

**NGAJI ASIK!: APLIKASI PEMBELAJARAN  
AL-QURAN SECARA PRIVAT  
(STUDI KASUS: MAJELIS QURAN MESJID  
SYAMSUL ULUM)**

**NGAJI ASIK!: PRIVATELY AL-QURAN  
LEARNING APPLICATION  
(STUDY CASE: MAJELIS QURAN SYAMSUL  
ULUM MOSQUE)**

**PROYEK AKHIR**

**Muh Alif Al Gibran Arif  
6701164163**



**PROGRAM STUDI D3 SISTEM INFORMASI  
FAKULTAS ILMU TERAPAN  
UNIVERSITAS TELKOM  
BANDUNG, 2018**

*Untuk Semua yang memegang teguh prinsip integritas*

**LEMBAR PENGESAHAN PROYEK AKHIR  
APLIKASI PEMESANAN DAN PENGIRIMAN MAKANAN STUDI  
KASUS UNIVERSITAS TELKOM**

Penulis

Muh Alif Al Gibran Arif

NIM 6701164163

---

Pembimbing I

Siska Komala Sari, S.T., M.T.

NIP 07810044

---

Pembimbing II

Agus Pratondo, Ph.D.

NIP 09770043

---

Ketua Program Studi

Wawa Wikusna, S.T., M.Kom.

NIP 14740031

---

## PERNYATAAN

---

Dengan ini saya menyatakan bahwa:

1. Proyek Akhir ini adalah asli dan belum pernah diajukan untuk mendapatkan gelar akademik (Ahli Madya, Sarjana, Magister dan Doktor), baik di Fakultas Ilmu Terapan Universitas Telkom maupun di perguruan tinggi lainnya;
2. karya tulis ini murni gagasan, rumusan, dan penelitian saya sendiri, tanpa bantuan pihak lain, kecuali arahan tim pembimbing atau tim promotor atau penguji;
3. dalam karya tulis ini tidak terdapat cuplikan karya atau pendapat yang telah ditulis atau dipublikasikan orang lain, kecuali secara tertulis dengan jelas dicantumkan sebagai acuan dalam naskah dengan menyebutkan nama pengarang dan dicantumkan dalam daftar pustaka;
4. saya mengizinkan karya tulis ini dipublikasikan oleh Fakultas Ilmu Terapan Universitas Telkom, dengan tetap mencantumkan saya sebagai penulis; dan

Pernyataan ini saya buat dengan sesungguhnya dan apabila pada kemudian hari terdapat penyimpangan dan ketidakbenaran dalam pernyataan ini maka saya bersedia menerima sanksi akademik berupa pencabutan gelar yang telah diperoleh karena karya tulis ini, serta sanksi lainnya sesuai norma yang berlaku di Fakultas Ilmu Terapan Universitas Telkom.

Bandung, 21 Desember 2018

Pembuat pernyataan,

Muh Alif Al Gibran

## KATA PENGANTAR

---

Alhamdulillahirabbil'alamin. Puji syukur kehadiran ALLAH SWT, atas rahmat dan hidayah-Nya sehingga penulis dapat menyelesaikan tugas akhir dengan judul: "NGAJI ASIK!: APLIKASI PEMBELAJARAN AL-QURAN SECARA PRIVAT" Proyek akhir ini disusun sebagai syarat dalam menyelesaikan pendidikan Diploma-III Sistem Informasi pada Fakultas Ilmu Terapan Universitas Telkom.

Pada kesempatan ini Penulis ingin menyampaikan rasa syukur dan terima kasih yang sebesar-besarnya kepada seluruh pihak yang telah membantu Penulis dalam menyelesaikan proyek akhir ini. Penulis sadar bahwa semua ini tidak akan tercapai tanpa adanya doa, dukungan, dorongan, serta bimbingan dari semua pihak. Oleh karena itu, dengan ketulusan Penulis ingin mengucapkan rasa syukur dan terima kasih kepada:

1. Allah SWT, Segala puji syukur kepada-Mu ya Allah atas segala nikmat, rahmat, karunia, berkah, hidayah dan izin yang telah Engkau berikan kepada hamba-Mu ini sehingga bisa menyelesaikan tugas akhir ini beserta Rasul-Nya Muhammad SAW yang telah menyiarkan agama Islam, agama yang membawa kedamaian, sehingga Penulis dapat merasakan nikmat iman dan islam.
2. Ayah Arif dan Ibu Hatifah yang tidak pernah lelah mendoakan Penulis, memberikan dorongan semangat, nasehat, cinta kasih dan ketulusan serta kerja kerasnya yang tak kenal lelah guna merawat dan membiayai Penulis sehingga Penulis dapat menyelesaikan tugas akhir ini.
3. Adik Penulis Maipha Deapati Arif, Zahra Ramadhanty dan Muh Ilham Hafidz yang selalu menghibur dan memberikan semangat kepada Penulis.
4. Ibu Siska Komala Sari, S.T., M.T., selaku pembimbing I yang selalu memberikan masukan dan memberikan bimbingan, arahan serta ilmu dalam menyelesaikan tugas akhir ini.

5. Bapak Agus Pratondo, Ph.D., selaku pembimbing II yang telah sabar membimbing Penulis, memberikan arahan serta motivasi kepada Penulis.
6. Seluruh Dosen Universitas Telkom yang tidak dapat Penulis sebutkan satu persatu, terimakasih atas semua ilmu yang telah diberikan.
7. Teman-teman D3 Manajemen Informatika seperjuangan, terimakasih atas segala waktu, bantuan, dukungan untuk Penulis.
8. Seluruh pihak yang telah membantu menyelesaikan proyek akhir ini yang tidak bisa Penulis sebutkan satu persatu. Semoga kebaikan semua pihak baik langsung maupun tidak mendapatkan amal yang berlipat ganda dari Allah SWT

Bandung, 21 Desember 2018

Penulis

## ABSTRAK

---

Al-Quran merupakan pedoman hidup bagi umat muslim, maka dari itu sudah sepatutnya umat muslim mempelajari lebih dalam mengenai Al-Quran. Dewasa ini, ketertarikan manusia akan Al-Quran semakin menurun dilihat dari data Rumah Quran Universitas Islam Negeri Sunan Gunung Djati Bandung, sekitar 54% umat muslim di Indonesia belum lancar membaca Al-Quran. Data tersebut menunjukkan minat membaca Al-Quran rendah. Berdasarkan data kuesioner dengan 93 responden, 67.7% mahasiswa Universitas Telkom tidak fasih dalam membaca Al-Quran. Maka dari itu, salah satu solusi adalah dibuatlah aplikasi “Ngaji Asik!: Aplikasi Pembelajaran Al-Quran Secara Privat”. Aplikasi ini dibuat untuk memudahkan para santri untuk belajar kapan saja dan dimana saja dibantu dengan tenaga pengajar secara privat. Aplikasi ini memungkinkan santri untuk mengatur jadwal dan tempat belajar sendiri. Dari sisi bisnis pengajar mendapatkan bayaran atas jasa dengan parameter tertentu. Target user utama adalah remaja yang cenderung tidak memiliki waktu luang dalam mempelajari Al-Quran tetapi masih memiliki keinginan belajar. Metode pengembangan aplikasi menggunakan metode *waterfall* dari Roger S. Pressmann. Metode analisis menggunakan metode kuantitatif dengan survey dan kualitatif dengan wawancara. Aplikasi dibangun diatas bahasa pemrograman Java.

Kata Kunci: Android, Al-Quran, Belajar

## ABSTRACT

---

*Al-Quran is a living guide for Muslims, therefore Muslims should learn more about the Al-Quran. Nowadays, human interest in the Al-Quran is decreasing. from the House Quran data of the Sunan Gunung Djati State Islamic University in Bandung, about 54% of Muslims in Indonesia have not able read the Al-Quran. These data show the low of interest to read Al-Quran .Based on questionnaire with 93 respondents, 67.7% of Telkom University students were not fluent in reading the Al-Quran. hence, there ara many solutions, one of them is develop an application called "Ngaji Asik!: Privately Al-Quran Learning Application " be made. This application is made to facilitate the students to study anytime and anywhere assisted by private teaching staff. This application allows students to arrange their own schedules and places of study. From the business side, teachers get paid for services with certain parameters. The main user target is teenagers who tend not to have free time in learning the Al-Quran but still have desires to learn to read the Al-Quran. Application development methods use the waterfall method from Roger S. Pressmann. The analytical method uses quantitative methods with surveys and qualitative interviews. Applications are built on the Java programming language.*

*Keywords: Android, Quran, Learn*



## DAFTAR ISI

---

KATA PENGANTAR.....	I
ABSTRAK .....	III
ABSTRACT .....	IV
DAFTAR ISI .....	V
DAFTAR GAMBAR.....	VIII
DAFTAR TABEL .....	IX
DAFTAR LAMPIRAN .....	X
<b>BAB 1 PENDAHULUAN.....</b>	<b>1</b>
1.1 Latar Belakang.....	1
1.2 Rumusan Masalah .....	2
1.3 Tujuan .....	2
1.4 Batasan Masalah .....	2
1.5 Metode Pengerjaan.....	3
1.6 Jadwal Pengerjaan .....	5
<b>BAB 2 TINJAUAN PUSTAKA .....</b>	<b>6</b>
2.1 Madrasah Quran Mesjid Syamsul Ulum .....	6
2.2 Al-Quran .....	6
2.3 Aplikasi .....	6
2.4 Website.....	6

<b>2.5</b>	<b>Android .....</b>	<b>7</b>
<b>2.6</b>	<b>Tools Pemodelan Aplikasi.....</b>	<b>7</b>
2.6.1	Business Process Model and Notation (BPMN) .....	7
2.6.2	Entity Relationship Diagram (ERD) .....	9
2.6.3	Unified Modeling Language (UML) .....	10
2.6.4	Use Case Diagram .....	11
2.6.5	Class Diagram .....	12
2.6.6	Activity Diagram .....	12
2.6.7	Package Diagram .....	13
2.6.8	Deployment Diagram .....	14
2.6.9	Balsamiq Mockups .....	15
2.6.10	yEd Graph Editor .....	15
2.6.11	StarUML .....	15
<b>2.7</b>	<b>Tools Pembangunan Aplikasi.....</b>	<b>15</b>
2.7.1	Kotlin .....	16
2.7.2	SQLite .....	16
2.7.3	Android Studio .....	16
2.7.4	Atom.....	16
2.7.5	X Apache-MySQL-PHP-Perl (XAMPP) .....	16
2.7.6	Hypertext Preprocessor (PHP) .....	17
2.7.7	Codeigniter .....	17
2.7.8	HTML, CSS dan JavaScript .....	17
<b>2.8</b>	<b>Tools Pengujian Aplikasi.....</b>	<b>17</b>
2.8.1	Blackbox Testing.....	18
2.8.2	User Acceptance Test (UAT).....	18
<b>BAB 3</b>	<b>ANALISIS DAN PERANCANGAN .....</b>	<b>20</b>
<b>3.1</b>	<b>Analisis.....</b>	<b>20</b>
3.1.1	Gambaran Sistem Saat Ini .....	20
3.1.2	Gambaran Sistem Usulan .....	29
3.1.3	Analisis Kebutuhan Sistem (atau Produk) .....	33
3.1.4	Kebutuhan Perangkat Keras dan Perangkat Lunak .....	38
<b>3.2</b>	<b>Perancangan .....</b>	<b>40</b>
3.2.1	Model Aplikasi Berbasis Objek .....	40
3.2.2	Perancangan Basis Data .....	53
3.2.3	Perancangan Antar Muka.....	58
<b>BAB 4</b>	<b>IMPLEMENTASI DAN PENGUJIAN .....</b>	<b>65</b>
<b>4.1</b>	<b>Implementasi .....</b>	<b>65</b>
4.1.1	Pemesanan Kelas .....	65

4.1.2	Tampilan Jenis Kelas.....	67
4.1.3	Kelola Akun .....	69
<b>4.2</b>	<b>Pengujian .....</b>	<b>70</b>
4.2.1	<i>Black box Testing</i> .....	70
BAB 5 KESIMPULAN DAN SARAN .....		75
DAFTAR PUSTAKA .....		76
LAMPIRAN.....		78

## DAFTAR GAMBAR

---

Gambar 3 . 1 Proses Bisnis Utama MQMSU .....	21
Gambar 3 . 2 Proses Bisnis Registrasi Mahasiswa.....	23
Gambar 3 . 3 Proses Bisnis Registrasi Calon Pengajar .....	25
Gambar 3 . 4 Usulan Proses Bisnis Utama MQMSU .....	30
Gambar 3 . 5 Usulan Proses Bisnis Registrasi Calon Pengajar .....	32
Gambar 3 . 6 Use Case Diagram Aplikasi .....	41
Gambar 3 . 7 Activity Diagram Login.....	42
Gambar 3 . 8 Activity Diagram Kelola Kelas .....	42
Gambar 3 . 9 Activity Diagram Pilih Jenis Pembayaran .....	43
Gambar 3 . 10 Activity Diagram Konfirmasi Kelas.....	43
Gambar 3 . 11 Activity Diagram Chat.....	44
Gambar 3 . 12 Activity Diagram Membuat Akun .....	44
Gambar 3 . 13 Activity Diagram Edit Akun.....	45
Gambar 3 . 14 Activity Diagram Ganti Password .....	45
Gambar 3 . 15 Activity Diagram Hapus Jadwal Ketersediaan .....	47
Gambar 3 . 16 Activity Diagram Melihat Poin.....	47
Gambar 3 . 17 Activity Diagram Lihat Jadwal Ketersediaan .....	48
Gambar 3 . 18 Activity Diagram Menerima Notifikasi .....	49
Gambar 3 . 19 Activity Diagram Melihat Daftar User .....	49
Gambar 3 . 20 Activity Diagram Lihat Daftar Keseluruhan Kelas.....	50
Gambar 3 . 21 Activity Diagram Input Nilai Calon Pengajar .....	50
Gambar 3 . 22 Class Diagram Aplikasi.....	51
Gambar 3 . 23 Package Diagram Aplikasi.....	52
Gambar 3 . 24 Deployment Diagram Aplikasi .....	52
Gambar 3 . 25 Struktur Entitiy Relationship Diagram.....	53
Gambar 4 . 1 Tampilan Daftar Kelas Tersedia.....	65
Gambar 4 . 2 Tampilan Menambahkan Event .....	66
Gambar 4 . 3 Tampilan Memesan Kelas .....	66
Gambar 4 . 4 Tampilan Jenis Kelas.....	67
Gambar 4 . 5 Tampilan Deskripsi Jenis Kelas .....	68
Gambar 4 . 6 Tampilan Daftar Pengajar.....	68
Gambar 4 . 7 Tampilan Login Menggunakan Akun Gmail.....	69
Gambar 4 . 8 Tampilan Login .....	69
Gambar 4 . 9 Tampilan Profil .....	70

## DAFTAR TABEL

Tabel 1 . 1 Jadwal Pengerjaan Aplikasi .....	5
Tabel 2 . 1 Notasi pada BPMN.....	7
Tabel 2 . 2 Notasi Umum pada ERD .....	10
Tabel 2 . 3 Notasi pada Use Case Diagram.....	11
Tabel 2 . 4 Class Diagram .....	12
Tabel 2 . 5 Notasi Pada Activity Diagram .....	13
Tabel 2 . 6 Notasi pada Package Diagram .....	14
Tabel 2 . 7 Notasi pada Deployment Diagram .....	14
Tabel 2 . 8 Contoh Format Black Box Testing.....	18
Tabel 2 . 9 Contoh Format User Acceptance Test.....	18
Tabel 3 . 1 Pembandingan Aplikasi Sejenis .....	26
Tabel 3 . 2 Kelemahan Sistem Berjalan dan Usulan Perbaikan.....	27
Tabel 3 . 3 Fungsionalitas Aplikasi .....	34
Tabel 3 . 4 Analisis Pengguna .....	38
Tabel 3 . 5 Kebutuhan Perangkat Keras Pengembangan Aplikasi.....	39
Tabel 3 . 6 Kebutuhan Perangkat Lunak Pengembangan Aplikasi .....	39
Tabel 3 . 7 Kebutuhan Perangkat Keras Implementasi Sistem .....	40
Tabel 3 . 8 Struktur Tabel Santri.....	56
Tabel 3 . 9 Struktur Tabel Kelas.....	56
Tabel 3 . 10 Struktur Tabel Poin.....	57
Tabel 3 . 11 Struktur Tabel Penilaian .....	57
Tabel 3 . 12 Struktur Tabel Pengajar .....	58
Tabel 3 . 13 Perancangan Antar Muka Aplikasi.....	59
Tabel 4 . 1 Pengujian Lihat Jadwal Kelas Tersedia .....	70
Tabel 4 . 2 Pengujian Fungsi Membuat <i>Event</i> .....	71
Tabel 4 . 3 Pengujian Melakukan Pemesanan Kelas .....	72
Tabel 4 . 4 Pengujian Menampilkan Penjelasan Jenis Kelas .....	72
Tabel 4 . 5 Pengujian Menampilkan Daftar Pengajar.....	73

## DAFTAR LAMPIRAN

---

Lampiran 1 Hasil Wawancara Pengurus MQMSU Ulum Universitas Telkom .....	78
Lampiran 2 Hasil Kuesioner.....	79
Lampiran 3 Struktur Organisasi MQMSU.....	81

## BAB 1

### PENDAHULUAN

---

#### 1.1 Latar Belakang

Al-Quran merupakan pedoman bagi setiap etape hidup manusia, sejak alam kandungan hingga kembali menghadap Allah *Subhanahu wata'ala*. Mengamalkannya adalah kemestian sedangkan abai akan petunjuknya hanya mengantar pada keterbelakangan [1]. Mengenal Al-Quran sejak dini merupakan langkah utama sebelum mempelajari pelajaran lainnya. Bagi setiap keluarga muslim, nilai-nilai Al-Quran harus ditanamkan dalam lingkungan keluarga maupun lingkungan sosial. Namun, pada saat ini minat membaca Al-Quran semakin menurun.

Berdasarkan survey menggunakan metode kuesioner, dari 93 responden 67.7% mahasiswa Universitas Telkom tidak fasih dalam membaca Al-Quran. Hal ini dikarenakan jadwal kuliah yang padat serta kesibukan organisasi. Faktor yang paling signifikan adalah kurang dan terbatasnya waktu serta tempat dalam belajar Al-Quran. Selain itu faktor malu dalam belajar Al-Quran dengan umur 19 tahun keatas juga mempengaruhi malasnya mahasiswa dalam belajar membaca Al-Quran. Jika hal ini terus dibiarkan terjadi maka minat membaca Al-Quran bisa menurun drastis.

Berdasarkan kasus tersebut, maka salah satu solusi yang ditawarkan adalah dibuatnya aplikasi belajar Al-Quran dimana santri dapat belajar secara privat bersama tenaga pengajar dengan jadwal yang dapat diatur oleh santri. Aplikasi ini berbasis android karena menyesuaikan dengan target *user* yaitu mahasiswa yang hampir semua telah memiliki ponsel pintar atau gawai. Kelebihan lainnya dari Android adalah mudahnya dibawa kemana saja sehingga *user* dapat mengakses aplikasi kapan dan dimanapun [2].

Dengan adanya aplikasi ini diharapkan para mahasiswa yang ingin belajar membaca Al-Quran dapat terfasilitasi agar proses belajar membaca Al-quran menjadi lebih mudah. Santri tidak perlu datang ke kelas pembelajaran Al-Quran melainkan memilih

tempat sesuai keinginan santri begitupun dengan pengajar akan mendapatkan bayaran atas jasa yang diberikan.

## 1.2 Rumusan Masalah

Berdasarkan latar belakang, maka rumusan masalah yang akan dibahas yaitu:

- a. Bagaimana memfasilitasi pengajar dan santri dalam melakukan proses belajar membaca Al-Quran?
- b. Bagaimana membantu santri untuk mengetahui informasi terkait pengajar yang berkompeten?
- c. Bagaimana membantu pengajar dan santri dalam membuat *schedule* proses belajar membaca Al-Quran?

## 1.3 Tujuan

Berdasarkan rumusan masalah yang telah dipaparkan, tujuan dari pembuatan aplikasi yaitu:

- a. Memfasilitasi pengajar dan santri dalam melakukan proses belajar membaca Al-Quran.
- b. Menyediakan informasi yang dibutuhkan terkait tenaga pengajar.
- c. Menyediakan layanan dalam membuat *schedule* proses belajar membaca Al-Quran.

## 1.4 Batasan Masalah

Batasan masalah dari aplikasi yang dibangun, diantaranya:

1. Aplikasi hanya dapat dijalankan pada *smartphone* Android versi minimum 5. Lollipop.
2. Terdapat 3 user yaitu, santri, pengajar, dan pengurus Madrasah Quran.
3. pengajar dapat menambahkan waktu luang agar memudahkan santri dalam memilih jadwal.
4. Pengajar dan santri mendaftarkan diri melalui aplikasi.
5. pengajar tidak dapat melakukan penghitungan hasil bayaran atas jasa pada aplikasi.



6. Pengurus MQ dapat mengakses aplikasi website untuk pemantauan kelas yang lebih rinci.
7. Hanya diberikan satu akun pada pengurus MQ, sehingga pengurus MQ tidak perlu dimasukan kedalam basis data.
8. Pembayaran dilakukan secara tatap muka atau transfer bank, tidak dalam aplikasi.
9. Perangkat yang digunakan harus terhubung dengan jaringan internet.
10. Pada pengerjaan ini hanya sampai implementasi, tidak melakukan tahap perawatan.

## 1.5 Metode Pengerjaan

Metode pembangunan aplikasi yang digunakan adalah *waterfall* dari Roger S. Presmann. Berikut tahapan dalam mengembangkan aplikasi:

### 1. Analisis

Pada tahap ini, pengumpulan data yang berkaitan akan dilakukan menggunakan teknik wawancara dan teknik survey. Teknik wawancara dengan narasumber salah satu pengurus Madrasah Quran Mesjid Syamsul Ulum. Survey dilakukan menggunakan metode kuesioner dengan target user adalah mahasiswa.

### 2. Desain

Pada tahap ini, setelah data terkumpul maka akan dilakukan desain sistem, mulai dari desain antarmuka, proses bisnis hingga perancangan *backend* dengan UML. Beberapa *tools* yang digunakan dalam mendesain yaitu Balsamiq Mockups, yEd Graph Editor dan StarUML.

### 3. Implementasi

Pada tahap ini sistem mulai dibangun. Pengkodean terhadap aplikasi dilakukan secara keseluruhan, mulai dari pengkodean antarmuka hingga pengkodean *backend* menggunakan *tools* Android Studio, Atom dan XAMPP.

#### 4. Integrasi dan tes

Setelah unit dikembangkan akan dilakukan integrasi sistem dan *testing* menggunakan *black box* serta UAT (*user acceptance test*).

#### 5. Operasional dan perawatan

Setelah melakukan testing dan mengevaluasi aplikasi maka aplikasi dapat digunakan oleh para pengguna baik itu santri maupun pengajar. Setelah itu dilakukan perawatan sistem dengan mengevaluasi kekurangan dari aplikasi selama aplikasi digunakan pengguna.

## 1.6 Jadwal Pengerjaan

Berikut merupakan jadwal pengerjaan aplikasi Pemesanan Makanan Studi Kasus Universitas Telkom:

**Tabel 1 . 1 Jadwal Pengerjaan Aplikasi**

No	Tahapan	Agustus				September				Oktober				November				Desember				Januari				Februari				Maret				April				Mei			
		1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4				
1	Analisis Kebutuhan																																								
	A. wawancara																																								
	B. Kuesioner																																								
	C. Penyusunan																																								
2	Desain Sistem																																								
	A. BPMN																																								
	B. Use Case																																								
	C. ERD																																								
	D. Mockup																																								
3	Implementasi																																								
	A. Kelas Diagram																																								
	B. Skema Relasi																																								
4.	Pengujian Aplikasi																																								
	A. Blackbox testing																																								
	B. UAT																																								
5.	Dokumentasi Aplikasi																																								

## BAB 2

### TINJAUAN PUSTAKA

---

#### 2.1 Madrasah Quran Mesjid Syamsul Ulum

Majelis Quran Syamsul Ulum merupakan salah satu badan yang dibawah oleh Syamsul Ulum Family yang berfokus memberikan layanan pembelajaran Al-Quran kepada mahasiswa dan masyarakat sekitar lingkungan Universitas Telkom. Mesjid Syamsul Ulum (MSU) sendiri merupakan masjid yang berada di lingkungan kampus Universitas Telkom yang didirikan tanggal 28 Desember 1994.

#### 2.2 Al-Quran

Al-Quran merupakan kitab pedoman umat muslim dalam melakukan aktivitas sehari-hari. Al-Quran diturunkan oleh Allah *Subhanahu Wata'ala* kepada Nabi Muhammad *Shallallahu Alaihi wa Sallam* melalui perantarat malaikat Jibril. Al-Quran sangat penting bagi kehidupan manusia karena didalamnya membahas mengenai segala lingkup kehidupan [1].

#### 2.3 Aplikasi

Aplikasi merupakan program komputer yang dibuat untuk mempermudah manusia dalam melakukan proses atau tugas tertentu. Pembuatan aplikasi tergantung dari kebutuhan pengguna maka tujuan dibuatnya aplikasi harus memenuhi atau memfasilitasi keinginan dari penggunanya [3].

#### 2.4 Website

*Website* merupakan kumpulan dokumen multimedia yang saling berhubungan menggunakan *hypertext link*. Melalui *website* pengguna bisa mengakses informasi-informasi yang tidak hanya berupa teks tetapi dapat juga berupa gambar, video serta animasi [4].

## 2.5 Android

Android merupakan sistem operasi berbasis Linux yang dirancang untuk perangkat seluler layar sentuh seperti ponsel pintar hingga tablet. Pada awalnya android dikembangkan oleh Android Inc., lalu dibeli oleh google pada 2005 yang sebelumnya mendanai terkait finansial perusahaan. Android lalu dirilis pertama kali dipasaran pada tahun 2007 [2].

## 2.6 Tools Pemodelan Aplikasi


Tools pemodelan aplikasi merupakan serangkaian perangkat yang digunakan dalam memodelkan aplikasi. Tujuannya memudahkan pengembang dalam merancang atau memodelkan aplikasi. *Tools* yang digunakan adalah Use Case Diagram, Business Process Model and Notation (BPMN), Entity Relationship Diagram (ERD), dan beberapa *tools* dalam Unified Modeling Language seperti (Use Case Diagram, Activity Diagram, Package Diagram dan Deployment Diagram).






### 2.6.1 Business Process Model and Notation (BPMN)







*Business Process Model and Notation* (BPMN) merupakan diagram yang menggambarkan alur proses bisnis suatu sistem atau kegiatan. Tujuan utamanya adalah untuk menyediakan notasi yang mudah dimengerti dalam pemodelan alur bisnis baik yang sudah ada maupun alur bisnis yang baru dirancang. Para pengembang selanjutnya bertanggung jawab membangun aplikasi sesuai dengan proses bisnis yang direncanakan [5].

Berikut merupakan notasi pada *tools* BPMN:

Tabel 2 . 1 Notasi pada BPMN

Notasi	Nama	Penjelasan
	<i>Start Event</i>	Notasi untuk memulai proses

Notasi	Nama	Penjelasan
	<i>Intermediate Event</i>	Notasi Intermediate terletak diantara Start Event dan End Event. Notasi Ini akan mempengaruhi alur proses, tetapi tidak akan memulai atau secara langsung menghentikan proses.
	<i>End Event</i>	Notasi untuk mengindikasikan proses telah berakhir
	<i>Pool</i>	Batasan suatu proses
	<i>Lane</i>	Merupakan batasan suatu objek pelaksana dalam suatu proses bisnis
	<i>Task Activity</i>	Merupakan Notasi untuk suatu pekerjaan
	<i>Subprocess Activity</i>	mengidikasikan adanya proses secara terperinci dengan penggambaran yang sederhana
	<i>Data Object</i>	Notasi yang mengidentikasikan suatu objek data pada suatu proses
	<i>Data Store</i>	Notasi yang menunjukan adanya suatu tempat penyimpanan data

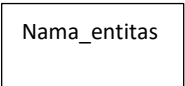
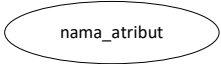
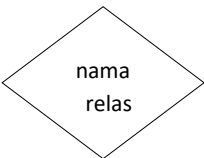
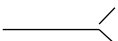
Notasi	Nama	Penjelasan
	<i>Message Event</i>	Mengidentifikasi adanya suatu pesan yang dikirim dari suatu aktivitas ke aktivitas lainnya
	<i>Timer Event</i>	Notasi yang mengidentifikasi adanya waktu yang berjalan atau delay pada suatu aktivitas
	<i>Data Object (output)</i>	Mengidentifikasi adanya data objek yang keluar dari hasil berjalannya suatu aktivitas
	<i>Exclusive gateway</i>	Mengidentifikasi adanya kondisi; kondisi benar hanya pada satu kondisi dan tidak dapat benar pada setiap kondisi
	<i>Parallel gateway</i>	Mengidentifikasi adanya kondisi; kondisi dapat berjalan dengan benar dan dapat berjalan secara parallel atau bersamaan
	Message Flow	Mengidentifikasi adanya alur pesan pada suatu proses bisnis
	Sequence Flow	Notasi adanya alur secara langsung atau direct dari suatu aktivitas ke aktivitas lain

### 2.6.2 Entity Relationship Diagram (ERD)

*Entity Relationship Diagram* (ERD) adalah Teknik umum untuk struktur data perancangan sistem database. Sintaks dan semantik ERD untuk pemodelan data. Pendekatan terstruktur untuk pengembangan model ERD [6].

Berikut merupakan Notasi umum pada ERD:

**Tabel 2 . 2 Notasi Umum pada ERD**

Notasi	Nama	Deskripsi
	Entitas / <i>entity</i>	Entitas merupakan data inti yang akan disimpan; bakal tabel pada basis data; biasanya mengacu pada benda yang terlibat dalam aplikasi yang akan dibuat
	Atribut	<i>Field</i> atau kolom data yang butuh disimpan dalam suatu entitas
	Relasi	Relasi yang menghubungkan antar entitas; biasanya diawali dengan kata kerja
	Asosiasi / <i>association</i>	Penghubung antara relasi dan entitas dimana kedua ujungnya memiliki <i>multiplicity</i> kemungkinan jumlah pemakaian. Kemungkinan jumlah maksimum keterhubungan antara entitas satu dengan entitas yang lain disebut dengan kardinalitas. Misalkan ada kardinalitas 1 ke N atau sering disebut <i>one to many</i> menghubungkan entitas A dan entitas B.

### 2.6.3 Unified Modeling Language (UML)

Unified Modeling Language (UML) adalah himpunan struktur dan teknik untuk pemodelan desain program berorientasi objek (OOP) serta aplikasinya. UML adalah metodologi untuk mengembangkan sistem OOP dan sekelompok perangkat tool untuk mendukung pengembangan sistem tersebut [7].




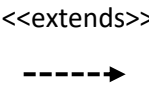
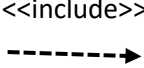


## 2.6.4 Use Case Diagram

Use case diagram adalah alat visual yang memberi jalan bagi pengembang untuk mencapai pemahaman bersama dengan pengguna akhir sistem dan pakar domain. Perilaku kasus penggunaan dapat ditentukan dengan menggambarkan arus kejadian dalam teks agar orang luar dapat memahaminya dengan mudah [7].

Berikut merupakan beberapa Notasi pada Use Case Diagram:

Tabel 2 . 3 Notasi pada Use Case Diagram

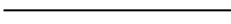
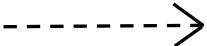
Notasi	Nama	Deskripsi
	<i>Use case</i>	Fungsionalitas atau proses-proses yang disediakan aplikasi sebagai unit-unit yang saling bertukar pesan antar unit atau aktor; biasanya dinyatakan dengan menggunakan kata kerja diawal frase nama <i>use case</i>
	Aktor / <i>actor</i>	Orang, proses, atau aplikasi lain yang berinteraksi dengan aplikasi yang akan dibuat diluar aplikasi yang akan dibuat itu sendiri, jadi walaupun Notasi dari aktor adalah gambar orang, tapi aktor belum tentu merupakan orang.
	Asosiasi / <i>association</i>	Komunikasi antara aktor dan <i>use case</i> yang berpartisipasi pada <i>use case</i> atau <i>use case</i> memiliki interaksi dengan aktor.
	<i>Extends</i>	Relasi <i>use case</i> tambahan ke sebuah <i>use case</i> di mana <i>use case</i> yang ditambahkan berhubungan saat menjalankan <i>usecase</i> lainnya.
	<i>Include</i>	Relasi <i>use case</i> tambahan ke sebuah <i>use case</i> di mana <i>use case</i> yang ditambahkan memerlukan <i>use case</i> ini untuk menjalankan fungsinya atau sebagai syarat dijalankan <i>use case</i> ini.

### 2.6.5 Class Diagram

Kelas Diagram digunakan untuk memodelkan hubungan statis antar komponen dengan suatu sistem. Model uml tunggal dapat memiliki banyak diagram kelas yang menunjukkan sistem yang sama dari pandangan yang berbeda. misalnya, diagram kelas mungkin menunjukkan beberapa kelas menggunakan antarmuka subsistem namun tidak menjelaskan secara terperinci mengenai implementasi subsistem. Diafram kelas yang berbeda, yang digunakan oleh pengembang subsistem, dapat menunjukkan antarmuka subsistem dan kelas yang membantu menyesuaikan subsistem tersebut.

Berikut merupakan notasi umum dari kelas diagram:

Tabel 2 . 4 Class Diagram



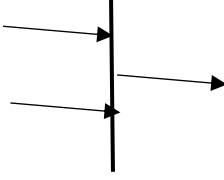
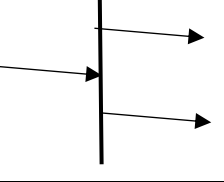

Notasi		Nama	Deskripsi
<div style="border: 1px solid black; padding: 5px; display: inline-block;"> Nama_kelas  (+) atribut  (+) operator </div>		Class/kelas	Kelas pada struktur sistem
		asosiasi / <i>association</i>	Menggambarkan hubungan langsung terhadap kelas lain
		Dependency	Menggambarkan ketergantungan terhadap kelas lain

### 2.6.6 Activity Diagram

Menggambarkan proses bisnis serta urutan aktivitas dalam suatu proses pada sistem [6]. Salah satu cara menjabarkan use case adalah menggunakan activity diagram.

Berikut merupakan notasi umum pada Activity Diagram:

Tabel 2 . 5 Notasi Pada Activity Diagram

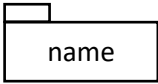
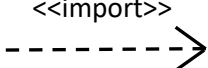
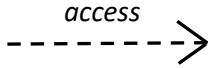
Notasi	Nama	Deskripsi
	Start	Untuk memulai suatu aktifitas
	End	Mengakhiri aktifitas
	join	Menggambarkan adanya penggabungan
	Fork	Menggambarkan adanya percabangan
	Decision	Menggambarkan adanya percabangan yang harus dipilih salah satunya. Notasi ini juga dapat digunakan dalam menyatukan notasi yang dipisah oleh Decision sebelumnya

### 2.6.7 Package Diagram

Package merupakan kontainer serbaguna yang merepresentasikan hubungan antara UML *elements*. Package memiliki perbedaan secara grafik dengan class. Tujuannya untuk menggambarkan sifat hirarki. Sebuah paket diberi nama yang menggambarkan isi dari package [6].

Berikut merupakan beberapa Notasi pada Class Diagram:

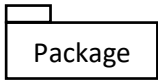
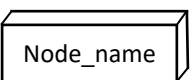
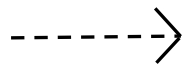
**Tabel 2 . 6 Notasi pada Package Diagram**

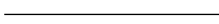

Notasi	Nama	Deskripsi
	Package	Sekelompok elemen model
	Import	Mengindikasikan isi tujuan paket secara umum yang akan ditambahkan kedalam sumber paket
	Access	Mengindikasikan isi tujuan paket secara umum yang dapat digunakan pada sumber paket

### 2.6.8 Deployment Diagram

Deployment diagram menunjukkan node secara fisik yang dieksekusi oleh sistem dan biasanya mengandung artifacts, nodes, components, dan hubungan antara nodes dan components [6]. Tujuannya untuk menspesifikasi dan mendokumentasikan proses yang berjalan pada suatu sistem.

**Tabel 2 . 7 Notasi pada Deployment Diagram**

Notasi	Nama	Deskripsi
	Package	Bertujuan untuk membungkus satu atau beberapa node
	Node	Biasanya mengacu pada <i>hardware/software</i> .
	Dependency/ ketergantungan	Menggambarkan kebergantungan antar node, arah panah mengacu pada node yang digunakan

Notasi	Nama	Deskripsi
	Link	Menggambarkan relasi antar node
	Objek	Menggambarkan adanya objek, biasanya menjelaskan objek yang ada didalam node

### 2.6.9 Balsamiq Mockups

Balsamiq Mockups merupakan alat untuk mendesain antarmuka pengguna dalam bentuk wireframe (maket atau prototype). Balsamiq Mockups dapat digunakan untuk membuat sketsa secara digital dari produk yang akan dibuat [8].

### 2.6.10 yEd Graph Editor

yEd Graph Editor merupakan aplikasi yang membantu dalam pembangunan aplikasi yang memungkinkan visualisasi grafik, diagram dan jaringan yang jelas. yEd Graph Editor menyatukan struktur data yang efisien, kompleks, dan teknik berkelanjutan yang memberikan interaksi pengguna kepada tarket perangkat [9].

### 2.6.11 StarUML

StarUML merupakan aplikasi yang berfungsi sebagai alat perancangan perangkat lunak yang bertujuan untuk mendukung pemodelan sifat dan struktur perangkat lunak [10].

## 2.7 Tools Pembangunan Aplikasi

*Tools* dalam pembangunan aplikasi merupakan perangkat yang digunakan untuk membantu atau mempermudah pembuatan aplikasi yang telah dirancang. Adapun *Tools* yang digunakan adalah Kotlin, SQLite, Android Studio, Balsamiq Mockups, yEd Graph Editor, StarUML, Atom, X Apache-MySQL-PHP-Perl (XAMPP), *hypertext preprocessor* (PHP), Codeigniter (CI), Hypertext Markup Language (HTML), CSS, dan JavaScript.

### 2.7.1 Kotlin

Kotlin merupakan bahasa pemrograman yang dapat berjalan diatas mesin virtual java. Pada google I/O 2017 kotlin diumumkan sebagai bahasa pemrograman yang termasuk kedalam bahasa kelas satu yang mendukung pembuatan aplikasi Android. Kelebihan dari kotlin adalah mudah dipahami dan menghemat penulisan program hingga 30% [11].

### 2.7.2 SQLite

SQLite merupakan database yang bersifat *open source* yang dirilis pada tahun 2000. Fungsinya adalah menyediakan cara yang mudah untuk aplikasi mengelola data tanpa *overhead* yang sering terjadi pada sistem basisdata relasional. SQLite berorientasi portable sehingga mudah digunakan, ringkas, efisien dan andal [12].

### 2.7.3 Android Studio

Android Studio merupakan tools dalam membangun aplikasi android, keunggulannya android yang merupakan sistem operasi yang berbasis linux dapat dibangun diatas sistem operasi windows dengan peformansi yang tidak berbeda jauh dengan linux. Android studio memudahkan para pengembang dalam membangun aplikasi android karena fitur yang disajikan mudah dipahami oleh pengembang [13].

### 2.7.4 Atom

Atom merupakan aplikasi *text editor* yang bersifat *open source*. Bagi pengembang, Atom sangat membantu karena memungkinkan memasukan package tersendiri sesuai dengan kebutuhan aplikasi yang dibangun. Atom sangat populer dalam melakukan pengkodean aplikasi berbasis *website*. Atom dapat terhubung kebeberapa *platform* lainnya seperti GitHub dan Heroku [14].

### 2.7.5 X Apache-MySQL-PHP-Perl (XAMPP)

X Apache-MySQL-PHP-Perl (XAMPP) merupakan *control panel* untuk pemrograman web, dengan menyediakan paket instalasi Apache, MySQL, PHP dan Perl dengan cepat. XAMPP menyediakan beberapa fitur dalam pembuatan *website* dengan

komputer sebagai server lokal. Untuk basis data dapat menggunakan *Graphical User Interface* (GUI) PHPMyAdmin untuk konfigurasi atau pengaturan [15].

#### 2.7.6 Hypertext Preprocessor (PHP)

Hypertext Preprocessor (PHP) merupakan bahasa pemrograman yang mampu berjalan diatas server dan memungkinkan pembuatan *website* secara statis maupun dinamis. PHP mampu mengolah data dari komputer klien serta mampu mengolah data server sehingga informasi yang disajikan dapat ditampilkan menggunakan browser [15].

Berdasarkan perkembangan teknologi saat ini PHP menjadi bahasa pemrograman *website* yang sangat populer didunia karena sebagian besar *website* yang ada dibangun diatas bahasa pemrograman PHP.

#### 2.7.7 Codeigniter

Codeigniter merupakan sebuah *framework* yang memudahkan pengguna dalam membangun sebuah aplikasi *website*. Konsep yang ditawarkan adalah konsep MVC (*model-view-controller*). *Model* mengatur bagian basis data, *view* bertugas mengatur tampilan antarmuka dan *controller* sebagai penggabung antara basis data dengan antarmuka. Selain itu Codeigniter bersifat *open source* sehingga para pengembang dapat memodifikasi struktur dari framework Codeigniter [16].

#### 2.7.8 HTML, CSS dan JavaScript

Hypertext Markup Language (HTML), Cascading Style Sheet (CSS), JavaScript dan Bootstrap merupakan teknologi dasar untuk *Website*. Jika PHP dan MySql mengatur *backend* dan *database* maka HTML dan CSS merupakan bahasa markup dasar yang bertugas mengatur antarmuka sedangkan JavaScript dan Bootstrap merupakan *framework* yang memudahkan dalam mengatur antarmuka aplikasi *website* [17].

### 2.8 Tools Pengujian Aplikasi

*Tools* pengujian aplikasi merupakan perangkat yang digunakan untuk menguji aplikasi dari kesesuaian kebutuhan pengguna. *Tools* yang digunakan adalah *blackbox testing* dan *User Acceptance Test* (UAT).

### 2.8.1 Blackbox Testing

*Blackbox testing* merupakan pengujian yang berfokus terhadap fungsional aplikasi. Tujuannya memungkinkan penguji mendapatkan serangkaian kondisi yang sesuai dengan keinginan berdasarkan input yang dimasukan [18].

Tabel 2 . 8 Contoh Format Black Box Testing

Input	Output	Expectation
"laki laki"	Gambar laki-laki	Gambar laki-laki
"unta"	Suara kelelawar	Suara unta

### 2.8.2 User Acceptance Test (UAT)

*User Acceptance Test* (UAT) merupakan jenis pengujian kepada pengguna dari aplikasi. Tujuannya agar aplikasi yang dibangun sesuai dengan kebutuhan serta dapat menangani tugas-tugas yang diinginkan oleh pengguna aplikasi.

Pengujian ini membantu menemukan kesalahan yang berkaitan dengan fungsional aplikasi dengan cara memeriksa fungsi pada setiap menu yang ada pada dokumen *requirements*. Hasil dari pengujian UAT adalah dokumen yang dijadikan bukti bahwa aplikasi yang diuji dapat diterima atau tidak oleh pengguna [19].

Adapun contoh merupakan format dari UAT adalah sebagai berikut:

Tabel 2 . 9 Contoh Format User Acceptence Test

No	Pertanyaan	Sangat Baik	Cukup Baik	Baik	Buruk	Sangat Buruk
1	Apakah tampilan dari aplikasi memudahkan anda memahami alur aplikasi?		√			



No	Pertanyaan	Sangat Baik	Cukup Baik	Baik	Buruk	Sangat Buruk
2	Apakah aplikasi telah memberikan kemudahan terhadap anda dalam melakukan pemesanan kelas?	√				

## BAB 3

### ANALISIS DAN PERANCANGAN

---

#### 3.1 Analisis

Adapun analisis pada sistem yang berjalan dan diusulkan untuk membangun aplikasi adalah sebagai berikut.

##### 3.1.1 Gambaran Sistem Saat Ini

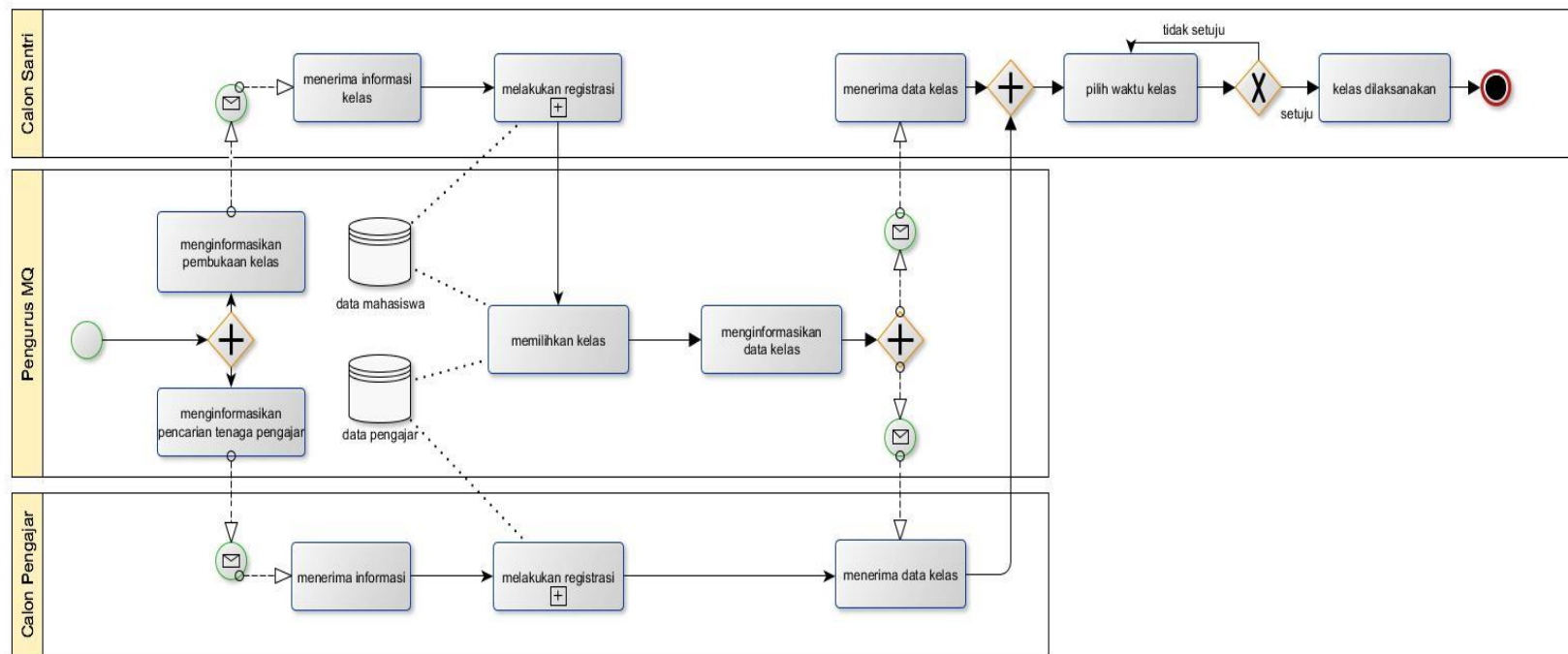
Berikut merupakan gambaran sistem yang sedang berjalan digambarkan dengan proses bisnis.

##### 3.1.1.1 Proses Bisnis yang Berjalan Menggunakan BPMN

Berikut merupakan gambaran sistem yang berjalan saat ini menggunakan BPMN (*Business Process Model Notation*):

##### 3.1.1.1.1 Proses Bisnis utama Madrasah Quran Mesjid Syamsul Ulum

Berikut merupakan proses bisnis utama Madrasah Quran Mesjid Syamsul Ulum:



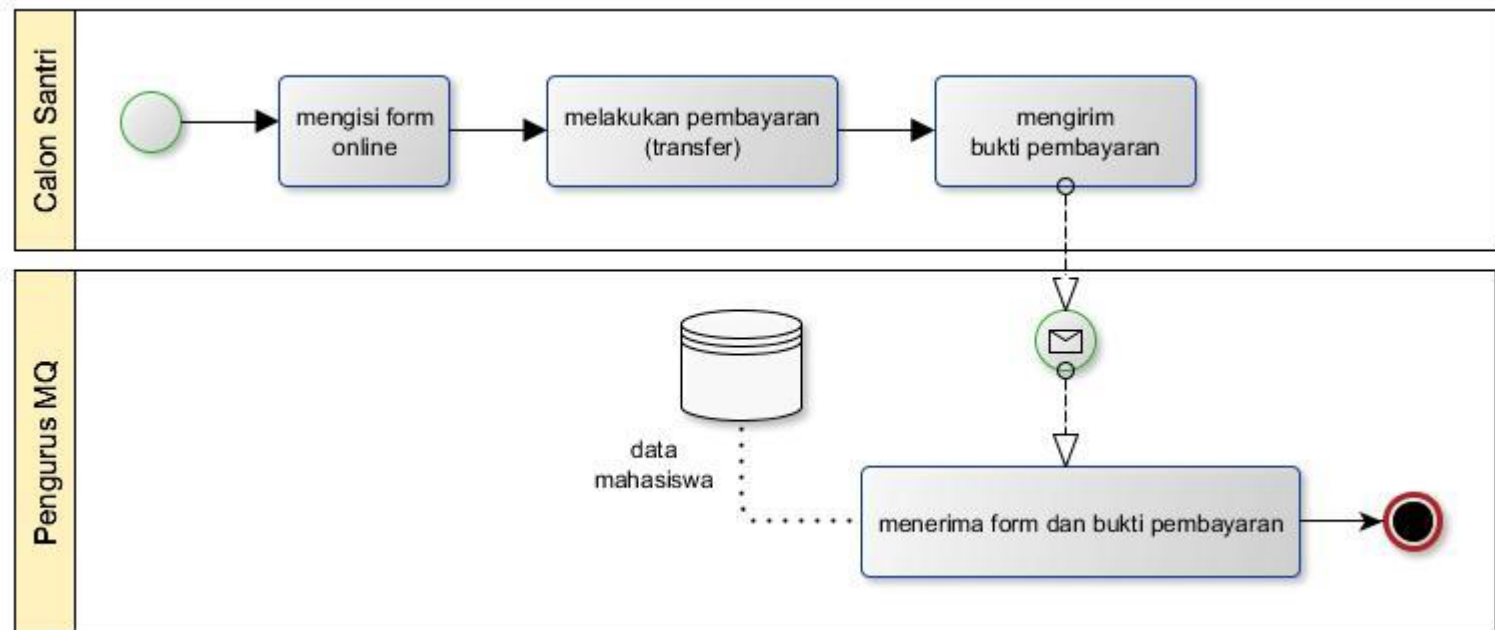
Gambar 3 . 1 Proses Bisnis Utama MQMSU

Berikut merupakan penjelasan dari alur proses bisnis yang berjalan:

1. Pengurus MQMSU menginformasikan pembukaan kelas belajar membaca mengaji bersamaan dengan menginformasikan pencarian tenaga pengajar.
2. Calon santri lalu menerima informasi kelas melalui *Official Account* LINE MQMSU.
3. Calon Tenaga pengajar yang telah mendapatkan informasi selanjutnya melakukan registrasi untuk mendaftarkan dirinya sebagai pengajar.
4. Selanjutnya calon santri melakukan registrasi (isi formulir online, membayar kelas via transfer bank).
5. Setelah mahasiswa melakukan pembayaran selanjutnya pengurus MQ melakukan pemilihan kelas dengan mengelompokkan beberapa calon santri dalam satu pengajar.
6. Setelah selesai maka pengurus MQ menginformasikan kelas kepada santri dan pengajar.
7. Selanjutnya pengajar dan calon santri melakukan negosiasi terkait jadwal kelas.
8. Setelah itu jika belum ada persetujuan terkat jadwal maka kembali pengajar dan calon santri kembali mengatur jadwal kelas.
9. Setelah menemukan jadwal kelas maka pengajar dan santri telah dapat melakukan pembelajaran.

### 3.1.1.1.2 Proses Bisnis Registrasi Calon Santri

Berikut merupakan proses bisnis ketika mahasiswa melakukan registrasi pendaftaran menjadi santri pada MQMSU:



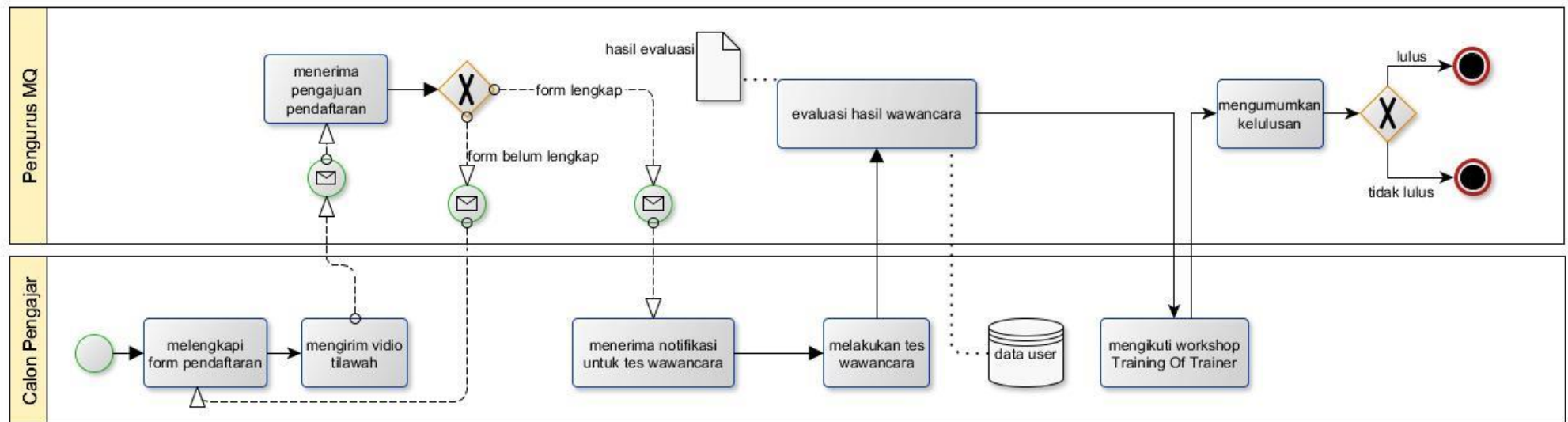
Gambar 3 . 2 Proses Bisnis Registrasi Mahasiswa

Berikut merupakan proses bisnis registrasi calon santri yang sedang berjalan:

1. Setelah calon santri menerima informasi terkait kelas yang dibuka, selanjutnya calon santri mengisi formulir secara online.
2. Setelah mengisi formulir, calon santri melakukan pembayaran dengan transfer bank, selanjutnya mengirim bukti pembayaran.
3. Setelah itu pengurus MQ menerima formulir dan bukti pembayaran.
4. Setelah lengkap maka pengurus MQ akan membuka memilih kelas yang akan diikuti oleh mahasiswa terkait. Jika mahasiswa telah mendapatkan kelas maka mahasiswa telah dapat dikatakan menjadi santri.

#### **3.1.1.1.3 Proses Bisnis Registrasi Calon Pengajar**

Dibawah ini merupakan alur calon pengajar dalam melakukan pendaftaran menjadi tenaga pengajar pada MQMSU:



Gambar 3 . 3 Proses Bisnis Registrasi Calon Pengajar

Berikut merupakan deskripsi dari proses bisnis registrasi calon pengajar:

1. Calon pengajar melengkapi formulir pendaftaran yang telah tersedia secara online.
2. Calon pengajar mengirim video tilawah sebagai salah satu syarat pendaftaran menjadi pengajar.
3. Setelah itu pengurus MQ menerima pengajuan pendaftaran. Jika belum lengkap maka calon pengajar kembali mengisi formulir pendaftaran dan mengirim video tilawah. Apabila telah lengkap maka calon pengajar akan menerima notifikasi untuk melakukan tes wawancara.
4. Setelah menerima notifikasi calon pengajar melakukan tes wawancara.
5. Setelah tes wawancara panitia MQ melakukan evaluasi hasil wawancara lalu menyimpan data calon pengajar.
6. Setelah mengevaluasi hasil wawancara, calon pengajar mengikuti *Workshop Training Of Trainer* untuk memberikan pengetahuan terkait metode pembelajaran membaca Al-Quran.
7. Setelah mengikuti workshop sekolah mentor, pengurus MQ mengumumkan kelulusan para calon pengajar.

### 3.1.1.2 Perbandingan Aplikasi Sejenis

Berikut merupakan tabel perbandingan dengan aplikasi sejenis:

**Tabel 3 . 1 Pembanding Aplikasi Sejenis**

No	Fungsionalitas	Ngaji Asik!	Mengaji	Learns Quran
1	<i>Sign Up</i>	√	√	
2	<i>Sign In</i>	√	√	
3	<i>Hadist collection</i>		√	
4	Mendengarkan Surah		√	
5	Monitoring Kualitas Bacaan	√		
6	Potongan bacaan ayat			√



No	Fungsionalitas	Ngaji Asik!	Mengaji	Learns Quran
7	Mengatur jadwal tersedia	√		
8	Memilih Pengajar	√		
9	Memilih jadwal pembelajaran	√		
10	Video pembelajaran		√	√
11	Animasi pembelajaran			√
12	<i>Setting profile</i>	√	√	
13	Penghitung Zikir		√	
14	Jadwal Sholat		√	
15	Doa Harian		√	√
16	<i>Chat</i>	√		
17	Iklan			√

### 3.1.1.3 Kelemahan Sistem Berjalan dan Usulan Perbaikan

Berikut merupakan tabel kelemahan sistem yang sedang berjalan serta usulan perbaikannya:

Tabel 3 . 2 Kelemahan Sistem Berjalan dan Usulan Perbaikan

No	Kelemahan	Usulan
1	Registrasi calon santri yang memakan waktu yang lama, sehingga calon santri enggan untuk mengikuti tahapan registrasi	Menyederhanakan tahapan registrasi dengan pembayaran dapat dilakukan setelah calon santri telah mendaftarkan diri melalui aplikasi
2	Calon pengajar mengikuti <i>workshop training of trainer</i> padahal calon pengajar belum tentu lulus sebagai pengajar, sehingga jalannya <i>workshop training of trainer</i> tidak begitu baik dikarenakan banyaknya calon pengajar yang mengikuti kegiatan tersebut	<i>Workshop training of trainer</i> dilakukan setelah adanya pengumuman kelulusan dari pengurus MQ agar <i>workshop training of trainer</i> menjadi lebih cepat karena jumlah calon pengajar lebih sedikit.

No	Kelemahan	Usulan
3	Pembayaran dilakukan pada saat registrasi sehingga calon santri harus membayar diawal	Calon santri dapat mendaftarkan dirinya terlebih dahulu sebelum melakukan pembayaran. Jenis pembayaran juga dapat ditentukan sesuai dengan keinginan calon santri.
4	Santri yang belajar membaca dikelompokkan sehingga ada kemungkinan pemahaman terkait membaca Al-Quran berbeda tiap santrinya. Hal ini berpengaruh dalam baiknya proses pembelajaran	Bentuk pembelajaran tidak berkelompok melainkan privat antar muka. Sehingga, pengajar dapat mengetahui kondisi/kemampuan santri lebih dalam. Santri juga dapat lebih paham dalam membaca Al-Quran.
5	Santri biasanya tidak dapat mengikuti pembelajaran setiap pertemuan, akibatnya santri tertinggal dalam hal pelajaran.	Santri tidak akan mengalami ketertinggalan pembelajaran dikarenakan santri belajar secara privat dan dapat mengatur jadwal pembelajaran.
6	Pembelajaran hanya ketika bertemu, akibatnya jika ada santri yang ingin mengajukan pertanyaan maka harus bertatap muka	Santri yang memiliki pertanyaan dapat mengajukan melalui fitur <i>chat</i> , sehingga santri tidak perlu menunggu bertemu secara langsung.
7	Tempat pembelajaran ditentukan oleh pengurus MQ, sehingga santri harus menyesuaikan diri.	Tempat dapat disesuaikan dengan keinginan santri.
8	Pengajar ditentukan oleh pengurus MQ tanpa diinformasikan terkait latar belakang dari pengajar kepada santri.	Santri dapat memilih pengajar yang diinginkan, selain itu santri dapat melihat latar belakang dari pengajar tersebut.

### **3.1.2 Gambaran Sistem Usulan**

Adapun gambaran sistem yang diusulkan adalah sebagai berikut:

#### **3.1.2.1 Gambaran Umum Sistem Usulan**

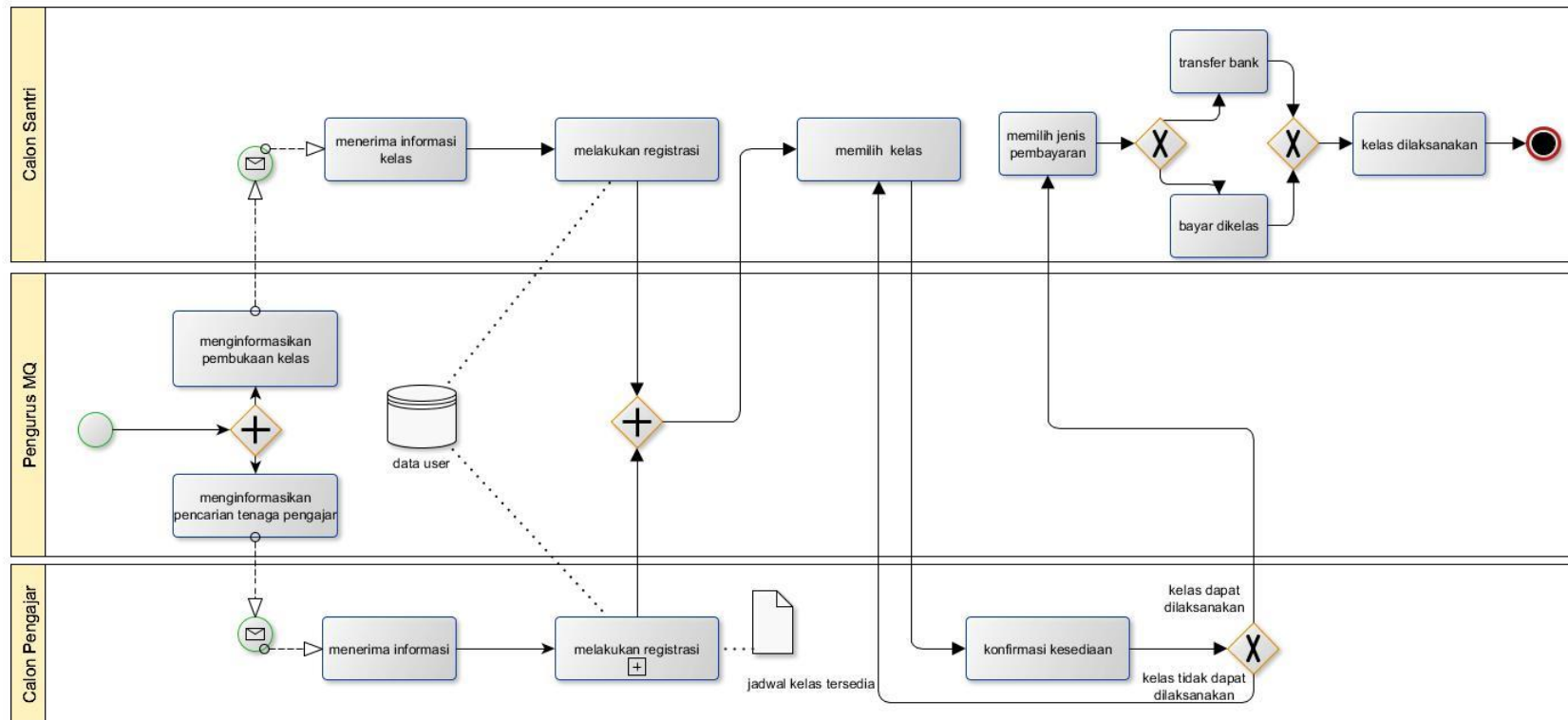
Sistem yang diusulkan bertujuan agar mempermudah proses belajar membaca Al-Quran. Gambaran yang utama adalah bagaimana santri dapat menentukan jadwal, tempat dan pengajar yang diinginkan. Pembayaran kelas dapat dilakukan dengan transfer bank atau pembayaran langsung ketika melakukan proses belajar. Dari sisi bisnis, pengajar akan langsung mendapatkan bayaran atas jasa yang ditawarkan. Dapat dilihat bahwa orientasi dari sistem usulan ini adalah santri itu sendiri, dimana santri sangat diutamakan kenyamanannya. Harapannya adalah santri menjadi nyaman dalam belajar membaca Al-Quran sehingga santri akan semakin mudah dan cepat dalam memahami cara membaca Al-Quran.

#### **3.1.2.2 Proses Bisnis Usulan Menggunakan BPMN**

Berikut merupakan gambaran sistem yang diusulkan menggunakan BPMN (*Business Process Model Notation*):

##### **3.1.2.2.1 Usulan Proses Bisnis Utama Madrasah Quran Mesjid Syamsul Ulum**

Berikut merupakan usulan proses bisnis utama Madrasah Quran Mesjid Syamsul Ulum:



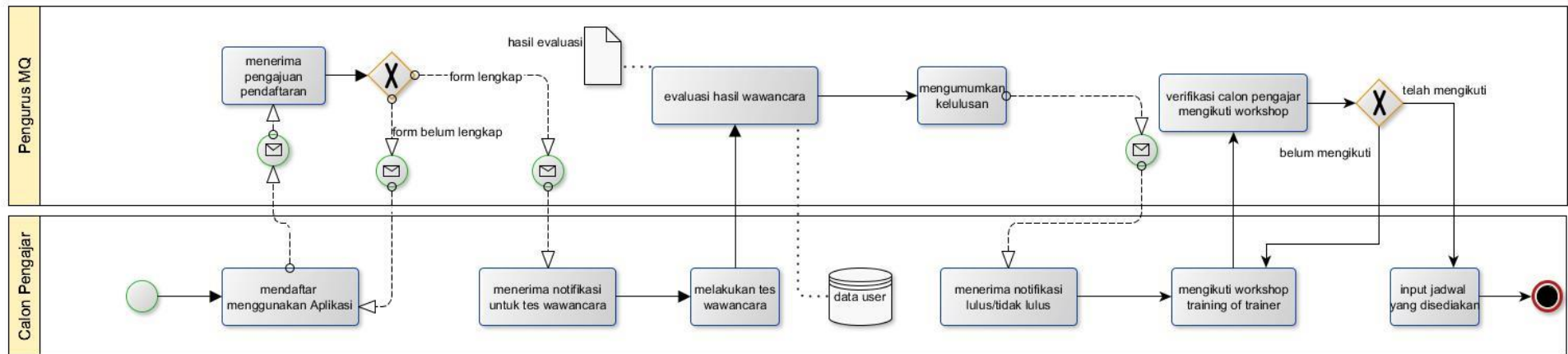
Gambar 3 . 4 Usulan Proses Bisnis Utama MQMSU

Berikut merupakan penjelasan usulan proses bisnis utama MQMSU:

1. Pengurus menginformasikan pembukaan kelas belajar membaca Al-Quran untuk mahasiswa. Mahasiswa yang mendaftar dikatakan sebagai calon santri.
2. Pengurus juga menginformasikan pembukaan pendaftaran pagi para mahasiswa yang ingin menjadi pengajar membaca Al-Quran.
3. Mahasiswa yang berminat dan telah mendapatkan informasi pendaftaran, melakukan registrasi. Output dari registrasi adalah jadwal kelas yang dapat dipilih oleh santri. Mahasiswa dikatakan sebagai pengajar apabila telah lulus dari proses registrasi.
4. Calon santri selanjutnya melakukan registrasi melalui aplikasi setelah menerima informasi kelas belajar membaca Al-Quran. Pada tahap ini mahasiswa hanya mengisi formulir registrasi pada aplikasi sehingga tidak perlu mendaftarkan diri melalui formulir online pihak ketiga lainnya.
5. Setelah melakukan pendaftaran, data calon santri disimpan lalu calon santri dapat langsung memilih jadwal kelas yang tersedia. Daftar pengajar akan otomatis sesuai dengan jenis kelamin calon santri. Sehingga menutup kemungkinan adanya interaksi lawan jenis ketika melakukan proses pembelajaran.
6. Selanjutnya, pengajar mendapatkan notifikasi pemesanan, lalu mengonfirmasi kesediaan.
7. Jika pengajar tidak bersedia maka calon santri mendapatkan notifikasi untuk memilih jadwal/pengajar lainnya.
8. Jika pengajar bersedia maka santri mendapatkan notifikasi untuk melakukan pembayaran, pembayaran dapat dilakukan dengan transfer bank atau bayar dikelas.
9. Selanjutnya santri dan pengajar bertemu untuk melakukan proses belajar membaca Al-Quran.

#### **3.1.2.2.2 Usulan Proses Bisnis Registrasi Calon Pengajar**

Berikut merupakan usulan proses bisnis registrasi Calon Pengajar:



Gambar 3 . 5 Usulan Proses Bisnis Registrasi Calon Pengajar

Berikut merupakan penjelasan usulan proses bisnis registrasi Calon Pengajar:

1. Mahasiswa yang ingin menjadi pengajar mendaftar melalui aplikasi.
2. Selanjutnya pengurus MQ menerima pengajuan pendaftaran, jika formulir belum lengkap maka calon pengajar mengisi kembali formulir pendaftaran.
3. Setelah lengkap calon pengajar mendapatkan notifikasi untuk melakukan tes wawancara, jadwal akan langsung diinformasikan.
4. Selanjutnya calon pengajar melakukan tes wawancara (mengajukan beberapa pertanyaan dan tes mengaji).
5. Setelah melakukan tes wawancara, pengurus MQ mengevaluasi hasil wawancara lalu mengumumkan kelulusan.
6. Bagi calon pengajar yang lulus selanjutnya mengikuti *workshop training of trainer*.
7. Pengurus MQ selanjutnya melakukan verifikasi calon pengajar dalam mengikuti *workshop training of trainer* jika belum maka harus mengikuti. Aktifitas Ini merupakan syarat utama dalam menjadi pengajar.
8. Jika telah mengikuti maka calon pengajar berubah status menjadi pengajar dan dapat menginput jadwal ketersediaan kelas.

### **3.1.3 Analisis Kebutuhan Sistem (atau Produk)**

Adapun di bawah ini merupakan analisis kebutuhan sistem (atau produk):

#### **3.1.3.1 Analisis Fungsionalitas**

Berikut merupakan tabel analisis fungsionalitas:

Tabel 3 . 3 Fungsionalitas Aplikasi

No	Fungsionalitas	Deskripsi	Data/Dokumen input	Data/dokumen output	USER		
					Santri	Pengajar	Pengurus MQ
1	Kelola akun	Bertujuan mengatur hak akses dari setiap user					
	1.1 Buat akun	Membuat akun ( <i>sign up</i> )	Nama lengkap, alamat, surel, foto profil, umur		√	√	
	1.2 Edit akun	Mengedit akun seperti nama, surel, foto profil dan status	Sesuai dengan item yang ingin diedit (Nama lengkap, alamat, surel, foto profil, umur)		√	√	√
	1.3 Ubah password	Mengubah password dari akun	Password baru		√	√	√
	1.4 Hapus akun	Akun dapat dihapus apabila melakukan pelanggaran, akun yang dihapus adalah santri dan pengajar					√
	1.5 tutup akun	Akun dapat ditutup oleh user			√	√	
2	Jadwal ketersediaan						
	2.1 Input jadwal kesediaan	Tujuannya agar santri dapat melihat jadwal yang tersedia	Hari dan jam			√	



No	Fungsionalitas	Deskripsi	Data/Dokumen input	Data/dokumen output	USER		
					Santri	Pengajar	Pengurus MQ
	2.2 update jadwal kesediaan	Dapat mengupdate jadwal yang tersedia	Hari dan jam			√	
	2.3 hapus jadwal kesediaan	Jadwal dapat dihapus oleh pemilik akun				√	
	2.4 lihat jadwal tersedia	Melihat semua jadwal yang tersedia		Jadwal yang tersedia	√	√	√
	2.5 membatalkan jadwal	Jadwal dapat dibatalkan dengan alasan tertentu			√	√	
3	Kelola kelas						
	3.1 Pilih jadwal kelas	Jadwal yang telah tersedia dapat dipilih			√		
	3.2 pilih pengajar	Pengajar yang telah menginput jadwal dapat dipilih			√		
	3.3 konfirmasi kelas	Kelas yang telah dipesan harus dikonfirmasi oleh pengajar				√	
4	Pilih jenis pembayaran	Jenis pembayaran dibedakan menjadi 2, transfer bank atau bayar dikelas			√		

No	Fungsionalitas	Deskripsi	Data/Dokumen input	Data/dokumen output	USER		
					Santri	Pengajar	Pengurus MQ
5	Notifikasi						
	5.1 notifikasi wawancara					√	
	5.2 notifikasi pengumuman					√	
	5.3 notifikasi janji				√	√	
	5.4 notifikasi kesediaan pengajar				√		
6	Poin	Semakin tinggi poin artinya semakin sering orang tersebut melakukan proses belajar, juga sebagai monitoring kemampuan baca Al-Quran. Poin menggambarkan keaktifan dari <i>user</i> .					
	6.1 lihat poin			Poin yang dimiliki	√	√	

No	Fungsionalitas	Deskripsi	Data/Dokumen input	Data/dokumen output	USER		
					Santri	Pengajar	Pengurus MQ
7	<i>Chat</i>	Tujuan fungsionalitas <i>chat</i> agar interaksi belajar antara pengajar dan santri lebih sering dilakukan			√	√	
8	Lihat daftar keseluruhan kelas	Melihat daftar keseluruhan kelas yang akan dan telah terlaksana					√
9	Melihat daftar user	Dapat melihat informasi pengajar dan santri					√
10	Hapus akun <i>user</i>	<i>user</i> dapat dihapus akunya apabila dianggap melakukan pelanggaran oleh pihak MQ					√
11	Input nilai calon pengajar	Menginput nilai sebagai bahan pertimbangan kelulusan calon pengajar					√

### 3.1.3.2 Analisis Pengguna

Berikut merupakan analisis *user* yang menggunakan aplikasi:

**Tabel 3 . 4 Analisis Pengguna**

No	Pengguna	Deskripsi	Kompetensi IT/ Non IT yang dimiliki
1	Pengurus MQ	Merupakan mahasiswa yang memiliki jabatan sebagai pengurus MQMSU	Mampu memahami penggunaan aplikasi secara umum
2	Pengajar	Merupakan mahasiswa yang mendaftar sebagai pengajar. Apabila belum diterima sebagai pengajar maka status masih calon pengajar. Berkompetensi dibidang baca Al-Quran	Mampu menggunakan perangkat android
3	Santri	Merupakan mahasiswa yang belajar Al-Quran, sebelum melakukan pembelajaran maka status masih calon santri. Mahasiswa ini kurang memiliki kompetensi dalam membaca Al-Quran	Memiliki kompetensi IT karena sedang berkuliah

### 3.1.4 Kebutuhan Perangkat Keras dan Perangkat Lunak

Berikut merupakan kebutuhan perangkat keras dan perangkat lunak pada aplikasi:

#### 3.1.4.1 Pengembangan Aplikasi

Adapun beberapa kebutuhan perangkat keras dan perangkat lunak yang dibutuhkan dalam pengembangan sistem selama proses pembuatan aplikasi adalah:

##### 3.1.4.1.1 Kebutuhan Perangkat Keras Pengembangan Aplikasi

Berikut merupakan spesifikasi kebutuhan perangkat keras yang dibutuhkan dalam pembuatan aplikasi:

**Tabel 3 . 5 Kebutuhan Perangkat Keras Pengembangan Aplikasi**

No	Jenis Hardware	Spesifikasi Minimum Perangkat Lunak	Spesifikasi Perangkat yang digunakan
1	RAM Laptop	512 MB	12 GB
2	Harddisk	2 GB	1 Tera
3	Solid State Disk	2 Gb	250 GB
4	Processor	Minimum kecepatan 1.5 GHz	Intel core i7 – 5005U up to 1.7 GHz
5	Android	API 20 Lollipop 5.0	API 21 Marshmello 6.1
6	RAM Android	1.5 GB	3GB

#### 3.1.4.1.2 Kebutuhan Perangkat Lunak Pengembangan Aplikasi

Berikut merupakan kebutuhan perangkat lunak yang menunjang pembuatan aplikasi:

**Tabel 3 . 6 Kebutuhan Perangkat Lunak Pengembangan Aplikasi**

No	Jenis Software	Perangkat yang digunakan	Spesifikasi
1	Operating System	Windows 10	Processor: 1 gigahertz (GHz) or faster, RAM: 1 gigabyte (GB) (32-bit) or 2 GB (64-bit), Free hard disk space: 16 GB, Graphics card: Microsoft DirectX 9 graphics device with WDDM driver. A Microsoft account and Internet access.
		Android Lollipop 5.0	64 Bit, 3GB RAM
2	Browser	Chrome	Windows 10
3	Dokumentasi	Word 2016	Windows 10
4	Database	MySQL, SQLite	
5	Android Studio	Android Studio 3.0	Windows 10, 6GB RAM, 1.5 GHz Processor
6	CodeIgniter	CodeIgniter 3	Web browser
7	Atom	Atom 3	Windows 10, 2GB RAM

### 3.1.4.2 Implementasi Sistem

Adapun yang merupakan implementasi kebutuhan pada sisi *Server* dan *Client* adalah sebagai berikut.

#### 3.1.4.2.1 Kebutuhan Perangkat Keras

Berikut merupakan kebutuhan implementasi kebutuhan perangkat keras pada sisi *Server* dan *Client*:

**Tabel 3 . 7 Kebutuhan Perangkat Keras Implementasi Sistem**

No	Jenis Software	Server	Client
1	RAM	-	3GB
2	ROM/Penyimpanan	1GB	500MB
3	Perangkat	-	Android, Komputer

#### 3.1.4.2.2 Kebutuhan Perangkat Lunak Implementasi Aplikasi

Berikut merupakan kebutuhan implementasi kebutuhan perangkat lunak pada sisi *Server* dan *Client*:

No	Jenis Software	Server	Client
1	Browser	-	Chrome,Firefox,Opera
2	Sistem Operasi	-	Android 5.0

## 3.2 Perancangan

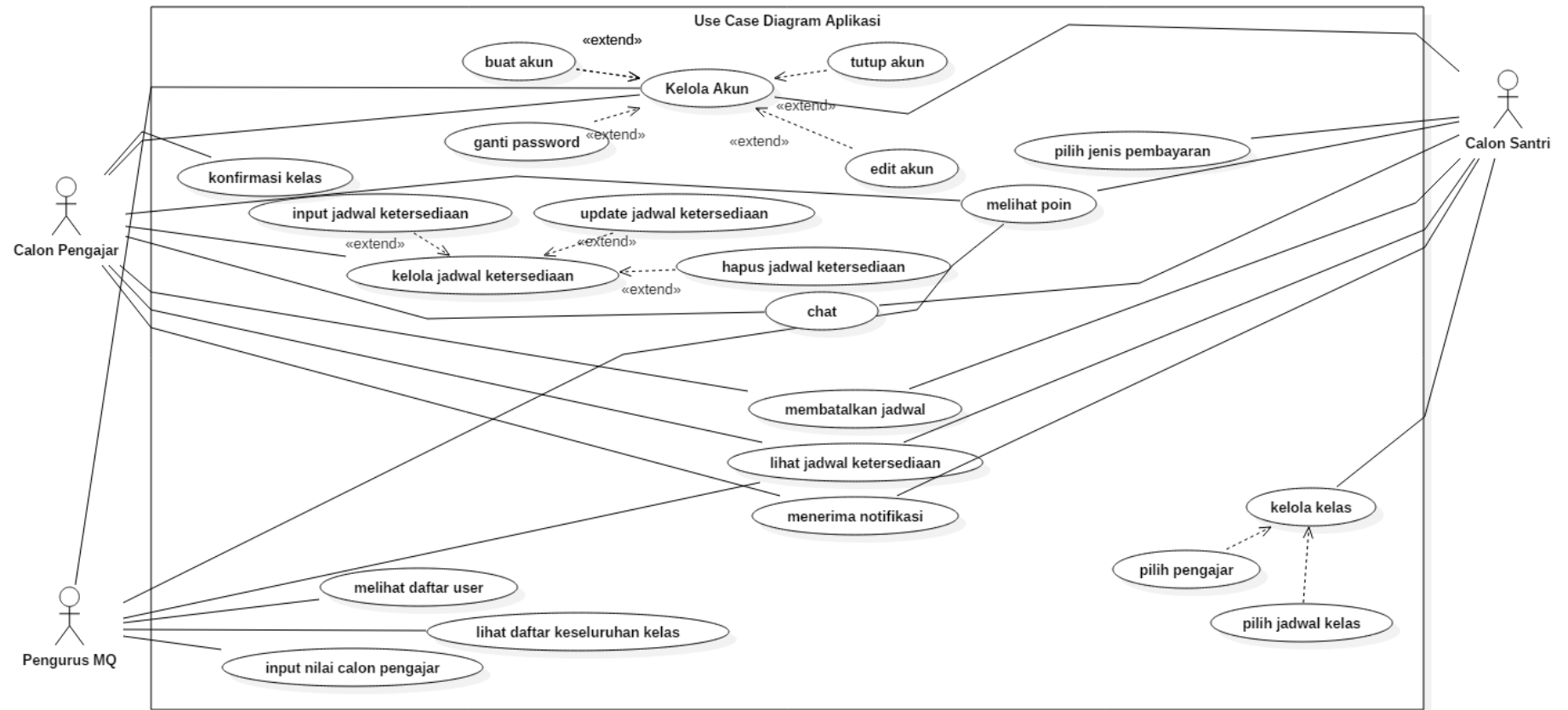
Adapun perancangan aplikasi dengan pemodelan berbasis objek, perancangan basis data dan perancangan antar muka adalah sebagai berikut.

### 3.2.1 Model Aplikasi Berbasis Objek

Pemodelan aplikasi berbasis objek menggunakan *tools* UML. Adapun pemodelan aplikasi adalah sebagai berikut.

#### 3.2.1.1 Use Case Diagram

Berikut merupakan Use Case Diagram yang menggambarkan hak akses dalam setiap fungsionalitas (Use Case) dalam perancangan aplikasi:



Gambar 3 . 6 Use Case Diagram Aplikasi

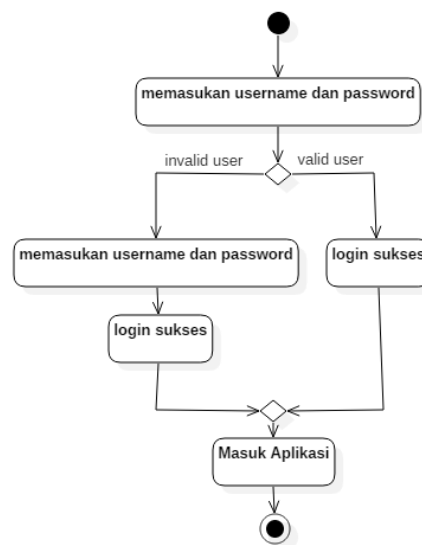
Berikut merupakan penjelasan gambar diatas:

Use Case *login* tidak dimasukan karena sudah menjadi syarat untuk setiap Use Case dan semua *user* pasti melakukan *login*.

### 3.2.1.2 Activity Diagram

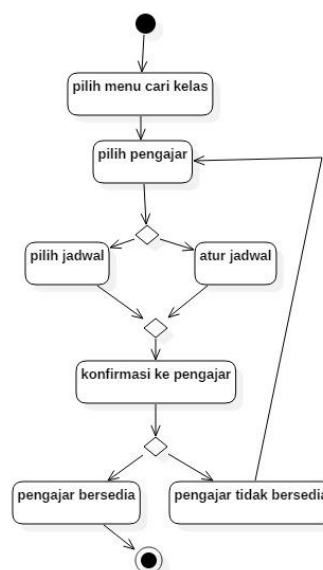
Berikut merupakan pendefinisian *Use Case* dari aplikasi dari masing-masing usecase yang telah didefinisikan sebelumnya menggunakan *Activity diagram*:

Use Case: Login



**Gambar 3 . 7 Activity Diagram Login**

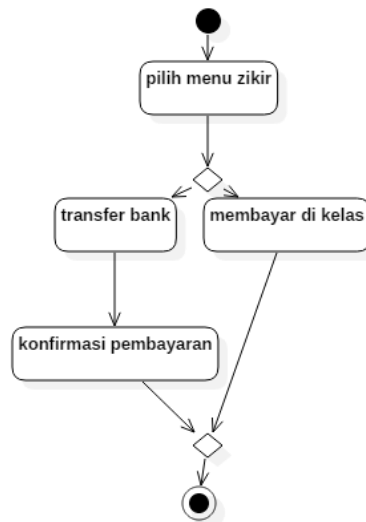
Use Case: Kelola Kelas



**Gambar 3 . 8 Activity Diagram Kelola Kelas**

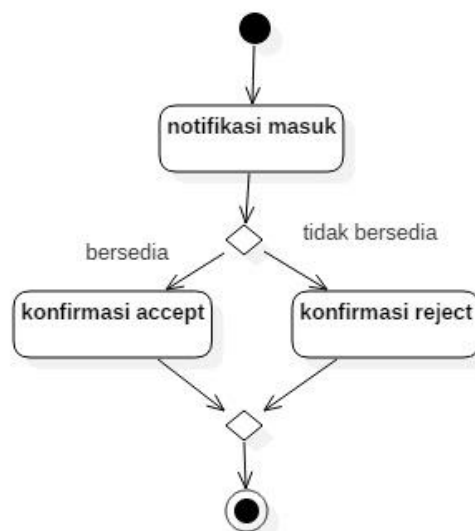


Use case: pilih jenis pembayaran



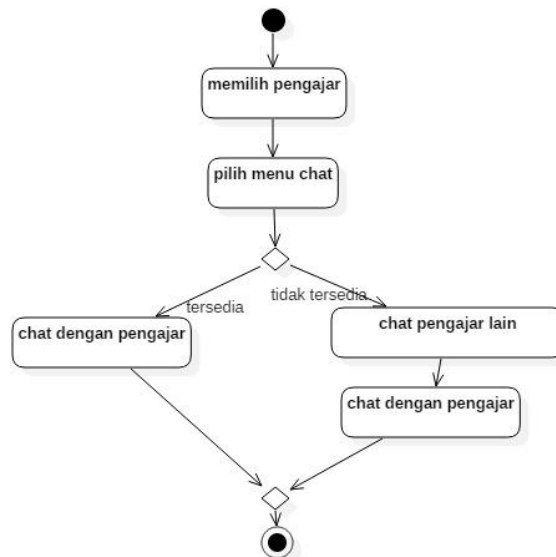
Gambar 3 . 9 Activity Diagram Pilih Jenis Pembayaran

Use Case: konfirmasi kelas



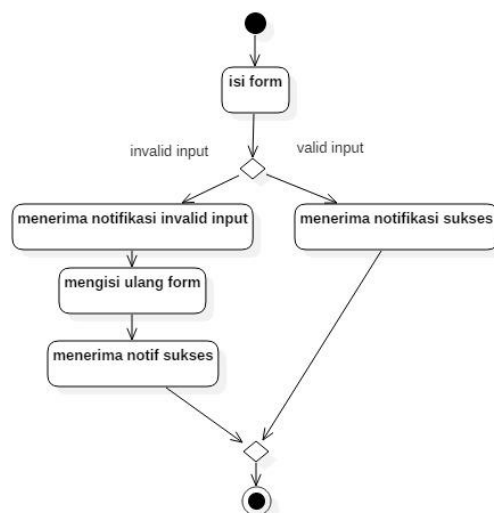
Gambar 3 . 10 Activity Diagram Konfirmasi Kelas

### Use Case: Chat



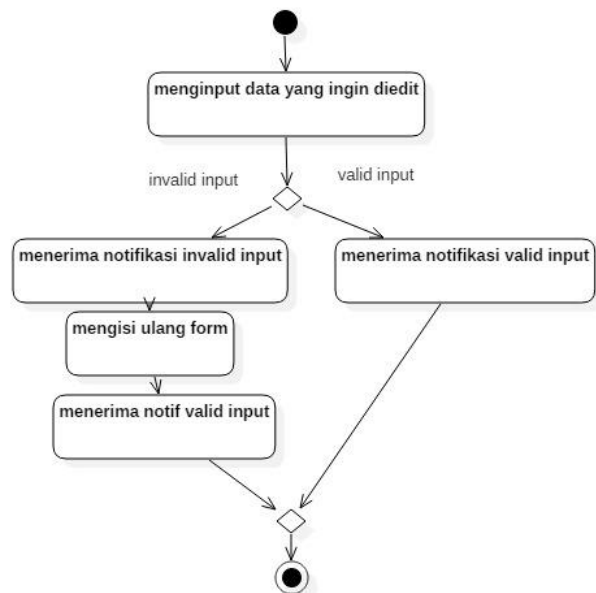
**Gambar 3 . 11 Activity Diagram Chat**

### Use Case: Buat Akun



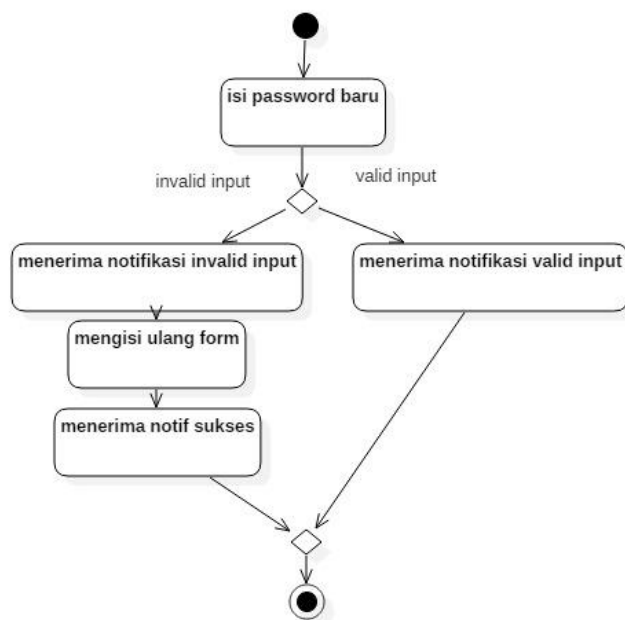
**Gambar 3 . 12 Activity Diagram Membuat Akun**

### Use Case: Edit Akun



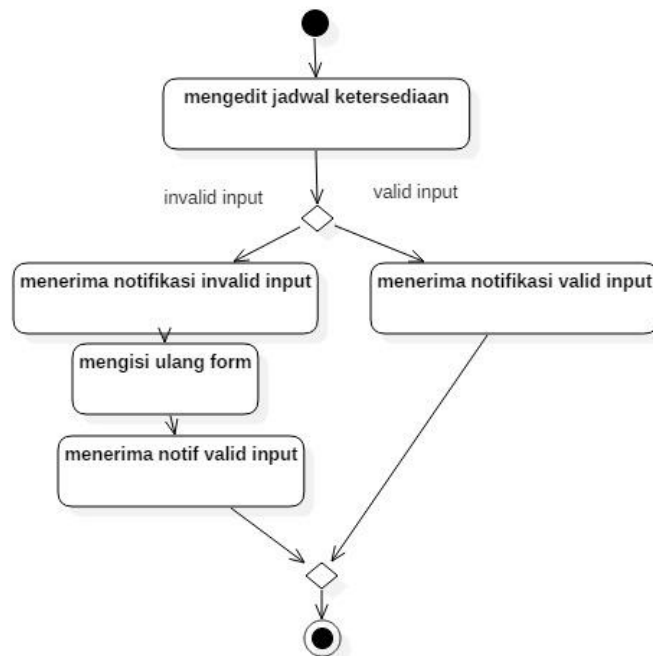
**Gambar 3 . 13 Activity Diagram Edit Akun**

### Use Case: Ganti Password



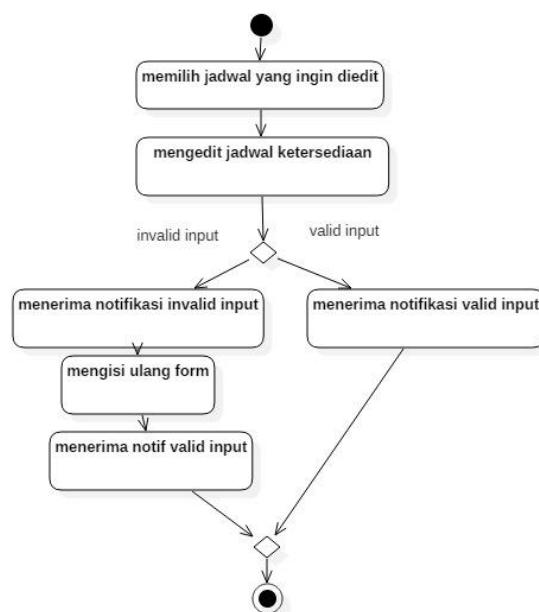
**Gambar 3 . 14 Activity Diagram Ganti Password**

### Use Case: Input Jadwal Ketersediaan



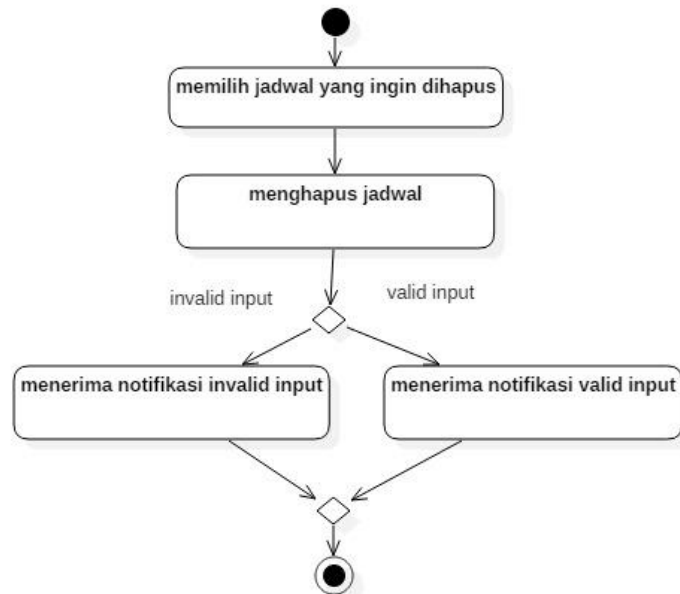
**Gambar 3 . 15 Activity Diagram Input Jadwal Ketersediaan**

### Use Case: Update Jadwal Ketersediaan



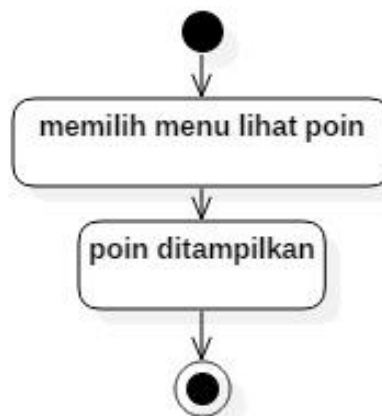
**Gambar 3 . 16 Activity Diagram Update Jadwal Ketersediaan**

Use Case: Hapus Jadwal Ketersediaan



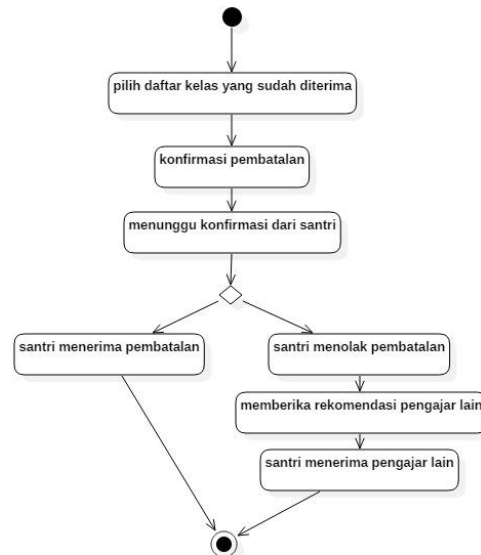
**Gambar 3 . 15 Activity Diagram Hapus Jadwal Ketersediaan**

Use Case: Melihat Poin



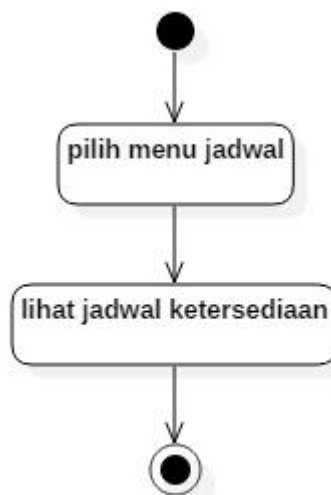
**Gambar 3 . 16 Activity Diagram Melihat Poin**

Use Case: Membatalkan Jadwal



Gambar 3 . 19 Activity Diagram Membatalkan Jadwal

Use Case: Lihat Jadwal Ketersediaan



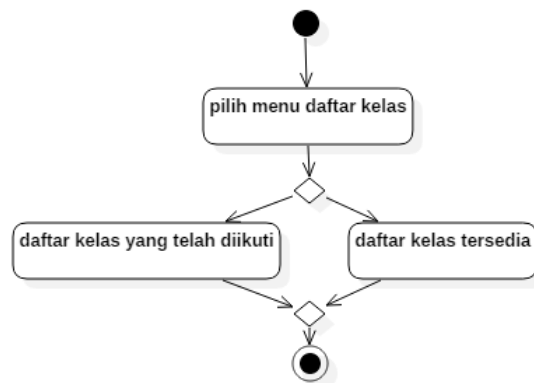
Gambar 3 . 17 Activity Diagram Lihat Jadwal Ketersediaan

Use Case: Menerima Notifikasi



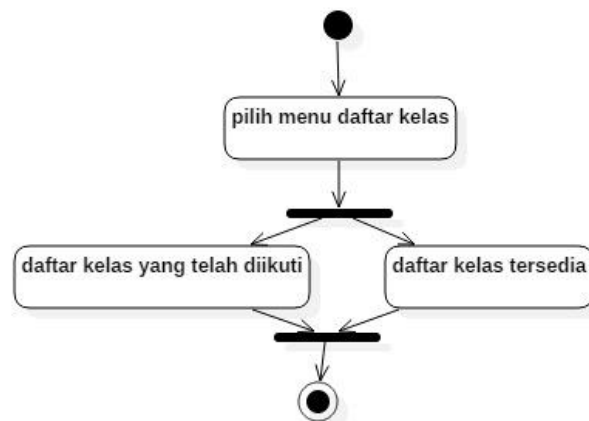
Gambar 3 . 18 Activity Diagram Menerima Notifikasi

Use Case: Melihat Daftar User



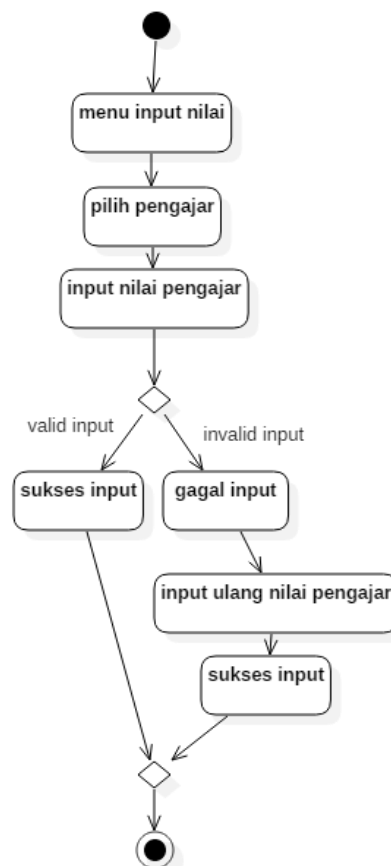
Gambar 3 . 19 Activity Diagram Melihat Daftar User

Use Case: Lihat Daftar Keseluruhan Kelas



Gambar 3 . 20 Activity Diagram Lihat Daftar Keseluruhan Kelas

Use Case: Input Nilai Calon Pengajar

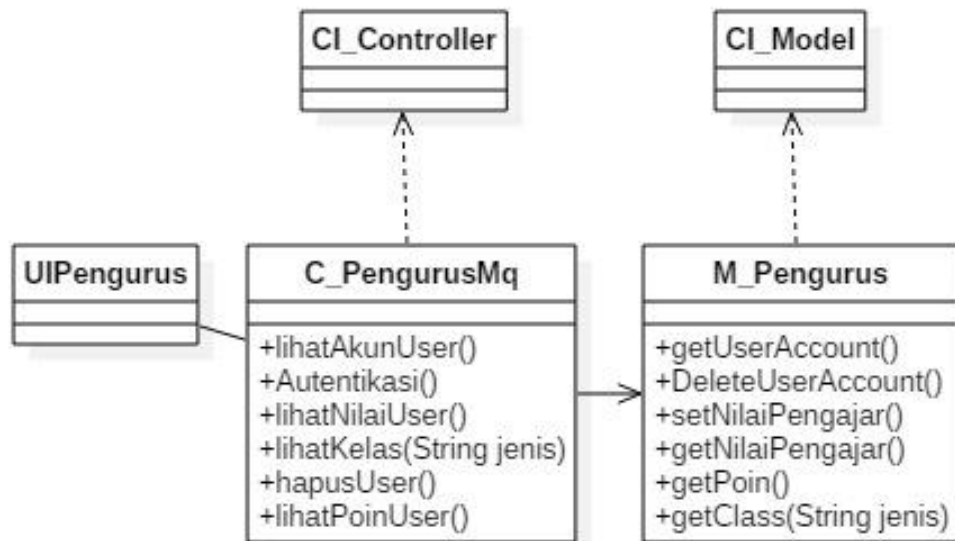


Gambar 3 . 21 Activity Diagram Input Nilai Calon Pengajar



### 3.2.1.3 Class Diagram

Berikut merupakan gambaran struktur kelas aplikasi *website* digambarkan dengan Class Diagram:

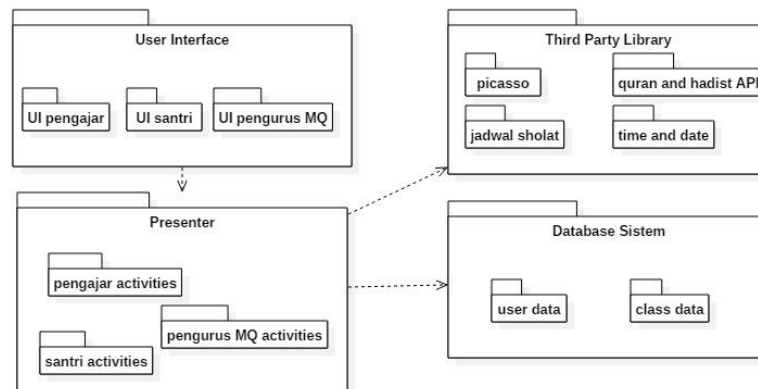


Gambar 3 . 22 Class Diagram Aplikasi

Aplikasi *website* diperuntukan kepada Pengurus MQMSU tidak kepada santri dan pengajar. Pada gambar diatas menunjukkan beberapa kelas, CI\_Controller dan CI\_Model merupakan kelas bawaan dari CodeIgniter yang harus menjadi kelas *parent*. Framework CodeIgniter menggunakan prinsip MVC (*model-view-controller*) sehingga untuk struktur dari kelas dapat terlihat jelas. Pada gambar diatas, *view* disatukan didalam kelas UIPengurus agar penulisan dapat lebih mudah untuk dimengerti.

### 3.2.1.4 Package Diagram

Berikut merupakan Package Diagram dari aplikasi yang menggambarkan pengelompokan struktur aplikasi secara umum:

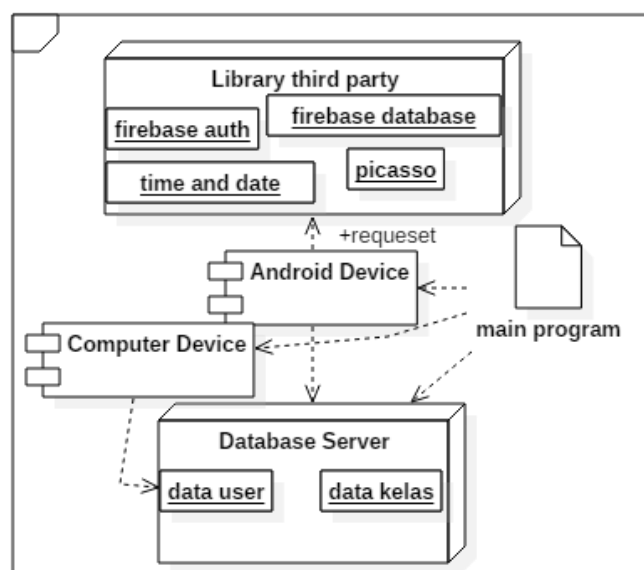


Gambar 3 . 23 Package Diagram Aplikasi

Terdapat empat package yang menggambarkan struktur aplikasi. Pada package *User Interface* membungkus tiga *sub-package* yang berisi tampilan dari aplikasi yang semuanya memiliki ketergantungan terhadap *package Presenter*. Pada *package Presenter* yang membungkus aktifitas-aktifitas dari setiap *user* dan memiliki ketergantungan terhadap *package Third Party Library* yang berisi *Library* pihak ketiga yang memudahkan pengembangan/tampilan dari aplikasi dan *package Database sistem* yang berfungsi sebagai penyimpanan data dari *users*.

### 3.2.1.5 Deployment Diagram

Berikut merupakan Deployment Diagram dari aplikasi yang menggambarkan hubungan antara perangkat keras dan perangkat lunak:



Gambar 3 . 24 Deployment Diagram Aplikasi

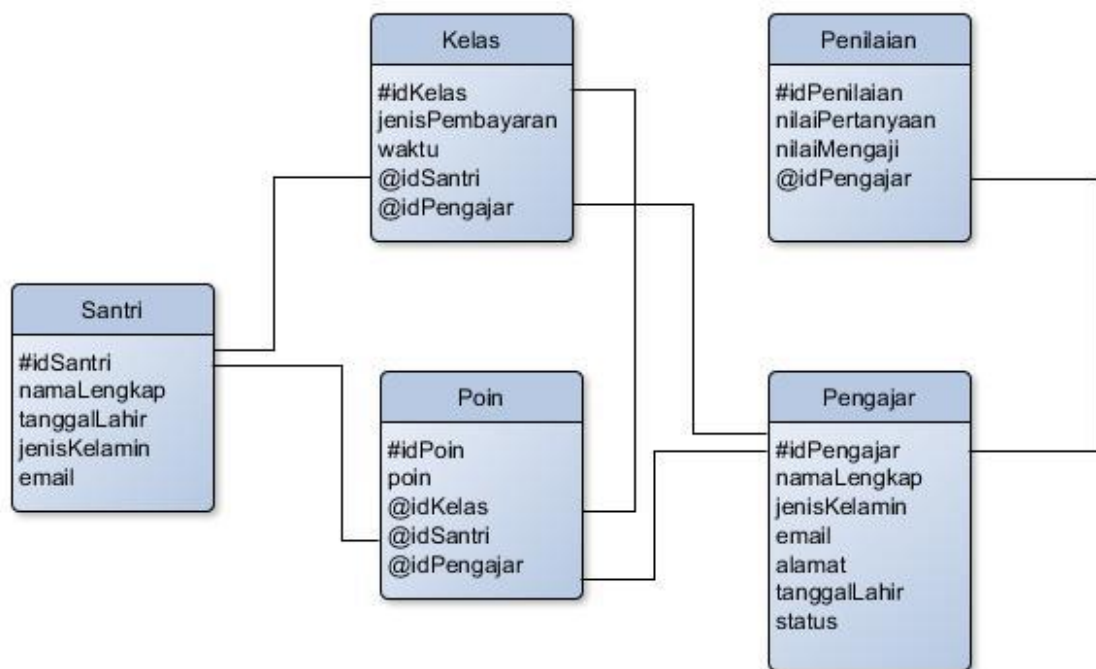


Adapun penjelasan hubungan antar tabel adalah sebagai berikut:

1. Pada tabel santri memiliki hubungan dengan tabel poin dimana satu santri dapat memiliki banyak poin dan satu poin hanya dapat dimiliki oleh satu santri.
2. Pada tabel santri memiliki hubungan dengan tabel kelas dimana santri dapat memesan satu kelas dan satu kelas hanya bisa dipesan oleh satu santri.
3. Pada tabel pengajar memiliki hubungan dengan tabel kelas dimana satu pengajar dapat membuka banyak kelas dan satu kelas hanya dapat dibuka oleh satu pengajar.
4. Tabel pengajar memiliki hubungan dengan tabel poin dimana satu pengajar dapat memiliki banyak poin dan satu poin hanya dapat dimiliki oleh satu pengajar.
5. Pada tabel pengajar memiliki hubungan dengan tabel penilaian dimana satu pengajar merujuk kepada satu penilaian dan satu hanya dapat dirujuk oleh satu pengajar. Tabel penilaian berpengaruh terhadap status pengajar, apakah pengajar dapat dikatakan pengajar atau masih calon pengajar.

#### **3.2.2.2 Skema Relasi**

Berikut merupakan gambaran skema relasi yang menggambarkan hubungan antar tabel:



Gambar 3 . 29 Skema Relasi Aplikasi

Adapun penjelasan dari skema relasi adalah sebagai berikut:

1. Tabel kelas memiliki *foreign key* idSantri dan idPengajar karena kelas akan diisi oleh pengajar dan santri. Hubungan tabel kelas dengan tabel santri dan tabel pengajar adalah satu banyak terhadap tabel kelas.
2. Pada tabel poin memiliki *foreign key* idKelas, idSantri dan idPengajar karena tabel kelas, santri dan pengajar merupakan tabel yang memiliki hubungan satu banyak terhadap tabel poin.
3. Tabel penilaian memiliki *foreign key* idPengajar, hubungan tabel pengajar adalah satu banyak terhadap tabel penilaian.

### 3.2.2.3 Struktur Tabel

Terdapat 5 tabel yang digunakan pada aplikasi yaitu, tabel santri, tabel kelas, tabel poin, tabel penilaian dan tabel pengajar. Berikut merupakan struktur tabel dari aplikasi:

PK = Primary Key

FK = Foreign Key

### 3.2.2.3.1 Santri

Tabel santri merupakan tabel yang menyimpan data diri santri. Tabel santri akan menyimpan data ketika santri melakukan registrasi atau jika menggunakan Firebase Authentication data santri diinput ketika santri mengedit profil.

**Tabel 3 . 8 Struktur Tabel Santri**

Tabel Santri				
Atribut	Tipe Data	Nullable	Keterangan	Length
idSantri	Number		PK	11
namaLengkap	Varchar			30
tanggalLahir	Date			-
jenisKelamin	Enum('Laki-laki','Perempuan')			-

### 3.2.2.3.2 Kelas

Tabel kelas menyimpan data dari kelas yang telah dilaksanakan atau belum dilaksanakan. Jika telah dilaksanakan maka semua kolom akan diisi oleh data sedangkan jika belum dilaksanakan maka kolom jenis pembayaran dan idsantri akan kosong.

**Tabel 3 . 9 Struktur Tabel Kelas**

Tabel Kelas				
Atribut	Tipe Data	Nullable	Keterangan	Length
idKelas	Number		PK	11
jenisPembayaran	Enum('transfer','bayar ditempat')	√		30
Waktu	Date			-
idSantri	Number	√	FK	11
idPengajar	Number		FK	11

### 3.2.2.3.3 Poin

Tabel poin merupakan tabel yang menyimpan poin dari *user*. Poin akan didapatkan ketika user telah melakukan suatu kegiatan, contohnya telah melakukan kegiatan belajar mengajar.

**Tabel 3 . 10 Struktur Tabel Poin**

Tabel Poin				
Atribut	Tipe Data	Nullable	Keterangan	Length
idPoin	Number		PK	11
Poin	Number	√		11
idKelas	Number		FK	11
idSantri	Number	√	FK	11
idPengajar	Number	√	FK	11

### 3.2.2.3.4 Penilaian

Tabel penilaian merupakan tabel yang menyimpan data penilaian dari pengajar. Tabel penilaian digunakan ketika pengurus MQ melakukan proses rekrutasi pengajar.

**Tabel 3 . 11 Struktur Tabel Penilaian**

Tabel Penilaian				
Atribut	Tipe Data	Nullable	Keterangan	Length
idPenilaian	Number		PK	11
nilaiPertanyaan	Number	√		3
nilaiMengaji	Number	√		3
idPengajar	Number	√	FK	11

### 3.2.2.3.5 Pengajar

tabel pengajar merupakan tabel yang menyimpan data pengajar. Data dari tabel diinput ketika mendaftar, lalu kolom status akan berubah ketika calon pengajar telah melalui seluruh rangkaian rekrutasi dan dinyatakan layak menjadi pengajar.

**Tabel 3 . 12 Struktur Tabel Pengajar**

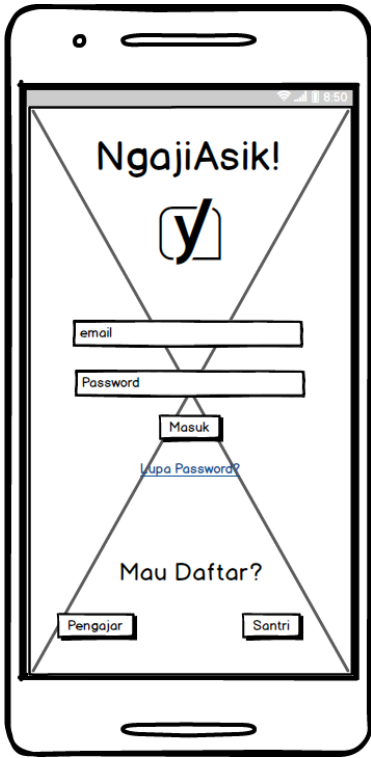
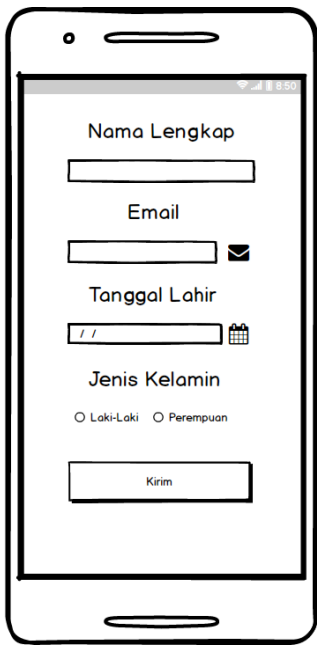
Tabel Pengajar				
Atribut	Tipe Data	Nullable	Keterangan	Length
idPengajar	Number		PK	11
namaLengkap	Varchar			30
jenisKelamin	Enum('Laki-laki','perempuan')			-
email	Varchar	√		30
alamat	Varchar			225
tanggalLahir	Date	√		-
Status	Enum('calon pengajar','pengajar')			-

### 3.2.3 Perancangan Antar Muka

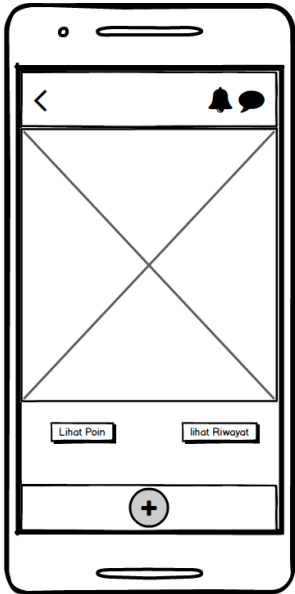
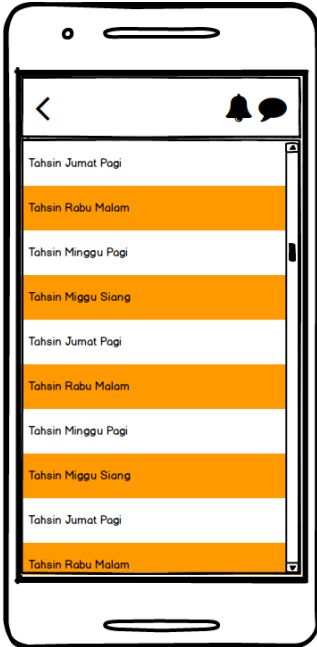
Adapun perancangan Antar Muka dari aplikasi berdasarkan proses bisnis utama adalah sebagai berikut:

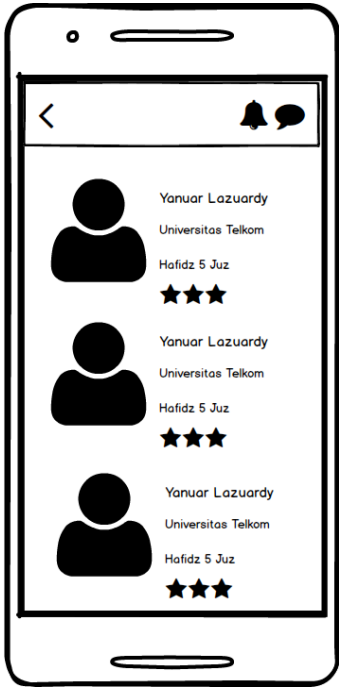
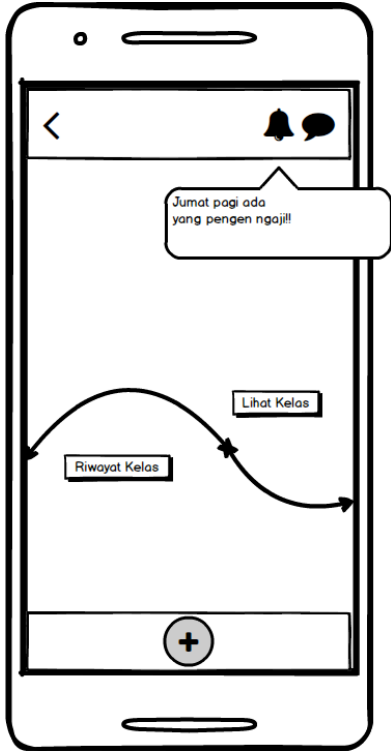


**Tabel 3 . 13 Perancangan Antar Muka Aplikasi**

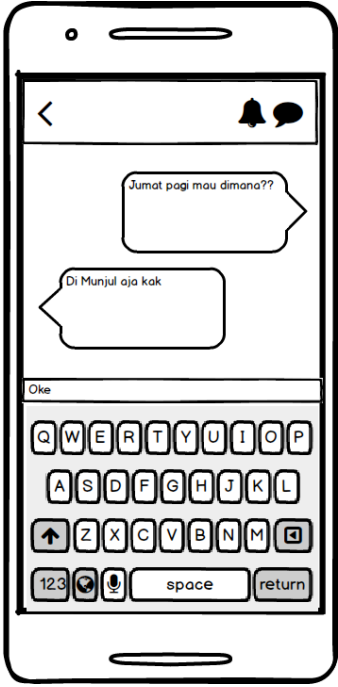
Gambar	Keterangan
	<p>Tampilan awal dari aplikasi, pada layar tersebut disajikan login dan daftar, pendaftaran dibagi dua yaitu sebagai pengajar dan sebagai santri</p>
	<p>Tampilan berikut merupakan layar untuk mengisi form data diri calon pengajar, untuk password akan dikirimkan melalui email</p>

Gambar	Keterangan
	<p>Sama seperti sebelumnya tampilan pendaftaran santri hanya memiliki sedikit isian, untuk password juga akan dikirimkan melalui email</p>
	<p>Diasumsikan proses registrasi telah selesai. Selanjutnya, pada dashboard pengajar ada tombol utama untuk menambahkan ketersediaan kelas. Dua icon diatas menunjukan notifikasi dan chat</p>

Gambar	Keterangan
	<p>Gambar berikut merupakan tampilan dari dashboard santri dimana tombol utama adalah untuk mencari kelas.</p>
	<p>Berikut merupakan tampilan jadwal yang tersedia, jadwal ini merupakan jadwal yang telah dibuat oleh pengajar.</p>

Gambar	Keterangan
	<p>Selanjutnya adalah daftar pengajar beserta informasi yang dibutuhkan.</p>
	<p>Pengajar akan menerima notifikasi apabila ada santri yang ingin belajar sesuai jadwal yang disediakan oleh pengajar.</p>

Gambar	Keterangan
	<p>Apabila setuju maka akan langsung ke fitur chat untuk negosiasi tempat. Apakah ingin ditempat yang telah disediakan oleh pengajar (MSU)</p>
	<p>Apabila telah selesai maka akan keluar alert pilihan pembayaran pada layar santri. Apabila transfer maka akan di direct ke layar rekening tujuan. Dan apa bila tidak maka pembayaran dapat dilakukan ketika kelas berlangsung</p>

Gambar	Keterangan
	<p>Selanjutnya dilakukan chat untuk negosiasi terkait tempat pelaksanaan kelas.</p>

## BAB 4

### IMPLEMENTASI DAN PENGUJIAN

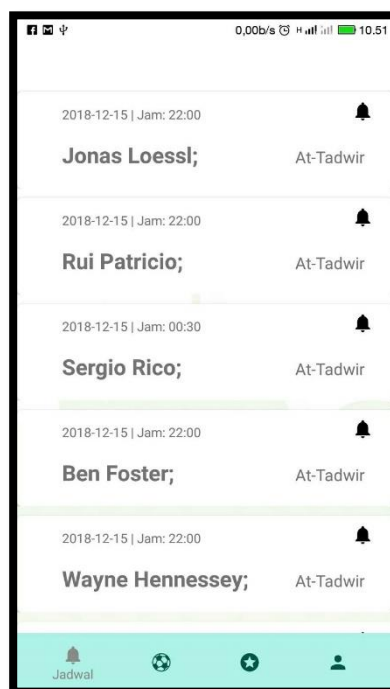
Pada bab ini membahas mengenai implementasi dan pengujian dari aplikasi yang dibangun.

#### 4.1 Implementasi

##### 4.1.1 Pemesanan Kelas

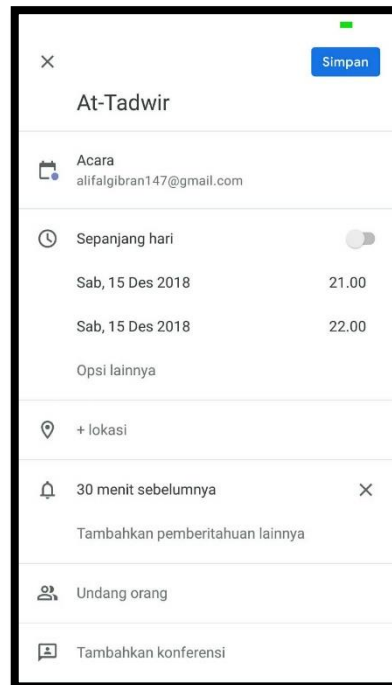
Pada pemesanan kelas terdapat beberapa bagian yang dapat dilakukan oleh pengguna seperti daftar jadwal tersedia, atur notifikasi, dan pesan kelas.

Gambar 4.1 merupakan tampilan daftar jadwal yang tersedia. Pengguna dapat melihat jadwal daftar yang tersedia dan dapat dipesan oleh santri.



Gambar 4 . 1 Tampilan Daftar Kelas Tersedia

Gambar 4.2 merupakan tampilan menambahkan *event* agar terdapat notifikasi ketika kelas akan dilaksanakan.



**Gambar 4 . 2 Tampilan Menambahkan Event**

Gambar 4.3 merupakan tampilan memesan kelas, menggunakan *shared preferences* dari Whatsapp



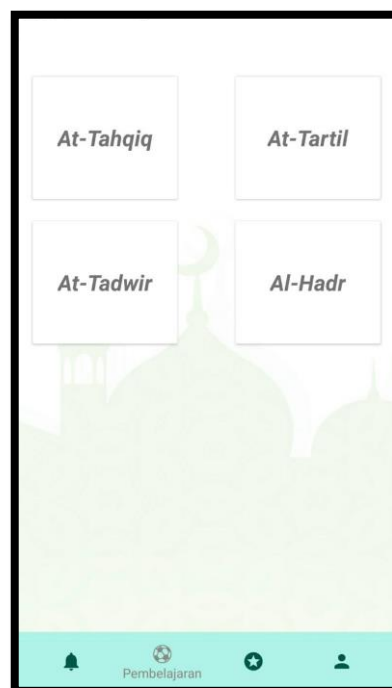
**Gambar 4 . 3 Tampilan Memesan Kelas**



#### 4.1.2 Tampilan Jenis Kelas

Tampilan jenis kelas menjelaskan tentang kelas yang dapat dipesan oleh santri. Tampilan jenis kelas terbagi beberapa tampilan seperti deksripsi jenis kelas serta daftar pengajar yang berkompeten dalam mengajar kelas tersebut.

Gambar 4.4 menampilkan empat bagian kelas yang dapat dipesan oleh santri berdasarkan kemampuan membaca.



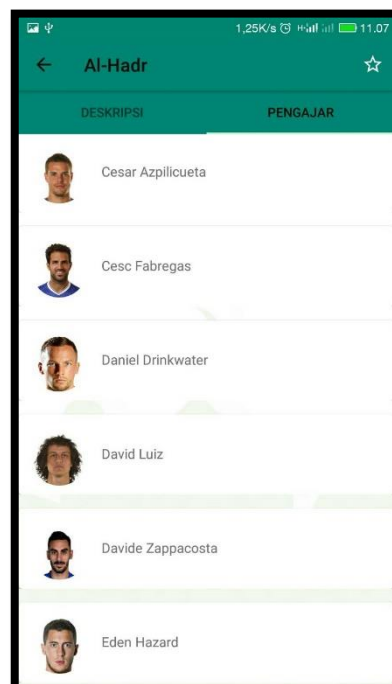
**Gambar 4 . 4 Tampilan Jenis Kelas**

Gambar 4.5 menjelaskan deskripsi dari jenis kelas yang dapat dipilih oleh santri.



Gambar 4 . 5 Tampilan Deskripsi Jenis Kelas

Gambar 4.6 menampilkan daftar pengajar yang berkompeten pada kelas yang terkait.

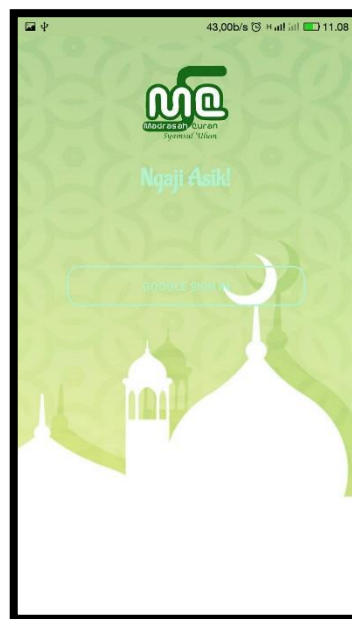


Gambar 4 . 6 Tampilan Daftar Pengajar

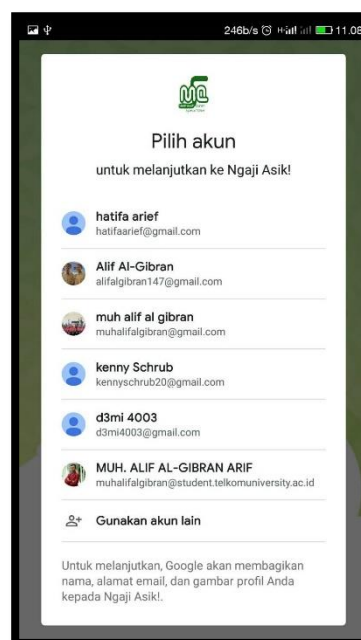
### 4.1.3 Kelola Akun

Kelola akun pada aplikasi memungkinkan pengguna untuk mengelola akun yang telah didaftarkan.

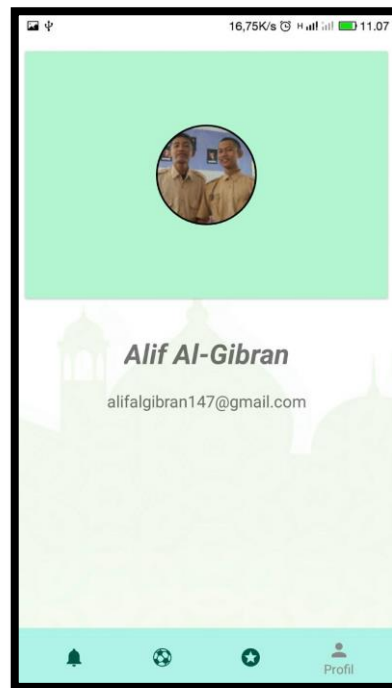
Gambar 4.7 – 4.9 merupakan tampilan login menggunakan akun gmail. Sehingga memudahkan santri agar dalam melakukan registrasi.



**Gambar 4 . 8 Tampilan Login**



**Gambar 4 . 7 Tampilan Login Menggunakan Akun Gmail**



Gambar 4 . 9 Tampilan Profil

## 4.2 Pengujian

Pengujian aplikasi dilakukan dengan dua cara yaitu *black box testing*. Berikut merupakan hasil dari kedua jenis pengujian tersebut.

### 4.2.1 *Black box Testing*

Berikut merupakan pengujian menggunakan *black box testing* pada aplikasi.

#### 4.2.1.1 Fungsionalitas Kelola Kelas

Fungsionalitas kelola kelas memungkinkan user untuk melihat kelas tersedia, membuat *event*, dan melakukan pemesanan kelas.

##### 4.2.1.1.1 Lihat Kelas Tersedia

Pengujian ini dilakukan untuk mengetahui sistem telah menampilkan kelas yang tersedia.

Tabel 4 . 1 Pengujian Lihat Jadwal Kelas Tersedia

<b>Nama Kelas Uji</b>	Pengujian fungsi lihat jadwal kelas yang tersedia
<b>Tujuan</b>	Memastikan jadwal kelas yang tersedia tampil pada layar
<b>Kondisi awal</b>	Layar kosong sebelum <i>request</i> data
<b>Skenario</b>	[1] Cek apakah Sistem menampilkan fragment

		[2] <i>Recyclerview</i> pada fragment menampilkan daftar kelas yang tersedia		
Hasil				
No	Use case	Test Data (Input)	Expected Result	Hasil Pengujian
1	Lihat data kelas yang tersedia	-	Menampilkan daftar kelas yang tersedia	Daftar kelas yang terseda tampil pada fragment

Dari hasil pengujian di atas dapat disimpulkan bahwa sistem telah mampu untuk menampilkan data kelas yang tersedia.

#### 4.2.1.1.2 Membuat Event Pengingat

Pengujian ini dilakukan untuk mengetahui apakah sistem mampu membuat *event* pengingat kelas yang akan dilaksanakan.

Tabel 4 . 2 Pengujian Fungsi Membuat Event

Nama Kelas Uji	Pengujian fungsi membuat event pengingat kelas yang akan dilaksanakan			
Tujuan	Memastikan <i>event</i> dibuat			
Kondisi awal	Daftar kelas yang tersedia tampil pada layar			
Skenario	[1] Cek apakah Sistem menampilkan <i>fragment</i> [2] <i>Recyclerview</i> pada fragment menampilkan daftar kelas yang tersedia [3] Menekan icon notifikasi [4] Intent ke google calendar [5] Event dibuat			
Hasil				
No	Use case	Test Data (Input)	Expected Result	Hasil Pengujian
1	Menekan icon notifikasi	-	Event dapat dibuat setelah melakukan pemesanan	Event ter-create sehingga terdapat notifikasi apabila kelas akan dilaksanakan

Pada tabel pengujian di atas, dapat disimpulkan bahwa hasil pengujian berhasil sehingga event dapat dibuat oleh santri.

#### 4.2.1.1.3 Pemesanan Kelas

Pengujian ini dilakukan untuk mengetahui apakah sistem mampu mengakomodasi permintaan santri dalam melakukan pemesanan kelas.

**Tabel 4 . 3 Pengujian Melakukan Pemesanan Kelas**

Nama Kelas Uji	Pengujian fungsi melakukan pemesanan kelas			
Tujuan	Memastikan kelas dapat dipesan			
Kondisi awal	Daftar kelas yang tersedia tampil pada layar			
Skenario	[1] Cek apakah Sistem menampilkan <i>fragment</i> [2] <i>Recyclerview</i> pada fragment menampilkan daftar kelas yang tersedia [3] Menekan <i>cardview</i> atau daftar kelas yang tersedia [4] Intent keaplikasi WhatsApp			
Hasil				
No	Use case	Test Data (Input)	Expected Result	Hasil Pengujian
1	Menekan <i>cardview</i>	-	Kelas dapat dipesan oleh santri	<i>Carview</i> yang ditampilkan dapat diklik sehingga intent ke aplikasi WhatsApp

#### 4.2.1.2 Fungsionalitas Menampilkan Jenis Kelas

Pada fungsionalitas menampilkan jenis kelas ini terdapat dua sub-fungsi yang memungkinkan santri untuk melihat penjelasan jenis kelas serta melihat daftar pengajar.

##### 4.2.1.2.1 Menampilkan Penjelasan Jenis Kelas

Pengujian ini bertujuan untuk menguji sistem apakah mampu daam menampilkan penjelasan dari jenis kelas.

**Tabel 4 . 4 Pengujian Menampilkan Penjelasan Jenis Kelas**

<b>Nama Kelas Uji</b>	Pengujian fungsi menampilkan penjelasan jenis kelas			
<b>Tujuan</b>	Memastikan penjelasan kelas tampil pada layar			
<b>Kondisi awal</b>	Tampilan empat jenis kelas			
<b>Skenario</b>	[1] Cek apakah Sistem menampilkan empat jenis kelas [2] klik salah satu <i>cardview</i>			

		[3] Menampilkan penjelasan dari jenis kelas yang tersedia		
Hasil				
No	Use case	Test Data (Input)	Expected Result	Hasil Pengujian
1	Menekan <i>cardview</i>	-	Penjelasan kelas dapat ditampilkan	<i>Carview</i> berhasil diklik sehingga menampilkan penjelasan kelas yang tersedia

Berdasarkan tabel pengujian di atas dapat disimpulkan bahwa sistem mampu untuk menampilkan penjelasan dari jenis kelas yang tersedia.

#### 4.2.1.2.2 Menampilkan Daftar Pengajar

Pada pengujian ini, sistem diharapkan mampu untuk menampilkan daftar pengajar yang berkompeten sesuai dengan jenis kelas.

Tabel 4 . 5 Pengujian Menampilkan Daftar Pengajar

Nama Kelas Uji		Pengujian fungsi menampilkan daftar pengajar		
Tujuan		Memastikan daftar pengajar tampil pada layar		
Kondisi awal		Tampilan empat jenis kelas		
Skenario		[1] Cek apakah Sistem menampilkan empat jenis kelas [2] klik salah satu <i>cardview</i> [3] menampilkan penjelasan kelas [4] <i>action swipe left</i> untuk menampilkan daftar pengajar yang tersedia		
Hasil				
No	Use case	Test Data (Input)	Expected Result	Hasil Pengujian
1	Menekan <i>cardview</i>	-	Daftar pengajar dapat dilihat oleh santri	Daftar pengajar dapat ditampilkan oleh sistem sehingga santri dapat melihat daftar pengajar yang tersedia sesuai kompetensi masing-masing

Berdasarkan tabel diatas, dapat ditarik kesimpulan bahwa sistem dapat mengakomodir permintaan user yaitu menampilkan daftar pengajar yang tersedia.



## BAB 5

### KESIMPULAN DAN SARAN

---

#### 5.1 Kesimpulan

Berdasarkan hasil penelitian selama proses pengumpulan dan pengolahan data, analisis sistem, dan implementasi sistem usulan maka dapat diambil kesimpulan sebagai berikut:

1. Dengan membuat aplikasi Ngaji Asik! Diharapkan pengajar dan santri dapat melakukan proses belajar membaca Al-Quran dengan baik.
2. Santi dapat mengetahui informasi terkait pengajar yang berkompeten.
3. Pengajar dan santri dapat membuat *schedule* yang memungkinkan kelas dapat terlaksana.

#### 5.2 Saran

Saran yang diberikan untuk pengembangan aplikasi Ngaji Asik! untuk pengembangan selanjutnya adalah sebagai berikut:

1. sebaiknya dilakukan pemeliharaan aplikasi secara berkala, sehingga aplikasi dapat berjalan dengan baik.
2. Fungsionalitas aplikasi dapat ditambahkan sehingga tidak monoton hanya pemesanan kelas.

## DAFTAR PUSTAKA

---

- [1] A. Hidayat, Bekasi Selatan: Institut Quantum Akhyar, 2018.
- [2] J. Enterprise, "Apa Itu Android ?," in *Mengenal Dasar-Dasar Pemrograman Android*, Jakarta, Elex Media Komputindo, 2015, p. 1.
- [3] S. Chan, Membuat Aplikasi Database dengan PowerBuilder 12.6 dan MySQL, Elex Media Komputindo, 2017, 2017.
- [4] R. W. A. R. Novhirtamely Kahar, "APLIKASI PEMESANAN MAKANAN ONLINE BERBASIS WEB," *Jurnal Informatika*, vol. VII, no. 2, p. 794, 2013.
- [5] Y. C. Qing Li, Entity-Relationship Diagram. In: *Modeling and Analysis of Enterprise and Information Systems*, Berlin: Springer, 2010.
- [6] D. Pilone, UML 2.0 Pocket Reference, O'Reilly Media Inc, 2006.
- [7] N. S. S. Bhattacharya, "Formalization of UML use case diagram-a Z notation based approach," in *ICOCI'06. International Conference on (pp. 1-6) IEEE*, Kuala Lumpur, 2006.
- [8] Balsamiq Studios, "Balsamiq Mockups 3 Application Overview," Balsamiq, 2008. [Online]. Available: <https://docs.balsamiq.com/desktop/overview/>. [Accessed 23 October 2018].
- [9] yWorks , "The Company," yWorks the Diagram Company, 2018. [Online]. Available: <https://www.yworks.com/company>. [Accessed 23 October 2018].
- [10] MKLab, "StarUML documentation-Introduction," MKLab Co., Ltd., 2018 August 17. [Online]. Available: <https://docs.staruml.io/user-guide/basic-concepts>. [Accessed 23 October 2018].
- [11] J. Team, "Using Kotlin for Android Development," JetBrains, Mei 2017. [Online]. Available: <https://kotlinlang.org/docs/reference/android-overview.html>. [Accessed 23 October 2018].
- [12] M. Owens, "The Definitive Guide to SQLite," in *The Definitive Guide to SQLite*, Apress, 2006, pp. 1-16.

- [13] A. G. Clifton Craig, "Convention Used in This Book," in *Learn Android Studio: Build Android Apps Quickly and Effectively*, Apress, 2015, p. xxvii.
- [14] Atom, "Atom Documentation," GitHub, 2018 October 6. [Online]. Available: <https://atom.io/docs>. [Accessed 2018 October 2018].
- [15] w. komputer, "ShortCourse," in *PHP Programming, Semarang*, semarang, wahana komputer, 2009.
- [16] E. Prasetio, *Buku Sakti WEBMASTER*, Jakarta: Mediakita, 2014.
- [17] B. Sidik, *Framework Codeigniter*, Bandung: Informatika, 2012.
- [18] R. Patton, *Software Testing*, USA: SAMS Publishing, 2005.
- [19] M. Weske, *Business Process Management Concepts Languages*, Berlin: Springer, 2007.

## LAMPIRAN

---

### Lampiran 1 Hasil Wawancara Pengurus MQMSU Ulum Universitas Telkom

Wawancara dilakukan dengan saudara Haddad Alwi Yafie, beliau telah mejadi pengurus MQMSU sejak tahun 2016. Waktu dan tempat wawancara sekitar pukul 19.15 WIB bertempat di ruang utama Masjid Syamsul Ulum. Pada wawancara tersebut diajukan beberapa pertanyaan terkait kebutuhan dari MQMSU.

Pertanyaan pertama terkait proses bisnis utama MQMSU. Proses bisnis utama adalah memberikan wadah kepada mahasiswa dan masyarakat sekitar Universitas Telkom dalam belajar membaca dan memahami Al-Quran. Masyarakat yang dimaksud adalah anak berusia 5 tahun hingga 10 tahun. Implementasi dari proses bisnis berjalan dengan baik hanya kepada masyarakat sekitar, tetapi tidak berjalan sama sekali kepada mahasiswa.

Pertanyaan selanjutnya adalah mengapa proses bisnis belajar mengaji pada kalangan mahasiswa tidak berjalan? Hal ini dikarenakan pengaturan waktu serta tempat sangat sulit ditentukan oleh mahasiswa. Mahasiswa sering tersendat jadwal kuliah yang padat sehingga sulit mencari jadwal. Awalnya berjalan dengan baik, akan tetapi semakin hari semakin sulit dalam mencari jadwal. Mencari tenaga pengajar juga menjadi permasalahan.

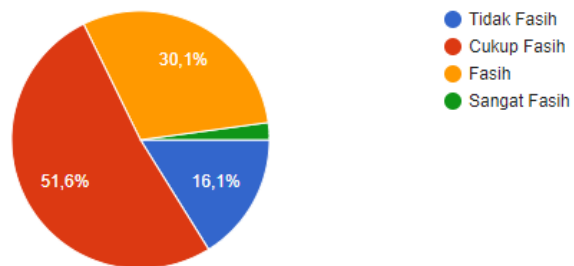
Berikutnya mengenai tata cara mahasiswa melakukan pendaftaran. Mahasiswa mendapatkan info pembelajaran Al-Quran dari beberapa *official account* LINE, setelah itu mahasiswa yang berminat selanjutnya mengisi form online terkait registrasi, lalu mahasiswa melakukan transfer kerekening tujuan dan mengirim bukti transfer sebagai bentuk pembayaran kelas, setelah pengurus menerima form dan bukti transfer maka mahasiswa akan dimasukan kedalam salah satu grup semacam mentoring yang didalamnya membahas waktu dan tempat pembelajaran.

Lalu pertanyaan selanjutnya terkait solusi permasalahan. Haddad menanggapi dengan baik adanya solusi dengan memberikan wadah berupa aplikasi yang dapat mempertemukan tenaga pengajar dan santri dengan memungkinkan melakukan pengaturan membuat janji bertemu.

## Lampiran 2 Hasil Kuesioner

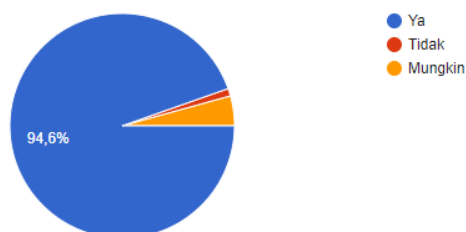
Menurut kamu, seberapa lancar kamu dalam membaca Al-Quran?

93 tanggapan



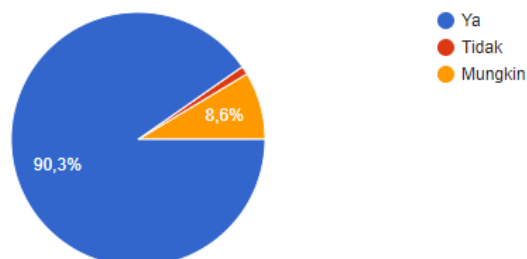
Apakah kamu tertarik dalam mempelajari Al-Quran dengan umur kamu yang sekarang?

93 tanggapan



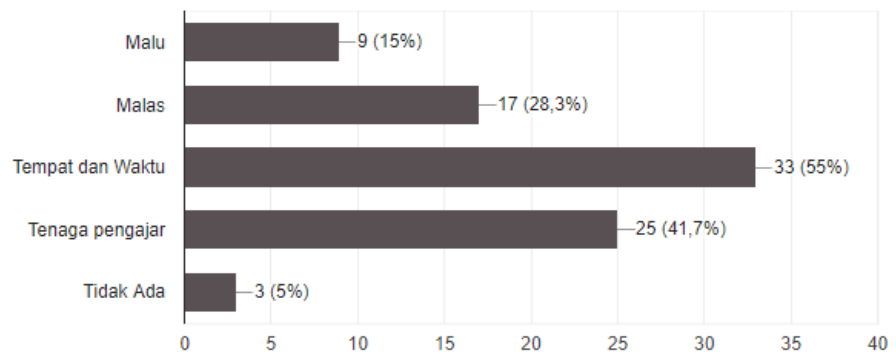
Apakah kamu ingin meluangkan waktu untuk belajar Mengaji?

93 tanggapan



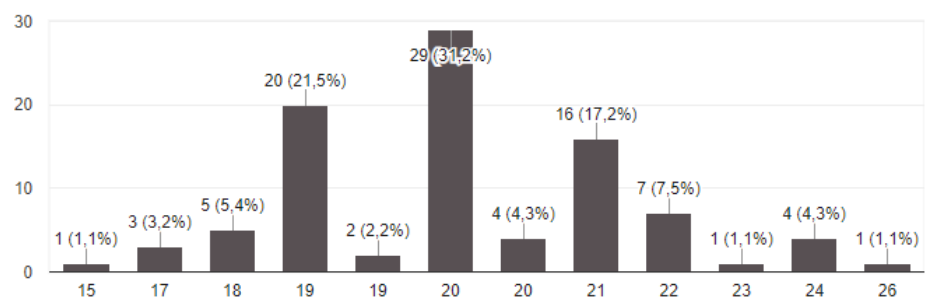
## Apa faktor yang membuat kamu terkendala dalam belajar Mengaji?

60 tanggapan



## Umur

93 tanggapan



### Lampiran 3 Struktur Organisasi MQMSU

