

**RANCANG BANGUN APLIKASI
SISTEM ADMINISTRASI BERBASIS WEB MENGGUNAKAN
ICONIX PROCESSDI LPK CIPTA KERJA DPN PERKASA JATENG**



SKRIPSI

**Disusun Sebagai Salah Satu Syarat
untuk Memperoleh Gelar Sarjana Komputer
pada Program Studi Informatika**

Disusun oleh:

Yesy Margharetta Munthe

24060121120031

**DEPARTEMEN INFORMATIKA
FAKULTAS SAINS DAN MATEMATIKA
UNIVERSITAS DIPONEGORO
2024**

HALAMAN PERNYATAAN KEASLIAN SKRIPSI

Saya yang bertanda tangan di bawah ini :

Nama : Yesy Margharetta Munthe

NIM : 24060121120031

Judul : Rancang Bangun Aplikasi Sistem Administrasi Berbasis Web Menggunakan ICONIX Process di LPK Cipta Kerja DPN Perkasa Jateng

Dengan ini saya menyatakan bahwa dalam skripsi ini tidak terdapat karya yang pernah diajukan untuk memperoleh gelar kesarjanaan di suatu Perguruan Tinggi, dan sepanjang pengetahuan saya juga tidak terdapat karya atau pendapat yang pernah ditulis atau diterbitkan oleh orang lain, kecuali yang secara tertulis diacu dalam naskah ini dan disebutkan di dalam daftar pustaka.

Semarang, 09 Juni 2024

Yesy Margharetta Munthe

24060121120031

HALAMAN PENGESAHAN

Judul : Rancang Bangun Aplikasi Sistem Administrasi Berbasis Web Menggunakan ICONIX Process di LPK Cipta Kerja DPN Perkasa Jateng

Nama : Yesy Margareta Munthe

NIM : 24060121120031

Telah diujikan pada sidang tugas akhir dan dinyatakan lulus pada tanggal

Semarang, 13 Januari 2024

Pembimbing 1,

Pembimbing 2,

Nurdin Bahtiar, S.Si., M.T.

Dr. Eng. Adi Wibowo S.Si., M.Kom.

NIP. 19790720 200312 1 002

NIP.

Mengetahui,

Ketua

Ketua

Departemen Informatika

Panitia Penguji Tugas Akhir

Drs Eko Adi Sarwoko M.Kom.

Bismillah yang baik aamiin yaallah

NIP. 196511071992031003

NIP.

KATA PENGANTAR

Segala puji syukur penulis ucapkan kepada Tuhan Yang Maha Esa atas karunia-Nya penulis dapat menyelesaikan penelitian “Rancang Bangun Aplikasi Sistem Administrasi Berbasis Web Menggunakan ICONIX Process di LPK Cipta Kerja DPN Perkasa Jateng” dengan baik. Laporan ini dapat diselesaikan dengan dukungan dari berbagai pihak yang telah berperan dalam kegiatan ini. Oleh karena itu, penulis mengucapkan terima kasih kepada:

1. Bapak Dr. Aris Sugiharto, S.Si., M.Kom., selaku Ketua Departemen Ilmu Komputer/ Informatika.
2. Bapak Nurdin Bahtiar, S.Si., M.T., selaku Dosen Pembimbing 1 Skripsi dan selaku Koordinator Skripsi yang telah membimbing penulis hingga terselesaiannya penelitian ini.
3. Bapak Drs Eko Adi Sarwoko, M.Kom selaku Dosen Pembimbing 2 Skripsi yang telah membantu penulis hingga terselesaiannya penelitian ini.
4. LPK Cipta Kerja DPN Perkasa Jateng selaku instansi yang menjadi instansi tempat penelitian dan pengembangan website ini.
5. Orang tua yang telah senantiasa bersama kelancaran penelitian ini dengan doa dan dukungan semangat.
6. Teman-teman angkatan informatika 2021 khususnya teman terdekat (Arynda, Kharisma dan Shidki) yang telah menjadi rekan seperjuangan penulis dalam pelaksanaan skripsi ini.
7. Semua pihak yang telah membantu kelancaran dalam pelaksanaan skripsi yang tidak dapat penulis sebutkan satu persatu.

Penulis menyadari terdapat kekurangan dalam penulisan laporan ini sehingga kritik dan saran yang membangun sangat penulis harapkan untuk perbaikan penulisan ilmiah yang akan datang. Penulis berharap laporan ini dapat bermanfaat bagi semua pihak.

Semarang, 03 Juni 2024

Yesy Marghareta Munthe

HALAMAN PERNYATAAN PERSETUJUAN PUBLIKASI SKRIPSI UNTUK KEPENTINGAN AKADEMIS

Sebagai civitas akademik Universitas Diponegoro, saya yang bertanda tangan di bawah ini:

Nama : Yesy Margharetta Munthe
NIM : 24060121120031
Program Studi : Informatika
Departemen : Informatika
Fakultas : Sains dan Matematika
Jenis Karya : Skripsi

demi pengembangan ilmu pengetahuan, menyetujui untuk memberikan **Hak Bebas Royalti Noneksklusif (Non-exclusive Royalty Free Right)** kepada Universitas Diponegoro atas karya ilmiah saya yang berjudul:

Rancang Bangun Aplikasi Sistem Administrasi Berbasis Web Menggunakan ICONIX Process di LPK Cipta Kerja DPN Perkasa Jateng

beserta perangkat yang ada (jika diperlukan). Dengan Hak Bebas Royalti Non-eksklusif ini Universitas Diponegoro berhak menyimpan, mengalihmedia/formatkan, mengelola dalam bentuk pangkalan data (*database*), merawat, dan mempublikasikan skripsi saya tanpa meminta izin dari saya selama tetap mencantumkan nama saya sebagai penulis/pencipta dan sebagai pemilik Hak Cipta.

Demikian pernyataan ini saya buat dengan sebenarnya.

Semarang, 09 Juni 2024

Materai
Rp. 10.000,-

Yesy Margharetta Munthe

24060121120031

ABSTRAK

LPK Cipta Kerja DPN Perkasa Jateng, yang berlokasi di Kabupaten Pekalongan, berperan penting dalam mempersiapkan tenaga kerja Indonesia agar memenuhi standar pasar kerja Jepang. Namun, sistem administrasi manual yang diterapkan lembaga ini menyebabkan ketidakefisienan dalam pengelolaan data siswa, instruktur, kelas, dan jadwal pelatihan. Untuk mengatasi permasalahan tersebut, dikembangkan aplikasi sistem administrasi berbasis web yang dirancang menggunakan metode *ICONIX Process*. Aplikasi ini dirancang untuk memenuhi kebutuhan utama lembaga dalam pengelolaan data administrasi secara efisien. Fitur-fitur yang disediakan meliputi pengelolaan data siswa, instruktur, kelas, dan jadwal pelatihan, serta pemantauan perkembangan siswa. Metode *ICONIX Process* diterapkan untuk memastikan aplikasi dibangun secara terstruktur, mulai dari analisis kebutuhan hingga pengujian, sehingga aplikasi dapat memenuhi kebutuhan pengguna secara optimal. Hasil implementasi menunjukkan bahwa aplikasi ini mampu meningkatkan efisiensi operasional, akurasi pengelolaan data, dan kualitas layanan lembaga. Dengan demikian, aplikasi ini diharapkan dapat mendukung LPK Cipta Kerja DPN Perkasa Jateng dalam mencetak tenaga kerja profesional yang siap bersaing di pasar kerja internasional.

Kata kunci : Sistem administrasi, LPK, *ICONIX Process*, aplikasi berbasis web, efisiensi.

ABSTRACT

LPK Cipta Kerja DPN Perkasa Jateng, located in Pekalongan Regency, plays a vital role in preparing Indonesian workers to meet the standards of the Japanese labor market. However, the manual administrative system currently implemented has led to inefficiencies in managing student, instructor, class, and training schedule data. To address these issues, a web-based administrative system application was developed using the *ICONIX Process* methodology. The application is designed to meet the institution's primary needs for efficient administrative data management. Its features include managing student, instructor, class, and training schedule data, as well as monitoring student progress. The *ICONIX Process* methodology ensures a structured development process, from requirement analysis to testing, enabling the application to optimally address user needs. The implementation results demonstrate that this application improves operational efficiency, data management accuracy, and the quality of institutional services. Consequently, the application is expected to support LPK Cipta Kerja DPN Perkasa Jateng in producing professional workers ready to compete in the international job market.

Keywords : administrative system, LPK, ICONIX Process, web-based application, efficiency.

DAFTAR ISI

KATA PENGANTAR.....	iv
HALAMAN PERNYATAAN PERSETUJUAN PUBLIKASI SKRIPSI UNTUK KEPENTINGAN AKADEMIS	v
ABSTRAK	vi
ABSTRACT	vii
DAFTAR ISI	viii
DAFTAR TABEL	x
DAFTAR GAMBAR.....	xi
BAB I.....	1
PENDAHULUAN	1
1.1 Latar Belakang.....	1
1.2 Rumusan Masalah.....	2
1.3 Tujuan.....	2
1.4 Manfaat.....	3
1.5 Ruang Lingkup.....	3
1.6 Sistematika Penulisan.....	4
BAB II	6
TINJAUAN PUSTAKA.....	6
2.1 Literature Review.....	6
2.2 Aplikasi Berbasis <i>Web</i>	8
2.3 Sistem Administrasi.....	10
2.4 Bahasa Pemrograman dan Database Management System.....	10
2.4.1 Bahasa Pemograman PHP	10
2.4.2 Bahasa Pemograman Javascript.....	10
2.4.3 Framework Laravel.....	11
2.4.4 Sistem Manajemen Basis Data MySQL.....	12
2.4.5 HTML & CSS.....	13
2.4.6 PHP My Admin.....	13
2.5 Metode Pengembangan ICONIX.....	14
2.5.1. Requirements	15
2.5.2. Use Case Modeling.....	17

2.6 Pengujian Perangkat Lunak.....	20
BAB III.....	22
METODE PENELITIAN	22
3.1 Tahap Requirements.....	22
4.2 Tahap Analysis/Preliminary Design.....	22
4.3 Tahap Detailed Design.....	23
4.4 Tahap Implementation.....	23
BAB IV.....	25
HASIL DAN PEMBAHASAN	25
4.1 Tahap Requirements.....	25
4.1.1 Proses Bisnis.....	25
4.1.2 Desain GUI (Grapichap User Interface)	40
4.1.3 Domain Model.....	49
4.2 Tahap Analysis/Preliminary Design.....	51
4.2.1. Robustness Diagram.....	51
4.2.2. Update Domain Model.....	57
4.3 Tahap Detailed Design.....	58
4.3.1 Sequence Diagram	58
4.3.2 Class Diagram	67
4.4 Tahap Implementation.....	68
4.4.1 Tahap Koding.....	68
4.4.2 Implementasi Antarmuka.....	72
BAB V.....	87
PENUTUP	87
DAFTAR PUSTAKA.....	88

DAFTAR TABEL

Tabel 2. 1 Penelitian Pengembangan Sistem Administrasi	6
Tabel 2. 2 Penelitian Pengembangan Aplikasi Menggunakan Metode ICONIX <i>Process</i>	7
Tabel 2. 3 Tabel simbol-simbol use case.....	17
Tabel 4. 1 Kebutuhan Fungsional.....	50
Tabel 4. 2 Kebutuhan Non Fungsional	51
Tabel 4. 3 Implementasi Model Sistem Manajemen Admininitrasi	71
Tabel 4. 4 Implementasi View Sistem Manajemen Administrasi	71
Tabel 4. 5 Implementasi Contoller Sistem Manajemen Administrasi.....	72

DAFTAR GAMBAR

Gambar 2. 1 Gambaran Kinerja ICONIX Process	15
Gambar 2. 2 domain modeling sistem penjualan online	16
Gambar 2. 3 contoh GUI Story Board.....	17
Gambar 2. 4 Contoh Use case ATM.....	19
Gambar 2. 5 Use case dengan include	20
Gambar 2. 6 Use case dengan extend	20
Gambar 4. 2 Activity Diagram untuk login.....	25
Gambar 4. 3 Activity Diagram untuk logout.....	26
Gambar 4. 4 Activity Diagram untuk dashboard Admin	26
Gambar 4. 5 Activity Diagram untuk menampilkan data bidang.....	27
Gambar 4. 6 Activity Diagram untuk menambah data bidang.....	27
Gambar 4. 7 Activity Diagram untuk mengubah data bidang	28
Gambar 4. 8 Activity Diagram hapus data bidang	29
Gambar 4. 9 Activity Diagram untuk menampilkan data kelas	29
Gambar 4. 10 Activity Diagram untuk menambah data kelas.....	30
Gambar 4. 11 Activity Diagram untuk mengubah data kelas.....	30
Gambar 4. 12 Activity Diagram untuk menghapus data kelas	31
Gambar 4. 13 Activity Diagram untuk menampilkan profil siswa.....	31
Gambar 4. 14 Activity Diagram profil instruktur.....	32
Gambar 4. 15 Activity Diagram profil admin	32
Gambar 4. 16 Activity Diagram menambahkan profil siswa	33
Gambar 4. 17 Activity Diagram menambahkan profil instruktur	34
Gambar 4. 18 Activity Diagram untuk menambahkan profil admin.....	34

Gambar 4. 19 Activity Diagram untuk mengubah profil siswa.....	35
Gambar 4. 20 Activity Diagram mengubah profil instruktur	35
Gambar 4. 21 Activity Diagram mengubah profil admin.....	36
Gambar 4. 22 Activity Diagram menghapus profil siswa	36
Gambar 4. 23 Activity Diagram menghapus profil instruktur.....	37
Gambar 4. 24 Activity Diagram menghapus profil admin	37
Gambar 4. 25 Activity Diagram untuk menambahkan akun	38
Gambar 4. 26 Activity Diagram untuk menghapus akun	38
Gambar 4. 27 Activity Diagram untuk menambahkan akun	39
Gambar 4. 28 Activity Diagram untuk mengedit status akun	39
Gambar 4. 29 Desain Halaman Login	40
Gambar 4. 30 Desain Halaman Menu	40
Gambar 4. 31 Desain Halaman <i>Reset Password</i>	41
Gambar 4. 32 Desain Halaman Admin.....	42
Gambar 4. 33 Desain Modal Ubah bidang	42
Gambar 4. 34 Desain Halaman Kelola Bidang.....	43
Gambar 4. 35 Desain Halaman Tambah Bidang	43
Gambar 4. 36 Desain Halaman Hapus Bidang	43
Gambar 4. 37 Desain Halaman Kelola Kelas	44
Gambar 4. 38 Desain Halaman Detail Kelas	44
Gambar 4. 39 Desain Modal Tambah Kelas	45
Gambar 4. 40 Desain Halaman Ubah Kelas	45
Gambar 4. 41 Desain Halaman Kelola Kelas	45
Gambar 4. 42 Desain Halaman Kelola Profil	46
Gambar 4. 43 Desain modal tambah siswa	46

Gambar 4. 44 modal tambah siswa.....	47
Gambar 4. 45 Desain Halaman Kelola Akun	47
Gambar 4. 46 Desain Halaman Tambah Akun.....	48
Gambar 4. 47 Desain Modal Ubah akun	48
Gambar 4. 48 Desain dropdown status	48
Gambar 4. 49 Desain Halaman Daftar Akun.....	49
Gambar 4. 50 Domain Model	49
Gambar 4. 51 Usecase Diagram	50
Gambar 4. 52 Robustness Diagram melakukan login	52
Gambar 4. 53 Robustness Diagram melakukan login	52
Gambar 4. 54 Robustness Diagram menampilkan dashboard admin.....	53
Gambar 4. 55 Robustness Diagram menampilkan dat bidang.....	53
Gambar 4. 56 Robustness Diagram menambah data bidang	53
Gambar 4. 57 Robustness Diagram mengubah data bidang	54
Gambar 4. 58 Robustness Diagram mengubah data bidang	54
Gambar 4. 59 Robustness Diagram menambah data kelas.....	54
Gambar 4. 60 Robustness Diagram mengubah data kelas	55
Gambar 4. 61 Robustness Diagram menghapus data kelas	55
Gambar 4. 62 Robustness Diagram menampilkan data kelas	55
Gambar 4. 63 Robustness Diagram menampilkan data akun.....	56
Gambar 4. 64 Robustness Diagram menambah data akun	56
Gambar 4. 65 Robustness Diagram mengubah data akun	56
Gambar 4. 66 Robustness Diagram menghapus data akun	57
Gambar 4. 67 <i>Update Domain Model</i>	58
Gambar 4. 68 <i>Sequence diagram</i> mengakses halaman login.....	59

Gambar 4. 69 <i>Sequence</i> diagram melakukan logout	59
Gambar 4. 70 <i>Sequence</i> diagram mengakses dashboard	59
Gambar 4. 71 Sequence diagram menampilkan daftar profil	60
Gambar 4. 72 Sequence diagram menambah profil	60
Gambar 4. 73 Sequence diagram mengubah profil	61
Gambar 4. 74 Sequence diagram menghapus profil.....	61
Gambar 4. 75 Sequence diagram menampilkan daftar akun	62
Gambar 4. 76 Sequence diagram menambah akun.....	62
Gambar 4. 77 Sequence diagram mengubah akun	62
Gambar 4. 78 Sequence diagram menghapus akun.....	63
Gambar 4. 79 Sequence diagram mengubah status akun	63
Gambar 4. 80 Sequence diagram menampilkan bidang	64
Gambar 4. 81 Sequence diagram menambah bidang	64
Gambar 4. 82 Sequence diagram mengubah bidang	64
Gambar 4. 83 Sequence diagram menghapus bidan.....	65
Gambar 4. 84 Sequence diagram mengelolah kelas	65
Gambar 4. 85 Sequence diagram menambah kelas	66
Gambar 4. 86 Sequence diagra mengubah kelas	66
Gambar 4. 87 Sequence diagram menghapus kelas	67
Gambar 4. 88 Sequence diagram melakukan daftar akun	67
Gambar 4. 89 Class Diagram.....	68
Gambar 4. 90 Tabel admin	69
Gambar 4. 91 Tabel user.....	69
Gambar 4. 92 Tabel Siswa.....	70
Gambar 4. 93 Tabel instruktur.....	70

Gambar 4. 94 Tabel Kelas	70
Gambar 4. 95 Tabel bidang_minat	70
Gambar 4. 96 Antarmuka Halaman Login	73
Gambar 4. 97 Antarmuka Halaman Menu.....	73
Gambar 4. 98 Antarmuka Halaman Daftar Akun.....	74
Gambar 4. 99 Antarmuka Halaman Reset Password.....	74
Gambar 4. 100 Antarmuka Dashboard Admin	75
Gambar 4. 101 Antarmuka Halaman Kelola Akun	75
Gambar 4. 102 Antarmuka modal tambah akun.....	76
Gambar 4. 103 Antarmuka modal konfirmasi	76
Gambar 4. 104 Atarmuka modal ubah akun.....	76
Gambar 4. 105 Atarmuka modal berhasil ubah akun	77
Gambar 4. 106 Antarmuka Ubah status akun.....	77
Gambar 4. 107 Antarmuka Ubah status akun.....	77
Gambar 4. 108 Antarmuka berhasil Ubah status siswa	78
Gambar 4. 109 Antarmuka Halaman Hapus Siswa	78
Gambar 4. 110 Antarmuka Halaman Kelola Akun Instruktur.....	78
Gambar 4. 111 Antarmuka pop up Tambah Akun Instruktur.....	79
Gambar 4. 112 Antarmuka Konfirmasi berhasil menambah instruktur	79
Gambar 4. 113 Antarmuka modal ubah akun instruktur	79
Gambar 4. 114 Antarmuka modal konfirmasi berhasil ubah akun instruktur	80
Gambar 4. 115 Antarmuka Halaman Hapus Akun Instruktur	80
Gambar 4. 116 Antarmuka Halaman Kelola Akun Admin	80
Gambar 4. 117 Antarmuka Halaman Tambah Akun Admin.....	81
Gambar 4. 118 Antarmuka Halaman Ubah Akun Admin	81

Gambar 4. 119 Antarmuka Halaman Hapus Akun Admin.....	81
Gambar 4. 120 Antarmuka Halaman Kelola Bidang.....	82
Gambar 4. 121 Antarmuka Halaman tambah bidang	82
Gambar 4. 122 Antarmuka Halaman ubah bidang	83
Gambar 4. 123 Antarmuka Halaman hapus Bidang	83
Gambar 4. 124 Antarmuka Halaman Kelola Kelas	84
Gambar 4. 125 Antarmuka Halaman Tambah Kelas.....	84
Gambar 4. 126 Antarmuka Halaman Ubah Kelas	84
Gambar 4. 127 Antarmuka Halaman Hapus Kelas Kondisi Sukses Hapus	85
Gambar 4. 128 Antarmuka Halaman Hapus Kelas Kondisi Gagal hapus	85
Gambar 4. 130 Antar muka Halaman Kelola Profil	86
Gambar 4. 131 Antar muka Halaman Tambah Profil.....	86
Gambar 4. 132 Antar muka Halaman tambah Profi	86

BAB I

PENDAHULUAN

1.1 Latar Belakang

Indonesia merupakan negara pengirim tenaga kerja migran terbesar kedua di ASEAN, setelah Filipina [1]. Berdasarkan laporan Bank Indonesia (BI), jumlah pekerja migran Indonesia tercatat mencapai 3,44 juta orang pada tahun 2022 [2]. Selain itu, laporan bulanan Data Layanan Penempatan Pekerja Migran Indonesia periode Januari hingga November 2024 dari Badan Nasional Penempatan dan Perlindungan Tenaga Kerja Indonesia (BNP2TKI) menunjukkan bahwa Jepang menjadi salah satu dari lima negara dengan layanan penempatan tertinggi, menyumbangkan 11.758 pekerja [3]. Hal ini menunjukkan bahwa Jepang adalah salah satu tujuan utama pekerja migran Indonesia, yang membutuhkan tenaga kerja terampil, khususnya yang menguasai bahasa Jepang dan memiliki keterampilan teknis tertentu.

Dalam hal ini, lembaga pelatihan kerja (LPK) memiliki peranan penting dalam mempersiapkan calon tenaga kerja Indonesia untuk memenuhi standar yang dibutuhkan oleh pasar kerja Jepang. LPK Cipta Kerja DPN Perkasa Jateng, yang berlokasi di Kabupaten Pekalongan, Jawa Tengah, merupakan lembaga yang fokus pada peningkatan kompetensi calon tenaga kerja Indonesia, khususnya dalam penguasaan bahasa Jepang, guna memenuhi kebutuhan tenaga kerja di Jepang. Jawa Tengah sendiri tercatat sebagai provinsi pengirim tenaga kerja migran terbesar kedua di Indonesia [3]. Oleh karena itu, LPK Cipta Kerja DPN Perkasa Jateng memegang peranan strategis dalam mencetak tenaga kerja yang terampil dan siap bersaing di pasar kerja internasional.

Namun, seiring dengan meningkatnya permintaan terhadap pelatihan ini, LPK Cipta Kerja DPN Perkasa Jateng menghadapi tantangan besar dalam hal sistem administrasi yang diterapkan. Proses administrasi yang masih dilakukan secara manual, termasuk pengelolaan data siswa, instruktur, kelas, dan jadwal pelatihan, menghambat efisiensi operasional dan kualitas layanan. Ketergantungan pada sistem manual ini berisiko menyebabkan kesalahan data serta ketidakefisienan dalam pengelolaan informasi, terutama dengan semakin banyaknya siswa yang terdaftar setiap tahunnya.

Untuk mengatasi tantangan tersebut, LPK Cipta Kerja DPN Perkasa Jateng memerlukan sistem administrasi yang lebih efektif dan efisien. Meskipun pendaftaran siswa dilakukan secara offline, penerapan sistem administrasi berbasis website dapat menjadi solusi yang tepat. Sistem ini memungkinkan pengelolaan data secara online, yang dapat mempercepat proses administrasi dan mengurangi risiko kesalahan atau kehilangan data. Selain itu, dengan sistem yang terintegrasi, lembaga dapat memantau perkembangan siswa secara real-time, sehingga setiap peserta pelatihan mendapatkan perhatian yang sesuai dengan bidang minat dan kebutuhan mereka.

Pengembangan sistem administrasi berbasis website ini telah terbukti efektif dalam meningkatkan efisiensi operasional dan kualitas layanan di berbagai lembaga pendidikan seperti pada berbagai universitas di Indonesia dan pelatihan lainnya. Dalam konteks LPK Cipta Kerja DPN Perkasa Jateng, penerapan sistem ini diharapkan dapat memberikan dampak positif yang signifikan. Dengan sistem yang terintegrasi, lembaga dapat meningkatkan pengelolaan data, mempercepat proses administrasi, serta memastikan kualitas pembelajaran dan layanan yang lebih baik bagi siswa.

Oleh karena itu, pengembangan sistem administrasi berbasis website di LPK Cipta Kerja DPN Perkasa Jateng merupakan langkah strategis yang perlu segera diambil untuk mengatasi masalah yang ada. Sistem ini tidak hanya akan membantu lembaga dalam mengelola data dan proses administrasi dengan lebih efisien, tetapi juga mendukung upaya lembaga dalam mencetak tenaga kerja profesional yang siap bersaing di pasar kerja internasional. Dengan meningkatkan efisiensi operasional dan kualitas layanan, LPK ini akan semakin siap untuk mengakomodasi peningkatan jumlah siswa yang terus berkembang, sekaligus memperkuat kontribusinya dalam pengiriman tenaga kerja terampil ke Jepang.

1.2 Rumusan Masalah

Berdasarkan permasalahan yang telah dijabarkan di latar belakang, disimpulkan sebuah rumusan masalah yaitu bagaimana cara membangun aplikasi sistem administrasi berbasis web di LPK Cipta Kerja DPN Perkasa Jateng.

1.3 Tujuan

Tujuan dari pengerjaan tugas akhir Rancang Bangun Aplikasi Sistem Administrasi LPK Cipta Kerja DPN Perkasa Jateng adalah sebagai berikut.

1. Mengembangkan sistem administrasi berbasis *web* di LPK Cipta Kerja DPN Perkasa Jateng menggunakan metode ICONIX *Process* untuk menggantikan proses manual dan mempermudah proses administrasi seperti pengelolaan siswa, instruktur, kelas dan jadwal pelatihan.
2. Meningkatkan efisiensi operasional dan kualitas layanan LPK dalam mendukung pencetakan tenaga kerja terampil yang siap bersaing khususnya di Jepang.

1.4 Manfaat

Berikut adalah manfaat pengembangan sistem administrasi di LPK Cipta Kerja DPN Perkasa Jateng.

1. Menyediakan solusi inovatif bagi LPK Cipta Kerja DPN Jateng pengembangan sistem administrasi yang dirancang untuk meningkatkan efisiensi layanan.
2. Mempermudah dan meningkatkan efisiensi proses administrasi dan mengurangi kesalahan data.
3. Mendukung optimalisasi layanan untuk membantu mempersiapkan tenaga kerja yang berkualitas untuk siap dikirim ke Jepang.

1.5 Ruang Lingkup

Penelitian dan pengembangan ini difokuskan pada solusi untuk meningkatkan efisiensi administrasi di LPK Cipta Kerja DPN Perkasa Jateng. Ruang lingkupnya mencakup aspek berikut.

1. Perjanjian kontrak kerjasama dilakukan di luar aplikasi.
2. Aplikasi memiliki 3 *user*, yaitu admin, instruktur, dan siswa.
3. Sistem hanya dapat digunakan oleh admin, instruktur, dan siswa yang telah terdaftar di LPK Cipta Kerja DPN Perkasa Jateng.
4. Activity Diagram untuk pengelolaan siswa, instruktur, kelas dan jadwal kelas yang dapat diakses oleh admin.

5. Activity Diagram untuk pendaftaran akun siswa yang memungkinkan siswa melakukan pendaftaran akun secara individu dan diaktivasi oleh admin.
6. Activity Diagram untuk menampilkan detail data siswa dan instruktur pada profil akun.
7. Aplikasi dapat menampilkan kelas dan jadwal kelas yang telah dibuat oleh admin.
8. Penelitian ini hanya mencakup pengembangan sistem untuk kebutuhan internal LPK Cipta Kerja DPN Perkasa Jateng. Sistem ini belum dirancang untuk mendukung pendaftaran online atau integrasi dengan pihak eksternal.

1.6 Sistematika Penulisan

Sistematika penulisan ini disusun untuk memberikan gambaran yang terstruktur dan jelas mengenai isi dan alur pembahasan dalam penyusunan laporan skripsi berjudul “Rancang Bangun Aplikasi Administrasi di LPK Cipta Kerja DPN Perkasa Jateng”. Adapun sistematika pembahasan laporan ini adalah sebagai berikut.

BAB I PENDAHULUAN

Bab ini memuat penjelasan mengenai latar belakang masalah, rumusan masalah, tujuan, manfaat, ruang lingkup, dan sistematika penulisan yang digunakan pada dokumen Skripsi.

BAB II TINJAUAN PUSTAKA

Bab ini memuat kajian terkait penelitian-penelitian terdahulu yang relevan sebagai referensi untuk mendukung penelitian ini dan berisi teori-teori yang menjadi landasan dalam proses perancangan dan pengembangan aplikasi.

BAB III METODOLOGI PENELITIAN

Bab ini menjelaskan metode yang diterapkan dalam proses perancangan dan pembangunan Aplikasi Aplikasi Sistem Administrasi Berbasis Web Menggunakan ICONIX Process di LPK Cipta Kerja DPN Perkasa Jateng yaitu dengan menggunakan metode ICONIX *Process* yang mencakup tahap

requirements, analysis/premiliary design, detailed design, dan implementation.

BAB IV PEMBAHASAN

Bab ini menjelaskan hasil penelitian Rancang Bangun Aplikasi Sistem Administrasi Berbasis Web Menggunakan ICONIX Process di LPK Cipta Kerja DPN Perkasa Jateng menggunakan metode ICONIX *Process*.

BAB V PENUTUP

Bab ini berisi ringkasan berupa kesimpulan penelitian yang telah dilaksanakan, mencakup pencapaian tujuan yang ditetapkan dengan rekomendasi dan saran untuk pengembangan lebih lanjut guna meningkatkan kualitas dan fungsionalitasnya di masa mendatang.

BAB II

TINJAUAN PUSTAKA

2.1 Literature Review

Dalam mendukung pengembangan aplikasi sistem administrasi, tinjauan pustaka ini bertujuan untuk menyajikan teori dasar dan hasil penelitian yang relevan terkait pengelolaan data administrasi. Literatur sebelumnya memberikan wawasan mengenai penerapan teknologi informasi untuk meningkatkan efisiensi dalam pengelolaan data seperti data registrasi, akun, kelas dan bidang sekaligus meminimalkan risiko kesalahan manual dalam proses administrasi.

Perbandingan penelitian terdahulu dilakukan untuk mengevaluasi pendekatan yang telah digunakan dalam pengembangan aplikasi administrasi. Analisis ini menjadi referensi dalam merancang aplikasi yang efektif dan efisien sesuai dengan kebutuhan lembaga. Dengan landasan teori yang kuat, aplikasi sistem administrasi yang dikembangkan diharapkan mampu mendukung operasional lembaga secara optimal dan meningkatkan kualitas pelayanan. Perbandingan penelitian yang relevan dapat dilihat pada Tabel 2.1.

Tabel 2. 1 Penelitian Pengembangan Sistem Administrasi

No	Penulis	Topik	Fitur
1	(Widiyatmoko, 2024)	Pengembangan Sistem Administrasi Sekolah TK Islam Pelita Insan Berbasis Web Menggunakan Nestjs	<ul style="list-style-type: none">• Pendaftaran akun siswa• Form pendaftaran siswa baru• Pengeloaan siswa dana kelas
2	(Yulianti, 2013)	Implementasi Arsitektur Client-Server dan MVC (Model-View-Controller) Untuk Membangun Aplikasi Administrasi Sekolah	<ul style="list-style-type: none">• Pengelolaan data staf administrasi, guru dan siswa• Pengelolaan data kelompok mata pelajaran, jurusan, kelas dan jadwal.• Pendaftaran dan penerimaan siswa serta pembayaran
3	(Yuzistin, dkk., 2016)	Sistem Informasi Administrasi Siswa Berbasis Website Pada SMA Islam Putradarma Bekasi	<ul style="list-style-type: none">• Melaksanakan KBM dan evaluasi• Melaksanakan Penilaian• Melaksanakan absensi

4	(Cobantoro,dkk, 2029)	Implementasi Sistem Administrasi Sekolah Berbasis ICT	<ul style="list-style-type: none"> Pengelolaan Administrasi sekolah
5	(Habibah, dkk., 2022)	Digitalisasi Administrasi Sekolah Melalui Pengembangan Website	<ul style="list-style-type: none"> Layanan administrasi kurikulum Layanan administrasi kesiswaan Layanan administrasi pendidik

Setelah penelitian-penelitian di atas, dilakukan riset mengenai aplikasi menggunakan metode yang akan diterapkan pada penelitian ini yaitu metode *ICONIX Process*. Perbandingan penelitian yang pernah dilakukan terdapat pada Tabel 2.2.

Tabel 2. 2 Penelitian Pengembangan Aplikasi Menggunakan Metode *ICONIX Process*

No	Penulis	Topik	Hasil
1	(Fitri Ana Wari, Seftin, dkk. 2024)	Sistem Informasi Bimbingan Tugas Akhir Mahasiswa menggunakan Model SDLC Berbasis Iconix Process	Penelitian ini mengembangkan sistem bimbingan tugas akhir berbasis SDLC dengan Iconix Process, yang terbukti efektif dalam memenuhi kebutuhan fungsional. Hasil pengujian menunjukkan sistem mendukung manajemen bimbingan di perguruan tinggi.
2	(Agustha, dkk.2024)	Pengembangan Aplikasi Monitoring Informasi Proyek Berbasis Web Menggunakan Metode <i>ICONIX Process</i> (Studi Kasus: PT Telkom Akses Witel Semarang)	Aplikasi "Monitoring Informasi Proyek" berbasis web berhasil mengatasi masalah integrasi data dan pemantauan tagihan pekerjaan di PT Telkom Akses Witel Semarang. Pengujian menunjukkan aplikasi ini memenuhi kebutuhan pengguna dan diterima dengan baik dalam setiap skenario pengujian..
3	(Yulianta dkk., 2016)	Pengembangan Aplikasi Web dengan <i>ICONIX Process</i> dan UML (Studi Kasus: Sistem Manajemen Isi)	Hasil pengembangan xCMS menggunakan <i>ICONIX Process</i> adalah prototipe aplikasi manajemen isi yang efektif untuk mengelola situs web PT X. Metode <i>ICONIX Process</i> dengan UML

			digunakan untuk mendokumentasikan sistem secara terstruktur.
4	(Ananda, Fikri dkk., 2024)	Penerapan Metode ICONIX Process Dalam Pengembangan Sistem Informasi Kepegawaian Universitas Catur Insan Cendekia	Aplikasi sistem informasi kepegawaian berbasis web di Universitas Catur Insan Cendekia berhasil mengelola data pegawai secara terstruktur dan efisien. Metode ICONIX Process memastikan pengembangan sistem yang terarah sesuai kebutuhan pengguna.
5	(Indra Gunawan Hutasuhut dkk., 2021)	<i>E-Learning</i> Pembelajaran Ilustrasi Menggunakan Metode ICONIX Process	Pengembangan sistem ini dirancang untuk memenuhi kebutuhan, sebagai platform bagi ilustrator profesional untuk menyalurkan aspirasi dan berbagi pengetahuan yang dapat mendukung ilustrator pemula.

Metode ICONIX terbukti efektif dalam pengembangan berbagai aplikasi web, termasuk dalam pengembangan sistem manajemen konten (CMS) dan aplikasi pengelolaan kepegawaian di Universitas Catur Insan Cendekia. Dalam pengembangan aplikasi CMS berbasis web, metode ini membantu merancang aplikasi yang dapat mengelola sumber daya situs web dengan lebih terstruktur, menggunakan UML untuk menggambarkan sistem secara jelas, dan menghasilkan prototipe yang sesuai dengan kebutuhan pengguna. Proses pengembangan dilakukan secara terperinci, dimulai dengan identifikasi kebutuhan hingga pengujian aplikasi untuk memastikan aplikasi sesuai dengan ekspektasi pengguna.

Begitu juga dalam pengembangan sistem informasi kepegawaian di Universitas Catur Insan Cendekia, penerapan ICONIX Process memastikan aplikasi dapat mengintegrasikan berbagai data pegawai secara efisien dan memudahkan akses informasi secara terpusat. Dengan pendekatan yang sistematis, pengembangan aplikasi ini mengarahkan proses pengelolaan data kepegawaian menjadi lebih terstruktur, akses mudah, dan efisien, serta memberikan solusi sesuai kebutuhan. Penerapan metode ICONIX dalam kedua proyek ini menunjukkan efektivitasnya dalam membangun aplikasi web yang memenuhi standar kualitas dan kebutuhan pengguna.

2.2 Aplikasi Berbasis Web

Aplikasi berbasis web adalah perangkat lunak yang dijalankan melalui server dan diakses menggunakan peramban internet tanpa memerlukan instalasi pada perangkat pengguna. Keunggulan utamanya adalah kemudahan akses, fleksibilitas, serta pengelolaan data yang terpusat, menjadikannya solusi ideal bagi lembaga pendidikan atau pelatihan yang membutuhkan efisiensi dalam pengelolaan administrasi. Dengan aplikasi berbasis web, data dapat diakses secara real-time dari berbagai perangkat, asalkan terhubung ke internet, sehingga mempermudah pengelolaan informasi dalam skala yang lebih luas.

Aplikasi berbasis web juga menawarkan sejumlah keunggulan lainnya, seperti efisiensi biaya dan pengurangan risiko kehilangan data. Sistem ini memungkinkan pengelolaan data dilakukan secara terpusat pada server, sehingga pencadangan menjadi lebih mudah dan aman. Selain itu, tidak ada kebutuhan untuk perangkat keras khusus atau instalasi perangkat lunak tambahan, sehingga biaya operasional dapat ditekan. Fitur-fitur ini menjadikan aplikasi berbasis web solusi yang sangat relevan untuk memenuhi kebutuhan administrasi modern di berbagai sektor.

Dalam pengembangan aplikasi berbasis web, metodologi ICONIX Process dapat digunakan untuk memastikan sistem yang dibangun sesuai dengan kebutuhan pengguna. Metode ini terdiri dari beberapa tahapan utama, yaitu analisis kebutuhan, pemodelan domain, desain kasus penggunaan, serta pengujian dan analisis ketahanan sistem. Pendekatan ini menitikberatkan pada pemahaman mendalam terhadap kebutuhan pengguna dan interaksi sistem, sehingga menghasilkan aplikasi yang tidak hanya fungsional tetapi juga dapat diandalkan. Dengan struktur yang sistematis, ICONIX Process memastikan pengembangan aplikasi berjalan efisien dan terarah.

Sistem administrasi berbasis web berperan penting dalam meningkatkan efisiensi operasional lembaga pendidikan atau pelatihan. Digitalisasi proses administrasi memungkinkan lembaga mengotomatisasi berbagai kegiatan, seperti pendaftaran peserta, pengelolaan jadwal, dan pemantauan perkembangan siswa. Selain itu, sistem ini juga dapat meningkatkan akurasi data serta mengurangi risiko kesalahan manual, yang sering terjadi dalam pengelolaan berbasis kertas. Dengan demikian, penerapan sistem administrasi berbasis web tidak hanya mendukung pengelolaan yang lebih efisien, tetapi juga membantu lembaga menyediakan layanan yang lebih berkualitas bagi pengguna.

Aspek keamanan dalam aplikasi berbasis web juga menjadi perhatian utama, terutama karena sistem ini sering menangani data sensitif. Untuk menjaga kepercayaan pengguna dan melindungi data, aplikasi berbasis web harus dilengkapi dengan fitur keamanan seperti autentikasi pengguna yang kuat, enkripsi data, serta audit keamanan secara berkala. Langkah-langkah ini diperlukan untuk meminimalkan risiko kebocoran data dan memastikan sistem beroperasi dengan aman. Dengan pendekatan yang terintegrasi terhadap efisiensi dan keamanan, aplikasi berbasis web dapat menjadi solusi yang andal untuk mendukung operasional lembaga secara optimal.

2.3 Sistem Administrasi

2.4 Bahasa Pemrograman dan Database Management System

2.4.1 Bahasa Pemograman PHP

PHP atau *Hypertext Preprocessor* adalah bahasa pemrograman *script server side* yang digunakan untuk memproses kode di sisi *server* sebelum mengirimkan halaman *web* yang dapat diakses oleh pengguna. Tujuan utamanya adalah untuk menghasilkan halaman *web* yang dinamis dan dapat berinteraksi dengan pengguna. Hal ini sejalan dengan penelitian oleh (Sari, 2023) yang menyatakan bahwa PHP adalah bahasa pemograman yang efektif dalam pengembangan aplikasi *web* dinamis karena kemampuannya dalam memproses permintaan dari pengguna dengan cepat dan efisien.

Seiring dengan popularitasnya, PHP telah menjadi salah satu bahasa pemrograman *server-side* yang paling banyak digunakan di dunia *web*. Kemudahan dalam mempelajari dan menggunakan PHP, bersama dengan dukungan luas komunitas pengembang, telah membuatnya menjadi pilihan yang populer untuk berbagai jenis proyek *web*. Dengan fitur-fitur yang terus berkembang dan perbaikan yang dilakukan secara teratur oleh komunitas PHP, bahasa ini terus mempertahankan posisinya sebagai salah satu alat utama dalam pengembangan aplikasi *web*. Dengan demikian, PHP tidak hanya menjadi alat yang efektif dalam menciptakan aplikasi *web* yang dinamis, tetapi juga menjadi fondasi yang kuat bagi berbagai inovasi dan proyek *web* masa depan (Sari, 2023).

2.4.2 Bahasa Pemograman Javascript

JavaScript adalah bahasa pemrograman yang sering digunakan dalam pengembangan aplikasi *web* untuk memberikan interaktivitas dan dinamika pada halaman *web*. Dengan JavaScript, pengembang dapat membuat efek visual, animasi, validasi formulir, dan berinteraksi dengan pengguna melalui elemen-elemen interaktif seperti tombol dan formulir. Bahasa ini berjalan di sisi klien (*client-side*), yang berarti kode JavaScript dieksekusi di perangkat pengguna akhir, seperti *browser web*, sehingga memungkinkan halaman *web* untuk merespons tindakan pengguna secara cepat dan dinamis.

Selain itu, JavaScript juga dapat digunakan di sisi server (*server-side*) dengan *platform* seperti Node.js, yang memungkinkan pengembang untuk membuat aplikasi *web* yang bersifat *real-time* dan dapat menangani banyak permintaan dengan efisien. Dengan keunggulan dalam memanipulasi DOM (*Document Object Model*) dan berinteraksi dengan API *browser*, JavaScript memainkan peran penting dalam mengubah halaman *web* menjadi aplikasi *web* yang interaktif dan responsif. Oleh karena itu, JavaScript telah menjadi salah satu teknologi yang paling penting dalam pengembangan *web* modern.

2.4.3 Framework Laravel

Laravel adalah sebuah *framework* atau kerangka kerja PHP 5.3 yang *open-source* dan mendapatkan popularitas yang tinggi di kalangan pengembang *web*. Kerangka kerja ini menawarkan sejumlah alat dan struktur yang diperlukan untuk membangun aplikasi *web* dengan kecepatan dan efisiensi yang tinggi (Selvaraj, 2023). Dengan menggunakan Laravel, pengembang dapat mempercepat proses pengembangan aplikasi *web*, sekaligus memanfaatkan fitur-fitur canggih yang disediakan dalam *framework* ini.

Dengan motto “*PHP doesn’t hurt, code happy & enjoy the fresh air*”, laravel bertujuan untuk memberikan kemudahan dalam pengembangan *website* dengan menciptakan lingkungan yang menyenangkan bagi para pengembang(Nugraha, n.d.). Pendekatan ini menjadikan laravel sebagai salah satu *framework* yang populer di kalangan pengembang *web*. Bahkan, pada tahun 2014 Laravel telah diakui sebagai salah satu *framework* PHP terbaik menurut webdesignmoo. Dengan popularitasnya yang terus meningkat, Laravel telah membuktikan dirinya sebagai salah satu alat yang sangat berharga dalam dunia pengembangan website.

Untuk menjalankan Laravel, terdapat beberapa kebutuhan dasar yang perlu dipenuhi. Pertama, memastikan tersedianya lingkungan pengembangan seperti *server web* Apache atau Nginx, serta interpreter PHP dalam versi yang direkomendasikan oleh Laravel. Selain itu, pengguna juga harus menginstal manajer paket PHP seperti Composer untuk mengelola dependensi proyek Laravel(Nugraha, n.d.). *Database* juga merupakan kebutuhan penting, dan Laravel mendukung berbagai jenis *database*, termasuk MySQL, PostgreSQL, SQLite, dan SQL Server. Selain itu, pengetahuan dasar tentang terminologi dan konsep dasar dalam pemrograman *web*, seperti HTTP, *routing*, dan MVC (*Model-View-Controller*). Dengan memenuhi kebutuhan dasar ini, pengembang dapat memulai menjalankan dan mengembangkan aplikasi *web* menggunakan Laravel.

2.4.4 Sistem Manajemen Basis Data MySQL

MySQL adalah sistem manajemen basis data relasional yang paling umum digunakan di dunia pengembangan perangkat lunak. Dalam MySQL, data disimpan dalam tabel-tabel yang terdiri dari baris-baris dan kolom-kolom, dengan setiap kolom mewakili atribut data tertentu. Kunci primer digunakan untuk mengidentifikasi secara unik setiap baris data dalam tabel, sementara kunci asing membangun hubungan antara tabel-tabel dalam basis data. Bahasa pemograman yang digunakan dalam mengelola MySQL adalah bahasa pemograman SQL (*Structured Query Language*).

SQL adalah sebuah bahasa pemrograman standar yang digunakan untuk mengelola basis data relasional. SQL memberikan kemudahan bagi pengguna untuk mengekstrak informasi yang dibutuhkan dari berbagai tabel dengan menggunakan perintah seperti *SELECT*, *INSERT*, *UPDATE*, dan *DELETE* (Beaulieu, 2009). SQL memiliki berbagai perintah seperti perintah DDL (*Data Definition Language*) yang digunakan untuk mendefinisikan struktur basis data, perintah DML (*Data Manipulation Language*) yang digunakan untuk memanipulasi data, dan perintah DQL (*Data Query Language*) yang digunakan untuk mengambil data dari basis data.

Keunggulan utama SQL adalah kemampuannya untuk berinteraksi dengan berbagai jenis basis data relasional seperti MySQL, PostgreSQL, SQLite, SQL Server, dan Oracle. Hal ini membuat SQL menjadi fleksibel dan dapat digunakan dalam berbagai lingkungan pengembangan serta sistem manajemen basis data (O'Neil et al., 2012). Dengan kelebihan

ini, SQL menjadi salah satu bahasa yang penting dalam dunia pengolahan basis data dalam memfasilitasi pengguna untuk mengelola data dengan efisien.

2.4.5 HTML & CSS

Hypertext Transfer Protocol (HTTP) adalah protokol komunikasi yang digunakan untuk mentransfer data melalui *World Wide Web*. HTTP merupakan dasar dari komunikasi data di internet dan memungkinkan *browser web* untuk mengambil halaman *web* dari server dan menampilkannya kepada pengguna. Protokol ini bekerja dalam model permintaan-respons, di mana klien mengirim permintaan HTTP ke *server web* untuk mengambil sumber daya tertentu, seperti halaman *web* atau file gambar, dan *server* merespon dengan mengirimkan sumber daya yang diminta kembali kepada klien. HTTP adalah bagian integral dari pengalaman berselancar di internet dan memungkinkan interaksi yang mulus antara pengguna dan aplikasi *web*.

Cascading Style Sheets (CSS) adalah bahasa pemrograman yang digunakan untuk mengontrol tampilan atau gaya presentasi halaman *web* yang ditulis dalam bahasa *markup* seperti HTML. CSS memungkinkan pengembang *web* untuk memisahkan struktur konten (HTML) dari presentasi atau gaya (CSS) halaman *web*, sehingga memungkinkan pembaruan dan modifikasi desain tanpa harus mengubah struktur dasar konten. Dengan CSS, pengembang dapat menentukan properti seperti warna, font, ukuran, dan tata letak elemen HTML yang memungkinkan mereka untuk menciptakan halaman *web* yang menarik dan responsif. CSS juga memungkinkan penggunaan gaya berlapis (*layered style*), di mana pengembang dapat menetapkan gaya yang berbeda untuk perangkat dan situasi yang berbeda, seperti tampilan desktop, tablet, dan ponsel.

2.4.6 PHP My Admin

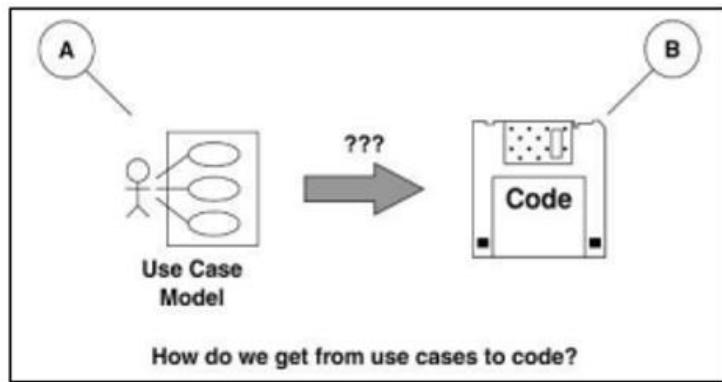
PHPMyAdmin dikembangkan dalam bahasa pemograman PHP, memungkinkan pengguna melakukan pengelolaan terhadap *database* dengan menjalankan *queri SQL* seperti membuat, mengedit, dan menghapus basis data, tabel, dan kueri SQL (Offner & Mendel, 2007). Desain antarmuka yang komunikatif dan kemudahannya dalam mengelola MySQL menjadikan aplikasi ini mudah digunakan dan banyak diminati oleh pengembang *web* dan *administrator database* (Cebulski & Suehring, 2009).

Selain itu, PHPMyAdmin juga menawarkan berbagai fitur tambahan yang mempermudah pengelolaan basis data, seperti visualisasi struktur basis data dalam bentuk diagram ER (*Entity-Relationship*), *eksport* dan *import* data dalam berbagai format file, dan manajemen hak akses pengguna. Fitur-fitur ini membuat PHPMyAdmin menjadi pilihan yang populer di kalangan pengembang *web* dan *administrator* basis data, karena dapat mempercepat dan menyederhanakan proses pengelolaan *database* MySQL. Dengan demikian, PHPMyAdmin menjadi salah satu alat yang penting dalam pengembangan dan pemeliharaan aplikasi *web* yang menggunakan MySQL sebagai basis datanya.

4.1 2.5 Metode Pengembangan ICONIX

ICONIX *Process* adalah sebuah model pengembangan perangkat lunak yang menggunakan pendekatan berorientasi objek dalam memproduksi sistem perangkat lunak yang berkualitas tinggi (Stephens & Rosenberg, 2010). Pendekatan ini berfokus pada pengembangan model *visual* komunikatif, yang didukung dengan pengujian *intensif* guna menjamin keakuratan dan keandalan sistem yang dikembangkan. Dengan demikian, ICONIX *Process* menawarkan pendekatan yang holistik dalam menghasilkan sistem perangkat lunak yang tidak hanya efisien dan andal, tetapi juga sesuai dengan kebutuhan dan spesifikasi pengguna.

Tujuan utama dari ICONIX *Process* adalah untuk menyusun strategi implementasi *use case* ke dalam kode seperti yang diilustrasikan dalam Gambar 2.1 (Yulianta & Mursanto, n.d.). Pada gambar tersebut, ICONIX *Process* digambarkan dalam tanda tanya diantara titik A dan B. Titik A merupakan *use case* yang disusun berdasarkan *requirement* yang ada, sementara titik B merupakan kumpulan kode utuh yang telah berhasil di uji untuk mewujudkan *use case* yang dijelaskan di titik A.



Gambar 2. 1 Gambaran Kinerja ICONIX Process

ICONIX *Process* ditemukan oleh Doug Rosenberg pada awal tahun 1990-an. Penemuan ini didasarkan pada pengabungan praktik terbaik dari tiga metedologi pengembangan lunak yang ada, yaitu *Object Modeling Techniques* (OMT) oleh James Rumbaugh, *Object-Oriented Software Engineering* (OOSE) oleh Ivar Jacobson, dan *Structural Method (Booch)* oleh Grady Booch (Rosenberg, 2002). Melalui penggabungan ini, ICONIX Process menggabungkan kekuatan dan pendekatan yang berbeda dari ketiga metodologi tersebut untuk membentuk suatu pendekatan yang lebih terstruktur dalam pengembangan perangkat lunak.

Menurut (Ristyawan & Harini, 2019)terdapat empat langkah dalam ICONIX *Process* yaitu *Requirements*, *Analysis dan Preliminary Design*, *Detailed Design* dan *Implementation* yang digambarkan dalam Gambar 3.1. Berikut akan dibahas lebih jelas terkait langkah-langkah pengembangan perangkat lunak dengan model pengembangan ICONIX Proces.

2.5.1. Requirements

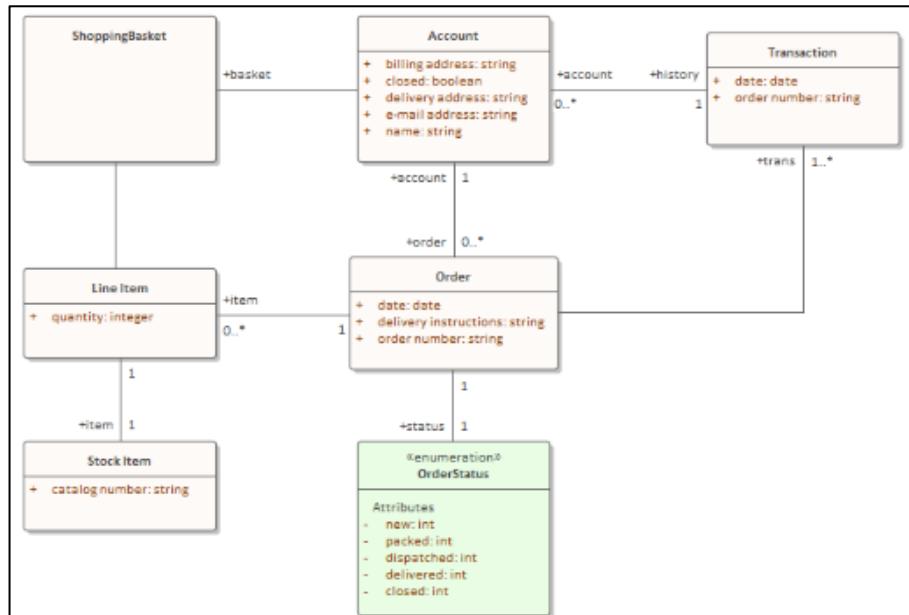
Pada bagian ini, terdapat beberapa tahapan yang harus dilakukan untuk memahami dan mendokumentasikan kebutuhan sistem secara rinci yaitu sebagai berikut.

1. Functional Requirements

Tahapan pertama adalah mendefinisikan *functional requirement*. *Functional Requirements* merupakan spesifikasi tentang apa yang dibutuhkan untuk pengembangan perangkat lunak. Proses ini melibatkan identifikasi kebutuhan fungsional secara terperinci yang akan menjadi acuan utama dalam mengembangkan perangkat lunak.

2. Domain Modeling

Selanjutnya, membuat *domain modeling* berupa kelas diagram yang merupakan representasi dari objek-objek dalam domain serta hubungan antar kelas tersebut. Tujuan utamanya adalah untuk memudahkan dalam memahami objek-objek yang ada dengan pemahaman dan konteks yang jelas. *Domain modeling* dimulai dengan mengidentifikasi entitas-entitas utama dalam sistem kemudian menentukan hubungan diantaranya. *Domain modeling* juga melibatkan penentuan atribut-atribut yang dimiliki oleh setiap entitas yang merepresentasikan karakteristik yang penting dalam pengembangan sistem. Proses ini akan membantu pengembang untuk memahami tentang lingkungan dimana sistem akan beroperasi dengan mudah. Gambar 2.2 merupakan gambar *domain modeling* dari sistem penjualan *online*.

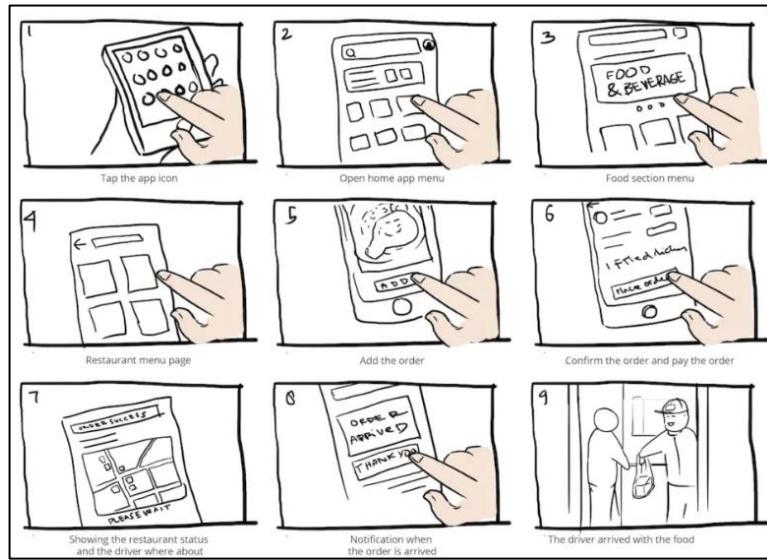


Gambar 2. 2 domain modeling sistem penjualan online

3. GUI *Storyboard*

Langkah selanjutnya adalah membuat GUI *storyboard* berupa gambaran awal dari tampilan perangkat lunak yang direncanakan. Tujuan dari pembuatan GUI adalah untuk memberikan visualisasi gambaran awal dari tampilan dan interaksi antar pengguna dengan sistem yang dikembangkan. Contoh dari GUI *Storyboard* dapat dilihat pada Gambar 2.3. Langkah pertama dalam membuat GUI adalah merancang sketsa kasar atau *mock up* dari halaman-halaman antarmuka pengguna utama. Selanjutnya, menyusun sketsa yang dibuat dalam urutan logis yang menggambarkan interaksi pengguna. Kemudian *story board* dapat ditambahkan

detail seperti warna, font, ukuran dan bahkan catatan atau keterangan tambahan yang dibutuhkan untuk memperjelas fungsi dari setiap elemen antarmuka pengguna.



Gambar 2. 3 contoh GUI Story Board

2.5.2. Use Case Modeling

Use case modeling bertujuan untuk menggambarkan interaksi pengguna dengan sistem. Pemodelan ini dilakukan dengan identifikasi, dokumentasi, dan analisis berbagai skenario penggunaan sistem. Setiap *use case* menggambarkan tindakan yang dilakukan oleh pengguna atau sistem, serta respon yang dihasilkan. Simbol-simbol yang digunakan dalam pembuatan *use case* dijelaskan lebih detail pada Tabel 2.3.

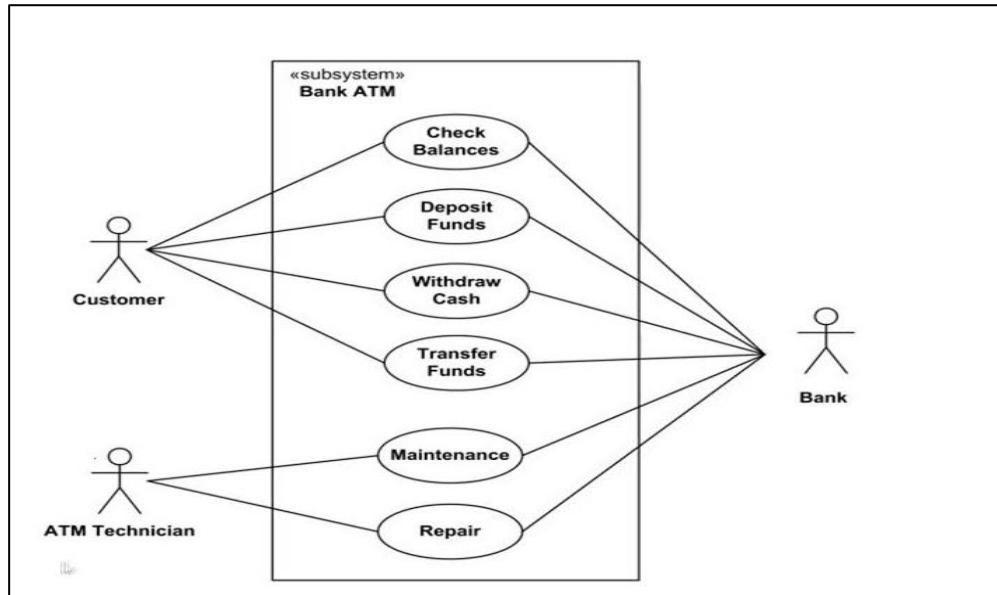
Tabel 2. 3 Tabel simbol-simbol use case

Simbol	Nama	Keterangan
○	<i>Actor</i>	Menspesifikasikan himpunan peran yang pengguna mainkan ketika berinteraksi dengan <i>use case</i> .
-----→	<i>Dependency</i>	Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri (<i>independent</i>) akan mempengaruhi

Simbol	Nama	Keterangan
		elemen yang bergantung padanya elemen yang tidak mandiri.
	<i>Generalization</i>	Hubungan dimana objek anak (<i>descendent</i>) berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk (<i>ancestor</i>).
	<i>Include</i>	Menspesifikasi bahwa <i>use case</i> sumber secara <i>eksplisit</i> .
	<i>Extend</i>	Menspesifikasi bahwa <i>use case</i> target memperluas perilaku dari <i>use case</i> sumber pada suatu titik yang diberikan.
	<i>Association</i>	Apa yang menghubungkan antara objek satu dengan objek lainnya.
	<i>System</i>	Menspesifikasi paket yang menampilkan sistem secara terbatas.
	<i>Use Case</i>	Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu aktor.
	<i>Collaboration</i>	Interaksi aturan-aturan dan elemen lain yang bekerja sama untuk menyediakan perilaku yang lebih besar dari jumlah dan elemen-elemennya (sinergi).
	<i>Note</i>	Elemen fisik yang eksis saat aplikasi dijalankan dan mencerminkan suatu sumber daya komputasi.

Langkah-langkah dalam membuat *use case* yang efektif dimulai dengan pembuatan template yang mencakup *Basic Course* dan *Alternate Course*. *Basic Course* menggambarkan jalur utama atau skenario paling umum dalam penggunaan sistem, sedangkan *Alternate Course* menggambarkan kemungkinan jalur *alternatif*. Proses identifikasi dimulai dengan pertanyaan "Apa yang akan terjadi?" diikuti dengan pertanyaan berulang untuk memperjelas

detail dari *Basic Course*. Setelah itu, skenario *alternatif* diidentifikasi dengan pertanyaan "Hal lain apa yang mungkin terjadi?". Adapun contoh *use case* dapat dilihat pada gambar 2.4.



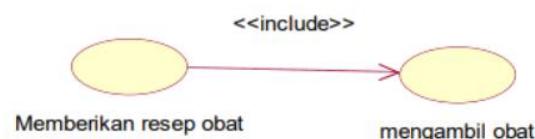
Gambar 2. 4 Contoh Use case ATM

Dalam diagram *use case*, terdapat beberapa jenis relasi yang menggambarkan hubungan antara *use case-use case*. Pertama, *Association* adalah jenis relasi yang menghubungkan dua elemen terkait tanpa menunjukkan ketergantungan fungsional. Kemudian, *Generalization* menggambarkan hubungan spesialisasi dan generalisasi antara *use case*. Relasi ini mencerminkan konsep pewarisan di mana karakteristik dan perilaku dari *use case* yang lebih umum diwariskan oleh *use case* yang lebih spesifik. Selanjutnya, *Dependency* menunjukkan ketergantungan antara dua elemen, di mana perubahan pada satu elemen dapat memengaruhi elemen lainnya. Relasi ini digunakan ketika satu *use case* menggunakan fungsionalitas yang disediakan oleh *use case* lain. Terakhir, *Aggregation* menggambarkan hubungan bagian ke keseluruhan di mana sebuah *use case* terdiri dari atau berisi beberapa *use case* lainnya.

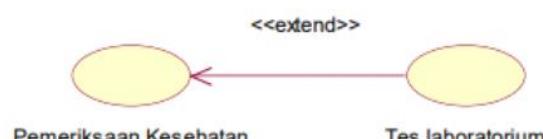
Selain relasi tersebut *use case* juga memiliki tiga tipe relasi yaitu *include*, *extends* dan *communicates*. Dalam diagram *use case*, arah panah pada relasi *extend* dan *include* menentukan aliran kontrol antara *use case* yang terlibat. Pada relasi *Include*, panah menunjukkan arah aliran kontrol dari *use case* induk ke *use case* yang di-*include*. Ini menggambarkan bahwa *use case* induk menentukan kapan dan di mana *use case* yang

diinclude akan dieksekusi. Dengan kata lain, *include* merupakan kelakuan yang harus terpenuhi agar sebuah *event* dapat terjadi.

Kedua, pada relasi *Extend*, panah menunjukkan arah ekstensi dari *use case* yang ada ke *use case* yang memberikan ekstensi. Ini mengindikasikan bahwa ekstensi yang ditambahkan hanya terjadi jika kondisi tertentu terpenuhi dalam *use case* yang ada. Dalam realisasinya di diagram tandah panah terbuka harus terarah ke *parent* atau ke *base use case*. Contoh *use case* dengan *extend* dan *include* lebih detail pada Gambar 2.5 dan Gambar 2.6.



Gambar 2. 5 Use case dengan include



Gambar 2. 6 Use case dengan extend

2.6 Pengujian Perangkat Lunak

Metode pengujian adalah cara atau teknik untuk menguji perangkat lunak. Metode pengujian berhubungan dengan perancangan data uji yang akan dieksekusi pada perangkat lunak yang dikembangkan. Metode pengujian diharapkan mempunyai mekanisme untuk menentukan data uji yang dapat menguji perangkat lunak secara lengkap (*completeness of test*) dan mempunyai kemungkinan tinggi untuk menemukan kesalahan (*high likelihood for uncovering error*).

Perangkat Lunak sendiri dapat diuji dengan dua cara yaitu:

1. Pengujian dilakukan dengan mengeksekusi data uji dan mengecek apakah fungsional perangkat lunak bekerja dengan baik. Data uji dibangkitkan dari spesifikasi perangkat lunak, yang dalam hal ini menjelaskan fungsional perangkat lunak. Cara pengujian ini disebut dengan *Black Box Testing*.

2. Pengujian dilakukan dengan mengenakan data uji untuk menguji semua elemen program perangkat lunak (data internal, lelaran (*loop*), logika keputusan, jalur). Data uji dibangkitkan dengan mengetahui struktur internal (kode sumber) dari perangkat lunak. Cara pengujian ini disebut dengan *White Box Testing*.

Metode pengujian white-box *testing* dan black-box *testing* mempunyai banyak jenis dan kriterianya. Penjelasan lebih lengkap untuk metode pengujian ini dapat dilihat pada lampiran. Selain *black box testing* dan *white box testing* terdapat pendekatan lain dalam metode pengujian, metode tersebut adalah pengujian berdasar kode sumber dan pengujian berdasar spesifikasi.

BAB III

METODE PENELITIAN

Bab ini menjelaskan metode yang digunakan dalam pengembangan Aplikasi Sistem Administrasi berbasis *web* di LPK Cipta Kerja DPN Perkasa Jateng, yaitu ICONIX Process. Metode ini meliputi empat tahap utama: *requirements*, *analysis/preliminary design*, *detailed design*, dan *implementation*.

3.1 Tahap Requirements

Tahap pertama dalam proses pengembangan adalah *requirements*, yang bertujuan untuk mendapatkan gambaran umum terkait aplikasi yang akan dikembangkan. Pada tahap ini, fokus utama adalah memahami proses bisnis yang ada, kebutuhan fungsional sistem, serta ruang lingkup aplikasi. Proses ini dimulai dengan melakukan wawancara dengan pemangku kepentingan untuk menggali informasi terkait kebutuhan sistem yang sebenarnya. Hasil dari wawancara ini menjadi dasar bagi perumusan kebutuhan fungsional yang akan diwujudkan dalam aplikasi.

Setelah kebutuhan fungsional teridentifikasi, langkah selanjutnya adalah pembuatan *GUI storyboard*, yang berfungsi sebagai representasi awal antarmuka pengguna. *GUI storyboard* ini menggambarkan desain awal sistem secara sederhana, mengacu pada kebutuhan yang telah dikumpulkan. Langkah berikutnya adalah pengembangan *domain model*, yang memberikan gambaran mengenai struktur statis sistem. *Domain model* ini menghubungkan elemen-elemen penting dalam sistem, menggambarkan kelas-kelas yang akan dibentuk, serta hubungan antar elemen tersebut. Proses ini dilanjutkan dengan pembuatan *use case diagram*, yang memetakan interaksi antara pengguna dan sistem serta menggambarkan berbagai aktivitas yang terjadi dalam aplikasi.

4.2 Tahap Analysis/Preliminary Design

Pada tahap *analysis/preliminary design*, dilakukan analisis lebih mendalam untuk merinci komponen-komponen yang belum teridentifikasi sebelumnya. Pembuatan *robustness diagram* menjadi langkah penting pada tahap ini, di mana diagram tersebut digunakan untuk menemukan objek-objek baru yang akan berperan dalam sistem.

Robustness diagram ini berfungsi sebagai jembatan antara analisis kebutuhan dan desain sistem, dengan fokus pada penentuan objek-objek yang berperan dalam setiap *use case*.

Setelah pembuatan *robustness diagram*, dilakukan pembaruan terhadap *domain model*. Pada tahap ini, kelas-kelas baru yang sebelumnya belum teridentifikasi akan ditambahkan. Selain itu, atribut-atribut yang dibutuhkan untuk setiap kelas juga ditambahkan agar *domain model* lebih lengkap dan siap digunakan dalam desain selanjutnya. Pembaruan ini juga bertujuan untuk memastikan bahwa semua elemen yang diperlukan untuk membangun sistem sudah terdefinisi dengan jelas.

4.3 Tahap Detailed Design

Setelah tahap analisis selesai, pengembangan berlanjut ke tahap *detailed design*. Pada tahap ini, fungsi-fungsi yang telah diidentifikasi pada tahap sebelumnya mulai dialokasikan ke dalam desain teknis yang lebih rinci. Salah satu langkah penting dalam tahap ini adalah pembuatan *sequence diagram*, yang menggambarkan interaksi antara objek-objek dalam sistem secara rinci sesuai dengan *use case* yang telah dibuat. *Sequence diagram* ini menjadi panduan untuk memahami alur eksekusi dari setiap fitur dalam sistem, yang akan menjadi acuan implementasi lebih lanjut.

Selanjutnya, dilakukan pembuatan *class diagram*, yang merupakan pemodelan statis lanjutan. Diagram ini dibuat berdasarkan pembaruan *domain model* dan diagram-diagram sebelumnya. *Class diagram* menggambarkan struktur dan hubungan antar kelas yang ada dalam sistem serta mendeskripsikan perilaku sistem berdasarkan fungsi yang dialokasikan pada masing-masing kelas. Diagram ini menjadi dasar untuk implementasi kode pada tahap berikutnya.

4.4 Tahap Implementation

Tahap terakhir dalam pengembangan sistem adalah *implementation*, di mana desain yang telah disusun diterjemahkan ke dalam kode program yang sesungguhnya. Pada tahap ini, seluruh hasil desain dari tahap-tahap sebelumnya, mulai dari *domain model*, *use case diagram*, hingga *class diagram*, diterjemahkan menjadi aplikasi yang berfungsi dengan baik menggunakan bahasa pemrograman PHP dan framework Laravel. Proses implementasi ini dilakukan dengan mematuhi spesifikasi teknis yang telah ditentukan sebelumnya.

Setelah implementasi selesai, dilakukan pengujian untuk memastikan bahwa aplikasi yang dikembangkan memenuhi kebutuhan yang telah digariskan di awal. Pengujian dilakukan menggunakan metode *black box testing* yang berfokus pada pengujian fungsionalitas aplikasi tanpa melihat struktur internal sistem. Pengujian dilakukan dengan menyusun rencana pengujian yang terdiri dari beberapa kelas uji, yang masing-masing berisi butir-butir uji berdasarkan skenario yang telah disusun. Setelah pengujian selesai, hasilnya dianalisis untuk memastikan bahwa sistem telah berfungsi sesuai dengan yang diharapkan dan dapat digunakan oleh pengguna dengan baik.

BAB IV

HASIL DAN PEMBAHASAN

Analisis kebutuhan adalah proses penting yang bertujuan untuk mengidentifikasi dan memahami kebutuhan fungsional dan nonfungsional dari aplikasi yang akan dibangun. Bab ini membahas hasil penelitian terkait pengembangan Aplikasi Sistem Administrasi Berbasis Web Menggunakan ICONIX Process di LPK Cipta Kerja DPN Perkasa Jateng.

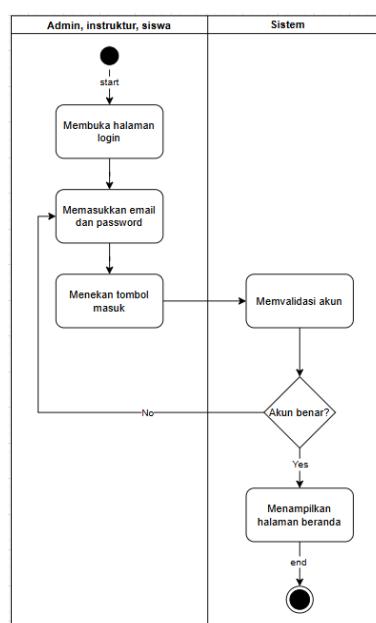
4.1 Tahap Requirements

Pada tahap ini, dilakukan identifikasi dan penentuan kebutuhan yang mencakup berbagai aspek seperti proses bisnis yang digambarkan melalui alur proses, desain antarmuka pengguna (GUI), model domain, serta diagram use case yang menggambarkan interaksi antara pengguna dan sistem. Selain itu, kebutuhan fungsional dari aplikasi juga ditetapkan pada tahap ini.

4.1.1 Proses Bisnis

1. Activity Diagram untuk login

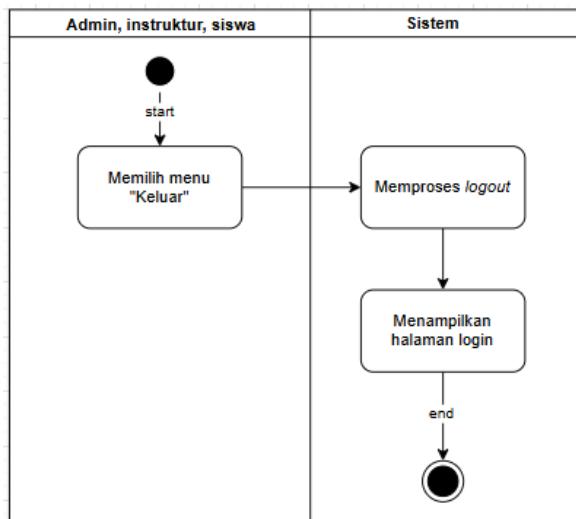
Diagram aktivitas pada Gambar 4.1 menggambarkan alur proses login pengguna ke dalam sistem.



Gambar 4. 1 Activity Diagram untuk login

2. Activity Diagram untuk logout

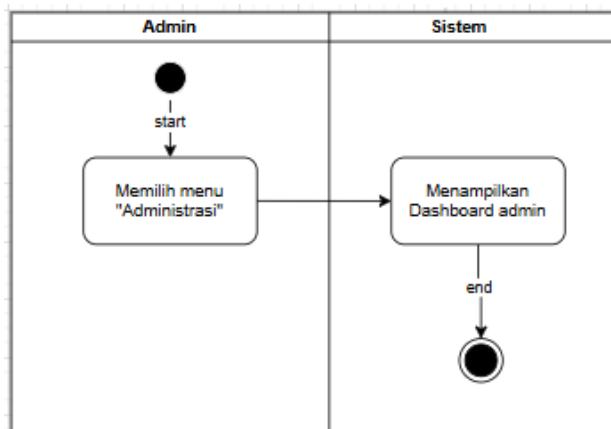
Diagram aktivitas pada Gambar 4.2 menggambarkan alur proses logout pengguna dari sistem.



Gambar 4. 2 Activity Diagram untuk logout

3. Activity Diagram untuk menampilkan dashboard Admin

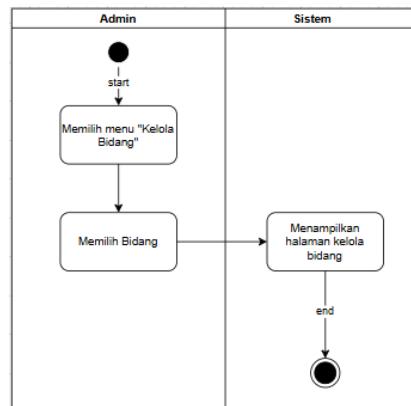
Diagram aktivitas pada Gambar 4.3 menggambarkan interaksi admin dengan sistem.



Gambar 4. 3 Activity Diagram untuk dashboard Admin

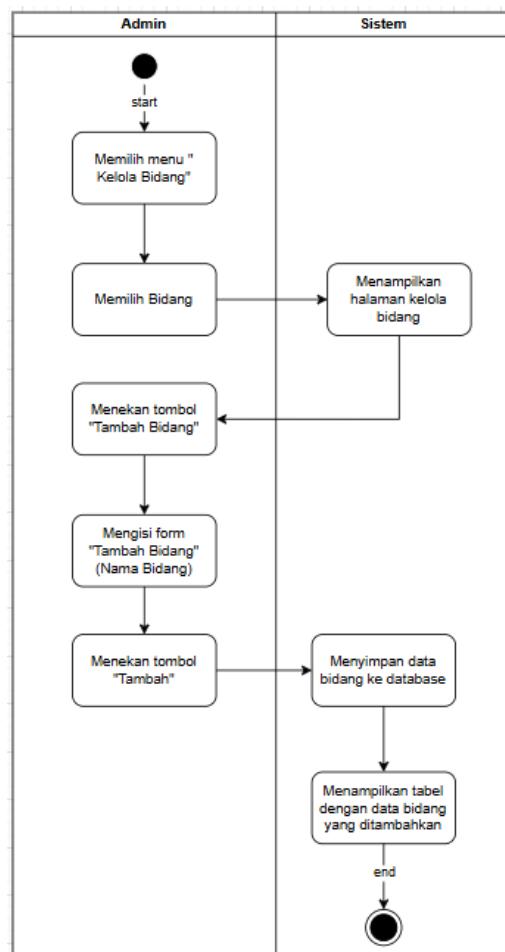
4. Activity Diagram untuk mengelola bidang

Diagram aktivitas pada Gambar 4.4 menggambarkan interaksi admin dengan sistem dalam mengelola data bidang meliputi



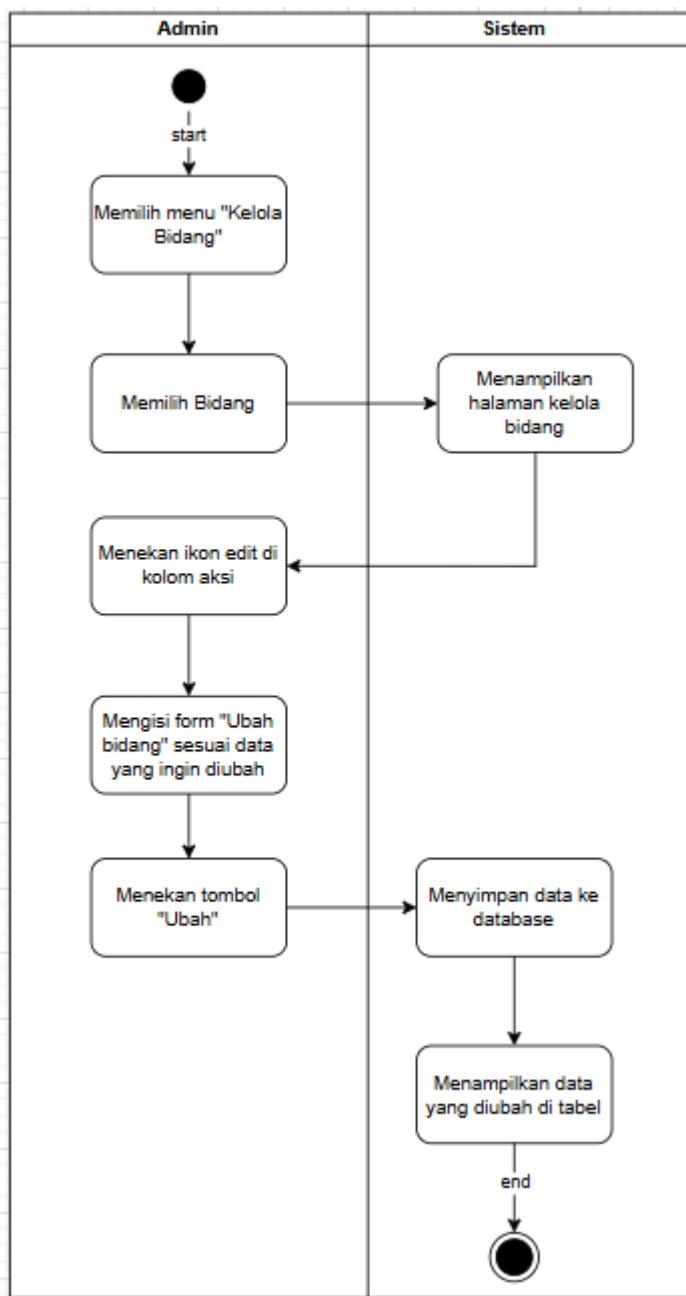
Gambar 4. 4 Activity Diagram untuk menampilkan data bidang

Diagram aktivitas pada Gambar 4.5 menggambarkan interaksi admin dengan sistem dalam menampilkan data bidang



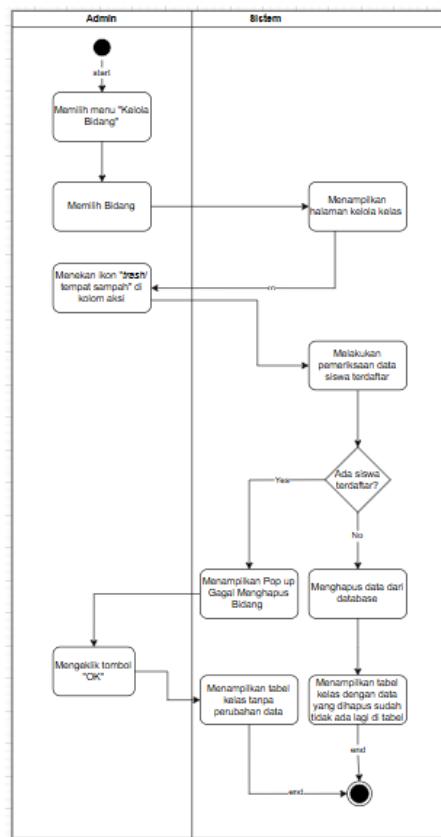
Gambar 4. 5 Activity Diagram untuk menambah data bidang

Diagram pada Gambar 4.6 menggambarkan interaksi admin dengan sistem dalam untuk mengubah data bidang.



Gambar 4. 6 Activity Diagram untuk mengubah data bidang

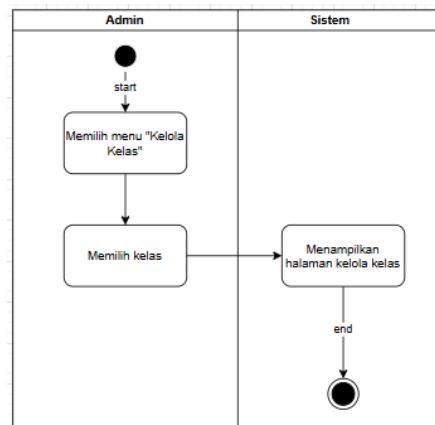
Diagram aktivitas pada Gambar 4.7 menggambarkan interaksi admin dengan sistem dalam untuk menghapus data bidang.



Gambar 4. 7 Activity Diagram hapus data bidang

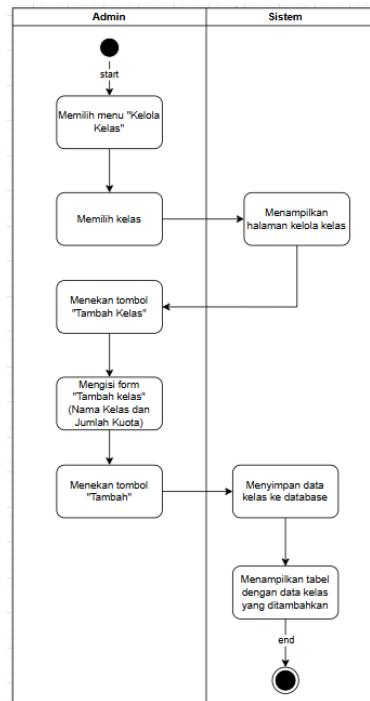
5. Activity Diagram untuk Mengelola data kelas

Diagram aktivitas pada Gambar 4.8 menggambarkan interaksi admin dengan sistem dalam mengelola data kelas dalam meliputi beberapa pengelolaan yakni untuk menampilkan kelola kelas.



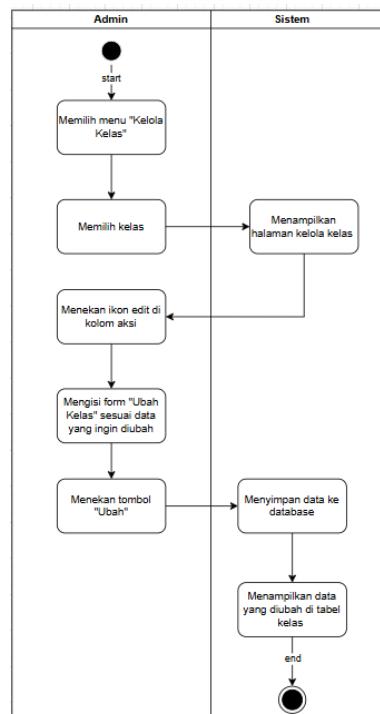
Gambar 4. 8 Activity Diagram untuk menampilkan data kelas

Diagram aktivitas pada Gambar 4.9 menggambarkan interaksi admin dengan sistem dalam menambah data kelas.



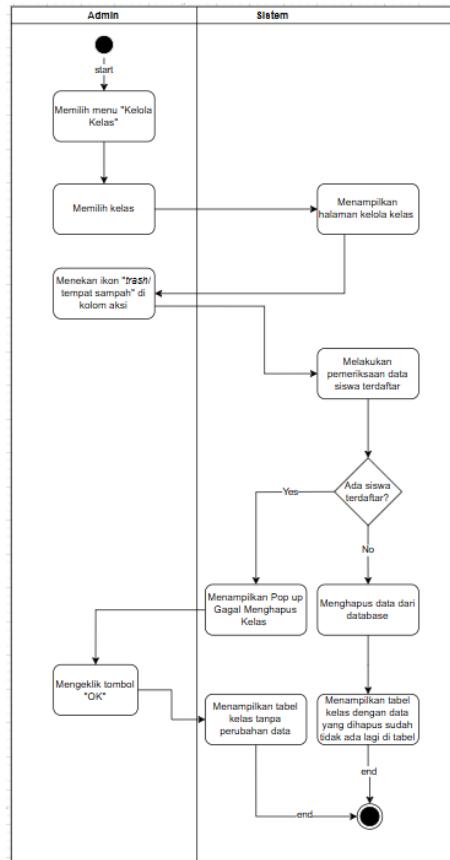
Gambar 4. 9 Activity Diagram untuk menambah data kelas

Diagram aktivitas pada Gambar 4.10 menggambarkan interaksi admin dengan sistem dalam mengubah data kelas.



Gambar 4. 10 Activity Diagram untuk mengubah data kelas

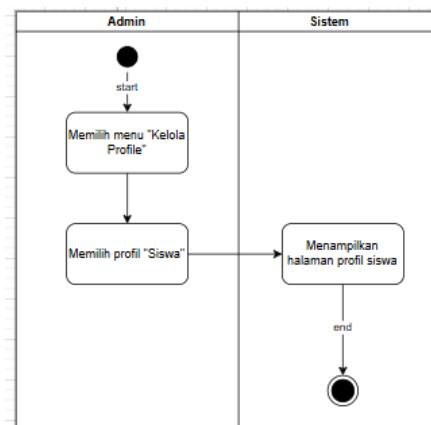
Diagram aktivitas pada Gambar 4.11 menggambarkan interaksi admin dengan sistem dalam menghapus data kelas.



Gambar 4. 11 Activity Diagram untuk menghapus data kelas

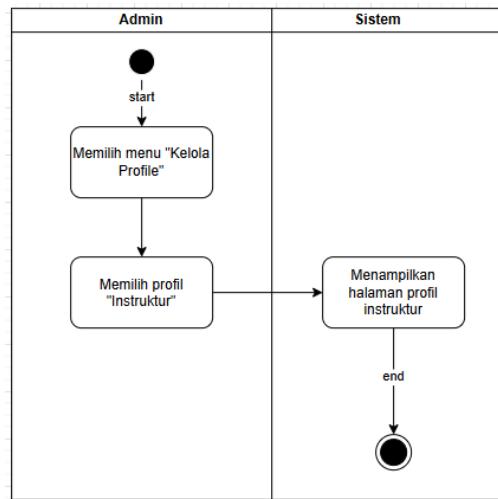
6. Activity Diagram untuk mengelola profil

Diagram aktivitas pada Gambar 4.12 menggambarkan interaksi admin dengan sistem dalam menampilkan profil siswa.



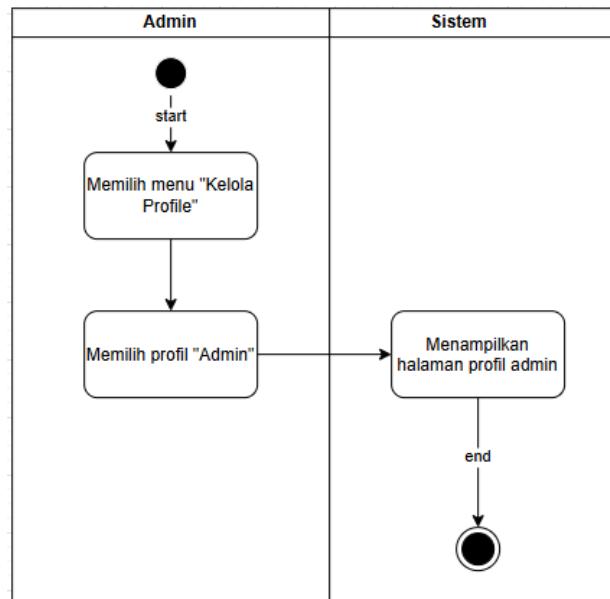
Gambar 4. 12 Activity Diagram untuk menampilkan profil siswa

Diagram aktivitas pada Gambar 4.13 menggambarkan interaksi admin dengan sistem dalam menampilkan profil instruktur.



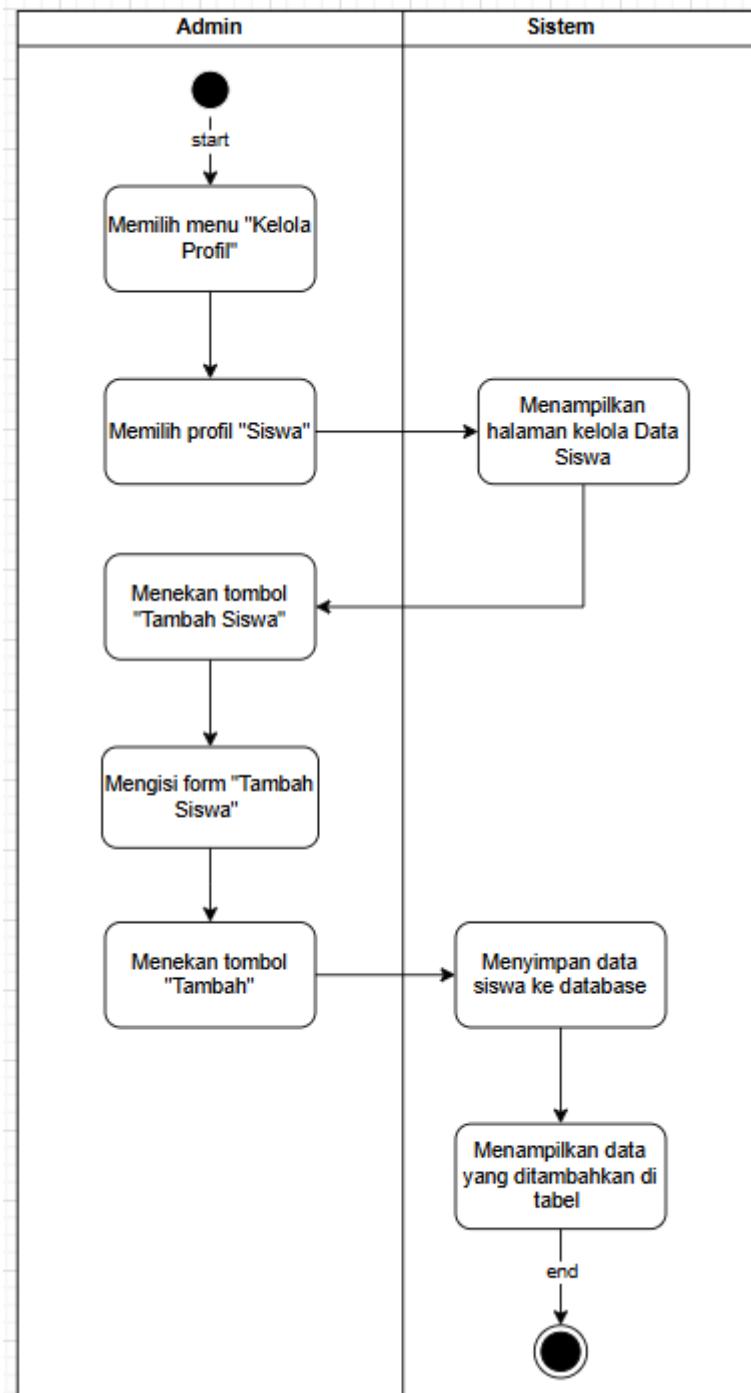
Gambar 4. 13 Activity Diagram profil instruktur

Diagram aktivitas pada Gambar 4.4 menggambarkan interaksi admin dengan sistem dalam menampilkan profil admin.



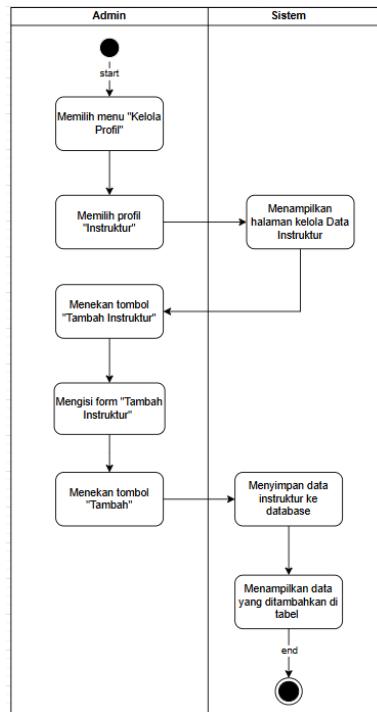
Gambar 4. 14 Activity Diagram profil admin

Diagram aktivitas pada Gambar 4.15 menggambarkan interaksi admin dengan sistem untuk menambahkan profil siswa.



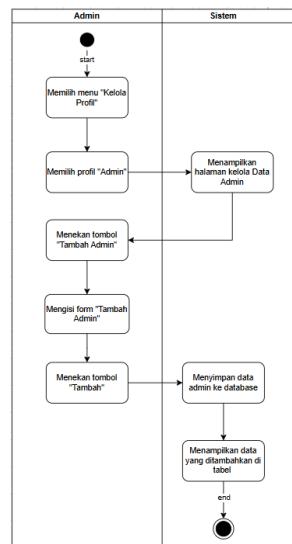
Gambar 4. 15 Activity Diagram menambahkan profil siswa

Diagram aktivitas pada Gambar 4.16 menggambarkan interaksi admin dengan sistem untuk menambahkan profil instruktur.



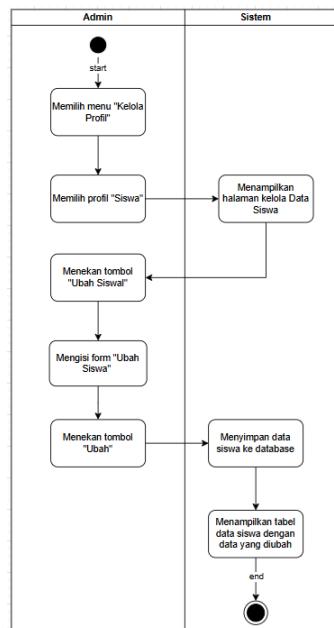
Gambar 4. 16 Activity Diagram menambahkan profil instruktur

Diagram aktivitas Gambar 4.17 menggambarkan interaksi admin dengan sistem untuk menambahkan profil admin.



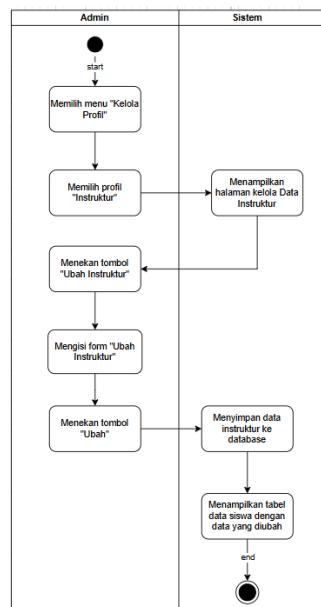
Gambar 4. 17 Activity Diagram untuk menambahkan profil admin

Diagram aktivitas Gambar 4.18 menggambarkan interaksi admin dengan sistem untuk menambahkan profil siswa.



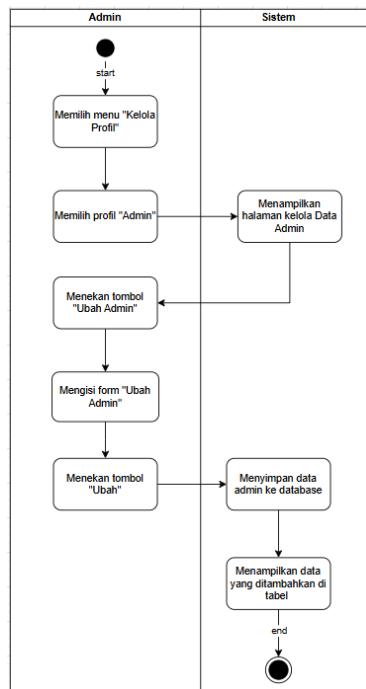
Gambar 4. 18 Activity Diagram untuk mengubah profil siswa

Diagram aktivitas Gambar 4.19 menggambarkan interaksi admin dengan sistem untuk menambahkan profil instruktur.



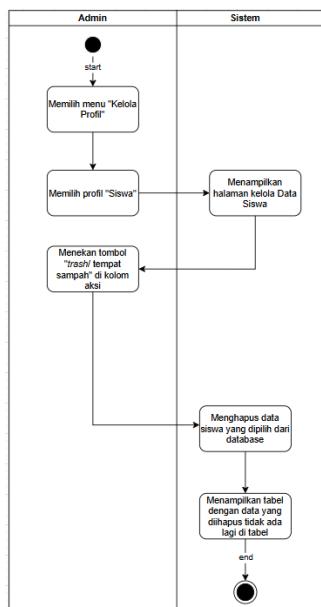
Gambar 4. 19 Activity Diagram mengubah profil instruktur

Diagram aktivitas Gambar 4.20 menggambarkan interaksi admin dengan sistem untuk menambahkan profil admin.



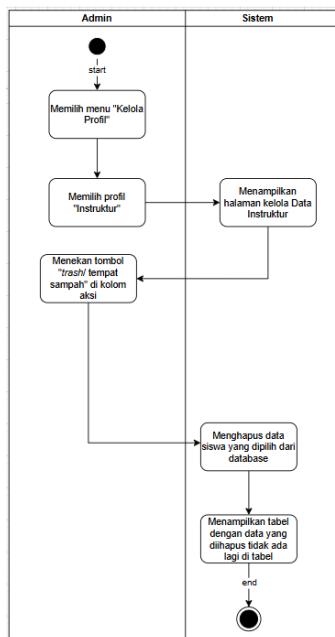
Gambar 4. 20 Activity Diagram mengubah profil admin

Diagram aktivitas Gambar 4.21 menggambarkan interaksi admin dengan sistem untuk menghapus profil siswa.



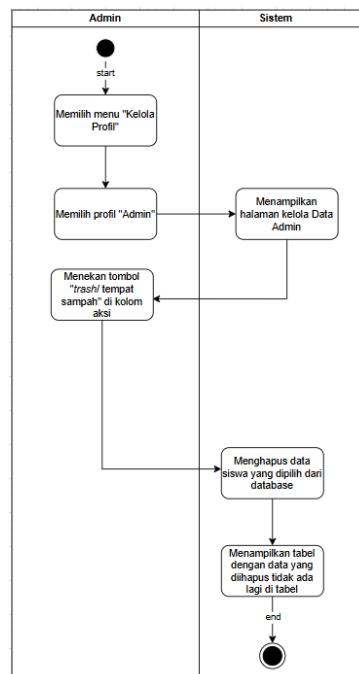
Gambar 4. 21 Activity Diagram menghapus profil siswa

Diagram aktivitas Gambar 4.22 menggambarkan interaksi admin dengan sistem untuk menghapus profil instruktur.



Gambar 4. 22 Activity Diagram menghapus profil instruktur

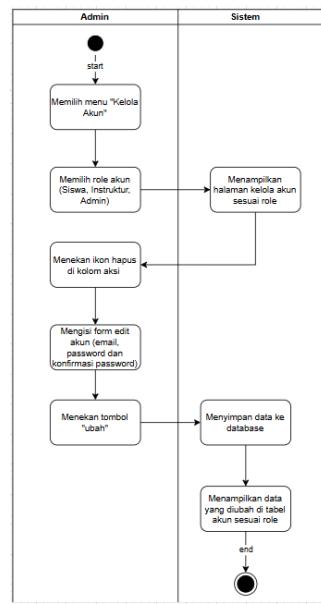
Diagram aktivitas Gambar 4.23 menggambarkan interaksi admin dengan sistem untuk menghapus profil admin.



Gambar 4. 23 Activity Diagram menghapus profil admin

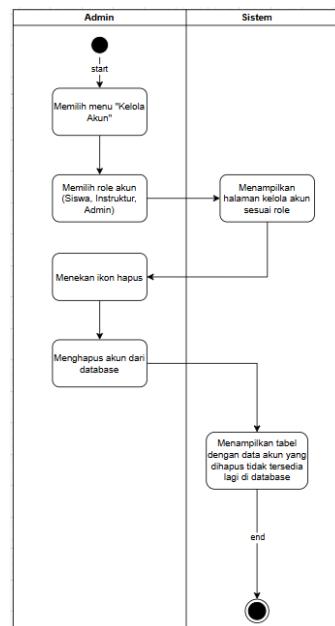
7. Activity Diagram Mengelola Akun

Diagram aktivitas Gambar 4.24 menggambarkan interaksi admin dengan sistem untuk menambah akun.



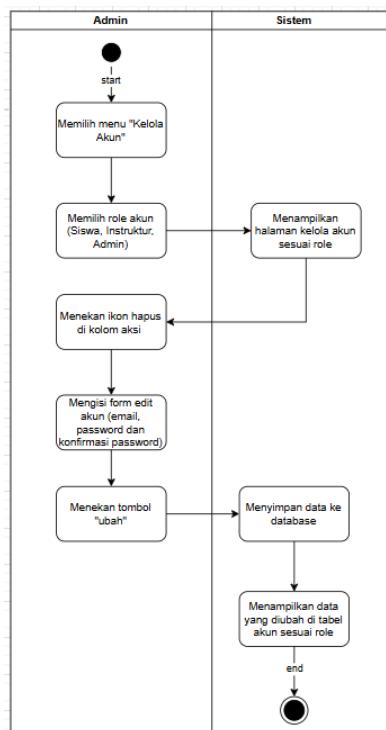
Gambar 4. 24 Activity Diagram untuk menambahkan akun

Diagram aktivitas pada Gambar 4.25 menggambarkan interaksi admin dengan sistem untuk menghapus akun.



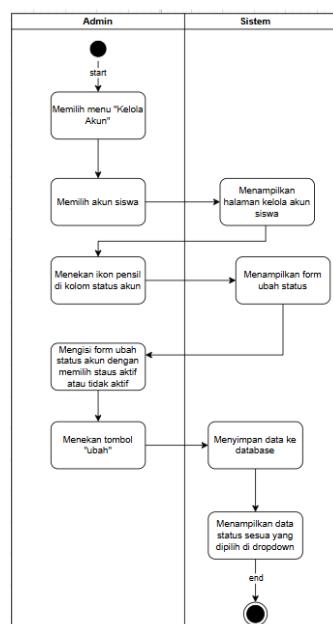
Gambar 4. 25 Activity Diagram untuk menghapus akun

Diagram aktivitas pada Gambar 4.26 menggambarkan interaksi admin dengan sistem untuk mengedit akun.



Gambar 4. 26 Activity Diagram untuk menambahkan akun

Diagram aktivitas pada Gambar 4.27 menggambarkan interaksi admin dengan sistem untuk mengedit status akun.



Gambar 4. 27 Activity Diagram untuk mengedit status akun

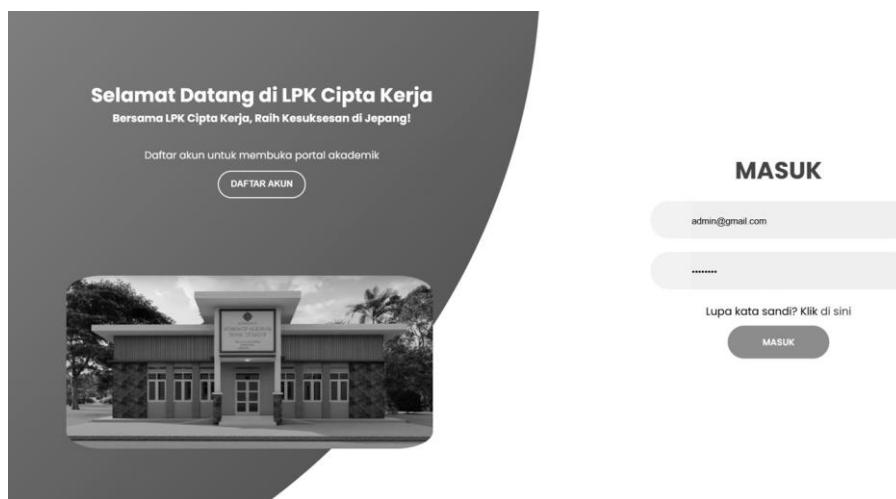
4.1.2 Desain GUI (Grapichap User Interface)

GUI *Storyboard* adalah representasi visual dari antarmuka pengguna (GUI) dalam bentuk urutan gambar atau sketsa yang menampilkan setiap layar atau halaman yang akan ada dalam aplikasi atau sistem.

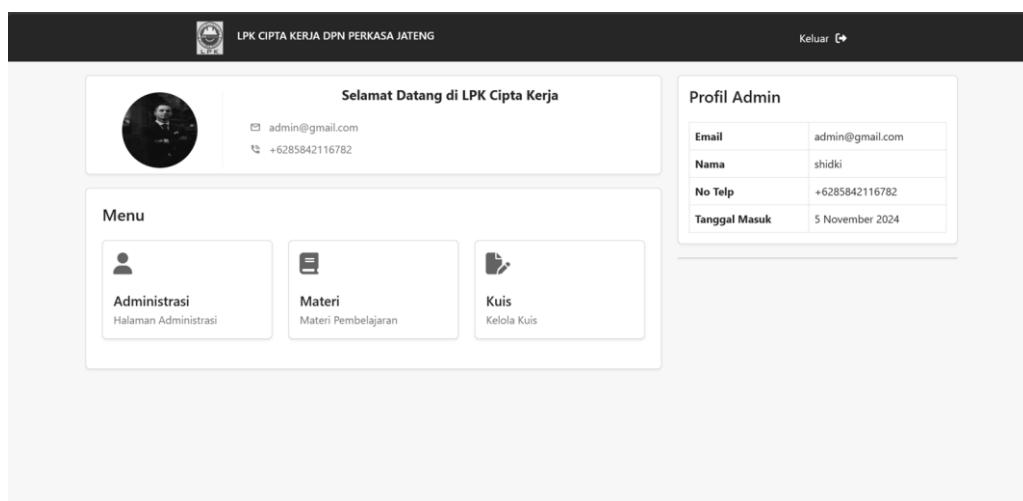
Adapun GUI dari Aplikasi Sistem Manajemen Administrasi Berbasis Web di LPK Cipta Kerja DPN Perkasa Jateng adalah sebagai berikut.

1. Desain Halaman Login

Berikut merupakan desain GUI untuk melakukan *login* yang dapat dilihat pada Gambar 4. 28. Setelah Login Admin akan diarahkan ke halaman menu admin yaitu pada Gambar 4.29.



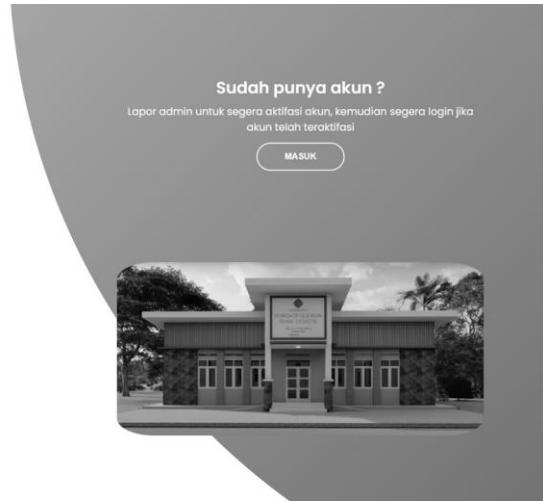
Gambar 4. 28 Desain Halaman Login



Gambar 4. 29 Desain Halaman Menu

2. Desain Halaman Daftar Akun

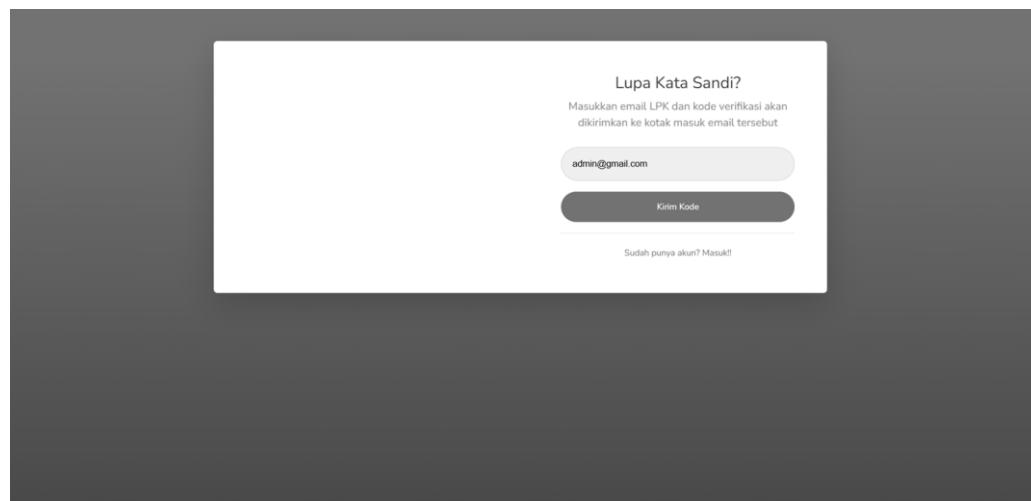
Berikut merupakan desain GUI untuk melakukan daftar akun yang dapat dilihat pada Gambar 4. 30.



Gambar 4. 30 Desain Halaman Daftar Akun

3. Desain Halaman *Reset Password*

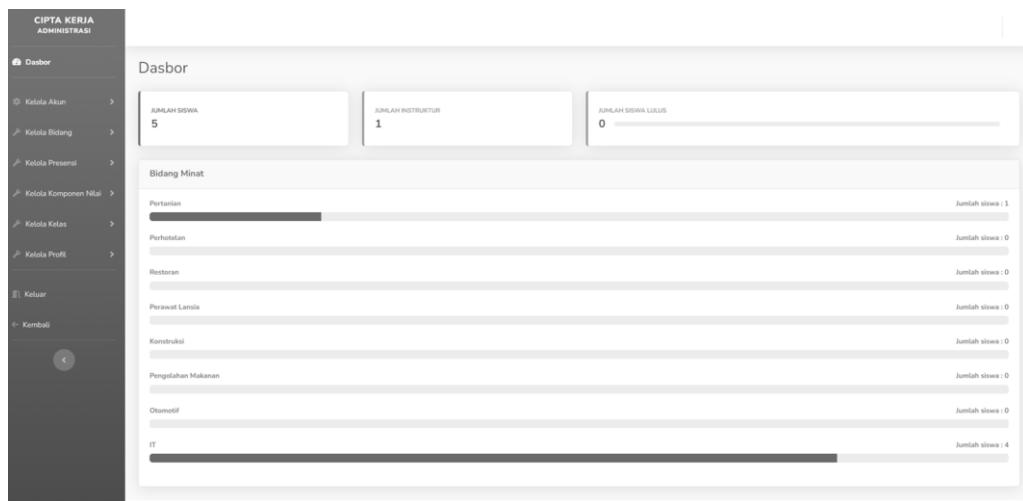
Berikut merupakan desain GUI untuk melakukan *reset password* yang dapat dilihat pada Gambar 4. 30.



Gambar 4. 31 Desain Halaman *Reset Password*

4. Desain Dashboard Admin

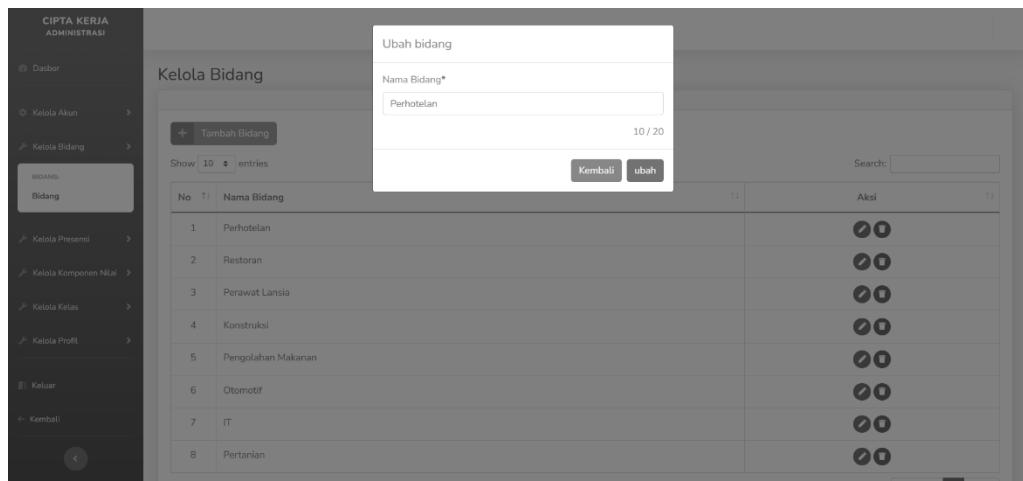
Berikut merupakan desain GUI untuk melakukan daftar akun yang dapat dilihat pada Gambar 4. 31.



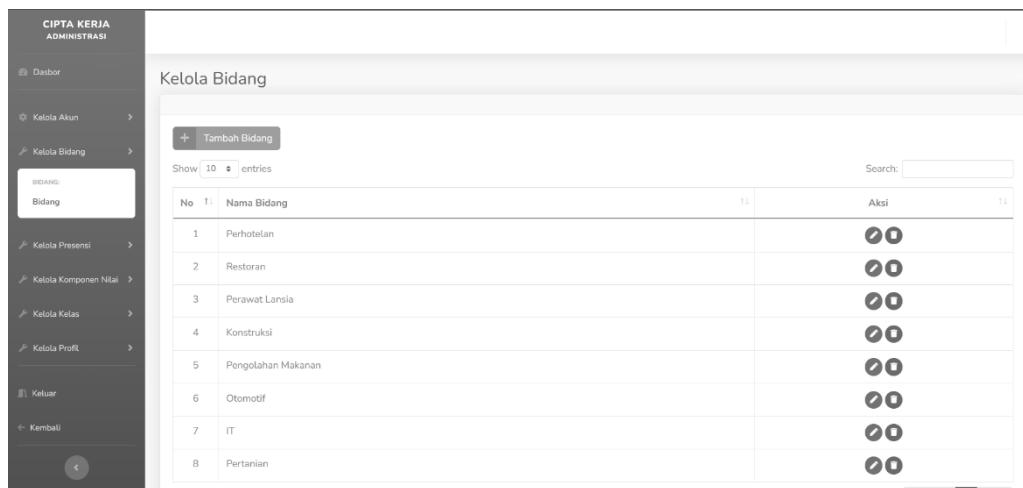
Gambar 4. 32 Desain Halaman Admin

5. Desain Halaman Kelola Bidang

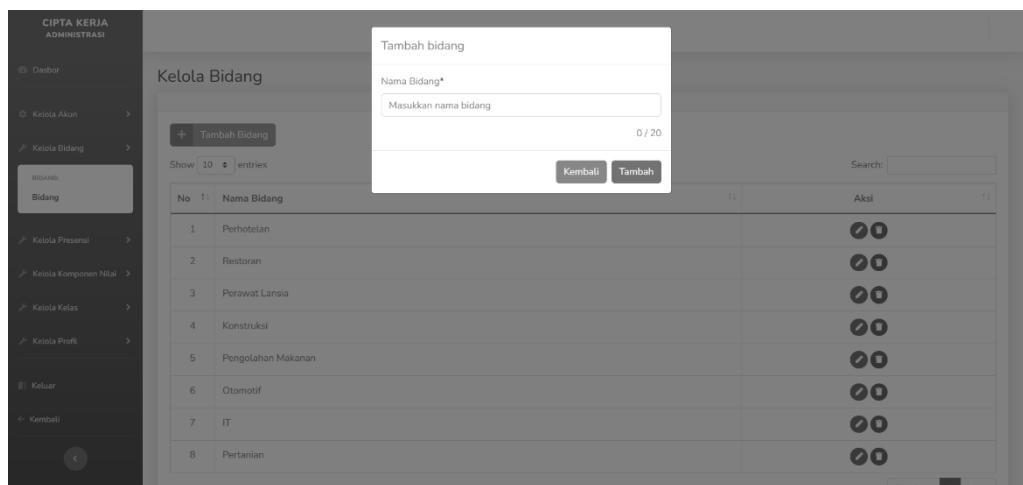
Berikut merupakan desain GUI untuk melakukan kelola bidang seperti menampilkan halaman kelola bidang, serta menambah, megubah dan menghapus bidang yang dapat dilihat pada Gambar 4. 32, Gambar 4. 33, Gambar 4. 34, dan Gambar 4. 35.



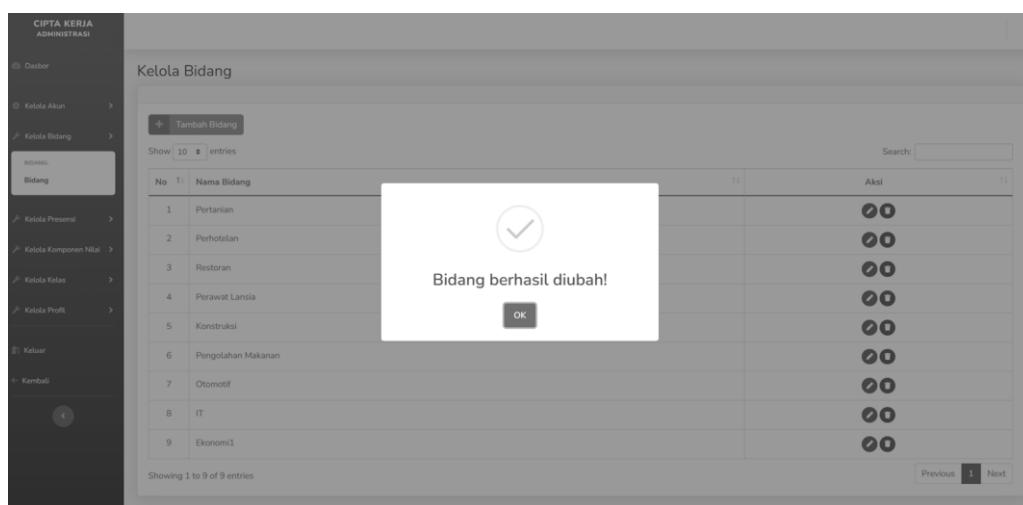
Gambar 4. 33 Desain Modal Ubah bidang



Gambar 4. 34 Desain Halaman Kelola Bidang



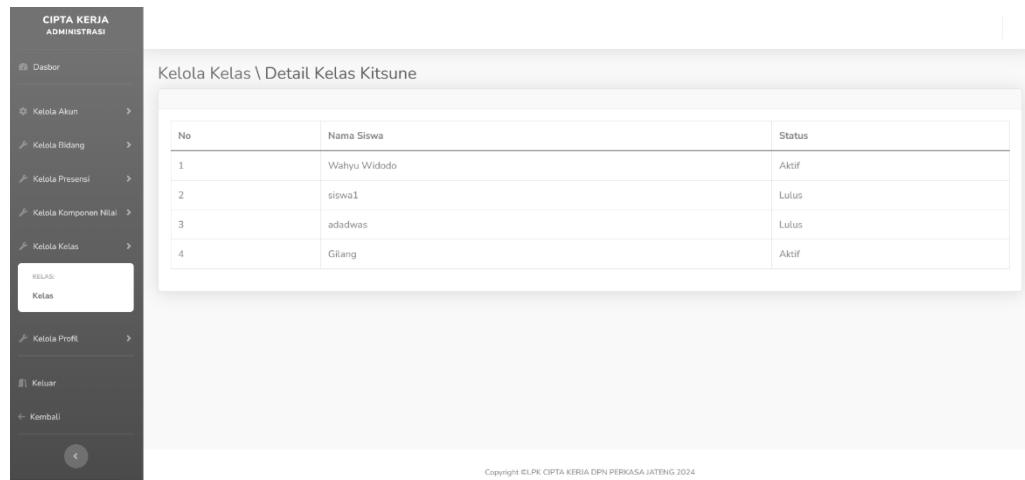
Gambar 4. 35 Desain Halaman Tambah Bidang



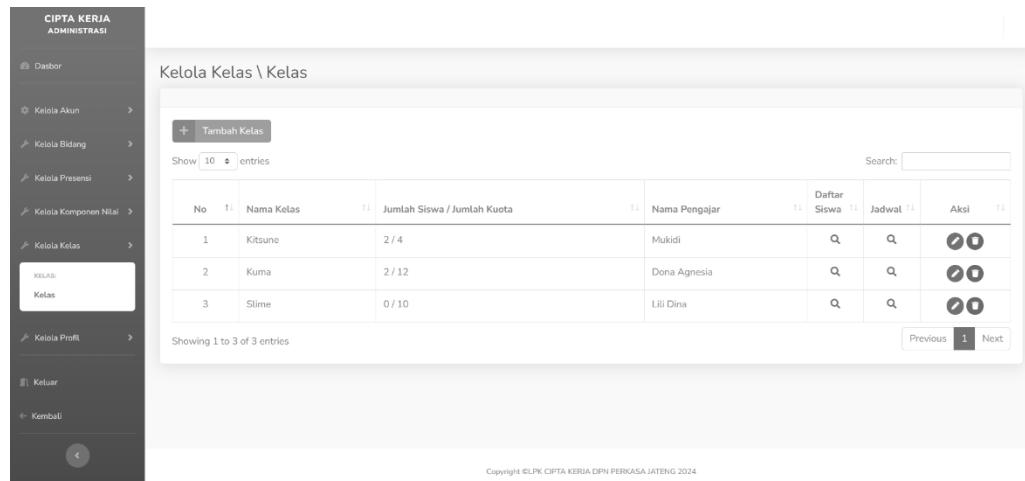
Gambar 4. 36 Desain Halaman Hapus Bidang

6. Desain Halaman Kelola Kelas

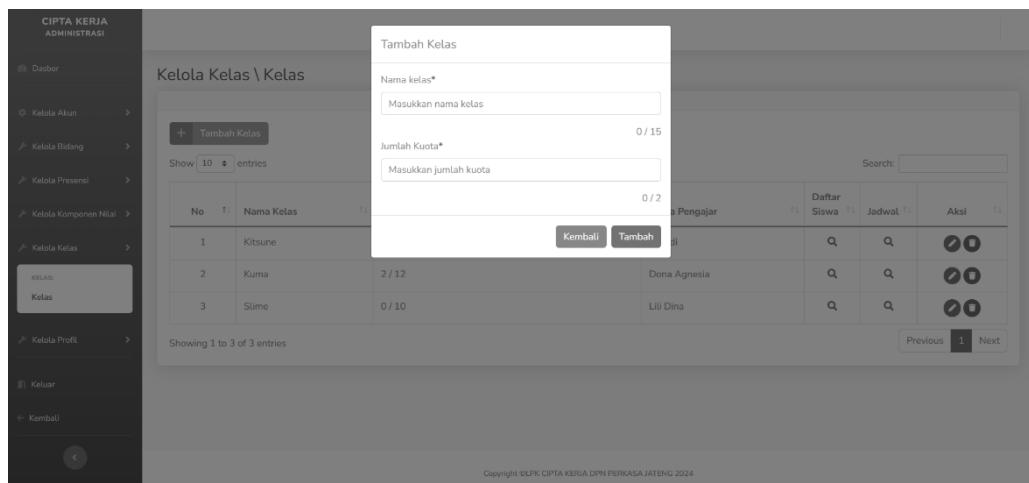
Berikut merupakan desain GUI untuk melakukan kelola kelas seperti menampilkan halaman kelola kelas, serta menambah, mengubah dan menghapus kelas yang dapat dilihat pada Gambar 4. 36, Gambar 4. 37, Gambar 4. 38, Gambar 4. 39 dan Gambar 4. 40.



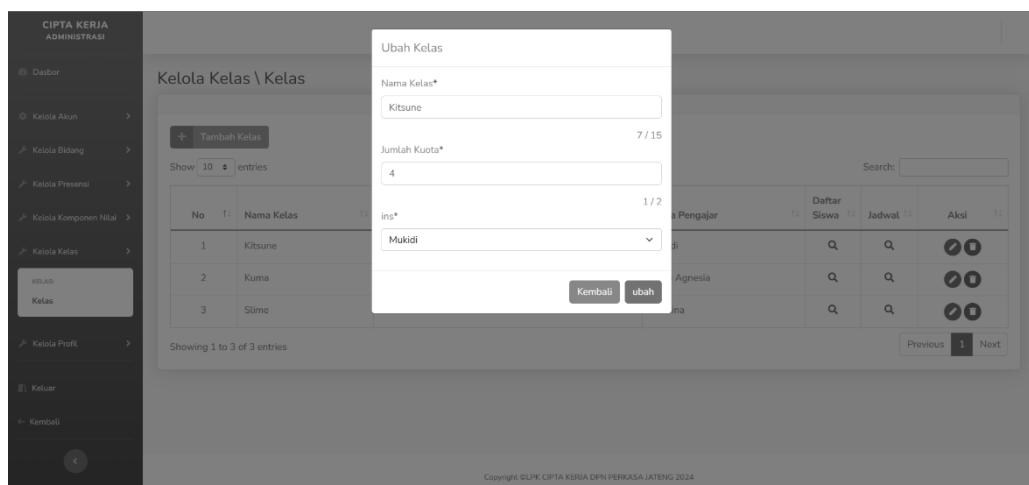
Gambar 4. 37 Desain Halaman Kelola Kelas



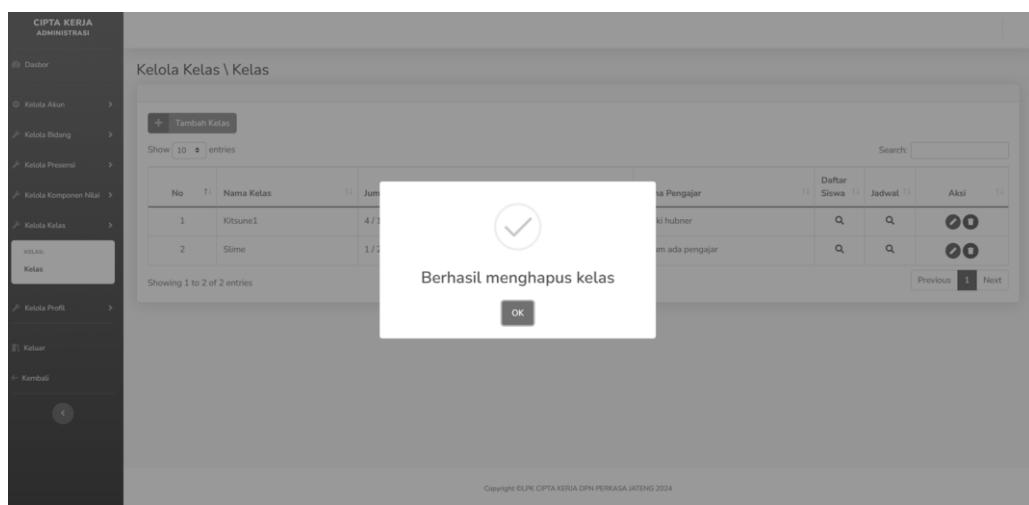
Gambar 4. 38 Desain Halaman Detail Kelas



Gambar 4. 39 Desain Modal Tambah Kelas



Gambar 4. 40 Desain Halaman Ubah Kelas



Gambar 4. 41 Desain Halaman Kelola Kelas

7. Desain Halaman Kelola Profil

Berikut merupakan desain GUI untuk melakukan kelola profil seperti menampilkan halaman kelola profil, serta menambah, mengubah dan menghapus profil yang dapat dilihat pada Gambar 4. 41, Gambar 4. 41, dan Gambar 4. 43.

The screenshot shows a table titled "Data Siswa" with the following data:

No	Nama	Email	No HP	Alamat	Kelas	Bidang	Tgl Masuk	Tgl Lulus	Status	Aksi	
1	Wahyu Widodo	siswa1@siswa.com	088903129894	Jl. Tembalang Selatan VI No. 25	Kitsune	Perhotelan	5 Feb 2024	-	Aktif		
2	siswa1	siswa2@siswa.com	98903212989	Houston	Kitsune	Pertanian	5 Mei 2024	5 Mei 2025	Lulus	-	
3	adadwas	shidkigaming7@gmail.com	989031292389	adwda	Kitsune	Pertanian	5 Mei 2024	5 Juli 2024	Lulus	-	
4	Gilang	siswa5@siswa.com	098876586756	Jl. Widuri No. 90	Kitsune	Restoran	5 Feb 2024	-	Aktif		
5	Sinta Mariska	siswa3@siswa.com	081921216675	Jl. Tembalang Utara VI No. 25	Kuma	Perhotelan	5 Mei 2024	-	Aktif		
6	Tyas Puji	siswa4@siswa.com	098876545578	Jl. Tembalang Selatan	Kuma	Pengolahan Makanan	5 Mei 2024	-	Aktif		

Gambar 4. 42 Desain Halaman Kelola Profil

The modal window "Tambah Siswa" contains the following fields:

- Nama Siswa*
- Email Siswa*
- No HP*
- Alamat*
- Bidang Minat*
- Pilih Bidang
- Kelas*
- Kelas Siswa

Gambar 4. 43 Desain modal tambah siswa

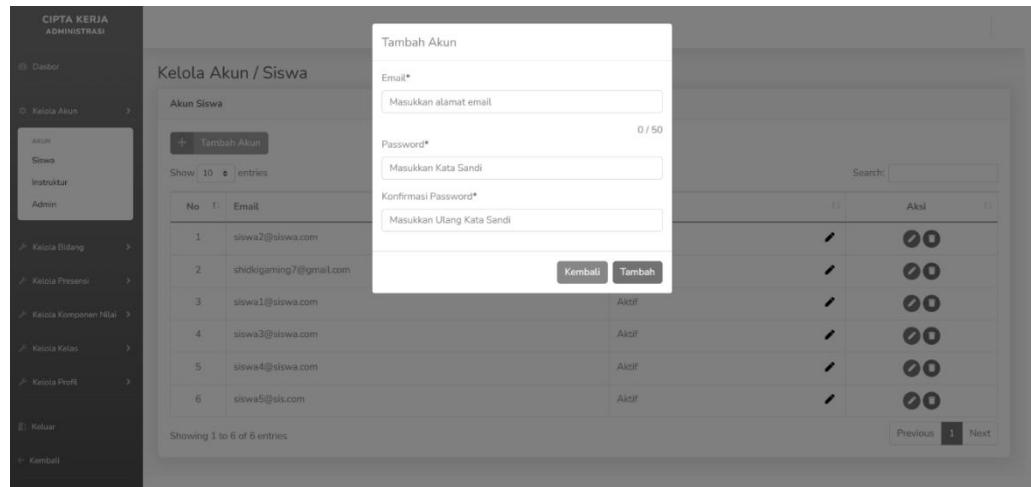
Gambar 4. 44 modal tambah siswa

8. Desain Halaman Kelola Akun

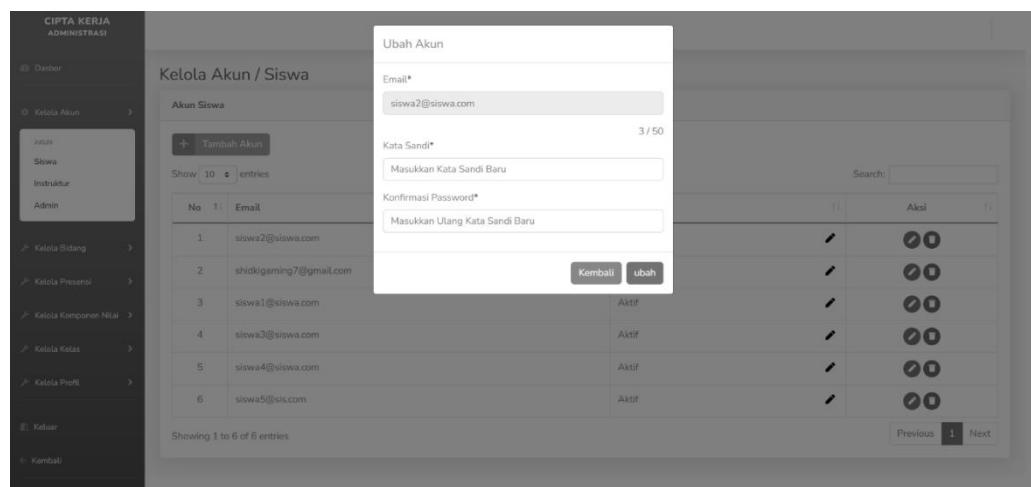
Berikut merupakan desain GUI untuk melakukan kelola akun seperti menampilkan halaman kelola akun, serta menambah, mengubah dan menghapus akun yang dapat dilihat pada Gambar 4. 44, Gambar 4. 45, Gambar 4.46 dan Gambar 4. 47.

No	Email	Status Akun	Aksi
1	siswa2@siswa.com	Tidak Aktif	
2	shidigaming7@gmail.com	Tidak Aktif	
3	siswa1@siswa.com	Aktif	
4	siswa3@siswa.com	Aktif	
5	siswa4@siswa.com	Aktif	
6	siswa5@sis.com	Aktif	

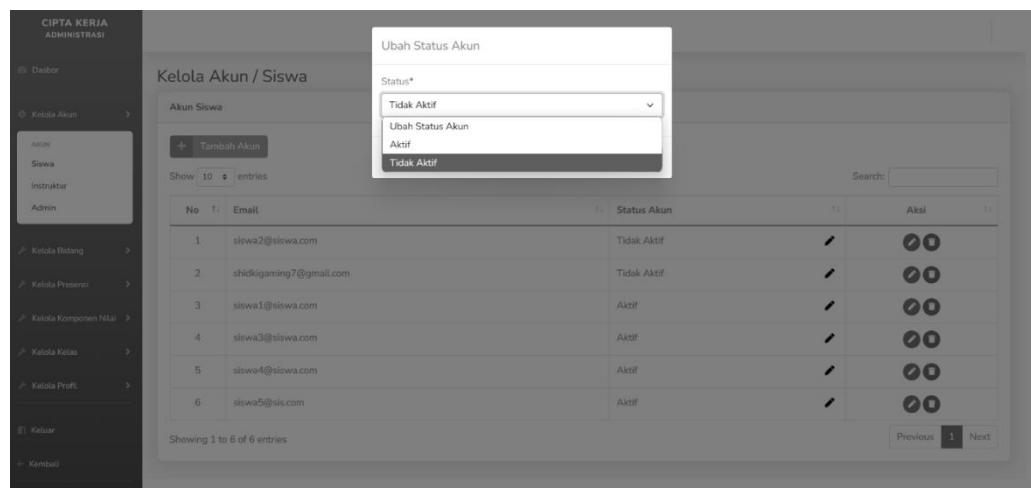
Gambar 4. 45 Desain Halaman Kelola Akun



Gambar 4. 46 Desain Halaman Tambah Akun



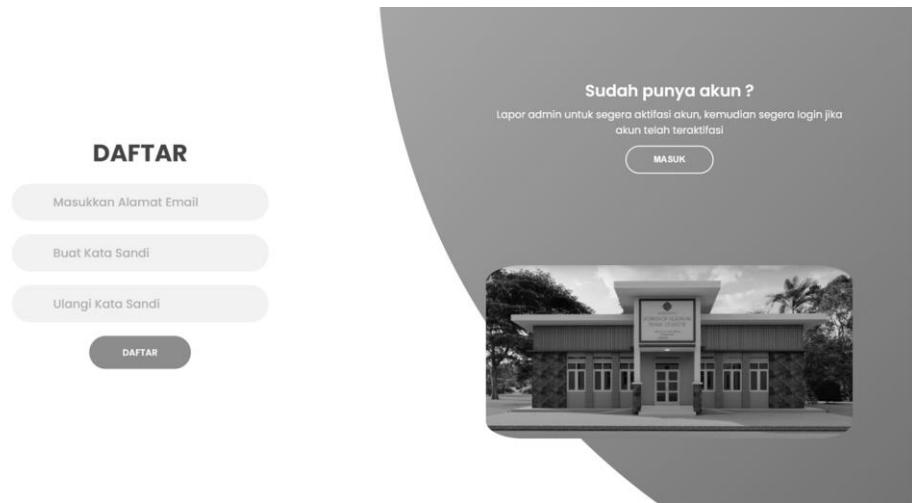
Gambar 4. 47 Desain Modal Ubah akun



Gambar 4. 48 Desain dropdown status

9. Desain Halaman Daftar Akun

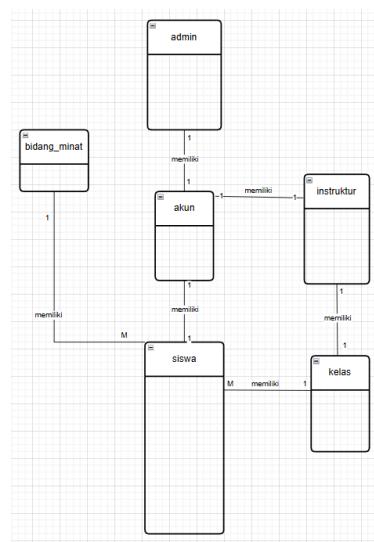
Berikut merupakan desain GUI untuk melakukan Daftarkan akun dapat dilihat pada Gambar 4.48



Gambar 4. 49 Desain Halaman Daftar Akun

4.1.3 Domain Model

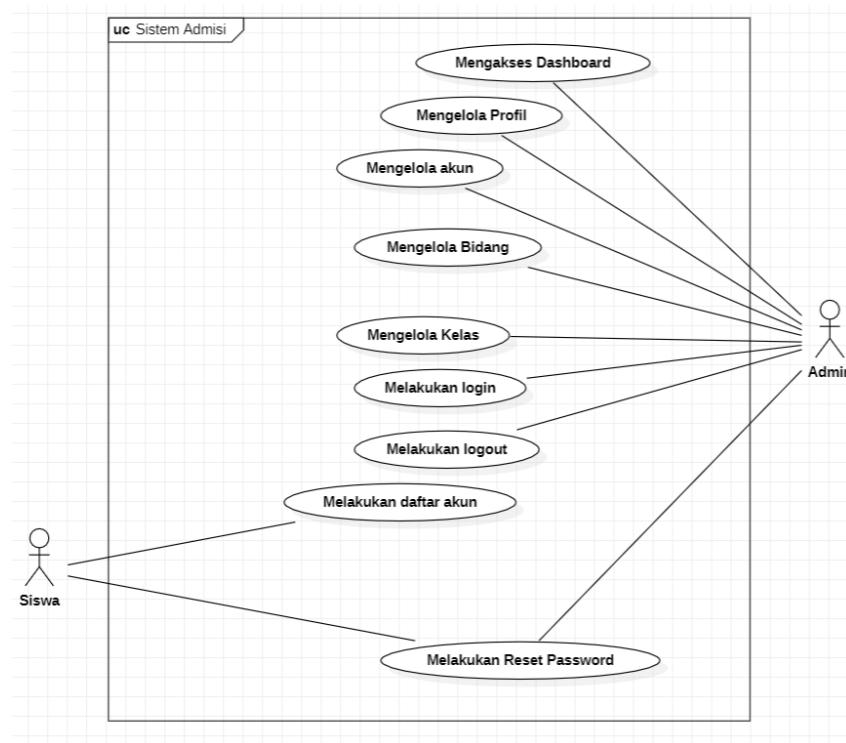
Domain model adalah alat penting dalam memahami struktur dan hubungan dalam suatu sistem. Melalui domain model, kita dapat menggambarkan elemen-elemen kunci dan keterkaitannya secara visual, yang membantu menyelaraskan pemahaman antar pemangku kepentingan. Berikut merupakan *domain diagram* dalam pengembangan aplikasi ini direpresentasikan pada Gambar 4.49



Gambar 4. 50 Domain Model

4.1.1 Use case Diagram dan Kebutuhan Aplikasi

Adapun hubungan aktor dengan sistem yang juga merupakan representasi dari kebutuhan aplikasi ini adalah dapat dilihat pada Gambar 4.29.



Gambar 4. 51 Usecase Diagram

Kebutuhan fungsional dari Aplikasi Sistem Manajemen Administrasi LPK Cipta Kerja DPN Perkasa Jateng dapat dilihat dalam representasi *use case* untuk kebutuhan fungsional dan non fungsional dideskripsikan dalam Tabel 4.1.

Tabel 4. 1 Kebutuhan Fungsional

No.	Use Case	Deskripsi	Aktor
1.	Melakukan Login	Activity Diagram untuk login.	Admin
2.	Melakukan Logout	Activity Diagram untuk logout yang dapat diakses oleh admin.	Admin
3.	Mengakses dashboard	Activity Diagram Mengakses Dashboard	Admin
4.	Mengelola kelas	Activity Diagram untuk kelola data kelas (tambah, edit dan hapus data kelas).	Admin
5.	Mengelola bidang	Activity Diagram untuk kelola data bidang (tambah, edit dan hapus data bidang).	Admin

No.	Use Case	Deskripsi	Aktor
6.	Mengelola profil	Activity Diagram untuk kelola data profil (tambah, edit dan hapus data kelas).	Admin
7.	Mengelola akun	Activity Diagram untuk kelola akun (tambah, edit dan hapus data kelas).	Admin
8.	Melakukan daftar akun	Activity Diagram Melakukan daftar akun	Siswa
9.	Melakukan reser password	Activity Diagram Melakukan Reset Password	Admin

Aplikasi Sistem Manajemen Administrasi LPK Cipta Kerja DPN Perkasa Jateng memiliki kebutuhan non fungsional sebagai fungsi yang mendukung fungsi utama yang dijelaskan pada Tabel 4.2.

Tabel 4. 2 Kebutuhan Non Fungsional

No.	Deskripsi
1.	Aplikasi harus kompatibel dengan berbagai perangkat dan sistem operasi.
2.	Aplikasi <i>web</i> harus responsif dan mampu memberikan respon yang cepat dan konsisten untuk pengguna, bahkan pada saat intensitas penggunaan yang tinggi.
3.	Aplikasi harus memiliki antarmuka yang sederhana, intuitif dan mudah digunakan oleh berbagai tingkat pengguna.
4.	Aplikasi web harus dapat dikembangkan lagi sesuai dengan perubahan dan peningkatan jumlah pengguna.

4.2 Tahap Analysis/Preliminary Design

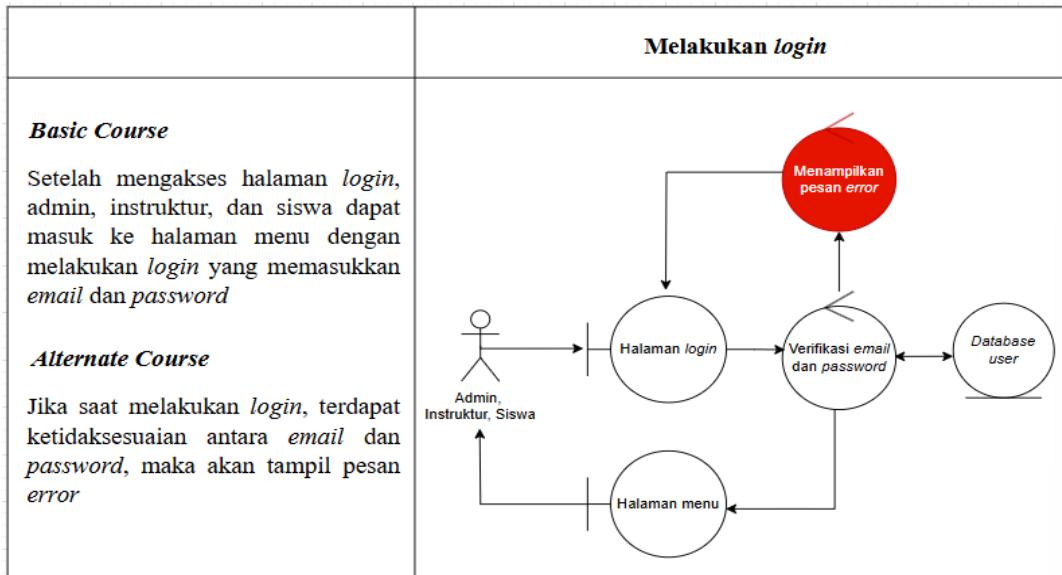
Proses yang dilakukan pada tahap ini yaitu pembuatan *robustness diagram* dan *update domain model* sebagai pembaruan dari domain model yang telah dilakukan di tahap sebelumnya.

4.2.1. Robustness Diagram

Berikut merupakan *robustness diagram* sebagai analisis yang berfungsi untuk menemukan objek-objek baru yang belum teridentifikasi sebelumnya.

1. Robustness Diagram melakukan login

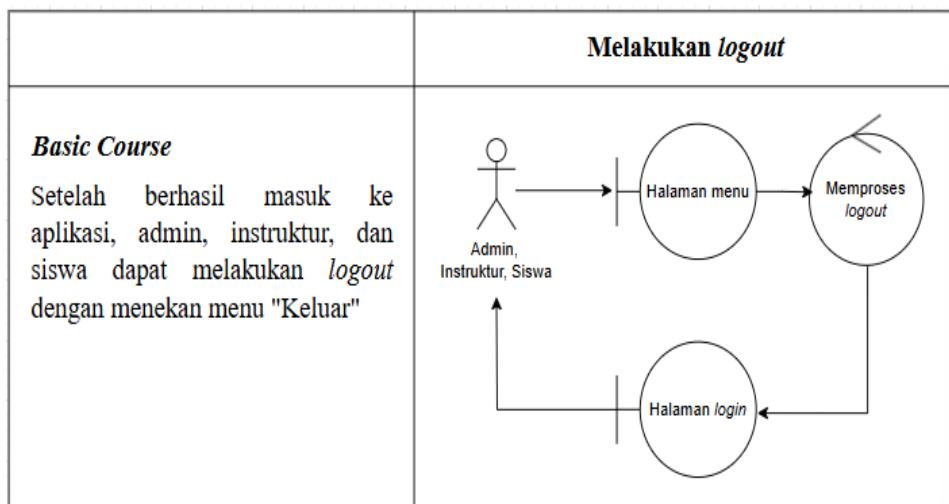
Aktor yang memiliki akun seperti admin, siswa dan instruktur dapat melakukan login untuk mengakses portal akademik LPK yang di gambarkan dalam robustness diagram pada Gambar 4.51.



Gambar 4. 52 Robustness Diagram melakukan login

2. Robustness Diagram melakukan logout

Setelah login, pengguna juga dapat melakukan log out untuk mengakhiri sesi akses terhadap halaman menu. Robustness Diagram logout untuk melakukan logout dapat dilihat pada Gambar 4.52.



Gambar 4. 53 Robustness Diagram melakukan login

3. Robustness Diagram menampilkan dashboard admin

Setelah login, admin dapat mengakses halaman dashboard secara langsung. Robustness Diagram untuk melihat dashbord admin dapat dilihat pada Gambar 4.53

	Menampilkan Dashboard Admin
<p>Basic Course</p> <p>Setelah melakukan login admin dapat menhaskses fitur dashboard amin dengan masuk ke menu Administrasi dan sistem akan menampilkan dashboard admin</p>	<pre> graph LR Admin((Admin)) --> HalamanMenu([Halaman menu]) HalamanMenu --> Menangani[Menangani permintaan administrasi] Menangani --> DashboardAdmin([Dashboard admin]) </pre>

Gambar 4. 54 Robustness Diagram menampilkan dashboard admin

4. Robustness Diagram Mengelola Bidang

Pada fitur kelola bidang, admin dapat menampilkan data bidang dengan proses pada Robustness Diagram pada Gambar 4.54.

	Menampilkan bidang
<p>Basic Course</p> <p>Setelah melakukan login admin memilih menu kelola bidang kemudian memilih bidang dan menampilkan kelola bidang.</p>	<pre> graph LR Admin((Admin)) --> Dashboard([Dashboard]) Dashboard --> Menangani[Menangani permintaan kelola bidang] Menangani --> HalamanKelolaBidang([Halaman kelola bidang]) </pre>

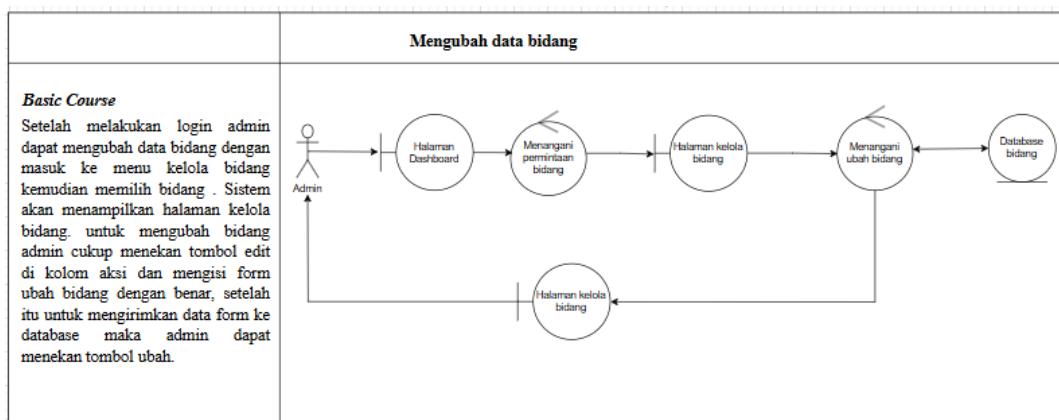
Gambar 4. 55 Robustness Diagram menampilkan dat bidang

Pada fitur kelola bidang, admin dapat menambahkan data bidang dengan proses pada Robustness Diagram pada Gambar 4.55.

	Menambah data bidang
<p>Basic Course</p> <p>Setelah melakukan login admin, dapat menambahkan data bidang dengan masuk ke menu kelola bidang, kemudian memilih bidang. Pada halaman kelola bidang tekan tombol tambah bidang kemudian mengisi form tambah bidang dan menekan tombol tambah untuk menambah data baru.</p>	<pre> graph LR Admin((Admin)) --> HalamanDashboard([Halaman Dashboard]) HalamanDashboard --> Menangani[Menangani permintaan bidang] Menangani --> HalamanKelolaBidang([Halaman kelola bidang]) HalamanKelolaBidang --> MenanganiTambahBidang[Menangani tambah bidang] MenanganiTambahBidang <--> DatabaseBidang([Database Bidang]) DatabaseBidang --> HalamanKelolaBidang </pre>

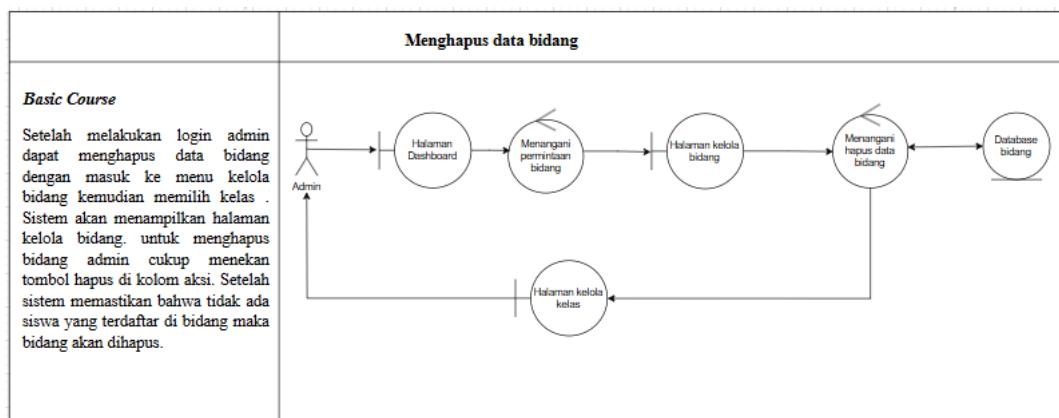
Gambar 4. 56 Robustness Diagram menambah data bidang

Pada fitur kelola bidang, admin dapat mengubah data bidang dengan proses pada Robustness Diagram pada Gambar 4.56.



Gambar 4. 57 Robustness Diagram mengubah data bidang

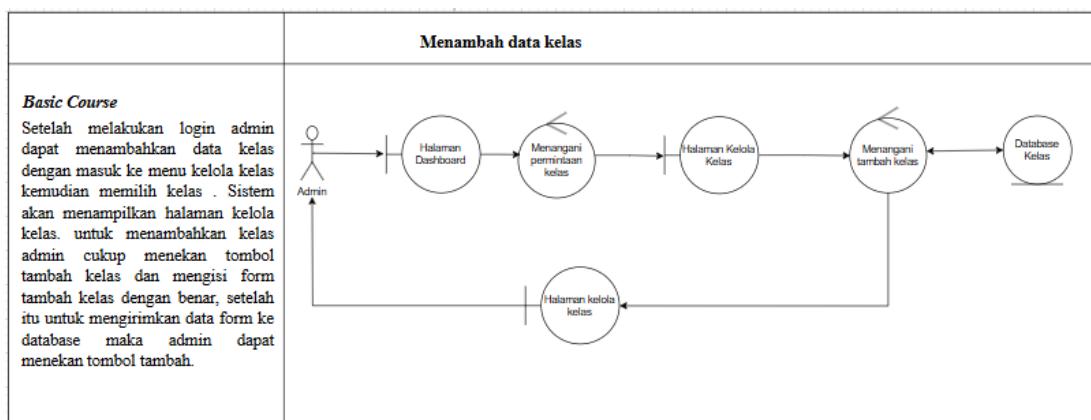
Pada fitur kelola bidang, admin dapat menghapus data bidang dengan proses pada Robustness Diagram pada Gambar 4.57.



Gambar 4. 58 Robustness Diagram mengubah data bidang

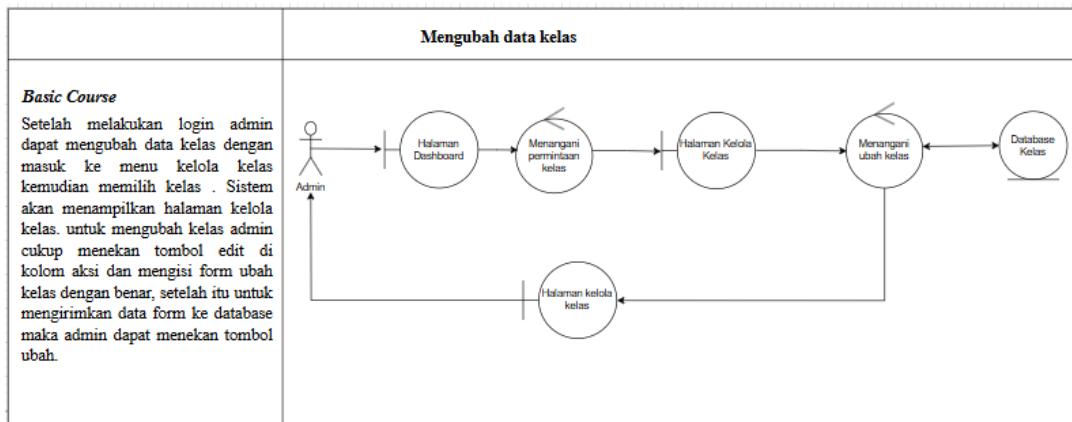
5. Robustness Diagram Mengelola Kelas

Pada fitur kelola kelas, admin dapat menambahkan data kelas dengan proses pada Robustness Diagram pada Gambar 4.58.



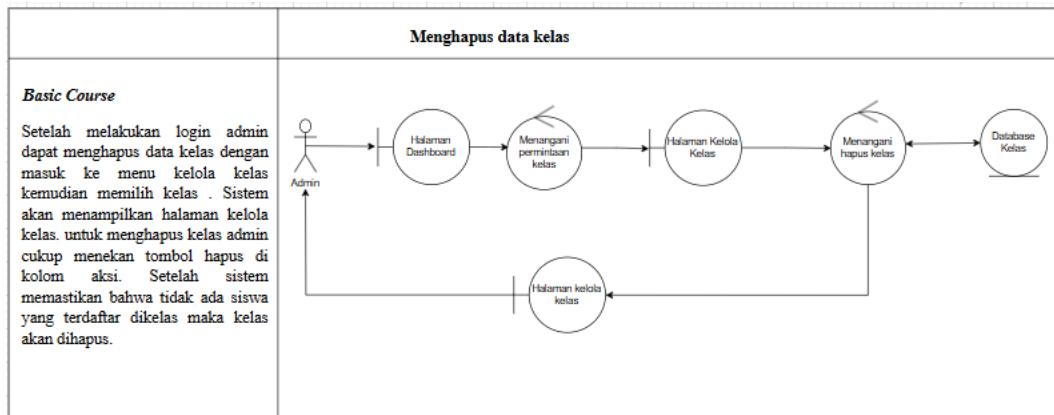
Gambar 4. 59 Robustness Diagram menambah data kelas

Pada fitur kelola kelas, admin dapat mengubah data kelas dengan proses pada Robustness Diagram pada Gambar 4.59.



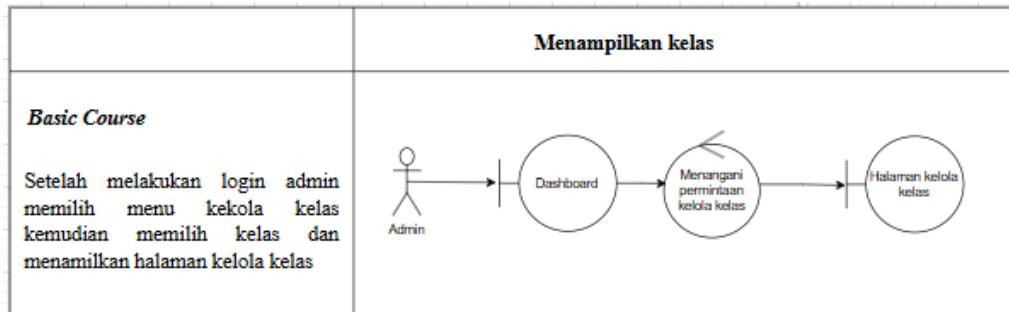
Gambar 4. 60 Robustness Diagram mengubah data kelas

Pada fitur kelola kelas, admin dapat menghapus data kelas dengan proses pada Robustness Diagram pada Gambar 4.60.



Gambar 4. 61 Robustness Diagram menghapus data kelas

Pada fitur kelola kelas, admin dapat menampilkan data kelas dengan proses pada Robustness Diagram pada Gambar 4.61.



Gambar 4. 62 Robustness Diagram menampilkan data kelas

6. Robustness Diagram mengelola akun

Pada fitur kelola akun, admin dapat menampilkan data akun dengan proses pada Robustness Diagram pada Gambar 4.62.

	Menampilkan Akun
Basic Course Setelah melakukan login admin memilih menu kelola profil memilih role yang diinginkan dan menampilkan halaman kelola akun sesuai role yang dipilih.	<pre> graph LR Admin((Admin)) --> Dashboard([Dashboard]) Dashboard --> Menangani[Menangani permintaan kelola akun] Menangani --> Halaman([Halaman kelola akun]) Halaman --> Menangani </pre>

Gambar 4. 63 Robustness Diagram menampilkan data akun

Pada fitur kelola akun, admin dapat menambahkan data akun dengan proses pada Robustness Diagram pada Gambar 4.63.

	Menambah akun siswa, instruktur dan admin
Basic Course Setelah melakukan login admin dapat menambahkan akun siswa, instruktur dan admin dengan masuk ke kelola akun memilih tambah akun di masing-masing role kemudian mengisi form dan menambahkan data form ke database dengan menekan tombol tambah.	<pre> graph LR Admin((Admin)) --> HalamanDashboard([Halaman Dashboard]) HalamanDashboard --> Menangani[Menangani permintaan kelola akun] Menangani --> Halamankelola([Halaman kelola akun]) Halamankelola --> Menangani Halamankelola --> Menangantambah[Menangani tambah akun] Menangantambah --> Database([Database akun]) Database --> Menangantambah </pre>

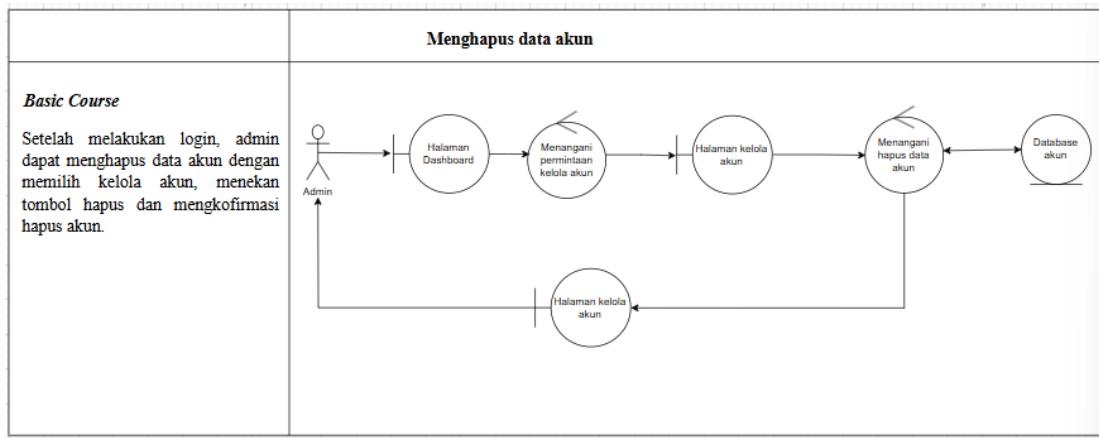
Gambar 4. 64 Robustness Diagram menambah data akun

Pada fitur kelola akun, admin dapat mengubah data akun dengan proses pada Robustness Diagram pada Gambar 4.64.

	Mengubah data akun
Basic Course Setelah melakukan login, admin dapat mengubah akun siswa, instruktur dan admin dengan masuk ke kelola akun menekan tombol edit di kolom aksi, megisi form ubah data, menekan tombol ubah.	<pre> graph LR Admin((Admin)) --> HalamanDashboard([Halaman Dashboard]) HalamanDashboard --> Menangani[Menangani permintaan ke kelola akun] Menangani --> Halamankelola([Halaman kelola akun]) Halamankelola --> Menangani Halamankelola --> Menanganiubah[Menangani ubah akun] Menanganiubah --> Database([Database akun]) Database --> Menanganiubah </pre>

Gambar 4. 65 Robustness Diagram mengubah data akun

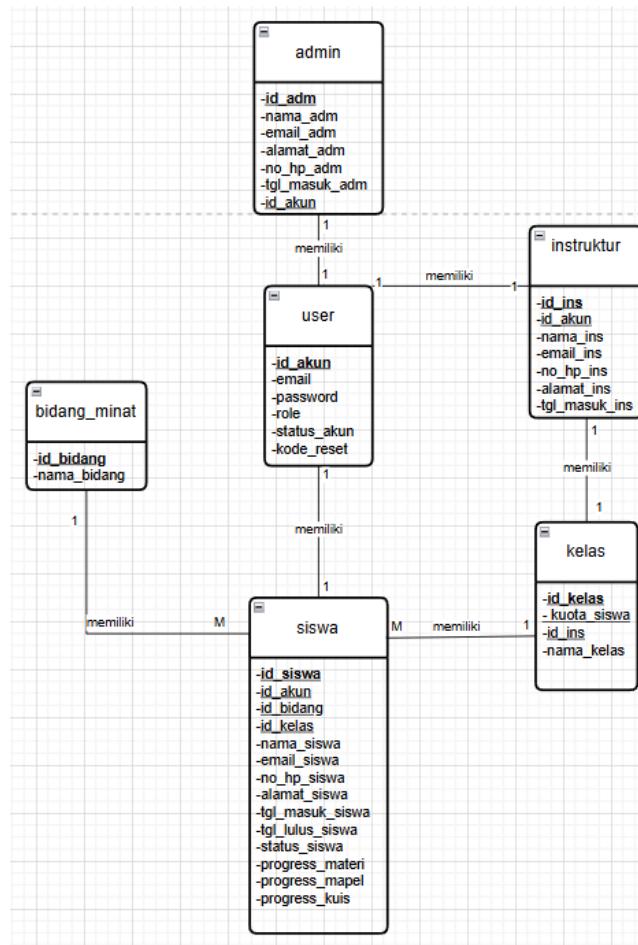
Pada fitur kelola akun, admin dapat menghapus data akun dengan proses pada Robustness Diagram pada Gambar 4.65.



Gambar 4. 66 Robustness Diagram menghapus data akun

4.2.2. Update Domain Model

Berikut merupakan *update domain model* sebagai pembaruan kelas-kelas dan interaksinya disertai penambahan atribut yang terdapat pada Gambar.



Gambar 4. 67 *Update Domain Model*

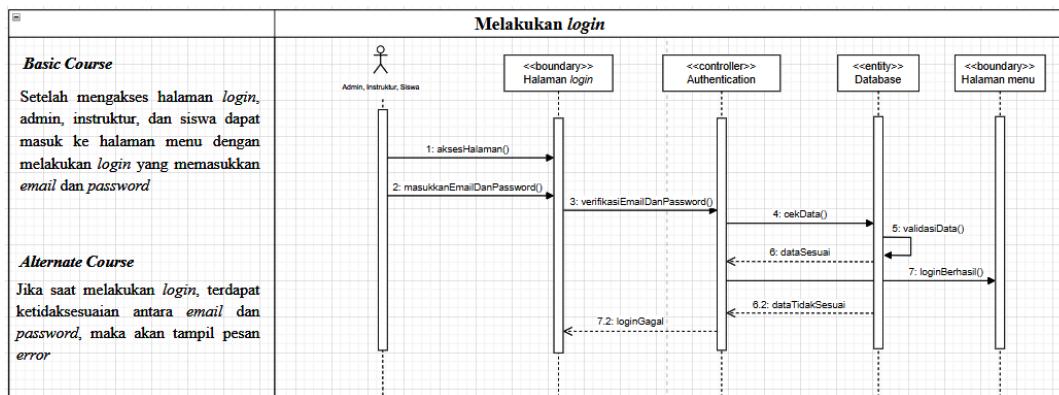
4.3 Tahap Detailed Design

4.3.1 Sequence Diagram

Sequence diagram adalah diagram interaksi yang memvisualisasikan urutan pesan yang dikirim antara objek-objek dalam sistem serta kapan pesan tersebut dikirim. Ini memberikan gambaran tentang bagaimana objek-objek berinteraksi satu sama lain dalam suatu skenario atau proses tertentu. *Sequence* diagram untuk Aplikasi Sistem Manajemen Administrasi LPK Cipta Kerja DPN Perkasa Jateng adalah sebagai berikut.

1. *Sequence* diagram mengakses halaman login

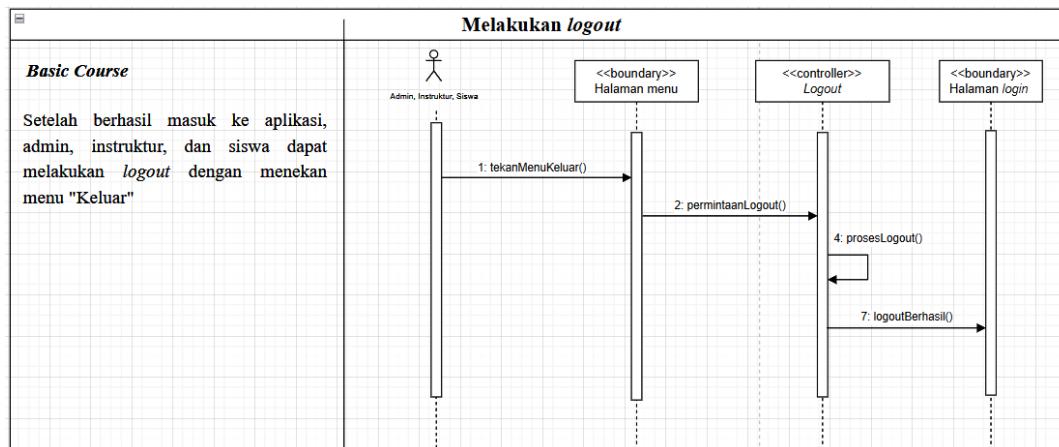
Sequence diagram berikut menunjukkan alur login yang mencakup proses autentikasi untuk admin, instruktur, dan siswa pada Gambar.



Gambar 4. 68 Sequence diagram mengakses halaman login

2. Sequence diagram melakukan logout

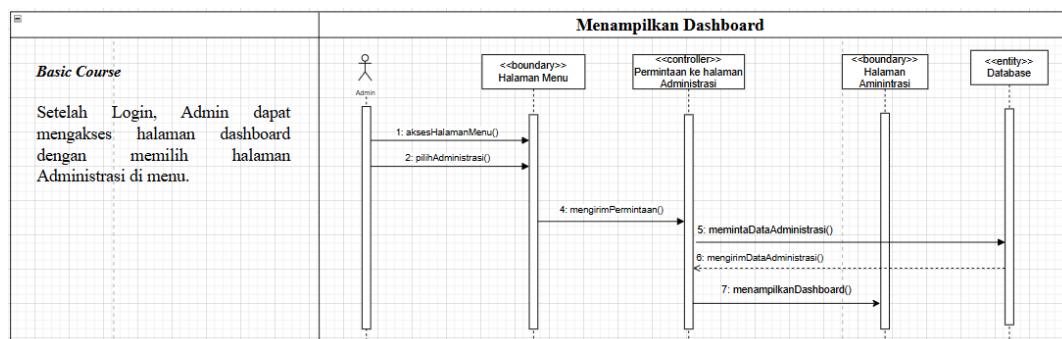
Sequence diagram berikut menunjukkan alur logout yang dapat diakses oleh admin, instruktur, dan siswa yang dilihat pada Gambar.



Gambar 4. 69 Sequence diagram melakukan logout

3. Sequence diagram mengakses dashboard

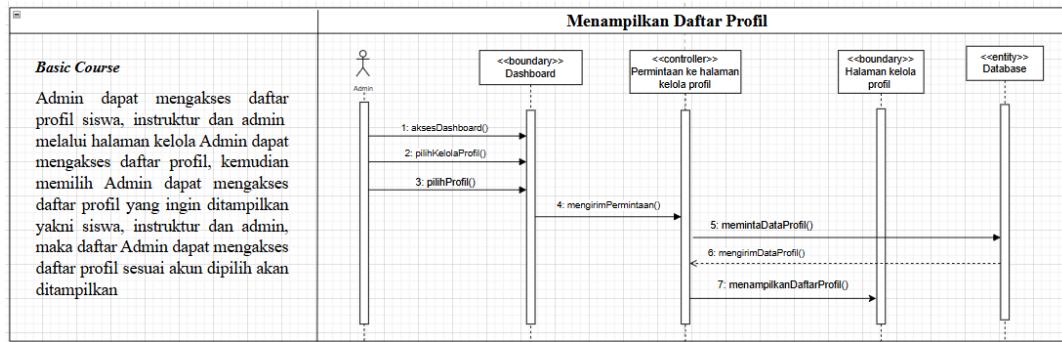
Sequence diagram berikut menunjukkan alur mengakses dashboard yang dapat diakses oleh admin yang dilihat pada Gambar 4.69.



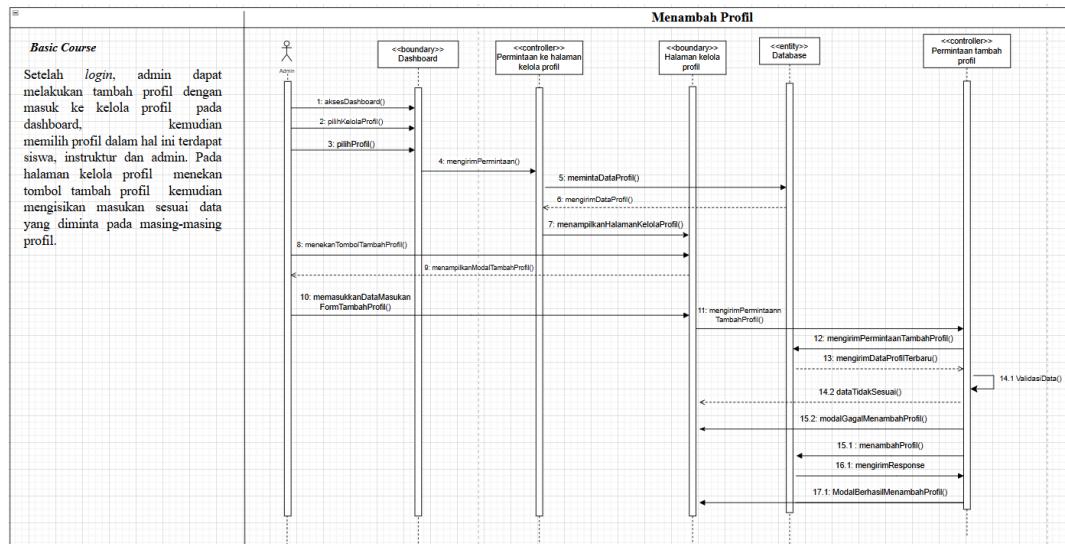
Gambar 4. 70 Sequence diagram mengakses dashboard

4. Sequence diagram mengelola profil

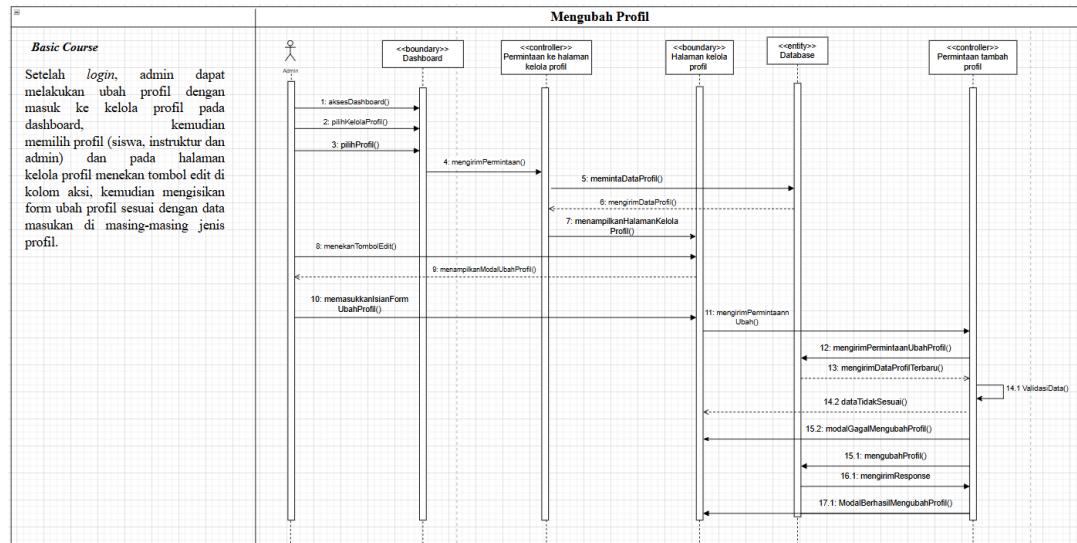
Sequence diagram berikut menunjukkan berbagai alur pengelolaan profil yang dapat diakses oleh admin yang dilihat pada Gambar 4.70, Gambar 4.71, Gambar 4.72, dan Gambar 4.73.



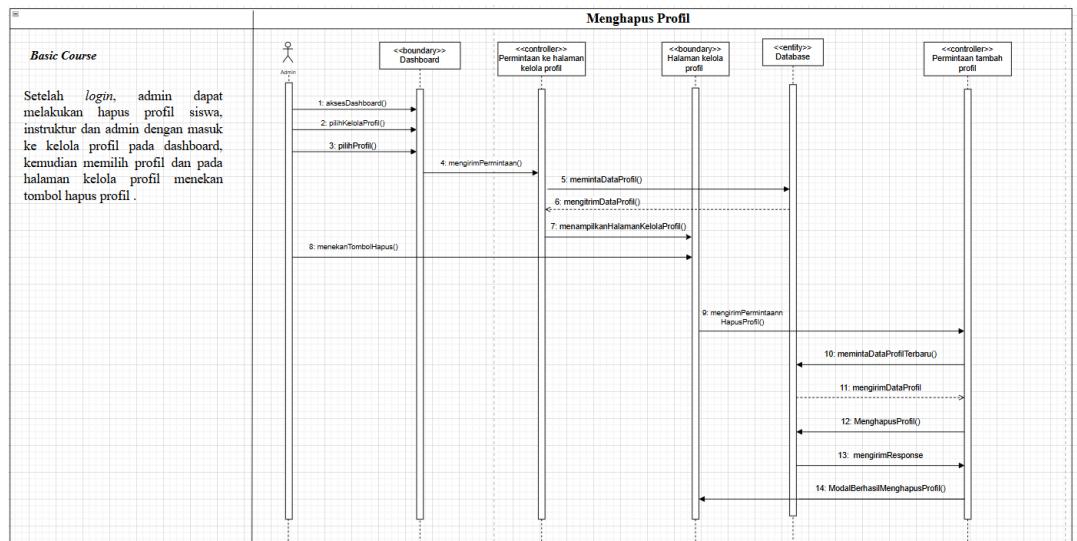
Gambar 4. 71 Sequence diagram menampilkan daftar profil



Gambar 4. 72 Sequence diagram menambah profil



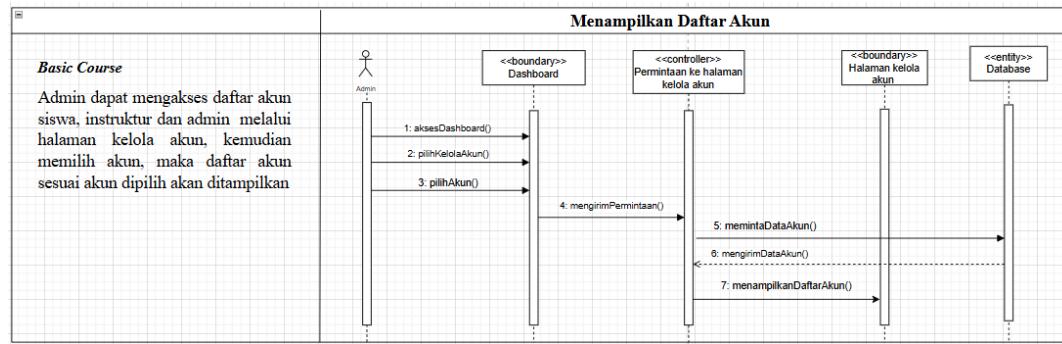
Gambar 4. 73 Sequence diagram mengubah profil



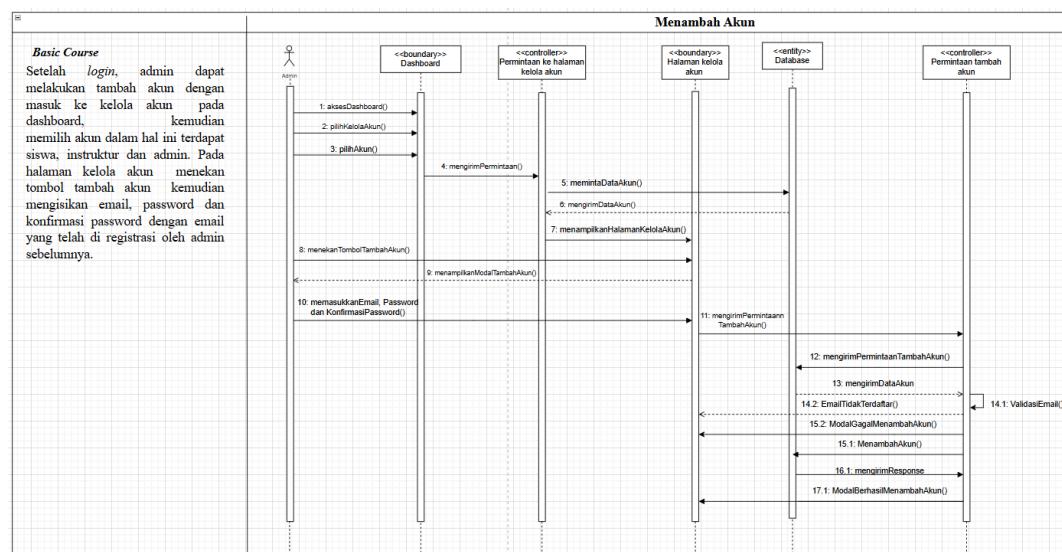
Gambar 4. 74 Sequence diagram menghapus profil

5. Sequence diagram mengelola akun

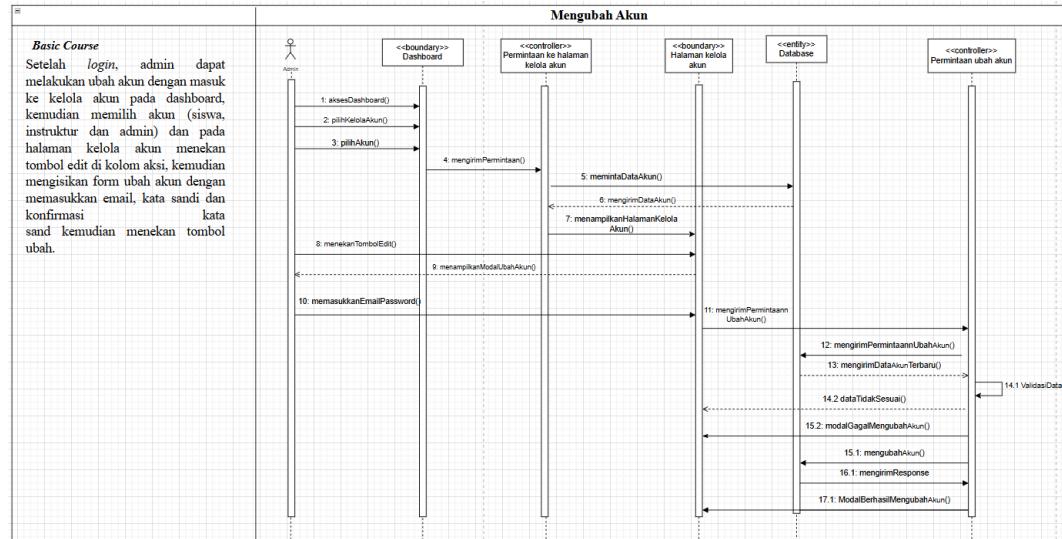
Sequence diagram berikut menunjukkan berbagai alur pengelolaan akun yang dapat diakses oleh admin yang dilihat pada Gambar 4.74, Gambar 4.75, Gambar 4.76, Gambar 4.77 dan Gambar 4.78.



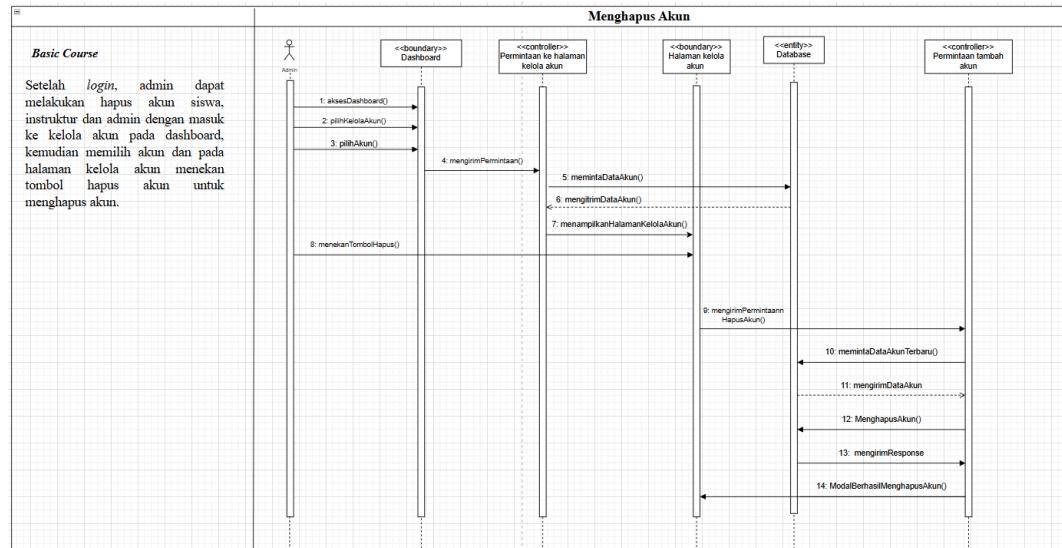
Gambar 4. 75 Sequence diagram menampilkan daftar akun



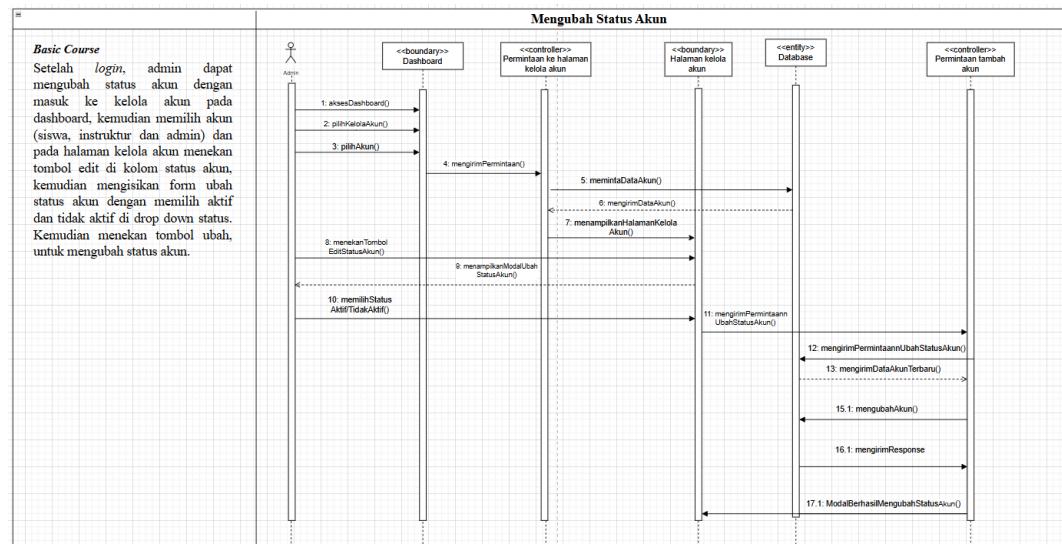
Gambar 4. 76 Sequence diagram menambah akun



Gambar 4. 77 Sequence diagram mengubah akun



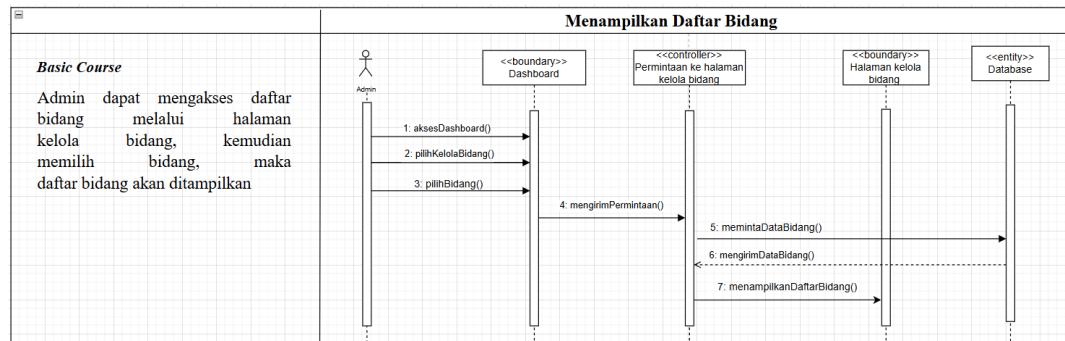
Gambar 4. 78 Sequence diagram menghapus akun



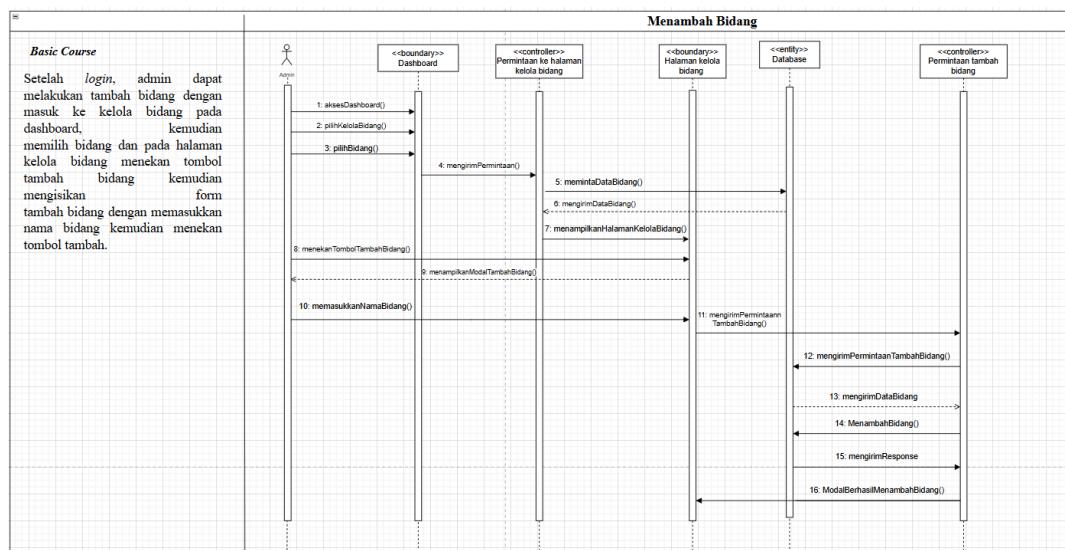
Gambar 4. 79 Sequence diagram mengubah status akun

6. Sequence diagram mengelola bidang

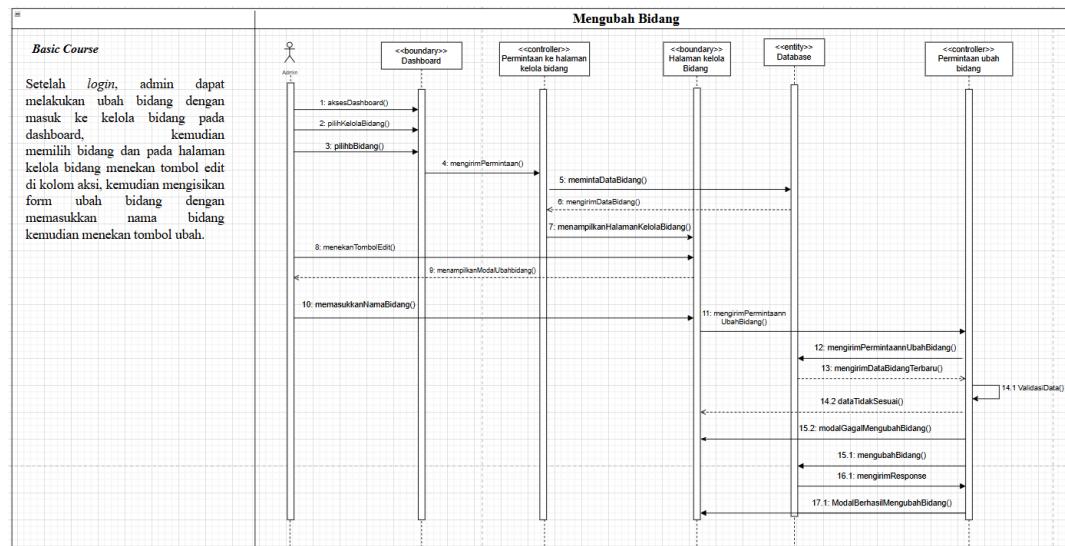
Sequence diagram berikut menunjukkan berbagai alur pengeloaan bidang yang dapat diakses oleh admin yang dilihat pada Gambar 4.79, Gambar 4.80, Gambar 4.81, dan Gambar 4.82.



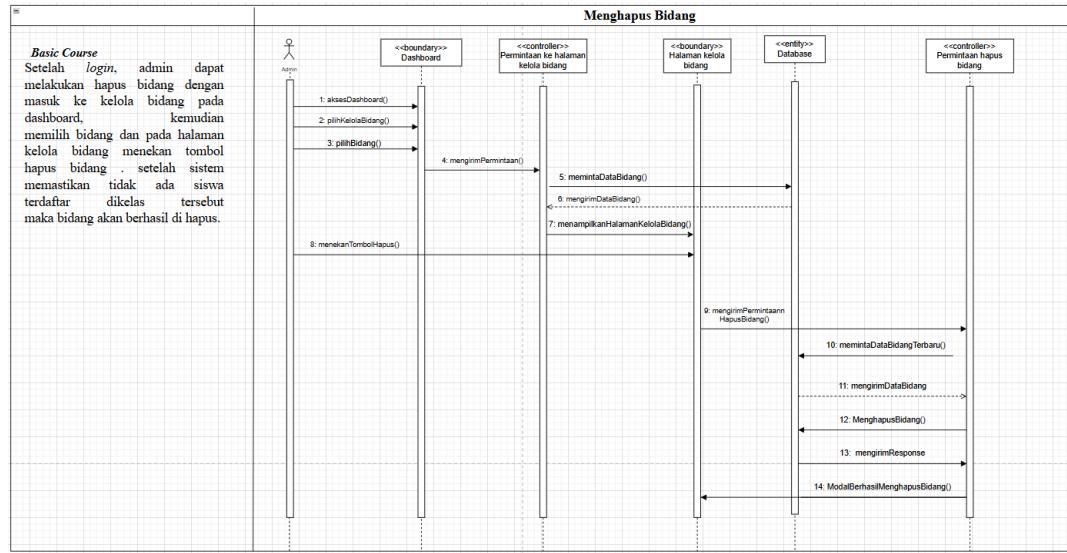
Gambar 4. 80 Sequence diagram menampilkan bidang



Gambar 4. 81 Sequence diagram menambah bidang



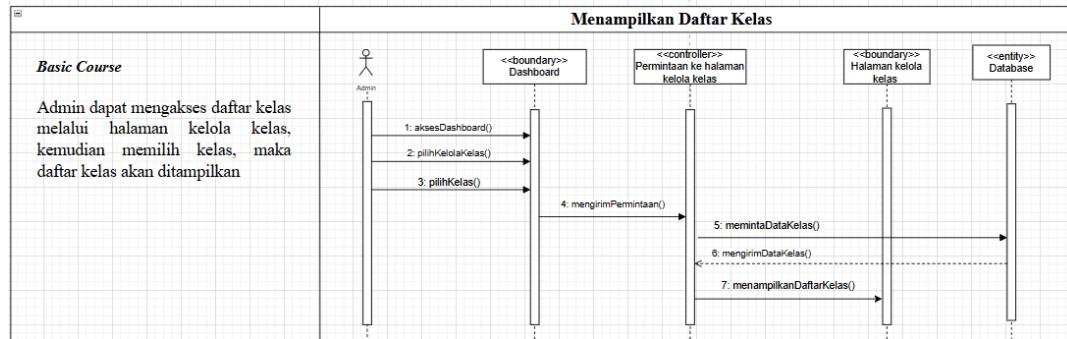
Gambar 4. 82 Sequence diagram mengubah bidang



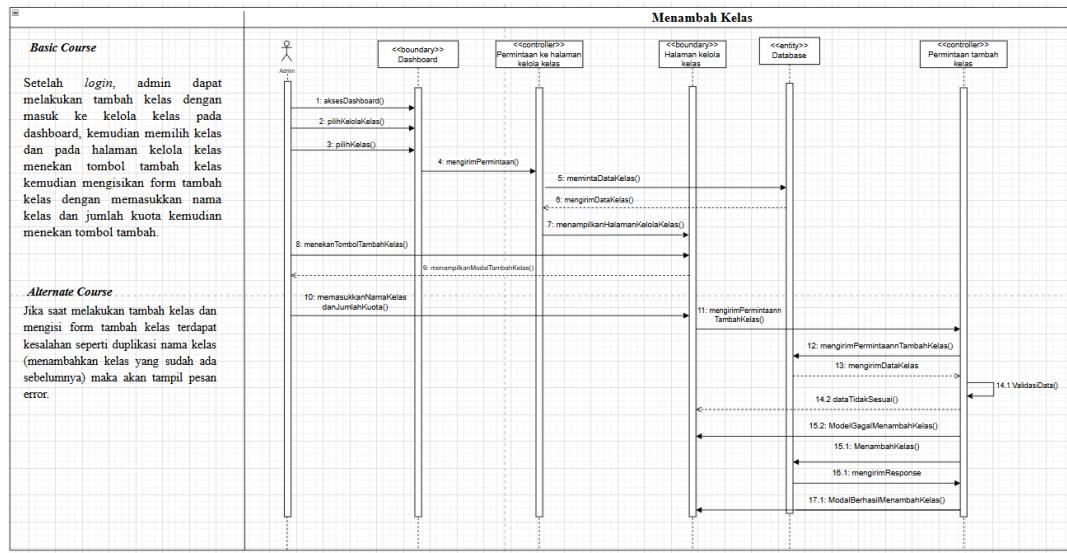
Gambar 4. 83 Sequence diagram menghapus bidan

7. Sequence diagram mengelola kelas

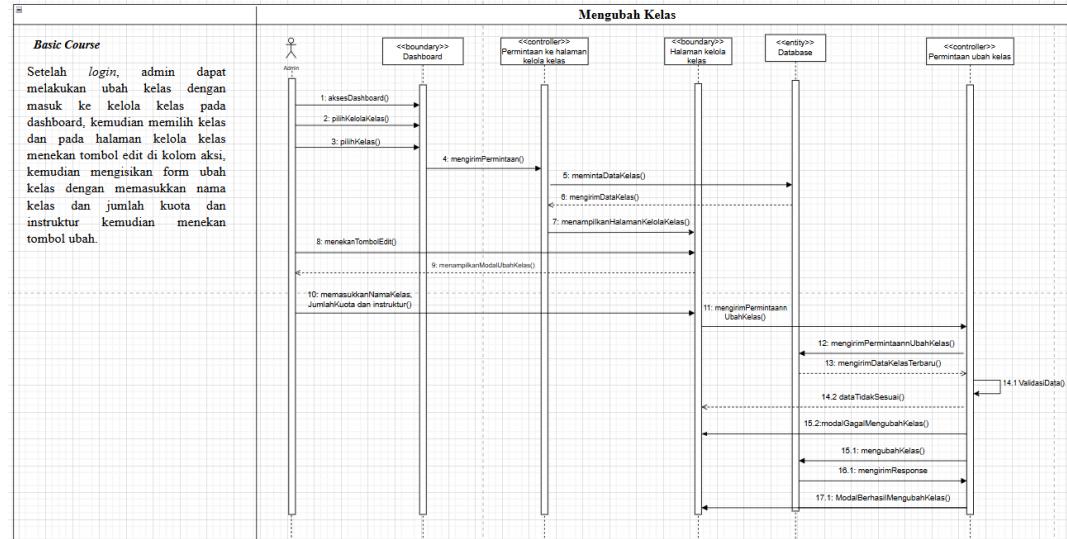
Sequence diagram berikut menunjukkan berbagai alur pengelolaan kelas yang dapat diakses oleh admin yang dilihat pada Gambar 4.83, Gambar 4.84, Gambar 4.85, dan Gambar 4.86.



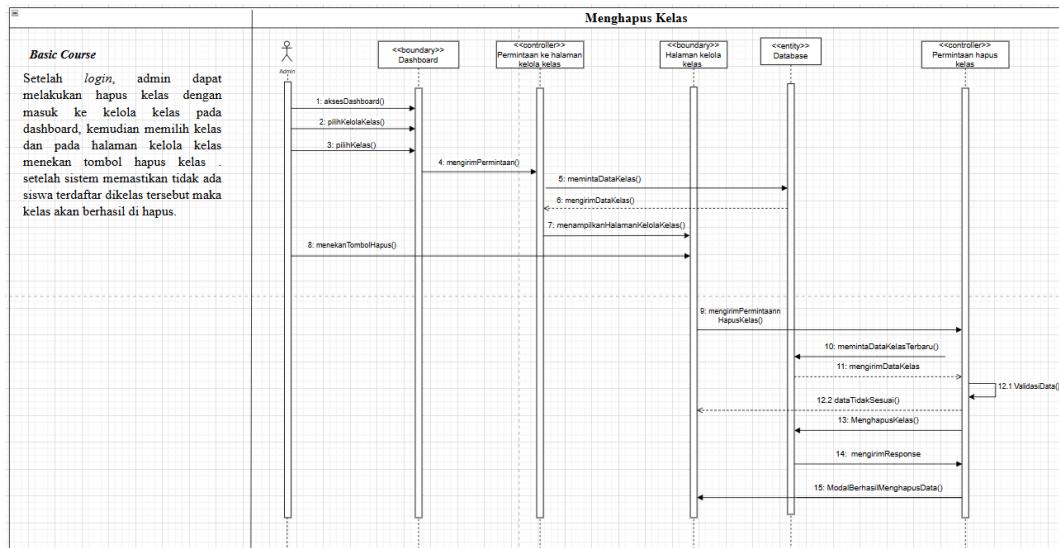
Gambar 4. 84 Sequence diagram mengelolah kelas



Gambar 4. 85 Sequence diagram menambah kelas



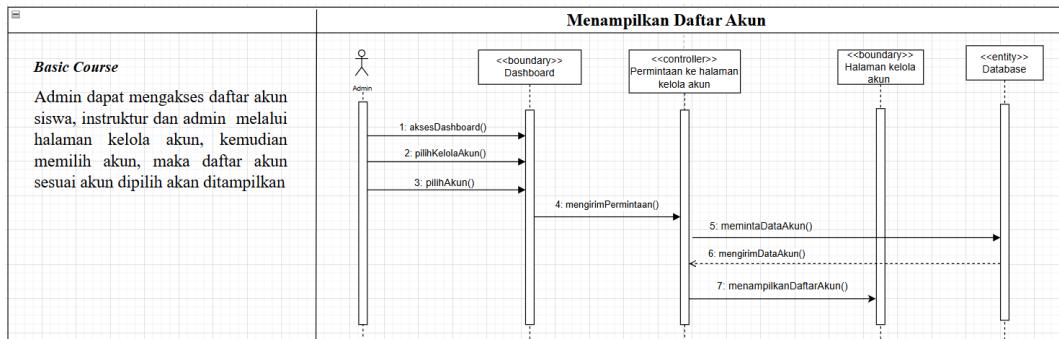
Gambar 4. 86 Sequence diagram mengubah kelas



Gambar 4. 87 Sequence diagram menghapus kelas

8. Sequence diagram melakukan daftar akun

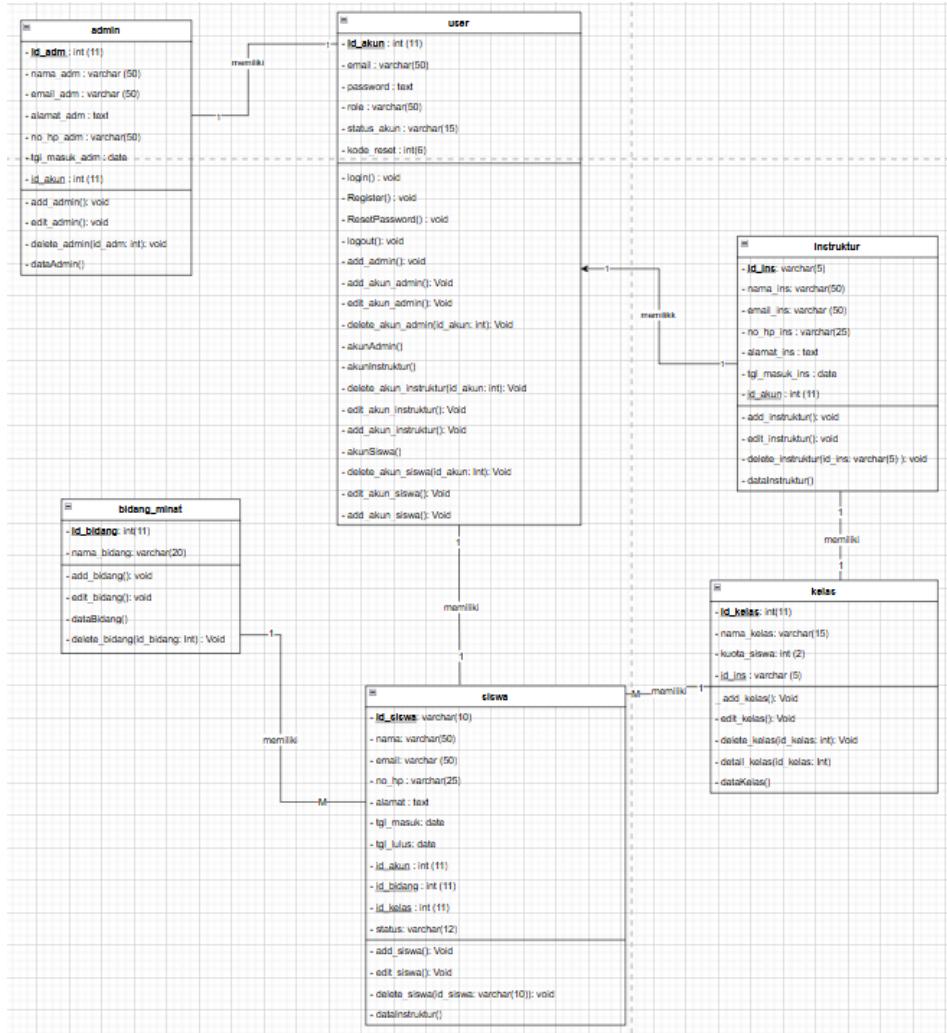
Sequence diagram berikut menunjukkan berbagai alur pengeloaan bidang yang dapat diakses oleh admin yang dilihat pada Gambar 4.87.



Gambar 4. 88 Sequence diagram melakukan daftar akun

4.3.2 Class Diagram

Diagram ini menyajikan gambaran visual tentang entitas-entitas utama dalam sistem, termasuk atribut-atribut dan metode-metode yang dimiliki oleh setiap kelas, serta hubungan dan asosiasi antara kelas-kelas tersebut. Class diagram membantu dalam memahami struktur keseluruhan sistem, serta memberikan landasan untuk implementasi kode dan pengembangan lebih lanjut. Class diagram untuk Aplikasi Sistem Manajemen Administrasi LPK Cipta Kerja DPN Perkasa Jateng digambarkan pada Gambar 4.88.



Gambar 4. 89 Class Diagram

4.4 Tahap Implementation

4.4.1 Tahap Koding

4.4.1.1 Implementasi Perangkat

Spesifikasi perangkat keras da perangkat luak yang digunakan untuk membuat Aplikasi Sistem Manajemen Administrasi LPK Cipta Kerja DPN Perkasa Jateng adalah sebagai berikut.

1. CPU : AMD Ryzen 5 5500U with Radeon Graphics 2.10 GHz
2. Sistem Operasi : Windows 11 Home Single Language
3. Bahasa Pemrograman : PHP 8.2.0, JavaScript
4. Framework : Laravel 11.9, Bootstrap

5. Database Management System : MySQL
6. Tools : Visual Studio Code
7. Web-browser : Microsoft Edge

4.4.1.2 Implementasi Basis Data

Implementasi basis data untuk Aplikasi Sistem Manajemen Administrasi LPK Cipta Kerja DPN Perkasa Jateng di buat menggunakan PHPMyAdmin. Setiap tabel dalam sistem memiliki berbagai kolom dengan nama yang telah direncanakan dan diimplementasikan sebagai berikut.

1. Tabel admin

Implementasi perancangan tabel dapat dilihat pada tabel admin pada Gambar 4.89.

#	Nama	Jenis	Penyortiran	Atribut	Tak Ternilai	Bawaan	Komentar	Ekstra	Tindakan
1	id_adm	int(11)			Tidak	Tidak ada		AUTO_INCREMENT	Ubah Hapus Lainnya
2	nama_adm	varchar(50)	utf8mb4_general_ci		Tidak	Tidak ada			Ubah Hapus Lainnya
3	email_adm	varchar(50)	utf8mb4_general_ci		Tidak	Tidak ada			Ubah Hapus Lainnya
4	alamat_adm	text	utf8mb4_general_ci		Tidak	Tidak ada			Ubah Hapus Lainnya
5	no_hp_adm	varchar(25)	utf8mb4_general_ci		Tidak	Tidak ada			Ubah Hapus Lainnya
6	tgl_masuk_adm	date			Tidak	Tidak ada			Ubah Hapus Lainnya
7	id_akun	int(11)			Ya	NULL			Ubah Hapus Lainnya

Gambar 4. 90 Tabel admin

2. Tabel user

Implementasi perancangan tabel dapat dilihat pada tabel user pada Gambar 4.90.

#	Nama	Jenis	Penyortiran	Atribut	Tak Ternilai	Bawaan	Komentar	Ekstra	Tindakan
1	id_akun	int(11)			Tidak	Tidak ada		AUTO_INCREMENT	Ubah Hapus Lainnya
2	email	varchar(50)	utf8mb4_unicode_ci		Tidak	Tidak ada			Ubah Hapus Lainnya
3	password	text	utf8mb4_unicode_ci		Tidak	Tidak ada			Ubah Hapus Lainnya
4	role	varchar(15)	utf8mb4_unicode_ci		Ya	NULL			Ubah Hapus Lainnya
5	status_akun	varchar(15)	utf8mb4_unicode_ci		Tidak	tidak_aktif			Ubah Hapus Lainnya
6	kode_reset	int(6)			Ya	NULL			Ubah Hapus Lainnya

Gambar 4. 91 Tabel user

3. Tabel siswa

Implementasi perancangan tabel dapat dilihat pada tabel siswa pada Gambar 4.91.

Nama	Jenis	Penyortiran	Atribut	Tak Ternilai	Bawaan	Komentar	Ekstra	Tindakan
id_siswa	varchar(10)	utf8mb4_general_ci		Tidak	Tidak ada		Ubah Hapus Lainnya	
nama	varchar(50)	utf8mb4_general_ci		Tidak	Tidak ada		Ubah Hapus Lainnya	
email	varchar(50)	utf8mb4_general_ci		Tidak	Tidak ada		Ubah Hapus Lainnya	
no_hp	varchar(25)	utf8mb4_general_ci		Tidak	Tidak ada		Ubah Hapus Lainnya	
alamat	text	utf8mb4_general_ci		Tidak	Tidak ada		Ubah Hapus Lainnya	
tgl_masuk	date			Tidak	Tidak ada		Ubah Hapus Lainnya	
tgl_lulus	date			Ya	NULL		Ubah Hapus Lainnya	
id_akun	int(11)			Ya	NULL		Ubah Hapus Lainnya	
id_bidang	int(11)			Tidak	Tidak ada		Ubah Hapus Lainnya	
id_kelas	int(11)			Tidak	Tidak ada		Ubah Hapus Lainnya	
status	varchar(12)	utf8mb4_general_ci		Tidak	Tidak ada		Ubah Hapus Lainnya	

Gambar 4. 92 Tabel Siswa

4. Tabel instruktur

Implementasi perancangan tabel dapat dilihat pada tabel instruktur pada Gambar 4.92.

Nama	Jenis	Penyortiran	Atribut	Tak Ternilai	Bawaan	Komentar	Ekstra	Tindakan
id_ins	varchar(5)	utf8mb4_general_ci		Tidak	Tidak ada		Ubah Hapus Lainnya	
nama_ins	varchar(50)	utf8mb4_general_ci		Tidak	Tidak ada		Ubah Hapus Lainnya	
email_ins	varchar(50)	utf8mb4_general_ci		Tidak	Tidak ada		Ubah Hapus Lainnya	
no_hp_ins	varchar(25)	utf8mb4_general_ci		Tidak	Tidak ada		Ubah Hapus Lainnya	
alamat_ins	text	utf8mb4_general_ci		Tidak	Tidak ada		Ubah Hapus Lainnya	
tgl_masuk_ins	date			Tidak	Tidak ada		Ubah Hapus Lainnya	
id_akun	int(11)			Ya	NULL		Ubah Hapus Lainnya	

Gambar 4. 93 Tabel instruktur

5. Tabel kelas

Implementasi perancangan tabel dapat dilihat pada tabel kelas pada Gambar 4.93.

Nama	Jenis	Penyortiran	Atribut	Tak Ternilai	Bawaan	Komentar	Ekstra	Tindakan
id_kelas	int(11)			Tidak	Tidak ada	AUTO_INCREMENT	Ubah Hapus Lainnya	
nama_kelas	varchar(15)	utf8mb4_general_ci		Tidak	Tidak ada		Ubah Hapus Lainnya	
kuota_siswa	int(2)			Tidak	Tidak ada		Ubah Hapus Lainnya	
id_ins	varchar(5)	utf8mb4_general_ci		Ya	NULL		Ubah Hapus Lainnya	

Gambar 4. 94 Tabel Kelas

6. Tabel bidang_minat

Implementasi perancangan tabel dapat dilihat pada tabel bidang_minat pada Gambar 4.94.

#	Nama	Jenis	Penyortiran	Atribut	Tak Ternilai	Bawaan	Komentar	Ekstra	Tindakan
1	id_bidang	int(11)			Tidak	Tidak ada	AUTO_INCREMENT	Ubah Hapus Lainnya	
2	nama_bidang	varchar(20)	utf8mb4_general_ci		Tidak	Tidak ada		Ubah Hapus Lainnya	

Gambar 4. 95 Tabel bidang_minat

4.4.1.3 Implementasi Model

Implementasi model pada sistem administrasi berbasis *web* dengan ICONIX berbasis *web* di LPK Cipta Kerja DPN Perkasa Jateng dapat dilihat pada Tabel 4.3.

Tabel 4. 3 Implementasi Model Sistem Manajemen Administrasi

No	Nama View	Path File
1	admin.php	app/Models/admin.php
2	bidang_minat.php	app/Models/bidang_minat.php
3	instruktur.php	app/Models/instruktur.php
4	kelas.php	app/Models/kelas.php
5	siswa.php	app/Models/siswa.php
6	users.php	app/Models/users.php

4.4.1.4 Implementasi View

Implementasi view pada sistem administrasi berbasis *web* dengan ICONIX *process* berbasis *web* di LPK Cipta Kerja DPN Perkasa Jateng dapat dilihat pada Tabel 4.4.

Tabel 4. 4 Implementasi View Sistem Manajemen Administrasi

No.	Nama View	Path File
1	login.blade.php	resources/views/auth/Login/login.blade.php
2	dashboard_admin.blade.blade.php	resources/views/admin/dashboard_admin/dashboard_admin.blade.blade.php
3	jadwal.blade.php	resources/views/admin/jadwal/jadwal.blade.php
4	akun_admin.blade.php	resources/views/admin/kelola_akun/akun_admin.blade.php
5	akun_siswa.blade.php	resources/views/admin/kelola_akun/akun_siswa.blade.php
6	akun_instruktur.blade.php	resources/views/admin/kelola_akun/akun_instruktur.blade.php
7	profile_admin.blade.php	resources/views/admin/kelola_profile/profile_admin.blade.php
8	profile_siswa.blade.php	resources/views/admin/kelola_profile/profile_siswa.blade.php
9	profile_instruktur.blade.php	resources/views/admin/kelola_profile/profile_instruktur.blade.php
10	bidang.blade.php	resources/views/admin/kelola_bidang/bidang.blade.php
11	kelas.blade.php	resources/views/admin/kelola_kelas/kelas.blade.php
12	detail.blade.php	resources/views/admin/kelola_kelas/detail.blade.php

13	resetPassword.blade.php	resources/views/auth/reset_password/resetPassword.blade.php
14	templatePesanEmail.blade.php	resources/views/auth/reset_password/templatePesanEmail.blade.php
15	dashboard.blade.php	resources/views/dashboard_menu/dashboard.blade.php
16	edit_profile.blade.php	resources/views/siswa/edit_profile/edit_profile.blade.php

4.4.1.5 Implementasi Controller

Implementasi controller pada sistem administrasi berbasis *web* dengan ICONIX process berbasis *web* di LPK Cipta Kerja DPN Perkasa Jateng dapat dilihat pada Tabel 4.5.

Tabel 4. 5 Implementasi Controller Sistem Manajemen Administrasi

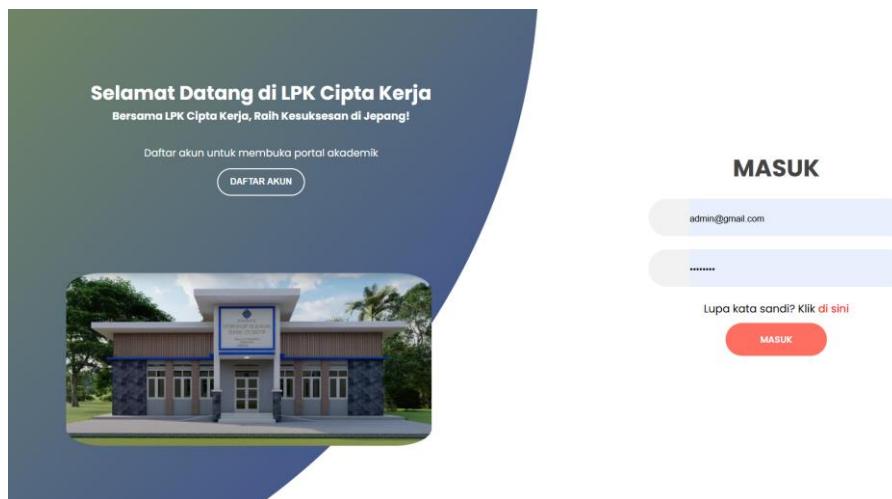
No	Nama View	Path File
1	halamanController.php	app/Http/Controllers/halamanController.php
2	AuthController.php	app/Http/Controllers/AuthController.php
3	BidangController.php	app/Http/Controllers/BidangController.php
4	ProfileController.php	app/Http/Controllers/ProfileController.php
5	akunController.php	app/Http/Controllers/akunController.php
6	jadwalController.php	app/Http/Controllers/jadwalController.php
7	kelasController.php	app/Http/Controllers/kelasController.php

4.4.2 Implementasi Antarmuka

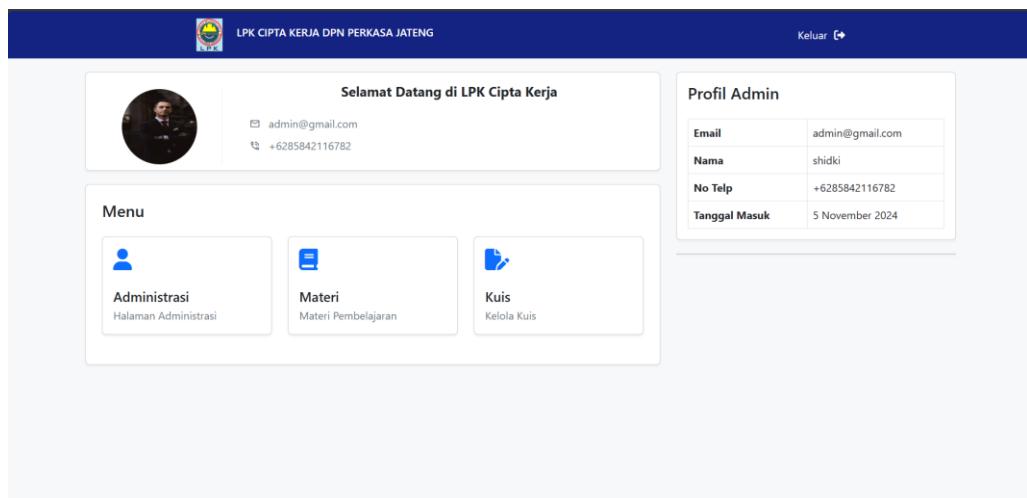
Implementasi antarmuka merupakan implementasi dari perancangan antarmuka yang telah dibuat dalam GUI *storyboard* sebelumnya

1. Antarmuka Halaman Login

Halaman login adalah halaman pertama ketika website ini diakses seperti pada Gambar 4.95 dan Halaman menua salah halaman yang akan ditampilkan setelah melakukan login dengan detail pada Gambar 4.96.



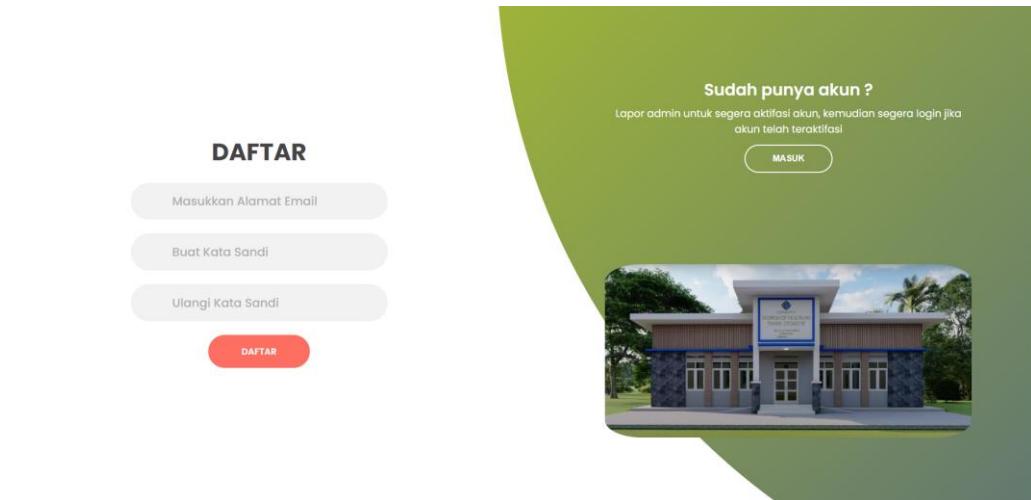
Gambar 4. 96 Antarmuka Halaman Login



Gambar 4. 97 Antarmuka Halaman Menu

2. Antarmuka Halaman Daftar Akun

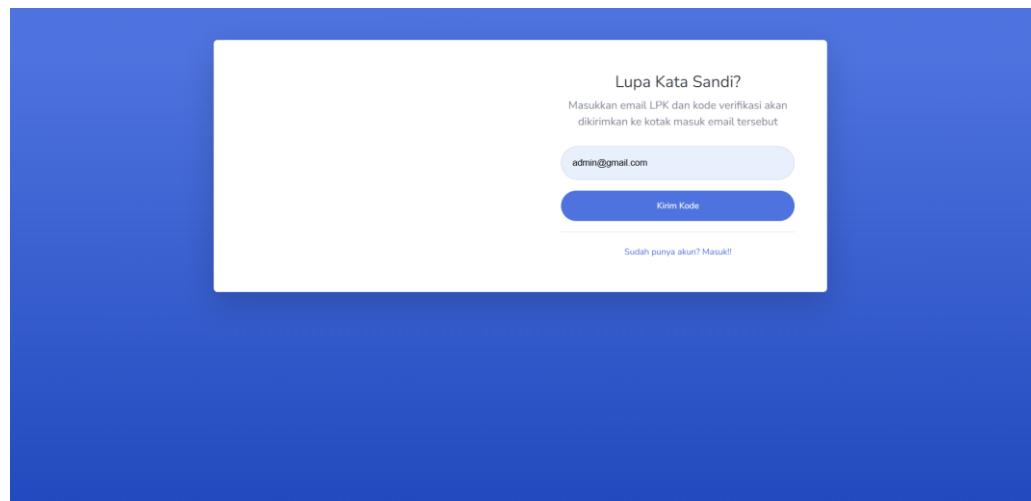
Halaman daftar akun adalah halaman yang dirancang khusus agar siswa yang telah melakukan registrasi dapat membuat akun dengan detail lengkap pada Gambar 4.97.



Gambar 4. 98 Antarmuka Halaman Daftar Akun

3. Antarmuka Halaman Reset Password

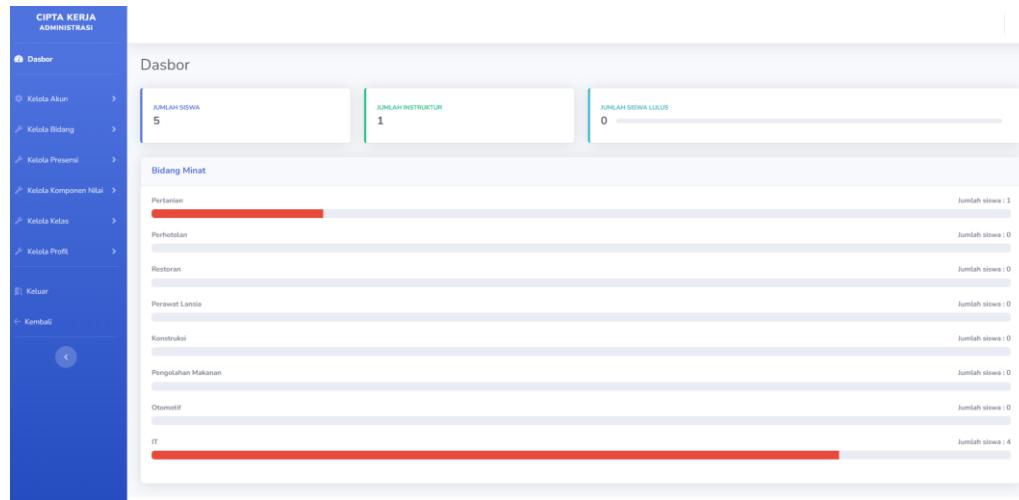
Halaman reset password adalah halaman untuk menangani lupa kata sandi dengan antarmuka tampilan pada Gambar 4.98.



Gambar 4. 99 Antarmuka Halaman Reset Password

4. Antarmuka Dashboard Admin

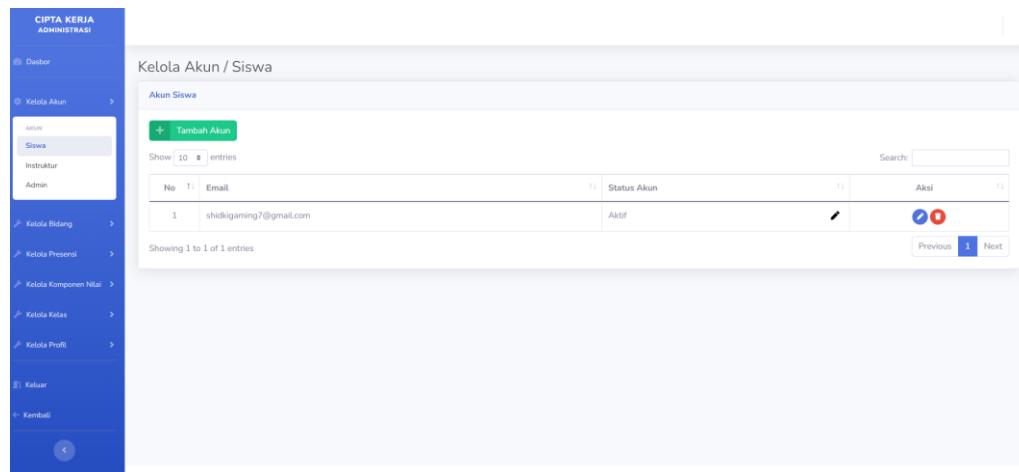
Dashboard Admin adalah halaman yang memberikan ringkasan data yang dikelola admin dengan antarmuka yang dapat lihat pada Gambar 4.99.



Gambar 4. 100 Antarmuka Dashboard Admin

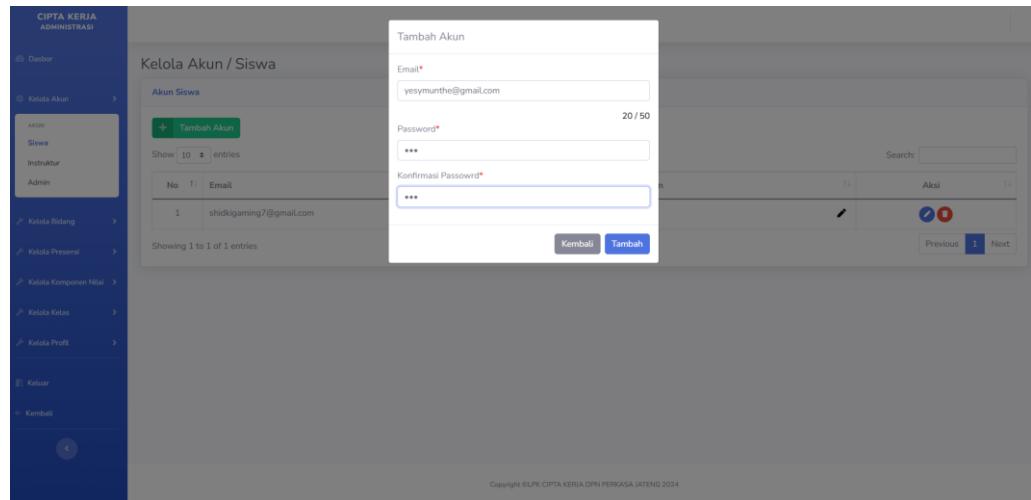
5. Antarmuka Halaman Kelola Akun

Halaman kelola akun adalah halaman yang memungkinkan admin untuk menambahkan, mengedit dan menghapus akun siswa, instruktur maupun admin. Untuk salah satu antarmuka halaman kelola akun dapat dilihat pada Gambar 4.100.

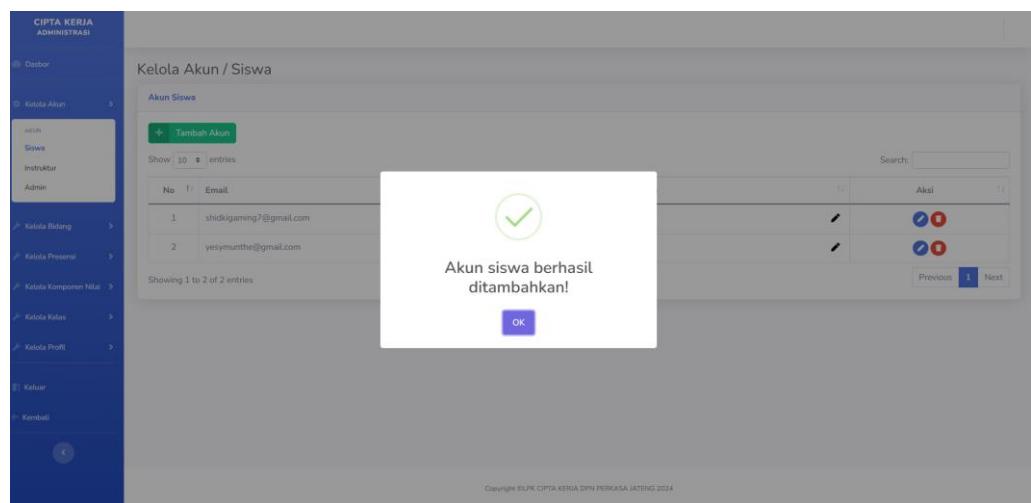


Gambar 4. 101 Antarmuka Halaman Kelola Akun

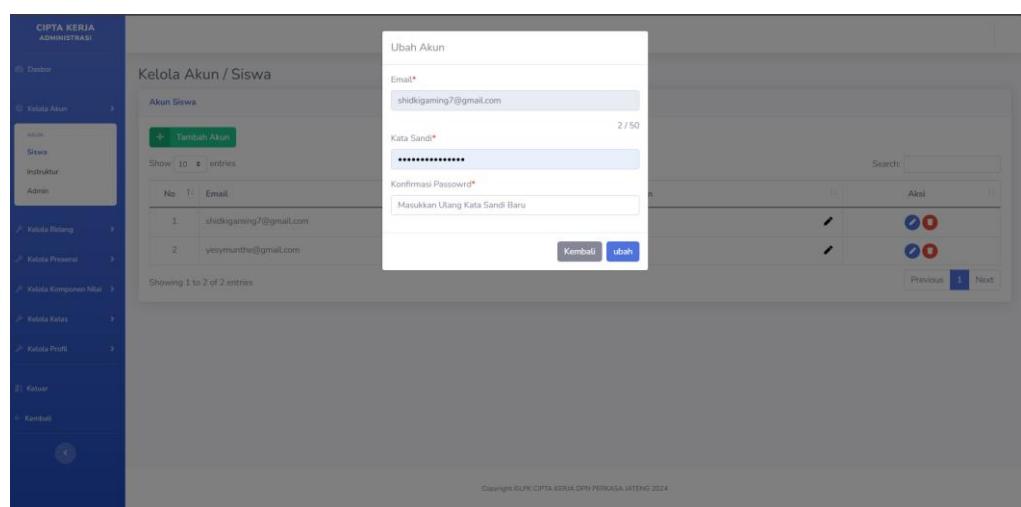
Untuk salah satu antarmuka halaman tambah akun dapat dilihat pada Gambar 4.101 dan antarmuka model konfirmasi berhasil tambah akun dapat dilihat pada Gambar 4.102.



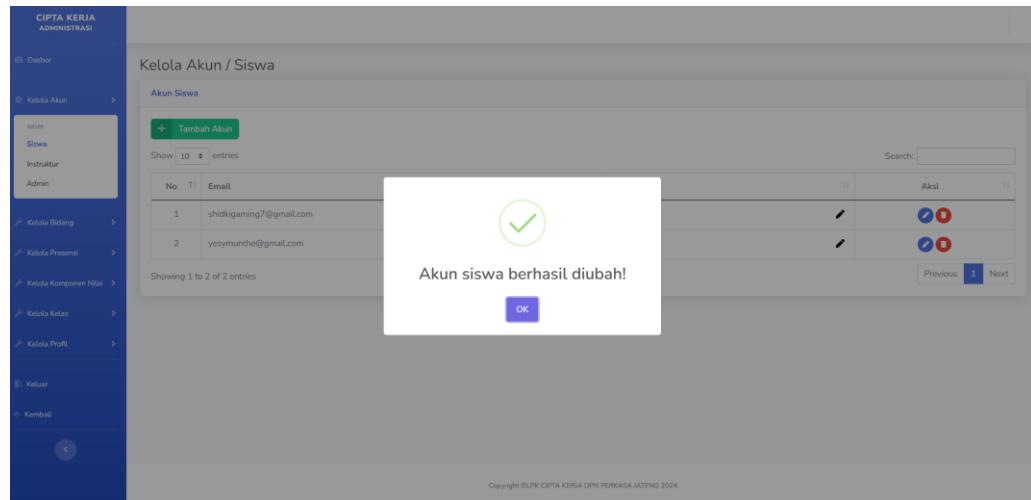
Gambar 4. 102 Antarmuka modal tambah akun



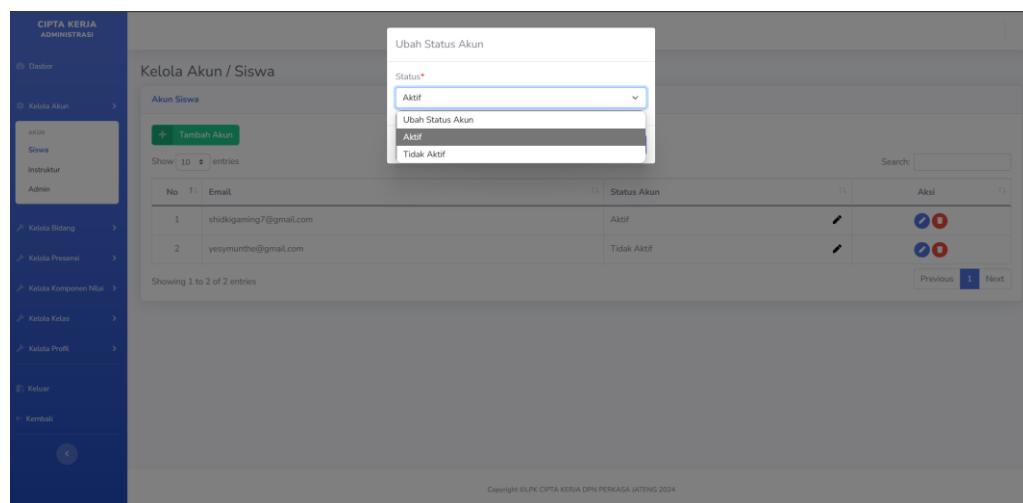
Gambar 4. 103 Antarmuka modal konfirmasi



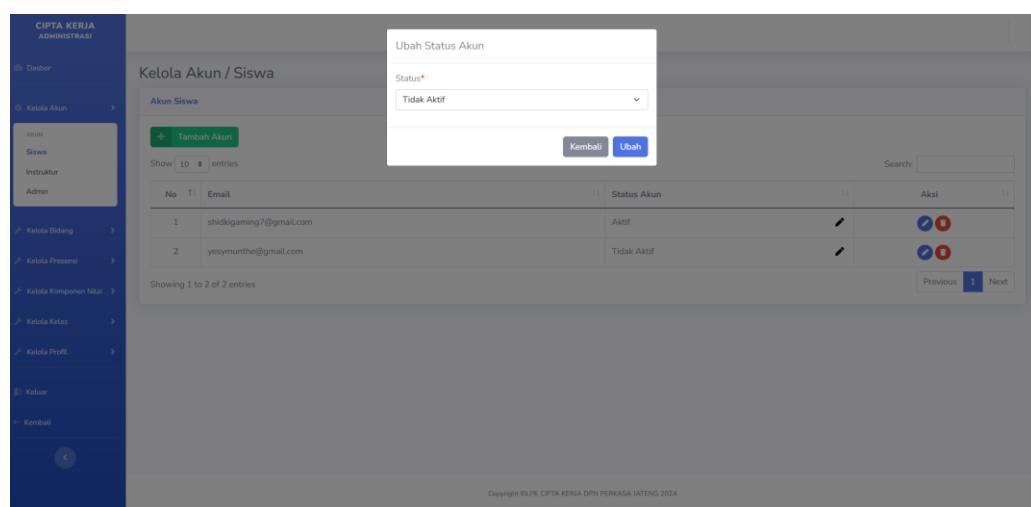
Gambar 4. 104 Atarmuka modal ubah akun



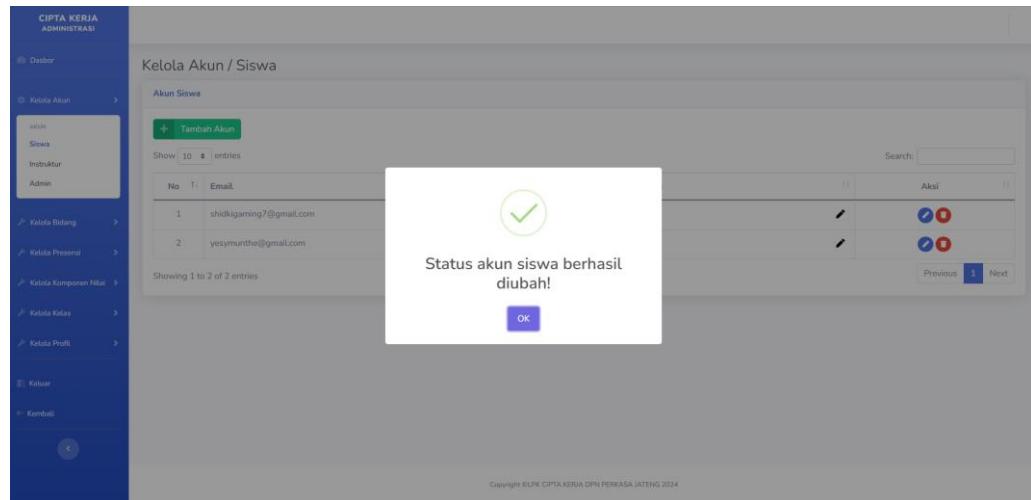
Gambar 4. 105 Atarmuka modal berhasil ubah akun



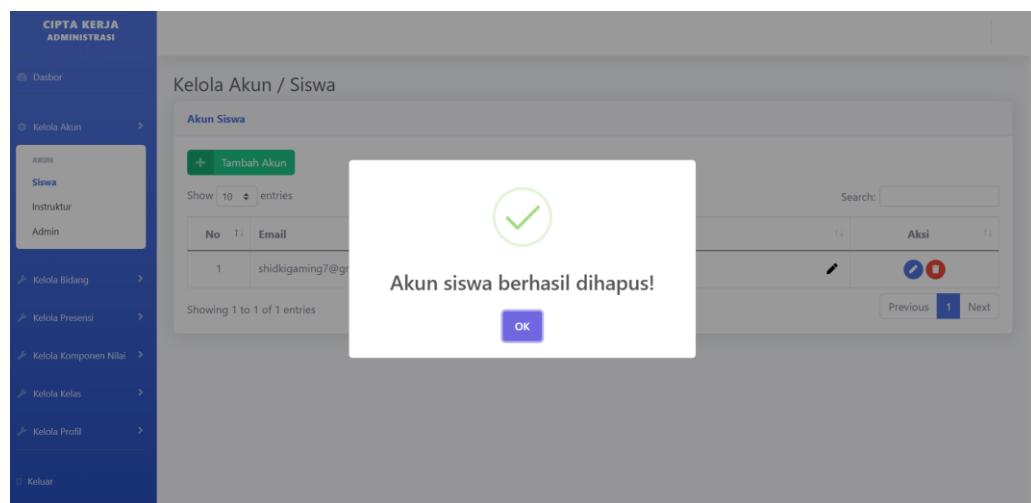
Gambar 4. 106 Antarmuka Ubah status akun



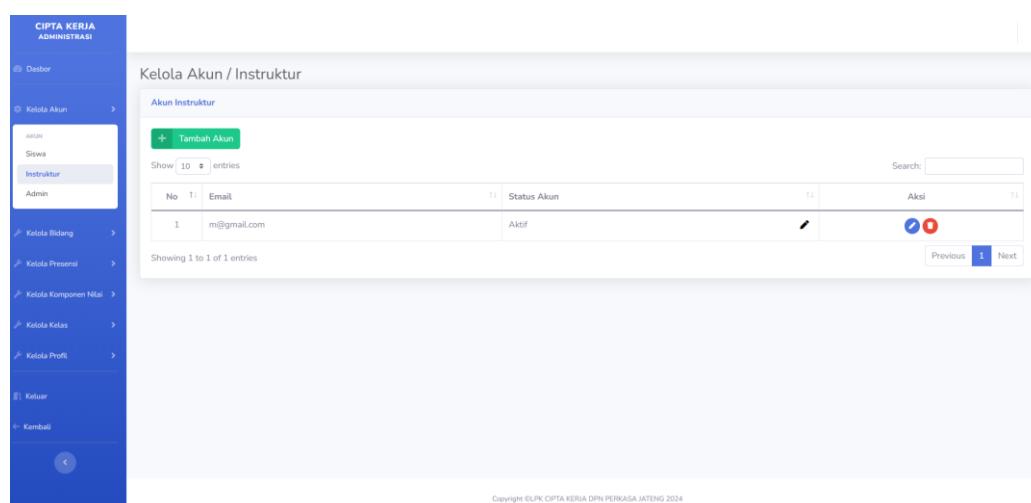
Gambar 4. 107 Antarmuka Ubah status akun



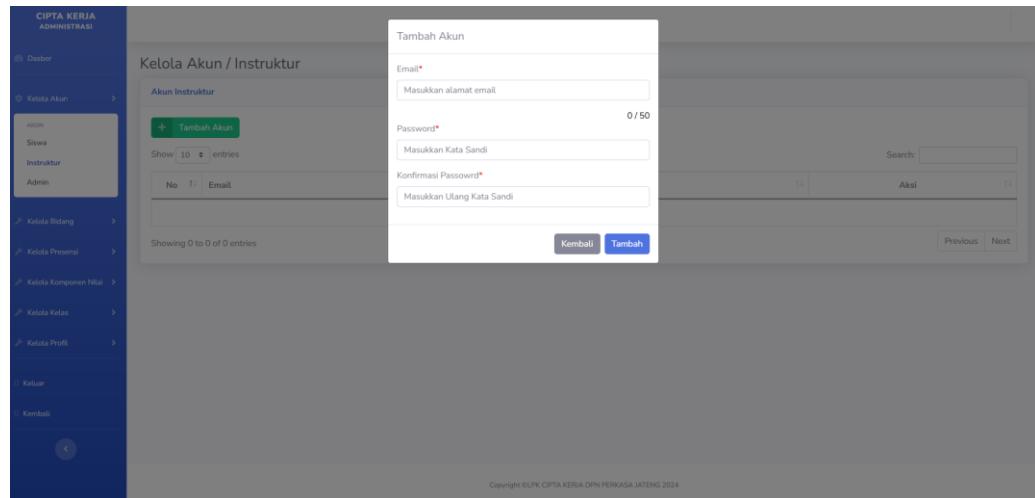
Gambar 4. 108 Antarmuka berhasil Ubah status akun siswa



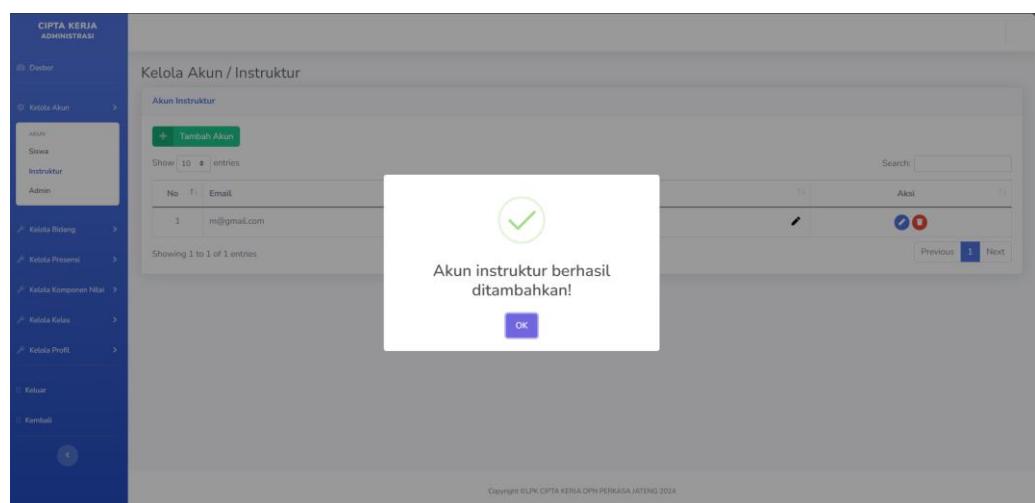
Gambar 4. 109 Antarmuka Halaman Hapus Siswa



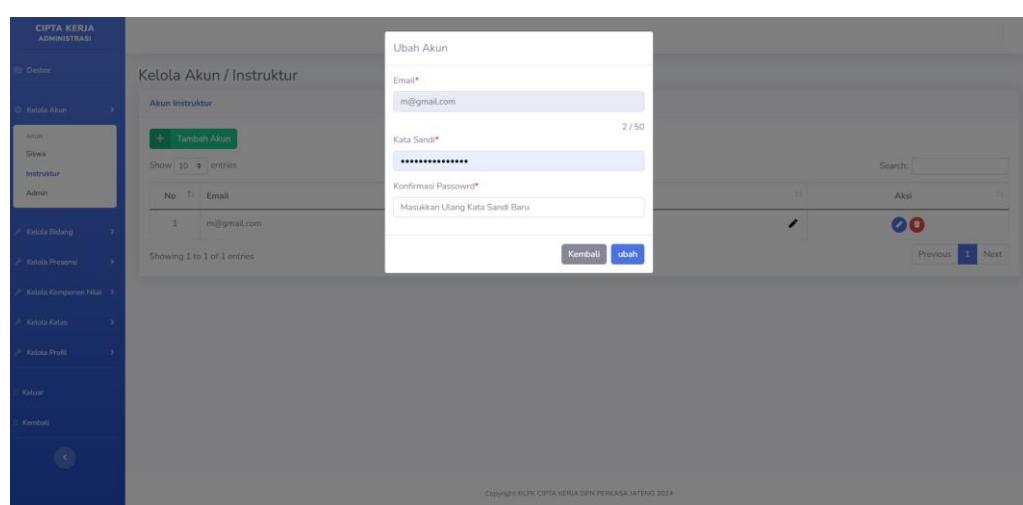
Gambar 4. 110 Antarmuka Halaman Kelola Akun Instruktur



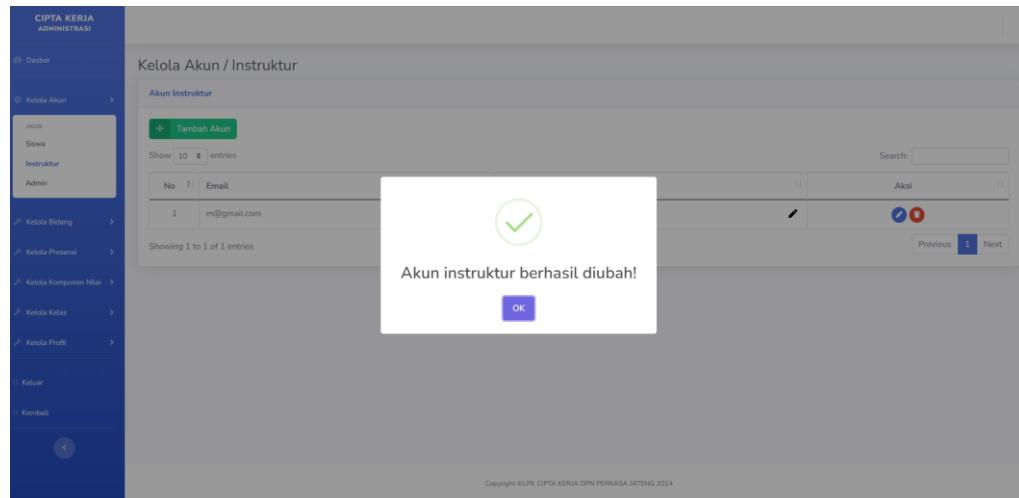
Gambar 4. 111 Antarmuka pop up Tambah Akun Instruktur



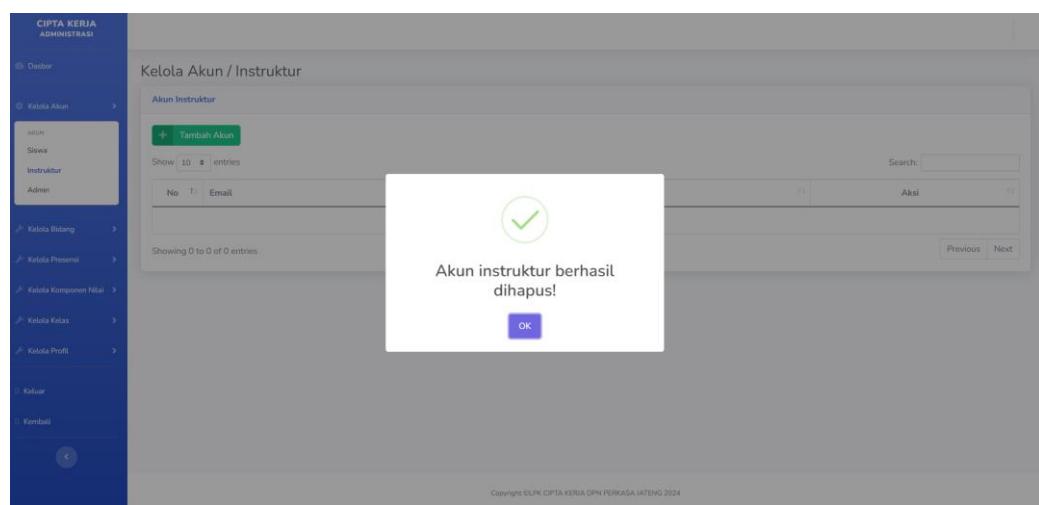
Gambar 4. 112 Antarmuka Konfirmasi berhasil menambah instruktur



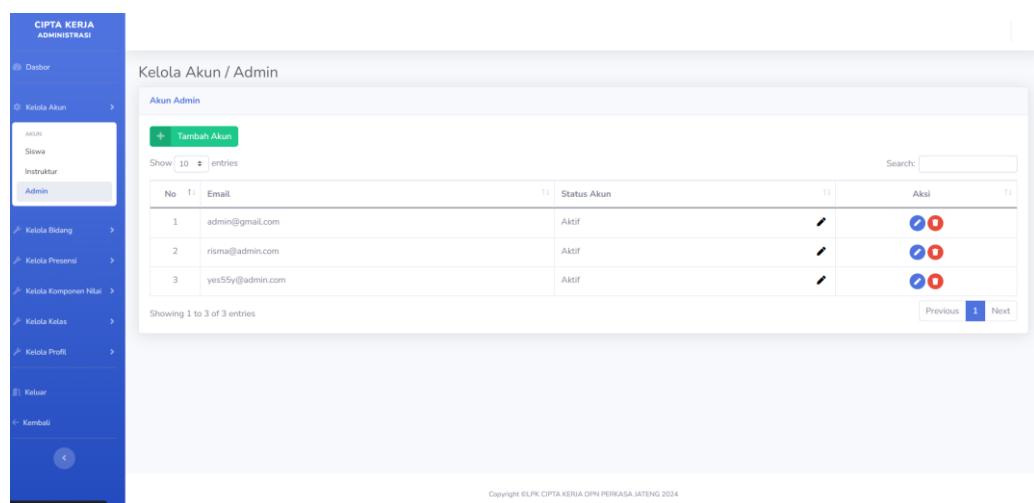
Gambar 4. 113 Antarmuka modal ubah akun instruktur



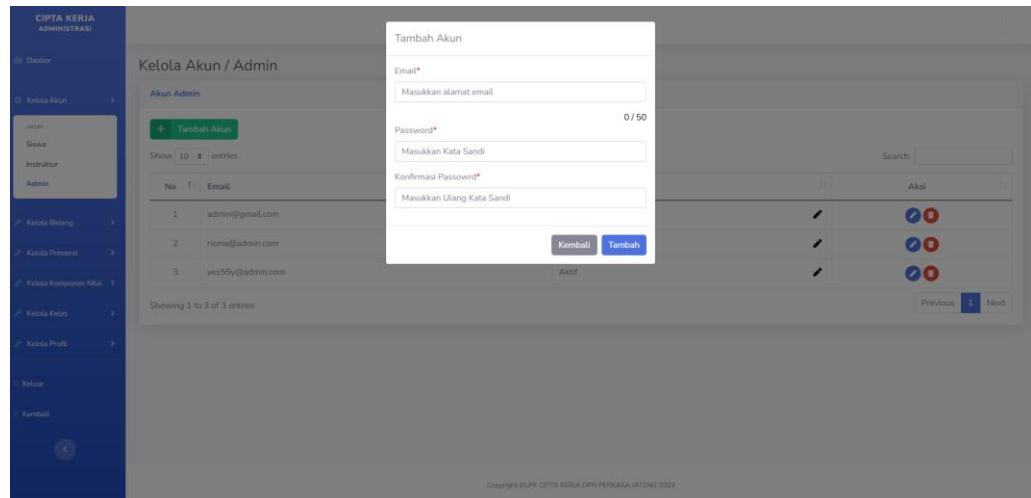
Gambar 4. 114 Antarmuka modal konfirmasi berhasil ubah akun instruktur



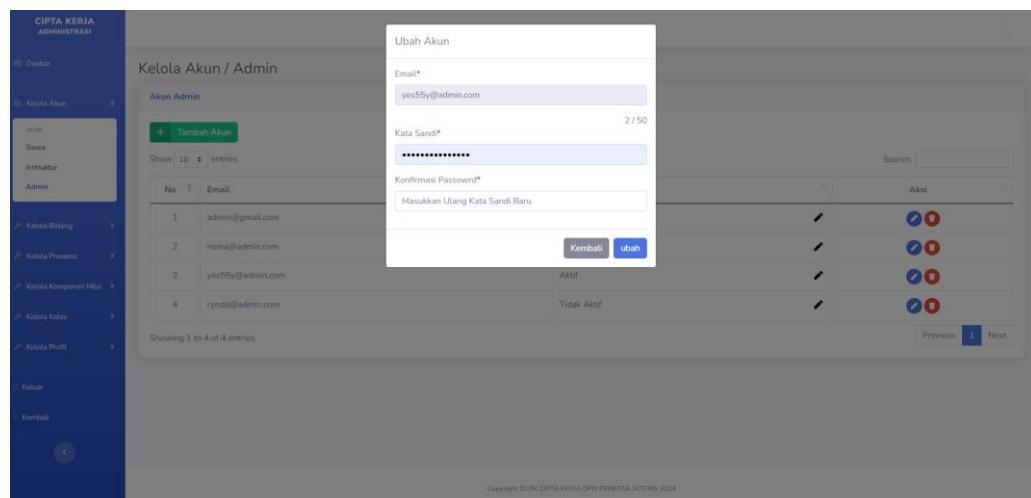
Gambar 4. 115 Antarmuka Halaman Hapus Akun Instruktur



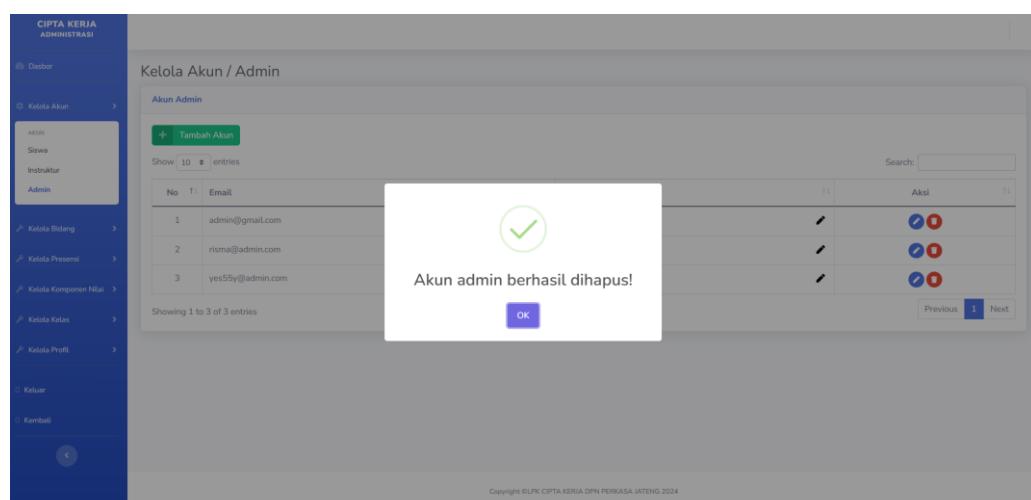
Gambar 4. 116 Antarmuka Halaman Kelola Akun Admin



Gambar 4. 117 Antarmuka Halaman Tambah Akun Admin



Gambar 4. 118 Antarmuka Halaman Ubah Akun Admin



Gambar 4. 119 Antarmuka Halaman Hapus Akun Admin

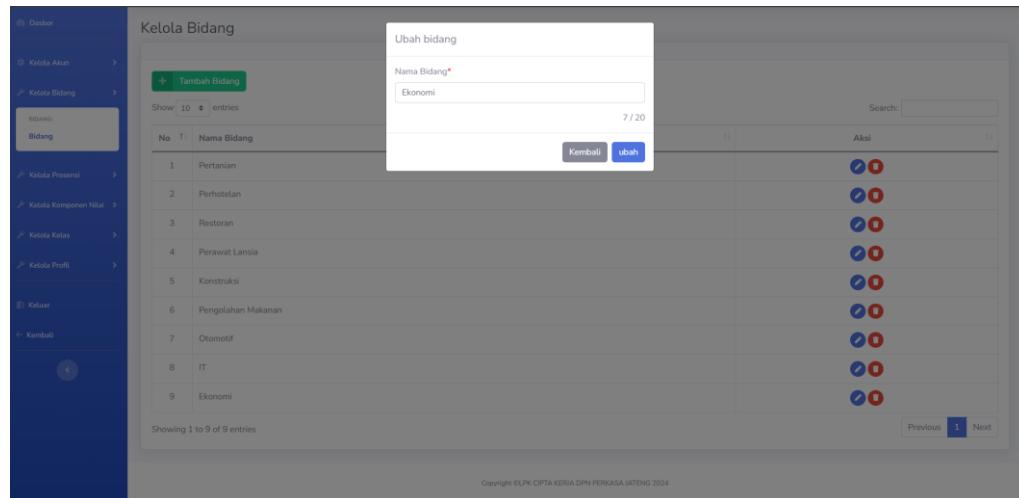
6. Antarmuka Halaman Kelola Bidang

Halaman kelola bidang adalah halaman yang memungkinkan admin untuk mengakses halaman kelola bidang dapat dilihat pada Gambar 4.120. Selain itu admin juga dapat menambahkan dengan antarmuka pada Gambar 4.121, mengedit dengan antarmuka pada Gambar 4.122 dan menghapus bidang minat dengan antarmuka Gambar 4.123.

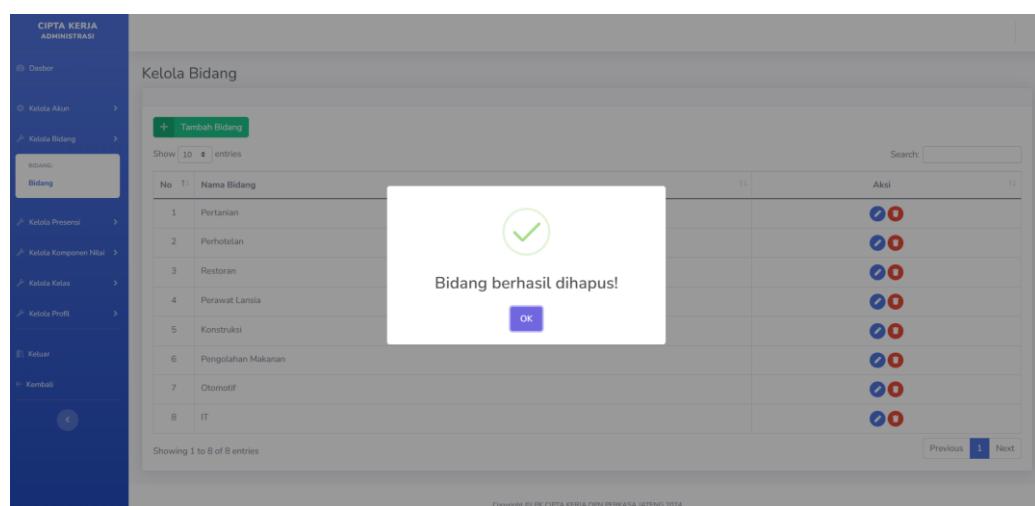
No	Nama Bidang	Aksi
1	Pertanian	
2	Perhotelan	
3	Restoran	
4	Perawat Lansia	
5	Konstruksi	
6	Pengolahan Makanan	
7	Otomotif	
8	IT	

Gambar 4. 120 Antarmuka Halaman Kelola Bidang

Gambar 4. 121 Antarmuka Halaman tambah bidang



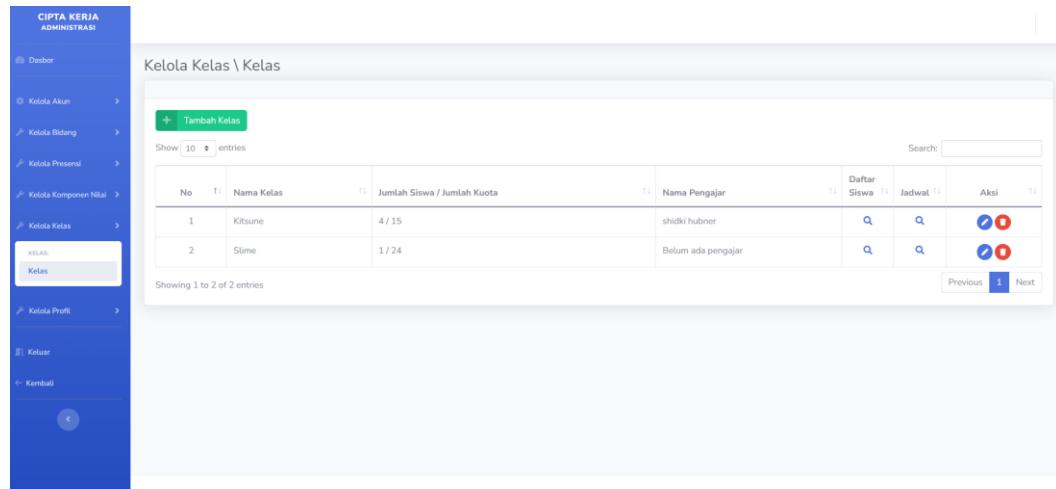
Gambar 4. 122 Antarmuka Halaman ubah bidang



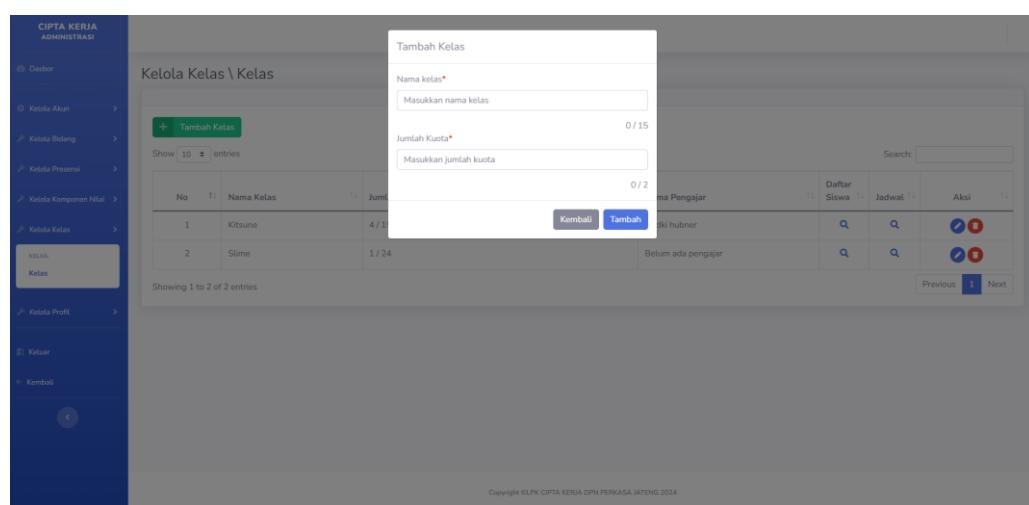
Gambar 4. 123 Antarmuka Halaman hapus Bidang

7. Antarmuka Halaman Kelola Kelas

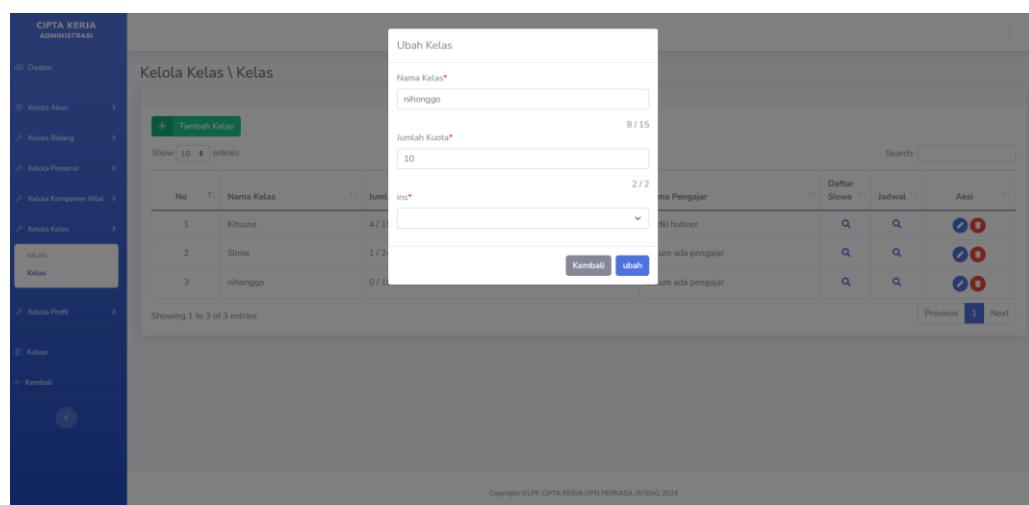
Halaman kelola Kelas adalah halaman yang memungkinkan admin untuk mengakses halaman kelola kelas dapat dilihat pada Gambar 4.124. Selain itu admin juga dapat menambahkan dengan antarmuka pada Gambar 4.125, mengedit dengan antarmuka pada Gambar 4.126 dan menghapus kelas dengan antarmuka Gambar 4.126.



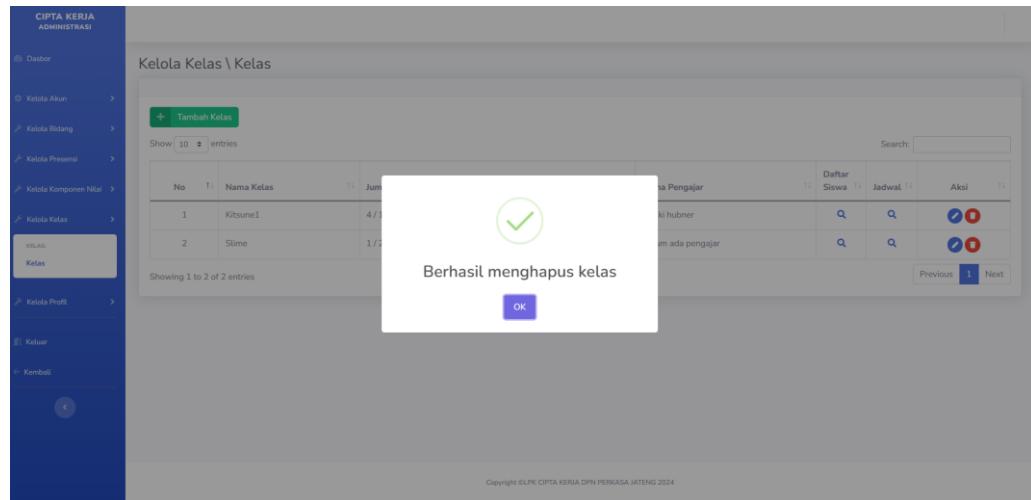
Gambar 4. 124 Antarmuka Halaman Kelola Kelas



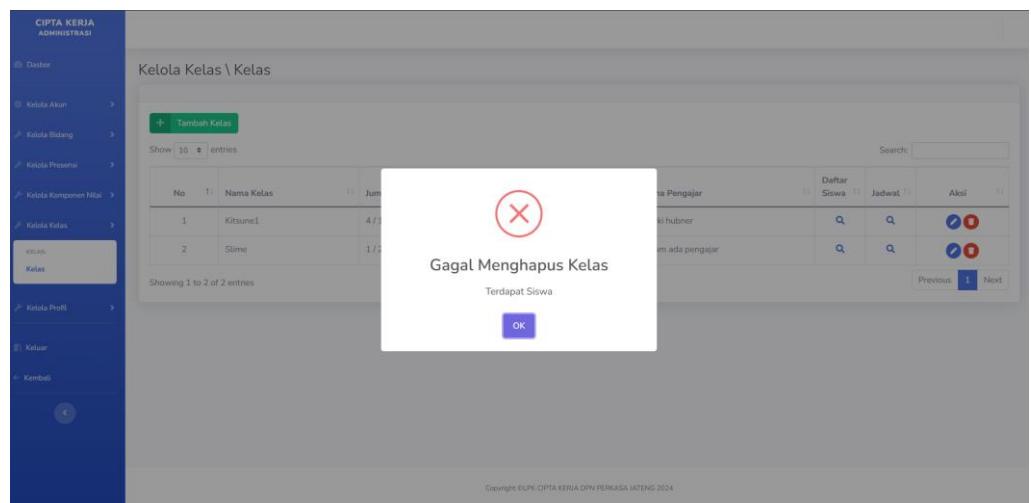
Gambar 4. 125 Antarmuka Halaman Tambah Kelas



Gambar 4. 126 Antarmuka Halaman Ubah Kelas



Gambar 4. 127 Antarmuka Halaman Hapus Kelas Kondisi Sukses Hapus



Gambar 4. 128 Antarmuka Halaman Hapus Kelas Kondisi Gagal hapus

8. Antar muka Halaman Kelola Profil

Halaman kelola Profil adalah halaman yang memungkinkan admin untuk mengakses halaman kelola Profil untuk siswa, instruktur dan admin.

The screenshot shows the 'Kelola Profil' section of the application. On the left, there's a sidebar with navigation links: Dasbor, Kelola Akun, Kelola Bidang, Kelola Presensi, Kelola Komponen Nilai, Kelola Kelas, and Kelola Profil. Under 'Kelola Profil', there are sub-links for PROFIL, Siswa, Instruktur, and Admin. Below these are 'Keluar' and 'Kembali' buttons. The main area is titled 'Kelola Data Profile / Siswa' and contains a table with columns: No, Tgl, Nama, Email, No HP, Alamat, Tgl, Kelas, Bidang, Tgl Masuk, Tgl Lulus, Status, and Aksi. The table lists six student profiles with various details like names, emails, phone numbers, addresses, and assigned fields.

Gambar 4. 129 Antar muka Halaman Kelola Profil

This screenshot shows the 'Tambah Siswa' (Add Student) form. It includes fields for 'Nama Siswa*', 'Email Siswa*', 'No HP*', 'Alamat*', 'Bidang Minat*', 'Kelas*', and 'Tanggal Masuk*'. The 'Bidang Minat' field has a dropdown menu labeled 'Pilih Bidang'. The 'Kelas*' field has a dropdown menu labeled 'Kelas Siswa'. The 'Tanggal Masuk*' field has a date input set to '05/02/2024'. The background shows a blurred view of the 'Data Siswa' table from the previous screenshot.

Gambar 4. 130 Antar muka Halaman Tambah Profil

This screenshot shows the 'Tambah Siswa' form with all fields filled in. The 'Nama Siswa' field contains 'siswa1@siswa.com', 'Email Siswa' contains '088903129894', 'No HP' contains '088903129894', 'Alamat' contains 'Jl. Tembalang Selatan VI No. 25', 'Bidang Minat' is set to 'Perhotelan', 'Kelas' is set to 'Kitsune', 'Tanggal Masuk' is set to '05/02/2024', and 'Status' is set to 'Aktif'. At the bottom right are 'Kembali' and 'Ubah' buttons.

Gambar 4. 131 Antar muka Halaman tambah Profi

BAB V

PENUTUP

5.1 Kesimpulan

Kesimpulan dari penelitian Rancang Bangun Aplikasi Sistem Administrasi berbasis *web* menggunakan ICONIX *process* di LPK Cipta Kerja DPN Perkasa Jateng yaitu sebagai berikut.

1. Aplikasi Sistem Manajemen Administrasi ini berhasil dikembangkan dengan menggunakan bahasa pemrograman PHP yang dipadukan dengan framework Laravel, serta teknologi web seperti JavaScript, HTML, dan CSS. Pengelolaan database dilakukan melalui PHPMyAdmin dengan MySQL. Metode pengembangan yang digunakan adalah ICONIX Process, sedangkan pengujian dilakukan dengan metode black box.
2. Aplikasi ini dirancang untuk mempermudah proses administrasi di LPK Cipta Kerja DPN Perkasa Jateng, terutama dalam mengelola data siswa, instruktur, dan admin. Sistem ini memungkinkan pengelolaan data yang lebih efisien, transparan, dan mudah diakses, serta meningkatkan kualitas layanan dan pelayanan administrasi yang pada akhirnya dapat meningkatkan efektivitas dan efisiensi operasional LPK Cipta Kerja DPN Perkasa Jateng.

5.2 Saran

Beberapa saran yang dapat diberikan untuk pengembangan lebih lanjut dari Aplikasi Sistem Manajemen Administrasi di LPK Cipta Kerja DPN Perkasa Jateng adalah sebagai berikut:

1. Diperlukan peningkatan sumber daya yang lebih banyak untuk mendukung pengembangan sistem agar dapat berfungsi lebih optimal dan sesuai dengan kebutuhan yang berkembang.
2. Penggunaan metode pengembangan dan pengujian lain dapat menjadi alternatif untuk melihat perbedaan hasil dan memastikan bahwa sistem yang dibangun semakin baik.

DAFTAR PUSTAKA

LAMPIRAN 1. Dokumen Pengambilan Data dan Informasi

**Dokumentasi Pengambilan Data dan Informasi di LPK Cipta Kerja DPNPerkasa
Jateng Untuk Pembuatan Aplikasi Sistem Adminitrasi**

Berikut merupakan beberapa hal yang menjadi output diskusi permbuatan Aplikasi Sitem Adminitrasi Berbasis *web* yang digunakan sebagai batasan masalah.

1. Hanya terdapat role admin, admin dapat mengedit profil dan akun admin satu sama lain. Tidak ada role master admin
2. Email dan no telp pada profil siswa, instruktur, dan admin bersifat unik, tidak boleh sama
3. Status siswa tidak bisa langsung dianggap siswa aktif setelah melakukan pendaftaran siswa baru
4. Tidak bisa menghapus data siswa ketika siswa sudah mempunyai nilai
5. Ketika siswa sudah lulus, maka data profile siswa tidak bisa di edit dan tidak bisa dihapus karena untuk menghindari human error yang mengubah data siswa yang telah lulus
6. Data kelas dapat ditambahkan terlebih dahulu walaupun belum ada data instruktur yang mengampu
7. 1 Instruktur hanya dapat mengampu 1 Kelas
8. Ketika siswa tidak aktif, lulus, mangkir, status akun siswa terkait akan menjadi aktif dan tidak bisa diakifkan selama siswa tersebut bukan menjadi siswa aktif

Lampiran 2. Tabel Pengujian Aplikasi

Tabel 5.1 Rencana Pengujian

Kelas Uji	Butir Uji	Identifikasi		Aktor	Jenis Pengujian	Teknik Pengujian
		Use Case	DUPL			
Login	Pengujian memasukkan data masukan email dan password yang valid	1	DUPL- BSO-1-01	Admin	Sistem	<i>Black box</i>
	Pengujian memasukkan email terdaftar dengan password yang salah.		DUPL- BSO-1-02	Admin	Sistem	<i>Black box</i>
	Pengujian memasukkan email yang tidak terdaftar.		DUPL- BSO-1-03	Admin	Sistem	<i>Black box</i>
Logout	Menekan tombol keluar di halaman Administrasi	2	DUPL- BSO-2-01	Admin	Sistem	<i>Black box</i>
	Menekan tombol keluar di halaman Menu		DUPL- BSO-2-02	Admin	Sistem	<i>Black box</i>
Mengakses Dashboard	Pengujian menekan menu “Administrasi”	3	DUPL- BSO-3-01	Admin	Sistem	<i>Black box</i>
Kelola Kelas	Pengujian menekan <i>dropdown</i> kelola kelas	4	DUPL- BSO-4-01	Admin	Sistem	<i>Black box</i>
	Pengujian menekan pilihan kelas		DUPL- BSO-4-02	Admin	Sistem	<i>Black box</i>
	Pengujian menekan tombol “Tambah Kelas”		DUPL- BSO-4-03	Admin	Sistem	<i>Black box</i>
	Pengujian memasukkan data masukan kelas yang valid		DUPL- BSO-4-04	Admin	Sistem	<i>Black box</i>
	Pengujian memasukkan data tambah kelas dengan nama kelas melebihi jumlah masukan yang telah ditentukan		DUPL- BSO-4-05	Admin	Sistem	<i>Black box</i>
	Pengujian memasukkan data tambah kelas dengan jumlah kuota melebihi jumlah masukan yang telah ditentukan		DUPL- BSO-4-06	Admin	Sistem	<i>Black box</i>
	Pengujian memasukkan data tambah kelas dengan jumlah kuota dengan karakter selain angka seperti huruf dan tanda baca.		DUPL- BSO-4-07	Admin	Sistem	<i>Black box</i>

Kelas Uji	Butir Uji	Identifikasi		Aktor	Jenis Pengujian	Teknik Pengujian
		Use Case	DUPL			
Kelas Kelas	Pengujian menekan tombol “Tambah” di modal “Tambah Kelas”					
	Pengujian menekan tombol “Kembali” di modal “Tambah Kelas”		DUPL- BSO-4-08	Admin	Sistem	
	Menekan ikon <i>search</i> di daftar siswa		DUPL- BSO-4-9	Admin	Sistem	<i>Black box</i>
	Pengujian menekan ikon edit		DUPL- BSO-4-10	Admin	Sistem	<i>Black box</i>
	Pengujian menekan ikon hapus		DUPL- BSO-4-11	Admin	Sistem	<i>Black box</i>
	Pengujian menekan <i>dropdown ins</i>		DUPL- BSO-4-12	Admin	Sistem	<i>Black box</i>
	Pengujian memasukkan data ubah kelas dengan dengan ins yang sudah di menjadi pengajar di kelas lain		DUPL- BSO-4-13	Admin	Sistem	<i>Black box</i>
	Pengujian memasukkan data ubah kelas yang valid		DUPL- BSO-4-14	Admin	Sistem	<i>Black box</i>
	Pengujian memasukkan huruf di kolom <i>search</i>		DUPL- BSO-4-15	Admin	Sistem	<i>Black box</i>
	Pengujian menekan tombol “ <i>Previous</i> ”		DUPL- BSO-4-16	Admin	Sistem	<i>Black box</i>
	Pengujian menekan tombol “ <i>Next</i> ”		DUPL- BSO-4-17	Admin	Sistem	<i>Black box</i>
	Pengujian menekan <i>dropdown kelola bidang</i>		DUPL- BSO-4-19	Admin	Sistem	<i>Black box</i>
	Pengujian menekan pilihan “Bidang”					
Kelola Bidang	Pengujian menekan tombol “Tambah Bidang”	5	DUPL- BSO-5-01	Admin	Sistem	<i>Black box</i>
	Pengujian memasukkan data masukan tambah bidang yang valid		DUPL- BSO-5-02	Admin	Sistem	<i>Black box</i>
	Pengujian memasukkan data masukan dengan mengisi nama bidang melebihi jumlah masukan yang telah ditentukan		DUPL- BSO-5-03	Admin	Sistem	<i>Black box</i>
			DUPL- BSO-5-04	Admin	Sistem	<i>Black box</i>
			DUPL- BSO-5-05	Admin	Sistem	<i>Black box</i>

Kelas Uji	Butir Uji	Identifikasi		Aktor	Jenis Pengujian	Teknik Pengujian
		Use Case	DUPL			
	Pengujian memasukkan data masukan dengan mengisi nama bidang dengan nama bidang yang sudah terdaftar di tabel bidang		DUPL- BSO-5-06	Admin	Sistem	<i>Black box</i>
	Pengujian menekan ikon edit		DUPL- BSO-5-07		Sistem	<i>Black box</i>
	Pengujian memasukkan data masukan ubah bidang dengan nama bidang yang valid		DUPL- BSO-5-08		Sistem	<i>Black box</i>
	Pengujian menekan tombol “kembali” di modal “Ubah bidang”		DUPL- BSO-5-09		Sistem	<i>Black box</i>
	Pengujian menekan ikon hapus		DUPL- BSO-5-10		Sistem	<i>Black box</i>
	Pengujian memasukkan huruf di kolom <i>search</i>		DUPL- BSO-5-11		Sistem	<i>Black box</i>
	Pengujian menekan tombol “Previous”		DUPL- BSO-5-12		Sistem	<i>Black box</i>
	Pengujian menekan tombol “Next”		DUPL- BSO-5-13		Sistem	<i>Black box</i>
Kelola Profil	Pengujian menekan dropdown kelola profil	6	DUPL- BSO-6-01	Admin	Sistem	<i>Black box</i>
	Pengujian menekan pilihan “Siswa”		DUPL- BSO-6-02		Sistem	<i>Black box</i>
	Pengujian menekan pilihan “Instruktur”		DUPL- BSO-6-03		Sistem	<i>Black box</i>
	Pengujian menekan pilihan “Admin”		DUPL- BSO-6-04		Sistem	<i>Black box</i>
	Pengujian menekan tombol “Tambah Siswa”		DUPL- BSO-6-05		Sistem	<i>Black box</i>
	Pengujian menekan drop down “Bidang Minat” dimodal Tambah Siswa		DUPL- BSO-6-06		Sistem	<i>Black box</i>
	Pengujian menekan dropd down “Kelas” dimodal Tambah Siswa		DUPL- BSO-6-07		Sistem	<i>Black box</i>
	Pengujian menekan dropd down “Status” dimodal Tambah Siswa		DUPL- BSO-6-08		Sistem	<i>Black box</i>
	Pengujian menekan ikon kalender dikolom “Tanggal Masuk” di modal Tambah Siswa		DUPL- BSO-6-09		Sistem	<i>Black box</i>

Kelas Uji	Butir Uji	Identifikasi		Aktor	Jenis Pengujian	Teknik Pengujian
		Use Case	DUPL			
	Pengujian memasukkan data masukan tambah siswa yang valid		DUPL- BSO-6-10	Admin	Sistem	<i>Black box</i>
	Pengujian memasukkan data masukan tambah siswa dengan data siswa yang sudah terdaftar di tabel.		DUPL- BSO-6-11	Admin	Sistem	<i>Black box</i>
	Pengujian memasukkan data masukan tambah siswa dengan memasukkan email siswa tanpa format @.		DUPL- BSO-6-12	Admin	Sistem	<i>Black box</i>
	Pengujian memasukkan data masukan tambah siswa dengan No Hp dalam format tidak valid		DUPL- BSO-6-13	Admin	Sistem	<i>Black box</i>
	Pengujian memasukkan data masukan tambah siswa dengan mengosongkan 1 kolom isian		DUPL- BSO-6-14	Admin	Sistem	<i>Black box</i>
	Pengujian memasukkan data masukan tambah siswa dengan mengosongkan semua kolom		DUPL- BSO-6-15	Admin	Sistem	<i>Black box</i>
	Pengujian menekan tombol “Tambah” di modal Tambah Siswa		DUPL- BSO-6-16	Admin	Sistem	<i>Black box</i>
	Pengujian menekan tombol “Kembali” di modal Tambah Siswa		DUPL- BSO-6-17	Admin	Sistem	<i>Black box</i>
	Pengujian menekan ikon edit		DUPL- BSO-6-18	Admin	Sistem	<i>Black box</i>
	Pengujian memasukkan data masukan ubah siswa dengan email siswa yang sudah terdaftar di tabel siswa		DUPL- BSO-6-19	Admin	Sistem	<i>Black box</i>
	Pengujian memasukkan data masukan ubah siswa dengan pilihan status “Lulus” tetapi tidak mengisi kolom tanggal lulus.		DUPL- BSO-6-20	Admin	Sistem	<i>Black box</i>
	Pengujian menekan tombol “Kembali” di modal ubah siswa		DUPL- BSO-6-21	Admin	Sistem	<i>Black box</i>
	Pengujian menekan ikon hapus di tabel data siswa		DUPL- BSO-6-22	Admin	Sistem	<i>Black box</i>
	Pengujian menekan tombol “Tambah Instruktur”		DUPL- BSO-6-26	Admin	Sistem	<i>Black box</i>

Kelas Uji	Butir Uji	Identifikasi		Aktor	Jenis Pengujian	Teknik Pengujian
		Use Case	DUPL			
	Pengujian menekan ikon kalender dikolom “Tanggal Masuk” di modal Tambah Siswa		DUPL- BSO-6-28	Admin	Sistem	<i>Black box</i>
	Pengujian memasukkan data masukan tambah instruktur dengan data masukan email dan No Hp yang sudah terdaftar di tabel profil instruktur		DUPL- BSO-6-29	Admin	Sistem	<i>Black box</i>
	Pengujian menekan ikon edit di tabel profil instruktur		DUPL- BSO-6-30	Admin	Sistem	<i>Black box</i>
	Pengujian menekan ikon hapus di tabel profil instruktur		DUPL- BSO-6-31	Admin	Sistem	<i>Black box</i>
	Pengujian memasukkan data masukan ubah instruktur dengan data instruktur yang sudah ada di tabel profil instruktur		DUPL- BSO-6-32	Admin	Sistem	<i>Black box</i>
	Pengujian menekan tombol “ubah” di modal ubah instruktur		DUPL- BSO-6-33	Admin	Sistem	<i>Black box</i>
	Pengujian menekan tombol “kembali” di modal ubah intruktur		DUPL- BSO-6-34	Admin	Sistem	<i>Black box</i>
	Pengujian memasukkan huruf di kolom search data instruktur		DUPL- BSO-6-35	Admin	Sistem	<i>Black box</i>
	Pengujian menekan tombol “Previous” data instruktur		DUPL- BSO-6-36	Admin	Sistem	<i>Black box</i>
	Pengujian menekan tombol “Next” data instruktur		DUPL- BSO-6-37	Admin	Sistem	<i>Black box</i>
	Pengujian menekan tombol “Tambah Admin”		DUPL- BSO-6-38	Admin	Sistem	<i>Black box</i>
	Pengujian memasukkan data masukan tambah admin yang valid		DUPL- BSO-6-39	Admin	Sistem	<i>Black box</i>
	Pengujian memasukkan data masukan tambah admin dengan data admin yang sudah ada di daftar admin		DUPL- BSO-6-40	Admin	Sistem	<i>Black box</i>
	Pengujian memasukkan huruf di kolom search data		DUPL- BSO-6-41	Admin	Sistem	<i>Black box</i>

Kelas Uji	Butir Uji	Identifikasi		Aktor	Jenis Pengujian	Teknik Pengujian
		Use Case	DUPL			
	admin					
	Pengujian menekan tombol “Previous” data admin		DUPL- BSO-6-42	Admin	Sistem	Black box
	Pengujian menekan tombol “Next” data admin		DUPL- BSO-6-43	Admin	Sistem	Black box
Kelola Profil	Pengujian menekan drop down kelola akun	7	DUPL- BSO-7-01	Admin	Sistem	Black box
	Pengujian menekan pilihan akun “Siswa”		DUPL- BSO-7-02	Admin	Sistem	Black box
	Pengujian menekan pilihan akun “Instruktur”		DUPL- BSO-7-03	Admin	Sistem	Black box
	Pengujian menekan pilihan akun “Admin”		DUPL- BSO-7-04	Admin	Sistem	Black box
	Pengujian menekan tombol “Tambah Akun” di halaman kelola akun siswa		DUPL- BSO-7-05	Admin	Sistem	Black box
	Pengujian memasukkan masukan tambah akun siswa dengan email dengan format yang salah.		DUPL- BSO-7-06	Admin	Sistem	Black box
	Pengujian memasukkan masukan tambah akun siswa dengan konfirmasi password yang berbeda dengan password.		DUPL- BSO-7-07	Admin	Sistem	Black box
	Pengujian memasukkan masukan tambah akun siswa dengan email yang tidak terdaftar di profil siswa.		DUPL- BSO-7-08	Admin	Sistem	Black box
	Pengujian menekan tombol “Tambah” di modal tambah akun siswa		DUPL- BSO-7-09	Admin	Sistem	Black box
	Pengujian menekan tombol “Kembali” di modal tambah akun siswa		DUPL- BSO-7-10	Admin	Sistem	Black box
	Pengujian menekan ikon edit di kolom aksi akun siswa		DUPL- BSO-7-11	Admin	Sistem	Black box
	Pengujian menekan ikon hapus di kolom aksi akun siswa		DUPL- BSO-7-12	Admin	Sistem	Black box
	Pengujian menekan ikon edit di kolom status akun siswa		DUPL- BSO-7-13	Admin	Sistem	Black box

Kelas Uji	Butir Uji	Identifikasi		Aktor	Jenis Pengujian	Teknik Pengujian
		Use Case	DUPL			
	Pengujian menekan drop down status di modal ubah status akun siswa		DUPL- BSO-7-14	Admin	Sistem	<i>Black box</i>
	Pengujian menekan tombol “Ubah” di modal ubah status akun siswa		DUPL- BSO-7-15	Admin	Sistem	<i>Black box</i>
	Pengujian menekan tombol “kembali” di modal ubah status akun siswa		DUPL- BSO-7-16	Admin	Sistem	<i>Black box</i>
	Pengujian memilih pilihan “Ubah Status Akun” di kolom status siswa		DUPL- BSO-7-17	Admin	Sistem	<i>Black box</i>
	Pengujian memasukkan huruf di kolom <i>search</i> data siswa		DUPL- BSO-7-18	Admin	Sistem	<i>Black box</i>
	Pengujian menekan tombol “ <i>Previous</i> ” data siswa		DUPL- BSO-7-19	Admin	Sistem	<i>Black box</i>
	Pengujian menekan tombol “ <i>Next</i> ” data siswa			Admin	Sistem	<i>Black box</i>
	Pengujian menekan tombol “Tambah Akun” di halaman kelola akun instruktur		DUPL- BSO-7-20	Admin	Sistem	<i>Black box</i>
	Pengujian memasukkan masukan tambah akun Instruktur dengan email dengan format yang salah.		DUPL- BSO-7-21	Admin	Sistem	<i>Black box</i>
	Pengujian memasukkan masukan tambah akun Instruktur dengan konfirmasi password yang berbeda dengan password.		DUPL- BSO-7-22	Admin	Sistem	<i>Black box</i>
	Pengujian memasukkan masukan tambah akun instruktur dengan email yang tidak terdaftar di profil instruktur.		DUPL- BSO-7-23	Admin	Sistem	<i>Black box</i>
	Pengujian menekan ikon edit di kolom aksi akun instruktur		DUPL- BSO-7-24	Admin	Sistem	<i>Black box</i>
	Pengujian menekan ikon hapus di kolom aksi akun instruktur		DUPL- BSO-7-25	Admin	Sistem	<i>Black box</i>
	Pengujian menekan ikon edit di kolom status akun		DUPL- BSO-7-26	Admin	Sistem	<i>Black box</i>

Kelas Uji	Butir Uji	Identifikasi		Aktor	Jenis Pengujian	Teknik Pengujian
		Use Case	DUPL			
instruktur	Pengujian menekan drop down status di modal ubah status akun instruktur		DUPL- BSO-7-27	Admin	Sistem	<i>Black box</i>
	Pengujian menekan tombol “Ubah” di modal ubah status akun instruktur		DUPL- BSO-7-28	Admin	Sistem	<i>Black box</i>
	Pengujian menekan tombol “kembali” di modal ubah status akun instruktur		DUPL- BSO-7-29	Admin	Sistem	<i>Black box</i>
	Pengujian memilih pilihan “Ubah Status Akun” di kolom status instruktur		DUPL- BSO-7-30	Admin	Sistem	<i>Black box</i>
	Pengujian memasukkan huruf di kolom <i>search</i> data Instruktur		DUPL- BSO-7-31	Admin	Sistem	<i>Black box</i>
	Pengujian menekan tombol “ <i>Previous</i> ” data instruktur		DUPL- BSO-7-32	Admin	Sistem	<i>Black box</i>
	Pengujian menekan tombol “ <i>Next</i> ” data instruktur		DUPL- BSO-7-33	Admin	Sistem	<i>Black box</i>
	Pengujian menekan tombol “Tambah Akun” di halaman kelola akun admin		DUPL- BSO-7-34	Admin	Sistem	<i>Black box</i>
	Pengujian memasukkan masukan tambah akun admin dengan email dengan format yang salah.		DUPL- BSO-7-35	Admin	Sistem	<i>Black box</i>
	Pengujian memasukkan masukan tambah akun admin dengan konfirmasi password yang berbeda dengan password.		DUPL- BSO-7-36	Admin	Sistem	<i>Black box</i>
	Pengujian memasukkan masukan tambah akun admin dengan email yang tidak terdaftar di profil admin.		DUPL- BSO-7-37	Admin	Sistem	<i>Black box</i>
	Pengujian menekan ikon edit di kolom aksi akun admin		DUPL- BSO-7-38	Admin	Sistem	<i>Black box</i>
	Pengujian menekan ikon hapus di kolom aksi akun		DUPL- BSO-7-39	Admin	Sistem	<i>Black box</i>

Kelas Uji	Butir Uji	Identifikasi		Aktor	Jenis Pengujian	Teknik Pengujian
		Use Case	DUPL			
admin	Pengujian menekan ikon edit di kolom status akun admin		DUPL- BSO-7-40	Admin	Sistem	<i>Black box</i>
	Pengujian menekan drop down status di modal ubah status akun admin		DUPL- BSO-7-41	Admin	Sistem	<i>Black box</i>
	Pengujian menekan tombol “Ubah” di modal ubah status akun admin		DUPL- BSO-7-42	Admin	Sistem	<i>Black box</i>
	Pengujian menekan tombol “kembali” di modal ubah status akun admin		DUPL- BSO-7-43	Admin	Sistem	<i>Black box</i>
	Pengujian memilih pilihan “Ubah Status Akun” di kolom status admin		DUPL- BSO-7-44	Admin	Sistem	<i>Black box</i>
	Pengujian memasukkan huruf di kolom <i>search</i> data admin		DUPL- BSO-7-45	Admin	Sistem	<i>Black box</i>
	Pengujian menekan tombol “ <i>Previous</i> ” data admin		DUPL- BSO-7-46	Admin	Sistem	<i>Black box</i>
	Pengujian menekan tombol “ <i>Next</i> ” data admin		DUPL- BSO-7-47	Admin	Sistem	<i>Black box</i>

Tabel 5.2 Deskripsi dan Hasil Uji

Identifi-kasi	Deskripsi	Prosedur Pengujian	Masukkan	Keluaran yang Diharapkan	Kriteria Evaluasi Hasil	Hasil	Kesimpulan
DUPL-BSO-1-01	Pengujian memasukkan data masukan email terdaftar dan password yang sesuai	<ol style="list-style-type: none"> 1. Mengisi kolom email dengan email terdaftar. 2. Mengisi kolom password dengan password yang sesuai. 3. Menekan tombol “Masuk” 	Email terdaftar, password yang sesuai	Tampil halaman menu	-	Tampil halaman kelola menu	Diterima
DUPL-BSO-1-02	Pengujian memasukkan email terdaftar dengan password yang salah.	<ol style="list-style-type: none"> 1. Mengisi kolom email dengan email terdaftar. 2. Mengisi kolom password dengan password yang salah (bukan password tidak cocok dengan email) 3. Menekan tombol “Masuk” 	Email terdaftar, password yang salah	Tetap dihalaman login dengan pesan kesalahan “Password Salah!”	Pesan kesalahan “Password Salah!”	Tetap dihalaman login dengan pesan kesalahan “Password Salah!” dan tetap dihalaman login	Diterima
DUPL-BSO-1-03	Pengujian memasukkan email yang tidak terdaftar.	<ol style="list-style-type: none"> 1. Mengisi kolom email dengan email tidak terdaftar. 2. Mengisi kolom password dengan password benar. 3. Menekan tombol “Masuk” 	Email tidak terdaftar dan password	Tetap dihalaman login dengan pesan kesalahan “Email Tidak Tersedia!”	Pesan kesalahan “Email Tidak Tersedia!”	Tetap dihalaman login dengan pesan kesalahan “Email Tidak Tersedia!”	Diterima

Identifi-kasi	Deskripsi	Prosedur Pengujian	Masukkan	Keluaran yang Diharapkan	Kriteria Evaluasi Hasil	Hasil	Kesimpulan
DUPL-BSO-2-01	Menekan tombol keluar di halaman Administrasi	Menekan tombol “keluar”	Tidak ada	Kembali ke halaman login	-	Kembali ke halaman login	Diterima
DUPL-BSO-2-02	Menekan tombol keluar di halaman Menu	Menekan tombol “keluar”	Tidak ada	Kembali ke halaman login	-	Kembali ke halaman login	Diterima
DUPL-BSO-3-01	Pengujian menekan menu “Administrasi”	Menekan menu “Administrasi”	Tidak ada	Menampilkan halaman administrasi	-	Halaman Administrasi berhasil ditampilkan dengan dasahboard sebagai halaman awal	Diterima
DUPL-BSO-4-01	Pengujian menekan dropdown kelola kelas	Menekan dropdown menu kelola kelas	Tidak ada	Menampilkan pilihan kelas	-	Menampilkan pilihan kelas	Diterima
DUPL-BSO-4-02	Pengujian menekan pilihan kelas	1. Menekan dropdown menu kelola kelas 2. Menekan pilihan kelas	Tidak ada	Menampilkan halaman kelola kelas	-	Menampilkan halaman kelola kelas	Diterima
DUPL-BSO-4-03	Pengujian menekan tombol “Tambah Kelas”	1. Menekan dropdown menu kelola kelas 2. Menekan pilihan kelas 3. Menekan tombol “Tambah Kelas”	Tidak ada	Menampilkan modal Tambah Kelas	-	Menampilkan modal Tambah Kelas	Diterima

Identifi-kasi	Deskripsi	Prosedur Pengujian	Masukkan	Keluaran yang Diharapkan	Kriteria Evaluasi Hasil	Hasil	Kesimpulan
DUPL-BSO-4-04	Pengujian memasukkan data masukan kelas yang valid	<ol style="list-style-type: none"> 1. Menekan dropdown menu kelola kelas 2. Menekan pilihan kelas 3. Menekan tombol “Tambah Kelas” 4. Memasukkan nama kelas dan jumlah kuota yang tidak melebihi batas jumlah yang ditentukan. 	Nama kelas dan jumlah kuota	Data berhasil ditambah	Modal berhasil menambah data	Data berhasil kelas ditambah	Diterima
DUPL-BSO-4-05	Pengujian memasukkan data tambah kelas dengan nama kelas melebihi jumlah masukan yang telah ditentukan	<ol style="list-style-type: none"> 1. Menekan dropdown menu kelola kelas 2. Menekan pilihan kelas 3. Menekan tombol “Tambah Kelas” 4. Memasukkan nama kelas lebih dari 15 Karakter. 	Nama kelas	Menampilkan peringatan maksimal jumlah maskimal 15 karakter	Peringatan maksimal 15 Karakter	Menampilkan peringatan maksimal 15 karakter	Diterima
DUPL-BSO-4-06	Pengujian memasukkan data tambah kelas dengan jumlah kuota melebihi jumlah masukan yang telah ditentukan	<ol style="list-style-type: none"> 1. Menekan menu “Menekan dropdown menu kelola kelas 2. Menekan pilihan kelas 3. Menekan tombol “Tambah Kelas” 	Jumlah kuota	Menampilkan peringatan jumlah maksimal 2 angka	Peringatan maksimal 2 karakter angka	Menampilkan peringatan maksimal 2 karakter angka	Diterima

Identifi-kasi	Deskripsi	Prosedur Pengujian	Masukkan	Keluaran yang Diharapkan	Kriteria Evaluasi Hasil	Hasil	Kesimpulan
		4. Memasukkan jumlah kuota lebih dari 2 angka yaitu 100					
DUPL-BSO-4-07	Pengujian memasukkan data tambah kelas dengan jumlah kuota dengan karakter selain angka seperti huruf dan tanda baca.	1. Menekan menu “Menekan dropdown menu kelola kelas 2. Menekan pilihan kelas 3. Menekan tombol “Tambah Kelas” 4. Memasukkan jumlah kuota dengan menekan huruf dan karakter lain seperti tanda baca. 5. Menekan tombol “Tambah”	Jumlah Kuota	Menampilkan alert <i>“Please enter a number”</i>	Alert <i>“Please enter a number”</i>	Menampilkan alert <i>“Please enter a number”</i>	Diterima
DUPL-BSO-4-08	Pengujian menekan tombol “Tambah” di modal “Tambah Kelas”	1. Menekan menu “Menekan dropdown menu kelola kelas 2. Menekan pilihan kelas 3. Menekan tombol “Tambah Kelas” 4. Menekan tombol “Tambah”	Tidak ada	Peringatan kesalahan atau modal konfirmasi	Muncul peringatan kesalahan atau modal konfirmasi	Peringatan kesalahan atau modal konfirmasi	Diterima

Identifi-kasi	Deskripsi	Prosedur Pengujian	Masukkan	Keluaran yang Diharapkan	Kriteria Evaluasi Hasil	Hasil	Kesimpulan
DUPL-BSO-4-9	Pengujian menekan tombol “Kembali” di modal “Tambah Kelas”	1. Menekan menu “Menekan dropdown menu kelola kelas 2. Menekan pilihan kelas 3. Menekan tombol “Tambah Kelas” 4. Menekan tombol “Kembali”	Tidak ada	Modal tambah kelas ditutup	-	Menutup modal tambah kelas	Diterima
DUPL-BSO-4-10	Menekan ikon <i>search</i> di daftar siswa	1. Menekan menu “Menekan dropdown menu kelola kelas 2. Menekan pilihan kelas 3. Menekan ikon <i>search</i> di daftar siswa	Tidak ada	Menampilkan daftar siswa	-	Menampilkan halaman detail kelas yang berisi daftar siswa	Diterima
DUPL-BSO-4-11	Pengujian menekan ikon edit	1. Menekan menu “Menekan dropdown menu kelola kelas 2. Menekan pilihan kelas 3. Menekan ikon “edit”	Tidak ada	Menampilkan pop up edit kelas	-	Menampilkan pop up edit kelas	Diterima
DUPL-BSO-4-12	Pengujian menekan ikon hapus	1. Menekan menu “Menekan dropdown menu kelola kelas 2. Menekan pilihan kelas	Tidak ada	Menghapus data kelas yang tidak memiliki siswa dan data kelas terhapus	Pop up gagal hapus dan data kelas terhapus	Menghapus data kelas yang tidak memiliki siswa dan menampilkan modal gagal	Diterima

Identifi-kasi	Deskripsi	Prosedur Pengujian	Masukkan	Keluaran yang Diharapkan	Kriteria Evaluasi Hasil	Hasil	Kesimpulan
		3. Menekan ikon “hapus”				hapus untuk kelas dengan siswa terdaftar	
DUPL-BSO-4-13	Pengujian menekan <i>dropdown</i> ins di modal ubah kelas	1. Menekan menu “Menekan dropdown menu kelola kelas 2. Menekan pilihan kelas 3. Menekan ikon “edit” 4. Menekan <i>dropdown</i> ins	Tidak ada	<i>Dropdown</i> berfungsi dengan baik	-	Menampilkan pilihan ins	Diterima
DUPL-BSO-4-14	Pengujian memasukkan data ubah kelas dengan dengan ins yang sudah di menjadi pengajar di kelas lain	1. Menekan menu “Menekan dropdown menu kelola kelas 2. Menekan pilihan kelas 3. Menekan ikon “edit” 4. Menekan <i>dropdown</i> ins 5. Memilih pilihan ins yang sudah ada di tabel	ins	Ins tidak berhasil ditambahkan	modal gagal menambah instruktur dengan pesan error instruktur telah mengajak dikelas lain.	Menampilkan modal gagal menambah instruktur	Diterima
DUPL-BSO-4-15	Pengujian memasukkan data ubah kelas yang valid	1. Menekan menu “Menekan dropdown menu kelola kelas 2. Menekan pilihan kelas 3. Menekan ikon “edit”	Tidak ada	Berhasil mengubah data	modal sukses ubah data	Menampilkan modal sukses ubah data	Diterima

Identifi-kasi	Deskripsi	Prosedur Pengujian	Masukkan	Keluaran yang Diharapkan	Kriteria Evaluasi Hasil	Hasil	Kesimpulan
		4. Menekan <i>dropdown</i> ins 5. Memilih pilihan ins yang belum menjadi pengajar di kelas lain					
DUPL-BSO-4-16	Pengujian memasukkan karakter di kolom <i>search</i>	1. Menekan dropdown menu kelola kelas 2. Menekan pilihan kelas 3. Memasukkan karakter di kolom search	karakter	Tabel menampilkan hasil real time sesuai karakter yang dimasukkan	No matching records found atau hasil sesuai karakter yang di search	Tabel menampilkan hasil real time sesuai karakter yang dimasukkan	Diterima
DUPL-BSO-4-17	Pengujian menekan tombol “Previous”	1. Menekan dropdown menu kelola kelas 2. Menekan pilihan kelas 3. Menekan tombol “Previous”	Tidak ada	Jika berasa di page 2 dan seterusnya maka akan menampilkan halaman sebelumnya.	Tampil halaman sebelumnya	Tombol berfungsi dengan baik	Diterima
DUPL-BSO-4-20	Pengujian menekan tombol “Next”	1. Menekan dropdown menu kelola kelas 2. Menekan pilihan kelas 3. Menekan tombol “Next”	Tidak ada	Jika berasa di page 1 dan bukan halaman terakhir maka akan menampilkan	Tampil halaman berikutnya	Tombol berfungsi dengan baik	Diterima

Identifi-kasi	Deskripsi	Prosedur Pengujian	Masukkan	Keluaran yang Diharapkan	Kriteria Evaluasi Hasil	Hasil	Kesimpulan
				halaman berikutnya.			
DUPL-BSO-5-01	Pengujian menekan dropdown kelola bidang	Menekan <i>drop down</i> kelola bidang	Tidak ada	Menampilkan pilihan kelola bidang	-	Menampilkan bidang	Diterima
DUPL-BSO-5-02	Pengujian menekan pilihan “Bidang”	1. Menekan <i>drop down</i> kelola bidang 2. Menekan pilihan “Bidang”	Tidak ada	Menampilkan halaman kelola bidang	-	Menampilkan halaman kelola bidang	Diterima
DUPL-BSO-5-03	Pengujian menekan tombol “Tambah Bidang”	1. Menekan <i>drop down</i> kelola bidang 2. Menekan pilihan “Bidang ” 3. Menekan tombol “Tambah Bidang”	Tidak ada	Menampilkan modal Tambah Bidang	-	Menampilkan modal Tambah Bidang	Diterima
DUPL-BSO-5-04	Pengujian memasukkan data masukan tambah bidang yang valid	1. Menekan <i>drop down</i> kelola bidang 2. Menekan pilihan “Bidang ” 3. Menekan tombol “Tambah Bidang” 4. Memasukkan nama bidang yang valid dan tidak melebihi 20 karakter	Nama bidang	Bidang baru	-	Menampilkan modal sukses bidang berhasil di tambahkan ”	Diterima
DUPL-BSO-5-05	Pengujian memasukkan data masukan dengan	1. Menekan <i>drop down</i> kelola bidang	Nama bidang	Menampilkan peringatan	Peringatan Maksimal	Menampilkan peringatan	Diterima

Identifi-kasi	Deskripsi	Prosedur Pengujian	Masukkan	Keluaran yang Diharapkan	Kriteria Evaluasi Hasil	Hasil	Kesimpulan
	mengisi nama bidang melebihi jumlah masukan yang telah ditentukan	2. Menekan pilihan “Bidang” 3. Menekan tombol “Tambah Bidang” 4. Memasukkan nama bidang melebihi 20 karakter		jumlah maksimal	20 Karakter	maksimal 20 karakter	
DUPL-BSO-5-06	Pengujian memasukkan data masukan dengan mengisi nama bidang dengan nama bidang yang sudah terdaftar di tabel bidang	1. Menekan <i>drop down</i> kelola bidang 2. Menekan pilihan “Bidang” 3. Menekan tombol “Tambah Bidang” 4. Memasukkan nama bidang dengan nama bidang yang sudah terdaftar di tabel bidang dan memilih “benar” atau “salah” 5. Menekan tombol “Tambah”	Nama bidang	Menangani duplikasi data	Modal gagal berisi Bidang sudah Tersedia	Menampilkan modal gagal menambahkan bidang	Diterima
DUPL-BSO-5-07	Pengujian menekan ikon edit	1. Menekan <i>drop down</i> kelola bidang 2. Menekan pilihan “Bidang” 3. Menekan ikon edit di kolom aksi	-	Menampilkan modal ubah bidang	-	Menampilkan modal ubah bidang	Diterima

Identifi-kasi	Deskripsi	Prosedur Pengujian	Masukkan	Keluaran yang Diharapkan	Kriteria Evaluasi Hasil	Hasil	Kesimpulan
DUPL-BSO-5-08	Pengujian memasukkan data masukan ubah bidang dengan nama bidang yang valid	<ol style="list-style-type: none"> 1. Menekan <i>drop down</i> kelola bidang 2. Menekan pilihan “Bidang” 3. Menekan ikon edit di kolom aksi 4. Memasukkan nama bidang dengan nama yang belum tersedia di daftar dan tidak lebih dari 20 karakter 	Nama bidang	Mengubah nama bidang	-	Menampilkan modal berhasil mengubah bidang	Diterima
DUPL-BSO-5-09	Pengujian menekan tombol “kembali” di modal “Ubah bidang”	<ol style="list-style-type: none"> 1. Menekan <i>drop down</i> kelola bidang 2. Menekan pilihan “Bidang” 3. Menekan ikon edit di kolom aksi 4. Menekan tombol “kembali” 	Tidak ada	Menutup modal ubah bidang	-	Menutup modal ubah bidang	Diterima
DUPL-BSO-5-10	Pengujian menekan ikon hapus	<ol style="list-style-type: none"> 1. Menekan <i>drop down</i> kelola bidang 2. Menekan pilihan “Bidang” 3. Menekan ikon hapus 	Tidak ada	Menghapus data bidang yang tidak memiliki siswa terdaftar dan modal gagal untuk bidang dengan siswa terdaftar	-	Menghapus data bidang yang tidak memiliki siswa terdaftar dan modal gagal untuk bidang dengan siswa terdaftar	Diterima

Identifi-kasi	Deskripsi	Prosedur Pengujian	Masukkan	Keluaran yang Diharapkan	Kriteria Evaluasi Hasil	Hasil	Kesimpulan
				dengan siswa terdaftar			
DUPL-BSO-5-11	Pengujian memasukkan huruf di kolom <i>search</i>	<ol style="list-style-type: none"> 1. Menekan dropdown menu kelola bidang 2. Menekan pilihan bidang 3. Memasukkan karakter di kolom search 	karakter	Tabel menampilkan hasil real time sesuai karakter yang dimasukkan	<i>No matching records found</i> atau hasil sesuai karakter yang di search	Tabel menampilkan hasil real time sesuai karakter yang dimasukkan	Diterima
DUPL-BSO-5-12	Pengujian menekan tombol “ <i>Previous</i> ”	<ol style="list-style-type: none"> 1. Menekan dropdown menu kelola bidang 2. Menekan pilihan bidang 3. Menekan tombol “<i>Previous</i>” 	Tidak ada	Jika berada di page 2 dan seterusnya maka akan menampilkan halaman sebelumnya.	Tampil halaman sebelumnya	Tombol berfungsi dengan baik	Diterima
DUPL-BSO-5-13	Pengujian menekan tombol “ <i>Next</i> ”	<ol style="list-style-type: none"> 1. Menekan dropdown menu kelola bidang 2. Menekan pilihan bidang 3. Menekan tombol “<i>Next</i>” 	Tidak ada	Jika berada di page 1 dan bukan halaman terakhir maka akan menampilkan halaman berikutnya.	Tampil halaman berikutnya	Tombol berfungsi dengan baik	Diterima

Identifi-kasi	Deskripsi	Prosedur Pengujian	Masukkan	Keluaran yang Diharapkan	Kriteria Evaluasi Hasil	Hasil	Kesimpulan
DUPL-BSO-6-01	Pengujian menekan <i>dropdown</i> kelola profil	Menekan <i>drop down</i> kelola profil	Tidak ada	Menampilkan pilihan profil	-	Menampilkan pilihan profil	Diterima
DUPL-BSO-6-02	Pengujian menekan pilihan “Siswa”	1. Menekan <i>drop down</i> kelola profil 2. Menekan pilihan profile “Siswa”	Tidak ada	Menampilkan halaman profile siswa	-	Menampilkan halaman profile siswa	Diterima
DUPL-BSO-6-03	Pengujian menekan pilihan “Instruktur”	1. Menekan <i>drop down</i> kelola profil 2. Menekan pilihan profile “Instruktur”	Tidak ada	Menampilkan halaman profile Instruktur	-	Menampilkan halaman profile Instruktur	Diterima
DUPL-BSO-6-04	Pengujian menekan pilihan “Admin”	1. Menekan <i>drop down</i> kelola profil 2. Menekan pilihan profile “Admin”	Tidak ada	Menampilkan halaman profile Admin	-	Menampilkan halaman profile Admin	Diterima
DUPL-BSO-6-05	Pengujian menekan tombol “Tambah Siswa”	1. Menekan <i>drop down</i> kelola profil 2. Menekan pilihan profile “Siswa” 3. Menekan tombol “Tambah Siswa”	Tidak ada	Menampilkan modal tambah siswa	-	Menampilkan modal tambah siswa	Diterima
DUPL-BSO-6-06	Pengujian menekan <i>dropdown</i> “Bidang Minat” dimodal Tambah Siswa	1. Menekan <i>drop down</i> kelola profil 2. Menekan pilihan profile “Siswa” 3. Menekan tombol “Tambah Siswa”	Tidak ada	Dropdown berfungsi	-	Menampilkan pilihan bidang minat	Diterima

Identifi-kasi	Deskripsi	Prosedur Pengujian	Masukkan	Keluaran yang Diharapkan	Kriteria Evaluasi Hasil	Hasil	Kesimpulan
		4. Menekan <i>drop down</i> Bidang Minat					
DUPL-BSO-6-07	Pengujian menekan dropd down “Kelas” dimodal Tambah Siswa	1. Menekan <i>drop down</i> kelola profil 2. Menekan pilihan profile “Siswa” 3. Menekan tombol “Tambah Siswa” 4. Menekan <i>drop down</i> kelas	Tidak ada	Dropdown berfungsi	-	Menampilkan pilihan kelas	Diterima
DUPL-BSO-6-08	Pengujian menekan dropd down “Status” dimodal Tambah Siswa	1. Menekan <i>drop down</i> kelola profil 2. Menekan pilihan profile “Siswa” 3. Menekan tombol “Tambah Siswa” 4. Menekan <i>drop down</i> status	Tidak ada	Dropdown berfungsi	-	Menampilkan pilihan bidang status	Diterima
DUPL-BSO-6-09	Pengujian menekan ikon kalender dikolom “Tanggal Masuk” di modal Tambah Siswa	1. Menekan <i>drop down</i> kelola profil 2. Menekan pilihan profile “Siswa” 3. Menekan tombol “Tambah Siswa” 4. Menekan ikon kalender di kolom tanggal masuk	Tidak ada	Dropdown berfungsi	-	Menampilkan kalender	Diterima

Identifi-kasi	Deskripsi	Prosedur Pengujian	Masukkan	Keluaran yang Diharapkan	Kriteria Evaluasi Hasil	Hasil	Kesimpulan
DUPL-BSO-6-10	Pengujian memasukkan data masukan tambah siswa yang valid dan belum terdaftar	<ol style="list-style-type: none"> Menekan <i>drop down</i> kelola profil Menekan pilihan profile “Siswa” Menekan tombol “Tambah Siswa” Memasukkan data siswa yang valid dan belum terdaftar 	Nama siswa, email, no hp, alamat, bidang minat, kelas, tanggal masuk dan status	Data siswa baru disimpan ke database	-	Menampilkan modal berhasil menambahkan data siswa	Diterima
DUPL-BSO-6-11	Pengujian memasukkan data masukan tambah siswa dengan data siswa yang sudah terdaftar di tabel.	<ol style="list-style-type: none"> Menekan <i>drop down</i> kelola profil Menekan pilihan profile “Siswa” Menekan tombol “Tambah Siswa” Memasukkan data siswa yang valid dan sudah terdaftar terdaftar 	Nama siswa, email, no hp, alamat, bidang minat, kelas, tanggal masuk dan status	Menangani duplikasi dengan pesan error	Data siswa sudah terdaftar	Menampilkan modal gagal menambahkan data siswa	Diterima
DUPL-BSO-6-12	Pengujian memasukkan data masukan tambah siswa dengan memasukkan email siswa tanpa format @.	<ol style="list-style-type: none"> Menekan <i>drop down</i> kelola profil Menekan pilihan profile “Siswa” Menekan tombol “Tambah Siswa” Memasukkan email siswa dengan contoh 	Nama siswa, email tanpa @, no hp, alamat, bidang minat, kelas, tanggal masuk dan status	Tidak berhasil menambah siswa	“Email dalam format yang tidak valid”	Menampilkan modal gagal menambahkan data siswa	Diterima

Identifi-kasi	Deskripsi	Prosedur Pengujian	Masukkan	Keluaran yang Diharapkan	Kriteria Evaluasi Hasil	Hasil	Kesimpulan
		masukan siswa@email.com					
DUPL-BSO-6-13	Pengujian memasukkan data masukan tambah siswa dengan No Hp dalam format tidak valid	1. Menekan <i>drop down</i> kelola profil 2. Menekan pilihan profile “Siswa” 3. Menekan tombol “Tambah Siswa” 4. Memasukkan no hp dengan mengandung huruf dan karakter selain angka dan +	Nama siswa, email tanpa @, no hp, alamat, bidang minat, kelas, tanggal masuk dan status	Tidak berhasil menambah siswa	“ No hp dalam format yang tidak valid”	Menampilkan modal gagal menambahkan data siswa	Diterima
DUPL-BSO-6-14	Pengujian memasukkan data masukan tambah siswa dengan mengosongkan 1 kolom isian	1. Menekan <i>drop down</i> kelola profil 2. Menekan pilihan profile “Siswa” 3. Menekan tombol “Tambah Siswa” 4. Mengosongkan alamat	Nama siswa, email tanpa @, no hp, alamat, bidang minat, kelas, tanggal masuk dan status	Tidak berhasil menambah siswa	Alert “please fill out the field”	Menampilkan alert di kolom alamat	Diterima
DUPL-BSO-6-15	Pengujian memasukkan data masukan tambah siswa dengan mengosongkan semua kolom	1. Menekan <i>drop down</i> kelola profil 2. Menekan pilihan profile “Siswa” 3. Menekan tombol “Tambah Siswa” 4. Mengosongkan semua isian	Tidak ada	Tidak berhasil menambah siswa	Alert “please fill out the field”	Menampilkan alert di kolom	Diterima

Identifi-kasi	Deskripsi	Prosedur Pengujian	Masukkan	Keluaran yang Diharapkan	Kriteria Evaluasi Hasil	Hasil	Kesimpulan
DUPL-BSO-6-16	Pengujian menekan tombol “Tambah” di modal Tambah Siswa	1. Menekan <i>drop down</i> kelola profil 2. Menekan pilihan profile “Siswa” 3. Menekan tombol “Tambah Siswa” 4. Menekan tombol “Tambah”	Tidak ada	Menampilkan alert untuk memastikan kolom tidak kosong	<i>Alert “please fill out the field”</i>	Menampilkan alert di kolom	Diterima
DUPL-BSO-6-17	Pengujian menekan tombol “Kembali” di modal Tambah Siswa	1. Menekan <i>drop down</i> kelola profil 2. Menekan pilihan profile “Siswa” 3. Menekan tombol “Tambah Siswa” 4. Menekan tombol “kembali”	Tidak ada	Menutup modal tambah siswa	-	Menutup modal tambah siswa	Diterima
DUPL-BSO-6-18	Pengujian menekan ikon edit	1. Menekan <i>drop down</i> kelola profil 2. Menekan pilihan profile “Siswa” 3. Menekan ikon edit	Tidak ada	Menampilkan modal edit	-	Menampilkan modal edit	Diterima
DUPL-BSO-6-19	Pengujian memasukkan data masukan ubah siswa dengan email siswa yang sudah terdaftar di tabel siswa	1. Menekan <i>drop down</i> kelola profil 2. Menekan pilihan profile “Siswa” 3. Menekan ikon edit 4. Memasukkan email dengan email siswa	Nama siswa, email tanpa @, no hp, alamat, bidang minat, kelas, tanggal	Data siswa tidak diubah	“Email sudah tersedia”	Menampilkan modal gagal	Diterima

Identifi-kasi	Deskripsi	Prosedur Pengujian	Masukkan	Keluaran yang Diharapkan	Kriteria Evaluasi Hasil	Hasil	Kesimpulan
	sementara kolom lain dengan data baru	yang sudah terdaftar sementara kolom lain dengan data baru.	masuk dan status				
DUPL-BSO-6-20	Pengujian memasukkan data masukan ubah siswa dengan pilihan status “Lulus” tetapi tidak mengisi kolom tanggal lulus.	<ol style="list-style-type: none"> Menekan <i>drop down</i> kelola profil Menekan pilihan profile “Siswa” Menekan ikon edit Memilih status lulus Mengosongkan tanggal lulus Menekan tombol ubah 	Nama siswa, email tanpa @, no hp, alamat, bidang minat, kelas, tanggal masuk dan status	Data siswa tidak diubah	<i>Alert “please fill out the field”</i>	Menampilkan alert kolom harus diisi	Diterima
DUPL-BSO-6-21	Pengujian menekan tombol “Kembali” di modal ubah siswa	<ol style="list-style-type: none"> Menekan <i>drop down</i> kelola profil Menekan pilihan profile “Siswa” Menekan ikon edit Menekan tombol kembali 	Tidak ada	Menutup modal ubah siswa	-	Menutup modal ubah siswa	
DUPL-BSO-6-22	Pengujian menekan ikon hapus di tabel data siswa	<ol style="list-style-type: none"> Menekan <i>drop down</i> kelola profil Menekan pilihan profile “Siswa” Menekan ikon hapus 	Tidak ada	Menghapus siswa	berhasil menghapus data siswa	Menampilkan modal berhasil hapus data	

Identifi-kasi	Deskripsi	Prosedur Pengujian	Masukkan	Keluaran yang Diharapkan	Kriteria Evaluasi Hasil	Hasil	Kesimpulan
DUPL-BSO-6-26	Pengujian menekan tombol “Tambah Instruktur”	5.					