



PENGEMBANGAN SISTEM INFORMASI MZ DARUL KHALIK MENGUNAKAN FLUTTER

Disusun dan Diajukan untuk Memenuhi Persyaratan Ujian Akhir Memperoleh
Gelar Sarjana Komputer pada Fakultas Sains dan Teknologi
Universitas Pembangunan Panca Budi
Medan

SKRIPSI

OLEH:

NAMA : RUDI SALDONI KURNIAWAN
NPM : 1814370334
PROGRAM STUDI : SISTEM KOMPUTER

FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS PEMBANGUNAN PANCA BUDI
MEDAN
2022

PENGESAHAN SKRIPSI

JUDUL : PENGEMBANGAN SISTEM INFORMASI MZ DARUL KHALIK
MENGUNAKAN FLUTTER

NAMA : RUDI SALDONI KURNIAWAN
N.P.M : 1814370334
FAKULTAS : SAINS & TEKNOLOGI
PROGRAM STUDI : Sistem Komputer
TANGGAL KELULUSAN : 19 Februari 2022

DIKETAHUI

DEKAN

KETUA PROGRAM STUDI



Hamdani, ST., MT.



Eko Hariyanto, S.Kom., M.Kom

**DISETUJUI
KOMISI PEMBIMBING**

PEMBIMBING I

PEMBIMBING II



Dr Muhammad Iqbal, S.Kom., M.Kom.



Muhammad Muttaqin, S.Kom., M.Kom

Hal : Permohonan Meja Hijau

Medan, 04 Januari 2022
Kepada Yth : Bapak/Ibu Dekan
Fakultas SAINS & TEKNOLOGI
UNPAB Medan
Di -
Tempat

Dengan hormat, saya yang bertanda tangan di bawah ini :

Nama : RUDI SALDONI KURNIAWAN
Tempat/Tgl. Lahir : RANTAU KASAI /
Nama Orang Tua : SUFIRMAN
N. P. M : 1814370334
Fakultas : SAINS & TEKNOLOGI
Program Studi : Sistem Komputer
No. HP : 085275391671
Alamat :

Datang bermohon kepada Bapak/Ibu untuk dapat diterima mengikuti Ujian Meja Hijau dengan judul **PENGEMBANGAN SISTEM INFORMASI MZ DARUL KHALIK MENGGUNAKAN FLUTTER**, Selanjutnya saya menyatakan :

1. Melampirkan KKM yang telah disahkan oleh Ka. Prodi dan Dekan
2. Tidak akan menuntun ujian perbaikan nilai mata kuliah untuk perbaikan indeks prestasi (IP), dan mohon diterbitkan ijazahnya setelah lulus ujian meja hijau.
3. Telah tercap keterangan bebas pustaka
4. Terlampir surat keterangan bebas laboratorium
5. Terlampir pas photo untuk ijazah ukuran 4x6 = 5 lembar dan 3x4 = 5 lembar Hitam Putih
6. Terlampir foto copy STTB SLTA dilegalisir 1 (satu) lembar dan bagi mahasiswa yang lanjutan D3 ke S1 lampirkan ijazah dan transkripnya sebanyak 1 lembar.
7. Terlampir pelunasan kwintasi pembayaran uang kuliah berjalan dan wisuda sebanyak 1 lembar
8. Skripsi sudah dijilid lux 2 exemplar (1 untuk perpustakaan, 1 untuk mahasiswa) dan jilid kertas jeruk 5 exemplar untuk penguji (bentuk dan warna penjilidan diserahkan berdasarkan ketentuan fakultas yang berlaku) dan lembar persetujuan sudah di tandatangi dosen pembimbing, prodi dan dekan
9. Soft Copy Skripsi disimpan di CD sebanyak 2 disc (Sesuai dengan Judul Skripsinya)
10. Terlampir surat keterangan BKKOL (pada saat pengambilan ijazah)
11. Setelah menyelesaikan persyaratan point-point diatas berkas di masukan kedalam MAP
12. Bersedia melunaskan biaya-biaya uang dibebankan untuk memproses pelaksanaan ujian dimaksud, dengan perincian sbb :

1. [102] Ujian Meja Hijau	: Rp.	1,000,000
2. [170] Administrasi Wisuda	: Rp.	1,750,000
Total Biaya	: Rp.	2,750,000

Ukuran Toga :

Diketahui/Disetujui oleh :

Hormat saya



Hamdani, ST., MT.
Dekan Fakultas SAINS & TEKNOLOGI



RUDI SALDONI KURNIAWAN
1814370334

Catatan :

- 1. Surat permohonan ini sah dan berlaku bila ;
 - a. Telah dicap Bukti Pelunasan dari UPT Perpustakaan UNPAB Medan.
 - b. Melampirkan Bukti Pembayaran Uang Kuliah aktif semester berjalan
- 2. Dibuat Rangkap 3 (tiga), untuk - Fakultas - untuk BPAA (asli) - Mhs.ybs.

SURAT PERNYATAAN

Saya yang bertanda tangan dibawah ini:

Nama : Rudi Saldoni Kurniawan

NPM : 1814370334

Prodi : Sistem Komputer

Judul Skripsi : Pengembangan Sistem Informasi MZ Darul Khaliq Menggunakan Flutter

Dengan ini saya menyatakan bahwa :

1. Skripsi ini merupakan hasil karya tulis saya sendiri dan bukan merupakan hasil karya tulis orang lain.
2. Memberi izin dan hak bebas Royalti Non-Eksklusif kepada UNPAB untuk menyimpan, mengalih-media/formatkan pengelola mendistribusikan, dan mempublikasikan karya skripsinya melalui internet dan media lain bagi kepentingan akademik.

Pernyataan ini saya perbuat dengan penuh tanggung jawab dan saya bersedia menerima konsekuensi apapun sesuai dengan aturan yang berlaku apabila dikemudian hari diketahui apabila pernyataan ini tidak benar.

Medan, 3 Maret 2022





Rudi Saldoni Kurniawan

NPM : 1814370334

SURAT PERNYATAAN

Dengan ini saya menyatakan bahwa dalam pembuatan skripsi ini tidak terdapat karya yang diajukan untuk memperoleh gelar kesarjanaan didalam perguruan tinggi, dan sepanjang pengetahuan saya juga tidak terdapat karya atau pendapat yang pernah ditulis atau diterbitkan oleh orang lain, kecuali yang secara tertulis didalam skripsi ini dan disebutkan dalam daftar Pustaka.

Medan, 3 Maret 2022



Rudi Saldoni Kurniawan

NPM : 1814370334



UNIVERSITAS PEMBANGUNAN PANCA BUDI FAKULTAS SAINS & TEKNOLOGI

Jl. Jend. Gatot Subroto Km 4,5 Medan Fax. 061-8458077 PO.BOX : 1099 MEDAN

PROGRAM STUDI TEKNIK ELEKTRO	(TERAKREDITASI)
PROGRAM STUDI ARSITEKTUR	(TERAKREDITASI)
PROGRAM STUDI SISTEM KOMPUTER	(TERAKREDITASI)
PROGRAM STUDI TEKNIK KOMPUTER	(TERAKREDITASI)
PROGRAM STUDI AGROTEKNOLOGI	(TERAKREDITASI)
PROGRAM STUDI PETERNAKAN	(TERAKREDITASI)
PROGRAM STUDI TEKNOLOGI INFORMASI	(TERAKREDITASI)

PERMOHONAN JUDUL TESIS / SKRIPSI / TUGAS AKHIR*

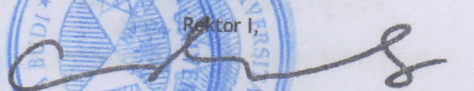
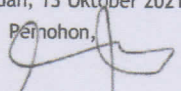
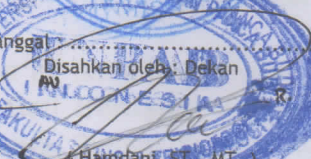
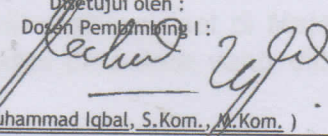

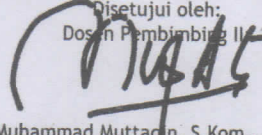
Saya yang bertanda tangan di bawah ini :

Nama Lengkap : RUDI SALDONI KURNIAWAN
 Tempat/Tgl. Lahir : RANTAU KASAI / 22 Februari 1998
 Nomor Pokok Mahasiswa : 1814370334
 Program Studi : Sistem Komputer
 Konsentrasi : Rekayasa Perangkat Lunak
 Jumlah Kredit yang telah dicapai : 125 SKS, IPK 3.62
 Nomor Hp : 085275391671
 Dengan ini mengajukan judul sesuai bidang ilmu sebagai berikut :

No.	Judul
1.	PENGEMBANGAN SISTEM INFORMASI MZ DARUL KHALIK MENGGUNAKAN FLUTTER

Catatan : Diisi Oleh Dosen Jika Ada Perubahan Judul

*Coret Yang Tidak Perlu

 Rektor I. (Cahyo Pramono, S.E., M.M.)	Medan, 13 Oktober 2021 Pemohon,  (Rudi Saldoni Kurniawan)	
 Tanggal : Disahkan oleh: Dekan (Haridanti, S.T., M.T.)	Tanggal : 14 Oktober 2021 Disetujui oleh : Dosen Pembimbing I :  (Dr. Muhammad Iqbal, S.Kom., M.Kom.)	
Tanggal : Disetujui oleh: Ka. Prodi Sistem Komputer  (Eko Hariyanto, S.Kom., M.Kom.)	Tanggal : 15 Oktober 2021 Disetujui oleh: Dosen Pembimbing II :  (Muhammad Muttakin, S.Kom., M.Kom.)	
No. Dokumen: FM-UPBM-18-02	Revisi: 0	Tgl. Eff: 22 Oktober 2018

Sumber dokumen: <http://mahasiswa.pancabudi.ac.id>

Dicetak pada: Rabu, 13 Oktober 2021 11:54:45

ABSTRAK

RUDI SALDONI KURNIAWAN
PENGEMBANGAN SISTEM INFORMASI MZ DARUL KHALIK
MENGGUNAKAN FLUTTER
2021

Informasi data jamaah dan jadwal kegiatan merupakan salah satu data informasi yang harus diatur dan dicatat agar kegiatan berlangsung dengan baik. Permasalahan yang sering terjadi adalah tidak tersampaikan informasi tersebut kepada jamaah dalam menunjang kegiatan pada Majelis Zikir Darul khalik tersebut sehingga target yang ditentukan tidak tersampaikan akibat kekurangan data informasi. Permasalahan ini juga dihadapi oleh Majelis Zikir Darul khalik. Lembaga ini yang melakukan Pengembangan pendidikan pesantren modern dan pengajaran secara modern, Mengembangkan ajaran agama Islam berdasarkan Al Quran, Al Hadits dan Tasawuf Islam. Badan Kordinator Majelis Zikir Darul khalik melakukan sistem pendataan secara manual dan sederhana, Majelis Zikir Darul khalik juga membutuhkan informasi dalam melakukan kerjanya. Hasil rancangan sistem informasi dapat mendata kesediaan informasi, sehingga data jamaah dan jadwal kelas lembaga dapat terlaksana dengan baik. Informasi jamaah dan informasi jadwal kelas dapat tersampaikan dan dimanfaatkan dengan baik dan kegiatan dapat diikuti oleh semua jamaah.

KATA KUNCI: Sistem informasi, Flutter, Android

KATA PENGANTAR

Puji syukur ke hadirat Allah SWT, karena dengan berkat dan kasih anugerah-Nya penulis masih diberikan kesempatan untuk menyelesaikan skripsi ini sebagaimana mestinya. Skripsi ini berjudul ” **Sistem Informasi Mz Darul Khalik Menggunakan Flutter**”. Dalam kesempatan ini, penulis mengucapkan rasa terima kasih yang tak terhingga kepada pihak-pihak yang telah membantu dalam penyelesaian skripsi ini. Penulis ingin mengucapkan terima kasih kepada :
Orang tua saya yang selalu memberikan semangat, dukungan dan motivasi dalam penyusunan skripsi ini:

1. Bapak Dr. H. Muhammad Isa Indrawan, S.E., M.M., selaku Rektor Universitas Pembangunan Panca Budi Medan.
2. Bapak Hamdani, S.T., M.T., selaku Dekan Fakultas Sains dan Teknologi Universitas Pembangunan Panca Budi Medan.
3. Bapak Eko Harianto, S. Kom., M. Kom. selaku Ketua Program Studi Teknik Sistem Komputer Universitas Pembangunan Panca Budi Medan.
4. Bapak DR. Muhammad Iqbal, S.Kom., M. Kom. selaku Dosen Pembimbing I yang telah memberikan arahan dan membimbing dalam penyelesaian skripsi ini.
5. Bapak Muhammad Muttaqin, S. Kom., M. Kom. selaku Dosen Pembimbing II yang telah memberikan ilmu pengetahuan, serta bimbingan dalam penyelesaian skripsi ini.
6. Dosen-dosen pada Program Studi Sistem Komputer Fakultas Sains dan Teknologi Universitas Pembangunan Panca Budi Medan.
7. Staff dan karyawan Universitas Pembangunan Panca Budi Medan.
8. Rekan-rekan dari program studi Sistem Komputer, Fakultas Sains dan Teknologi, Universitas Pembangunan Panca Budi, Medan.

Penulis juga menyadari bahwa penyusunan skripsi ini belum sempurna baik dalam penulisan maupun isi disebabkan keterbatasan kemampuan penulis. Oleh karena itu, penulis mengharapkan kritik dan saran yang sifatnya membangun dari pembaca untuk kesempurnaan isi skripsi ini.

Medan, 08 Desember 2021
Penulis

Rudi Saldoni Kurniawan
1814370334

DAFTAR ISI

KATA PENGANTAR	i
DAFTAR ISI	ii
DAFTAR GAMBAR	v
DAFTAR TABEL	vi
BAB I PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Identifikasi Masalah	3
1.3 Rumusan Masalah	3
1.4 Batasan Masalah.....	4
1.5 Tujuan Penelitian	4
1.6 Manfaat Penelitian	4
BAB II LANDASAN TEORI	6
2.1 <i>Mobile Aplication Framework</i>	6
2.2 Flutter	7
2.2.1 Dart	8
2.2.2 Widget	8
2.2.3 <i>Architecture Flutter</i>	9
2.2.4 <i>Rest Fire</i>	10
2.2.5 <i>JSON (JavaScript Object Notation)</i>	13
2.3 React Native	13
2.3.1 <i>JSX</i>	14
2.3.2 <i>Component</i>	14
2.3.3 <i>Arsitektur React Native</i>	15
2.4 <i>State Management</i>	16
2.4.1 <i>Flutter BloC</i>	16
2.5 Domain.....	18
2.6 Sistem Informasi	21

2.7	Konsep Informasi	23
2.7.1	Siklus Informasi.....	23
2.7.2	Kualitas Informasi	24
2.7.3	Konsep Dasar Sistem Informasi	24
2.8	Majelis Zikir Abdul Khalik.....	25
2.8.1	Sejarah MZ Darul khalik	26
2.9	<i>Unified Modelling Language</i>	27
2.9.1	<i>Use Case Diagram</i>	28
2.10	Aplikasi	31
2.11	Database	32
2.12	Structured Query Language	35
2.12.1	Sejarah SQL.....	35
2.12.2	Standardisasi.....	36
2.13	Visual Studio Code	36
2.13.1	Sejarah Visual Studio Code.....	38
2.13.2	Fitur Visual Studio Code	38
2.14	Riset Terkait	39

BAB III METODE PENELITIAN 41

3.1	Tahapan Penelitian	41
3.2	Analisis Kebutuhan Perangkat Lunak.....	43
3.3	Analisis Kebutuhan Perangkat Keras.....	43
3.3.1	<i>Use Case Diagram</i>	44
3.3.2	<i>Activity Diagram</i>	45
3.3.3	<i>Sequence Diagram</i>	47
3.3.4	<i>Flowchart</i>	47
3.4	Perancangan Antarmuka Program.....	48
3.4.1	Rancangan Menu Dashboard.....	49
3.4.2	Rancangan Menu Jadwal Kelas.....	50
3.4.3	Rancangan Menu Informasi	50
3.4.4	Rancangan Menu <i>Registrasi</i> dan <i>Login</i>	51

3.4.5	Rancangan Menu Kontak Kami	52
BAB IV HASIL DAN PEMBAHASAN.....		53
4.1	Spesifikasi Sistem	53
4.1.1	Spesifikasi Perangkat Keras	54
4.1.2	Spesifikasi Perangkat Lunak	54
4.2	Hasil Aplikasi Android	55
4.2.1	Halaman Menu Dashboard	55
4.2.2	Halaman Menu Galeri	56
4.2.3	Halaman Menu Login.....	57
4.2.4	Halaman Menu Kontak Kami.....	57
4.2.5	Halaman Menu Inventaris	58
BAB V PENUTUP.....		60
5.1	Kesimpulan	60
5.2	Saran.....	60
DAFTAR PUSTAKA		1

DAFTAR TABEL

Tabel 2.1 Simbol Use Case Diagram	29
Tabel 2.8 Riset terkait	39
Tabel 4.1 Spesifikasi Perangkat Keras.....	54
Tabel 4.2 Spesifikasi Perangkat Lunak.....	54

DAFTAR GAMBAR

Gambar 2.1 Struktur MZ Darul khalik.....	26
Gambar 2.2 Hierarki Data.....	34
Gambar 2.3 Microsoft Visual Studio Code.....	37
Gambar 3.1 Tahapan Penelitian	41
Gambar 3.2 <i>Use case diagram</i> pencatatan jemaah	44
Gambar 3.3 <i>Activity diagram</i> pencatatan Jemaah.....	46
Gambar 3.4 <i>Sequence diagram</i> pencatatan inventaris	47
Gambar 3.5 <i>Flowchart</i> pencatatan inventaris	48
Gambar 3.6 Rancangan menu dashboard.....	49
Gambar 3.7 Rancangan menu Jadwal Kelas	50
Gambar 3.8 Rancangan Menu Informasi	51
Gambar 3.9 Rancangan menu registrasi dan <i>login</i>	52
Gambar 3.10 Rancangan menu kontak kami	52
Gambar 4.1 Halaman menu beranda.....	55
Gambar 4.2 Halaman menu Jadwal kelas	56
Gambar 4.3 Halaman Menu Login.....	57
Gambar 4.4 Halaman menu kontak kami.....	58
Gambar 4.5 Halaman menu informasi	59

BAB I

PENDAHULUAN

1.1 Latar Belakang

Kemajuan teknologi memberikan banyak manfaat dan dampak positif dalam penggunaannya, (Fallo, 2017). Teknologi informasi adalah jenis teknologi yang melakukan fungsi seperti pemrosesan data, akuisisi data, organisasi data, penyimpanan data, dan modifikasi data dalam berbagai metode untuk menghasilkan informasi yang berharga atau berkualitas tinggi. Selain itu, teknologi informasi berfungsi untuk memecahkan masalah, merangsang inovasi, serta meningkatkan efektivitas dan efisiensi operasional penggunaannya.

Badan Kordinator Majelis Zikir Darul khalik di Jalan Tj Selamat kec. Sunggal Kabupaten Deli Serdang Sumut 20351, adalah Majelis dibawah Yayasan abdul khalik fajduani yang melakukan Pengembangan pendidikan pesantren modern dan pengajaran secara modern, Mengembangkan ajaran agama Islam berdasarkan Al Quran, Al Hadits dan Tasawuf Islam. Badan Kordinator Majelis Zikir Darul khalik melakukan sistem pendataan secara manual dan sederhana, jika jamaah tersebut lupa membawa salah satu dokumen tersebut, sehingga memerlukan waktu dalam proses pendataan. Sistem kerja tersebut dirasa masih kurang efektif dan efisien, karena dalam pencatatan manual bisa saja menyebabkan beberapa kesalahan dan suatu masalah. Serta membutuhkan banyak waktu untuk suatu proses pendataan dan pembuatan laporan karena harus mencari dan membuka kembali formulir pendataan terlebih dahulu.

Banyak jamaah yang tidak tau perkembangan kegiatan yang dilaksanakan badan kordinator Majelis Zikir Darul khalik karena jarak tempuh dari daerah yang berbeda. Penyediaan informasi untuk pendataan Jemaah Majelis Zikir Darul khalik dengan memanfaatkan kemajuan teknologi agar informasi lebih mudah diakses oleh jemaah Majelis Zikir. Memberikan informasi yang tepat waktu, akurat, dan benar dapat membantu dalam menyediakan konferensi dengan layanan terbaik.

Untuk mendapatkan informasi yang akurat, murah, tepat waktu dan efektif, maka dibutuhkan sebuah aplikasi yang dapat diakses oleh user secara mobile kapan dan dimana saja, serta dapat berjalan dengan koneksi internet, lebih cepat, akses informasi lebih mudah. Seiring dengan perkembangan teknologi, aplikasi mobile dapat digunakan untuk mengumpulkan data secara realtime menggunakan smartphone, karena smartphone sudah dilengkapi dengan sensor Wifi untuk koneksi internet, kamera untuk merekam foto, dan sistem GPS untuk mendapatkan koordinat alamat lengkap user (Rizzo, F., 2020).

Flutter adalah pilihan yang tepat bagi pengembang aplikasi cross-platform mobile application (iOS dan Android), karena menyediakan banyak layanan, efisien dari segi waktu dan biaya, lebih mudah, dan lebih produktif. Disamping memiliki User Interface yang lebih menarik, flutter juga dapat digunakan untuk merancang aplikasi mobile secara original. Flutter dikembangkan oleh Google menggunakan bahasa pemrograman Dart. Dart merupakan bahasa pemrograman terstruktur (perograman berorientasi objek / OOP) dan open source. Alasan menggunakan flutter adalah karena flutter sudah menghilangkan bridgedan OEM platform untuk dapat berkomunikasi langsung

dengan CPU dan GPU. (Rizzo, F., 2020). Tujuannya adalah memudahkan pengembang melakukan perubahan aplikasi sesering mungkin, memudahkan penggunaan platforms API melakukan sinkronisasi sistem pesan. Sistem informasi merupakan suatu aplikasi yang dapat dibuka secara umum oleh admin yang bertanggung jawab terhadap kontrol kualitas dari bagian admin (Pratama, E. B., & Hendini, 2019). Dengan menggunakan aplikasi ini, admin dapat memantau dan berkordinasi langsung Jemaah Majelis Zikir Darul khalik sehingga kegiatan pekerjaan tidak terhambat. Berdasarkan latar belakang yang sudah dijelaskan, maka dengan ini penulis mengambil judul: **PENGEMBANGAN SISTEM INFORMASI MZ DARUL KHALIK MENGGUNAKAN FLUTTER.**

1.2 Identifikasi Masalah

Berdasarkan latar belakang masalah yang telah diuraikan di atas, maka dapat disimpulkan bahwa identifikasi masalah adalah:

1. Sulitnya melakukan pendataan jemaah dikarenakan jarak tempuh dari daerah yang berbeda-beda.
2. Jemaah yang sering Ketinggalan Informasi yang di laksanakan Kordinator Majelis zikir.
3. Sistem informasi merupakan suatu aplikasi yang tepat untuk mendukung program dan kordinasi di majelis zikir darul khalik.

1.3 Rumusan Masalah

Adapun rumusan masalah yang digunakan dalam penelitian adalah sebagai berikut:

1. Bagaimana merancang Sistem informasi merupakan suatu aplikasi yang tepat untuk mendukung program dan kordinasi di majelis zikir darul khalik?
2. Bagaimana cara merancang Framework Flutter dengan metode yang tepat sebagai solusi pendukung system informasi berbasis android?

1.4 Batasan Masalah

Adapun batasan masalah yang digunakan dalam penelitian ini adalah sebagai berikut:

- 1 Jemaah majelis zikir darul khalik supaya terbantu dengan Informasi yang di laksanakan Kordinator Majelis zikir.
- 2 Sistem informasi di Majelis zikir darul khalik.
- 3 Merancang Framework Flutter dengan pendukung system informasi berbasis android.

1.5 Tujuan Penelitian

Adapun tujuan penelitian yang digunakan dalam penelitian ini adalah sebagai berikut:

1. Merancang dan membuat Sistem informasi untuk mendukung program dan kordinasi di majelis zikir darul khalik.
2. Melakukan Mengenalkan Aplikasi Framework Flutter untuk pendukung system informasi berbasis android.
3. Melakukan pendataan jemaah.

1.6 Manfaat Penelitian

Adapun tujuan penelitian yang digunakan dalam penelitian ini adalah sebagai berikut:

1. Menambah serta memperdalam pengetahuan atas teori-teori pembuatan Sitem informasi berbasis Android.
2. Mempermudah dalam kordinasi antar Jemaah dalam mendapatkan informasi yang tepat waktu.
3. Menjaga kelestarian data Jemaah.

BAB II

LANDASAN TEORI

2.1 *Mobile Application Framework*

Banyak kerangka kerja dan SDK (Software development kits) yang tersedia untuk membangun sebuah perangkat bergerak yang kompatibel dengan dua sistem operasi, dalam kasus ini IOS dan android. Mobile Application Framework tersebut digunakan untuk memudahkan pengembang dalam membuat perangkat bergerak dengan single codebase. Single codebase memungkinkan pengembang untuk membuat aplikasi yang berbeda platform hanya dengan satu bahasa pemrograman. Dengan single codebase pengembang dapat menghasilkan perangkat lunak bergerak yang dapat berjalan di berbagai sistem operasi terutama IOS dan android. Kerangka kerja yang demikian dikenal juga dengan istilah cross platform framework.

Cross platform framework merupakan native atau kombinasi antara native aplikasi dan web aplikasi yang dapat membuat lebih dari satu platform, dikembangkan menggunakan library yang bersifat terbuka dan memiliki akses ke dalam beberapa bagian fungsi perangkat seperti kamera, GPS, map, dst (Caicedo, 2015). Untuk pembuatan aplikasi dalam kasus ini menggunakan framework Flutter dan React Native karena kedua framework tersebut merupakan cross platform framework yang disukai dan diinginkan (Stackoverflow, 2019)

Menurut (Liu & Yu, 2011), dapat dikatakan android framework jika meliputi:

- a. Dapat mengakses data dari aplikasi lain maupun data sendiri.

- b. Terdapat banyak komponen yang dapat dikembangkan dan digunakan untuk membangun antarmuka seperti text, tombol, dan lainnya.
- c. Notification manager menampilkan status peringatan khusus.
- d. Resource manager menyediakan sumber daya seperti grafik, tata letak berkas.
- e. Activity manager mengelola siklus aplikasi dan menyediakan navigasi.

2.2 Flutter

Flutter adalah toolkit antar muka untuk membuat aplikasi yang dikompilasi secara native untuk seluler, website, desktop dari single codebase (Flutter web, 2020). Flutter dirilis pertama kali pada tahun 2017 oleh google, dibangun menggunakan bahasa C, C++, Dart, dan Skia (2D rendering engine).

Saat pengembangan aplikasi, Flutter didukung dengan adanya fitur hot-reload yang dianggap salah satu fitur utama yang diunggulkan untuk meningkatkan pengembangan. Fitur hot-reload memudahkan pengembang dalam membuat user interface, menambahkan fitur aplikasi, dan memperbaiki bug. Hot-reload berkerja dengan memasukkan kode yang sudah diperbarui ke dalam dart virtual machine yang sedang berjalan. Setelah virtual machine memperbarui class dan fungsi, secara otomatis kerangka kerja Flutter akan membangun ulang widget sehingga pengembang dapat secara langsung melihat efek perubahan (Flutter web, 2020).

Flutter adalah SDK yang dikembangkan oleh Google untuk mengembangkan aplikasi seluler yang memungkinkan pembuatan aplikasi berkinerja tinggi dan dapat dipublikasikan untuk platform Android dan iOS dari satu basis kode.

Flutter mudah dipelajari karena ditulis dalam bahasa pemrograman Dart, yang akan terlihat familier bagi siapa saja yang pernah bekerja dengan Java atau Javascript. Selain itu, Flutter hadir dengan kerangka kerja fungsionalitas reaktif, mesin tampilan 2D, widget bawaan, dan alat pengembang. (CrossTechno, 2019).

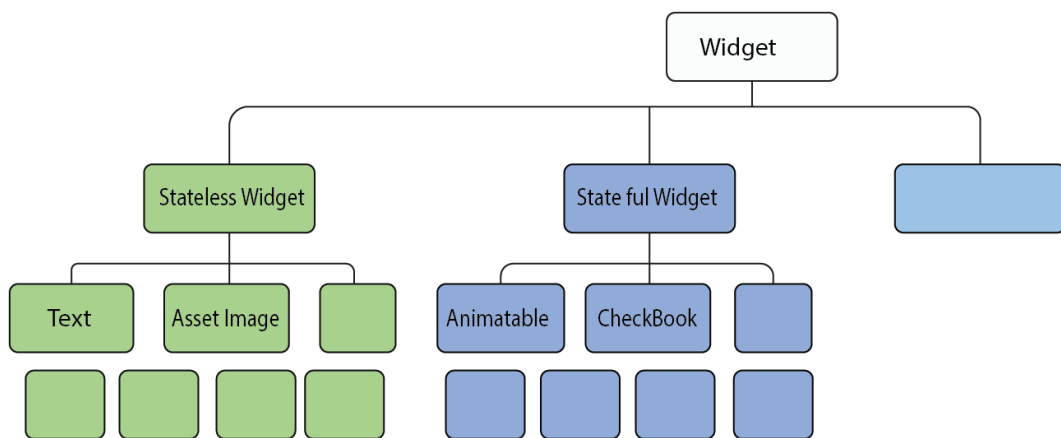
2.2.1 Dart

Flutter menggunakan bahasa pemrograman Dart sebagai bahasa utamanya. Dart adalah bahasa pemrograman yang dikembangkan dan dikelola oleh google pada tahun 2011. Google menggunakan dart untuk membuat aplikasi berkualitas tinggi untuk iOS, android, dan web dengan fitur yang ditujukan untuk pengembangan sisi klien (Sharma & Gupta, 2020). Pada awalnya dart dikembangkan sebagai pengganti dan penerus javascript. Dart mengimplementasi sebagian karakteristik dalam javascript (ES7) seperti “async” dan “await”. Namun, untuk menarik pengembang yang tidak terbiasa dengan JavaScript, Dart memiliki sintaksis yang lebih mirip dengan java (Wu, 2018).

2.2.2 Widget

Widget merupakan komponen penting dalam kerangka kerja Flutter untuk membangun tampilan. Flutter memiliki model objek yang konsisten dan terpadu, sehingga setiap widget memiliki deklarasi dalam pembangunan tampilan yang tidak dapat diubah. Flutter memiliki struktur widget yang dapat disesuaikan dan diperluas, serta yang paling utama Flutter tidak menggunakan komponen atau widget OEM tetapi menyediakan widget sendiri (Fayzullaev et al., 2018).

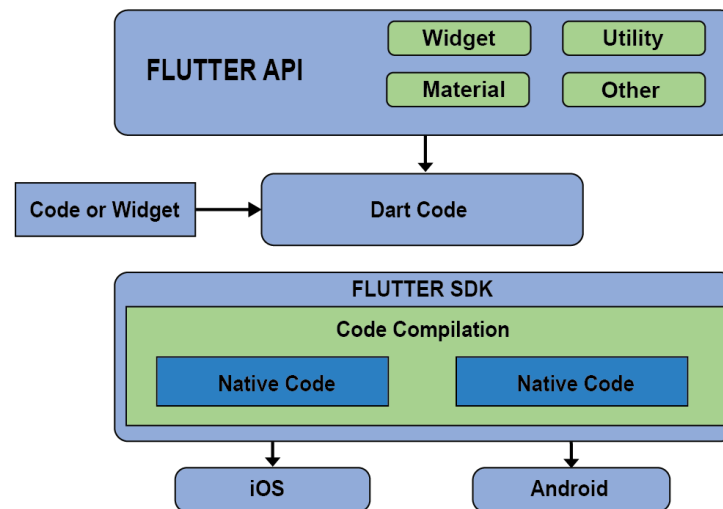
Ada dua jenis widget utama yang ada dalam Flutter yaitu stateless widget dan stateful widget. Stateful widget memiliki state yang dapat berubah. State adalah informasi yang dapat dibaca secara bersama ketika sebuah widget dibuat dan dapat berubah selama pemakaian widget (Flutter Api, 2020). Sedangkan stateless widget tidak membutuhkan state, keduanya berada dalam level hierarki yang sama seperti terlihat pada Gambar 2.1.



Gambar 2.1 Hierarki Flutter Class Widget Sumber: (Fayzullaev et al.,2018)

2.2.3 *Architecture Flutter*

Flutter menjalankan sebagian besar kerangka kerja dan kode aplikasi di dalam Dart virtual machine yang ringan. Kerangka kerja Flutter dibangun menggunakan Dart sedangkan rendering diimplementasikan dalam C++. Kode sumber dikompilasikan ke kode native menggunakan fitur kompilasi AoT (Ahead of Time) Dart seperti pada Gambar 2.2. Dalam android kode program C/C++ dikompilasi oleh AoT ke dalam kode native dengan NDK Android, sedangkan iOS akan dikompilasi dengan LLVM (Low Level Virtual Machine), dan juga semua kode program Dart (Fayzullaev et al., 2018).



Gambar 2.2 Arsitektur Flutter

2.2.4 Rest Fire

REST (*Representational State Transfer*) adalah kerangka komunikasi berbasis web yang sering digunakan untuk mengembangkan aplikasi berbasis web. Umumnya transmisi data dilakukan melalui protokol HTTP (Hypertext Transfer Protocol). Roy Fielding memperkenalkan REST pada tahun 2000. Arsitektur REST terdiri dari dua komponen: server REST menawarkan sumber daya (sumber daya/data), dan klien REST mengakses dan menampilkan sumber daya ini untuk penggunaan lebih lanjut. URI (Universal Resource Identifiers) atau pengidentifikasi global digunakan untuk mengidentifikasi setiap sumber daya. Materi ini tersedia dalam format teks, format JSON atau format XML. Secara umum, formatnya didasarkan pada JSON dan XML.

1) Keuntungan dari REST

Secara desain dan konsep, bahasa dan platform agnostik lebih mudah dibangun daripada SOAP. Mereka mudah dimengerti, tidak bergantung pada alat kecil, dan tidak memerlukan lapisan pesan tambahan.

2) Kelemahan REST

Asumsikan arsitektur komunikasi point-to-point; tidak dapat digunakan dalam pengaturan komputasi terdistribusi dengan satu atau lebih perantara. Karena kurangnya dukungan standar untuk keamanan, regulasi, dan keandalan pesan, antara lain, mengembangkan layanan dengan persyaratan yang lebih kompleks menjadi lebih menantang (“selesaikan sendiri”). Properti ini terkait dengan model transport HTTP.

Beberapa metode HTTP sering digunakan dalam sistem berbasis REST, antara lain sebagai berikut:

- a) *GET*, yang memungkinkan akses read-only ke sumber daya.
- b) *PUT*, yang digunakan untuk menambah sumber daya baru.
- c) *DELETE*, yang digunakan untuk memadamkan sumber daya secara permanen.
- d) *POST*, yang digunakan untuk mengupdate atau membuat resource baru.
- e) *OPTIONS*, yang digunakan untuk mendapatkan dukungan untuk operasi sumber daya.

Karena aplikasi yang mengirimkan data dapat dikembangkan dalam beberapa bahasa pemrograman atau beroperasi pada berbagai platform, standar layanan web digunakan untuk bertukar data antar aplikasi atau sistem. SOAP dan REST adalah dua contoh implementasi layanan web. Layanan web RESTFUL adalah layanan web yang sesuai dengan arsitektur REST. Layanan web ini mengimplementasikan ide arsitektur REST melalui penggunaan metode HTTP.

Klien mengirimkan data atau permintaan ke server melalui Permintaan HTTP, dan server merespons melalui Respons HTTP. Komponen permintaan HTTP meliputi: Kata kerja atau metode HTTP yang digunakan, seperti GET, POST, DELETE atau PUT. Uniform Resource Identifier (URI) digunakan untuk mengidentifikasi lokasi sumber daya di server. Versi HTTP, menunjukkan versi HTTP yang digunakan, mis. HTTPv1.1. Header Permintaan mencakup informasi yang terkait dengan Permintaan HTTP. Misalnya, jenis klien/browser, format yang didukung klien, format isi pesan, dan pengaturan cache. Tubuh permintaan, data itu sendiri. Sedangkan komponen HTTP Response Status/Response menyediakan status server untuk resource yang diminta. Misalnya, 404 menunjukkan bahwa sumber tidak dapat dilacak, tetapi 200 menunjukkan bahwa sumber itu ditemukan. Versi HTTP.

Header Respons menyertakan informasi tentang respons HTTP. Misalnya, jenis server, panjang konten, jenis materi, dan waktu respons. Badan respons, materi data yang ditransmisikan (Feridi, 2019).

2.2.5 JSON (*JavaScript Object Notation*)

JSON (JavaScript Object Notation) adalah format transmisi data ringan yang mudah dibaca dan ditulis orang serta komputer untuk menerjemahkan dan menghasilkan (generate). Format ini berasal dari bagian Bahasa Pemrograman JavaScript, Standar ECMA-262, Edisi Ketiga, Desember 1999. JSON adalah format teks yang tidak bergantung pada bahasa pemrograman apa pun karena penggunaan gaya bahasa yang sering digunakan oleh pemrogram keluarga C seperti C , C++, C#, Java, JavaScript, Perl, dan Python. Sebagai hasil dari kualitas ini, JSON adalah bahasa pertukaran data yang tepat (Hyderabad, 2012).

2.3 React Native

Flutter React Native adalah kerangka kerja yang digunakan untuk membuat aplikasi seluler cross platform yang pertama kali dikenalkan pada tahun 2015. React Native dirilis pertama kali pada awal tahun 2015 oleh facebook yang hanya mendukung pengembangan aplikasi iOS, tetapi pada bulan September 2015 kerangka kerja diperluas untuk mendukung pengembangan aplikasi android (Witte & Weitershausen, 2015).

React Native dibangun dengan prinsip dan konsep seperti ReactJS yang merupakan kerangka kerja open source pada 2013. Secara struktur kode sangat mirip, yang membedakan antara React Native dengan ReactJs adalah bahwa ReactJS beroperasi pada Document Object Model (DOM) dalam web browser, sedangkan React Native beroperasi pada aplikasi seluler (Hansson & Vidhall, 2016). Sama halnya dengan Flutter yang memiliki fitur hot-reload, untuk

mempercepat dalam pengembangan aplikasi, React Native memiliki fitur fast refresh. Fast refresh merupakan fitur yang memungkinkan pengembang mendapatkan umpan balik secara langsung dalam perubahan component.

2.3.1 *JSX*

Dalam pengembangan, React Native menggunakan bahasa pemrograman Javascript sebagai bahasa utamanya. Untuk mempermudah pembangunan aplikasi React Native, facebook mengembangkan ekstensi sintaksis JavaScript yang disebut JSX (JavaScript Syntax Transformer) yang bertujuan untuk mempermudah dalam membaca dan menulis kode program. JSX merupakan ekstensi untuk mengatur tampilan. JSX akan dikompilasi menjadi objek Javascript normal ketika sedang dikompilasi (Wu, 2018). Penulisan sintaksis JSX dengan Javascript regular tidak jauh berbeda, JSX terlihat mirip dengan XML, perbedaan penulisan sintaksis dalam React bisa dilihat pada Gambar 2.3..

<pre>var React = require('react'); var ReactDOM = require('react-dom'); ReactDOM.render(<h1> Hello World!</h1>; document.getElementById('example'));</pre>	<pre>var React = require('react'); var ReactDOM = require('react-dom'); ReactDOM.render(React.DOM.h1 (null, "Hello World!"), document.getElementById('example'));</pre>
JSX	JS

Gambar 2.3 Menulis Hello World dalam React dengan JSX dan JavaScript regular. Sumber: (Hansson & Vidhall, 2016)

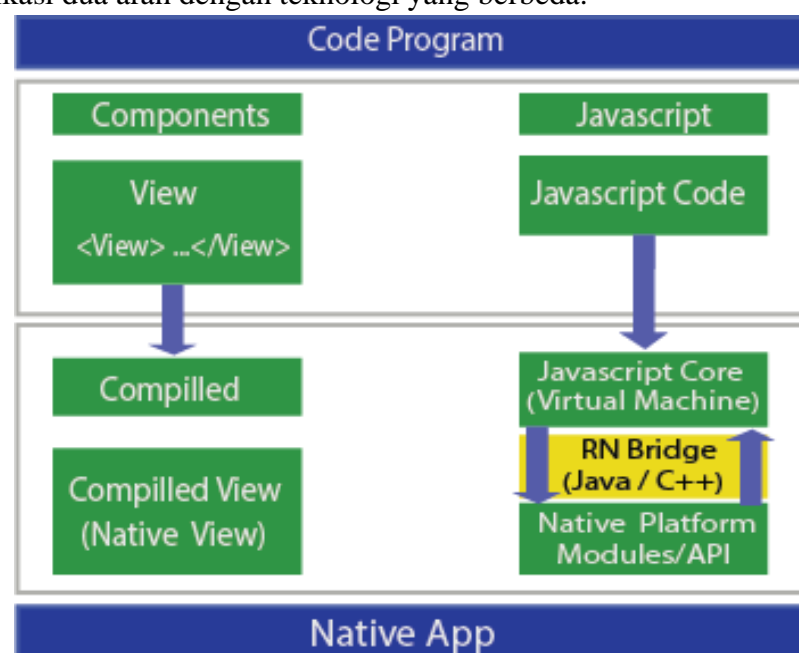
2.3.2 *Component*

Component merupakan bagian penting dalam pembangunan tampilan pada React Native. Secara konsep, component seperti function Javascript digunakan untuk membuat tampilan yang terbagi dalam beberapa bagian serta dapat digunakan kembali (ReactJS, 2020). Component memiliki dua model data yaitu

prop dan state. Prop atau property biasanya digunakan untuk pertukaran data antara component parent dengan component child, sedangkan state hanya bisa dilakukan dalam suatu component dan data tidak bisa diakses dari component lain. State digunakan untuk mengelola data yang berubah seiring waktu yang berasal dari interaksi pengguna (React Fundamentals, 2020).

2.3.3 Arsitektur React Native

Arsitektur utama framework React Native terdiri dari javascript core (virtual machine), React Native bridge, dan modul native seperti terlihat pada Gambar 2.4. Kode program javascript berjalan pada virtual machine bersama library yang digunakan dalam pengembangan. Untuk menghubungkan kode program pada virtual machine dengan modul native dan library pihak ketiga, React Native menggunakan bridge (Kuitunen, 2019). React Native Bridge merupakan konsep paling penting dalam arsitektur React Native karena bridge digunakan untuk berkomunikasi dua arah dengan teknologi yang berbeda.



Gambar 2. Arsitektur React Native

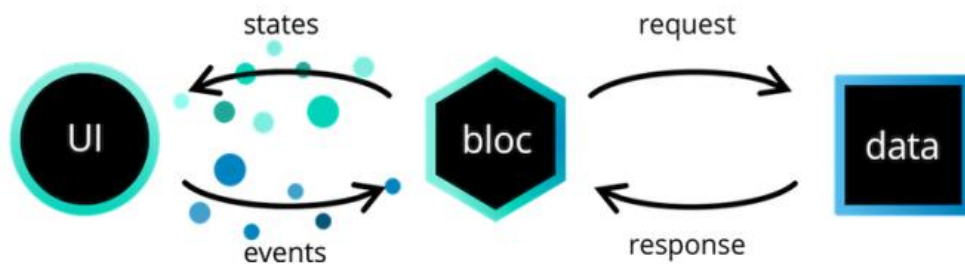
2.4 State Management

State management atau pola desain berfungsi untuk memisahkan bisnis logika dengan tampilan antarmuka. Pemilihan dan penggunaan state management berpengaruh terhadap proses pengembangan tertentu karena hampir semua bagian dari aplikasi bergantung pada management yang dipilih (Faust & Klein, 2020). Penggunaan state management berpengaruh terhadap domain penelitian yang digunakan terutama performance, salah satunya pertukaran state antara component. Ada banyak state management yang dapat digunakan dalam Flutter maupun React Native, tetapi pada penelitian ini menggunakan BloC sebagai state management Flutter dan Redux sebagai state management React Native karena kedua state management tersebut direkomendasikan oleh setiap pengembang kedua framework tersebut, selain itu juga BloC dan Redux sangat populer sehingga banyak digunakan oleh pengembang aplikasi.

2.4.1 Flutter BloC

The Business Logic Component atau Flutter BloC dibuat oleh Google dan di kenalkan dalam konferensi Google I/O pada tahun 2018, BloC merupakan library yang membantu untuk memisahkan antara business logic dan presentation seperti terlihat pada Gambar 2.5. Dalam BloC memungkinkan kita untuk memisahkan aplikasim menjadi 3 lapisan utama yaitu:

- a. *Presentation* (UI), menampilkan antar muka sesuai dengan state BloC dengan me render dirinya sendiri berdasarkan satu atau dua BloC state, dan juga menangani masukan dari pengguna dengan events kedalam BloC.
- b. *Bussiness Logic*, semua perubahan state terjadi pada lapisan ini, Bussiness Logic akan menanggapi masukan dari lapisan presentation dengan state baru, dengan mengambil data pada satu atau beberapa repository untuk membangun state dalam aplikasi.
- c. *Data* (Repository dan Data Provider), mengambil satu atau lebih sumber data seperti database, network request.



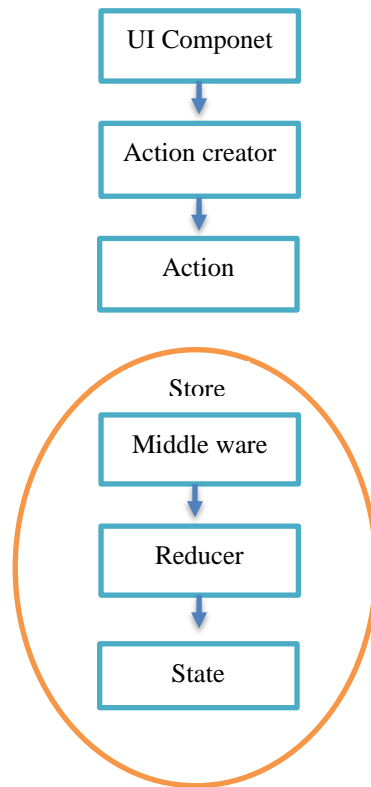
Gambar 2.5 Arsitektur data BloC. Sumber: (BloC Architecture, 2020)

2.4.2 React Native Redux

Redux merupakan library javascript terbuka dengan menerapkan ide dasar dari pola flux yang digunakan untuk manajemen state. Redux memiliki aliran data seperti terlihat pada Gambar 2.6. Aliran data tersebut terbagi dalam 3 bagian utama (Abramov, 2020).

- a. *Action*, merupakan proses yang memuat informasi dengan mengirim data dari aplikasi menuju ke store.
- b. *Reducer*, bertugas mengolah data dengan merubah state menjadi response.

c. Store, bertugas untuk menggabungkan antara action dengan reducer.



Gambar 2.6 Arsitektur data Redux. Sumber: (Kuparinen, 2019)

2.5 Domain

Menurut Gerdessen (2007), domain adalah bidang pengetahuan untuk masalah atau tugas tertentu dengan serangkaian konsep dan istilah yang terbatas. Tujuan digunakannya domain antara lain yaitu untuk mengurangi kompleksitas masalah dengan memisahkan menjadi area yang lebih kecil agar mudah untuk dikelola.

Terdapat banyak domain yang dapat digunakan untuk memkomparasikan framework seperti pada Tabel 2.1.

Tabel 2.1 Domain Komparasi Sumber (Gerdessen, 2007)

<i>No.</i>	<i>Domain</i>
1	Scalability
2	Persistency
3	Security
4	Availability
5	Customisability
6	Interoperability
7	Skeleton implementation
8	Performance
9	Separation of concern
10	Modifiability
11	Testability

Dari beberapa domain pada Tabel 2.1 dipilih beberapa domain yang memungkinkan untuk diimplementasikan dengan melakukan analisis pada kerangka kerja berdasarkan domain yang paling berdampak pada pengembang. Sehingga didapat domain serta parameter yang digunakan dalam membandingkan kedua framework sebagai berikut:

- a. Customisability, kemudahan pengubahan perangkat lunak oleh pengguna dengan memodifikasi parameter desain (Jiao & Tseng, 2004). Pemilihan customizability untuk melakukan komparasi framework karena berdampak terhadap kemudahan pengembang pada saat memulai pembangunan aplikasi dan penambahan fitur aplikasi. Untuk mengukur customizability dengan melakukan pengukuran tingkat kustomisasi pada fitur yang ada pada masing-masing framework.
- b. Modifiability, kemudahan dalam perubahan dan penambahan sistem (Saliu et al., 2009). Pemilihan modifiability untuk komparasi karena berdampak terhadap pengembang ketika melakukan perubahan dan penambahan fitur pada aplikasi. Untuk mengukur modifiability dengan mengukur jumlah fungsi dan dependensi, pengukuran jumlah fungsi dan dependensi mempengaruhi proses lamanya pengembang dalam melakukan perubahan pada aplikasi karena banyak komponene yang harus diubah.
- c. Performance, sejauh mana sistem atau perangkat lunak dalam mencapai tujuan dengan waktu yang ditentukan (Oshana, 2013). Pemilihan performance berdampak terhadap pengembang pada saat melakukan pembuatan aplikasi karena memudahkan pengembang dalam merepresentasikan performance aplikasi yang sedang dikembangkan terhadap performance setelah aplikasi rilis. Pengukuran dapat dilakukan dengan mengukur parameter memori, CPU, dan frame per second karena sudah tersedia fitur pada framework.

- d. Testability, karakteristik yang menunjukkan suatu komponen dalam menentukan status secara akurat (Garousi et al., 2019). Pemilihan testability berdampak terhadap pengembang ketika melakukan pengujian aplikasi agar sesuai yang diinginkan. Pengujian yang dapat dilakukan pada kedua kerangka kerja dapat dilakukan dengan mengimplementasikan unit testing dan integration testing.

2.6 Sistem Informasi

Sistem Informasi ketika diimplementasikan dengan benar, sistem informasi adalah kumpulan operasi terstruktur yang menghasilkan data untuk membantu pengambilan keputusan dan pengendalian dalam organisasi.

Sistem informasi terstruktur, struktur organisasi sosioteknik yang digunakan untuk mengumpulkan, memproses, menyimpan dan menyebarkan data. Sudut pandang sosioteknik mengidentifikasi empat komponen sistem informasi: tugas, orang, struktur (atau peran), dan teknologi. Sistem informasi komputer (CIS) adalah kumpulan manusia dan komputer yang menganalisis atau menafsirkan data. Selain itu, frasa ini juga digunakan dalam pengertian yang lebih terbatas untuk merujuk hanya pada perangkat lunak yang diperlukan untuk mengoperasikan basis data terkomputerisasi atau sistem komputer. Sistem Informasi adalah studi tentang sistem akademik dengan fokus pada informasi dan jaringan pendukung perangkat keras dan perangkat lunak yang digunakan oleh individu dan organisasi untuk mengumpulkan, menyaring, memproses, menghasilkan, dan menyebarkan data. Penekanannya adalah pada sistem informasi dengan batasan yang berbeda,

termasuk pengguna, prosesor, penyimpanan, input, output, dan jaringan komunikasi (Astuti, 2017).

Setiap sistem informasi dirancang untuk membantu operasi, manajemen dan pengambilan keputusan. Istilah "sistem informasi" mengacu pada teknologi informasi dan komunikasi (TIK) yang digunakan oleh perusahaan dan cara teknologi ini digunakan untuk mendukung kegiatan bisnis. Banyak penulis dengan jelas membedakan antara sistem informasi, sistem komputer dan proses bisnis. Meskipun sistem informasi sering memiliki komponen TIK, mereka tidak fokus pada TIK saja, melainkan berfokus pada aplikasi akhir teknologi informasi. Selain itu, sistem informasi berbeda dari proses bisnis. Mengontrol kinerja operasi perusahaan disederhanakan dengan sistem informasi.

Alter membuat kasus untuk memahami sistem informasi sebagai bagian dari sistem kerja yang berbeda. Sistem kerja adalah seperangkat prosedur dan aktivitas yang dilakukan oleh orang atau mesin untuk menciptakan barang atau jasa tertentu bagi pelanggan. Sistem informasi adalah jenis sistem kerja yang berfokus pada memperoleh, mentransfer, menyimpan, mengambil, memodifikasi dan menyajikan data. Oleh karena itu, sistem informasi berkaitan erat dengan sistem data dan sistem aktivitas. Sistem informasi adalah jenis sistem komunikasi di mana data direpresentasikan dan diproses dengan cara yang sama seperti memori sosial ditampilkan dan diproses. Selain itu, sistem informasi dapat dianggap sebagai bahasa semi formal yang memfasilitasi pengambilan keputusan dan tindakan manusia. Informatika organisasi terutama berkaitan dengan sistem informasi.

2.7 Konsep Informasi

Informasi adalah kumpulan fakta yang telah diproses untuk memberikan lebih banyak pengetahuan yang dapat digunakan untuk menyelesaikan suatu tugas. Suatu informasi dianggap bermanfaat jika memberikan manfaat yang melebihi nilai data saat ini.

Ada beberapa definisi informasi. H.M. Jogiyanto mendefinisikan informasi sebagai berikut dalam bukunya *Analysis and Design of a Structured Approach to Information Systems*: “Informasi adalah data yang telah diolah menjadi bentuk yang memiliki arti bagi penerimanya dan memiliki nilai yang nyata dan dirasakan untuk keputusan saat ini atau masa depan.” Gordon B. Davis mendefinisikan informasi sebagai berikut dalam bukunya *The Basic Framework of Management Information Systems*: “Informasi adalah data yang telah diolah menjadi bentuk yang dapat digunakan dan berwujud, atau menjadi bentuk nilai yang dapat dipahami dalam pilihan sekarang dan yang akan datang.” (Jogiyanto, 2016)

2.7.1 Siklus Informasi

Siklus informasi dimulai dengan data mentah diubah menjadi informasi (*output*), dan kemudian penerima menerima informasi tersebut untuk digunakan sebagai dasar pengambilan keputusan dan tindakan, yang memerlukan perubahan data kembali menjadi data mentah. Data kemudian akan dikumpulkan sebagai input, membentuk loop.

2.7.2 Kualitas Informasi

Kualitas informasi ditentukan oleh beberapa faktor. Tiga faktor yang mempengaruhi antara lain:

1. Akurat

Informasi tersebut harus akurat dan tidak salah atau menyesatkan. Akurasi juga menyiratkan bahwa data harus secara akurat mewakili tujuan. Informasi harus akurat karena dapat terjadi banyak gangguan (*noise*) antara sumber dan penerima informasi yang dapat mengubah atau merugikan informasi tersebut.

2. Tepat Pada Waktunya

Informasi penerima tidak dapat ditunda. Informasi yang buruk akan kehilangan nilainya, karena informasi berfungsi sebagai dasar pengambilan keputusan. Jika suatu pilihan dibuat terlambat, itu dapat membawa bencana bagi organisasi.

3. Relevan

Relevan dalam skenario ini adalah di mana pengetahuan memiliki nilai dan kegunaan dalam penerapannya. Relevansi informasi bervariasi menurut individu.

2.7.3 Konsep Dasar Sistem Informasi

Sistem informasi terdiri dari banyak prinsip dasar. Definisi berikut berlaku untuk sistem informasi:

1. Berbasis komputer dan Sistem Manusia/Mesin

- a. Berbasis komputer: sebuah. Berbasis komputer: desainer harus memiliki pemahaman yang baik tentang komputer dan pemrosesan informasi.
 - b. Sistem manusia-mesin: manusia berfungsi sebagai manajer dan robot beroperasi sebagai alat pemrosesan informasi. Ada operasi yang harus dilakukan secara manual oleh orang dan proses yang dapat diotomatisasi oleh mesin. Oleh karena itu, diperlukan suatu sistem prosedur/manual.
2. Sistem basis data terintegrasi
- a. Kehadiran database yang digunakan secara kolaboratif (bersama) dalam sistem manajemen database. (*database management system*).
3. Mendukung Operasi
- a. Informasi yang diproses dan dihasilkan digunakan untuk membantu kegiatan organisasi.

2.8 Majelis Zikir Abdul Khalik

Badan Kordinator Majelis Zikir Darul khalik di Jalan Tj Selamat kec. Sunggal Kabupaten Deli Serdang Sumut 20351, adalah Majelis dibawah Yayasan abdul khalik fajduani yang melakukan Pengembangan pendidikan pesantren modern dan pengajaran secara modern, Mengembangkan ajaran agama Islam berdasarkan Al Quran, Al Hadits dan Tasawuf Islam. Badan Kordinator Majelis Zikir Darul khalik melakukan sistem pendataan secara manual dan sederhana, jika jamaah tersebut lupa membawa salah satu dokumen tersebut, sehingga memerlukan

waktu dalam proses pendataan. Sistem kerja tersebut dirasa masih kurang efektif dan efisien, karena dalam pencatatan manual bisa saja menyebabkan beberapa kesalahan dan suatu masalah. Serta membutuhkan banyak waktu untuk suatu proses pendataan dan pembuatan laporan karena harus mencari dan membuka kembali formulir pendataan terlebih dahulu.



Gambar 2.1 Struktur MZ Darul khalik

2.8.1 Sejarah MZ Darul khalik

Majelis Zikir Darul Khaliq dibangun pada bulan Mei 2019 bertepatan dengan bulan Ramadhan 1440 H. Nama Majelis Zikir Darul Khaliq berdasarkan surat Pemberitahuan BAKORMAZ nomor: 112/2-BAKORMAZ//X/2019 pada tanggal, 21 Oktober 2019. Nama Majelis Zikir Darul tersebut digunakan sebagai identitas yang mudah dikenal oleh pemerintah dan masyarakat. Pembangunan

Majelis Zikir Darul Khaliq dengan penerapan pembangunan nilai-nilai bukan hanya sekedar pembangunan fisik bangunan.

Tujuan sekaligus ciri khas Majelis Zikir Darul Khaliq adalah kemegahan spiritual bukan kemegahan fisik, fisik adalah penunjang spiritual yang megah. Spiritual megah adalah terjadinya masyarakat yang menjalankan islam kafah; mencintai zikir, mencintai Rasulullah SAW, mencintai Alquran, bermanfaat bagi sesama dengan kata lain masyarakat yang diridhoi Allah SWT.

2.9 *Unified Modelling Language*

Unified Modeling Language adalah bahasa pemrograman berbasis model yang memungkinkan pengembang untuk menggambarkan, melihat, menghasilkan dan merekam artefak sistem perangkat lunak. Oleh karena itu, UML memastikan skalabilitas, keamanan, dan kekokohan objek-objek ini selama implementasi. UML merupakan komponen penting dalam pengembangan perangkat lunak berorientasi objek. Ini menciptakan representasi visual dari sistem perangkat lunak melalui penggunaan notasi grafis. Arsitektur UML dibangun di atas kenyamanan objek meta, yang memberikan dasar untuk mengembangkan bahasa pemodelan. Mereka cukup tepat untuk menghasilkan aplikasi yang lengkap. Model UML yang sepenuhnya dapat dieksekusi dapat digunakan pada berbagai platform dan dengan berbagai teknologi, dan dapat diintegrasikan ke dalam semua fase siklus pengembangan perangkat lunak. UML adalah bahasa pemodelan visual yang memungkinkan pengguna untuk membuat model visual yang ekspresif dan siap digunakan. Selain itu, ini memungkinkan persepsi pengembangan tingkat tinggi seperti kerangka kerja, pola, dan kolaborasi.

Tujuan penggunaan model ini adalah untuk memastikan komponen mana yang berada dalam lingkup sistem yang dibahas dan sifat hubungan antara sistem dan subsistemnya dengan sistem lainnya (Sukmawati & Priyadi, 2019).

2.9.1 Use Case Diagram


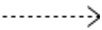
Use Case Diagram adalah Use Case Diagram adalah representasi dari bagaimana berbagai jenis pengguna berinteraksi dengan sistem untuk menyelesaikan masalah. Dengan demikian, ini menggambarkan tujuan pengguna, interaksi antara pengguna dan sistem, dan perilaku sistem yang diperlukan untuk mencapai tujuan ini. Model use case terdiri dari banyak bagian model. Komponen yang paling penting dari model adalah kasus penggunaan, aktor, dan hubungan mereka. Untuk memfasilitasi komunikasi, diagram use case digunakan untuk menggambarkan bagian dari model secara visual. Biasanya, model tertentu mungkin memiliki banyak diagram use case, masing-masing menggambarkan subset fitur model yang penting untuk tujuan tertentu. Beberapa use case diagram mungkin memiliki potongan model yang sama, tetapi setiap instance harus konsisten. Jika alat digunakan untuk memelihara model kasus penggunaan, persyaratan konsistensi ini diotomatisasi, sehingga setiap perubahan pada elemen model (misalnya, mengubah nama) secara otomatis tercermin dalam setiap diagram kasus penggunaan yang berisi elemen itu.


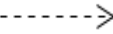


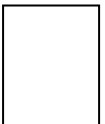


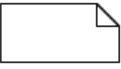
Model use-case dapat dimasukkan dalam model kasus penggunaan untuk memfasilitasi analisis, komunikasi, navigasi, pengembangan, pemeliharaan, dan perencanaan. Faktanya, sebagian besar model use case adalah teks, dengan bahasa

yang terkait dengan setiap aspek model use case yang ditentukan dalam Spesifikasi *Use Case*. Spesifikasi ini merinci urutan kejadian yang terjadi dalam use case. Model *use case* bertindak sebagai utas pemersatu selama proses pengembangan sistem. Ini berfungsi sebagai definisi utama persyaratan fungsional sistem, titik awal untuk analisis dan desain, masukan untuk perencanaan iterasi, titik awal untuk mengembangkan kasus uji, dan titik awal untuk dokumentasi pengguna. (Kurniawan, 2018).

Use case diagram Use case diagram adalah representasi visual dari use case, aktor dan interaksinya. *Use Case Diagram* dapat digunakan untuk merepresentasikan setiap kebutuhan dalam sistem, sehingga pengoperasian sistem dapat dipahami dengan jelas. Tabel 2.1 merupakan representasi *use case* dalam diagram *use case*.

Tabel 2.1 Simbol Use Case Diagram

No.	Gambar	Nama	Keterangan
1		<i>Actor</i>	Menspesifikasikan himpunan peran yang pengguna mainkan ketika berinteraksi dengan <i>use case</i> .
2		<i>Dependency</i>	Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri (<i>independent</i>) akan mempengaruhi elemen yang bergantung padanya elemen yang tidak mandiri.

3		<i>Generalization</i>	Hubungan dimana objek anak berbagi perilaku dan struktur data dari objek yang ada di atasnya.
4		<i>Include</i>	Menspesifikasikan bahwa <i>use case</i> sumber secara <i>eksplisit</i> .
5		<i>Extend</i>	Menspesifikasikan bahwa <i>use case</i> target memperluas perilaku dari <i>use case</i> sumber pada suatu titik yang diberikan.
6		<i>Association</i>	Apa yang menghubungkan antara objek satu dengan objek lainnya.
7		<i>System</i>	Menspesifikasikan paket yang menampilkan sistem secara terbatas.
8		<i>Use Case</i>	Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu actor
9		<i>Collaboration</i>	Interaksi aturan-aturan dan elemen lain yang bekerja sama untuk menyediakan perilaku yang lebih besar dari jumlah dan elemen-elemennya (sinergi).
10		<i>Note</i>	Elemen fisik yang eksis saat aplikasi dijalankan dan mencerminkan suatu sumber daya komputasi

Sumber: (Kurniawan, 2018)

2.10 Aplikasi

Aplikasi Perangkat lunak (disingkat aplikasi) adalah program atau kumpulan program yang ditujukan untuk pengguna akhir. Pengolah kata, spreadsheet, aplikasi akuntansi, browser web, klien email, pemutar video, penampil file, simulator, dan konsol game adalah contoh aplikasi. Perangkat lunak aplikasi adalah kata kolektif yang mengacu pada semua aplikasi. Hal ini berbeda dengan perangkat lunak sistem, yang terutama terkait dengan pengoperasian komputer (Pratama, 2010).

Aplikasi dapat dikemas dengan komputer dan perangkat lunak sistem operasinya atau didistribusikan secara independen, dan dapat dikembangkan sebagai proyek pribadi, sumber terbuka, atau akademis. Aplikasi seluler adalah yang dikembangkan untuk platform seluler.

Dua bentuk utama perangkat lunak adalah perangkat lunak aplikasi dan perangkat lunak sistem. Perangkat lunak sistem menangani aktivitas internal komputer, biasanya melalui sistem operasi (OS). Selain itu, ia memelihara perangkat periferan seperti perangkat penyimpanan, printer, dan layar. Sebaliknya, perangkat lunak aplikasi, sering dikenal sebagai program aplikasi, menginstruksikan komputer untuk mengeksekusi instruksi pengguna.

Perangkat lunak sistem terdiri dari aplikasi latar belakang yang memungkinkan program aplikasi dijalankan. Kompiler, penginstal, dan alat manajemen file adalah contoh aplikasi perangkat lunak sistem, seperti sistem operasi itu sendiri. Aplikasi aplikasi berjalan di atas perangkat lunak sistem, yang terdiri dari program "tingkat rendah". Perangkat lunak sistem diinstal secara

otomatis selama instalasi sistem operasi. Pengguna, di sisi lain, memiliki kemampuan untuk memilih aplikasi mana yang diinstal pada mesin mereka.

Beberapa contoh program aplikasi meliputi:

1. Suite aplikasi: Kumpulan aplikasi terkait rangkaian aplikasi.
2. Perangkat lunak perusahaan: Memastikan aliran data dan persyaratan proses organisasi terpenuhi di semua divisi.
3. Perangkat lunak untuk pekerja informasi: Memungkinkan pengguna untuk mengembangkan dan mengelola informasi.
4. Perangkat lunak akses materi: Terutama digunakan untuk mendapatkan akses tak terbatas ke konten tanpa mengedit.
5. Software untuk pengembangan media: Membuat media elektronik dan cetak.
6. Perangkat lunak pendidikan terdiri dari materi dan/atau fitur yang ditujukan untuk siswa atau instruktur.
7. Rekayasa produk perangkat lunak: Membuat produk perangkat lunak dan perangkat keras.

2.11 Database

Basis data (database) adalah kumpulan data yang dihubungkan bersama; database menunjukkan kumpulan data yang digunakan dalam batas-batas bisnis atau organisasi pemerintah. Database digunakan untuk menyimpan data atau informasi yang terintegrasi dengan baik dengan komputer. Sistem manajemen basis

data (DBMS) diperlukan untuk mengoperasikan basis data (*Database Management System*) (Hung et al., 2018)..

Sistem manajemen basis data (DBMS) adalah jenis perangkat lunak yang memungkinkan pengguna untuk membangun, memelihara, mengontrol, dan mengakses basis data dengan cara yang praktis dan efisien. Sistem manajemen basis data (DBMS) memungkinkan pengguna untuk menangani dan memodifikasi data yang ada dengan lebih mudah. Sedangkan RDBMS, atau Sistem Basis Data Relasional, adalah bagian dari sistem manajemen basis data yang memungkinkan komunikasi antar label. Selain RDBMS, ada jenis DBMS lain, seperti DBMS Hierarki dan DBMS Berorientasi Objek.

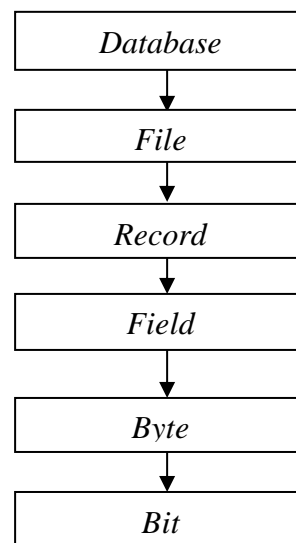
Tingkat data dapat diklasifikasikan menurut tingkat kerumitannya, dari yang paling sederhana hingga yang paling kompleks. Database diatur atau hierarkis dengan cara berikut:

Berdasarkan tingkat kompleksitas nilai data, tingkatan data dapat disusun ke dalam sebuah hierarki, mulai dari yang paling sederhana hingga yang paling kompleks. Urutan atau hierarki *database* adalah sebagai berikut:

1. Basis data adalah kumpulan dari berbagai jenis dokumen yang dihubungkan bersama dalam beberapa cara.
2. *File* adalah kumpulan record data yang berhubungan dengan suatu entitas.
3. *Record* adalah kumpulan bagian yang saling berhubungan yang mencakup semua informasi entitas. Sebuah record terdiri dari satu atau lebih field.

4. *Field* adalah unit data terkecil yang tidak dapat dibagi menjadi unit yang lebih berarti.
5. Satu *byte* adalah unit terkecil dari memori yang dapat diakses.
6. *Bit* adalah sistem bilangan biner yang terdiri dari dua jenis nilai yang berbeda yaitu 0 dan 1. Sistem biner merupakan dasar komunikasi antara manusia dan mesin. Ini terdiri dari urutan komponen listrik dan hanya mampu membedakan dua jenis kondisi, yaitu tegangan dan tidak adanya tegangan dalam suatu rangkaian.

Hierarki *database* dapat dilihat sebagai berikut:



Gambar 2.2 Hierarki Data

Akibat dari dua pengertian ini, istilah "basis data" mengacu pada kumpulan file atau tabel yang terhubung yang disimpan dalam media penyimpanan elektronik.

(*Database*) adalah kumpulan *file* atau tabel yang saling berelasi (terhubung) yang disimpan dalam media penyimpanan komputer.

2.12 *Structured Query Language*

Structured Query Language (SQL) adalah bahasa query untuk mengakses data yang disimpan dalam database relasional. Ini adalah bahasa standar de facto untuk mengelola database kontak. Hampir semua server database yang tersedia saat ini mendukung bahasa ini untuk administrasi data.

2.12.1 Sejarah SQL

Sejarah SQL dimulai dengan sebuah esai yang ditulis pada bulan Juni 1970 oleh seorang peneliti IBM bernama Jhonny Oracle, yang menjelaskan tentang konsep membangun database relasional. Selain itu, artikel ini membahas prospek pengembangan bahasa umum untuk akses database. SEQUEL akhirnya berganti nama menjadi bahasa (*Structured English Query Language*).

Setelah publikasi artikel, IBM memulai proyek untuk mengembangkan database relasional menggunakan bahasa pemrograman SEQUEL. Namun, karena masalah hukum atas nama SEQUEL, IBM mengganti nama menjadi SQL. System/R adalah nama implementasi database relasional.

Pada akhir 1970-an, sebuah perusahaan bernama Oracle didirikan dan dengan cepat menjadi terkenal karena server database populernya dengan nama yang sama. SQL telah berkembang secara signifikan sebagai akibat dari popularitas

John Oracle yang meningkat, sampai pada titik di mana sekarang menjadi bahasa standar de facto untuk administrasi database.

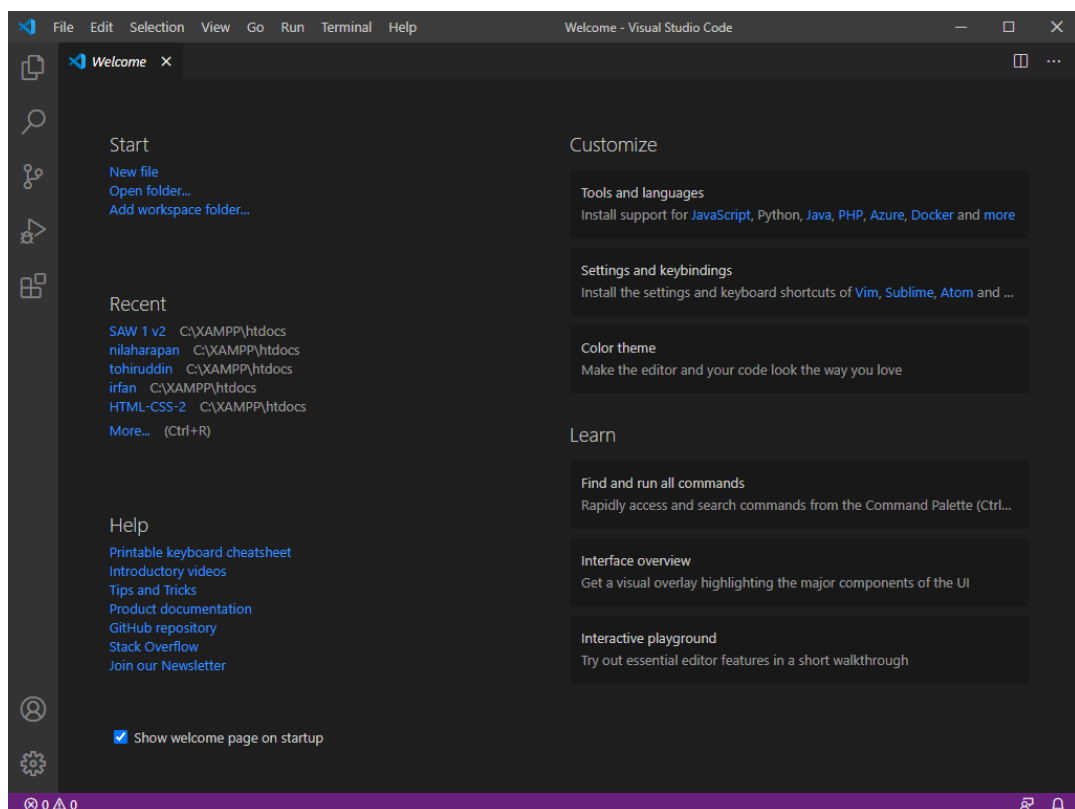
2.12.2 Standardisasi

Standardisasi SQL dimulai pada tahun 1986, dengan publikasi ANSI dari standar SQL. Standar ini sering disingkat SQL86. Standar diubah lagi pada tahun 1989 dan 1992. SQL92 adalah versi terbaru. Pada tahun 1999, standar baru, SQL99, atau umumnya dikenal sebagai SQL99, diterbitkan, namun sebagian besar implementasi masih mematuhi SQL92. Saat ini, tidak ada server database yang sepenuhnya mendukung SQL92. Hal ini disebabkan oleh fakta bahwa setiap server berbicara dengan dialeknnya sendiri.

2.13 Visual Studio Code

Microsoft memproduksi Visual Studio Code, editor kode sumber gratis untuk Windows, Linux, dan macOS. Debugging, penyorotan sintaks, penyelesaian kode cerdas, cuplikan, pemfaktoran ulang kode, dan Git terintegrasi adalah semua fitur yang disertakan. Pengguna dapat menyesuaikan tema desktop, pintasan dan setelan keyboard, serta memasang ekstensi yang menawarkan fungsionalitas baru. Microsoft telah membuat kode sumber untuk Visual Studio Code tersedia di GitHub.com di bawah Lisensi MIT yang permisif, sedangkan kode biner adalah perangkat lunak bebas. Kode Visual Studio terpilih sebagai alat lingkungan pengembangan paling populer dalam Survei Pengembang Overflow 2019, dengan 50,7 persen dari 87.317 responden mengatakan mereka menggunakannya.

Visual Studio Code adalah editor kode sumber yang mendukung beberapa bahasa pemrograman yang berbeda, termasuk Java, JavaScript, Go, Node.js dan C++. Itu dibangun di atas kerangka Electron, yang digunakan untuk membuat aplikasi Web Node.js yang menggunakan mesin tata letak Blink. Kode Visual Studio menggunakan komponen editor yang sama (dijuluki "Monaco") seperti yang dilakukan Azure DevOps (sebelumnya Visual Studio Online dan Visual Studio Team Services).



Gambar 2.3 Microsoft Visual Studio Code

2.13.1 Sejarah Visual Studio Code

Visual Studio Code diumumkan pada 29 April 2015 oleh Microsoft pada konferensi Build 2015. Pratinjau Build dirilis tak lama kemudian. Pada 18 November 2015, Visual Studio Code dirilis di bawah Lisensi Expat dan kode sumbernya diserahkan ke GitHub. Dukungan lebih lanjut juga diumumkan. Pada 14 April 2016, Visual Studio Code lolos tahap pratinjau publik dan dirilis ke web, (Rahmel, 2018).

Bahasa pemrograman terstruktur mirip dengan bahasa Inggris yang biasa digunakan oleh programmer untuk menulis program komputer sederhana yang berfungsi sebagai pembelajaran untuk konsep dasar pemrograman komputer. Sejak itu, banyak versi BASIC telah dikembangkan untuk digunakan di berbagai platform komputer.

2.13.2 Fitur Visual Studio Code

Visual Studio Code memungkinkan pengguna untuk membuka satu atau lebih direktori, yang kemudian dapat disimpan di ruang kerja untuk digunakan kembali di masa mendatang. Ini memungkinkan untuk beroperasi sebagai editor kode agnostik bahasa untuk bahasa apa pun. Ini mendukung beberapa bahasa pemrograman dan set fitur yang berbeda untuk setiap bahasa. File dan folder yang tidak diinginkan dapat dikecualikan dari pohon proyek melalui pengaturan. Banyak fitur *Visual Studio Code* tidak diungkapkan melalui menu atau antarmuka pengguna, tetapi dapat diakses melalui palet perintah.

Visual Studio Code dapat diperpanjang melalui ekstensi, tersedia melalui repositori pusat. Ini termasuk penambahan editor dan dukungan bahasa. Fitur penting adalah kemampuan untuk membuat ekstensi yang menambahkan dukungan untuk bahasa, tema, dan debugger baru, melakukan analisis kode statis, dan menambahkan linter kode menggunakan Protokol Server Bahasa. *Visual Studio Code* menyertakan beberapa ekstensi untuk FTP, memungkinkan perangkat lunak untuk digunakan sebagai alternatif gratis untuk pengembangan web. Kode dapat disinkronkan antara editor dan server, tanpa mengunduh perangkat lunak tambahan. *Visual Studio Code* memungkinkan pengguna untuk mengatur halaman kode tempat dokumen aktif disimpan, karakter baris baru, dan bahasa pemrograman dokumen aktif. Ini memungkinkannya untuk digunakan pada platform apa pun, di lokasi mana pun, dan untuk bahasa pemrograman apa pun.

2.14 Riset Terkait

Berikut ini adalah beberapa riset terkait dalam membangun sistem yang dirancang.

Tabel 2.2 Riset terkait

Penulis	Judul	Keterangan
(Ibrahim & Gustina, 2020)	Rancang Bangun Aplikasi Berbasis Android Untuk Brand Clothing Sand Beach Dengan Skema Diskon Menggunakan Hungarian Algorithm	Aplikasi berbasis android ini berisikan informasi mengenai produk yang dikemas rapih dan menarik guna mempermudah pembeli untuk melihat jelas deskripsi , model, hingga harga dari produk yang ingin dibeli. Aplikasi Brand Clothing Sand Beach berbasis android ini juga memiliki akun admin yang dapat mengakses serta mengelola pesanan dari para pembeli dengan tertata dan sistematis

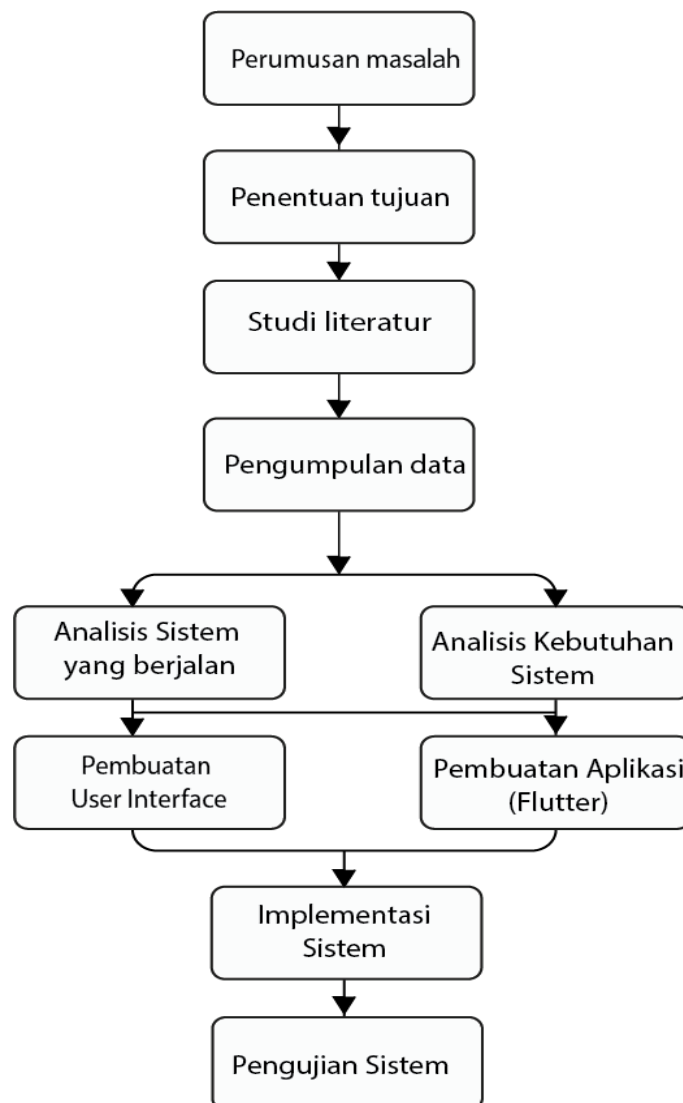
(Enggar Krisnada & Tanone, 2020)	Aplikasi Penjualan Tiket Kelas Pelatihan Berbasis Mobile menggunakan Flutter	Aplikasi pincher ini memberikan informasi dan mendistribusikan kelas pelatihan dengan menjual tiket kelas. Dalam perkembangannya sebagai platform online, Pincher masih menggunakan website sebagai sarana penyebaran kelas pelatihan yang dimiliki oleh Pincher, sehingga alat ini kurang efektif dan efisien bagi pengguna karena website Pincher terlalu rumit untuk diakses menggunakan ponsel atau smartphone. Oleh karena itu, untuk membantu pengguna mengakses aplikasi Pincher menggunakan ponsel mereka, aplikasi Pincher dibuat dalam bentuk sederhana yang diterapkan pada aplikasi seluler
(Ramadhan et al., 2020)	Monitoring Kualitas Air Tambak Udang Menggunakan NodeMCU, Firebase, dan Flutter	Tujuan dari penelitian ini adalah mengembangkan sistem monitoring kualitas air tambak udang khususnya udang vaname menggunakan perangkat berbasis Internet of Things (IoT) dengan metode data logging. Peran sistem adalah untuk memantau kondisi air, merekam data sensor, dan memberikan status kualitas air tambak berdasarkan pergerakan air, kekeruhan air, dan suhu air. Perangkat data logger ini menggunakan mikrokontroler bernama NodeMCU ESP8266. Hasil pemantauan kualitas air dapat diakses melalui aplikasi pemantauan berbasis Android yang dibangun menggunakan framework Flutter yang berisi informasi.

BAB III

METODE PENELITIAN

3.1 Tahapan Penelitian

Penelitian dalam membangun sistem informasi majelis zikir darul khalik memiliki beberapa tahapan yang digambarkan pada gambar 3.1



Gambar 3.1 Tahapan Penelitian

Gambar 3.1 menjelaskan langkah yang perlu diambil dalam menyelesaikan rancangan sistem. Tahapan penelitian diatur agar penelitian dan hasil penelitian memberikan kontribusi yang tepat. Hal ini bertujuan agar tidak terjadi kekeliruan dalam melaksanakan penelitian tersebut. Tahapan berikut adalah prosedur yang diambil dalam melakukan penelitian ini:

1. Rumusan masalah ditetapkan sebagai penentu permasalahan yang akan diselesaikan menggunakan sistem informasi.
2. Penentuan tujuan merupakan kegiatan dan arah dari penelitian yang dilakukan.
3. Studi literatur merupakan pencarian referensi yang berhubungan dengan sistem informasi. Kelayakan studi didapat dari jurnal, internet dan buku yang berhubungan dengan metode yang digunakan.
4. Pengumpulan data dilakukan dengan mendapatkan data-data yang digunakan di MZ darul Khalik.
5. Analisa dilakukan untuk menentukan teknik penyelesaian suatu rumusan masalah dan membandingkan dengan cara sebelumnya. Perancangan dilakukan berdasarkan kriteria yang dalam pencatatan data jemaah.
6. Perancangan sistem bertujuan untuk menentukan model yang digunakan dalam pembuatan aplikasi.
7. Pembuatan Aplikasi bertujuan untuk membuat sistem yang dapat berkomunikasi dengan pengguna.
8. Pembahasan berfungsi untuk mengkaji hasil yang telah dihasilkan. Pembahasan dilakukan untuk memberikan sinkronisasi.

9. Pengujian Sistem merupakan pengujian Aplikasi yang telah dibuat menggunakan Flutter dalam menentukan kelebihan dan kekurangan sistem.

3.2 Analisis Kebutuhan Perangkat Lunak

Pada tahapan ini ditentukan perangkat lunak yang akan digunakan dalam membangun dan mengimplementasikan aplikasi, kebutuhan perangkat lunak sebagai berikut :

- a. Sistem Operasi Windows 10/Mac OSX Catalina
- b. Flutter Stable SDK (Software Development Kit)
- c. Android SDK
- d. iOS SDK
- e. Visual Studio Code
- f. Postman

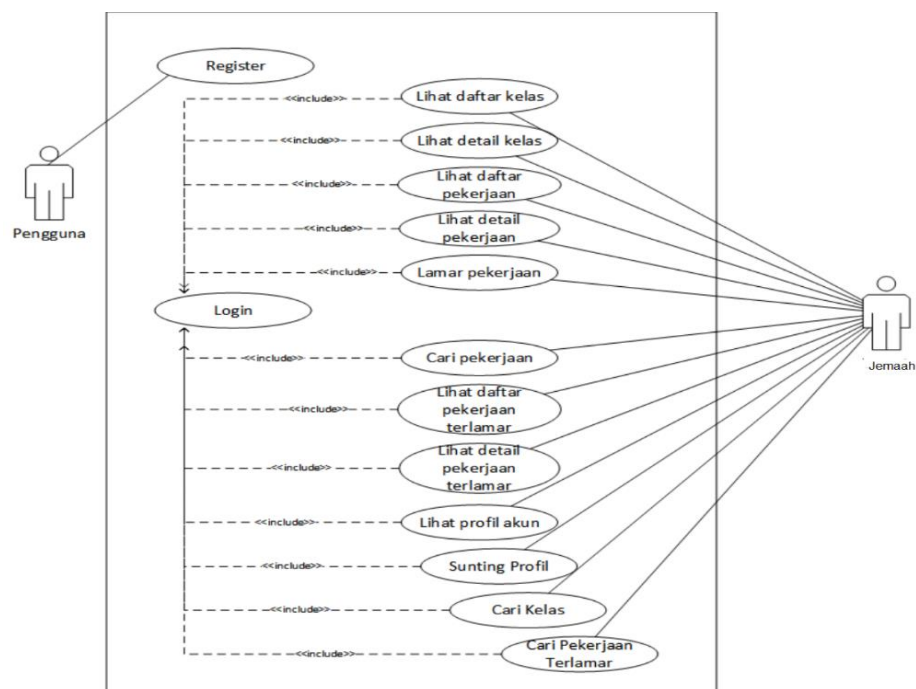
3.3 Analisis Kebutuhan Perangkat Keras

Pada tahapan ini ditentukan spesifikasi minimum perangkat keras yang dibutuhkan untuk menjalankan aplikasi, sebagai berikut:

- a. Android Device dengan spesifikasi minimum:
 - Memori : 1GB RAM
 - Versi OS : Lollipop 5.1
- b. iOS Device dengan spesifikasi minimum :
 - Memori : 512 MB
 - Versi OS : iOS 12.4.1

3.3.1 Use Case Diagram

Use case diagram memberikan gambaran deskripsi dari fungsi sistem melalui sudut pandang pengguna dan membantu analisis sistem memahami cara kerja sistem informasi yang dibangun. Usecase diagram menggambarkan interaksi antara aktor dan system. Dari hasil analisis, terdapat dua actor yang berinteraksi dengan aplikasi, yaitu jemaah dan Pengunjung, Pengunjung hanya dapat melakukan registrasi pada aplikasi tanpa bisa mengakses fitur lain. jemaah merupakan pengguna yang sudah melakukan registrasi sebelumnya dan terverifikasi dapat melakukan login akun pada aplikasi. Setelah login, jemaah dapat mencari kelas, melihat, mencari Informasi, melihat daftar apasaja detail informasi, melihat profil dan menyuntingnya. Usecase diagram dari aplikasi yang akan dibuat adalah seperti pada Gambar 3.1

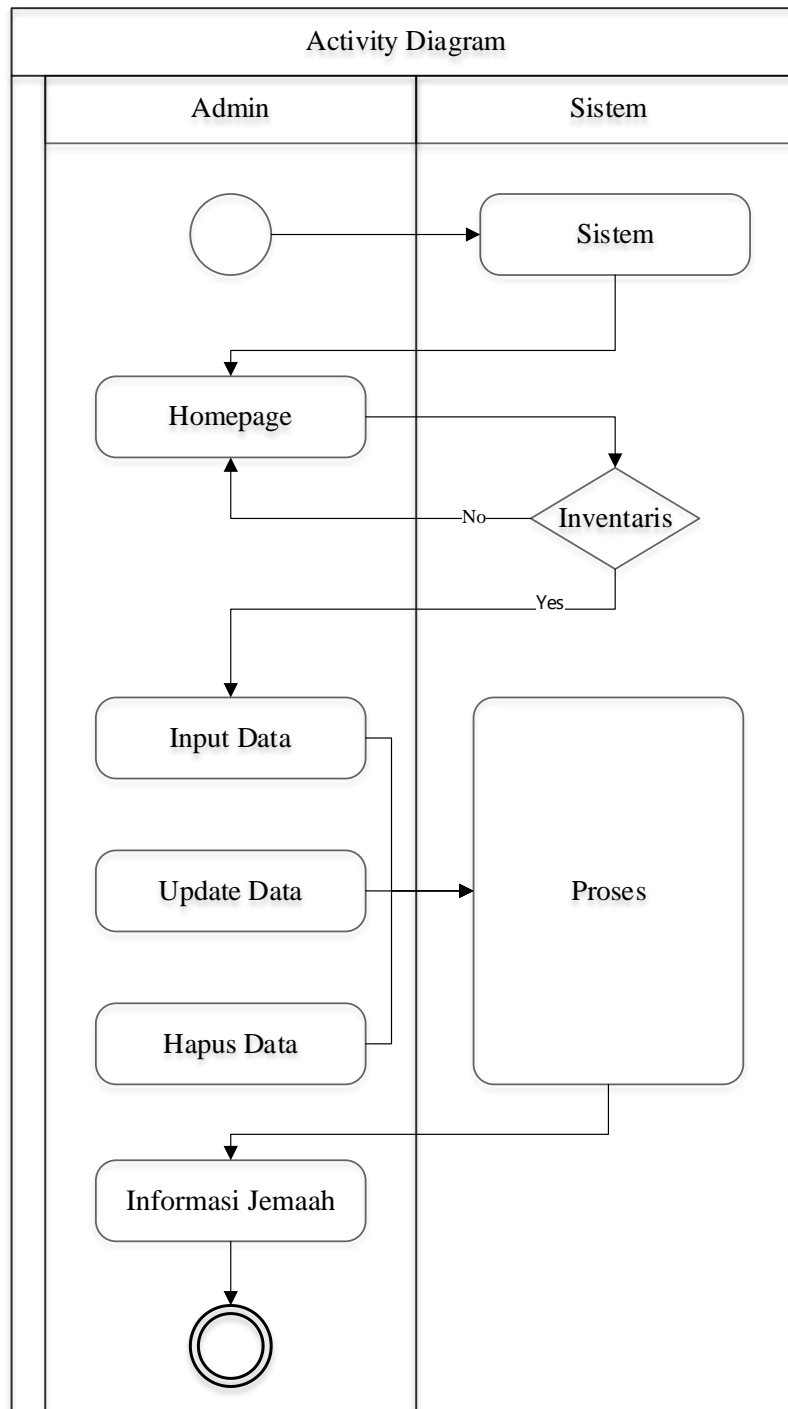


Gambar 3.2 Use case diagram pencatatan jemaah

Gambar 3.2 menjelaskan bahwa pengguna dan Jemaah merupakan user memiliki fungsi berbeda. Pengguna memiliki akun yang bersifat user. Jemaah dapat melakukan input data pembaharuan data dan juga dapat menghapus data profile.

3.3.2 Activity Diagram

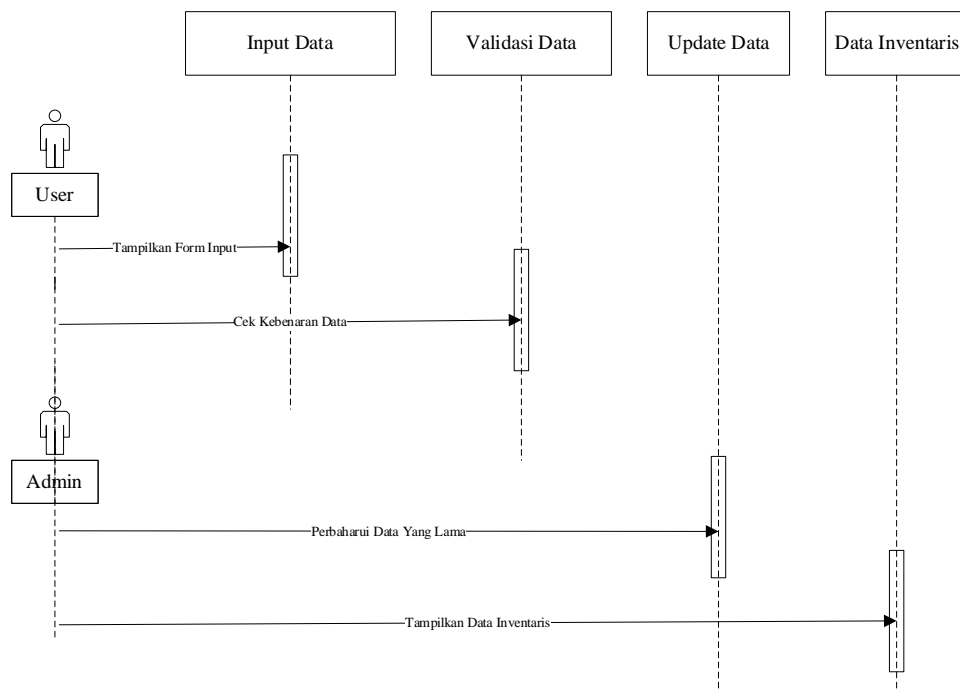
Activity Diagram menjelaskan arah atau aktifitas yang dilakukan oleh pengguna terhadap sistem informasi pencatatan data jemaah tersebut. Diagram ini menggambarkan alur kerja urutan aktifitas yang terjadi pada sistem. Diagram ini menunjukkan langka-langkah proses kerja yang terjadi pada sistem dari awal hingga akhir. Gambar 3.3



Gambar 3.3 Activity diagram pencatatan Jemaah

3.3.3 Sequence Diagram

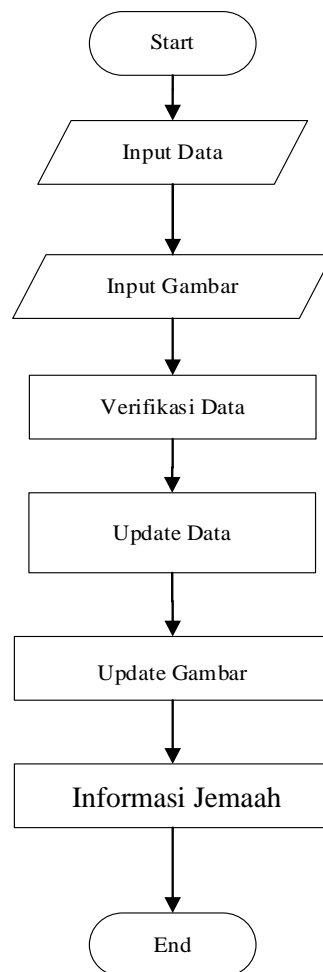
Sequence diagram merupakan urutan pengerjaan yang dilakukan pengguna dalam melaksanakan pencatatan inventaris. Gambar 3.4 adalah *sequence diagram* yang digunakan pada penelitian ini.



Gambar 3.4 *Sequence diagram* pencatatan inventaris

3.3.4 Flowchart

Flowchart menggambarkan alur dari pencatatan data informasi Jamaah dalam sistem informasi yang akan dibangun di penelitian ini. Gambar 3.5 adalah *flowchart* pencatatan data informasi Jamaah.



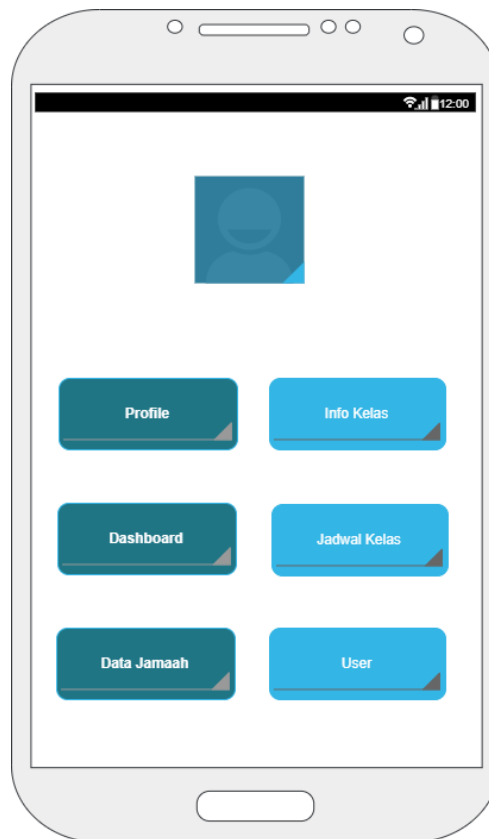
Gambar 3.5 Flowchart pencatatan inventaris

3.4 Perancangan Antarmuka Program

Tampilan antarmuka program harus dirancang dengan baik dan benar agar pengguna dapat menggunakan Aplikasi tanpa mengalami kendala. Kesalahan informasi yang ditampilkan merupakan kesalahan fatal yang terjadi dalam membangun suatu sistem informasi. Semakin *user friendly* suatu Aplikasi maka semakin mudah digunakan oleh orang awam. Aplikasi yang akan dibangun ditujukan kepada Seluruh jamaah yayasan abdul khalik Fajduani.

3.4.1 Rancangan Menu Dashboard

Rancangan menu dashboard merupakan halaman *homepage* dalam *website* sistem informasi pencatatan data inventaris. Gambar 3.6 merupakan hasil perancangan menu beranda.



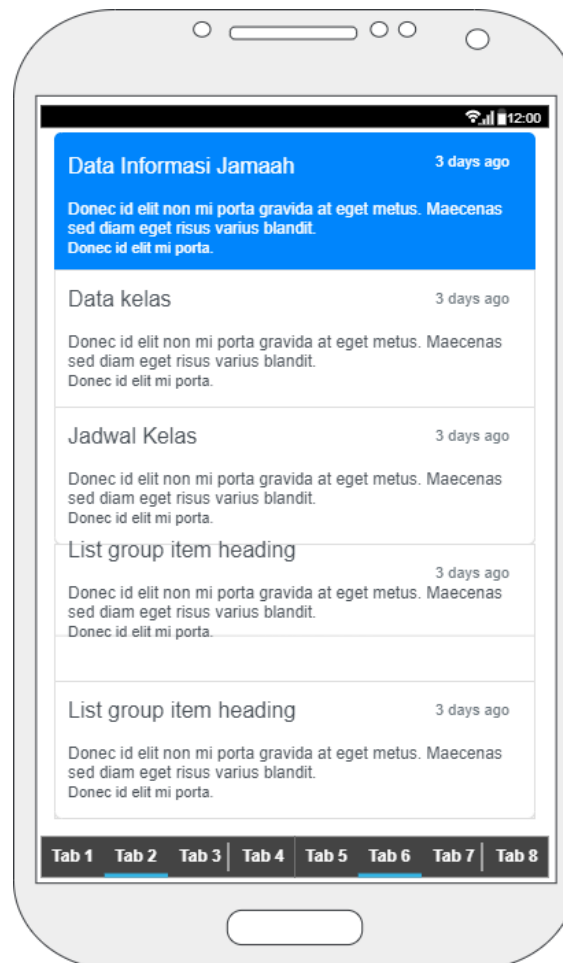
Gambar 3.6 Rancangan menu dashboard

Menu beranda memiliki beberapa menu lainnya, antara lain:

1. Profile
2. Informasi Kelas
3. Jadwal Kelas
4. Data Jamaah
5. Login dan Logout
6. Kontak Kami

3.4.2 Rancangan Menu Jadwal Kelas

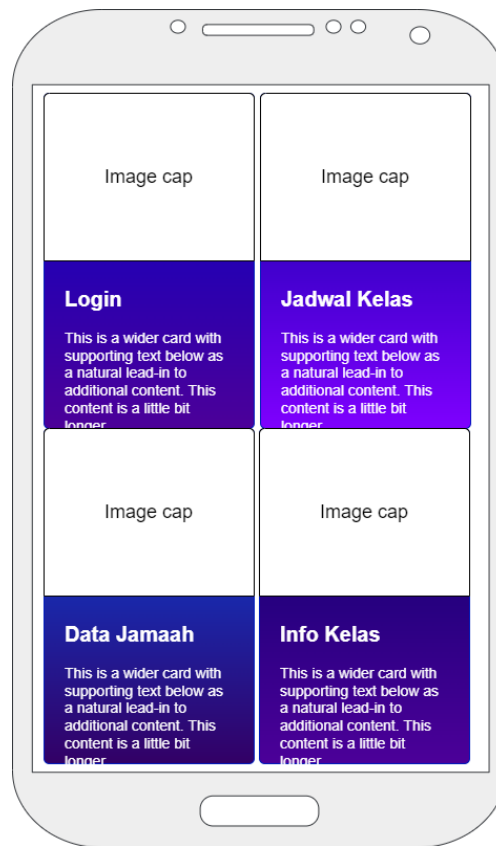
Rancangan menu ini dibangun untuk menampilkan jadwal tentang kegiatan di Majelis Zikir Darul Khalik. Gambar 3.7 merupakan hasil perancangan menu galeri.



Gambar 3.7 Rancangan menu Jadwal Kelas

3.4.3 Rancangan Menu Informasi

Rancangan menu inventaris berfungsi untuk mencatat data Informasi Jamaah dan dan informasi kelas. Gambar 3.8 adalah rancangan menu aplikasi.



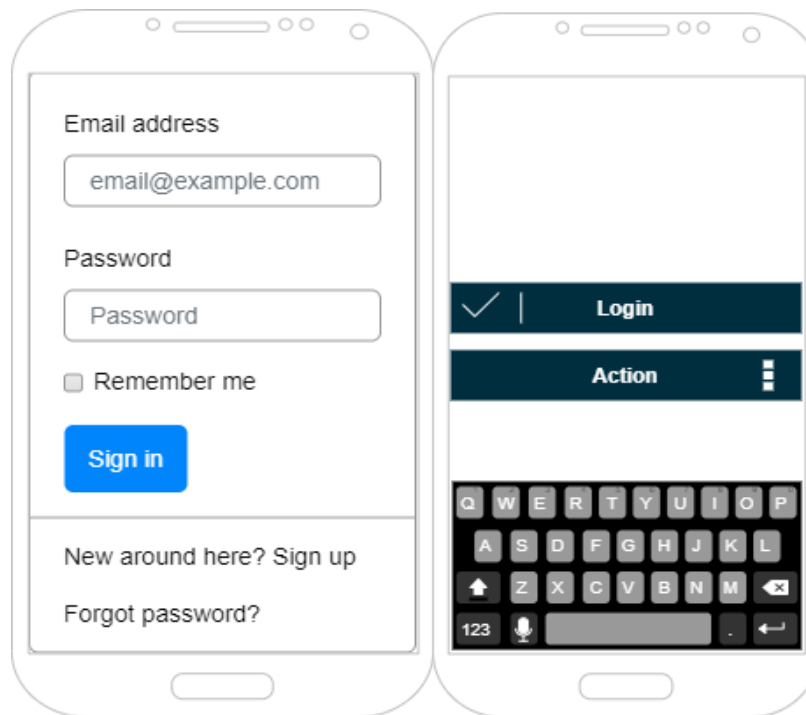
Gambar 3.8 Rancangan Menu Informasi

Menu inventaris terdiri dari beberapa bagian antara lain:

1. Pencatatan Jamaah
2. Pencatatan Informasi Kelas
3. Jadwal Kelas
4. Keterangan

3.4.4 Rancangan Menu *Registrasi* dan *Login*

Menu *login* merupakan tampilan yang memberikan akses masuk kepada pengguna terotorisasi. Gambar 3.9 adalah hasil perancangan dari menu *login*.



Gambar 3.9 Rancangan menu registrasi dan login

3.4.5 Rancangan Menu Kontak Kami

Menu info merupakan tampilan yang memberikan informasi sekilas tentang kontak yang dapat di hubungi.



Gambar 3.10 Rancangan menu kontak kami

BAB IV

HASIL DAN PEMBAHASAN

Implementasi adalah tahapan dalam membangun sistem informasi pencatatan data Jamaah di Majelis Zikir Darul khalik di Jalan Tj Selamat kec. Sunggal Kabupaten Deli Serdang Sumut 20351, Bagian ini menjelaskan beragam implementasi dan *interface* dari komponen yang berperan serta pada penelitian ini yaitu implementasi sistem informasi pencatatan data Jamaah. Penelitian menggunakan beberapa model tampilan dalam menghubungkan pengguna dan sistem. Penjelasan sistem akan dijelaskan pada bagian berikut ini.

4.1 Spesifikasi Sistem

Spesifikasi sistem penting dalam memastikan suatu sistem berjalan dengan baik dan lancar. Hal ini guna mendukung kearaban jamaah dari *Aplikasi* yang akan dibangun untuk melakukan pencatatan data informasi jamaah. Diperlukan persyaratan sistem atau program, yang akan memungkinkan pembuatan aplikasi dengan flutter yang mendukung penelitian ini agar penelitian ini dapat berjalan dengan baik dan lancar. Sistem pada aplikasi terdiri dari dua bagian yaitu perangkat keras dan perangkat lunak. Saat membuat aplikasi, kedua sistem ini bekerja sama dan juga terikat satu sama lain dalam menunjang kelancaran aplikasi yang dibuat. Hal ini juga akan mempermudah pengguna dalam berinteraksi dengan sistem. Adapun spesifikasi perangkat keras dan lunak dalam uji coba aplikasi akan dijelaskan berikutnya.

4.1.1 Spesifikasi Perangkat Keras

Aplikasi yang dibangun memerlukan perangkat keras atau *hardware* dalam menjalankan sistem operasi dan perangkat lunak lainnya. Komputer dan komponen harus saling mendukung kemampuan. Tabel 4.1 adalah spesifikasi perangkat keras yang digunakan dalam penelitian.

Tabel 4.1 Spesifikasi Perangkat Keras

<i>No.</i>	<i>Nama Komponen</i>	<i>Spesifikasi</i>
1	<i>Processor</i>	Intel Core i5 2.5 GHz
2	RAM	8 GB
3	<i>Harddisk</i>	500 GB
4	Monitor	14 inches

4.1.2 Spesifikasi Perangkat Lunak

Perangkat lunak atau *software* merupakan antarmuka yang akan melaksanakan tugas. Tabel 4.2 adalah spesifikasi perangkat lunak yang digunakan dalam penelitian.

Tabel 4.2 Spesifikasi Perangkat Lunak

<i>No.</i>	<i>Nama Komponen</i>	<i>Spesifikasi</i>
1	Sistem Operasi	Windows 10 64bit
2	IDE Pemrograman	Microsoft Visual Studio Code
3	Desainer UI	MockFlow
4	Data Editor	Microsoft Excel 2019
5	Word Processing	Microsoft Word 2019

4.2 Hasil Aplikasi Android

Hasil aplikasi Android dibangun berdasarkan desain yang telah dirancang sebelumnya. Ada beberapa tampilan yang ada dalam Android. Berikut ini akan dijelaskan lebih lanjut untuk masing-masing tampilan halaman yang digunakan dalam penelitian ini.

4.2.1 Halaman Menu Dashboard

Halaman menu *dashboard* merupakan *homepage* pada sistem aplikasi Android pencatatan data Jamaah pada Majelis Zikir Darul khalik. Gambar 4.1 merupakan hasil tampilan menu beranda.



Gambar 4.1 Halaman menu beranda

4.2.2 Halaman Menu Jadwal Kelas

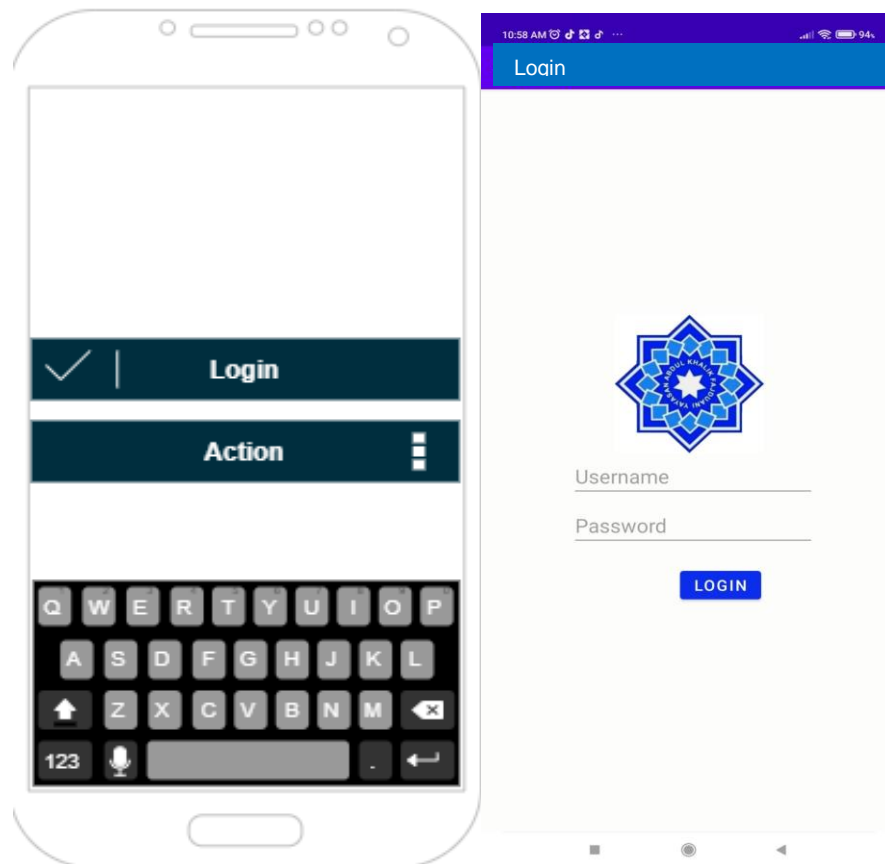
Halaman menu galeri menampilkan foto-foto kegiatan yang terjadi di area Majelis Zikir Darul khalik. Gambar 4.2 adalah tampilan dari halaman menu Jadwal kelas.



Gambar 4.2 Halaman menu Jadwal kelas

4.2.3 Halaman Menu Login

Halaman menu login berfungsi untuk melakukan akses kepada admin yang telah terdaftar di sistem informasi. Gambar 4.3 merupakan tampilan dari halaman menu login.



Gambar 4.3 Halaman Menu Login

4.2.4 Halaman Menu Kontak Kami

Halaman menu kontak kami adalah halaman petunjuk bagi pengguna yang ingin mengetahui lokasi Majelis Zikir Darul khalik. Di halaman ini tertera juga alamat lengkap. Gambar 4.4 merupakan tampilan dari halaman menu kontak kami.



Gambar 4.4 Halaman menu kontak kami

4.2.5 Halaman Menu Informasi

Ada beberapa bagian yang berada di halaman ini. Halaman ini menunjukkan data-data inventaris lengkap yang dimiliki oleh Majelis Zikir Darul khalik. Menu-menu yang terdapat di dalam halaman ini meliputi, penambahan, pengubahan dan

penghapusan data. Setiap data informasi memiliki foto untuk mengetahui bentuk deskripsi informasi tersebut. Gambar 4.6 adalah halaman menu informasi.



Gambar 4.5 Halaman menu informasi

BAB V

PENUTUP

5.1 Kesimpulan

Setelah melakukan penelitian di Majelis Zikir Darul khalik, ada beberapa kesimpulan yang diperoleh, antara lain:

1. Sistem informasi pencatatan data Jamaah dilakukan dengan menggunakan aplikasi berbentuk android menggunakan Flutter.
2. Informasi yang dijadikan sampel untuk data percobaan diambil dari jadwal kelas jadwal kegiatan di Majelis Zikir Darul khalik.
3. Data pencatatan jamaah semua disimpan dalam database yang dapat dimodifikasi apabila terjadi kesalahan.

5.2 Saran

Penelitian ini memiliki saran yang sebaiknya dikembangkan oleh penelitian berikutnya. Berikut ini adalah beberapa saran tersebut, antara lain:

1. Hendaknya menambah data informasi lain selain jadwal kegiatan agar sistem informasi memiliki data yang lebih lengkap.
2. Hendaknya aplikasi dikembangkan agar memiliki versi yang berbasis Android dan ios yang lebih lengkap.

DAFTAR PUSTAKA

- Astuti, P. D. (2017). *Sistem Informasi Penjualan Obat Pada Apotek Jati Farma Arjosari. Speed-sentra penelitian engineering dan edukasi*. 3(4), 34–39.
- Berisha-Shaqiri, A. (2015). Management Information System and Competitive Advantage. *Mediterranean Journal of Social Sciences*. <https://doi.org/10.5901/mjss.2015.v6n1p204>
- Enggar Krisnada, F., & Tanone, R. (2020). Aplikasi Penjualan Tiket Kelas Pelatihan Berbasis Mobile menggunakan Flutter. *Jurnal Teknik Informatika Dan Sistem Informasi*, 5(3), 281–295. <https://doi.org/10.28932/jutisi.v5i3.1865>
- Fallo, F. C. (2017). Perancangan Prototype Sistem Otomasi Berbasis Mikrokontroler Untuk Lampu Penerangan Ruangan Di UAJY. (*Doctoral Dissertation, UAJY*).
- Hung, N. V., van Hung, P., & Anh, B. T. (2018). Database Design For E-Governance Applications: A Framework For The Management Information Systems Of The Vietnam Commitee For Ethnic Minority Affairs (CEMA). *International Journal of Civil Service Reform and Practice*, 3(1).
- Ibrahim, K. A. B., & Gustina, D. (2020). Rancang Bangun Aplikasi Berbasis Android Untuk Brand Clothing Sand Beach Dengan Skema Diskon Menggunakan Hungarian Algorithm. *Jurnal Teknologi Informasi Dan Komunikasi, Media History of Documents*, 47–56. Universitas Persada Indonesia Y.A.I
- Jogiyanto, H. M. (2016). *Analisis Dan Desain Sistem Informasi, Pendekatan Terstruktur Teori Dan Praktek Aplikasi Bisnis*. Andi Offset.
- Kurniawan, T. A. (2018). Pemodelan Use Case (UML): Evaluasi Terhadap beberapa Kesalahan dalam Praktik. *Jurnal Teknologi Informasi Dan Ilmu Komputer*, 5(1), 77. <https://doi.org/10.25126/jtiik.201851610>
- Ladjamudin, A.-B. bin. (2017). *Analisis dan Desain Sistem Informasi*. Graha Ilmu.
- Mangematin, V., & Baden-Fuller, C. (2008). Global Contests in the Production of Business Knowledge. *Long Range Planning*, 41(1), 117–139. <https://doi.org/10.1016/j.lrp.2007.11.005>
- Pratama, E. B., & Hendini, A. (2019). Sistem Pakar Diagnosa Penyakit Kulit Pada Kucing Berbasis Web Menggunakan Metode Decision Tree. *Jurnal Sistem Informasi*, 8(2), 254–264.

- Pratama, A. N. W. (2010). *Codelgniter: Cara Mudah Membangun Aplikasi PHP*.
- Rahmel, D. (2018). *Visual Basic.NET*. McGraw-Hill.
- Ramadhan, H. P., Kartiko, C., & Prasetiadi, A. (2020). Monitoring Kualitas Air Tambak Udang Menggunakan NodeMCU, Firebase, dan Flutter. *Jurnal Teknik Informatika Dan Sistem Informasi*, 6(1), 102–114. <https://doi.org/10.28932/jutisi.v6i1.2365>
- Sukmawati, R., & Priyadi, Y. (2019). Perancangan Proses Bisnis Menggunakan UML Berdasarkan Fit/Gap Analysis Pada Modul Inventory Odoo. *INTENSIF: Jurnal Ilmiah Penelitian Dan Penerapan Teknologi Sistem Informasi*, 3(2), 104. <https://doi.org/10.29407/intensif.v3i2.12697>
- Caicedo, J. C., & Lazebnik, S. (2015). Active object localization with deep reinforcement learning. In Proceedings of the IEEE international conference on computer vision (pp. 2488-2496).