

**APLIKASI WEB UNTUK SELEKSI *ON-THE-SPOT*
PENERIMAAN MAHASISWA BARU
(STUDI KASUS: *LASALLE MARKETING CENTER*)**

LAPORAN KERJA PRAKTIK

**Disusun oleh:
DUSTIN IGNATIUS KUMALA
21013003**



**PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS TEKNIK
UNIVERSITAS KATOLIK DE LA SALLE
MANADO
2026**

**APLIKASI WEB UNTUK SELEKSI *ON-THE-SPOT*
PENERIMAAN MAHASISWA BARU
(STUDI KASUS: *LASALLE MARKETING CENTER*)**

LAPORAN KERJA PRAKTIK

Ditulis untuk Memenuhi Persyaratan Mata Kuliah Kerja Praktik
(INF2417401)

Disusun oleh:

**DUSTIN IGNATIUS KUMALA
21013003**



**PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS TEKNIK
UNIVERSITAS KATOLIK DE LA SALLE
MANADO
2026**



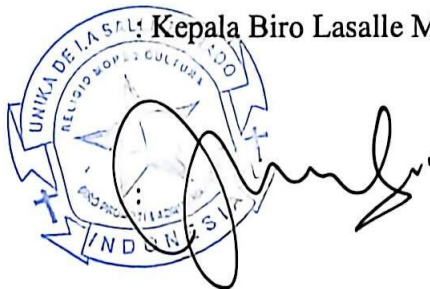
**PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS TEKNIK
UNIVERSITAS KATOLIK DE LA SALLE MANADO**

FORM KP - 003

FORMULIR DATA UMUM PERUSAHAAN

NAMA MAHASISWA : Dustin Ignatius Kumala
NIM : 21013003
NAMA PERUSAHAAN : Universitas Katolik De La Salle Manado
ALAMAT PERUSAHAAN : Kombos, Jl. Kairagi I Jl. Kombos-Manado,
Kairagi Satu, Kec. Mapanget, Kota Manado,
Sulawesi Utara
DIDIRIKAN TAHUN : 2000
IZIN USAHA : Keputusan Menteri Pendidikan Nasional RI Nomor
113/D/O/2000
BIDANG BISNIS : Pendidikan
JUMLAH KARYAWAN : 185 Karyawan
PEMILIK : Yayasan Perguruan Tinggi Universitas Katolik
De La Salle Manado.
DEWAN DIREKTUR : Dr. Gregorius Hertanto Dwi Wibowo, S.S., M.Th.
WAKIL PERUSAHAAN
Tanggal : 5 Juni 2026
Nama : Junaidy Budi Sanger, S.Kom., M.Kom.
Jabatan : Kepala Biro Lasalle Marketing Center

(Tanda tangan dan
cap perusahaan)





PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS TEKNIK
UNIVERSITAS KATOLIK DE LA SALLE MANADO

FORM KP - 004

FORMULIR PENILAIAN KEMAJUAN KERJA PRAKTIK

A. UMUM

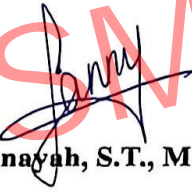
Nama Mahasiswa : Dustin Ignatius Kumala
NIM Mahasiswa : 21013003
Program Studi : Teknik Informatika
Dosen Pembimbing Akademik : Junaidy Budi Sanger, S.Kom., M.Kom.
Pembimbing 1 : Lanny Sitanayah, S.T., M.Sc., Ph.D.
Pembimbing 2 : Thomas Christian Suwanto, S.Kom., M.Mm
Terhitung Mulai : 9 Februari 2026
Target Selesai : 5 Juni 2026

B. KEGIATAN PELAKSANAAN KERJA PRAKTEK

No.	Tanggal	Jenis Kegiatan	Paraf Pembimbing
1.	14 April 2026	Proposal Kerja Praktik	
2.	14 April 2026	Proposal Kerja Praktik	
3.	7 Mei 2026	Proposal Kerja Praktik	
4.	7 Mei 2026	Proposal Kerja Praktik	
5.	11 Mei 2026	Konsultasi laporan bab 2	
6.	11 Mei 2026	Konsultasi laporan bab 2	
7.	18 Mei 2026	Revisi laporan bab 2	
8.	18 Mei 2026	Revisi laporan bab 2	

No.	Tanggal	Jenis Kegiatan	Paraf Pembimbing
9.	7 Juni 2026	Konsultasi laporan bab 3	
10.	7 Juni 2026	Konsultasi laporan bab 3	
11.	8 Juni 2026	Konsultasi laporan bab 4	
12.	8 Juni 2026	Konsultasi laporan bab 4	
13.	10 Juni 2026	Revisi laporan bab 4	
14.	10 Juni 2026	Revisi laporan bab 4	
15.	11 Juni 2026	Revisi laporan akhir	
16.	12 Juni 2026	Revisi laporan akhir	

Dosen Pembimbing KP


Lanny Sitanaayah, S.T., M.Sc., Ph.D.



**PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS TEKNIK
UNIVERSITAS KATOLIK DE LA SALLE MANADO**

FORM KP - 005

FORMULIR PENILAIAN PELAKSANAAN KERJA PRAKTIK

Mohon diisi dan dicek seperlunya.

NAMA MAHASISWA : Dustin Ignatius Kumala
NIM : 21013003
NAMA PERUSAHAAN : Universitas Katolik De La Salle Manado
ALAMAT PERUSAHAAN : Kombos, Jl. Kairagi I Jl. Kombos-Manado,
Kairagi Satu, Kec. Mapanget, Kota Manado,
Sulawesi Utara
TGL KERJA PRAKTEK : 9 Februari 2026
TOPIK YANG DIBAHAS : Aplikasi Web Untuk Seleksi *On-The-Spot*
Penerimaan Mahasiswa Baru

Nilai Sikap	=	50	60	70	80	90	100
Kerajinan	=	50	60	70	80	90	100
Prestasi	=	50	60	70	80	90	100

KOMENTAR/SARAN

-

NILAI RATA-RATA : 90
Tanggal : 5 Juni 2026
NAMA PENILAI : Junaidy Budi Sanger, S.Kom., M.Kom.
Jabatan : Kepala Biro Lasalle Marketing Center
(Tanda tangan dan cap perusahaan)

KATA PENGANTAR

Puji dan syukur dipanjatkan kepada Tuhan yang Maha Esa atas rahmat dan kemurahan-Nya sehingga penulis dapat menyelesaikan proses penulisan Kerja Praktik dengan judul “Aplikasi Web Untuk Seleksi *On-The-Spot* Penerimaan Mahasiswa Baru”. Penulis menyadari bahwa proses Kerja Praktik ini tidak akan berjalan dengan baik tanpa adanya bantuan dari orang-orang terdekat, maka dari itu di kesempatan ini penulis ingin mengucapkan terima kasih sebesar-besarnya kepada:

1. Bapak Dr. Gregorius Hertanto Dwi Wibowo, S.S., M.Th. selaku Rektor Universitas Katolik De La Salle Manado.
2. Bapak Ronald A. Rachmadi, S.T., M.T. selaku Dekan Fakultas Teknik.
3. Ibu Vivie D. Kumenap, S.T., M.Cs. selaku Ketua Program Studi Teknik Informatika.
4. Ibu Lanny Sitanayah, S.T., M.Sc., Ph.D. selaku Dosen Pembimbing I yang telah membantu dan menuntun dari awal hingga akhir Kerja Praktik.
5. Bapak Thomas Christian Suwanto, S.Kom., M.Mm. selaku Dosen Pembimbing II yang telah membantu dan memberikan arahan dalam proses penulisan laporan Kerja Praktik.
6. Bapak Junaidy Budi Sanger, S.Kom., M.Kom., selaku Dosen Pembimbing Akademik yang memberikan bimbingan selama masa perkuliahan.
7. Kepada seluruh anggota keluarga yang telah mendukung penulis selama masa perkuliahan.
8. Kepada teman-teman seperjuangan Teknik Informatika Angkatan 21.

Penulis menyadari bahwa laporan Kerja Praktik ini masih memiliki banyak kekurangan, karena itu penulis mengharapkan saran dan kritik membangun dari para pembaca.

Manado, Juni 2026

Penulis

DAFTAR ISI

HALAMAN JUDUL.....	i
LEMBAR PENGESAHAN.....	ii
LEMBAR PERNYATAAN	iii
FORMULIR DATA UMUM PERUSAHAAN (FORM KP-003)	iv
FORMULIR PENILAIAN KEMAJUAN KERJA PRAKTIK (FORM KP-004) ...	v
FORMULIR PENILAIAN PELAKSANAAN KERJA PRAKTIK (FORM KP-005).....	vii
KATA PENGANTAR.....	vii
DAFTAR ISI.....	ix
DAFTAR TABEL	xi
DAFTAR GAMBAR	xii
DAFTAR LAMPIRAN	xiii
BAB I PENDAHULUAN	1
1.1 Latar Belakang.....	1
1.2 Rumusan Masalah	2
1.3 Tujuan Kerja Praktik	2
1.4 Manfaat Kerja Praktik	2
1.5 Batasan Masalah.....	2
1.6 Sistematika Penulisan.....	3
BAB II DATA UMUM PERUSAHAAN	4
2.1 Sejarah Singkat Perusahaan.....	4
2.2 Lingkungan Pekerjaan Perusahaan.....	4
2.2.1 Fungsi Biro LMC	4
2.2.2 Tugas Pokok Biro LMC	4
2.2.3 Struktur	5
2.2.4 Logo Perusahaan	6
2.2.5 Arti Logo Perusahaan	6
2.3 Lingkup Pekerjaan yang Dilakukan	7
BAB III LANDASAN TEORI	8
3.1 Teori Pendukung	8
3.1.1 Sistem Penerimaan Mahasiswa Baru.....	8
3.1.2 Teknologi Pengembangan Sistem	8
3.2 Metodologi Pengembangan Perangkat Lunak.....	13
3.2.1 Metode Pengembangan Sistem	13
3.2.2 Kakas Pemodelan	14
BAB IV PEMBAHASAN	19
4.1 Analisis	19
4.1.1 Analisis Target Pengguna.....	19
4.1.2 Sumber Data	19
4.1.3 Pemodelan Sistem yang Sedang Berjalan	19
4.1.4 Persyaratan dan Preferensi Sistem Baru.....	20

4.1.5	Identifikasi Masalah dan Kesempatan.....	21
4.2	Perancangan.....	21
4.2.1	<i>Data Flow Diagram</i>	22
4.2.2	Rancangan Antarmuka Aplikasi.....	25
4.2.3	Rancangan Struktur Basis Data.....	36
4.3	Implementasi	37
4.3.1	Lingkungan Implementasi.....	37
4.3.2	Aturan Implementasi	38
4.3.3	Implementasi Antarmuka Aplikasi.....	38
4.3.4	Implementasi Basis Data	47
4.3.5	Implementasi Modul Program.....	48
4.4	Pengujian	48
4.4.1	Tujuan Pengujian.....	48
4.4.2	Kriteria Pengujian.....	48
4.4.3	<i>Black Box Testing</i>	49
4.4.4	<i>User Acceptance Test</i>	58
4.4.5	Analisis Hasil Pengujian.....	59
BAB V KESIMPULAN DAN SARAN.....		60
5.1	Kesimpulan	60
5.2	Saran	60
DAFTAR PUSTAKA.....		62
LAMPIRAN		

UKDLSM

DAFTAR TABEL

Tabel 3.1	Simbol-Simbol Data Flow Diagram [11]	15
Tabel 3.2	Simbol-Simbol Flowchart [12].....	16
Tabel 3.3	Simbol-Simbol <i>Document Collection Diagram</i> [13].....	17
Tabel 4.1	Tabel Masalah dan Kesempatan.....	21
Tabel 4.2	Spesifikasi Perangkat Pengujian.....	49
Tabel 4.3	Hasil Pengujian <i>Black Box Testing</i>	50

UKDLSM

DAFTAR GAMBAR

Gambar 2.1	Struktur Biro LMC Unika De La Salle.....	5
Gambar 2.2	Logo Unika De La Salle Manado	6
Gambar 4.1	Pemodelan Sistem yang Sedang Berjalan	20
Gambar 4.2	<i>Data Flow Diagram</i> Level 0	22
Gambar 4.3	<i>Data Flow Diagram</i> Level 1	23
Gambar 4.4	<i>Data Flow Diagram</i> Level 2 1.0	24
Gambar 4.5	<i>Data Flow Diagram</i> Level 2 2.0	24
Gambar 4.6	<i>Data Flow Diagram</i> Level 2 3.0	25
Gambar 4.7	Rancangan Halaman <i>Landing</i>	26
Gambar 4.8	Rancangan Halaman Seleksi	27
Gambar 4.9	Rancangan Halaman <i>Dashboard</i>	28
Gambar 4.10	Rancangan Halaman Data Pendaftar	29
Gambar 4.11	Rancangan Halaman Manajemen Halaman.....	30
Gambar 4.12	Rancangan Halaman Manajemen Pertanyaan	31
Gambar 4.13	Rancangan Halaman Manajemen Program Studi.....	32
Gambar 4.14	Rancangan Halaman Pengaturan	33
Gambar 4.15	Rancangan Tampilan <i>Login Admin</i>	33
Gambar 4.16	Rancangan Halaman Detail Pendaftar.....	34
Gambar 4.17	Rancangan Tampilan Edit Halaman	34
Gambar 4.18	Rancangan Tampilan Edit Soal.....	35
Gambar 4.19	Rancangan Tampilan Edit Program Studi.....	35
Gambar 4.20	<i>Document Collection Diagram</i>	36
Gambar 4.21	Implementasi Halaman <i>Landing</i>	38
Gambar 4.22	Impelementasi Halaman Seleksi.....	39
Gambar 4.23	Implementasi Halaman <i>Dashboard</i>	39
Gambar 4.24	Implementasi Halaman Data Pendaftar	40
Gambar 4.25	Implementasi Halaman Manajemen Halaman.....	40
Gambar 4.26	Implementasi Halaman Manajemen Pertanyaan	41
Gambar 4.27	Implementasi Halaman Manajemen Prodi	42
Gambar 4.28	Implementasi Halaman Pengaturan	42
Gambar 4.29	Implementasi Tampilan <i>Login Admin</i>	43
Gambar 4.30	Impelemntasi Tampilan Detail Pendaftar	43
Gambar 4.31	Implementasi Tampilan Edit Halaman	44
Gambar 4.32	Implementasi Halaman Edit Soal	45
Gambar 4.33	Implementasi Tampilan Edit Program Studi.....	46
Gambar 4.34	Tampilan Hasil Ekspor CSV.....	47
Gambar 4.35	Implementasi Basis Data.....	47

DAFTAR LAMPIRAN

LAMPIRAN A KODE PROGRAM.....	A-1
LAMPIRAN B <i>USER ACCEPTANCE TEST</i>	B-1
LAMPIRAN C TRANSKRIP WAWANCARA.....	C-1

UKDLSM

BAB I

PENDAHULUAN

1.1 Latar Belakang

Biro *La Salle Marketing Center* (LMC) merupakan salah satu unit kerja di Universitas Katolik De La Salle Manado yang berada di bawah naungan Wakil Rektor bidang Penelitian dan Kerja Sama. LMC bertanggung jawab atas pengelolaan kegiatan Penerimaan Mahasiswa Baru (PMB) dan pelayanan promosi universitas. Tugas utama LMC meliputi penyelenggaraan promosi, melayani pendaftaran calon mahasiswa, melakukan proses seleksi, pengumuman hasil seleksi, pengendalian informasi promosi, sampai mencapai target penerimaan mahasiswa baru [1].

Sebelumnya, proses seleksi memanfaatkan *Google Forms* sebagai media pengumpulan data peserta. Meskipun data dapat diperoleh dengan mudah, hasil yang diperoleh ini masih harus dipindahkan secara manual agar dapat dikelola dan dianalisis lebih lanjut. Proses pemindahan ini justru menambah waktu kerja dan menimbulkan risiko terjadinya kehilangan data.

Proses pelaksanaan seleksi *On-The-Spot* PMB menuntut proses pengelolaan data yang tak hanya cepat, tapi juga akurat dan dapat terintegrasi secara keseluruhan karena proses seleksi dilakukan secara langsung di lokasi kegiatan. Dalam praktik seleksi *On-The-Spot* masih terdapat beberapa kendala, seperti pengelolaan data peserta yang belum terintegrasi secara optimal dan proses pencatatan yang masih menggunakan metode manual, sampai penggunaan aplikasi umum yang tidak dapat memenuhi kebutuhan khusus seleksi *On-The-Spot*. Hal ini berpotensi menyebabkan kendala dalam proses pengolahan data serta keterlambatan dalam penyampaian informasi kepada calon mahasiswa.

Berdasarkan permasalahan yang telah dipaparkan, solusi untuk permasalahan ini, yaitu dengan mengembangkan sebuah aplikasi web sederhana yang mampu menggantikan penggunaan aplikasi umum yang dapat memenuhi kebutuhan khusus seleksi *On-The-Spot*. Aplikasi yang dikembangkan akan memiliki fungsionalitas yang serupa dengan sistem sebelumnya dan dilengkapi dengan kemampuan

tambahan seperti *filtering data* dan *export data* dalam format CSV. Fitur ini dapat membantu mempermudah proses pengolahan dan pengelolaan data. Aplikasi ini diharapkan dapat membantu mempermudah proses seleksi *On-The-Spot* dengan metode yang lebih terstruktur dan mudah dimengerti.

1.2 Rumusan Masalah

Berdasarkan latar belakang yang telah dipaparkan, rumusan masalah yang diangkat untuk Kerja Praktik ini adalah, bagaimana merancang aplikasi web untuk membantu proses seleksi *On-The-Spot* PMB di Unika De La Salle Manado?

1.3 Tujuan Kerja Praktik

Tujuan dari dilaksanakannya Kerja Praktik ini, yaitu untuk merancang dan membangun aplikasi web yang dapat membantu proses seleksi *On-The-Spot* PMB di Unika De La Salle Manado.

1.4 Manfaat Kerja Praktik

Berikut ini merupakan manfaat dari dilaksanakannya Kerja Praktik ini untuk masing-masing pihak:

1. Berikut ini adalah manfaat yang diterima oleh Biro LMC:
 - Mendapatkan aplikasi web yang dapat mempermudah proses seleksi *On-The-Spot* PMB.
2. Berikut ini adalah manfaat yang diterima oleh penulis:
 - a. Mendapatkan pengalaman untuk mengembangkan sebuah aplikasi berbasis web untuk pengelolaan data PMB di Unika De La Salle Manado.
 - b. Mendapatkan kesempatan untuk berkontribusi secara aktif dalam lingkungan pengembangan kampus.

1.5 Batasan Masalah

Demi menjaga besaran ruang lingkup pengembangan, berikut ini adalah beberapa batasan masalah yang telah ditentukan:

1. Aplikasi yang dibangun memerlukan koneksi Internet yang stabil agar dapat berjalan dengan optimal.

2. Aplikasi yang dibangun memerlukan *browser* dengan versi terbaru agar aplikasi dapat berjalan dengan optimal.
3. Aplikasi hanya dapat melakukan ekspor laporan dengan format CSV (*Comma Separated Value*).

1.6 Sistematika Penulisan

Berikut adalah sistematika penulisan yang dipakai di dalam laporan ini:

BAB I: PENDAHULUAN

Bab ini berisi tentang permasalahan yang ditemukan selama melakukan Kerja Praktik. Bagian ini berisi latar belakang, rumusan masalah, tujuan Kerja Praktik, manfaat Kerja Praktik, batasan masalah, dan sistematika penulisan.

BAB II: DATA UMUM PERUSAHAAN

Bab ini membahas tentang data umum perusahaan yang terdiri dari sejarah perusahaan, lingkup pekerjaan perusahaan, dan lingkup pekerjaan yang dilakukan.

BAB III: LANDASAN TEORI

Bab ini membahas terkait topik Kerja Praktik, teknologi yang digunakan, metodologi pengembangan perangkat lunak, kaskas pemodelan, dan prosedur pengumpulan dan pengolahan data.

BAB IV: PEMBAHASAN

Pada bab ini akan dibahas permasalahan yang dihadapi dan solusinya dengan mengikuti langkah-langkah metodologi pengembangan perangkat lunak.

BAB V: KESIMPULAN DAN SARAN

Bab ini berisi tentang kesimpulan dan saran untuk pengembangan selanjutnya.

BAB II

DATA UMUM PERUSAHAAN

2.1 Sejarah Singkat Perusahaan

Biro La Salle Marketing Center (LMC) merupakan salah satu unit kerja di Universitas Katolik De La Salle Manado yang bertanggung jawab dalam pengelolaan kegiatan Penerimaan Mahasiswa Baru (PMB) dan promosi universitas. Biro LMC berada di bawah naungan WR II, yaitu Wakil Rektor Bidang Penelitian dan Kerja Sama, dan berperan sebagai pusat koordinasi untuk berbagai kegiatan pemasaran, promosi, hingga administrasi PMB [1].

2.2 Lingkungan Pekerjaan Perusahaan

Bagian ini akan membahas tentang fungsi, tugas pokok, dan struktur dari biro LMC [1].

2.2.1 Fungsi Biro LMC

1. Pelaksanaan kebijakan Yayasan dan Universitas dalam kegiatan Penerimaan Mahasiswa Baru.
2. Koordinator bagi Tim Promosi Fakultas.
3. Pengendali kegiatan promosi atau sosialisasi seluruh Universitas.
4. Pengendali informasi mengenai Universitas dan Penerimaan Mahasiswa Baru.

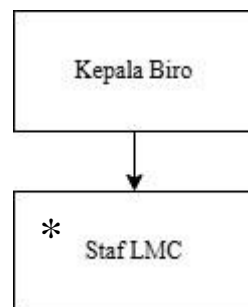
2.2.2 Tugas Pokok Biro LMC

1. Merancang dan mengusulkan untuk kemudian ditetapkan oleh Rektor atau Yayasan tentang Sistem Penerimaan Mahasiswa Baru.
2. Melakukan kajian dan desain strategi promosi.
3. Merancang jadwal kegiatan promosi atau sosialisasi Universitas Katolik De La Salle Manado.
4. Melaksanakan tugas melakukan kegiatan promosi atau sosialisasi bagi calon mahasiswa di:
 - a. SMA/SMK sederajat.
 - b. Paroki-paroki dan gereja-gereja Katolik se-Keuskupan Manado.
 - c. Lokasi-lokasi strategis (mall, pusat perbelanjaan, dsb).

- d. Media-media massa.
- e. Media sosial.
5. Menyiapkan materi-materi promosi.
6. Melaksanakan pelayanan pendaftaran calon mahasiswa baru dan pindahan.
7. Melaksanakan serangkaian proses tes saringan masuk bagi calon mahasiswa baru.
8. Mengumumkan hasil penetapan calon mahasiswa baru yang lulus seleksi.
9. Melaksanakan fungsi administrasi pendaftaran kembali calon mahasiswa yang sudah dinyatakan lulus dan membayar biaya studi.
10. Menetapkan target capaian jumlah calon mahasiswa baru yang disesuaikan dengan penetapan kuota mahasiswa baru tiap-tiap Program Studi.
11. Setiap awal tahun ajaran, merancang Program Kerja dan Rencana Anggaran Biaya.
12. Setiap akhir tahun membuat laporan pertanggungjawaban Program Kerja dan Keuangan.

2.2.3 Struktur

Berikut ini adalah struktur dari biro LMC Universitas Katolik De La Salle Manado:



Keterangan:

“*” Menandakan tempat penulis melakukan Kerja Praktik

Gambar 2.1 Struktur Biro LMC Unika De La Salle Manado

2.2.4 Logo Perusahaan

Berikut merupakan logo Universitas Katolik De La Salle Manado tempat dilakukannya Kerja Praktik:



Gambar 2.2 Logo Unika De La Salle Manado

2.2.5 Arti Logo Perusahaan

Berikut ini adalah makna dari logo perusahaan [1]:

1. Lambang dan Logo

- a. Lambang Unika De La Salle Manado berupa perisai yang bagian atasnya dilingkari dengan pita warna kuning dengan tulisan Unika De La Salle Manado, dan di bawahnya dilingkari dengan pita kuning dengan tulisan Indonesia. Pita atas dan pita bawah dibatasi dengan tanda bentuk salib di kiri kanan.
- b. Di tengah perisai yang berwarna hijau terdapat bintang yang mempunyai lima sudut yang berwarna kuning yang diselengi warna biru.
- c. Di tengah bintang terdapat kitab yang terbuka.

2. Arti dari warna

- a. Warna kuning berarti keagungan ilmu yang dicita-citakan.
- b. Warna hijau yang melatarbelakangi lambang universitas berarti sikap pengharapan yang kokoh terhadap kesuksesan tanpa mengenal rasa putus asa.
- c. Warna biru yang menyelengi lambang bintang berarti sikap ketabahan yang harus dipupuk untuk mencapai cita-cita.

3. Arti dari simbol-simbol

- a. Bintang ialah lambang iman akan Tuhan sebagai landasan ilmu yang dikejar.

- b. Buku yang terbuka ialah lambang ilmu pengetahuan yang tidak kaku melainkan senantiasa terbuka dan dinamis.
- c. Salib ialah lambang pengorbanan Yesus Kristus yang hendaknya menjadi tauladan kita untuk berani berkorban.
- d. Tulisan Religio, Mores, Cultura berarti Iman, Moral, dan Budaya.

2.3 Lingkup Pekerjaan yang Dilakukan

Selama melakukan Kerja Praktik di biro LMC Unika De La Salle sebagai staf biro, berikut adalah beberapa aktivitas yang dilakukan:

1. Melakukan promosi untuk universitas baik secara daring atau luring.
2. Mengolah data PMB.
3. Membantu staf yang lain dalam tugas sehari-hari.

UKDLSM

BAB III

LANDASAN TEORI

3.1 Teori Pendukung

Pada bagian ini akan dibahas tentang teori yang digunakan sebagai dasar dalam pengembangan aplikasi seleksi *On-The-Spot* PMB Unika De La Salle Manado.

3.1.1 Sistem Penerimaan Mahasiswa Baru

Sistem Penerimaan Mahasiswa Baru (PMB) merupakan suatu sistem yang digunakan oleh banyak perguruan tinggi untuk mengatur dan mengelola seluruh proses penerimaan mahasiswa baru, mulai dari pendaftaran, pengumpulan data calon mahasiswa, pengolahan data calon mahasiswa, seleksi, pengolahan hasil seleksi, hingga pengumuman kelulusan. Sistem PMB dirancang agar dapat mempermudah proses administrasi dan pengelolaan data agar kegiatan penerimaan dapat berlangsung dengan lebih terstruktur [2].

Seiring dengan berkembangnya teknologi, proses PMB juga terdorong untuk ikut berkembang, dengan memanfaatkan sistem informasi berbasis web dalam pelaksanaan PMB memungkinkan proses pendaftaran dilakukan secara daring sehingga dapat mempermudah akses bagi calon mahasiswa serta membantu pihak universitas menyebarkan informasi terbaru. Selain itu sistem PMB berbasis web juga dapat mempercepat proses pengolahan data, meminimalisir *human error*, serta meningkatkan kualitas layanan bagi calon mahasiswa [2].

Dalam penerapannya sistem informasi PMB berbasis web terdiri atas beberapa komponen utama, yaitu data pendaftar, basis data untuk menyimpan informasi, antarmuka pengguna, serta fungsi untuk melakukan *export* dengan format pilihan. Integrasi antar komponen juga diperlukan agar data yang dimasukkan oleh pengguna dapat langsung tersimpan dan diproses secara otomatis [3].

3.1.2 Teknologi Pengembangan Sistem

Berikut ini adalah teori-teori seputar teknologi yang digunakan dalam pengembangan aplikasi seleksi *On-The-Spot* PMB Unika De La Salle Manado.

1. Aplikasi Web

Web App atau aplikasi web merupakan sebuah sistem perangkat lunak yang dijalankan melalui *browser* dan diakses melalui jaringan Internet tanpa harus melakukan instalasi secara langsung pada perangkat pengguna. Aplikasi web menggunakan struktur *client-server*, di mana sisi *client* bertanggung jawab untuk mengatur antarmuka yang ditampilkan kepada pengguna melalui *browser* dan sisi *server* bertugas untuk mengatur logika aplikasi, pemrosesan data, dan komunikasi dengan basis data. Perbedaan utama aplikasi web dengan *website*, yaitu umumnya *website* hanya menyediakan informasi sedangkan aplikasi web memungkinkan pengguna untuk berinteraksi langsung seperti mengelola data, melakukan autentikasi, bertransaksi, dan sebagainya [4].

Saat ini aplikasi web modern menggunakan berbagai macam teknologi seperti HTML, CSS, dan JavaScript untuk mengatur *front-end* dan bahasa pemrograman seperti python, java, PHP, atau yang paling populer JavaScript untuk mengatur *back-end*. Penggunaan teknologi tersebut memungkinkan aplikasi web untuk memiliki tampilan yang responsif dan dinamis, dapat diakses oleh berbagai macam perangkat, serta dapat mendukung berbagai proses yang kompleks. Dalam konteks ini aplikasi yang dikembangkan untuk membantu proses seleksi *On-The-Spot* PMB Unika De La Salle Manado merupakan sebuah aplikasi berbasis web atau *web app* [4].

2. HTML

HTML (*HyperText Markup Language*) merupakan pondasi utama dari WWW (*World Wide Web*) dan merupakan bahasa markup standar yang biasa digunakan untuk membuat struktur dan konten halaman sebuah web. HTML pertama kali diciptakan tahun 1991 oleh tim Berners-Lee. Sejak itu HTML telah mengalami banyak perubahan yang signifikan. Versi standar HTML saat ini yaitu HTML5 menjadi standar pengembangan web modern dan dikelola oleh W3C (*World Wide Web Consortium*) dan WHATWG (*Web HyperText Application Technology Working Group*). HTML sendiri bukan sebuah bahasa pemrograman, melainkan dokumen web yang menggunakan *tags* untuk mendefinisikan elemen-elemen dalam suatu dokumen web. HTML memiliki peran yang sangat penting karena merupakan

fundamental utama yang menyediakan struktur dasar dari sebuah web yang kemudian akan diperindah dengan CSS dan diatur dengan JavaScript [5] [6].

Struktur dokumen HTML diatur secara hierarkis dalam bentuk *Document Object Model* (DOM), diawali dengan deklarasi `<!DOCTYPE html>` yang menandakan dokumen HTML5, diikuti oleh elemen *root* `<html>`, yang berisikan `<head>` untuk *metadata* dan `<body>` untuk konten yang ditampilkan. HTML5 juga memperkenalkan elemen skematik yang memberikan makna lebih jelas terhadap struktur dokumen, seperti `<header>`, `<main>`, `<section>`, dan masih banyak lagi. Elemen ini tidak hanya meningkatkan *readability code*, tetapi juga membantu *search engine optimization* (SEO) dan aksesibilitas untuk *screen readers* bagi mereka yang berkebutuhan khusus. HTML5 membawa inovasi yang signifikan untuk memperluas kemampuan *web platform* tanpa bergantung pada *plugin* eksternal seperti Flash [6]. Dalam konteks pengembangan aplikasi web untuk seleksi *On-The-Spot* PMB Unika De La Salle Manado, HTML digunakan sebagai struktur dasar atau kerangka konten web.

3. CSS

CSS (*Cascading Style Sheets*) merupakan bahasa *stylesheet* yang bertugas untuk mengatur rancangan *visual* dan *layout* dokumen HTML, memisahkan antara aspek desain dan struktur sesuai prinsip *separation of concerns* dalam *web development*. CSS pertama kali diperkenalkan tahun 1996 oleh W3C dan sejak itu juga telah mengalami banyak perkembangan. Saat ini versi CSS yang menjadi standar industri adalah CSS3. Konsep ‘*Cascading*’ dalam CSS mengacu pada mekanisme di mana terdapat *multiple style rules* yang diterapkan pada satu elemen dengan *specificity* dan *source order* yang menentukan *rule* mana yang akhirnya diaplikasikan. CSS memungkinkan para pengembang untuk menciptakan identitas visual yang tak hanya konsisten dalam sebuah *website*, tapi juga dapat meningkatkan *maintainability* karena perubahan desain semuanya dapat dilakukan dalam satu tempat tanpa harus memodifikasi struktur HTML itu sendiri [5] [7].

Arsitektur CSS terdiri dari selektor yang menargetkan elemen HTML dengan deklarasi yang mendefinisikan properti dan *style value*. CSS3 memperkenalkan *advanced selectors* seperti *attribute selectors*, *pseudo-classes* (`:hover`, `:focus`) dan

pseudo-elements (::before, ::after) yang dapat memberikan kontrol granular tanpa menambah markup HTML. CSS3 juga memperkenalkan *flexible layout systems* seperti *Flexbox (Flexible Box Layout)* untuk *layout* satu dimensi dan *CSS Grid* untuk *layout* dua dimensi. Banyak juga fitur-fitur lanjutan CSS3 yang dalam implementasinya perlu digunakan bersamaan dengan JavaScript. Ditambah dengan transisi dan animasi yang memungkinkan perubahan visual secara halus untuk meningkatkan pengalaman penggunaan, kontrol atas *timing functions*, *keyframe animation*, dan masih banyak lagi [7]. Dalam konteks pengembangan aplikasi *web* untuk seleksi *On-The-Spot* PMB Unika De La Salle Manado, CSS digunakan untuk menciptakan antarmuka yang intuitif dan atraktif.

4. JavaScript

JavaScript merupakan bahasa pemrograman *high-level*, *interpreted*, dan juga *multi-paradigm* yang saat ini sudah menjadi standar industri untuk bahasa pemrograman yang digunakan dalam *web development*. JavaScript pertama kali diciptakan tahun 1995 oleh Brendan Eich hanya dalam waktu 10 hari saja untuk digunakan di Netscape Navigator. Saat ini JavaScript telah berkembang jauh melampaui fungsi awalnya sebagai *scripting language* sederhana. JavaScript kini telah menjadi ekosistem lengkap yang mencakup *frontend frameworks*, *backend runtime*, *mobile development*, *desktop application*, dan bahkan *IoT programming*. ECMAScript (ES) oleh ECMA International menjadi *blueprint* yang mengatur standarisasi JavaScript untuk memastikan konsistensi implementasi di berbagai *browser* [8].

Paradigma pemrograman JavaScript terkenal atas fleksibilitasnya JavaScript tidak hanya mendukung *imperative programming*, tapi juga *object-oriented programming*, dan *functional programming styles*. JavaScript adalah *dynamically typed language* dengan fungsi kelas satu, memungkinkan fungsi-fungsi ini diperlakukan sebagai nilai yang dapat di-*pass* sebagai argumen atau sebagai *value return* dari fungsi yang lain. Satu hal yang membedakan JavaScript dengan bahasa pemrograman OOP tradisional adalah penerapan *prototypal inheritance* di mana bahasa pemrograman tradisional masih menggunakan konsep *classical inheritance*. *Event-driven* dan *asynchronous programming* menjadi ciri khas utama JavaScript,

dengan *event loop mechanism* yang memungkinkan *non-blocking operations* untuk diterapkan di *responsive web applications* [8]. Dalam konteks pengembangan aplikasi web untuk seleksi *On-The-Spot* PMB Unika De La Salle Manado, JavaScript digunakan untuk mengatur *behavior* dari komponen dalam web.

5. *Firestore Database Management System*

Firestore merupakan *platform Backend-as-a-Service* (BaaS) yang dikembangkan oleh *Firestore* dan diakuisisi oleh Google tahun 2014 silam. *Firestore* menyediakan berbagai macam layanan dan *tools* berbasis *cloud* untuk mengembangkan atau mengelola aplikasi baik itu *mobile* ataupun web. *Platform* ini menghilangkan kebutuhan untuk mengatur dan mengelola infrastruktur *server* sendiri, sehingga dapat membantu para pengembang untuk fokus pada pengembangan fitur-fitur penting aplikasi. *Firestore* mengadopsi filosofi *serverless architecture* di mana para pengembang hanya perlu menuliskan logika aplikasi tanpa harus mengatur *server provisioning, scaling, ataupun maintenance* [9].

Ekosistem *Firestore* ini terdiri dari beberapa layanan yang dapat digunakan modularly sesuai dengan kebutuhan masing-masing aplikasi. *Cloud Firestore* dan *Realtime Database* merupakan beberapa fitur yang dapat digunakan dalam pengembangan aplikasi seleksi *On-The-Go* PMB. RTDB dan *Firestore* ini merupakan NoSQL *database* dengan kemampuan *real-time synchronization*. RTDB menawarkan simple *JSON-based data structure* dengan tingkat latensi yang rendah sementara *Firestore* menawarkan data model yang lebih terstruktur dengan skalabilitas yang lebih kompleks. Selain itu *Firestore* juga menawarkan banyak fitur lain seperti *Authentication, Hosting, A/B Testing*, dan masih banyak lagi [9]. Dalam konteks pengembangan aplikasi web untuk seleksi *On-The-Spot* PMB Unika De La Salle Manado, *Firestore* digunakan untuk menyimpan data aplikasi seperti informasi mahasiswa ataupun soal pertanyaan.

6. *Visual Studio Code*

Visual Studio Code (VS Code) merupakan *source code editor* yang dikembangkan oleh Microsoft dan dirilis sebagai *open-source software* tahun 2015

silam, selang beberapa waktu VS Code menjadi salah satu *tools development* terpopuler di kalangan para pengembang di dunia. VS Code menggabungkan simplisitas dan *lightweight nature* dari *text editor* tradisional dengan fitur-fitur yang modern dan kompleks dan menciptakan titik nyaman yang menjadi daya tarik utama VS Code. Dibangun menggunakan Electron, *Framework* VS Code dapat berjalan di berbagai *platform* baik itu Windows, Mac OS, ataupun Linux [10].

Fitur utama VS Code mencakup IntelliSense yang dapat memberikan rekomendasi kode sesuai dengan tipe variabel, definisi fungsi, dan modul yang diimpor. *Integrated debugging* juga salah satu fitur unggulan VS Code dengan dukungan untuk *breakpoints*, *call stacks*, *interactive console*, dan *debug configurations* untuk bahasa pemrograman dan *framework* berbeda tanpa memerlukan *debugger* eksternal. Selain dua fitur tadi ada juga ekstensibilitas yang menjadi *marketplace* berisikan ribuan bahkan puluhan ribu ekstensi *virtual* yang dapat digunakan untuk mendukung pengembangan aplikasi. Ekstensi dapat menambahkan dukungan bahasa, tema kustom untuk kustomisasi *visual*, *snippets* untuk *code generation*, dan sebagainya [10]. Dalam konteks pengembangan aplikasi web untuk seleksi *On-The-Spot* PMB Unika De La Salle Manado, *Visual Studio Code* digunakan untuk sebagai *text editor* utama untuk pengembangan aplikasi web.

3.2 Metodologi Pengembangan Perangkat Lunak

Berikut ini adalah teori-teori seputar metodologi dan kakas pemodelan yang digunakan dalam pengembangan aplikasi seleksi *On-The-Spot* PMB Unika De La Salle Manado.

3.2.1 Metode Pengembangan Sistem

Dalam penelitian ini metode pengembangan sistem yang digunakan adalah Kanban. Kanban sendiri merupakan metode manajemen alur kerja yang bertujuan untuk mempermudah proses pengembangan perangkat lunak melalui visualisasi pekerjaan, pembatasan pekerjaan yang sedang berjalan, sampai perbaikan proses secara berkelanjutan. Menurut studi dari Anderson dan Carmichael, Kanban dapat membantu tim pengembang dalam mengelola alur pekerjaan dengan cara memvisualisasikan setiap tugas pada papan Kanban sehingga perkembangan

pengembangan dapat dipantau dengan lebih mudah dan transparan. Metode Kanban juga memungkinkan pengembang untuk mengidentifikasi hambatan yang terjadi selama proses pengembangan dan melakukan penyesuaian yang diperlukan secara cepat tanpa mengganggu pekerjaan yang sedang berlangsung [11].

Kanban banyak dipilih karena tingkat fleksibilitasnya yang tinggi sehingga dapat merespon terhadap perubahan kebutuhan dengan lebih baik dibandingkan metode lain seperti *Waterfall*. Dalam papan Kanban biasanya terbagi menjadi beberapa bagian, setiap pekerjaan yang akan dilakukan sebelumnya akan dicatat dalam daftar pekerjaan (*Backlog*) terlebih dahulu. Papan Kanban kemudian akan dibagi untuk setiap tahapan, yaitu *To do*, *In Progress*, *Testing*, dan *Done*. Tahapan *To do* berisikan daftar pekerjaan yang akan dikerjakan, tahapan *In Progress* berisikan daftar pekerjaan yang sedang dikerjakan, *Testing* berisikan daftar pekerjaan yang dalam tahapan pengujian, dan tahapan *Done* berisikan daftar pekerjaan yang telah selesai dikerjakan [11]. Dalam konteks pengembangan aplikasi web untuk seleksi *On-The-Spot* PMB Unika De La Salle Manado, metodologi Kanban dipilih agar pengembangan dapat berjalan lebih terstruktur, mudah dipantau, dan mampu beradaptasi dengan perubahan.

3.2.2 Kakas Pemodelan

Berikut ini adalah teori-teori seputar kakas pemodelan yang digunakan dalam pengembangan aplikasi seleksi *On-The-Spot* PMB Unika De La Salle Manado.





1. *Data Flow Diagram*

Data Flow Diagram (DFD) merupakan salah satu teknik pemodelan yang banyak digunakan dalam analisis dan perancangan sistem atau aplikasi. DFD memiliki peran penting dalam proses pengembangan sistem karena dapat membantu mengidentifikasi kebutuhan fungsional aplikasi. Dengan menggambarkan aliran data dan hubungan antar proses, pengembang mendapatkan representasi visual tentang bagaimana sistem bekerja dan dapat mengetahui fungsi-fungsi utama yang harus tersedia pada sistem serta dapat meminimalisir kemungkinan terjadi kendala dalam proses pengembangan [12].

Dalam implementasinya, DFD terdiri dari empat komponen utama, entitas eksternal yang merepresentasikan pihak diluar dari sistem, proses yang

merepresentasikan aktivitas pengolahan data dalam sistem, *data store* yang merepresentasikan tempat penyimpanan data, dan *data flow* yang digunakan untuk merepresentasikan aliran data yang menghubungkan entitas dengan proses atau *data store*. DFD umumnya disusun secara hierarki. Tingkatan pertama biasa disebut Diagram Konteks (Level 0) yang menggambarkan sistem sebagai satu proses tunggal dan menunjukkan hubungan antara sistem dengan entitas eksternal. DFD Level 1 digunakan untuk menggambarkan proses utama menjadi beberapa sub proses yang lebih terperinci, dan apabila penjelasan masih diperlukan maka akan dibuat DFD Level 2 dan seterusnya dan dalam pengembangan ini, akan digunakan DFD dengan notasi Yourdon & De Marco [12].

Tabel 3.1 Simbol-Simbol *Data Flow Diagram* [12]

Simbol	Keterangan
	Entitas eksternal: Menggambarkan pihak diluar dari sistem.
	Proses: Menggambarkan aktivitas pengolahan data dalam sistem
	<i>Data flow</i> : Menggambarkan aliran data yang menghubungkan entitas dengan proses atau <i>data store</i> .
	<i>Data Store</i> : Menggambarkan tempat untuk menyimpan data.

2. *Flowchart*

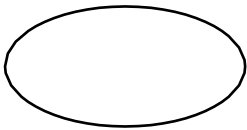

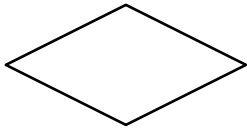

Flowchart merupakan suatu diagram umum yang banyak digunakan untuk menggambarkan urutan langkah-langkah dalam suatu proses, sistem, atau algoritma dengan menggunakan simbol-simbol sederhana yang saling terhubung satu sama lain. *Flowchart* berfungsi sebagai alat visual yang dapat membantu pengguna untuk memahami bagaimana proses suatu sistem atau algoritma berlangsung dari awal

hingga akhir secara berurutan. *Flowchart* dapat membantu menganalisis, merancang, mendokumentasikan, bahkan mengkomunikasikan suatu proses bisnis atau logika sehingga dapat mempermudah proses identifikasi masalah atau perbaikan [13].

Dalam pengembangan perangkat lunak, *Flowchart* memiliki peranan penting sebagai media untuk memodelkan proses sebelum implementasi dilakukan. *Flowchart* dapat membantu pengembang untuk memahami alur proses, hubungan antar aktivitas, keputusan, *input*, hingga *output* yang dihasilkan. Implementasi *Flowchart* yang tepat juga dapat meningkatkan efisiensi, mempermudah pemeliharaan sistem, dan membantu proses pengambilan keputusan dalam perancangan dan implementasi [13].

Dalam konteks pengembangan aplikasi Seleksi *On-The-Spot* PMB Unika De La Salle *Flowchart* digunakan untuk menggambarkan alur bisnis atau proses penerimaan mahasiswa baru yang sedang berjalan, hal ini dilakukan untuk memberikan pengembang gambaran visual tentang bagaimana alur sistem harus dibangun.

Tabel 3.2 Simbol-Simbol *Flowchart* [13]

Simbol	Keterangan
	Mulai/Akhir: Simbol yang digunakan menandakan awal atau akhir dari diagram alur.
	Proses: Simbol yang digunakan untuk menandakan aksi atau tindakan spesifik dalam sebuah proses.
	<i>Decision</i> : Simbol yang digunakan untuk menandakan pilihan dengan hasil <i>output</i> yang berbeda,
	<i>Flow</i> : Simbol yang digunakan untuk menandakan alur diagram.

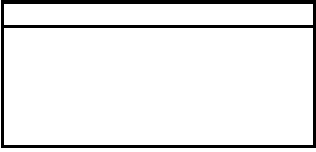


3. *Collection Document Diagram*

Collection Document Diagram merupakan diagram yang sering digunakan untuk merepresentasikan struktur data pada basis data NoSQL berorientasi dokumen. CDD menggambarkan bagaimana data diorganisasikan ke dalam serangkaian kumpulan (*collection*) dan dokumen (*document*). Berbeda dengan model basis data relasional yang menggunakan tabel dan baris, CDD menyimpan data dalam bentuk dokumen yang umumnya menyerupai struktur JSON (*JavaScript Object Notation*) dan memiliki tingkat fleksibilitas data yang tinggi. Salah satu basis data ternama yang menggunakan struktur ini yaitu *Firebase Database* oleh Google. Salah satu fitur yang ditawarkan oleh *Firebase* yaitu *Firestore* yang mana merupakan basis data berbasis *cloud* yang menyimpan data dalam dokumen yang dikelompokkan dalam *collection*, sehingga membentuk struktur data yang bersifat hierarkis [14].

Komponen utama dalam *Collection Document Diagram* terdiri dari *collection*, *document*, dan *field*. *Collection* bertindak sebagai wadah untuk mengelompokkan beberapa dokumen dengan jenis data yang serupa. Sedangkan *document* merupakan tempat penyimpanan utama untuk menyimpan sekumpulan data dalam bentuk *key-value*, dan *field* adalah atribut yang tersimpan dalam dokumen seperti teks, angka, dan lain sebagainya. Dalam CDD terdapat juga kondisi disebut *subcollection* dan *nested object*. *Subcollection* adalah peristiwa di mana *collection* berada di dalam sebuah *document*, sedangkan *nested object* merupakan struktur objek yang tersimpan sebagai bagian dari *field* dalam sebuah *document* [14].

Dalam konteks pengembangan aplikasi Seleksi *On-The-Spot* PMB Unika De La Salle *Collection Document Diagram* digunakan untuk menggambarkan struktur basis data, hal ini dilakukan agar dapat mempermudah pengembang ketika mengembangkan proses CRUD (*Create, Read, Update, Delete*) yang konsisten.

Tabel 3.3 Simbol-Simbol *Collection Document Diagram* [14]

Simbol	Keterangan
	<i>Collection/Document:</i> Digunakan untuk menggambarkan <i>Collection</i> atau <i>Document</i> .
	<i>Field:</i> Digunakan untuk menggambarkan data yang disimpan.
	<i>Flow:</i> Digunakan untuk menggambarkan relasi antar <i>Collection</i> .

UKDLSM

BAB IV PEMBAHASAN

4.1 Analisis

Pada tahap ini akan dilakukan identifikasi kebutuhan dan masalah yang dialami melalui observasi, wawancara, dan studi literatur. Dalam Kanban, tahapan ini dimulai dari penyusunan daftar kebutuhan (*To-Do*), analisis data (*In Progress*), dan rumusan kebutuhan sistem (*Done*). Tahapan ini akan menjelaskan analisis target pengguna, sumber data yang digunakan, pemodelan sistem berjalan, persyaratan dan preferensi sistem baru, dan identifikasi masalah dan kesempatan.

4.1.1 Analisis Target Pengguna

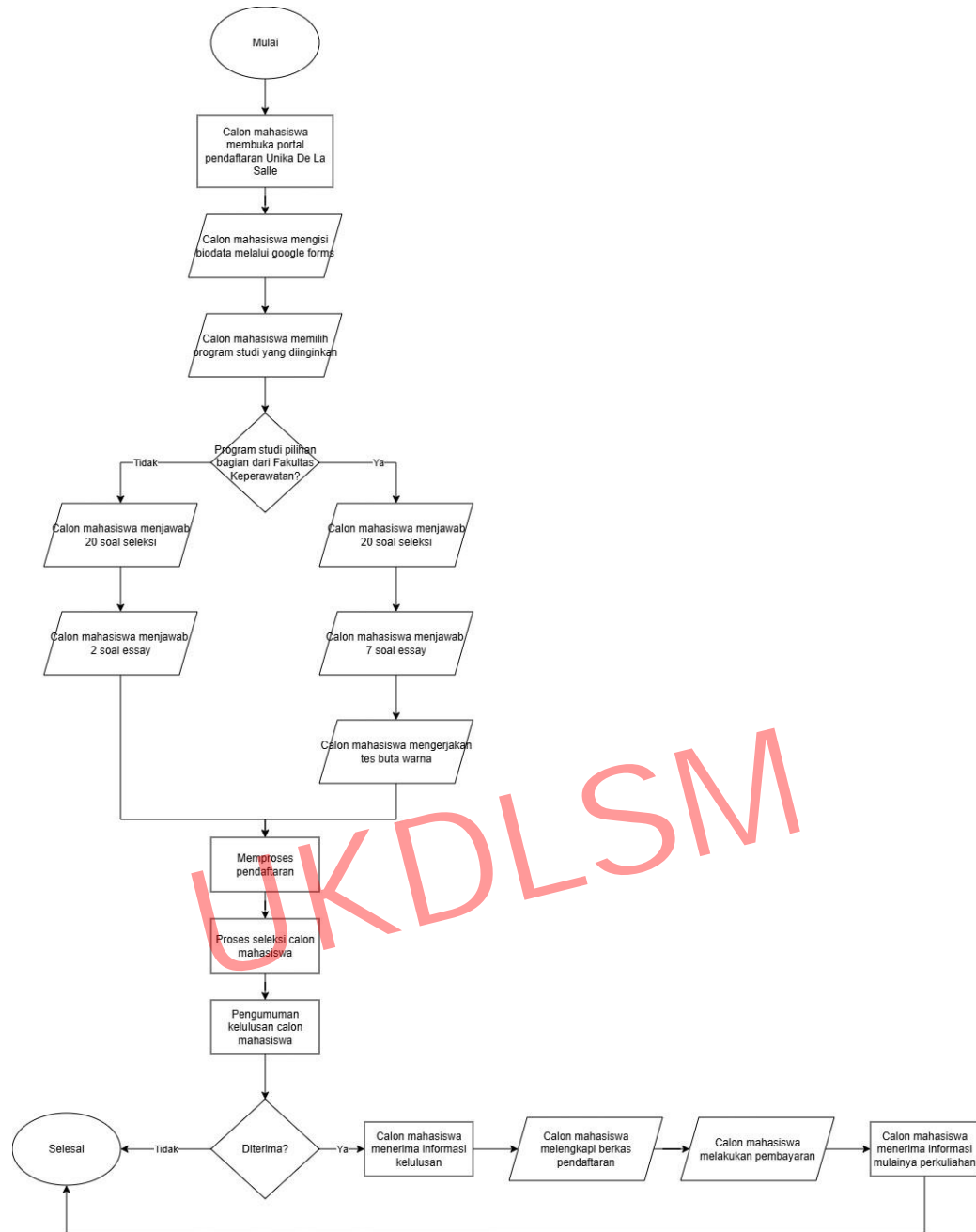
Target pengguna dari aplikasi *On-The-Spot* PMB ini tentunya calon mahasiswa baru yang ingin mendaftarkan diri di Universitas Katolik De La Salle Manado dan juga staf LMC. Aplikasi seleksi *On-The-Spot* PMB Unika De La Salle Manado dibangun untuk memudahkan proses pendaftaran dan seleksi calon mahasiswa baru dan membantu staf LMC untuk mengolah data mahasiswa yang mendaftar.

4.1.2 Sumber Data

Data yang digunakan selama pengembangan aplikasi seleksi *On-The-Spot* PMB Unika De La Salle Manado diperoleh dari hasil observasi dan juga wawancara dengan staf dan kepala biro LMC Unika De La Salle. Data yang diperoleh kemudian akan diolah menjadi informasi yang diatur sesuai dengan kebutuhan aplikasi.

4.1.3 Pemodelan Sistem yang Sedang Berjalan

Berikut ini adalah pemodelan sistem yang sedang berjalan untuk saat ini yang digambarkan menggunakan *flowchart*.



Gambar 4.1 Pemodelan Sistem yang Sedang Berjalan

4.1.4 Persyaratan dan Preferensi Sistem Baru

Pada Kerja Praktik ini akan dibangun sebuah aplikasi seleksi *On-The-Spot* PMB yang dirancang khusus untuk Unika De La Salle Manado. Aplikasi ini dibuat agar dapat membantu staf LMC mengolah data calon mahasiswa baru di Universitas Katolik De La Salle Manado. Aplikasi ini juga dibuat untuk mempermudah calon mahasiswa yang ingin mendaftar sehingga seluruh proses pendaftaran sampai seleksi dapat dilakukan secara daring. Aplikasi ini akan dibangun di lingkungan

web sehingga dapat diakses dengan mudah oleh siapa saja dan di mana saja terlepas dari perangkat apa yang digunakan. Basis data yang digunakan adalah *Firestore* karena menawarkan banyak kemudahan dalam pengembangan aplikasi.

Berdasarkan uraian di atas, berikut ini adalah daftar fitur yang akan tersedia dalam aplikasi seleksi *On-The-Spot* PMB Unika De La Salle Manado nantinya:

1. Pendaftaran dan seleksi calon mahasiswa baru.
2. Admin dapat menambahkan, mengubah, ataupun menambahkan pertanyaan baru.
3. Admin dapat menambahkan, mengubah, ataupun menambahkan halaman baru.
4. Sistem dapat melakukan *export* laporan pendaftaran dalam bentuk *.CSV*.

4.1.5 Identifikasi Masalah dan Kesempatan

Berikut ini merupakan hasil identifikasi masalah beserta solusi yang ditawarkan untuk sistem seleksi *On-The-Spot* PMB Unika De La Salle Manado.

Tabel 4.1 Tabel Masalah dan Kesempatan

Masalah	Kesempatan
Jumlah calon mahasiswa yang cukup banyak sehingga mempersulit pengolahan data pendaftar secara manual.	Mengembangkan aplikasi seleksi <i>On-The-Spot</i> PMB berbasis web yang dapat membantu bukan hanya staf LMC dalam mengolah data calon mahasiswa tapi juga agar calon mahasiswa dapat melakukan pendaftaran dengan lebih mudah.

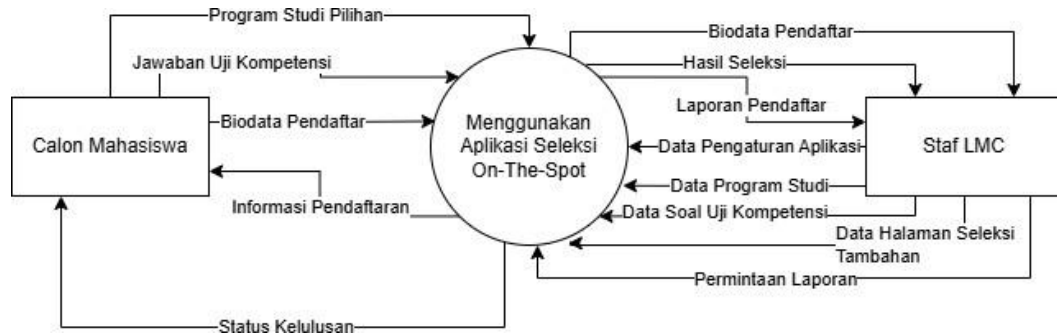
4.2 Perancangan

Pada tahap ini akan dilakukan rancangan arsitektur sistem dan tampilan antarmuka aplikasi. Dalam Kanban, tahapan ini dimulai dari identifikasi arsitektur sistem dan kebutuhan antarmuka aplikasi (*To-Do*), merancang pemodelan arsitektur aplikasi menggunakan *Data Flow Diagram* dan merancang antarmuka aplikasi (*In Progress*), hingga menghasilkan rancangan antarmuka dan arsitektur sistem yang siap di implementasikan (*Done*). Tahapan ini akan mencakup *Data Flow Diagram*, rancangan antarmuka aplikasi, dan rancangan struktur basis data.

4.2.1 Data Flow Diagram

Berikut ini merupakan rancangan pemodelan *Data Flow Diagram* yang digunakan pada sistem seleksi *On-The-Spot* PMB Unika De La Salle Manado.

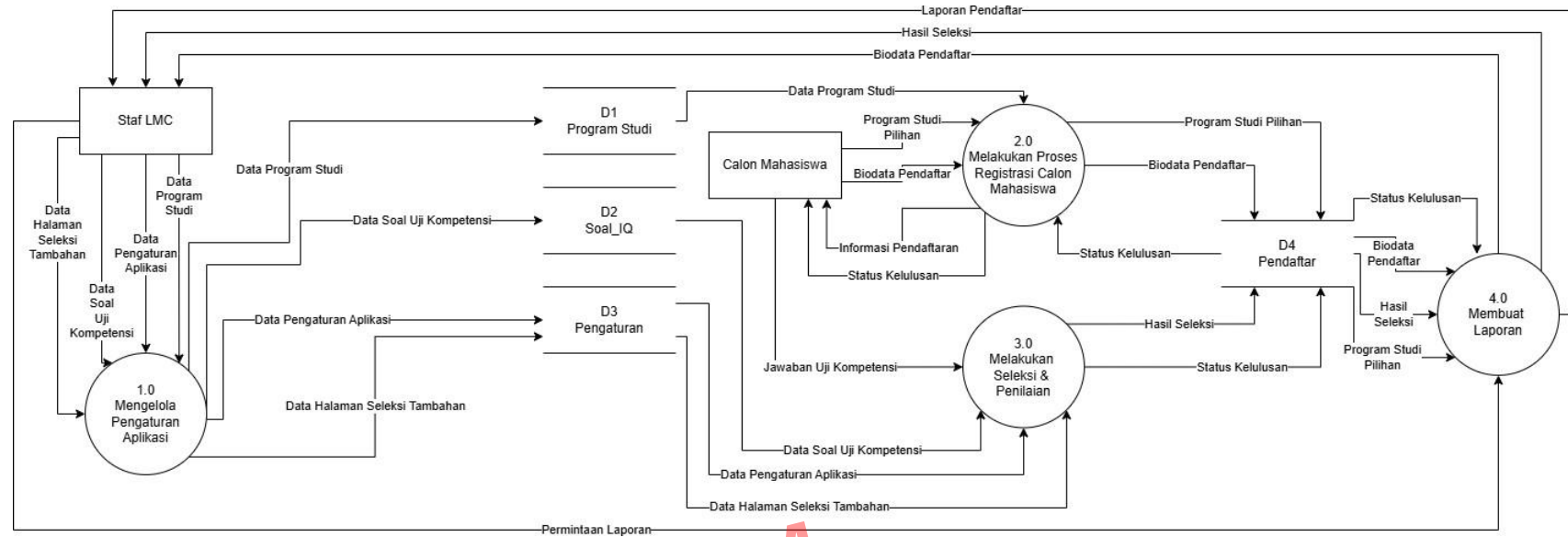
1. Context Diagram (Level 0)



Gambar 4.2 Data Flow Diagram Level 0

UKDLSM

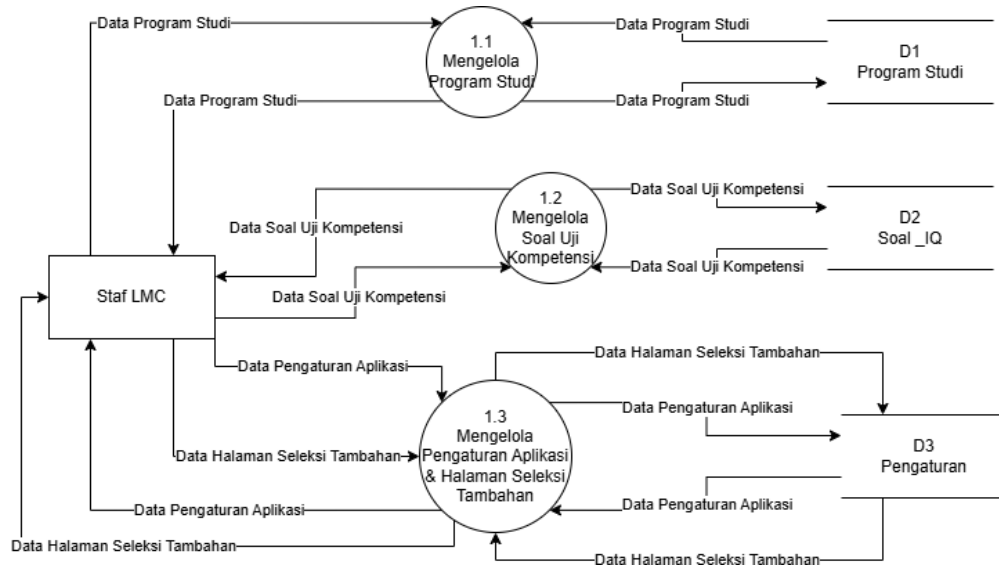
2. Data Flow Diagram Level 1



Gambar 4.3 Data Flow Diagram Level 1

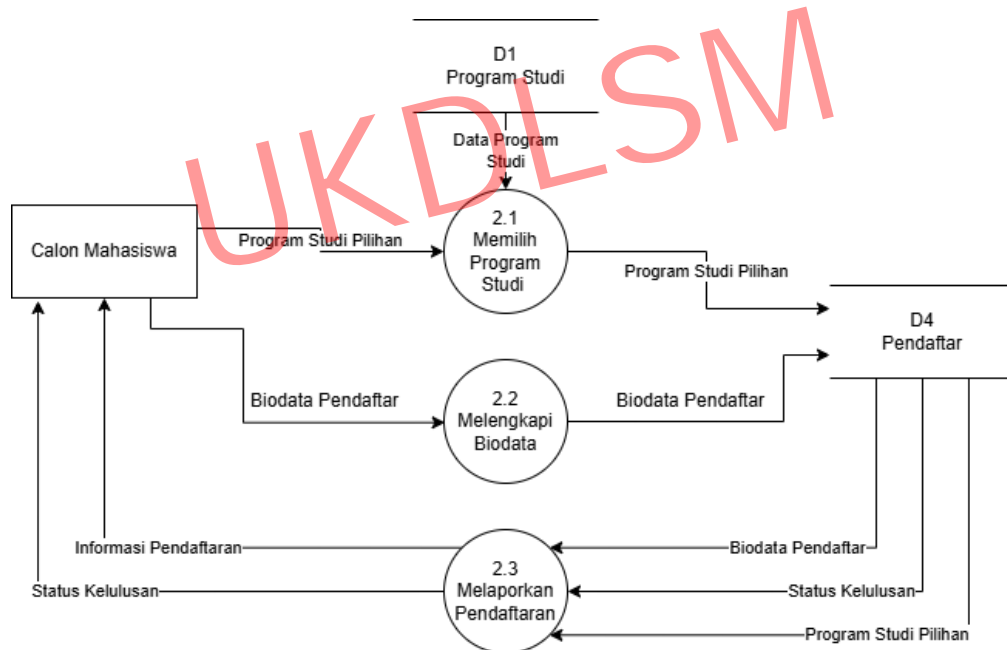
UKDLSM

3. *Data Flow Diagram Level 2 1.0*



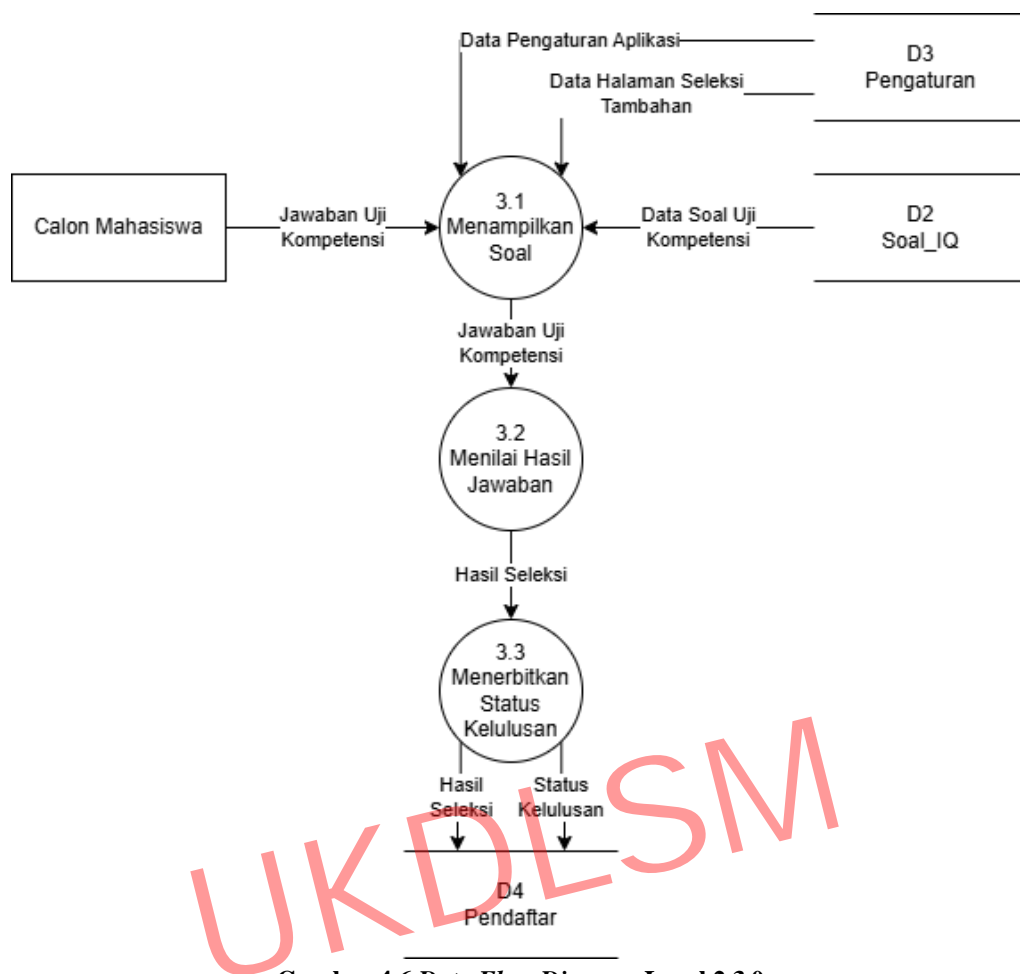
Gambar 4.4 Data Flow Diagram Level 2 1.0

4. *Data Flow Diagram Level 2 2.0*



Gambar 4.5 Data Flow Diagram Level 2 2.0

5. Data Flow Diagram Level 2 3.0



Gambar 4.6 Data Flow Diagram Level 2 3.0

4.2.2 Rancangan Antarmuka Aplikasi

Berikut ini merupakan rancangan antarmuka dari aplikasi aplikasi *On-The-Spot* Unika De La Salle yang terdiri dari 8 halaman dan 5 tampilan *pop-up*: Halaman *Landing*, Halaman Seleksi, Halaman *Dashboard*, Halaman Data Pendaftar, Halaman Manajemen Halaman, Halaman Manajemen Pertanyaan, Halaman Manajemen Program Studi, dan Halaman Pengaturan; Tampilan *Login Admin*, Tampilan Detail Pendaftar, Tampilan Edit Halaman, Tampilan Edit Soal, Tampilan Edit Program Studi.

1. Halaman *Landing*

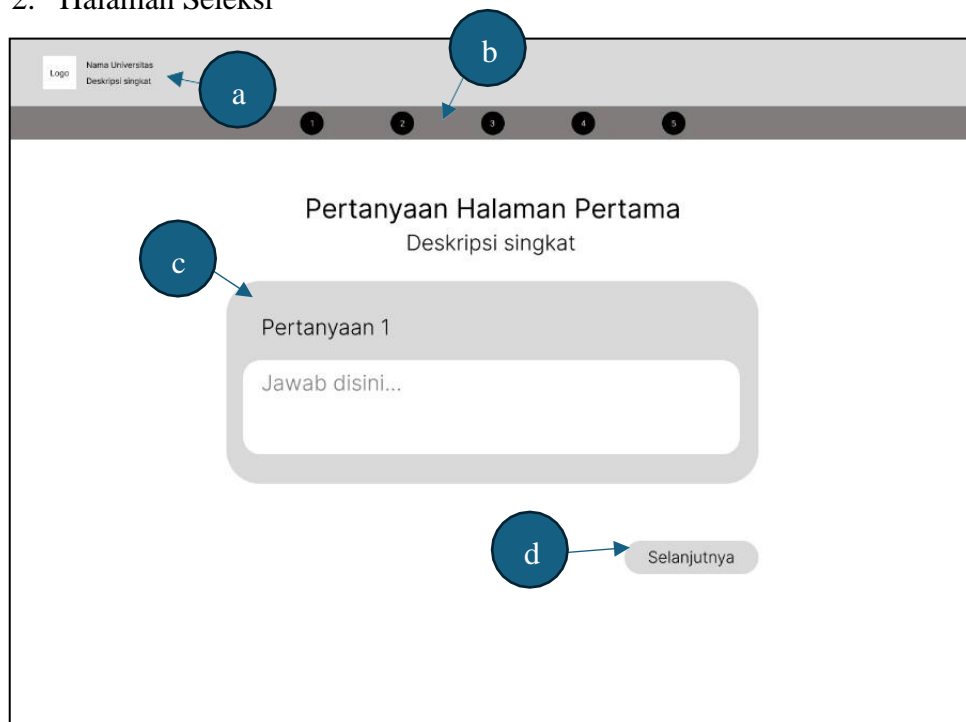


Gambar 4.7 Rancangan Halaman *Landing*

Keterangan:

- a. Logo dan nama Universitas.
- b. Tombol untuk *Login* Admin.
- c. Tombol untuk mulai proses pendaftaran dan seleksi.

2. Halaman Seleksi

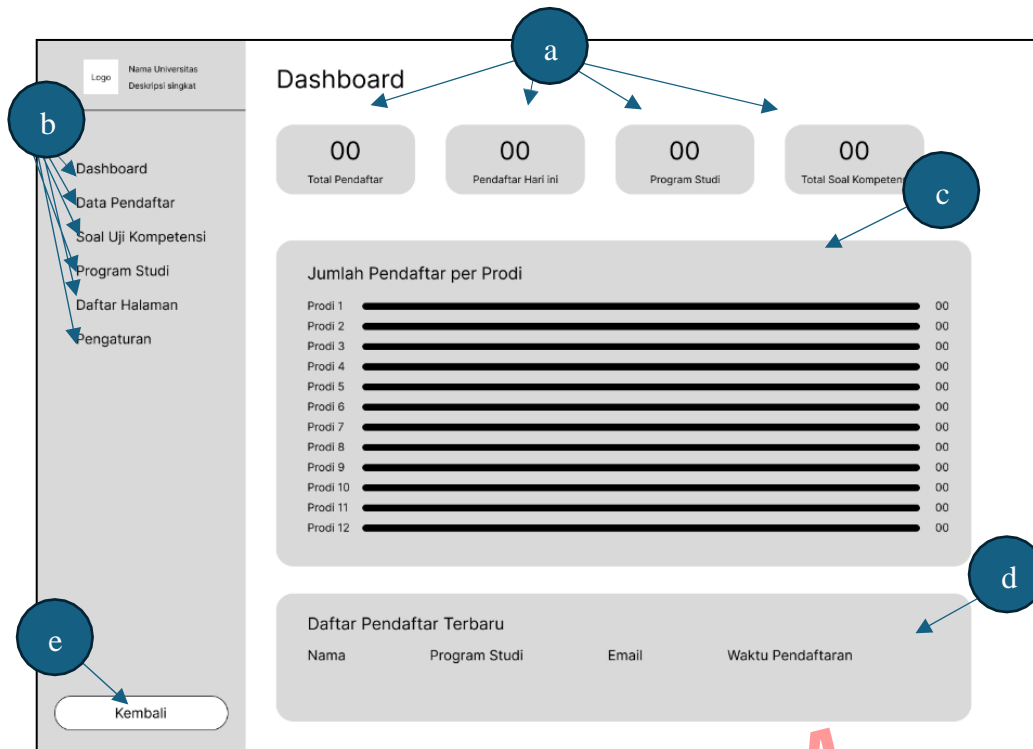


Gambar 4.8 Rancangan Halaman Seleksi

Keterangan:

- a. Logo dan nama Universitas.
- b. *Progress bar* untuk melihat kemajuan seleksi.
- c. *Card view* untuk menampilkan pertanyaan seleksi.
- d. Tombol untuk lanjut ke halaman selanjutnya.

3. Halaman *Dashboard*

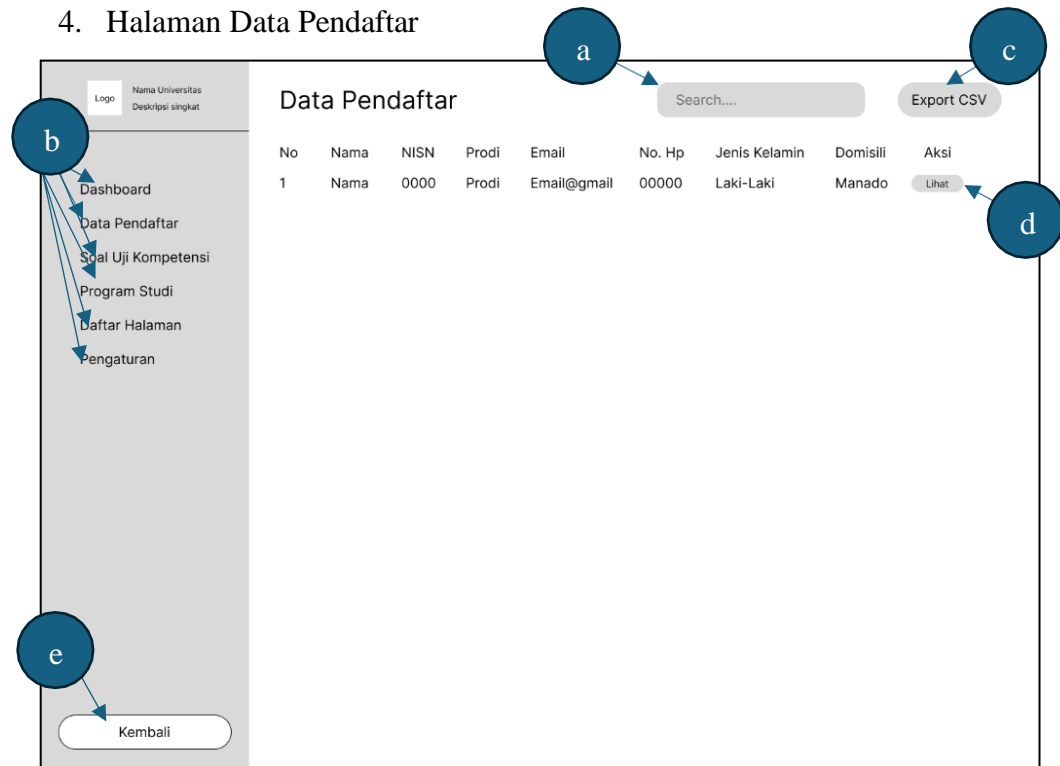


Gambar 4.9 Rancangan Halaman *Dashboard*

Keterangan:

- Card view* yang menampilkan informasi dasar.
- Pilihan untuk mengganti halaman yang ditampilkan.
- Card view* untuk menampilkan jumlah pendaftar per prodi.
- Card view* untuk menampilkan pendaftar terbaru.
- Tombol untuk kembali ke halaman *landing*.

4. Halaman Data Pendaftar

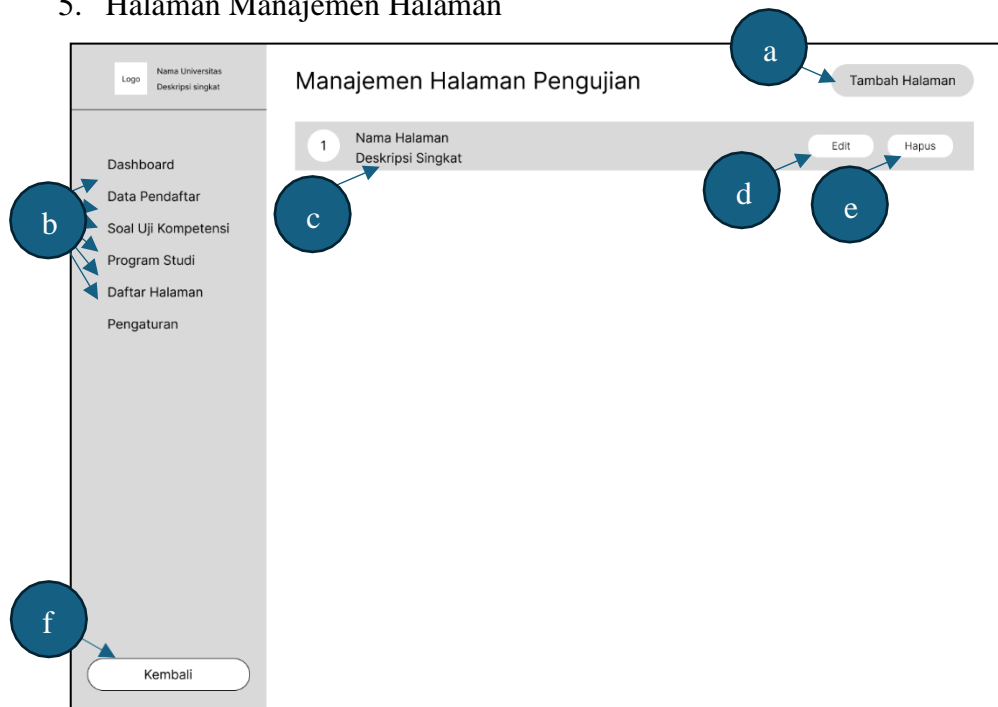


Gambar 4.10 Rancangan Halaman Data Pendaftar

Keterangan:

- a. *Search bar* yang dapat digunakan untuk mencari pendaftar.
- b. Pilihan untuk mengganti halaman yang ditampilkan.
- c. Tombol untuk ekspor data ke format CSV.
- d. Tombol untuk melihat lebih lengkap informasi pendaftar.
- e. Tombol untuk kembali ke halaman *landing*.

5. Halaman Manajemen Halaman

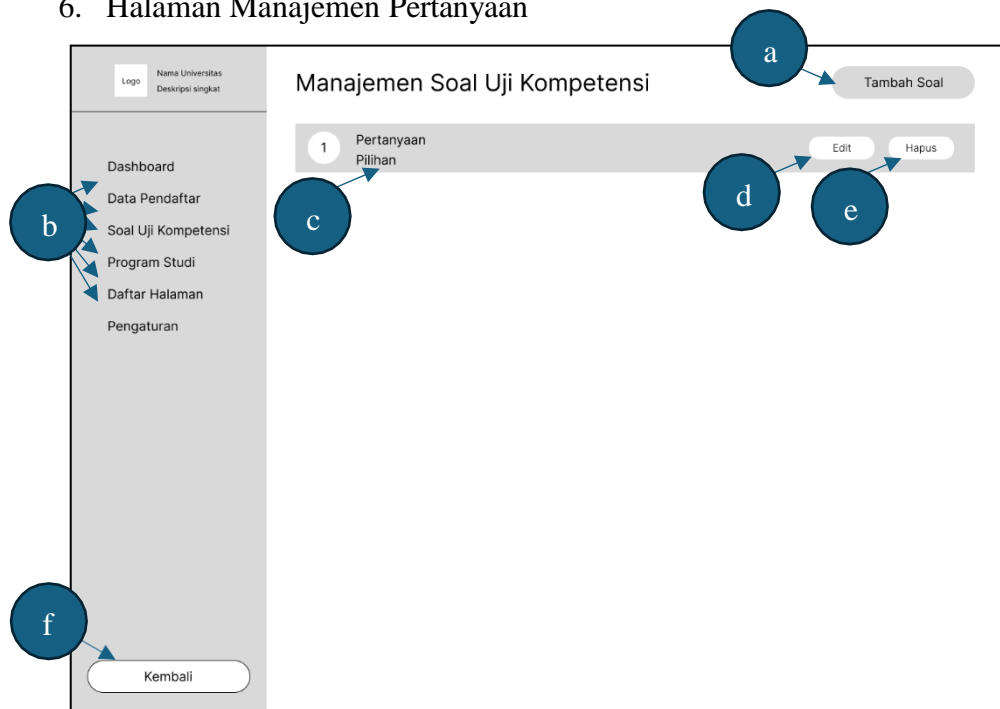


Gambar 4.11 Rancangan Halaman Manajemen Halaman

Keterangan:

- a. Tombol untuk menambahkan halaman seleksi tambahan.
- b. Pilihan untuk mengganti halaman yang ditampilkan.
- c. *Card view* untuk menampilkan halaman seleksi tambahan.
- d. Tombol untuk memperbarui isi halaman seleksi tambahan.
- e. Tombol untuk menghapus halaman seleksi tambahan.
- f. Tombol untuk kembali ke halaman *landing*.

6. Halaman Manajemen Pertanyaan

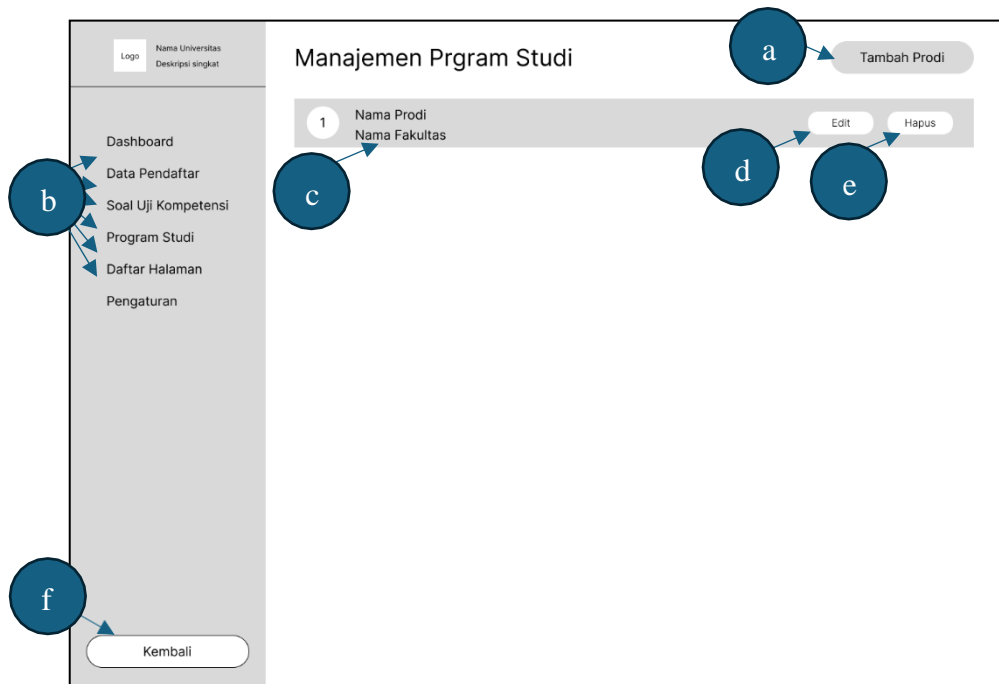


Gambar 4.12 Rancangan Halaman Manajemen Pertanyaan

Keterangan:

- a. Tombol untuk menambahkan soal uji kompetensi.
- b. Pilihan untuk mengganti halaman yang ditampilkan.
- c. *Card view* untuk menampilkan pertanyaan uji kompetensi.
- d. Tombol untuk memperbarui isi pertanyaan.
- e. Tombol untuk menghapus pertanyaan.
- f. Tombol untuk kembali ke halaman *landing*.

7. Halaman Manajemen Program Studi

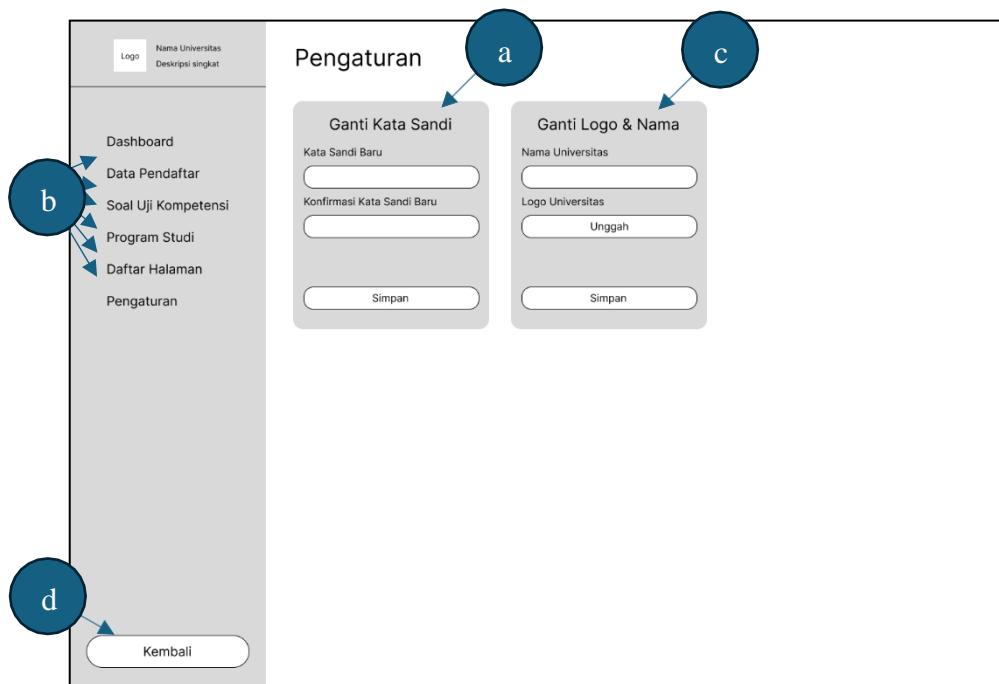


Gambar 4.13 Rancangan Halaman Manajemen Program Studi

Keterangan:

- a. Tombol untuk menambahkan program studi.
- b. Pilihan untuk mengganti halaman yang ditampilkan.
- c. *Card view* untuk menampilkan program studi yang terdaftar.
- d. Tombol untuk memperbarui informasi program studi.
- e. Tombol untuk menghapus program studi.
- f. Tombol untuk kembali ke halaman *landing*.

8. Halaman Pengaturan

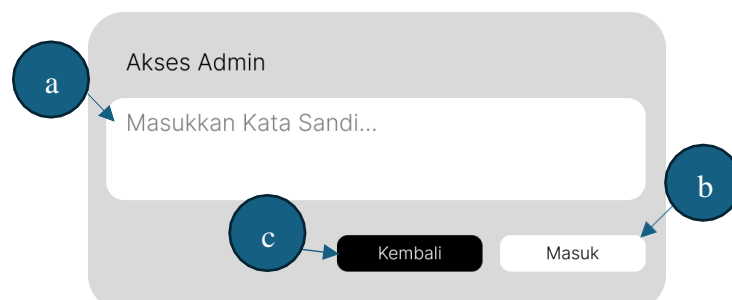


Gambar 4.14 Rancangan Halaman Pengaturan

Keterangan:

- Card view* untuk mengganti kata sandi.
- Pilihan untuk mengganti halaman yang ditampilkan.
- Card view* untuk mengganti nama dan logo universitas.
- Tombol untuk kembali ke halaman *landing*.

9. Tampilan *Login Admin*



Gambar 4.15 Rancangan Tampilan *Login Admin*

Keterangan:

- Text field* untuk memasukkan kata sandi.
- Tombol untuk menutup tampilan *login*.
- Tombol untuk menyelesaikan proses *login*.

10. Tampilan Detail Pendaftar

Detail Pendaftar

Nama	NISN
Tempat/Tanggal Lahir	Jenis Kelamin
Email	No HP
Domisili	Waktu Daftar
Skor Uji Kompetensi	Status

a → Kembali Hapus ← b

Gambar 4.16 Rancangan Halaman Detail Pendaftar

Keterangan:

- Tombol untuk menutup tampilan detail pendaftar.
- Tombol untuk menghapus data pendaftar yang dipilih.

11. Tampilan Edit Halaman

Edit Halaman

a → Nama Halaman

Deskripsi

b →

c → Urutan Halaman

Tampilkan Khusus Prodi

d →

e → Kembali Simpan ← f

Gambar 4.17 Rancangan Tampilan Edit Halaman

Keterangan:

- Text field* untuk mengisi nama halaman.
- Text field* untuk mengisi deskripsi singkat.
- Text field* untuk mengisi urutan halaman.
- Text field* untuk mengisi tampilan khusus prodi.
- Tombol untuk menutup tampilan edit halaman.
- Tombol untuk menyimpan halaman tambahan.

12. Tampilan Edit Soal

Gambar 4.18 Rancangan Tampilan Edit Soal

Keterangan:

- a. *Text field* untuk mengisi pertanyaan.
- b. *Text field* untuk mengisi pilihan yang ditampilkan.
- c. Tombol untuk tambah pilihan.
- d. *Text field* untuk mengisi jawaban yang benar.
- e. Tombol untuk menutup tampilan edit soal.
- f. Tombol untuk menyimpan soal tambahan.

13. Tampilan Edit Program Studi

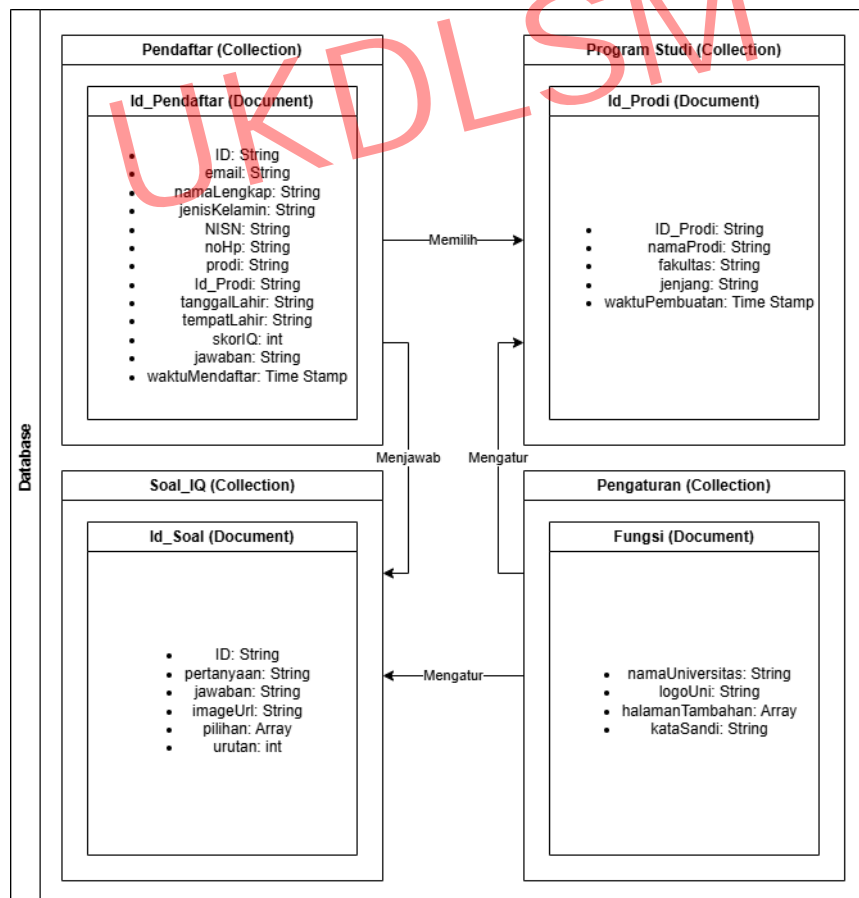
Gambar 4.19 Rancangan Tampilan Edit Program Studi

Keterangan:

- Text field* untuk mengisi nama prodi.
- Text field* untuk mengisi fakultas.
- Text field* untuk mengisi jenjang pendidikan.
- Text field* untuk mengisi deskripsi singkat.
- Tombol untuk menutup tampilan edit program studi.
- Tombol untuk menyimpan program studi.

4.2.3 Rancangan Struktur Basis Data

Aplikasi seleksi *On-The-Spot* PMB Unika De La Salle menggunakan *Firestore* sebagai basis data utama untuk menyimpan dan mengolah data. *Firestore* merupakan basis data *NoSQL* karena itu struktur basis data akan digambarkan menggunakan *Collection-Document Diagram* (CDD). Berikut ini adalah gambaran struktur basis data yang digunakan untuk mengembangkan aplikasi *On-The-Spot* PMB Unika De La Salle.



Gambar 4.20 Document Collection Diagram

4.3 Implementasi

Pada tahap ini akan dilakukan implementasi arsitektur sistem dan tampilan antarmuka aplikasi. Dalam Kanban, tahapan ini dimulai dari mengatur alur algoritma aplikasi, membuat daftar fitur yang diperlukan, dan membuat kerangka struktur aplikasi (*To-Do*), mengembangkan aplikasi sesuai dengan daftar fitur, arsitektur sistem dan rancangan antarmuka yang telah dibuat di tahap perancangan (*In Progress*), hingga menghasilkan aplikasi berbasis web yang dapat membantu proses seleksi *On-The-Spot* PMB di Unika De La Salle Manado (*Done*). Tahapan ini akan mencakup lingkungan implementasi, aturan implementasi, implementasi antarmuka aplikasi, implementasi basisdata, dan implementasi modul program.

4.3.1 Lingkungan Implementasi

Berikut ini merupakan lingkungan implementasi untuk pengembangan aplikasi seleksi *On-The-Spot* PMB Unika De La Salle Manado.

Tabel 4.2 Tabel Lingkungan Implementasi

No	Perangkat Lunak	
1.	Visual Studio Code (V.1.123)	<i>Compiler</i> dan <i>code editor</i> utama yang digunakan untuk mengembangkan aplikasi seleksi <i>On-The-Spot</i> PMB Unika De La Salle Manado.
2.	<i>Firebase</i> (V.12.14.0)	Basis data utama yang digunakan untuk menyimpan data dan informasi, dan untuk <i>hosting</i> aplikasi.
3.	Google Chrome (V.149.0.7827.102)	<i>Browser</i> utama yang digunakan untuk demonstrasi dan pengujian aplikasi yang dikembangkan.
4.	Figma (V.125.4.9)	Aplikasi <i>editor</i> yang digunakan untuk merancang antarmuka aplikasi.
No.	Perangkat Keras	
1.	ROG Zephyrus G15	<i>Processor</i> : AMD Ryzen 7 4800HS RAM: 16 GB ROM: 1.4 TB <i>Operating System</i> : Windows 11

4.3.2 Aturan Implementasi

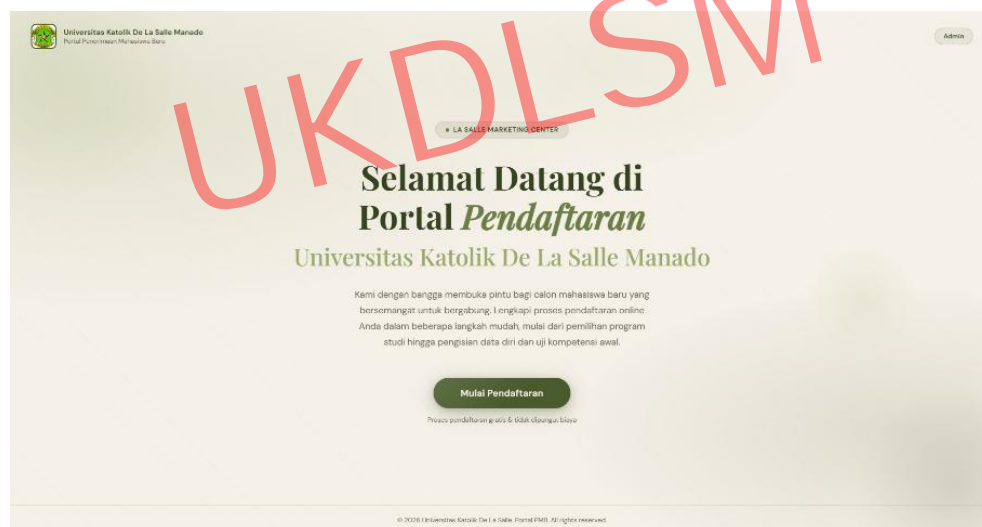
Berikut ini merupakan aturan implementasi yang telah ditentukan agar aplikasi dapat berjalan dengan baik:

1. Perangkat pengguna harus terhubung dengan jaringan Internet yang stabil.
2. *Browser* pengguna harus menggunakan versi terbaru.

4.3.3 Implementasi Antarmuka Aplikasi

Berikut ini merupakan implementasi antarmuka dari aplikasi seleksi *On-The-Spot* Unika De La Salle yang terdiri dari 8 halaman dan 5 tampilan *pop-up*: Halaman *Landing*, Halaman Seleksi, Halaman *Dashboard*, Halaman Data Pendaftar, Halaman Manajemen Halaman, Halaman Manajemen Pertanyaan, Halaman Manajemen Program Studi, dan Halaman Pengaturan; Tampilan *Login Admin*, Tampilan Detail Pendaftar, Tampilan Edit Halaman, Tampilan Edit Soal, Tampilan Edit Program Studi.

1. Halaman *Landing*



Gambar 4.21 Implementasi Halaman *Landing*

Gambar 4.21 adalah implementasi tampilan antarmuka untuk halaman *landing page*, di mana halaman ini adalah halaman pertama yang akan tampil ketika pengguna mengakses situs web. Di halaman ini terdapat kata sambutan dan tombol untuk melakukan pendaftaran. Adapun tombol di bagian kanan atas untuk melakukan *login* ke halaman admin.

2. Halaman Seleksi

Universitas Katolik De La Salle Manado
PORTAL PENDAFTARAN MAHASISWA BARU

1 Pilih Program Studi 2 Data Diri 3 Uji Kompetensi

LANGKAH 1
Pilih Program Studi

Silakan pilih program studi pilihan pertama dan kedua. Pilihan kedua akan digunakan sebagai cadangan apabila pilihan pertama tidak tersedia.

Pilihan Pertama *

— Pilih Program Studi —

Pilihan Kedua *

— Pilih Program Studi —

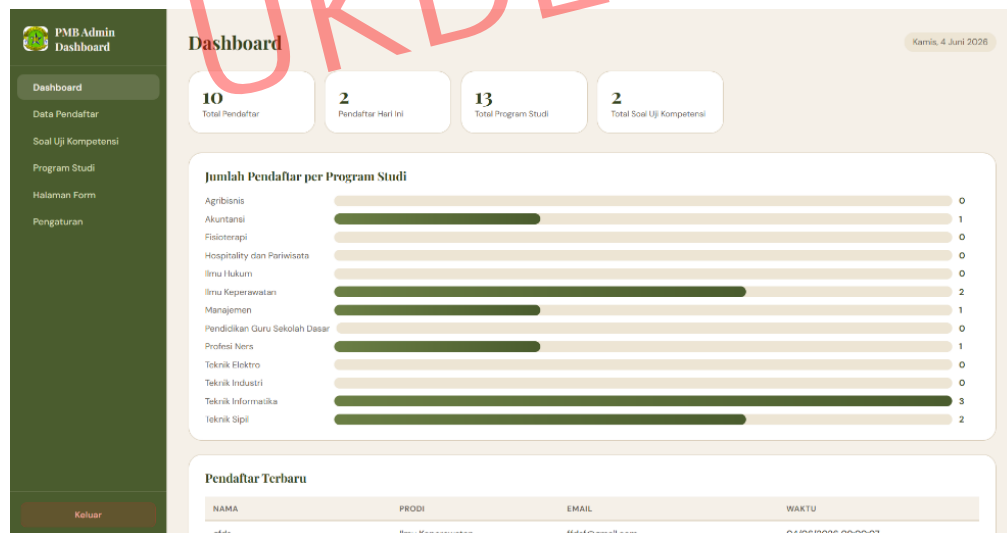
Lanjut →

© 2026 Universitas Katolik De La Salle, Portal PMB. All rights reserved.
Made by JTT

Gambar 4.22 Implementasi Halaman Seleksi

Gambar 4.22 adalah implementasi tampilan antarmuka untuk halaman seleksi. Halaman ini terdiri dari beberapa bagian sesuai dengan ketentuan dari admin. Tiga bagian utama dalam halaman ini adalah bagian pemilihan prodi, bagian biodata pendaftar, dan bagian uji kompetensi.

3. Halaman Dashboard

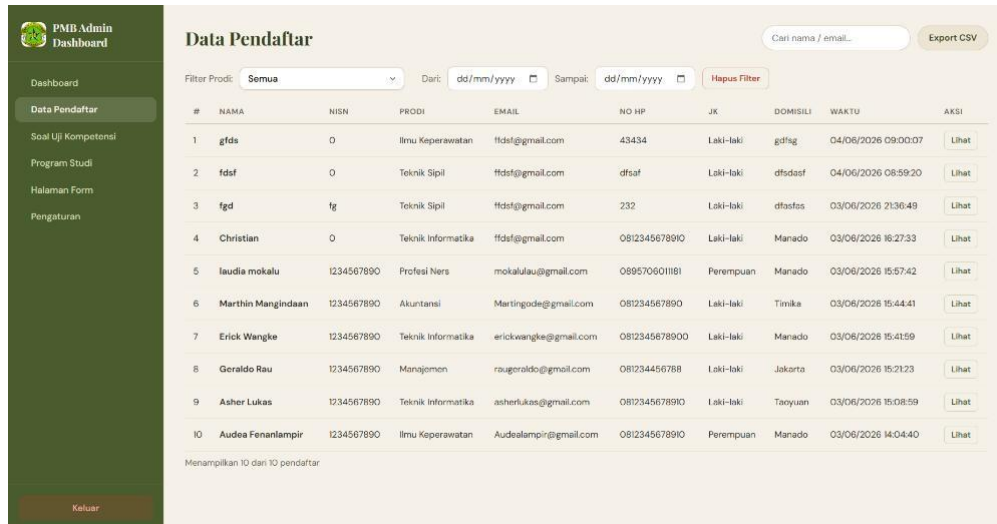


Gambar 4.23 Implementasi Halaman Dashboard

Berikut pada Gambar 4.23 adalah implementasi tampilan antarmuka untuk halaman *dashboard*. Halaman ini berisikan informasi dasar tentang proses penerimaan mahasiswa baru. Di bagian atas terdapat 4 *card view* yang menampilkan total pendaftar, pendaftar hari ini, total program studi yang terdaftar,

dan total soal uji kompetensi yang terdaftar. Adapun *card view* untuk menampilkan jumlah pendaftar per program studi dan data pendaftar yang baru saja mendaftar.

4. Halaman Data Pendaftar

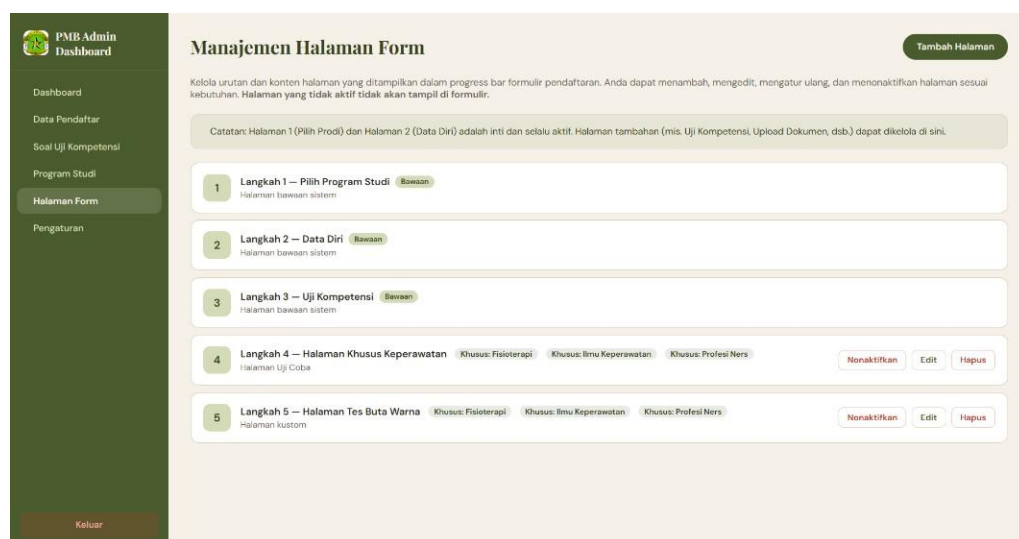


#	NAMA	NISN	PRODI	EMAIL	NO HP	JK	DOMISILI	WAKTU	AKSI
1	gfds	0	Ilmu Keperawatan	ffdfst@gmail.com	43434	Laki-laki	gdffg	04/06/2026 09:00:07	Lihat
2	fdsf	0	Teknik Sipil	ffdfst@gmail.com	dfsf	Laki-laki	dfsdasf	04/06/2026 08:59:20	Lihat
3	fgd	fg	Teknik Sipil	ffdfst@gmail.com	232	Laki-laki	dfsdafs	03/06/2026 21:30:49	Lihat
4	Christian	0	Teknik Informatika	ffdfst@gmail.com	081234567890	Laki-laki	Manado	03/06/2026 16:27:33	Lihat
5	laudia mokalu	1234567890	Profesi Ners	mokalulau@gmail.com	089570601181	Perempuan	Manado	03/06/2026 15:57:42	Lihat
6	Marthin Mangindaan	1234567890	Akuntansi	Martingode@gmail.com	081234567890	Laki-laki	Timika	03/06/2026 15:44:41	Lihat
7	Erick Wangke	1234567890	Teknik Informatika	erickwangke@gmail.com	081234567890	Laki-laki	Manado	03/06/2026 15:41:59	Lihat
8	Geraldo Rau	1234567890	Manajemen	raugeraldo@gmail.com	081234456788	Laki-laki	Jakarta	03/06/2026 15:21:23	Lihat
9	Asher Lukas	1234567890	Teknik Informatika	asherlukas@gmail.com	081234567890	Laki-laki	Taoyuan	03/06/2026 15:08:59	Lihat
10	Audea Fenanlampir	1234567890	Ilmu Keperawatan	Audealampir@gmail.com	081234567890	Perempuan	Manado	03/06/2026 14:04:40	Lihat

Gambar 4.24 Implementasi Halaman Data Pendaftar

Gambar 4.24 adalah implementasi tampilan antarmuka untuk halaman data pendaftar. Halaman ini berisikan informasi para pendaftar lengkap dengan filter dan fungsi pencarian. Adapun tombol untuk ekspor laporan dalam bentuk CSV yang terdapat pada halaman ini.

5. Halaman Manajemen Halaman

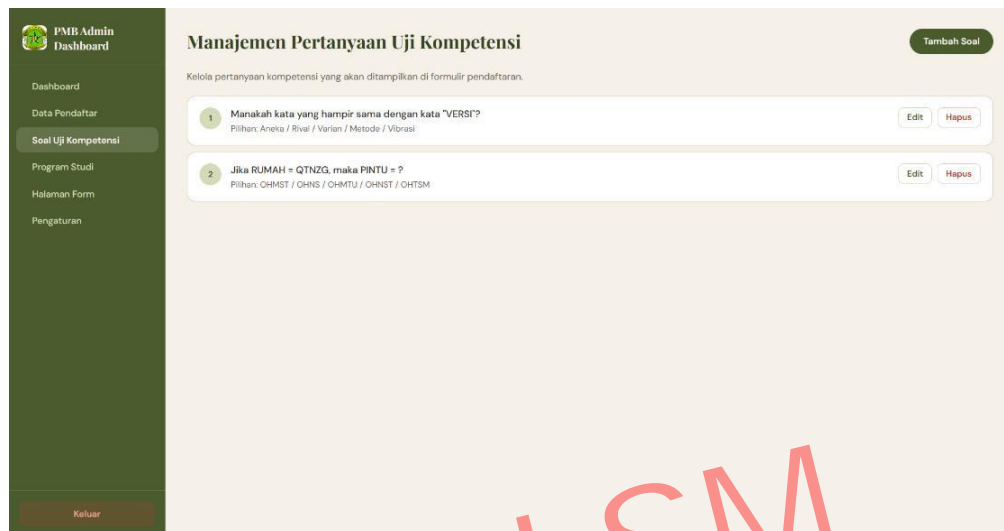


Langkah	Nama Langkah	Status	Aksi
1	Langkah 1 - Pilih Program Studi	Bawaan	
2	Langkah 2 - Data Diri	Bawaan	
3	Langkah 3 - Uji Kompetensi	Bawaan	
4	Langkah 4 - Halaman Khusus Keperawatan	Khusus: Fisioterapi, Khusus: Ilmu Keperawatan, Khusus: Profesi Ners	Nonaktifkan, Edit, Hapus
5	Langkah 5 - Halaman Tes Buta Warna	Khusus: Fisioterapi, Khusus: Ilmu Keperawatan, Khusus: Profesi Ners	Nonaktifkan, Edit, Hapus

Gambar 4.25 Implementasi Halaman Manajemen Halaman

Gambar 4.25 adalah implementasi tampilan antarmuka untuk halaman manajemen halaman. Halaman ini berisikan informasi tentang halaman tambahan. Admin dapat menambahkan halaman baru, mengatur halaman yang sudah ada, ataupun menghapus halaman yang tidak lagi relevan.

6. Halaman Manajemen Pertanyaan



Gambar 4.26 Implementasi Halaman Manajemen Pertanyaan

Berikut pada Gambar 4.26 adalah implementasi tampilan antarmuka untuk halaman manajemen pertanyaan. Halaman ini berisikan informasi tentang pertanyaan yang ditampilkan pada halaman uji kompetensi. Admin dapat menambahkan pertanyaan baru, mengatur pertanyaan yang sudah ada, ataupun menghapus pertanyaan yang tidak lagi relevan.

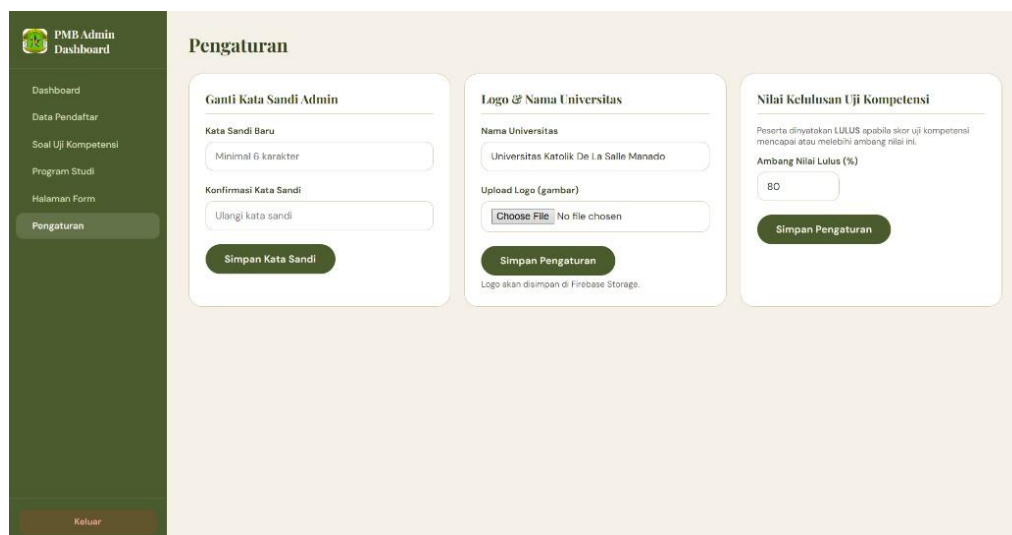
7. Halaman Manajemen Program Studi



Gambar 4.27 Implementasi Halaman Manajemen Prodi

Berikut pada Gambar 4.27 adalah implementasi tampilan antarmuka untuk halaman manajemen program studi. Halaman ini berisikan informasi tentang program studi yang terdaftar dalam Aplikasi PMB Unika De La Salle. Admin dapat menambahkan prodi baru, mengatur prodi yang sudah terdaftar sebelumnya, ataupun menghapus prodi yang tidak lagi aktif.

8. Halaman Pengaturan



Gambar 4.28 Implementasi Halaman Pengaturan

Gambar 4.28 adalah implementasi tampilan antarmuka untuk halaman pengaturan. Di halaman ini admin dapat mengubah informasi seputar aplikasi PMB.

Seperti kata sandi login admin, nama universitas, logo universitas, atau nilai standar kelulusan calon pendaftar.

9. Tampilan *Login Admin*



Gambar 4.29 Implementasi Tampilan *Login Admin*

Gambar 4.29 adalah implementasi antarmuka untuk tampilan *login admin*. Di tampilan ini, admin harus mengisi kata sandi untuk dapat mengakses halaman admin.

10. Tampilan Detail Pendaftar

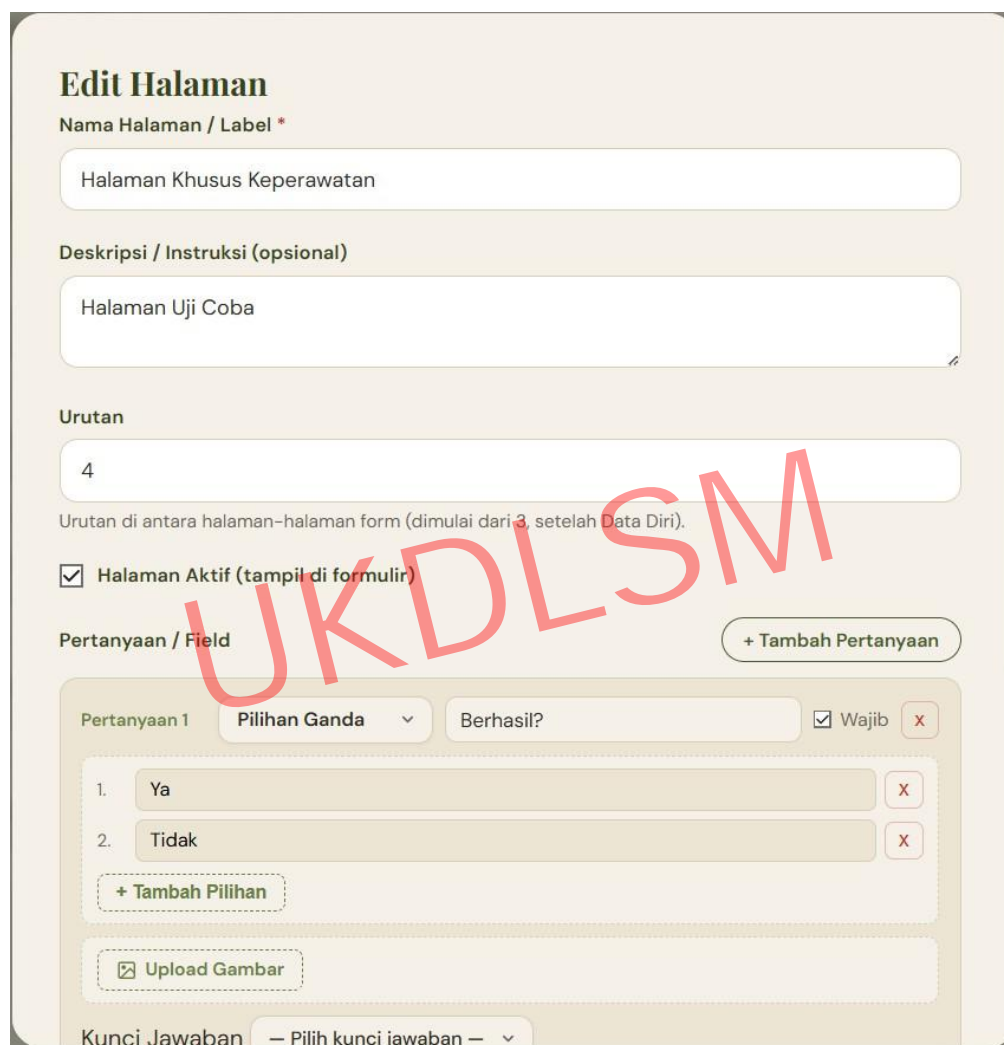


Detail Pendaftar	
NAMA LENGKAP gfds	TEMPAT, TANGGAL LAHIR gdfsg, 43443-03-04
JENIS KELAMIN Laki-laki	EMAIL ffdsf@gmail.com
NO HP 43434	DOMISILI gdfsg
PROGRAM STUDI Ilmu Keperawatan	NO. PENDAFTARAN PMB-0021-J9ER4
WAKTU DAFTAR 04/06/2026 09:00:07	NISN 0
NIK -	SKOR UJI KOMPETENSI 2/2 benar (100%)
STATUS KELULUSAN LULUS	
HALAMAN TES BUTA WARNA	
JAWAB APA SAJA	
Apa saja	
SEBUTKAN ANGKA PADA GAMBAR	
2	
HALAMAN KHUSUS KEPERAWATAN	
BERHASIL?	
Ya	

Gambar 4.30 Impelemntasi Tampilan Detail Pendaftar

Gambar 4.30 adalah implementasi antarmuka untuk tampilan detail pendaftar. Di tampilan ini admin dapat melihat informasi tentang pendaftar, baik itu biodata mereka, skor hasil uji kompetensi, sampai respon terhadap pertanyaan dalam bentuk paragraf.

11. Tampilan Edit Halaman



Edit Halaman

Nama Halaman / Label *

Halaman Khusus Keperawatan

Deskripsi / Instruksi (opsional)

Halaman Uji Coba

Urutan

4

Urutan di antara halaman-halaman form (dimulai dari 3, setelah Data Diri).

Halaman Aktif (tampil di formulir)

Pertanyaan / Field + Tambah Pertanyaan

Pertanyaan 1 Pilihan Ganda Berhasil? Wajib X

1. Ya X

2. Tidak X

+ Tambah Pilihan

Upload Gambar

Kunci Jawaban — Pilih kunci jawaban —

Gambar 4.31 Implementasi Tampilan Edit Halaman

Gambar 4.31 adalah implementasi antarmuka untuk tampilan edit halaman. Di tampilan ini admin dapat melihat informasi tentang halaman seleksi tambahan, mulai dari nama halaman, deskripsi, urutan penampilan halaman, pertanyaan yang terdaftar, sampai pilihan prodi khusus.

12. Tampilan Edit Soal

Edit Soal

Isi pertanyaan dan pilihan jawaban di bawah ini.

Pertanyaan *

Manakah kata yang hampir sama dengan kata "VERSI"?

Pilihan Jawaban

Aneka

Rival

Varian

Metode

Vibrasi

+ Tambah Pilihan

Jawaban Benar (opsional, untuk skoring)

Varian

Upload Gambar (opsional)

Choose File No file chosen

Gambar akan disimpan di Firebase Storage.

Gambar 4.32 Implementasi Halaman Edit Soal

Gambar 4.32 adalah implementasi antarmuka untuk tampilan edit soal. Di tampilan ini admin dapat melihat informasi tentang soal yang terdaftar di halaman uji kompetensi, mulai dari pertanyaan, opsi pilihan, kunci jawaban, juga pilihan untuk mengunggah gambar untuk mendukung pertanyaan.

13. Tampilan Edit Program Studi

Edit Program Studi

Nama Prodi *

Agribisnis

Fakultas

Fakultas Pertanian

Jenjang

S1

Deskripsi Singkat

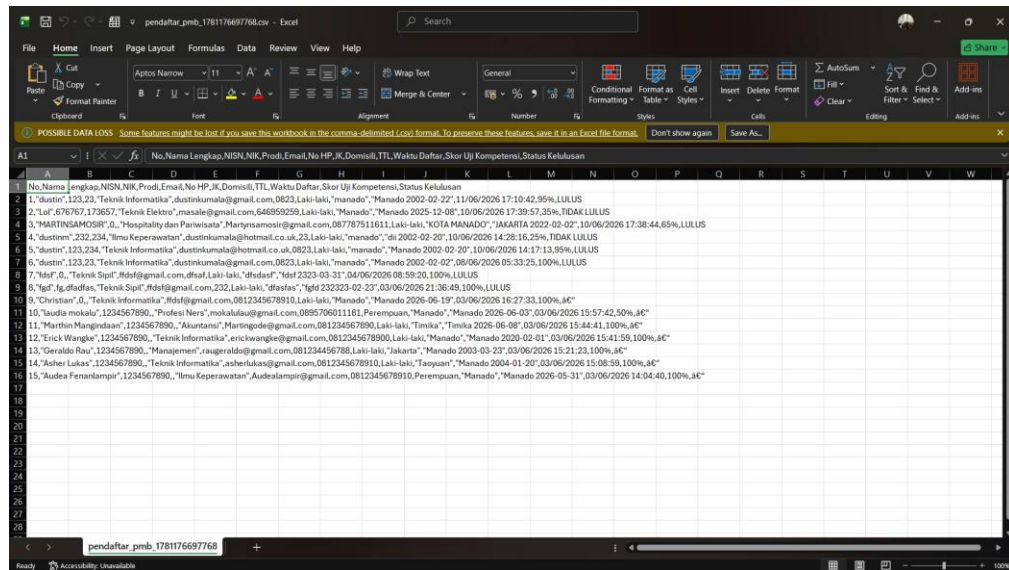
Deskripsi singkat prodi...

Batal Simpan

Gambar 4.33 Implementasi Tampilan Edit Program Studi

Berikut pada Gambar 4.33 adalah implementasi antarmuka untuk tampilan edit program studi. Di tampilan ini admin dapat melihat informasi tentang program studi yang terdaftar di halaman program studi, mulai dari nama program studi, fakultas, jenjang pendidikan, sampai deskripsi singkat program studi.

14. Tampilan Hasil Ekspor CSV



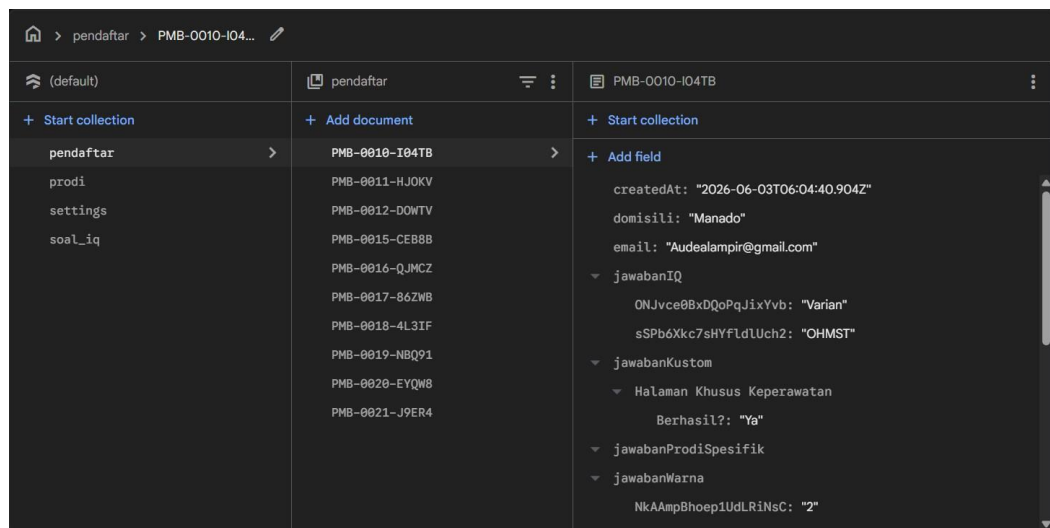
No	Nama Lengkap	NISN	NIK	Prodi	Email	No HP	JK	Domisili	TTL	Waktu Daftar	Skor Uji Kompetensi	Status Kelulusan
1	"dustin"	123.23	"Teknik Informatika"	dustinkumala@gmail.com	0823	Laki-laki	"Manado"	"Manado 2002-02-22"	11/06/2026	17:10:42.95%	LULUS	
2	"Lor"	676767.173657	"Teknik Elektro"	masale@gmail.com	646959259	Laki-laki	"Manado"	"Manado 2025-12-08"	10/06/2026	17:39:57.35%	TIDAK LULUS	
3	"MARTINSA MOSIR"	"Hospitality dan Pariwisata"	Maryamsas@gmail.com	087787911611	Laki-laki	"KOTA MANADO"	"JAKARTA 2022-02-02"	10/06/2026	17:38:44.85%	LULUS		
4	"dustin"	232.234	"Ilmu Keperawatan"	dustinkumala@hotmail.co.uk	23	Laki-laki	"manado"	"di 2002-02-20"	10/06/2026	14:28:16.25%	TIDAK LULUS	
5	"dustin"	123.234	"Teknik Informatika"	dustinkumala@hotmail.co.uk	0823	Laki-laki	"manado"	"Manado 2002-02-20"	10/06/2026	14:17:13.95%	LULUS	
6	"dustin"	123.23	"Teknik Informatika"	dustinkumala@gmail.com	0823	Laki-laki	"manado"	"Manado 2002-02-02"	08/06/2026	05:33:25.100%	LULUS	
7	"dustin"	"Teknik Sipil"	hsd@gmail.com	hsd@gmail.com	hsd	Laki-laki	"dhdasd"	"hsd 2323 03-31"	04/06/2026	08:59:20.100%	LULUS	
8	"dustin"	123.234	"Teknik Sipil"	hsd@gmail.com	232	Laki-laki	"hsdhsd"	"hsd 2323 03-31"	03/06/2026	21:38:49.100%	LULUS	
9	"Christan"	"Teknik Informatika"	hsd@gmail.com	0812345678910	Laki-laki	"Manado"	"Manado 2026-06-19"	03/06/2026	16:27:33.100%	ac"		
10	"Audea mokalu"	1234567890	"Profesi Ners"	mokalu@gmail.com	089570601181	Perempuan	"Manado"	"Manado 2026-06-03"	03/06/2026	15:57:42.50%	ac"	
11	"Martin Mangindaan"	1234567890	"Akuntansi"	Marstingode@gmail.com	081234567890	Laki-laki	"Timika"	"Timika 2026-06-08"	03/06/2026	15:44:41.100%	ac"	
12	"Trick Wangke"	1234567890	"Teknik Informatika"	erickwangke@gmail.com	0812345678900	Laki-laki	"Manado"	"Manado 2020-02-01"	03/06/2026	15:41:56.100%	ac"	
13	"Ovado Riu"	1234567890	"Manajemen"	ruvada@gmail.com	081234567890	Laki-laki	"Jakarta"	"Manado 2003-03-23"	03/06/2026	15:21:23.100%	ac"	
14	"Asher Lukas"	1234567890	"Teknik Informatika"	asherlukas@gmail.com	0812345678910	Laki-laki	"Tayayan"	"Manado 2004-01-20"	03/06/2026	15:08:59.100%	ac"	
15	"Audea Fenarampir"	1234567890	"Ilmu Keperawatan"	Audealampir@gmail.com	0812345678910	Perempuan	"Manado"	"Manado 2026-05-31"	03/06/2026	14:04:40.100%	ac"	

Gambar 4.34 Tampilan Hasil Ekspor CSV

Berikut pada Gambar 4.34 adalah contoh hasil ekspor data calon mahasiswa dengan format CSV.

4.3.4 Implementasi Basis Data

Berikut ini merupakan contoh bentuk implementasi *Firestore Database* sebagai basis data utama aplikasi seleksi *On-The-Spot* PMB Unika De La Salle Manado. Fitur utama *Firestore* yang kita gunakan adalah *Firestore* untuk menyimpan data berupa *string*, *integer*, dan *array*.



Field	Value
prodi	"PMB-0011-HJOKV"
settings	"PMB-0012-DOWTV"
soal_iq	"PMB-0015-CE88B"
createdAt	"2026-06-03T06:04:40.904Z"
domisili	"Manado"
email	"Audealampir@gmail.com"
jawabanIQ	["ONJvce0BxDQoPqJixYvb: \"Varian\"", "sSPb6Xkc7sHYfLdLUch2: \"OHMST\""]
jawabanKustom	["Halaman Khusus Keperawatan", "Berhasil?: \"Ya\""]
jawabanProdiSpesifik	"NkAAmpBhoep1UdLRiNsC: \"2\""
jawabanWarna	"2"

Gambar 4.35 Implementasi Basis Data

4.3.5 Implementasi Modul Program

Terdapat tiga modul utama dalam aplikasi sistem seleksi *On-The-Spot* PMB Unika De La Salle Manado, yaitu *Landing Page*, *Main App*, *Admin Dashboard*. Kode program dapat dilihat di Lampiran A.

4.4 Pengujian

Pada tahap ini akan dilakukan pengujian untuk aplikasi seleksi *On-The-Spot* PMB yang telah dikembangkan. Dalam Kanban, tahapan ini dimulai dari identifikasi interaksi antar pengguna dan sistem dan malfungsi yang mungkin terjadi (*To-Do*), melakukan pengujian menggunakan metode *Black Box Testing* untuk memastikan setiap fungsi dalam aplikasi dapat berjalan dengan baik dan *User Acceptance Test* untuk mendapatkan *feedback* dari calon pengguna (*In Progress*), hingga menghasilkan laporan analisis hasil pengujian (*Done*). Tahapan ini akan mencakup tujuan pengujian, kriteria pengujian, *Black Box Testing*, *User Acceptance Test*, dan analisis hasil pengujian.

4.4.1 Tujuan Pengujian

Proses pengujian dilakukan untuk memastikan aplikasi yang dikembangkan dapat berfungsi sesuai dengan spesifikasi dan kebutuhan yang telah ditentukan sebelumnya. *Black Box Testing* digunakan untuk menguji fungsionalitas aplikasi sehingga dapat mencegah masalah sedini mungkin. Selain itu, pengujian menggunakan metode *User Acceptance Test* juga digunakan untuk mendapatkan *feedback* terkait pengalaman penggunaan aplikasi dari sudut pandang pengguna akhir.

4.4.2 Kriteria Pengujian

Kriteria pengujian merupakan nilai-nilai dasar yang harus dipenuhi oleh aplikasi yang dikembangkan. Terdapat beberapa kriteria pengujian yang telah ditetapkan, yaitu sebagai berikut:

1. Aplikasi dapat diakses dan berjalan dengan baik menggunakan *browser* yang mendukung standar ES6+.
2. Aplikasi berhasil terhubung dengan *Firebase*.
3. Aplikasi dapat berinteraksi secara interaktif dengan pendaftar.

4. Aplikasi dapat menyimpan data pendaftaran dari calon mahasiswa.
5. Proses CRUD (*Create, Read, Update, Delete*) untuk bagian Program Studi, Pertanyaan, dan Halaman Seleksi Tambahan, dapat bekerja dengan baik.
6. Aplikasi dapat menampilkan data pendaftar sesuai dengan filter atau kriteria pencarian yang ditentukan.
7. Pengujian akan dilakukan melalui link <https://pmb-delasalle.web.app>.

4.4.3 *Black Box Testing*

Pengujian aplikasi menggunakan metode *Black Box Testing* dilakukan untuk memastikan aplikasi seleksi *On-The-Spot* PMB Unika De La Salle dapat berjalan dengan baik dan sesuai dengan kebutuhan dan harapan pengguna. Pengujian ini berfokus pada aspek fungsionalitas aplikasi tanpa mempertimbangkan kode program atau logika sistem. Tujuannya yaitu untuk mengevaluasi apakah fitur-fitur yang dikembangkan dapat berjalan sesuai dengan harapan dan fungsinya.

Melalui metode ini, setiap fitur utama akan diuji dari sudut pandang pengguna, dimana penguji akan memberikan masukan dan mengamati hasil atau respon yang diberikan oleh aplikasi. Pengujian akan dilakukan menggunakan Hp Pavilion Laptop 14 dengan spesifikasi sebagai berikut:

Tabel 4.2 Spesifikasi Perangkat Pengujian

No	Nama Kriteria	Spesifikasi Perangkat Pengujian
1.	<i>Operating System</i>	Windows 11
2.	<i>Processor</i>	11 th Gen Intel Core i7
3.	RAM	16 GB
4.	ROM	512 GB NVMe SSD
5.	<i>Network</i>	Telkomsel 5G

Berikut ini adalah hasil dari pengujian yang dilakukan menggunakan metode *Black Box Testing*:

Tabel 4.3 Hasil Pengujian *Black Box Testing*

No	Aktivitas Pengujian	Hasil yang Diharapkan	Hasil Pengujian	Kesimpulan
1.	Menjalankan aplikasi menggunakan <i>browser</i> Opera One.	Aplikasi dapat dimuat dan <i>landing page</i> ditampilkan.	Aplikasi berhasil dimuat dan <i>landing page</i> berhasil ditampilkan.	<input checked="" type="checkbox"/> Diterima <input type="checkbox"/> Ditolak
2.	Melakukan proses pendaftaran calon mahasiswa.	Aplikasi dapat menampilkan setiap halaman pendaftaran, mulai dari pemilihan program studi sampai halaman tambahan dan menyimpan data pendaftar di <i>Firebase</i> .	Aplikasi berhasil menampilkan setiap halaman pendaftaran, mulai dari pemilihan program studi sampai halaman tambahan dan menyimpan data pendaftar di <i>Firebase</i> .	<input checked="" type="checkbox"/> Diterima <input type="checkbox"/> Ditolak
3.	Mengakses halaman admin.	Aplikasi dapat menampilkan tampilan <i>login</i> admin dan menampilkan halaman <i>dashboard</i> .	Aplikasi berhasil menampilkan tampilan <i>login</i> admin dan menampilkan halaman <i>dashboard</i> .	<input checked="" type="checkbox"/> Diterima <input type="checkbox"/> Ditolak

No	Aktivitas Pengujian	Hasil yang Diharapkan	Hasil Pengujian	Kesimpulan
4.	Mengakses halaman data pendaftar.	<p>Aplikasi dapat menampilkan halaman data pendaftar beserta data pendaftar sesuai dengan filter atau hasil pencarian yang ditentukan.</p> <p>Aplikasi juga harus dapat mengekspor data pendaftar dalam format CSV.</p>	<p>Aplikasi berhasil menampilkan halaman data pendaftar beserta data pendaftar sesuai dengan filter atau hasil pencarian yang ditentukan.</p> <p>Aplikasi berhasil mengekspor data pendaftar dalam format CSV.</p>	<p><input checked="" type="checkbox"/> Diterima <input type="checkbox"/> Ditolak</p>
5.	Memasukkan data tidak valid di kolom NISN atau NIK.	Aplikasi akan memberikan peringatan bahwa data yang dimasukkan tidak valid.	Aplikasi memberikan peringatan bahwa data yang dimasukkan tidak valid.	<p><input checked="" type="checkbox"/> Diterima <input type="checkbox"/> Ditolak</p>
6.	Memasukkan tanggal lahir tidak valid (usia pendaftar < 15 tahun).	Aplikasi akan memberikan peringatan bahwa data yang	Aplikasi memberikan peringatan bahwa data yang	<p><input checked="" type="checkbox"/> Diterima <input type="checkbox"/> Ditolak</p>

No	Aktivitas Pengujian	Hasil yang Diharapkan	Hasil Pengujian	Kesimpulan
		dimasukan tidak valid dan bahwa usia pendaftar harus lebih dari 15 tahun.	dimasukan tidak valid dan bahwa usia pendaftar harus lebih dari 15 tahun.	
7.	Mengakses halaman manajemen soal.	Aplikasi dapat menampilkan halaman manajemen soal dan menampilkan pertanyaan yang terdaftar.	Aplikasi berhasil menampilkan halaman manajemen soal dan menampilkan pertanyaan yang terdaftar.	<input checked="" type="checkbox"/> Diterima <input type="checkbox"/> Ditolak
8.	Menampilkan Soal Pengujian	Aplikasi dapat menampilkan soal pengujian sesuai dengan program studi yang telah dipilih oleh calon mahasiswa.	Aplikasi dapat menampilkan soal pengujian sesuai dengan program studi yang telah dipilih oleh calon mahasiswa.	<input checked="" type="checkbox"/> Diterima <input type="checkbox"/> Ditolak
9.	Menjawab Soal Pengujian	Aplikasi dapat menyimpan jawaban yang dimasukkan oleh pengguna untuk setiap	Aplikasi berhasil menyimpan jawaban yang dimasukkan oleh pengguna	<input checked="" type="checkbox"/> Diterima <input type="checkbox"/> Ditolak

No	Aktivitas Pengujian	Hasil yang Diharapkan	Hasil Pengujian	Kesimpulan
		pertanyaan yang ditampilkan.	untuk setiap pertanyaan yang ditampilkan.	
10.	Mengirimkan Pendaftaran	Aplikasi dapat mencatat pendaftaran calon mahasiswa termasuk biodata, hasil seleksi, dan status kelulusan.	Aplikasi berhasil mencatat pendaftaran calon mahasiswa termasuk biodata, hasil seleksi, dan status kelulusan.	<input checked="" type="checkbox"/> Diterima <input type="checkbox"/> Ditolak
11.	Melakukan Proses <i>Login Admin</i>	Aplikasi dapat memastikan kata sandi yang dimasukkan oleh pengguna sesuai dengan kata sandi yang telah ditetapkan sebelum memberikan akses halaman admin.	Aplikasi mampu memastikan kata sandi yang dimasukkan oleh pengguna sesuai dengan kata sandi yang telah ditetapkan sebelum memberikan akses halaman admin.	<input checked="" type="checkbox"/> Diterima <input type="checkbox"/> Ditolak

No	Aktivitas Pengujian	Hasil yang Diharapkan	Hasil Pengujian	Kesimpulan
12.	Tambah Soal	Aplikasi dapat menampilkan tampilan tambah soal dan menyimpan soal baru di <i>database</i> .	Aplikasi mampu menampilkan tampilan tambah soal dan menyimpan soal baru di <i>database</i> .	<input checked="" type="checkbox"/> Diterima <input type="checkbox"/> Ditolak
13.	Ubah Soal	Aplikasi dapat menampilkan tampilan ubah soal dan menyimpan perubahan soal di <i>database</i> .	Aplikasi dapat menampilkan tampilan ubah soal dan menyimpan perubahan soal di <i>database</i> .	<input checked="" type="checkbox"/> Diterima <input type="checkbox"/> Ditolak
14.	Hapus Soal	Aplikasi dapat menghapus soal yang dipilih oleh admin dari <i>database</i> .	Aplikasi berhasil menghapus soal yang dipilih oleh admin dari <i>database</i> .	<input checked="" type="checkbox"/> Diterima <input type="checkbox"/> Ditolak
15.	Mengkases halaman manajemen program studi.	Aplikasi dapat menampilkan halaman manajemen program studi	Aplikasi berhasil menampilkan halaman manajemen program studi	<input checked="" type="checkbox"/> Diterima <input type="checkbox"/> Ditolak

No	Aktivitas Pengujian	Hasil yang Diharapkan	Hasil Pengujian	Kesimpulan
		dan menampilkan daftar program studi yang terdaftar.	dan menampilkan daftar program studi yang terdaftar.	
16.	Menampilkan daftar program studi yang terdaftar.	Aplikasi dapat menampilkan daftar program studi yang terdaftar dalam <i>database</i> .	Aplikasi berhasil menampilkan daftar program studi yang terdaftar dalam <i>database</i> .	[x] Diterima [] Ditolak
17.	Tambah program studi	Aplikasi dapat menampilkan tampilan tambah program studi dan menyimpan program studi baru di <i>database</i> .	Aplikasi berhasil menampilkan tampilan tambah program studi dan menyimpan program studi baru di <i>database</i> .	[x] Diterima [] Ditolak
18.	Ubah program studi	Aplikasi dapat menampilkan tampilan ubah program studi dan menyimpan	Aplikasi berhasil menampilkan tampilan ubah program studi dan	[x] Diterima [] Ditolak

No	Aktivitas Pengujian	Hasil yang Diharapkan	Hasil Pengujian	Kesimpulan
		perubahan program studi di <i>database</i> .	menyimpan perubahan program studi di <i>database</i> .	
19.	Hapus program studi	Aplikasi dapat menghapus program studi terdaftar yang dipilih oleh admin dari <i>database</i> .	Aplikasi berhasil menghapus program studi terdaftar yang dipilih oleh admin dari <i>database</i> .	[x] Diterima [] Ditolak
20.	Mengakses halaman manajemen halaman.	Aplikasi dapat menampilkan halaman manajemen halaman dan menampilkan daftar halaman seleksi tambahan yang terdaftar.	Aplikasi berhasil menampilkan halaman manajemen halaman dan menampilkan daftar halaman seleksi tambahan yang terdaftar.	[x] Diterima [] Ditolak
21.	Tambah halaman seleksi tambahan	Aplikasi dapat menampilkan tampilan tambah halaman seleksi	Aplikasi berhasil menampilkan tampilan tambah halaman	[x] Diterima [] Ditolak

No	Aktivitas Pengujian	Hasil yang Diharapkan	Hasil Pengujian	Kesimpulan
		<p>tambahan dan menyimpan halaman seleksi tambahan baru di <i>database</i>.</p>	<p>seleksi tambahan dan menyimpan halaman seleksi tambahan baru di <i>database</i>.</p>	
22.	<p>Ubah halaman seleksi tambahan</p>	<p>Aplikasi dapat menampilkan tampilan ubah halaman seleksi tambahan dan menyimpan perubahan halaman seleksi tambahan di <i>database</i>.</p>	<p>Aplikasi berhasil menampilkan tampilan ubah halaman seleksi tambahan dan menyimpan perubahan halaman seleksi tambahan di <i>database</i>.</p>	<p><input checked="" type="checkbox"/> Diterima <input type="checkbox"/> Ditolak</p>
23.	<p>Hapus halaman seleksi tambahan</p>	<p>Aplikasi dapat menghapus halaman seleksi tambahan yang dipilih oleh admin dari <i>database</i>.</p>	<p>Aplikasi berhasil menghapus halaman seleksi tambahan yang dipilih oleh admin dari <i>database</i>.</p>	<p><input checked="" type="checkbox"/> Diterima <input type="checkbox"/> Ditolak</p>

No	Aktivitas Pengujian	Hasil yang Diharapkan	Hasil Pengujian	Kesimpulan
24.	Mengubah kata sandi <i>login</i> admin	Aplikasi dapat mengganti kata sandi <i>login</i> admin sesuai masukan dari pengguna.	Aplikasi berhasil mengganti kata sandi <i>login</i> admin sesuai masukan dari pengguna.	<input checked="" type="checkbox"/> Diterima <input type="checkbox"/> Ditolak
25.	Mengubah nama atau logo universitas	Aplikasi dapat memperbarui nama atau logo universitas sesuai masukan dari pengguna.	Aplikasi berhasil memperbarui nama atau logo universitas sesuai masukan dari pengguna.	<input checked="" type="checkbox"/> Diterima <input type="checkbox"/> Ditolak
26.	Mengubah batas kelulusan uji kompetensi	Aplikasi dapat memperbarui nilai batas kelulusan uji kompetensi.	Aplikasi berhasil memperbarui nilai batas kelulusan uji kompetensi.	<input checked="" type="checkbox"/> Diterima <input type="checkbox"/> Ditolak

4.4.4 User Acceptance Test

Selain pengujian dengan *Black Box Testing*, pengujian menggunakan metode UAT juga dilakukan untuk mendapatkan respon dan tanggapan dari sudut pandang pengguna. Proses pengujian dilakukan dengan tiga orang responden sebagai calon mahasiswa dan satu orang responden sebagai staf. Hasil pengujian mendapatkan hasil yang positif dari setiap responden. Hasil UAT dapat dilihat di Lampiran B.

4.4.5 Analisis Hasil Pengujian

Berdasarkan hasil dari pengujian yang dilakukan, aplikasi seleksi *On-The-Spot* PMB berbasis web ini telah berhasil memenuhi standar untuk semua semua fungsi dalam aplikasi sesuai dengan skenario pengujian. Melalui pengujian menggunakan metode *Black Box Testing* dan *User Acceptance Test*, dapat kita ketahui bahwa fungsi-fungsi utama semua dapat berjalan dengan baik, mulai dari proses registrasi untuk calon mahasiswa, daftar pilihan program studi berhasil ditampilkan dengan baik, halaman biodata juga berhasil ditampilkan dengan baik, halaman seleksi pun berhasil ditampilkan sesuai dengan program studi pilihan calon mahasiswa. Di sisi staf LMC juga setiap fungsi aplikasi dapat berjalan dengan baik. Aplikasi juga mampu untuk mengidentifikasi *miss-input* dan memberi peringatan terhadap *miss-input*. Aplikasi juga berhasil untuk mengekspor laporan dalam bentuk CSV.

UKDLSM

BAB V

KESIMPULAN DAN SARAN

5.1 Kesimpulan

Berdasarkan hasil perancangan, pembangunan, dan pengujian aplikasi web untuk seleksi *On-The-Spot* Penerimaan Mahasiswa Baru (PMB) di Universitas Katolik De La Salle Manado, dapat ditarik beberapa kesimpulan sebagai berikut:

1. Aplikasi berbasis web telah berhasil dirancang untuk membantu proses seleksi *On-The-Spot* PMB Universitas Katolik De La Salle Manado, dengan menyediakan berbagai fitur lengkap seperti proses pendaftaran dan seleksi calon mahasiswa baru, yang mana dapat mempermudah calon mahasiswa untuk mendaftar dan staf LMC untuk melakukan seleksi. Adapun fitur lainnya seperti manajemen soal uji kompetensi, manajemen halaman seleksi, manajemen program studi, dan fitur ekspor data dengan format CSV, di mana dapat mempermudah staf LMC untuk mengelola data calon mahasiswa.
2. Hasil pengujian menggunakan metode *Black Box Testing* menunjukkan bahwa seluruh fungsionalitas aplikasi, mulai dari proses pendaftaran, manajemen data, operasi CRUD untuk soal, program studi, dan halaman seleksi, hingga ekspor data CSV, semuanya dapat berjalan dengan baik sesuai dengan harapan dan ketentuan diawal. Pengujian dengan metode *User Acceptance Test* (UAT) juga memberikan respon yang positif, dari tiga orang responden sebagai calon mahasiswa dan satu orang responden sebagai staf LMC. Mengindikasikan bahwa aplikasi telah memenuhi kebutuhan pengguna dan dapat diterima dengan oleh pengguna.

5.2 Saran

Berdasarkan hasil Kerja Praktik yang telah dilaksanakan, terdapat beberapa saran yang dapat dipertimbangkan untuk pengembangan aplikasi selanjutnya:

1. Menambahkan autentikasi yang lebih terstruktur untuk meningkatkan keamanan akses halaman admin dan melindungi data calon mahasiswa.
2. Menambahkan fitur ekspor data yang lebih beragam, tidak hanya format CSV tapi juga format lain seperti PDF atau Excel.

3. Menambahkan fitur pengingat otomatis untuk calon mahasiswa agar dapat membantu menyalurkan informasi terbaru kepada calon mahasiswa.
4. Menambahkan fitur analitik dan grafik visual untuk membantu visualisasi data.

UKDLSM

DAFTAR PUSTAKA

- [1] "Biro La Salle Marketing Center," De La Salle Catholic University Manado, [Online]. Available: <https://unikadelasalle.ac.id/lasalle-marketing-center/>. [Accessed 01 06 2026].
- [2] K. Beda, "Sistem Informasi Penerimaan Mahasiswa Baru," *JUPITER: Jurnal Penelitian Ilmu dan Teknologi Komputer*, vol. 14, no. 2-B, pp. 433-443, 2022.
- [3] E. Y. Pratiwi and N. Hasanah, "Sistem Informasi Penerimaan Mahasiswa Baru Fakultas Syariah dan Hukum," *STORAGE: Jurnal Ilmiah Teknik dan Ilmu Komputer*, vol. 1, no. 1, pp. 21-25, 2022.
- [4] A. I. Alchinov, O. Y. Yalovenko and M. V. Sygotina, "Main architectural patterns of web applications and web services using the example of banking systems," *International Conference on IT in Business and Industry*, 2021.
- [5] J. Duckett, *Web Design with HTML, CSS, JavaScript and jQuery Set*, Indianapolis: John Wiley & Sons, Inc., 2016.
- [6] J. N. Robbins, *Learning Web Design: A Beginner's Guide to HTML, CSS, JavaScript, and Web Graphics 5th Edition*, Canada: O'Reily Media, 2018.
- [7] E. A. Meyer and E. Weyl, *CSS: The Definitive Guide 4th Edition*, Canada: O'Reily Media, 2018.
- [8] P. Lops, M. d. Gemmis and G. Semeraro, *Recommender Systems HandBook*, New York: Springer, 2022.
- [9] L. Moroney, *The Definitive Guide to Firebase*, Washington: Apress, 2017.
- [10] A. D. Sole, *Visual Studio Code Distilled*, Cremona: Apress, 2023.
- [11] D. J. Anderson and A. Carmichael, *Essential Kanban Condensed*, Seattle: Lean Kanban University Press, 2015.
- [12] S. Seifermann, R. Heinrich, D. Werle and R. Reussner, "Detecting violations of access control and information flow policies in data flow diagrams," *The Journal of Systems & Software*, 2021.

- [13] P. Zhang, W. Dou and H. Liu, "Hierarchical Data Structures for Flowchart," *Scientific Reports*, no. 13, 2023.
- [14] M. R. Alifi, T. Semiawan, D. Leiharyani and H. Hayati, "Relational Data Modelling on the Document-Based NoSQL," *Jurnal Nasional Teknik Elektro dan Teknologi Informasi*, vol. 11, no. 3, 2022.

UKDLSM

LAMPIRAN A KODE PROGRAM

Pada bagian ini akan dilampirkan kode program untuk modul penting dalam aplikasi seleksi *On-The-Spot* PMB Unika De La Salle Manado.

1. Main Page

```
firebase.initializeApp(firebaseConfig);
const db = firebase.firestore();
const storage = firebase.storage();

let currentPage = 1;
let formPages = [];
let activeFormPages = [];
let formData = {};
let iqAnswers = {};
let warnaAnswers = {};
let customPageAnswers = {};
let iqQuestions = [];
let warnaQuestions = [];
let prodilist = [];
let settings = {};

function computeActivePages(selectedProdiId) {
  return formPages.filter(p => {
    if (!p.aktif && p.aktif !== undefined) return false;
    const BUILTIN = ['__prodi__', '__data__', '__iq__', '__warna__'];
    if (BUILTIN.includes(p.id)) return true;
    const ids = p.khususProdiIds && p.khususProdiIds.length > 0
      ? p.khususProdiIds
      : (p.khususProdiId ? [p.khususProdiId] : []);
    if (ids.length === 0) return true;
    return selectedProdiId && ids.includes(selectedProdiId);
  });
}

const DEFAULT_PAGES = [
  { id: '__prodi__', label: 'Program Studi', urutan: 1, aktif: true, builtin: true },
  { id: '__data__', label: 'Data Diri', urutan: 2, aktif: true, builtin: true },
  { id: '__iq__', label: 'Uji Kompetensi', urutan: 3, aktif: true, builtin: true },
  { id: '__warna__', label: 'Uji Buta Warna', urutan: 4, aktif: true, builtin: true },
];
```

```

function hideLoading() {
  const screen = document.getElementById('loadingScreen');
  if (screen) screen.classList.add('hidden');
}

async function init() {
  await loadSettings();
  await loadFormPages();
  await loadProdi();
  await loadQuestions();
  activeFormPages = computeActivePages('');
  createCustomPageElements();
  renderProgressSteps();
  renderProdiDropdown();
  renderIQQuestions();
  renderWarnaQuestions();
  goToPage(1, true);

  setTimeout(hideLoading, 500);
}

async function loadSettings() {
  try {
    const doc = await db.collection('settings').doc('main').get();
    if (doc.exists) {
      settings = doc.data();
      if (settings.uniName) {
        document.getElementById('uniNameDisplay').textContent =
settings.uniName;
        document.title = `Portal PMB - ${settings.uniName}`;
        const loadingTitle =
document.getElementById('loadingUniName');
        if (loadingTitle) loadingTitle.textContent =
settings.uniName;
      }
      if (settings.logoUrl) {
        document.getElementById('uniLogo').src = settings.logoUrl;
        const loadingLogo = document.getElementById('loadingLogo');
        if (loadingLogo) loadingLogo.src = settings.logoUrl;
      }
    }
  } catch (e) {}
}

async function loadFormPages() {
  try {

```

```

    const doc = await
db.collection('settings').doc('form_pages').get();
    if (doc.exists && doc.data().pages && doc.data().pages.length >
0) {
        formPages = doc.data().pages
            .filter(p => p.aktif !== false)
            .sort((a, b) => (a.urutan||99) - (b.urutan||99));
    } else {
        formPages = [...DEFAULT_PAGES];
    }
} catch (e) {
    formPages = [...DEFAULT_PAGES];
}
}

async function loadProdi() {
    try {
        const snap = await
db.collection('prodi').orderBy('createdAt').get();
        prodiList = !snap.empty ? snap.docs.map(d => ({ id: d.id,
...d.data() })) : defaultProdiList;
    } catch (e) {
        prodiList = defaultProdiList;
    }
}

async function loadQuestions() {
    try {
        const iqSnap = await
db.collection('soal_iq').orderBy('urutan').get();
        iqQuestions = !iqSnap.empty ? iqSnap.docs.map(d => ({ id: d.id,
...d.data() })) : [];
    } catch (e) { iqQuestions = []; }

    try {
        const wSnap = await
db.collection('soal_warna').orderBy('urutan').get();
        warnaQuestions = !wSnap.empty ? wSnap.docs.map(d => ({ id: d.id,
...d.data() })) : defaultWarnaQuestions;
    } catch (e) { warnaQuestions = defaultWarnaQuestions; }
}

function renderProdiDropdown() {
    const sel = document.getElementById('prodiSelect');
    sel.innerHTML = '<option value="">- Pilih Program Studi -
</option>';
    prodiList.forEach(p => {

```

```

    const opt = document.createElement('option');
    opt.value = p.id;
    opt.textContent = `${p.jenjang || 'S1'} - ${p.nama}`;
    sel.appendChild(opt);
  });
  sel.removeEventListener('change', onProdiChange);
  sel.addEventListener('change', onProdiChange);

  const sel2 = document.getElementById('prodiSelect2');
  sel2.innerHTML = '<option value="">- Pilih Program Studi -
</option>';
  prodiList.forEach(p => {
    const opt = document.createElement('option');
    opt.value = p.id;
    opt.textContent = `${p.jenjang || 'S1'} - ${p.nama}`;
    sel2.appendChild(opt);
  });
  sel2.removeEventListener('change', onProdiChange2);
  sel2.addEventListener('change', onProdiChange2);
}

function onProdiChange() {
  const val = document.getElementById('prodiSelect').value;
  const card = document.getElementById('prodiCard');
  const hint = document.getElementById('prodiHint');
  if (!val) {
    card.style.display = 'none';
    hint.textContent = '';
    updateProdiSpecificFields('');
    activeFormPages = computeActivePages('');
    renderProgressSteps();
    updateProgress(currentPage);
    return;
  }
  const p = prodiList.find(x => x.id === val);
  if (!p) return;
  document.getElementById('prodiCardName').textContent = p.nama;
  document.getElementById('prodiCardFakultas').textContent =
p.fakultas || '';
  document.getElementById('prodiCardJenjang').textContent =
p.jenjang || '';
  hint.textContent = p.deskripsi || '';
  card.style.display = 'flex';

  updateProdiSpecificFields(val);

  activeFormPages = computeActivePages(val);

```

```

renderProgressSteps();
updateProgress(currentPage);
}

function onProdiChange2() {
  const val = document.getElementById('prodiSelect2').value;
  const card = document.getElementById('prodiCard2');
  const hint = document.getElementById('prodiHint2');
  if (!val) {
    card.style.display = 'none';
    hint.textContent = '';
    return;
  }
  const p = prodiList.find(x => x.id === val);
  if (!p) return;
  document.getElementById('prodiCardName2').textContent = p.nama;
  document.getElementById('prodiCardFakultas2').textContent =
p.fakultas || '';
  document.getElementById('prodiCardJenjang2').textContent =
p.jenjang || '';
  hint.textContent = p.deskripsi ? `Cadangan: ${p.deskripsi}` : '';
  card.style.display = 'flex';
}

function updateProdiSpecificFields(prodiId) {
  const containers = [
    document.getElementById('prodiSpecificFieldsPage1'),
    document.getElementById('prodiSpecificFieldsPage2')
  ];
  containers.forEach(c => { if (c) c.innerHTML = ''; });

  if (!prodiId) return;

  const prodi = prodiList.find(p => p.id === prodiId);
  if (!prodi || !prodi.extraFields || prodi.extraFields.length ===
0) return;

  prodi.extraFields.forEach((field, idx) => {
    const pageTarget = field.showOnPage === '__data__' ?
'prodiSpecificFieldsPage2' : 'prodiSpecificFieldsPage1';
    const container = document.getElementById(pageTarget);
    if (!container) return;

    const wrapper = document.createElement('div');
    wrapper.className = 'field-group prodi-specific-field visible';
    wrapper.id = `psf-${prodiId}-${idx}`;
    wrapper.dataset.prodiId = prodiId;

```

```

wrapper.dataset.fieldIdx = idx;

const req = field.required ? '<span class="required">*</span>' :
'';
let inputHtml = '';

if (field.type === 'textarea') {
  inputHtml = `<textarea class="field-input psf-input" id="psf-
input-${prodiId}-${idx}" placeholder="Tulis jawaban Anda..."
rows="3" ${field.required ? 'required' : ''}></textarea>`;
} else if (field.type === 'radio') {
  const opts = (field.options || []).map((opt, oi) => `
<label class="option-item">
  <input type="radio" name="psf-${prodiId}-${idx}"
value="${escHtmlApp(opt)}" class="psf-input" data-psfid="${prodiId}-
${idx}" />
  ${escHtmlApp(opt)}
</label>`);
  inputHtml = `<div class="options-list">${opts}</div>`;
} else if (field.type === 'select') {
  const opts = (field.options || []).map(opt => `<option
value="${escHtmlApp(opt)}">${escHtmlApp(opt)}</option>`);
  inputHtml = `<div class="select-wrapper"><select class="field-
select psf-input" id="psf-input-${prodiId}-${idx}" ${field.required
? 'required' : ''}><option value="">- Pilih -
</option>${opts}</select><span class="select-arrow">▼</span></div>`;
} else {
  inputHtml = `<input type="${field.type === 'number' ? 'number'
: field.type === 'date' ? 'date' : 'text'}" class="field-input psf-
input" id="psf-input-${prodiId}-${idx}" placeholder="Tulis jawaban
Anda..." ${field.required ? 'required' : ''} />`;
}

wrapper.innerHTML = `
<label class="field-label">${escHtmlApp(field.label)}
${req}</label>
  ${inputHtml}
`;
container.appendChild(wrapper);
});
}

function collectProdiSpecificAnswers() {
  const prodiId = document.getElementById('prodiSelect').value;
  if (!prodiId) return {};
  const prodi = prodiList.find(p => p.id === prodiId);
  if (!prodi || !prodi.extraFields) return {};
}

```

```

const answers = {};
prodi.extraFields.forEach((field, idx) => {
  if (field.type === 'radio') {
    const checked = document.querySelector(`input[name="psf-
${prodiId}-${idx}"]:checked`);
    answers[field.label] = checked ? checked.value : '';
  } else {
    const el = document.getElementById(`psf-input-${prodiId}-
${idx}`);
    answers[field.label] = el ? el.value.trim() : '';
  }
});
return answers;
}

function validateProdiSpecificFields(pageId) {
  const prodiId = document.getElementById('prodiSelect').value;
  if (!prodiId) return true;
  const prodi = prodiList.find(p => p.id === prodiId);
  if (!prodi || !prodi.extraFields) return true;

  let valid = true;
  prodi.extraFields.forEach((field, idx) => {
    if (!field.required) return;
    const targetPage = field.showOnPage === '__data__' ? '2' : '1';
    if (targetPage !== pageId) return;

    if (field.type === 'radio') {
      const checked = document.querySelector(`input[name="psf-
${prodiId}-${idx}"]:checked`);
      if (!checked) {
        const grp = document.getElementById(`psf-${prodiId}-
${idx}`);
        if (grp) grp.classList.add('error');
        valid = false;
      }
    } else {
      const el = document.getElementById(`psf-input-${prodiId}-
${idx}`);
      if (el && !el.value.trim()) {
        el.classList.add('error');
        valid = false;
      }
    }
  });
  return valid;
}

```

```

}

function renderProgressSteps() {
  const container = document.getElementById('progressSteps');
  container.innerHTML = '';
  activeFormPages.forEach((p, i) => {
    const step = document.createElement('div');
    step.className = 'progress-step';
    step.id = `step-dot-${i+1}`;
    const dot = document.createElement('div');
    dot.className = 'step-dot';
    dot.textContent = i + 1;
    const label = document.createElement('span');
    label.className = 'step-label';
    label.textContent = p.label;
    step.appendChild(dot);
    step.appendChild(label);
    container.appendChild(step);
    if (i < activeFormPages.length - 1) {
      const line = document.createElement('div');
      line.className = 'step-line';
      line.id = `step-line-${i+1}`;
      container.appendChild(line);
    }
  });
}

function updateProgress(page) {
  const total = activeFormPages.length;
  for (let i = 1; i <= total; i++) {
    const stepEl = document.getElementById(`step-dot-${i}`);
    if (!stepEl) continue;
    stepEl.classList.remove('active', 'done');
    if (i < page) stepEl.classList.add('done');
    else if (i === page) stepEl.classList.add('active');
    const lineEl = document.getElementById(`step-line-${i}`);
    if (lineEl) lineEl.classList.toggle('done', i < page);
  }
  const fill = total > 1 ? ((page - 1) / (total - 1)) * 100 : 100;
  document.getElementById('progressFill').style.width = fill + '%';
}

function createCustomPageElements() {
  formPages.forEach((page) => {
    const builtinIds =
    ['__prodi__', '__data__', '__iq__', '__warna__'];

```

```

if (builtinIds.includes(page.id)) return;
if (document.getElementById(`page-${page.id}`)) return;

customPageAnswers[page.id] = customPageAnswers[page.id] || {};

const pageEl = document.createElement('div');
pageEl.className = 'form-page';
pageEl.id = `page-${page.id}`;

const fields = page.fields || [];
const fieldsHtml = fields.map((field, fi) => {
  const req = field.required ? '<span class="required">*</span>'
: '';
  let inputHtml = '';
  if (field.type === 'textarea') {
    inputHtml = `<textarea class="field-input custom-field"
id="cf-${page.id}-${fi}" placeholder="Tulis jawaban Anda..."
rows="3" ${field.required?'required':''}></textarea>`;
  } else if (field.type === 'radio') {
    const opts = (field.options||[]).map((opt,oi) => `
<label class="option-item" id="cfopt-${page.id}-${fi}-${
oi}">
    <input type="radio" name="cf-${page.id}-${fi}"
value="${escHtmlApp(opt)}" class="custom-field" data-
pageid="${page.id}" data-fi="${fi}" />
    ${escHtmlApp(opt)}
  </label>`);
    inputHtml = `<div class="options-list">${opts}</div>`;
  } else if (field.type === 'checkbox') {
    const opts = (field.options||[]).map((opt,oi) => `
<label class="option-item">
    <input type="checkbox" value="${escHtmlApp(opt)}"
class="custom-field custom-checkbox" data-pageid="${page.id}" data-
fi="${fi}" />
    ${escHtmlApp(opt)}
  </label>`);
    inputHtml = `<div class="options-list">${opts}</div>`;
  } else if (field.type === 'select') {
    const opts = (field.options||[]).map(opt => `<option
value="${escHtmlApp(opt)}">${escHtmlApp(opt)}</option>`);
    inputHtml = `<div class="select-wrapper"><select
class="field-select custom-field" id="cf-${page.id}-${fi}" data-
pageid="${page.id}" data-fi="${fi}"
${field.required?'required':''}><option value="">- Pilih -
</option>${opts}</select><span class="select-arrow">▼</span></div>`;
  } else if (field.type === 'number') {

```

```

        inputHtml = `

```

```

if (!page) return null;
const idMap = {
  '__prodi__': '1',
  '__data__': '2',
  '__iq__': '3',
  '__warna__': '4'
};
return idMap[page.id] || page.id;
}

function renderIQQuestions() {
  const container = document.getElementById('iqQuestionsContainer');
  container.innerHTML = '';
  iqQuestions.forEach((q, idx) => {
    container.appendChild(buildQuestionCard(q, idx + 1, 'iq',
iqAnswers));
  });
}

function renderWarnaQuestions() {
  const container =
document.getElementById('colorQuestionsContainer');
  container.innerHTML = '';
  warnaQuestions.forEach((q, idx) => {
    container.appendChild(buildQuestionCard(q, idx + 1, 'warna',
warnaAnswers));
  });
}

function buildQuestionCard(q, num, type, answersObj) {
  const card = document.createElement('div');
  card.className = 'question-card';
  card.id = `qcard-${type}-${q.id}`;
  const numEl = document.createElement('div');
  numEl.className = 'question-num';
  numEl.textContent = `Soal ${num}`;
  const textEl = document.createElement('div');
  textEl.className = 'question-text';
  textEl.textContent = q.pertanyaan;
  card.appendChild(numEl);
  if (q.imageUrl) {
    const imgWrap = document.createElement('div');
    imgWrap.className = 'question-image';
    const img = document.createElement('img');
    img.src = q.imageUrl;
    img.alt = `Gambar soal ${num}`;
    imgWrap.appendChild(img);
  }
}

```

```

    card.appendChild(imgWrap);
  } else if (type === 'warna' && q.imageDesc) {
    const imgWrap = document.createElement('div');
    imgWrap.className = 'question-image no-img';
    imgWrap.style.cssText =
'width:200px;height:200px;background:linear-
gradient(135deg,#d4dbb8,#9aab6a);display:flex;align-
items:center;justify-content:center;border-radius:12px;margin-
bottom:14px;';
    imgWrap.innerHTML = `

```

```

inp.value = answersObj[q.id] || '';
inp.addEventListener('input', () => {
  answersObj[q.id] = inp.value;
  card.classList.remove('error');
});
card.appendChild(inp);
}
return card;
}

function goToPage(page, isInit = false, skipValidation = false) {
  if (!isInit && !skipValidation) {
    const valid = validatePage(currentPage);
    if (!valid) return;
  }

  document.querySelectorAll('.form-page').forEach(p =>
p.classList.remove('active'));

  currentPage = page;
  const pageId = getPageId(page);
  const pageEl = document.getElementById(`page-${pageId}`);
  if (pageEl) pageEl.classList.add('active');

  updateProgress(page);
  window.scrollTo({ top: 0, behavior: 'smooth' });
  updateNavButtons(page);
}

function updateNavButtons(page) {
  const pageId = getPageId(page);
  const pageEl = document.getElementById(`page-${pageId}`);
  if (!pageEl) return;

  const prevBtn = pageEl.querySelector('.btn-back');
  const nextBtn = pageEl.querySelector('.btn-next');

  if (prevBtn) {
    if (page > 1) {
      prevBtn.onclick = () => goToPage(page - 1, false, true);
      prevBtn.style.visibility = '';
    } else {
      prevBtn.onclick = () => { window.location.href = 'index.html';
};
      prevBtn.style.visibility = '';
    }
  }
}
}

```

```

if (nextBtn) {
  if (page < activeFormPages.length) {
    nextBtn.onclick = () => goToPage(page + 1);
    nextBtn.innerHTML = 'Lanjut →';
    nextBtn.classList.remove('btn-submit');
  } else {
    nextBtn.onclick = () => submitForm();
    nextBtn.innerHTML = '<span>Kirim Pendaftaran</span>';
    nextBtn.classList.add('btn-submit');
  }
}
}

function validatePage(page) {
  clearErrors();
  let valid = true;
  const pageId = getPageId(page);

  if (pageId === '1') {
    const prodi = document.getElementById('prodiSelect').value;
    if (!prodi) {
      document.getElementById('prodiSelect').classList.add('error');
      showFieldError('prodiSelect', 'Pilih program studi terlebih
dahulu');
      valid = false;
    }
    const prodi2 = document.getElementById('prodiSelect2').value;
    if (!prodi2) {
      document.getElementById('prodiSelect2').classList.add('error')
;
      showFieldError('prodiSelect2', 'Pilih program studi pilihan
kedua');
      valid = false;
    }
    if (prodi && prodi2 && prodi === prodi2) {
      document.getElementById('prodiSelect2').classList.add('error')
;
      showFieldError('prodiSelect2', 'Pilihan kedua harus berbeda
dari pilihan pertama');
      valid = false;
    }
    if (valid && !validateProdiSpecificFields('1')) {
      alert('Mohon lengkapi semua field tambahan yang wajib
diisi.');
```

```

if (pageId === '2') {
  const fields =
['namaLengkap', 'tempatLahir', 'tanggalLahir', 'email', 'noHp', 'jenisKel
amin', 'domisili'];
  fields.forEach(id => {
    const el = document.getElementById(id);
    if (!el.value.trim()) { el.classList.add('error'); valid =
false; }
  });
  const email = document.getElementById('email').value;
  if (email && !/^[^\s@]+@[^\s@]+\.[^\s@]+$/.test(email)) {
    document.getElementById('email').classList.add('error');
    valid = false;
  }
  if (!validateProdiSpecificFields('2')) valid = false;
  if (!valid) alert('Mohon lengkapi semua field yang wajib
diisi.');
```

```

}

if (pageId === '3') {
  let unanswered = false;
  iqQuestions.forEach(q => {
    if (!iqAnswers[q.id]) {
      const card = document.getElementById(`qcard-iq-${q.id}`);
      if (card) card.classList.add('error');
      unanswered = true;
    }
  });
  if (unanswered) { alert('Mohon lengkapi jawaban uji
kompetensi.');
```

```

  valid = false; }
}

if (pageId === '4') {
  let unanswered = false;
  warnaQuestions.forEach(q => {
    if (!warnaAnswers[q.id]) {
      const card = document.getElementById(`qcard-warna-${q.id}`);
      if (card) card.classList.add('error');
      unanswered = true;
    }
  });
  if (unanswered) { alert('Mohon lengkapi jawaban uji buta
warna.');
```

```

  valid = false; }
}

const pageObj = activeFormPages[page - 1];
```

```

    if (pageObj &&
    ![ '__prodi__', '__data__', '__iq__', '__warna__'].includes(pageObj.id))
    {
        const fields = pageObj.fields || [];
        fields.forEach((field, fi) => {
            if (!field.required) return;
            if (field.type === 'radio') {
                const checked = document.querySelector(`input[name="cf-
                ${pageObj.id}-${fi}"]:checked`);
                if (!checked) {
                    const grp = document.getElementById(`cfg-${pageObj.id}-
                    ${fi}`);
                    if (grp) grp.classList.add('error');
                    valid = false;
                }
            } else if (field.type === 'checkbox') {
                const checked = document.querySelectorAll(`.custom-
                checkbox[data-pageid="${pageObj.id}"][data-fi="${fi}"]:checked`);
                if (checked.length === 0) {
                    const grp = document.getElementById(`cfg-${pageObj.id}-
                    ${fi}`);
                    if (grp) grp.classList.add('error');
                    valid = false;
                }
            } else {
                const el = document.getElementById(`cf-${pageObj.id}-
                ${fi}`);
                if (el && !el.value.trim()) {
                    el.classList.add('error');
                    valid = false;
                }
            }
        });
        if (!valid) alert('Mohon lengkapi semua field yang wajib
        diisi.');
```

```

    }

    return valid;
}

function clearErrors() {
    document.querySelectorAll('.error').forEach(el =>
    el.classList.remove('error'));
}

function showFieldError(id, msg) {
    const el = document.getElementById(id);

```

```

const parent = el.closest('.field-group');
if (!parent) return;
let hint = parent.querySelector('.field-hint');
if (!hint) { hint = document.createElement('p'); hint.className =
'field-hint'; parent.appendChild(hint); }
hint.style.color = 'var(--red)';
hint.textContent = msg;
}

async function submitForm() {
  if (!validatePage(currentPage)) return;

  const submitBtns = document.querySelectorAll('.btn-submit');
  submitBtns.forEach(b => { b.disabled = true; b.innerHTML =
  '<span>Mengirim...</span>'; });

  try {
    const prodiId = document.getElementById('prodiSelect').value;
    const prodiData = prodiList.find(p => p.id === prodiId);

    let pendaftaranId;
    try {
      const counterRef = db.collection('settings').doc('counter');
      pendaftaranId = await db.runTransaction(async (transaction) =>
{
      const counterDoc = await transaction.get(counterRef);
      const currentCount = counterDoc.exists ?
(counterDoc.data().count || 0) : 0;
      const newCount = currentCount + 1;
      transaction.set(counterRef, { count: newCount }, { merge:
true });
      const paddedNum = String(newCount).padStart(4, '0');
      const rand = Math.random().toString(36).substr(2,
5).toUpperCase();
      return `PMB-${paddedNum}-${rand}`;
    });
    } catch (e) {
      pendaftaranId = generateId();
    }

    const customAnswersCollected = {};
    activeFormPages.forEach(page => {
      const builtinIds =
['__prodi__', '__data__', '__iq__', '__warna__'];
      if (builtinIds.includes(page.id)) return;
      const fields = page.fields || [];
      customAnswersCollected[page.label] = {};
    });
  }
}

```

```

    fields.forEach((field, fi) => {
      if (field.type === 'radio') {
        const checked = document.querySelector(`input[name="cf-
        ${page.id}-${fi}"]:checked`);
        customAnswersCollected[page.label][field.label] = checked
        ? checked.value : '';
      } else if (field.type === 'checkbox') {
        const checked = document.querySelectorAll(`.custom-
        checkbox[data-pageid="${page.id}"][data-fi="${fi}"]:checked`);
        customAnswersCollected[page.label][field.label] =
        Array.from(checked).map(c => c.value);
      } else {
        const e1 = document.getElementById(`cf-${page.id}-${fi}`);
        customAnswersCollected[page.label][field.label] = e1 ?
        e1.value.trim() : '';
      }
    });
  });

  const prodiSpecificAnswers = collectProdiSpecificAnswers();

  const data = {
    pendaftaranId,
    prodi: prodiData ? prodiData.nama : prodiId,
    prodiId,
    prodi2: (() => { const id2 =
    document.getElementById('prodiSelect2').value; const p2 =
    prodiList.find(p => p.id === id2); return p2 ? p2.nama : id2; })(),
    prodiId2: document.getElementById('prodiSelect2').value,
    namaLengkap:
    document.getElementById('namaLengkap').value.trim(),
    nisn: document.getElementById('nisn').value.trim() || '0',
    tempatLahir:
    document.getElementById('tempatLahir').value.trim(),
    tanggalLahir: document.getElementById('tanggalLahir').value,
    email: document.getElementById('email').value.trim(),
    noHp: document.getElementById('noHp').value.trim(),
    jenisKelamin: document.getElementById('jenisKelamin').value,
    domisili: document.getElementById('domisili').value.trim(),
    jawabanIQ: { ...iqAnswers },
    jawabanWarna: { ...warnaAnswers },
    jawabanKustom: customAnswersCollected,
    jawabanProdiSpesifik: prodiSpecificAnswers,
    timestamp: firebase.firestore.FieldValue.serverTimestamp(),
    createdAt: new Date().toISOString()
  };

```

```

    let skorIQ = 0;
    iqQuestions.forEach(q => { if (q.jawaban && iqAnswers[q.id] ===
q.jawaban) skorIQ++; });
    data.skorIQ = skorIQ;
    data.totalSoalIQ = iqQuestions.length;

    let skorWarna = 0;
    warnaQuestions.forEach(q => { if (q.jawaban &&
warnaAnswers[q.id] === q.jawaban) skorWarna++; });
    data.skorWarna = skorWarna;
    data.totalSoalWarna = warnaQuestions.length;

    await db.collection('pendaftar').doc(pendaftaranId).set(data);

    document.getElementById('successName').textContent =
data.namaLengkap;
    document.getElementById('successProdi').textContent = data.prodi
+ (data.prodi2 ? ` (Cadangan: ${data.prodi2})` : '');
    document.getElementById('successId').textContent =
pendaftaranId;
    document.getElementById('successEmail').textContent =
data.email;

    document.querySelectorAll('.form-page').forEach(p =>
p.classList.remove('active'));
    document.getElementById('page-success').classList.add('active');
    document.getElementById('progressFill').style.width = '100%';

} catch (err) {
    console.error(err);
    alert('Terjadi kesalahan saat mengirim. Periksa koneksi Anda dan
coba lagi.');
```

```

    submitBtns.forEach(b => { b.disabled = false; b.innerHTML =
'<span>Kirim Pendaftaran</span>'; });
}
}

function generateId() {
    const ts = Date.now().toString(36).toUpperCase();
    const rand = Math.random().toString(36).substr(2,
5).toUpperCase();
    return `PMB-${ts}-${rand}`;
}

function openAdminModal() {
    document.getElementById('adminModal').classList.add('open');
    document.getElementById('adminPasswordInput').value = '';
}

```

```

document.getElementById('adminError').textContent = '';
setTimeout(() =>
document.getElementById('adminPasswordInput').focus(), 100);
}

function closeAdminModal() {
  document.getElementById('adminModal').classList.remove('open');
}

async function checkAdminPassword() {
  const pass = document.getElementById('adminPasswordInput').value;
  if (!pass) { document.getElementById('adminError').textContent =
'Masukkan kata sandi.'; return; }
  try {
    const doc = await db.collection('settings').doc('admin').get();
    const storedPass = doc.exists ? doc.data().password :
'admin123';
    if (pass === storedPass) {
      sessionStorage.setItem('pmb_admin', '1');
      window.location.href = 'admin.html';
    } else {
      document.getElementById('adminError').textContent = 'Kata
sandi salah.';
      document.getElementById('adminPasswordInput').value = '';
    }
  } catch (e) {
    if (pass === 'admin123') {
      sessionStorage.setItem('pmb_admin', '1');
      window.location.href = 'admin.html';
    } else {
      document.getElementById('adminError').textContent = 'Kata
sandi salah. (Default: admin123)';
    }
  }
}

document.addEventListener('DOMContentLoaded', () => {
  document.getElementById('adminPasswordInput').addEventListener('ke
ydown', e => {
    if (e.key === 'Enter') checkAdminPassword();
  });
  init();
});

```

LAMPIRAN B

USER ACCEPTANCE TEST

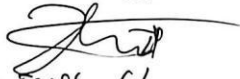
FORMULIR USER ACCEPTANCE TEST (UAT) APLIKASI WEB UNTUK SELEKSI ON-THE-SPOT PENERIMAAN MAHASISWA BARU — Admin	
Nama Penguji	Frankie

BAGIAN A — PENGALAMAN PENGGUNAAN						
No.	Pertanyaan	1	2	3	4	5
A1	Secara keseluruhan, seberapa mudah aplikasi ini digunakan? (1=Sangat Sulit, 5=Sangat Mudah)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
A2	Apakah tampilan (desain/antarmuka) aplikasi menarik dan nyaman dilihat? (1=Sangat Tidak Menarik, 5=Sangat Menarik)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
A3	Seberapa besar aplikasi ini membantu pekerjaan Anda sebagai Admin? (1=Tidak Membantu, 5=Sangat Membantu)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>

BAGIAN B — MASUKAN DAN SARAN	
B1	Fitur apa yang menurut Anda paling membantu dalam pekerjaan Admin? Jawaban: Export untuk mendapatkan history
B2	Apakah ada fitur yang perlu diperbaiki atau ditambahkan agar aplikasi lebih membantu pekerjaan Anda? Jawaban: untuk hasil export bisa ditambahkan ikon dari pdf

Manado, 10 Juni 2026

Penguji


 (Frankie)

Nama & Tanda Tangan

FORMULIR USER ACCEPTANCE TEST (UAT) APLIKASI WEB UNTUK SELEKSI ON-THE-SPOT PENERIMAAN MAHASISWA BARU — Calon Mahasiswa	
Nama Penguji	Rusell Herhan Rombejany

BAGIAN A — TAMPILAN & KEMUDAHAN PENGGUNAAN						
No.	Pertanyaan	1	2	3	4	5
A1	Apakah tampilan (desain) aplikasi ini menarik? (1=Sangat Tidak Menarik, 5=Sangat Menarik)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
A2	Apakah aplikasi ini mudah digunakan selama proses pendaftaran? (1=Sangat Sulit, 5=Sangat Mudah)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
A3	Apakah ukuran teks dan tombol mudah dibaca dan diklik? (1=Sangat Tidak Mudah, 5=Sangat Mudah)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
A4	Apakah aplikasi berjalan lancar di perangkat Anda (HP/Laptop)? (1=Sangat Tidak Lancar, 5=Sangat Lancar)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
A5	Secara keseluruhan, berikan penilaian pengalaman Anda menggunakan aplikasi ini! (1=Sangat Buruk, 5=Sangat Baik)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>

BAGIAN B — MASUKAN DAN SARAN	
B1	Apakah ada hal yang membingungkan atau menyulitkan Anda selama menggunakan aplikasi ini? Jawaban: Pengalaman saya dengan menggunakan aplikasi lancar dan tanpa tanpa kendala apapun. aplikasi tersebut memenuhi ekspektasi yang diharapkan dan tujuan form UAT
B2	Hal apa yang dapat dikembangkan atau ditambahkan dari aplikasi ini? Jawaban: Dari pengamatan saya program ini baik dilengkapi dengan fitur login supaya user yang terdaftar & berdaftar dapat melihat status pendaftaran mahasiswa baru

Manado, 10 Juni 2026

Penguji


(Rusell A. Rombejany)
Nama & Tanda Tangan

FORMULIR USER ACCEPTANCE TEST (UAT) APLIKASI WEB UNTUK SELEKSI ON-THE-SPOT PENERIMAAN MAHASISWA BARU — Calon Mahasiswa	
Nama Penguji	Jusuf Sumantri Mangalcom

BAGIAN A — TAMPILAN & KEMUDAHAN PENGGUNAAN

No.	Pertanyaan	1	2	3	4	5
A1	Apakah tampilan (desain) aplikasi ini menarik? (1=Sangat Tidak Menarik, 5=Sangat Menarik)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
A2	Apakah aplikasi ini mudah digunakan selama proses pendaftaran? (1=Sangat Sulit, 5=Sangat Mudah)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
A3	Apakah ukuran teks dan tombol mudah dibaca dan diklik? (1=Sangat Tidak Mudah, 5=Sangat Mudah)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
A4	Apakah aplikasi berjalan lancar di perangkat Anda (HP/Laptop)? (1=Sangat Tidak Lancar, 5=Sangat Lancar)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
A5	Secara keseluruhan, berikan penilaian pengalaman Anda menggunakan aplikasi ini! (1=Sangat Buruk, 5=Sangat Baik)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>

BAGIAN B — MASUKAN DAN SARAN

B1	Apakah ada hal yang membingungkan atau menyulitkan Anda selama menggunakan aplikasi ini?
Jawaban:	Sejalan pengalaman saya, Aplikasi berjalan sesuai dengan ekspektasi saya, dan menu mudah dilahani, Informatif, dan berisi informasi yang dibutuhkan untuk proses pendaftaran.
B2	Hal apa yang dapat dikembangkan atau ditambahkan dari aplikasi ini?
Jawaban:	Penambahan contact person atau website tambahan, group facebook.

Manado, 10 Juni 2026

Penguji


 (Jusuf Sumantri Mangalcom)
 Nama & Tanda Tangan

FORMULIR USER ACCEPTANCE TEST (UAT)

APLIKASI WEB UNTUK SELEKSI ON-THE-SPOT PENERIMAAN MAHASISWA BARU — Calon Mahasiswa

Nama Penguji Tiara n. wekv**BAGIAN A — TAMPILAN & KEMUDAHAN PENGGUNAAN**

No.	Pertanyaan	1	2	3	4	5
A1	Apakah tampilan (desain) aplikasi ini menarik? (1=Sangat Tidak Menarik, 5=Sangat Menarik)	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
A2	Apakah aplikasi ini mudah digunakan selama proses pendaftaran? (1=Sangat Sulit, 5=Sangat Mudah)	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
A3	Apakah ukuran teks dan tombol mudah dibaca dan diklik? (1=Sangat Tidak Mudah, 5=Sangat Mudah)	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
A4	Apakah aplikasi berjalan lancar di perangkat Anda (HP/Laptop)? (1=Sangat Tidak Lancar, 5=Sangat Lancar)	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
A5	Secara keseluruhan, berikan penilaian pengalaman Anda menggunakan aplikasi ini! (1=Sangat Buruk, 5=Sangat Baik)	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

BAGIAN B — MASUKAN DAN SARAN**B1** Apakah ada hal yang membingungkan atau menyulitkan Anda selama menggunakan aplikasi ini?Jawaban:
Tidak ada**B2** Hal apa yang dapat dikembangkan atau ditambahkan dari aplikasi ini?Jawaban:
Tanggal ujian bisa ditambahManado, 16 Juni 2026
PengujiTiara n. wekv
Nama & Tanda Tangan

LAMPIRAN C

TRANSKRIP WAWANCARA

TRANSKRIP WAWANCARA 1

Hari/Tanggal : Senin, 16 Januari 2026
Waktu : 13.00 – 13.30
Tempat : Ruang Biro La Salle Marketing Center (LMC)
Narasumber : Junaidy Budi Sanger (Kepala Biro LMC)
Pewawancara : Dustin Ignatius Kumala

Keterangan: P = Dustin Ignatius Kumala (Penulis) | N = Junaidy Budi Sanger (Kepala Biro LMC)

P : Selamat pagi, Sir. Terima kasih sudah meluangkan waktu untuk wawancara ini. Boleh Sir ceritakan secara umum bagaimana proses seleksi On-The-Spot PMB yang saat ini berjalan di Biro LMC?

N : Selamat pagi, Dustin. Jadi selama ini kami masih mengandalkan Google Forms untuk proses pendaftaran dan ujian. Calon mahasiswa yang datang langsung kami arahkan untuk mengisi formulir dan mengerjakan soal melalui Google Forms di perangkat yang kami sediakan.

P : Bagaimana alur lengkap proses On-The-Spot dari awal calon mahasiswa datang hingga selesai, Sir?

N : Alurnya begini: calon mahasiswa datang ke biro LMC, lalu staf kami mengarahkan mereka untuk mengisi data diri melalui Google Forms terlebih dahulu. Setelah itu mereka mengerjakan soal ujian kompetensi yang juga ada di Google Forms. Khusus untuk program studi Ilmu Keperawatan dan Fisioterapi, ada tambahan tes buta warna Ishihara yang dilakukan secara terpisah. Setelah semua selesai, hasil ujian dikumpulkan dan kami proses secara manual.

P : Lalu bagaimana proses pengumuman kelulusan dan tindak lanjutnya, Sir?

N : Setelah hasil diproses, calon mahasiswa yang dinyatakan lulus akan mendapatkan informasi dan diarahkan untuk mengurus administrasi keuangan. Prosesnya cukup panjang karena masih banyak dilakukan manual.

P : Apakah ada kendala utama dengan sistem yang berjalan sekarang, Sir?

N : Kendala terbesar ada di pengolahan data. Data dari Google Forms harus kami ekspor dan pindahkan ke Excel secara manual, ini makan waktu dan rawan kesalahan input. Selain itu kami tidak bisa membedakan alur pengerjaan per program studi dalam satu form yang sama.

P : Dari sisi kebijakan, siapa saja yang seharusnya memiliki akses ke sistem baru yang akan dikembangkan?

N : Untuk akses admin hanya staf LMC yang ditunjuk, satu sampai dua orang. Calon mahasiswa cukup mengakses halaman pendaftaran dan ujian saja tanpa perlu login.

P : Fitur apa yang paling Sir prioritaskan dalam sistem baru ini?

N : Yang paling penting adalah rekap data pendaftar otomatis, pengelolaan soal dan program studi yang bisa dilakukan staf sendiri, serta fitur ekspor data ke CSV atau Excel

untuk kebutuhan pelaporan. Keamanan data juga jadi perhatian kami karena datanya bersifat pribadi.

P : Apakah ada hal lain yang ingin Sir sampaikan sebagai masukan untuk pengembangan aplikasi ini?

N : Tampilannya harus sesederhana mungkin karena penggunanya termasuk calon mahasiswa yang baru pertama kali ke sini. Stabilitas sistem saat hari H seleksi juga sangat penting. Kalau bisa ada notifikasi atau konfirmasi setelah calon mahasiswa selesai mengisi formulir.

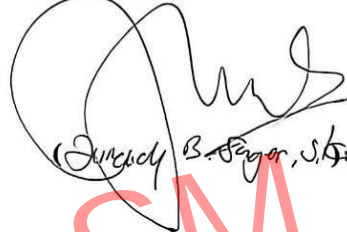
P : Baik Sir, terima kasih atas masukan dan waktunya. Informasi ini sangat berguna untuk perancangan sistem.

N : Sama-sama, Dustin. Semangat terus ya.

Pewawancara,

Dustin Ignatius Kumala

Narasumber,



(Jureddy B. Pagar, S.Sn., M.Kn.)

UKDLSM

TRANSKRIP WAWANCARA 2

Hari/Tanggal : Kamis, 19 Januari 2026
Waktu : 14.00 – 14.30
Tempat : Ruang Biro La Salle Marketing Center (LMC)
Narasumber : Hizkia R.M Joseph (Staf Biro LMC)
Pewawancara : Dustin Ignatius Kumala

Keterangan: P = Dustin Ignatius Kumala (Penulis) | S = Hizkia R.M Joseph (Staf Biro LMC)

P : Selamat pagi, Kak. Boleh Kak ceritakan tugas sehari-hari sebagai staf LMC, khususnya yang berkaitan dengan proses seleksi On-The-Spot PMB?

S : Selamat pagi, Dustin. Jadi kalau ada jadwal seleksi On-The-Spot, kami yang menyiapkan perangkat, memastikan Google Forms sudah siap, lalu mengarahkan calon mahasiswa satu per satu untuk mengisi formulir dan mengerjakan ujian. Setelah selesai kami juga yang mengumpulkan dan memeriksa data hasilnya.

P : Bisa Kak jelaskan lebih detail alur tes On-The-Spot dari awal sampai akhir seperti yang biasa Kak lakukan?

S : Jadi begini urutannya: pertama calon mahasiswa datang dan kami sambut di meja registrasi. Kami minta mereka mengisi data diri lewat Google Forms di laptop atau tablet yang sudah kami siapkan, mulai dari nama, tempat tanggal lahir, asal sekolah, NISN, NIK, nomor HP, email, dan pilihan program studi. Setelah itu mereka lanjut ke bagian ujian kompetensi, ada sekitar dua puluh soal pilihan ganda dan beberapa soal esai tergantung program studi. Khusus untuk calon mahasiswa yang memilih Ilmu Keperawatan atau Fisioterapi, ada tambahan tes buta warna Ishihara yang kami lakukan dengan buku tes fisik. Setelah semua tahap selesai, calon mahasiswa menunggu dan kami proses datanya. Kalau lulus, mereka diarahkan ke bagian administrasi keuangan.

P : Berapa lama biasanya waktu yang dibutuhkan untuk satu calon mahasiswa menyelesaikan seluruh proses tersebut, Kak?

S : Tergantung program studinya. Untuk yang tidak ada tes Ishihara biasanya sekitar tiga puluh sampai empat puluh menit. Yang ada tes Ishihara bisa sampai lima puluh menit atau lebih.

P : Kesulitan apa yang paling sering Kak hadapi saat menjalankan proses ini?

S : Yang paling merepotkan itu setelah seleksi selesai. Data dari Google Forms harus kami salin satu per satu ke Excel karena formatnya tidak selalu cocok. Kalau ada banyak pendaftar di satu hari, bisa makan waktu berjam-jam hanya untuk rekap data. Selain itu kadang ada calon mahasiswa yang salah mengisi data karena tidak ada validasi di Google Forms, seperti NIK atau NISN yang kurang atau kelebihan angka, atau tanggal lahir yang tidak masuk akal.

P : Apakah pernah ada kejadian teknis yang mengganggu proses seleksi, Kak?

S : Pernah. Waktu koneksi internet lambat, Google Forms susah diakses jadi calon mahasiswa harus menunggu lama. Kami juga tidak bisa memantau secara real-time siapa

yang sudah selesai dan siapa yang belum kecuali harus lihat satu per satu di Google Forms.

P : Kalau ada sistem baru yang menggantikan Google Forms, fitur apa yang menurut Kak paling membantu kerja sehari-hari?

S : Yang paling saya butuhkan itu validasi input otomatis supaya data yang masuk sudah benar dari awal, terus bisa lihat langsung siapa saja yang sudah selesai mengisi dan mengerjakan ujian dalam satu tampilan. Kalau ada rekap nilai otomatis juga sangat membantu. Dan tentunya ekspor data ke Excel atau CSV biar kami tidak perlu salin manual lagi.

P : Apakah ada masukan lain dari sisi operasional yang perlu saya perhatikan dalam membangun sistem ini, Kak?

S : Tampilan harus simpel dan cepat dimuat karena kadang kami pakai perangkat yang spesifikasinya tidak terlalu tinggi. Kalau bisa ada konfirmasi di layar setelah calon mahasiswa selesai mengerjakan, supaya kami tahu mereka sudah submit dan tidak perlu submit ulang.

P : Baik Kak, terima kasih atas waktunya. Penjelasan ini sangat membantu saya memahami kebutuhan sistem dari sisi operasional.

S : Sama-sama, semangat ya Dustin.

Pewawancara,

Narasumber,

Dustin Ignatius Kumala



(Hizki A.M. J. Saqih)