

LAPORAN TUGAS AKHIR

**SISTEM INFORMASI KERJA PRAKTEK BAGIAN MAHASISWA
PROGRAM STUDI KOMPUTER DAN SISTEM INFORMASI SV UGM
BERBASIS WEB**

***WEB BASED INFORMATION SYSTEM OF PRACTICAL WORK STUDENT
COLLEGE IN COMPUTER AND INFORMATION SYSTEM SV UGM***



DEAR NASYITA

17/410830/SV/12757

**PROGRAM STUDI D3 KOMPUTER DAN SISTEM INFORMASI
DEPARTEMEN TEKNIK ELEKTRO DAN INFORMATIKA
SEKOLAH VOKASI
UNIVERSITAS GADJAH MADA
YOGYAKARTA
2020**

LAPORAN TUGAS AKHIR

SISTEM INFORMASI KERJA PRAKTEK BAGIAN MAHASISWA
PROGRAM STUDI KOMPUTER DAN SISTEM INFORMASI SV UGM
BERBASIS WEB

***WEB BASED INFORMATION SYSTEM OF PRACTICAL WORK STUDENT
COLLEGE IN COMPUTER AND INFORMATION SYSTEM SV UGM***
Diajukan untuk memenuhi salah satu syarat memperoleh derajat Ahli Madya
Komputer dan Sistem Informasi



DEAR NASYITA
17/410830/SV/12757

PROGRAM STUDI D3 KOMPUTER DAN SISTEM INFORMASI
DEPARTEMEN TEKNIK ELEKTRO DAN INFORMATIKA
SEKOLAH VOKASI
UNIVERSITAS GADJAH MADA
YOGYAKARTA
2020

**HALAMAN PENGESAHAN
LAPORAN TUGAS AKHIR**

**SISTEM INFORMASI KERJA PRAKTEK BAGIAN MAHASISWA PADA
PROGRAM STUDI KOMPUTER DAN SISTEM INFORMASI SV UGM
BERBASIS WEB**

Disusun oleh :

Dear Nasyita

17/410830/SV/12757

Diajukan untuk memenuhi salah satu syarat memperoleh gelar Ahli Madya

Pada Program Studi Diploma III Komputer dan Sistem Informasi,

Departemen Teknik Elektro dan Informatika,

Sekolah Vokasi,

Universitas Gadjah Mada

Diterima dan disetujui oleh,

PERNYATAAN BEBAS PLAGIASI

Saya yang bertanda tangan di bawah ini :

Nama : Dear Nasyita

NIM : 17/410830/SV/12757

Tahun terdaftar : 2017

Program studi : D3 Komputer dan Sistem Informasi

Fakultas/Sekolah : Sekolah Vokasi

Menyatakan bahwa dalam dokumen ilmiah Tugas Akhir ini tidak terdapat bagian dari karya ilmiah lain yang telah diajukan untuk memperoleh gelar akademik di suatu lembaga Pendidikan Tinggi dan juga tidak terdapat karya atau pendapat yang pernah ditulis atau diterbitkan oleh orang atau lembaga lain kecuali yang secara tertulis disitasi dalam dokumen dan disebutkan sumbernya secara lengkap dalam daftar pustaka.

Dengan demikian saya menyatakan bahwa dokumen ilmiah ini bebas dari unsur unsur plagiasi dan apabila dokumen ilmiah Tugas Akhir ini di kemudian hari terbukti merupakan plagiasi dari hasil karya penulis lain dan/atau dengan sengaja mengajukan karya atau pendapat yang merupakan hasil karya penulis lain, maka penulis bersedia menerima sanksi akademik dan/atau sanksi hukum yang berlaku.

Klaten April 2020

Dear Nasyita

NIM. 17/410830/SV/12757

KATA PENGANTAR

Puji syukur kehadirat Allah SWT atas limpahan rahmat dan hidayah sehingga penulis dapat menyelesaikan Tugas Akhir yang berjudul “Sistem Informasi Kerja Praktek Bagian Mahasiswa Program Studi Komputer dan Sistem Informasi SV UGM Berbasis Web”. Laporan Tugas Akhir ini telah disusun sebagai salah satu syarat untuk mendapatkan gelar Ahli Madya pada program studi Komputer dan Sistem Informasi Sekolah Vokasi Universitas Gadjah Mada.

Dalam penyusunan Tugas Akhir ini penulis mendapatkan dukungan, bimbingan, dan bantuan dari berbagai pihak yang telah membantu dalam menyusun tugas akhir ini baik secara langsung maupun tidak langsung. Maka dari itu, perkenankanlah penulis menyampaikan ucapan terima kasih kepada :

1. Allah SWT yang telah melimpahkan keberkahan dan karunia Nya.
2. Orang tua tercinta Ibu Dewi Sulistyorini dan Bapak Supriyadi yang senantiasa menemani, memberi doa, dukungan, nasihat, dan motivasi kepada penulis.
3. Nenek dan adik yang selalu memberikan doa dan semangat kepada penulis.
4. Bapak Muhammad Fakhruddin, M.Cs., selaku dosen pembimbing Tugas Akhir penulis.
5. Bapak Irkham Huda, M.Cs., selaku klien yang telah memberikan penjelasan tentang sistem dan arahan kepada penulis.
6. Rama, Febi, dan Nofa sebagai tim seperjuangan dalam menyelesaikan Tugas Akhir.
7. Enno, Nisa, Jannah, dan sahabat kodew yang sudah menemani hari-hari selama kuliah, memberikan semangat, dan memberikan banyak kenangan selama berada di KOMSI.
8. Indra Wirawan yang telah banyak membantu dalam penggerjaan tugas akhir, juga memberikan dorongan besar kepada penulis.
9. Hanif Panji yang memberikan dukungan, doa, dan semangat kepada penulis.
10. Seluruh teman KOMSI atas kebersamaan, doa, dukungan, dan semangat selama kuliah hingga penggerjaan Tugas Akhir selesai.

11. Mas Abdur, Ilham mastah, dan Elang yang menjadi rujukan penulis saat mengalami kendala dalam mengerjakan Tugas Akhir ini.
12. Seluruh pihak yang telah mendukung dan membantu dalam pembuatan Tugas Akhir yang tidak bisa penulis sebutkan satu per satu.

Akhir kata penulis mengucapkan terima kasih dan memohon maaf jika Tugas Akhir ini masih jauh dari kata sempurna. Semoga Tugas Akhir ini dapat bermanfaat bagi diri penulis, pembaca dan semua pihak yang berkepentingan dalam Tugas Akhir ini.

Klaten, April 2020

Dear Nasyita

DAFTAR ISI

LAPORAN TUGAS AKHIR	i
LAPORAN TUGAS AKHIR	ii
HALAMAN PENGESAHAN	iii
PERNYATAAN BEBAS PLAGIASI	iv
KATA PENGANTAR	v
DAFTAR ISI	vii
DAFTAR GAMBAR	xi
DAFTAR TABEL	xvi
INTISARI	xvii
ABSTRACT	xviii
BAB I PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	1
1.3 Batasan Masalah	2
1.4 Manfaat Penelitian	2
1.5 Metode Penelitian	2
1.6 Sistematika Penulisan	3
BAB II TINJAUAN PUSTAKA	5
BAB III LANDASAN TEORI	7
3.1 Kerja Praktek	7
3.2 Sistem	7
3.2.1 Elemen Sistem	7
3.2.2 Karakteristik Sistem	7
3.2.3 Komponen Sistem Informasi	9
3.3 Siklus Hidup Pengembangan Sistem	9
3.3.1 Perancangan Sistem	10
3.3.2 Analisis Sistem	10
3.3.3 Desain Sistem	10

3.3.4	Implementasi Sistem	11
3.4	Diagram Alir Data	11
3.5	<i>Activity Diagram</i>	11
3.6	Basis Data.....	12
3.6.1	<i>Database Management System</i>	12
3.6.2	Entity Relationship Model	13
3.7	Internet.....	15
3.7.1	<i>Web</i>	15
3.7.2	<i>Web Server</i>	15
3.7.3	<i>Web Browser</i>	15
3.8	Bahasa Pemrograman	16
3.8.1	HTML	16
3.8.2	PHP (<i>Hypertext Preprocessor</i>)	16
3.8.3	CSS3.....	16
3.8.4	<i>Javascript</i>	16
3.8.5	<i>jQuery</i>	16
3.8.6	<i>MySQL</i>	17
3.8.7	Framework Laravel	17
3.8.8	JSON	17
3.8.9	<i>Application Programming Interface (API)</i>	17
3.9	Alat Implementasi	17
3.9.1	Visual Studio Code	17
3.9.2	XAMPP	18
3.9.3	Balsamiq Mockup	18
3.9.4	Postman	18
	BAB IV ANALISIS DAN PERANCANGAN SISTEM	19
4.1	Analisis Sistem	19
4.1.1	Analisis Masalah	19
4.1.2	Analisis Kebutuhan Pengguna	19
4.1.3	Analisis Kebutuhan Fungsional	20
4.1.4	Analisis Kebutuhan Nonfungsional	20

4.2	Perancangan Sistem.....	21
4.2.1	Perancangan Basis Data	21
4.2.2	Perancangan Proses	39
4.2.3	Perancangan Struktur Menu.....	46
4.2.4	Perancangan Antarmuka Pengguna.....	47
	BAB V IMPLEMENTASI SISTEM.....	65
5.1	Implementasi Perangkat Lunak Pembangun	65
5.2	Implementasi Perangkat Keras Pembangun	65
5.3	Implementasi Basis Data	66
5.4	Implementasi Antarmuka	78
	BAB VI HASIL PENGUJIAN DAN PEMBAHASAN	126
6.1	Skenario Pengujian.....	126
6.2	Hasil Pengujian.....	127
6.2.1	Pengujian <i>login</i> sistem	127
6.2.2	Pengujian Ubah Foto Profil.....	128
6.2.3	Pengujian Ubah Data Profil	128
6.2.4	Pengujian Lihat CV.....	128
6.2.5	Pengujian Buat Kelompok	129
6.2.6	Pengujian Tambah Anggota.....	129
6.2.7	Pengujian Hapus Undangan Anggota	130
6.2.8	Pengujian Penilaian Anggota	130
6.2.9	Pengujian Tambah Usulan Instansi	131
6.2.10	Pengujian Lihat Semua Lowongan	131
6.2.11	Pengujian Lihat Detail Lowongan	132
6.2.12	Pengujian <i>Apply</i> Lowongan	132
6.2.13	Pengujian Tambah Laporan Harian	133
6.2.14	Pengujian Tambah Laporan Akhir	133
6.2.15	Pengujian Lihat Laporan Akhir.....	134
6.2.16	Pengujian Lihat Semua Pengumuman	134
6.2.17	Pengujian Lihat Detail Pengumuman.....	135
6.2.18	Pengujian Ubah Password.....	136

BAB VII KESIMPULAN DAN SARAN	137
7.1 Kesimpulan.....	137
7.2 Saran	137
DAFTAR PUSTAKA	138
LAMPIRAN	140

DAFTAR GAMBAR

Gambar 3. 1 Siklus Hidup Pengembangan Sistem Waterfall	10
Gambar 3. 2 Komponen ERD	14
Gambar 4. 1 ERD Sistem Informasi Kerja Praktek	22
Gambar 4. 2 Diagram usecase.....	40
Gambar 4. 3 Activity diagram login sistem	41
Gambar 4. 4 Activity diagram melihat halaman menu	41
Gambar 4. 5 Activity diagram buat kelompok.....	42
Gambar 4. 6 Activity diagram penilaian anggota	43
Gambar 4. 7 Activity diagram usulan instansi	43
Gambar 4. 8 Activity diagram melamar lowongan.....	44
Gambar 4. 9 Activity diagram laporan harian.....	45
Gambar 4. 10 Activity diagram laporan akhir	45
Gambar 4. 11 Activity diagram ubah avatar	46
Gambar 4. 12 Rancangan struktur menu.....	47
Gambar 4. 13 Rancangan antarmuka halaman login	48
Gambar 4. 14 Rancangan antarmuka halaman dashboard	49
Gambar 4. 15 Rancangan antarmuka halaman profil.....	49
Gambar 4. 16 Rancangan antarmuka halaman edit profil.....	50
Gambar 4. 17 Rancangan halaman buat kelompok.....	50
Gambar 4. 18 Rancangan antarmuka halaman tambah anggota	51
Gambar 4. 19 Rancangan antarmuka halaman riwayat buat kelompok.....	51
Gambar 4. 20 Rancangan antarmuka halaman daftar anggota kelompok.....	52
Gambar 4. 21 Rancangan antarmuka halaman penilaian anggota kelompok	53
Gambar 4. 22 Rancangan antarmuka halaman form penilaian	53
Gambar 4. 23 Rancangan antarmuka halaman detail kelompok tab anggota	54
Gambar 4. 24 Rancangan antarmuka halaman detail kelompok tab info kerja praktek.....	55
Gambar 4. 25 Rancangan antarmuka halaman detail kelompok tab jadwal presentasi.....	56
Gambar 4. 26 Rancangan antarmuka halaman tambah usulan instansi	57
Gambar 4. 27 Rancangan antarmuka halaman edit usulan instansi	57
Gambar 4. 28 Rancangan antarmuka halaman detail usulan instansi	58
Gambar 4. 29 Rancangan antarmuka halaman daftar lowongan.....	58
Gambar 4. 30 Rancangan antarmuka halaman apply lowongan	59
Gambar 4. 31 Rancangan antarmuka halaman riwayat apply lowongan	59
Gambar 4. 32 Rancangan antarmuka halaman detail laporan harian.....	60

Gambar 4. 33 Rancangan antarmuka halaman tambah laporan harian	61
Gambar 4. 34 Rancangan antarmuka halaman edit laporan harian.....	61
Gambar 4. 35 Rancangan antarmuka halaman detail laporan akhir.....	62
Gambar 4. 36 Rancangan antarmuka halaman tambah laporan akhir.....	62
Gambar 4. 37 Rancangan antarmuka halaman pengumuman.....	63
Gambar 4. 38 Rancangan antarmuka halaman detail pengumuman	63
Gambar 4. 39 Rancangan antarmuka halaman ubah password.....	64
Gambar 5. 1 Implementasi tabel admin	66
Gambar 5. 2 Implementasi tabel dosen	67
Gambar 5. 3 Implementasi tabel mahasiswa.....	67
Gambar 5. 4 Implementasi tabel aspek penilaian.....	68
Gambar 5. 5 Implementasi tabel buku harian	68
Gambar 5. 6 Implementasi tabel jadwal presentasi.....	69
Gambar 5. 7 Implementasi tabel kelompok	69
Gambar 5. 8 Implementasi tabel kelompok detail	70
Gambar 5. 9 Implementasi tabel kelompok penilai	70
Gambar 5. 10 Implementasi tabel laporan	70
Gambar 5. 11 Implementasi tabel lowongan.....	71
Gambar 5. 12 Implementasi tabel magang.....	71
Gambar 5. 13 Implementasi tabel nilai	72
Gambar 5. 14 Implementasi tabel pelamar	72
Gambar 5. 15 Implementasi tabel pengumuman.....	73
Gambar 5. 16 Implementasi tabel periode	73
Gambar 5. 17 Implementasi tabel roles.....	73
Gambar 5. 18 Implementasi tabel ruang	74
Gambar 5. 19 Implementasi tabel sesiwaktu.....	74
Gambar 5. 20 Implementasi tabel users	74
Gambar 5. 21 Implementasi tabel usulan	75
Gambar 5. 22 Implementasi tabel instansi	76
Gambar 5. 23 Hubungan relasi antar tabel.....	77
Gambar 5. 24 Implementasi halaman login	78
Gambar 5. 25 Implementasi halaman dashboard	79
Gambar 5. 26 Kode program jumlah anggota.....	79
Gambar 5. 27 Implementasi halaman profil.....	80
Gambar 5. 28 Kode program halaman profil	81
Gambar 5. 29 Implementasi halaman edit profil.....	81
Gambar 5. 30 Kode program untuk edit profil.....	82
Gambar 5. 31 Kode program edit profil.....	83
Gambar 5. 32 Pop up update avatar	84

Gambar 5. 33 Kode program ubah avatar	84
Gambar 5. 34 Implementasi halaman buat kelompok.....	85
Gambar 5. 35 Kode program create nama kelompok	86
Gambar 5. 36 Implementasi halaman tambah anggota	86
Gambar 5. 37 Kode program daftar mahasiswa.....	87
Gambar 5. 38 Kode program add anggota	88
Gambar 5. 39 Implementasi halaman riwayat buat kelompok.....	88
Gambar 5. 40 Kode program halaman riwayat kelompok	89
Gambar 5. 41 Implementasi halaman daftar anggota kelompok.....	90
Gambar 5. 42 Kode program halaman daftar anggota kelompok	91
Gambar 5. 43 Kode program delete anggota kelompok	92
Gambar 5. 44 Implementasi halaman penilaian anggota_kelompok	92
Gambar 5. 45 Kode program halaman daftar penilaian anggota kelompok	93
Gambar 5. 46 Implementasi halaman form penilaian	94
Gambar 5. 47 Implementasi halaman form penilaian	94
Gambar 5. 48 Implementasi halaman detail kelompok tab anggota	95
Gambar 5. 49 Implementasi halaman detail kelompok tab info kerja praktek	96
Gambar 5. 50 Halaman detail kelompok tab jadwal presentasi	96
Gambar 5. 51 Kode program untuk menampilkan halaman detail kelompok	98
Gambar 5. 52 Implementasi halaman detail usulan instansi	99
Gambar 5. 53 Kode program halaman daftar usulan instansi	100
Gambar 5. 54 Implementasi halaman tambah usulan instansi	101
Gambar 5. 55 Kode program halaman tambah usulan instansi.....	102
Gambar 5. 56 Kode program tambah usulan instansi	103
Gambar 5. 57 Implementasi edit usulan instansi	104
Gambar 5. 58 Kode program halaman edit usulan instansi	104
Gambar 5. 59 Kode program edit usulan instansi	105
Gambar 5. 60 Implementasi halaman daftar lowongan	106
Gambar 5. 61 Kode program halaman daftar lowongan	106
Gambar 5. 62 Implementasi halaman apply lowongan.....	107
Gambar 5. 63 Kode program halaman apply lowongan	108
Gambar 5. 64 Kode program apply lowongan	109
Gambar 5. 65 Implementasi halaman riwayat apply lowongan.....	109
Gambar 5. 66 Kode program halaman riwayat apply lowongan	110
Gambar 5. 67 Implementasi.halaman detail laporan harian.....	111
Gambar 5. 68 Kode program halaman daftar laporan harian.....	111
Gambar 5. 69 Implementasi halaman tambah laporan harian.....	113
Gambar 5. 70 Kode program halaman tambah laporan harian	113
Gambar 5. 71 Kode program tambah laporan harian	114

Gambar 5. 72 Implementasi halaman edit laporan harian.....	115
Gambar 5. 73 Kode program halaman edit laporan	115
Gambar 5. 74 Kode program edit laporan harian.....	116
Gambar 5. 75 Implementasi halaman detail laporan akhir	117
Gambar 5. 76 Kode program halaman daftar laporan akhir.....	117
Gambar 5. 77 Implementasi halaman tambah laporan akhir.....	118
Gambar 5. 78 Kode program halaman tambah laporan akhir	119
Gambar 5. 79 Kode program tambah laporan akhir.....	119
Gambar 5. 80 Implementasi halaman edit laporan akhir	120
Gambar 5. 81 Kode program edit laporan akhir	120
Gambar 5. 82 Kode program ubah laporan akhir.....	121
Gambar 5. 83 Pop up menampilkan laporan akhir.....	122
Gambar 5. 84 Kode program pop up halaman laporan akhir	122
Gambar 5. 85 Implementasi halaman pengumuman	122
Gambar 5. 86 Kode program menampilkan daftar pengumuman di halaman mahasiswa	123
Gambar 5. 87 Implementasi halaman detail pengumuman	123
Gambar 5. 88 Kode program detail pengumuman	124
Gambar 5. 89 Implementasi halaman ubah.password.....	124
Gambar 5. 90 Kode program untuk menampilkan halaman ubah password	124
Gambar 5. 91 Kode program untuk update password.....	125
Gambar 6. 1 Tampilan pesan error username atau password tidak sesuai	127
Gambar 6. 2 Notifikasi foto profil berhasil diubah	128
Gambar 6. 3 Notifikasi berhasil ubah data profil	128
Gambar 6. 4 Tampilan berhasil lihat CV	129
Gambar 6. 5 Notifikasi kelompok berhasil dibuat	129
Gambar 6. 6 Anggota berhasil ditambahkan.....	130
Gambar 6. 7 Notifikasi berhasil hapus undangan anggota.....	130
Gambar 6. 8 Notifikasi nilai berhasil ditambahkan	130
Gambar 6. 9 Notifikasi masukan data tidak valid	131
Gambar 6. 10 Notifikasi berhasil menambahkan usulan	131
Gambar 6. 11 Tampilan lihat semua lowongan	132
Gambar 6. 12 Tampilan lihat semua lowongan	132
Gambar 6. 13 Notifikasi berhasil apply lowongan.....	133
Gambar 6. 14 Notifikasi berhasil menambah laporan harian.....	133
Gambar 6. 15 Notifikasi laporan akhir berhasil ditambahkan	133
Gambar 6. 16 Notifikasi file laporan terlalu besar.....	133
Gambar 6. 17 Notifikasi format file laporan tidak sesuai ketentuan.....	134
Gambar 6. 18 Pop up laporan akhir	134
Gambar 6. 19 Halaman lihat semua pengumuman	135

Gambar 6. 20 Pop up detail pengumuman	135
Gambar 6. 21 Notifikasi berhasil ubah password	136
Gambar 6. 22 Notifikasi error data masukan terlalu pendek	136
Gambar 6. 23.Notifikasi error masukan data confirm password tidak sama	136

DAFTAR TABEL

Tabel 2. 1 Perbandingan Tinjauan Pustaka	6
Tabel 3. 1 Simbol Activity Diagram	11
Tabel 4. 1 Rancangan tabel admin	23
Tabel 4. 2 Rancangan tabel dosen.....	24
Tabel 4. 3 Rancangan tabel mahasiswa.....	25
Tabel 4. 4 Rancangan tabel aspek_penilaian	26
Tabel 4. 5 Rancangan tabel buku_harian	26
Tabel 4. 6 Rancangan tabel jadwal_presentasi	27
Tabel 4. 7 Rancangan tabel kelompok	28
Tabel 4. 8 Rancangan tabel kelompok_detail	29
Tabel 4. 9 Rancangan tabel kelompok_penilai	30
Tabel 4. 10 Rancangan tabel laporan	30
Tabel 4. 11 Rancangan tabel lowongan	31
Tabel 4. 12 Rancangan tabel magang	32
Tabel 4. 13 Rancangan tabel nilai	33
Tabel 4. 14 Rancangan tabel pelamar	33
Tabel 4. 15 Rancangan tabel pengumuman	34
Tabel 4. 16 Rancangan tabel periode	35
Tabel 4. 17 Rancangan tabel roles	35
Tabel 4. 18 Rancangan tabel ruang	36
Tabel 4. 19 Rancangan tabel sesiwaktu	36
Tabel 4. 20 Rancangan tabel users	37
Tabel 4. 21 Rancangan tabel usulan.....	38
Tabel 4. 22 Rancangan tabel instansi.....	39
Tabel 6. 1 Skenario pengujian.....	126

INTISARI

SISTEM INFORMASI KERJA PRAKTEK BAGIAN MAHASISWA PROGRAM STUDI KOMPUTER DAN SISTEM INFORMASI SV UGM BERBASIS WEB

Oleh:
Dear Nasyita
17/410830/SV/12757

Kerja Praktek (KP) adalah kegiatan penerapan ilmu yang diperoleh mahasiswa di bangku perkuliahan pada suatu pekerjaan dan bertujuan untuk melatih mahasiswa agar mengenal situasi dunia kerja sekaligus untuk meningkatkan kualitas mahasiswa. Dalam pelaksanaan kerja praktek pada program studi Komputer dan Sistem Informasi SV UGM masih dilakukan secara konvensional sehingga menimbulkan masalah yaitu kurang praktis dan tidak efisien dalam proses pendaftaran hingga penilaian anggota kelompok kerja praktek.

Berdasarkan permasalahan tersebut, perlu dibangun sebuah sistem yang dapat mengatur proses pendaftaran hingga penilaian anggota kelompok kerja praktek pada program studi Komputer dan Sistem Informasi SV UGM. Sistem bertujuan untuk memproses dari tahapan buat kelompok, mendaftar ke instansi, pengelolaan laporan harian saat kerja praktek, hingga penilaian antar anggota kelompok kerja praktek.

Oleh karena itu, penulis membuat Sistem Informasi Kerja Praktek Bagian Mahasiswa pada program studi Komputer dan Sistem Informasi SV UGM Berbasis Web. Sistem dibangun menggunakan *framework* Laravel dengan bahasa pemrograman PHP, web server Apache, dan basis data MySQL. Permbuatan sistem diharapkan dapat mengatasi masalah terhadap proses pendaftaran hingga penilaian pada kerja praktek.

Kata kunci : Sistem informasi , kerja praktek, website, laravel

ABSTRACT

WEB BASED INFORMATION SYSTEM OF PRACTICAL WORK STUDENT COLLEGE IN COMPUTER AND INFORMATION SYSTEM SV UGM

By

Dear Nasyita

17/410830/SV/12757

Practical work is an implementation of knowledge which is gained by students in lectures, which aims to train students to recognize the situation of work and also improve the quality of students. In the implementation of the practical work in Computer and Information System study program still done conventional, so it has problems that are less practical and inefficient in registration process until group member assessment of practical work.

Based on these problems, we need to build a system that can regulate the registration process up to reporting on practical work in Computer and Information Systems SV UGM. The purpose of the system is to facilitate preparation of groups, registering with institutions, organizing daily reports on practical work, and assessment between members of practical work groups.

Therefore, the authors make the Web Based Information System of Practical Work Student College in Computer and Information System SV UGM. The system was built using the Laravel framework with the PHP programming language, Apache web server, and the MySQL database. The system development is expected to be able to overcome problems with the registration process and the assessment of practical work.

Keywords: Information systems, practical work , website, laravel

BAB I

PENDAHULUAN

1.1 Latar Belakang

Teknologi informasi mengalami perkembangan yang sangat pesat, ditandai dengan adanya bidang pekerjaan yang pada awalnya dilakukan menggunakan cara konvensional sekarang dilakukan dengan teknologi komputer yang membuat pekerjaan menjadi lebih praktis dan efisien.

Kerja Praktek (KP) merupakan kegiatan penerapan ilmu yang diperoleh mahasiswa pada suatu bidang pekerjaan dan bertujuan untuk melatih mahasiswa agar mengenal situasi dunia kerja sekaligus untuk meningkatkan kualitas mahasiswa. Kerja praktek merupakan salah satu kurikulum wajib yang harus ditempuh mahasiswa Komputer dan Sistem Informasi Sekolah Vokasi Universitas Gadjah Mada sebelum penulisan tugas akhir sebagai syarat kelulusan.

Dalam pelaksanaan kerja praktek pada program studi Komputer dan Sistem Informasi SV UGM belum optimal karena masih dilaksanakan dengan cara konvensional. Diantara beberapa kendala yang ditemui adalah pendaftaran yang belum terpusat, pemilihan instansi tempat melakukan kerja praktek yang masih kurang efektif, laporan harian yang masih menggunakan kertas, dan penilaian anggota kelompok yang masih menggunakan kertas sehingga kurang praktis dan tidak efisien.

Dari permasalahan diatas, penulis memberikan solusi dengan membangun sebuah sistem informasi yang akan memudahkan mahasiswa dalam proses pelaksanaan kerja praktek sehingga penulis membuat sistem ini dengan mengangkat judul Sistem Informasi Kerja Praktek Bagian Mahasiswa pada program studi Komputer dan Sistem Informasi SV UGM Berbasis Web.

1.2 Rumusan Masalah

Berdasarkan latar belakang yang telah dijelaskan sebelumnya, dapat disimpulkan permasalahan yang dihadapi yaitu bagaimana membuat sistem informasi kerja praktek bagian mahasiswa pada program studi Komputer dan Sistem Informasi SV UGM.

1.3 Batasan Masalah

Batasan masalah dalam pembuatan sistem informasi pendaftaran kerja praktek adalah sebagai berikut :

- a. Aktivitas penilaian yang dilakukan oleh mahasiswa tidak menggunakan sistem ranking
- b. Tidak ada fitur *planning calendar*

1.4 Manfaat Penelitian

Manfaat penelitian ini adalah :

- a. Pendaftaran kerja praktek menjadi terpusat.
- b. Memudahkan mahasiswa dalam pelaporan kegiatan harian saat kerja praktek.
- c. Dapat menghasilkan suatu sistem yang tepat, efektif, dan berguna bagi program studi komputer dan sistem informasi.

1.5 Metode Penelitian

Metodologi penelitian yang digunakan dalam pembuatan sistem ini adalah:

1. Metode pengumpulan data

- a. Riset

Metode riset dilakukan dengan cara memahami alur proses kerja praktek pada program studi D3 Komputer dan Sistem Informasi SV UGM.

- b. Wawancara

Metode wawancara dilakukan dengan tanya jawab kepada Bapak Irkham Huda.,M.Cs, selaku client. Wawancara dilakukan supaya mendapatkan gambaran tentang sistem sesuai dengan kebutuhan user sehingga implementasinya sesuai dengan yang diinginkan.

- c. Studi Literatur

Metode studi literatur dilakukan dengan cara mencari referensi sistem serupa yang berkaitan dengan sistem informasi kerja praktek.

2. Pengembangan sistem

a. Analisis Sistem

Dilakukan dengan menganalisa sistem yang akan dibangun. Hasil dari analisis adalah ditemukannya kebutuhan fungsional dan kebutuhan non fungsional sistem.

b. Perancangan Sistem

Perancangan sistem berisi tentang perancangan berdasarkan analisis yang telah dilakukan. Perancangan sistem meliputi perancangan proses, perancangan struktur menu, perancangan basis data, dan perancangan antarmuka.

c. Implementasi Sistem

Implementasi sistem berisi tentang pengimplementasian dari perancangan yang telah telah dilakukan.

d. Pengujian Sistem

Pengujian sistem dilakukan dengan cara melakukan pengujian pada sistem. Tujuannya untuk menguji kesesuaian sistem yang dibuat dengan analisis dan perancangan yang telah dilakukan. Fungsi dari pengujian sistem adalah untuk mengurangi kesalahan sistem dan memastikan sistem dapat berjalan dengan baik.

1.6 Sistematika Penulisan

Penulisan laporan tugas akhir ini disusun dalam tujuh bab. Sistematika penulisan laporan diuraikan sebagai berikut :

BAB I PENDAHULUAN

Bab ini menjelaskan tentang latar belakang, rumusan masalah, batasan masalah, tujuan penelitian, manfaat penelitian, metodologi penelitian dan sistematika penulisan.

BAB II TINJAUAN PUSTAKA

Bab ini berisi deskripsi singkat sistem sejenis yang pernah dibuat sebelumnya dan digunakan sebagai referensi dalam pembuatan tugas akhir ini.

BAB III LANDASAN TEORI

Berisi teori yang sesuai dengan sistem yang dibuat, konsep, pendapat dan prinsip yang berasal dari berbagai sumber seperti buku maupun karya tulis ilmiah yang lain dan dapat dipertanggungjawabkan secara ilmiah.

BAB IV ANALISIS DAN PERANCANGAN SISTEM

Berisi penjelasan tahapan analisis dan perancangan dari sistem yang dibuat. Tahap analisis terdiri dari deskripsi proses bisnis sistem dan kebutuhan sistem, sedangkan proses perancangan meliputi perancangan basis data, perancangan proses yang ada, perancangan struktur menu dan perancangan antarmuka pengguna.

BAB V IMPLEMENTASI SISTEM

Berisi detail penerapan dari rancangan yang telah dibuat berupa potongan kode program dan penjelasan.

BAB VI HASIL PENGUJIAN DAN PEMBAHASAN

Penguraian yang berhubungan dengan pengujian dan pembahasan dari sistem yang dibuat.

BAB VII PENUTUP

Berisi kesimpulan dari pembuatan sistem yang telah selesai dilakukan dan berisi saran untuk penelitian selanjutnya.

DAFTAR PUSTAKA

Berisikan sumber referensi literatur sebagai acuan penulis.

LAMPIRAN

Lampiran berisi dokumen pendukung tambahan.

BAB II

TINJAUAN PUSTAKA

Peneliti melakukan penelitian terhadap pembuatan sistem informasi kerja praktek di lingkungan program studi Komputer dan Sistem Informasi SV UGM. Penelitian tersebut bertujuan untuk memberi kemudahan bagi mahasiswa dalam melakukan pendaftaran hingga penilaian anggota kelompok kerja praktek. Penelitian dan peninjauan juga dilakukan pada beberapa karya yang memiliki kesamaan sistem sejenis dan dapat membantu peneliti dalam mengembangkan sistem ini.

Trimanto (2018) mengembangkan sistem informasi praktek kerja lapangan berbasis web menggunakan framework CodeIgniter untuk membantu pendataan seluruh kegiatan mahasiswa yang melakukan praktek kerja lapangan pada suatu instansi, mulai dari pendaftaran hingga penilaian.

Andriyanto & Aswi (2016) mengembangkan sistem informasi berjudul “Rancang Bangun Sistem Informasi Praktek Kerja Lapangan Terintegrasikan Menggunakan Webservice”. Sistem dibangun menggunakan RESTful Webservice dan terintegrasi dengan Sistem Informasi Akademik, sistem digunakan untuk mendukung sistem pelaporan dan monitoring pelaksanaan PKL.

Setiawan (2016) mengembangkan sistem informasi berjudul “Sistem Informasi Pendaftaran Praktek Kerja Lapangan (PKL) Berbasis Web Pada Prodi Sistem Informasi Universitas Nusantara PGRI Kediri”. Sistem dibuat dengan bahasa pemrograman PHP dan metode pengembangan waterfall, sistem ini dapat membantu admin dalam proses pendaftaran.

Perbandingan tinjauan pustaka penelitian dapat dilihat pada Tabel 2.1

Tabel 2. 1 Perbandingan Tinjauan Pustaka

No	Pengarang	Tahun	Judul	Isi
1	Abdurrahman Trimanto	2018	Sistem Informasi Praktik Kerja Lapangan Berbasis Web Prodi Komputer dan Sistem Informasi	Pembuatan sistem informasi praktik kerja lapangan berbasis web menggunakan framework CodeIgniter
2	Teguh Andriyanto, Risky Aswi R	2016	Rancang Bangun Sistem Informasi Praktek Kerja Lapangan Terintegrasi Menggunakan Webservice	Pembuatan sistem informasi praktek kerja lapangan yang terintegrasi dengan SIA menggunakan RESTful Webservice
3	Pendik Setiawan	2016	Sistem Informasi Pendaftaran Praktek Kerja Lapangan (PKL) Berbasis Web Pada Prodi Sistem Informasi Universitas Nusantara PGRI Kediri	Perancangan sistem informasi pendaftaran praktek kerja lapangan menggunakan bahasa pemrograman PHP dan metode pengembangan waterfall

BAB III

LANDASAN TEORI

3.1 Kerja Praktek

Kerja praktek merupakan kegiatan penerapan ilmu yang diperoleh mahasiswa dan diterapkan dalam bidang pekerjaan. Kerja praktek memiliki tujuan untuk meningkatkan kualitas mahasiswa dan melatih mahasiswa agar dapat mengenal situasi dunia kerja. Menurut Sinambela (2008), PKL merupakan upaya perguruan tinggi untuk mewujudkan profil lulusan program studi dan kompetensi yang lebih unggul dan siap kerja.

3.2 Sistem

Menurut Romney dan Steinbart (2015) sistem adalah rangkaian dari dua atau lebih komponen-komponen yang saling berhubungan, yang berinteraksi untuk mencapai suatu tujuan. Pendekatan sistem yang merupakan jaringan kerja dari suatu prosedur lebih menekankan dari urutan-urutan operasi dalam sistem.

3.2.1 Elemen Sistem

Kristanto (2008) menyatakan elemen yang terdapat dalam sistem terdiri dari tujuan sistem, batasan sistem, kontrol, input, proses, output, dan umpan balik. Elemen-elemen sistem tersebut akan membuat suatu siklus dimana akan menghasilkan output yang dianalisa menjadi umpan balik yang akan dipakai sebagai pertimbangan input selanjutnya.

3.2.2 Karakteristik Sistem

Menurut Jogiyanto (2005) karakteristik sistem yaitu:

1. Komponen Sistem

Sistem tersusun oleh berbagai komponen yang saling berhubungan dan bekerjasama membentuk suatu kesatuan. Komponen sistem disebut juga dengan subsistem dan merupakan bagian dari sistem yang berfungsi untuk menjalankan fungsi dan mempengaruhi keseluruhan proses dalam sistem.

2. Batasan Sistem

Sistem memiliki batasan yang membatasi sebuah sistem dengan sistem yang lain dan dengan lingkungan luar sistem. Batasan sistem memungkinkan sebuah sistem dipandang sebagai satu kesatuan.

3. Lingkungan luar sistem (*environment*)

Merupakan semua hal diluar batasan sistem yang dapat berpengaruh terhadap proses dari sebuah sistem. Lingkungan luar sistem bersifat merugikan ataupun menguntungkan, lingkungan yang menguntungkan harus tetap dipertahankan sedangkan lingkungan yang besifat merugikan harus dapat dikontrol supaya tidak mengganggu suatu sistem.

4. Penghubung Sistem (*interface*)

Berfungsi sebagai penghubung antara suatu subsistem lain. Penghubung sistem mengalirkan sumber daya dari subsistem ke subsistem lainnya.

5. Masukan Sistem

Masukan (*input*) merupakan daya yang dimasukkan ke dalam sistem. Input berupa pemeliharaan (*maintenance*) dan masukan sinyal (*signal input*). Masukan sinyal adalah daya yang diproses untuk didapatkan output atau keluaran sedangkan *maintenance* merupakan energi yang dimasukkan agar sistem dapat beroperasi.

6. Keluaran Sistem

Keluaran (*output*) adalah hasil olahan dari energi yang diklasifikasikan menjadi *output* yang berguna dan sisa-sisa pembuangan. Output dapat menjadi input untuk subsistem yang lain sedangkan informasi merupakan keluaran yang dibutuhkan.

7. Pengolah Sistem

Merupakan komponen sistem yang berfungsi memproses *input* menjadi *output*.

8. Sasaran Sistem

Sebuah sistem harus mempunyai sasaran (*objective*) dan tujuan (*goal*) sampai operasi sistem menjadi berguna, tujuan sistem sangat menentukan

input dan *output* yang dihasilkan. Suatu sistem dapat dikatakan berhasil jika dapat mencapai sebuah tujuan.

3.2.3 Komponen Sistem Informasi

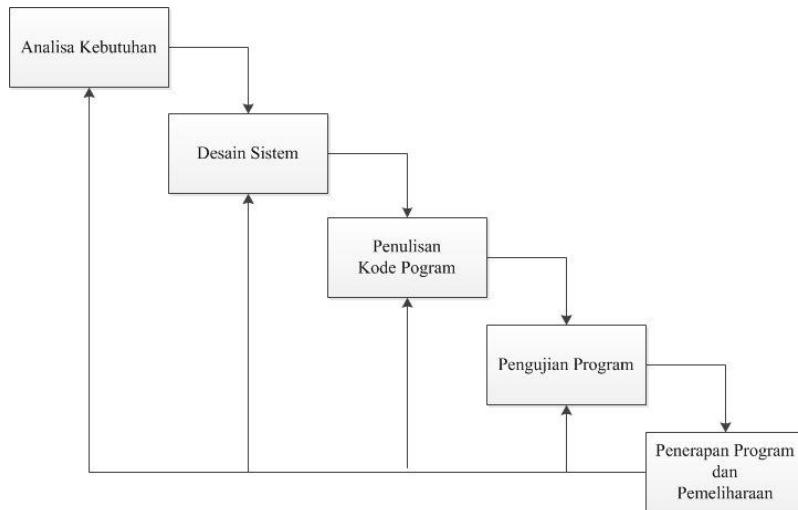
Menurut Anggraeni dan Irviani (2017) sistem informasi memiliki komponen berikut :

- a. Perangkat keras (*hardware*) yaitu seperti komputer, printer, dan lain lain.
- b. Perangkat lunak (*software*) atau program berisi macam-macam instruksi yang memungkinkan *hardware* untuk dapat mengolah data.
- c. Prosedur merupakan ketentuan pengolahan data dan pembangkitan *output* sesuai keinginan.
- d. Orang yaitu seluruh pihak yang memiliki tanggung jawab dalam mengembangkan sistem informasi, pengolahan, dan penggunaan hasil dari sistem informasi.
- e. Basisidata (*database*) berisi kumpulan tabel, relasi, dan penyimpanan data.
- f. Jaringan komputer dan komunikasi data merupakan sistem yang menghubungkan sumber (*resources*) agar dapat dipakai dan diakses oleh sejumlah orang.

3.3 Siklus Hidup Pengembangan Sistem

Menurut Djahir dan Pratita (2015) siklus hidup sistem merupakan penerapan pendekatan sistem, untuk tugas mengembangkan dan menggunakan sistem berbasis komputer.

Menurut Bassil (2012) ada berbagai metode pengembangan sistem, satu diantaranya yaitu metode waterfall. Metode waterfall merupakan proses pembangunan perangkat lunak dimana kemajuan dianggap mengalir kebawah (mirip seperti air terjun) melalui fase yang harus dijalankan untuk membangun perangkat lunak komputer.



Gambar 3. 1 Siklus Hidup Pengembangan Sistem Waterfall

Pada Gambar 3.1 metode pengembangan *waterfall* memiliki beberapa tahap, diantaranya adalah analisis kebutuhan, desain sistem, penulisan kode program, *testing*, implementasi program dan pemeliharaan. Setelah program selesai diuji dan berhasil, tahap yang dilakukan selanjutnya adalah *maintenance* yaitu pemeliharaan terhadap sistem.

3.3.1 Perancangan Sistem

Menurut Satzinger, Jackson dan Burd (2011) perancangan sistem merupakan gambaran detail cara kerja sistem. Berfungsi untuk menghasilkan perangkat lunak sesuai kebutuhan *user*.

3.3.2 Analisis Sistem

Menurut Anggraeni dan Irviani (2017) analisis sistem memiliki tujuan utama sebagai penentu aktivitas yang biasa sistem kerjakan. Analisis sistem terdiri dari studi kelayakan dan analisis kebutuhan. Menentukan keberhasilan dari solusi yang diusulkan merupakan fungsi dari studi kelayakan, sedangkan analisis kebutuhan dilakukan untuk menghasilkan spesifikasi kebutuhan.

3.3.3 Desain Sistem

Menurut Anggraeni dan Irviani (2017) desain sistem merupakan tahap setelah analisis dari pengembangan suatu sistem, pendefinisian dari kebutuhan

fungsional dan persiapan untuk implementasi. Desain sistem adalah tahap penggambaran perencanaan dan pembuatan dengan menyatukan beberapa fungsi yang sudah ditentukan.

3.3.4 Implementasi Sistem

Menurut Sutabri (2012) implementasi sistem adalah prosedur untuk menyelesaikan desain sistem di dokumentasi desain sistem yang disetujui dan menguji, termasuk menginstal dan memulai penggunaan sistem baru atau sistem yang diperbaiki.

3.4 Diagram Alir Data

Menurut Kristanto (2008) DFD (*Data Flow Diagram*) merupakan proses yang dibuat untuk menggambarkan darimana data berasal dan kemana tujuan data yang keluar dari sistem, dimana data disimpan, proses apa yang menghasilkan data tersebut dan interaksi antara data yang tersimpan dan proses yang dikenakan pada data tersebut.

3.5 Activity Diagram

Menurut Irwanto (2004) *Activity diagram* adalah penggambaran langkah sebuah use case yang digambarkan secara grafis dalam bentuk diagram. *Activity diagram* menjelaskan secara grafis alir kegiatan dalam sistem yang sedang dirancang, bagaimana awal dari masing-masing alir, keputusan yang mungkin terjadi, dan kegiatan berakhir seperti apa.

Tabel 3. 1 Simbol Activity Diagram

.No..	Simbol	Nama	Deskripsi
1.		<i>Initial Node</i>	Suatu aktivitas sistem diawali dan dibentuk
2.		<i>Decision</i>	Simbol untuk kondisi yang akan menghasilkan keputusan

3.		<i>Activity</i>	Aktivitas yang system lakukan yang biasanya diawali dengan kata kerja
4.		<i>Fork Node</i>	Perubahan dari satu aliran menjadi beberapa aliran pada tahap tertentu.
5.		<i>.Swimlane.</i>	Pengelompokan <i>activity</i> di dalam aktor yang sama untuk menjelaskan urutan aktivitas suatu proses dalam sebuah sistem
6.		<i>Activity Final</i>	Suatu aktivitas diakhiri

3.6 Basis Data

Menurut Sutanta (2004) basis data merupakan sekumpulan data yang mempunyai hubungan satu sama lain dan disimpan secara bersama pada suatu media, disimpan dengan cara-cara tertentu sehingga mudah untuk digunakan atau ditampilkan kembali. Basis data memiliki fungsi dalam mengembangkan sistem informasi. Tujuan dibentuknya basis data yaitu untuk penyimpanan data yang ditujukan oleh sistem informasi ke dalam tabel yang memiliki banyak baris dan kolom.

3.6.1 *Database Management System*

Menurut Connolly dan Carolyn (2002), *Database Management System* merupakan sistem perangkat lunak untuk mendefinisikan, membuat, memelihara, dan mengontrol akses ke database.

Menurut Connolly dan Carolyn (2002) *Database Management System* mempunya fasilitas sebagai berikut :

1. *Data Definition Language (DDL)*

Digunakan user untuk mendefinisikan tipe data, struktur, dan batasan data yang disimpan pada *database*.

2. *Data Manipulation Language (DML)*

Merupakan operasi *insert*, *update*, *delete*, dan mengambil data dari *database* (*retrieve*) yang dilakukan oleh user.

3. *Access control*

Penyedia kontrol akses ke *database* meliputi integrasi sistem, keamanan sistem, *concurrency control system*, kontrol perbaikan sistem dan katalog yang dapat diakses oleh user.

3.6.2 Entity Relationship Model

Connolly dan Carolyn (2002) menyatakan, *Entity Relationship Model* merupakan gambaran dari basis data. *ER Model* terdapat kumpulan entitas dan relasi. *ER Model* digambarkan dengan sebuah diagram yang disebut *Entity Relationship Diagram* (ERD). Komponen-komponen ERD, adalah :

1. Entitas

Simbol entitas adalah persegi panjang dan diberi nama entitas tersebut. Entitas merupakan objek yang mewakili sesuatu yang nyata dengan sifat yang sama.

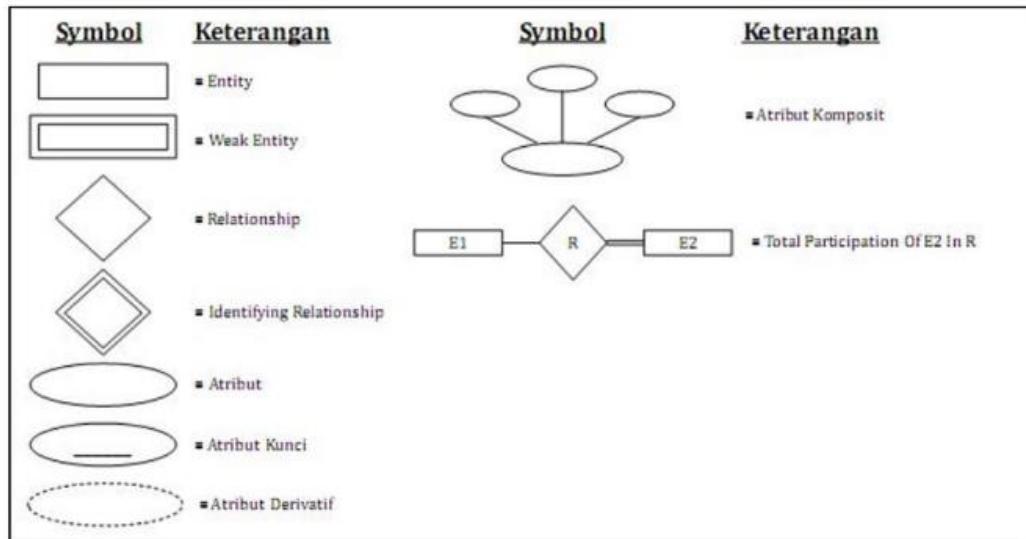
2. Relasi

Merupakan hubungan antar entitas yang memiliki makna. Simbol relasi adalah jajar genjang.

3. Atribut

Setiap entitas atau relasi mempunyai elemen yang disebut atribut. Atribut disimbolkan dengan elips dan di bagian dalamnya diberi nama. Atribut digunakan untuk menyimpan nilai yang mendefinisikan masing-masing entitas dari data yang tersimpan dalam basis data.

Komponen ERD terdapat dalam Gambar 3.2.



Gambar 3. 2 Komponen ERD

Menurut Nugroho (2011), ERD berfungsi untuk pembuatan model basis data yang dapat menggambarkan struktur dan batasan suatu basis data tersebut. Simbol komponen ERD yaitu:

- Persegi panjang, melambangkan entitas.
- Double persegi panjang, menggambarkan entitas lemah.
- Elips, melambangkan sebuah atribut.
- Double elips, mewakili atribut yang mempunyai berbagai nilai.
- Elips putus-putus, menggambarkan atribut turunan.
- Garis, adalah symbol hubungan antar atribut.
- Garis double, menggambarkan jumlah partisipasi suatu entitas dalam sebuah relasi.
- Belah ketupat, menggambarkan relasi antar entitas.

4. Kardinalitas

Kardinalitas menampilkan total maksimum entitas yang bisa berelasi dengan entitas yang lain, kardinalitas mengacu kepada hubungan maksimum dari himpunan entitas yang satu dengan entitas yang lain. Jenis kardinalitas ada tiga,yaitu:

- One to one* (1:1)

Setiap entitas pertama hanya memiliki satu hubungan pada entitas yang kedua.

b. *One to many/ many to one* (1:n / n:1)

Entitas pertama dapat berhubungan dengan lebih dari satu pada entitas yang kedua dan sebaliknya.

c. *Many to many* (n:n)

Pada entitas pertama memiliki banyak hubungan pada entitas yang kedua.

3.7 Internet

Menurut Ramadhani (2003) Internet atau inter-network merupakan istilah yang merujuk pada kumpulan jaringan komputer yang menghubungkan bermacam situs dan layanan yang diakses oleh pengguna yang tersebar di seluruh dunia, sekumpulan jaringan tersebut menggunakan protokol jaringan terstandar yang dikenal sebagai *Transmission Control Protocol/ Information Protocol* (TCP/IP) yang berguna sebagai pemantau lalu lintas jaringan. Jika komputer terkoneksi dengan jaringan internet, maka secara otomatis komputer tersebut memperoleh IP numerik dengan format tertentu sebagai tanda pengenal. Selain itu, internet mempunyai *Gateway* untuk mengakses jaringan dan layanan berbasis protokol lain.

3.7.1 Web

Menurut Ramadhani (2003) *web* merupakan sebuah sistem yang memuat informasi seperti gambar, tulisan, audio atau multimedia dan tersimpan dalam *web server* yang diubah menjadi bentuk *hypertext* kemudian dapat diakses melalui *web browser*.

3.7.2 Web Server

Menurut Solichin (2006) *web server* merupakan alat penerima *request* dari sisi client dan direspon kembali dengan halaman *hypertext*. *Web server* menggunakan protokol HTTP dan bertujuan untuk menyediakan dokumen web bagi *client* sehingga dapat diakses dengan mudah.

3.7.3 Web Browser

Ramadhani (2003) menyatakan, *web browser* merupakan perangkat lunak sisi *client*. Halaman yang tersimpan dalam *web server* kemudian dibaca melalui

protokol yang disebut dengan HTTP (*Hypertext Transfer Protocol*). Beberapa jenis diantaranya Google Chrome, Internet Explorer, Mozilla Firefox, Safari dan Opera.

3.8 Bahasa Pemrograman

Pengembangan ini menggunakan bahasa pemrograman HTML5, PHP, CSS3, *javascript*, *jQuery*, AJAX, MySQL dan *framework* Laravel.

3.8.1 HTML

Astamal (2006) menyatakan, HTML (Hyper Text Markup Language) adalah file yang berisi kode untuk menampilkan halaman pada web browser. Perintah yang terdapat dalam HTML memiliki pembuka <html> dan berakhiran dengan </html>.

3.8.2 PHP (*Hypertext Preprocessor*)

Supono dan Putratama (2016) mengemukakan bahwa PHP adalah suatu bahasa pemrograman yang digunakan untuk menerjemahkan baris kode program menjadi kode mesin yang dapat dipahami oleh komputer yang berbasis server-side yang dapat ditambahkan ke dalam HTML.

3.8.3 CSS3

Menurut Untung Raharja (2009) CSS adalah bahasa style sheet yang digunakan untuk mengatur tampilan dokumen. Dengan adanya CSS memungkinkan kita untuk menampilkan halaman yang sama dengan format yang berbeda. CSS membuat desain halaman *web* menjadi lebih baik dan efektif.

3.8.4 Javascript

Nixon (2009) menyatakan, *javascript* merupakan kode pemrograman yang sepenuhnya berjalan dalam browser. *Javascript* merupakan bahasa pemrograman yang bersifat client side yang pemrosesannya dilakukan oleh client dan digunakan pada web browser untuk menghasilkan halaman web yang menarik. *javascript* menggunakan tag <script> </script> dalam dokumen HTML.

3.8.5 jQuery

Menurut Bekti (2015) *jQuery* adalah *library* dalam *javascript* untuk membuat program halaman web tanpa menambahkan event atau property pada halaman web tersebut. *jQuery* berjalan pada HTML dengan tag pembuka dan penutup <script> </script>.

3.8.6 MySQL

Menurut Nixon (2009) SQL (*Structured Query Language*) adalah bahasa untuk permintaan akses dan kelola basis data. MySQL penggunaanya paling banyak dalam pemrograman web dengan bahasa pemrograman PHP.

3.8.7 Framework Laravel

Menurut Stauffer (2017) Laravel merupakan PHP *framework* dibawah lisensi MIT. Laravel disusun dengan konsep *Model View Controller* (MVC).

Model View Controller (MVC) adalah suatu teknologi yang memisahkan aplikasi logika dari presentasi. MVC memisahkan aplikasi bedasarkan komponen aplikasi, seperti: manipulasi data, *controller*, dan *user interface*.

1. *Model*, berisi fungsi-fungsi untuk mengelola basis data.
2. *View*, merupakan bagian yang berfungsi untuk menampilkan halaman web ke pengguna.
3. *Controller*, adalah bagian yang menjembatani antara *model* dan *view*.

3.8.8 JSON

Menurut Hakim dan Rizki (2012) *Java Script Object Notation* (JSON) adalah format pertukaran data yang ditemukan oleh Douglas Crockford pada tahun 2006 yang memiliki ukuran data yang lebih kecil serta waktu proses yang lebih cepat dibandingkan dengan XML.

3.8.9 Application Programming Interface (API)

API merupakan kumpulan dari perintah, fungsi, dan protokol yang digunakan oleh programmer saat mengembangkan perangkat lunak untuk sebuah sistem operasi. Dengan API, programmer dapat menggunakan fungsi standar untuk berinteraksi dengan sistem operasi (Wikipedia, 2018).

3.9 Alat Implementasi

Pengembangan sistem ini menggunakan beberapa alat dalam tahap implementasi hasil dari perencanaan, analisis dan desain.

3.9.1 Visual Studio Code

Dilansir dari code.visualstudio.com visual studio code merupakan sebuah teks editor untuk berbagai macam bahasa pemrograman seperti PHP yang dibuat

oleh Microsoft dan berguna untuk mengembangkan aplikasi berbasis web menggunakan ASP.NET dan Node.js, Visual Studio Code juga mendukung teknologi seperti HTML, CSS, JSON, Sass dan Less. Visual Studio Code memiliki banyak fitur diantaranya *IntelliSense*, *Debugging*, Integrasi dengan Git dan memiliki banyak ekstensi.

3.9.2 XAMPP

Menurut Riyanto (2010) dikutip dalam Choliviana, Triyono, & Sukadi (2012), XAMPP adalah paket PHP dan MySQL yang open source. Tool ini berguna sebagai alat pengembangan aplikasi berbasis PHP.

3.9.3 Balsamiq Mockup

Menurut Hanifah (2015) *Balsamiq Mockup* merupakan *software* untuk pembuatan desain atau *prototype* dalam pembuatan tampilan user *interface* sebuah aplikasi. Keunggulannya adalah ukuran tidak besar dan *user-friendly*.

3.9.4 Postman

Menurut Rohman (2017) Postman merupakan sebuah *REST Client* yang dapat digunakan untuk menguji *REST API*. Fungsi utama postman adalah sebagai GUI API Caller, selain itu terdapat fitur lain yaitu Testing API (free), Realtime Collaboration Team (paid), Sharing Collection API for Documentation (free), Monitoring API (paid), dan Integration (paid). Postman tersedia untuk beberapa platform, yaitu Windows, Linux dan MAC.

BAB IV

ANALISIS DAN PERANCANGAN SISTEM

4.1 Analisis Sistem

Analisis sistem merupakan pemahaman mengenai sistem yang akan dikembangkan. Fungsi dari tahapan ini adalah untuk mengidentifikasi masalah pada sistem dan untuk menentukan kebutuhan sistem yang akan dikembangkan. Berdasarkan tahapan penelitian yang telah dilakukan, didapat beberapa analisis yang digunakan untuk membangun Sistem Informasi Kerja Praktek Bagian Mahasiswa Program Studi Komputer dan Sistem Informasi.

4.1.1 Analisis Masalah

Pendataan kerja praktek masih dilakukan dengan cara konvensional. Mulai dari proses pendaftaran, pelaporan harian hingga penilaian masih dilakukan dengan mengisi data menggunakan kertas. Dalam melaksanakan kegiatan kerja praktek mahasiswa dalam pelaporan kegiatan harian dan presensi masih menggunakan buku yang diberikan dari pihak program studi.

Dari permasalahan tersebut diperlukan sebuah sistem yang memberikan kemudahan dalam melakukan proses pendaftaran hingga penilaian kerja praktek. Dengan adanya sistem tersebut diharapkan dapat memudahkan mahasiswa dalam proses pelaksanaan kerja praktek, sehingga dalam pelaksanaan kerja praktek lebih efektif dan efisien.

4.1.2 Analisis Kebutuhan Pengguna

User yang berkaitan dengan sistem ini adalah mahasiswa. Mahasiswa dapat membuat kelompok dan memilih anggota kelompok, melakukan pendaftaran kerja praktek dengan mengusulkan instansi atau dengan melamar lowongan yang tersedia. Selama kerja praktek, mahasiswa

menuliskan laporan kinerja harian pada menu laporan harian yang telah disediakan dalam sistem.

4.1.3 Analisis Kebutuhan Fungsional

Berdasarkan._penjelasan._permasalahan._yang._ada, sistem_ini_memiliki kebutuhan-fungsional-berdasarkan-penggunaanya, yaitu :

1. Mahasiswa dapat login ke dalam sistem.
2. Mahasiswa dapat mengubah profil.
3. Mahasiswa dapat mengubah password.
4. Mahasiswa dapat membuat kelompok.
5. Mahasiswa dapat melihat data kelompok.
6. Mahasiswa dapat menghapus undangan anggota kelompok.
7. Mahasiswa dapat melihat jadwal presentasi.
8. Mahasiswa dapat melihat data instansi tempat untuk kerja praktek.
9. Mahasiswa dapat melihat data dosen pembimbing.
10. Mahasiswa dapat melamar lowongan.
11. Mahasiswa dapat melihat lowongan.
12. Mahasiswa dapat mengajukan usulan instansi untuk kerja praktek.
13. Mahasiswa dapat melihat status usulan instansi untuk kerja praktek.
14. Mahasiswa dapat mengunggah laporan akhir.
15. Mahasiswa dapat mengedit laporan akhir.
16. Mahasiswa dapat menambahkan laporan harian.
17. Mahasiswa dapat mengedit laporan harian.
18. Mahasiswa dapat melihat pengumuman.
19. Mahasiswa dapat melihat jumlah mahasiswa dalam kelompoknya.
20. Mahasiswa dapat mengunggah CV.
21. Mahasiswa dapat memberi nilai kepada anggota kelompok dan dirinya sendiri.

4.1.4 Analisis Kebutuhan Nonfungsional

Kebutuhan non fungsional sistem adalah kebutuhan yang mendukung sistem. Kebutuhan nonsfungisional dari sistem ini sebagai berikut :

1. Sistem_berbasis_website.

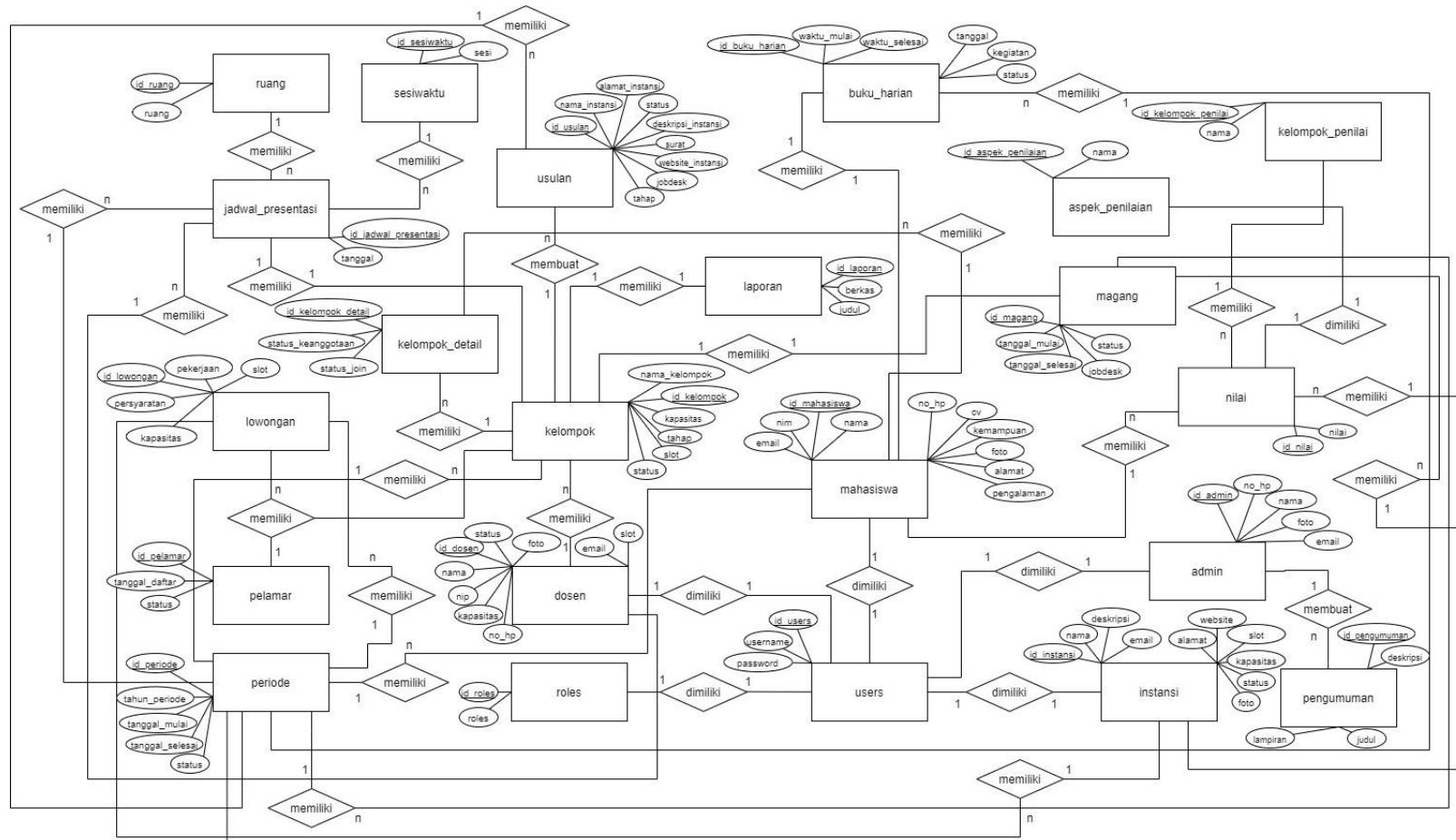
2. Sistem menggunakan framework Laravel 6.
3. Mahasiswa harus login terlebih dahulu untuk dapat menggunakan sistem.
4. File laporan akhir, cv, surat pemberitahuan instansi hanya dapat di unggah dalam bentuk (.pdf).
5. Mahasiswa hanya dapat bergabung dalam 1 kelompok.
6. Kelompok mahasiswa tidak dapat mengusulkan usulan instansi jika status usulan sebelumnya “disetujui” atau “diproses”.
7. Kelompok mahasiswa dapat mengusulkan usulan instansi jika status usulan instansi sebelumnya “ditolak”.
8. Mahasiswa yang melakukan usulan hanya mahasiswa dengan status “ketua”.
9. Mahasiswa yang dapat mengunggah laporan akhir hanya mahasiswa dengan status “ketua”.
10. Mahasiswa tidak dapat mengubah laporan harian jika status laporan harian “disetujui”.
11. Setiap mahasiswa yang membuat/ create kelompok otomatis menjadi ketua kelompok.
12. Penilaian mahasiswa memiliki range dari 1 sampai 5.

4.2 Perancangan Sistem

Perancangan sistem digunakan untuk menggambarkan rancangan bangun yang lengkap dalam merancang sistem. Terdapat empat perancangan dalam penelitian ini yaitu perancangan *database*, perancangan proses, perancangan struktur menu, dan perancangan antarmuka pengguna.

4.2.1 Perancangan Basis Data

Entity Relationship Diagram (ERD) digunakan dalam perancangan basis data sistem ini. ERD ini menjadi rujukan dalam membuat tabel basis data sistem informasi kerja praktik program studi komputer dan sistem informasi terdapat dalam Gambar 4.1.



Gambar 4. 1 ERD Sistem Informasi Kerja Praktek

Hasil dari rancangan ERD pada Gambar 4.1, dapat diimplementasikan menjadi 22 tabel, yaitu : tabel admin, dosen, mahasiswa, aspek_penilaian, buku_harian, jadwal_presentasi, kelompok, kelompok_detail, kelompok_penilai, laporan, lowongan, magang, nilai, pelamar, pengumuman, periode, roles, ruang, sesiwaktu, users, tabel usulan, dan tabel instansi.

1. Tabel admin

Tabel admin menyimpan data admin yang terdiri dari id_admin, id_users, nama, email, no_hp, foto, isDeleted, created_by, created_at, updated_at.

Rincian rancangan tabel admin terdapat dalam Tabel 4.1.

Tabel 4. 1 Rancangan tabel admin

Atribut	Tipe data	Panjang	Keterangan
id_admin	int	11	<i>Primary Key</i>
id_users	int	11	<i>Foreign Key</i> dari tabel users
nama	varchar	255	Nama Admin
email	varchar	255	Email Admin
no_hp	varchar	25	Nomor telepon admin
foto	text		Foto admin
isDeleted	tinyint	4	Keterangan apakah data dihapus (1) atau tidak (0)
created_by	int	11	User yang menambahkan data
created_at	timestamp		Waktu tambah data
updated_at	timestamp		Waktu update data

2. Tabel dosen

Tabel dosen menyimpan profil data dosen yang terdiri dari id_dosen, id_users, nama, nip, email, no_hp, slot, kapasitas, status, foto, isDeleted, created_by, created_at, updated_at._Rincian_rancangan-tabel dosen terdapat dalam-Tabel-4.2.

Tabel 4. 2 Rancangan tabel dosen

Atribut	Tipe data	Panjang	Keterangan
id_dosen	int	11	<i>Primary Key</i>
id_users	int	11	<i>Foreign Key</i> dari tabel user
nama	varchar	150	Nama-Dosen
nip	varchar	50	Nip Dosen
email	varchar	100	Email-Dosen
no_hp	varchar	20	No Telepon Dosen
slot	tinyint	4	Slot dosen untuk menerima kelompok
kapasistas	tinyint	4	Kapasitas maksimal dosen untuk menerima kelompok
status	enum('open', 'close')		Status dosen apakah slot tersedia(open) atau penuh (close)
foto	varchar	255	Foto dosen
isDeleted	tinyint	4	Keterangan apakah data dihapus (1) atau tidak (0)
created_by	int	11	User yang menambahkan data
created_at	datetime		Waktu tambah data
updated_at	datetime		Waktu update data

3. Tabel mahasiswa

Tabel mahasiswa menyimpan profil data mahasiswa yang terdiri dari id_mahasiswa, id_users, id_periode, nama, nim, angkatan, no_hp, email, alamat, kemampuan, pengalaman, foto, cv, isDeleted, created_by, created_at, updated_at. Rincian rancangan tabel mahasiswa terdapat dalam Tabel 4.3.

Tabel 4. 3 Rancangan tabel mahasiswa

Atribut	Tipe data	Panjang	Keterangan
id_mahasiswa	int	11	<i>Primary Key</i>
id_users	int	11	<i>Foreign Key</i> dari tabel users
id_periode	int	11	<i>Foreign key</i> dari tabel periode
nama	varchar	150	Nama mahasiswa
nim	varchar	20	NIM mahasiswa
angkatan	year	4	Angkatan mahasiswa
no_hp	varchar	20	Nomor Telepon mahasiswa
email	varchar	100	Email mahasiswa
alamat	text		Alamat mahasiswa
kemampuan	text		Kemampuan mahasiswa
pengalaman	text		Pengalaman mahasiswa
foto	varchar	255	Foto mahasiswa
cv	varchar	255	Cv mahasiswa
isDeleted	tinyint	4	Keterangan apakah data dihapus (1) atau tidak (0)
created_by	int	11	User yang menambahkan data
created_at	datetime		waktu tambah data
updated_at	datetime		waktu update data

4. Tabel aspek_penilaian

Tabel aspek_penilaian menyimpan data aspek penilaian yang terdiri dari sikap, keaktifan, kemampuan, kebersamaan, kerapihan, perhatian, kehadiran, keterkaitan, kesesuaian, sistematika, ketepatan, kekompakan. Rincian rancangan tabel aspek_penilaian dalam Tabel 4.4.

Tabel 4. 4 Rancangan tabel aspek penilaian

Atribut	Tipe data	Panjang	Keterangan
id_aspek_penilaian	int	11	<i>Primary Key</i>
nama	varchar	30	Nama aspek penilaian
isDeleted	tinyint	4	Keterangan apakah data dihapus (1) atau tidak (0)
created_by	int	11	User yang menambahkan data
created_at	datetime		waktu tambah data
updated_at	datetime		waktu update data

5. Tabel buku_harian

Tabel buku_harian merupakan tabel yang menyimpan data buku harian atau laporan harian mahasiswa yang melakukan kerja praktek. Tabel buku harian terdiri dari id_buku_harian, id_mahasiswa, id_periode, waktu_mulai, waktu_selesai, tanggal, kegiatan, status, isDeleted, created_by, created_at, updated_at. Rincian tabel buku_harian dalam Tabel 4.5.

Tabel 4. 5 Rancangan tabel buku_harian

Atribut	Tipe data	Panjang	Keterangan
id_buku_harian	int	11	<i>Primary Key</i>
id_mahasiswa	int	11	<i>Foreign Key</i> dari tabel mahasiswa
id_periode	int	11	<i>Foreign Key</i> dari tabel periode
waktu_mulai	time		waktu mulai kerja praktek
waktu_selesai	time		waktu selesai kerja praktek
tanggal	date		tanggal kerja praktek
kegiatan	text		kegiatan yang dilakukan mahasiswa saat kerja praktek
status	enum(status yang sedang diproses

	‘diproses’ , ‘diperiksa’)		atau sudah diperiksa oleh instansi
isDeleted	tinyint	4	Keterangan apakah data dihapus (1) atau tidak (0)
created_by	int	11	User yang menambahkan data
created_at	datetime		waktu tambah data
updated_at	datetime		waktu update data

6. Tabel jadwal_presentasi

Tabel jadwal_presentasi menyimpan data jadwal presentasi kelompok yang terdiri dari id_jadwal_presentasi, id_kelompok, id_periode, id_dospeng, id_sesiwaktu, id_ruang, tanggal, created_by, isDeleted, created_at, updated_at. Rincian tabel jadwal_presentasi dalam Tabel 4.6.

Tabel 4. 6 Rancangan tabel jadwal_presentasi

Atribut	Tipe data	Panjang	Keterangan
id_jadwal_presentasi	int	11	<i>Primary Key</i>
id_kelompok	int	11	<i>Foreign Key</i> dari tabel kelompok
id_periode	int	11	<i>Foreign Key</i> dari tabel periode
id_dospeng	int	11	<i>Foreign Key</i> dari tabel dosen
id_sesiwaktu	int	11	<i>Foreign key</i> dari tabel sesi waktu
id_ruang	int	11	<i>Foreign key</i> dari tabel ruang
tanggal	date		Tanggal presentasi
created_by	int	11	User yang menambahkan data
isDeleted	tinyint	4	Keterangan apakah data dihapus (1) atau tidak (0)

created_at	timestamp		Waktu tambah data
updated_at	timestamp		Waktu update data

7. Tabel kelompok

Tabel kelompok merupakan tabel yang menyimpan data kelompok yang akan melakukan kerja praktek. Tabel kelompok terdiri dari id_kelompok, id_periode, nama_kelompok, id_dosen, kapasitas, tahap, slot, status, isDeleted, created_by, created_at, updated_at. Rincian tabel kelompok dalam Tabel 4.7.

Tabel 4.7 Rancangan tabel kelompok

Atribut	Tipe data	Panjang	Keterangan
id_kelompok	int	11	<i>Primary Key</i>
id_periode	int	11	<i>Foreign Key</i> dari tabel periode
nama_kelompok	varchar	100	Nama kelompok
id_dosen	int	11	<i>Foreign key</i> dari tabel dosen
kapasitas	tinyint	4	kapasitas maksimal dalam kelompok
tahap	enum ('diproses', 'diterima', 'ditolak')		tahap status kelompok yang mendaftar
slot	tinyint	4	slot dalam kelompok
status	enum ('open', 'close')		Status kelompok apakah slot tersedia(open) atau penuh (close)
isDeleted	tinyint	4	Keterangan apakah data dihapus (1) atau tidak (0)
created_by	int	11	User yang menambahkan data
created_at	datetime		waktu tambah data
updated_at	datetime		waktu update data

8. Tabel kelompok_detail

Tabel kelompok_detail menyimpan data detail kelompok terdiri dari id_kelompok_detail, id_kelompok, id_mahasiswa, status_keanggotaan, status_join, created_by, isDeleted, created_at, updated_at. Rincian tabel kelompok_detail dalam Tabel 4.8.

Tabel 4.8 Rancangan tabel kelompok_detail

Atribut	Tipe data	Panjang	Keterangan
id_kelompok_detail	int	11	<i>Primary Key</i>
id_kelompok	int	11	<i>Foreign key</i> dari tabel kelompok
id_mahasiswa	int	11	<i>Foreign key</i> dari tabel mahasiswa
status_keanggotaan	enum ('ketua', 'anggota')		status keanggotaan mahasiswa dalam kelompok
status_join	enum ('diinvite', 'diterima', 'ditolak', 'join', 'create')		Status mahasiswa dalam kelompok
created_by	int	11	User yang menambahkan data
isDeleted	tinyint	4	Keterangan apakah data dihapus (1) atau tidak (0)
created_at	datetime		waktu tambah data
updated-at	datetime		waktu update data

9. Tabel kelompok_penilai

Tabel kelompok_penilai merupakan tabel yang menyimpan data kelompok penilai yang akan memberikan penilaian terhadap mahasiswa. Tabel

kelompok_penilai terdiri dari id_kelompok_penilai, nama, isDeleted, created_by, created_at, updated_at. Rincian tabel kelompok_penilai dalam Tabel 4.9

Tabel 4. 9 Rancangan tabel kelompok_penilai

Atribut	Tipe data	Panjang	Keterangan
id_kelompok_penilai	int	11	<i>Primary key</i>
nama	varchar	30	Nama kelompok penilai
bobot	int	11	Bobot penilaian
isDeleted	tinyint	4	Keterangan apakah data dihapus (1) atau tidak (0)
created_by	int	11	User yang menambahkan data
created_at	datetime		Waktu tambah data
updated_at	datetime		Waktu update data

10. Tabel laporan

Tabel laporan menyimpan data laporan akhir kelompok mahasiswa saat melaksanakan kerja praktek. Tabel laporan terdiri dari id_laporan, id_kelompok, judul, berkas, isDeleted, created_by, created_at, updated_at.

Rincian tabel laporan dalam Tabel 4.10.

Tabel 4. 10 Rancangan tabel laporan

Atribut	Tipe data	Panjang	Keterangan
id_laporan	.int	11	<i>Primary key</i>
id_kelompok	int	11	<i>Foreign Key</i> dari tabel kelompok
judul	varchar	150	Judul laporan
berkas	varchar	255	File/Berkas laporan
isDeleted	tinyint	4	Keterangan apakah data dihapus (1) atau tidak (0)
created_by	int	11	User yang menambahkan data

created_at	datetime		Waktu tambah data
updated_at	datetime		Waktu update data

11. Tabel lowongan

Tabel lowongan merupakan tabel yang akan menyimpan data lowongan yang diberikan oleh instansi kepada mahasiswa yang akan melaksanakan kerja praktik. Tabel lowongan terdiri dari id_lowongan, id_instansi,id_periode, pekerjaan, persyaratan, kapasitas, slot, isDeleted, created_by, created_at, updated_at. Rincian tabel lowongan dalam Tabel 4.11.

Tabel 4. 11 Rancangan tabel lowongan

Atribut	Tipe data	Panjang	Keterangan
id_lowongan	int	11	<i>Primary Key</i>
id_instansi	int	11	<i>Foreign key</i> dari tabel instansi
id_periode	int	11	<i>Foreign key</i> dari tabel periode
pekerjaan	varchar	100	pekerjaan dalam tawaran lowongan
persyaratan	text		persyaratan lowongan
kapasitas	tinyint	4	kapasitas maksimal lowongan
slot	tinyint	4	slot yang tersedia dalam lowongan
isDeleted	tinyint	4	Keterangan apakah data dihapus (1) atau tidak (0)
created_by	int	11	User yang menambahkan data
created_at	datetime		Waktu tambah data
updated_at	datetime		Waktu update data

12. Tabel magang

Tabel magang menyimpan data mahasiswa yang melakukan kerja praktek. Tabel magang terdiri dari id_magang, id_kelompok, id_instansi, id_periode, tanggal_mulai, tanggal_selesai, jobdesk, status, isDeleted, created_by, created_at, updated_at. Rincian tabel magang dalam Tabel 4.12.

Tabel 4. 12 Rancangan tabel magang

Atribut	Tipe data	Panjang	Keterangan
id_magang	int	11	<i>Primary Key</i>
id_kelompok	int	11	<i>Foreign key</i> dari tabel kelompok
id_instansi	int	11	<i>Foreign key</i> tabel instansi
id_periode	int	11	<i>Foreign key</i> dari tabel periode
tanggal_mulai	date		tanggal mulai kerja praktek
tanggal_selesai	date		tanggal selesai kerja praktek
jobdesk	text		jobdesk saat kerja praktek
status	enum (‘belum magang’ , ‘magang’ , ‘selesai’)		status mahasiswa dalam pelaksanaan kerja praktek
isDeleted	tinyint	4	Keterangan apakah data dihapus (1) atau tidak (0)
created_by	int	11	User yang menambahkan data
created_at	timestamp		Waktu tambah data
updated_at	timestamp		Waktu update data

13. Tabel nilai

Tabel nilai merupakan tabel yang akan menyimpan data nilai mahasiswa yang diberikan oleh kelompok penilai. Tabel nilai terdiri dari id_nilai, id_periode, id_mahasiswa, id_aspek_penilaian, nilai, isDeleted, created_by, created-at, updated-at. Rincian tabel nilai dalam Tabel 4.13.

Tabel 4. 13 Rancangan tabel nilai

Atribut	Tipe data	Panjang	Keterangan
id_nilai	int	11	<i>Primary Key</i>
id_periode	int	11	<i>Foreign key</i> dari tabel periode
id_mahasiswa	int	11	<i>Foreign key</i> dari tabel mahasiswa
id_aspek_penilaian	int	11	<i>Foreign key</i> dari tabel aspek penilaian
id_kelompok_penilaian	int	11	<i>Foreign key</i> dari tabel kelompok penilaian
nilai	float		nilai mahasiswa
isDeleted	tinyint	4	Keterangan apakah data dihapus (1) atau tidak (0)
created_by	int	11	User yang menambahkan data
created_at	datetime		waktu tambah data
updated_at	datetime		waktu update data

14. Tabel pelamar

Tabel pelamar merupakan tabel yang akan menyimpan data mahasiswa yang melamar lowongan yang diberikan oleh instansi. Tabel pelamar terdiri dari id_pelamar, id_lowongan, id_kelompok, tanggal_daftar, status, created_by, isDeleted, created_at, updated_at. Rincian tabel pelamar dalam Tabel 4.14.

Tabel 4. 14 Rancangan tabel pelamar

Atribut	Tipe data	Panjang	Keterangan
id_pelamar	int	11	<i>Primary key</i>
id_lowongan	int	11	<i>Foreign key</i> dari tabel lowongan
id_kelompok	int	11	<i>Foreign key</i> dari tabel kelompok
tanggal_daftar	date		Tanggal saat mendaftar lowongan

status	enum (‘melamar’ , ‘diterima’ , ‘ditolak’)		Status mahasiswa yang melamar lowongan
created_by	int	11	User yang menambahkan data
isDeleted	tinyint	4	Keterangan apakah data dihapus (1) atau tidak (0)
created_at	datetime		Waktu tambah data
updated_at	datetime		Waktu update data

15. Tabel pengumuman

Tabel pengumuman menyimpan data pengumuman tentang penyelenggaraan kerja praktek. Tabel pengumuman terdiri dari id_pengumuman, judul, deskripsi, lampiran, isDeleted, created_by, created_at, updated_at. Rincian tabel pengumuman dapat dilihat dalam Tabel 4.15.

Tabel 4. 15 Rancangan tabel pengumuman

Atribut	Tipe data	Panjang	Keterangan
id_pengumuman	int	11	<i>Primary Key</i>
judul	varchar	150	Judul pengumuman
deskripsi	text		Isi dari pengumuman
lampiran	varchar	255	Lampiran pengumuman
isDeleted	tinyint	4	Keterangan apakah data dihapus (1) atau tidak (0)
created_by	int	11	User yang menambahkan data
created_at	datetime		Waktu tambah data
updated_at	datetime		Waktu update data

16. Tabel periode

Tabel periode menyimpan data periode untuk men-setting periode kerja praktek. Tabel periode terdiri dari id_periode, tahun_periode, tgl_mulai, tgl_selesai, status, isDeleted, created_by, created_at, updated_at. Rincian tabel periode dalam Tabel 4.16.

Tabel 4. 16 Rancangan tabel periode

Atribut	Tipe data	Panjang	Keterangan
id_periode	int	11	<i>Primary Key</i>
tahun_periode	int	11	tahun periode kerja praktek
tgl_mulai	date		tanggal periode mulai kerja praktek
tgl_selesai	date		tanggal periode selesai kerja praktek
status	enum (‘open’ , ‘close’)		Status periode
isDeleted	tinyint	4	Keterangan apakah data dihapus (1) atau tidak (0)
created_by	int	11	User yang menambahkan data
created_at	datetime		Waktu tambah data
updated_at	datetime		Waktu update data

17. Tabel roles

Tabel roles menyimpan kategori user. Tabel roles terdiri dari id_roles dan roles. Rincian tabel roles dalam Tabel 4.17.

Tabel 4. 17 Rancangan tabel roles

Atribut	Tipe data	Panjang	Keterangan
id_roles	tinyint	4	<i>Primary Key</i>

roles	varchar	20	Nama kategori user
-------	---------	----	--------------------

18. Tabel ruang

Tabel ruang menyimpan data ruang atau tempat yang akan digunakan untuk presentasi kelompok. Tabel ruang terdiri dari id_ruang, ruang, isDeleted, created_by, created_at, updated_at. Rincian tabel ruang dalam Tabel 4.18.

Tabel 4. 18 Rancangan tabel ruang

Atribut	Tipe data	Panjang	Keterangan
id_ruang	int	11	<i>Primary Key</i>
ruang	varchar	255	Nama ruang untuk presentasi
isDeleted	tinyint	4	Keterangan apakah data dihapus (1) atau tidak (0)
created_by	int	11	User yang menambahkan data
created_at	timestamp		Waktu tambah data
updated_at	timestamp		Waktu ubah data

19. Tabel sesiwaktu

Tabel sesiwaktu menyimpan data waktu untuk presentasi kelompok. Tabel sesi waktu tersiri dari id_sesiwaktu, sesi, isDeleted, created_by, created_at, updated_at. Rincian tabel sesiwaktu dalam Tabel 4.19.

Tabel 4. 19 Rancangan tabel sesiwaktu

Atribut	Tipe data	Panjang	Keterangan
id_sesiwaktu	int	11	<i>Primary Key</i>
sesi	time		Sesi waktu presentasi
isDeleted	tinyint	4	Keterangan apakah data dihapus (1) atau tidak (0)
created_by	int	11	User yang menambahkan data
created_at	timestamp		Waktu tambah data

updated_at	timestamp		Waktu update data
------------	-----------	--	-------------------

20. Tabel users

Tabel users berfungsi sebagai penyimpanan data akun user. Tabel users terdiri dari id_users, username, password, id_roles, api_token, isDeleted, created_by, updated_by, created_at, updated_at. Rincian tabel users dalam Tabel 4.20.

Tabel 4. 20 Rancangan tabel users

Atribut	Tipe data	Panjang	Keterangan
id_users	int	11	<i>Primary Key</i>
username	varchar	12	<i>Username</i>
password	varchar	255	<i>password hash</i>
id_roles	tinyint	4	<i>Foreign key</i> dari tabel roles
api_token	text		api token user
isDeleted	tinyint	4	Keterangan apakah data dihapus (1) atau tidak (0)
created_at	datetime		Waktu tambah data
updated_at	datetime		Waktu update data
created_by	int	11	User yang menambah data
updated_by	int	11	User yang mengupdate data

21. Tabel usulan

Tabel usulan digunakan untuk menyimpan data usulan kelompok mahasiswa yang ingin melakukan kerja praktik. Tabel usulan terdiri dari id_usulan, id_kelompok, id_periode, nama_instansi, deskripsi_instansi, alamat_instansi, website_instansi, jobdesk, surat, status, tahap, created_by, created_at, updated_at. Rincian tabel usulan dalam Tabel 4.21.

Tabel 4. 21 Rancangan tabel usulan

Atribut	Tipe data	Panjang	Keterangan
id_usulan	int	11	<i>Primary Key</i>
id_kelompok	int	11	<i>foreign key</i> dari tabel kelompok
id_periode	int	11	<i>foreign key</i> dari tabel periode
nama_instansi	varchar	150	nama instansi
deskripsi_instansi	text		Isi penjelasan singkat instansi
alamat_instansi	text		alamat instansi
website_instansi	varchar	100	website instansi
jobdesk	text		Jobdesk mahasiswa saat melakukan kerja praktek
surat	text		Surat pemberitahuan dari instansi
status	enum ('diproses', 'diterima', 'ditolak')		Status usulan
isDeleted	tinyint	4	Keterangan apakah data dihapus (1) atau tidak (0)
created_by	int	11	User yang menambah data
created_at	datetime		waktu tambah data
updated_at	datetime		waktu update data

22. Tabel instansi

Tabel instansi berfungsi sebagai penyimpanan data instansi. Tabel instansi terdiri dari id_instansi, id_users, nama, deskripsi , email, alamat, website, slot, kapasitas, status, foto , isBlacklist, isDeleted, created_by, created_at, updated_at. Rincian tabel instansi terdapat dalam Tabel 4.22.

Tabel 4. 22 Rancangan tabel instansi

Atribut	Tipe data	Panjang	Keterangan
id_instansi	int	11	<i>Primary Key</i>
id_users	int	11	<i>Foreign key</i> dari tabel users
nama	varchar	150	Nama instansi
deskripsi	text		Deskripsi singkat instansi
email	varchar	100	Email Instansi
alamat	text		Alamat instansi
website	varchar	100	Website instansi
slot	tinyint	4	Slot instansi untuk menerima kelompok
kapasitas	tinyint	4	Kapasitas maksimal instansi untuk menerima kelompok
status	enum ('open', 'close')		Status instansi apakah slot tersedia (open) atau penuh (close)
foto	varchar	255	Foto instansi
isDeleted	tinyint	4	Keterangan apakah data dihapus (1) atau tidak (0)
created_by	int	11	User yang menambah data
created_at	datetime		Waktu tambah data
updated_at	datetime		Waktu update data

4.2.2 Perancangan Proses

Merupakan gambaran dari suatu tahapan pada sistem yang dibangun dan menghasilkan informasi yang benar. Perancangan proses dalam pembangunan sistem ini dibuat dengan UML yang digambarkan dengan use case diagram dan activity diagram.

a. Use case diagram

Penggambaran skenario sistem dari sisi pengguna. Use case diagram dalam penelitian ini terdapat 1 aktor yaitu mahasiswa terdapat dalam Gambar 4.2.



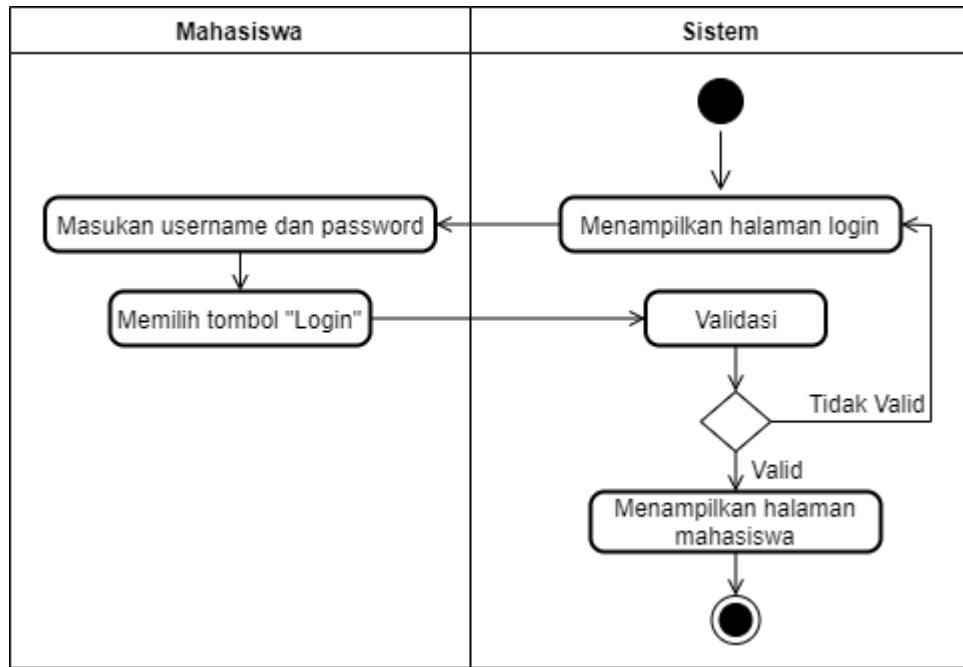
Gambar 4. 2 Diagram usecase

b. Activity diagram

Menggambarkan langkah suatu aktivitas pada sistem, dari mulai aktivitas tersebut sampai selesai. *Activity* diagram merupakan penjelasan suatu langkah dalam use case. Berikut adalah uraian mengenai *activity* diagram pada sistem ini.

1. *Activity* diagram login sistem

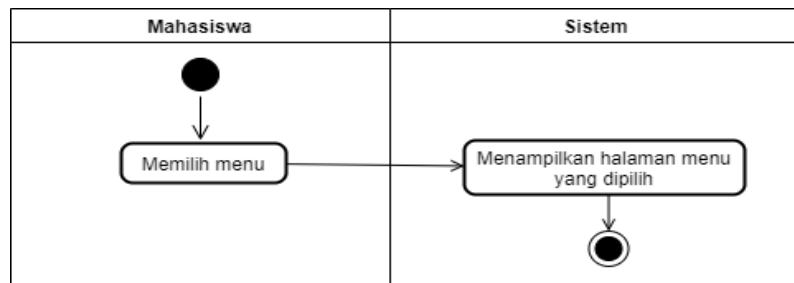
Activity diagram login sistem merupakan kegiatan mahasiswa untuk memasuki sistem menggunakan username dan password. *Activity* diagram login sistem dapat dilihat dalam Gambar 4.3.



Gambar 4. 3 Activity diagram login sistem

2. *Activity* diagram melihat halaman menu

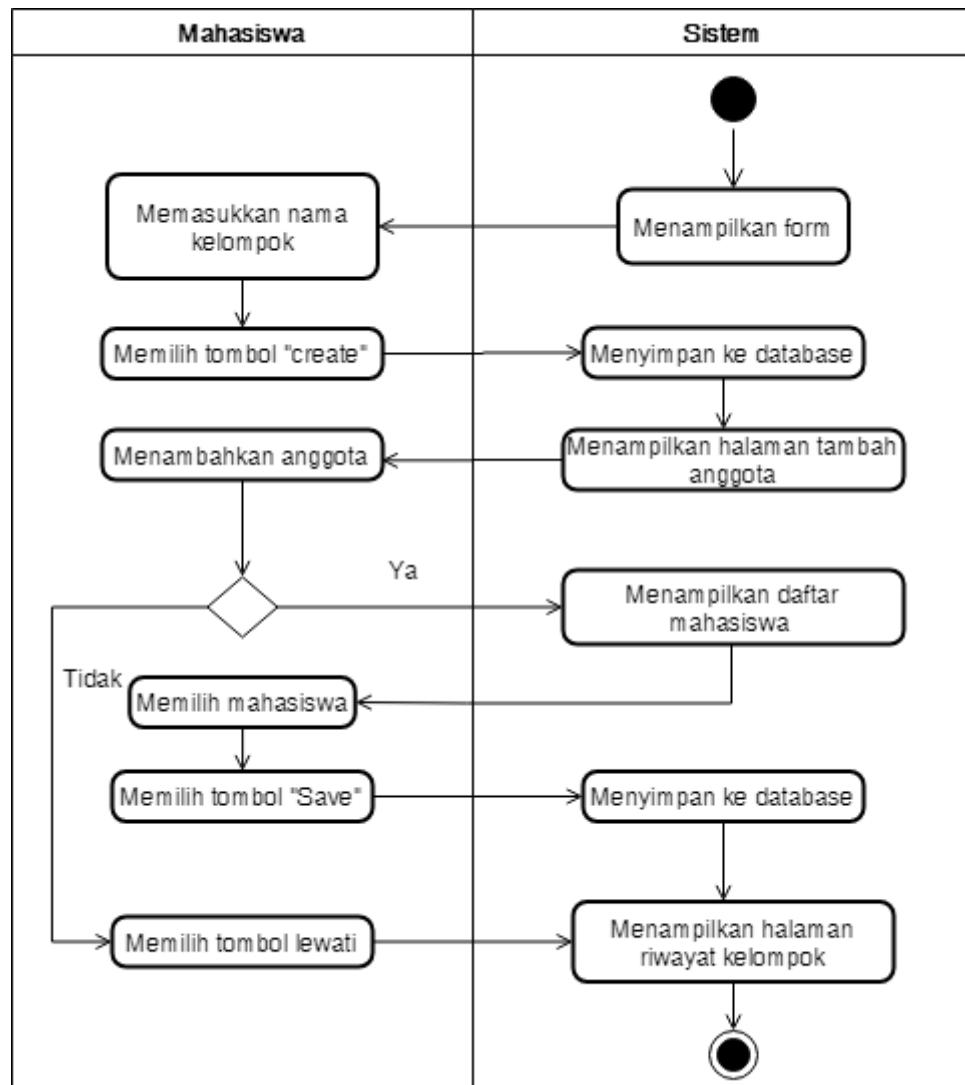
Activity diagram melihat halaman menu merupakan kegiatan dari mahasiswa untuk melihat berbagai halaman menu seperti melihat halaman kelompok, pendaftaran, laporan, dan pengumuman. *Activity* diagram melihat halaman menu terdapat dalam Gambar 4.4.



Gambar 4. 4 Activity diagram melihat halaman menu

3. *Activity* diagram buat kelompok

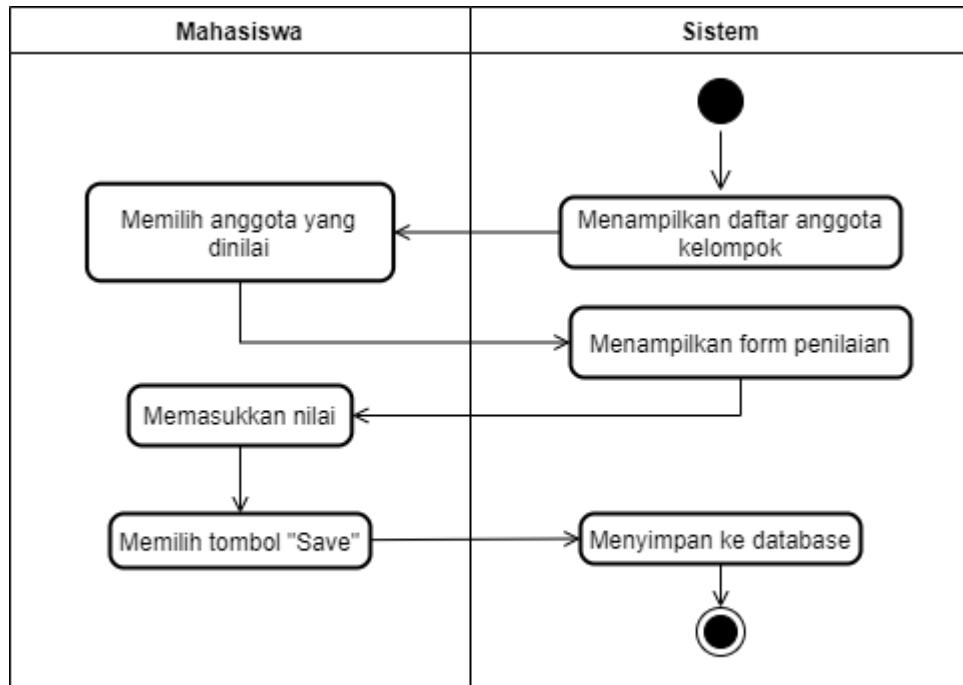
Activity diagram buat kelompok merupakan kegiatan dari mahasiswa untuk membuat kelompok kerja praktik. *Activity* diagram buat kelompok terdapat dalam Gambar 4.5.



Gambar 4. 5 Activity diagram buat kelompok

4. *Activity* diagram penilaian anggota

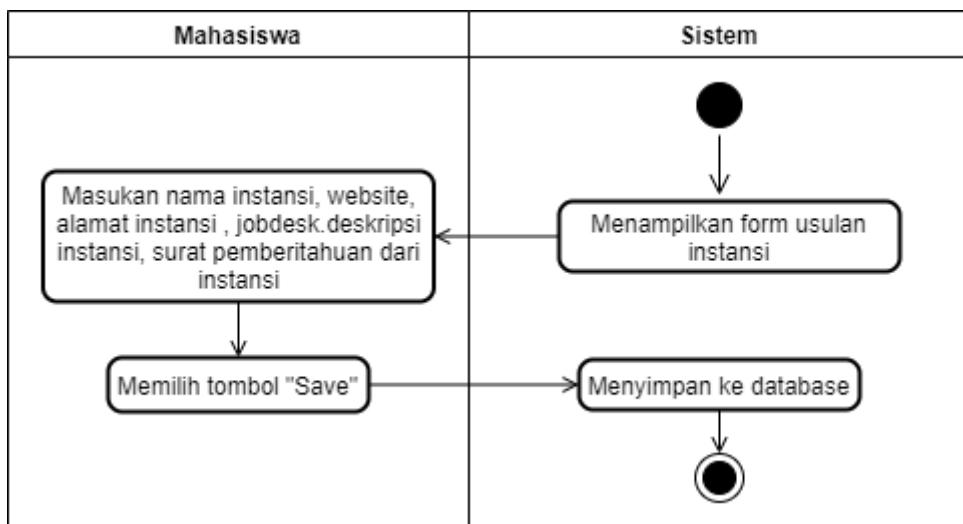
Dalam *activity* diagram ini menjelaskan penilaian yang diberikan kepada anggota kelompok yang telah melaksanakan kerja praktek. *Activity* diagram penilaian anggota dapat dilihat dalam Gambar 4.6.



Gambar 4. 6 Activity diagram penilaian anggota

5. Activity diagram usulan instansi

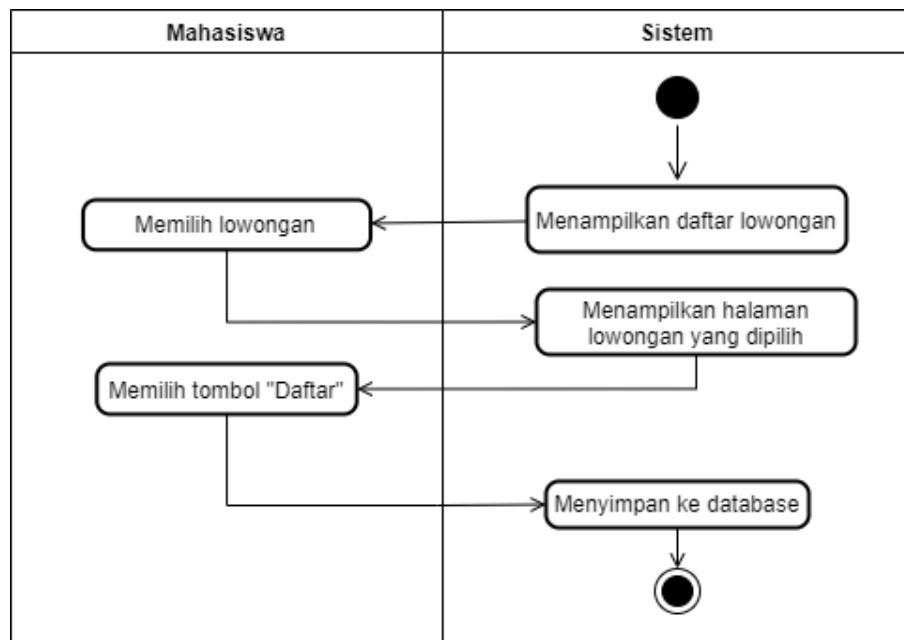
Dalam *activity* diagram ini mahasiswa dapat mengajukan usulan instansi untuk kerja praktek. *Activity* diagram usulan instansi terdapat dalam Gambar 4.7.



Gambar 4. 7 Activity diagram usulan instansi

6. Activity diagram melamar lowongan

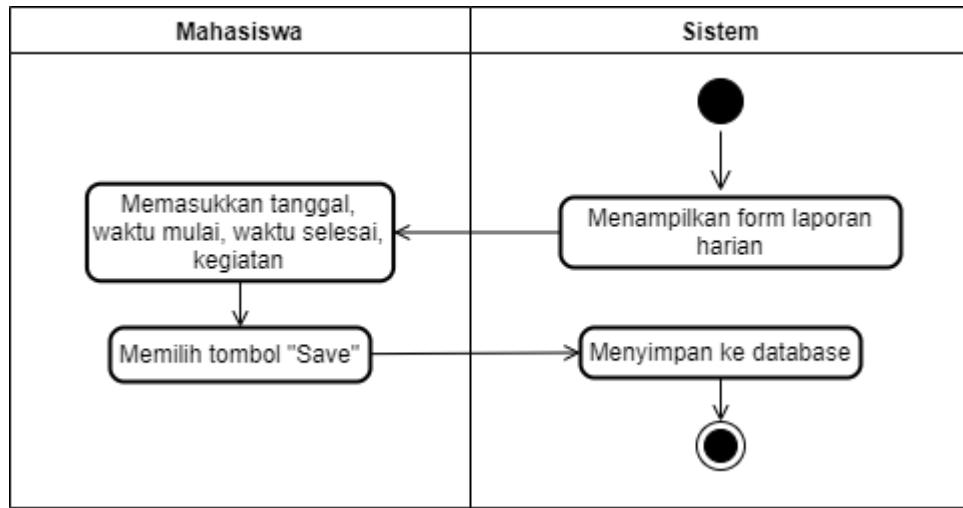
Dalam *activity* diagram ini mahasiswa dapat melamar lowongan kerja praktek yang tersedia. *Activity* diagram usulan instansi terdapat dalam Gambar 4.8.



Gambar 4. 8 Activity diagram melamar lowongan

7. Activity diagram laporan harian

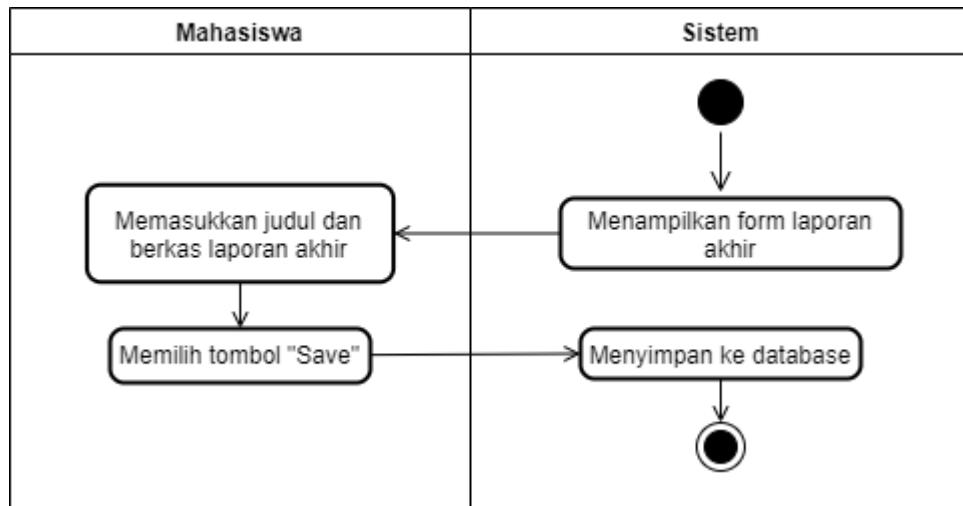
Dalam *activity* diagram ini menjelaskan mahasiswa melakukan pengisian laporan kinerja harian/laporan harian pada saat kerja praktek. *Activity* diagram laporan harian dapat dilihat dalam Gambar 4.9.



Gambar 4. 9 Activity diagram laporan harian

8. Activity diagram laporan akhir

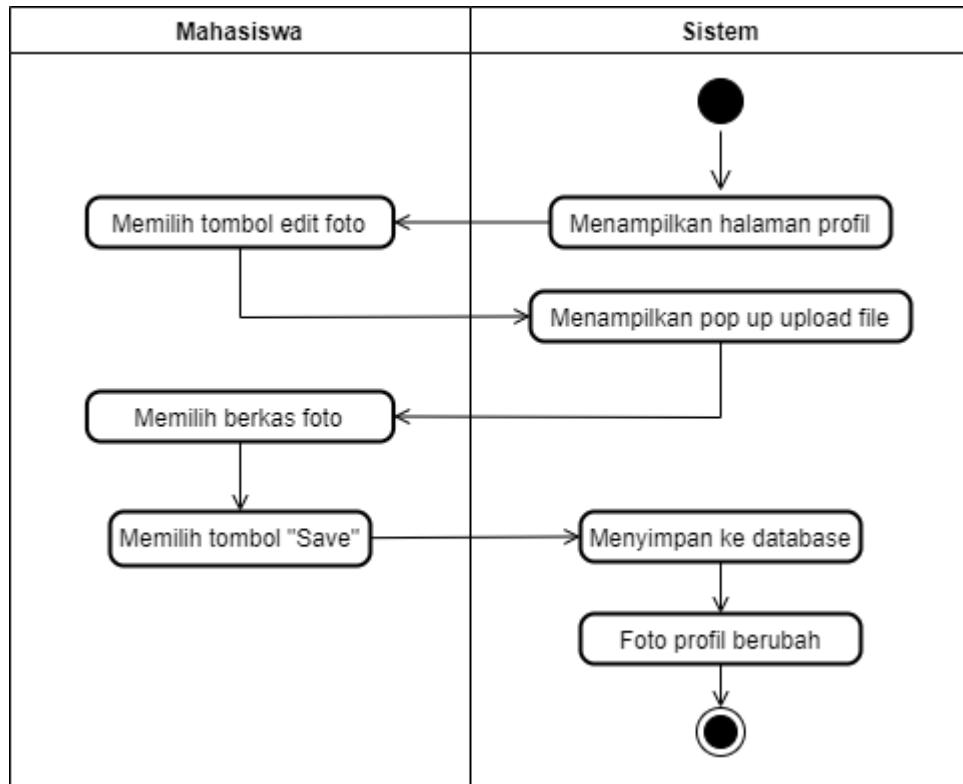
Dalam *activity diagram* ini menjelaskan mahasiswa dapat mengunggah laporan akhir kerja praktek. *Activity diagram* laporan akhir terdapat dalam Gambar 4.10.



Gambar 4. 10 Activity diagram laporan akhir

9. Activity diagram ubah avatar

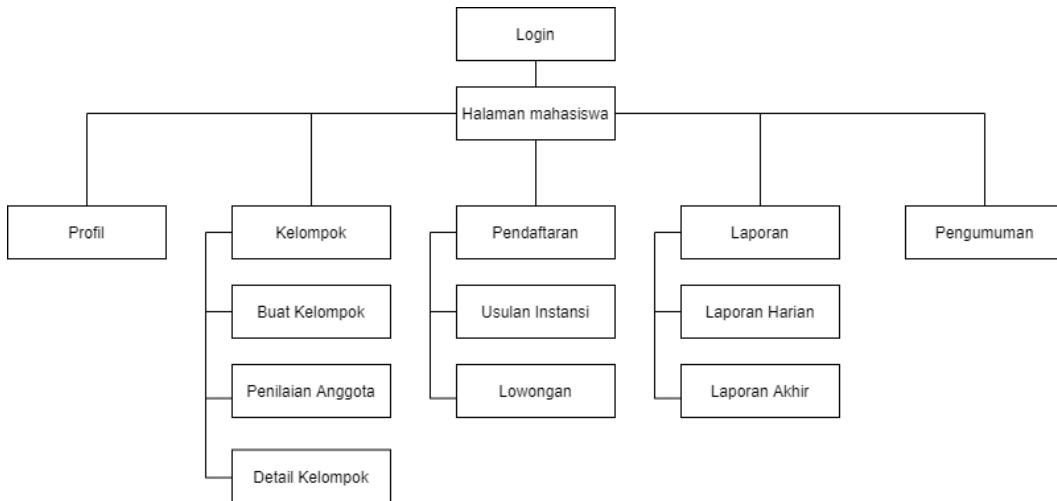
Activity diagram ubah avatar merupakan kegiatan mahasiswa mengubah foto profil atau avatar. *Activity diagram* ubah avatar terdapat dalam Gambar 4.11.



Gambar 4. 11 Activity diagram ubah avatar

4.2.3 Perancangan Struktur Menu

Sistem informasi kerja praktek bagian mahasiswa pada program studi komputer dan sistem informasi memiliki satu pengguna yaitu mahasiswa. Rancangan struktur menu mahasiswa dapat mengakses profil mahasiswa, kelompok, pendaftaran, laporan , dan pengumuman setelah melakukan login ke dalam sistem. Rancangan struktur menu Mahasiswa dalam Gambar 4.12.



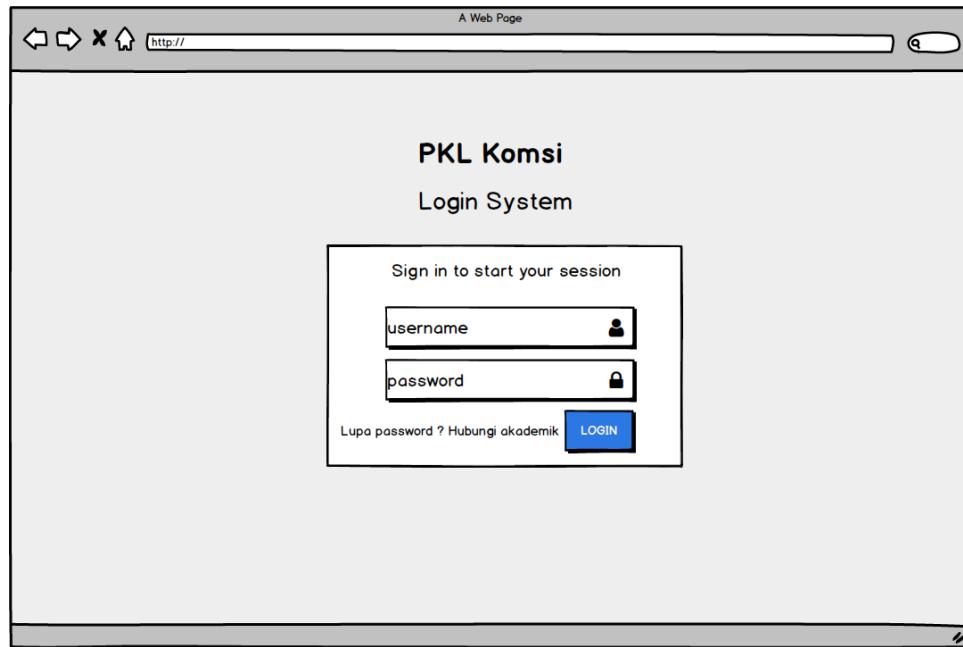
Gambar 4. 12 Rancangan struktur menu

4.2.4 Perancangan Antarmuka Pengguna

Perancangan antarmuka adalah media yang menjadi penghubung interaksi antara sistem dengan pengguna. Rancangan antarmuka menunjukkan hubungan antara pengguna sistem dengan komputer secara visual.

1. Rancangan-antarmuka halaman login

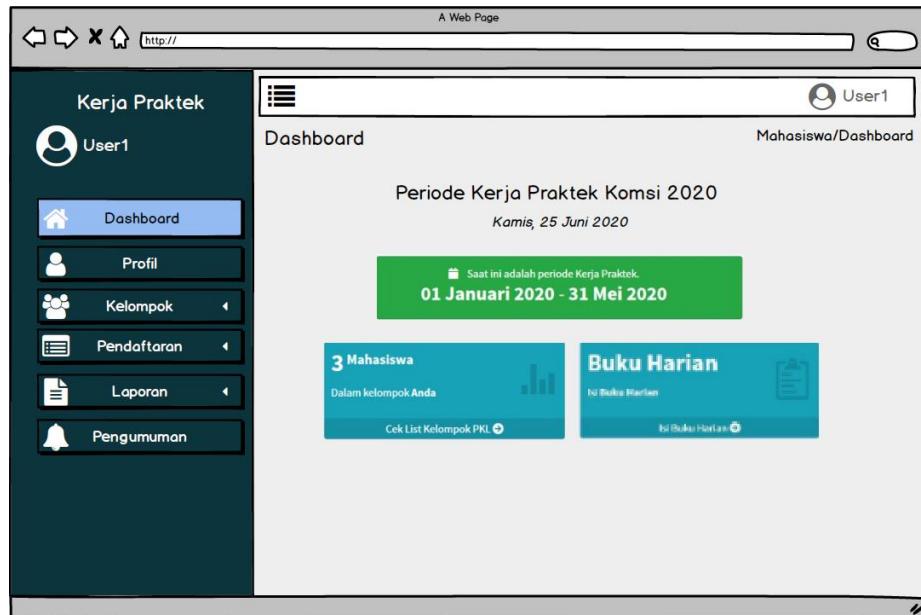
Halaman login merupakan halaman utama ketika mengakses sistem informasi kerja praktek bagian mahasiswa. Pada halaman ini terdapat kolom *username* dan *password*, dan tombol login. Pengguna diharuskan menginputkan *username* dan *password* dengan benar supaya dapat masuk ke sistem. Rancangan antarmuka halaman login dalam Gambar 4.13.



Gambar 4. 13 Rancangan antarmuka halaman login

2. Rancangan antarmuka halaman dashboard

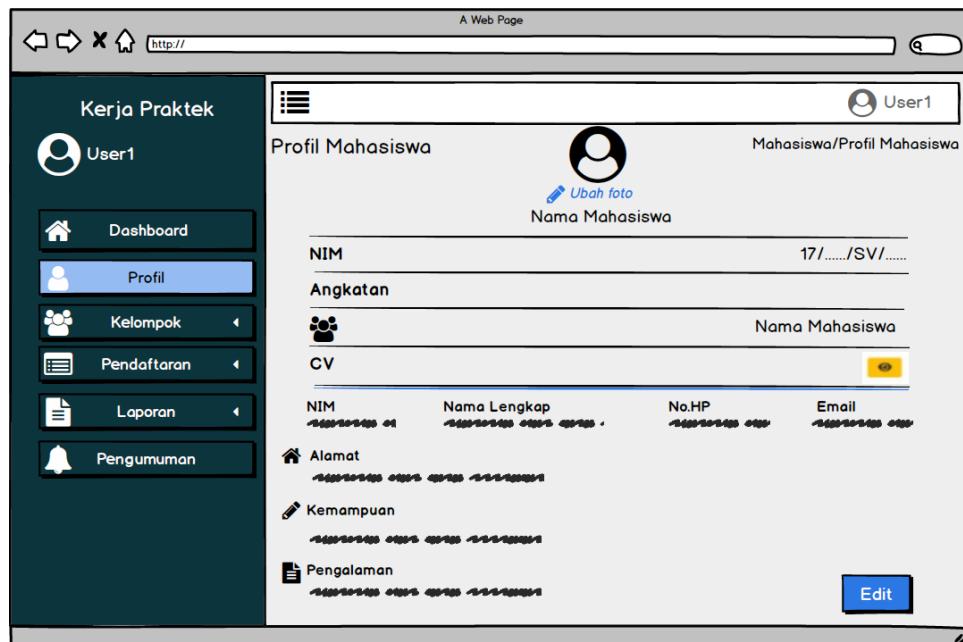
Sistem akan menampilkan halaman *dashboard* ketika mahasiswa sukses login. Halaman *dashboard* menampilkan periode kerja praktek, informasi jumlah mahasiswa yang berada dalam kelompok, dan menu pintas untuk mengisi buku harian. Rancangan antarmuka halaman dashboard dalam Gambar 4.14.



Gambar 4. 14 Rancangan antarmuka halaman dashboard

3. Rancangan antarmuka halaman profil

Halaman profil menampilkan informasi profil mahasiswa yaitu foto profil, nama, nim, angkatan, cv, alamat, nomor telepon, email, kemampuan(*skill*), dan pengalaman yang dapat dilihat dalam Gambar 4.15.



Gambar 4. 15 Rancangan antarmuka halaman profil

4. Rancangan antarmuka halaman edit profil

Halaman edit profil adalah halaman untuk mengubah informasi profil seperti foto profil, nama, nim, cv, alamat, nomor telepon, email, kemampuan(*skill*), dan pengalaman yang terdapat dalam Gambar 4.16.

A screenshot of a web application interface titled 'Edit Profil Mahasiswa'. The left sidebar, titled 'Kerja Praktek' and 'User1', contains navigation links: Dashboard, Profil, Kelompok, Pendaftaran, Laporan, and Pengumuman. The main content area shows fields for 'CV' (with a 'Telusuri...' button), 'NIM' (input field), 'Nama Lengkap' (input field), 'Angkatan' (input field), 'No.HP' (input field), 'Email' (input field), 'Alamat' (input field), 'Kemampuan' (input field), and 'Pengalaman' (input field). A blue 'Save' button is located at the bottom right.

Gambar 4. 16 Rancangan antarmuka halaman edit profil

5. Rancangan antarmuka halaman buat kelompok

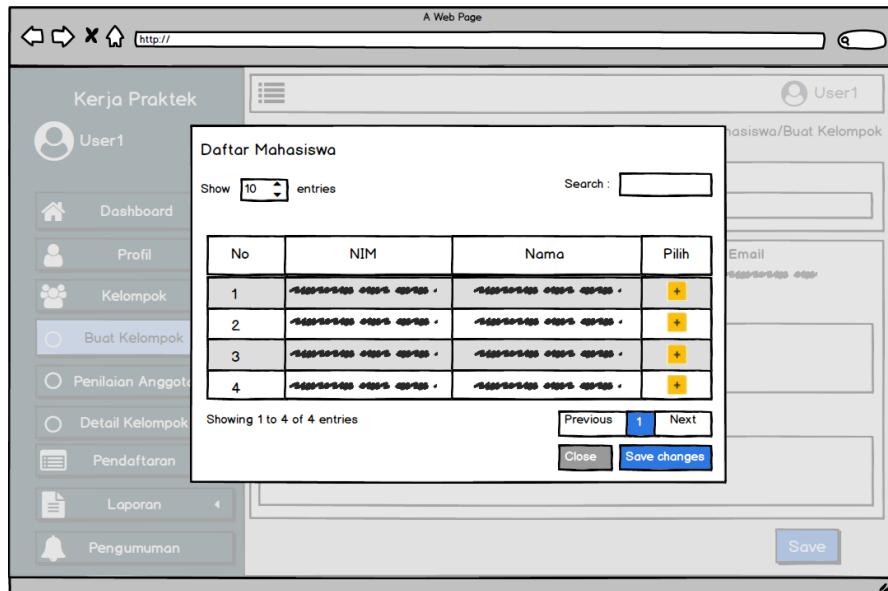
Halaman buat kelompok digunakan untuk membuat kelompok kerja praktek

A screenshot of a web application interface titled 'Buat Kelompok'. The left sidebar, titled 'Kerja Praktek' and 'User1', contains navigation links: Dashboard, Profil, Kelompok (selected), Buat Kelompok (selected), Penilaian Anggota, Detail Kelompok, Pendaftaran, Laporan, and Pengumuman. The main content area shows a 'Buat Kelompok' button and a 'Nama Kelompok*' input field. A blue 'Create' button is located at the bottom right. A link '>cek riwayat kelompok' is also present.

Gambar 4. 17 Rancangan halaman buat kelompok

6. Rancangan antarmuka halaman tambah anggota

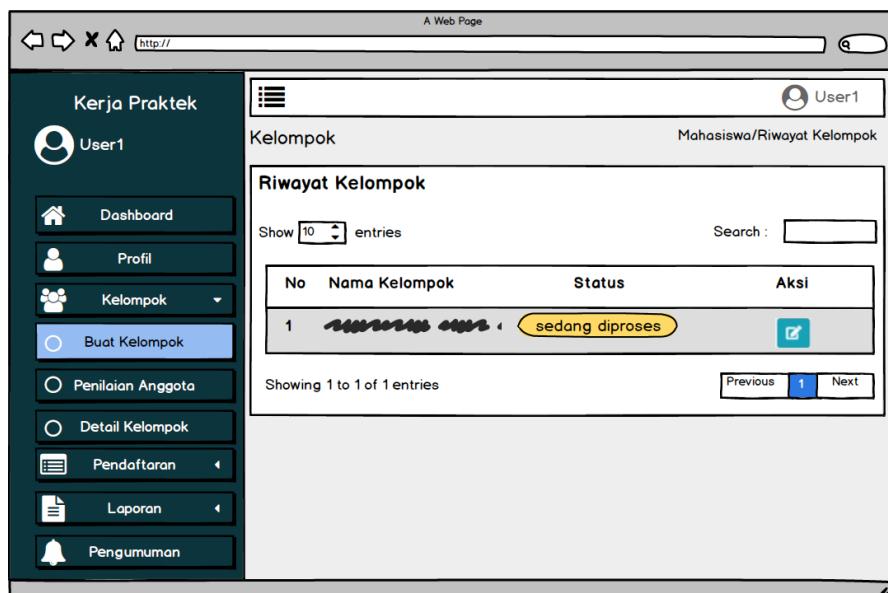
Halaman tambah anggota berfungsi untuk menambahkan anggota kelompok kerja praktek. Pemilihan anggota kelompok dilakukan dengan memilih daftar mahasiswa yang tersedia.



Gambar 4. 18 Rancangan antarmuka halaman tambah anggota

7. Rancangan antarmuka halaman riwayat kelompok

Halaman riwayat kelompok menampilkan daftar kelompok yang pernah didaftarkan.



Gambar 4. 19 Rancangan antarmuka halaman riwayat buat kelompok

8. Rancangan antarmuka daftar anggota kelompok

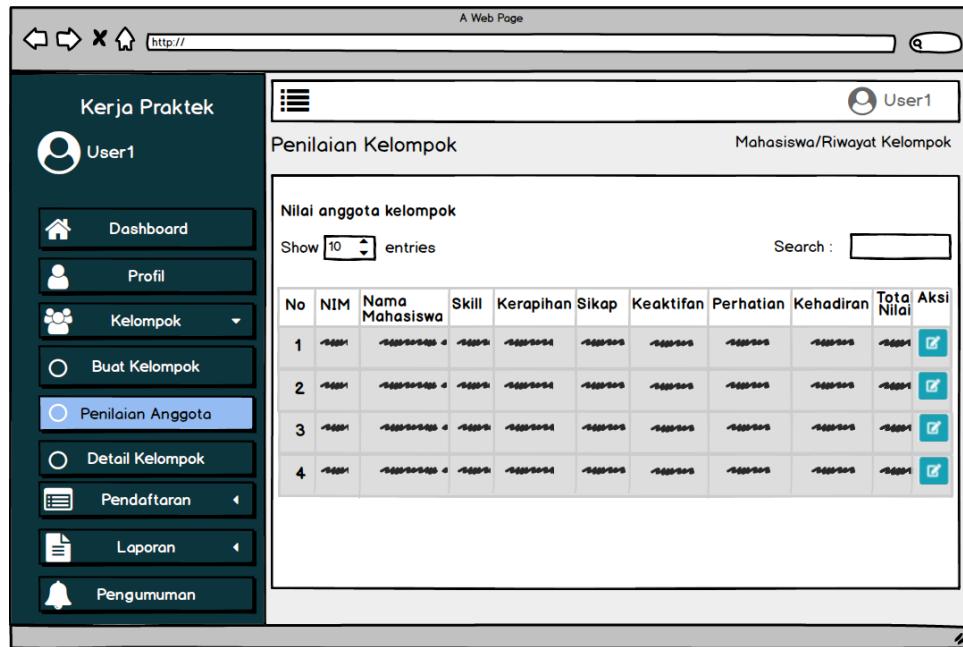
Halaman daftar anggota kelompok adalah halaman yang menampilkan daftar anggota kelompok. Dalam halaman ini ketua dapat menghapus undangan anggota kelompok jika tahap kelompok berstatus “diproses”.

The screenshot shows a web-based application interface. On the left, there's a sidebar with a dark blue header 'Kerja Praktek' and a user profile 'User1'. Below the header, the sidebar contains several menu items: 'Dashboard', 'Profil', 'Kelompok' (which is highlighted in blue), 'Pendaftaran', 'Usulan Instansi', 'Lowongan', 'Laporan', and 'Pengumuman'. The main content area has a header 'Daftar Anggota' and 'Mahasiswa/Daftar Kelompok'. Below this, it says 'Kelompok Ultramen'. There's a green button '+ Tambah anggota'. A table lists one member: 'No' (1), 'NIM' (redacted), 'Nama' (redacted), 'Status' (anggota, highlighted in yellow), and 'Aksi' (a red square button). At the bottom, it says 'Showing 1 to 1 of 1 entries' and has 'Previous' and 'Next' buttons.

Gambar 4. 20 Rancangan antarmuka halaman daftar anggota kelompok

9. Rancangan antarmuka-halaman penilaian anggota kelompok

Halaman penilaian anggota kelompok untuk menampilkan nilai terhadap masing-masing anggota kelompok.



Gambar 4. 21 Rancangan antarmuka halaman penilaian anggota kelompok

10. Rancangan antarmuka halaman form penilaian

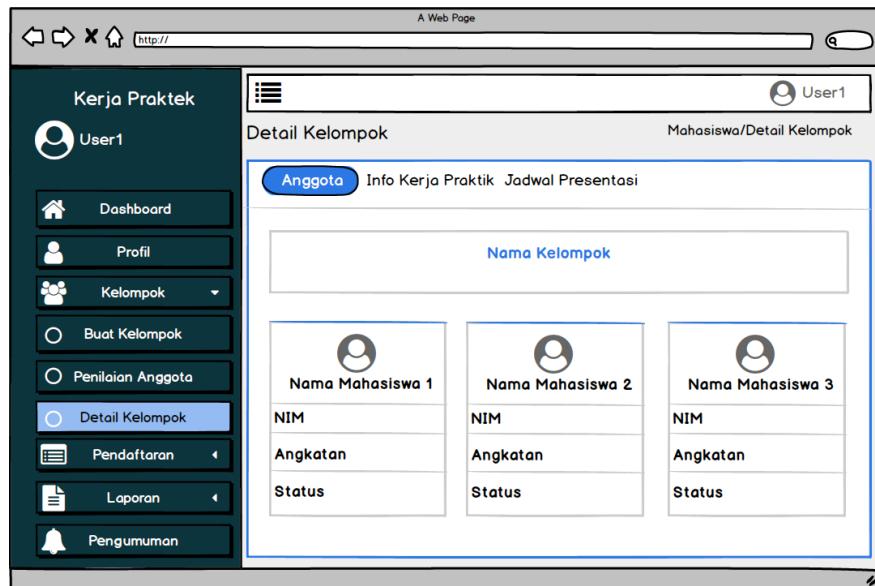
Halaman form penilaian merupakan halaman yang digunakan untuk memberikan penilaian kepada anggota kelompok. Indikator yang dinilai yaitu *skill*, kerapihan, sikap, keaktifan, dan perhatian dengan skala penilaian 1-5. Rancangan antarmuka halaman form penilaian dalam Gambar 4.22

The screenshot shows the 'Penilaian Anggota Kelompok' form. The top part is identical to the previous screenshot. The main form area is titled 'Penilaian Anggota Kelompok' and 'Mahasiswa/ Penilaian Kelompok'. It features a horizontal 5-point rating scale with colored boxes: green for 'Sangat Baik' (5), blue for 'Baik' (4), yellow for 'Cukup' (3), red for 'Kurang' (2), and grey for 'Sangat Kurang' (1). Below the scale, there are five dropdown menus for 'Nama Mahasiswa', 'Skill', 'Kerapihan', 'Sikap', and 'Keaktifan'. A 'Perhatian' section is also present with a dropdown menu. A 'Keterangan' section at the bottom defines the abbreviations: 1. Skill, 2. Kerapihan, 3. Sikap, 4. Keaktifan, 5. Perhatian. At the bottom right are 'Cancel' and 'Save' buttons.

Gambar 4. 22 Rancangan antarmuka halaman form penilaian

11. Rancangan antarmuka halaman detail kelompok tab anggota

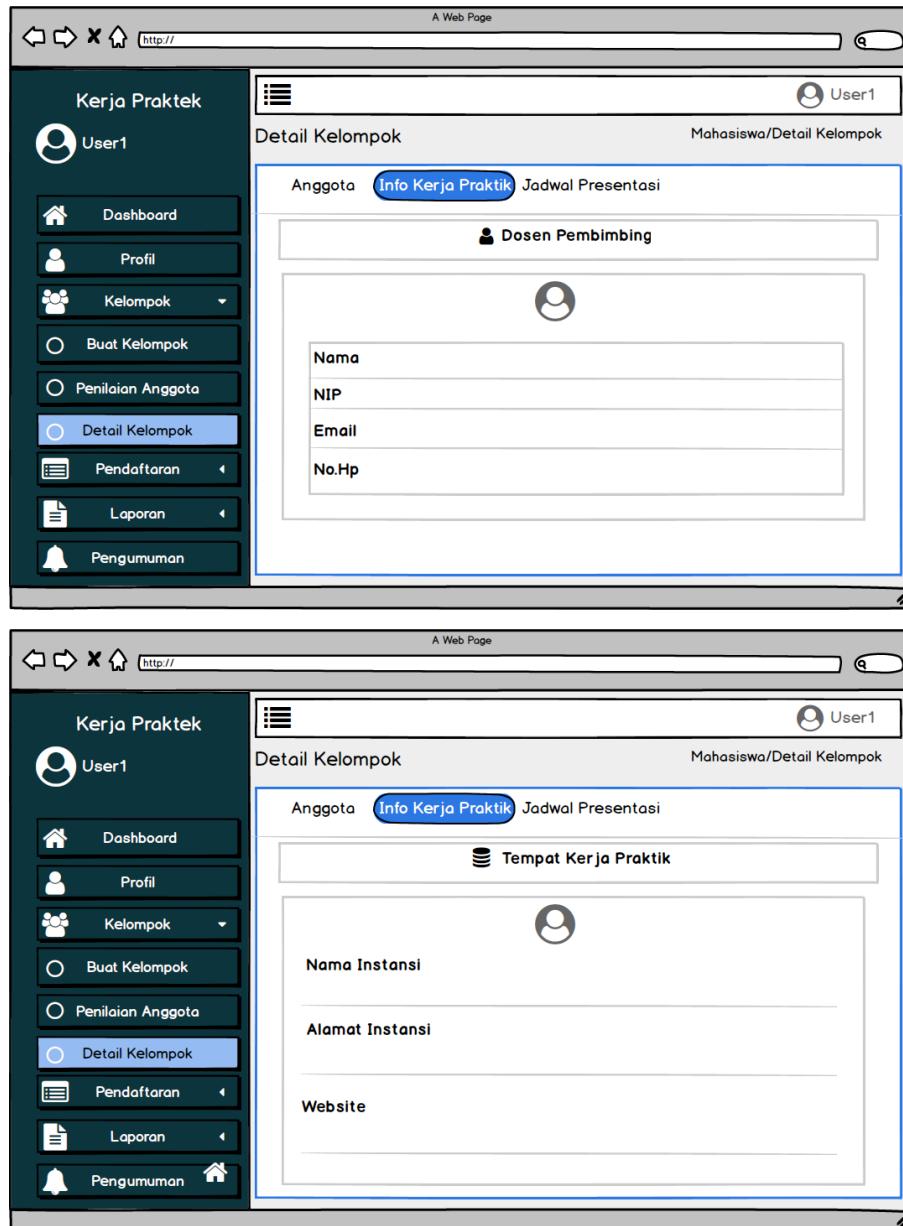
Halaman detail kelompok tab anggota berisi nama kelompok dan informasi anggota pada kelompok kerja praktek. Informasi anggota kelompok berisi nama mahasiswa, nim, angkatan dan status dalam kelompok. Rancangan antarmuka halaman detail kelompok tab anggota dalam Gambar 4.23



Gambar 4. 23 Rancangan antarmuka halaman detail kelompok tab anggota

12. Rancangan antarmuka halaman detail kelompok tab info kerja praktek

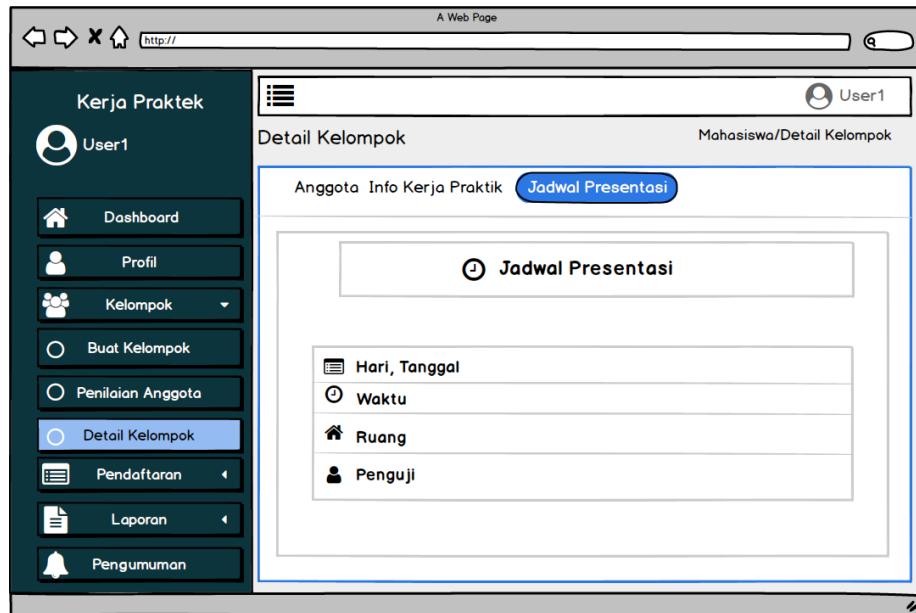
Halaman detail kelompok tab info kerja praktek menampilkan informasi dosen pembimbing dan detail info tempat kerja praktek. Rancangan antarmuka halaman detail kelompok tab info kerja praktek dalam Gambar 4.24



Gambar 4. 24 Rancangan antarmuka halaman detail kelompok tab info kerja praktek

13. Rancangan antarmuka halaman detail kelompok tab jadwal presentasi

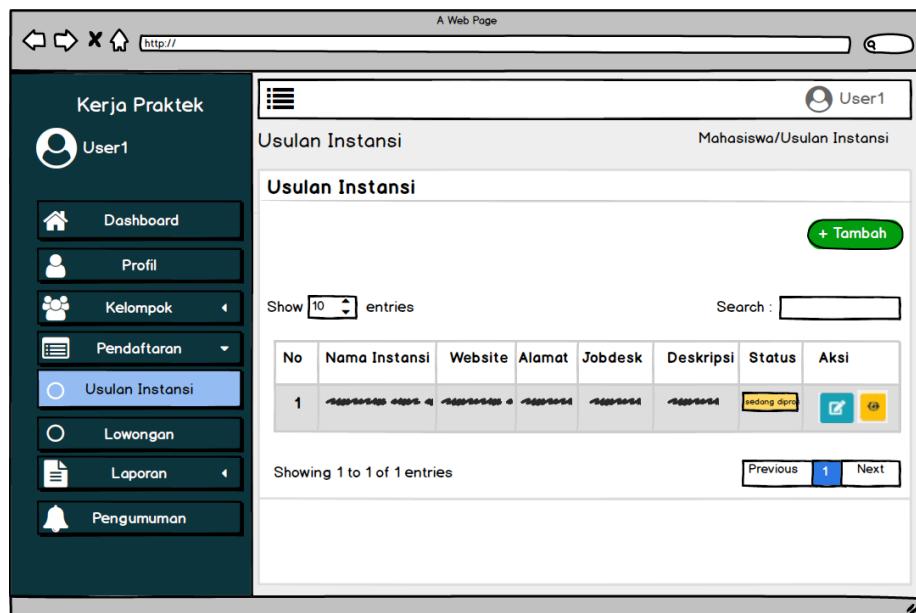
Halaman detail kelompok tab presentasi menampilkan informasi tentang jadwal presentasi laporan akhir kerja praktek yang berisi hari, tanggal, waktu, ruang dan penguji. Rancangan antarmuka dalam Gambar 4.25



Gambar 4. 25 Rancangan antarmuka halaman detail kelompok tab jadwal presentasi

14. Rancangan antarmuka halaman tambah usulan instansi

Halaman tambah usulan instansi menampilkan usulan instansi yang diajukan untuk tempat kerja praktek.



A screenshot of a web application interface. The left sidebar is dark blue with white text and icons. It includes 'Dashboard', 'Profil', 'Kelompok', 'Pendaftaran', 'Usulan Instansi' (which is highlighted in blue), 'Lowongan', 'Laporan', and 'Pengumuman'. The main content area has a header 'Mahasiswa/Tambah Usulan'. Below it is a form titled 'Tambah Usulan' with fields: 'Nama instansi*' (text input), 'Website*' (text input), 'Alamat*' (text input), 'Jobdesk*' (text input), 'Deskripsi*' (text input), and 'Surat Pemberitahuan' (file input with placeholder 'Telusuri....'). At the bottom are 'Cancel' and 'Save' buttons.

Gambar 4. 26 Rancangan antarmuka halaman tambah usulan instansi

15. Rancangan antarmuka halaman edit usulan instansi

Halaman edit usulan instansi digunakan untuk mengubah informasi tentang detail usulan instansi yang diajukan.

A screenshot of a web application interface, similar to the one above but for editing. The left sidebar is identical. The main content area has a header 'Mahasiswa/Tambah Usulan'. Below it is a form titled 'Tambah Usulan' with fields: 'Nama instansi*' (text input), 'Website*' (text input), 'Alamat*' (text input), 'Jobdesk*' (text input), 'Deskripsi*' (text input), and 'Surat Pemberitahuan*' (text input with placeholder 'Telusuri....'). At the bottom are 'Cancel' and 'Save' buttons.

Gambar 4. 27 Rancangan antarmuka halaman edit usulan instansi

16. Rancangan antarmuka halaman detail usulan instansi

Halaman detail usulan instansi menampilkan tentang informasi instansi yang akan digunakan sebagai tempat kerja praktek.

No	Nama Instansi	Website	Alamat	Jobdesk	Deskripsi	Status	Aksi
1	[REDACTED]	[REDACTED]	[REDACTED]	[REDACTED]	[REDACTED]	sedang diproses	[Edit] [Delete]

Gambar 4. 28 Rancangan antarmuka halaman detail usulan instansi

17. Rancangan antarmuka halaman daftar lowongan

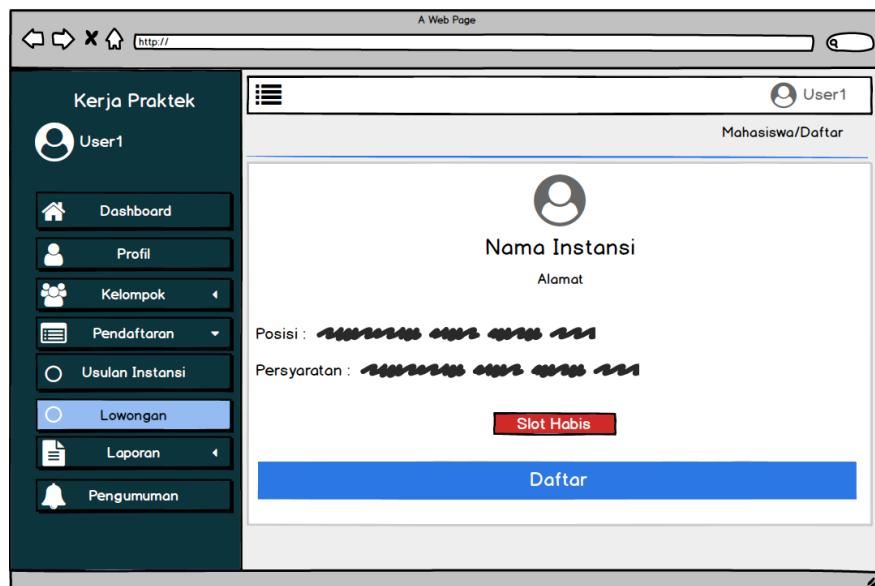
Halaman daftar lowongan menampilkan daftar lowongan yang tersedia dan bisa di-*apply* oleh mahasiswa. Berisi informasi tentang nama instansi, posisi, dan kapasitas yang tersedia.

No	Nama Instansi	Posisi	Kapasitas	Aksi
1	[REDACTED]	[REDACTED]	[REDACTED]	[> Cek Status Pendaftaran]
2	[REDACTED]	[REDACTED]	[REDACTED]	[> Cek Status Pendaftaran]

Gambar 4. 29 Rancangan antarmuka halaman daftar lowongan

18. Rancangan antarmuka halaman apply lowongan

Halaman *apply* lowongan berfungsi sebagai tempat mahasiswa untuk mendaftar kerja praktek. Berisi informasi tentang posisi yang akan di-*apply* dan persyaratan yang dibutuhkan.



Gambar 4. 30 Rancangan antarmuka halaman apply lowongan

19. Rancangan antarmuka halaman riwayat apply lowongan

Halaman riwayat *apply* lowongan menampilkan daftar lowongan yang pernah didaftarkan beserta statusnya.

No	Nama Instansi	Posisi	Status
1	██████████	██████████	ditolak
2	██████████	██████████	sedang diproses

Gambar 4. 31 Rancangan antarmuka halaman riwayat apply lowongan

20. Rancangan antarmuka halaman detail laporan harian

Halaman detail laporan harian berisi daftar kegiatan selama kerja praktek yang berisi tanggal, waktu datang dan pulang, kegiatan yang dilakukan, dan status.

No	Tanggal	Datang	Pulang	Kegiatan	Status	Aksi
1	2023-09-01	08:00	16:00	Praktikum	disetujui	
2	2023-09-02	08:00	16:00	Praktikum	disetujui	
3	2023-09-03	08:00	16:00	Praktikum	sedang diproses	

Gambar 4. 32 Rancangan antarmuka halaman detail laporan harian

21. Rancangan antarmuka halaman tambah laporan harian

Halaman tambah laporan harian digunakan untuk menambahkan list kegiatan yang dilakukan pada hari tersebut.

Gambar 4. 33 Rancangan antarmuka halaman tambah laporan harian

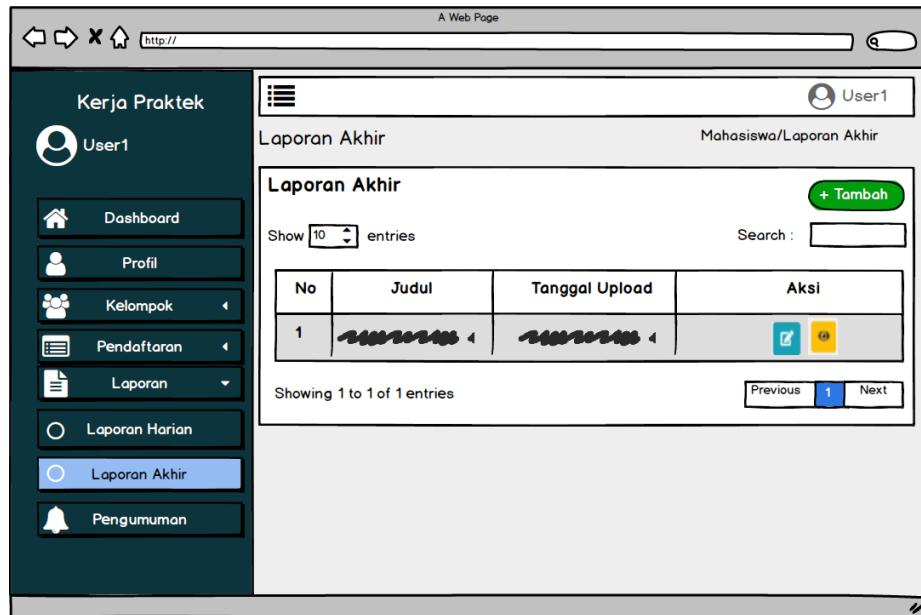
22. Rancangan antarmuka halaman edit laporan harian

Halaman edit laporan harian berfungsi untuk mengubah informasi tentang laporan harian.

Gambar 4. 34 Rancangan antarmuka halaman edit laporan harian

23. Rancangan-antarmuka-halaman detail laporan akhir

Halaman detail laporan akhir berisi laporan akhir yang telah diupload dan tanggal upload.



Gambar 4. 35 Rancangan antarmuka halaman detail laporan akhir

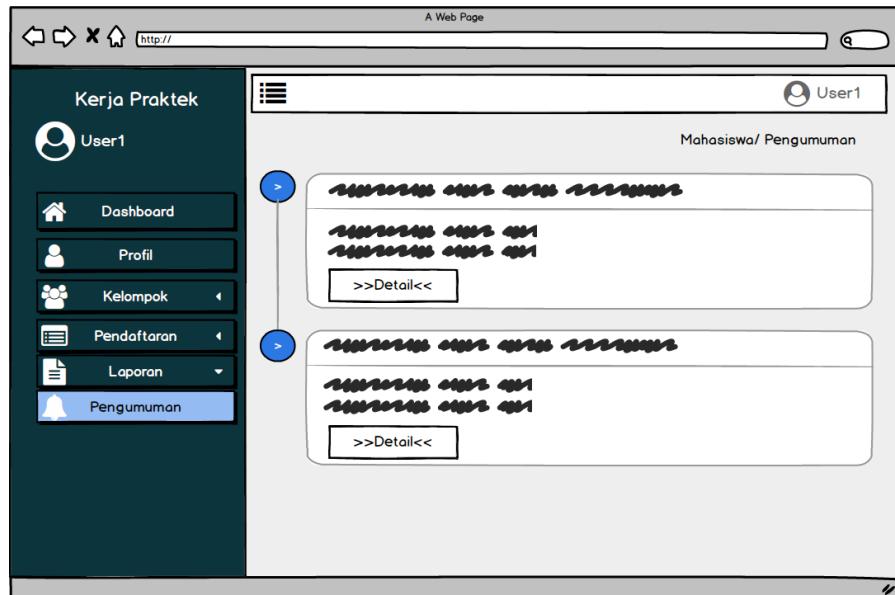
24. Rancangan antarmuka halaman tambah laporan akhir

Halaman tambah laporan akhir digunakan untuk mengupload berkas dokumen laporan akhir.

Gambar 4. 36 Rancangan antarmuka halaman tambah laporan akhir

25. Rancangan antarmuka halaman pengumuman

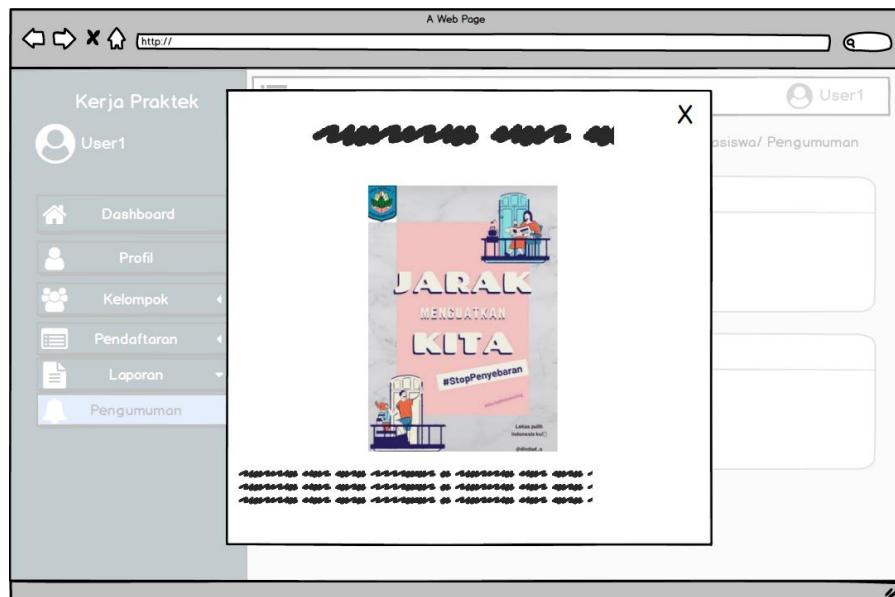
Halaman pengumuman berisi tentang pengumuman selama kegiatan kerja praktik.



Gambar 4. 37 Rancangan antarmuka halaman pengumuman

26. Rancangan antarmuka halaman detail pengumuman

Berisi tentang detail informasi pengumuman .



Gambar 4. 38 Rancangan antarmuka halaman detail pengumuman

27. Rancangan antarmuka halaman ubah password

Halaman yang menampilkan form untuk mengubah password.

The screenshot shows a web browser window titled "A Web Page". The address bar displays "http://". The main content area has a dark blue header with the text "Kerja Praktek" and "User1". Below the header is a vertical navigation menu with the following items: Dashboard (selected), Profil, Kelompok, Pendaftaran, Laporan, and Pengumuman. The main content area is titled "Mahasiswa/ Ubah Password". It contains a form titled "Ubah Password" with two input fields: "New Password*" and "Confirm Paassword*". At the bottom right of the form are two buttons: "Cancel" (red) and "Save" (blue).

Gambar 4. 39 Rancangan antarmuka halaman ubah password

BAB V

IMPLEMENTASI SISTEM

Analisis dan perancangan sistem diterapkan dalam tahap implementasi sistem. Tahap implementasi dalam penelitian ini terdiri dari implementasi perangkat keras pembangun, implementasi perangkat lunak pembangun, implementasi basis data, dan implementasi antarmuka.

5.1 Implementasi Perangkat Lunak Pembangun

Perangkat lunak pembangun adalah perangkat lunak yang digunakan untuk membangun sistem informasi kerja praktek bagian mahasiswa pada program studi komputer dan sistem informasi. Dalam penelitian ini perangkat lunak yang digunakan sebagai berikut :

1. Windows 10pro 64-bit sebagai *operating system*
2. Laravel v6.0 sebagai framework
3. PHP v7.2.9 sebagai bahasa pemrograman sisi server
4. HTML 5, CSS3, JavaScript, jQuery, sebagai bahasa pemrograman sisi client
5. MySQL 5.0.12 sebagai server database
6. Apache v2.4.34 sebagai web server
7. XAMPP v3.2.2 sebagai paket aplikasi Apache, MySQL , dan PHP
8. Google Chrome v83.0.4103.106 sebagai web browser
9. Bootstrap versi 4
10. Visual Studio Code 1.42.1 sebagai editor text
11. Admin LTE 3 sebagai template HTML
12. Postman versi 7.27.0

5.2 Implementasi Perangkat Keras Pembangun

Perangkat keras yang digunakan untuk membangun sistem ini adalah:

1. Processor Intel(R) Core(TM) i3-3227U CPU @ 1.90GHz 1.90GHz
2. RAM-8.00-GB

3. HDD Hitachi 500GB
4. Monitor Touchscreen LCD 14”
5. Keyboard
6. Mouse logitech

5.3 Implementasi Basis Data

Implementasi basis data adalah implementasi dari hasil analisis rancangan basis data. Dalam tahapan sebelumnya, analisis rancangan basis data terdiri dari rancangan ERD, rancangan struktur tabel, dan rancangan tabel.

Dalam penggerjaan proyek ini database yang digunakan diberi nama “kerjapraktek” yang memiliki 22 tabel yang terdiri dari atribut dan relasi didalamnya. Implementasi dari pembuatan basis data sistem informasi sebagai berikut :

1. Tabel admin

Tabel admin menyimpan data admin yaitu id_admin, id_users, nama, email, no_hp, foto, isDeleted, created_by, created_at, updated_at.

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
1	id_admin	int(11)			No	None		AUTO_INCREMENT
2	id_users	int(11)			No	None		
3	nama	varchar(255)	latin1_swedish_ci		No	None		
4	email	varchar(255)	latin1_swedish_ci		No	None		
5	no_hp	varchar(25)	latin1_swedish_ci		No	None		
6	foto	text	latin1_swedish_ci		No	None		
7	isDeleted	tinyint(4)			No	0		
8	created_by	int(11)			No	None		
9	created_at	timestamp			No	CURRENT_TIMESTAMP		
10	updated_at	timestamp			No	CURRENT_TIMESTAMP		

Gambar 5. 1 Implementasi tabel admin

2. Tabel dosen

Tabel dosen menyimpan profil dosen yang terdiri dari id_dosen, id_users, nama, nip, email, no_hp, slot, kapasitas, status, foto, isDeleted, created_by, created_at, updated_at.

	#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
	1	id_dosen 	int(11)			No	None		AUTO_INCREMENT
	2	id_users 	int(11)			No	None		
	3	nama	varchar(150)	latin1_swedish_ci		No	None		
	4	nip	varchar(50)	latin1_swedish_ci		No	None		
	5	email	varchar(100)	latin1_swedish_ci		No	None		
	6	no_hp	varchar(20)	latin1_swedish_ci		No	None		
	7	slot	tinyint(4)			No	None		
	8	kapasitas	tinyint(4)			No	None		
	9	status	enum('open', 'close')	latin1_swedish_ci		No	None		
	10	foto	varchar(255)	latin1_swedish_ci		No	None		
	11	isDeleted	tinyint(4)			No	None		
	12	created_by	int(11)			No	None		
	13	created_at	datetime			No	None		
	14	updated_at	datetime			No	None		

Gambar 5. 2 Implementasi tabel dosen

3. Tabel mahasiswa

Tabel mahasiswa menyimpan profil mahasiswa yaitu id_mahasiswa, id_users, id_periode, nama, nim, angkatan, no_hp, email, alamat, kemampuan, pengalaman, foto, cv, isDeleted, created_by, created_at, updated_at.

	#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
	1	id_mahasiswa 	int(11)			No	None		AUTO_INCREMENT
	2	id_users 	int(11)			No	None		
	3	id_periode 	int(11)			No	None		
	4	nama	varchar(150)	latin1_swedish_ci		No	None		
	5	nim	varchar(20)	latin1_swedish_ci		Yes	NULL		
	6	angkatan	year(4)			Yes	NULL		
	7	no_hp	varchar(20)	latin1_swedish_ci		Yes	NULL		
	8	email	varchar(100)	latin1_swedish_ci		Yes	NULL		
	9	alamat	text	latin1_swedish_ci		Yes	NULL		
	10	kemampuan	text	latin1_swedish_ci		Yes	NULL		
	11	pengalaman	text	latin1_swedish_ci		Yes	NULL		
	12	foto	varchar(255)	latin1_swedish_ci		Yes	NULL		
	13	cv	varchar(255)	latin1_swedish_ci		Yes	NULL		
	14	isDeleted	tinyint(4)			Yes	NULL		
	15	created_by	int(11)			Yes	NULL		
	16	created_at	datetime			No	None		
	17	updated_at	datetime			No	None		

Gambar 5. 3 Implementasi tabel mahasiswa

4. Tabel aspek_penilaian

Tabel aspek_penilaian menyimpan data aspek penilaian yang terdiri dari sikap, keaktifan, kemampuan, kebersamaan, kerapihan, perhatian, kehadiran, keterkaitan, kesesuaian, sistematika, ketepatan, kekompakan.

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
1	id_aspek_penilaian	int(11)			No	None		AUTO_INCREMENT
2	nama	varchar(30)	latin1_swedish_ci		No	None		
3	isDeleted	tinyint(4)			No	None		
4	created_by	int(11)			No	None		
5	created_at	datetime			No	None		
6	updated_at	datetime			No	None		

Gambar 5. 4 Implementasi tabel aspek_penilaian

5. Tabel buku_harian

Tabel buku_harian menyimpan data buku harian atau laporan harian mahasiswa yang melakukan kerja praktek. Tabel buku harian terdiri dari id_buku_harian, id_mahasiswa, id_periode, waktu_mulai, waktu_selesai, tanggal, kegiatan, status, isDeleted, created_by, created_at, updated_at.

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
1	id_buku_harian	int(11)			No	None		AUTO_INCREMENT
2	id_mahasiswa	int(11)			No	None		
3	id_periode	int(11)			No	None		
4	waktu_mulai	time			No	None		
5	waktu_selesai	time			No	None		
6	tanggal	date			No	None		
7	kegiatan	text	latin1_swedish_ci		No	None		
8	status	enum('diproses', 'diperiksa')	latin1_swedish_ci		No	diproses		
9	isDeleted	tinyint(4)			No	0		
10	created_by	int(11)			No	None		
11	created_at	datetime			No	CURRENT_TIMESTAMP		
12	updated_at	datetime			No	CURRENT_TIMESTAMP		

Gambar 5. 5 Implementasi tabel buku_harian

6. Tabel jadwal_presentasi

Tabel jadwal_presentasi menyimpan data jadwal presentasi kelompok terdiri dari id_jadwal_presentasi, id_kelompok, id_periode, id_dospeng, id_sesiwaktu, id_ruang, tanggal, created_by, isDeleted, created_at, updated_at.

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
1	id_jadwal_presentasi	int(11)			No	None		AUTO_INCREMENT
2	id_kelompok	int(11)			No	None		
3	id_periode	int(11)			No	None		
4	id_dospeng	int(11)			No	None		
5	id_sesiwaktu	int(11)			No	None		
6	id_ruang	int(11)			No	None		
7	tanggal	date			No	None		
8	created_by	int(11)			No	None		
9	isDeleted	tinyint(4)			No	0		
10	created_at	timestamp			No	CURRENT_TIMESTAMP		
11	updated_at	timestamp			No	CURRENT_TIMESTAMP		

Gambar 5. 6 Implementasi tabel jadwal_presentasi

7. Tabel kelompok

Tabel kelompok menyimpan data kelompok yang akan melakukan kerja praktek. Tabel kelompok terdiri dari id_kelompok, id_periode, nama_kelompok, id_dosen, kapasitas, tahap, slot, status, isDeleted, created_by, created_at, updated_at.

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
1	id_kelompok	int(11)			No	None		AUTO_INCREMENT
2	id_periode	int(11)			No	None		
3	nama_kelompok	varchar(100)	latin1_swedish_ci		No	None		
4	id_dosen	int(11)			Yes	NULL		
5	kapasitas	tinyint(4)			No	3		
6	tahap	enum('diproses', 'diterima', 'ditolak')	latin1_swedish_ci		No	diproses		
7	slot	tinyint(4)			No	2		
8	status	enum('open', 'close')	latin1_swedish_ci		No	open		
9	isDeleted	tinyint(4)			No	0		
10	created_by	int(11)			No	None		
11	created_at	datetime			No	None		
12	updated_at	datetime			No	None		

Gambar 5. 7 Implementasi tabel kelompok

8. Tabel kelompok_detail

Tabel kelompok_detail menimpan data detail kelompok terdiri dari id_kelompok_detail, id_kelompok, id_mahasiswa, status_keanggotaan, status_join, created_by, isDeleted, created_at, updated_at.

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
1	id_kelompok_detail	int(11)		No	None			AUTO_INCREMENT
2	id_kelompok	int(11)		No	None			
3	id_mahasiswa	int(11)		No	None			
4	status_keanggotaan	enum('Ketua', 'Anggota')	latin1_swedish_ci	No	None			
5	status_join	enum('diinvite', 'diterima', 'ditolak', 'join', 'c')	latin1_swedish_ci	No	None			
6	created_by	int(11)		No	None			
7	isDeleted	tinyint(4)		No	0			
8	created_at	datetime		No	None			
9	updated_at	datetime		No	None			

Gambar 5. 8 Implementasi tabel kelompok_detail

9. Tabel kelompok_penilai

Tabel kelompok_penilai menyimpan data kelompok penilai yang akan memberikan penilaian terhadap mahasiswa. Tabel kelompok_penilai terdiri dari id_kelompok_penilai, nama, isDeleted, created_by, created_at, updated_at.

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
1	id_kelompok_penilai	int(11)		No	None			AUTO_INCREMENT
2	nama	varchar(30)	latin1_swedish_ci	No	None			
3	isDeleted	tinyint(4)		No	None			
4	created_by	int(11)		No	None			
5	created_at	datetime		No	None			
6	updated_at	datetime		No	None			

Gambar 5. 9 Implementasi tabel kelompok_penilai

10. Tabel laporan

Tabel laporan menyimpan data laporan akhir kelompok mahasiswa saat melaksanakan kerja praktek. Tabel laporan terdiri dari id_laporan, id_kelompok, judul, berkas, isDeleted, created_by, created_at, updated_at.

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
1	id_laporan	int(11)		No	None			AUTO_INCREMENT
2	id_kelompok	int(11)		No	None			
3	judul	varchar(150)	latin1_swedish_ci	No	None			
4	berkas	varchar(255)	latin1_swedish_ci	No	None			
5	isDeleted	tinyint(4)		No	0			
6	created_by	int(11)		No	None			
7	created_at	datetime		No	None			
8	updated_at	datetime		No	None			

Gambar 5. 10 Implementasi tabel laporan

11. Tabel lowongan

Tabel lowongan menyimpan data lowongan yang diberikan oleh instansi kepada mahasiswa yang akan melaksanakan kerja praktek. Tabel lowongan terdiri dari id_lowongan, id_instansi,id_periode, pekerjaan, persyaratan, kapasitas, slot, isDeleted, created_by, created_at, updated_at.

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
1	id_lowongan	int(11)			No	None		AUTO_INCREMENT
2	id_instansi	int(11)			No	None		
3	id_periode	int(11)			No	None		
4	pekerjaan	varchar(100)	latin1_swedish_ci		No	None		
5	persyaratan	text	latin1_swedish_ci		No	None		
6	kapasitas	tinyint(4)			No	None		
7	slot	tinyint(4)			No	None		
8	isDeleted	tinyint(4)			No	None		
9	created_by	int(11)			No	None		
10	created_at	datetime			No	None		
11	updated_at	datetime			No	None		

Gambar 5. 11 Implementasi tabel lowongan

12. Tabel magang

Tabel magang menyimpan data mahasiswa yang melakukan kerja praktek. Tabel magang terdiri dari id_magang, id_kelompok, id_instansi, id_periode, tanggal_mulai, tanggal_selesai, jobdesk, status, isDeleted, created_by, created_at, updated_at.

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
1	id_magang	int(11)			No	None		AUTO_INCREMENT
2	id_kelompok	int(11)			No	None		
3	id_instansi	int(11)			No	None		
4	id_periode	int(11)			No	None		
5	tanggal_mulai	date			No	None		
6	tanggal_selesai	date			No	None		
7	jobdesk	text	latin1_swedish_ci		No	None		
8	status	enum('belum magang', 'magang', 'selesai')	latin1_swedish_ci		No	belum magang		
9	isDeleted	tinyint(4)			No	0		
10	created_by	int(11)			No	None		
11	created_at	timestamp			No	CURRENT_TIMESTAMP		
12	updated_at	timestamp			No	CURRENT_TIMESTAMP		

Gambar 5. 12 Implementasi tabel magang

13. Tabel nilai

Tabel nilai menyimpan data nilai mahasiswa yang diberikan oleh kelompok penilai yang terdiri dari id_nilai, id_periode, id_mahasiswa, id_aspek_penilaian, nilai, isDeleted, created_by, created-at, updated-at.

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
1	id_nilai	int(11)			No	None		AUTO_INCREMENT
2	id_periode	int(11)			No	None		
3	id_mahasiswa	int(11)			No	None		
4	id_aspek_penilaian	int(11)			No	None		
5	id_kelompok_penilai	int(11)			No	None		
6	nilai	float			No	None		
7	isDeleted	tinyint(4)			No	0		
8	created_by	int(11)			No	None		
9	created_at	datetime			No	None		
10	updated_at	datetime			No	None		

Gambar 5. 13 Implementasi tabel nilai

14. Tabel pelamar

Tabel pelamar menyimpan data mahasiswa yang melamar lowongan yang diberikan oleh instansi. Tabel pelamar terdiri dari id_pelamar, id_lowongan, id_kelompok, tanggal_daftar, status, created_by, isDeleted, created_at, updated_at.

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
1	id_pelamar	int(11)			No	None		AUTO_INCREMENT
2	id_lowongan	int(11)			No	None		
3	id_kelompok	int(11)			No	None		
4	tanggal_daftar	datetime			No	CURRENT_TIMESTAMP		
5	status	enum('melamar', 'diterima', 'ditolak')	latin1_swedish_ci		No	melamar		
6	created_by	int(11)			No	None		
7	isDeleted	tinyint(4)			No	0		
8	created_at	datetime			No	CURRENT_TIMESTAMP		
9	updated_at	datetime			No	CURRENT_TIMESTAMP		

Gambar 5. 14 Implementasi tabel pelamar

15. Tabel pengumuman

Tabel pengumuman menyimpan data pengumuman tentang penyelenggaraan kerja praktek. Tabel pengumuman terdiri dari id_pengumuman, judul, deskripsi, lampiran, isDeleted, created_by, created_at, updated_at.

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
1	id_pengumuman	int(11)	latin1_swedish_ci		No	None		AUTO_INCREMENT
2	judul	varchar(150)	latin1_swedish_ci		No	None		
3	deskripsi	text	latin1_swedish_ci		No	None		
4	lampiran	varchar(255)	latin1_swedish_ci		Yes	NULL		
5	created_at	datetime			No	None		
6	updated_at	datetime			No	None		

Gambar 5. 15 Implementasi tabel pengumuman

16. Tabel periode

Tabel periode menyimpan data periode untuk men-setting periode kerja praktik. Tabel periode terdiri dari id_periode, tahun_periode, tgl_mulai, tgl_selesai, status, isDeleted, created_by, created_at, updated_at.

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
1	id_periode	int(11)	latin1_swedish_ci		No	None		AUTO_INCREMENT
2	tahun_periode	int(11)			No	None		
3	tgl_mulai	date			No	None		
4	tgl_selesai	date			No	None		
5	status	enum('open', 'close')	latin1_swedish_ci		No	None		
6	isDeleted	tinyint(4)			No	None		
7	created_by	int(11)			No	None		
8	created_at	datetime			No	None		
9	updated_at	datetime			No	None		

Gambar 5. 16 Implementasi tabel periode

17. Tabel roles

Tabel menyimpan kategori user yang terdiri dari id_roles dan roles.

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
1	id_roles	tinyint(4)	latin1_swedish_ci		No	None		AUTO_INCREMENT
2	roles	varchar(20)	latin1_swedish_ci		No	None		

Gambar 5. 17 Implementasi tabel roles

18. Tabel ruang

Tabel ruang menyimpan data ruang atau tempat yang akan digunakan untuk presentasi kelompok. Tabel ruang terdiri dari id_ruang, ruang, isDeleted, created_by, created_at, updated_at.

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
1	id_ruang	int(11)			No	None		AUTO_INCREMENT
2	ruang	varchar(255)	latin1_swedish_ci		No	None		
3	isDeleted	tinyint(4)			No	0		
4	created_by	int(11)			No	None		
5	created_at	timestamp			No	CURRENT_TIMESTAMP		
6	updated_at	timestamp			No	CURRENT_TIMESTAMP		

Gambar 5. 18 Implementasi tabel ruang

19. Tabel sesiwaktu

Tabel sesiwaktu menyimpan data waktu untuk presentasi kelompok. Tabel sesi waktu tersiri dari id_sesiwaktu, sesi, isDeleted, created_by, created_at, updated_at.

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
1	id_sesiwaktu	int(11)			No	None		AUTO_INCREMENT
2	sesi	time			No	None		
3	isDeleted	tinyint(4)			No	0		
4	created_by	int(11)			No	None		
5	created-at	timestamp			No	CURRENT_TIMESTAMP		
6	updated_at	timestamp			No	CURRENT_TIMESTAMP		

Gambar 5. 19 Implementasi tabel sesiwaktu

20. Tabel-users

Tabel users menyimpan data akun user. Tabel users terdiri dari id_users, username, password, id_roles, api_token, isDeleted, created_by, updated_by, created_at, updated_at.

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
1	id_users	int(11)			No	None		AUTO_INCREMENT
2	username	varchar(12)	latin1_swedish_ci		No	None		
3	password	varchar(255)	latin1_swedish_ci		No	None		
4	id_roles	tinyint(4)			No	None		
5	api_token	varchar(255)	latin1_swedish_ci		Yes	NULL		
6	isDeleted	tinyint(4)			No	None		
7	created_at	datetime			No	None		
8	updated_at	datetime			No	None		
9	created_by	int(11)			No	None		
10	updated_by	int(11)			No	None		

Gambar 5. 20 Implementasi tabel users

21. Tabel usulan

Tabel usulan menyimpan data usulan kelompok mahasiswa yang ingin melakukan kerja praktek. Tabel usulan terdiri dari id_usulan, id_kelompok, id_periode, nama_instansi, deskripsi_instansi, alamat_instansi, website_instansi, jobdesk, surat, status, tahap, created_by, created_at, updated_at.

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
1	id_usulan 🔑	int(11)			No	None		AUTO_INCREMENT
2	id_kelompok 💬	int(11)			No	None		
3	id_periode 📅	int(11)			No	None		
4	nama_instansi	varchar(150)	latin1_swedish_ci		No	None		
5	deskripsi_instansi	text	latin1_swedish_ci		No	None		
6	alamat_instansi	text	latin1_swedish_ci		No	None		
7	website_instansi	varchar(100)	latin1_swedish_ci		No	None		
8	jobdesk	varchar(150)	latin1_swedish_ci		No	None		
9	surat	varchar(255)	latin1_swedish_ci		No	None		
10	status	enum('diproses', 'diterima', 'ditolak')	latin1_swedish_ci		No	diproses		
11	tahap	enum('mendaftar', 'magang', 'selesai', 'ditolak')	latin1_swedish_ci		No	None		
12	created_by	tinyint(11)			No	None		
13	created_at	datetime			No	None		
14	updated_at	datetime			No	None		

Gambar 5. 21 Implementasi tabel usulan

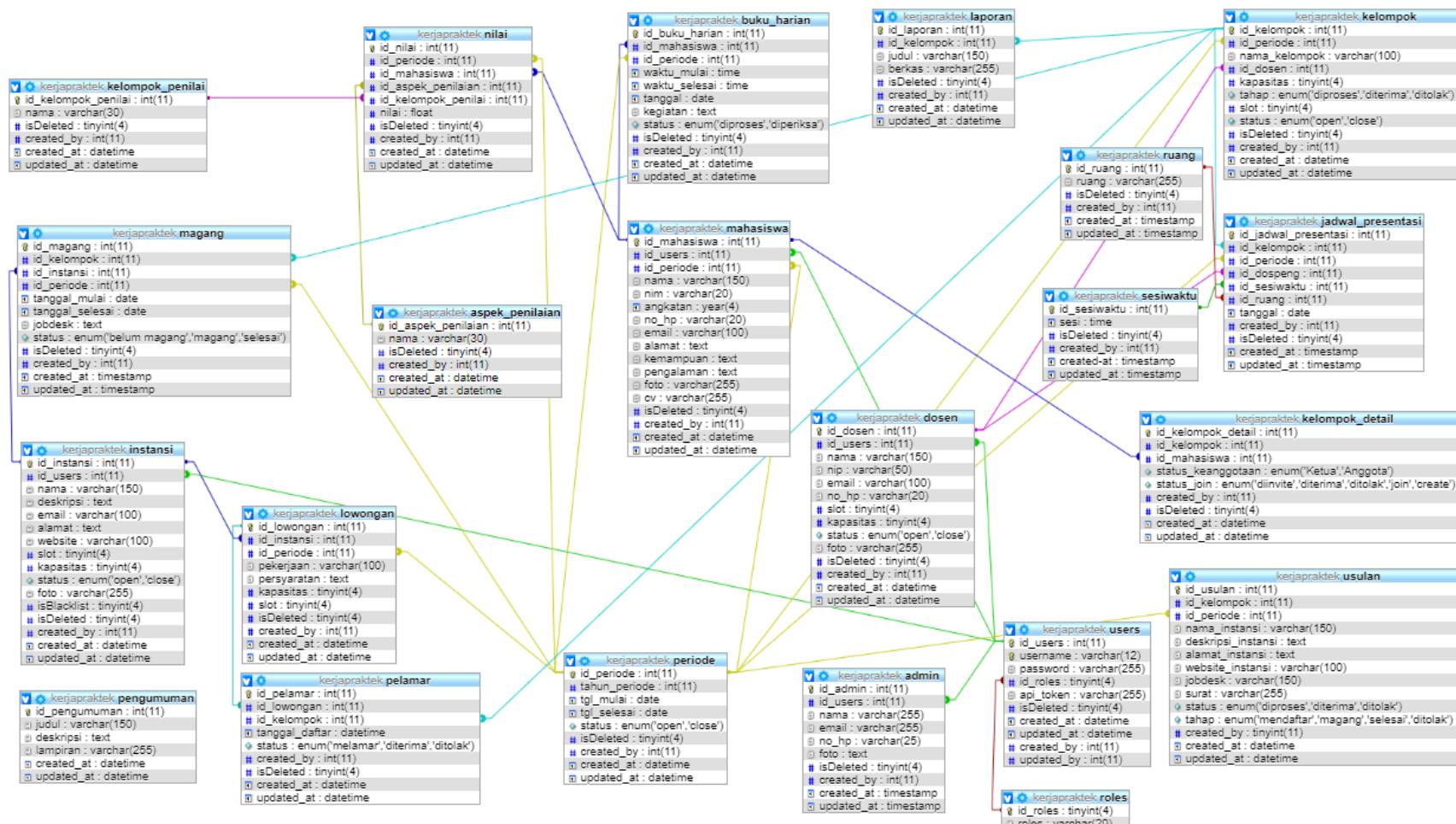
22. Tabel instansi

Tabel instansi menyimpan data instansi yang terdiri dari id_instansi, id_users, nama, deskripsi , email, alamat, website, slot, kapasitas, status, foto , isBlacklist, isDeleted, created_by, created_at, updated_at.

	#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
<input type="checkbox"/>	1	id_instansi 	int(11)		No	None			AUTO_INCREMENT
<input type="checkbox"/>	2	id_users 	int(11)		No	None			
<input type="checkbox"/>	3	nama	varchar(150)	latin1_swedish_ci	No	None			
<input type="checkbox"/>	4	deskripsi	text	latin1_swedish_ci	No	None			
<input type="checkbox"/>	5	email	varchar(100)	latin1_swedish_ci	No	None			
<input type="checkbox"/>	6	alamat	text	latin1_swedish_ci	No	None			
<input type="checkbox"/>	7	website	varchar(100)	latin1_swedish_ci	No	None			
<input type="checkbox"/>	8	slot	tinyint(4)		No	None			
<input type="checkbox"/>	9	kapasitas	tinyint(4)		No	None			
<input type="checkbox"/>	10	status	enum('open', 'close')	latin1_swedish_ci	No	None			
<input type="checkbox"/>	11	foto	varchar(255)	latin1_swedish_ci	Yes	NULL			
<input type="checkbox"/>	12	isBlacklist	tinyint(4)		No	None			
<input type="checkbox"/>	13	isDeleted	tinyint(4)		No	None			
<input type="checkbox"/>	14	created_by	int(11)		No	None			
<input type="checkbox"/>	15	created_at	datetime		No	None			
<input type="checkbox"/>	16	updated_at	datetime		No	None			

Gambar 5. 22 Implementasi tabel instansi

Tabel di atas memiliki relasi antar tabel yang menjelaskan hubungan antar objek. Fungsi dari relasi antar tabel yaitu sebagai pengatur operasi database yang dibutuhkan sistem. Hubungan antar tabel dalam prototipe sistem ini dalam Gambar 5.23.

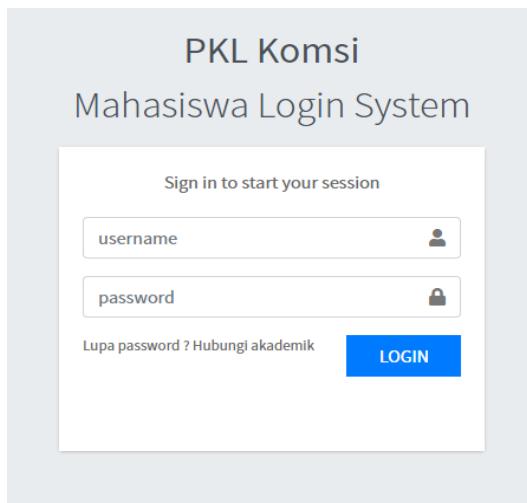


Gambar 5. 23 Hubungan relasi antar tabel

5.4 Implementasi Antarmuka

1. Halaman *login*

Ketika mengakses sistem informasi ini halaman yang muncul pertama adalah halaman login. Mahasiswa login menggunakan username dan password, kemudian sistem akan melakukan validasi . Jika valid maka akan masuk ke halaman dashboard. Implementasi halaman login dalam Gambar 5.24.



Gambar 5. 24 Implementasi-halaman-login

2. Halaman *dashboard*

Halaman *dashboard* merupakan halaman utama setelah mahasiswa berhasil login. Dalam halaman dashboard menampilkan informasi periode kerja praktek, menampilkan menu pintas untuk mengisi laporan harian dan menampilkan info jumlah anggota dalam kelompok. Implementasi halaman *dashboard* dalam Gambar 5.25.



Gambar 5. 25 Implementasi halaman *dashboard*

```

72 | public function kelompokCount()
73 | {
74 |     $userId = Auth::id();
75 |     $idMahasiswa = Mahasiswa::select('id_mahasiswa')
76 |     ->where('id_users', $userId)->first();
77 |     $idKelompok = DetailKelompok::join('kelompok', 'kelompok_detail.id_kelompok', 'kelompok.id_kelompok')
78 |     ->select('kelompok_detail.id_kelompok', 'kelompok_detail.isDeleted')
79 |     ->where('kelompok.tahap', '!=', 'ditolak')
80 |     ->orderBy('id_kelompok', 'desc')
81 |     ->where('kelompok_detail.id_mahasiswa', $idMahasiswa->id_mahasiswa)->first();
82 |     if(!$idKelompok&&empty($idKelompok) || $idKelompok->isDeleted == 1 ){
83 |         return response()->json([
84 |             'jumlah' => 0,
85 |             'message' => 'success',
86 |         ]);
87 |     }else{
88 |         $data = Kelompok::join('kelompok_detail', 'kelompok.id_kelompok', 'kelompok_detail.id_kelompok')
89 |             ->where('kelompok_detail.id_kelompok', $idKelompok->id_kelompok)
90 |             ->where('kelompok_detail.isDeleted', '!=', '1')
91 |             ->where(function($q) {
92 |                 $q->where('kelompok_detail.status_join', 'create')
93 |                 ->orWhere('kelompok_detail.status_join', 'diinvite')
94 |                 ->orWhere('kelompok_detail.status_join', 'diterima');
95 |             })
96 |             ->count();
97 |
98 |         return response()->json([
99 |             'jumlah' => $data,
100 |             'message' => "succes",
101 |         ]);
102 |     }
}

```

Gambar 5. 26 Kode program jumlah anggota

Gambar 5.26 diatas adalah potongan kode program untuk menghitung jumlah anggota kelompok. Potongan kode program diatas dapat dijelaskan sebagai berikut :

- Baris 74 sampai 81 merupakan kode program untuk mengambil id kelompok sesuai dengan id kelompok mahasiswa yang login.
- Baris 82 sampai 86 merupakan kode program untuk menampilkan angka 0 jika mahasiswa belum memiliki kelompok.

iii. Baris 88 sampai 102 merupakan kode program untuk menghitung jumlah anggota dalam kelompok.

3. Halaman profil

Halaman profil menampilkan data diri mahasiswa yaitu nama lengkap, nim ,angkatan, cv, foto profil, no.hp, email, alamat, kemampuan dan pengalaman.

The image contains two screenshots of a web application interface for a student profile. Both screenshots show a dark sidebar on the left with navigation links: Dashboard, Profil, Kelompok, Pendaftaran, Laporan, and Pengumuman. The top right corner shows a user icon and the name 'Nofa Dwi Adeliaa'. The top right also has a breadcrumb navigation: Mahasiswa / Profil Mahasiswa.

Screenshot 1 (Top): Profil Mahasiswa

Nofa Dwi Adeliaa	
NIM	17/410830/SV/18888
Angkatan	2017
CV	Download

Screenshot 2 (Bottom): Profil Mahasiswa

Angkatan	
CV	Download

NIM	Nama Lengkap	No.HP	Email
17/410830/SV/18888	Nofa Dwi Adeliaa	085567883333	nofadwiadelia@gmail.com

Alamat
Guyangan, Cawas RT5 RW 10 , Klaten, Jateng

Kemampuan
Fullstack Developer

Pengalaman
Membuat SI Halodoc

[Edit](#)

Gambar 5. 27 Implementasi halaman profil

Berikut adalah potongan kode program untuk menampilkan halaman profil mahasiswa dalam Gambar 5.28.

```

31     public function indexmahasiswa()
32     {
33         $mahasiswa = Auth::user()->leftJoin('users', 'mahasiswa.id_users', 'users.id_users')
34             ->leftJoin('roles', 'users.id_roles', 'roles.id_roles')
35             ->select('mahasiswa.id_mahasiswa', 'mahasiswa.id_users', 'users.id_users', 'mahasiswa.nama', 'mahasiswa.
36                 foto', 'roles.id_roles', 'roles.roles', 'mahasiswa.no_hp', 'mahasiswa.email', 'mahasiswa.pengalaman',
37                 'mahasiswa.kemampuan', 'mahasiswa.nim', 'mahasiswa.cv', 'mahasiswa.alamat', 'mahasiswa.angkatan')
38             ->first();
39     }

```

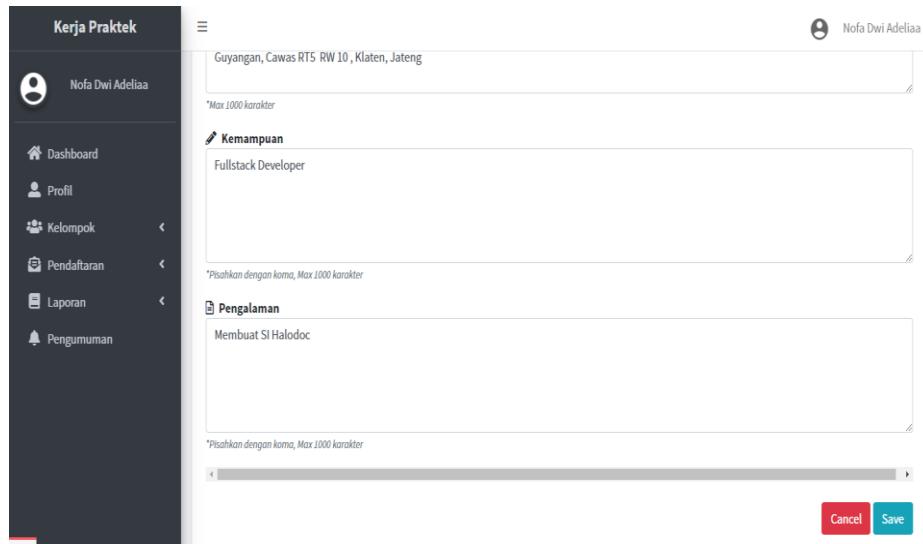
Gambar 5. 28 Kode program halaman profil

Potongan kode program diatas dapat dijelaskan sebagai berikut :

- i. Baris 33 sampai 36 merupakan kode program untuk mengambil data profil mahasiswa.
 - ii. Baris 37 merupakan kode program untuk menampilkan halaman profil.
4. Halaman edit profil

Menampilkan data profil mahasiswa yang akan diubah. Implementasi halaman edit profil dalam Gambar 5.29.

Gambar 5. 29 Implementasi halaman edit profil



```

136     public function edit($id_mahasiswa)
137     {
138         $mahasiswa = Mahasiswa::findOrFail($id_mahasiswa);
139         return view('mahasiswa.profil.editprofil', compact('mahasiswa'));
140     }

```

Gambar 5. 30 Kode program untuk edit profil

Gambar 5.30 merupakan potongan kode program untuk menampilkan halaman edit profil mahasiswa. Potongan kode program diatas dapat dijelaskan sebagai berikut:

- i. Baris 136 sampai 140 kode program untuk menampilkan halaman edit profil.

```

156
157
158    public function update(Request $request, $id_mahasiswa)
159    {
160        $this->validate($request, [
161            'nama' => 'required|string|max:100',
162            'nim' => 'required|string|max:100',
163            'angkatan' => 'required|string|max:4',
164            'email' => 'required|string|max:100',
165            'no_hp' => 'required|string|max:25',
166            'alamat' => 'required|string|max:1000',
167            'kemampuan' => 'required|string|max:1000',
168            'pengalaman' => 'required|string|max:1000',
169            'cv' => 'mimes:pdf|max:3000',
170        ], [
171            'nama.required' => 'nama tidak boleh kosong !',
172            'nim.required' => 'nim tidak boleh kosong !',
173            'angkatan.required' => 'angkatan tidak boleh kosong !',
174            'email.required' => 'email tidak boleh kosong !',
175            'no_hp.required' => 'no_hp tidak boleh kosong !',
176            'alamat.required' => 'alamat tidak boleh kosong !',
177            'alamat.max' => 'alamat terlalu panjang !',
178            'kemampuan.required' => 'kemampuan tidak boleh kosong !',
179            'kemampuan.max' => 'kemampuan terlalu panjang !',
180            'pengalaman.required' => 'pengalaman tidak boleh kosong !',
181            'pengalaman.max' => 'pengalaman terlalu panjang !',
182            'cv.max' => 'File terlalu besar !',
183            'cv.mimes' => 'File harus pdf !'
184        ]);
185
186
187        $data = Mahasiswa::findOrFail($id_mahasiswa);
188        $cv = $data->cv;
189
190
191        if ($request->hasFile('cv')) {
192            !empty($cv) ? File::delete(public_path('uploads/cv' . $cv)):null;
193
194            $files=$request->file('cv');
195            $cv=str_slug($request->nim) . '.' . $files->getClientOriginalExtension();
196            $files->move(public_path('uploads/cv'),$cv);
197
198
199        $data -> update([
200            'nama' => $request->nama,
201            'nim' => $request->nim,
202            'angkatan' => $request->angkatan,
203            'email' => $request->email,
204            'no_hp' => $request->no_hp,
205            'alamat' => $request->alamat,
206            'kemampuan' => $request->kemampuan,
207            'pengalaman' => $request->pengalaman,
208            'cv' => $cv
209        ]);
210        $data->save();
211        return response()->json(['message' => 'Berhasil diubah !']);
212    }

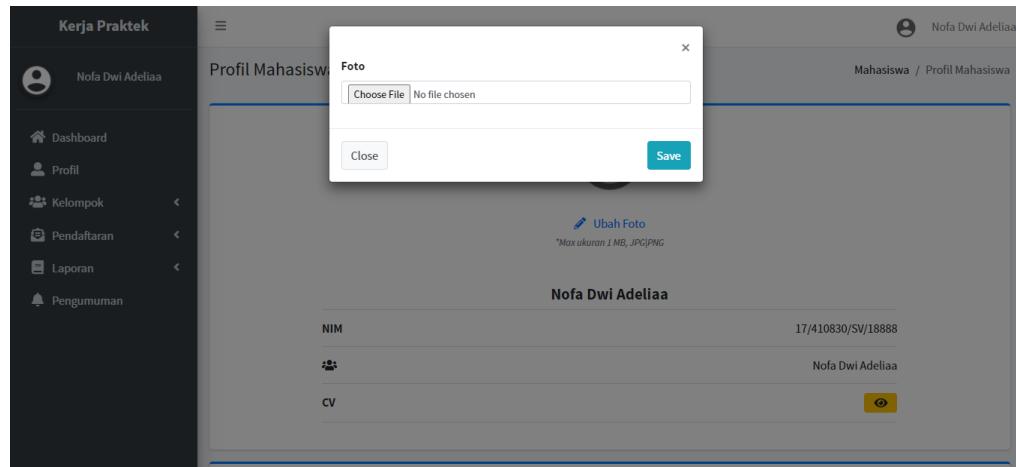
```

Gambar 5.31 Kode program edit profil

Dalam gambar 5.31 merupakan potongan kode program edit profil. Potongan kode program diatas dapat dijelaskan sebagai berikut :

- i. Baris 158 sampai 182 kode program untuk melakukan validasi data yang di inputkan.

- ii. Baris 184 sampai 207 merupakan kode program untuk menyimpan data yang diubah.
- iii. Baris 208 kode program untuk menampilkan notifikasi berhasil diubah.



Gambar 5. 32 Pop up update avatar

```

214     public function updateAvatar(Request $request, $id_mahasiswa){
215         $this->validate($request, [
216             'foto' => 'required|mimes:jpg,png,jpeg|max:1024',
217         ], [
218             'foto.max' => 'File terlalu besar !',
219             'foto.required' => 'File tidak boleh kosong !',
220             'foto.mimes' => 'File harus jpg,png,jpeg'
221         ]);
222
223         $data = Mahasiswa::findOrFail($id_mahasiswa);
224         $foto = $data->foto;
225
226         if ($request->hasFile('foto')) {
227             !empty($foto) ? File::delete(public_path('uploads/fotoprofile/' . $foto)) : null;
228
229             $files=$request->file('foto');
230             $foto=$data->id_mahasiswa;
231             $foto=str_slug($data->nim) . '.' . $files->getClientOriginalExtension();
232             $files->move(public_path('uploads/fotoprofile/'),$foto);
233         }
234         $data->update([
235             'foto' => $foto
236         ]);
237         return response()->json(['data' => $data,
238             'message' => 'Foto profil berhasil diubah.']);
239     }

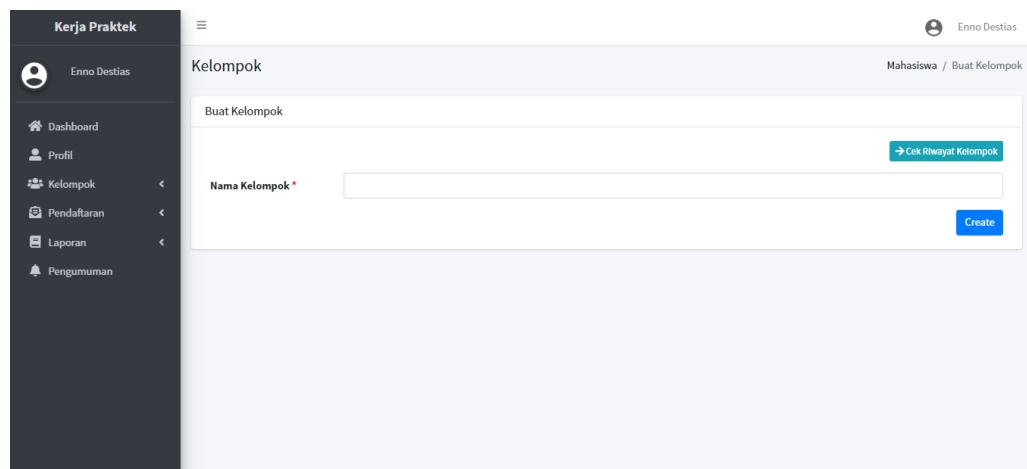
```

Gambar 5. 33 Kode program ubah avatar

Gambar 5.33 diatas merupakan potongan kode program ubah avatar. Potongan kode program diatas memiliki penjelasan sebagai berikut :

- i. Baris 215 sampai 221 merupakan kode program untuk validasi data
 - ii. Baris 223 sampai 236 merupakan kode program untuk menyimpan perubahan foto.
 - iii. Baris 237 sampai 238 merupakan kode program untuk menampilkan notifikasi foto berhasil diupdate.
5. Halaman buat kelompok

Halaman ini merupakan halaman untuk membuat kelompok kerja praktek. Langkah awal dalam membuat kelompok yaitu mahasiswa memasukkan nama kelompok. Implementasi halaman buat kelompok dalam Gambar 5.34.



The screenshot shows the 'Buat Kelompok' (Create Group) page of the 'Kerja Praktek' application. The page has a header with the user's name 'Enno Destias' and a navigation bar with links for 'Mahasiswa / Buat Kelompok', 'Cek Riwayat Kelompok', and 'Create'. On the left, there is a sidebar with links for 'Dashboard', 'Profil', 'Kelompok' (which is the active tab), 'Pendaftaran', 'Laporan', and 'Pengumuman'. The main content area contains a form with a label 'Nama Kelompok' and a red asterisk, followed by an input field and a 'Create' button.

Gambar 5. 34 Implementasi halaman buat kelompok

```

117     public function store(Request $request)
118     {
119         $periode = Periode::select('id_periode')
120             | | | ->where('status', 'open')->first();
121
122         $this->validate($request, [
123             'nama_kelompok' => 'required|string|max:100',
124         ],
125         [
126             'nama_kelompok.required' => 'Nama Kelompok tidak boleh kosong !',
127             'nama_kelompok.max' => 'Nama Kelompok terlalu panjang !'
128         ]);
129
130         $data = Kelompok::create([
131             'nama_kelompok' => $request->nama_kelompok,
132             'id_periode' => $periode->id_periode,
133             'created_by' => $request->created_by,
134         ]);
135
136
137         $data = DetailKelompok::create([
138             'id_kelompok' => $data->id_kelompok,
139             'id_mahasiswa' => $request->id_mahasiswa,
140             'status_keanggotaan' => 'Ketua',
141             'status_join' => 'create',
142             'created_by' => $request->created_by,
143         ]);
144
145         $data->save();
146         return response()->json(['message' => 'Kelompok berhasil ditambahkan.']);
147     }

```

Gambar 5. 35 Kode program create nama kelompok

Gambar 5.35 adalah potongan kode program untuk create nama kelompok.

Potongan kode program diatas memiliki penjelasan sebagai berikut:

- Baris 122 sampai 128 merupakan kode untuk validasi.
- Baris 130 sampai 145 merupakan kode program untuk menyimpan data yang telah diinputkan.

6. Halaman tambah anggota

No	NIM	Nama	Pilih
1	17/410830/SV/12757	Dear Nasitya	+
2	17/410839/SV/12766	Febi Fiolanda Sari S	+
3	17/410830/SV/18888	Nofa Dwi Adeliaa	+
4	17/414430/SV/12757	Marsekal Rama	+

Gambar 5. 36 Implementasi halaman tambah anggota

```

45     public function datamahasiswa()
46     {
47         $userId = Auth::id();
48         $idMahasiswa = Mahasiswa::select('id_mahasiswa')
49             | | | | ->where('mahasiswa.id_users', $userId)->first();
50
51         $idKelompok = DetailKelompok::join('kelompok', 'kelompok_detail.id_kelompok', 'kelompok.id_kelompok')
52             | | | | ->select('kelompok_detail.id_kelompok', 'kelompok.nama_kelompok')
53             | | | | ->where('kelompok_detail.id_mahasiswa', $idMahasiswa->id_mahasiswa)
54             | | | | ->orderBy('id_kelompok', 'desc')->first();
55
56         $periode = Periode::select('id_periode')
57             | | | | ->where('status', 'open')->first();
58
59         $data1 = \DB::table('mahasiswa')
60             ->join('kelompok_detail', 'kelompok_detail.id_mahasiswa', '=', 'mahasiswa.id_mahasiswa')
61             ->where('kelompok_detail.status_join', '!=', 'ditolak')
62             ->where('kelompok_detail.isDeleted', '=', "0")
63             ->select('mahasiswa.id_mahasiswa')
64             ->get();
65         $mahasiswa_memiliki_kelompok = $data1->pluck('id_mahasiswa');
66         $mahasiswa_memiliki_kelompok->all();
67
68
69         $data2 = \DB::table('mahasiswa')
70             ->join('kelompok_detail', 'kelompok_detail.id_mahasiswa', '=', 'mahasiswa.id_mahasiswa')
71             ->join('kelompok', 'kelompok.id_kelompok', '=', 'kelompok_detail.id_kelompok')
72             ->where('kelompok_detail.status_join', '=', "create")
73             ->where('kelompok.tahap', '!=', 'ditolak')
74             ->select('mahasiswa.id_mahasiswa')
75             ->get();
76         $mahasiswa_menjadi_ketua = $data2->pluck('id_mahasiswa');
77         $mahasiswa_menjadi_ketua->all();
78
79         $data3 = \DB::table('mahasiswa')
80             ->join('kelompok_detail', 'kelompok_detail.id_mahasiswa', '=', 'mahasiswa.id_mahasiswa')
81             ->where('kelompok_detail.status_join', '=', "diinvite")
82             ->where('kelompok_detail.isDeleted', '=', "0")
83             ->select('mahasiswa.id_mahasiswa')
84             ->get();
85         $mahasiswa_terinvite_users = $data3->pluck('id_mahasiswa');
86         $mahasiswa_terinvite_users->all();
87
88         $mahasiswa_tersedia = \DB::table('mahasiswa')
89             ->where('mahasiswa.id_periode', '=', $periode->id_periode)
90             ->whereNotIn('mahasiswa.id_mahasiswa', $mahasiswa_memiliki_kelompok)
91             ->whereNotIn('mahasiswa.id_mahasiswa', $mahasiswa_menjadi_ketua)
92             ->whereNotIn('mahasiswa.id_mahasiswa', $mahasiswa_terinvite_users)
93             ->select('mahasiswa.id_mahasiswa', 'mahasiswa.nama', 'mahasiswa.nim')
94             ->get();
95
96         return view('mahasiswa.kelompok.buatkelompok', compact('userId', 'data', 'mahasiswa_tersedia', 'idMahasiswa', 'idKelompok'));
}

```

Gambar 5. 37 Kode program daftar mahasiswa

Gambar 5.37 merupakan potongan kode program untuk menampilkan daftar mahasiswa yang tersedia. Potongan kode program dapat dijelaskan sebagai berikut :

- Baris 59 sampai 66 kode program untuk mengambil data mahasiswa yang memiliki kelompok.
- Baris 68 sampai 76 kode program untuk mengambil data mahasiswa yang menjadi ketua.

- iii. Baris 78 sampai 85 kode program untuk mengambil data mahasiswa yang sudah di invite oleh user.
- iv. Baris 87 sampai 93 kode program untuk mengambil mahasiswa yang masih tersedia.
- v. Baris 95 sampai 96 kode program untuk menampilkan halaman buat kelompok.

```

149  public function addAnggota(Request $request){
150
151      $data = DetailKelompok::create([
152          'id_kelompok' => $request->id_kelompok,
153          'id_mahasiswa' => $request->id_mahasiswa,
154          'status_keanggotaan' => 'Anggota',
155          'status_join' => 'diinvite',
156          'created_by' => $request->created_by,
157      ]);
158      return response()->json(['message' => 'Anggota added successfully.']);
159

```

Gambar 5. 38 Kode program add anggota

Gambar 5.38 merupakan potongan kode program untuk menyimpan anggota yang telah dipilih.

7. Halaman riwayat kelompok

Halaman riwayat kelompok merupakan halaman untuk menampilkan daftar riwayat kelompok mahasiswa. Implementasi halaman riwayat kelompok dalam Gambar 5.39.

No	Nama Kelompok	Status	Aksi
1	Flamboyan	diproses	[Action Button]

Gambar 5. 39 Implementasi halaman riwayat buat kelompok

```

170     public function riwayatkelompok(Request $request)
171     {
172         $userId = Auth::id();
173         $idMahasiswa = Mahasiswa::select('id_mahasiswa')
174             ->where('id_users', $userId)->first();
175         $idKelompok = DetailKelompok::join('kelompok', 'kelompok_detail.id_kelompok', 'kelompok.id_kelompok')
176             ->select('kelompok_detail.id_kelompok', 'kelompok.nama_kelompok')
177             ->where('kelompok_detail.isDeleted', 0)
178             ->where('kelompok_detail.id_mahasiswa', $idMahasiswa->id_mahasiswa)->first();
179         $kelompok = Kelompok::get();
180         if($request()->ajax()){
181             $data = [];
182             if ($idKelompok){
183                 $data = DetailKelompok::join('kelompok', 'kelompok_detail.id_kelompok', 'kelompok.id_kelompok')
184                     ->select('kelompok.*')
185                     ->where('kelompok_detail.isDeleted', 0)
186                     ->where('kelompok_detail.id_mahasiswa', $idMahasiswa->id_mahasiswa)->get();
187             }
188             return datatables()->of($data)->addIndexColumn()
189                 ->addColumn('action', function ($kelompok){
190                     if ($kelompok !=null){
191                         $disable = $kelompok->tahap != 'diproses'? "disabled" : " ";
192                         $btn = '<a href="'.$mahasiswa/daftaranggota/'.$kelompok->id_kelompok.'" class="btn btn-info
193                         btn-sm '$disable.'><i class="fas fa-list"></i></a>';
194                         return $btn;
195                     }
196                 })
197                 ->rawColumns(['action'])
198                 ->addIndexColumn()
199                 ->make(true);
200         }
200     return view('mahasiswa.kelompok.indexkelompok', compact('kelompok'));

```

Gambar 5. 40 Kode program halaman riwayat kelompok

Gambar 5.40 merupakan kode program untuk menampilkan daftar riwayat kelompok ke dalam tabel. Potongan kode program diatas dapat dijelaskan sebagai berikut :

- i. Baris 180 sampai 186 adalah kode program untuk mengambil data kelompok berdasarkan id mahasiswa yang login.
 - ii. Baris 188 sampai 199 merupakan kode program untuk menampilkan data kedalam datatable.
 - iii. Baris 200 kode program untuk menampilkan halaman riwayat kelompok
8. Halaman daftar anggota kelompok

Halaman daftar anggota kelompok menampilkan nama dan nim daftar anggota kelompok. Dalam halaman ini ketua bisa menghapus anggota jika kelompok masih dalam tahap diproses. Implementasi halaman daftar anggota kelompok dalam Gambar 5.41.

No	NIM	Nama Mahasiswa	Status	Aksi
1	17/414430/SV/12757	Marsekal Rama	Anggota	

Gambar 5. 41 Implementasi halaman daftar anggota kelompok

```

166     public function daftaranggota(Request $id_kelompok, $id)
167     {
168         $userId = Auth::id();
169         $idMahasiswa = Mahasiswa::select('id_mahasiswa')
170             | | | | ->where('id_users', $userId)->first();
171
172         $idKelompok = DetailKelompok::join('kelompok', 'kelompok_detail.id_kelompok', 'kelompok.id_kelompok')
173             | | | | ->select('kelompok_detail.id_kelompok', 'kelompok.nama_kelompok', 'kelompok_detail.
174             | | | | id_kelompok_detail')
175             | | | ->where('kelompok.id_kelompok', $id)
176             | | | ->where('kelompok_detail.isDeleted', 0)
177             | | | ->first();
178
179         $statusKeanggotaan = @DetailKelompok::select('kelompok_detail.status_keanggotaan')
180             | | | ->where('kelompok_detail.id_mahasiswa', $idMahasiswa->id_mahasiswa)
181             | | | ->orderBy('id_kelompok', 'desc')->first();
182
183         $anggota = Mahasiswa::get();
184         if(request()->ajax()){
185             $data = [];
186             if ($idKelompok){
187                 $data = Mahasiswa::leftJoin('kelompok_detail', 'mahasiswa.id_mahasiswa', 'kelompok_detail.id_mahasiswa')
188                     ->LeftJoin('kelompok', 'kelompok_detail.id_kelompok', 'kelompok.id_kelompok')
189                     ->select('kelompok.id_kelompok', 'mahasiswa.id_mahasiswa', 'mahasiswa.nama', 'mahasiswa.nim', 'kelompok_detail.
190                     status_keanggotaan', 'kelompok_detail.id_kelompok_detail')
191                     ->where('kelompok_detail.status_keanggotaan', '!=', 'Ketua')
192                     ->where('kelompok_detail.status_join', '!=', 'ditolak')
193                     ->where('kelompok_detail.id_kelompok', $idKelompok->id_kelompok)
194                     ->where('kelompok_detail.isDeleted', 0)
195                     ->get();
196             }
197         }
198     }

```

```

195 |     if(@$statusKeanggotaan->status_keanggotaan == 'Ketua')
196 |     {
197 |         return datatables()->of($data)->addIndexColumn()
198 |         ->addColumn('action', function ($data){
199 |             if($data != null){
200 |                 $btn = '<button type="button" name="delete" id="" . $data->id_kelompok_detail . '" class="btn btn-danger btn-sm
201 |                 deleteAnggota "><i class="fas fa-trash">/i></button>';
202 |                 return $btn;
203 |             }
204 |             return;
205 |         })
206 |         ->rawColumns(['action'])
207 |         ->addIndexColumn()
208 |         ->make(true);
209 |     }
210 |     else{
211 |         return datatables()->of($data)->addIndexColumn()
212 |         ->addColumn('action', function ($data){
213 |             if($data != null){
214 |                 $btn = '<a href="" type="button" class="btn btn-danger btn-sm disabled "><i class="fas fa-trash">/i></a>';
215 |                 return $btn;
216 |             }
217 |             return;
218 |         })
219 |         ->rawColumns(['action'])
220 |         ->addIndexColumn()
221 |         ->make(true);
222 |     }
223 |     return view('mahasiswa.kelompok.daftaranggota', compact('anggota','idKelompok','statusKeanggotaan'));
224 |
---
```

Gambar 5. 42 Kode program halaman daftar anggota kelompok

Gambar 5.42 diatas adalah potongan kode program berfungsi menampilkan halaman daftar anggota kelompok. Penjelasan kode program diatas sebagai berikut :

- i. Baris 168 sampai 176 merupakan kode program untuk mengambil id kelompok.
- ii. Baris 178 sampai 180 merupakan kode program untuk mengambil data status keanggotaan.
- iii. Baris 184 sampai 194 merupakan kode program untuk mengambil data mahasiswa yang berada dalam kelompok.
- iv. Baris 195 sampai 208 merupakan kode program untuk menampilkan button delete jika user memiliki status keanggotaan ketua.
- v. Baris 209 sampai 222 merupakan kode program untuk disable button delete jika status keanggotaan user adalah anggota.
- vi. Baris 223 merupakan kode program untuk menampilkan halaman daftar anggota kelompok.

```

226     public function kick($id_kelompok_detail)
227     {
228         $anggota = DetailKelompok::findOrFail($id_kelompok_detail);
229         $anggota->isDeleted = '1';
230         $anggota->save();
231         return response()->json(['message' => 'Anggota berhasil dihapus.']);
232     }

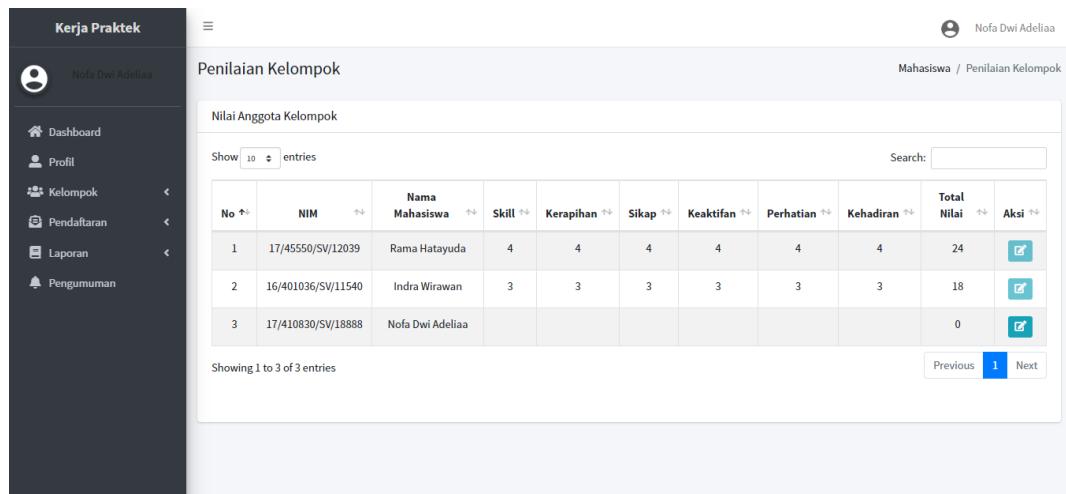
```

Gambar 5. 43 Kode program delete anggota kelompok

Gambar 5.43 merupakan kode program untuk delete undangan anggota kelompok.

9. Halaman penilaian anggota kelompok

Halaman penilaian anggota kelompok adalah halaman untuk menampilkan data nilai anggota kelompok. Implementasi halaman penilaian anggota kelompok dalam Gambar 5.44.



The screenshot shows a web application interface titled 'Penilaian Kelompok'. On the left, there is a sidebar with a user profile picture and the name 'Nofa Dwi Adeliaa'. Below the profile are navigation links: 'Dashboard', 'Profil', 'Kelompok', 'Pendaftaran', 'Laporan', and 'Pengumuman'. The main content area has a header 'Nilai Anggota Kelompok'. Below it is a search bar with 'Search:' and a dropdown for 'Show' entries. A table displays student scores across various criteria. The table has columns for 'No', 'NIM', 'Nama Mahasiswa', 'Skill', 'Kerapihan', 'Sikap', 'Keaktifan', 'Perhatian', 'Kehadiran', 'Total Nilai', and 'Aksi'. Three rows of data are shown:

No	NIM	Nama Mahasiswa	Skill	Kerapihan	Sikap	Keaktifan	Perhatian	Kehadiran	Total Nilai	Aksi
1	17/45550/SV/12039	Rama Hatayuda	4	4	4	4	4	4	24	<input type="button" value="Edit"/>
2	16/401036/SV/11540	Indra Wirawan	3	3	3	3	3	3	18	<input type="button" value="Edit"/>
3	17/410830/SV/18888	Nofa Dwi Adeliaa							0	<input type="button" value="Edit"/>

At the bottom, a message says 'Showing 1 to 3 of 3 entries' and includes 'Previous' and 'Next' buttons.

Gambar 5. 44 Implementasi halaman penilaian anggota_kelompok

```

23 |     public function show()
24 |     {
25 |
26 |         $userId = Auth::id();
27 |
28 |         $idMahasiswa = Mahasiswa::select('id_mahasiswa')
29 |                               ->where('id_users', $userId)->first();
30 |
31 |         $idKelompok = DetailKelompok::join('kelompok', 'kelompok_detail.id_kelompok', 'kelompok.id_kelompok')
32 |                               ->select('kelompok_detail.id_kelompok', 'kelompok.nama_kelompok')
33 |                               ->where('kelompok_detail.isDeleted', '=', '0')
34 |                               ->where('kelompok.tahap', '!=', 'ditolak')
35 |                               ->orderBy('id_kelompok', 'desc')
36 |                               ->where('kelompok_detail.id_mahasiswa', $idMahasiswa->id_mahasiswa)->first();
37 |
38 |         $data = [];
39 |         if ($idKelompok){
40 |             $data = Mahasiswa::leftJoin('kelompok_detail', 'mahasiswa.id_mahasiswa', 'kelompok_detail.
41 |                                         id_mahasiswa')
42 |                                         ->LeftJoin('kelompok', 'kelompok_detail.id_kelompok', 'kelompok.id_kelompok')
43 |                                         ->select('kelompok.id_kelompok', 'mahasiswa.id_mahasiswa', 'mahasiswa.nama',
44 |                                                 'kelompok_detail.status_keanggotaan', 'mahasiswa.nim')
45 |                                         ->where('kelompok_detail.id_kelompok', $idKelompok->id_kelompok)
46 |                                         ->where('kelompok_detail.status_join', '!=', 'ditolak')
47 |                                         ->where('kelompok_detail.isDeleted', '!=', '1')
48 |                                         ->get();
49 |
50 |         }
51 |
52 |         return view('mahasiswa.penilaian.indexpenilaian', compact('data', 'userId'));

```

Gambar 5.45 Kode program halaman daftar penilaian anggota kelompok

Gambar 5.45 merupakan potongan kode program untuk menampilkan daftar anggota kelompok. Potongan kode program diatas dapat memiliki penjelasan sebagai berikut :

- i. Baris 31 sampai 36 merupakan kode program untuk mengambil id kelompok sesuai dengan mahasiswa yang login.
- ii. Baris 40 sampai 46 merupakan kode program untuk mengambil data mahasiswa yang berada dalam kelompok yang sama dengan mahasiswa yang login.
- iii. Baris 49 merupakan kode program untuk menampilkan data ke halaman index penilaian.

10. Halaman form penilaian

Halaman form penilaian merupakan halaman yang menampilkan form untuk memberikan nilai kepada anggota kelompok. Implementasi halaman form penilaian dalam Gambar 5.46.

Gambar 5.46 Implementasi halaman form penilaian

```

85     public function edit($id_mahasiswa)
86     {
87         $anggota = Mahasiswa::findOrFail($id_mahasiswa);
88         $userId = Auth::id();
89         return view('mahasiswa.penilaian.formnilai', compact('anggota','userId'));
90     }
91
92     public function store(Request $request)
93     {
94         $periode=Periode::where(['status'=>'open'])->first();
95         $userId = Auth::id();
96
97         foreach($request->id_aspek_penilaian as $key => $value)
98         {
99             $model = new Nilai;
100            $model->id_aspek_penilaian = $value;
101            $model->nilai = $request->nilai[$key];
102            $model->id_periode = $periode->id_periode;
103            $model->id_kelompok_penilai=$request->id_kelompok_penilai;
104            $model->id_mahasiswa = $request->id_mahasiswa;
105            $model->created_by= $request->created_by;
106            $model->save();
107        }
108    }
109
110
111
112
113
114
115
116
117
118

```

Gambar 5.47 Implementasi halaman form penilaian

Gambar 5.47 adalah potongan kode program yang berfungsi menampilkan form penilaian dan untuk menambahkan nilai. Potongan kode program diatas dapat dijelaskan sebagai berikut :

- i. Baris 85 sampai 90 merupakan kode program untuk menampilkan form penilaian.
- ii. Baris 92 sampai 108 merupakan kode program untuk menyimpan inputan nilai.

11. Halaman detail kelompok tab anggota

Halaman ini menampilkan data kelompok yang meliputi nama kelompok, anggota kelompok, jabatan anggota, nama, nim, angkatan dan foto anggota kelompok. Halaman detail kelompok tab anggota dalam Gambar 5.48.

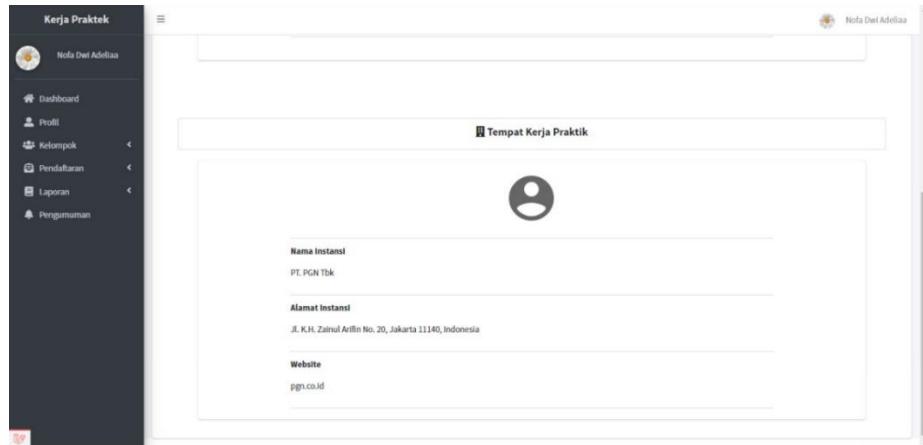
Anggota	NIM	Angkatan	Status
Dear Nasyita	17/410830/SV/12757	2017	Ketua
Febi Fiolanda Sari S	17/410839/SV/12766	2017	Anggota
Nofa Dwi Adeliaa	17/410830/SV/18888	2017	Anggota

Gambar 5. 48 Implementasi halaman detail kelompok tab anggota

12. Halaman detail kelompok tab info kerja praktek

Halaman ini menampilkan informasi tempat kerja praktek dan dosen pembimbing kerja praktek. Implementasi halaman detail kelompok tab info kerja praktek dalam Gambar 5.49

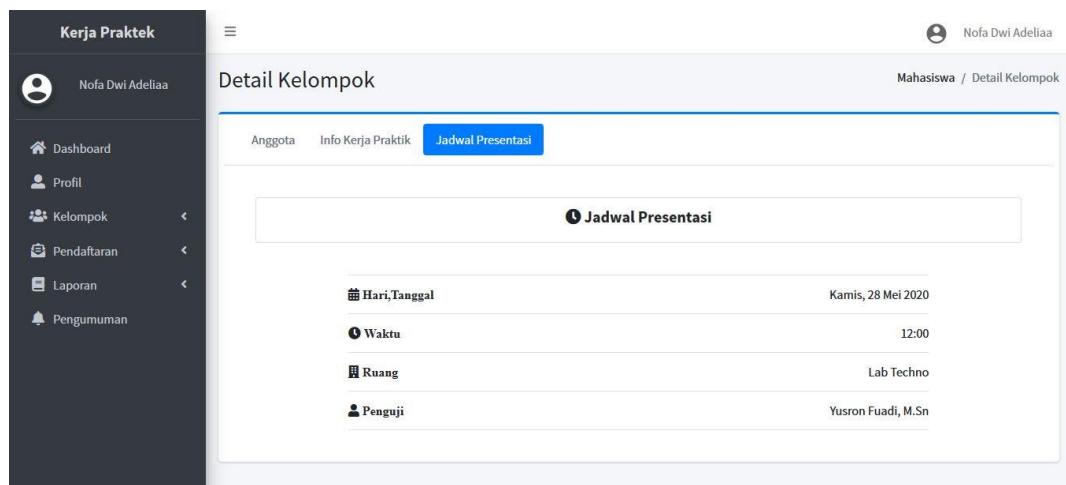
Info Kerja Praktik	Detail
Name	Yusron Fuadi, M.Si
NIP	1000000000
Email	yusron@ugm.ac.id
No.Hp	08999999999



Gambar 5. 49 Implementasi halaman detail kelompok tab info kerja praktek

13. Halaman detail kelompok tab jadwal presentasi

Halaman detail kelompok tab jadwal presentasi merupakan halaman untuk menampilkan jadwal presentasi kelompok yang telah selesai melakukan kerja praktek. Halaman tab jadwal presentasi menampilkan hari dan tanggal, waktu, ruang, dan penguji.



Gambar 5. 50 Halaman detail kelompok tab jadwal presentasi

```

57     public function show()
58     {
59         //initial var;
60         $jadwal = $detail_kelompok = $mahasiswa = $kelompok = $anggota = $magang = $instansi = [];
61         $auth = auth()->user()->id_users;
62
63         $mahasiswa = Mahasiswa::where('id_users', $auth)
64             ->first();
65
66         $id_mahasiswa = $mahasiswa != null ? $mahasiswa->id_mahasiswa : [];
67
68         if ($id_mahasiswa) {
69
70             $group = DetailKelompok::where('id_mahasiswa', $id_mahasiswa)
71                 ->where('status_join', '!=', 'ditolak')
72                 ->where('isDeleted', '!=', '1')
73                 ->orderBy('id_kelompok_detail', 'desc')
74                 ->first();
75
76             $id_kelompok = $group != null ? $group->id_kelompok : [];
77
78             $mahasiswa = Mahasiswa::findOrFail($id_mahasiswa);
79             $magang = Mahasiswa::leftJoin('kelompok_detail', 'mahasiswa.id_mahasiswa', 'kelompok_detail.
80             id_mahasiswa')
81                 ->leftJoin('kelompok', 'kelompok_detail.id_kelompok', 'kelompok.id_kelompok')
82                 ->leftJoin('dosen', 'kelompok.id_dosen', 'dosen.id_dosen')
83                 ->select('dosen.nama', 'dosen.email', 'dosen.nip', 'dosen.no_hp', 'dosen.foto')
84                 ->where('mahasiswa.id_mahasiswa', $id_mahasiswa)
85                 ->first();
86
86             $instansi = Mahasiswa::join('kelompok_detail', 'mahasiswa.id_mahasiswa', 'kelompok_detail.id_mahasiswa')
87                 ->join('kelompok', 'kelompok_detail.id_kelompok', 'kelompok.id_kelompok')
88                 ->join('magang', 'kelompok.id_kelompok', 'magang.id_kelompok')
89                 ->join('instansi', 'magang.id_instansi', 'instansi.id_instansi')
90                 ->select('instansi.nama', 'instansi.deskripsi', 'instansi.alamat', 'instansi.foto', 'instansi.
91                 website')
92                 ->where('mahasiswa.id_mahasiswa', $id_mahasiswa)
93                 ->first();
94
94             $kelompok = Mahasiswa::leftJoin('kelompok_detail', 'mahasiswa.id_mahasiswa', 'kelompok_detail.
95             id_mahasiswa')
96                 ->leftJoin('kelompok', 'kelompok_detail.id_kelompok', 'kelompok.id_kelompok')
97                 ->select('kelompok.nama_kelompok', 'kelompok.tahap', 'kelompok_detail.status_keanggotaan')
98                 ->where('mahasiswa.id_mahasiswa', $id_mahasiswa)
99                 ->get();
100
100         if ($id_kelompok) {
101
102             $detail_kelompok = Kelompok::where('id_kelompok', $id_kelompok)
103                 ->first();
104
105             $jadwal = Jadwal::join('dosen', 'dosen.id_dosen', 'jadwal_presentasi.id_dospeng')
106                 ->join('sesiwaktu', 'sesiwaktu.id_sesiwaktu', 'jadwal_presentasi.id_sesiwaktu')
107                 ->join('ruang', 'ruang.id_ruang', 'jadwal_presentasi.id_ruang')
108                 ->where('jadwal_presentasi.id_kelompok', $id_kelompok)
109                 ->select('jadwal_presentasi.tanggal', 'sesiwaktu.sesi', 'ruang.ruang', 'dosen.nama')
110                 ->first();
111
111     }

```

```

113     $anggota = Mahasiswa::leftJoin('kelompok_detail', 'mahasiswa.id_mahasiswa', 'kelompok_detail.
114     id_mahasiswa')
115     ->LeftJoin('kelompok', 'kelompok_detail.id_kelompok', 'kelompok.id_kelompok')
116     ->select('kelompok.id_kelompok', 'mahasiswa.id_mahasiswa', 'mahasiswa.nama', 'mahasiswa.
117     foto', 'mahasiswa.nim', 'mahasiswa.angkatan', 'kelompok_detail.status_keanggotaan')
118     ->where('kelompok_detail.status_join', '!=', 'ditolak')
119     ->where('kelompok_detail.isDeleted', '!=', '1')
120     ->where('kelompok_detail.id_kelompok', $id_kelompok)
121     ->get();
122 }
123 return view('mahasiswa.kelompok.detailkelompok', compact(
124     'jadwal',
125     'detail_kelompok',
126     'mahasiswa',
127     'kelompok',
128     'anggota',
129     'magang',
130     'instansi'
131 ));
}

```

Gambar 5. 51 Kode program untuk menampilkan halaman detail kelompok

Gambar 5.51 merupakan kode program untuk menampilkan halaman detail kelompok. Penjelasan potongan kode program diatas dapat dijelaskan sebagai berikut :

- i. Baris 66 adalah kode program untuk mengambil id mahasiswa yang login.
- ii. Baris 76 adalah kode program untuk mengambil id kelompok.
- iii. Baris 79 sampai 84 kode program untuk mengambil data dosen pembimbing kerja praktek.
- iv. Baris 86 sampai 92 kode program untuk mengambil data instansi tempat kerja praktek.
- v. Baris 94 sampai 98 kode program untuk mengambil data kelompok kerja praktek.
- vi. Baris 106 sampai 111 kode program untuk mengambil data jadwal presentasi.
- vii. Baris 113 sampai 119 kode program untuk mengambil data anggota kelompok.
- viii. Baris 122 sampai 130 kode program untuk menampilkan halaman detail kelompok.

14. Halaman detail usulan instansi

Halaman daftar usulan instansi merupakan halaman yang menampilkan data usulan instansi yang pernah ditambahkan. Implementasi halaman daftar usulan instansi dapat di lihat dalam.Gambar.5.52.

The screenshot shows a user interface for managing institutional applications. On the left is a dark sidebar menu titled 'Kerja Praktek' with the user's name 'Nofa Dwi Adeliaa'. The main content area is titled 'Usulan Instansi' and displays a table of one entry. The table columns are: No, Nama Instansi, Website, Alamat, Jobdesk, Deskripsi, Status, and Aksi. The entry details are: No 1, Nama Instansi PT. PGN, Website www.kai.id, Alamat Jakarta, Jobdesk Membuat SI Absensi karyawan, Deskripsi PT PGN Adalah, Status sedang diproses, and Aksi (with edit and delete icons). Navigation buttons at the bottom include 'Previous 1 Next'.

No	Nama Instansi	Website	Alamat	Jobdesk	Deskripsi	Status	Aksi
1	PT. PGN	www.kai.id	Jakarta	Membuat SI Absensi karyawan	PT PGN Adalah	sedang diproses	

Gambar 5. 52 Implementasi halaman detail usulan instansi

```

34     public function index(Request $request)
35     {
36         // Get id user from Auth
37         $userId = Auth::id();
38
39         $idMahasiswa = Mahasiswa::select('id_mahasiswa')
40             ->where('id_users', $userId)->first();
41
42         $idKelompok = DetailKelompok::select('kelompok_detail.id_kelompok')
43             ->where('kelompok_detail.id_mahasiswa', $idMahasiswa->id_mahasiswa)
44             ->orderBy('kelompok_detail.id_kelompok_detail','desc')
45             ->where('kelompok_detail.isDeleted', '=', '0')->first();
46
47         $statusKeanggotaan = @DetailKelompok::select('kelompok_detail.status_keanggotaan')
48             ->where('kelompok_detail.id_mahasiswa', $idMahasiswa->id_mahasiswa)
49             ->orderBy('id_kelompok', 'desc')->first();
50
51         $usulan = Usulan::get();
52
53         $statusUsulan = @Usulan::leftJoin('kelompok', 'usulan.id_kelompok', '=', 'kelompok.id_kelompok')
54             ->select('usulan.status')
55             ->where('usulan.id_kelompok', $idKelompok->id_kelompok)->first();
56
57         $statusLamaran = @Lamaran::select('pelamar.status')
58             ->where('pelamar.id_kelompok', $idKelompok->id_kelompok)->first();
59
60         if (request()->ajax()) {
61             $data = [];
62             if ($idKelompok){
63                 $data = Usulan::leftJoin('kelompok', 'usulan.id_kelompok', '=', 'kelompok.id_kelompok')
64                     ->select('usulan.*')
65                     ->where('usulan.id_kelompok', $idKelompok->id_kelompok)->get();
66             }
67         if(@$statusKeanggotaan->status_keanggotaan == 'Ketua'){
68             return datatables()->of($data)->addIndexColumn()
69                 ->addColumn('action', function ($data) {
70                     if($data != null ){
71                         $disable = $data->status != 'diproses' ? "disabled" : " ";
72                         $btn = '<a href="#"mahasiswa/editusulan/' . $data->id_usulan . '" class="btn btn-info btn-sm ' . $disable .
73                             '<i class="fas fa-edit"></i></a>';
74                         $btn .= '&ampnbsp&ampnbsp';
75                         $btn .= '<button type="button" name="show" id="" . $data->id_usulan . '" class="btn btn-warning btn-sm surat" ><i class="fas fa-eye"></i></button>';
76                         return $btn;
77                     }
78                 })
79                 ->rawColumns(['action'])
80                 ->addIndexColumn()
81                 ->make(true);
82             else{
83                 return datatables()->of($data)->addIndexColumn()
84                     ->addColumn('action', function ($data) {
85                         if($data != null ){
86                             $btn = '<a href="#" class="btn btn-info btn-sm disabled"><i class="fas fa-edit"></i></a>';
87                             $btn .= '&ampnbsp&ampnbsp';
88                             $btn .= '<button type="button" name="show" id="" . $data->id_usulan . '" class="btn btn-warning btn-sm surat" ><i class="fas fa-eye"></i></button>';
89                             return $btn;
90                         }
91                     })
92                     ->rawColumns(['action'])
93                     ->addIndexColumn()
94                     ->make(true);
95                 }
96             return view('mahasiswa.usulan.indexusulan', compact('usulan', 'statusKeanggotaan', 'statusUsulan', 'statusLamaran'));
97         }
98     }

```

Gambar 5.53 Kode program halaman daftar usulan instansi

Gambar 5.53 adalah potongan kode program untuk menampilkan daftar data usulan instansi. Potongan kode program diatas dapat dijelaskan sebagai berikut :

- i. Baris 37 sampai 45 merupakan kode program untuk mengambil id kelompok sesuai dengan mahasiswa yang login.
- ii. Baris 47 sampai 49 merupakan kode program untuk mengambil status keanggotaan.
- iii. Baris 51 merupakan kode program untuk mengambil data usulan instansi.
- iv. Baris 59 sampai 82 merupakan kode program untuk mengambil data usulan instansi.
- v. Baris 63 sampai 65 kode program untuk menampilkan data ke dalam tabel.
- vi. Baris 67 sampai 79 kode program untuk menampilkan button edit jika user berstatus ketua.
- vii. Baris 80 sampai 92 kode program untuk disable button edit jika user berstatus anggota.
- viii. Baris 93 kode program untuk menampilkan halaman daftar usulan instansi.

15. Halaman tambah usulan instansi

Halaman tambah usulan instansi menampilkan form nama instansi, website, alamat, jobdesk, deskripsi dan surat pemberitahuan dari instansi. Implementasi dari halaman tambah usulan instansi dalam Gambar 5.54.

The screenshot shows a web application interface titled "Tambah Usulan". The left sidebar has a navigation menu with items like "Dashboard", "Profil", "Kelompok", "Penilaian", "Laporan", and "Pengumuman". The main content area is titled "Tambah Usulan" and contains the following fields:

- Nama Instansi ***: Input field
- Website ***: Input field
- Alamat ***: Input field
- Jobdesk ***: Textarea with placeholder "Penjelasan singkat jobdesk saat kerja praktik". Note: "Maksimum dengan 2000 karakter".
- Deskripsi ***: Textarea with placeholder "Penjelasan singkat tentang Instansi". Note: "Maksimum dengan 2000 karakter".
- Surat Pemberitahuan ***: Input field containing "Teksmu... Teks ada terlebih dipilih." Note: "Surat Survei PDF, Max ukuran 1 MB".

At the bottom right are "Cancel" and "Save" buttons.

Gambar 5. 54 Implementasi halaman tambah usulan instansi

```

97     public function create()
98     {
99         $userId = Auth::id();
100        // Mengambil id_mahasiswa yang sesuai dengan id yang login sekarang
101        $idMahasiswa = Mahasiswa::select('id_mahasiswa', 'id_periode')
102            ->where('id_users', $userId)->first();
103
104        $idKelompok = DetailKelompok::select('kelompok_detail.id_kelompok')
105            ->where('kelompok_detail.id_mahasiswa', $idMahasiswa->id_mahasiswa)
106            ->orderBy('id_kelompok', 'desc')->first();
107
108        $periode = Periode::select('id_periode')
109            ->where('status', 'open')->first();
110
111        $datas = Usulan::get();
112        return view('mahasiswa.usulan.tambahusulan', compact('datas', 'idKelompok', 'periode', 'userId'));
113    }

```

Gambar 5.55 Kode program halaman tambah usulan instansi

Gambar 5.55 adalah potongan kode program untuk menampilkan halaman tambah usulan instansi. Potongan kode program diatas dapat dijelaskan sebagai berikut :

- i. Baris 99 sampai 106 merupakan kode program untuk mengambil id kelompok sesuai dengan mahasiswa yang login.
- ii. Baris 108 sampai 109 merupakan kode program untuk mengambil id periode sesuai dengan periode berstatus open.
- iii. Baris 111 merupakan kode program untuk mengambil data usulan.
- iv. Baris 112 merupakan kode program untuk menampilkan halaman tambah usulan instansi.

```

125     public function store(Request $request)
126     {
127         $this->validate($request, [
128             'nama_instansi' => 'required|string|max:100',
129             'website_instansi' => 'required|string|max:100',
130             'alamat_instansi' => 'required|string|max:1000',
131             'jobdesk' => 'required|string|max:100',
132             'deskripsi_instansi' => 'required|string|max:1000',
133             'surat' => 'required|mimes:pdf|max:3072',
134         ],
135         [
136             'nama_instansi.required' => 'Nama instansi tidak boleh kosong !',
137             'nama_instansi.max' => 'Nama instansi terlalu panjang !',
138             'website_instansi.required' => 'Website instansi tidak boleh kosong !',
139             'website_instansi.max' => 'Website instansi terlalu panjang !',
140             'alamat_instansi.required' => 'Alamat instansi tidak boleh kosong !',
141             'alamat_instansi.max' => 'Alamat instansi terlalu panjang !',
142             'jobdesk.required' => 'Jobdesk tidak boleh kosong !',
143             'jobdesk.max' => 'Jobdesk terlalu panjang !',
144             'deskripsi_instansi.required' => 'Deskripsi instansi tidak boleh kosong !',
145             'deskripsi_instansi.max' => 'Deskripsi instansi terlalu panjang !',
146             'surat.required' => 'File tidak boleh kosong !',
147             'surat.max' => 'File terlalu besar !',
148             'surat.mimes' => 'File harus pdf'
149         ]);
150
151         $surat = null;
152         if ($request->hasFile('surat')) {
153             $files = $request->file('surat');
154             $surat = str_slug('Kelompok-' . $request->id_kelompok) . '.' . $files->getClientOriginalExtension();
155             $files->move(public_path('uploads/suratpersetujuan'), $surat);
156         }
157
158         $data = Usulan::create([
159             'nama_instansi' => $request->nama_instansi,
160             'website_instansi' => $request->website_instansi,
161             'alamat_instansi' => $request->alamat_instansi,
162             'jobdesk' => $request->jobdesk,
163             'deskripsi_instansi' => $request->deskripsi_instansi,
164             'surat' => $surat,
165             'created_by' => $request->created_by,
166             'id_periode' => $request->id_periode,
167             'id_kelompok' => $request->id_kelompok,
168         ]);
169
170         $data->save();
171         return response()->json(['message' => 'Usulan berhasil ditambahkan']);
172     }

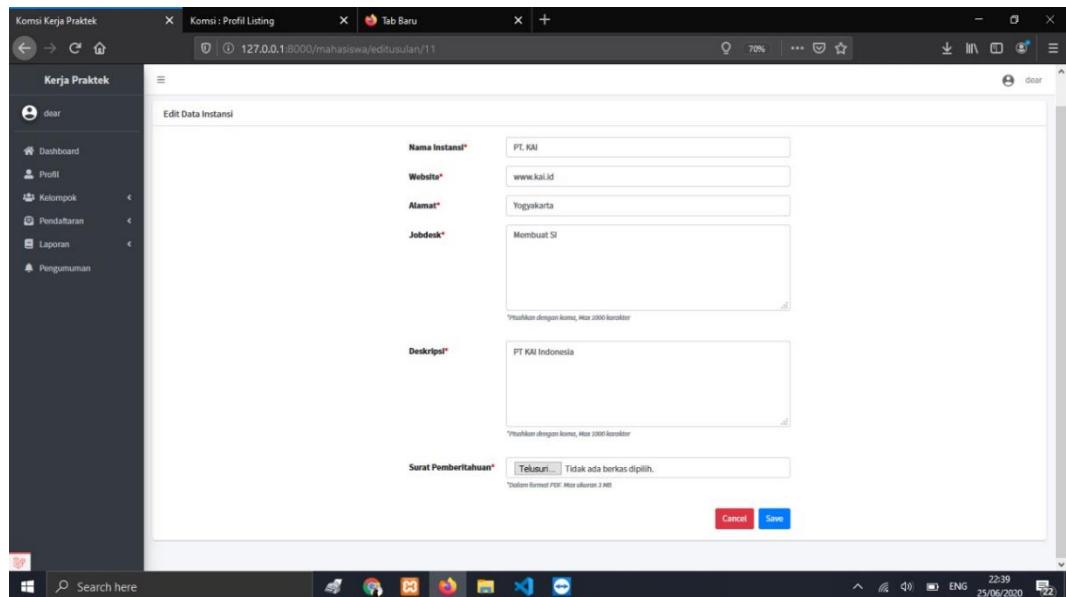
```

Gambar 5.56 Kode program tambah usulan instansi

Gambar 5.56 merupakan potongan kode program untuk menambahkan usulan instansi. Potongan kode program diatas dapat dijelaskan sebagai berikut :

- Baris 127 sampai 149 merupakan kode program untuk validasi data.
- Baris 151 sampai 169 merupakan kode program untuk menyimpan data yang diinputkan.
- Baris 170 merupakan kode program untuk menampilkan pesan berhasil ditambahkan.

16. Halaman edit usulan instansi



Gambar 5.57 Implementasi edit usulan instansi

```

186     public function edit($id_usulan)
187     {
188         // Get id user from Auth
189         $userId = Auth::id();
190
191         // Mengambil id_mahasiswa yang sesuai dengan id yang login sekarang
192         $idMahasiswa = Mahasiswa::select('id_mahasiswa')
193             ->where('mahasiswa.id_users', $userId)->first();
194         $idKelompok = DetailKelompok::select('kelompok_detail.id_kelompok')
195             ->where('kelompok_detail.id_mahasiswa', $idMahasiswa->id_mahasiswa)
196             ->orderBy('id_kelompok','desc')->first();
197
198         $data = Usulan::findOrFail($id_usulan);
199
200     return view('mahasiswa.usulan.editusulan', compact('data', 'idKelompok'));
201 }

```

Gambar 5.58 Kode program halaman edit usulan instansi

Gambar 5.58 adalah potongan kode program untuk menampilkan halaman edit usulan instansi. Potongan kode program tersebut dapat dijelaskan sebagai berikut :

- Baris 189 sampai 196 merupakan kode program untuk mengambil id kelompok sesuai dengan kelompok mahasiswa yang login.
- Baris 198 merupakan kode program untuk mengambil data usulan instansi sesuai dengan id usulan yang dipilih.

- iii. Baris 200 merupakan kode program untuk menampilkan halaman edit usulan instansi.

```

215 public function update(Request $request, $id_usulan)
216 {
217     $this->validate($request,
218     [
219         'nama_instansi' => 'required|string|max:100',
220         'website_instansi' => 'required|string|max:100',
221         'alamat_instansi' => 'required|string|max:1000',
222         'jobdesk' => 'required|string|max:100',
223         'deskripsi_instansi' => 'required|string|max:100',
224         'surat' => 'mimes:pdf|max:3072',
225     ],
226     [
227         'nama_instansi.required' => 'Nama instansi tidak boleh kosong !',
228         'nama_instansi.max' => 'Nama instansi terlalu panjang !',
229         'website_instansi.required' => 'Website instansi tidak boleh kosong !',
230         'website_instansi.max' => 'Website instansi terlalu panjang !',
231         'alamat_instansi.required' => 'Alamat instansi tidak boleh kosong !',
232         'alamat_instansi.max' => 'Alamat instansi terlalu panjang !',
233         'jobdesk.required' => 'Jobdesk tidak boleh kosong !',
234         'jobdesk.max' => 'Jobdesk terlalu panjang !',
235         'deskripsi_instansi.required' => 'Deskripsi instansi tidak boleh kosong !',
236         'deskripsi_instansi.max' => 'Deskripsi instansi terlalu panjang !',
237         'surat.max' => 'File terlalu besar !',
238         'surat.mimes' => 'File harus pdf'
239     ]);
240
241     $data = Usulan::findOrFail($id_usulan);
242     $surat = $data->surat;
243     if ($request->hasFile('surat')) {
244         !empty($surat) ? File::delete(public_path('uploads/surat' . $surat)) : null;
245
246         $files = $request->file('surat');
247         $surat = str_slug('Kelompok-' . $request->id_kelompok) . '.' . $files->getClientOriginalExtension();
248         $files->move(public_path('uploads/suratpersetujuan'), $surat);
249     }
250
251     $data->update([
252         'nama_instansi' => $request->nama_instansi,
253         'website_instansi' => $request->website_instansi,
254         'alamat_instansi' => $request->alamat_instansi,
255         'jobdesk' => $request->jobdesk,
256         'deskripsi_instansi' => $request->deskripsi_instansi,
257         'surat' => $surat,
258         'id_kelompok' => $request->id_kelompok,
259     ]);
260     $data->save();
261     return response()->json(['message' => 'Berhasil diubah !']);
262 }
```

Gambar 5.59 Kode program edit usulan instansi

Dalam Gambar 5.59 diatas merupakan potongan kode program untuk mengubah usulan instansi. Potongan kode program diatas dapat dijelaskan sebagai berikut :

- Baris 217 sampai 239 merupakan kode program untuk validasi data.
- Baris 235 merupakan kode program untuk mengambil data usulan instansi yang akan diubah.

- iii. Baris 242 sampai 260 merupakan kode program untuk menyimpan perubahan berkas.
- iv. Baris 261 merupakan kode program untuk menampilkan pesan “Berhasil diubah”.

17. Halaman daftar lowongan

Halaman daftar lowongan merupakan halaman untuk menampilkan daftar lowongan yang tersedia. Implementasi halaman daftar lowongan dalam Gambar 5.60.

Gambar 5. 60 Implementasi halaman daftar lowongan

```

26 public function indexmahasiswa()
27 {
28     $userId = Auth::id();
29     $idMahasiswa = Mahasiswa::select('id_mahasiswa', 'id_periode')
30             | ->where('id_users', $userId)->first();
31
32     $lowongan = Lowongan::get();
33     if(request()->ajax()){
34
35         $data = Lowongan::leftJoin('instansi', 'lowongan.id_instansi', 'instansi.id_instansi')
36             | ->select('lowongan.*', 'instansi.nama')
37             | ->where('lowongan.id_periode', $idMahasiswa->id_periode)
38             | ->get();
39         return datatables()->of($data)->addIndexColumn()
40
41             ->addColumn('action', function ($lowongan){
42
43                 $btn = '<a href="/mahasiswa/applylowongan/'.$lowongan->id_lowongan.'" class="btn btn-warning
44                 btn-sm"><i class="fas fa-arrow-right">/<i></a>';
45                 return $btn;
46             })
47
48             ->rawColumns(['action'])
49             ->addIndexColumn()
50             ->make(true);
51
52     }
53     return view('mahasiswa.lowongan.indexlowongan', compact('lowongan'));
54 }
```

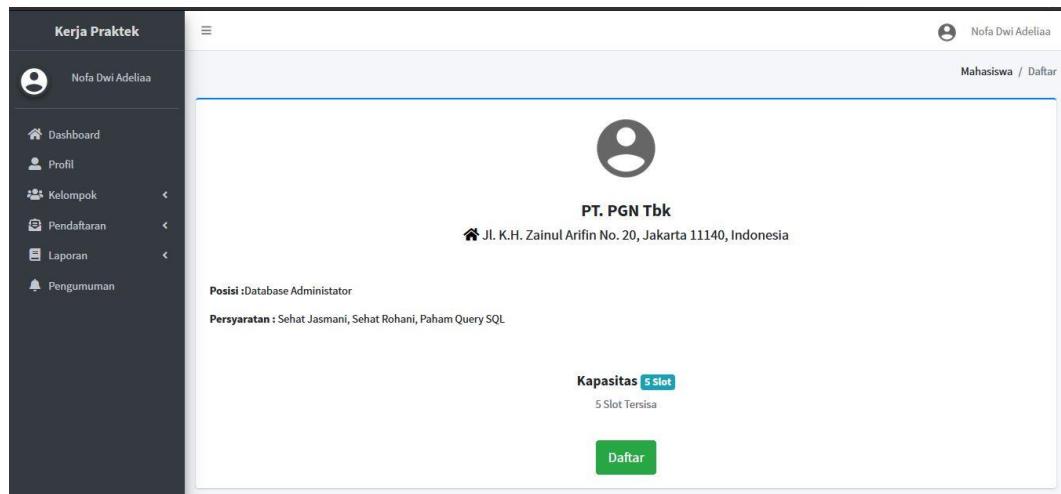
Gambar 5. 61 Kode program halaman daftar lowongan

Dalam gambar 5.61 diatas merupakan potongan kode program daftar lowongan . Potongan kode program diatas dapat dijelaskan sebagai berikut:

- i. Baris 28 sampai 30 merupakan kode program untuk mengambil id mahasiswa dan id periode sesuai dengan mahasiswa yang login.
- ii. Baris 32 merupakan kode program untuk mengambil data lowongan.
- iii. Baris 35 sampai 38 merupakan kode program untuk mengambil data lowongan yang sesuai dengan periode mahasiswa yang login.
- iv. Baris 41 sampai 45 merupakan kode program untuk menambahkan button dalam kolom aksi.
- v. Baris 51 merupakan kode program untuk menampilkan halaman daftar lowongan.

18. Halaman *apply* lowongan

Halaman *apply* lowongan adalah halaman yang menampilkan detail lowongan yang dipilih. Implementasi halaman apply lowongan dalam Gambar 5.62.



Gambar 5. 62 Implementasi halaman *apply* lowongan

```

109     public function show($id_lowongan)
110     {
111         // Get id user from Auth
112         $userId = Auth::id();
113
114         // Mengambil id_mahasiswa yang sesuai dengan id yang login sekarang
115         $idMahasiswa = Mahasiswa::select('id_mahasiswa')
116             ->where('id_users', $userId)->first();
117
118         $idKelompok = DetailKelompok::select('kelompok_detail.id_kelompok')
119             ->where('kelompok_detail.id_mahasiswa', $idMahasiswa)
120             ->where('kelompok_detail.isDeleted', '=', '0')
121             ->orderBy('kelompok_detail.id_kelompok_detail','desc')->first();
122
123         $statusKeanggotaan = @DetailKelompok::select('kelompok_detail.status_keanggotaan')
124             ->where('kelompok_detail.id_mahasiswa', $idMahasiswa)
125             ->orderBy('id_kelompok', 'desc')->first();
126
127         $statusUsulan = @Usulan::leftJoin('kelompok', 'usulan.id_kelompok', '=', 'kelompok.id_kelompok')
128             ->select('usulan.status')
129             ->where('usulan.id_kelompok', $idKelompok->id_kelompok)->first();
130
131         $statusLamaran = @Lamaran::select('pelamar.status')->where('pelamar.id_kelompok', $idKelompok->id_kelompok)->first();
132         $instansi = Instansi::get();
133         $datas = Periode::get();
134         $lowongan = Lowongan::findOrFail($id_lowongan);
135         return view('mahasiswa.lowongan.applylowongan',compact('instansi','lowongan','datas','idKelompok','data','userId',
136         'statusKeanggotaan','statusUsulan','statusLamaran'));
    }

```

Gambar 5. 63 Kode program halaman apply lowongan

Gambar 5.63 merupakan potongan kode program untuk menampilkan detail lowongan. Potongan kode program dapat dijelaskan sebagai berikut :

- i. Baris 112 sampai 121 merupakan kode program untuk mengambil id kelompok sesuai dengan mahasiswa yang login.
- ii. Baris 123 sampai 125 merupakan kode program untuk mengambil status keanggotaan.
- iii. Baris 127 sampai 129 kode program untuk mengambil status usulan
- iv. Baris 131 kode program untuk mengambil status lamaran
- v. Baris 132 merupakan kode program untuk mengambil data dari instansi
- vi. Baris 133 merupakan kode program untuk mengambil data periode
- vii. Baris 134 merupakan kode program untuk mengambil data lowongan sesuai dengan id lowongan yang dipilih.
- viii. Baris 135 sampai 136 merupakan kode program untuk menampilkan halaman apply lowongan.

```

141     public function lamarLowongan(Request $request){
142
143         $data = Lamaran::create([
144             'id_lowongan' => $request->id_lowongan,
145             'id_kelompok'=> $request->id_kelompok,
146             'created_by'=> $request->created_by,
147         ]);
148         $data->save();
149         return response()->json(['message' => 'Berhasil mendaftar lowongan.']);
150     }

```

Gambar 5. 64 Kode program apply lowongan

19. Halaman riwayat *apply* lowongan

Halaman riwayat *apply* lowongan merupakan halaman yang menampilkan daftar lamaran kelompok yang pernah dilamar. Implementasi halaman riwayat *apply* lowongan dalam-Gambar-5.65

No	Nama Instansi	Posisi	Status
1	PT KAI	Front end	ditolak
2	Aino Indonesia	Backend	sedang diproses

Gambar 5. 65 Implementasi halaman riwayat apply lowongan

```

19     public function index(Request $request)
20     {
21         // Get id user from Auth
22         $userId = Auth::id();
23
24         $idMahasiswa = Mahasiswa::select('id_mahasiswa')
25             ->where('id_users', $userId)->first();
26
27         $idKelompok = DetailKelompok::select('kelompok_detail.id_kelompok')
28             ->where('kelompok_detail.id_mahasiswa', $idMahasiswa->id_mahasiswa)
29             ->where('kelompok_detail.isDeleted', '=', '0')
30             ->orderBy('id_kelompok', 'desc')
31             ->first();
32
33         $lamaran = Lamaran::get();
34         if ($request->ajax()) {
35
36             $data = [];
37             if ($idKelompok){
38                 $data = Lamaran::where('id_kelompok',$idKelompok)
39                     ->join('lowongan', 'lowongan.id_lowongan','pelamar.id_lowongan')
40                     ->join('instansi', 'lowongan.id_instansi', 'instansi.id_instansi')
41                     ->select('pelamar.status','instansi.nama','lowongan.pekerjaan')
42                     ->get();
43             }
44
45             return datatables()->of($data)->addIndexColumn()
46             ->addIndexColumn()
47             ->make(true);
48         }
49     }
50 }
```

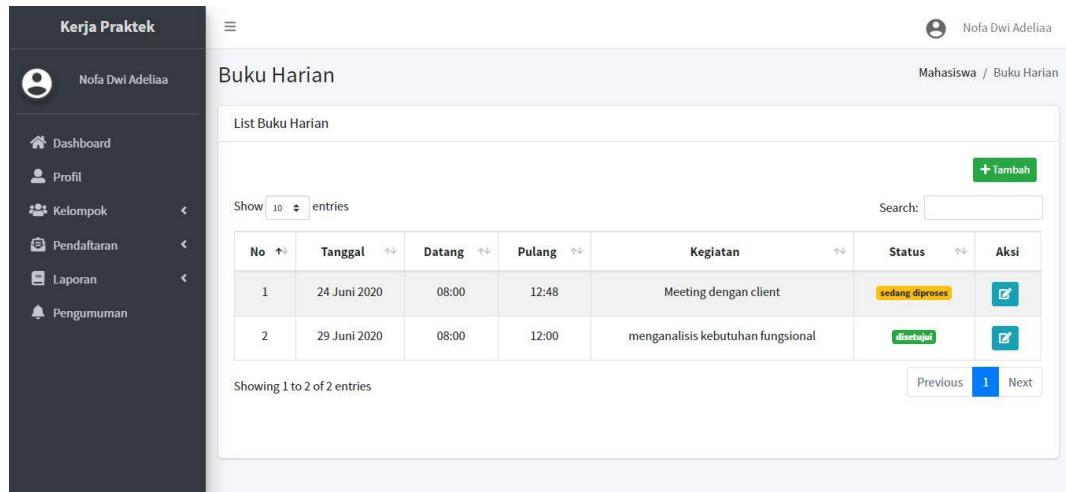
Gambar 5. 66 Kode program halaman riwayat apply lowongan

Gambar 5.66 merupakan potongan kode program untuk menampilkan halaman riwayat apply lowongan. Potongan kode program diatas memiliki penjelasan sebagai berikut :

- i. Baris 22 sampai 23 merupakan kode program untuk mengambil id kelompok sesuai dengan mahasiswa yang login.
- ii. Baris 33 untuk mengambil data lamaran.
- iii. Baris 34 sampai 48 merupakan kode program untuk menampilkan data lowongan yang pernah dilamar oleh kelompok ke dalam tabel.
- iv. Baris 49 kode program untuk menampilkan halaman lamaran.

20. Halaman daftar laporan harian

Halaman daftar laporan harian merupakan halaman yang menampilkan data laporan harian yang telah ditambah oleh mahasiswa.



Gambar 5. 67 Implementasi halaman detail laporan harian

```

21 | public function index(Request $request)
22 | {
23 |     $laporanharian = LaporanHarian::get();
24 |
25 |     // Get id user from Auth
26 |     $userId = Auth::id();
27 |
28 |     // Mengambil id_mahasiswa yang sesuai dengan id yang login sekarang
29 |     $idMahasiswa = Mahasiswa::select('mahasiswa.id_mahasiswa')
30 |                             ->where('mahasiswa.id_users', $userId)->first();
31 |
32 |     $periode = Periode::select('id_periode')->where('status', 'open')->first();
33 |
34 |     if($request()->ajax()){
35 |         $data = LaporanHarian::join('mahasiswa', 'buku_harian.id_mahasiswa', '=', 'mahasiswa.id_mahasiswa')
36 |             ->select('buku_harian.*')
37 |             ->where('buku_harian.id_mahasiswa', $idMahasiswa->id_mahasiswa)->get();
38 |
39 |         return datatables()->of($data)->addIndexColumn()
40 |
41 |             ->editColumn('tanggal', function ($laporanharian) {
42 |                 return Carbon::parse($laporanharian->tanggal)->translatedFormat('d F Y');
43 |             })
44 |             ->editColumn('waktu_mulai', function ($laporanharian) {
45 |                 return date('H:i', strtotime($laporanharian->waktu_mulai));
46 |             })
47 |
48 |             ->editColumn('waktu_selesai', function ($laporanharian) {
49 |                 return date('H:i', strtotime($laporanharian->waktu_selesai));
50 |             })
51 |     }
52 |
53 |     ->addColumn('action', function ($laporanharian){
54 |         $disable = $laporanharian->status == 'diperiksa'? "disabled" : " ";
55 |         $btn = '<a href="/mahasiswa/editlaporanharian/'.$laporanharian->id_buku_harian.'" class="btn btn-info btn-sm '.
56 |             '$disable.' "><i class="fas fa-edit">/i></a>';
57 |         $btn .= '&ampnbsp&ampnbsp';
58 |         return $btn;
59 |     })
60 |     ->rawColumns(['action'])
61 |     ->addIndexColumn()
62 |     ->make(true);
63 |     return view('mahasiswa.laporan.laporanharian', compact('laporanharian', 'idMahasiswa','userId','periode'));
64 |

```

Gambar 5. 68 Kode program halaman daftar laporan harian

Gambar 5.68 merupakan potongan kode program untuk menampilkan halaman detail laporan harian. Potongan kode diatas dapat dijelaskan sebagai berikut :

- i. Baris 23 merupakan potongan kode program untuk mengambil data laporan harian.
- ii. Baris 26 potongan kode program untuk mengambil data user id sesuai dengan user yang login.
- iii. Baris 29 merupakan potongan kode program untuk mengambil id mahasiswa sesuai user id yang login.
- iv. Baris 32 merupakan potongan kode program untuk mengambil id periode yang status periodenya open.
- v. Baris 35 sampai 37 merupakan potongan kode program untuk mengambil data laporan harian yang sesuai dengan laporan harian id mahasiswa.
- vi. Baris 39 merupakan potongan kode program untuk menampilkan data ke tabel.
- vii. Baris 41 sampai 42 merupakan potongan kode program untuk mengedit format dalam kolom tanggal. Format tanggal diubah ke dalam nama bulan yang dikonversi dari bahasa inggris ke bahasa indonesia.
- viii. Baris 44 sampai 45 merupakan potongan kode program untuk mengedit format dalam kolom waktu mulai. Format waktu di tampilkan diubah menjadi jam dan menit.
- ix. Baris 48 sampai 49 merupakan potongan kode program untuk mengedit format dalam kolom waktu selesai. Format waktu di tampilkan diubah menjadi jam dan menit.
- x. Baris 53- 57 merupakan potongan kode program untuk menambahkan button edit dalam kolom action. Button edit akan menjadi disable jika status laporan harian berubah menjadi ‘diperiksa’.

- xi. Baris 63 merupakan potongan kode untuk menampilkan data kedalam halaman laporan harian.

21. Halaman tambah laporan harian

Saat menambahkan laporan harian mahasiswa harus menginputkan tanggal, waktu datang, waktu selesai, dan kegiatan yang telah mahasiswa lakukan saat kerja praktik.

The screenshot shows a modal window titled "Tambah List Kegiatan Harian". It has four input fields: "Tanggal" (29/06/2020), "Waktu Mulai" (00:00), "Waktu Selesai" (02:55), and "Kegiatan" (empty). At the bottom are "Cancel" and "Save" buttons.

Gambar 5. 69 Implementasi halaman tambah laporan harian

```

78     public function create()
79     {
80
81         $data = LaporanHarian::all();
82         return view('mahasiswa.laporan.laporanharian',compact('data'));
83     }

```

Gambar 5. 70 Kode program halaman tambah laporan harian

Gambar 5.70 diatas merupakan potongan program untuk menampilkan halaman tambah laporan harian. Potongan kode diatas dapat dijelaskan sebagai berikut :

- Baris 81 merupakan kode program untuk mengambil data laporan harian.
- Baris 82 merupakan kode program untuk menampilkan data laporan harian kedalam halaman tambah laporan harian.

```

93     public function store(Request $request)
94     {
95         $this->validate($request, [
96             'tanggal' => 'required',
97             'waktu_mulai' => 'required',
98             'waktu_selesai' => 'required',
99             'kegiatan' => 'required',
100        ], [
101            'tanggal.required' => 'Tanggal tidak boleh kosong !',
102            'waktu_mulai.required' => 'Waktu mulai tidak boleh kosong !',
103            'waktu_selesai.required' => 'Waktu selesai tidak boleh kosong !',
104            'kegiatan.required' => 'Kegiatan tidak boleh kosong !'
105        ]);
106
107        $data = LaporanHarian::create([
108            'tanggal' => $request->tanggal,
109            'waktu_mulai' => $request->waktu_mulai,
110            'waktu_selesai' => $request->waktu_selesai,
111            'kegiatan' => $request->kegiatan,
112            'id_mahasiswa' => $request->id_mahasiswa,
113            'id_periode' => $request->id_periode,
114            'created_by' => $request->created_by,
115        ]);
116        $data->save();
117
118        return response()->json(['message' => 'Berhasil ditambahkan !']);
119
120

```

Gambar 5. 71 Kode program tambah laporan harian

Gambar 5.71 merupakan kode program untuk menambah laporan harian. Potongan kode program diatas dapat dijelaskan sebagai berikut :

- i. Baris 86 sampai 91 merupakan kode program untuk validasi data yang akan di inputkan.
- ii. Baris 93 sampai 101 merupakan kode program untuk menyimpan inputan data ke dalam kolom tabel laporan harian.
- iii. Baris 102 merupakan kode program untuk menyimpan data .
- iv. Baris 104 merupakan kode program untuk menampilkan pesan “Berhasil ditambahkan !” saat data disimpan.

22. Halaman edit laporan harian

Halaman edit laporan halaman yang digunakan mengubah isi laporan harian yang dipilih. Implementasi dari halaman edit laporan harian dalam Gambar 5.72.

The screenshot shows a user interface for editing a daily report. The main title is 'Edit Laporan Harian'. It includes fields for 'Tanggal' (24/06/2020), 'Waktu Mulai' (08:00), 'Waktu Selesai' (12:48), and 'Kegiatan' (Meeting dengan client). There are 'Cancel' and 'Save' buttons at the bottom. The sidebar on the left lists 'Dashboard', 'Profil', 'Kelompok', 'Pendaftaran', 'Laporan' (selected), and 'Pengumuman'.

Gambar 5.72 Implementasi halaman edit laporan harian

```

124 |     public function edit($id_buku_harian)
125 |     {
126 |
127 |         $data = LaporanHarian::findOrFail($id_buku_harian);
128 |         return view('mahasiswa.laporan.editlaporanharian', compact('data'));
129 |

```

Gambar 5.73 Kode program halaman edit laporan

Gambar 5.73 diatas adalah potongan kode program untuk menampilkan data kedalam halaman edit laporan. Potongan kode program diatas dapat dijelaskan sebagai berikut :

- i. Baris 127 merupakan kode program untuk mengambil data laporan harian sesuai dengan id laporan harian yang telah dipilih untuk diubah.
- ii. Baris 128 merupakan kode program untuk menampilkan data kedalam halaman edit laporan harian.

```

152     public function update(Request $request, $id_buku_harian)
153     {
154         $this->validate($request, [
155             'tanggal' => 'required',
156             'waktu_mulai' => 'required',
157             'waktu_selesai' => 'required',
158             'kegiatan' => 'required',
159         ],
160         [
161             'tanggal.required' => 'Tanggal tidak boleh kosong !',
162             'waktu_mulai.required' => 'Waktu mulai tidak boleh kosong !',
163             'waktu_selesai.required' => 'Waktu selesai tidak boleh kosong !',
164             'kegiatan.required' => 'Kegiatan tidak boleh kosong !'
165         ]);
166         $data = LaporanHarian::findOrFail($id_buku_harian);
167         $data -> update([
168             'tanggal' => $request->tanggal,
169             'waktu_mulai' => $request->waktu_mulai,
170             'waktu_selesai' => $request->waktu_selesai,
171             'kegiatan' => $request->kegiatan,
172             'id_mahasiswa' => $request->id_mahasiswa,
173         ]);
174         $data->save();
175         return response()->json(['message' => 'Berhasil diubah !']);
176     }

```

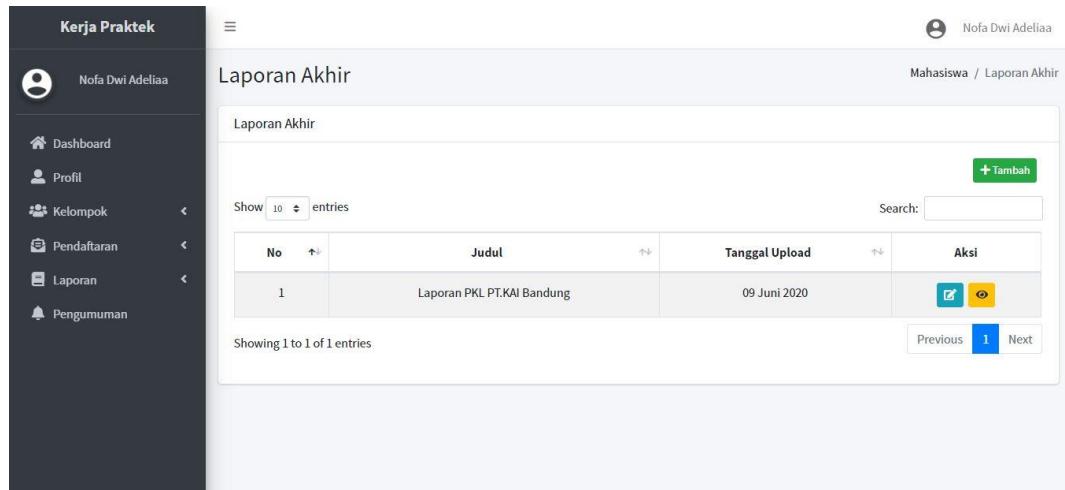
Gambar 5. 74 Kode program edit laporan harian

Gambar 5.74 diatas merupakan potongan kode program untuk mengubah data laporan harian. Potongan kode program diatas dapat dijelaskan sebagai berikut :

- i. Baris 138 sampai 144 merupakan kode program untuk validasi data laporan harian.
- ii. Baris 146 sampai 153 merupakan kode program untuk menyimpan data perubahan.
- iii. Baris 154 merupakan kode program untuk menampilkan pesan “Berhasil diubah !”.

23. Halaman daftar laporan akhir

Halaman daftar laporan akhir merupakan halaman untuk menampilkan data laporan akhir yang telah di tambahkan. Implementasi halaman daftar laporan akhir dalam Gambar 5.75.



Gambar 5.75 Implementasi halaman detail laporan akhir

```

22 public function index(Request $request)
23 {
24     $userId = Auth::id();
25     $idMahasiswa = Mahasiswa::select('id_mahasiswa')
26         ->where('id_users', $userId)->first();
27
28     $idKelompok = DetailKelompok::select('kelompok_detail.id_kelompok')
29         ->where('kelompok_detail.id_mahasiswa', $idMahasiswa)->first();
30
31     $status = DetailKelompok::select('kelompok_detail.status_keanggotaan')
32         ->where('kelompok_detail.id_mahasiswa', $idMahasiswa)->first();
33     $laporanakhir = LaporanAkhir::get();
34     if($request->ajax()){
35         $data = LaporanAkhir::leftJoin('kelompok', 'laporan.id_kelompok','=',
36             'kelompok.id_kelompok')
37             ->select('laporan.*')
38             ->where('laporan.id_kelompok', $idKelompok->id_kelompok)->get();
39             return datatables()->of($data)->addIndexColumn()
40             ->editColumn('created_at', function ($laporanakhir) {
41                 return Carbon::parse($laporanakhir->created_at)->translatedFormat('d F Y');
42             })
43             ->addColumn('action', function ($laporanakhir){
44                 $btn = '<a href="/mahasiswa/editlaporanpk1/'.$laporanakhir->id_laporan.'" class="btn btn-info
45                 btn-sm"><i class="fas fa-edit">/<i></a>';
46                 $btn .= '&ampnbsp&ampnbsp';
47                 $btn .= '<button type="button" name="show" id="'.$laporanakhir->id_laporan.'" class="btn btn-warning
48                 btn-sm detaillaporanakhir" ><i class="fas fa-eye">/<i></button>';
49                 return $btn;
50             })
51             ->rawColumns(['action'])
52             ->addIndexColumn()
53             ->make(true);
54     }
55
56     return view('mahasiswa.laporan.laporanpk1', compact('laporanakhir', 'status'));
57 }
```

Gambar 5.76 Kode program halaman daftar laporan akhir

Gambar 5.76 diatas adalah potongan kode program untuk menampilkan halaman daftar laporan akhir. Potongan kode program dapat dijelaskan sebagai berikut :

- i. Baris 24 sampai 26 merupakan kode program untuk mengambil id mahasiswa sesuai dengan user yang login.

- ii. Baris 28 sampai 29 kode program untuk mengambil id kelompok sesuai dengan id kelompok mahasiswa yang login.
- iii. Baris 31 sampai 32 kode program untuk mengambil data status keanggotaan sesuai dengan is mahasiswa yang login.
- iv. Baris 33 merupakan kode program untuk mengambil data laporan akhir.
- v. Baris 35 sampai 37 merupakan kode program untuk mengambil data laporan akhir berdasarkan id kelompok mahasiswa yang login.
- vi. Baris 38 merupakan kode program untuk menampilkan data kedalam tabel.
- vii. Baris 39 sampai 41 merupakan kode program untuk mengubah format tanggal dalam kolom tanggal upload.
- viii. Baris 42 sampai 47 merupakan kode program untuk menambah button edit dalam kolom aksi.
- ix. Baris 52 kode program untuk menampilkan data ke dalam halaman laporan akhir.

24. Halaman tambah laporan akhir

Halaman tambah laporan akhir merupakan halaman yang menampilkan form untuk menambahkan laporan akhir. Implementasi dalam Gambar 5.77

Gambar 5.77 Implementasi halaman tambah laporan akhir

```

93     public function create()
94     {
95         $userId = Auth::id();
96         // Mengambil id_mahasiswa yang sesuai dengan id yang login sekarang
97         $idMahasiswa = Mahasiswa::select('id_mahasiswa')
98             ->where('id_users', $userId)->first();
99         // mengambil id kelompok
100        $idKelompok = DetailKelompok::select('kelompok_detail.id_kelompok')
101            ->where('kelompok_detail.id_mahasiswa', $idMahasiswa->id_mahasiswa)
102            ->orderBy('kelompok_detail.id_kelompok_detail','desc')->first();
103
104        return view('mahasiswa.laporan.tambahlaporanpk1',compact('datas', 'idKelompok','userId'));
105    }

```

Gambar 5.78 Kode program halaman tambah laporan akhir

Gambar 5.78 diatas adalah potongan kode program untuk menampilkan halaman tambah laporan akhir. Potongan kode program diatas dapat dijelaskan sebagai berikut :

- Baris 95 sampai 102 merupakan kode program untuk mengambil id kelompok yang sesuai dengan mahasiswa yang login.
- Baris 104 merupakan kode program untuk menampilkan halaman tambah laporan akhir.

```

130    public function store(Request $request)
131    {
132        $this->validate($request, [
133            'judul' => 'required|string|max:100',
134            'berkas' => 'required|mimes:pdf|max:10240',
135        ],
136        [
137            'judul.required' => 'Judul tidak boleh kosong !',
138            'judul.max' => 'Judul terlalu panjang !',
139            'berkas.max' => 'File terlalu besar !',
140            'berkas.mimes' => 'File harus pdf',
141            'berkas.required' => 'Berkas tidak boleh kosong !'
142        ]);
143
144        $berkas= null;
145        if($request->hasFile('berkas')){
146            $files=$request->file('berkas');
147            $berkas=str_slug('Laporan Kelompok-'.$request->id_kelompok) . '.' . $files->getClientOriginalExtension();
148            $files->move(public_path('uploads/laporanakhir'),$berkas);
149        }
150
151        $data = LaporanAkhir::create([
152            'judul' => $request->judul,
153            'berkas' => $berkas,
154            'id_kelompok' => $request->id_kelompok,
155            'created_by' => $request->created_by,
156        ]);
157
158        $data->save();
159        return response()->json(['message' => 'Laporan Akhir berhasil ditambahkan.']);
160    }

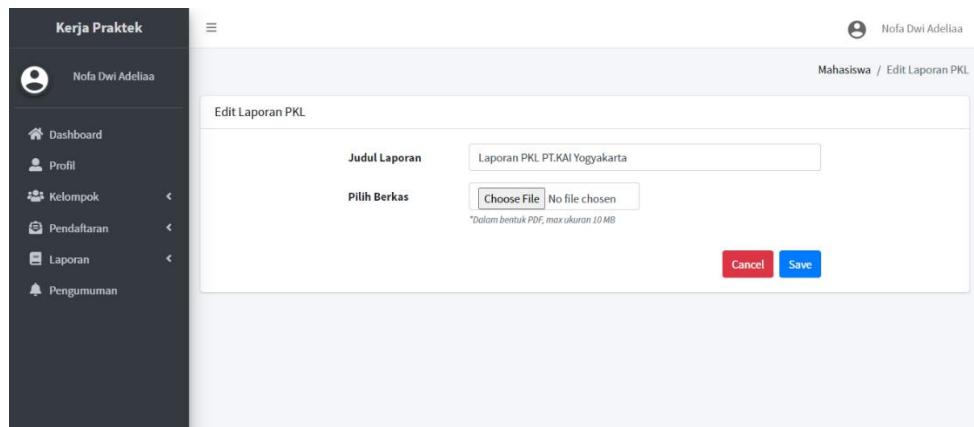
```

Gambar 5.79 Kode program tambah laporan akhir

Gambar 5.79 merupakan potongan kode program untuk menambahkan laporan akhir. Potongan kode program dapat dijelaskan sebagai berikut :

- Baris 132 sampai 143 merupakan kode program untuk validasi masukan data.

- ii. Baris 145 sampai 159 merupakan kode program untuk menyimpan file berkas laporan akhir kedalam folder.
- iii. Baris 160 kode program untuk menampilkan pesan berhasil di tambah.



Gambar 5. 80 Implementasi halaman edit laporan akhir

```

164     public function edit($id_laporan)
165     {
166         $userId = Auth::id();
167
168         // Mengambil id_mahasiswa yang sesuai dengan id yang login sekarang
169         $idMahasiswa = Mahasiswa::select('id_mahasiswa')
170             | | | | ->where('id_users', $userId)->first();
171
172         $idKelompok = DetailKelompok::select('kelompok_detail.id_kelompok')
173             | | | | ->where('kelompok_detail.id_mahasiswa', $idMahasiswa->id_mahasiswa)
174             | | | | ->orderBy('kelompok_detail.id_kelompok_detail','desc')->first();
175         $datas = Kelompok::get();
176         $data = LaporanAkhir::findOrFail($id_laporan);
177
178         return view('mahasiswa.laporan.editlaporanpkl', compact('data','idKelompok','userId'));
179     }

```

Gambar 5. 81 Kode program edit laporan akhir

Gambar 5.81 diatas merupakan potongan kode program untuk menampilkan halaman edit laporan akhir. Potongan kode program tersebut dapat dijelaskan sebagai berikut :

- i. Baris 166 sampai 174 merupakan kode program untuk mengambil id kelompok berdasarkan kelompok mahasiswa yang login.
- ii. Baris 175 merupakan kode program untuk mengambil data laporan akhir.
- iii. Baris 178 merupakan kode program untuk menampilkan halaman edit laporan akhir.

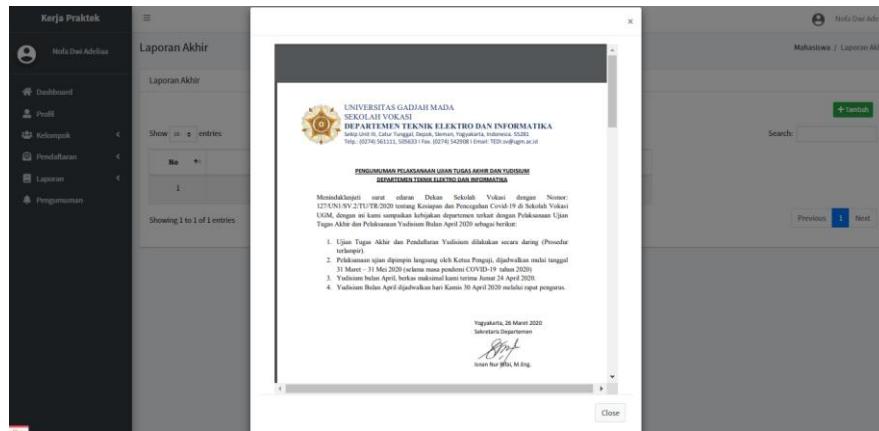
```

204     public function update(Request $request, $id_laporan)
205     {
206         $this->validate($request, [
207             'judul' => 'required|string|max:100',
208             'berkas' => 'mimes:pdf|max:10000',
209         ],
210         [
211             'judul.required' => 'Judul tidak boleh kosong !',
212             'judul.max' => 'Judul terlalu panjang !',
213             'berkas.max' => 'File terlalu besar!',
214             'berkas.mimes' => 'File harus pdf'
215         ]);
216     }
217
218
219     $data = LaporanAkhir::findOrFail($id_laporan);
220     $berkas = $data->berkas;
221
222     if ($request->hasFile('berkas')) {
223         !empty($berkas) ? File::delete(public_path('uploads/laporanakhir' . $berkas)) : null;
224         $files=$request->file('berkas');
225         $berkas=str_slug('Laporan Kelompok-' . $request->id_kelompok) . '.' . $files->getClientOriginalExtension();
226         $files->move(public_path('uploads/laporanakhir'),$berkas);
227     }
228     $data -> update([
229         'judul' => $request->judul,
230         'berkas' => $berkas,
231         'id_kelompok' => $request->id_kelompok,
232     ]);
233     $data->save();
234     return response()->json(['message' => 'Berhasil diubah !']);
235 }
```

Gambar 5.82 Kode program ubah laporan akhir

Gambar 5.82 merupakan potongan kode program untuk mengubah laporan akhir. Potongan kode program tersebut dapat dijelaskan sebagai berikut :

- i. Baris 207 sampai 216 merupakan kode program untuk validasi data.
- ii. Baris 220 sampai 233 merupakan kode program untuk menyimpan data yang diubah.
- iii. Baris 234 merupakan kode program untuk menampilkan pesan berhasil diubah.



Gambar 5. 83 Pop up menampilkan laporan akhir

Gambar 5.83 merupakan tampilan pop up untuk menampilkan laporan akhir.

```

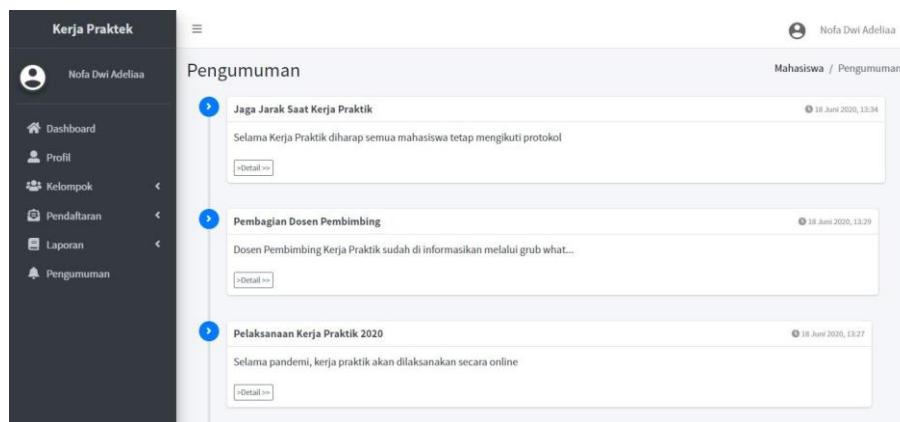
114     public function show(Request $request)
115 {
116     $data = LaporanAkhir::where('id_laporan', $request->id_laporan)->first();
117     return response()->json($data);
118 }
```

Gambar 5. 84 Kode program pop up halaman laporan akhir

Gambar 5.84 merupakan potongan kode program untuk menampilkan laporan akhir. Potongan kode program dapat dijelaskan sebagai berikut :

- Baris 116 sampai 177 merupakan kode program untuk mengambil data laporan akhir yang akan ditampilkan.

25. Halaman pengumuman



Gambar 5. 85 Implementasi halaman pengumuman

```

38 |     // menampilkan pengumuman
39 |     public function indexmahasiswa()
40 |     {
41 |         $data = Pengumuman::orderBy('created_at','DESC')->get();
42 |         return view('mahasiswa.pengumuman',compact('data'));
43 |     }

```

Gambar 5. 86 Kode program menampilkan daftar pengumuman di halaman mahasiswa

Gambar 5.86 adalah kode program untuk menampilkan daftar pengumuman dalam halaman mahasiswa. Potongan kode diatas dapat dijelaskan sebagai berikut :

- i. Baris 41 merupakan kode program untuk mengambil seluruh data atau daftar pengumuman. Daftar pengumuman akan ditampilkan secara urut , urutan teratas adalah pengumuman yang paling terbaru.
- ii. Baris 42 merupakan kode program untuk menampilkan data ke dalam halaman pengumuman.

26. Halaman detail pengumuman



Gambar 5. 87 Implementasi halaman detail pengumuman

```

108 |     // menampilkan detail pengumuman
109 |     public function show(Request $request)
110 |     {
111 |         $pengumuman = Pengumuman::where('id_pengumuman', $request->id_pengumuman)->first();
112 |         return response()->json($pengumuman);
113 |
}

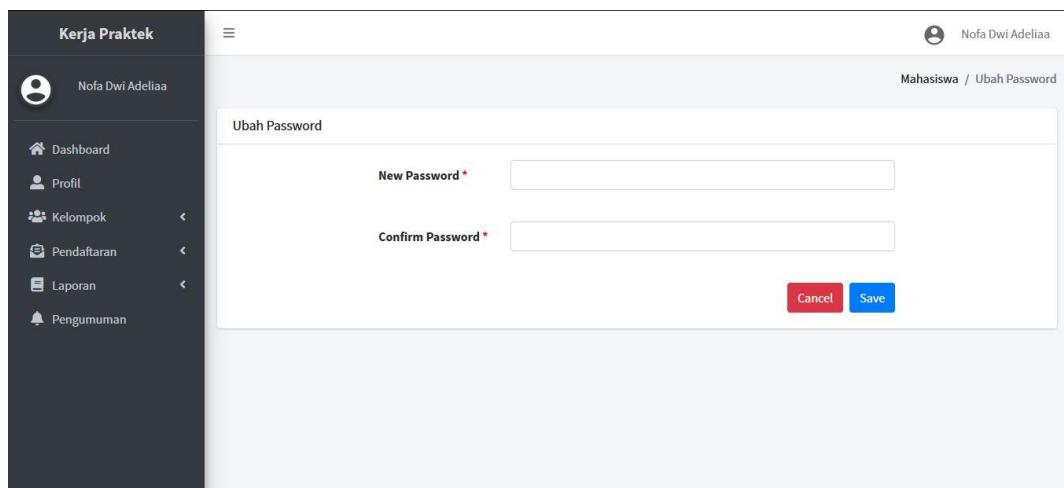
```

Gambar 5. 88 Kode program detail pengumuman

Gambar 5.88 adalah kode program untuk menampilkan detail pengumuman setelah tombol detail di pilih. Potongan kode diatas dapat dijelaskan sebagai berikut :

- i. Baris 111 merupakan kode program untuk mengambil data pengumuman yang telah dipilih berdasarkan id pengumuman.
- ii. Baris 112 merupakan kode program untuk menampilkan data pengumuman.

27. Halaman ubah password



Gambar 5. 89 Implementasi halaman ubah.password

```

218 |     public function showchangePassword(){
219 |         $user = Auth::user();
220 |         return view('mahasiswa.ubah_password', compact('user'));
221 |
}

```

Gambar 5. 90 Kode program untuk menampilkan halaman ubah password

Gambar 5.90 adalah potongan kode program untuk menampilkan halaman ubah password.

```
84     public function updatePassword(Request $request, $id_users){
85         $this->validate($request, [
86             'password' => 'required|min:6|max:191|confirmed'
87         ],
88         [
89             'password.required' => 'Password tidak boleh kosong !',
90             'password.min' => 'Password terlalu pendek !',
91             'password.max' => 'Password terlalu panjang !',
92             'password.confirmed' => 'Confirm password tidak sama !',
93         ]);
94         $data = User::where ('id_users',$id_users)->first();
95         $data->password = Hash::make($request->password);
96         $data->save();
97         return response()->json(['message'=>'Password updated successfully.']);
98     }
99 }
```

Gambar 5.91 Kode program untuk update password

Gambar 5.91 adalah potongan kode program untuk update password.

Penjelasan potongan kode program diatas sebagai berikut :

- i. Baris 85 sampai 94 merupakan kode program untuk validasi
- ii. Baris 96 merupakan kode program untuk mengambil data user sesuai id user yang login.
- iii. Baris 96 sampai 97 kode program untuk menyimpan perubahan password.
- iv. Baris 98 adalah kode program untuk menampilkan notifikasi password berhasil diubah.

BAB VI

HASIL PENGUJIAN DAN PEMBAHASAN

6.1 Skenario Pengujian

Skenario pengujian merupakan urutan dari proses yang digunakan untuk uji coba Sistem Kerja Praktek bagian Mahasiswa Program Studi Komputer dan Sistem Informasi SV UGM Berbasis Web. Skenario pengujian sistem terdapat dalam Tabel 6.1.

Tabel 6. 1 Skenario pengujian

No	Pengujian	Materi
1	<i>Login</i> sistem	Isi data masukan
2	Ubah foto profil	Pilih text ubah foto dan isi data masukan
3	Ubah data profil	Isi data masukan
4	Lihat CV	Pilih icon <i>view</i>
5	Buat kelompok	Isi data masukan
6	Tambah anggota	Pilih tombol tambah anggota dan pilih tombol <i>save</i>
7	Hapus undangan anggota	Pilih icon hapus
8	Penilaian anggota	Isi data masukan
9	Tambah usulan instansi	Pilih tombol tambah instansi dan isi data masukan
10	Lihat semua lowongan	Pilih menu lowongan
11	Lihat detail lowongan	Pilih icon panah
12	<i>Apply</i> lowongan	Pilih tombol tambah
13	Tambah laporan harian	Pilih tombol tambah dan isi data masukan
14	Tambah laporan akhir	Pilih tombol tambah dan isi data masukan

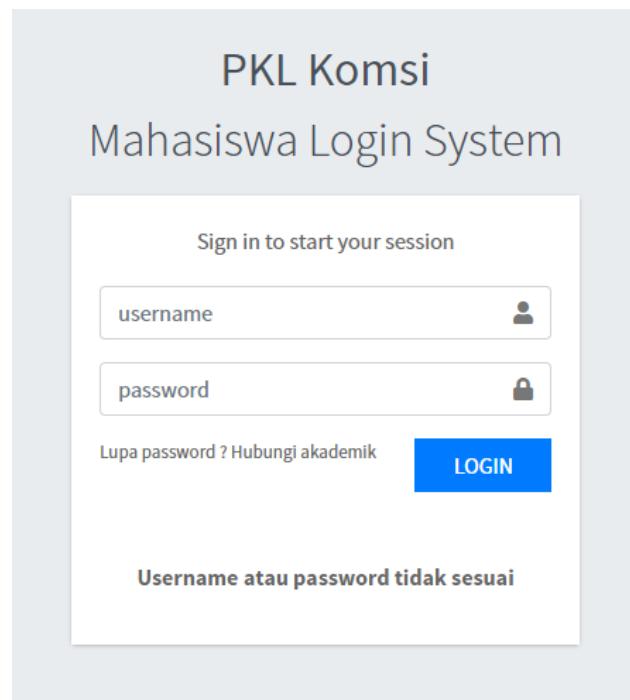
15	Lihat laporan akhir	Pilih icon <i>view</i>
16	Lihat semua pengumuman	Pilih menu pengumuman
17	Lihat detail pengumuman	Pilih tombol detail
18	Ubah password	Isi data masukan

6.2 Hasil Pengujian

Tahap ini merupakan bagian dari pengujian yang disusun dalam skenario pengujian yang dikhususkan untuk fungsional sistem dengan memperhatikan masukan dan keluaran data.

6.2.1 Pengujian *login* sistem

Pengujian *login* sistem bertujuan untuk menyeleksi user yang bisa memasuki sistem. User yang dapat memasuki sistem merupakan pengguna yang data masukannya terdaftar dalam database sistem. Pengguna yang berhasil memasuki sistem akan diarahkan ke halaman dashboard. Namun apabila data yang dimasukkan tidak sesuai akan muncul error seperti dalam Gambar 6.1.



Gambar 6. 1 Tampilan pesan error username atau password tidak sesuai

6.2.2 Pengujian Ubah Foto Profil

Pengujian ubah foto profil merupakan kegiatan untuk mengubah foto profil milik mahasiswa tersebut dengan memilih text “Ubah foto” . Untuk dapat mengubah foto profil, mahasiswa harus mengunggah foto. Apabila foto yang diunggah sesuai ketentuan muncul notifikasi seperti dalam Gambar 6.2



Gambar 6. 2 Notifikasi foto profil berhasil diubah

6.2.3 Pengujian Ubah Data Profil

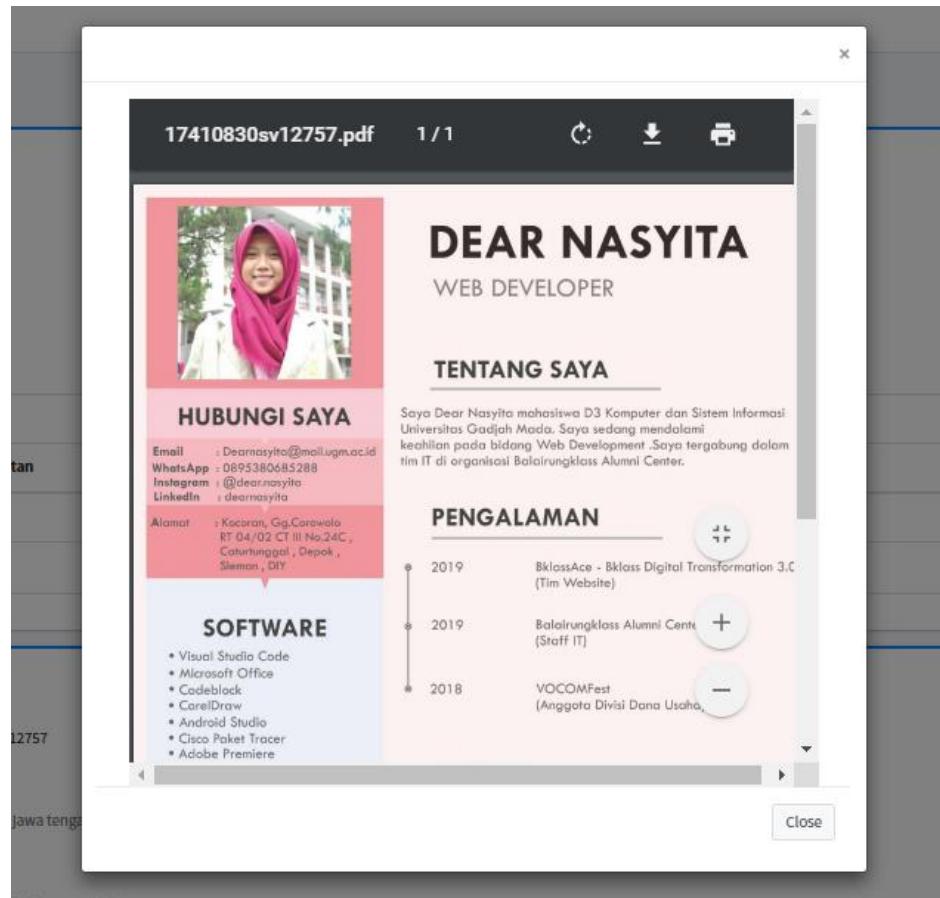
Pengujian ini merupakan kegiatan untuk memastikan fungsi ubah data profil dapat berjalan sesuai harapan dan data tersebut tersimpan dalam database. Pengujian ini mahasiswa melengkapi data masukan profil kemudian memilih tombol save. Apabila data yang dimasukan benar maka akan muncul notifikasi seperti dalam Gambar 6.3.



Gambar 6. 3 Notifikasi berhasil ubah data profil

6.2.4 Pengujian Lihat CV

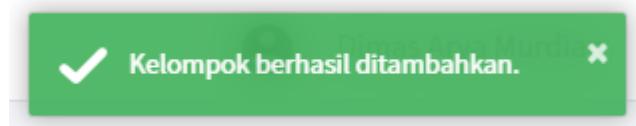
Pengujian lihat cv bertujuan untuk memastikan fungsi lihat cv dapat menampilkan berkas cv yang telah tersimpan. Pengujian ini dengan cara memilih icon view yang terdapat dalam halaman profil. Tampilan cv yang akan ditampilkan seperti Gambar 6.4.



Gambar 6. 4 Tampilan berhasil lihat CV

6.2.5 Pengujian Buat Kelompok

Pengujian buat kelompok bertujuan untuk memastikan fitur buat kelompok berjalan sesuai yang diharapkan dan data kelompok tersimpan dalam basis data. Dalam pengujian ini mahasiswa mengisi data masukan lalu memilih tombol create. Apabila berhasil tampil notifikasi seperti Gambar 6.5.

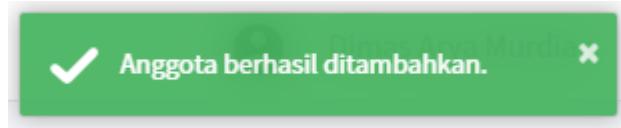


Gambar 6. 5 Notifikasi kelompok berhasil dibuat

6.2.6 Pengujian Tambah Anggota

Pengujian tambah anggota bertujuan untuk memastikan fungsi tambah anggota dapat berjalan sesuai yang diharapkan. Dalam pengujian ini mahasiswa memilih tombol “Tambah Anggota” lalu memilih anggota dengan memilih icon

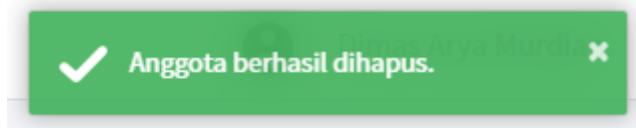
plus kemudian pilih tombol “save”. Apabila berhasil muncul notifikasi seperti dalam Gambar 6.6.



Gambar 6. 6 Anggota berhasil ditambahkan

6.2.7 Pengujian Hapus Undangan Anggota

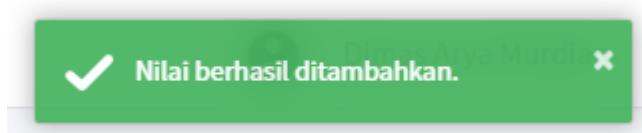
Pengujian hapus undangan anggota bertujuan untuk memastikan fitur hapus undangan anggota berjalan sesuai yang diharapkan. Pengujian ini dengan cara memilih icon hapus lalu memilih tombol “OK”. Apabila berhasil muncul notifikasi seperti Gambar 6.7



Gambar 6. 7 Notifikasi berhasil hapus undangan anggota

6.2.8 Pengujian Penilaian Anggota

Pengujian penilaian anggota bertujuan untuk memastikan fitur penilaian anggota berjalan dengan baik. Pengujian ini dengan mengisi data masukan. Apabila data masukan valid muncul notifikasi seperti Gambar 6.8.



Gambar 6. 8 Notifikasi nilai berhasil ditambahkan

Apabila data yang dimasukkan tidak valid muncul notifikasi seperti Gambar 6.9.

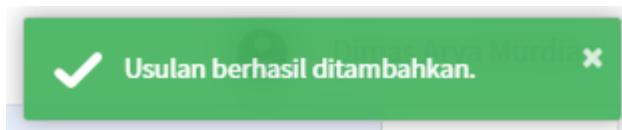
Penilaian Anggota Kelompok

	5 Sangat Baik	4 Baik	3 Cukup	2 Kurang	1 Sangat Kurang		
Nama Mahasiswa	Dimas Arya Murdian	Skill	Kerapihan	Sikap	Keaktifan	Perhatian	Kehadiran
	<input type="text" value="6"/>	<input type="button" value="▼"/>	<input type="text" value="4"/>	<input type="text" value="5"/>	<input type="text" value="5"/>	<input type="text" value="5"/>	<input type="text" value="3"/>
Keterangan	<p>! Value must be less than or equal to 5.</p> <ol style="list-style-type: none"> 1. Skill : Kemampuan dalam menyelesaikan tugas 2. Kerapihan : Berpakaian, cara kerja, penampilan 3. Sikap : Kesopanan, menghormati, menghargai orang lain 4. Keaktifan : Bertanya, mengeluarkan pendapat, tidak malas-malasan 5. Perhatian : Keingintahuan, kepatuhan dalam bimbingan 6. Kehadiran : Kehadiran PKL, efisien waktu 						
	<input style="background-color: red; color: white; border-radius: 5px; padding: 2px 10px; margin-right: 10px;" type="button" value="Cancel"/> <input style="background-color: teal; color: white; border-radius: 5px; padding: 2px 10px;" type="button" value="Save"/>						

Gambar 6. 9 Notifikasi masukan data tidak valid

6.2.9 Pengujian Tambah Usulan Instansi

Pengujian tambah usulan instansi bertujuan untuk memastikan fitur tambah usulan instansi berjalan sesuai dengan harapan. Pengujian ini dengan cara memilih tombol “Tambah Usulan” lalu mengisi data masukan kemudian memilih tombol *save*. Apabila berhasil menambahkan usulan instansi muncul notifikasi seperti Gambar 6.10.



Gambar 6. 10 Notifikasi berhasil menambahkan usulan

6.2.10 Pengujian Lihat Semua Lowongan

Pengujian lihat semua lowongan bertujuan untuk memastikan fitur lihat semua lowongan berfungsi dengan baik. Pengujian ini dengan cara memilih menu “Lowongan”. Apabila berhasil tampilan menjadi seperti Gambar 6.11.

No	Nama Instansi	Posisi	Kapasitas	Aksi
1	PT. PGN Tbk	UI UX Designer	3	
2	PT. PGN Tbk	Membuat Sistem Analisis Obat	3	

Gambar 6. 11 Tampilan lihat semua lowongan

6.2.11 Pengujian Lihat Detail Lowongan

Pengujian ini bertujuan untuk memastikan fitur detail lowongan berfungsi dengan baik. Pengujian ini dengan cara memilih icon panah dalam kolom aksi pada tabel daftar lowongan. Apabila berhasil tampilan detail lowongan seperti Gambar 6.12.

Gambar 6. 12 Tampilan lihat semua lowongan

6.2.12 Pengujian Apply Lowongan

Pengujian ini bertujuan untuk memastikan fitur *apply* lowongan berjalan dengan baik. Pengujian ini dilakukan dengan cara memilih tombol “Daftar” pada

halaman detail lowongan. Apabila berjalan dengan baik muncul notifikasi seperti Gambar 6.13.



Gambar 6. 13 Notifikasi berhasil *apply* lowongan

6.2.13 Pengujian Tambah Laporan Harian

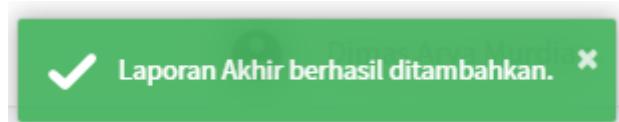
Pengujian tambah laporan harian bertujuan untuk memastikan fitur tambah laporan harian berjalan dengan baik. Cara pengujian ini memilih tombol “tambah” selanjutnya mengisi data masukan lalu memilih tombol save. Apabila berhasil muncul notifikasi seperti Gambar 6.14.



Gambar 6. 14 Notifikasi berhasil menambah laporan harian

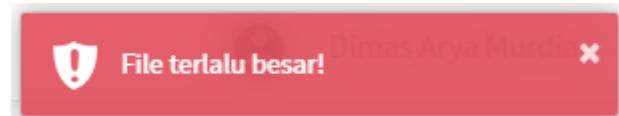
6.2.14 Pengujian Tambah Laporan Akhir

Pengujian tambah laporan akhir berfungsi untuk memastikan fitur tambah laporan akhir berjalan dengan baik. Pengujian ini dengan cara memilih tombol tambah kemudian mengisi judul laporan dan berkas laporan. Apabila berhasil muncul notifikasi seperti Gambar 6.15.



Gambar 6. 15 Notifikasi laporan akhir berhasil ditambahkan

Apabila masukan data tidak valid muncul notifikasi eror seperti Gambar 6.16.



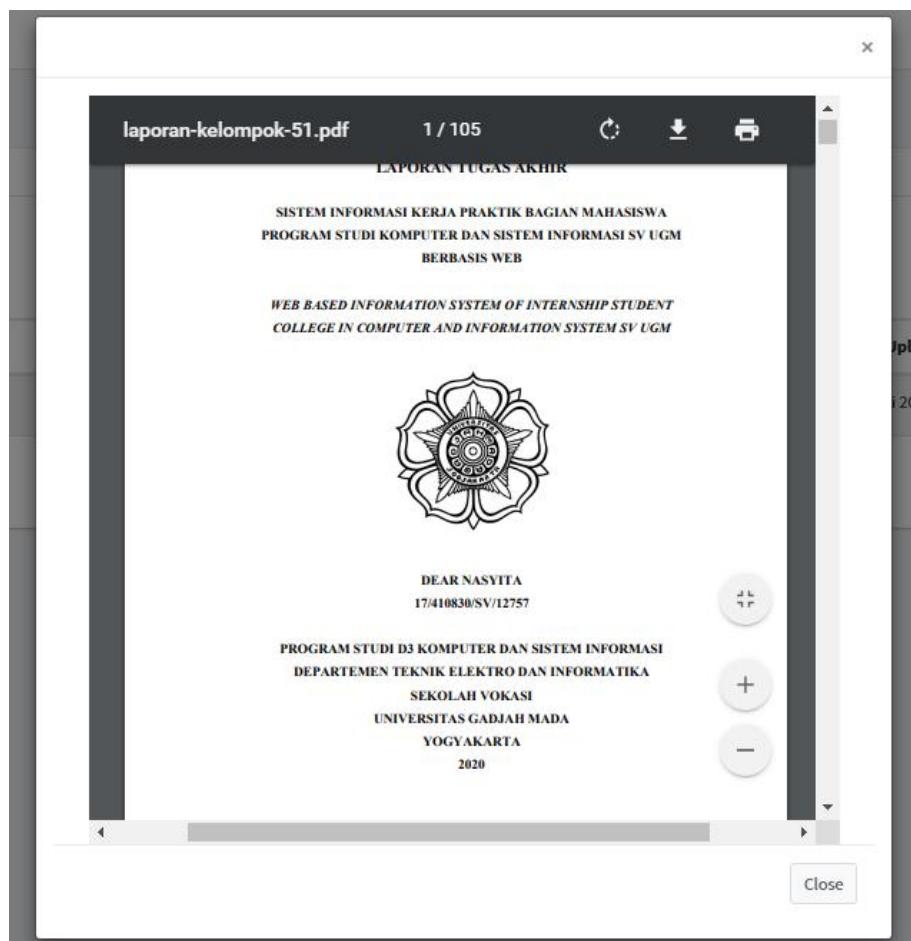
Gambar 6. 16 Notifikasi file laporan terlalu besar



Gambar 6. 17 Notifikasi format file laporan tidak sesuai ketentuan

6.2.15 Pengujian Lihat Laporan Akhir

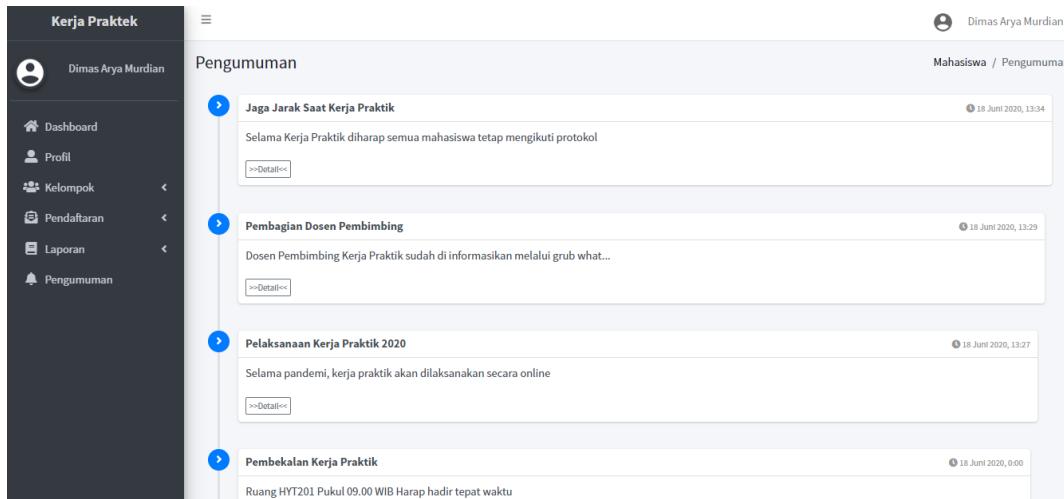
Pengujian ini berfungsi untuk memastikan fitur lihat laporan akhir berjalan dengan baik. Pengujian ini dilakukan dengan cara memilih icon view pada kolom aksi halaman laporan pkl. Apabila berhasil muncul pop up seperti Gambar 6.18.



Gambar 6. 18 Pop up laporan akhir

6.2.16 Pengujian Lihat Semua Pengumuman

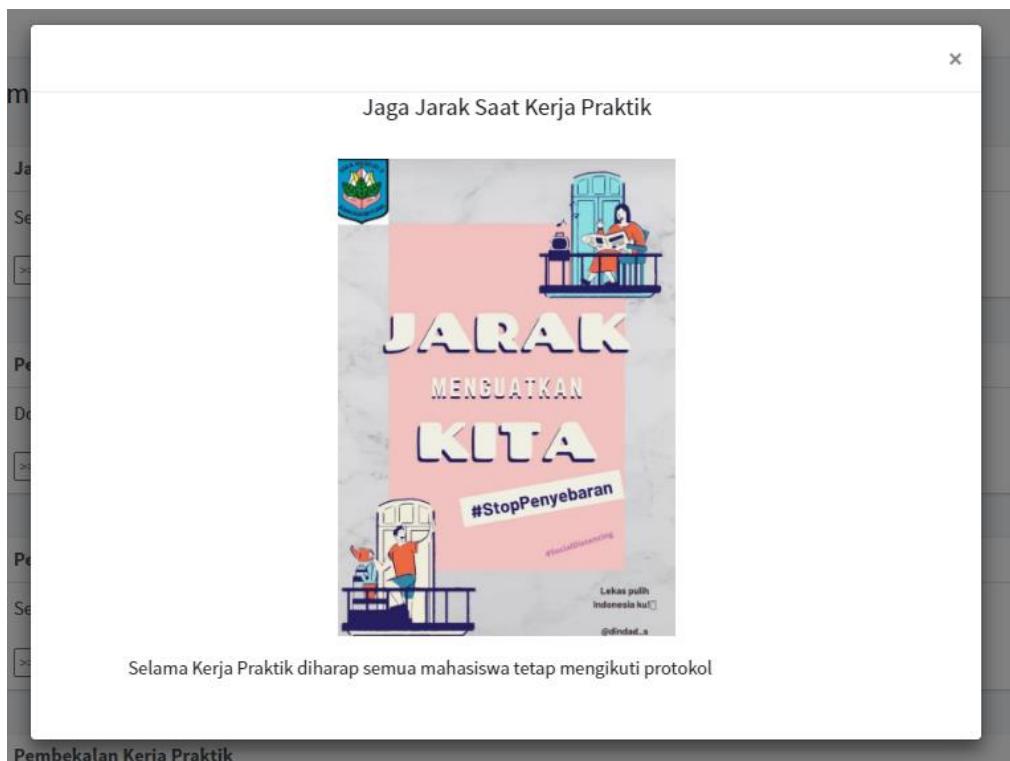
Pengujian ini bertujuan untuk memastikan fungsi lihat semua pengumuman berjalan dengan baik. Pengujian ini dilakukan dengan cara memilih menu pengumuman. Apabila berhasil akan menampilkan halaman seperti Gambar 6.19.



Gambar 6. 19 Halaman lihat semua pengumuman

6.2.17 Pengujian Lihat Detail Pengumuman

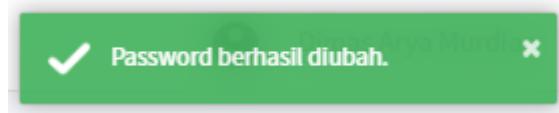
Pengujian ini bertujuan untuk memastikan fitur detail pengumuman berfungsi sesuai harapan. Pengujian ini dilakukan dengan cara memilih tombol Detail pada halaman pengumuman. Apabila berhasil, muncul pop up seperti Gambar 6.20.



Gambar 6. 20 Pop up detail pengumuman

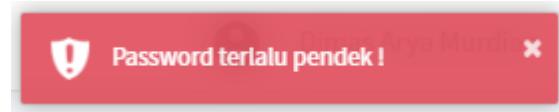
6.2.18 Pengujian Ubah Password

Pengujian ubah password berfungsi untuk memastikan fitur ubah password berfungsi sesuai dengan yang diharapkan. Pengujian ini dengan memasukkan password baru dan konfirmasi password baru. Apabila berhasil muncul notifikasi seperti dalam Gambar 6.21.

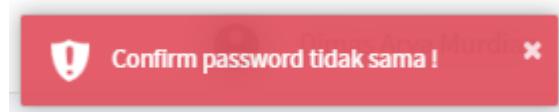


Gambar 6. 21 Notifikasi berhasil ubah password

Apabila data yang dimasukkan tidak valid muncul notifikasi error seperti Gambar 6.22.



Gambar 6. 22 Notifikasi error data masukan terlalu pendek



Gambar 6. 23. Notifikasi error masukan data confirm password tidak sama

BAB VII

KESIMPULAN DAN SARAN

7.1 Kesimpulan

Kesimpulan yang didapat dari hasil pengujian sistem ini sebagai berikut:

1. Sistem Informasi Kerja Praktek bagian Mahasiswa Program Studi Komputer dan Sistem Informasi telah berhasil dibangun sesuai dengan fungsionalitas yang ada.
2. Sistem dapat memudahkan mahasiswa dalam melakukan pendaftaran kerja praktek.
3. Sistem dapat memudahkan mahasiswa dalam pelaporan kegiatan harian dan penilaian anggota kelompok kerja praktek.
4. Secara keseluruhan sistem mampu melaksanakan proses-proses yang telah dirancang pada tahap analisis dan memenuhi kebutuhan sistem.

7.2 Saran

Saran yang peneliti berikan apabila terdapat penelitian dengan tema yang sama agar hasil lebih baik, antara lain :

1. Sistem dapat terhubung langsung dengan sistem akademik milik Universitas.
2. Fitur penilaian terhadap anggota kelompok dapat menggunakan sistem ranking.
3. Diperlukan pengembangan dalam hal *interface* agar lebih menarik.

DAFTAR PUSTAKA

- Andriyanto, T. & Aswi, R. R., 2016. Rancang Bangun Sistem Informasi Praktek Kerja Lapangan Terintegrasi Menggunakan Webservice. *Jurnal SIMETRIS*, Volume 7.
- Anggraeni, E. & Irviani, R., 2017. *Pengantar Sistem Informasi*. Yogyakarta: Andi Offset.
- Anon., t.thn. *Visual Studio Code*. [Online]
Available at: <https://code.visualstudio.com/>
[Diakses 14 July 2020].
- Astamal, R., 2006. *Mastering Kode HTML (edisi 2)*. 2 penyunt. Surabaya: s.n.
- Bassil, Y., 2012. *A Simulation Model For The Waterfall Software Development Life*. Beirut: International journal of Engineering & Technology.
- Bekti, H. B., 2015. *Mahir Membuat Website dengan Adobe Dreamweaver CS6, CSS dan JQuery*. Yogyakarta: Andi.
- Connolly, T. & Carolyn, E., 2002. *Database Systems : A Practical Approach to Design, Implementation, and Management*. Third Edition penyunt. Massachusetts: Addison-Wesley, Reading.
- Djahir, Y. & Pratita, D., 2015. *Bahan Ajar Sistem Informasi Manajemen*. Yogyakarta: Deepublish.
- Hakim & Rizki, M., 2012. *Prototype Sistem Informasi Akademik Berbasis Mobile Menggunakan Java Script Object Notation (JSON)*. Surabaya: STIKOM Surabaya.
- Hutahean, J., 2015. *Konsep Sistem Informasi*. Yogyakarta: Deepublish.
- Irwanto, D., 2004. *Perancangan Object Oriented Software dengan UML*. Yogyakarta: ANDI.
- Jogiyanto, H., 2005. *Analisis dan Desain*. Yogyakarta: ANDI.
- Kristanto, A., 2008. *Perancangan Sistem Informasi dan Aplikasinya*. Yogyakarta: Gava Media.
- Nixon, R., 2009. *Learning PHP, MySQL and Javascript*. s.l.:O'Reilly Media, Inc

- Nugroho, A., 2011. *Perancangan dan Implementasi Sistem Basis Data*. Yogyakarta: ANDI.
- Ramadhani, G., 2003. *Modul Pengenalan Internet*. Yogyakarta: s.n.
- Riyanto, 2010. *Sistem Informasi Penjualan Dengan PHP Dan MySQL*. Yogyakarta: Gava Media.
- Rohman, A., 2017. *Documentation & Testing API dengan Postman part 1*. [Online]
Available at: <https://medium.com/skyshidigital/documentation-testing-api-dengan-postman-part-1>
[Diakses 14 July 2020].
- Satzinger, J. W., Jackson, R. B. & Burd, S. D., 2011. *System Analysisi and Design in a Changing World*. 6 penyunt. s.l.:s.n.
- Setiawan, P., 2016. *Sistem Informasi Pendaftaran Praktek Kerja Lapangan (PKL) Berbasis Web Pada Prodi Sistem Informasi Universitas Nusantara PGRI Kediri*. Kediri: Universitas Nusantara PGRI Kediri.
- Solichin, A., 2006. *Diktat Kuliah Pemrograman Web 2*. Semarang: Universitas Negeri Semarang.
- Stauffer, M., 2017. *Laravelc- Up and Running*. California: O'Reilly Media, Inc.
- Supono & Putratama, V., 2016. *Pemograman Web Dengan Menggunakan PHP dan Framework Codeigniter*. Yogyakarta: Deepublish.
- Sutabri, T., 2012. *Analisis Sistem Informasi*. Yogyakarta: Andi Offset.
- Sutanta, E., 2004. *Sitem Basis Data*. Yogyakarta: Penerbit Graha Ilmu.
- Trimanto, A., 2018. *Sistem Informasi Praktek Kerja Lapangan Berbasis Web*. Yogyakarta: Universitas Gadjah Mada.
- Trimanto, A., 2018. *Sistem Informasi Praktek Kerja Lapangan Berbasis Web*. Yogyakarta: Universitas Gadjah Mada.
- Untung Rahardja, d., 2009. *Siapa Saja bisa membuat website dengan CSS dan HTML*. Yogyakarta: Andi Offset.

LAMPIRAN

KUISIONER

SISTEM INFORMASI KERJA PRAKTEK BAGIAN MAHASISWA PROGRAM STUDI KOMPUTER DAN SISTEM INFORMASI SV UGM BERBASIS WEB

Nama : Marsekal Rama Hattayudha

Jabatan : Mahasiswa

Berikan tanda centang (✓) pada nilai yang Anda anggap sesuai,

Keterangan :

1 : Sangat Tidak Setuju

2 : Tidak Setuju

3 : Cukup

4 : Setuju

5 : Sangat Setuju

No	Pertanyaan	Nilai				
		1	2	3	4	5
1	Sistem memudahkan pengguna dalam melakukan pendaftaran kerja praktek					✓
2	Sistem memudahkan pengguna dalam melakukan laporan harian kerja praktek				✓	
3	Sistem memudahkan pengguna dalam memberikan penilaian terhadap anggota kelompok				✓	
4	Sistem membuat kegiatan kerja praktek berjalan lebih efisien					✓
5	Secara umum sistem sudah sesuai dengan kebutuhan					✓
6	Sistem memiliki desain antarmuka yang baik					✓

7. Fitur manakah yang menurut Anda kurang dan perlu dikembangkan? Jelaskan!

Fitur Buat Kelompok membutuhkan pengembangan agar sistem dapat memenuhi kebutuhan yang sangat sesuai dengan sistem yang sudah berjalan.

8. Bagaimana pendapat Anda setelah menggunakan sistem tersebut? Berikan kritik dan saran sesuai pendapat anda

Ketika terjadi perubahan wajibkan pengguna untuk reload page mungkin perlu disempurnakan dan beberapa halaman kadang terjadi 404 Not Found mungkin karena terjadinya linux sensitive case.

Pengaji

Marsekal Rama Hattayudha

KUISIONER

SISTEM INFORMASI KERJA PRAKTEK BAGIAN MAHASISWA PROGRAM STUDI KOMPUTER DAN SISTEM INFORMASI SV UGM BERBASIS WEB

Nama : Nofa Dwi Adelia

Jabatan : Mahasiswa

Berikan tanda centang (✓) pada nilai yang Anda anggap sesuai,

Keterangan :

1 : Sangat Tidak Setuju

2 : Tidak Setuju

3 : Cukup

4 : Setuju

5 : Sangat Setuju

No	Pertanyaan	Nilai				
		1	2	3	4	5
1	Sistem memudahkan pengguna dalam melakukan pendaftaran kerja praktek					✓
2	Sistem memudahkan pengguna dalam melakukan laporan harian kerja praktek					✓
3	Sistem memudahkan pengguna dalam memberikan penilaian terhadap anggota kelompok					✓
4	Sistem membuat kegiatan kerja praktek berjalan lebih efisien					✓
5	Secara umum sistem sudah sesuai dengan kebutuhan					✓
6	Sistem memiliki desain antarmuka yang baik					✓

7. Fitur manakah yang menurut Anda kurang dan perlu dikembangkan? Jelaskan!
 Perlu ditambahkan fitur untuk update status magang supaya dosen bisa mengetahui status magang mahasiswa yang dibimbing
8. Bagaimana pendapat Anda setelah menggunakan sistem tersebut? Berikan kritik dan saran sesuai pendapat anda
 Sistem sangat membantu dan mudah digunakan, tampilan juga sudah bagus
 Pengaji



Nofa Dwi Adelia

KUISIONER

**SISTEM INFORMASI KERJA PRAKTEK BAGIAN MAHASISWA PROGRAM STUDI
KOMPUTER DAN SISTEM INFORMASI SV UGM BERBASIS WEB**

Nama : Febi Fiolanda Sari S

Jabatan : Mahasiswa

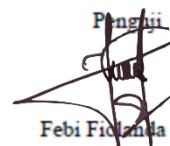
Berikan tanda centang (✓) pada nilai yang Anda anggap sesuai,

Keterangan :

- 1 : Sangat Tidak Setuju
- 2 : Tidak Setuju
- 3 : Cukup
- 4 : Setuju
- 5 : Sangat Setuju

No	Pertanyaan	Nilai				
		1	2	3	4	5
1	Sistem memudahkan pengguna dalam melakukan pendaftaran kerja praktek			✓		
2	Sistem memudahkan pengguna dalam melakukan laporan harian kerja praktek				✓	
3	Sistem memudahkan pengguna dalam memberikan penilaian terhadap anggota kelompok			✓		
4	Sistem membuat kegiatan kerja praktek berjalan lebih efisien				✓	
5	Secara umum sistem sudah sesuai dengan kebutuhan			✓		
6	Sistem memiliki desain antarmuka yang baik				✓	

7. Fitur manakah yang menurut Anda kurang dan perlu dikembangkan? Jelaskan!
Pada bagian penilaian, agar dapat memberikan penilaian kepada 1 kelompok
8. Bagaimana pendapat Anda setelah menggunakan sistem tersebut? Berikan kritik dan saran sesuai pendapat anda
Agar kelompok dapat memberikan nilai langsung terhadap teman nya berdasarkan 1 kelompok



Pengaji
Febi Fiolanda Sari S

KUISIONER

SISTEM INFORMASI KERJA PRAKTEK BAGIAN MAHASISWA PROGRAM STUDI
KOMPUTER DAN SISTEM INFORMASI SV UGM BERBASIS WEB

Nama : Muhamad Saiful Anam

Jabatan : Mahasiswa

Berikan tanda centang (✓) pada nilai yang Anda angap sesuai.

Keterangan :

1 : Sangat Tidak Setuju

2 : Tidak Setuju

3 : Cukup

4 : Setuju

5 : Sangat Setuju

No	Pertanyaan	Nilai				
		1	2	3	4	5
1	Sistem memudahkan pengguna dalam melakukan pendaftaran kerja praktek					✓
2	Sistem memudahkan pengguna dalam melakukan laporan harian kerja praktek				✓	
3	Sistem memudahkan pengguna dalam memberikan penilaian terhadap anggota kelompok				✓	
4	Sistem membuat kegiatan kerja praktek berjalan lebih efisien				✓	
5	Secara umum sistem sudah sesuai dengan kebutuhan					✓
6	Sistem memiliki desain antarmuka yang baik			✓		

7. Fitur manakah yang menurut Anda kurang dan perlu dikembangkan? Jelaskan!

Fitur yang masih kurang dan perlu dikembangkan bagi saya adalah bagian detail lowongan diberikan job deskripsi yang memberikan pandangan user untuk mengetahui pekerjaan apa yang akan dia lakukan selama Kerja Praktek.

Lalu untuk detail kelompok atau daftar anggota pada ketua dipastikan apabila menghapus anggota dilakukan antara kedua belah pihak yang dihapus serta yang menghapus daftar tersebut agar tidak terjadi kesengajaan penghapusan oleh ketua dengan niat yang salah.

8. Bagaimana pendapat Anda setelah menggunakan sistem tersebut? Berikan kritik dan saran sesuai pendapat anda

Secara kebutuhan, sistem ini sudah memenuhi berbagai kebutuhan yang dibutuhkan untuk mempermudah kegiatan Kerja Praktek, namun perlu dipertimbangkan di bagian mendetail seperti tampilan maupun seperti hal yang saya sebutkan sebelumnya di atas

Pengisi



Muhamad Saiful Anam

KUISIONER

SISTEM INFORMASI KERJA PRAKTEK BAGIAN MAHASISWA PROGRAM STUDI KOMPUTER DAN SISTEM INFORMASI SV UGM BERBASIS WEB

Nama : Bintang Lazuardi

Jabatan : Mahasiswa

Berikan tanda centang (✓) pada nilai yang Anda anggap sesuai,

Keterangan :

1 : Sangat Tidak Setuju

2 : Tidak Setuju

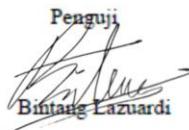
3 : Cukup

4 : Setuju

5 : Sangat Setuju

No	Pertanyaan	Nilai				
		1	2	3	4	5
1	Sistem memudahkan pengguna dalam melakukan pendaftaran kerja praktek					✓
2	Sistem memudahkan pengguna dalam melakukan laporan harian kerja praktek					✓
3	Sistem memudahkan pengguna dalam memberikan penilaian terhadap anggota kelompok					✓
4	Sistem membuat kegiatan kerja praktek berjalan lebih efisien				✓	
5	Secara umum sistem sudah sesuai dengan kebutuhan					✓
6	Sistem memiliki desain antarmuka yang baik				✓	

7. Fitur manakah yang menurut Anda kurang dan perlu dikembangkan? Jelaskan!
Pada fitur kelompok dapat ditambahkan sub-fitur untuk mengirimkan permintaan bergabung dengan kelompok lain, sehingga pengguna yang belum memiliki kelompok sendiri bisa aktif mencari kelompok dan tidak hanya menunggu undangan dari kelompok lain.
8. Bagaimana pendapat Anda setelah menggunakan sistem tersebut? Berikan kritik dan saran sesuai pendapat anda
Secara keseluruhan sistem informasi kerja praktek ini sudah bagus karena dapat memudahkan pengguna (mahasiswa) dalam melaksanakan kegiatan kerja praktek selain itu sistem ini juga mudah untuk dipelajari sehingga tidak membingungkan pengguna ketika akan menggunakan sistem.

Pengujii

Bintang Lazuardi