PENGEMBANGAN SISTEM INFORMASI PENGELOLAAN KEGIATAN DAN MONITORING PROGRAM PENGEMBANGAN DIRI QUR'ANI (PDQ) UNIVERSITAS ISLAM INDONESIA



Disusun Oleh:

N a m a : Arif Subarkah

NIM : 16523221

PROGRAM STUDI INFORMATIKA – PROGRAM SARJANA

FAKULTAS TEKNOLOGI INDUSTRI

UNIVERSITAS ISLAM INDONESIA

2020

HALAMAN PENGESAHAN DOSEN PEMBIMBING

PENGEMBANGAN SISTEM INFORMASI PENGELOLAAN KEGIATAN DAN MONITORING PROGRAM PENGEMBANGAN DIRI QUR'ANI (PDQ) UNIVERSITAS ISLAM INDONESIA

TUGAS AKHIR

Disusun Oleh:

N a m a : Arif Subarkah

NIM : 16523221

Yogyakarta, 10 Juli 2020

Pembimbing,

(Kholid Haryono, S.T, M.Kom)

HALAMAN PENGESAHAN DOSEN PENGUJI

PENGEMBANGAN SISTEM INFORMASI PENGELOLAAN KEGIATAN DAN MONITORING PROGRAM PENGEMBANGAN DIRI QUR'ANI (PDQ) UNIVERSITAS ISLAM INDONESIA

TUGAS AKHIR

Telah dipertahankan di depan sidang penguji sebagai salah satu syarat untuk memperoleh gelar Sarjana Komputer dari Program Studi Informatika di Fakultas Teknologi Industri Universitas Islam Indonesia

Yogyakarta, 21 Juli 2020

Tim Penguji

Kholid Haryono, S.T., M.Kom.

Anggota 1

Galang Prihadi Mahardhika, S.Kom., M.Kom

Anggota 2

Andhik Budi Cayhyono, S.T., M.T.

Mengetahui,
Ketua Program Studi Informatika – Program Sarjana
Fakultas Teknologi Industri
Universitas Islam Indonesia

(Dr. Raden Teduh Dirgahayu, S.T., M.Sc.)

HALAMAN PERNYATAAN KEASLIAN TUGAS AKHIR

Yang bertanda tangan di bawah ini:

Nama : Arif Subarkah

NIM : 16523221

Tugas akhir dengan judul:

PENGEMBANGAN SISTEM INFORMASI PENGELOLAAN KEGIATAN DAN MONITORING PROGRAM PENGEMBANGAN DIRI QUR'ANI (PDQ) UNIVERSITAS ISLAM INDONESIA

Menyatakan bahwa seluruh komponen dan isi dalam tugas akhir ini adalah hasil karya saya sendiri. Apabila dikemudian hari terbukti ada beberapa bagian dari karya ini adalah bukan hasil karya sendiri, tugas akhir yang diajukan sebagai hasil karya sendiri ini siap ditarik kembali dan siap menanggung resiko dan konsekuensi apapun.

Demikian surat pernyataan ini dibuat, semoga dapat dipergunakan sebagaimana mestinya.

Yogyakarta, 15 Juli 2020



(Arif Subarkah)

HALAMAN PERSEMBAHAN

"Teruntuk pembaca,

semoga apa yang ada dalam tulisan ini bisa diambil manfaatnya "

HALAMAN MOTO

"Maka sesungguhnya bersama kesulitan ada kemudahan"

(QS. Al Insyirah 94: Ayat 5)

KATA PENGANTAR

Bismillahirrahmanirrahim

Alhamdulillah dengan izin Allah SWT laporan tugas akhir *Sistem Informasi Pengelolaan Kegiatan dan Monitoring Program Pengembangan Diri Qur'an (PDQ) Universitas Islam Indonesia* selesai disusun. Laporan tugas akhir ini disusun sebagai salah satu syarat memperoleh gelar sarjana di Jurusan Informatika Universitas Islam Indonesia. Penulis mengucapkan terimakasih kepada semua pihak yang telah mendukung dan membantu sehingga penelitian tugas akhir bisa diselesaikan. Terimakasih khususnya penulis ucapkan kepada:

1. Kedua orang tua yang senantiasa memberikan dukungan dalam berbagai hal.
2. Pemerintah dan pihak Universitas Islam Indonesia yang telah memberikan saya kesempatan untuk menempuh pendidikan tinggi dengan beasiswa.
3. Prof. Fathul Wahid, S.T., M.Sc., Ph.D. sebagai Rektor Universitas Islam Indonesia.
4. Prof. Dr. Ir. Hari Purnomo., M.T. sebagai Dekan Fakultas Teknologi Industri Universitas Islam Indonesia.
5. Dr. Raden Teduh Dirgahayu, S.T., M.Sc. sebagai Ketua Ketua Prodi Studi Informatika Program Sarjana Fakultas Teknologi Industri Universitas Islam Indonesia.
6. Kholid Haryono, S.T., M.Kom. sebagai dosen pembimbing Tugas Akhir yang senantiasa memberikan dukungan dan membimbing saya dengan baik.
7. Teman-teman satu perjuangan di Pondok Pesantren Ki Ageng Giring yang selalu memberikan banyak dukungan dan pembelajaran.
8. Teman dekat saya Mufti Leriansyah dan Sandy Vrianda yang senantiasa menemani dan mendukung selama saya kuliah di jurusan Informatika.

Penulis sepenuhnya menyadari, bahwa laporan tugas akhir ini memiliki banyak kekurangan. Maka dari itu penulis memohon maaf kepada semua pihak terutama pembaca, masih terdapat kekeliruan dalam laporan ini. Penulis juga mengharapkan kritik maupun saran yang membangun dari semua pembaca. Penulis berharap laporan tugas akhir ini bisa bermanfaat bagi penulis pribadi maupun bagi para pembaca sekalian.

Yogyakarta, 30 Juni 2020



(Arif Subarkah)

SARI

Program Pengembangan Diri Qur'ani (PDQ) merupakan salah satu program pembinaan keagamaan yang wajib diikuti oleh mahasiswa Universitas Islam Indonesia. Direktorat Pembinaan dan Pengembangan Agama Islam (DPPAI) sebagai pihak penanggung jawab program PDQ sudah menggunakan sistem informasi dalam pengelolaan dan *monitoring* program tersebut. Permasalahannya sistem yang sudah ada belum bisa memberikan kontribusi yang maksimal. Hal tersebut dikarenakan sistem masih memiliki beberapa kekurangan. Diantara kekurangan tersebut yaitu tahapan atau proses pengelolaan program PDQ belum memberikan kemudahan bagi pengguna. Sistem seharusnya mampu mempermudah proses pengelolaan dan program PDQ, baik dari alur kerja maupun tampilan sistem. Namun pada kenyataannya pengguna masih merasa kesulitan dengan alur kerja dan tampilan yang dimiliki sistem sebelumnya. Selain itu sistem dirasa belum efisien, salah satu faktornya karena kecepatan respons dari sistem yang tergolong lambat. Hal tersebut mengakibatkan pada penambahan beban kerja dan waktu yang terbuang. Penelitian dilakukan untuk mengembangkan sebuah sistem baru guna membantu mengelola dan *monitoring* program PDQ secara lebih mudah dan efisien. Model pengembangan perangkat lunak yang digunakan dalam pengembangan sistem adalah model *Waterfall*. Pengujian pada sistem menggunakan metode *User Acceptance Test* (UAT) dan pengukuran efisiensi memanfaatkan *tool* GTMetrix. Hasil pengujian menunjukan bahwa sistem sudah termasuk kategori efisien dan mampu memberikan kemudahan.

Kata kunci: pengembangan diri qur'ani, *waterfall*, sistem informasi, efisiensi, *user acceptance test*.

GLOSARIUM

DPPAI	Direktorat Pembinaan dan Pengembangan Agama Islam.
PDQ	Pengembangan Diri Qur'ani.
Ta'lim	Kegiatan pengajaran ilmu agama dari seseorang kepada satu atau beberapa orang.
Package	Paket yang berisi kumpulan program yang bisa digunakan untuk membuat fitur tertentu secara lebih mudah.
Library	Kumpulan file yang berisi *code program* yang bisa dipanggil atau digunakan dalam proses *coding*.
Framework	Kerangka kerja yang berisi kode program untuk memudahkan proses pengembangan perangkat lunak.
Model	Kelas yang digunakan untuk memanipulasi data.
View	Bagian program yang berinteraksi langsung dengan pengguna.
Controller	Kelas yang memuat logika program untuk mengatur bagaimana sistem berinteraksi.
Route	Jalur untuk melakukan permintaan atau mengakses suatu alamat web.
Waterfall	Metode pengembangan perangkat lunak.
Wireframe	Rancangan hitam putih yang fokus menggambarkan struktur halaman website/aplikasi.

DAFTAR ISI

HALAMAN JUDUL i
HALAMAN PENGESAHAN DOSEN PEMBIMBING ii
HALAMAN PENGESAHAN DOSEN PENGUJI iii
HALAMAN PERNYATAAN KEASLIAN TUGAS AKHIR iv
HALAMAN PERSEMBAHAN v
HALAMAN MOTO vi
KATA PENGANTAR vii
SARI viii
GLOSARIUM ix
DAFTAR ISI x
DAFTAR TABEL xii
DAFTAR GAMBAR xiii
BAB I PENDAHULUAN 1
1.1 Latar Belakang 1
1.2 Rumusan Masalah 2
1.3 Tujuan Penelitian 2
1.4 Manfaat Penelitian 2
1.5 Batasan Masalah 2
1.6 Metode Penelitian 3
1.7 Sistematika Penulisan 4
BAB II LANDASAN TEORI 5
2.1 Pembinaan Keagamaan 5
2.2 Pengembangan Diri Qur'ani (PDQ) 5
2.3 Model Pengembangan Perangkat Lunak Waterfall 6
2.4 Sistem Informasi 7
2.5 Efisiensi 7
2.6 Pengujian Perangkat Lunak 8
2.7 Penelitian Terkait 9
BAB III METODOLOGI PENELITIAN 11
3.1 Studi Literatur 11
3.2 Pengembangan Sistem 12
3.2.1 Komunikasi 12

3.2.2 Perencanaan 13
3.2.3 Pemodelan Sistem 14
BAB IV IMPLEMENTASI DAN PENGUJIAN 46
4.1 Implementasi 46
4.1.1 Membuat Basis Data 46
4.1.2 Membuat Model 46
4.1.3 Membuat Controller 47
4.1.4 Membuat Route 49
4.1.5 Membuat View 49
4.2 Pengujian Sistem 58
4.2.1 *User Acceptance Testing* 58
4.2.2 Pengukuran Efisiensi 60
4.2.3 Evaluasi dan Kesimpulan Hasil Pengujian 61
BAB V KESIMPULAN DAN SARAN 62
5.1 Kesimpulan 62
5.2 Saran 62
DAFTAR PUSTAKA 63
LAMPIRAN 64

DAFTAR TABEL

Tabel 2.1 Penelitian Terkait 9
Tabel 3.1 Rangkuman Pertemuan dengan *Stakeholder* 12
Tabel 3.2 Daftar Tugas 13
Tabel 3.3 Teknologi yang dibutuhkan untuk pengembangan 13
Tabel 3.4 Aktor Sistem PDQ 15
Tabel 3.5 Rancangan Tabel Basis Data Mahasiswa 30
Tabel 3.6 Rancangan Tabel Basis Data Mahasiswa_taklim 31
Tabel 3.7 Rancangan Tabel Basis Data Organisasi 31
Tabel 3.8 Rancangan Tabel Basis Data Muallim 32
Tabel 3.9 Rancangan Tabel Basis Data Muallim_organisasi 32
Tabel 3.10 Rancangan Tabel Basis Data Kelompok 33
Tabel 3.11 Rancangan Tabel Basis Data Level 33
Tabel 3.12 Rancangan Tabel Basis Data Status 33
Tabel 3.13 Rancangan Tabel Basis Data Mahasiswa_nilai 34
Tabel 3.14 Rancangan Tabel Basis Data Periode 34
Tabel 3.15 Rancangan Tabel Basis Data Nilai_akhir 34
Tabel 3.16 Rancangan Tabel Basis Data Users 35
Tabel 3.17 Rancangan Pengujian 45
Tabel 4.1 Hasil Pengujian UAT berdasarkan wawancara 58
Tabel 4.2 Hasil Pengukuran Menggunakan GTmetrix. 60

DAFTAR GAMBAR

Gambar 2.1 Tahapan *Software Development Life Cycle* 6
Gambar 2.2 Tahapan Pengembangan Perangkat Lunak Model *Waterfall* 7
Gambar 3.1 Bagan Alur Penelitian 11
Gambar 3.2 *Use Case Diagram* Sistem Informasi Program PDQ 15
Gambar 3.3 *Activity Diagram* Unggah Hasil *Placement Test* Agama 17
Gambar 3.4 *Activity Diagram* Melihat Data Mahasiswa Program PDQ 18
Gambar 3.5. *Activity Diagram* Melihat Data Muallim 19
Gambar 3.6 *Activity Diagram* Melihat Data Kelompok 19
Gambar 3.7 *Activity Diagram* Melihat Nilai Semester Mahasiswa 20
Gambar 3.8. *Activity Diagram* Melihat Nilai Akhir Mahasiswa 21
Gambar 3.9 *Activity Diagram* Pengelolaan Akun Sistem 22
Gambar 3.10 *Activity Diagram* Melihat Data Mahasiswa Fakultas 23
Gambar 3.11 *Activity Diagram* Pengelolaan Data Kelompok Program PDQ 24
Gambar 3.12 *Activity Diagram* Pengelolaan Data Muallim Fakultas 25
Gambar 3.13 *Activity Diagram* Pengelolaan Data Muallim Fakultas 26
Gambar 3.14 *Activity Diagram* Pengelolaan Nilai Akhir Program PDQ 27
Gambar 3.15 *Activity Diagram* Melihat Data Mahasiswa yang Mengikuti Remediasi 28
Gambar 3.16 Arsitektur Sistem Pengelolaan dan Monitoring Program PDQ 29
Gambar 3.17 *ERD* Sistem Informasi Pengelolaan dan *Monitoring* Program PDQ 30
Gambar 3.18 Rancangan Antarmuka Halaman *Login* 36
Gambar 3.19 Rancangan Antarmuka Halaman Unggah Hasil *Placement Test* Agama 36
Gambar 3.20 Rancangan Antarmuka Halaman Data Mahasiswa 37
Gambar 3.21 Rancangan Antarmuka Halaman Data Muallim 37
Gambar 3.22 Rancangan Antarmuka Halaman Data Kelompok 38
Gambar 3.23 Rancangan Antarmuka Halaman Nilai Semester 38
Gambar 3.24 Rancangan Antarmuka Halaman Nilai Akhir 39
Gambar 3.25 Rancangan Antarmuka Halaman Kelola Akun 40
Gambar 3.26 Rancangan Antarmuka Halaman Data Mahasiswa Fakultas 40
Gambar 3.27 Rancangan Antarmuka Halaman Data Muallim Fakultas 41
Gambar 3.28 Rancangan Antarmuka Halaman Data Kelompok 41
Gambar 3.29 Rancangan Antarmuka Halaman Buat Kelompok 42
Gambar 3.30 Rancangan Antarmuka Halaman Nilai Semester Fakultas 42

Gambar 3.31 Rancangan Antarmuka Nilai Akhir Fakultas 43
Gambar 3.32 Rancangan Antarmuka Halaman *Input* Nilai Akhir Fakultas 43
Gambar 3.33 Rancangan Antarmuka Halaman Remediasi Nilai Ujian Semester 44
Gambar 3.34 Rancangan Antarmuka Halaman Remediasi Berdasarkan Kehadiran 44
Gambar 4.1 MySQL Workbench 46
Gambar 4.2 Contoh Kelas Model Data Muallim 47
Gambar 4.3 Contoh *Command* Untuk Membuat *Controller* Datamuallim 47
Gambar 4.4 Contoh Kode Program Kelas *Controller* Datamuallim 49
Gambar 4.5 Contoh *Endpoint* Fakultas Untuk Mengakses Sistem. 49
Gambar 4.6 Halaman Login Sistem 50
Gambar 4.7 Halaman *Dashboard* Administrator DPPAI 50
Gambar 4.8 Halaman *Dashboard Administrator* Fakultas 51
Gambar 4.9 Halaman Unggah Hasil *Placement Test Agama* 51
Gambar 4.10 Halaman Data Mahasiswa *Administrator* DPPAI 52
Gambar 4.11 Halaman Data Mahasiswa *Administrator* Fakultas 52
Gambar 4.12 Halaman Data Muallim *Administrator* DPPAI 53
Gambar 4.13 Halaman Data Muallim *Administrator* Fakultas 53
Gambar 4.14 *Form* Tambah Data Muallim *Administrator* Fakultas 54
Gambar 4.15 Halaman Data Kelompok *Administrator* DPPAI 54
Gambar 4.16 Halaman Data Kelompok *Administrator* Fakultas 55
Gambar 4.17 Halaman Pembuatan Kelompok *Administrator* Fakultas 55
Gambar 4.18 Halaman *Input* Nilai Akhir *Administrator* Fakultas 56
Gambar 4.19 Halaman Data Nilai Akhir *Administrator* Fakultas 56
Gambar 4.20 Halaman Remediasi *Administrator* Fakultas 57
Gambar 4.21 Halaman Pengelolaan Akun *Administrator* DPPAI 57

BAB I
PENDAHULUAN

1.1 Latar Belakang

Universitas Islam Indonesia memiliki berbagai macam program pembinaan keagamaan untuk mahasiswa. Program pembinaan keagamaan tersebut dikelola oleh Direktorat Pembinaan dan Pengembangan Agama Islam (DPPAI). Salah satu dari program pembinaan keagamaan tersebut adalah Pengembangan Diri Qur'ani (PDQ) yang sering disebut dengan kegiatan ta'lim. PDQ termasuk kedalam aktivitas wajib yang harus diikuti oleh mahasiswa. Hal tersebut tercantum dalam Peraturan Rektor No. 11 tahun 2017 Tentang Capaian Pembelajaran Lulusan Universitas dan Mata Kuliah Wajib Universitas. Bentuk dari aktivitas ini yaitu mahasiswa dibagi menjadi halaqah-halaqah atau kelompok untuk kemudian dibimbing oleh seorang pengajar yang disebut muallim.

Sebelum ditempatkan dalam suatu kelompok mahasiswa terlebih dahulu mengikuti sebuah tes keagamaan. Hasil tes tersebut untuk mengetahui seberapa jauh kemampuan dan keilmuan agama si mahasiswa. Kelompok ta'lim dibuat berdasarkan kemampuan dari mahasiswa. Setelah kelompok dibuat maka proses selanjutnya adalah penempatan muallim ke kelompok-kelompok yang sudah dibuat. Saat ini sudah ada sistem informasi yang membantu pengelolaan kegiatan ta'lim, namun sayangnya sistem informasi yang ada belum bisa membantu proses bisnis yang ada secara maksimal. Bahkan ada proses bisnis yang cukup menyulitkan bagi pengguna dan terbilang menambah beban pekerjaan. Salah satu contohnya adalah hasil dari ujian tes keagamaan yang berupa level kemampuan mahasiswa harus dipilah secara manual untuk kemudian dibentuk kelompok-kelompok ta'lim. Setelah kelompok dibentuk, daftar kelompok tersebut kemudian diserahkan ke muallim kelompoknya masing-masing. Setelah itu muallim kembali harus memasukan nama-nama mahasiswa kelompoknya ke dalam sistem.

Selaian pada proses yang kurang memudahkan pengguna, sistem yang sudah ada juga belum memiliki performa yang efisien. Pengguna sistem seringkali harus menunggu dengan waktu yang tidak sebentar untuk mengakses sebuah halaman atau memanfaatkan suatu fitur. Hal tersebut salah satunya disebabkan oleh desain basis data yang kurang baik, sehingga mempengaruhi kecepatan *browser* dalam memuat sebuah halaman maupun fitur.

Penelitian ini dilakukan untuk mengembangkan sebuah sistem informasi yang baru guna melayani pengelolaan kegiatan ta'lim secara lebih mudah dan efisien. Sistem informasi

digunakan oleh organisasi untuk membantu proses bisnis yang dimiliki organisasi menjadi lebih efisien dan juga memberikan faktor pembeda kompetitif yang utama, (O'brien, 2006; Winarno, 2007). Maka dari itu penelitian ini penting dan menarik untuk diteliti karena dapat memberikan kontribusi dalam hal membantu proses bisnis yang telah ada agar menjadi lebih baik.

1.2 Rumusan Masalah

Berdasarkan beberapa persoalan yang diuraikan pada latar belakang penelitian, maka rumusan masalah untuk peneltian ini adalah sebagai berikut.

a. Bagaimana mengembangkan sistem informasi yang mampu memberikan kemudahan untuk proses pengelolaan dan memonitor program PDQ?
b. Bagaimana mengembangkan sistem informasi yang memiliki performa efisien untuk mengelola dan memonitor program PDQ ?

1.3 Tujuan Penelitian

Tujuan penelitian ini adalah mengembangkan sistem informasi yang efisien dan memberikan kemudahan dalam mengelola dan melakukan *monitoring* program PDQ bagi DPPAI dan penyelenggara program PDQ fakultas.

1.4 Manfaat Penelitian

Manfaat dari penelitian ini adalah pihak DPPAI maupun pihak panitia penyelenggara program PDQ fakultas akan lebih mudah dalam melakukan pengelolaan dan *monitoring*.

1.5 Batasan Masalah

a. Proses bisnis yang diakomodasi dalam sistem dimulai dari proses mengunggah hasil *Placement Test* Agama .
b. Proses pada sistem berakhir sampai dengan mahasiswa mendapatkan nilai akhir program PDQ.
c. Sistem yang dikembangkan berbasis website.

1.6 Metode Penelitian

Dalam melaksanakan penelitian ini, peneliti melakukan studi literatur kemudian diteruskan dengan mengimplementasikan pengembangan perangkat lunak menggunakan metode *Waterfall*. Jenis pendekatan atau metode ini dipilih karena kebutuhan atau *requirement* dari sistem mudah untuk diidentifikasi secara jelas di awal. Adapun tahap pengembangan sistem dengan metode *Waterfall* adalah sebagai berikut:

a. *Communication* (Komunikasi)

Tahap ini peneliti fokus melakukan komunikasi dengan pihak terkait untuk mengumpulkan permasalahan sekaligus kebutuhan dari sistem yang akan dikembangkan. Dimulai dari memahami proses bisnis yang dikerjakan oleh DPPAI. Kemudian mengidentifikasi permasalahan yang ada pada sistem yang sebelumnya sudah berjalan, serta membantu mendefinisikan fitur pada sistem baru yang akan dikembangkan.

b. *Planning* (Perencanaan)

Tahap perencanaan peneliti mulai mendefinisikan sumber daya yang diperlukan. Kemudian peneliti juga mendefinisikan tugas-tugas atau pekerjaan yang akan dilakukan selama penelitian berikut dengan waktu estimasi yang diperlukan.

c. *Modelling* (Pemodelan)

Tahap pemodelan aktivitas yang dilakukan adalah melakukan perancangan terkait dengan sistem. Rancangan yang dibuat antara lain *Use Case Diagram, Activity Diagram,* Arsitektur Sistem, *Entity Relationship Diagram*, dan rancangan antarmuka sistem.

d. *Construction* (Pengkodean)

Setelah perancangan model basis data dan juga rancangan antarmuka sistem selesai dibuat, tahap selanjutnya adalah pengkodean. Rancangan sistem diterjemahkan kedalam kode-kode program.

e. *Deployment*

Setelah proses pengkodean sistem yang sudah berhasil dikembangkan kemudian sistem diuji dan ditunjukan kepada pengguna untuk mendapatkan *feedback*.

1.7 Sistematika Penulisan

Laporan tugas akhir ini disusun menjadi lima bab. Adapun pembagian kelima bab tersebut adalah sebagai berikut.

BAB 1 PENDAHULUAN

Bab pendahuluan memuat uraian latar belakang masalah, rumusan masalah, batasan masalah, tujuan penelitian, manfaat penelitian dan metodologi penelitian.

BAB II LANDASAN TEORI

Bab ini memuat teori terkait dengan penelitian yang dilaksanakan diantaranya adalah tentang teknologi yang digunakan dalam tahap implementasi.

BAB III METODOLOGI PENELITIAN

Bab ini memuat beberapa tahapan yang dilakukan seperti pengumpulan data, analisis masalah, identifikasi kebutuhan *user*, perancangan arsitektur sistem, dan perancangan beberapa diagram.

BAB IV HASIL DAN PEMBAHASAN

Bab ini memuat tahapan yang dilakukan selama proses implementasi dan pengujian sistem beserta hasilnya.

BAB V KESIMPULAN DAN SARAN

Bab ini memuat kesimpulan dari penelitian yang telah dilaksanakan dan sekaligus saran untuk penelitian ke depannya.

BAB II
LANDASAN TEORI

2.1 Pembinaan Keagamaan

Menurut kamus besar bahasa Indonesia (KBBI), pembinaan merupakan usaha, tindakan, dan kegiatan yang dilakukan secara efisien dan efektif untuk memperoleh hasil yang lebih baik. Sedangkan pengertian keagamaan menurut KBBI adalah sesuatu yang berhubungan dengan agama. Maka bisa diambil sebuah kesimpulan bahwa pembinaan keagamaan merupakan sebuah usaha atau upaya untuk menghasilkan maupun meningkatkan pengetahuan dan kemampuan keagamaan seseorang maupun kelompok.

Mengacu kepada Pola Pembinaan dan Pengembangan Mahasiswa (POLBANGMAWA) dan rancangan kurikulum Ulil Albab 2016, pembinaan keagamaan di Universitas Islam Indonesia dilaksanakan dalam berbagai jalur, baik jalur akademik kurikuler, maupun non kurikuler Pola pembinaan keagamaan di UII akan banyak menyentuh aspek aplikatif, sehingga mahasiswa dapat segera menerapkan apa yang diperoleh dari pembinaan keagamaan di masyarakat (Universitas Islam Indonesia, 2017). Secara umum, pembinaan keagamaan di UII bertujuan membangun profil mahasiswa UII yang memiliki pemahaman yang utuh terhadap ajaran agama Islam, berakhlak mulia, mampu membaca al-qur'an secara tartil, melaksanakan sholat berjamaah secara konsisten, dan mampu melaksanakan dakwah islamiyah (DPPAI, 2019).

2.2 Pengembangan Diri Qur'ani (PDQ)

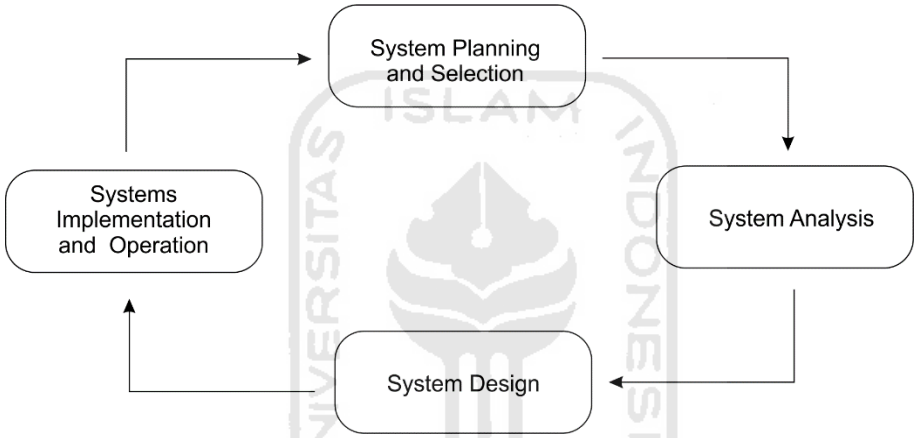
Pengembangan Diri Qurani (PDQ) merupakan salah satu aktivitas kemahasiswaan yang bersifat wajib untuk mahasiswa program sarjana (Universitas Islam Indonesia, 2017). Adapun tujuan secara umum dari PDQ yaitu mendidik, menghafal, dan memonitor mahasiswa untuk mampu dan menguasai baca tulis al-qur'an (DPPAI, 2019). PDQ terdiri dari serangkaian kegiatan. Mahasiswa baru sebelum mengikuti kegiatan PDQ akan mengikuti *Placement Test* Agama (PTA). Ujian yang diikuti terdiri dari ujian bacaan dan hafalannya. Hasil dari ujian tersebut akan digunakan dalam pengelompokan mahasiswa untuk kegiatan PDQ.

Kelompok PDQ memiliki empat macam level. Setiap level mewakili seberapa kemampuan mahasiswa yang ada pada level tersebut. Adapun level kelompok PDQ terdiri dari kelompok Pra Dasar, Dasar, Menengah, dan Lanjut. Setiap level tentu akan mendapatkan

perlakuan atau pembinaan yang berbeda, begitu juga capaian maupun target akan disesuaikan dengan level kelompok.

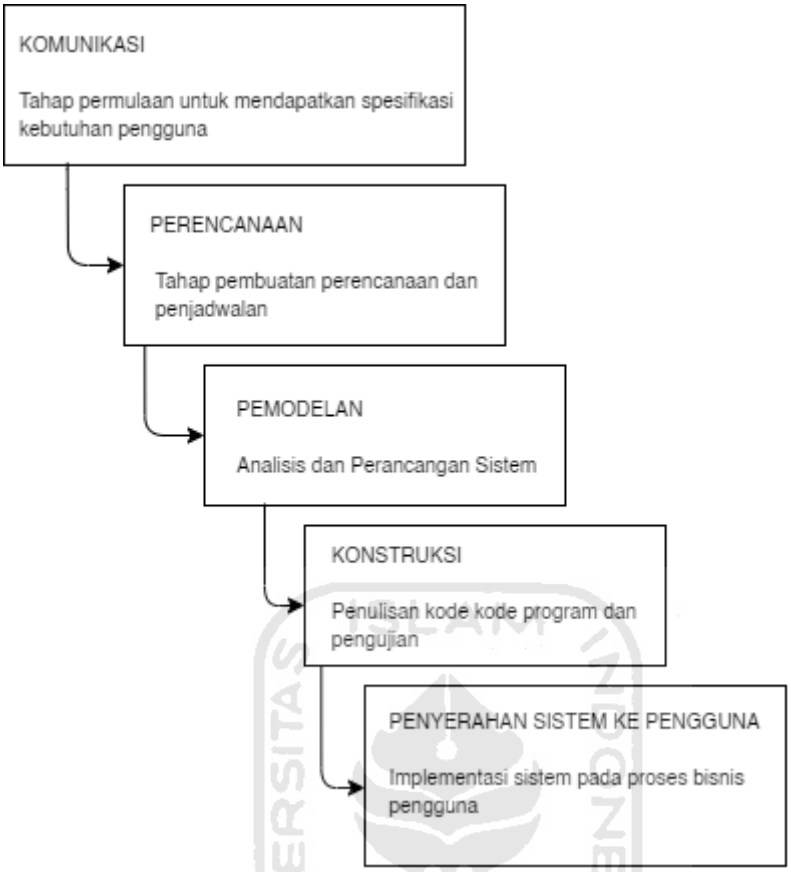
2.3 Model Pengembangan Perangkat Lunak Waterfall

Dalam melakukan pengembangan perangkat lunak ada istilah yang disebut dengan metodologi pengembangan. Metodologi pengembangn perangkat lunak adalah pendekatan formal untuk mengimplementasikan *Software Development Life Cycle* (SDLC) (Dennis, 2012). SDLC merupakan siklus hidup dari pengembangan sistem. Tahapan utama dari SDLC yaitu terdiri dari tahap perencanaan sistem, analisis sistem, perancangan sistem, dan implementasi sistem (Valacich, 2012).



Gambar 2.1 Tahapan *Software Development Life Cycle*

SDLC telah banyak dipelajari dan diselidiki oleh banyak peneliti dan praktisi di seluruh dunia. SDLC memiliki banyak model atau metode yang diusulkan, dan masing-masing memiliki kelebihan dan kekurangan (Bassil, 2012). Setiap metode paling tidak akan memiliki empat fase SDLC, hanya saja metode dalam menjalankan fasenya yang berbeda. Salah satu implementasi SDLC adalah metode *Waterfall*. Model pengembangan perangkat lunak dengan pendekatan *Waterfall* setiap fase yang ada dikerjakan secara berurutan (*sequential*). Ketika pekerjaan dalam satu tahap telah disetujui baru melangkah ke tahap selanjutnya (Dennis, 2012). Tahapan metode *Waterfall* ada lima tahapan, yaitu komunikasi, perencanaan, pemodelan, konstruksi, dan deployment (Pressman, 2015). Metode *Waterfall* termasuk salah satu metode pengembangan perangkat lunak yang mudah untuk dipahami. Penelitian oleh Firmansyah dan Udi (2018) menyimpulkan bahwa metode *Waterfall* mudah untuk dipahami dan cocok untuk pengembangan sistem informasi. Gambar metode *Waterfall* ditunjukan pada Gambar 2.2.



Gambar 2.2 Tahapan Pengembangan Perangkat Lunak Model *Waterfall*

2.4 Sistem Informasi

Sistem informasi secara teknis adalah suatu rangkaian komponen yang saling terkait untuk mengumpulkan, memproses, menyimpan dan mendistribusikan informasi untuk mendukung pengambilan keputusan dan mengendalikan perusahaan (Laudon, 2014). Sistem informasi melibatkan manusia, *hardware, software*, data, dan jaringan untuk melakukan *input*, proses, *output*, penyimpanan dan kontrol dengan tujuan mengubah sumber data menjadi sebuah informasi (O'Brien & Marakas, 2013).

2.5 Efisiensi

Salah satu aspek atau karakteristik yang menentukan suatu perangkat lunak dikatakan berkualitas atau tidak adalah karakteristik efisiensinya. Berdasarkan ISO 9126-1 yang memuat standar kualitas perangkat lunak, disebutkan bahwa ada enam karakteristik utama kualitas dari

perangkat lunak yang mana salah satu aspeknya adalah efisiensi. Karakteristik dalam standar ISO 9126-1 adalah sebagai berikut:

a. *Functionality* (Fungsionalitas). Berkaitan dengan kemampuan perangkat lunak dalam memenuhi kebutuhan pengguna.

b. *Reliability* (Kehandalan). Berkaitan dengan kinerja perangkat lunak untuk menyediakan fungsi atau layanan dalam kondisi atau periode waktu yang ditentukan.

c. *Usability* (Kebergunaan). Sangat berkaitan dengan aspek fungsionalitas, dimana *usability* mengacu pada kemudahan penggunaan untuk setiap fungsi yang ada pada perangkat lunak.

d. *Efficiency* (Efisiensi). Berkaitan dengan sumber daya sistem yang digunakan saat menyediakan fungsionalitas yang diperlukan.

e. *Maintainability* (Pemeliharaan). Berkaitan dengan kemudahan dalam mengidentifikasi dan memperbaiki kesalahan.

f. *Portability* (Portabilitas). Karakter ini mengacu pada seberapa baik perangkat lunak dapat mengadopsi perubahan di lingkungannya dan persyaratannya.

Berdasarkan standar kualitas perangkat lunak tersebut, karakteristik efisiensi tidak lepas dari dua aspek yaitu *time behavior* dan *resource behavior*. *Time Behavior* merupakan aspek yang berkaitan dengan waktu yang dibutuhkan suatu sistem untuk memberikan *response* terhadap *input* yang diberikan. Sedangkan aspek *resource behavior* berhubungan dengan penggunaan sumber daya seperti *memory, cpu, disk,* dan *network usage*. Penelitian ini salah satunya akan menguji karakteristik efisiensi berdasarkan dua aspek tersebut. Sebuah website dapat dikatakan baik jika waktu *load* atau *response* tidak lebih dari sepuluh detik (Nielsen, 2010).

2.6 Pengujian Perangkat Lunak

Tujuan utama dilakukan pengujian perangkat lunak adalah memastikan bahwa perangkat lunak yang dihasilkan sesuai dengan kebutuhan yang sebelumnya telah ditentukan. Pengujian perangkat lunak memiliki beberapa tipe diantaranya (Filipova, 2018) :

a. *Unit Testing* (Pengujian Unit). Menguji sebagian kecil dari kode program secara individual, seperti menguji sebuah *function, method, class,* maupun *modul*.

b. *Integration Testing* (Pengujian Integrasi). Menguji modul yang berbeda apakah dapat saling bekerja sama atau tidak.

c. *System Testing* (Pengujian Sistem). Menggabungkan beberapa jenis pengujian dengan tujuan memverifikasi bahwa sistem dapat bekerja dengan benar secara keseluruhan.
d. *Acceptance Testing* (Pengujian Penerimaan). Menguji kesesuaian antara spesifikasi kebutuhan dan implementasinya serta fokus pada bagaimana pengguna akhir akan menggunakan produk.
e. *Regression Testing* (Pengujian Regresi). Memastikan bahwa perubahan kode baru tidak memengaruhi versi perangkat lunak yang sebelumnya telah diuji dan stabil.

Dari beberapa tipe pengujian diatas, penilitian ini memilih tipe *Acceptance Testing*. Dimana pengujian dilakukan oleh klien atau pengguna akhir langsung atau disebut dengan *User Acceptance Testing*. Alasan utama dilakukan pengujian ini adalah untuk mengidentifikasi bagaimana sistem bekerja dan bagaimana hal itu akan bermanfaat bagi pengguna akhir sebelum sistem diimplementasikan pada proses bisnis mereka (Mohd & Shahbodin, 2015). Hal ini penting dilakukan dalam penelitian ini untuk mengetahui sistem yang dikembangkan sesuai dengan kebutuhan pengguna atau tidak.

2.7 Penelitian Terkait

Penulis menemukan beberapa penelitian terdahulu yang memiliki relevansi dan dapat digunakan sebagai salah satu acuan dalam melakukan penelitian. Beberapa penelitian yang mengangkat tema yang sejenis disajikan pada Tabel 2.1

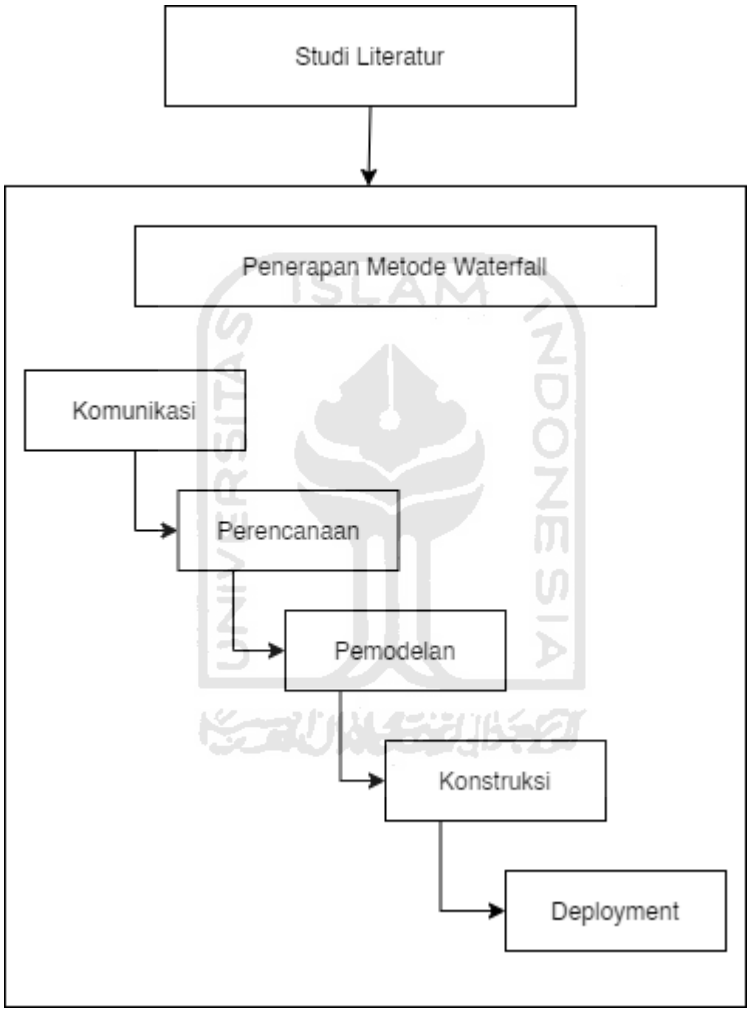
Tabel 2.1 Penelitian Terkait

No	Nama Peneliti	Judul Penelitian	Permasalahan yang diangkat	Metode
1.	Arif Pramana Putra (2019)	Pengembangan Website untuk Program Taklim UII	Bagaimana mengembangkan sistem informasi yang dapat memenuhi kebutuhan informasi bagi segenap *stakeholder* yaitu peserta, muallim, fakultas dan DPPAI	Metode pengembangan perangkat lunak *Rapid Application Development* (RAD)
Hasil Penelitian : Berdasarkan pengujian, sistem yang dikembangkan memberikan dampak yang baik bagi *stakeholder*				

<table>
<tr><td colspan="5">Perbedaan : Permasalahan yang diangkat berbeda, penulis mengangkat permasalahan sistem informasi program PDQ yang tidak efisien dan tidak mudahnya proses pengelolaan dan monitoring program PDQ. Metode yang dipilih berbeda, peneliti sebelumnya menggunakan RAD sedangkan penulis menggunakan metode pengembangan perangkat lunak Waterfall.</td></tr>
<tr><td>2.</td><td>Candra Dewi (2017)</td><td>Sistem Monitoring Capaian Program Pendampingan Materi Keislaman Untuk Mahasiswa UII</td><td>Bagaiamana mengembangkan sistem yang bisa dimanfaatkan oleh muallim, PIC fakultas dan DPPAI untuk memantau keberhasilan dan capaian program pendampingan materi keislaman dan dapat digunakan untuk memilih kader mualim.</td><td>Studi literatur, analisis proses bisnis, perancangan, coding, pengujian, penyusunan laporan.</td></tr>
<tr><td colspan="5">Hasil Penelitian : Sistem yang dikembangkan dapat membantu kinerja pengguna, seperti memantau perkembangan penguasaan agama oleh mahasiswa, melakukan penilaian dan pelaporan pertemuan, melakukan proses kaderisasi terhadap calon mualim.</td></tr>
<tr><td colspan="5">Perbedaan : Peneliti sebelumnya hanya berfokus pada bagaiamana sistem yang dikembangkan bisa dimanfaatkan oleh pengguna. Sementara penulis memilik fokus yang lebih spesifik bagaiamana sistem yang dikembangkan bisa memberikan kemudahan dan efisiensi bagai pengelolaan program PDQ.</td></tr>
</table>

BAB III
METODOLOGI PENELITIAN

Dalam bab ini menjelaskan tahapan dan bagaimana penelitian dilakukan. Dimulai dari melakukan studi literatur sampai dengan proses pengembangan sistem. Adapun tahapan penelitian yang dilakukan oleh peneliti ditunjukan pada Gambar 3.1.



Gambar 3.1 Bagan Alur Penelitian

3.1 Studi Literatur

Studi literatur dilakukan untuk mendapatkan informasi yang relevan guna memudahkan langkah-langkah yang ditempuh dalam melaksanakan penelitian. Adapun bahan yang digunakan untuk studi literatur adalah jurnal penelitian, *textbook* dan artikel dari internet berisi topik yang relevan dengan penelitian.

3.2 Pengembangan Sistem

Pengembangan sistem pada penelitian ini menggunakan metode *Waterfall.* Pada metode ini terdapat beberapa tahapan diantaranya yaitu:

3.2.1 Komunikasi

Pada tahap ini peneliti melakukan komunikasi dengan pemangku kepentingan (*stakeholder*) yaitu Direktorat Pembinaan dan Pengembangan Agama Islam (DPPAI) Universitas Islam Indonesia. Komunikasi dilakukan agar sistem yang dikembangkan sesuai dengan kebutuhan (*requirement*). Rangkuman dari proses komunikasi dengan DPPAI ditunjukan pada Tabel 3.1.

Tabel 3.1 Rangkuman Pertemuan dengan *Stakeholder*

Pertemuan dengan *stakeholder* tahap 1	
Agenda	Pemaparan Proses Bisnis Pembinaan Keagamaan UII
Narasumber	Junaidi Safitri, S.E., M.E.I
Tempat	Kantor DPPAI
Hasil Pertemuan	a.Informasi pembinaan keagamaan yang ada di UII secara rinci b.Proses bisnis program PDQ dari awal sampai dengan akhir. c.Pedoman penyelenggaraan program Pengembangan Diri Qur'ani d.Permasalahan sistem yang sudah digunakan
Pertemuan dengan *stakeholder* tahap 2	
Agenda	Pemaparan Hasil Analisis Proses Bisnis Pembinaan Keagamaan UII
Tempat	Kantor DPPAI
Hasil pertemuan	Perbaikan alur pada proses bisnis yang bisa diakomodasi oleh sistem
Pertemuan dengan *stakeholder* tahap 3	
Agenda	Konfirmasi Akhir Kebutuhan Sistem
Tempat	Kantor DPPAI
Hasil pertemuan	Kesepakatan proses bisnis yang diakomodasi oleh sistem

3.2.2 Perencanaan

Tahap perencanaan peneliti mulai mendefinisikan tugas-tugas yang akan dikerjakan selama proses pengembangan sistem. Dimulai dari merencanakan tugas pada tahap pemodelan atau perancangan. Pada tahap tersebut peneliti memutuskan untuk membuat diagram *Unified Modeling Language* (UML) yang terdiri dari beberapa diagram guna memudahkan dalam memberikan gambaran sistem. Selain membuat diagram UML, peneliti juga akan membuat rancangan antarmuka sistem yang dibuat berdasarkan gambaran dari diagram UML. Rancangan antarmuka penting untuk dibuat agar memudahkan proses pengembangan sistem terutama dari sisi *frontend*. Detail tugas yang akan dikerjakan disajikan pada Tabel 3.2.

Tabel 3.2 Daftar Tugas

No	Nama *Task*	Estimasi Waktu
	Pemodelan Sistem (Perancangan)	
1	Membuat *use case diagram*	2 Hari
2	Membuat *activity diagram*	2 Hari
3	Membuat arsitektur sistem	2 Hari
4	Membuat *entity relationship diagram*	2 Minggu
5	Membuat rancangan antarmuka sistem	2 Minggu
6	Membuat perancangan pengujian sistem	1 Minggu
	Konstruksi (Implementasi)	
1	Membuat basis data	2 Hari
2	Memprogram sistem	4 Bulan
3	Perbaikan bug	2 Minggu
	Deployment (Delivery ke User)	
1	Deploy sistem ke hosting	1 Hari
2	Pengujian efisiensi sistem	3 Hari
3	Melakukan *User Acceptance Testing*	1 Hari

Selain mendefinisikan tugas-tugas yang akan dikerjakan, dibuat juga perencanaan teknologi yang akan digunakan pada tahap konstruksi atau implementasi. Pemilihan teknologi dilakukan atas dasar berbagai pertimbangan. Teknologi yang akan digunakan dalam mengembangkan sistem ini disajikan pada Tabel 3.3.

Tabel 3.3 Teknologi yang dibutuhkan untuk pengembangan

Backend	Bahasa pemrograman untuk *server side*	PHP 7.3
	Framework	Laravel 7

	Database	Mysql
Frontend	Bahasa pemrograman untuk *client side*	Javascript (AJAX)
	Framework HTML CSS	Bootstrap 4

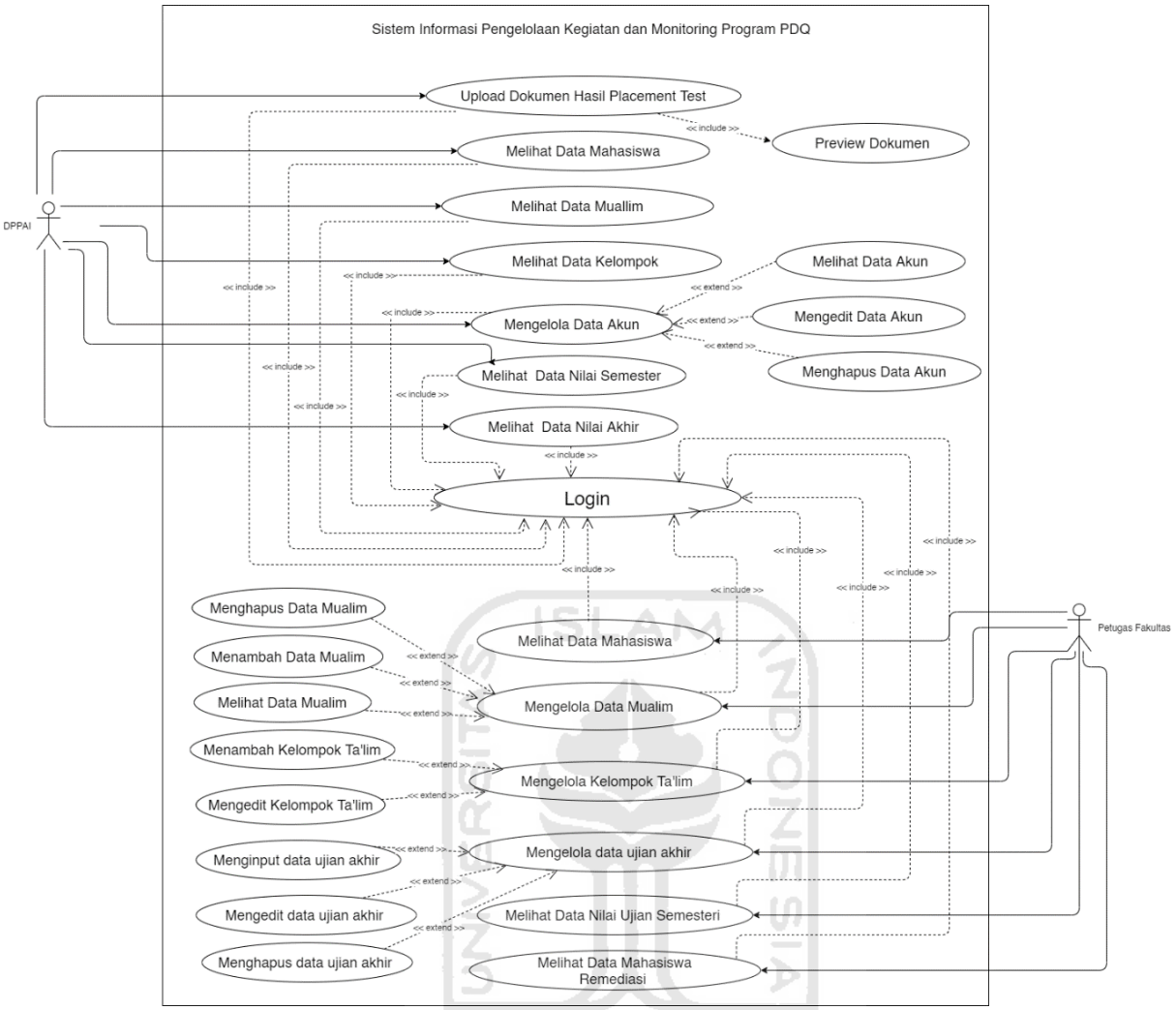
Pada tahap ini peneliti juga merencanakan bagaimana sistem akan diuji ketika sudah berhasil dikembangkan. Dikarenakan penelitian ini berfokus bagaimana mengembangkan sistem yang memberikan kemudahan bagi pengguna dan memiliki performa efisien yang baik maka pengujian yang dipilih merupakan pengujian yang spesifik dengan hal tersebut. Peneliti berencana menguji sistem dengan metode *User Acceptance Testing* dan juga uji pengukuran efisiensi sistem dengan menggunakan sebuah *tool*.

3.2.3 Pemodelan Sistem

Tahap ini membuat beberapa rancangan dari sistem. Pemodelan sistem berguna untuk memberikan gambaran yang lebih spesifik mengenai sistem yang akan dikembangkan. Pada tahapan ini dibuatlah beberapa rancangan diantaranya *use case diagram, activity diagram,* arsitektur sistem, rancangan basis data sampai dengan rancangan antarmuka sistem.

A. Use Case Diagram

Dalam *use case diagram* menampilkan bagaimana aktor atau pengguna berinteraksi dengan sistem. Pada Gambar 3.2 menampilkan fungsi apa saja yang ada pada sistem dan siapa saja yang dapat memanfaatkan fungsi tersebut.



Gambar 3.2 *Use Case Diagram* Sistem Informasi Program PDQ

Pada Gambar 3.2 terdapat dua aktor utama yang akan menggunakan sistem. Berikut penjelasan bagaimana aktor berinteraksi dengan sistem yang disajikan pada Tabel 3.4.

Tabel 3.4 Aktor Sistem PDQ

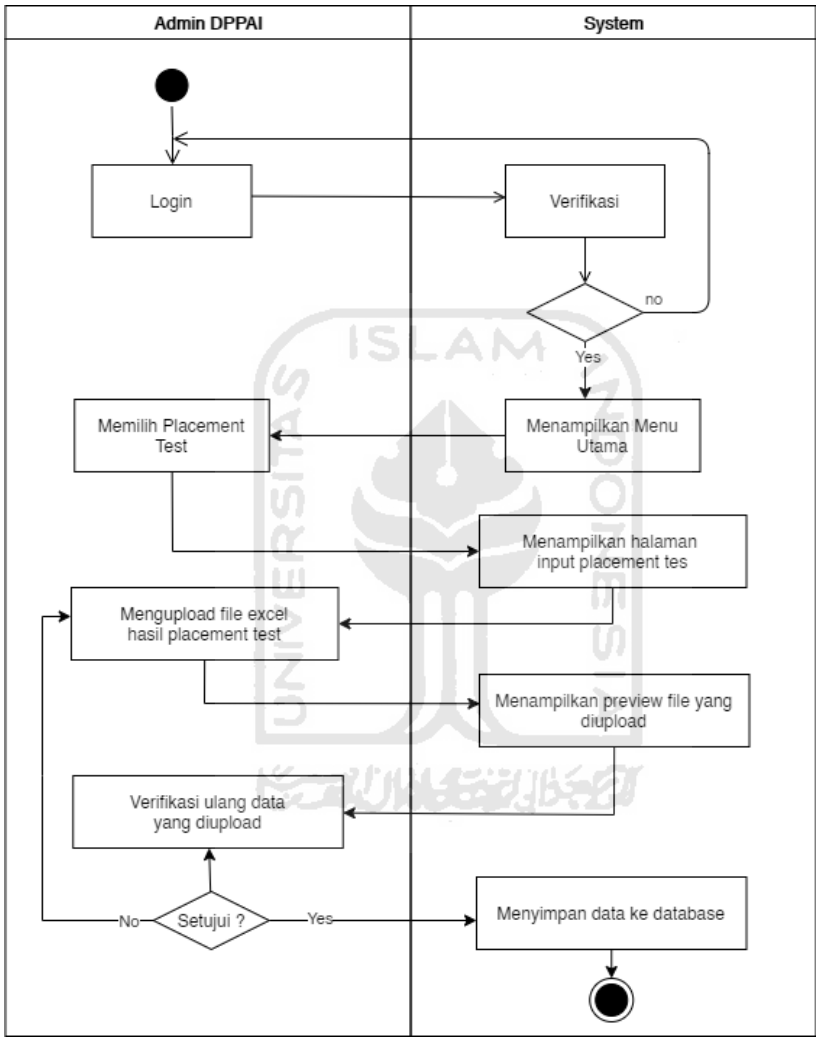
No	Aktor	Use Case	Keterangan
1	Admin DPPAI	Mengunggah Hasil *Placement Test* Agama	Admin DPPAI mengunggah dokumen/file excel yang berisi hasil *Placement Test* Agama agar sistem dapat membaca dan menyimpan level mahasiswa.
		Melihat Data Mahasiswa	Admin DPPAI melihat data mahasiswa program PDQ di seluruh fakultas untuk mengetahui informasi secara lebih spesifik tentang mahasiswa

			yang ingin diketahui.
		Melihat Data Muallim	Admin DPPAI melihat data muallim di seluruh fakultas untuk mengetahui informasi secara lebih spesifik tentang muallim yang ingin diketahui
		Melihat Data Kelompok	Admin DPPAI melihat data kelompok di seluruh fakultas untuk mengetahui informasi secara lebih spesifik tentang suatu kelompok yang ingin diketahui
		Melihat Data Nilai Semester	Admin DPPAI melihat data nilai ujian semester dari mahasiswa di semua fakultas
		Melihat Data Nilai Ujian Akhir PDQ	Admin DPPAI melihat data nilai ujian akhir dari mahasiswa di semua fakultas
		Mengelola Akun Sistem	Admin DPPAI dapat mengelola akun yang ada pada sistem, seperti menambah, mengedit, maupun menghapus
2	Admin Fakultas	Melihat Data Mahasiswa	Admin Fakultas dapat melihat data mahasiswa peserta program PDQ di masing-masing fakultasnya.
		Mengelola Data Muallim	Admin Fakultas dapat mengelola data muallim di fakultasnya mulai dari melihat, menambah, menyunting, maupun menghapus.
		Mengelola Data Kelompok	Admin Fakultas dapat mengelola data kelompok program PDQ di fakultasnya mulai dari melihat, membuat , menyunting, maupun menghapus suatu kelompok.
		Melihat Data Nilai Ujian Semester	Admin Fakultas dapat melihat data nilai ujian semester PDQ setiap mahasiswa di fakultasnya masing-masing
		Mengelola Data Nilai Ujian Akhir	Admin Fakultas dapat mengelola data nilai akhir mahasiswa program PDQ di fakultasnya mulai dari melihat, menyimpan , menyunting, maupun menghapus.
		Mengelola Data Mahasiswa Remediasi	Admin Fakultas dapat mengelola data mahasiswa yang harus mengikuti proses remediasi PDQ di fakultasnya .

B. Activity Diagram

a. Memasukan data hasil *Placement Test* Agama

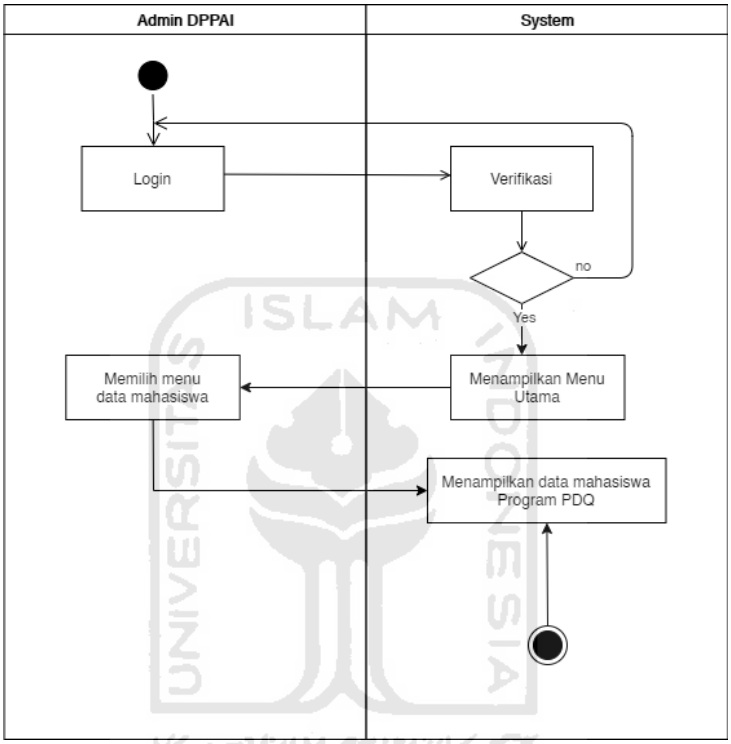
Activity diagram ini menggambarkan alur aktivitas proses mengunggah dokumen hasil *Placement Test* Agama. Admin DPPAI harus melakukan *login* terlebih dahulu ke sistem untuk bisa mengakses fitur unggah hasil *Placement Test* Agama. Alur detail aktivitas tersebut disajikan pada Gambar 3.3.



Gambar 3.3 *Activity Diagram* Unggah Hasil *Placement Test* Agama

b. Melihat data mahasiswa program PDQ

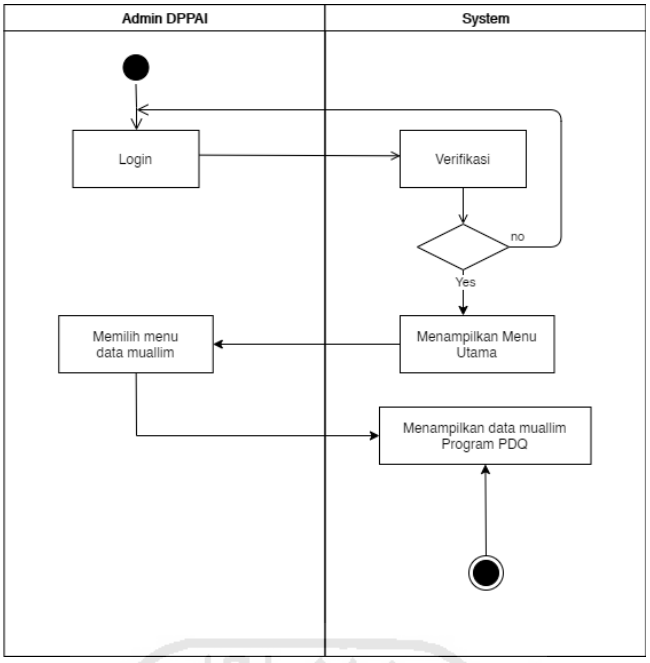
Activity diagram ini menggambarkan alur aktivitas dari Admin DPPAI untuk mengakses data mahasiswa peserta program PDQ. Admin harus *login* ke sistem terlebih dahulu untuk kemudian bisa memilih menu data mahasiswa dan bisa mengakses seluruh data mahasiswa peserta program PDQ di semua fakultas UII. Alur aktivitas tersebut disajikan pada Gambar 3.4.



Gambar 3.4 *Activity Diagram* Melihat Data Mahasiswa Program PDQ

c. Melihat data muallim progam PDQ

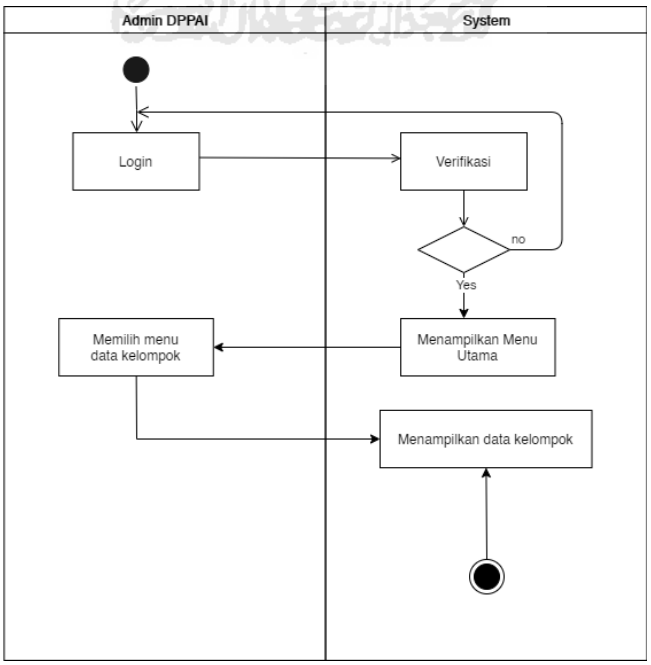
Activity diagram ini menggambarkan alur aktivitas dari Admin DPPAI untuk mengakses data muallim program PDQ. Admin harus *login* ke sistem terlebih dahulu untuk kemudian bisa memilih menu data muallim dan bisa mengakses seluruh data muallim program PDQ di semua fakultas UII. Alur aktivitas tersebut disajikan pada Gambar 3.5.



Gambar 3.5. *Activity Diagram* Melihat Data Muallim

d. Melihat data kelompok progam PDQ

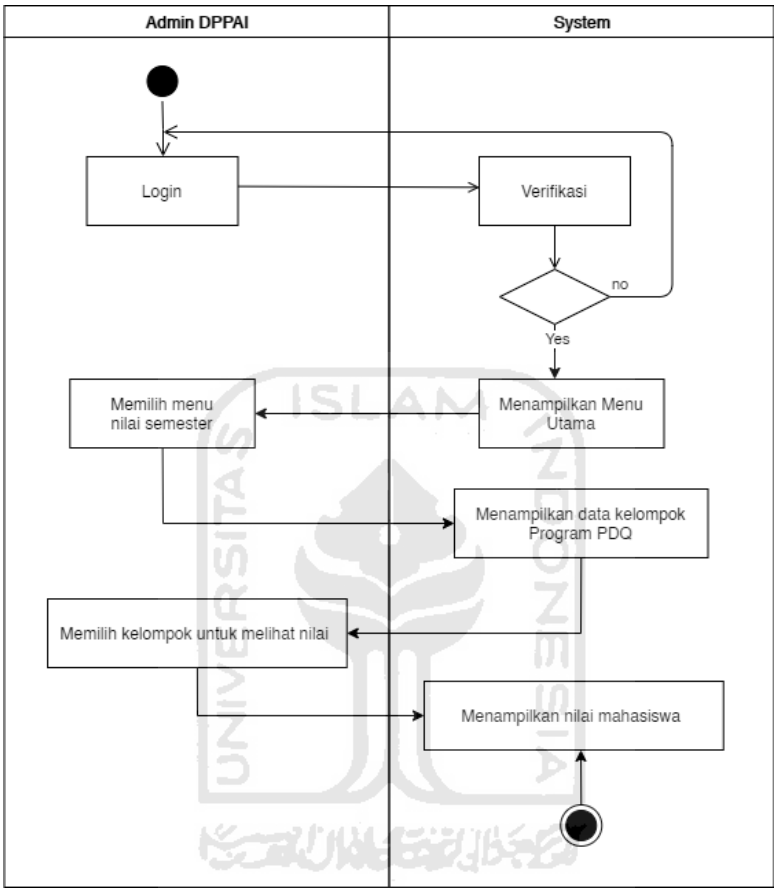
Activity diagram ini menggambarkan alur aktivitas dari Admin DPPAI untuk mengakses data kelompok program PDQ. Admin harus *login* ke sistem terlebih dahulu untuk kemudian bisa memilih menu data kelompok dan bisa mengakses seluruh data kelompok program PDQ di semua fakultas UII. Alur aktivitas tersebut disajikan pada Gambar 3.6.



Gambar 3.6 *Activity Diagram* Melihat Data Kelompok

e. Melihat nilai ujian semester mahasiswa

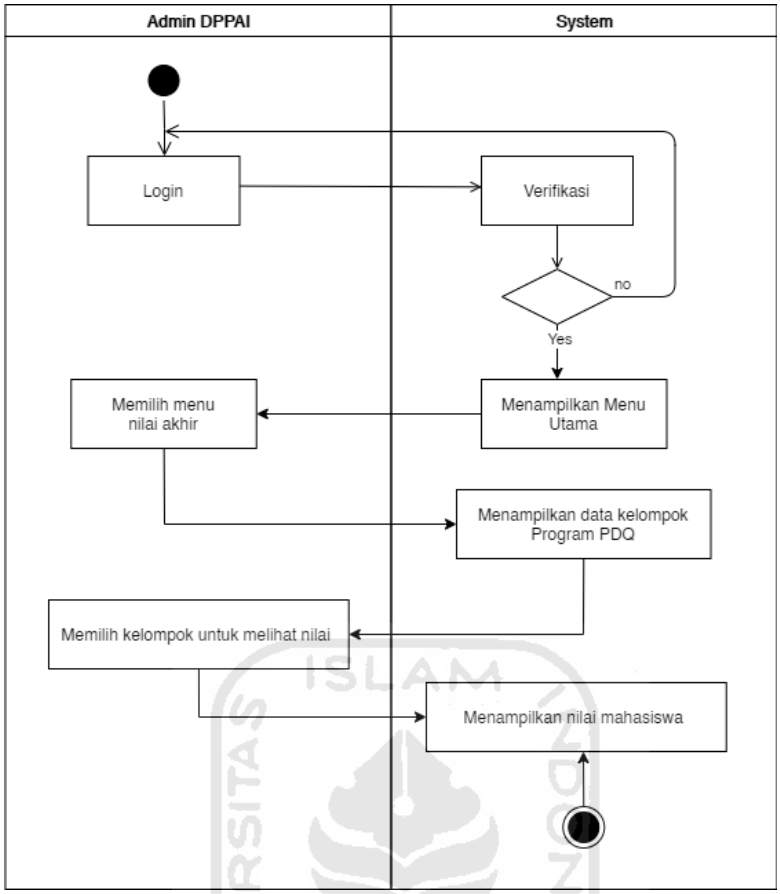
Activity diagram ini menggambarkan alur aktivitas dari Admin DPPAI untuk melihat nilai ujian semester PDQ mahasiswa. Admin harus *login* ke sistem terlebih dahulu untuk kemudian bisa memilih menu nilai semester dan bisa melihat nilai ujian semester mahasiswa. Alur aktivitas tersebut disajikan pada Gambar 3.7.



Gambar 3.7 *Activity Diagram* Melihat Nilai Semester Mahasiswa

f. Melihat nilai ujian akhir PDQ mahasiswa

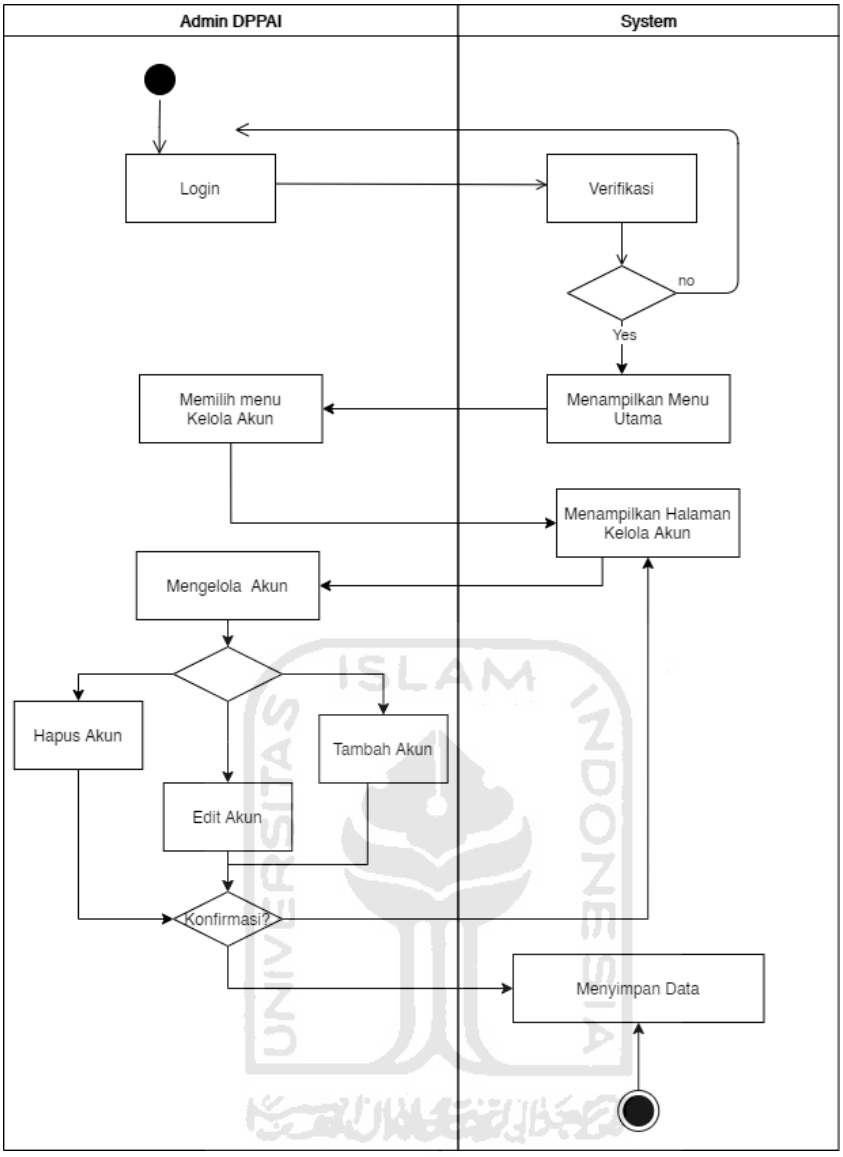
Activity diagram ini menggambarkan alur aktivitas dari Admin DPPAI untuk melihat nilai akhir PDQ mahasiswa. Admin harus *login* ke sistem terlebih dahulu untuk kemudian bisa memilih menu nilai akhir dan bisa melihat nilai ujian akhir mahasiswa. Alur aktivitas tersebut disajikan pada Gambar 3.8.



Gambar 3.8. *Activity Diagram* Melihat Nilai Akhir Mahasiswa

g. Mengelola pengguna untuk masuk ke sistem

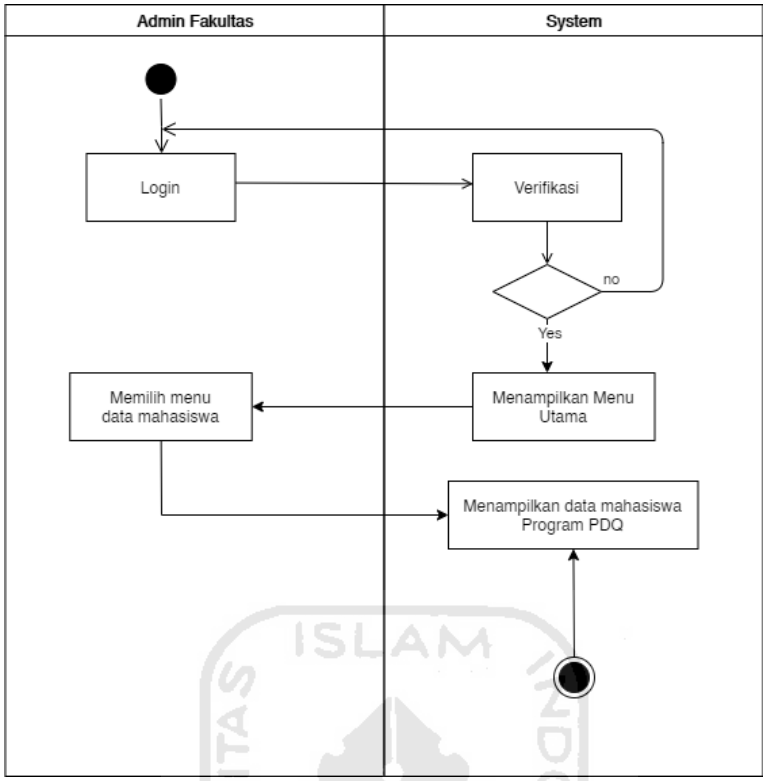
Activity diagram ini menggambarkan alur aktivitas dari Admin DPPAI untuk mengelola pengguna sistem. Admin harus *login* ke sistem terlebih dahulu untuk kemudian bisa memilih menu kelola akun dan bisa melakukan penambahan, perubahan maupun menghapus pengguna yang ada dalam sistem. Alur aktivitas tersebut disajikan pada Gambar 3.9.



Gambar 3.9 *Activity Diagram* Pengelolaan Akun Sistem

h. Melihat data mahasiswa fakultas

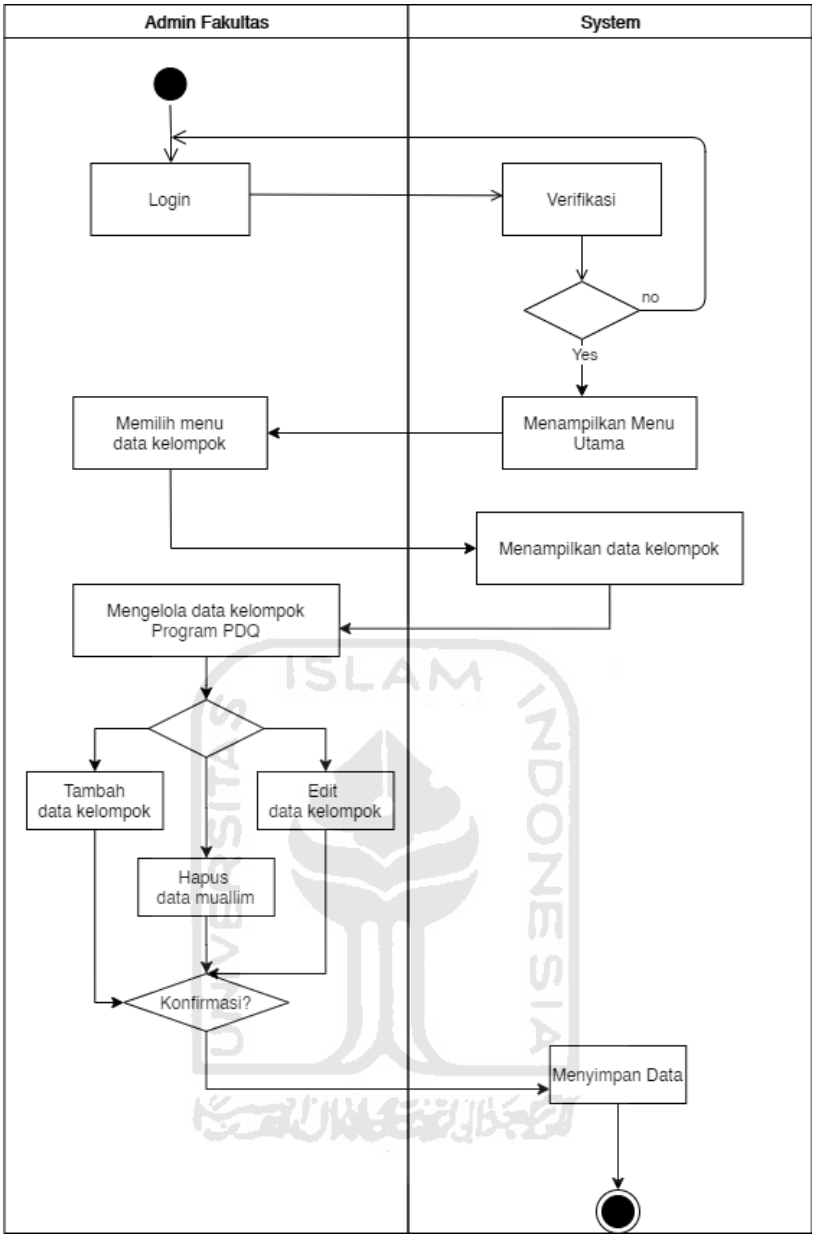
Activity diagram ini menggambarkan alur aktivitas dari Admin Fakultas untuk melihat data mahasiswa program PDQ di masing-masing fakultasnya. Admin harus *login* ke sistem terlebih dahulu untuk kemudian bisa memilih menu data mahasiswa dan bisa melihat data mahasiswasi program PDQ. Alur aktivitas tersebut disajikan pada Gambar 3.10.



Gambar 3.10 *Activity Diagram* Melihat Data Mahasiswa Fakultas

i. Mengelola kelompok PDQ

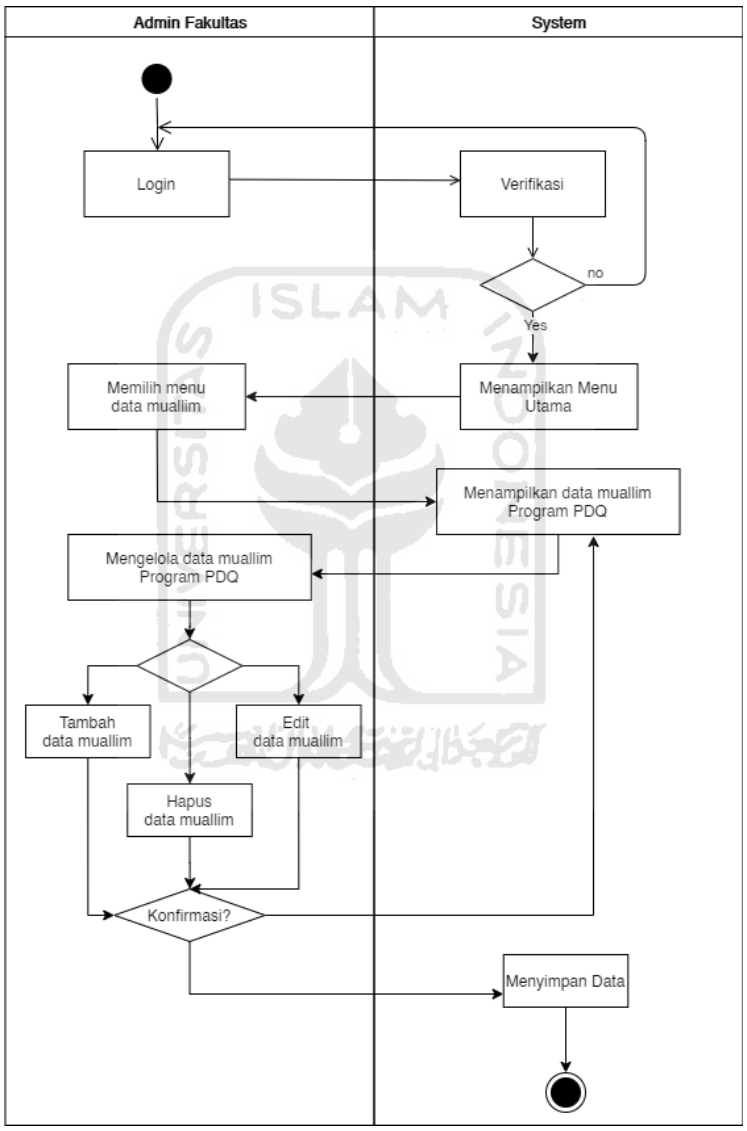
Activity diagram ini menggambarkan alur aktivitas dari Admin Fakultas untuk mengelola kelompok program PDQ di fakultasnya masing-masing. Admin harus *login* ke sistem terlebih dahulu untuk kemudian bisa memilih menu data kelompok dan bisa melakukan penambahan, perubahan maupun menghapus kelompok program PDQ. Alur aktivitas tersebut disajikan pada Gambar 3.11.



Gambar 3.11 *Activity Diagram* Pengelolaan Data Kelompok Program PDQ

j. Mengelola data muallim

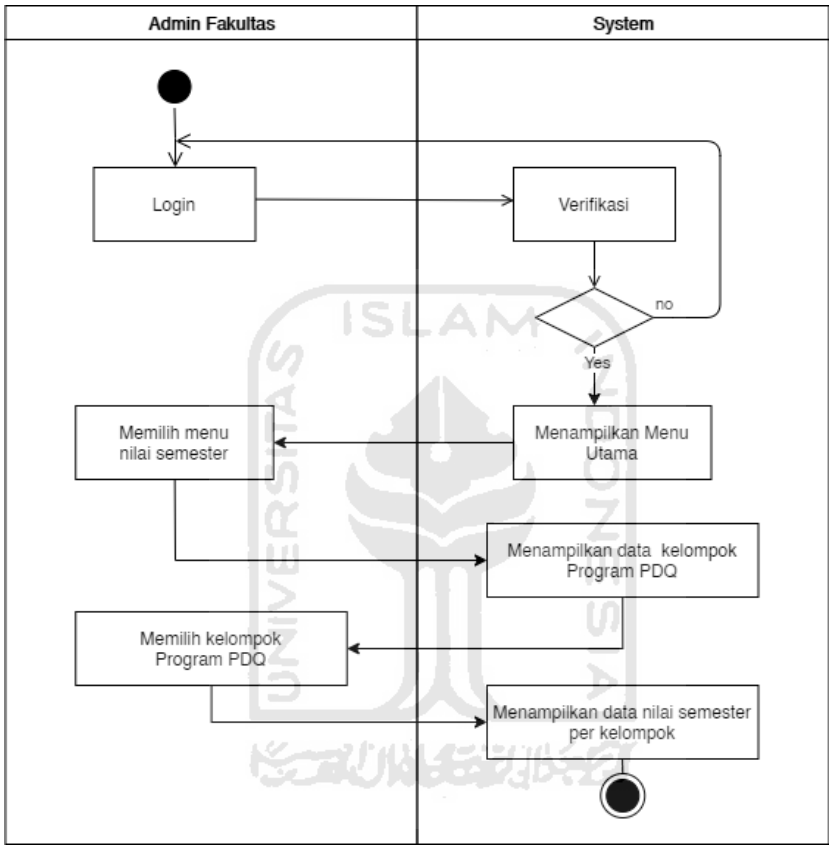
Activity diagram ini menggambarkan alur aktivitas dari Admin Fakultas untuk mengelola data muallim program PDQ di masing-masing fakultasnya. Admin harus *login* ke sistem terlebih dahulu untuk kemudian bisa memilih menu data muallim dan bisa melakukan penambahan, perubahan maupun menghapus data muallim program PDQ. Alur aktivitas tersebut disajikan pada Gambar 3.12.



Gambar 3.12 *Activity Diagram* Pengelolaan Data Muallim Fakultas

k. Melihat nilai ujian semester PDQ

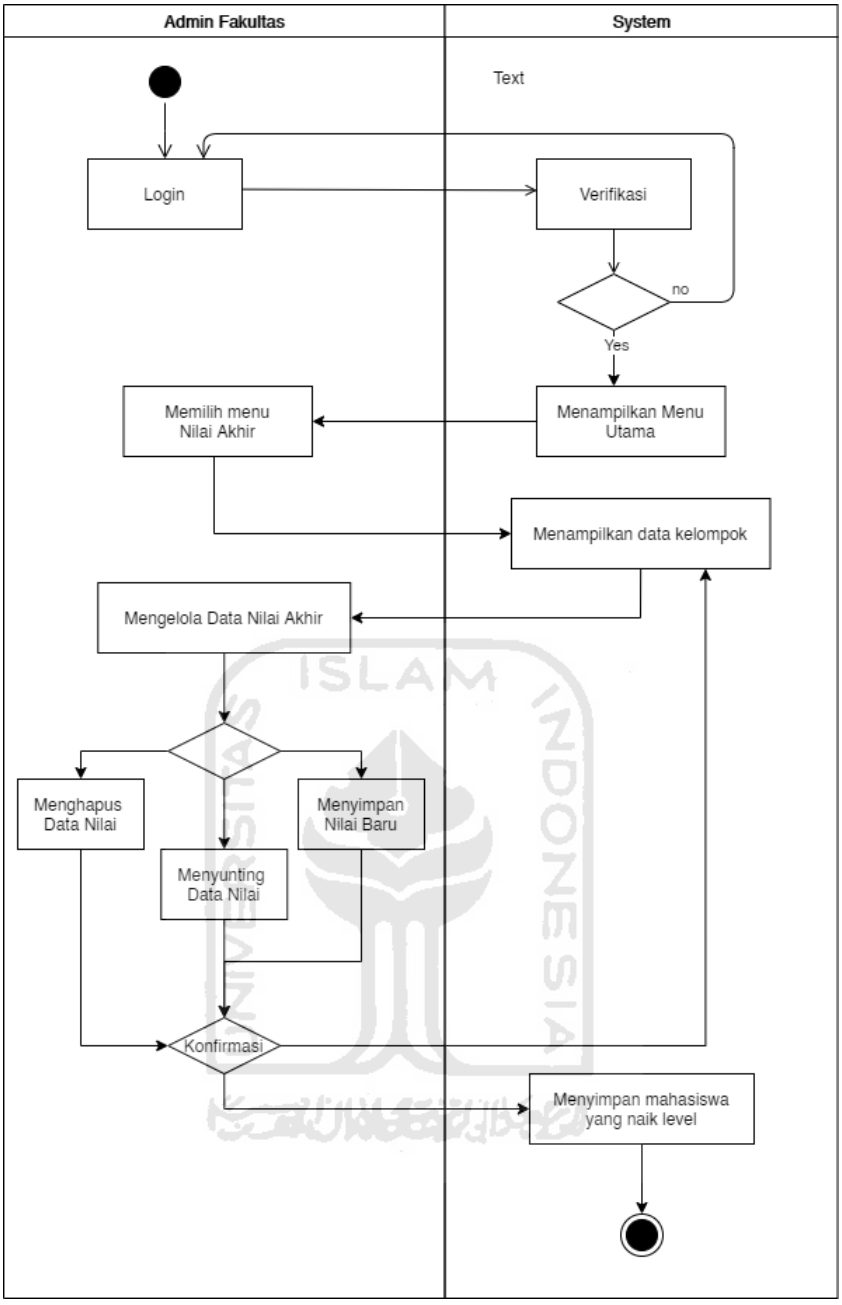
Activity diagram ini menggambarkan alur aktivitas dari Admin Fakultas untuk melihat nilai semester mahasiswa program PDQ di masing-masing fakultasnya. Admin harus *login* ke sistem terlebih dahulu untuk kemudian bisa memilih menu nilai semester dan bisa melihat nilai ujian semester mahasiswa program PDQ di fakultasnya. Alur aktivitas tersebut disajikan pada Gambar 3.13.



Gambar 3.13 *Activity Diagram* Pengelolaan Data Muallim Fakultas

l. Mengelola nilai akhir ujian PDQ

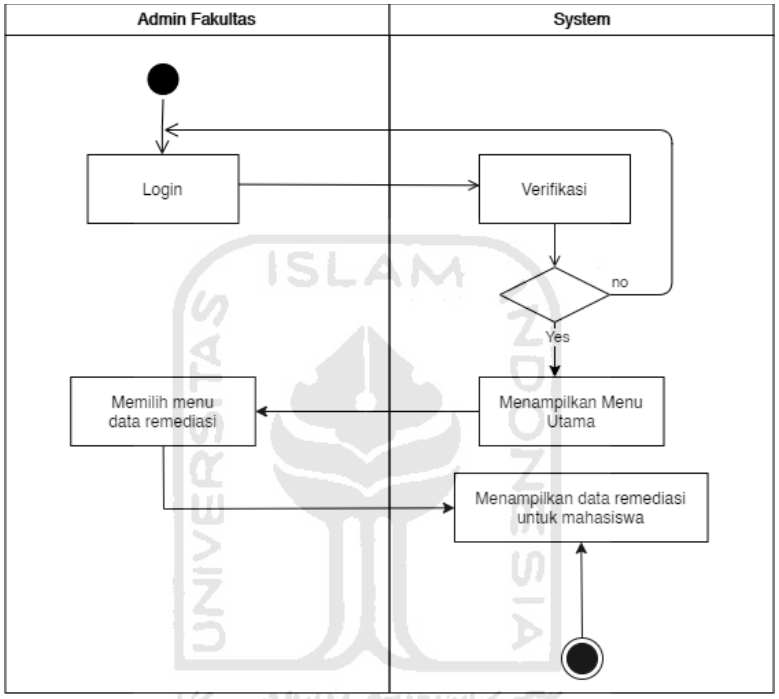
Activity diagram ini menggambarkan alur aktivitas dari Admin Fakultas untuk mengelola data nilai akhir mahasiswa program PDQ di masing-masing fakultasnya. Admin harus *login* ke sistem terlebih dahulu untuk kemudian bisa memilih menu nilai akhir dan bisa melakukan penambahan, perubahan maupun menghapus nilai akhir dari mahasiswa program PDQ. Alur aktivitas tersebut disajikan pada Gambar 3.14.



Gambar 3.14 *Activity Diagram* Pengelolaan Nilai Akhir Program PDQ

m. Melihat Data Mahasiswa Remediasi

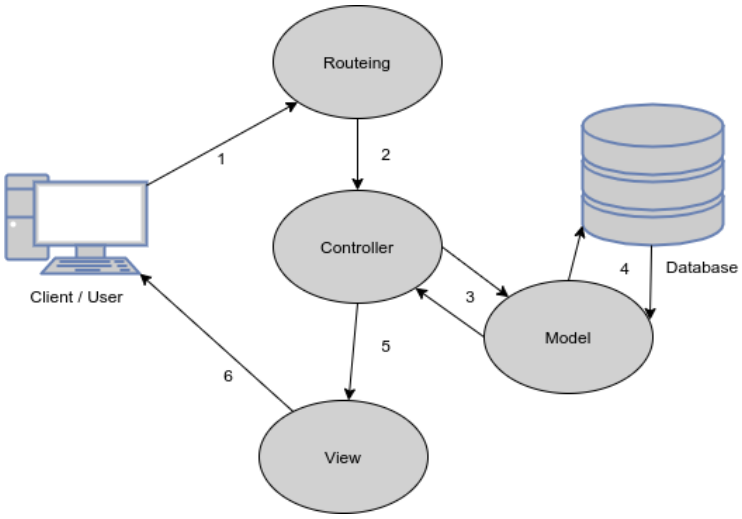
Activity diagram ini menggambarkan alur aktivitas dari Admin Fakultas untuk melihat data mahasiswa yang harus mengikuti proses remediasi program PDQ di masing-masing fakultasnya. Admin harus *login* ke sistem terlebih dahulu untuk kemudian bisa memilih menu remediasi dan bisa mengakses siapa saja mahasiswa program PDQ yang harus mengikuti proses remediasi. Alur aktivitas tersebut disajikan pada Gambar 3.15.



Gambar 3.15 *Activity Diagram* Melihat Data Mahasiswa yang Mengikuti Remediasi

C. Desain Arsitektur

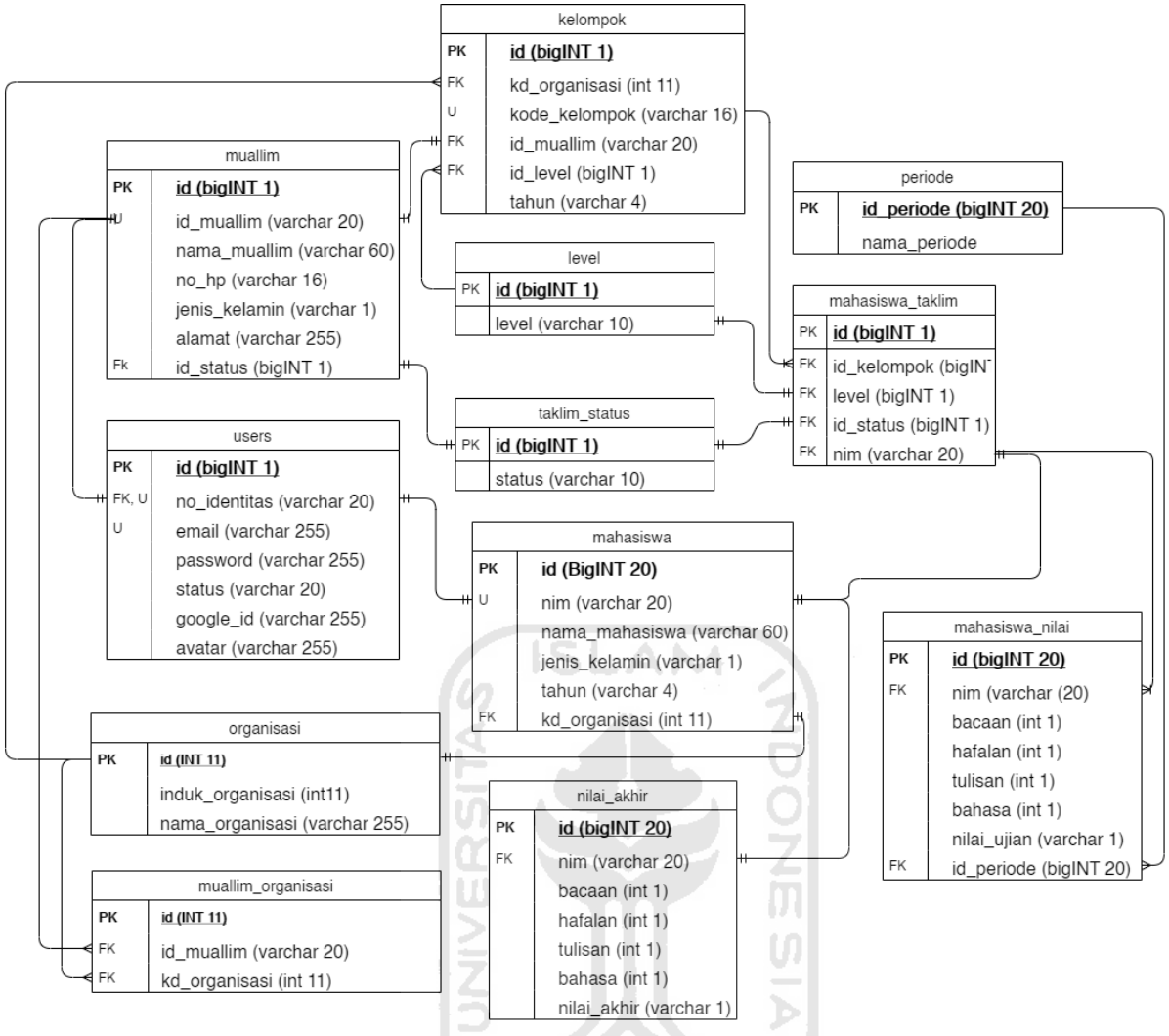
Pengembangan sistem menggunakan *framework* laravel yang mana menganut arsitektur MVC. Pertama komputer *client* akan melakukan request melalui *routing*. Sistem *route* akan mengarahkan kepada *controller* yang sesuai dengan *request* dari *client*. Apabila *request* membutuhkan data dari basis data maka akan diarahkan ke model terlebih dahulu sebelum mengakses langsung ke basis data. Sementara ketika *request* hanya membutuhkan tampilan halaman saja tanpa data maka *controller* langsung mengarahkan ke *view* untuk dikembalikan ke *client* sebagai *response*. Arsitektur sistem yang dikembangkan ditunjukan pada Gambar 3.16.



Gambar 3.16 Arsitektur Sistem Pengelolaan dan Monitoring Program PDQ

D. Entity Relationship Diagram

Basis data yang akan dibuat dan digunakan dalam sistem terlebih dahulu dibuat rancangan diagramnya terlebih dahulu. Diagram yang dibuat adalah *Entity Relationship Diagram* (ERD) yang mana memuat relasi antar entitas dalam basis data, dimana setiap entitas akan mewakili sebuah tabel. Relasi antar tabel yang terdapat pada Sistem Informasi Pengelolaan dan *Monitoring* Program PDQ Universitas Islam Indonesia disajikan pada Gambar 3.17.



Gambar 3.17 *ERD* Sistem Informasi Pengelolaan dan *Monitoring* Program PDQ

E. Deskripsi Tabel Basis Data

a. Rancangan Tabel Mahasiswa

Tabel mahasiswa merupakan tabel master yang berisi seluruh data mahasiswa Universitas Islam Indonesia. Adapun rancangan *field* yang digunakan pada tabel master mahasiswa disajikan pada Tabel 3.5.

Tabel 3.5 Rancangan Tabel Basis Data Mahasiswa

No	Nama Field	Tipe Data	Index	Keterangan
1	id	Bigint(20)	Primary Key	Id mahasiswa
2	nim	varchar(20)		Nomor induk mahasiswa
3	nama_mahasiswa	varchar(255)		Nama lengkap mahasiswa
4	jenis_kelamin	varchar(1)		Jenis kelamin mahasiswa

5	tahun	varchar(4)		Tahun angkatan mahasiswa
6	kd_organisasi	Int(2)	Foreign Key	Id jurusan mahasiswa

b. Rancangan Tabel Mahasiswa_taklim

Tabel mahasiswa_taklim merupakan tabel penghubung antara tabel satu dengan yang lainnya. Setiap data mahasiswa yang sudah diunggah hasil *Placement Test* Agamanya maka akan tersimpan di tabel ini. Selain menghubungkan tabel master mahasiswa, tabel ini juga menghubungkan tabel kelompok, level, dan status. Rancangan tabel mahasiswa_taklim ditunjukan pada Tabel 3.6

Tabel 3.6 Rancangan Tabel Basis Data Mahasiswa_taklim

No	Nama Field	Tipe Data	Index	Keterangan
1	id	Bigint(20)	Primary Key	Id mahasiswa
2	Nim	varchar(20)	Foreign key	Nomor induk mahasiswa
3	Kd_kelompok	varchar(16)	Foreign key	Kode kelompok mahasiswa
4	Id_level	int(1)	Foreign key	Id level mahasiswa taklim
5	Id_status	int(1)	Foreign key	Id status taklim mahasiswa

c. Rancangan Tabel Organisasi

Tabel organisasi merupakan tabel master yang memuat data fakultas dan jurusan yang ada di Universitas Islam Indonesia. Data nama fakultas dan jurusan dijadikan satu tabel menggunakan struktur hirarki. Rancangan tabel organisasi ditunjukan pada Tabel 3.7.

Tabel 3.7 Rancangan Tabel Basis Data Organisasi

No	Nama Field	Tipe Data	Index	Keterangan
1	Id	int(11)	Primary key	Id organisasi
2	Nama_organisasi	varchar(45)		Nama organisasi
3	Induk_organisasi	int(11)	Foreign key	Induk organisasi

d. Rancangan Tabel Muallim

Tabel muallim merupakan tabel master yang memuat data muallim program PDQ. Setiap data muallim yang dimasukan oleh *administrator* fakultas ke sistem maka akan disimpan pada tabel muallim. Rancangan tabel muallim ditunjukan pada Tabel 3.8.

Tabel 3.8 Rancangan Tabel Basis Data Muallim

No	Nama Field	Tipe Data	Index	Keterangan
1	id	Bigint(20)	Primary key	Id muallim
2	Id_muallim	varchar(20)		Nomor identitas muallim
3	Nama	varchar(45)		Nama muallim
4	Email	varchar(45)		Email muallim
5	No_hp	varchar(15)		Id level mahasiswa taklim
6	alamat	varchar(255)		Id status taklim mahasiswa
7	id_status	int(1)	Foreign Key	Id status taklim muallim

e. Rancangan Tabel Muallim_organisasi

Tabel muallim_organisasi dibuat agar *administrator* fakultas bisa mengetahui data muallim yang ada di fakultasnya masing-masing. Selain itu juga dibuat untuk mengantisipasi jika ada muallim yang tidak hanya membimbing satu fakultas saja. Maka dari itu dibuatlah tabel muallim_organisasi sebagai tabel penghubung antara tabel master muallim dan tabel master organisasi. Rancangan tabel muallim_organisasi ditunjukan pada Tabel 3.9.

Tabel 3.9 Rancangan Tabel Basis Data Muallim_organisasi

No	Nama Field	Tipe Data	Index	Keterangan
1	id	Bigint(20)	Primary Key	Id muallim
2	Id_muallim	varchar(20)	Foreign Key	Id Muallim
3	Kd_organisasi	Int(11)	Foreign Key	Id organisasi muallim

f. Rancangan Tabel Kelompok

Tabel kelompok merupakan tabel master yang memuat data kelompok program PDQ. Tabel ini terhubung dengan tabel organisasi dan juga tabel muallim. Rancangan tabel kelompok ditunjukan pada Tabel 3.10.

Tabel 3.10 Rancangan Tabel Basis Data Kelompok

No	Nama Field	Tipe Data	Index	Keterangan
1	id	Bigint(20)	Index	Id kelompok
2	Kd_kelompok	varchar(16)	Primary key	Kode kelompok
3	Kd_organisasi	int(11)	Foreign key	Kode organisasi kelompok
4	Id_muallim	Varchar(20)	Foreign Key	Id muallim
5	Angkatan	Varchar(4)		Tahun angkatan mahasiswa

g. Rancangan Tabel Level

Tabel level merupakan tabel master yang memuat nama level yang dimiliki oleh setiap mahasiswa program PDQ. Rancangan tabel level ditunjukan pada Tabel 3.11.

Tabel 3.11 Rancangan Tabel Basis Data Level

No	Nama Field	Tipe Data	Index	Keterangan
1	Id_level	int(1)	Primary Key	Id level
2	level	varchar(12)		Nama level

h. Rancangan Tabel Status

Tabel level merupakan tabel master yang memuat nama status yang dimiliki oleh setiap mahasiswa program PDQ maupun muallim. Rancangan tabel status ditunjukan pada Tabel 3.12.

Tabel 3.12 Rancangan Tabel Basis Data Status

No	Nama Field	Tipe Data	Index	Keterangan
1	Id	int(1)	Primary Key	Id status
2	status	varchar(12)		Nama status

i. Rancangan Tabel Mahasiswa_nilai

Tabel mahasiswa_nilai merupakan tabel yang memuat data nilai ujian semester mahasiswa program PDQ di setiap semesternya. Adapun rancangan dari tabel tersebut ditunjukan pada Tabel 3.13.

Tabel 3.13 Rancangan Tabel Basis Data Mahasiswa_nilai

No	Nama Field	Tipe Data	Index	Keterangan
1	Id	Bigint(20)	Primary Key	Id nilai akhir
2	Nim	varchar(20)	Foreign Key	Nomor induk mahasiswa
3	bacaan	Int(1)		Poin bacaan
4	Hafalan	Int(1)		Poin hafalan
5	Tulisan	Int(1)		Poin tulisan
6	bahasa	Int(1)		Poin bahasa
7	Nilai ujian	varchar(1)		Nilai akhir
8	Id_periode	BigInt(20)	Foreign key	Id untuk periode nilai semester

j. Rancangan Tabel Periode

Tabel periode merupakan tabel yang memuat data periode tahun ajaran. Informasi yang dimuat adalah tahun ajaran dan juga semester genap atau semester ganjil. Adapun rancangan dari tabel tersebut ditunjukan pada Tabel 3.14.

Tabel 3.14 Rancangan Tabel Basis Data Periode

No	Nama Field	Tipe Data	Index	Keterangan
1	Id	Bigint(20)	Primary Key	Id nilai akhir
2	nama_periode	varchar (45)		Nama periode tahun ajaran

k. Rancangan Tabel Nilai_akhir

Tabel nilai_akhir merupakan tabel yang memuat data nilai ujian akhir mahasiswa program PDQ. Adapun rancangan dari tabel tersebut ditunjukan pada Tabel 3.15.

Tabel 3.15 Rancangan Tabel Basis Data Nilai_akhir

No	Nama Field	Tipe Data	Index	Keterangan
1	Id	Bigint(20)	Primary Key	Id nilai akhir
2	Nim	varchar(20)	Foreign Key	Nomor induk mahasiswa
3	bacaan	Int(1)		Poin bacaan
4	Hafalan	Int(1)		Poin hafalan

5	Tulisan	Int(1)		Poin tulisan
6	bahasa	Int(1)		Poin bahasa
7	Nilai akhir	Varchar(1)		Nilai akhir

l. Rancangan Tabel Users

Tabel users merupakan tabel yang memuat data pengguna sistem. Adapun rancangan tabel tersebut ditunjukan pada Tabel 3.16.

Tabel 3.16 Rancangan Tabel Basis Data Users

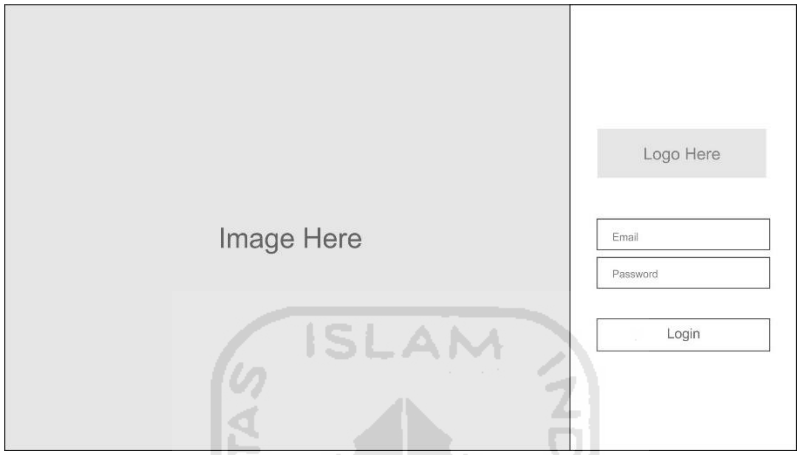
No	Nama Field	Tipe Data	Index	Keterangan
1	Id	Bigint(20)	Primary Key	Id organisasi
2	No_identitas	Varchar(16)	Foreign Key	Nomor identitas pengguna
3	Email	Varchar(32)		Email pengguna
4	Password	Hash(255)		Password pengguna
5	Google_id	Varchar(255)		Google id pengguna
6	Status	Varchar(12)		Status pengguna
7	Kd_organisasi	int(11)	Foreign key	Id organisasi pengguna

F. Rancangan Antarmuka

Rancangan antarmuka dibuat untuk menggambarkan tampilan antarmuka sistem. Rancangan antarmuka yang dibuat berupa *wireframe* yang dapat menyampaikan susunan ataupun *layout* dari setiap konten pada sebuah halaman.

a. Rancangan Halaman Login

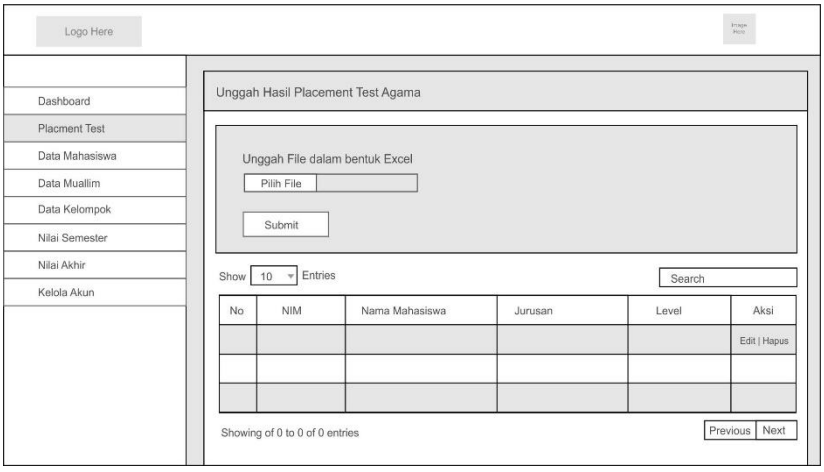
Pada rancangan antarmuka halaman login untuk admin DPPAI dan admin Fakultas disediakan sebuah *form* yang terdiri dari *input text* untuk email , *input password* untuk isian *password* dan sebuah *button* untuk melakukan *submit* data. Rancangan antarmuka halaman *login* disajikan pada Gambar 3.18.



Gambar 3.18 Rancangan Antarmuka Halaman *Login*

b. Rancangan Halaman Unggah Hasil *Placement Test* Agama

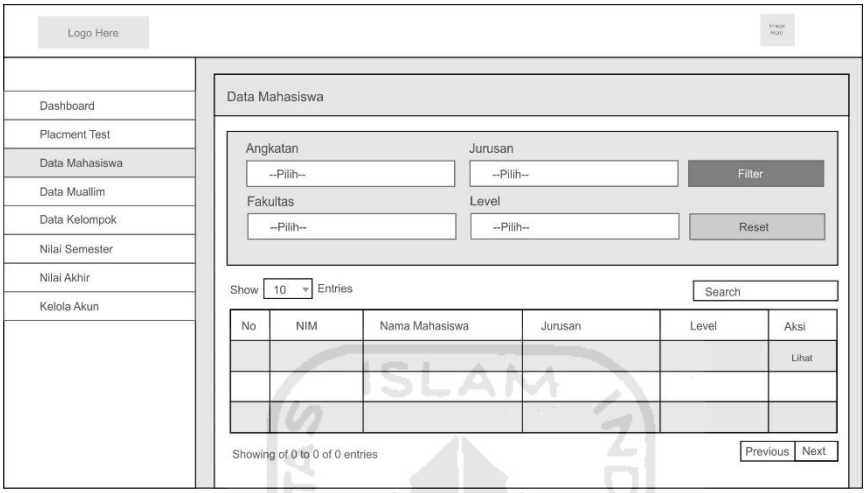
Pada Rancangan antarmuka halaman unggah hasil *Placement Test* Agama disediakan sebuah *input file* untuk memilih file yang akan diunggah. Selain itu juga disediakan sebuah *button* untuk melakukan *submit file* yang sebelumnya telah dipilih. Pada bagian bawah disediakan tabel untuk menampilkan data yang telah berhasil diunggah. Rancangan unggah hasil *Placement Test* Agama disajikan pada Gambar 3.19.



Gambar 3.19 Rancangan Antarmuka Halaman Unggah Hasil *Placement Test* Agama

c. Rancangan Halaman Data Mahasiswa

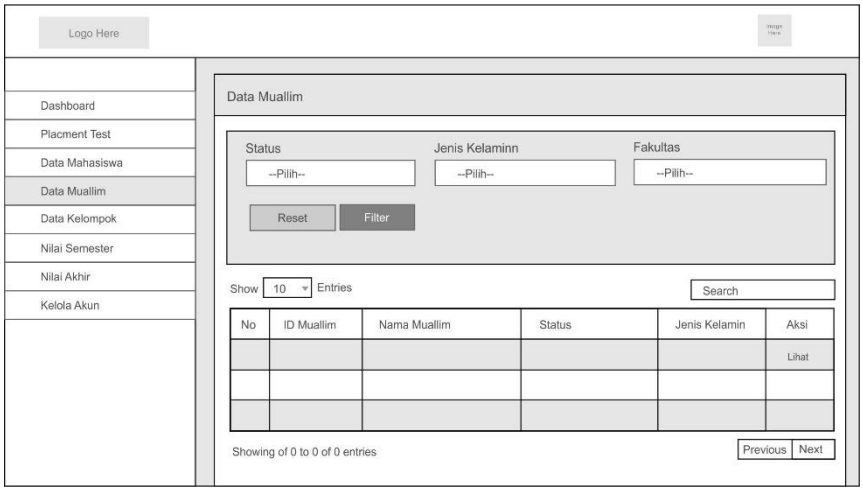
Pada rancangan antarmuka halaman data mahasiswa disediakan sebuah tabel yang menampilkan data mahasiswa. Selain itu juga disediakan sebuah kolom *filter* untuk melakukan pencarian data. Kolom aksi pada tabel memiliki fungsi untuk melihat informasi detail dari mahasiswa. Rancangan antarmuka halaman data mahasiswa disajikan pada Gambar 3.20.



Gambar 3.20 Rancangan Antarmuka Halaman Data Mahasiswa

d. Rancangan Halaman Data Muallim

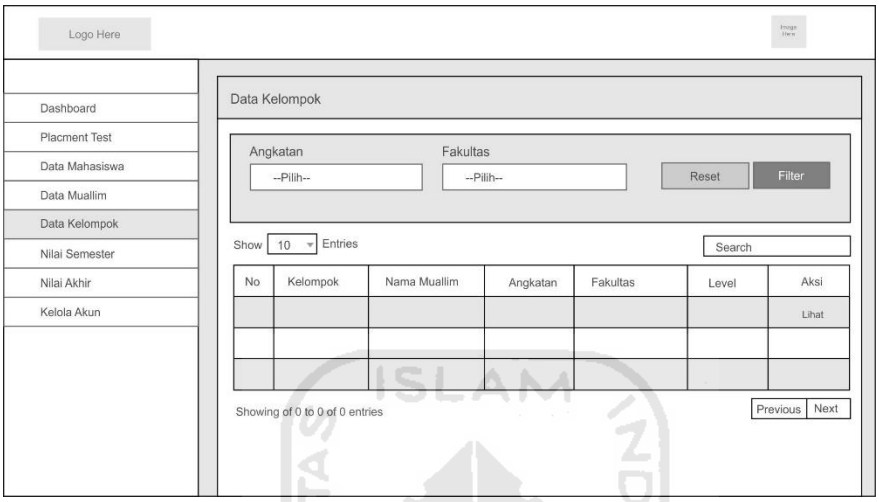
Pada rancangan antarmuka halaman data muallim disediakan sebuah tabel yang menampilkan data muallim. Selain itu juga disediakan sebuah kolom filter untuk melakukan pencarian data. Kolom aksi pada tabel memiliki fungsi untuk melihat informasi detail dari muallim. Rancangan antarmuka halaman data muallim disajikan pada Gambar 3.21.



Gambar 3.21 Rancangan Antarmuka Halaman Data Muallim

e. Rancangan Halaman Data Kelompok

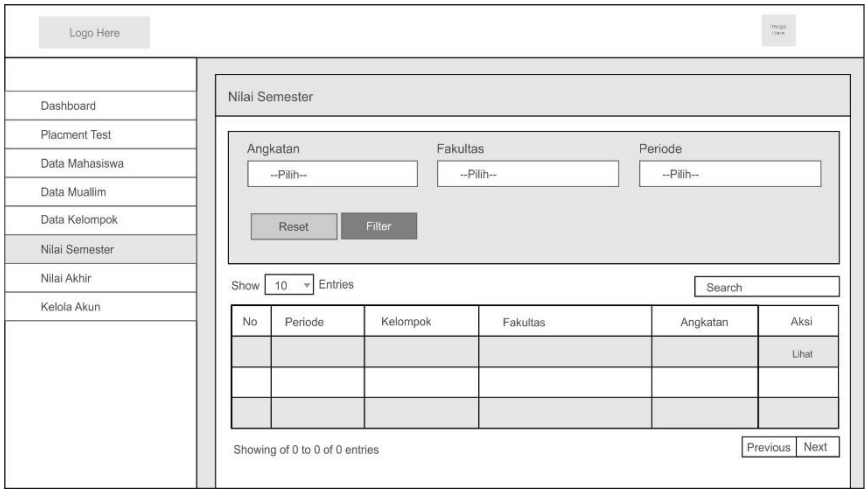
Pada rancangan antarmuka halaman data kelompok disediakan sebuah tabel yang menampilkan data kelompok. Selain itu juga disediakan sebuah kolom filter untuk melakukan pencarian data. Kolom aksi pada tabel memiliki fungsi untuk melihat informasi detail dari kelompok. Rancangan antarmuka halaman data muallim disajikan pada Gambar 3.22.



Gambar 3.22 Rancangan Antarmuka Halaman Data Kelompok

f. Rancangan Halaman Nilai Semester

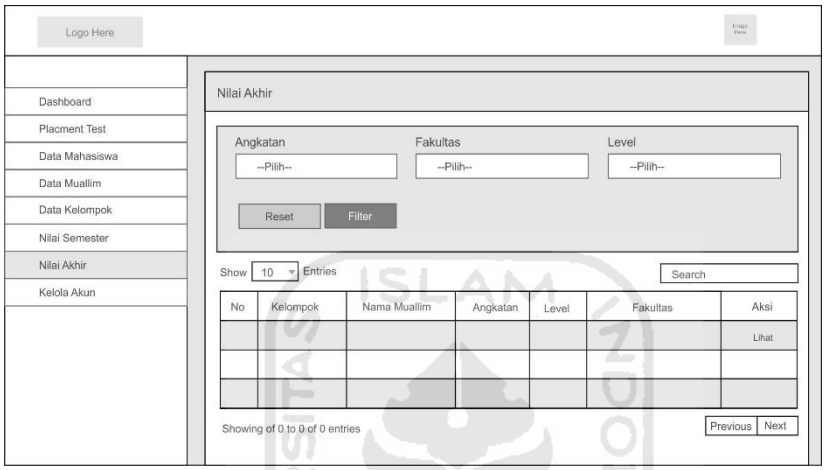
Pada rancangan antarmuka halaman nilai semester disediakan sebuah tabel yang menampilkan data nilai semester per kelompok. Selain itu juga disediakan sebuah kolom filter untuk melakukan pencarian data. Kolom aksi pada tabel memiliki fungsi untuk melihat informasi nilai mahasiswa dari suatu kelompok. Rancangan antarmuka halaman nilai semester disajikan pada Gambar 3.23.



Gambar 3.23 Rancangan Antarmuka Halaman Nilai Semester

g. Rancangan Halaman Nilai Akhir

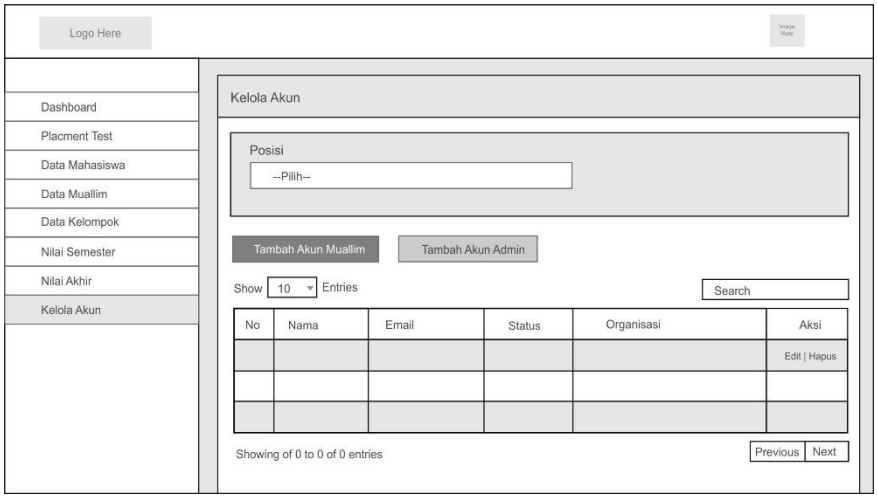
Pada rancangan antarmuka halaman nilai akhir disediakan sebuah tabel yang menampilkan data nilai akhir per kelompok. Selain itu juga disediakan sebuah kolom filter untuk melakukan pencarian data. Kolom aksi pada tabel memiliki fungsi untuk melihat informasi nilai akhir mahasiswa dari suatu kelompok. Rancangan antarmuka halaman nilai akhir disajikan pada Gambar 3.24.



Gambar 3.24 Rancangan Antarmuka Halaman Nilai Akhir

h. Rancangan Halaman Kelola Akun

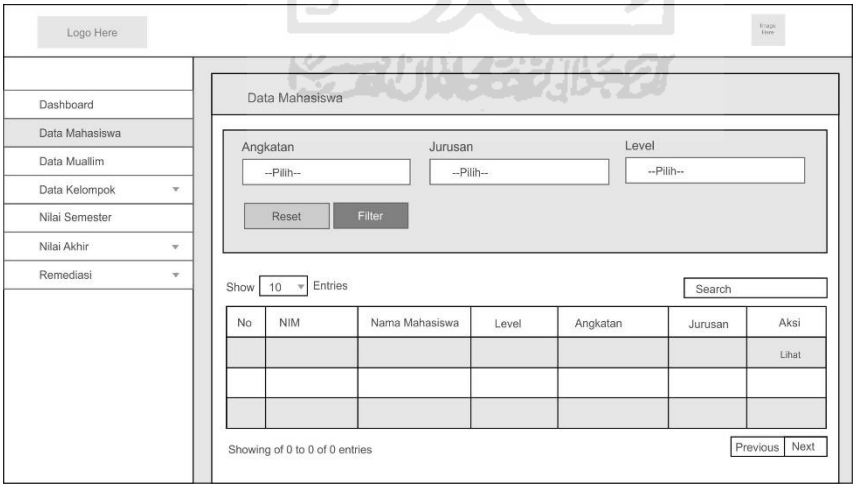
Pada rancangan antarmuka halaman kelola disediakan sebuah tabel yang menampilkan data pengguna berdasarkan posisi atau statusnya. Selain itu juga disediakan sebuah kolom filter untuk melakukan pencarian data. Kolom aksi pada tabel memiliki fungsi untuk melakukan perubahan akun pengguna dan juga menghapus akun pengguna. Ada dua *button* untuk menambahkan akun admin fakultas dan juga akun muallim. Rancangan antarmuka halaman nilai akhir disajikan pada Gambar 3.25



Gambar 3.25 Rancangan Antarmuka Halaman Kelola Akun

i. Rancangan Halaman Data Mahasiswa Fakultas

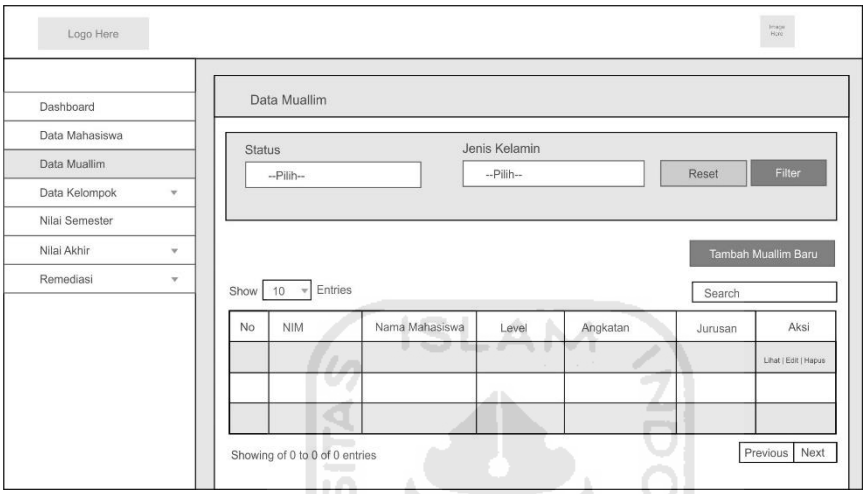
Pada rancangan antarmuka halaman data mahasiswa disediakan sebuah tabel yang menampilkan data mahasiswa sesuai dengan admin fakultas. Selain itu juga disediakan sebuah kolom filter untuk melakukan pencarian data. Kolom aksi pada tabel memiliki fungsi untuk melihat informasi detail dari mahasiswa. Rancangan antarmuka halaman data mahasiswa fakultas disajikan pada Gambar 3.26



Gambar 3.26 Rancangan Antarmuka Halaman Data Mahasiswa Fakultas

j. Rancangan Halaman Data Muallim Fakultas

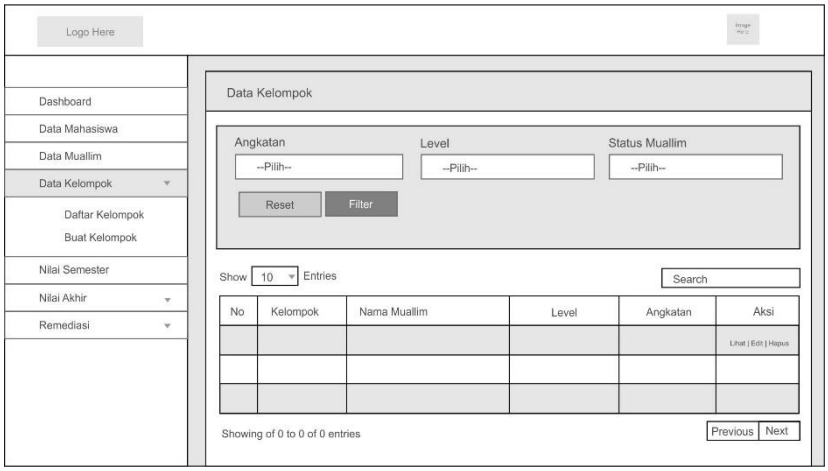
Pada rancangan antarmuka halaman data muallim disediakan sebuah tabel yang menampilkan data muallim sesuai dengan admin fakultas. Selain itu juga disediakan sebuah kolom filter untuk melakukan pencarian data. Kolom aksi pada tabel memiliki fungsi untuk melihat informasi detail dari muallim. Rancangan antarmuka halaman data muallim fakultas disajikan pada Gambar 3.27



Gambar 3.27 Rancangan Antarmuka Halaman Data Muallim Fakultas

k. Rancangan Halaman Data Kelompok Fakultas

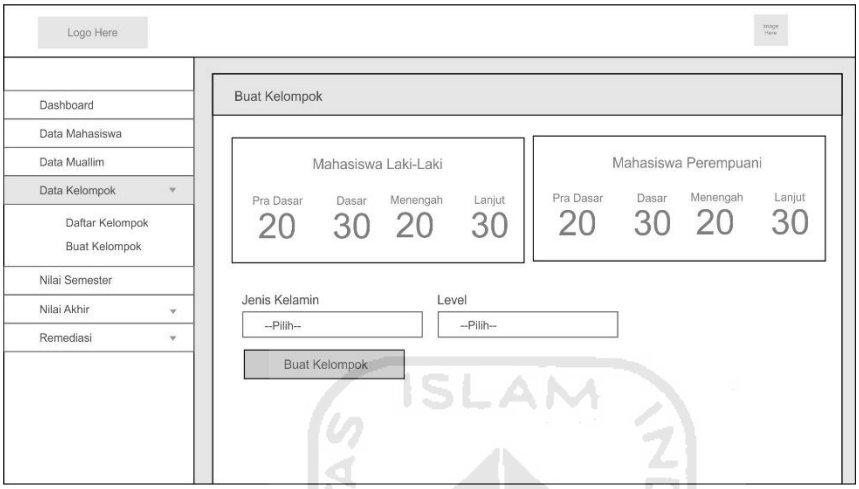
Pada rancangan antarmuka halaman data kelompok disediakan sebuah tabel yang menampilkan data kelompok sesuai dengan admin fakultas. Selain itu juga disediakan sebuah kolom filter untuk melakukan pencarian data. Kolom aksi pada tabel memiliki fungsi untuk melihat informasi detail dari kelompok. Rancangan antarmuka halaman data kelompok fakultas disajikan pada Gambar 3.28



Gambar 3.28 Rancangan Antarmuka Halaman Data Kelompok

l. Rancangan Halaman Buat Kelompok

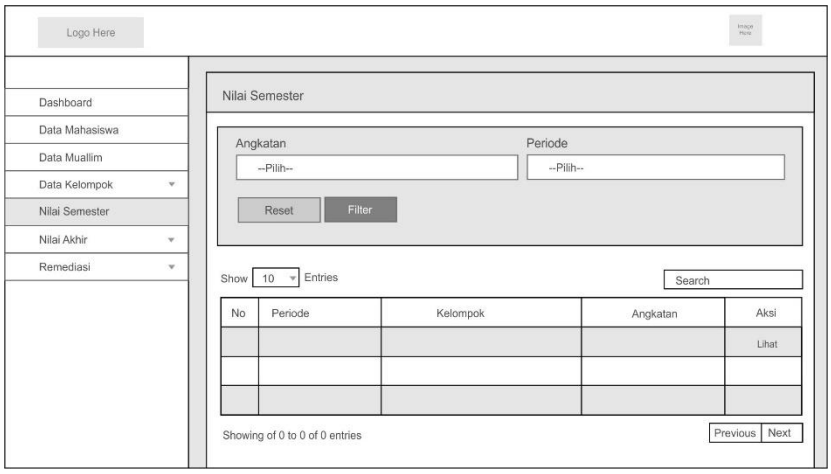
Pada rancangan antarmuka halaman buat kelompok disajikan informasi mengenai jumlah mahasiswa yang belum mendapatkan kelompok. Kemudian disediakan sebuah *select option* jenis kelamin dan level untuk memilih kelompok yang akan dibuat. Rancangan antarmuka buat kelompok disajikan pada Gambar 3.29



Gambar 3.29 Rancangan Antarmuka Halaman Buat Kelompok

m. Rancangan Halaman Nilai Semester Fakultas

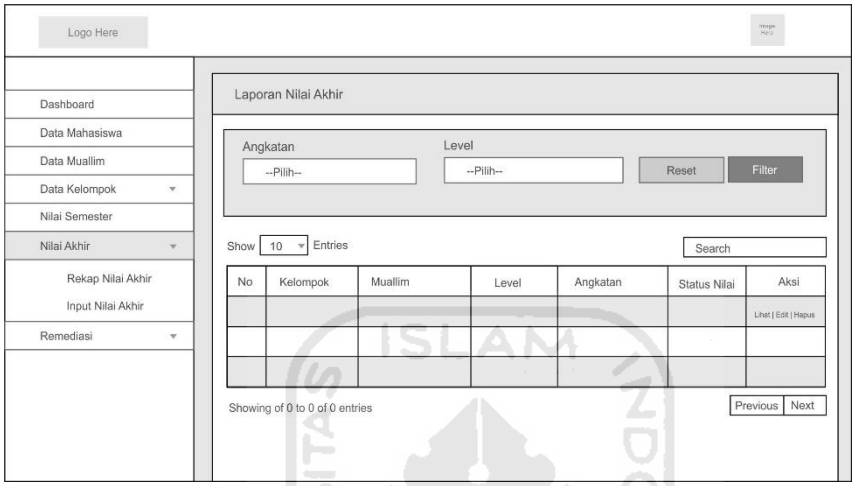
Pada rancangan antarmuka halaman nilai semester disediakan sebuah tabel yang menampilkan data nilai semester sesuai dengan admin fakultas. Selain itu juga disediakan sebuah kolom filter untuk melakukan pencarian data. Kolom aksi pada tabel memiliki fungsi untuk melihat informasi nilai semester mahasiswa per kelompok. Rancangan antarmuka halaman nilai semester fakultas disajikan pada Gambar 3.30.



Gambar 3.30 Rancangan Antarmuka Halaman Nilai Semester Fakultas

n. Rancangan Halaman Nilai Akhir Fakultas

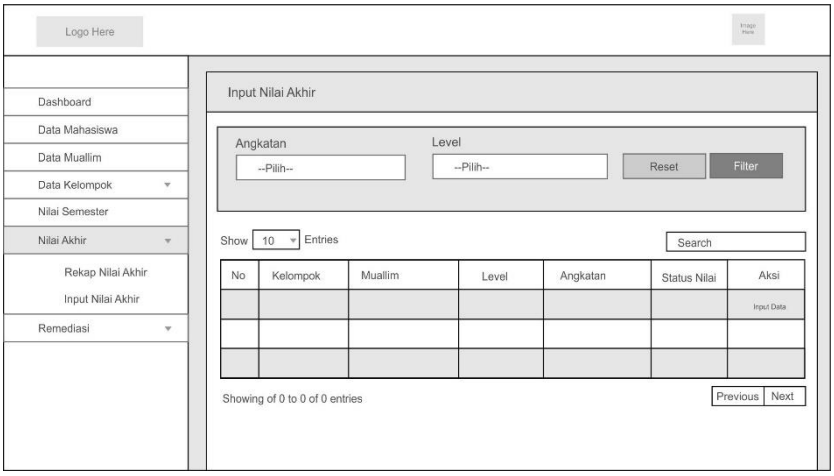
Pada rancangan antarmuka halaman nilai akhir disediakan sebuah tabel yang menampilkan data nilai akhir per kelompok di fakultasnya masing masing. Selain itu juga disediakan sebuah kolom filter untuk melakukan pencarian data. Kolom aksi pada tabel memiliki fungsi untuk melihat informasi nilai akhir mahasiswa per kelompok. Rancangan antarmuka halaman nilai akhir fakultas disajikan pada Gambar 3.31



Gambar 3.31 Rancangan Antarmuka Nilai Akhir Fakultas

o. Rancangan Halaman Input Nilai Akhir Fakultas

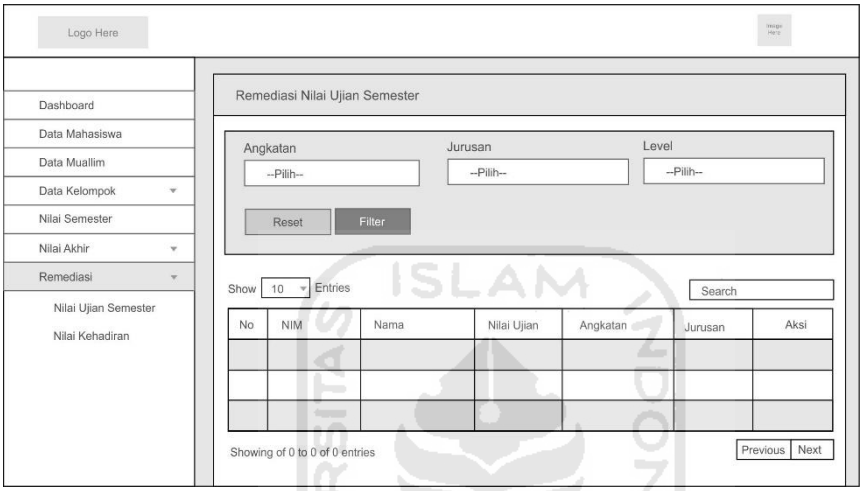
Pada rancangan antarmuka halaman input nilai akhir disediakan sebuah tabel yang menampilkan kelompok yang belum dimasukan nilai akhirnya. Selain itu juga disediakan sebuah kolom filter untuk melakukan pencarian data. Kolom aksi pada tabel memiliki fungsi untuk memasukan data nilai akhir mahasiswa per kelompok. Rancangan antarmuka halaman *input* nilai akhir fakultas disajikan pada Gambar 3.32



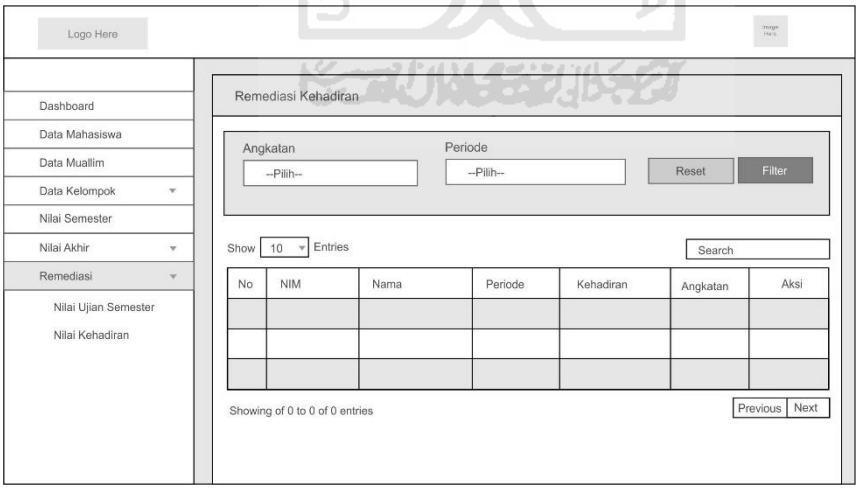
Gambar 3.32 Rancangan Antarmuka Halaman *Input* Nilai Akhir Fakultas

p. Rancangan Halaman Remediasi Fakultas

Pada rancangan antarmuka halaman remediasi dibedakan antara halaman remediasi untuk mahasiswa yang tidak memenuhi standar nilai semester dan juga yang tidak memenuhi jumlah kehadiran minimal. Disediakan sebuah tabel yang menampilkan data mahasiswa yang harus mengikuti proses remediasi. Selain itu juga disediakan sebuah kolom filter untuk melakukan pencarian data. Rancangan antarmuka remediasi fakultas disajikan pada Gambar 3.33 dan Gambar 3.34



Gambar 3.33 Rancangan Antarmuka Halaman Remediasi Nilai Ujian Semester



Gambar 3.34 Rancangan Antarmuka Halaman Remediasi Berdasarkan Kehadiran

G. Perancangan Pengujian

Pada penelitian ini, penulis melakukan dua jenis pengujian terhadap sistem yang dikembangkan. Pengujian yang pertama adalah *User Acceptance Testing* dan yang kedua adalah pengujian efisiensi. Rincian pengujian yang dilakukan ditunjukan pada Tabel 3.17.

Tabel 3.17 Rancangan Pengujian

NO	Nama Pengujian	Teknik	Tujuan
1	*User Acceptance Testing*	Wawancara dengan *stakeholder* (DPPAI)	Mengetahui apakah sistem yang dikembangkan memberikan kemudahan atau belum dengan memberikan pertanyaan-pertanyaan kepada *stakeholder* terkait dengan kemudahan penggunaan sistem.
2	Pengujian Efisiensi	Menggunakan *GTMetrix Tool*	Mengetahui apakah sistem yang sudah dikembangkan efisien atau belum. Baik dari segi kecepatan respons maupun penggunaan data.

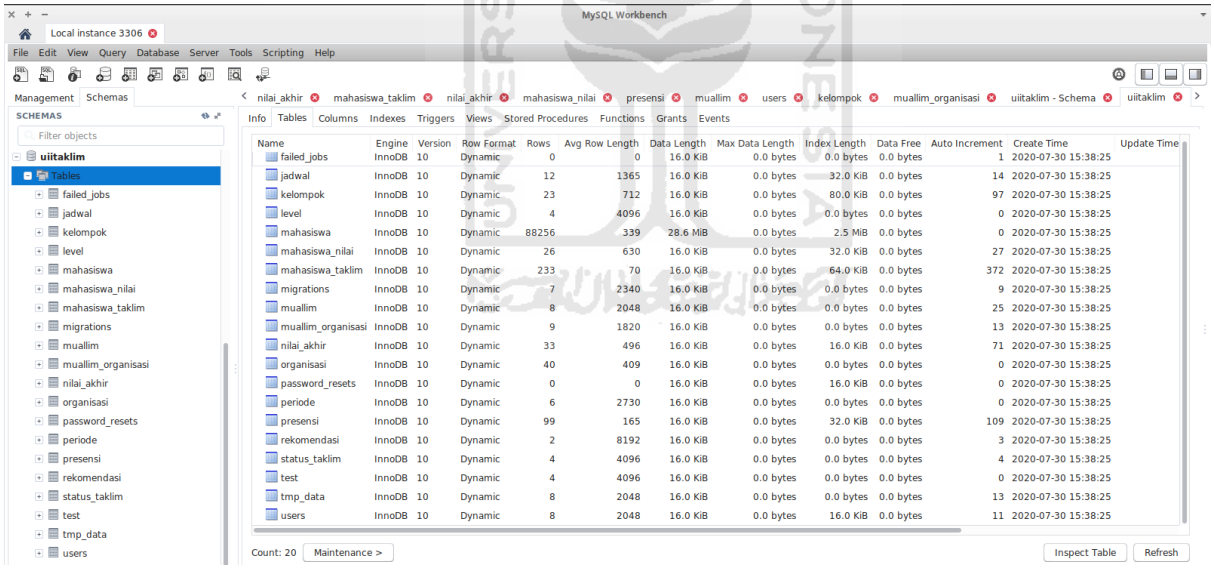
BAB IV
IMPLEMENTASI DAN PENGUJIAN

4.1 Implementasi

Pengembangan Sistem Informasi Pengelolaan Kegiatan dan *Monitoring* Program Pengembangan Diri Qur'ani (PDQ) menggunakan bahasa pemrograman *PHP* dengan memanfaatkan *framework* Laravel. Berikut tahapan yang dikerjakan dalam pengembangan sistem ini.

4.1.1 Membuat Basis Data

Tahap implementasi yang pertama adalah menerjamahkan ERD menjadi basis data yang bisa digunakan oleh sistem. Perangkat lunak digunakan untuk membuat basis data tersebut adalah *MySQL Workbench*. Basis data yang dibuat menggunakan *MySQL Workbench* ditunjukan pada Gambar 4.1.



Gambar 4.1 MySQL Workbench

4.1.2 Membuat Model

Model merupakan sebuah kelas yang menjembatani antara *controller* dengan *database*. *Model* disimpan pada direktori /App/Models. Salah satu model yang dibuat pada pengembangan sistem ini adalah model untuk tabel muallim. Berikut Gambar 4.2 menampilkan *source code* dari kelas model muallim.

```
<?php

namespace App\Models;
use Illuminate\Notifications\Notifiable;
use Illuminate\Database\Eloquent\Model;

class Muallim extends Model
{
    use Notifiable;

    protected $fillable = ['nama', 'no_hp', 'id_muallim', 'alamat', 'jenis_kelamin', 'status'];

    protected $table = 'muallim';

    const CREATED_AT = 'tgl_input';
    const UPDATED_AT = 'tgl_update';
}
```

Gambar 4.2 Contoh Kelas Model Data Muallim

Sebuah *model* mewakili satu tabel dalam basis data, yang mana *model* tersebut bisa digunakan untuk kebutuhan manipulasi data dengan mudah. Gambar 4.2 merupakan salah satu kelas model untuk proses manipulasi data pada tabel muallim.

4.1.3 Membuat Controller

Controller merupakan sebuah kelas yang bertugas untuk mengelola segala urusan logika program dari sistem. Pembuatan *controller* dilakukan melaui *terminal* dengan memanfaatkan *artisan command* yang disediakan oleh *framework* Laravel.

```
$ php artisan make controller:DataMuallimController -r
```

Gambar 4.3 Contoh *Command* Untuk Membuat *Controller* Datamuallim

Laravel memiliki berbagai macam *command* yang mempermudah dan mempercepat proses *coding.* Gambar 4.3 merupakan contoh *command artisan* dari laravel yang dimanfaatkan untuk membuat kelas *controller* DataMuallim. Hasil dari *command* tersebut adalah kelas *controller* DataMuallim akan otomatis dibuat beserta kerangka *function* dan *method* utama yang dibutuhkan. Contoh kelas *controller* untuk kelas DataMuallim ditunjukan pada Gambar 4.4.

```
<?php

namespace App\Http\Controllers\Fakultas;
```

```
use App\Http\Controllers\Controller;
use Illuminate\Http\Request;
use App\Models\Muallim;
use DataTables;
use DB;

class DataMuallimController extends Controller
{

    public function index()
    {

        return view('fakultas.datamuallim.index');
    }

    public function create()
    {
        $model = new Muallim();
        return view('fakultas.datamuallim.form', compact('model'));
    }

    public function store(Request $request)
    {
        $user = auth()->user();
        $this->validate($request, [
        'nama' => 'required|string|max:255',
        'no_hp' => 'required|string|max:12'
    ]);

    $model = Muallim::create($request->all());
    $organisasi = DB::table('muallim_organisasi')->insert(
    ['id_muallim' => $request->id_muallim,
    'kd_organisasi' => $user->kd_organisasi]
    );

    return $model;
    }

    public function show($id)
    {
        $model = Muallim::findOrFail($id);
        return view('fakultas.datamuallim.show', compact('model'));
    }

    public function edit($id)
    {
        $model = Muallim::findOrFail($id);
        return view('fakultas.datamuallim.form', compact('model'));
    }
```

```
    public function update(Request $request, $id)
    {
        $this->validate($request, [
        'nama' => 'required|string|max:255',
        'no_hp' => 'required|string|max:255',

    ]);

    $model = Muallim::findOrFail($id);
    $model->update($request->all());
    }

    public function destroy($id)
    {
        $model = Muallim::findOrFail($id);
        $model->delete();
    }

}
```

Gambar 4.4 Contoh Kode Program Kelas *Controller* Datamuallim

4.1.4 Membuat Route

Route merupakan sebuah cara untuk membuat jalur sebuah *request URL* dari sistem. *Route* berisi sebuah *endpoint* yang akan mengarahkan *request* ke tempat tujuannya. Seluruh *endpoint* didefinisikan pada file /routes/web.php. Contoh deklarasi *route* ditunjukan pada Gambar 4.5.

```
Route::namespace('Fakultas')
    ->prefix('fakultas')
    ->name('fakultas.')
    ->group(function (){
        Route::resource('/datamahasiswa', 'DataMahasiswaController');
        Route::resource('/datamuallim', 'DataMuallimController');
        Route::resource('/kelompoktaklim', 'KelompokTaklimController');
        Route::resource('/nilaiakhir', 'NilaiController');
        Route::resource('/inputNilaiAkhir', 'InputNilaiController');
        Route::resource('/remediasi', 'RemediasiController');

    }
```

Gambar 4.5 Contoh *Endpoint* Fakultas Untuk Mengakses Sistem.

4.1.5 Membuat View

Sebuah *view* berguna untuk menampung kode program dari seuatu tampilan halaman *website. Framework Laravel* menyimpan *view* pada direktori *resources/views*.

A. Halaman *Login*

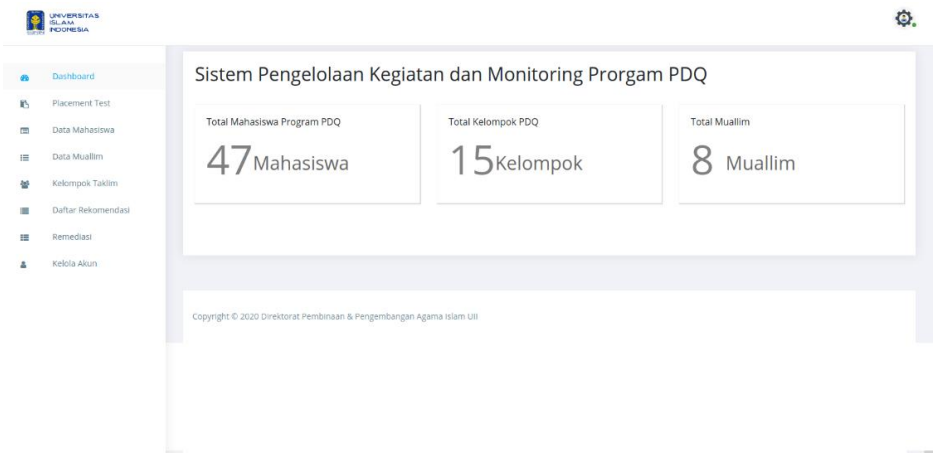
Administrator Fakultas dan DPPAI masuk ke sistem melalui halaman *login* dengan menggunakan akun email yang sudah terdaftar pada sistem. Halaman login ditunjukan pada Gambar 4.6.



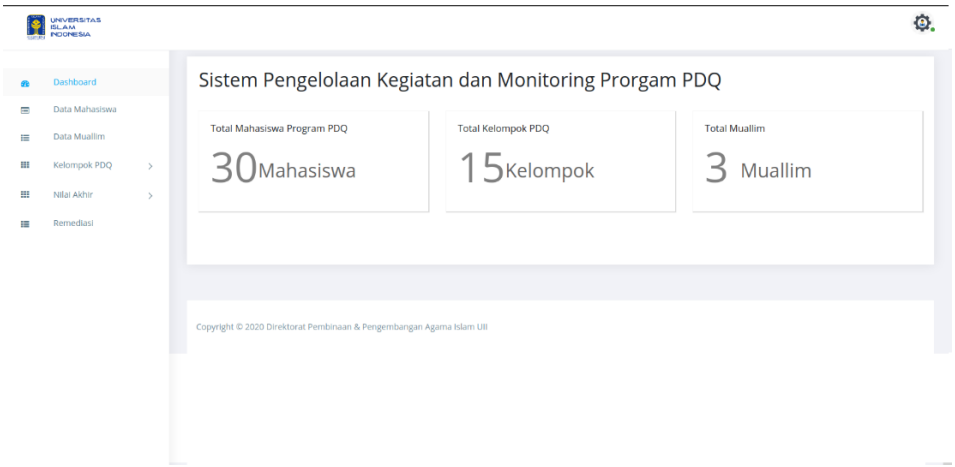
Gambar 4.6 Halaman Login Sistem

B. Halaman Dashboard

Halaman ini memuat berbabagai ringkasan data yang tersimpan pada sistem, seperti jumlah mahasiswa yang mengikuti program PDQ, jumlah muallim, jumlah kelompok, dan berbagai data lain yang menunjukan indikator penting.Data yang ditampilkan pada *dashboard* yang dimiliki oleh *administrator* DPPAI mencakup data ringkasan dari seluruh fakultas di UII. Sedangkan untuk *administrator* fakultas hanya mencakup data di masing masing fakultas saja. Halaman *dashboard administrator* DPPAI ditunjukan pada Gambar 4.7 sementara *dashboard administrator* fakultas ditunjukan pada Gambar 4.8



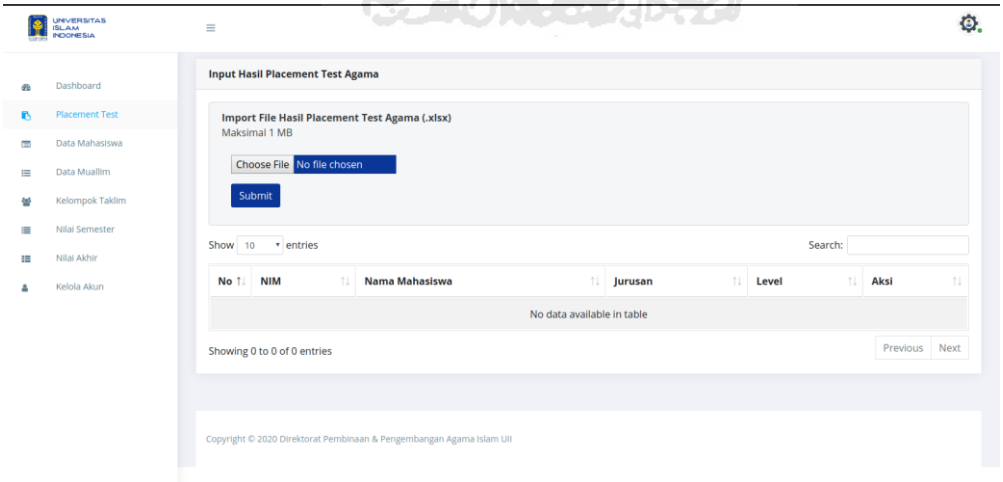
Gambar 4.7 Halaman *Dashboard* Administrator DPPAI



Gambar 4.8 Halaman *Dashboard Administrator* Fakultas

C. Halaman Unggah Hasil *Placement Test* Agama

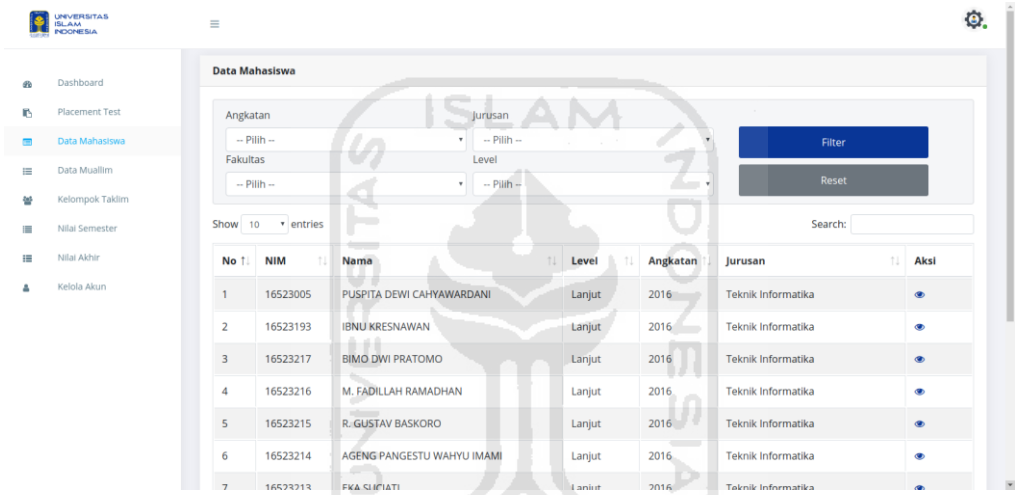
Halaman ini berguna untuk administrator DPPAI mengunggah hasil *Placment Test* Agama dalam bentuk *file excel* yang sudah diterima dari seluruh fakultas. Ketika proses unggah berhasil maka sistem akan menyimpan nim dari mahasiswa beserta level yang didapat dari proses *placement test* agama. Halaman unggah hasil *Placement Test* Agama ditunjukan pada Gambar 4.9



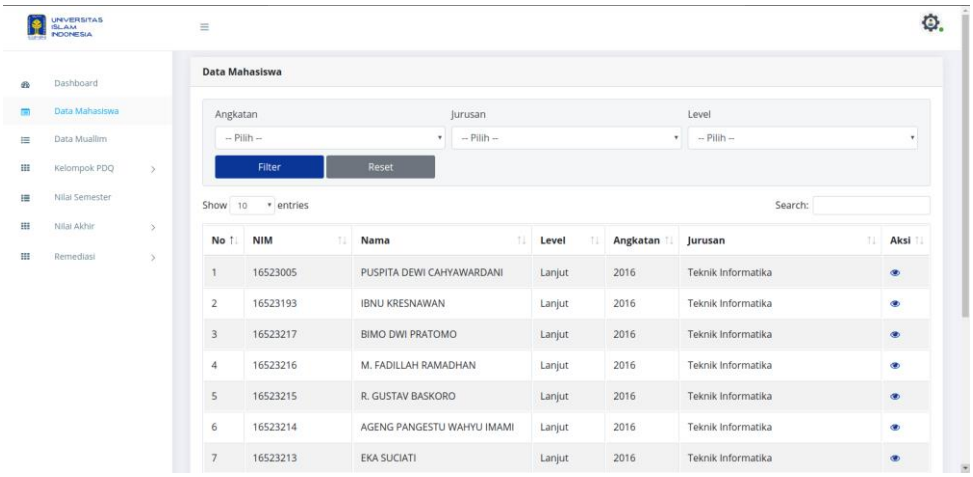
Gambar 4.9 Halaman Unggah Hasil *Placement Test Agama*

D. Halaman Data Mahasiswa

Pada halaman data mahasiswa *administrator* DPPAI dapat melihat data mahasiswa peserta program PDQ diseluruh fakultas yang sebelumnya telah diperbarui levelnya melalui laman unggah hasil *Placment Test* Agama. Sementara *administrator* fakultas hanya dapat melihat data mahasiswa yang ada dalam fakultasnya saja. Pada halaman ini terdapat fitur pencarian mahasiswa dengan kolom filter dan juga kolom pencarian manual untuk memudahkan dalam melakukan pencarian data mahasiswa. Halaman data mahasiswa *administrator* DPPAI ditunjukan pada Gambar 4.10 sementara *administrator* fakultas ditunjukan pada Gambar 4.11.



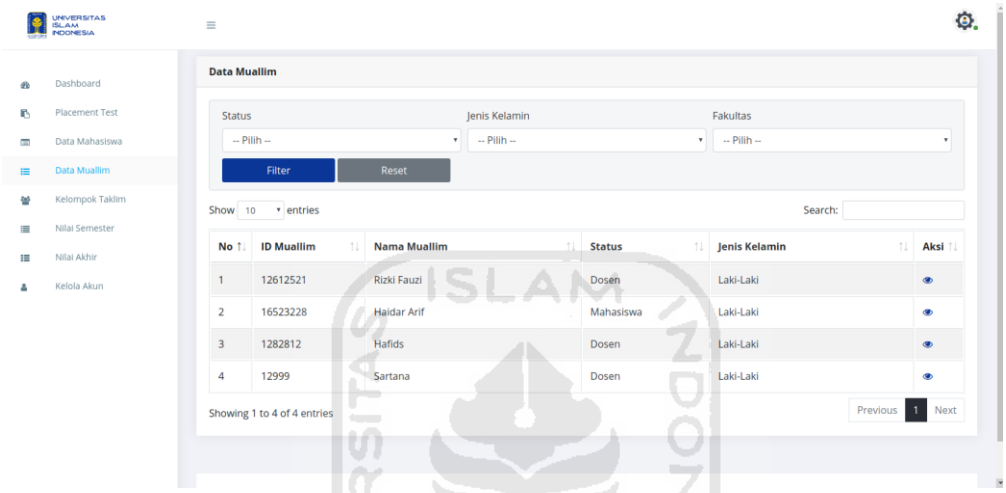
Gambar 4.10 Halaman Data Mahasiswa *Administrator* DPPAI



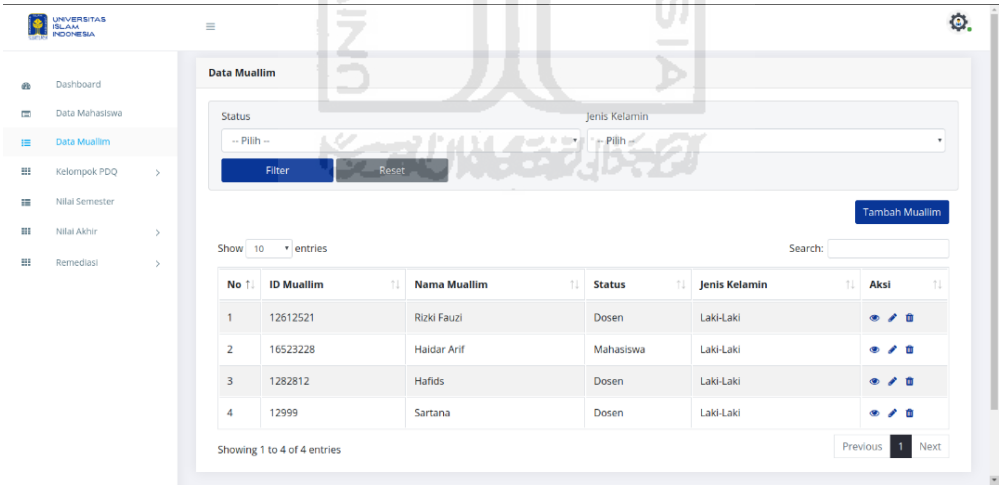
Gambar 4.11 Halaman Data Mahasiswa *Administrator* Fakultas

H. Halaman Data Muallim

Pada halaman data muallim administrator DPPAI dapat mencari dan melihat datam muallim dari seluruh fakultas. Namun hanya sebatas melihat saja tidak bisa untuk menambahakan, mengubah, maupun menghapus. Sementara pada menu data muallim *administrator* fakultas bisa menambahkan data muallim baru, mengubah dan menghapus. Halaman data muallim *administrator* DPPAI ditunjukan pada Gambar 4.12 sementara data muallim *administrator* fakultas ditunjukan pada Gambar 4.13.

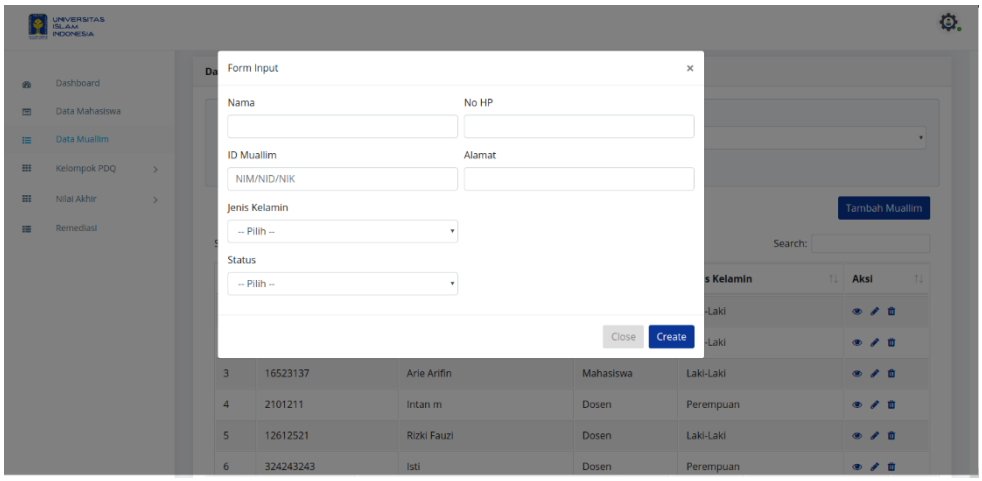


Gambar 4.12 Halaman Data Muallim *Administrator* DPPAI



Gambar 4.13 Halaman Data Muallim *Administrator* Fakultas

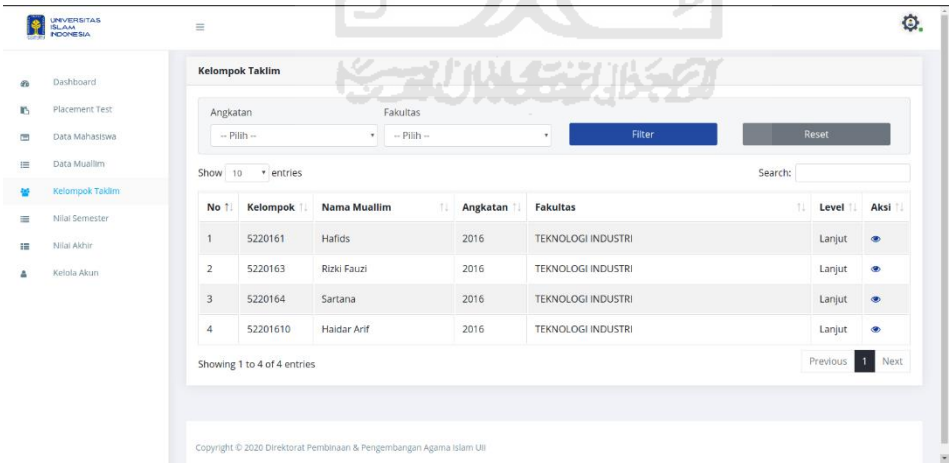
Administrator fakultas memiliki tugas untuk memasukan data muallim yang ada di fakultasnya masing-masing. Berikut Gambar 4.14 menampilkan *form* untuk pengisian data muallim.



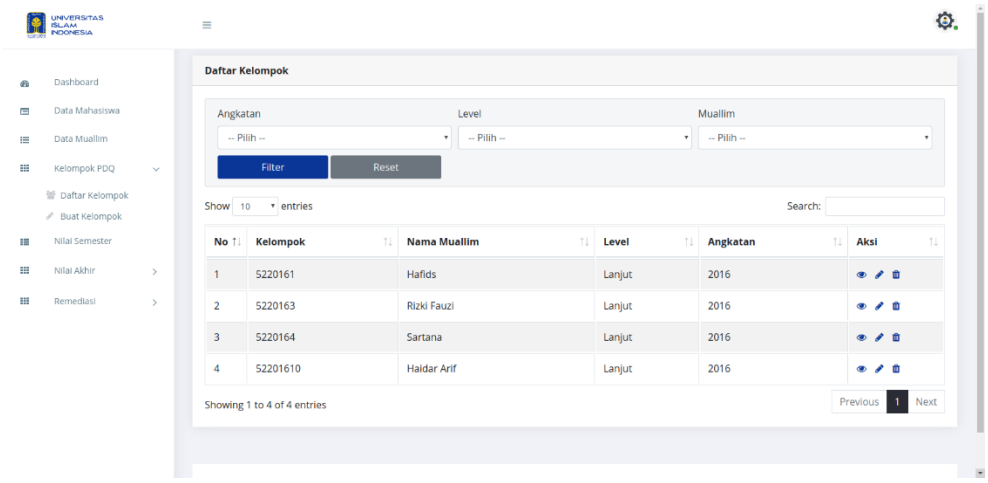
Gambar 4.14 *Form* Tambah Data Muallim *Administrator* Fakultas

I. Halaman Data Kelompok

Pada halaman data kelompok *administrator* DPPAI dapat mencari dan melihat datam muallim dari seluruh fakultas. Namun hanya sebatas melihat saja tidak bisa untuk menambahakan, mengubah, maupun menghapus. Sementara pada menu data kelompok *administrator* fakultas bisa menambahkan data kelompok baru, mengubah dan menghapus. Halaman data kelompok *administrator* DPPAI ditunjukan pada Gambar 4.15 , sementara halaman data kelompok *administrator* fakultas ditunjukan pada Gambar 4.16.

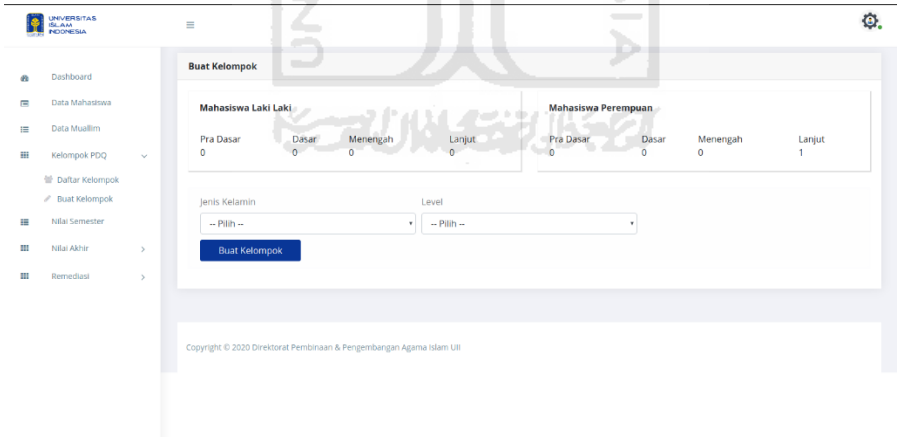


Gambar 4.15 Halaman Data Kelompok *Administrator* DPPAI



Gambar 4.16 Halaman Data Kelompok *Administrator* Fakultas

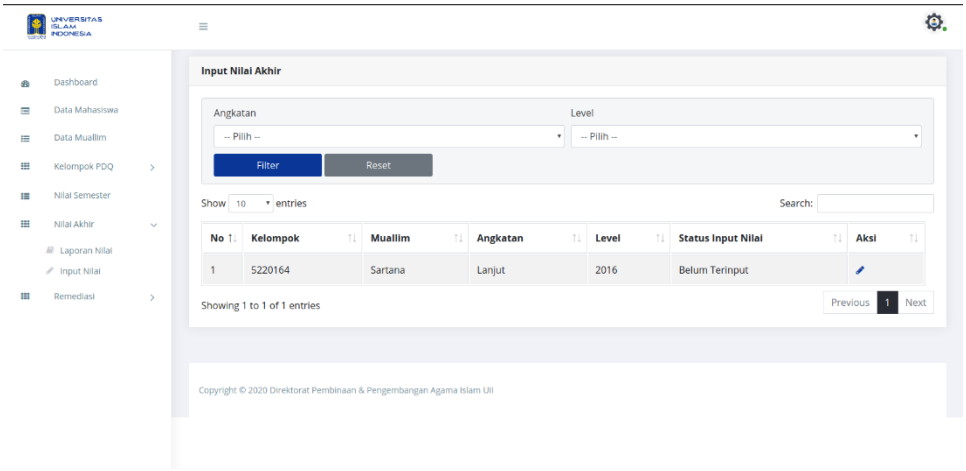
Pada menu pembuatan kelompok disajikan data rangkuman dari mahasiswa yang dibedakan menjadi dua kelompok berbeda. Data mahasiswa laki laki dan perempuan dipisah dan pada setiap data tersebut dibagi menjadi empat kategori level. Informasi tersebut digunakan untuk mempermudah *administrator* fakultas dalam membuat kelompok. Jumlah mahasiswa yang disajikan dalam tiap kategori adalah mahasiswa yang belum mendapatkan kelompok. Halaman pembuatan kelompok ditunjukan pada Gambar 4.17.



Gambar 4.17 Halaman Pembuatan Kelompok *Administrator* Fakultas

J. Halaman Input Nilai Akhir

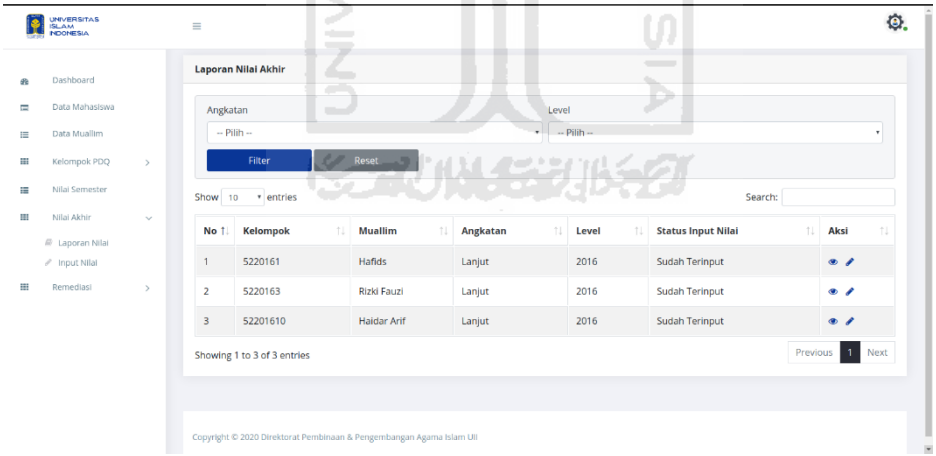
Pada halaman *input* nilai akhir *administrator* fakultas akan disajikan data per kelompok mana saja yang status nilai akhirnya belum dimasukan ke dalam sistem. Nilai akhir yang dijadikan sebagai syarat kelulusan program PDQ ini akan dimasukan per kelompok oleh *administrator* fakultas. Halaman *input* nilai akhir ditunjukan pada Gambar 4.18.



Gambar 4.18 Halaman *Input* Nilai Akhir *Administrator* Fakultas

K. Halaman Laporan Nilai Akhir

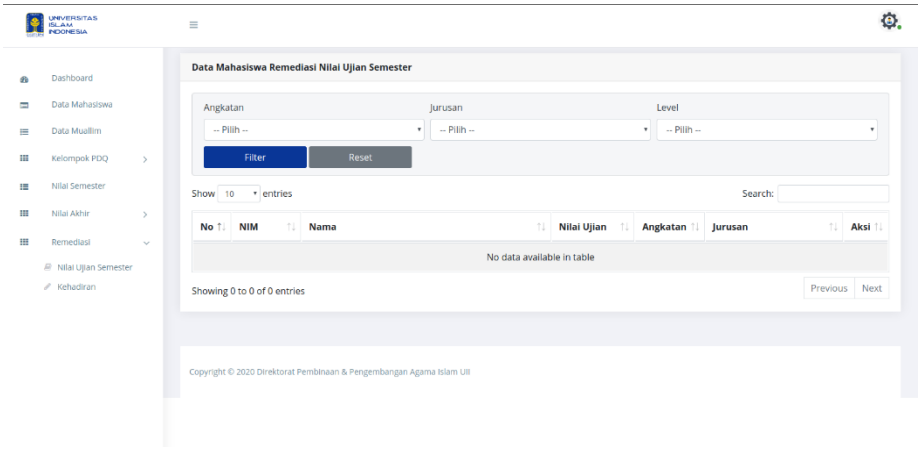
Pada halaman laporan nilai akhir *administrator* fakultas akan disajikan data per kelompok mana saja yang status nilai akhirnya sudah dimasukan ke dalam sistem. Nilai yang tersimpan dapat lihat hasilnya dalam bentuk pdf. Halaman laporan nilai akhir ditunjukan pada Gambar 4.19.



Gambar 4.19 Halaman Data Nilai Akhir *Administrator* Fakultas

L. Halaman Remediasi

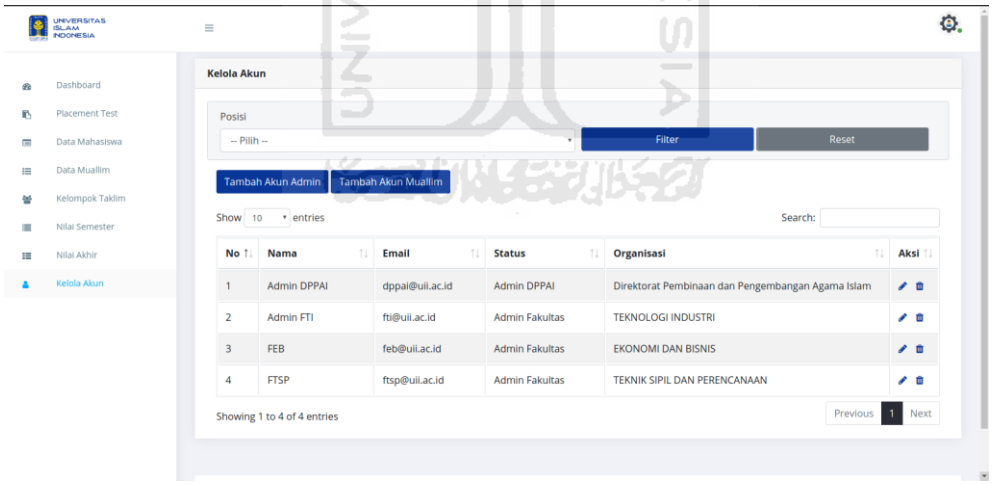
Pada halaman remediasi *administrator* fakultas disajikan data mahasiswa yang harus mengikuti proses remediasi dikarenakan jumlah kehadiran per semester yang tidak mencukupi maupun hasil ujian semester yang tidak memenuhi syarat ketentuan lulus. Halaman data mahasiswa remediasi ditunjukan pada Gambar 4.20.



Gambar 4.20 Halaman Remediasi *Administrator* Fakultas

M. Halaman Pengelolaan Akun

Pada halaman pengelolaan akun, administrator DPPAI memiliki otoritas untuk menambah, mengubah, dan menghapus akun dalam sistem. Menu ini dibuat untuk mempermudah pengelolan akun tanpa harus masuk ke dabasenya secara langsung. Halaman kelola akan *administrator* DPPAI ditunjukan pada Gambar 4.21.



Gambar 4.21 Halaman Pengelolaan Akun *Administrator* DPPAI

4.2 Pengujian Sistem

Tahap pengujian sistem dilakukan untuk memastikan bahwa sistem yang telah dikembangkan sesuai dengan rencana dan hasil yang diharapkan. Pengujian perangkat lunak menggunakan *User Acceptance Testing* dengan metode wawancara dan pengukuran efisiensi menggunakan tool GTMetrix. Pengujian *User Acceptance Testing* berfokus untuk menguji apakah sistem sudah dapat memberikan kemudahan bagi pengguna dalam melakukan pengelolaan dan *monitoring* program PDQ atau belum. Sementara pengukuran efisiensi berfokus untuk mengukur seberapa efisien sistem yang telah dikembangkan berdasarkan parameter kecepatan respons sistem dan juga penggunaan data.

4.2.1 *User Acceptance Testing*

Tahapan *User Acceptance Testing* dilakukan dengan wawancara dengan pihak DPPAI sebagai pengguna akhir dari sistem. Selain bertujuan untuk memastikan apakah sistem sudah menjawab kebutuhan dari pengguna, tahap pengujian ini juga untuk memastikan bahwa pengguna memperoleh manfaat kemudahan dengan adanya sistem yang dikembangkan ini. Berikut merupakan hasil wawancara dengan pihak DPPAI yang diwakili oleh Bapak Junaidi Safitri sebagai Kepala Divisi Pendidikan dan Dakwah ditunjukan pada Tabel 4.1.

Tabel 4.1 Hasil Pengujian UAT berdasarkan wawancara

No	Pertanyaan	Jawaban			Kebutuhan
		Ya	*Tidak*	*Alasan*	
1	Secara aspek antarmuka apakah sistem ini mudah untuk dipahami dan mudah digunakan?	Ya		Ya, secara tampilan sistem yang sudah dibuat memiliki *design* yang simple, tidak rumit. Sehingga mudah untuk dipahami dan digunakan	Pengelolaan dan *Monitoring*
2	Apakah ada fitur yang paling berguna dari sistem ini? Di bagian atau fitur mana dari sistem ini yang paling memberikan kemudahan bagi proses pengelolaan dan	Ya		Fitur yang paling memberikan kemudahan adalah proses memperbarui level dari mahasiswa untuk mengikuti progam PDQ. Karena cukup dengan mengunggah file excel, level mahasiswa sudah langsung bisa dipebarui.	Pengelolaan

	monitoring program PDQ daripada sistem sebelumnya?				
3	Secara keseluruhan apakah sistem ini sudah mampu menjawab kebutuhan DPPAI selaku pengguna akhir dari sistem ini?	Ya		Secara keseluruhan sistem sudah sesuai dengan harapan kami, karena sudah hampir sepenuhnya proses bisnis untuk program PDQ ini sudah diakomodasi oleh sistem, hanya ada satu dua saja yang belum ada pada sistem.	Pengelolaan dan *Monitoring*
4	Selain manfaat kemudahan, apakah ada manfaat dari aspek lain dengan adanyanya sistem ini ?	Ya		Ya, selain manfaat kemudahan dengan fitur yang ada pada sistem ini kami rasa manfaat dari segi efisiensi waktu juga bisa dirasakan jika sistem ini dapat diterapkan	Pengelolaan
5	Apakah dengan adanya sistem ini nantinya bisa meningkatkan proses bisnis yang ada di DPPAI menjadi lebih baik?	Ya		Ya, tentunya harapan kami dengan sistem yang sudah dibuat ini ketika diterapkan nanti bisa memberikan dampak positif bagi proses pengelolaan dan *monitoring* program pembinaan keagamaan di UII khususnya program PDQ.	Pengeloaan dan *Monitoring*
6	Apakah ada hal apa sekiranya harus diperbaiki lagi dari sistem ini ?	Ya		Ada, terutama mungkin dari aspek performa bisa ditingkatkan lagi supaya sistem bisa menangani data yang sekian banyak dari seluruh fakultas UII. Karena pengujian ini tadi kan baru data dari satu fakultas FTI saja.	Pengelolaan dan *Monitoring*

7	Adakah saran terkait fitur yang harus ditambahkan untuk tahap pengembangan lebih lanjut kedepan?	Ya		Ada, untuk memaksimalkan fungsi *monitoring* kami pikir perlu ada informasi mengenai perkembangan mahasiswa dalam mengikuti program PDQ. Artinya bukan hanya kehadiran dan nilai saja yang bisa kami lihat, tapi juga misalkan perkembangan hafalan mahasiswa	*Monitoring*

4.2.2 Pengukuran Efisiensi

Berpedoman pada standar ISO 9126 mengenai karakteristik perangkat lunak yang berkualitas dari aspek efisiensi dapat dilihat dari dua sisi yaitu *time behaviour* dan *resource behaviour*. Maka penulis melakukan pengujian menggunakan tool GTMetrix untuk mengukur waktu yang dibutuhkan dalam mengakses sebuah halaman dan juga seberapa besar *resource* data yang diperlukan. Hasil pengujian menggunakan *tool* GTMetrix ditunjukan pada Tabel 4.2.

Tabel 4.2 Hasil Pengukuran Menggunakan GTmetrix.

No	Halaman Web	*Page Load (detik)*	*Page Speed Grade*	Yslow *Grade*	Resource Data Usage
1	Halaman Login	3.1 s	A (97 %)	A (93%)	428 KB
2	Dashboard DPPAI	1.4 s	A (94 %)	A (93 %)	491 KB
3	Data Mahasiswa	2.9 s	A (94 %)	A (92%)	412 KB
4	Data Muallim	1.7 s	A (94 %)	A (92 %)	410 KB
5	Data Kelompok	2.3 s	A (94 %)	A (92 %)	410 KB
6	Nilai Semester	3.0 s	A (94%)	A (92 %)	410 KB
7	Nilai Akhir	1.8 s	A (94 %)	A (92 %)	410 KB
8	Kelola Akun	2.6 s	A (94 %)	A (92 %)	410 KB
9	Dashboard Fakultas	2.1 s	A (96%)	A(93%)	318 KB
10	Data Mahasiswa Fakultas	1.9s	A (95%)	A(92%)	340 KB
11	Data Muallim Fakultas	1.9 s	A (95%)	A(92%)	340 KB
12	Data Kelompok Fakultas	2.1 s	A (95%)	A (92%)	340 KB

13	Buat Kelompok Fakultas	3.1 s	A (96%)	A (93%)	340 KB
14	Nilai Semester Fakultas	2.0 s	A (96%)	A (92%)	336 KB
15	Nilai Akhir Fakultas	1.7 s	A (95%)	A (92%)	337 KB
16	*Input* nilai akhir Fakultas	1.8 s	A (95%)	A (92%)	335 KB
17	Remediasi nilai ujian	2.0 s	A (95%)	A (92%)	335 KB
18	Remediasi Kehadiran	2.6	A (95%)	A (92%)	335 KB

4.2.3 Evaluasi dan Kesimpulan Hasil Pengujian

Berdasarkan *User Acceptance Testing* dengan metode wawancara yang dilakukan dengan pihak DPPAI, sistem yang dikembangkan mampu memberikan kemudahan bagi pengguna dalam proses pengelolaan serta *monitoring* program PDQ. Sementara berdasarkan standar dari 9126 ISO mengenai karakterisitik sistem yang berkualitas dari aspek efisiensi, hasil pengujian menggunakan *tool* GTmetrix menunjukan hasil bahwa sistem masuk ke dalam *Grade* A untuk setiap halaman. Hasil tersebut menunujukan bahwa sistem sudah termasuk baik dalam hal efisiensi.

BAB V
KESIMPULAN DAN SARAN

5.1 Kesimpulan

Berdasarkan penelitian yang telah dilakukan, maka dapat diambil kesimpulan sebagai berikut:

a. Sistem ini dikembangkan dengan metode pengembangan perangkat lunak *Waterfall*. Proses pengembangan dibagi menjadi beberapa tahapan. Dimulai dari tahap komunikasi, kemudian dilanjut dengan tahap perencanaan, perancangan, konstruksi atau implementasi, dan berakhir pada tahap *deployment.*
b. Sistem yang telah berhasil dikembangkan diuji dengan dua metode. Metode pertama *User Acceptance Testing* berfokus kepada kemudahan pengguna dalam menggunakan sistem. Metode kedua pengujian performa efisiensi menggunakan *tool* GTMetrix berfokus untuk mengukur kecepatan respon sistem dan penggunaan *resource data.*
c. Hasil pengujian menggunakan metode *User Acceptance Testing* menunjukan bahwa sistem yang dikembangkan sudah memberikan kemudahan bagi pengguna. Hal ini ditunjukan dari hasil wawancara dengan narasumber. Jawaban dari seluruh pertanyaan yang berkaitan dengan kemudahan menunjukan bahwa pengguna mendapatkan kemudahan dalam penggunaan sistem.
d. Hasil pengujian menggunakan *tool* GTMetrix menunjukan bahwa sistem yang dikembangkan memiliki performa efisiensi yang baik. Hal ini ditunjukan dari hasil kecepatan respon sistem dan penggunaan *resource data* yang mendapatkan kategori *Grade A* untuk setiap halaman maupun fitur yang diuji coba.

5.2 Saran

Sistem Informasi Pengelolaan Kegiatan dan *Monitoring* Program Pengembangan Diri Qur'ani (PDQ) merupakan sebuah sistem yang baru mengakomodir salah satu program pembinaan dari DPPAI. Penulis menyarankan agar dapat dilakukan penelitian tentang pengembangan sistem informasi untuk pengelolaan program pembinaan agama dari DPPAI yang lain supaya seluruh rangkaian proses pembinaan keagamaan di Universitas Islam Indonesia bisa terintegrasi dengan baik. Diantara program pembinaan tersebut adalah ONDI dan pesantrenisasi.

DAFTAR PUSTAKA

Bassil, Y. (2012). *A Simulation Model for the Waterfall Software Development Life Cycle*. *2*(5). http://arxiv.org/abs/1205.6904

DPPAI. (2019). Pedoman Pengembangan Diri Qur'ani (PDQ)

Filipova, O. (2018). *Sof tware Development From A to Z*. apress. https://doi.org/10.1007/978-1-4842-3945-2

Firmansyah, Y., & Udi, U. (2017). Penerapan Metode SDLC Waterfall Dalam Pembuatan Sistem Informasi Akademik Berbasis Web Studi Kasus Pondok Pesantren Al-Habib Sholeh Kabupaten Kubu Raya, Kalimantan Barat. *Jurnal Teknologi Dan Manajemen Informatika*, *4*(1). https://doi.org/10.26905/jtmi.v4i1.1605

James A. O'Brien & George M. Marakas. (2010). Introduction ti information system. , *3*(2), 54–67. New York: McGraw-Hill/Irwin

Laudon, K. C., & Laudon, J. P. (2014). *Management Information Systems Managing The digital Firm Thirteen Edition Global Edition (SIB)*.

Mohd, C. K. N. C. K., & Shahbodin, F. (2015). Personalized Learning Environment: Alpha Testing, Beta Testing & User Acceptance Test. *Procedia - Social and Behavioral Sciences*, *195*, 837–843. https://doi.org/10.1016/j.sbspro.2015.06.319

Nielsen, J. 2010. *Website Response Times*. Diakses pada 17 Mei 2020, dari https://www.nngroup.com/articles/website-response-times/

Pressman, R.S. 2015. Rekayasa Perangkat Lunak: Pendekatan Praktisi Buku I. Yogyakarta: Andi.

UII. (2017). Pembinaan Keagamaan di UII Diselaraskan dengan Kurikulum Ulil Albab. Diakses pada 5 Mei 2020, dari https://www.uii.ac.id/pembinaan-keagamaan-di-uii-diselaraskan-dengan-kurikulum-ulil-albab/

Valacich, J. S., George, J. F., & Hoffer, J. A. (2012). *Essentials of systems analysis and design*.

LAMPIRAN

Lampiran A . Proses Pengujian UAT melalui wawancara



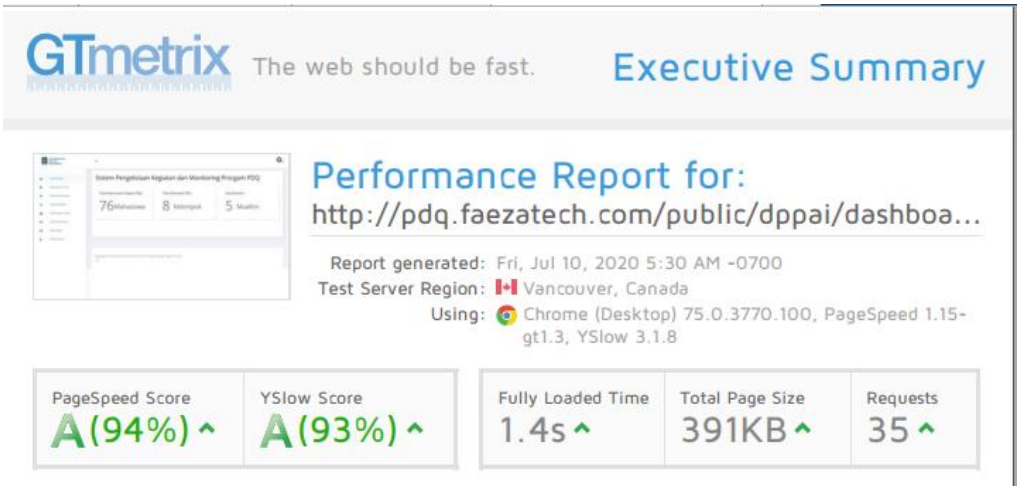


Lampiran B . Hasil Pengujian Efisiensi Menggunakan GTMetrix

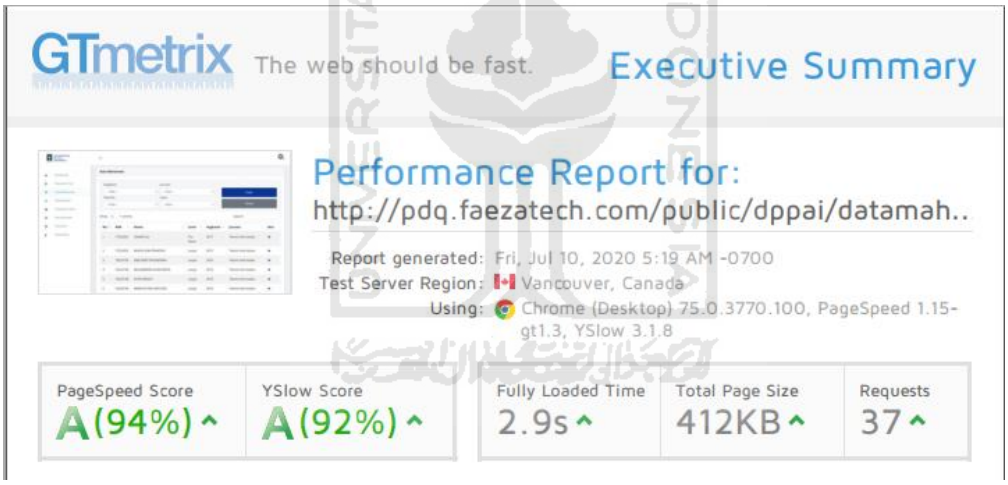
Pengujian terhadap halaman login



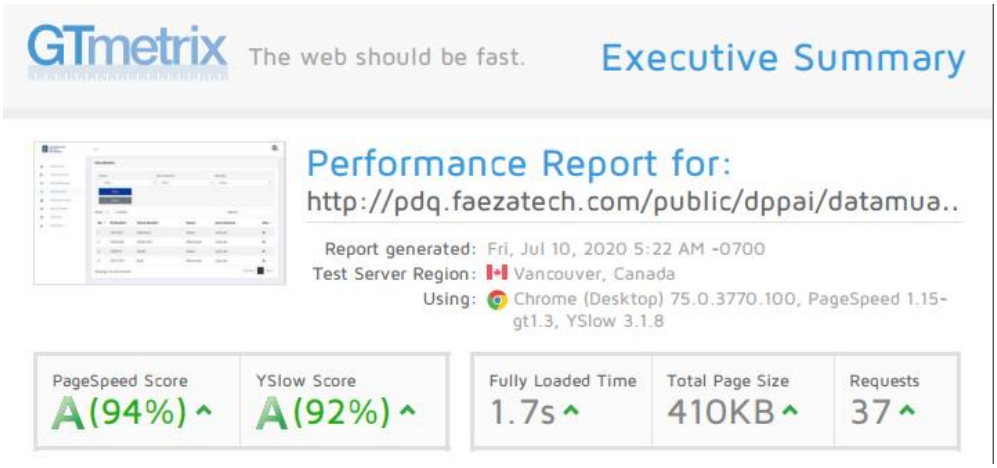
Pengujian terhadap halaman dashboard DPPAI



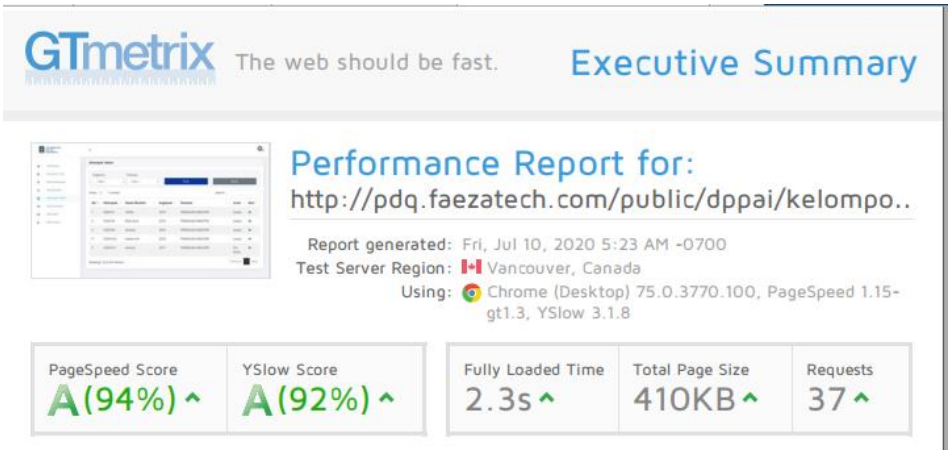
Pengujian terhadap halaman data mahasiswa



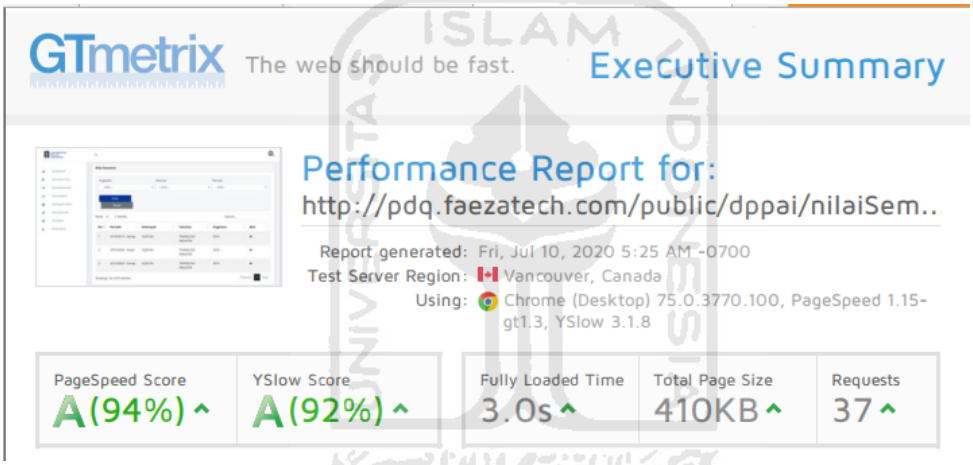
Pengujian terhadap halaman data muallim



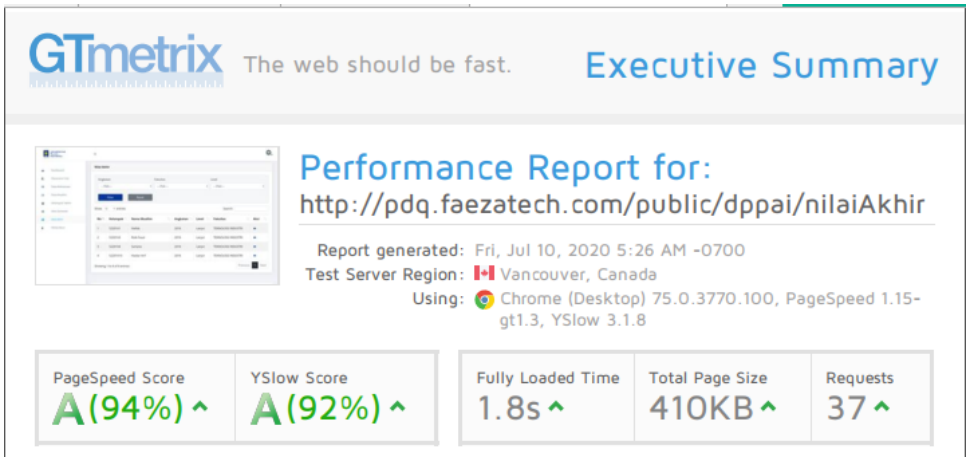
Pengujian terhadap halaman kelompok



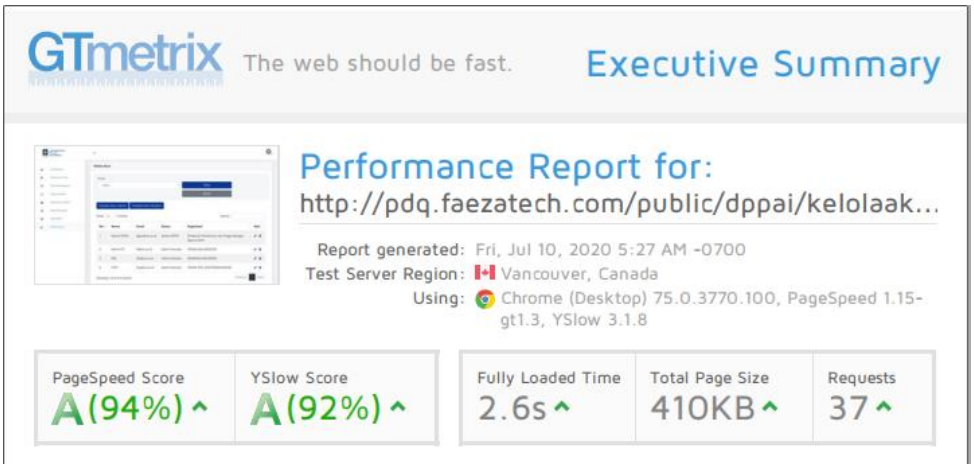
Pengujian terhadap halaman nilai semester



Pengujian terhadap halaman nilai akhir



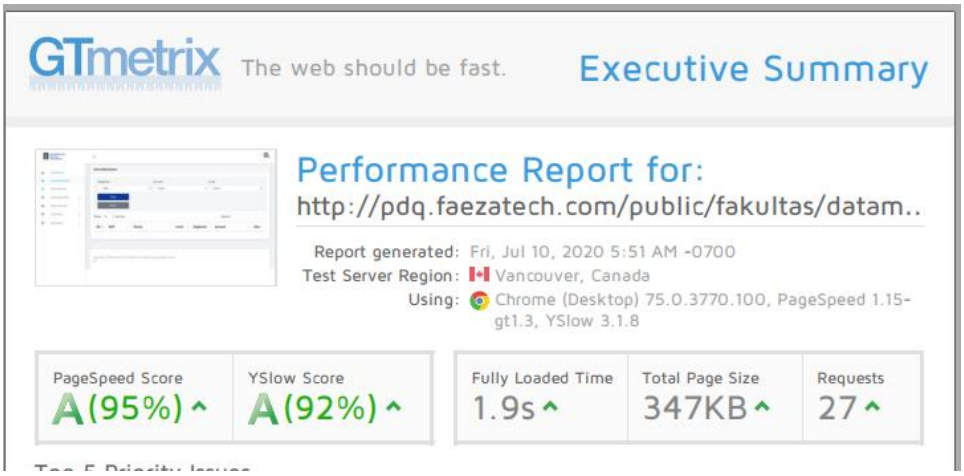
Pengujian terhadap halaman kelola akun



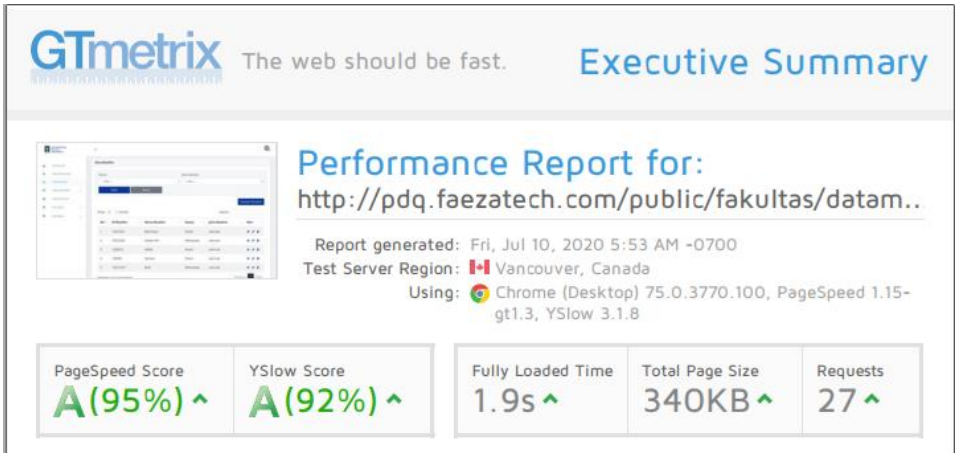
Pengujian terhadap halaman *dashboard* fakultas



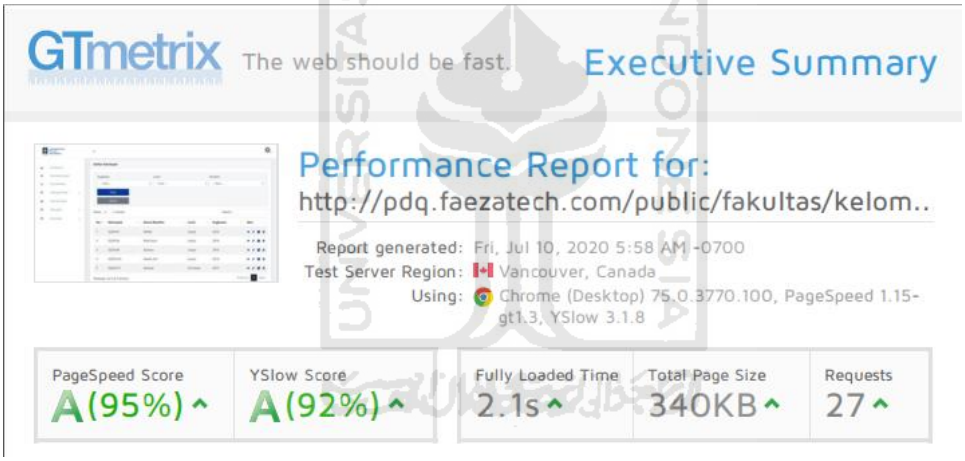
Pengujian terhadap halaman data mahasiswa fakultas



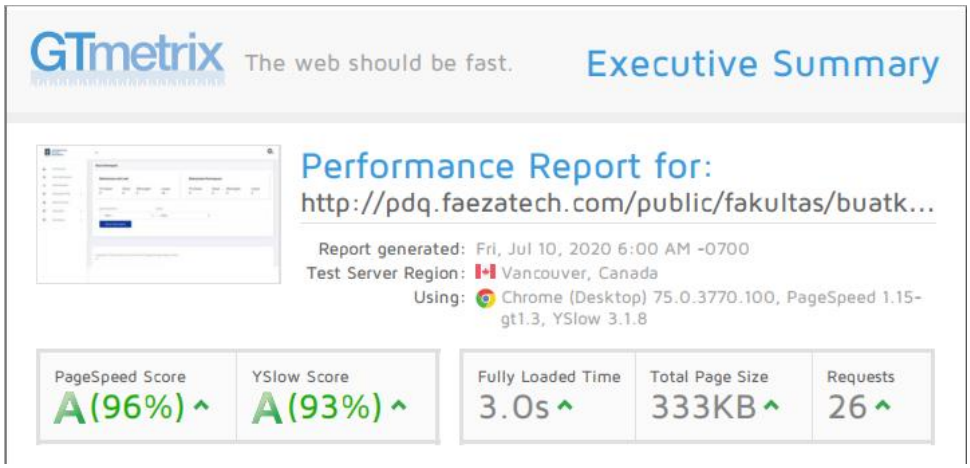
Pengujian terhadap halaman data muallim fakultas



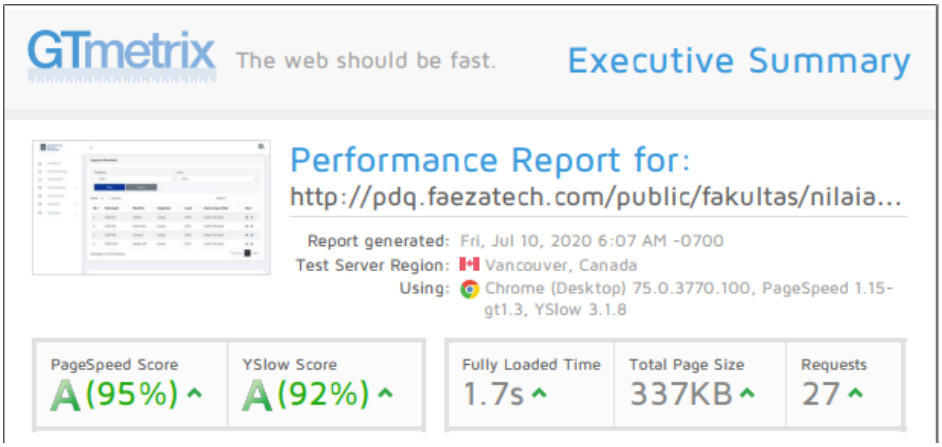
Pengujian terhadap halaman kelompok fakultas



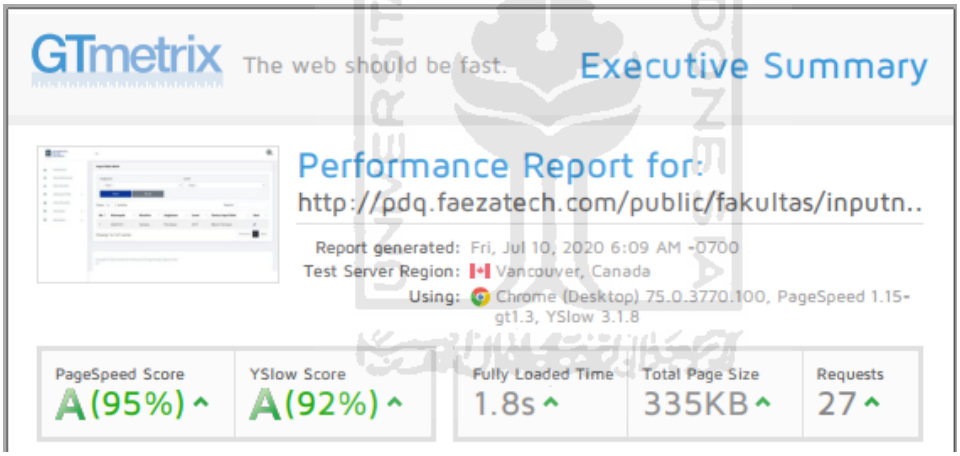
Pengujian terhadap halaman buat kelompok fakultas



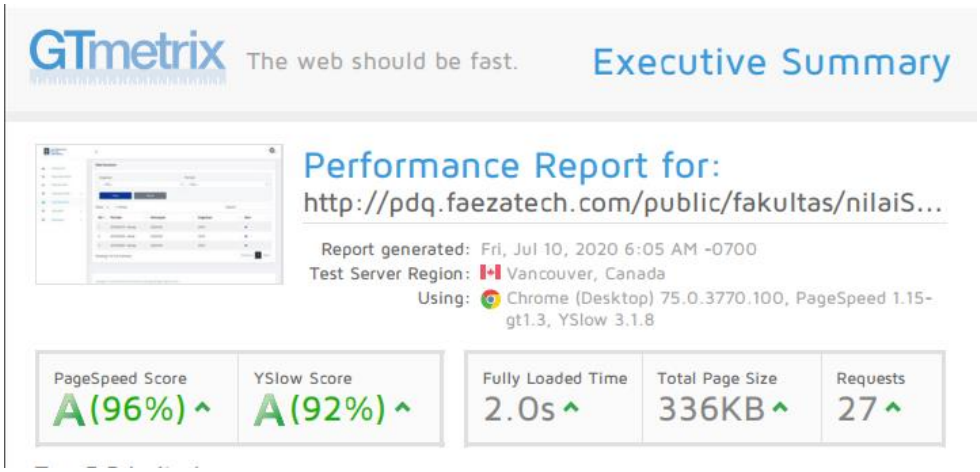
Pengujian terhadap halaman nilai akhir fakultas



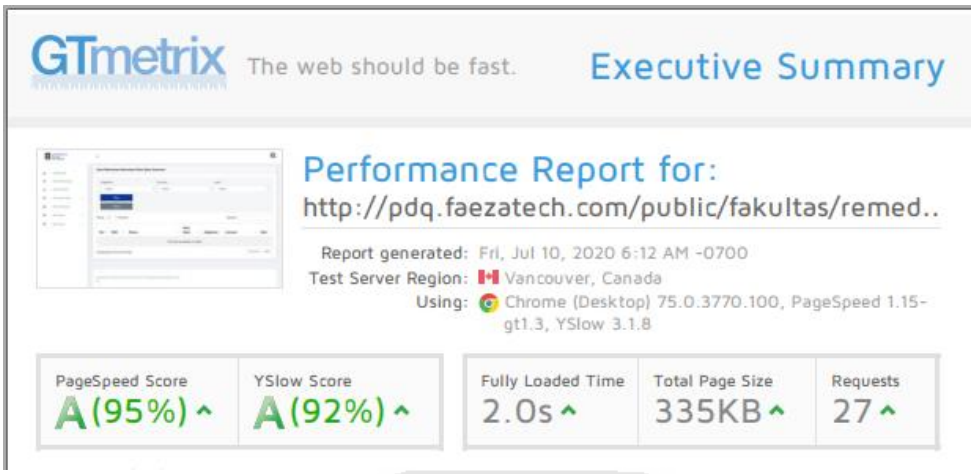
Pengujian terhadap halaman *input* nilai akhir fakultas



Pengujian terhadap halaman nilai semester



Pengujian terhadap halaman remediasi berdasarkan nilai ujian



Pengujian terhadap halaman remediasi berdasarkan kehadiran

