT Masalahat Amanah Umat
Actors	Admin MaslahatKita
Pre-condition	Admin belum melakukan verifikasi pada investor yang membelum mendapatkan status aktif
Main flow	1. Admin akan menekan tombol verifikasi 2. Admin memeriksa data pengajuan sebagai investor 3. Jika admin menyetujui pengajuan tersebut maka admin akan menekan tombol *approve* 4. Maka sistem akan membuat virtual *account* dan mengirim *email* kepada investor bahwa akun yang telah diajukan sudah terdaftar

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

Tabel 4.14 *Use Case Scenario* Verifikasi Pengajuan Sebagai Mitra Usaha (lanjutan)

Alternative flows	1. Jika admin menolak untuk pengajuan tersebut maka admin akan menekan tombol *reject.* 2. Kemudian sistem akan menampilkan form untuk mengisi pesan yang akan dikirimkan kepada pengguna yang mengajukan 3. Setelah itu admin menekan tombol *Okay* bila ingin mengirim pesan tersebut serta menolak pengajuan tersebut 4. Jika admin menekan tombol *cancel* maka tidak akan di lakukan verifikasi pada pengguna tersebut
Post-condition	Data pengajuan sebagai investor berhasil diverifikasi oleh admin

5. Pengajuan Pembiayaan Usaha

Use case scenario pengajuan pembiayaan usaha merupakan penjelasan lebih rinci dari *use case* pengajuan pembiayaan usaha. Tabel 4.15 menunjukkan *use case scenario* untuk pengajuan pembiayaan usaha.

Tabel 4.15 *Use Case Scenario* Pengajuan Pembiayaan Usaha

Kode Kebutuhan	MK-1-005
Objective	Mitra berhasil melakukan mengajukan pembiayaan pada usahanya
Actors	Mitra
Pre-condition	Mitra usaha akan melakukan pengajuan pembiayaan usaha
Main flow	1. Mitra mengisi form yang telah di sediakan 2. Kemudian mitra akan mengupload bisnis *plan* yang telah di buat. 3. Kemudian mitra menekan tombol daftar untuk mekalukan pengajuan pembiayaan
Alternative flows	1. Jika mitra menekan tombol *cancel* maka tidak akan di lakukan pengajuan pembiayaan
Post-condition	Data pengajuan pembiayaan berhasil terdaftar

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

6. Verifikasi Pengajuan Pembiayaan Usaha

Use case scenario verifikasi pengajuan pembiayaan usaha merupakan penjelasan lebih rinci dari *use case* verifikasi pengajuan pembiayaan usaha. Tabel 4.16 menunjukkan *use case scenario* untuk verifikasi pengajuan pembiayaan usaha.

Tabel 4.16 *Use Case Scenario* Verifikasi Pengajuan Sebagai Mitra Usaha

Kode Kebutuhan	MK-1-005
Objective	Admin berhasil melakukan verifikasi terhadap pengajuan pembiayaan usaha yang dilakukan oleh mitra
Actors	Admin MaslahatKita
Pre-condition	Admin belum melakukan verifikasi pada pengajuan pembiayaan yang dilakukan oleh mitra usaha
Main flow	1. Admin akan menekan tombol verifikasi pengajuan pembiayaan 2. Admin memilih pengajuan yang diinginkan 3. Admin memeriksa data pengajuan pemebiayaan yang di lakukan oleh mitra 4. Jika admin menyetujui pengajuan tersebut maka admin akan menekan tombol *approve* 5. Mitra usaha akan menerima pesan persejutuan pembiayaan dan juga data pembiayaan dimasukkan kedalam *market place* 6. Sistem akan menampilkan permintaan pembiayaan didalam *market place* dan juga
Alternative flows	1. Jika admin menolak untuk pengajuan tersebut maka admin akan menekan tombol *reject.* 2. Kemudian sistem akan menampilkan form untuk mengisi pesan yang akan dikirimkan kepada mitra yang mengajukan 3. Setelah itu admin menekan tombol *Okay* bila ingin mengirim pesan tersebut serta menolak pengajuan tersebut 4. Jika admin menekan tombol *cancel* maka tidak akan di lakukan verifikasi pada pengajuan pembiayaan tersebut
Post-condition	Data pengajuan pembiayaan usaha berhasil divalidasi oleh admin

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

7. Pengajuan Investasi

Use case scenario pengajuan investasi merupakan penjelasan lebih rinci dari *use case* pengajuan investasi. Tabel 4.17 menunjukkan *use case scenario* untuk pengajuan investasi.

Tabel 4.17 *Use Case Scenario* Pengajuan Investasi

Kode Kebutuhan	MK-1-007
Objective	Investor berhasil melakukan mengajukan investasi terhadap usaha yang dilakukan oleh mitra
Actors	Investor
Pre-condition	Investor akan melakukan pengajuan investasi
Main flow	1. Investor memilih usaha yang ada pada *market place* 2. Investor melihat ditail usaha, bisnis *plan* mitra usaha, dan memonitoring usaha mitra 3. Jika investor tetarik pada usaha yang diajukan oleh mitra, maka investor dapat mengirim *email* kepada mitra 4. Investor dapat memasukkan dana pembiayaan yang sudah di tentukan pada setiap pengajuan pembiayaan yang telah dilakukan oleh mitra
Alternative flows	1. Jika investor menekan tombol *cancel* setelah melihat usaha yang dilakukan mitra maka pengajuan tidak akan di lanjutkan
Post-condition	Data pengajuan investasi berhasil diajukan

8. Verifikasi Pengajuan Investasi

Use case scenario verifikasi pengajuan investasi merupakan penjelasan lebih rinci dari *use case* verifikasi pengajuan investasi. Tabel 4.18 sampai 4.19 menunjukkan *use case scenario* untuk verifikasi pengajuan investasi.

Tabel 4.18 *Use Case Scenario* Verifikasi Pengajuan Investasi

Kode Kebutuhan	MK-1-008
Objective	Admin berhasil melakukan verifikasi terhadap pengajuan investasi yang dilakukan oleh investor
Actors	Admin MaslahatKita
Pre-condition	Admin belum melakukan verifikasi pada pengajuan investasi yang dilakukan oleh investor

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

Tabel 4.19 *Use Case Scenario* Verifikasi Pengajuan Investasi (lanjutan)

Main flow	1. Admin akan menekan tombol verifikasi pengajuan investasi 2. Admin memilih pengajuan yang diinginkan 3. Admin memeriksa data pengajuan investasi yang di lakukan oleh investor 4. Jika admin menyetujui pengajuan investasi tersebut maka admin akan menekan tombol *approve* 5. Sistem mengirimkan pesan email konfrimasi kepada mitra 6. Sistem akan menyimpan data pengajuan untuk monitoring nantinya
Alternative flows	1. Jika admin menolak untuk pengajuan tersebut maka admin akan menekan tombol *reject* 2. Kemudian sistem akan menampilkan form untuk mengisi pesan yang akan dikirimkan kepada investor yang mengajukan 3. Setelah itu admin menekan tombol *Okay* bila ingin mengirim pesan tersebut serta menolak pengajuan tersebut 4. Jika admin menekan tombol *cancel* maka tidak akan di lakukan verifikasi pada pengajuan investasi tersebut
Post-condition	Data pengajuan investasi usaha berhasil verifikasi oleh admin

9. Pencairan Dana Usaha

Use case scenario pencairan dana usaha merupakan penjelasan lebih rinci dari *use case* pencairan dana usaha. Tabel 4.20 – 4.21 menunjukkan *use case scenario* untuk pencairan dana usaha.

Tabel 4.20 *Use Case Scenario* Pencairan Dana Usaha

Kode Kebutuhan	MK-1-009
Objective	Mitra berhasil melakukan pencairan dana usaha
Actors	Mitra
Pre-condition	Mitra belum melakukan pencarian dana usaha

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

Tabel 4.21 *Use Case Scenario* Pencairan Dana Usaha (lanjutan)

Main flow	1. Mitra menekan tombol pencairan dana usaha 2. Sistem akan memeriksa ketersedian dana 3. Jika dana tersedia maka sistem akan memberikan konfrimasi kepada mitra dan juga investor akan menerima konfrimasi pencairan dana
Alternative flows	1. Jika dana tidak tersedia maka muncul pesan penolakan
Post-condition	Mitra berhasil melakukan pencairan dana usaha

10. Pengangsuran

Use case scenario pengangsuran merupakan penjelasan lebih rinci dari *use case* pengangsuran. Tabel 4.22 menunjukkan *use case scenario* untuk verifikasi pengangsuran.

Tabel 4.22 *Use Case Scenario* Pengangsuran

Kode Kebutuhan	MK-1-010
Objective	Mitra berhasil melakukan pengangsuran
Actors	Mitra
Pre-condition	Mitra belum melakukan pengangsuran
Main flow	1. Mitra akan mentransfer melalui bank yang telah tersedia 2. Mitra mentransfer melalu virtual *account* yang telah disediakan
Alternative flows	1. Jika mitra menekan tombol *cancel* maka tidak akan di lakukan pengangsuran
Post-condition	Mitra berhasil melakukan pengangsuran

11. Verifikasi Pengangsuran

Use case scenario verifikasi pengangsuran merupakan penjelasan lebih rinci dari *use case* verifikasi pengangsuran. Tabel 4.23 menunjukkan *use case scenario* untuk verifikasi pengangsuran.

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

Tabel 4.23 *Use Case Scenario* Verifikasi Pengangsuran

Kode Kebutuhan	MK-1-011
Objective	Admin berhasil melakukan verifikasi pengangsuran
Actors	Admin
Pre-condition	Admin belum melakukan verifikasi pengangsuran
Main flow	1. Admin menekan tombol verifikasi pengangangsuran 2. Admin memilih pengangsuran yang sedang di lakukan 3. Admin memeriksa pengangsuran yang dilakukan oleh mitra 4. Jika pengangsuran disetujui maka dana angsuran akan masuk dalam rekening maslahat 5. dan bila ada margin maga sistem akan membaginya kepada investor dan persuhaan sesuai dengan peraturan yang telah di buat 6. Mitra akan menerima email konfrimasi pengangsuran yang telah di sepakati
Alternative flows	1. Jika admin menolak untuk pengangsuran tersebut maka admin akan menekan tombol *reject* 2. Kemudian sistem akan menampilkan form untuk mengisi pesan yang akan dikirimkan kepada mitra yang mengajukan 3. Setelah itu admin menekan tombol *Okay* bila ingin mengirim pesan tersebut serta menolak pengangsuran tersebut 4. Jika mitra menekan tombol *cancel* maka tidak akan di lakukan pengangsuran
Post-condition	Admin berhasil melakukan verifikasi pengangsuran

12. Pelaporan Angsuran

Use case scenario pelaporan angsuran merupakan penjelasan lebih rinci dari *use case* pelaporan angsuran. Tabel 4.24 menunjukkan *use case scenario* untuk pelaporan angsuran.

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

Tabel 4.24 *Use Case Scenario* Pelaporan Angsuran

Kode Kebutuhan	MK-1-012
Objective	Mitra berhasil melakukan pelaporan angsuran
Actors	Mitra
Pre-condition	Mitra belum melakukan pelaporan angsuran
Main flow	1. Mitra menekan tombol pelaporan pengangsuran 2. Mitra menekan tombol penambahan pelaporan pengangsuran 3. Mitra mengisi form untuk melakukan pelaporan pengangsuran 4. Sistem mamasukkan data pelaporan kedalam data base
Alternative flows	1. Jika mitra menekan tombol *cancel* maka tidak akan di lakukan pelaporan angsuran
Post-condition	Mitra berhasil melakukan angsuran

13. Pencairan Dana Hasil Investasi

Use case scenario pencairan dana hasil investasi merupakan penjelasan lebih rinci dari *use case* pencairan dana hasil investasi. Tabel 4.25 menunjukkan *use case scenario* untuk pencairan dana hasil investasi.

Tabel 4.25 *Use Case Scenario* Pencairan Dana Hasil Investasi

Kode Kebutuhan	MK-1-013
Objective	Investor berhasil melakukan pencairan dana hasil investasi
Actors	Investor
Pre-condition	Investor belum melakukan pencairan dana hasil investasi
Main flow	1. Mitra menekan tombol pelaporan pengangsuran 2. Mitra menekan tombol penambahan pelaporan pengangsuran 3. Mitra mengisi form untuk melakukan pelaporan pengangsuran 4. Sistem mamasukkan data pelaporan kedalam data base
Alternative flows	1. Jika mitra menekan tombol *cancel* maka tidak akan di lakukan pelaporan angsuran
Post-condition	Mitra berhasil melakukan angsuran

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

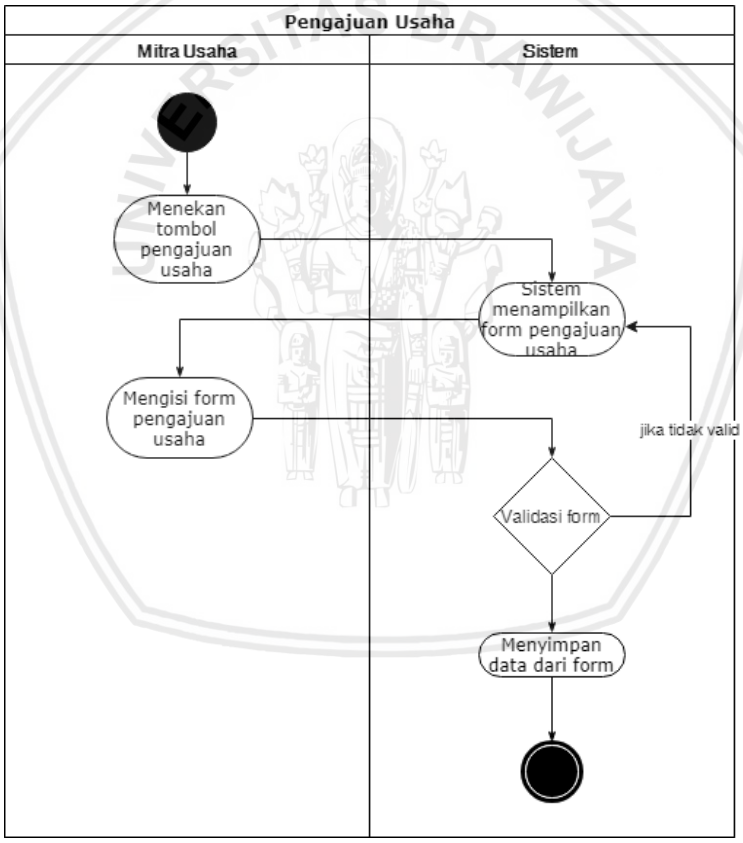
BAB 5 PERANCANGAN DAN IMPLEMENTASI

5.1 Perancangan

5.1.1 Perancangan *Activity Diagram*

Pada tahapan peracangan *activity diagram* ini merupakan penjabaran alur uang ada pada *use case* secara detail. *Actity diagram* yang ada dapat dilihat pada Gambar 5.1 – 5.3.

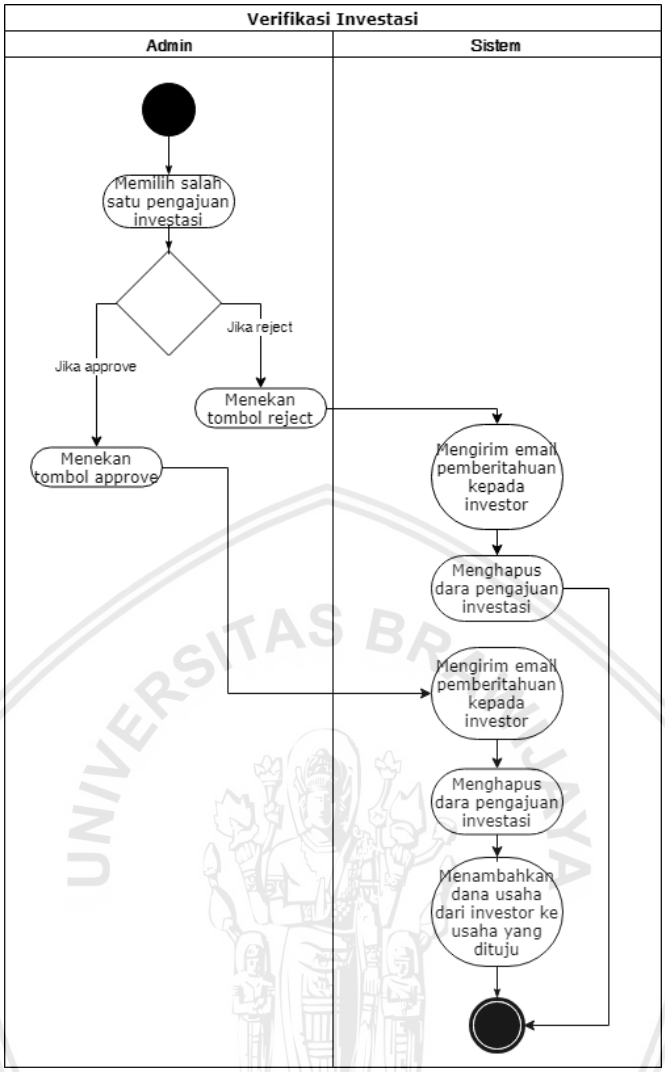
Activity diagram pengajuan usaha dalam Gambar 5.1 menjelaskan aliran kerja dari sistem saat melakukan proses pengajuan usaha. Mekanisme kerja dimulai dari Aktor mitra usaha menekan tombol pengajuan usaha yang kemudian sistem menampilkan form pengajuan usaha. Mitra usaha mengisi form tersebut, lalu sistem melakukan validasi terhadap form. Jika valid maka sistem menyimpan data dari form tersebut.



Gambar 5.1 *Activity* Diagram Pengajuan Usaha

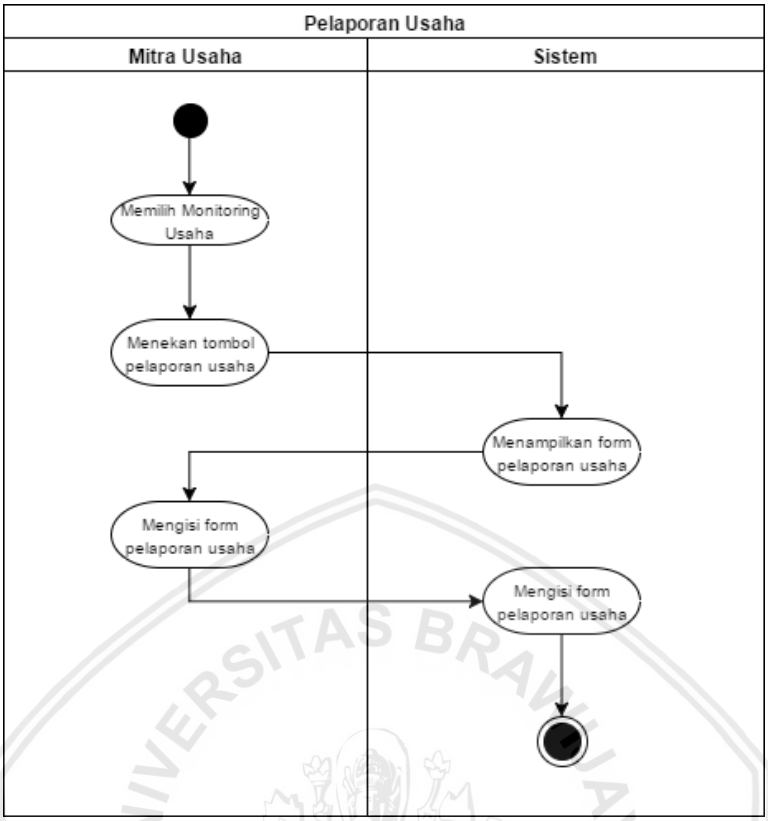
Activity diagram verifikasi investasi dalam Gambar 5.2 menjelaskan aliran kerja dari sistem saat melakukan proses verifikasi investasi. Mekanisme kerja dimulai dari Aktor admin memilih salah satu pengajuan investasi. Jika di *reject*, maka sistem mengirim email pemberitahuan kepada investor, dan menghapus data pengajuan investasi. Jika di *approve*, maka sistem mengirim email pemberitahuan kepada investor, menghapus data pengajuan investasi, dan menambahkan dana usaha dari investor ke usaha yang dituju.

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA



Gambar 5.2 *Activity Diagram* Verifikasi Investasi

Activity diagram pelaporan usaha dalam Gambar 5.3 menjelaskan aliran kerja dari sistem saat melakukan proses progres usaha. Mekanisme kerja dimulai dari Aktor mitra usaha memilih *monitoring* usaha dan menekan tombol pelaporan usaha. Lalu sistem menampilkan form pelaporan usaha. Mitra usaha mengisi form tersebut. Lalu sistem mengimpan data pelaporan usaha tersebut.



Gambar 5.3 *Activity Diagram* Pelaporan Usaha

5.1.2 Perancangan *Class Diagram*

Peranancangan *class diagram* yang digunakan pada penelitian ini dapat dilihat pada Gambar 5.4. Yang mana pada Gambar 5.4 akan terdapat beberapa class untuk melakukan fungsi-fungi untuk dilakukan sesuai dengan modul-modul kebutuhan yang telah dibuat. Salah satunya terdapat kelas-kelas model diantaranya *InvestorManagementModel*, *MarketPlaceModel*, *PengajuanUsaha-Model*, *UserManagementModel*, dan lain-lain. Serta terdapat juga beberapa *class activity* sebagai aktifitas yang akan dilakukan oleh aplikasi untuk melakukan fungsinya terhad tampilan yaitu *AdminActivity*, *InvestorActivity*, *MitraActivity*, *MonitoringUsahaActivity, PelaporanUsahaActivity*, dan lain-lain. Selain itu terdapat juga kelas *fragment* dimana kelas *fragment* ini di peruntukan untuk ada didalam sebuah *activity* diantaranya *DataPribadiInvestorFragment , DataPribadiMitraFragment, MarketPlaceFragment, MitraManagementFragment, PengajuanUsaha-BaruFragment, PengajuanUsahaDiterimaFragment, Pengajuan-UsahaDitolakFragment, SignupFragment,* dan lain-lain. Terdapat juga kelas adapter yang mana diperuntukan untuk mengatur sebuah *recyclerview* sehingga pada setiap data yang muncul akan diatur dapa kelas tersebut, diantaranya *InvestasiApproveAdapter, MonitoringUsahaAdapter, PengajuanUsahaBaru-Adptern,* dan kelas lainya. Serta terdapat juga kelas *listener* merupakan kelas *interface* untuk memverifikasi investor agar dapat dilakukan verifikasi. Dan juga terdapat beberapa file XML sebagai tampilan yang digunakan pada aplikasi tersebut.

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

Gambar 5.4 ***Class Diagram*** **Aplikasi MaslahatKita**

5.1.3 Perancangan Komponen

Pada perancangan komponen merupakan sebuah perancangan yang merincikan dari setiap komponen perangkat lunak yang dibuat. Pada perancangan komponen akan memaparkan alur kode program yang akan dibuat. Pada penelitian ini akan memaparkan tiga kode program pada *method* dalam sistem yang akan dikembangkan. Tiga *method* tersebut adalah *method onClick* pada kelas *PengajuanUsahaActivity*, *method onDataChange* pada kelas *VerifikasiInvestasi-Fragment* dan *method onClick* pada kelas *PelaporanUsahaActivity*. Berikut adalah hasil pemaparan perancangan komponen tiga kode program pada method.

5.1.3.1 Perancangan Komponen *Method* OnClick Pada Kelas PengajuanUsahaActivity

Komponen kode program tersebut berfungsi untuk memasukkan semua masukkan yang dilakukan *EditText* dan juga semua unggah gambar dan *file* kedalam *firebase*. Yang mana pada *method* tersebut terdapat pada kelas *PengajuanUsahaActivity.* Yang mana juga akan memanggil kelas model untuk memasukkan data-data tersebut terhadap *firebase* yang diberinama PengajuanUsaha. Komponen kode Program ini dapat dilihat pada Tabel 5.1.

Tabel 5.1 *Pseudocode Method* onClick

```
Pseudocode Method onClick
Start

Set EditText tujuanPengajuan,alamatUsaha,bidangUsaha,
deskripsiUsaha, proposal, namaUsaha
Set Uri filePathFoto1,filePathFoto2, filePathProposal

   If (nama != 0 & tanggalPengajuan != 0 & danaDiajukan != 0 & margin
   != 0 & tujuanPengajuan != 0 & namaUsaha != 0 & alamatUsaha != 0 &
   bidangUsaha != 0 & deskripsiUsaha != 0 & filePathFoto1 != kosong
   & filePathFoto2 != kosong & filePathFoto != kosong)

       Set nama pada kelas model
       Set jumlahWaktu pada kelas model
       Set danaDiButuhkan pada kelas model
       Set margin pada kelas model
       Set tujuanPengajuanpada kelas model
       Set namaUsaha pada kelas model
       Set alamatUsaha pada kelas model
       Set bidangUsaha pada kelas model
       Set deskripsiUsaha pada kelas model
       Set mitraId dengan intent dari kelas lain pada kelas model
       Set email dengan intent dari kelas lain pada kelas model
       Set statusUsaha = 0 pada kelas model
       Memanggil method uploadFoto1
       Memanggil method uploadFoto2
       Memanggil method uploadFile
       Memberi informasi "Berhasil Mendaftar"

           Else
           Memberikan informasi "Tolong Isi Semua Form"

   End If

End
```

5.1.3.2 Perancangan Komponen *Method* OnDataCahce Pada Kelas VerifikasiInvestasiFragment

Komponen kode program tersebut berfungsi untuk memasukkan semua masukkan yang dilakukan untuk melihat semua daftar investasi yang dilakukan investor. Yang mana pada *method* tersebut terdapat pada kelas *VerifikasiInvestasiFragment.* Yang mana juga akan memanggil kelas model untuk memasukkan data-data tersebut kedalam *firebase* yang di berinama approveModel. Dan juga terdapat pengecekan data yang dilakuakn berulang menggunakan *for*. Komponen kode Program ini dapat dilihat pada Tabel 5.2.

Tabel 5.2 *Pseudocode Method* onDataChance

```
Pseudocode Method onDataChance
Start
Set bersihkan array list
for sebanyak dataSnapshot
   Membuat objek ps untuk PengajuanUsahaModel
   Membuat HashMap untuk objek investor dengan tipe String
   If (objek investor tidak boleh kosong)
       For sebanyak insvestor
           If (status investor tidak kosong)
               Membuat objek approveModel
               Set jumlahInvestasi dari objek approveModel
               Set namaInvestor dari objek approveModel
               Set namaUsaha dari objek approveModel
               Set idInvestor dari objek approveModel
               Set idPengajuanUsaha dari objek approveModel
               Set danaYangTerkumpul dari objek approveModel
               Masukkan objek approveModel list
           End if
       End for
   End If
End for
Membuat objek adapter
Set layoutManager
Set adapter
End
```

5.1.3.3 Perancangan Komponen *Method* onClick Pada Kelas PelaporanUsahaActivity

Komponen kode program tersebut berfungsi untuk memasukkan semua masukkandata pelaporan usaha. Yang mana pada *method* tersebut terdapat pada kelas *PelaporanUsahaActivity.* Yang mana juga akan melakukan perhitungan untuk melakukan angsuran yang akan di bagiakan pada setiap investor yang telah

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

menginvestasikan pada usaha yang dijalankan oleh mitra sebanyak lima belas persen. Komponen kode Program ini dapat dilihat pada Tabel 5.3.

Tabel 5.3 *Pseudocode Method* onClick

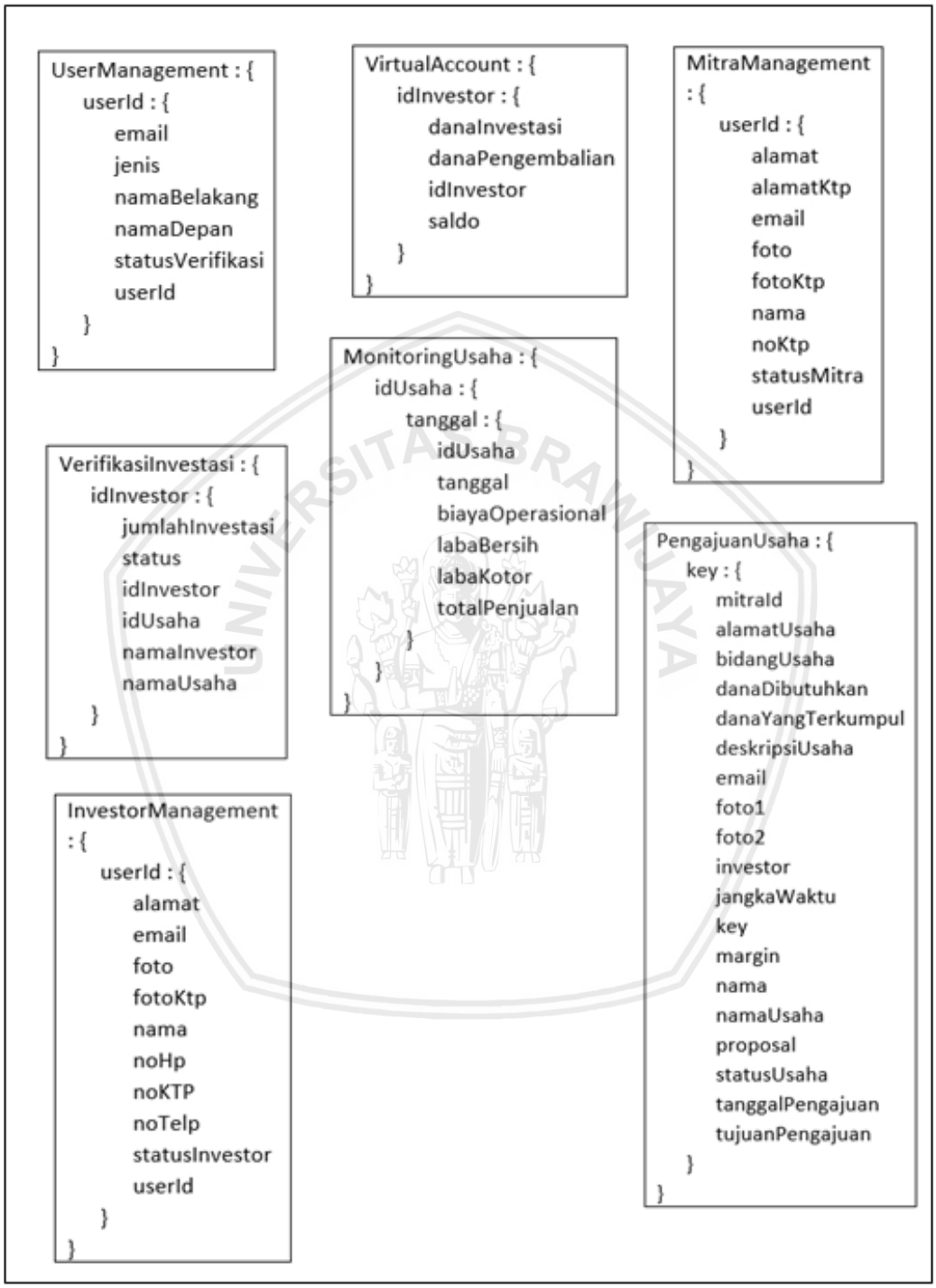
```
Pseudocode Method onClick
Start
If(totalPenjualan != 0 dan labaBersih != 0 dan labaKotor !=0 dan
biayaOprasiaonal !=0 dan angsuran !=0)
   Memuat idUsaha dari firebase
   Memuat totalPenjualan dari form
   Memuat labaBersih dari form
   Memuat labaKotor dari form
   Memuat biayaOprasional dari form
   Memuat tanggal for sebanyak dataSnapshot
   waktu double = mengambil jangkaWaktu
   angsuran_usaha double = ((danaDibutuhkan X 10
   /100)+danaDibutuhkan) : waktu
   angsuran_usaha_total double =((danaDibutuhkan X 10 :100)+
   danaDibutuhkan) : waktu
   angsuran_usaha_for_investor double =((danaDibutuhkan X 10 :100)+
   danaDibutuhkan) : waktu
   if (angsuran tidak = 0 dan angsuran = angsuran_usaha_total)
       li HashMap = mendapatkan investor
       For sebanyak jumlah investor yang telah di setujui
           mPengembalian double = jumlah investasi : total x
           angsuran_usaha_total
           Mengambil database VirtualAccount pada setiap investor
           Memanggil method onDataChange
           Memasukan data dari database kedalam model
           Memuat saldo awal + data mPengembalian
           Membuat isi database VirtualAccount dengan model yang
           sudah dirubah
           Memanggil method onCancelled
       End for
       Melakukan Instansiasi firebase
       Mengakses database pada Monitorring Usaha
       Menyimpan isi model pada firebase
       Memberikan infromasi "Pelaporan Usaha Telah Tersimpan"
       Menyudahi activity yang berjalan
   Else
   Memberikan informasi "Tolong Isi Angsuran Sesuai Dengan Angsuran
   Yang Tertera"
   End If
Else
Memberikan informasi "Tolong Isi Semua Form"
End If
End
```

5.1.4 Perancangan Basis Data

Perancangan basis data akan menggambarkan mengenai representasi dari *database* yang akan dibangan pada aplikasi ini. Dimana penggunaan *database* yang digunakan pada penelitian ini adalah *noSQl.* Yang mana akan terdapat beberapa objek utama dalam membangun aplikasi ini yaitu *UserManagement,*

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

MitraManagement, InvestorManagement, VirtualAccount, MonitoringUsaha, PengajuanUsaha, dan *VarifikasiInvestasi.* Pada setiap objek tersebut dapat dilihat pada Gambar 5.7 untuk perancangan basis.



Gambar 5.5 Perancangan Basis Data

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

Dapat dilihat secara keseluruhan akan data yang dibutuhkan pada perancangan basis data pada Gambar 5.5. Lebih terperinci lagi akan dijelaskn pada Tabel 5.4 sampai Tambel 5.11.

1. Tabel User Management

   Nama Tabel : User Management

   Jumlah Field : 6

   Fungsi : Untuk menyimpan data user

Tabel 5.4 Struktur Tabel UserManagement

No.	Nama Field	Tipe	Deskripsi
1	userId	String	id user
2	email	String	email user
3	jenis	String	jenis investasi
4	namabelakang	String	nama belakang user
5	namadepan	String	nama depan user
6	statusVerifikasi	integer	status verifikasi investasi

2. Tabel Virtual Account

   Nama Tabel : VirtualAccount

   Jumlah Field : 4

   Fungsi : Menyimpan data keuangan milik investor

Tabel 5.5 Struktur Tabel VirtualAccount

No.	Nama Field	Tipe	Deskripsi
1	idInvestor	String	id investor
2	danaInvestasi	String	jumlah dana investasi
3	danaPengembalian	double	jumlah dana pengem-balian
4	saldo	double	saldo user

3. Tabel Verifikasi Investasi

   Nama Tabel : VerifikasiInvestasi

   Jumlah Field : 6

   Fungsi : Menyimpan data verifikasi investasi

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

Tabel 5.6 Struktur Tabel VerifikasiInvestasi

No.	Nama Field	Tipe	Deskripsi
1	idInvestor	String	id investor
2	idUsaha	String	id usaha milik investor
3	jumlahInvestasi	double	jumlah investasi
4	namaInvestor	String	nama investor
5	namaUsaha	String	nama usaha
6	status	integer	Status

4. Tabel Mitra Management

   Nama Tabel : MitraManagement

   Jumlah Field : 9

   Fungsi : Untuk menyimpan data mita

Tabel 5.7 Struktur Tabel MitraManagement

No.	Nama Field	Tipe	Deskripsi
1	userId	String	id user
2	alamat	String	Alamat
3	alamatKtp	String	alamat pada KTP
4	email	String	alamat email
5	foto	String	Foto
6	fotoKtp	String	foto KTP
7	nama	String	nama user
8	noKtp	String	nomor KTP
9	statusMitra	integer	status mitra

5. Tabel Investor Management

   Nama Tabel : InvestorManagement

   Jumlah Field : 10

   Fungsi : Untuk menyimpan data investor

Tabel 5.8 Struktur Tabel InvestorManagement

No.	Nama Field	Tipe	Deskripsi
1	userId	String	id user
2	alamat	String	alamat user
3	email	String	alamat email
4	foto	String	Foto
5	fotoKtp	String	foto KTP

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

Tabel 5.9 Struktur Tabel InvestorManagement (lanjutan)

No.	Nama Field	Tipe	Deskripsi
6	nama	String	Nama
7	noHp	String	nomor HP
8	noKTP	String	nomor KTP
9	noTelp	String	nomor telepon
10	statusInvestor	integer	status investor

6. Tabel Pengajuan Usaha

   Nama Tabel : PengajuanUsaha

   Jumlah Field : 19

   Fungsi : Untuk menyimpan usaha yang dilakukan mitra

Tabel 5.10 Struktur Tabel PengajuanUsaha

No.	Nama Field	Tipe	Deskripsi
1	mitraId	String	id mitra
2	alamatUsaha	String	alamat tempat usaha
3	bidangUsaha	String	bidang usaha
4	danaDibutuhkan	double	dana yang dibutuhkan usaha
5	danaYangTerkumpul	double	dana yang terkumpul
6	deskripsiUsaha	String	deskripsi dari usaha
7	email	String	Email
8	foto1	String	foto pertama
9	foto2	String	foto kedua
10	investor	String	Investor
11	jangkaWaktu	String	jangka waktu
12	key	String	Key
13	margin	double	Margin
14	nama	String	Nama
15	namaUsaha	String	nama usaha
16	proposal	String	Proposal
17	statusUsaha	integer	status usaha
18	tanggalPengajuan	String	tanggal pengajuan
19	tujuanPengajuan	String	tujuan pengajuan

7. Tabel Monitoring Usaha

   Nama Tabel : MonitoringUsaha

   Jumlah Field : 6

   Fungsi : Untuk menyimpan monitoring usaha

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

Tabel 5.11 Struktur Tabel MonitoringUsaha

No.	Nama Field	Tipe	Deskripsi
1	idUsaha	String	id usaha
2	tanggal	String	tanggal submit monitoring
3	biayaOperasional	double	biaya operasional
4	labaBersih	double	laba bersih
5	labaKotor	double	laba kotor
6	totalPenjualan	double	total penjualan

5.1.5 Perancangan Antarmuka

Pada perancanngan antarmuka akan membahas mengenai tentang desain dari tampilan yang akan digunakan pada aplikasi perangkat bergerak berbasis android. Pada perancangan antarmuka ini akan memberikan alur dari setipan halaman antarmuka dan elemen apa saja dari sertiap halaman antar muka nantinya. Pada perancangan antarmuka akan terdiri *wireframe* dan *screenflow* yang dijelaskan pada sub bab 5.1.3.1 dan 5.1.3.2.

5.1.5.1 Perancangan *Screenflow*

Merupakan perancangan antarmuka, dimana menjelaskan fungsi-fungsi dari setiap item yang dapat berhungan dangan tampilan halaman lainnya. Sehingga dapat mempermudah dalam perancangan pada alur yang sudah dibuat. Pada perancangan ini *screenflow* dapat dilihat pada Gambar 5.6.

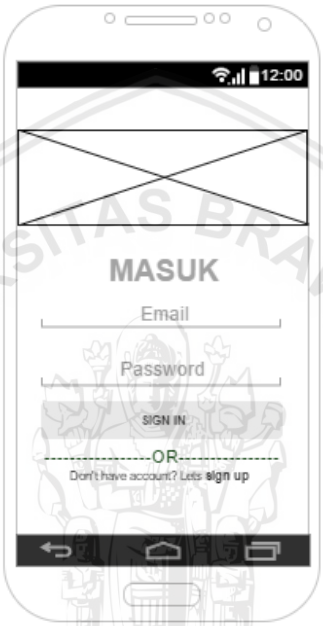
repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

Gambar 5.6 Perancangan *Screenflow*

5.1.5.1 Perancangan *wireframe*

Perancangan *wireframe* merupakan perancangan yang memuat kerangka dasar yang nantinya akan digunakan pada perangkat bergerak. Pada perancangan *wireframe* diberikan penjelasn tentang elemen-elemen yang terdapat pada sebuah halaman antar muka.

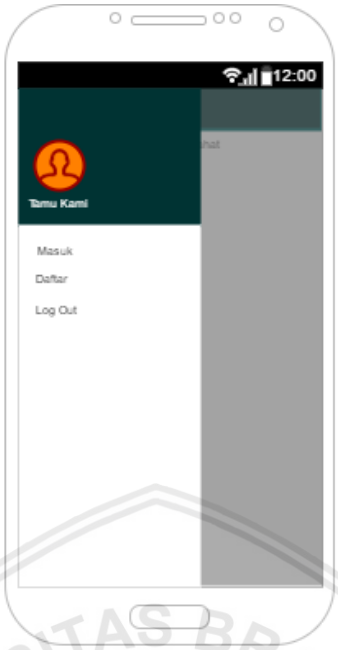
Halaman *login* dimana halaman ini yang akan digunakan pengguna untuk melakukan hak aksesnya terhadap sistem. Sehingga nantinya akan menampilkan halaman sesuai kategori hak akses yang telah di peruntukkan nantinya. Perancagan antarmuka *login* terdapat pada Gambar 5.7.



Gambar 5.7 Perancangan Antarmuka *Login*

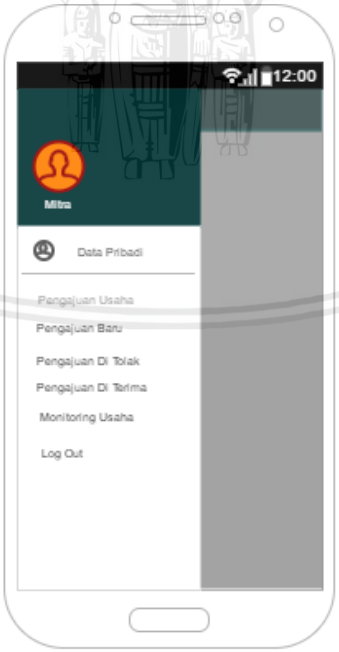
Halaman *Guest* dimana halaman ini yang akan digunakan pengguna untuk melakukan *login* maupun daftar pada navigasibar yang di berikan pada halaman tersebut. Perancagan antarmuka *guest* terdapat pada Gambar 5.8.

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA



Gambar 5.8 Perancangan Antarmuka *Guest*

Halaman mitra dimana halaman ini yang akan digunakan pengguna dengan hak akses sebagai mitra untuk melakukan fungsi-fungsi yang terdapat pada mitra di navigasibar. Sehingga nantinya akan menampilkan halaman sesuai kategori yang telah di peruntukkan. Perancagan antarmuka mitra terdapat pada Gambar 5.9.



Gambar 5.9 Perancangan Antarmuka Mitra

Halaman pengajuan usaha dimana halaman ini yang akan digunakan pengguna untuk melakukan pengajuan usaha dengan mengisi form-form yang

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

telah di berikan pada halaman tersebut. Perancangan antarmuka pengajuan usaha terdapat pada Gambar 5.10.

Gambar 5.10 Perancangan Antarmuka Pengajuan Usaha

Halaman *monitoring* usaha mitra dimana halaman ini yang akan digunakan pengguna untuk melakukan penarikan dana maupun melakukan pengangsuran serta ingin melakukan laporna usaha yang sedang di jalani pada halaman tersebut . Perancangan antarmuka *monitoring* mitra terdapat pada Gambar 5.11.

Gambar 5.11 Perancangan Antarmuka *Monitoring* Usaha Mitra

Halaman pelaporan usaha mitra dimana halaman ini yang akan digunakan pengguna untuk melakukan masukkan pelaporan usaha yang dilakukan selama sebulan. Dimana pengguna dapat mengisi form yang telah disediakan oleh halaman antarmuka.. Perancangan antarmuka pelaporan usaha terdapat pada Gambar 5.12.

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

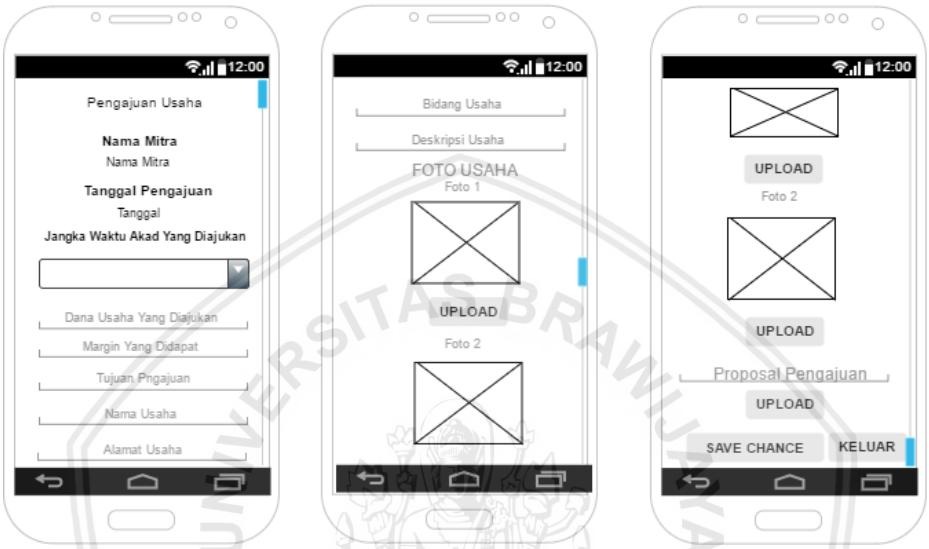
Gambar 5.12 Perancangan Antarmuka Pelaporan Usaha

Halaman *list* pengajuan usaha baru dimana halaman ini yang akan digunakan pengguna untuk melihat pengajuan usaha apa saja yang baru di daftarkan. Dimana pengguna dapat melihat seluruh *list* yang ada pada tampilan tersebu. Perancangan antarmuka *list* pengajuan usaha baru terdapat pada Gambar 5.13.

Gambar 5.13 Perancangan Antarmuka *List* Pengajuan Usaha Baru

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

Halaman detail pengajuan usaha baru dimana halaman ini yang akan digunakan pengguna untuk melihat seluruh data pengajuan usaha secara lebih rinci dan juga dapat mengubahnya bila di perlukan.Selain itu juga terdapat bebrapa form dan tombol untuk mengunggah *file* ataupun berkas dengan format pdf. Perancangan antarmuka detail pengajuan usaha baru terdapat pada Gambar 5.14.

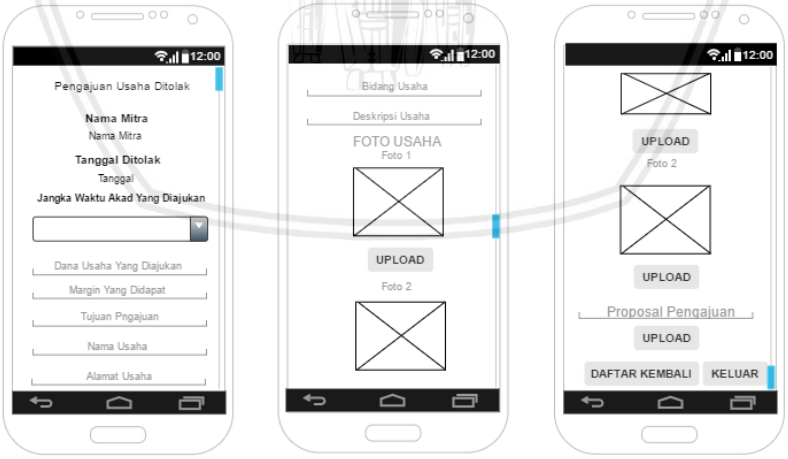


Gambar 5.14 Perancangan Antarmuka Detail Pengajuan Usaha Baru

Halaman *list* pengajuan usaha ditolak dimana halaman ini yang akan digunakan pengguna untuk melihat pengajuan usaha apa saja yang tellah ditolak oleh admin MaslahatKita . Dimana pengguna dapat melihat seluruh *list* yang ada pada tampilan tersebut dengan *recyclerview* dan terdapat tombol detail untuk melihat seluruh lebih rinci masing-masing pengajuan tersebut. Perancangan antarmuka *list* pengajuan usaha ditolak terdapat pada Gambar 5.15.

Gambar 5.15 Perancangan Antarmuka List Pengajuan Usaha Ditolak

Halaman detail pengajuan usaha ditolak dimana halaman ini yang akan digunakan pengguna untuk melihat seluruh data pengajuan usaha yang telah ditolak secara lebih rinci dan juga dapat mengubah data yang diperlukan.Selain itu juga terdapat bebrapa form dan tombol untuk mengunggah *file* ataupun berkas dengan format pdf dan juga tombol untuk mendaftarkan kembali pengajuan tersebut. Perancangan antarmuka detail pengajuan usaha ditolak terdapat pada Gambar 5.16.



Gambar 5.16 Perancangan Antarmuka Detail Pengajuan Usaha Ditolak

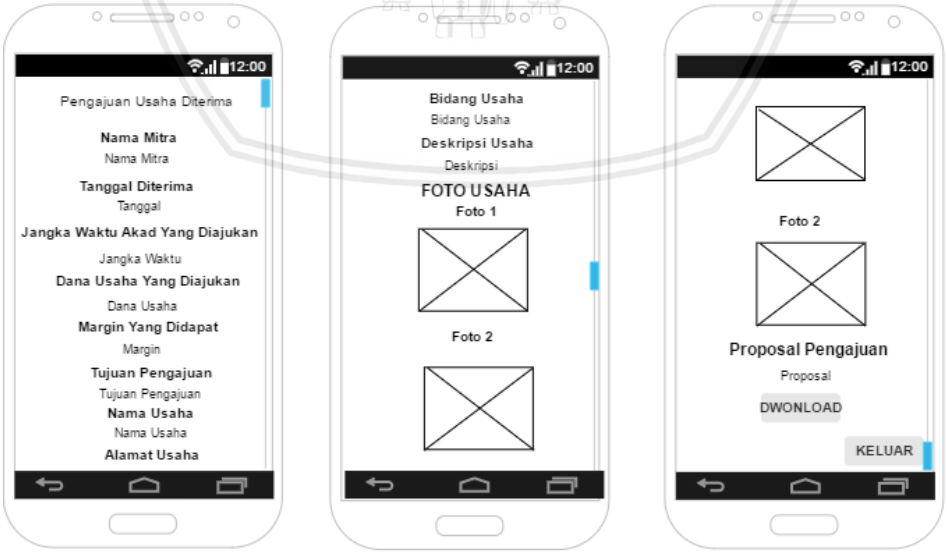
Halaman *list* pengajuan usaha diterima dimana halaman ini yang akan digunakan pengguna untuk melihat pengajuan usaha apa saja yang tellah diterima oleh admin MaslahatKita . Dimana pengguna dapat melihat seluruh *list* yang ada pada tampilan tersebut dengan *recyclerview* dan terdapat tombol detail untuk

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

melihat seluruh lebih rinci masing-masing usaha tersebut. Perancangan antarmuka *list* pengajuan usaha diterima terdapat pada Gambar 5.17.

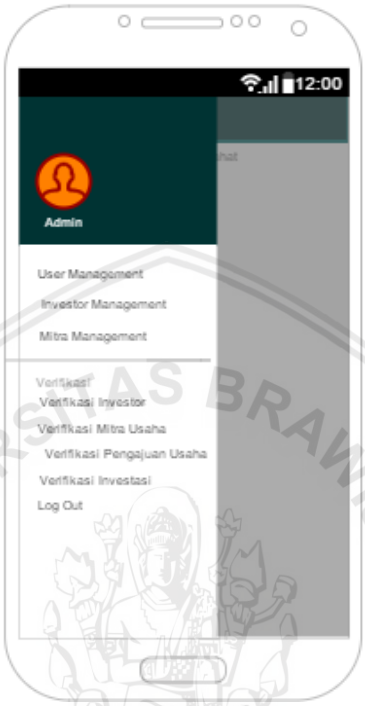
Gambar 5.17 Perancangan Antarmuka *List* Pengajuan Diterima

Halaman detail pengajuan usaha diterima dimana halaman ini yang akan digunakan pengguna untuk melihat seluruh data pengajuan usaha yang telah diterima secara lebih rinci.Selain itu juga terdapat tombol untuk mengunduh *file* ataupun berkas dengan format pdf pada pengajuan usaha tersebut. Perancangan antarmuka detail pengajuan usaha diterima terdapat pada Gambar 5.18.



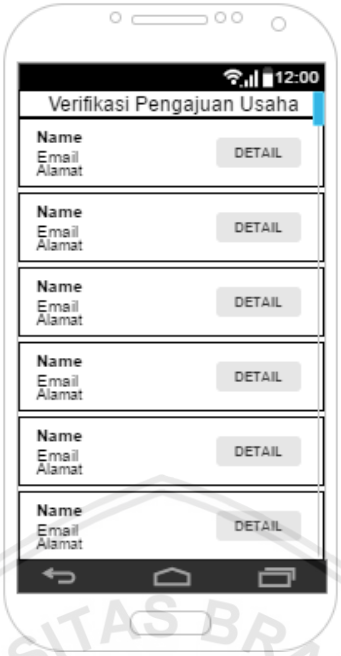
Gambar 5.18 Perancangan Antarmuka Detail Pengajuan Usaha Diterima

Halaman admin dimana halaman ini yang akan digunakan pengguna dengan hak akses sebagai admin untuk melakukan fungsi-fungsi yang terdapat pada admin di navigasibar. Sehingga nantinya akan menampilkan halaman sesuai kategori yang telah di peruntukkan. Perancagan antarmuka admin terdapat pada Gambar 5.19.



Gambar 5.19 Perancangan Antarmuka Admin

Halaman *list* verifikasi pengajuan usaha dimana halaman ini yang akan digunakan pengguna untuk melihat pengajuan usaha apa saja yang telah di daftarkan. Dimana pengguna dapat melihat seluruh *list* yang ada pada tampilan tersebut pada *recyclerview* yang telah dibuat. Perancagan antarmuka *list* verifikasi pengajuan usaha terdapat pada Gambar 5.20.



Gambar 5.20 Perancangan Antarmuka *List* Verifikasi Pengajuan Usaha

Halaman detail verifikasi pengajuan usaha dimana halaman ini yang akan digunakan pengguna untuk melihat seluruh data pengajuan usaha secara lebih rinci dan juga dapat mengubahnya bila di perlukan.Selain itu juga terdapat beberapa form dan tombol untuk mengunggah *file* ataupun berkas dengan format pdf. Perancangan antarmuka detail pengajuan usaha baru terdapat pada Gambar 5.21.

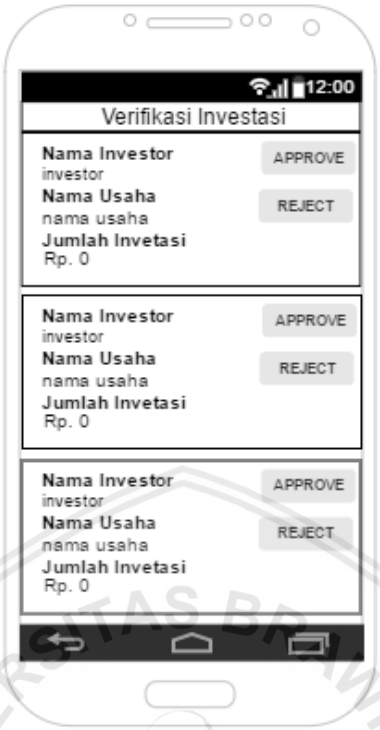
Gambar 5.21 Perancangan Antarmuka Detail Verifikasi Pengajuan Usaha

Halaman *form* penolakan dimana halaman ini yang akan digunakan pengguna untuk memberikan komentar atau info soal penolakan sebuah

pengajuan usaha yang dilakukan oleh mitra .Selain itu juga terdapat *form* untuk memasukkan alasan penolakan dan tombol kirim untuk mengirim pesan tersebut ataupun tombol keluar untuk keluar dari kotak dialog tersebut. Perancangan antarmuka *form* penolakan terdapat pada Gambar 5.22.

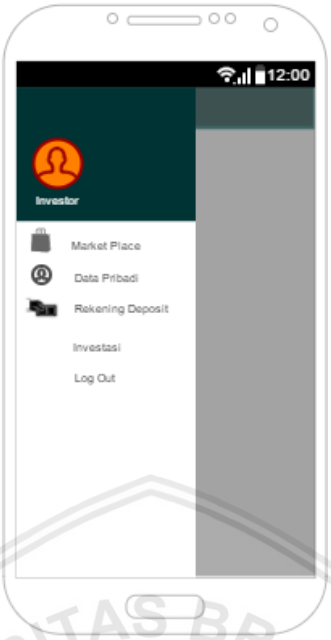
Gambar 5.22 Perancangan Antarmuka *Form* Penolakan

Halaman *list* verifikasi investasi dimana halaman ini yang akan digunakan pengguna untuk melihat investasi apa saja yang telah diajukan. Dimana pengguna dapat melihat seluruh *list* yang ada pada tampilan tersebut pada *recyclerview* yang telah dibuat. Dan juga terdapat tombol untuk persetujuan maupun penoakan pada investasi tersebut. Perancagan antarmuka *list* verifikasi investasi terdapat pada Gambar 5.23.



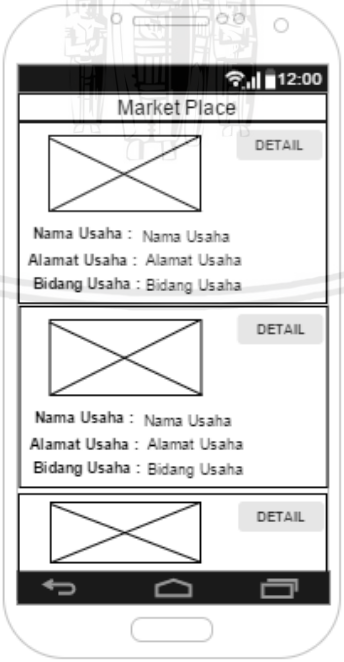
Gambar 5.23 Perancangan Antarmuka *List* Verifikasi Investasi

Halaman investor dimana halaman ini yang akan digunakan pengguna dengan hak akses sebagai investor untuk melakukan fungsi-fungsi yang terdapat pada investor di navigasibar. Sehingga nantinya akan menampilkan halaman sesuai dengan kategori yang telah di peruntukkan. Perancagan antarmuka investor terdapat pada Gambar 5.24.



Gambar 5.24 Perancangan Antarmuka Investor

Halaman *market place* dimana halaman ini yang akan digunakan pengguna untuk melihat saftar usaha apa saja yang telah terdaftar. Dimana pengguna dapat melihat seluruh daftar yang ada pada tampilan tersebut pada *recyclerview* yang telah dibuat dan juuga terdapat tombol investasi untuk melihat detail usaha. Perancagan antarmuka *list* verifikasi pengajuan usaha terdapat pada Gambar 5.25.



Gambar 5.25 Perancangan Antarmuka *Market Place*

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

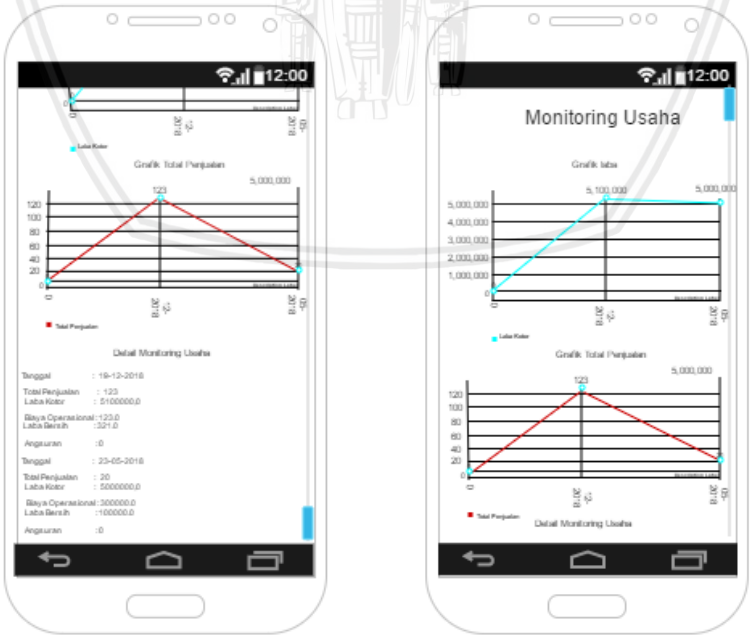
Halaman detail usaha mitra dimana halaman ini yang akan digunakan pengguna untuk melihat seluruh data usaha mitra secara lebih rincin.Selain itu juga terdapat beberapa tombol untuk memonitoring usaha mitra , tombol untuk mengunduh *file* proposal ataupun tombol untuk melakukan investasi pada usaha tersebut. Perancangan antarmuka detail pengajuan usaha baru terdapat pada Gambar 5.26.

Gambar 5.26 Perancangan Antarmuka Detail Usaha Mitra

Halaman *form* saran dimana halaman ini yang akan digunakan pengguna untuk memberikan komentar atau saran pada sebuah usaha mitra yang dilakukan oleh mitra .Selain itu juga terdapat *form* untuk memasukkan saran dan tombol kirim untuk mengirim pesan tersebut dan untuk sekaligus untuk berinvestasi pada usaha mitra ataupun tombol keluar untuk keluar dari kotak dialog tersebut. Perancangan antarmuka *form* saran terdapat pada Gambar 5.27.

Gambar 5.27 Perancangan Antarmuka *Form* Saran

Halaman *monitoring* usaha untuk investor dimana halaman ini yang akan digunakan pengguna untuk memonitoring usaha yang dilakukan oleh mitra secara detail. Selain itu juga terdapat grafik untuk melihat pergerakan usaha mitra untuk setiap bulannya. Perancangan antarmuka *monitoring* usaha untuk investor terdapat pada Gambar 5.28.



Gambar 5.28 Perancangan Antarmuka *Monitoring* Usaha Untuk Investor

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

5.2 Implementasi Sistem

5.2.1 Spesifikasi Piranti Pendukung

Pada sub bab spesifikasi piranti pendukung akan memaparkan piranti apa saja yang dipergunakan untuk mendukung adanya penelitian ini. Yang mana akan dipaparkan pada sub bab 5.2.1.1 dan sub bab 5.2.1.2.

5.2.1.1 Spesifikasi Perangkat Keras

Pada spesifikasi perangkat keras ini meparkan rincian spesifikasi dari perangkat keras yang digunakan pada penelitian ini. Dimana dipaparkan pada Tabel 5.12 sebagai penjabaran dalam perangkat yang digunakan.

Tabel 5.12 Spesifikasi Perangkat Keras

Nama Komponen	Spesifikasi
System Model	MSI SKU-GL62M
Processor	Kabylake i7-7700HQ+HM175
Memory	8 GB
Display	*nVidia Geforce GTX 1050Ti, 4GB GDDR5.*
Disk	1 TB SATA *Hard Disk Drive* 128 GB M.2 SATA *Solid State Disk*

5.2.1.2 Spesifikasi Perangkat Lunak

Pada spesifikasi perangkat lunak ini meparkan rincian dari spesifikasi perangkat lunak yang digunakan pada penelitian ini. Dimana dipaparkan pada Tabel 5.13 sebagai penjabaran dalam perangkat yang digunakan.

Tabel 5.13 Spesifikasi Perangkat Lunak

Nama Komponen	Spesifikasi
Operating System	Windows 10 Enterprise 64-bit
Programming Language	Java
Programming Environment	Android Studio
Support Device	ASUS Zenfone 2

5.2.2 Implementasi Kode Program

5.2.2.1 Implementasi Kode Program Pengajuan Usaha

Nama Kelas : PengajuanUsahaActivity

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

Nana Operasi : onClick

Kode Program :

```
1      if(nama.getText().toString().trim().length() != 0 &&
2  tanggalPengajuan.getText().toString().trim().length() != 0
3              && danaDiajukan.getText().toString().trim().length()
4  != 0
5              && margin.getText().toString().trim().length() != 0
6  && tujuanPengajuan.getText().toString().trim().length() != 0 &&
7              namaUsaha.getText().toString().trim().length() != 0
8
9  && alamatUsaha.getText().toString().trim().length() != 0 &&
10             bidangUsaha.getText().toString().trim().length() !=
11 0 && deskripsiUsaha.getText().toString().trim().length() != 0 &&
12             filePathFoto1 != null && filePathFoto2 != null &&
13 filePathProposal != null) {
14         model.setNama(nama.getText().toString());
15 model.setJangkaWaktu(jangkaWaktu.getSelectedItem().toString());
16         model.setTanggalPengajuan(tanggal_sekarang);
17
18 model.setDanaDibutuhkan(Integer.parseInt(danaDiajukan.getText().
19 toString()));
20
21 model.setMargin(Integer.parseInt(margin.getText().toString()));
23
24 model.setTujuanPengajuan(tujuanPengajuan.getText().toString());
25         model.setNamaUsaha(namaUsaha.getText().toString());
26         model.setAlamatUsaha(alamatUsaha.getText().toString());
27         model.setBidangUsaha(bidangUsaha.getText().toString());
28
29 model.setDiskripsiUsaha(deskripsiUsaha.getText().toString());
30         model.setMitraId(getIntent().getStringExtra("data1"));
31         model.setEmail(getIntent().getStringExtra("email"));
32         model.setStatusUsaha(0);
33         uploadFoto1();
34         uploadFoto2();
35         uploadFile();
36         Toast.makeText(PengajuanUsahaActivity.this,"Berhasil
37 Mendaftar Usaha", Toast.LENGTH_SHORT).show();
38         finish();
39     }
40     else{
41         Toast.makeText(PengajuanUsahaActivity.this,"Tolong Isi
42 Semua Form", Toast.LENGTH_SHORT).show();
43     }
```

Kode Program 5.1 Kode Program Untuk Menyimpan Pengajuan Usaha

Berikut merupakan penjelasan dari kode program yang ada dalam Kode Program 5.1 . Yaitu terdapat sebuah seleksi kondisi *if* dengan syarat dimana semua isian variable yang terdapat pada syarat tersebut tidak boleh kosong. Yang mana variabel tersebut yaitu nama, tanggalPengajuan, danaDiajukan, margin, tujuanPengajuan, namaUsaha, alamatUsaha, bidangUsaha, deskripsiUsaha, filePathFoto1, filePathFoto2, dan filePathFoto. Untuk mengecek apakah semua variabel telah terisi atau belum. Kemudian akan dimasukkannya data pada semua variabel tersebut kedalam sebuah *method set* pada kelas model. Kemudian akan dimasukkan delama *method* uploadFoto1, uploadFoto2, dan uploadFile yang sudah terdapat sebuah *method* onUploadSuccess untuk memasukkan semua data tersebut kadalam *firebase* yang telah disediakan. Kemudakan akan memunculkan

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

sebuah informasi bahwa semua data berhasil terdaftar. Setelah itu terdapat *method* finish untuk mengakhiri *activity* tersebut. Jika isi dari setiap variabel tersebut ada yang kosong maka akan menampilkan pesan "Tolong Isi Semua Form".

5.2.4.2 Implementasi Kode Verifikasi Investasi

Nama Kelas : VerifikasiInvestasiFragment

Nana Operasi : onDataChange

Kode Program :

```
1   list.clear();
2   for (DataSnapshot ds : dataSnapshot.getChildren()){
3       PengajuanUsahaModel ps =
4   ds.getValue(PengajuanUsahaModel.class);
5       HashMap<String, InvestorApproval> investor =
6   ps.getInvestor();
7       if (investor != null){
8           for ( String key : investor.keySet() ) {
9               if (!investor.get(key).getStatus()){
10                  ApproveModel approveModel = new ApproveModel();
11
12  approveModel.setJumlahInvestasi(investor.get(key).getJumlah());
13                  approveModel.setNamaInvestor(listNama.get(key));
14                  approveModel.setNamaUsaha(ps.getNamaUsaha());
15                  approveModel.setIdInvestor(key);
16                  approveModel.setIdPengajuanUsaha(ds.getKey());
17
18  approveModel.setDanaYangTerkumpul(ps.getDanaYangTerkumpul());
19                  list.add(approveModel);
20              }
21          }
23      }
24  }
25  VerifikasiInvestasiAdapter adapter = new
26  VerifikasiInvestasiAdapter(getContext(), list, listener);
27  rv.setLayoutManager(new LinearLayoutManager(getContext()));
28  rv.setAdapter(adapter);
```

Kode Program 5.2 Kode Program Untuk Memverifikasi Investasi

Berikut merupakan penjelasan dari kode program untuk memverifikasi investasi yang ada dalam Kode Program 5.2. Yaitu terdapat sebuah *method* untuk menghapus terlebih dahulu *array list* yang telah disediakan. Kemudian menginisialisai objek ps dengan kelas PengajuanUsahaModel. Dan juga menggunakan variabel investor dengan tipe data HashMap untuk mendata semua investor. Kemudian dilakukan seleksi kondisi *if* untuk melakukan pengecekkan investor tidak kosong. Jika syarat terpenuhi maka di lakukan perulangan sebanyak investor yang telah terdaftar. Lalu menginisialisai kelas ApproveModel pada objek approvemodel untuk menyeseikan semua data yang di butuhkan pada saat investasi. Yang mana semua data pada model tersebut akan dimasukkan kedalam *array list.* Jika semua oprasi sudah selesai makan akan dilakukan menyiapkan adapter dan *layout* untuk di tampilkan pada *recyclerview.*

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

5.2.4.3 Implementasi Kode Program Pelaporan Usaha

Nama Kelas : PelaporanUsahaActivity

Nana Operasi : onClick

Kode Program :

```
if (totalPenjualan.getText().toString().trim().length() != 0 &&
labaBersih.getText().toString().trim().length() != 0
                        &&
labaKotor.getText().toString().trim().length() != 0 &&
biayaOprasional.getText().toString().trim().length() != 0
                        &&
angsuran.getText().toString().trim().length() != 0){
                    model.setIdUsaha(pu.getKey());

model.setTotalPenjualan(Integer.valueOf(totalPenjualan.getText()
.toString()));

model.setLabaBersih(Double.valueOf(labaBersih.getText().toString
()));

model.setLabaKotor(Double.valueOf(labaBersih.getText().toString(
)));

model.setBiayaOprasional(Double.valueOf(biayaOprasional.getText(
).toString()));
                    model.setTanggal(tanggal_sekarang);
                    double waktu =
Double.valueOf(pu.getJangkaWaktu().split(" ")[0]);
                    final double angsuran_usaha_total =
((pu.getDanaDibutuhkan() * 15 / 100) + pu.getDanaDibutuhkan())
/ waktu;
                    final double angsuran_usaha_for_investor =
((pu.getDanaDibutuhkan() * 10 / 100) + pu.getDanaDibutuhkan())
/ waktu;
                    if
(angsuran.getText().toString().trim().length() != 0 &&
angsuran.getText().toString().trim().length() ==
angsuran_usaha_total) {
                        HashMap<String, InvestorApproval> li =
pu.getInvestor();
                        for (final String key : li.keySet()) {
                            final double mPengembalian =
li.get(key).getJumlah() / total * angsuran_usaha_for_investor;
                            //TODO ANGSURAN
//                      listInvestor.put(key, mPengembalian);

database.child("VirtualAccount").child(key).addListenerForSingle
ValueEvent(new ValueEventListener() {
                                @Override
                                public void
onDataChange(@NonNull DataSnapshot dataSnapshot) {
                                    VirtualAccountInvestorModel
virtualAccountInvestorModel =
dataSnapshot.getValue(VirtualAccountInvestorModel.class);

virtualAccountInvestorModel.setDanaPengembalian(virtualAccountIn
vestorModel.getDanaPengembalian() + mPengembalian);
```

Kode Program 5.3 Kode Program Untuk Pelaporan Usaha

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

```
53
54  virtualAccountInvestorModel.setSaldo(virtualAccountInvestorModel
55  .getSaldo() + mPengembalian);
56
57  database.child("VirtualAccount").child(key).setValue(virtualAcco
58  untInvestorModel);
59                                  }
60                                  @Override
61                                  public void onCancelled(@NonNull
62  DatabaseError databaseError) {
63
64                                  }
65                              });
66                          }
67                          FirebaseDatabase database =
68  FirebaseDatabase.getInstance();
69                          DatabaseReference myRef =
70  database.getReference("MonitoringUsaha").child(model.getIdUsaha(
71  )).child(tanggal.getText().toString());
72                          myRef.setValue(model);
73
74  Toast.makeText(PelaporanUsahaActivity.this,"Pelaporan Usaha
75  Telah Tersimpan", Toast.LENGTH_SHORT).show();
76                          finish();
77                      } else {
78
79  Toast.makeText(PelaporanUsahaActivity.this,"Tolong Isi Angsuran
80  Sesuai Dengan Angsuran Yang Tertera", Toast.LENGTH_LONG).show();
81                      }
82
83                  } else{
84
85  Toast.makeText(PelaporanUsahaActivity.this,"Tolong Isi Semua
86  Form", Toast.LENGTH_SHORT).show();
87                  }
```

Kode Program 5.4 Kode Program Untuk Pelaporan Usaha (lanjutan)

Berikut merupakan penjelasan dari kode program untuk pelaporan usaha yang ada dalam Kode Program 5.3 dan Kode Program 5.4. Yaitu terdapat sebuah selesksi kondisi *if* dimana dengan syarat untuk melihat semua variabel teresi dan tidak kosong. Kemudiakan akan memanggil kelas model untuk melakukan *method set* dimana akan memasukkan data dari setiap variabel yang bertipe EditText. Terdapat variabel waktu untuk mengambil jangka waktu pengembalian. Setelah itu terdapat rumus untuk mendapat nilai angsuran dana total yaitu angsuran = (dana dibutuhkan X 15 / 100) + dana dibutuhkan)) / waktu dan nilai angsuran dana untuk investor yaitu angsuran = (dana dibutuhkan X 10 / 100) + dana dibutuhkan)) / waktu. Dimana nantinya akan dilakukan selesksi kondisi kembali untuk melihat apakah angsuran yang telah di masukkan sesuai dengan angsuran total yang harus dibayarkan. Setelah itu akan dilakukan pembagian hasil dari setiap investor yang telah melakukan investasi dengan menggunakan perulangan *for.* Dimana akan masuk kedalam masing-masing virtual account investor dengan rumus mPengembalian = jumlah investasi yang di berikan / total seluruh investasi * angsuran untuk mitra sebanyak 10%. Lalu semua data akan masuk juga kedalam monitoring usaha. Terdapat juga beberapa informasi ketika sudah melakukan method tersebut.

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

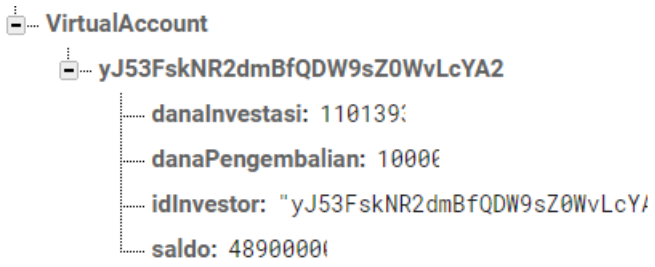
5.2.3 Implementasi Basis Data

Pada implementasi basis data ini akan memaparkan hasil dari perancangan basis data. Dimana pada implementasi basis data yang digunakan kedalam *firebase*. Pada implementasi basis data akan dipaparkan pada gambar 5.29 sampai 5.35.

Gambar 5.29 Implementasi Basis Data UserManagement

Pada Gambar 5.29 merupakan implementasi basis data UserManagement. Yang mana terdapat data email, jenis, namabelakang, namadepan, statusVerifikasi, dan userId. Yang telah dibuat pada *firebase* dengan struktur *noSQL*.

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA



Gambar 5.30 Implementasi Basis Data VirtualAccount

Pada Gambar 5.30 merupakan implementasi basis data VirtualAccount. Yang mana terdapat data danaInvestasi, danaPengembalian, idInvestor, dan saldo. Yang telah dibuat pada *firebase* dengan struktur *noSQL*.

Gambar 5.31 Implementasi Basis Data VerifikasiInvestasi

Pada Gambar 5.31 merupakan implementasi basis data VerifikasiInvestasi. Yang mana terdapat data danaInvestasi, idUsaha, namaUsaha, dan tglPembiayaan. Yang telah dibuat pada *firebase* dengan struktur *noSQL*.

repository.ub.ac.id


```
MitraManagement
  4yApgFAn6vcGjwfxYDmygtteW8I2
  deoYwsxFx7ZZlJ92mLLmlDcmcWQ2
    alamat: "jl.komik
    alamatKtp: "jl.ko'
    email: "juku@gmail.com
    foto: "https://firebasestorage.googleapis.com/v0/b/mas
    fotoKtp: "https://firebasestorage.googleapis.com/v0/b/mas
    nama: "juki cuki
    noHp: "085734348
    noKtp: "828394903
    statusMitra: 1
    userId: "deoYwsxFx7ZZlJ92mLLmlDcmcW(
```


Gambar 5.32 Implementasi Basis Data MitraManagement

Pada Gambar 5.32 merupakan implementasi basis data MitraManagement. Yang mana terdapat data alamat, alamatKtp, email, foto, fotoKtp, nama, noHp, noKtp, statusMitra, dan userId. Yang telah dibuat pada *firebase* dengan struktur *noSQL*.


```
InvestorManagement
  Qjiq1QfM0NUrlh1tNpMhVL9jE9p2
    alamat: "jl.mu'
    email: "runi@gmail.com
    foto: "https://firebasestorage.googleapis.com/v0/b/mas
    fotoKtp: "https://firebasestorage.googleapis.com/v0/b/mas
    nama: "runi marluni
    noHp: "08512346648
    noKTP: "0884664646
    noTelp: "012346483
    statusInvestor: 0
    userId: "Qjiq1QfM0NUrlh1tNpMhVL9jE9p
  yJ53FskNR2dmBfQDW9sZ0WvLcYA2
```


Gambar 5.33 Implementasi Basis Data InvestorManagement

Pada Gambar 5.33 merupakan implementasi basis data InvestorManagement. Yang mana terdapat data alamat, email, foto, fotoKtp, nama, noHp, noKtp,noTelp, statusInvestor, dan userId. Yang telah dibuat pada *firebase* dengan struktur *noSQL*.

UNIVERSITAS BRAWIJAYA

repository.ub.ac.id


```
PengajuanUsaha
  2a0da493-f857-43e5-b727-58307c8e0ecc
    alamatUsaha: "jl.damai
    bidangUsaha: "kuliner
    danaDibutuhkan: 200000(
    danaYangTerkumpul: 0
    diskripsiUsaha: "makanan
    email: "mitra@gmail.com
    foto1: "https://firebasestorage.googleapis.com/v0/b/mas
    foto2: "https://firebasestorage.googleapis.com/v0/b/mas
    jangkaWaktu: "12 bulan
    key: "2a0da493-f857-43e5-b727-58307c8e0e
    margin: 300000(
    mitraId: "4yApgFAn6vcGjwfxYDmygtteW8]
    nama: "mitra'
    namaUsaha: "warung asri
    proposal: "https://firebasestorage.googleapis.com/v0/b/mas
    statusUsaha: 2
    tanggalPengajuan: 14, 02, 2019"
    tujuanPengajuan: "membuat rumah maka
  44a081bd-7f81-4bb3-b989-d60613a1ed03
```


Gambar 5.34 Implementasi Basis Data PengajuanUsaha

Pada Gambar 5.34 merupakan implementasi basis data PengajuanUsaha. Yang mana terdapat data alamatUsaha, bidangUsaha, danaDibutuhkan, danaYangTerkumpul, diskripsiUsaha, email, foto1, foto2, jangkaWaktu, key, margin, mitraId, nama, namaUsaha, proposal, statusUsaha, tanggalPengajuan, dan tujuanPengajuan. Yang telah dibuat pada *firebase* dengan struktur *noSQL*.


```
MonitoringUsaha
  53ede635-38c3-4d7e-8948-d3e0800af2fa
    19-12-2018
      angsuran: 0
      biayaOprasional: 123
      idUsaha: "53ede635-38c3-4d7e-8948-d3e0800af2
      labaBersih: 321
      labaKotor: 510000(
      tanggal: "19-12-2018
      totalPenjualan: 123
    23-05-2018
```


Gambar 5.35 Implementasi Basis Data MonitoringUsaha

Pada Gambar 5.35 merupakan implementasi basis data PengajuanUsaha. Yang mana terdapat data angsuran, biayaOprasional, idUsaha, labaBersih,

UNIVERSITAS BRAWIJAYA

labaKotor, tanggal, dan totalPenjualan. Yang telah dibuat pada *firebase* dengan struktur *noSQL*.

5.2.4 Implementasi Antarmuka

Pada implementasi antarmuka ini memaparkan hasil dari perancangan antarmuka. Dimana implementasi antarmuka digunakan kedalam *markup language* XML. Dalam implementasi antarmuka pada penelitian ini akan dipaparkan pada sub bab 5.2.4.1 hingga 5.2.4.22.

5.2.4.1 Implementasi Antarmuka *Login*

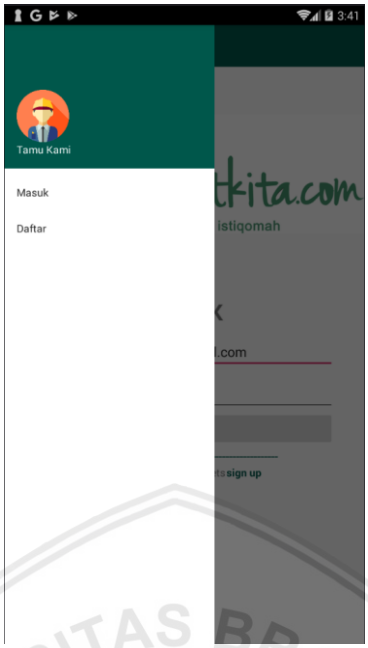
Pada Gambar 5.36 merupakan gambar hasil dari implementasi antar muka Halaman *login* dimana halaman ini yang akan digunakan pengguna untuk melakukan hak aksesnya terhadap sistem. Sehingga nantinya akan menampilkan halaman sesuai kategori hak akses yang telah di peruntukkan nantinya. dan terdapat tombol *sign in* untuk melakukan *login* pada hak akses yang diinginkan.

Gambar 5.36 Implementasi Antarmuka *Login*

5.2.4.2 Implementasi Antarmuka *Guest*

Pada Gambar 5.37 merupakan gambar hasil dari implementasi antarmuka navigasibar *guest* dimana halaman ini yang akan digunakan pengguna untuk melakukan aktivitas apa saja yang bisa dilakukan. Terdapat tombol pada navigasibar tersebut Masuk dan Daftar. Dimana tombol masuk untuk menampilkan halaman masuk atau halam *login* nantinya. Sedangkan untuk tombol daftar untuk menampilkan halaman daftar.

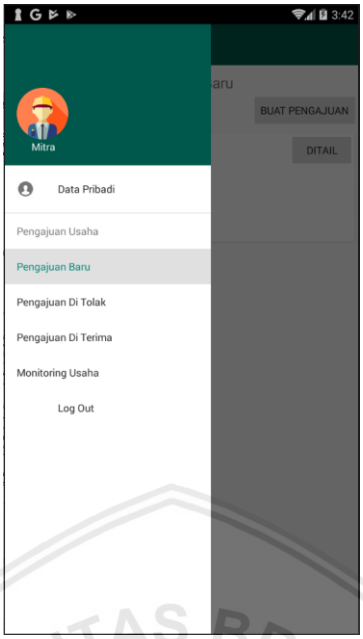
repository.ub.ac.id
UNIVERSITAS BRAWIJAYA



Gambar 5.37 Implementasi Antarmuka *Guest*

5.2.4.3 Implementasi Antarmuka Mitra

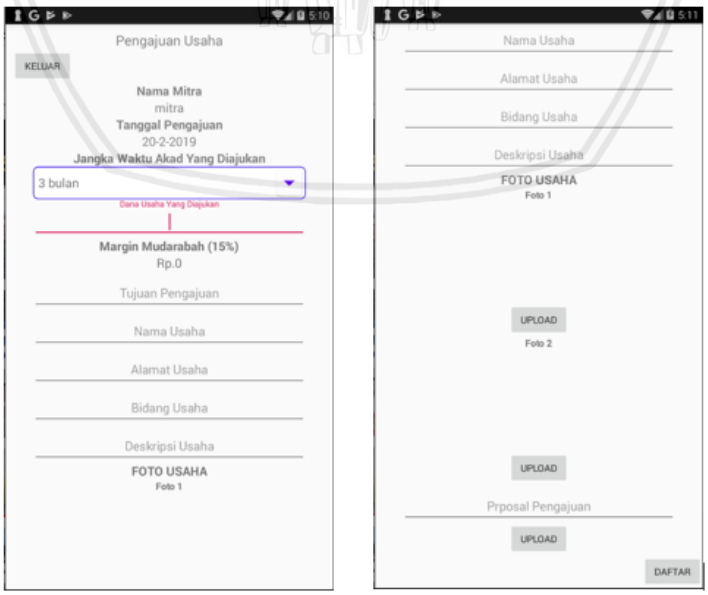
Pada Gambar 5.38 merupakan gambar hasil dari implementasi antarmuka navigasibar mitra dimana navigasibar ini yang akan digunakan pengguna untuk melakukan aktivitas apa saja yang bisa dilakukan. Terdapat tombol pada navigasibar tersebut data pribadi,pengajuan baru, pengajuan ditolak, pengajuan ditermia, dan monitoring usaha. Dimanapada setiap tombol akan menampilkan halaman yang akan dituju. Dan semua aktivitas tersebut yang dapat dilakukan oleh hak akses sebagai mitra.



Gambar 5.38 Implementasi Antarmuka Mitra

5.2.4.4 Implementasi Antarmuka Pengajuan Usaha

Pada Gambar 5.39 merupakan gambar hasil dari implementasi antarmuka halaman pengajuan usaha dimana halaman ini yang akan digunakan pengguna untuk melakukan pengajuan usaha yang akan dilakuakan. Terdapat *spinner* untuk melakukan pemilihan tengganggan waktu. Juga terdapat berberapa *form* untuk mengisi data yang dibutuhkan. Dan juga tombol untuk mengunggah foto maupun dokumen nantinya. Terutama foto usaha dan juga dokumen proposal pengajuan usaha.



Gambar 5.39 Implementasi Antarmuka Pengajuan Usaha

5.2.4.5 Implementasi Antarmuka Monitoring Usaha Mitra

Pada Gambar 5.40 merupakan gambar hasil dari implementasi antarmuka halaman pengajuan usaha dimana halaman ini yang akan digunakan pengguna untuk melakukan pengajuan usaha yang akan dilakuakan. Terdapat *spinner* untuk melakukan pemilihan tengganggan waktu. Juga terdapat berberapa *form* untuk mengisi data yang dibutuhkan. Dan juga tombol untuk mengunggah foto maupun dokumen nantinya. Terutama foto usaha dan juga dokumen proposal pengajuan usaha.

Gambar 5.40 Implementasi Antarmuka *Monitoring* Usaha Mitra

5.2.4.6 Implementasi Antarmuka Pelaporan Usaha

Pada Gambar 5.41 merupakan gambar hasil dari implementasi antarmuka halaman pelaporan usaha dimana halaman ini yang akan digunakan pengguna untuk melakukan pelporoan usaha maupun melakukan pengangsuran yang akan dilakuakan. Terdapat beberapa *form* untuk diisi data-data yang dibutuhkan. Dan juga terdapat dua tombol untuk menyimpan data pelaporan usaha yang sudah di isi yaitu tombol simpan. Dan juga terdapat tombol untuk keluar pada halaman tersebut yaitu tombol keluar.

Gambar 5.41 Implementasi Antarmuka Pelaporan Usaha

5.2.4.7 Implementasi Antarmuka *List* Pengajuan Usaha Baru

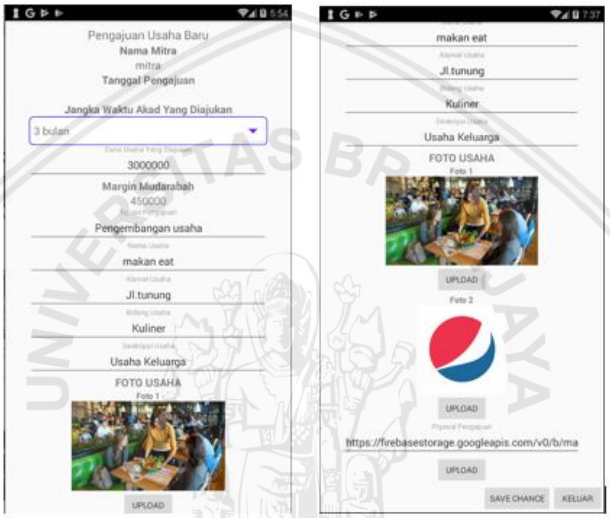
Pada Gambar 5.42 merupakan gambar hasil dari implementasi antarmuka halaman *list* pengajuan usaha baru dimana halaman ini yang akan digunakan pengguna untuk melihat daftar dari pengajuan usaha yang baru di daftarkan. Terdapat *recyclerview* untuk menamplikan daftar pengajuan dan terdapat beberapa tombol untuk fungsi tertentu. Terdapat tombol detail untuk melihat detail dari pengajuan usaha yang dilakukan. Dan terdapat juga tombol buat untuk membuat pengajuan usaha baru.

Gambar 5.42 Implementasi Antarmuka *List* Pengajuan Usaha Baru

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

5.2.4.8 Implementasi Antarmuka Detail Pengajuan Usaha Baru

Pada Gambar 5.43 merupakan gambar hasil dari implementasi antarmuka halaman detail pengajuan usaha baru dimana halaman ini yang akan digunakan pengguna untuk melihat detail dari pengajuan usaha yang baru di daftarkan serta dapat melakukan perubahan pada data yang telah didaftarkan. Terdapat beberapa *form* untuk merubah data yang dinginkan. Serta terdapat tombol beberapa tombol untuk melakukan fungsi yang berbeda. Terdapat tombol *upload* untuk menggunggah foto dan dokumen proposal yang dinginkan. Serta terdapat tombol *save chance* untuk menyimpan data yang telah dirubah. Dan terdapat tombol keluar untuk keluar dari halaman tersebut. Maupun *spinner* utuk merubah jangka waktu pengangsuran.



Gambar 5.43 Implementasi Antarmuka Detail Pengajuan Usaha Baru

5.2.4.9 Implementasi Antarmuka *List* Pengajuan Usaha Ditolak

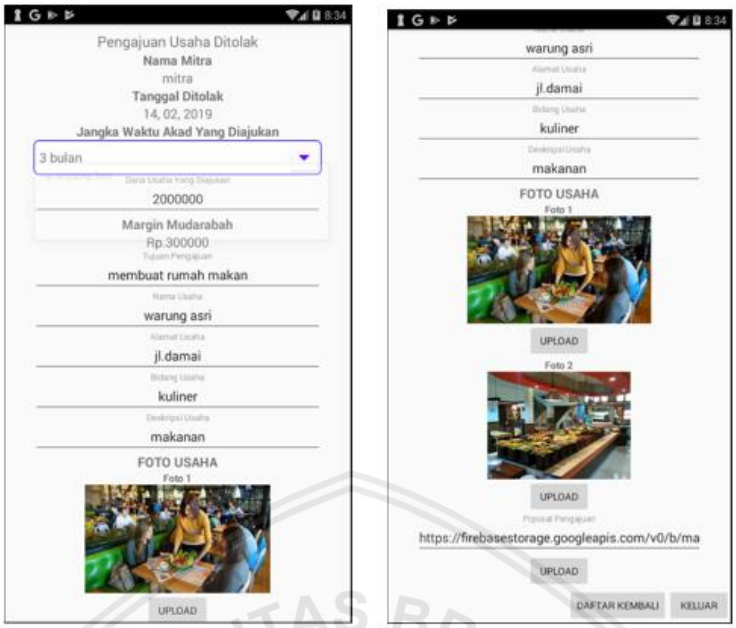
Pada Gambar 5.44 merupakan gambar hasil dari implementasi antarmuka halaman *list* pengajuan usaha ditolak dimana halaman ini yang akan digunakan pengguna untuk melihat daftar dari pengajuan usaha yang telah ditolak oleh admin maslahatkita. Terdapat *recyclerview* untuk menamplikan daftar pengajuan yang telah ditolak dan terdapat beberapa tombol untuk fungsi tertentu. Terdapat tombol detail untuk melihat detail dari pengajuan usaha yang dilakukan.

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

Gambar 5.44 Implementasi Antarmuka List Pengajuan Usaha Ditolak

5.2.4.10 Implementasi Antarmuka Detail Pengajuan Usaha Ditolak

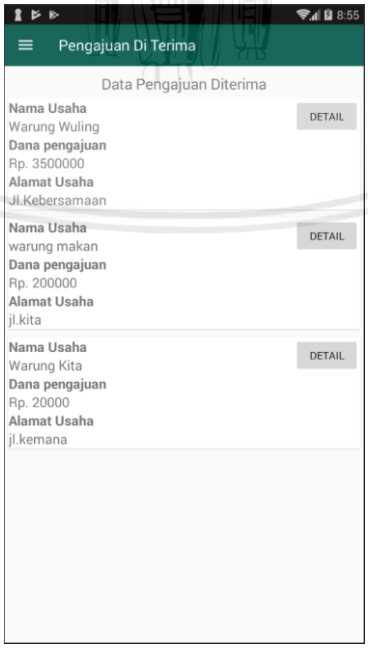
Pada Gambar 5.45 merupakan gambar hasil dari implementasi antarmuka halaman detail pengajuan usaha ditolak dimana halaman ini yang akan digunakan pengguna untuk melihat detail dari pengajuan usaha yang telah ditolak serta dapat melakukan perubahan pada data yang telah ditolak. Terdapat beberapa *form* untuk merubah data yang dinginkan. Serta terdapat tombol beberapa tombol untuk melakukan fungsi yang berbeda. Terdapat tombol *upload* untuk menggunggah foto dan dokumen proposal yang dinginkan. Serta terdapat tombol daftar kembali untuk menyimpan data yang telah dirubah dan mengajukan kembali usaha yang telah ditolak. Dan terdapat tombol keluar untuk keluar dari halaman tersebut. Maupun *spinner* utuk merubah jangka waktu pengangsuran.



Gambar 5.45 Implementasi Antarmuka Detail Pengajuan Usaha Ditolak

5.2.4.11 Implementasi Antarmuka *List* Pengajuan Usaha Diterima

Pada Gambar 5.46 merupakan gambar hasil dari implementasi antarmuka halaman *list* pengajuan usaha diterima dimana halaman ini yang akan digunakan pengguna untuk melihat daftar dari pengajuan usaha yang telah diterima oleh admin maslahatkita. Terdapat *recyclerview* untuk menamplikan daftar pengajuan yang telah ditolak dan terdapat beberapa tombol untuk fungsi tertentu. Terdapat tombol detail untuk melihat detail dari pengajuan usaha yang dilakukan.



Gambar 5.46 Implementasi Antarmuka *List* Pengajuan Diterima

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

5.2.4.12 Implementasi Antarmuka Detail Pengajuan Usaha Diterima

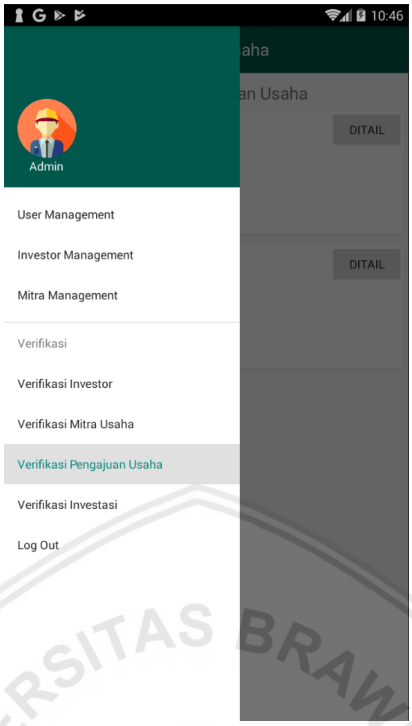
Pada Gambar 5.47 merupakan gambar hasil dari implementasi antarmuka halaman detail pengajuan usaha diterima dimana halaman ini yang akan digunakan pengguna untuk melihat detail dari pengajuan usaha yang telah diterima. Terdapat beberapa *text view* untuk memperlihatkan setiap data yang sudah diterima. Dan juga terdapat tombol *download* untuk mengunduh dokumen proposal. Dan juga terdapat tombol keluar untuk keluar dari halaman tersebut.

Gambar 5.47 Implementasi Antarmuka Detail Pengajuan Usaha Diterima

5.2.4.13 Implementasi Antarmuka Admin

Pada Gambar 5.48 merupakan gambar hasil dari implementasi antarmuka navigasibar admin dimana navigasibar ini yang akan digunakan pengguna untuk melakukan aktivitas apa saja yang bisa dilakukan. Terdapat tombol pada navigasibar tersebut *user management*, investor *management*, mitra *management*, verifikasi investor, verifikasi mitra usaha, verifikasi pengajuan usaha, verifikasi investasi, monitoring usaha, dan *log out*. Dimanapada setiap tombol akan menampilkan halaman yang akan dituju. Dan semua aktivitas tersebut yang dapat dilakukan oleh hak akses sebagai admin.

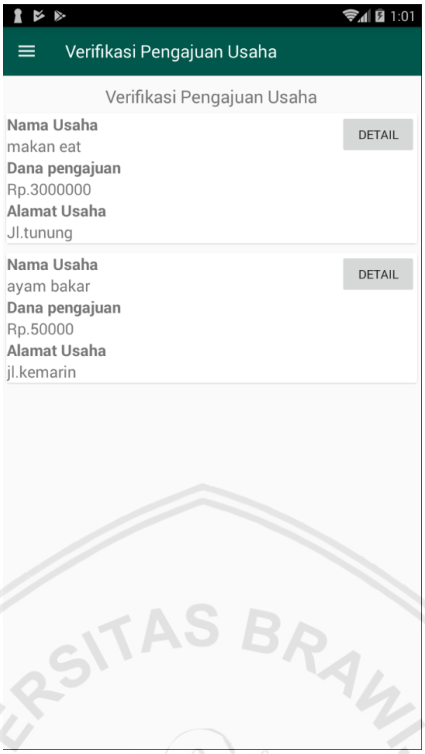
repository.ub.ac.id
UNIVERSITAS BRAWIJAYA



Gambar 5.48 Implementasi Antarmuka Admin

5.2.4.14 Implementasi Antarmuka *List* Verifikasi Pengajuan Usaha

Pada Gambar 5.49 merupakan gambar hasil dari implementasi antarmuka halaman *list* verifikasi pengajuan usaha dimana halaman ini yang akan digunakan pengguna untuk melihat daftar dari pengajuan usaha yang telah diterima oleh admin maslahatkita. Terdapat *recyclerview* untuk menamplikan daftar pengajuan yang telah ditolak dan terdapat beberapa tombol untuk fungsi tertentu. Terdapat tombol detail untuk melihat detail dari pengajuan usaha yang dilakukan.

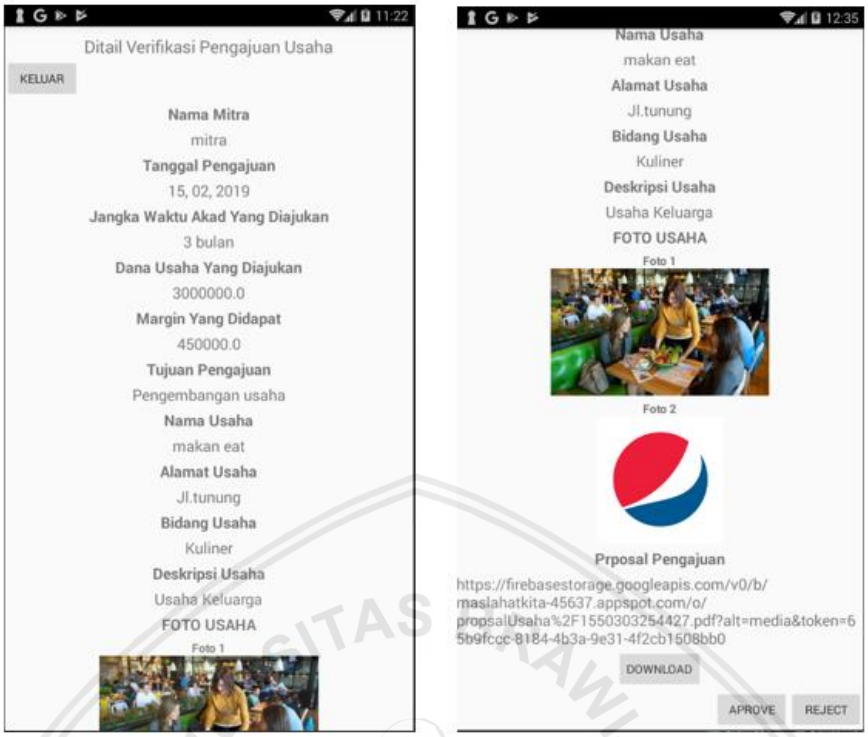


Gambar 5.49 Implementasi Antarmuka *List* Verifikasi Pengajuan Usaha

5.2.4.15 Implementasi Antarmuka Detail Verifikasi Pengajuan Usaha

Pada Gambar 5.50 merupakan gambar hasil dari implementasi antarmuka halaman detail verifikasi pengajuan usaha dimana halaman ini yang akan digunakan pengguna untuk melihat detail dari pengajuan usaha yang telah dididaftarkan. Terdapat beberapa *text view* untuk memperlihatkan setiap data yang sudah diterima. Dan juga terdapat tombol *download* untuk mengunduh dokumen proposal. Dan juga terdapat tombol keluar untuk keluar dari halaman tersebut.

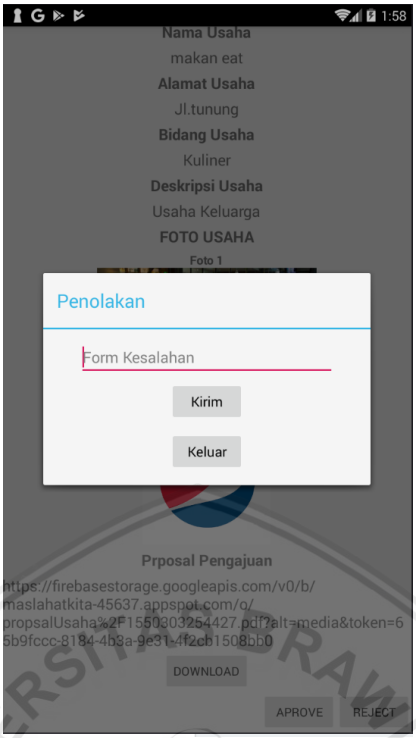
repository.ub.ac.id
UNIVERSITAS BRAWIJAYA



Gambar 5.50 Implementasi Antarmuka Detail Verifikasi Pengajuan Usaha

5.2.4.16 Implementasi Antarmuka Detail *Form* Penolakan

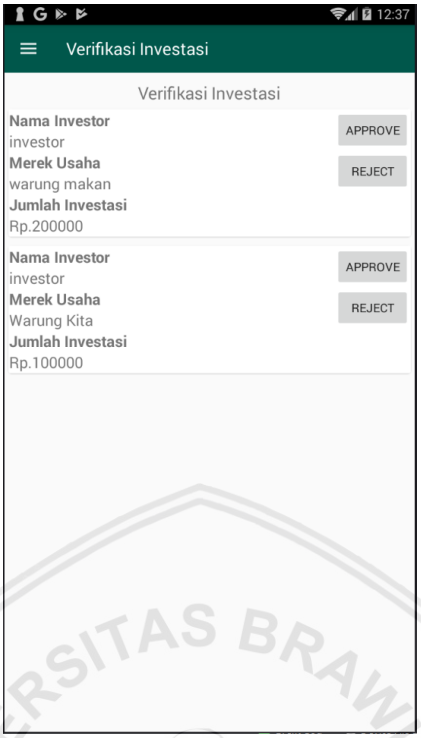
Pada Gambar 5.51 merupakan gambar hasil dari implementasi antarmuka halaman *form* penolakan dimana halaman ini yang akan digunakan pengguna untuk memberi pesan terhadap mitra usaha. Terdapat satu *form untuk* memberi pesan tersebut kepada mitra melalui *e-mail*. Dan juga terdapat tombol kirim untuk mengirim pesan . Dan juga terdapat tombol keluar untuk keluar dari halaman tersebut.



Gambar 5.51 Implementasi Antarmuka *Form* Penolakan

5.2.4.17 Implementasi Antarmuka *List* Verifikasi Investasi

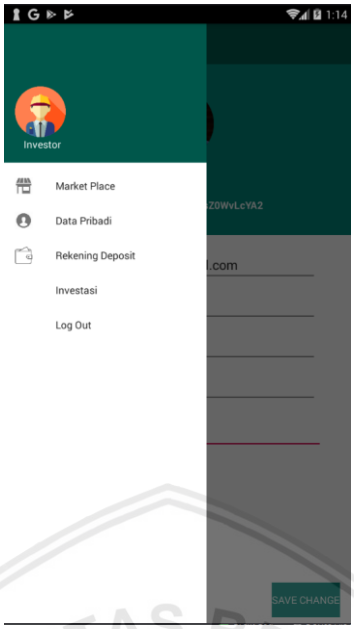
Pada Gambar 5.52 merupakan gambar hasil dari implementasi antarmuka halaman *list* verifikasi investasi dimana halaman ini yang akan digunakan pengguna untuk melihat daftar investai yang telah diajukan oleh investor. Terdapat *recyclerview* untuk menamplikan daftar pengajuan yang telah ditolak dan terdapat beberapa tombol untuk fungsi tertentu. Terdapat tombol *approve* untuk melakukan persetujuan pada investasi. Dan juga terdapat tombol *reject* untuk melakukan penolakan pada invsatasi tersebut.



Gambar 5.52 Implementasi Antarmuka *List* Verifikasi Investasi

5.2.4.18 Implementasi Antarmuka Investor

Pada Gambar 5.53 merupakan gambar hasil dari implementasi antarmuka navigasibar investor dimana navigasibar ini yang akan digunakan pengguna untuk melakukan aktivitas apa saja yang bisa dilakukan. Terdapat tombol pada navigasibar tersebut *marker place*, data pribadi, investasi,dan *log out*. Dimanapada setiap tombol akan menampilkan halaman yang akan dituju. Dan semua aktivitas tersebut yang dapat dilakukan oleh hak akses sebagai investor.



Gambar 5.53 Implementasi Antarmuka Investor

5.2.4.19 Implementasi Antarmuka *Market Place*

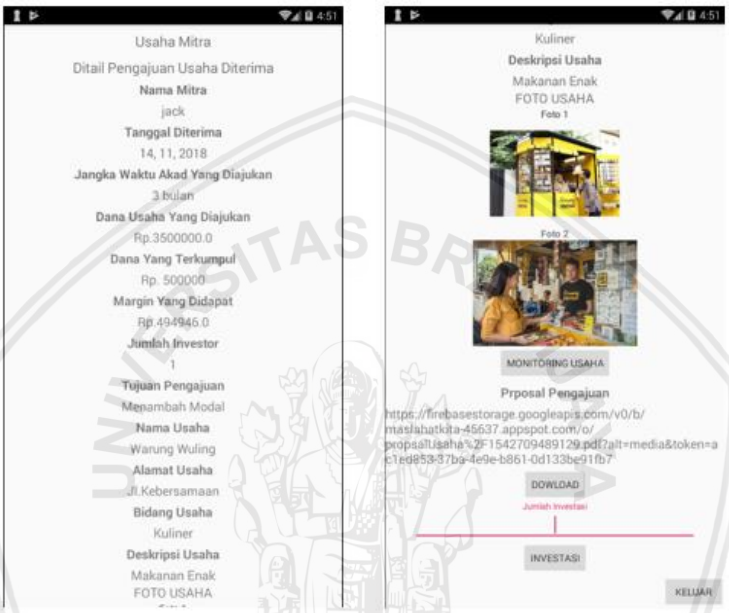
Pada Gambar 5.54 merupakan gambar hasil dari implementasi antarmuka halaman *market place* dimana halaman ini yang akan digunakan pengguna untuk melihat daftar dari usaha yang telah diverifikasi. Terdapat *recyclerview* untuk menamplikan daftar usaha milik mitra dan terdapat beberapa tombol untuk fungsi tertentu. Terdapat tombol detail untuk melihat detail dari pengajuan usaha yang dilakukan.

Gambar 5.54 ImplementasiAntarmuka *Market Place*

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

5.2.4.20 Implementasi Antarmuka Detail Verifikasi Pengajuan Usaha

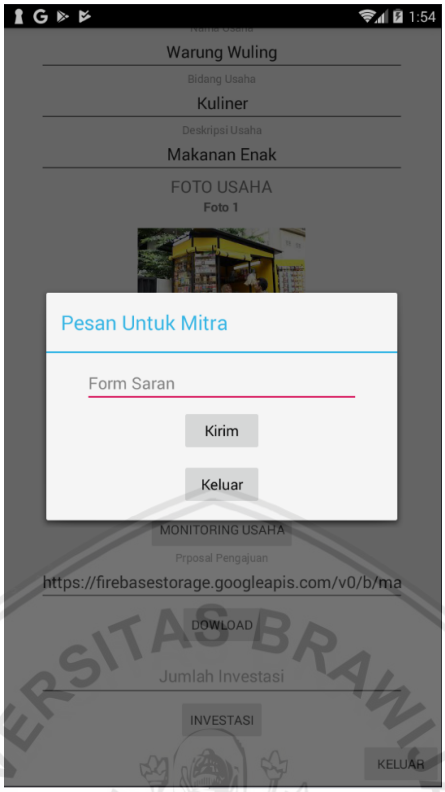
Pada Gambar 5.56 merupakan gambar hasil dari implementasi antarmuka halaman detail detail usaha mitra dimana halaman ini yang akan digunakan pengguna untuk melihat detail dari pengajuan usaha yang telah dididaftarkan. Terdapat beberapa *text view* untuk memperlihatkan setiap data yang sudah diterima. Terdapat *form* untuk menentukan jumlah investasi yang diinginkan. Terdapat tombol *download* untuk mengunduh dokumen proposal. Terdapat tombol investasi untuk melakukan investasi terhadap usaha tersebut. Dan juga terdapat tombol keluar untuk keluar dari halaman tersebut.



Gambar 5.55 Implementasi Antarmuka Detail Usaha Mitra

5.2.4.21 Implementasi Antarmuka *Form* Saran

Pada Gambar 5.56 merupakan gambar hasil dari implementasi antarmuka halaman *form* saran dimana halaman ini yang akan digunakan pengguna untuk memberi pesan terhadap mitra usaha. Terdapat satu *form untuk* memberi pesan tersebut kepada mitra melalui *e-mail*. Dan juga terdapat tombol kirim untuk mengirim pesan saran. Dan juga terdapat tombol keluar untuk keluar dari halaman tersebut.

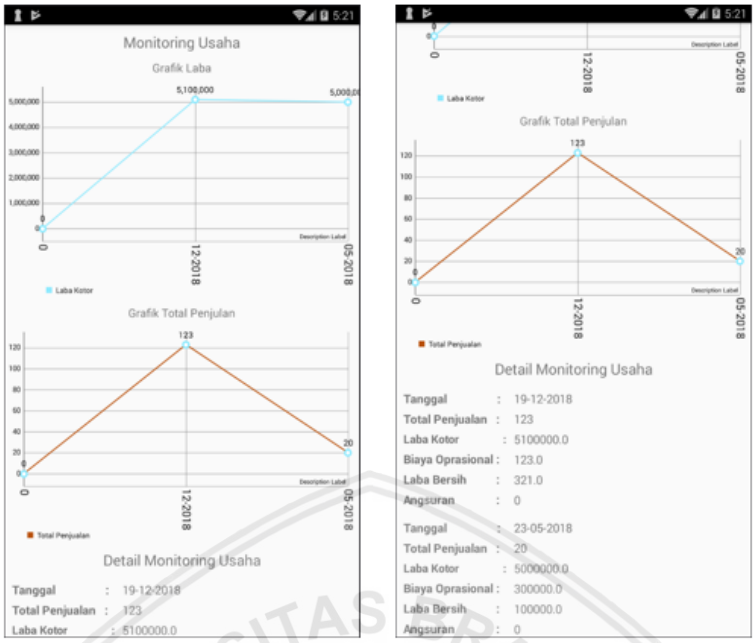


Gambar 5.56 Implementasi Antarmuka *Form* Saran

5.2.4.22 Implementasi Antarmuka Detail *Form* Saran

Pada Gambar 5.57 merupakan gambar hasil dari implementasi antarmuka halaman *monitoring* usaha untuk investor dimana halaman ini yang akan digunakan pengguna untuk melihat perkambangan usaha yang dilakukan oleh mitra. Terdapat dua grafik garis untuk melihat laba dan total penjualan setiap bulannya. Dan juga terdapat *recyclerview* untuk menamplikan daftar perkembangan usaha yang dilakukan setiap bulannya.

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA



Gambar 5.57 Implementasi Antarmuka *Monitoring* Usaha Untuk Investor

BAB 6 PENGUJIAN

6.1 Pengujian Unit

Pengujian unit merupakan pengujian dimana pada setiap unit dari sebuah perangkat lunak yang sedang dikembangkan. Pada pengujian unit akan di gunakan metode *white box* dimana dengan pengjuan ini akan dilakukan dengan *path testing*. Yang mana terdapat pada tigaa *method* yang akan di uji yaitu *method onClick* pada kelas *PengajuanUsahaActivity*, *method onDataChange* pada kelas *VerfikasiInvestasiFragment*, dan *onClick* pada kelas *PelaporanUsahaActivity*.

6.1.1 Pengujian Unit Kelas PengajuanUsahaActivity Pada Operasi onClick

1. *Pseudocode*

Pengujian unit dipaparkan pada *method onClick* dari kelas *PengajuanUsahaActivity* akan dijelaskan algoritme yang ditunjukkan dalam Pseudocode 6.1 dan Pseudocode 6.2.

Pseudocode	***No. Operasi***
Set EditText tujuanPengajuan,alamatUsaha,bidangUsaha, deskripsiUsaha, proposal, namaUsaha *Set* Uri filePathFoto1,filePathFoto2, filePathProposal	1
If (nama != 0 & tanggalPengajuan != 0 & danaDiajukan != 0 & margin != 0 & tujuanPengajuan != 0 & namaUsaha != 0 & alamatUsaha != 0 & bidangUsaha != 0 & deskripsiUsaha != 0 & filePathFoto1 != kosong & filePathFoto2 != kosong & filePathFoto != kosong)	2
Set nama pada kelas model Set jumlahWaktu pada kelas model Set danaDiButuhkan pada kelas model Set margin pada kelas model Set tujuanPengajuanpada kelas model Set namaUsaha pada kelas model Set alamatUsaha pada kelas model Set bidangUsaha pada kelas model Set deskripsiUsaha pada kelas model Set mitraId dengan intent dari kelas lain pada kelas model Set email dengan intent dari kelas lain pada kelas model Set statusUsaha = 0 pada kelas model Memanggil method uploadFoto1 Memanggil method uploadFoto2	3

Pseudocode 6.1 Algoritme *method onClick*

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

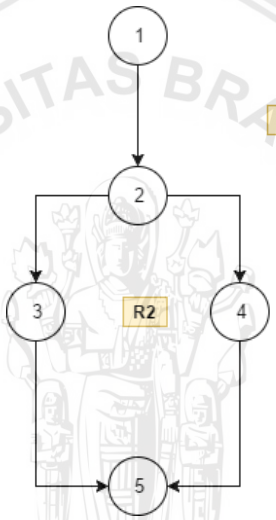
Memanggil method uploadFile Memberi informasi “Berhasil Mendaftar”	3
else Memberikan informasi “Tolong Isi Semua Form	4
end if	5

Pseudocode 6.2 Algoritme *method onClick* (lanjutan)

2. *Basis Path Testing*

2.1 *Flow Graph*

Dari *pseudocode* yang telah dibuat akan menjadi dasar dari pembuatan *flow graph* pada *method onClick* dari kelas *PengajuanUsahaActivity* dipaparkan pada Gambar 6.1.



Gambar 6.1 *Flow Graph Method onClick*

2.2. *Cyclomatic Complexity*

Dari gambar 6.1 *flow graph* maka diketahui jumlah dari jalur *independent* melalui kalkulasi yang dinamakan kalkulasi *cyclomatic complexity*. Berikut merupakan hasil dari kalkulasi *cyclomatic complexity* pada *flow graph method onClick.*

- V(G) = jumlah *region* = 2
- V(G) = jumlah *edge* – jumlah *node* + 2 = 5 – 5 + 2 = 2
- V(G) = jumlah *predicate* + 1 = 1 + 1 = 2

2.3 *Independent Path*

- Jalur 1 : 1 - 2 - 3 - 5
- Jalur 2 : 1 - 2 - 4 – 5
-

Tabel 6.1 Hasil Pengujian Unit *Method onClick*

Jalur Uji	Prosedur Uji	*Expected Result*	*Result*	Status
1	Semua varibel pada syarat *if* bernilai *null* atau kosong atau terdapat beberapa yang kosong	Memunculkan pemeberitahuan pada *method makeText* dari kelas *Toast* "Tolong Isi Semua Form"	Memunculkan pemeberitahuan pada *method makeText* dari kelas *Toast* "Tolong Isi Semua Form"	Valid
2	Semua varibel pada syarat *if* tidak kosong atau tidak *null*	Semua data yang di isi tersimpan dalam *firebase* dan memunculkan informasi "Berhasil Mendaftar Usaha"	Semua data yang di isi tersimpan dalam *firebase* dan memunculkan informasi "Berhasil Mendaftar Usaha"	Valid

6.1.2 Pengujian Unit Kelas VerifikasiInvestasiFragment untuk operasi onDataChange

1. *Pseudocode*

Pengujian unit dipaparkan pada *method onDataChange* dari kelas *VerifikasiInvestasiFragment* akan dijelaskan algoritme yang ditunjukkan dalam Pseudocode 6.3 dan Pseudocode 6.4.

Pseudocode	**No. Operasi**
Set bersihkan array list	1
for sebanyak dataSnapshot	2
Membuat objek ps untuk PengajuanUsahaModel Membuat HashMap untuk objek investor dengan tipe *String*	3
If (objek investor tidak boleh kosong)	4
For sebanyak insvestor	5
If (status investor tidak kosong)	6
Membuat objek approveModel	7

Pseudocode 6.3 Algoritme *method onDataChange*

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

Pseudocode	**No. Operasi**
Set jumlahInvestasi dari objek approveModel Set namaInvestor dari objek approveModel Set namaUsaha dari objek approveModel Set idInvestor dari objek approveModel Set idPengajuanUsaha dari objek approveModel Set danaYangTerkumpul dari objek approveModel Masukkan objek approveModel list	7
End if	8
End for	9
End if	10
End for	11
Membuat objek adapter Ser layoutManager Set adapter	12

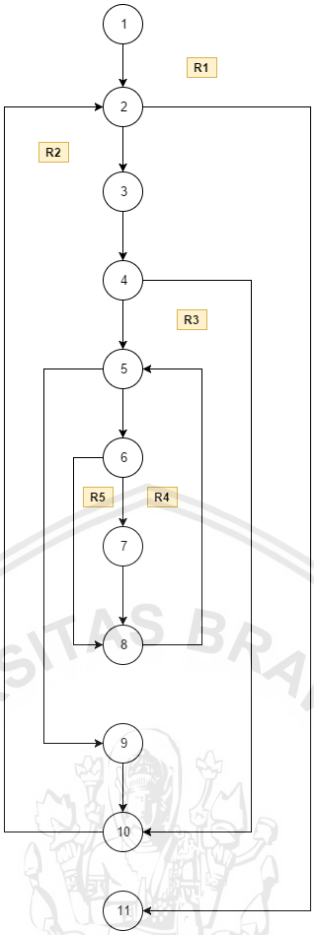
Pseudocode 6.4 Algoritme *method onDataChange* (lanjutan)

2. *Basis Path Testing*

2.1 *Flow Graph*

Dari *pseudocode* yang telah dibuat akan menjadi dasar dari pembuatan *flow graph* pada *method onDataChange* dari kelas *VerifikasiInvestasiFragment* dipaparkan pada Gambar 6.2.

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA



Gambar 6.2 *Flow Graph Method onDataChange*

2.2. *Cyclomatic Complexity*

Dari gambar 6.2 *flow graph* maka diketahui jumlah dari jalur *independent* melalui kalkulasi yang dinamakan kalkulasi *cyclomatic complexity*. Berikut merupakan hasil dari kalkulasi *cyclomatic complexity* pada *flow graph method onDataChange.*

- V(G) = jumlah *region* = 5
- V(G) = jumlah *edge* – jumlah *node* + 2 = 14 – 11 + 2 = 5
- V(G) = jumlah *predicate* + 1 = 4 + 1 = 5

2.3 *Independent Path*

- Jalur 1 : 1 - 2 - 11
- Jalur 2 : 1 - 2 - 3 - 4 - 10 - 2 - 11
- Jalur 3 : 1 - 2 - 3 - 4 - 5 - 9 - 10 - 2 – 11
- Jalur 4 : 1 - 2 - 3 - 4 - 5 - 6 - 8 - 9 - 10 - 2 - 11
- Jalur 5 : 1 - 2 - 3 - 4 - 5 - 6 - 7 - 8 - 5 - 9 - 10 - 2 – 11

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

Tabel 6.2 Hasil Pengujian Unit *Method onDataChange*

Jalur Uji	Prosedur Uji	*Expected Result*	*Result*	Status
1	Data yang ada pada *firebase* PengajuanUsaha kosong	Tidak memasukkan data dari kelas model Pengajuan-UsahaModel	Tidak memasukkan data dari kelas model Pengajuan-UsahaModel	Valid
2	Data yang ada pada *firebase* PengajuanUsaha tidak kosong dan mengambil semua data pengajuan usaha pada investor kosong atau *null*	Tidak menjalankan perulangan *for*	Tidak menjalankan perulangan *for*	Valid
3	Data yang ada pada *firebase* PengajuanUsaha tidak kosong dan mengambil semua data pengajuan usaha pada investor tidak kosong *method keySet()* kosong	Tidak menjalankan seleksi kondisi *if* dengan syarat status invetor	Tidak menjalankan seleksi kondisi *if* dengan syarat status invetor	Valid
4	Data yang ada pada *firebase* PengajuanUsaha tidak kosong dan mengambil semua data pengajuan usaha pada investor tidak kosong *method keySet()* kosong nilai *method getStatus*() = *true*	Tidak memasukkan data pada *firebase* kedalam model kelas *approveModel*	Tidak memasukkan data pada *firebase* kedalam model kelas *approveModel*	Valid

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

Tabel 6.3 Hasil Pengujian Unit *Method onDataChange* (lanjutan)

Jalur Uji	Prosedur Uji	*Expected Result*	*Result*	Status
5	Data yang ada pada *firebase* PengajuanUsaha tidak kosong dan mengambil semua data pengajuan usaha pada investor tidak kosong *method keySet()* kosong nilai *method getStatus*() = *false*	Data masuk dalam *list* dan dimunculkan ke dalam *recyclerview* melalui adapter	Data masuk dalam *list* dan dimunculkan ke dalam *recyclerview* melalui adapter	Valid

6.1.3 Pengujian Unit Kelas PelaporanUsahaActivity Untuk Operasi onClick

1. *Pseudocode*

Pengujian unit dipaparkan pada *method onClick* dari kelas *PelaporanUsahaActivity* akan dijelaskan algoritme yang ditunjukkan dalam Pseudocode 6.3.

Pseudocode	**No. Operasi**
If(totalPenjualan != 0 dan labaBersih != 0 dan labaKotor !=0 dan biayaOprasiaonal !=0 dan angsuran !=0)	1
Memuat idUsaha dari firebase Memuat totalPenjualan dari *form* Memuat labaBersih dari *form* Memuat labaKotor dari *form* Memuat biayaOprasional dari *form* Memuat tanggal waktu double = mengambil jangkaWaktu angsuran_usaha_total double =((danaDibutuhkan X 10 :100)+ danaDibutuhkan) : waktu angsuran_usaha_for_investor double =((danaDibutuhkan X 10 :100)+ danaDibutuhkan) : waktu	2
If (angsuran !=0 dan angsuran_usaha_total)	3

Pseudocode 6.5 Algoritme *method onClick*

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

Pseudocode	**No. Operasi**
listInvestor HashMap = mendapatkan investor li HashMap = mendapatkan investor	4
For sebanyak jumlah investor yang telah di setujui	5
mPengembalian double = jumlah investasi : total x angsuran_usaha Mengambil database VirtualAccount pada setiap investor Memanggil method onDataChange Memasukan data dari database kedalam model Memuat saldo awal + data mPengembalian Membuat isi database VirtualAccount dengan model yang sudah dirubah Memanggil *method* onCancelled	6
End for	7
Melakukan Instansiasi *firebase* Mengakses database pada Monitorring Usaha Menyimpan isi model pada *firebase* Memberikan infromasi "Pelaporan Usaha Telah Tersimpan" Menyudahi *activity* yang berjalan	8
Else Memberikan informasi "Tolong Isi Angsuran Sesuai Dengan Angsuran Yang Tertera"	9
End if	10
Else Memberikan informasi "Tolong Isi Semua Form"	11
End if	12

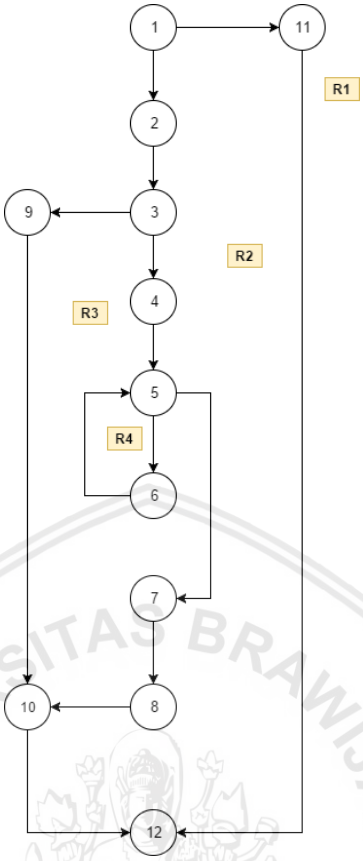
Pseudocode 6.6 Algoritme *method onClick* (lanjutan)

2. *Basis Path Testing*

2.1 *Flow Graph*

Dari *pseudocode* yang telah dibuat akan menjadi dasar dari pembuatan *flow graph* pada *method onClick* dari kelas *PelaporanUsahaActivity* dipaparkan pada Gambar 6.3.

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA



Gambar 6.2 *Flow graph method onClick*

2.2. *Cyclomatic Complexity*

Dari gambar 6.3 *flow graph* maka diketahui jumlah dari jalur *independent* melalui kalkulasi yang dinamakan kalkulasi *cyclomatic complexity*. Berikut merupakan hasil dari kalkulasi *cyclomatic complexity* pada *flow graph method onClick.*

- V(G) = jumlah *region* = 4
- V(G) = jumlah *edge* – jumlah *node* + 2 = 14 – 12 + 2 = 4
- V(G) = jumlah *predicate* + 1 = 3 + 1 = 4

2.3 *Independent Path*

- Jalur 1 : 1 - 14 - 15
- Jalur 2 : 1 - 2 - 3 - 12 - 13 - 15
- Jalur 3 : 1 - 2 - 3 - 4 - 5 - 7 - 8 - 10 - 11 - 13 - 15
- Jalur 4 : 1 - 2 - 3 - 4 - 5 - 6 - 5 - 7 - 8 - 9 - 8 - 10 - 11 - 13 - 15

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

Tabel 6.4 Hasil Pengujian Unit *Method onClick*

Jalur Uji	Prosedur Uji	*Expected Result*	*Result*	Status
1	Semua varibel pada syarat if bernilai null atau kosong atau terdapat beberapa yang kosong	Memunculkan pemeberitahuan pada *method makeText* dari kelas *Toast* "Tolong Isi Semua Form"	Memunculkan pemeberitahuan pada *method makeText* dari kelas *Toast* "Tolong Isi Semua Form"	Valid
2	Semua varibel pada syarat if tidak bernilai null atau tidak kosong dan Seleksi kondisi variabel angsuran = 0 dan angsuran tidak sama dengan angsuran_usaha_-total	Memunculkan pemeberitahuan pada *method makeText* dari kelas *Toast* " Tolong Isi Angsuran Sesuai Dengan Angsuran Yang Tertera"	Memunculkan pemeberitahuan pada *method makeText* dari kelas *Toast* " Tolong Isi Angsuran Sesuai Dengan Angsuran Yang Tertera"	Valid
3	Semua varibel pada syarat if tidak bernilai null atau tidak kosong dan Seleksi kondisi variabel angsuran tidak sama dengan 0 dan angsuran= angsuran_usaha_-total, tidak ada nilai pada *listInvestor* dan obejk *li*	Tidak melakukan kalkulasi penjumlahan untuk mendapat nilai investasi dari investor pada variabel total maupun memasukkan data mPengembalian kedalam *firebase*	Tidak melakukan kalkulasi penjumlahan untuk mendapat nilai investasi dari investor pada variabel total maupun memasukkan data mPengembalian kedalam *firebase*	Valid
4	Semua varibel pada syarat if tidak bernilai null atau tidak kosong dan Seleksi kondisi variabel	Melakukan kalkulasi penjumlahan untuk mendapat nilai investasi dari investor pada	Melakukan kalkulasi penjumlahan untuk mendapat nilai investasi dari investor	Valid

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

Tabel 6.5 Hasil Pengujian Unit *Method onClick* (lanjutan)

Jalur Uji	Prosedur Uji	*Expected Result*	*Result*	Status
4	angsuran tidak sama dengan 0 dan angsuran= angsuran_usaha_-total, tidak ada nilai pada *listinvestor*	variabel total maupun memasukkan data mPengembalian kedalam *firebase*	pada variabel total maupun memasukkan data mPengembalian kedalam *firebase*	Valid

6.2 Pengujuan Integrasi

Pada pengujian integrasi untuk penelitian ini bertujuan untuk menguji interaksi antar kelas berjalan dengan baik atau tidak. Dimana pada pengujian integrasi kali ini akan menggunakan pedekatan *white box* pada basii *path testing*. Yang mana pada pengujian kali ini akan menggunakan *method onDataChange* pada kelas *VerifikasiInvestasiFragment* dengan *method set* pada kelas *InvestorApproval* serta *method get* pada kelas *PengajuanUsahaModel.* Dalam interaksi dua kelas tersebut *method onDataChange* akan memanggil *method set* dan *method get*.

1. *Pseudocode*

Pengujian unit dipaparkan pada *method onDataChange* dari kelas *VerifikasiInvestasiFragment* akan dijelaskan algoritme yang ditunjukkan dalam Pseudocode 6.2.

Pseudocode	**No. Operasi**
Set bersihkan array list	1
for sebanyak dataSnapshot	2
Membuat objek ps untuk PengajuanUsahaModel Membuat HashMap untuk objek investor dengan tipe *String*	3
If (objek investor tidak boleh kosong)	4
For sebanyak insvestor	5
If (status investor tidak kosong)	6
Membuat objek approveModel Set jumlahInvestasi dari objek approveModel Set namaInvestor dari objek approveModel Set namaUsaha dari objek approveModel	7

Pseudocode 6.7 Algoritme *method onDataChange*

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

Pseudocode	**No. Operasi**
Set idInvestor dari objek approveModel Set idPengajuanUsaha dari objek approveModel Set danaYangTerkumpul dari objek approveModel Masukkan objek approveModel list	7
End if	8
End for	9
End if	10
End for	11
Membuat objek adapter Ser layoutManager Set adapter	12

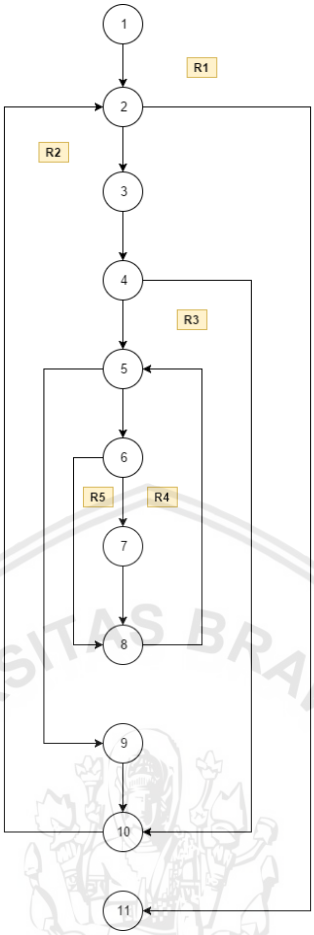
Pseudocode 6.8 Algoritme *method onDataChange* (lanjutan)

2. *Basis Path Testing*

2.1 *Flow Graph*

Dari *pseudocode* yang telah dibuat akan menjadi dasar dari pembuatan *flow graph* pada *method onDataChange* dari kelas *VerifikasiInvestasiFragment* dipaparkan pada Gambar 6.3.

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA



Gambar 6.3 *Flow Graph Method onDataChange*

2.2. *Cyclomatic Complexity*

Dari gambar 6.2 *flow graph* maka diketahui jumlah dari jalur *independent* melalui kalkulasi yang dinamakan kalkulasi *cyclomatic complexity*. Berikut merupakan hasil dari kalkulasi *cyclomatic complexity* pada *flow graph method onDataChange.*

- V(G) = jumlah *region* = 5
- V(G) = jumlah *edge* – jumlah *node* + 2 = 14 – 11 + 2 = 5
- V(G) = jumlah *predicate* + 1 = 4 + 1 = 5

2.3 *Independent Path*

- Jalur 1 : 1 - 2 - 11
- Jalur 2 : 1 - 2 - 3 - 4 - 10 - 2 - 11
- Jalur 3 : 1 - 2 - 3 - 4 - 5 - 9 - 10 - 2 – 11
- Jalur 4 : 1 - 2 - 3 - 4 - 5 - 6 - 8 - 9 - 10 - 2 - 11
- Jalur 5 : 1 - 2 - 3 - 4 - 5 - 6 - 7 - 8 - 5 - 9 - 10 - 2 – 11

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

Tabel 6.6 Hasil Pengujian Integrasi *Method onDataChange*

Jalur Uji	Prosedur Uji	*Expected Result*	*Result*	Status
1	Pada *method* onDataChange() mengambil data yang ada pada *firebase* PengajuanUsaha pada kelas *PengajuanUsaha-Model* pada *method get*()	Berhasil mengambil data dari method *getChildren*() dari kelas *Pengajuan-UsahaModel*	Berhasil mengambil data dari method *getChildren*() dari kelas *Pengajuan-UsahaModel*	Valid
2	Pada *method* onDataChange() mengambil data pada kelas *PengajuanUsaha-Model* pada *method get*() pada variabel investor	Berhasil mengambil data dari method *getInvestor* () dari kelas *Pengajuan-UsahaModel*	Berhasil mengambil data dari method *getInvestor* () dari kelas *Pengajuan-UsahaModel*	Valid
3	Pada *method* onDataChange() mengambil data pada kelas *PengajuanUsaha-Model* pada *method get*() pada variabel investor dan mendapatkan beberapa investor	Berhasil mengambil data dari method *getInvestor* () dari kelas *Pengajuan-UsahaModel* dengan idInvestor pada *method keySet*()	Berhasil mengambil data dari method *getInvestor* () dari kelas *Pengajuan-UsahaModel* dengan idInvestor pada *method keySet*()	Valid
4	Pada *method* onDataChange() mengambil data pada kelas *PengajuanUsaha-Model* pada *method get*() dengan masukkan idInvestor	Berhasil mengambil data status pada *method getStatus*() pada kelas *PengajuanUsaha-Model*	Berhasil mengambil data status pada *method getStatus*() pada kelas *PengajuanUsaha-Model*	Valid

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

Tabel 6.7 Hasil Pengujian Integrasi *Method onDataChange* (lanjutan)

Jalur Uji	Prosedur Uji	*Expected Result*	*Result*	Status
5	Pada *method* onDataChange() mengambil data pada kelas *PengajuanUsaha-Model* pada *method get*() yanga mana nilai *method getStatus*() = *false*	Berhasil mengakses *method set*() pada kelas *ApproveModel* serta bermhasil megambil data *method get*() dari kelas *PengajuanUsaha-Model*	Berhasil mengakses *method set*() pada kelas *ApproveModel* serta bermhasil megambil data *method get*() dari kelas *PengajuanUsaha-Model*	Valid

6.3 Pengujian Validasi

Pada pengujian validasi dimana akan dipaparkan pada setiap kebutuhan yang sudah didefinisakan. Dimana pengujian ini dilakukan untuk memastikan setiap skenario yang telah dibuat berjalan dengan baik. Pada pengujian validasi akan paparkan seluruh pengujian untuk kebutuhan fungsional metode *Blackbox.* Yang mana pengujian validasi pada penelitian ini akan dipaparkan pada sub bab 6.2.1 sampai 6.3.13.

6.2.1 Pengujian Validasi Pangajuan Sebagai Mitra Usaha

Pengujian validasi pengajuan sebagai mitra usaha merupakan pengujian yang dilakukan untuk menguji prosedur pada kebutuhan yang telah dipaparkan sebelumnya. Tabel 6.8 dan Tabel 6.9 menunjukkan pengujian validasi pengajuan sebagai mitra usaha.

Tabel 6.8 Pengujian Validasi Pengajuan Sebagai Mitra Usaha

Objek Uji	Kebutuhan Fungsional MK-1-001
Nama Kasus Uji	Pengajuan Sebagai Mitra Usaha
Prosedur Uji	1. Pengguna menekan tombol daftar 2. Sistem menambilkan form daftar sebagai mitra 3. Pengguna mengisi form yang telah tersedia 4. Sistem menyimpan data semetara dari pengguna 5. Pengguna melakukan *login* pada aplikasi 6. Pengguna mengisi form sebagai mitra aktif 7. Pengguna menekan tombol daftar pada form pengejuan untuk melakukan pengajuan

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

Tabel 6.9 Pengujian Validasi Pengajuan Sebagai Mitra Usaha (lanjutan)

Prosedur Uji	8. Sistem akan meberikan semua data tersebut kepada admin MaslahatKita untuk di verifikasi
Hasil Uji Yang Diharapkan	Data pengajuan sebagai mitra berhasil dikirim oleh mitra
Hasil Uji	Data pengajuan sebagai mitra berhasil dikirim oleh mitra
Status	VALID

6.2.2 Pengujian Validasi Verifikasi Pangajuan Sebagai Mitra Usaha

Pengujian validasi verifikasi pengajuan sebagai mitra usaha merupakan pengujian yang dilakukan untuk menguji prosedur pada kebutuhan yang telah dipaparkan sebelumnya. Tabel 6.10 menunjukkan pengujian validasi verifikasi pengajuan sebagai mitra usaha.

Tabel 6.10 Pengujian Validasi Verifikasi Pengajuan Sebagai Mitra Usaha

Objek Uji	Kebutuhan Fungsional MK-1-002
Nama Kasus Uji	Verifikasi Pengajuan Sebagai Mitra Usaha
Prosedur Uji	1. Admin akan menekan tombol verifikasi 2. Admin memeriksa data pengajuan sebagai mitra 3. Jika admin menyetujui pengajuan tersebut maka admin akan menekan tombol *approve* 4. Maka sistem akan mengirim *email* kepada mitra bahwa akun yang telah diajukan sudah terdaftar
Hasil Uji Yang Diharapkan	Data pengajuan sebagai mitra berhasil diverifikasi oleh admin
Hasil Uji	Data pengajuan sebagai mitra berhasil diverifikasi oleh admin
Status	VALID

6.2.3 Pengujian Validasi Pangajuan Sebagai Investor

Pengujian validasi pengajuan sebagai investor merupakan pengujian yang dilakukan untuk menguji prosedur pada kebutuhan yang telah dipaparkan sebelumnya. Tabel 6.11 menunjukkan pengujian validasi pengajuan sebagai investor.

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

Tabel 6.11 Pengujian Validasi Pengajuan Sebagai Investor

Objek Uji	Kebutuhan Fungsional MK-1-003
Nama Kasus Uji	Pengajuan Sebagai Investor
Prosedur Uji	1. Pengguna menekan tombol daftar 2. Sistem menambilkan form daftar sebagai investor 3. Pengguna mengisi form yang telah tersedia 4. Sistem menyimpan data semetara dari pengguna 5. Pengguna melakukan login pada aplikasi 6. Pengguna mengisi form sebagai investor aktif 7. Pengguna menekan tombol daftar pada form pengejuan untuk melakukan pengajuan 8. Sistem akan meberikan semua data tersebut kepada admin MaslahatKita untuk di verifikasi
Hasil Uji Yang Diharapkan	Data pengajuan sebagai mitra berhasil dikirim oleh investor
Hasil Uji	Data pengajuan sebagai mitra berhasil dikirim oleh investor
Status	VALID

6.2.4 Pengujian Validasi Verifikasi Pangajuan Sebagai Investor

Pengujian validasi verifikasi pengajuan sebagai investor merupakan pengujian yang dilakukan untuk menguji prosedur pada kebutuhan yang telah dipaparkan sebelumnya. Tabel 6.12 menunjukkan pengujian validasi verifikasi pengajuan sebagai investor.

Tabel 6.12 Pengujian Validasi Verifikasi Pangajuan Sebagai Investor

Objek Uji	Kebutuhan Fungsional MK-1-004
Nama Kasus Uji	Verifikasi Pengajuan Sebagai Investor
Prosedur Uji	1. Admin akan menekan tombol verifikasi 2. Admin memeriksa data pengajuan sebagai investor 3. Jika admin menyetujui pengajuan tersebut maka admin akan menekan tombol *approve* 4. Maka sistem akan membuat virtual *account* dan mengirim *email* kepada investor bahwa akun yang telah diajukan sudah terdaftar
Hasil Uji Yang Diharapkan	Data pengajuan sebagai investor berhasil diverifikasi oleh admin
Hasil Uji	Data pengajuan sebagai investor berhasil diverifikasi oleh admin
Status	VALID

6.2.5 Pengujian Validasi Pangajuan Pembiayaan Usaha

Pengujian validasi pangajuan pembiayaan usaha merupakan pengujian yang dilakukan untuk menguji prosedur pada kebutuhan yang telah dipaparkan

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

sebelumnya. Tabel 6.13 menunjukkan pengujian validasi pangajuan pembiayaan usaha.

Tabel 6.13 Pengujian Validasi Pangajuan Pembiayaan Usaha

Objek Uji	Kebutuhan Fungsional MK-1-005
Nama Kasus Uji	Pengajuan Pembiayaan Usaha
Prosedur Uji	1. Mitra mengisi form yang telah di sediakan 2. Kemudian mitra akan mengupload bisnis *plan* yang telah di buat. 3. Kemudian mitra menekan tombol daftar untuk mekalukan pengajuan pembiayaan
Hasil Uji Yang Diharapkan	Data pengajuan pembiayaan berhasil terdaftar
Hasil Uji	Data pengajuan pembiayaan berhasil terdaftar
Status	VALID

6.2.6 Pengujian Validasi Verifikasi Pangajuan Pembiayaan Usaha

Pengujian validasi verifikasi pangajuan pembiayaan usaha merupakan pengujian yang dilakukan untuk menguji prosedur pada kebutuhan yang telah dipaparkan sebelumnya. Tabel 6.14 menunjukkan pengujian validasi verifikasi pangajuan pembiayaan usaha.

Tabel 6.14 Pengujian Validasi Verifikasi Pangajuan Pembiayaan Usaha

Objek Uji	Kebutuhan Fungsional MK-1-006
Nama Kasus Uji	Verifikasi Pengajuan Sebagai Mitra Usaha
Prosedur Uji	1. Admin akan menekan tombol verifikasi pengajuan pembiayaan 2. Admin memilih pengajuan yang diinginkan 3. Admin memeriksa data pengajuan pemebiayaan yang di lakukan oleh mitra 4. Jika admin menyetujui pengajuan tersebut maka admin akan menekan tombol *approve* 5. Mitra usaha akan menerima pesan persejutuan pembiayaan dan juga data pembiayaan dimasukkan kedalam *market place* 6. Sistem akan menampilkan permintaan pembiayaan didalam *market place* dan juga
Hasil Uji Yang Diharapkan	Data pengajuan pembiayaan usaha berhasil divalidasi oleh admin
Hasil Uji	Data pengajuan pembiayaan usaha berhasil divalidasi oleh admin
Status	VALID

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

6.2.7 Pengajuan Investasi

Pengujian validasi pengajuan investasi merupakan pengujian yang dilakukan untuk menguji prosedur pada kebutuhan yang telah dipaparkan sebelumnya. Tabel 6.15 menunjukkan pengujian validasi pengajuan investasi.

Tabel 6.15 Pengujian Validasi Pengajuan Investasi

Objek Uji	Kebutuhan Fungsional MK-1-007
Nama Kasus Uji	Pengajuan Investasi
Prosedur Uji	1. Investor memilih usaha yang ada pada market place 2. Investor melihat ditail usaha, bisnis plan mitra usaha, dan monitoring usaha mitra 3. Jika investor tetarik pada usaha yang diajukan oleh mitra, maka investor dapat mengirim email kepada mitra 4. Investor dapat memasukkan dana pembiayaan yang sudah di tentukan pada setiap pengajuan pembiayaan yang telah dilakukan oleh mitra
Hasil Uji Yang Diharapkan	Data pengajuan investasi berhasil diajukan
Hasil Uji	Data pengajuan investasi berhasil diajukan
Status	VALID

6.2.8 Pengujian Verifikasi Pengajuan Investasi

Pengujian validasi verifikasi pengajuan investasi merupakan pengujian yang dilakukan untuk menguji prosedur pada kebutuhan yang telah dipaparkan sebelumnya. Tabel 6.16 dan Tabel 6.17 menunjukkan pengujian validasi verifikasi pengajuan investasi.

Tabel 6.16 Pengujian Validasi Pengajuan Investasi

Objek Uji	Kebutuhan Fungsional MK-1-008
Nama Kasus Uji	Verifikasi Pengajuan Investasi
Prosedur Uji	1. Admin akan menekan tombol verifikasi pengajuan investasi 2. Admin memilih pengajuan yang diinginkan 3. Admin memeriksa data pengajuan investasi yang di lakukan oleh investor 4. Jika admin menyetujui pengajuan investasi tersebut maka admin akan menekan tombol approve 5. Sistem mengirimkan pesan email konfrimasi kepada mitra 6. Sistem akan menyimpan data pengajuan untuk monitoring nantinya

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

Tabel 6.17 Pengujian Validasi Pengajuan Investasi (lanjutan)

Hasil Uji Yang Diharapkan	Data pengajuan investasi usaha berhasil verifikasi oleh admin
Hasil Uji	Data pengajuan investasi usaha berhasil verifikasi oleh admin
Status	VALID

6.2.9 Pengujian Validasi Pencairan Dana Usaha

Pengujian validasi pencairan dana usaha merupakan pengujian yang dilakukan untuk menguji prosedur pada kebutuhan yang telah dipaparkan sebelumnya. Tabel 6.18 menunjukkan pengujian validasi pencairan dana usaha.

Tabel 6.18 Pengujian Validasi Pencairan Dana Usaha

Objek Uji	Kebutuhan Fungsional MK-1-009
Nama Kasus Uji	Pencairan Dana Usaha
Prosedur Uji	1. Mitra menekan tombol pencairan dana usaha 2. Sistem akan memeriksa ketersedian dana 3. Jika dana tersedia makasistem akan memberikan konfrimasi kepada mitra dan juga investor akan menerima konfrimasi pencairan dana
Hasil Uji Yang Diharapkan	Mitra berhasil melakukan pencairan dana usaha
Hasil Uji	Mitra berhasil melakukan pencairan dana usaha
Status	VALID

6.2.10 Pengujian Validasi Pengangsuran

Pengujian validasi pengangsuran merupakan pengujian yang dilakukan untuk menguji prosedur pada kebutuhan yang telah dipaparkan sebelumnya. Tabel 6.19 menunjukkan pengujian validasi pengangsuran.

Tabel 6.19 Pengujian Validasi Pengangsuran

Objek Uji	Kebutuhan Fungsional MK-1-010
Nama Kasus Uji	Pengangsuran
Prosedur Uji	1. Mitra akan mentransfer melalui bank yang telah tersedia 2. Mitra mentransfer melalu virtual account yang telah disediakan
Hasil Uji Yang Diharapkan	Mitra berhasil melakukan pengangsuran
Hasil Uji	Mitra berhasil melakukan pengangsuran
Status	VALID

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

6.2.11 Pengujian Validasi Verifikasi Pengangsuran

Pengujian validasi verifikasi pengangsuran merupakan pengujian yang dilakukan untuk menguji prosedur pada kebutuhan yang telah dipaparkan sebelumnya. Tabel 6.20 menunjukkan pengujian validasi verifikasi pengangsuran.

Tabel 6.20 Pengujian Validasi Verifikasi Pengangsuran

Objek Uji	Kebutuhan Fungsional MK-1-011
Nama Kasus Uji	Verifikasi Pengangsuran
Prosedur	1. Admin menekan tombol verifikasi pengangangsuran 2. Admin memilih pengangsuran yang sedang di lakukan 3. Admin memeriksa pengangsuran yang dilakukan oleh mitra 4. Jika pengangsuran disetujui maka dana angsuran akan masuk dalam rekening maslahat 5. Dan bila ada margin maga sistem akan membaginya kepada investor dan persuhaan sesuai dengan peraturan yang telah di buat 6. Mitra akan menerima email konfrimasi pengangsuran yang telah di sepakati
Hasil Yang Diharapkan	Admin berhasil melakukan verifikasi pengangsuran
Hasil	Admin berhasil melakukan verifikasi pengangsuran
Status	VALID

6.2.12 Pengujian Validasi Pelaporan Pengangsuran

Pengujian validasi pelaporan pengangsuran merupakan pengujian yang dilakukan untuk menguji prosedur pada kebutuhan yang telah dipaparkan sebelumnya. Tabel 6.21 menunjukkan pengujian validasi pelaporan pengangsuran.

Tabel 6.21 Pengujian Validasi Pelaporan Pengangsuran

Objek Uji	Kebutuhan Fungsional MK-1-012
Nama Kasus Uji	Pelaporan Angsuran
Prosedur	1. Mitra menekan tombol pelaporan pengangsuran 2. Mitra menekan tombol penambahan pelaporan pengangsuran 3. Mitra mengisi form untuk melakukan pelaporan pengangsuran 4. Sistem mamasukkan data pelaporan kedalam data base
Hasil Yang Diharapkan	Mitra berhasil melakukan angsuran
Hasil	Mitra berhasil melakukan angsuran
Status	VALID

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

6.2.13 Pengujian Validasi Pencairan Dana Hasil Investasi

Pengujian validasi pencairan dana hasil investasi merupakan pengujian yang dilakukan untuk menguji prosedur pada kebutuhan yang telah dipaparkan sebelumnya. Tabel 6.22 menunjukkan pengujian validasi pencairan dana hasil investasi.

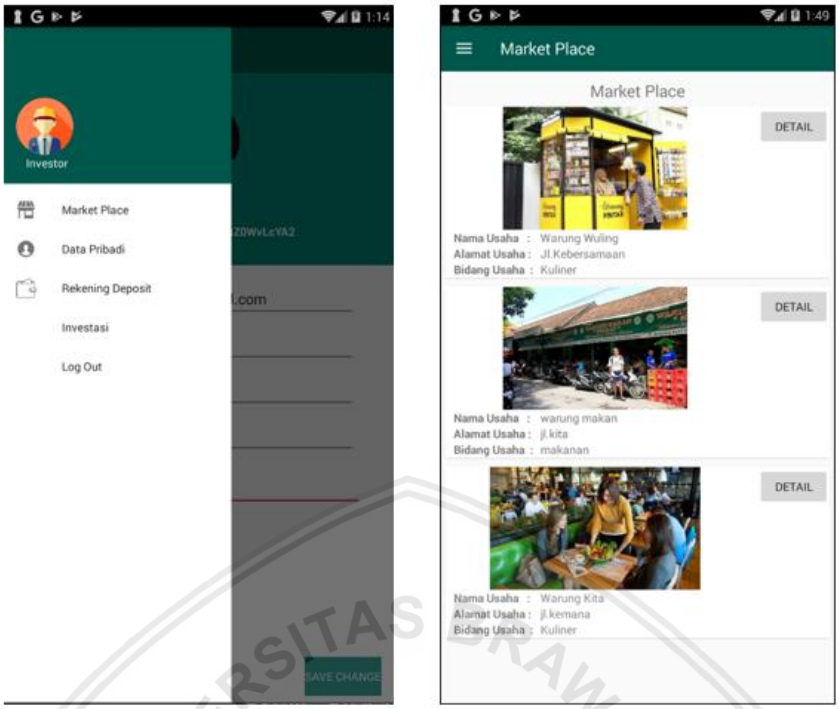
Tabel 6.22 Pengujian Validasi Pencairan Dana Hasil Investasi

Nama Kasus Uji	Pencairan Dana Hasil Investasi
Prosedur	1. Mitra menekan tombol pelaporan pengangsuran 2. Mitra menekan tombol penambahan pelaporan pengangsuran 3. Mitra mengisi form untuk melakukan pelaporan pengangsuran 4. Sistem mamasukkan data pelaporan kedalam data base
Hasil Yang Diharapkan	Mitra berhasil melakukan angsuran
Hasil	Mitra berhasil melakukan angsuran
Status	VALID

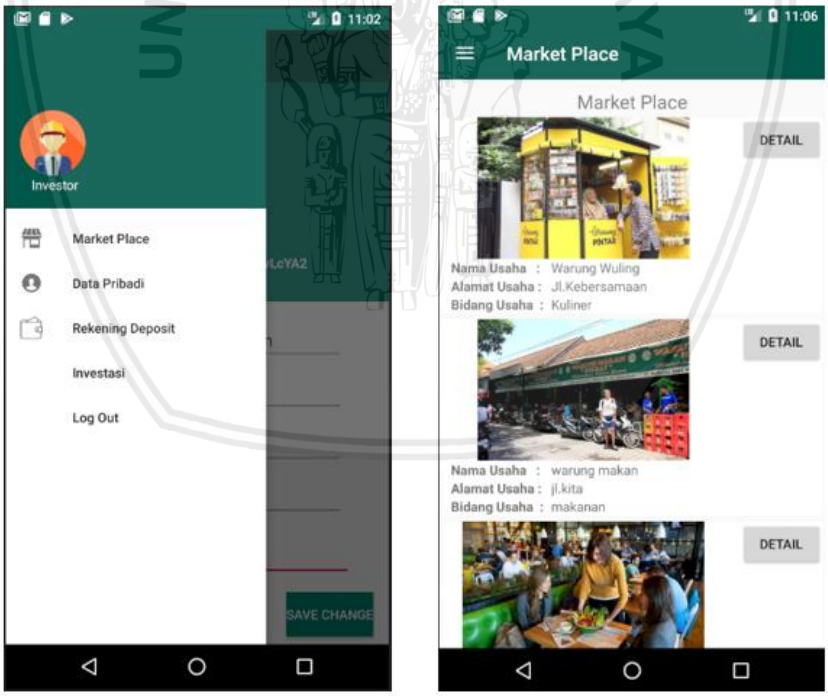
6.4 Pengujian *Compatibility*

Pada penelitian ini dilakukan pengujian *compatibility* dilakukan untuk mengetahui kompatibilitas dari sistem yang di bangun pada penelitian ini. Untuk memastikan bahwa kebutuhan non-fungsional system berjalan dengan baik. Pada pengujian *compatibility* akan dipaparkan untuk menguji sistem dengan berbagai versi sistem operasi android yaitu Android 4.4(*KitKat*), Android 7.0(*Nougat*), dan Android 8.0(*Oreo*). Dimana dapat dilihat tidak adanya perubahan tampilan mauapun fungsional aplikasi yang telah diujikan dapat dilihat pada Gambar 6.4 untuk Android Versi 4.4 (*Kitkat*), Gambar 6.5 untuk Android Versi 7.0 (*Nougat*), dan Gambar 6.6 Android Versi 8.0 (Oreo).

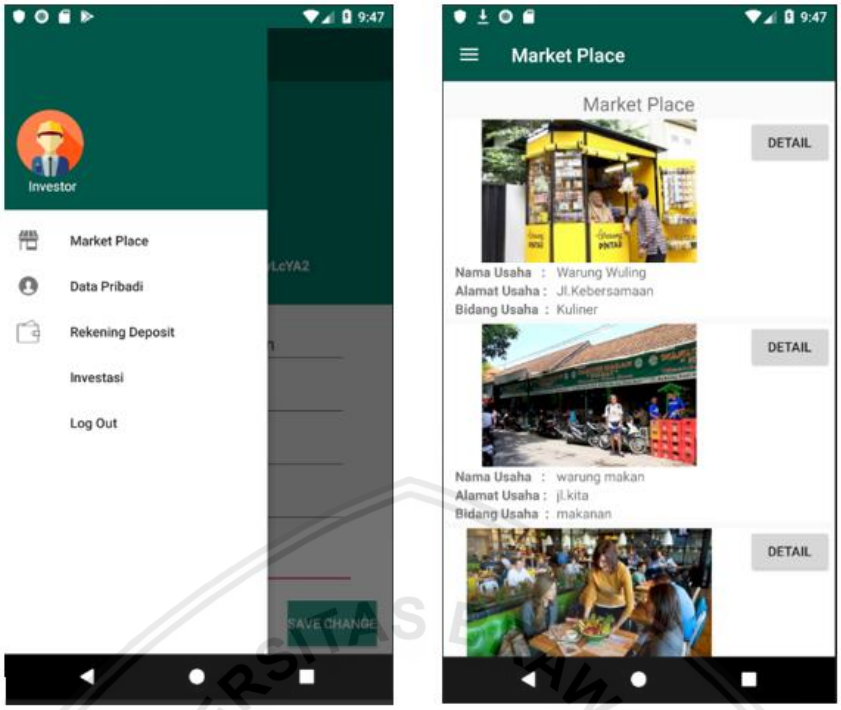
repository.ub.ac.id
UNIVERSITAS BRAWIJAYA



Gambar 6.4 Tampilan Aplikasi Pada Android Versi 4.4 (*Kitkat*)



Gambar 6.5 Tampilan Aplikasi Pada Android Versi 7.0 (*Nougat*)



Gambar 6.6 Tampilan Aplikasi Pada Android Versi 8.0 (*Oreo*)

Tabel 6.23 Hasil Pengujian *Compatibility*

Karakteristik	Android 4.4(*KitKat*)	Android 7.0(*Nougat*)	Android 8.0(*Oreo*)
Dapat menjalankan setiap fungsional dengan sebagai mana mestinya	Valid	Valid	Valid
Font, menu, tombol, dan gambar dapat terbaca dan terlihat dengan jelas	Valid	Valid	Valid
Penempatan setiap halaman tidak berubah dari posisinya	Valid	Valid	Valid

6.5 Analisis Hasil Pengujian

Pada hasil pengujian unit, pengujian integrasi, pengujian validasi, maupun pengujian *compatibility* yang telah dilakukan menghasilkan status valid pada semua kasus uji yang telah didefinisikan. Dimana pengujian unit dilakukan pada

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

tiga *method* yaitu *method onClick pada kelas PengajuanUsahaActivity, method onDataChange pada kelas VerfikasiInvestasiFragment, dan onClick pada kelas PelaporanUsahaActivity* yang menghasilkan 11 kasus uji. Yang mana pada 11 kasus uji tersebut terdiri dari dua untuk *method onClick pada kelas PengajuanUsaha-Activity*, lima untuk method onDataChange pada kelas VerfikasiInvestasiFragment, dan empat untuk *dan onClick pada kelas PelaporanUsahaActivity.* Dari 11 kasus uji tersebut menghasilkan status valid. Pada pengujian integrasi yang dilakukan untuk mengintegrasikan *method onDataChange pada kelas VerfikasiInvestasiFragment* dengan *method set* pada kelas *InvestorApproval* serta *method get* pada kelas *PengajuanUsahaModel* didapatkan satu kasus uji. Pada kasus uji tersebut dihasilkan status valid. Pada pengujian validasi menghasilkan sebanyak 13 kasus uji untuk kebutuhan fungsional. Dari pengujian validasi yang telah dilakukan menghasilkan status valid pada keseluruhan kasus uji. Untuk pengujian *compatibility* dilakuakn pada berbagai versi android diantaranya android 4.4(*KitKat*), android 7.0(*Nougat*), dan android 8.0(*Oreo*). Dengan hasil karakteristik pada setiap versi valid dan tidak terdapat perbuahan dan setiap fungsional yang didapat berjalan dengan sebagai mana mestinya.

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

BAB 7 KESIMPULAN DAN SARAN

Pada bab ini akan membahas tentang kesimpulan dari penelitan yang telah dilakukan. Dan juga terdapat saran untuk pengembangan penelitian yang telah dilakukan untuk kemudian hari.

7.1 Kesimpulan

Berdasarkan pada hasil dari analisis kebutuhan, perancangan, implementasi, dan pengujian yang telah dilakukan. Maka dapat disimpulkan sebagai berikut:

1. Pada analisi kebutuhan dapat disimpulkan bahwa proses bisnis yang didapatkan pada PT Maslahat Amanah Umat yang di dapat dapat di implementasikan pada perangkat *mobile.* Dan dijabarkan dengan adanya beberapa kebutuhan fungsional maupun non-fungsional yang telah dipaparkan pada penelitian ini.
2. Sesui dengan analisis kebutuhan maka didapatkan adanya perancangan sistem yang terbentuk dimana melibatkan tiga fungsionalitas utama yaitu pengajuan usaha, verifikasi investasi, dan pelaporan usaha. Dimana terdapat beberapa fitur diantarnya untuk menambah dokumen serta foto sebagai data pendukung.
3. Berdasarkan pada hasil pengujian validasi secara fungsionalitas didapatkan semua fungsi berjalan sesuai yang diharapkan. Dimana validasi pada tiga fitur utama yang telah dibuat dilakuakn pengujian secara *white box* maupun *black box* telah valid dan memenuhi perancangan aplikasi.
4. Secara aspek *compatibility* telah dilakukan pengujian dapat disimpulkan aplikasi dapat berjalan dengan baik pada sistem operasiAndroid 4.4(*KitKat*), sistem operasi Android 7.0(*Nougat*), dan sistem operrasi Android 8.0(*Oreo*). Dan tidak terdapat masalh akan perubahan tampilan atau fungsionalitas tidak dapat bekerja pada setiap sistem operasi yang dilakukan pengujian.

7.2 Saran

Nantinya aplikasi investasi maslatalah kita pada sistem Android dapat dikembangkan lebih lanjut kedepannya dengan saran sebagai berikut:

1. Kedepanya aplikasi maslahat kita ini dapat digunakan dengan sistem penyimpanan data lebih memadai lagi. Agar jalannya transaksi pada aplikasi ini dapat berjalan dengan cepat.
2. Nantinya juga dapat ditambah beberapa fitur-fitur pendukung untuk dapat melihat grafik progress usaha lebih detail dan lebih mudah untuk melakukan monitoring dari sisi investor, admin, maupun mitra usaha.

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

DAFTAR REFRENSI

Aditya, Imanuel., 2017. *Analisis Swot Implementasi Teknologi Finansial Terhadap Kualitas Layanan Perbankan Di Indonesia*. Universitas Pelita Harapan Tangerang: Vol. 20 No. 1

Adi, Taufik Nur., 2015. Generator Kode Unit Testing Untuk Javascript Berbasis Framework QUnit. Jurnal Rekayasa Sistem & Industri Vol.2, No.3

Alfian, Irfan., 2015.Analisis Kualitas Sistem Aplikasi M-Library Di Perpustakaan Universitas Gadjahmada. Surabaya : Perpustakaan Universitas Airlangga

Hendini, Ade. 2016. Pemodelan UML Sistem Informasi Monitoring Penjualan Dan Stok Barang (Studi Kasus : Distro Zhezha Pontianak). Pontianak : BSI Pontianak.

Heryanto, Deni., 2018. *Business Process Model And Notation (BPMN) Donat Madu Cihanjuang*. Garut: Sekolah Tinggi Teknik Garut

Gumiwang, Rinkang., 2018. Fintech Lending, Pinjam Meminjam Online yang Makin Menjamur. Tersedia di https://tirto.id/fintech-lending-pinjam-meminjam-online-yang-makin-menjamur-cDpo [Diakses 27 Agustus 2018]

Ilhami, Mirza,. 2017. *Pengenalan Google Firebase Untuk Hybrid Mobile Apps Berbasis Cordova*. STMIK Mikroskil: Vol. 3 No. 1

J. Myers, Glenford., 2004. *The Art of Software Testing 2nd Edition*. New Jersey : John Wiley & Sons, Inc.

Liana, Linda,. 2015. *PENGUJIAN PERANGKAT LUNAK*. Universitas Mercu Buana Jakarta: Fakultas Ilmu Komputer

Mustaqbal, M.Sidi dkk., 2015. Pengujian Aplikasi Menggunakan Black Box Testing Boundary Value Analysis (Studi Kasus : Aplikasi Prediksi Kelulusan SNMPTN). Bandung : Jurnal Ilmiah Teknologi Informasi Terapan.

Muzdalifa, Irma dkk., 2018. *Peran Fintech Dalam Meningkatkan Keuangan Inklusif Pada Umkm Di Indonesia (Pendekatan Keuangan Syariah)*. Masharaf Al-Syariah: Vol.3 No.1

Nerwork,M, D. 2019. Belajar UML – Use Case Diagram. Tersedia di https://msdn.microsoft.com/en-us/lirbrary/dd409445.aspx [Diakses 20 Februari 2019].

Prabowo, Heri dkk., 2014. Sistem Informasi Panduan Trayek Angkutan Umum Berbasis Mobile Smarthphone Pada Dinas Perhubungan Jakarta. Jakarta Pusat : Jurnal Pilar Nusa Mandiri.

Pratama, Aditya Rahmatullah. 2019. Belajar UML – Use Case Diagram. Tersedia di https://www.codepolitan.com/mengenal-uml-diagram-use-case#[Diakses 26 Februari 2019].

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

Pressman, R. S., 2010. *Softeware Engineering A Practitioner's Approach* (5th ed.). New York: McGraw-hill.

Pressman, R. S., 2015. *Softeware Engineering A Practitioner's Approach, Eighth Edition*. New York: McGraw-hill.

Sa'adah, Karimatus dkk., 2017. Aplikasi Panduan Sosialisasi Keselamatan Lalu Lintas "Road Safety" Menggunakan Phonegap Dengan Android. Semarang : Prosiding SINTAK

S, Rosa A dan Shalahuddin, M. 2015. Rekayasa Perangkay Lunak. Bandung : Informatika.

Sari, Siska Komala dan Asniar., 2015. *Analisis Dan Pemodelan Proses Bisnis Prosedur Pelaksanaan Proyek Akhir Sebagai Alat Bantu Identifikasi Kebutuhan Sistem*. Universitas Telkom: Fakultas Ilmu Terapan.

Setiay, hendry., 2013. Sistem Informasi Pemesanan Dan Penjualan Berbasis Web Pada Dewi Florist. Palembang: STMIK GI MDP.

Sommerville. (2011). *Software Engineering* (9th ed.). Jakarta: Erlangga.

Weske, M., 2007. Business Process Management Concepts Languages, Architectures. New York: Springer.

2018. *CSR Management & Development*.PT Maslahat Amanah Umat.

2018. Menggunakan Firebase, SDK terpadu dan lintas platform Google, untuk membuat aplikasi yang lebih baik. Tersedia di https://developer.android.com/distribute/best-practices/develop/build-with-firebase?hl=id [Diakses 31 Agustus 2018]

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA