

PENERAPAN METODE CANNY DALAM KOREKSI LEMBAR JAWABAN KOMPUTER UNTUK TRY OUT

LAPORAN TUGAS AKHIR

Oleh :

Ela Erminawati 3311401022

Disusun untuk memenuhi syarat kelulusan Program Diploma DIII



**PROGRAM STUDI TEKNIK INFORMATIKA
POLITEKNIK NEGERI BATAM
BATAM
2017**

HALAMAN PENGESAHAN

Penerapan Metode Canny dalam Koreksi Lembar Jawaban Komputer untuk Try Out

Oleh :

Ela Erminawati (3311401022)

Diajukan dan disahkan sebagai laporan Tugas Akhir II

PROGRAM STUDI DIPLOMA 3 TEKNIK INFORMATIKA
POLITEKNIK NEGERI BATAM

Batam, 15 Maret 2017

Disetujui oleh,

Pembimbing I,

Pembimbing II,

Mira Chandra Kirana, S.T., M.T.

NIP 197905302014042002

Sartikha, S.ST

NIK. 113115

HALAMAN PERNYATAAN

Dengan ini, saya:

NIM : 3311401022

Nama : Ela Erminawati

adalah mahasiswa Teknik Informatika Politeknik Batam yang menyatakan bahwa tugas akhir dengan judul:

PENERAPAN METODE CANNY DALAM KOREKSI LEMBAR JAWABAN KOMPUTER UNTUK TRY OUT

disusun dengan:

1. tidak melakukan plagiat terhadap naskah karya orang lain
2. tidak melakukan pemalsuan data
3. tidak menggunakan karya orang lain tanpa menyebut sumber asli atau tanpa izin pemilik

Jika kemudian terbukti terjadi pelanggaran terhadap pernyataan di atas, maka saya bersedia menerima sanksi apapun termasuk pencabutan gelar akademik.

Lembar pernyataan ini juga memberikan hak kepada Politeknik Negeri Batam untuk mempergunakan, mendistribusikan ataupun memproduksi ulang seluruh hasil tugas akhir ini.

Batam, 15 Maret 2017

Ela Erminawati
NIM 3311401022

HALAMAN PERSEMBAHAN

Karya tulis ini penulis persembahkan untuk.

1. Kedua orang tua saya yang selama ini telah memberikan dukungan, semangat serta motivasi untuk saya.
2. Kakak saya yang tak pernah bosan menasehati saya.
3. Dosen pembimbing yang selalu memberikan bimbingan dan arahan selama tugas akhir ini.
4. Sahabat-sahabat serta teman-teman saya yang telah membantu dan memberikan dorongan kepada saya.

KATA PENGANTAR

Puji syukur penulis panjatkan kehadiran Tuhan Yang Maha Esa yang telah melimpahkan rahmat, taufik serta hidayah-Nya sehingga laporan Tugas Akhir yang berjudul “PENERAPAN METODE CANNY DALAM KOREKSI LEMBAR JAWABAN KOMPUTER UNTUK TRY OUT” dapat terselesaikan.

Tujuan disusunnya Tugas Akhir ini adalah sebagai syarat kelulusan yang ada di Politeknik Negeri Batam dan untuk memenuhi mata kuliah Tugas Akhir pada program studi Informatika Diploma III. Dalam penyusunan laporan ini penulis banyak mendapat saran, dorongan, bimbingan dari berbagai pihak. Oleh karena itu penulis mengucapkan terimakasih kepada :

1. Tuhan Yang Maha Esa, yang senantiasa memberikan limpahan taufik dan hidayah-Nya yang selalu mengiringi penulis.
2. Ibu Mira Chandra Kirana, S.T., M.T. dan Ibu Sartikha, S.ST selaku dosen pembimbing Tugas Akhir yang telah meluangkan waktu untuk memberikan bimbingan serta arahan selama penyusunan tugas akhir ini.
3. SMA Negeri 2 Batam sebagai lokasi studi kasus yang telah memberikan banyak informasi untuk mendukung tugas akhir ini .
4. Semua teman-teman dan pihak yang tidak mungkin penulis sebutkan satu persatu yang telah banyak membantu sehingga Tugas Akhir ini dapat diselesaikan dengan tepat waktu.

Penulis menyadari bahwa laporan ini jauh dari kata sempurna sehingga masih ada banyak kekurangan dalam laporan tugas akhir ini oleh karena itu kritik dan saran yang membangun sangat penulis harapkan untuk membantu penelitian selanjutnya. Demikian kata pengantar ini penulis buat, semoga dapat bermanfaat, khususnya bagi penulis dan bagi pembaca pada umumnya.

Batam, 15 Maret 2017

Penulis

ABSTRAK

PENERAPAN METODE CANNY DALAM KOREKSI LEMBAR JAWABAN KOMPUTER UNTUK TRY OUT

Try out merupakan suatu kegiatan untuk simulasi ujian nasional yang biasanya diselenggarakan disetiap sekolah. Pelaksanaan kegiatan ini biasanya dengan mengisi lembar jawaban komputer. Untuk koreksi LJK ini menggunakan scanner khusus dengan harga yang cukup tinggi. Sehingga banyak pihak sekolah memilih untuk koreksi manual guna menekan pengeluaran dana sekolah. Koreksi manual ini yang membuat pihak sekolah memerlukan waktu yang cukup lama dalam koreksi sehingga pengumuman hasil *try out* siswa juga tidak cepat.

Untuk itu dari masalah yang telah diuraikan penulis memberikan sebuah solusi untuk mengatasi masalah yang ada dengan memanfaatkan teknologi *image processing* untuk mengoreksi LJK siswa dengan menerapkan metode canny. Teknologi ini tidak membutuhkan biaya yang besar karena hanya memanfaatkan citra digital hasil scanner biasa. Dengan adanya aplikasi yang menerapkan teknologi ini untuk koreksi LJK maka pihak sekolah tidak perlu mengeluarkan biaya yang tinggi. Aplikasi yang dibuat mempunyai fitur utama untuk mengoreksi LJK dengan salah satu metode deteksi tepi yang optimal sehingga tingkat kesalahan dalam koreksi dapat diminimalisir. Metode canny juga merupakan metode deteksi tepi yang dapat dikolaborasikan dengan metode lain seperti sobel, *hough transformation* dan lain – lain. Hal ini dikarenakan deteksi canny yang dapat mendeteksi tepi objek secara optimal. Dan operator yang digunakan adalah operator sobel. *Hough circle / Hough Transformation* pada penelitian ini digunakan untuk mendeteksi objek khusus lingkaran tanpa mengkhawatirkan noise / derau dari citra LJK. Dengan begitu akurasi untuk mendeteksi objek lingkaran dapat ditingkatkan. Sehingga aplikasi yang dibangun pada penelitian ini dapat mengelola data *try out* siswa dan dapat mendeteksi informasi yang terdapat pada citra LJK.

Kata kunci: *koreksi, LJK, sekolah, aplikasi*

ABSTRACT

APPLICATION METHODS OF CORRECTION CANNY ANSWER SHEET COMPUTER TO TRY OUT

Try out is an activity for the simulation of a national exam that is usually held in every school. Implementation of this activity typically by filling in a computer answer sheet. For this LJK correction using a special scanner at a high enough price. So many schools opt for manual correction in order to reduce the expenditure of funds the school. This manual correction that makes the school requires considerable time in corrections so that the announcement of the results of the students did not try out fast.

For that of the problems that have been outlined authors provide a solution to overcome the existing problems by utilizing image processing technology to correct LJK students by applying Canny. This technology does not require a high cost because only utilize digital image scanner results outstanding. With the application of applying this technology to the correction LJK then the school did not need to spend a high cost. Applications are made to have the main feature to correct LJK with one edge detection method is optimized so that the error rate in the correction can be minimized. Method is also a canny edge detection methods that can be collaborated with other methods such as Sobel, hough transformation and others - others. This is because the canny detection that can detect the edges of objects optimally. And operators used are Sobel operator. Hough circle / Hough Transformation in this study is used to detect specific objects circle without worrying about noise / noise from the image LJK. With so accuracy for detecting circular object can be improved. So that applications built on this research can manage data try out students and can detect the information contained in the image LJK.

Keywords: *correction, LJK, schools, application*

DAFTAR ISI

HALAMAN PENGESAHAN.....	ii
HALAMAN PERNYATAAN	iii
KATA PENGANTAR	v
ABSTRAK	vi
ABSTRACT	vii
DAFTAR GAMBAR	ix
BAB I PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	2
1.3 Batasan Masalah	2
1.4 Tujuan Penelitian.....	3
1.5 Manfaat Penelitian.....	3
1.6 Tinjauan Pustaka	4
1.7 Sistematika Penulisan.....	5
BAB II LANDASAN TEORI	6
2.1 Profil SMA Negeri 2 Batam	6
2.2 Sistem Pendukung Keputusan (SPK)	6
2.3 Image Processing	7
2.4 Metode Canny.....	7
2.5 Transformasi Hough	8
2.6 Pemrograman Matlab.....	9
2.7 Basis Data	10
2.8 MySQL	10
BAB III PERANCANGAN SISTEM	12
3.1 Deskripsi Umum Sistem.....	12
3.2 Spesifikasi Kebutuhan Fungsional dan Non Fungsional.....	13
3.3 Spesifikasi Kebutuhan Sistem	15
3.4 Metodologi Penelitian	16
3.5 Teknik Pengumpulan Data	17
3.6 Teknik Pengolahan dan Analisis Data.....	18
3.4 Penyajian Data.....	26
3.7 Analisis Sistem	27
3.8 Perancangan Antarmuka.....	91
BAB IV IMPLEMENTASI DAN PENGUJIAN	108
4.1 Implementasi Basis Data	108
4.2 Implementasi Antarmuka	111
4.3 Pengujian	125
BAB V KESIMPULAN DAN SARAN.....	126
5.1 Kesimpulan.....	126
5.2 Saran	126
DAFTAR PUSTAKA	127
LAMPIRAN I HASIL PENGUJIAN.....	129

DAFTAR GAMBAR

Gambar 2. 1 Rumus Persamaan Lingkaran.....	9
Gambar 2. 2 Ilustrasi Koordinat.....	9
Gambar 3. 1 Deskripsi Umum Sistem	12
Gambar 3. 2 Metodologi Penelitian	16
Gambar 3. 3 Hasil Membaca Image.....	18
Gambar 3. 4 Hasil Citra Grayscale	19
Gambar 3. 5 Hasil Citra Konversi ke Biner	19
Gambar 3. 6 Formula <i>Gaussian Mask</i>	20
Gambar 3. 7 Formula Konvolusi <i>Gaussian Mask</i>	20
Gambar 3. 8 Hasil Konvolusi <i>Gaussian Mask</i>	21
Gambar 3. 9 Operator Sobel.....	21
Gambar 3. 10 Contoh Matrik	22
Gambar 3. 11 Formula Gradient Arah Vertikal	22
Gambar 3. 12 Hasil Gradient Arah Vertikal	22
Gambar 3. 13 Formula Gradient Arah Horizontal	23
Gambar 3. 14 Hasil Gradient Arah Horizontal	23
Gambar 3. 15 Formula Magnitudo.....	23
Gambar 3. 16 Hasil Penerapan Magnitudo	24
Gambar 3. 17 Hasil Deteksi Tepi Canny	24
Gambar 3. 18 Contoh Penerapan <i>Centroid</i>	25
Gambar 3. 19 Penyajian Data Akhir	26
Gambar 3. 20 Use case diagram.....	27
Gambar 3. 21 Activity Diagram <i>Login</i>	46
Gambar 3. 22 Activity Diagram Mengubah <i>Password Admin</i>	47
Gambar 3. 23 Activity Diagram Menambah Admin.....	48
Gambar 3. 24 Activity Diagram Mereset <i>Password</i>	49
Gambar 3. 25 Activity Diagram Mencari Data Admin.....	50
Gambar 3. 26 Activity Diagram Menambah Data Peserta.....	51
Gambar 3. 27 Activity Diagram Mengubah Data Peserta.....	52
Gambar 3. 28 Activity Diagram Mencari Data Peserta	53
Gambar 3. 29 Activity Diagram Menambah Data Paket Soal	54
Gambar 3. 30 Activity Diagram Menghapus Data Paket Soal.....	55
Gambar 3. 31 Activity Diagram Mencari Data Paket Soal	56
Gambar 3. 32 Activity Diagram Memasukkan Citra LJK	57
Gambar 3. 33 Activity Diagram Mendeteksi Informasi LJK	58
Gambar 3. 34 Activity Diagram Memeriksa Jawaban <i>Try Out</i> Siswa.....	59
Gambar 3. 35 Activity Diagram Menyimpan Data Try Out Siswa.....	60
Gambar 3. 36 Activity Diagram Mencari Hasil Try Out Siswa.....	61
Gambar 3. 37 Activity Diagram Mencetak Data Hasil <i>Try Out</i>	62
Gambar 3. 38 Activity Diagram Mencari Jumlah Siswa Salah	63
Gambar 3. 39 Activity Diagram Menampilkan Grafik Jumlah Siswa salah.....	64
Gambar 3. 40 Activity Diagram Mengeksport Data Jumlah Siswa Salah	65
Gambar 3. 41 Activity Diagram Menambah Data Mapel	66

Gambar 3. 42 Activity Diagram Mengubah Data Mapel.....	67
Gambar 3. 43 Activity Diagram Mencari Data Mata Pelajaran.....	68
Gambar 3. 44 Activity Diagram Menambah Data Jurusan.....	69
Gambar 3. 45 Activity Diagram Mengubah Data Jurusan.....	70
Gambar 3. 46 Activity Diagram Mencari Data Jurusan.....	71
Gambar 3. 47 Communication Diagram Login	72
Gambar 3. 48 Communication Diagram Mengubah Password.....	72
Gambar 3. 49 Communication Diagram Menambah Data Admin	73
Gambar 3. 50 Communication Diagram Mereset Password.....	73
Gambar 3. 51 Communication Diagram Mencari Data Admin	74
Gambar 3. 52 Communication Diagram Menambah Data Peserta	74
Gambar 3. 53 Communication Diagram Mengubah Data Peserta.....	75
Gambar 3. 54 Communication Diagram Mencari Data Peserta.....	76
Gambar 3. 55 Communication Diagram Menambah Data Paket Soal	76
Gambar 3. 56 Communication Diagram Menghapus Data Paket Soal.....	77
Gambar 3. 57 Communication Diagram Mencari Data Paket Soal	77
Gambar 3. 58 Communication Diagram Memasukkan Citra LJK.....	78
Gambar 3. 59 Communication Diagram Mendeteksi informasi LJK	79
Gambar 3. 60 Communication Diagram Memeriksa Jawaban Try Out Siswa	79
Gambar 3. 61 Communication Diagram Menyimpan Data Try Out Siswa.....	80
Gambar 3. 62 Communication Diagram Cari Data Try Out.....	80
Gambar 3. 63 Communication Diagram Mencetak Data Hasil Try Out.....	81
Gambar 3. 64 Communication Diagram Cari Data Jumlah Siswa Salah.....	82
Gambar 3. 65 Communication Diagram Menampilkan Grafik JSS	82
Gambar 3. 66 Communication Diagram Mengeksport Data Jumlah Siswa Salah	83
Gambar 3. 67 Communication Diagram Menambah Data Mata Pelajaran	84
Gambar 3. 68 Communication Diagram Mengubah Data Mata Pelajaran	84
Gambar 3. 69 Communication Diagram Mencari Data Mata Pelajaran	85
Gambar 3. 70 Communication Diagram Menambah Data Jurusan	85
Gambar 3. 71 Communication Diagram Mengubah Data Jurusan	86
Gambar 3. 72 Communiaction Diagram Mencari Data Jurusan	87
Gambar 3. 73 Class Diagram	88
Gambar 3. 74 Entity Relational Diagram.....	89
Gambar 3. 75 Physical Data Model	90
Gambar 3. 76 <i>User Interface</i> Login	91
Gambar 3. 77 <i>User Interface</i> Menu Utama.....	92
Gambar 3. 78 <i>User Interface</i> Paket Soal.....	93
Gambar 3. 79 <i>User Interface</i> Hasil Try Out	96
Gambar 3. 80 <i>User Interface</i> Menampilkan Jumlah Siswa Salah	97
Gambar 3. 81 <i>User Interface</i> Mengelola Data Admin.....	98
Gambar 3. 82 <i>User Interface</i> Mengelola Data Peserta	100
Gambar 3. 83 <i>User Interface</i> Mengelola Jurusan	102
Gambar 3. 84 <i>User Interface</i> Mengelola Mata Pelajaran	103
Gambar 3. 85 <i>User Interface</i> Koreksi Citra LJK	105

DAFTAR TABEL

Tabel 3. 2	Kebutuhan Fungsional	13
Tabel 3. 3	Kebutuhan Non Fungsional	14
Tabel 3. 4	Spesifikasi kebutuhan sistem	15
Tabel 3. 5	Usecase Skenario Login.....	28
Tabel 3. 6	Usecase Skenario Mengubah <i>Password</i>	28
Tabel 3. 7	Usecase Skenario Menambah Data Admin.....	29
Tabel 3. 8	Usecase Skenario Mereset <i>Password</i>	30
Tabel 3. 9	Usecase Skenario Mencari Data Admin	31
Tabel 3. 10	Usecase Skenario Menambah Data Peserta	31
Tabel 3. 11	Usecase Skenario Mengubah Data Peserta	32
Tabel 3. 12	Usecase Skenario Mencari Data Peserta.....	33
Tabel 3. 13	Usecase Skenario Menambah Data Paket Soal.....	33
Tabel 3. 14	Usecase Skenario Menghapus Data Paket Soal	34
Tabel 3. 15	Usecase Skenario Mencari Data Paket Soal	35
Tabel 3. 16	Usecase Skenario Memasukkan Citra LJK.....	35
Tabel 3. 17	Usecase Skenario Mendeteksi Informasi LJK	36
Tabel 3. 18	Usecase Skenario Memeriksa Jawaban Try Out Siswa	37
Tabel 3. 19	Usecase Skenario Menyimpan Data Try Out Siswa	37
Tabel 3. 20	Usecase Skenario Mencari Data Try Out.....	38
Tabel 3. 21	Mencetak Data Hasil <i>Try Out</i>	38
Tabel 3. 22	Usecase Skenario Mencari Jumlah Jawaban Siswa Salah	39
Tabel 3. 23	Usecase Skenario Menampilkan Grafik Jumlah Siswa Salah.....	40
Tabel 3. 24	Usecase Skenario Mengeksport Data Jumlah Siswa Salah.....	40
Tabel 3. 25	Usecase Skenerio Menambah Data Mapel.....	41
Tabel 3. 26	Usecase Mengubah Data Mata Pelajaran.....	42
Tabel 3. 27	Usecase Skenario Mencari Data Mata Pelajaran	42
Tabel 3. 28	Usecase Skenerio Menambah Data Jurusan.....	43
Tabel 3. 29	Usecase Skenario Mengubah Data Jurusan	44
Tabel 3. 30	Usecase Skenario Mencari Data Jurusan	44
Tabel 3. 31	Penjelasan <i>User Interface</i> Login.....	91
Tabel 3. 32	Penjelasan <i>User Interface</i> Menu Utama	92
Tabel 3. 33	Penjelasan <i>User Interface</i> Paket Soal	93
Tabel 3. 34	Penjelasan <i>User Interface</i> Hasil Try Out	96
Tabel 3. 35	Penjelasan <i>User Interface</i> Jumlah Siswa Salah	97
Tabel 3. 36	Penjelasan <i>User Interface</i> Mengelola Data Admin	99
Tabel 3. 37	<i>User Interface</i> Mengelola Data Peserta	100
Tabel 3. 38	<i>User Interface</i> Mengelola Jurusan	102
Tabel 3. 39	<i>User Interface</i> Mengelola Mata Pelajaran	104
Tabel 3. 40	<i>User Interface</i> Koreksi Citra LJK.....	105

BAB I PENDAHULUAN

1.1 Latar Belakang

Try out merupakan simulasi ujian yang biasanya diselenggarakan setiap sekolah untuk menghadapi ujian nasional yang diselenggarakan negara. Tujuan dari ujian nasional ini sendiri adalah menguji hasil belajar setiap siswa dan sebagai syarat kelulusan sekolah. Oleh karena itu setiap sekolah berlomba-lomba untuk meningkatkan kemampuan siswa dalam belajar agar dapat menghadapi ujian nasional. Salah satunya adalah SMA Negeri 2 Batam merupakan sekolah yang terletak di kecamatan Belakang Padang tepatnya di Jl. Raya Sekanak, Belakang Padang. Disekolah ini sistem koreksi masih dilakukan secara manual sehingga untuk pengelolaan hasil *try out* pihak sekolah membutuhkan waktu yang cukup lama sehingga dapat memperlambat sistem pengambilan keputusan oleh pihak sekolah. Selain itu dikarenakan program pemerintah yang telah menggunakan komputer untuk ujian nasional, Sekolah ini belum mempunyai cukup infrastruktur komputer untuk seluruh siswa sehingga pelaksanaannya masih menggunakan lembar ujian.

Oleh karena itu penulis ingin merancang dan membangun sebuah sistem yang dapat membantu mengatasi masalah pengelolaan data *try out* yang telah diuraikan dengan teknologi *image processing* menggunakan metode canny untuk mendeteksi tepi objek lingkaran. Metode canny merupakan metode deteksi tepi yang optimal dan merupakan metode deteksi tepi yang sangat baik (Fitri Suryaningsih, 2012). Operator Canny menggunakan *Gaussian Derivative Kernel* untuk memfilter *noise* dari citra *original* / asli dengan tujuan mendapatkan hasil deteksi tepi yang halus dan jelas (Fitri Suryaningsih, 2012). Sehingga gambar / citra lebih mudah diproses dan dianalisis. Selain itu metode canny juga dapat dikombinasikan dengan metode lain untuk kasus ini peneliti menggunakan konvolusi dengan operator sobel dan metode hough circle.

Metode sobel ini mempunyai keunggulan dapat menghilangkan *noise* sebelum deteksi tepi sehingga dapat meningkatkan kualitas citra yang akan diproses ke tahap selanjutnya (Asahar Johar, dkk, 2014).

Teknologi ini tidak membutuhkan infrastruktur yang banyak sehingga pengeluaran dana sekolah dapat ditekan. Maka dari itu dengan adanya sistem ini diharapkan dapat mempermudah para guru untuk pengelolaan data *try out* siswa dan meningkatkan kualitas pembelajaran siswa sehingga setiap siswa siap menghadapi ujian nasional yang diselenggarakan oleh pemerintah dengan baik.

1.2 Rumusan Masalah

Berdasarkan latar belakang yang telah dijelaskan, maka dapat dirumuskan permasalahan sebagai berikut.

1. Bagaimana merancang dan membangun suatu aplikasi pengelolaan citra digital dalam koreksi LJK dengan menerapkan metode canny?
2. Bagaimana mengelola data *try out* dengan cepat dan mudah?

1.3 Batasan Masalah

Batasan masalah dari penelitian Tugas Akhir ini adalah sebagai berikut.

1. Aplikasi LJK tidak menerapkan pengenalan pola tanda tangan dan tulisan tanggal ujian secara manual.
2. Tanggal ujian diinputkan oleh pengguna secara langsung tidak melalui citra LJK.
3. Tahun dan periode ujian di inputkan secara manual
4. Aplikasi tidak menangani perbaikan kemiringan citra LJK

1.4 Tujuan Penelitian

Merancang dan membangun aplikasi koreksi lembar jawaban komputer yang mampu :

1. Menerapkan metode canny untuk mengenali jawaban siswa.
2. Menampilkan hasil koreksi berupa nilai serta data siswa (peserta).
3. Menampilkan rekapitulasi hasil koreksi jawaban yang benar dan salah.

1.5 Manfaat Penelitian

Manfaat dari adanya aplikasi ini adalah sebagai berikut.

1. Tidak membutuhkan infrastruktur yang banyak
2. Dapat meminimalisir pengeluaran dana sekolah sehingga lebih hemat
3. Membantu sistem pengambilan keputusan pihak sekolah
4. Lebih efisien karena pihak sekolah tidak membutuhkan waktu yang lama dalam koreksi lembar jawaban siswa.
5. Membantu pengelolaan data *try out* siswa

1.6 Tinjauan Pustaka

Untuk penelitian yang berhubungan dengan “Penerapan Metode Canny Dalam Koreksi Lembar Jawaban Komputer Untuk Try Out sebelumnya juga telah dilakukan oleh Emundari – 2013 dengan judul “Perancangan Aplikasi Peminda Lembar Jawaban Komputer (LJK) Dengan Teknik Smartscan” dan oleh Asahar Johar, dkk -2014 dengan judul “Aplikasi Pengolahan Citra Digital Untuk Pendeteksi Jawaban Pada Lembar Jawaban Komputer Menggunakan Algoritma Sobel (Studi Kasus SMP Negeri 2 Kota Bengkulu)”. Berikut tabel perbandingan sistem yang telah dibuat sebelumnya dengan sistem yang akan dibangun.

Tabel 1. 1 Tinjauan Pustaka

Penulis	Judul	Tahun	Metode	Bahasa Pemrograman	DBMS
Ermundari	Perancangan Aplikasi Peminda Lembar Jawaban Komputer (LJK) Dengan Teknik Smartscan	2013	Scanline	Delphi	MySQL
Asahar Johar, Desi Andreswari, Gita Triyana	Aplikasi Pengolahan Citra Digital Untuk Pendeteksi Jawaban Pada Lembar Jawaban Komputer Menggunakan Algoritma Sobel (Studi Kasus SMP Negeri 2 Kota Bengkulu)	2014	Sobel	Delphi	MySQL
Ela Erminwati	Penerapan Metode Canny dalam Koreksi Lembar Jawaban Komputer untuk Try Out (Studi Kasus SMA Negeri 2 Batam)	2016	Canny	Matlab	MySQL

Tabel 1.1 merupakan tabel perbandingan beberapa penelitian yang dilakukan sebelumnya dengan penelitian yang akan dilakukan dalam bidang yang sama.

1.7 Sistematika Penulisan

Sistematika penulisan laporan tugas akhir ini terdiri atas:

BAB I Pendahuluan

Berisi Latar belakang, rumusan masalah, batasan masalah, tujuan penelitian, manfaat penelitian, tinjauan pustaka dan sistematika penulisan.

BAB II Landasan Teori

Berisi tentang landasan teori yang berhubungan dengan penelitian.

BAB III Metodologi Penelitian

Berisi mengenai analisis penerapan metodologi pada sistem

BAB IV Hasil dan Pembahasan

Berisi tentang pembahasan yang berhubungan dengan implementasi / penerapan metodologi pada aplikasi yang akan dibangun.

BAB V Kesimpulan dan Saran

Berisi tentang kesimpulan yang dapat ditarik dari Tugas Akhir ini dan saran yang diharapkan untuk penelitian selanjutnya.

BAB II

LANDASAN TEORI

2.1 Profil SMA Negeri 2 Batam

SMA Negeri 2 Batam atau biasa dipanggil SMANDA merupakan salah satu sekolah menengah atas yang terletak di Belakang Padang tepatnya Jl. Raya Sekanak Belakang Padang, Kepulauan Riau. Sekolah ini mempunyai jumlah siswa sekitar 90 – 150 setiap angkatan dan sudah terakreditasi B. Disekolah ini terdapat 2 jurusan yaitu IPA dan IPS yang terdiri dari 1 kelas IPA dan 3 kelas IPS.

2.2 Sistem Pendukung Keputusan (SPK)

Sistem pendukung keputusan (SPK) merupakan kumpulan prosedur atau sebuah sistem untuk pemrosesan data yang dapat memberikan informasi baru sehingga informasi yang dihasilkan dapat mendukung pengambilan keputusan oleh pihak terkait. Selain itu SPK juga dapat membantu pengelolaan data sekaligus dapat menghasilkan solusi dalam berbagai masalah. Berikut ini merupakan definisi sistem pengambilan keputusan (SPK) menurut beberapa ahli.

1. Menurut Moore and Chang, SPK dapat digambarkan sebagai sistem yang dapat mendukung analisis *ad hoc data*, dan pemodelan keputusan, orientasi perencanaan masa depan berorientasi keputusan, dan digunakan pada waktu yang tidak biasa.
2. Menurut Mann dan Watson, Sistem Penunjang Keputusan adalah sistem yang interaktif, membantu pengambilan keputusan melalui penggunaan data dan model-model keputusan untuk memecahkan masalah-masalah yang sifatnya semi terstruktur dan tidak terstruktur.
3. Menurut Litle, Sistem Penunjang Keputusan adalah suatu sistem informasi berbasis komputer yang menghasilkan berbagai alternatif

keputusan untuk membantu manajemen dalam menangani berbagai permasalahan yang terstruktur ataupun tidak terstruktur dengan menggunakan data dan model.

2.3 Image Processing

Citra merupakan representasi atau suatu gambaran dari suatu objek. Sedangkan pada citra digital adalah citra yang dapat diolah dengan komputer (Sutojo, 2009). *Image Processing* merupakan sebuah proses pengolahan piksel-piksel yang ada dalam citra untuk tujuan tertentu. Tujuan utama *image processing* dulunya untuk memperbaiki kualitas citra. Namun seiring dengan berkembangnya dunia komputasi yang ditandai dengan meningkatnya kapasitas dan kecepatan proses komputer serta munculnya ilmu-ilmu komputasi di era ini yang memungkinkan manusia dapat mengambil informasi yang dibutuhkan dari suatu citra. Bahkan di era ini *image processing* dan *computer vision* sudah digunakan bagaikan mata manusia yang dapat menemukan informasi yang ada dalam citra sehingga muncul beberapa pecahan bidang yang menjadi penting dalam *computer vision*, antara lain: *pattern recognition* (pengenalan pola), *biometric* (pengenalan identifikasi manusia berdasarkan ciri-ciri biologis yang tampak pada badan manusia), *content based image and video retrieval* (mendapatkan kembali citra atau video dengan informasi tertentu), *video editing*, dan lain-lain (Basuki, 2005:1).

2.4 Metode Canny

Metode canny merupakan salah satu metode deteksi tepi yang optimal. Hal ini dikarenakan pada metode canny operator yang digunakan adalah *Gaussian Derivative Kernel* yang dapat menyaring kegaduhan dalam citra digital. Selain itu metode canny juga dapat diintegrasikan dengan metode lain

sehingga hasil deteksi tepi akan lebih jelas dan halus. Berikut merupakan proses-proses yang dilakukan dalam metode canny.

1. Penghalusan (*smoothing*) dengan *Gaussian derivative kernel* untuk mengurangi noise.
2. Proses konvolusi merupakan proses pembentukan citra baru dalam penelitian ini penulis menggunakan operator sobel.
3. *Non-maximal supression* bertujuan mengubah tepi yang kabur pada citra menjadi tepi yang tajam.
4. *Hypertesisis thresholding* untuk tahap klasifikasi untuk membedakan piksel yang merupakan tepi objek atau bukan.

2.5 Transformasi Hough

Hough Transform pertama kali diperkenalkan oleh Paul Hough pada tahun 1962 untuk mendeteksi garis lurus. Hough Transform adalah teknik transformasi citra yang dapat digunakan untuk mengisolasi atau dengan kata lain memperoleh fitur dari sebuah citra. Karena tujuan dari sebuah transformasi adalah mendapatkan suatu fitur yang lebih spesifik, Classical Hough Transform merupakan teknik yang paling umum digunakan untuk mendeteksi objek yang berbentuk kurva seperti garis, lingkaran, elips dan parabola. Keuntungan utama dari transformasi Hough adalah dapat mendeteksi sebuah tepian dengan celah pada batas fitur dan secara relatif tidak dipengaruhi oleh derau atau noise.

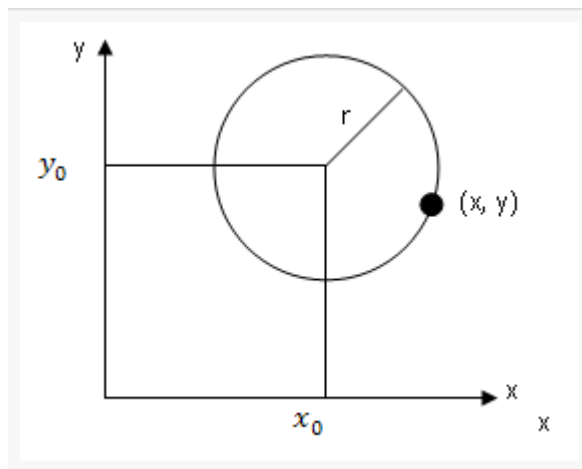
a. Fungsi Lingkaran

Jika objek yang dicari berupa lingkaran, maka digunakan transformasi lingkaran Hough. Prosedur yang digunakan dalam mendeteksi lingkaran adalah sama dengan transformasi Hough pada objek garis, tapi dikerjakan pada ruang dimensi yang lebih kompleks, yaitu dalam parameter ruang 3D (X_0, Y_0, R). Berikut merupakan rumus persamaan lingkaran:

$$(x-x_0)^2 + (y-y_0)^2 = r^2$$

Gambar 2. 1 Rumus Persamaan Lingkaran

Gambar 2.1 merupakan rumus persamaan lingkaran dimana x_0 dan y_0 merupakan koordinat pusat lingkaran dan r merupakan jari-jari lingkaran. Jika diilustrasikan dalam koordinat (X_0, Y_0, R) adalah sebagai berikut:



Gambar 2. 2 Ilustrasi Koordinat

Gambar 2.2 merupakan gambar ilustrasi koordinat lingkaran dimana fungsi hough circle dapat membentuk sebuah lingkaran berdasarkan nilai jari-jarinya.

2.6 Pemrograman Matlab

Matlab merupakan sebuah bahasa pemrograman (*high-performance*) dengan kinerja tinggi untuk mengatasi masalah-masalah komputasi teknis dalam berbagai bidang seperti:

- a) Matematika dan Komputasi
- b) Pembentukan Algorithm
- c) Akusisi Data
- d) Pemodelan, simulasi, dan pembuatan *prototype*

- e) Analisa data, explorasi, dan visualisasi
- f) Grafik Keilmuan dan bidang Rekayasa

MATLAB merupakan singkatan dari *matrix laboratory*. MATLAB pada awalnya dibuat dengan tujuan memudahkan akses perangkat lunak matrik yang telah dibentuk oleh LINPACK dan EISPACK. Kemudian seiring dengan perkembangan zaman, saat ini perangkat MATLAB telah diintegrasikan dengan LAPACK dan BLAS *library*, yang merupakan suatu kesatuan dari sebuah seni tersendiri dalam perangkat lunak untuk komputasi *matrix*. Dengan menggunakan matlab, proses pengolahan citra akan lebih mudah karena pada matlab citra dapat dibaca sebagai kumpulan nilai pixel warna pada citra digital yang membentuk sebuah *matrix*.

2.7 Basis Data

Basis data merupakan kumpulan data yang mendeskripsikan aktifitas struktur data yang saling berhubungan. Tujuan dari basis data atau pembuatan database adalah untuk menentukan informasi apa saja yang dibutuhkan oleh sebuah sistem. Selain itu dengan adanya basis data, informasi yang disimpan dapat terorganisir. Berikut ini definisi basis data menurut beberapa ahli.

1. Menurut “Siberschatz, dkk.; (2002) mendefinisikan basisdata sebagai kumpulan data berisi informasi yang sesuai untuk sebuah perusahaan. Sistem manajemen basisdata (DBMS) adalah kumpulan data yang saling berhubungan dan kumpulan program untuk mengakses data. Tujuan utama system manajemen basisdata adalah menyediakan cara menyimpan dan mengambil informasi basisdata secara mudah dan efisien”.
2. Menurut “Stephens dan Plew (2000), adalah mekanisme yang digunakan untuk menyimpan informasi atau data.

2.8 MySQL

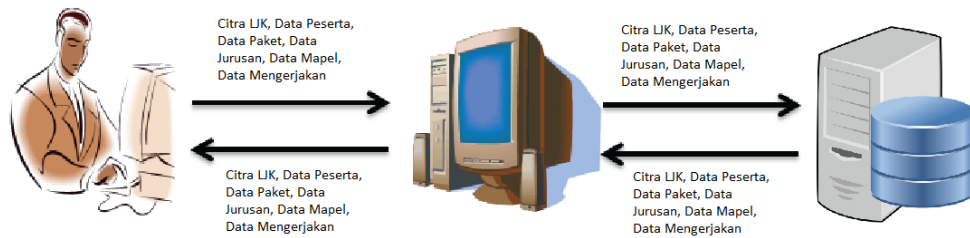
MySQL merupakan salah satu perangkat lunak pengelola basisdata SQL yang *multithread* dan *multi-user*. MySQL juga merupakan RDBMS

(*Relational Database Menegement Sytem*) yang didistribusikan secara gratis dibawah lisensi GPL (*General Public Licensi*) . MySQL ini merupakan produk *open source* dimana setiap orang dapat menggunakannya. Dengan adanya MySQL dapat membantu para *developer* dalam mengembangkan berbagai *software*.

BAB III PERANCANGAN SISTEM

3.1 Deskripsi Umum Sistem

Penerapan metode canny dalam koreksi lembar jawaban computer untuk try out merupakan sebuah aplikasi yang memanfaatkan teknologi *image processing* lebih tepatnya dengan menggunakan metode canny yang merupakan salah satu metode deteksi tepi untuk mengenali pola objek lingkaran pada LJK.



Gambar 3. 1 Deskripsi Umum Sistem

Gambar 3.1 merupakan deskripsi umum sistem dari aplikasi yang akan dibangun dimana user akan memasukkan data yang dibutuhkan seperti memasukkan citra LJK yang akan diproses, data peserta, data paket, data mapel, data jurusan dan data mengerjakan. Kemudian sistem akan pemroses data inputan tersebut sehingga dapat disimpan ke database dan data tersebut dapat ditampilkan kembali sesuai kebutuhan.

3.2 Spesifikasi Kebutuhan Fungsional dan Non Fungsional

A. Kebutuhan Fungsional

Berikut merupakan daftar kebutuhan fungsional dari aplikasi yang dibangun.

Tabel 3. 1 Kebutuhan Fungsional

F-001	Admin dapat mengakses sistem (Login) dengan username dan password yang benar
F-002	Admin dapat mengubah password pribadi
F-003	Admin dapat menambah admin
F-004	Admin dapat mereset password
F-005	Admin dapat mencari data admin berdasarkan username
F-006	Admin dapat menambah data peserta
F-007	Admin dapat mengubah data peserta
F-008	Admin dapat mencari data peserta nama / jurusan
F-009	Admin dapat menambah data paket soal
F-010	Admin dapat menghapus data paket soal
F-011	Admin dapat mencari data paket soal berdasarkan kode paket
F-012	Sistem dapat menyimpan data try out siswa
F-013	Sistem dapat memeriksa jawaban try out siswa
F-014	Admin dapat mencari hasil try out siswa berdasarkan nama dan jurusan
F-015	Admin dapat mencari jumlah siswa salah pada setiap soal berdasarkan kode paket dan periode
F-016	Sistem dapat menampilkan grafik untuk jumlah siswa salah pada setiap kode paket soal berdasarkan kode paket dan periode

F-017	Sistem dapat mencetak report hasil try out
F-018	Admin dapat memasukkan citra LJK
F-019	Sistem dapat mengolah citra LJK dengan metode canny
F-020	Admin dapat mengekspor data jumlah siswa salah dalam bentuk file excel
F-021	Admin dapat menambah data mapel
F-022	Admin dapat mengubah data mapel
F-023	Admin dapat mencari data mapel berdasarkan kode mapel
F-024	Admin dapat menambah data jurusan
F-025	Admin dapat mengubah data jurusan
F-026	Admin dapat mencari data jurusan berdasarkan kode jurusan

Tabel 3.1 merupakan tabel kebutuhan fungsional atau bisa disebut sebagai fungsi penting yang akan ada dalam aplikasi yang dibangun. Fungsi-fungsi ini merupakan *tools* yang akan membantu pengguna untuk mengelola data peserta try out.

B. Kebutuhan Non Fungsional

Berikut merupakan daftar kebutuhan non fungsional dari aplikasi yang dibangun.

Tabel 3. 2 Kebutuhan Non Fungsional

NF-001	Aplikasi yang dibangun memiliki tampilan yang sederhana
NF-002	Menggunakan Bahasa Indonesia

Tabel 3.2 merupakan daftar kebutuhan non fungsional / fungsi yang biasanya kurang diperhatikan oleh pengguna seperti tampilan aplikasi dan bahasa yang digunakan dalam aplikasi.

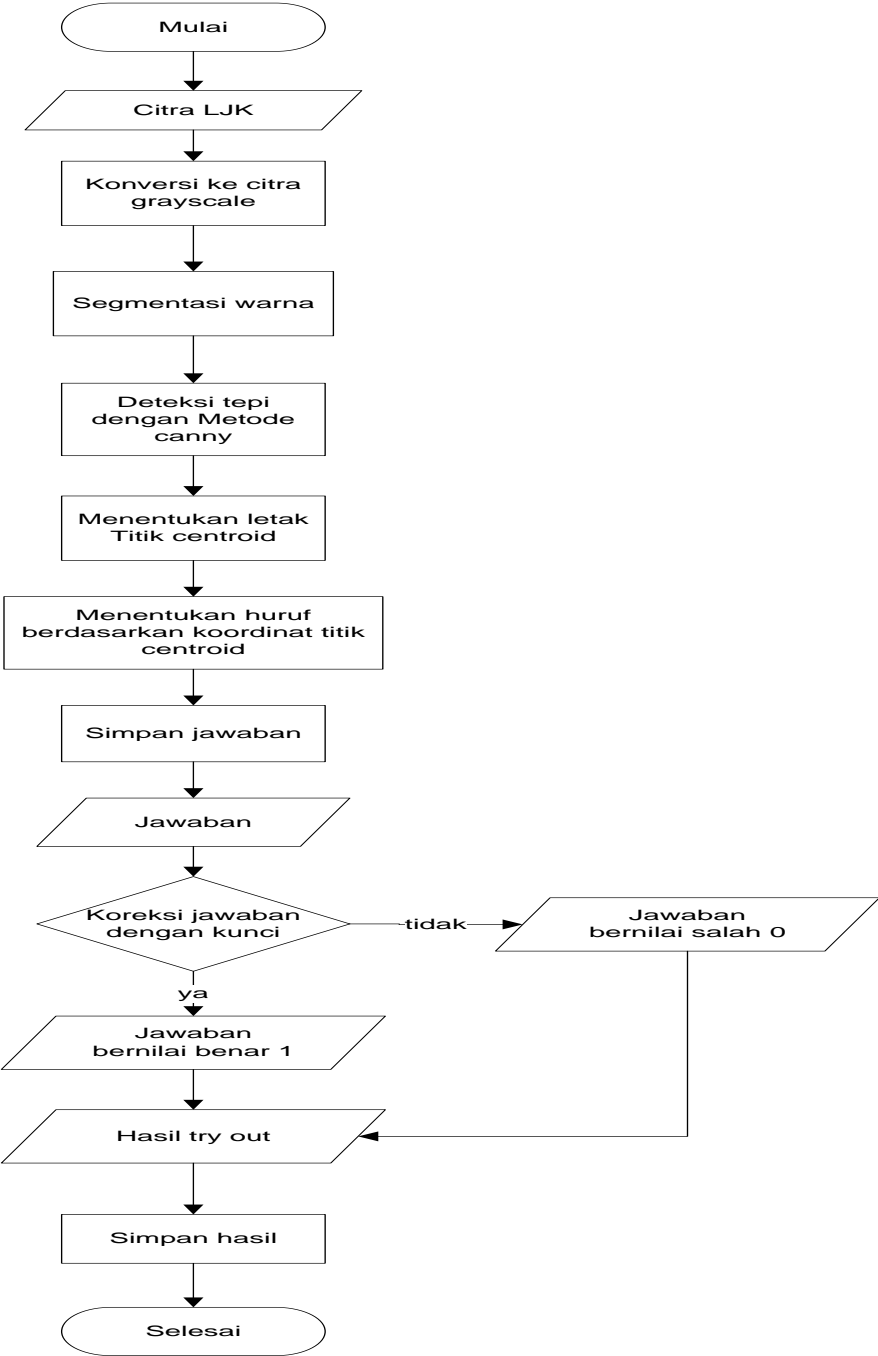
3.3 Spesifikasi Kebutuhan Sistem

Dalam penelitian ini penulis menggunakan spesifikasi sistem sebagai berikut.

Tabel 3. 3 Spesifikasi kebutuhan sistem

Perangkat	Hardware	Software
Laptop/PC	<ul style="list-style-type: none">• RAM 4 GB• Hardisk 1 TB• Processor Intel® Core™ i5-4200U CPU @ 1.60GHz, ~2.3GHz• VGA : NVIDIA GeForce 820M	<ul style="list-style-type: none">• OS Windows 10 Home Single Edition 64 bit• Matlab R2015a• XAMPP v3.2.1• MySQL ODBC 5.3 Unicode Driver 64 bit

3.4 Metodologi Penelitian



Gambar 3. 2 Metodologi Penelitian

Pada Gambar 3.2 menjelaskan algoritma dari metodologi penelitian yang akan diterapkan pada sistem yang akan dibangun. Untuk dapat mendeteksi sebuah objek pada citra LJK maka diperlukan beberapa teknik pengolahan citra (*image processing*). Pada *flowchart* di atas, langkah pertama admin memasukkan citra LJK hasil scan dengan format citra yang telah ditentukan. Kemudian langkah ke 2 citra dikonversi ke *grayscale* untuk menyederhanakan citra. Kemudian sistem akan melakukan segmentasi warna hitam. Citra Langkah ke 3 adalah deteksi tepi objek dengan metode *canny*. Kemudian sistem akan menentukan titik *centroid* dari objek. Titik *centroid* inilah yang akan dijadikan acuan untuk menentukan titik koordinat objek. Langkah ke 5 menentukan huruf pada setiap titik koordinat *centroid* objek. Kemudian rangkaian huruf yang terbentuk tersebut disimpan pada database dan data tersebut dipanggil kembali untuk dicocokkan dengan kunci jawaban. Data jawaban tersebut akan dipecah menjadi data array. Jika nilai array jawaban sama dengan nilai array pada kunci maka hasil akan bertambah 1 jika tidak maka nilai tambah = 0. Hasil tersebut kemudian di kalkulasi menjadi hasil akhir nilai *try out*. Kemudian langkah terakhir sistem mengupdate nilai *try out* siswa.

3.5 Teknik Pengumpulan Data

a. Observasi ke Lapangan

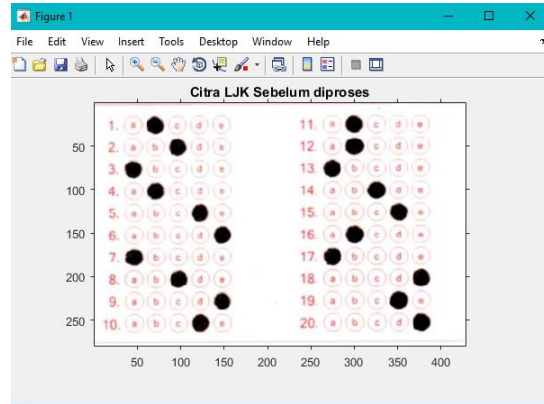
Metode pengumpulan data pada penelitian ini adalah dengan observasi atau pengamatan ke lapangan. Dimana sistem koreksi masih dilakukan secara manual

b. Dokumentasi

Peneliti mendapatkan data-data yang dibutuhkan seperti data siswa, kunci data paket soal, dan hasil *try out*. Data – data tersebut merupakan arsip sekolah yang pengelolaannya masih menggunakan ms. Excel.

3.6 Teknik Pengolahan dan Analisis Data

1. Sistem membaca citra yang diinputkan oleh admin dalam bentuk matiks



Gambar 3. 3 Hasil Membaca Image

Gambar 3.3 merupakan hasil pembacaan citra oleh program matlab dengan menggunakan fungsi `imread` pada matlab. Program matlab membaca citra dalam bentuk matrik dengan nilai yang sesuai pixel warna citra.

2. Image kemudian di konversi ke citra biner

Untuk dapat mengkonversi citra rgb ke biner maka citra tersebut harus dikonversi ke citra grayscale terlebih dahulu agar citra lebih sederhana.

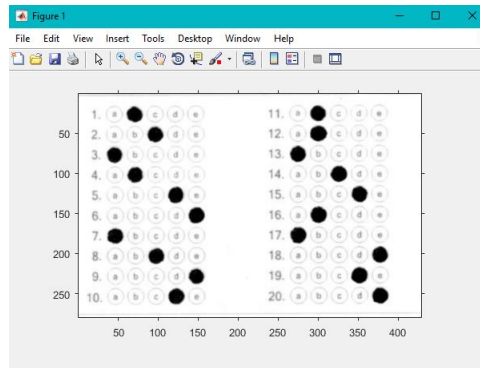
- Formula konversi dari citra RGB ke grayscale

$$\text{Grayscale} = 0.3 * \text{Red} + 0.5 * \text{Green} + 0.2 * \text{Blue};$$

Algoritma konversi citra RGB ke Grayscale

- 1) *Read image*
- 2) Ekstraksi *image* warna RGB
- 3) *Convert to grayscale*

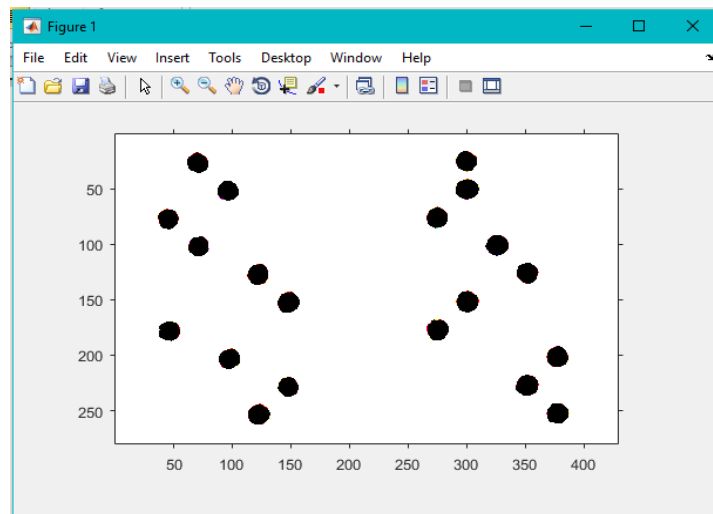
Fungsi konversi RGB ke *Grayscale*



Gambar 3. 4 Hasil Citra Grayscale

Gambar 3.4 merupakan hasil konversi citra RGB ke dalam versi citra grayscale. Nilai maksimum warna yang dapat diterapkan pada citra adalah 0 sampai 1. Maka dari itu nilai konstanta untuk merubah nilai warna merah, hijau, dan biru jika dijumlahkan hasil maksimumnya adalah 1 yaitu warna hitam.

- Konversi citra grayscale ke biner (hitam putih)
 $\text{Algorithm } \text{image} > \text{level} = 1; \text{image} \leq \text{level} = 0$
- Penerapan fungsi grayscale dan biner dengan nilai toleransi 0.3
 Hasil dari citra grayscale ke biner



Gambar 3. 5 Hasil Citra Konversi ke Biner

Gambar 3.5 merupakan citra hasil konversi citra grayscale ke biner dimana jika nilai pixel warna grayscale lebih dari nilai toleransi / nilai yang di tentukan maka nilai pixel warna akan diubah menjadi 1 (warna hitam) jika kurang dari sama dengan nilai toleransi maka nilai pixel diubah menjadi 0 (warna putih).

3. Penerapan metode canny

1) Pembentukan size mask

Ukuran_mask = floor(nilai*sigma) + 1. *Mask* merupakan ukuran dari matrik seperti matrik dengan ordo 3 x 3.

2) Pembentukan Gaussian mask

$$G(x, y) = \frac{1}{2\pi\sigma^2} e^{-\frac{(x^2+y^2)}{2\sigma^2}}$$

Gambar 3. 6 Formula *Gaussian Mask*

Gambar 3.6 merupakan formula/ kernel yang digunakan untuk mengkonversi nilai pixel citra. Hal ini bertujuan untuk mengurangi *noise* pada citra.

3) Konvolusi mask Gaussian

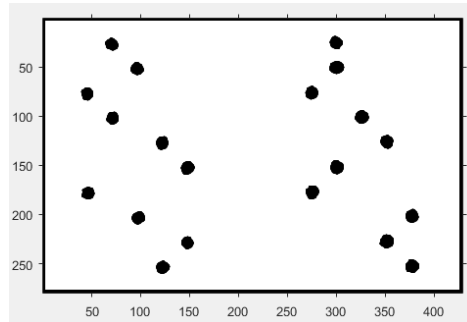
$$X = \begin{bmatrix} -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \end{bmatrix} \text{ and } Y = \begin{bmatrix} -1 & -1 & -1 \\ 0 & 0 & 0 \\ 1 & 1 & 1 \end{bmatrix}$$

$$e^{-\frac{(x^2+y^2)}{2\sigma^2}} = \begin{bmatrix} -2.7778 & -1.3889 & -2.7778 \\ -1.3889 & 0 & -1.3889 \\ -2.7778 & -1.3889 & -2.7778 \end{bmatrix}$$

Gambar 3. 7 Formula Konvolusi *Gaussian Mask*

Gambar 3.7 merupakan matrik yang digunakan untuk pembentukan citra baru dengan kernel yang sudah dijelaskan sebelumnya.

Hasil:



Gambar 3. 8 Hasil Konvolusi *Gaussian Mask*

Gambar 3.8 merupakan citra hasil konvolusi dengan Gaussian mask. Gambar yang dihasil akan kabur / *blur*. Biasanya hal ini untuk mengurangi *noise*.

4) Menentukan arah gradient dan magnitudo dengan operator sobel

-1	0	1
-2	0	2
-1	0	1

Mask (a) Vertikal

-1	-2	-1
0	0	0
1	2	1

Mask (b)
Horizontal

Gambar 3. 9 Operator Sobel

Gambar 3.9 merupakan operator sobel yang digunakan untuk menentukan arah *gradient* citra.

Contoh citra dengan matriks 3 x 3

a_0	a_1	a_2
a_7	a_{ij}	a_3
a_6	a_5	a_4

Gambar 3. 10 Contoh Matrik

Gambar 3.10 merupakan contoh matrik untuk dikalikan dengan operator sobel.

Maka didefinisikan arah penelusuran sebagai berikut:

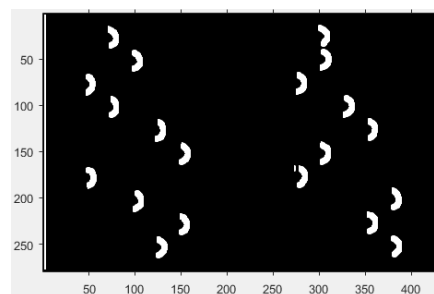
Arah vertical:

$$G_y = (a_0 - a_6) + (a_2 - a_4) + 2(a_1 - a_5)$$
$$G_y = (a_0 + 2a_1 + a_2) - (a_4 + 2a_5 + a_6)$$

Gambar 3. 11 Formula Gradient Arah Vertikal

Gambar 3.11 merupakan operasi apabila operator vertikal sobel dikalikan dengan contoh matrik.

Hasil:



Gambar 3. 12 Hasil Gradient Arah Vertikal

Gambar 3.12 merupakan hasil konvolusi citra dengan arah pembacaan hanya pada arah vertikal.

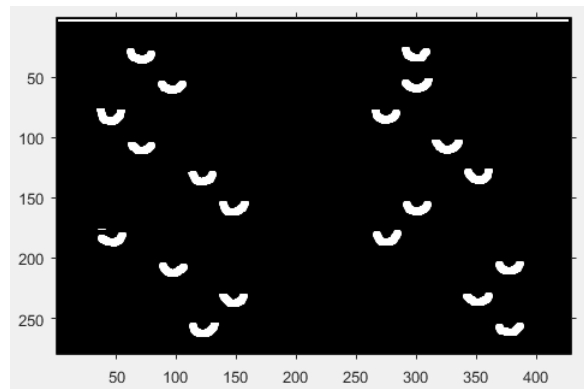
Arah horizontal:

$$G_x = (a_0 - a_2) + (a_6 - a_4) + 2(a_7 - a_3)$$
$$G_x = (a_0 + 2a_7 + a_6) - (a_2 + a_4 + 2a_3)$$

Gambar 3. 13 Formula Gradient Arah Horizontal

Gambar 3.13 merupakan operasi apabila operator horizontal sobel dikalikan dengan contoh matrik.

Hasil:



Gambar 3. 14 Hasil Gradient Arah Horizontal

Gambar 3.14 merupakan hasil konvolusi citra dengan arah pembacaan hanya pada arah horizontal.

Sedangkan untuk menentukan potensi gradient citra digunakan formula berikut.

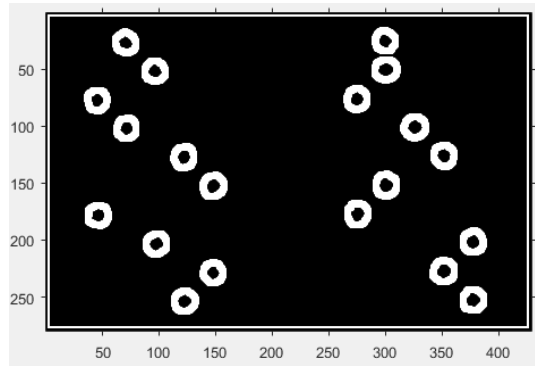
$$M = \sqrt{G_x^2 + G_y^2}$$

Gambar 3. 15 Formula Magnitudo

Gambar 3.15 merupakan rumus dari magnitudo maksud magnitudo disini adalah pembacaan konvolusi citra bukan hanya dari arah

horizontal / vertikal saja namun dari arah horizontal dan vertical sekaligus sehingga membentuk sebuah tepi lingkaran.

Hasil:

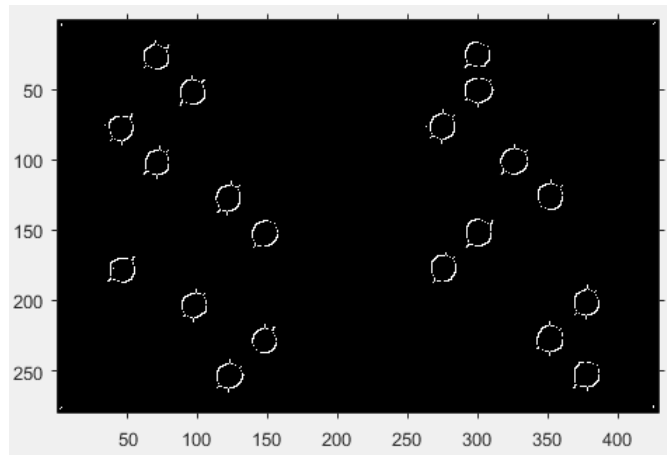


Gambar 3. 16 Hasil Penerapan Magnitudo

Gambar 3.16 merupakan hasil dari penerapan formula magnitudo pada citra LJK.

- 5) Penipisan tepi objek, teknik non-maximum suppression, hypertesis thresholding

Hasil:

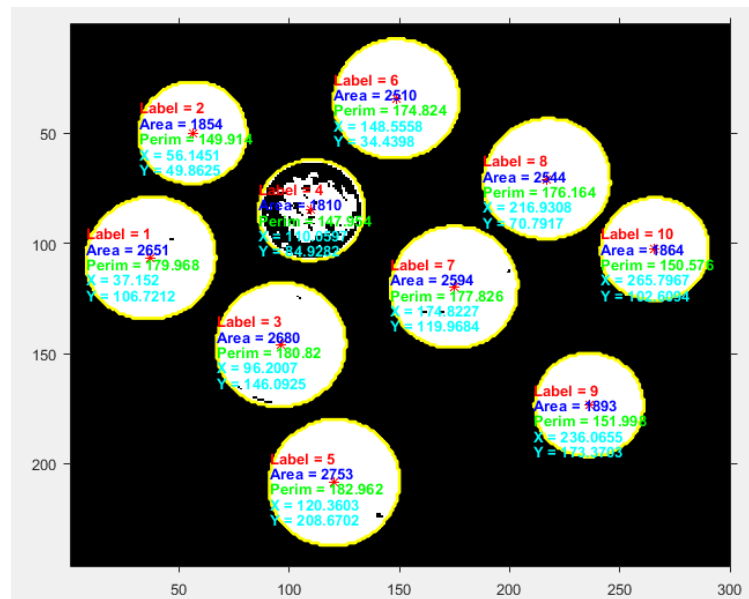


Gambar 3. 17 Hasil Deteksi Tepi Canny

Gambar 3.17 merupakan hasil deteksi tepi objek dengan metode canny yang telah melalui proses penipisan tepi, *non-maximum suppression* atau menghilangkan nilai tak *maximum* agar tepi yang dihasilkan lebih

tajam, dan terakhir adalah proses *hypertesis thresholding* merupakan pemisahan area objek dengan tepi telah dilakukan ketika sistem telah menemukan arah gradient dan magnitudo objek.

5. Menentukan titik centroid



Gambar 3. 18 Contoh Penerapan *Centroid*

Gambar 3.18 merupakan hasil contoh penerapan fungsi centroid pada matlab. Fungsi ini akan mencari titik tengah dari objek terdeteksi.

6. Menentukan huruf berdasarkan letak titik *centroid* (koordinat dari titik centroid)
7. Koreksi jawaban dengan mencocokkan dengan kunci di database
8. Kemudian menyimpan ke database

3.4 Penyajian Data

Hasil dari penelitian ini berupa sebuah daftar nilai peserta *try out* seperti berikut.

Jurusan : IPA						
Kode Peserta	Nama	Tgl ujian	Paket	J_salah	j_benar	Nilai
02 002 020 5	ADRIANI NURUL DIASTARY	16 Mei 2016	BIN01	10	40	80
02 002 001 8	ANANDA SET A NINGRUM	16 Mei 2016	BIN02	16	34	68
02 002 002 7	DEDE AFRIANA	16 Mei 2016	BIN03	17	33	66
02 002 021 4	DELLA INDIYANI	16 Mei 2016	BIN04	16	34	68
02 002 003 6	DENI SETIAWAN	16 Mei 2016	BIN05	16	34	68
02 002 022 3	DWI ANDINI RIZKY	16 Mei 2016	BIN06	16	34	68
02 002 004 5	DWI DINI LESTARI	16 Mei 2016	BIN07	14	36	72
02 002 023 2	ELMA NOVIANTI	16 Mei 2016	BIN08	21	29	58
02 002 005 4	FEBRI SILVIANI	16 Mei 2016	BIN09	23	27	54
02 002 024 9	FITRIA TAMARA	16 Mei 2016	BIN10	21	29	58

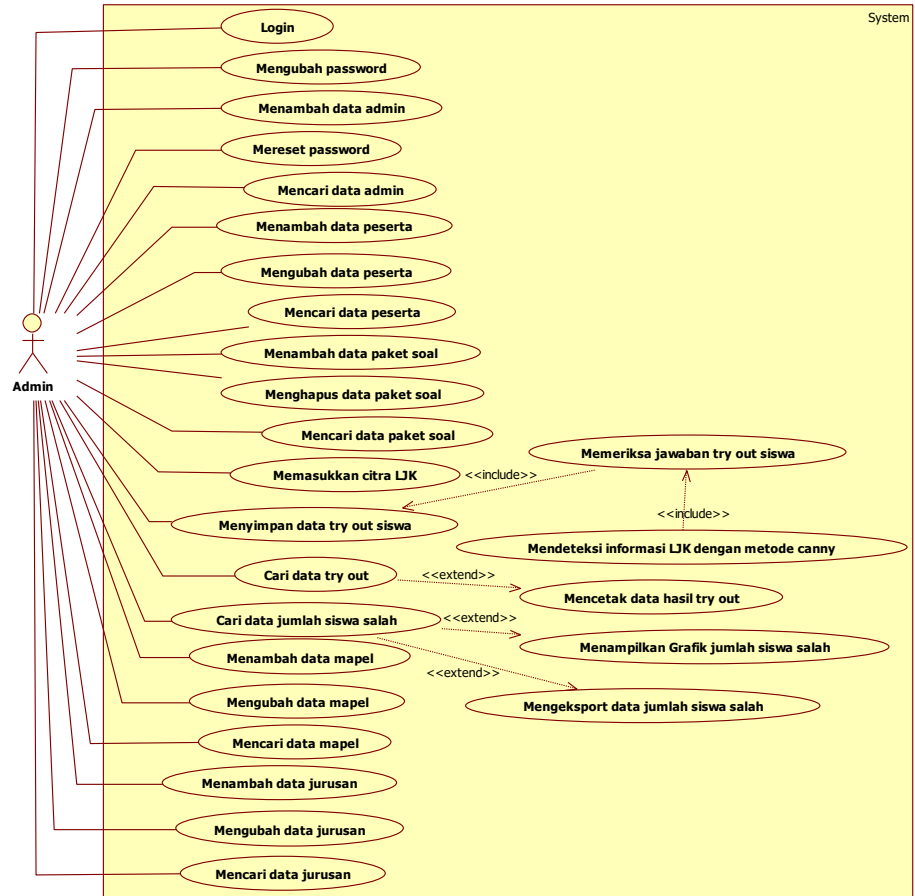
Gambar 3. 19 Penyajian Data Akhir

Gambar 3.19 merupakan penyajian data akhir dari penelitian yang akan dilakukan. Data tersebut merupakan hasil koreksi sistem dengan penerapan metode deteksi tepi canny.

3.7 Analisis Sistem

3.7.1 Use Case Diagram

Berikut merupakan use case diagram atau sebuah gambaran interaksi pengguna dengan sistem yang akan dibangun.



Gambar 3. 20 Use case diagram

Pada gambar 3.20 menggambarkan berbagai fitur dari aplikasi yang akan dibuat sesuai dengan kebutuhan dan fungsinya pengguna dapat login hingga dapat melakukan koreksi LJK serta sistem akan menampilkan hasil nilai try out.

3.7.2 Skenario Usecase

Berikut merupakan tabel-tabel usecase skenario / skenario jalannya sistem ketika pengguna mengoperasikan aplikasi yang akan dibangun.

Tabel 3. 4 Usecase Skenario Login

Nama Use Case	<i>Login</i>
Deskripsi	Digunakan untuk mengakses sistem yang dibangun serta untuk menjaga kerahasiaan data
Aktor	Admin
Kondisi Awal	Server dalam keadaan hidup dan admin telah membuka <i>GUI login</i>
Kondisi Akhir	Sistem menampilkan GUI menu utama
Skenario	<ol style="list-style-type: none">1. Admin memasukkan <i>username</i> dan <i>password</i>2. Sistem meverifikasi <i>username</i> dan <i>password</i> yang dimasukkan3. Sistem menampilkan GUI menu utama
Skenario Alternatif	2a. [<i>Username</i> atau <i>password</i> salah / kosong] menampilkan pesan <i>login</i> gagal kemudian kembali ke langkah 1.

Tabel 3.4 menjelaskan apabila pengguna melakukan login dengan benar sistem akan berjalan sesuai dengan skenario sedangkan jika pengguna melakukan login dengan salah maka scenario alternatif akan berjalan.

Tabel 3. 5 Usecase Skenario Mengubah *Password*

Nama Use Case	Mengubah <i>password</i>
Deskripsi	Digunakan untuk mengubah <i>password default</i> admin sesuai keinginan admin dan untuk membuka aplikasi sesuai dengan <i>username</i> dan <i>password</i> yang telah dimiliki masing-masing admin.
Aktor	Admin
Kondisi Awal	Admin telah <i>login</i> dan membuka <i>GUI</i> admin
Kondisi Akhir	Admin berhasil mengubah <i>password</i> pribadi
Skenario	<ol style="list-style-type: none">1. Admin memasukkan <i>password</i> lama2. Admin memasukkan <i>password</i> baru3. Admin memasukkan konfirmasi <i>password</i> baru.4. Sistem meverifikasi inputan <i>password</i> yang

	diinputkan admin 5. Sistem menyimpan perubahan <i>password</i> 6. Sistem menampilkan pesan <i>password</i> berhasil di ubah 7. Sistem menampilkan seluruh data admin kecuali <i>password</i>
Skenario Alternatif	4a. [Konfirmasi <i>password</i> tidak cocok] menampilkan pesan konfirmasi <i>password</i> tidak sesuai, kemudian kembali ke langkah 3.

Tabel 3.5 menjelaskan apabila pengguna mengubah *password* dengan prosedur yang benar sistem akan berjalan sesuai dengan skenario sedangkan jika pengguna mengubah *password* dengan prosedur yang salah maka skenario alternatif akan berjalan.

Tabel 3. 6 Usecase Skenario Menambah Data Admin

Nama Use Case	Menambah data admin
Deskripsi	Digunakan untuk menambah admin baru guna mempercepat pekerjaan
Aktor	Admin
Kondisi Awal	Admin telah <i>login</i> dan membuka <i>GUI</i> admin
Kondisi Akhir	Sistem berhasil menambahkan data admin baru
Skenario	<ol style="list-style-type: none"> 1. Admin memasukkan <i>username</i> dan <i>password</i> baru 2. Sistem memverifikasi inputan <i>username</i> dan memeriksa apakah <i>username</i> telah terdaftar / belum 3. Sistem menyimpan data admin baru 4. Sistem menampilkan pesan data admin berhasil ditambahkan 5. Sistem menampilkan kembali seluruh daftar admin
Skenario Alternatif	2a. [<i>Username</i> telah terdaftar] menampilkan pesan <i>username</i> telah terdaftar, penambahan admin gagal, kembali ke langkah 1.

Tabel 3.6 menjelaskan apabila pengguna menambah data admin dengan prosedur yang benar sistem akan berjalan sesuai dengan skenario sedangkan jika pengguna menambah data admin dengan prosedur yang salah maka skenario alternatif akan berjalan.

Tabel 3. 7 Usecase Skenario Mereset *Password*

Nama Use Case	Mereset <i>password</i>
Deskripsi	Digunakan untuk mereset <i>password</i> admin lain apabila ada admin lain yang lupa password untuk login ke sistem
Aktor	Admin
Kondisi Awal	Admin telah <i>login</i> dan GUI admin telah menampilkan seluruh data admin kecuali <i>password</i>
Kondisi Akhir	Sistem berhasil mereset <i>password</i> admin yang telah dipilih
Skenario	<ol style="list-style-type: none"> 1. Sistem menampilkan seluruh data admin 2. Admin mencari data admin yang dimaksud dengan memasukkan username admin tersebut 3. Sistem menampilkan data admin yang dicari 4. Admin memilih data yang akan direset 5. Sistem menampilkan pesan konfirmasi 6. Admin memasukkan instruksi sesuai dengan pilihan yang tersedia (yes jika menyetujui , no jika batal) 7. Sistem mereset password ke kondisi awal yaitu dengan password default admin123 8. Sistem menampilkan pesan data berhasil direset 9. Sistem menampilkan kembali data admin
Skenario Alternatif	5a. [Admin memilih opsi <i>no</i>] sistem menutup <i>message dialog</i> dan kemudian sistem menampilkan seluruh data admin kembali.

Tabel 3.7 menjelaskan apabila pengguna mereset *password* dengan prosedur yang benar sistem akan berjalan sesuai dengan skenario sedangkan jika pengguna mereset *password* dengan prosedur yang salah maka skenario alternatif akan berjalan.

Tabel 3. 8 Usecase Skenario Mencari Data Admin

Nama Use Case	Mencari data admin
Deskripsi	Digunakan untuk menampilkan data admin secara keseluruhan maupun secara spesifik
Aktor	Admin
Kondisi Awal	Admin telah <i>login</i> dan GUI admin telah menampilkan seluruh data admin kecuali <i>password</i>
Kondisi Akhir	Sistem berhasil menampilkan data admin yang dicari
Skenario	<ol style="list-style-type: none">1. Sistem menampilkan data admin secara keseluruhan2. Admin memasukkan kata kunci <i>username</i>3. Sistem melakukan pencarian sesuai dengan <i>username</i> yang diinputkan4. Sistem menampilkan data yang telah ditemukan
Skenario Alternatif	3a. [Sistem tidak menemukan data <i>username</i> yang diinputkan] kembali ke langkah 1.

Tabel 3.8 menjelaskan apabila pengguna mencari data admin dengan data yang benar sistem akan berjalan sesuai dengan skenario sedangkan jika pengguna mencari data admin yang salah maka skenario alternatif akan berjalan.

Tabel 3. 9 Usecase Skenario Menambah Data Peserta

Nama Use Case	Menambah data peserta
Deskripsi	Digunakan untuk menambah data peserta try out baru
Aktor	Admin
Kondisi Awal	Admin telah <i>login</i> dan membuka GUI peserta
Kondisi Akhir	Sistem berhasil menambahkan data peserta baru
Skenario	<ol style="list-style-type: none">1. Admin memasukkan data <i>peserta</i>2. Sistem memverifikasi inputan data peserta dan memeriksa apakah kode peserta sudah terdaftar atau belum3. Sistem menyimpan data peserta try out4. Sistem menampilkan pesan data peserta berhasil ditambahkan5. Sistem menampilkan kembali seluruh daftar data peserta
Skenario Alternatif	2a. [Kode peserta telah terdaftar] menampilkan pesan kode peserta telah terdaftar, penambahan data peserta gagal, kembali ke langkah 1.

	2b. [Data peserta tidak lengkap] menampilkan pesan data peserta tidak lengkap, penambahan data peserta gagal, kembali ke langkah 1.
--	---

Tabel 3.9 menjelaskan apabila pengguna menambah data peserta dengan prosedur yang benar sistem akan berjalan sesuai dengan skenario sedangkan jika pengguna menambah data peserta dengan prosedur yang salah maka skenario alternatif akan berjalan.

Tabel 3. 10 Usecase Skenario Mengubah Data Peserta

Nama Use Case	Mengubah data peserta
Deskripsi	Digunakan untuk mengubah semua data peserta try out baru kecuali kode peserta
Aktor	Admin
Kondisi Awal	Admin telah <i>login</i> dan membuka <i>GUI</i> peserta
Kondisi Akhir	Sistem berhasil mengubah data peserta baru
Skenario	<ol style="list-style-type: none"> 1. Sistem menampilkan seluruh data peserta try out yang telah terdaftar 2. Admin mencari data peserta dengan memasukkan kata kunci nama/jurusan/tahun 3. Sistem menampilkan data peserta 4. Admin memilih data yang akan diubah 5. Admin memasukkan data peserta / merubah data peserta 6. Sistem memverifikasi perubahan data peserta 7. Sistem menyimpan perubahan data admin 8. Sistem menampilkan pesan data peserta berhasil diubah 9. Sistem menampilkan kembali seluruh data peserta
Skenario Alternatif	4a. [Data peserta tidak lengkap] menampilkan pesan data peserta tidak lengkap, perubahan data peserta gagal, kembali ke langkah 1.

Tabel 3.10 menjelaskan apabila pengguna mengubah data peserta dengan prosedur yang benar sistem akan berjalan sesuai dengan skenario sedangkan jika pengguna mengubah data peserta dengan prosedur yang salah maka skenario alternatif akan berjalan.

Tabel 3. 11 Usecase Skenario Mencari Data Peserta

Nama Use Case	Mencari Data Peserta
Deskripsi	Digunakan untuk menampilkan data peserta secara keseluruhan maupun secara spesifik
Aktor	Admin
Kondisi Awal	Admin telah <i>login</i> dan GUI peserta telah menampilkan seluruh data peserta
Kondisi Akhir	Sistem berhasil menampilkan data peserta yang dicari
Skenario	<ol style="list-style-type: none">1. Sistem menampilkan data peserta secara keseluruhan2. Admin memasukkan nama / jurusan / tahun untuk mencari data peserta lebih cepat3. Sistem melakukan pencarian sesuai dengan nama / jurusan / tahun yang dimasukkan admin4. Sistem menampilkan data peserta yang telah ditemukan
Skenario Alternatif	4a. [Sistem tidak menemukan data peserta yang diinputkan] kembali ke langkah 1.

Tabel 3.11 menjelaskan apabila pengguna mencari data peserta dengan data yang benar sistem akan berjalan sesuai dengan skenario sedangkan jika pengguna mencari data admin yang salah maka skenario alternatif akan berjalan.

Tabel 3. 12 Usecase Skenario Menambah Data Paket Soal

Nama Use Case	Menambah data paket soal
Deskripsi	Digunakan untuk menambah data paket soal
Aktor	Admin
Kondisi Awal	Admin telah <i>login</i> dan membuka <i>GUI</i> paket soal
Kondisi Akhir	Sistem berhasil menambahkan data paket soal
Skenario	<ol style="list-style-type: none">1. Admin memasukkan data paket soal2. Sistem memverifikasi inputan data paket soal dan memeriksa apakah kode paket sudah terdaftar atau belum3. Sistem menyimpan data paket soal try out baru4. Sistem menampilkan pesan data paket berhasil ditambahkan5. Sistem menampilkan kembali seluruh daftar

	data paket
Skenario Alternatif	<p>2a. [Kode paket telah terdaftar] menampilkan pesan kode paket telah terdaftar, penambahan data paket gagal, kembali ke langkah 1.</p> <p>2b. [Data paket tidak lengkap] menampilkan pesan data paket tidak lengkap, penambahan data paket gagal, kembali ke langkah 1.</p>

Tabel 3.12 menjelaskan apabila pengguna menambah data paket soal dengan prosedur yang benar sistem akan berjalan sesuai dengan skenario sedangkan jika pengguna menambah data paket soal dengan prosedur yang salah maka skenario alternatif akan berjalan.

Tabel 3. 13 Usecase Skenario Menghapus Data Paket Soal

Nama Use Case	Menghapus data paket soal
Deskripsi	Digunakan untuk menghapus data paket soal apabila data tersebut terdapat kesalahan
Aktor	Admin
Kondisi Awal	Admin telah <i>login</i> dan GUI paket soal telah menampilkan seluruh data paket soal
Kondisi Akhir	Sistem berhasil menghapus data paket yang telah dipilih
Skenario	<ol style="list-style-type: none"> 1. Sistem menampilkan seluruh data paket soal 2. Admin mencari data paket soal dengan memasukkan kata kunci kode paket / periode 3. Sistem menampilkan data paket yang ditemukan 4. Admin memilih data yang akan dihapus 5. Sistem menampilkan pesan konfirmasi 6. Admin memasukkan instruksi sesuai dengan pilihan yang tersedia pada pesan konfirmasi (yes jika menyetujui , no jika batal) 7. Sistem menghapus data yang dipilih 8. Sistem menampilkan pesan data berhasil dihapus 9. Sistem menampilkan kembali data paket soal
Skenario Alternatif	5a. [Admin memilih opsi <i>no</i> / tidak] sistem menutup <i>message dialog</i> kemudian sistem merefresh data paket dan menampilkan kembali.

Tabel 3.13 menjelaskan apabila pengguna menghapus data paket soal dengan prosedur yang benar sistem akan berjalan sesuai dengan skenario sedangkan jika pengguna menghapus data paket soal dengan prosedur yang salah maka skenario alternatif akan berjalan.

Tabel 3. 14 Usecase Skenario Mencari Data Paket Soal

Nama Use Case	Mencari data paket soal
Deskripsi	Digunakan untuk menampilkan data paket soal secara keseluruhan maupun secara spesifik
Aktor	Admin
Kondisi Awal	Admin telah <i>login</i> dan GUI paket telah menampilkan seluruh data paket soal
Kondisi Akhir	Sistem berhasil menampilkan data paket soal yang dicari
Skenario	<ol style="list-style-type: none"> 1. Sistem menampilkan data paket secara keseluruhan 2. Admin memasukkan kode paket/ tahun / periode try out untuk mencari data paket soal 3. Sistem melakukan pencarian sesuai dengan kode paket / tahun / periode try out yang dimasukkan admin 4. Sistem menampilkan data paket soal yang telah ditemukan
Skenario Alternatif	4a. [Sistem tidak menemukan data paket yang diinputkan] kembali ke langkah 1.

Tabel 3.15 menjelaskan apabila pengguna mencari data paket soal dengan data yang benar sistem akan berjalan sesuai dengan skenario sedangkan jika pengguna mencari data paket soal yang salah maka skenario alternatif akan berjalan.

Tabel 3. 15 Usecase Skenario Memasukkan Citra LJK

Nama Use Case	Memasukkan citra LJK
Deskripsi	Digunakan untuk mengimport citra LJK yang akan diolah
Aktor	Admin
Kondisi Awal	Admin telah <i>login</i> dan GUI koreksi telah dibuka
Kondisi Akhir	Sistem berhasil menampilkan citra LJK yang diimport

Skenario	<ol style="list-style-type: none"> 1. Admin memilih citra yang ingin dimasukkan 2. Sistem memasukkan citra yang dipilih 3. Sistem menampilkan nama file citra yang sudah dipilih 4. Admin memilih citra yang akan diproses 5. Sistem menampilkan citra yang dipilih
Skenario Alternatif	-

Tabel 3.15 skenario pengguna dalam memasukkan citra LJK yang mana sistem akan memfilter citra yang dapat dimasukkan terlebih dahulu seperti citra dengan format jpg atau png.

Tabel 3. 16 Usecase Skenario Mendeteksi Informasi LJK dengan Metode Canny

Nama Use Case	Mendeteksi informasi LJK dengan metode Cannya
Deskripsi	Sistem dapat mendeteksi informasi berupa kode peserta, kode paket, jurusan, mata pelajaran dan jawaban dari LJK
Kondisi Awal	Admin telah memilih citra LJK yang akan diproses dan sistem telah menampilkan citra LJK yang akan diproses.
Kondisi Akhir	Sistem berhasil mendeteksi informasi dari citra LJK
Skenario	<ol style="list-style-type: none"> 1. Sistem melakukan pembagian part jawaban, kode soal, jurusan, kode peserta 2. Melakukan threshold pada setiap bagian yang sudah dibagi 3. Mendeteksi objek dengan metode canny 4. Mendeteksi objek lingkaran 5. Menentukan koordinat pada masing-masing bagian 6. Menentukan hasil (jawaban, kode peserta, kode soal, jurusan) 7. Menampilkan masing-masing hasil pada texfield yang sesuai dengan bagian yang ditentukan
Skenario Alternatif	-

Tabel 3.16 skenario bagaimana sistem menerapkan metode canny ketika pengguna telah memasukkan banyak citra dan sudah memilih data yang akan diproses terlebih dahulu.

Tabel 3. 17 Usecase Skenario Memeriksa Jawaban Try Out Siswa

Nama Use Case	Memeriksa jawaban siswa
Deskripsi	Admin memeriksa hasil jawaban siswa yang telah disimpan di database
Kondisi Awal	Sistem telah mendeteksi informasi serta jawaban dari LJK
Kondisi Akhir	Sistem berhasil mendeteksi jumlah jawaban yang benar dan yang salah serta menentukan nilai
Skenario	<ol style="list-style-type: none">1. Sistem telah mendeteksi informasi serta jawaban dari LJK2. Sistem mengambil data kunci jawaban berdasarkan informasi yang diterima dari LJK3. Sistem mencocokkan informasi yang diterima dari LJK4. Sistem berhasil mendeteksi jumlah jawaban yang benar dan yang salah serta menentukan nilai
Skenario Alternatif	2a. [Sistem Tidak berhasil menemukan informasi yang cocok dengan informasi LJK] sistem menampilkan pesan data informasi tidak ditemukan.

Tabel 3.17 skenario bagaimana sistem memeriksa informasi yang telah didapat dari LJK dan mencocokkannya dengan yang ada didatabase. Jika informasi tidak cocok maka scenario alternatif akan berjalan.

Tabel 3. 18 Usecase Skenario Menyimpan Data Try Out Siswa

Nama Use Case	Menyimpan Jawaban LJK
Deskripsi	Admin dapat menyimpan jawaban hasil deteksi citra LJK
Kondisi Awal	Gui koreksi dalam kondisi aktif dan sistem telah memeriksa informasi dari citra LJK
Kondisi Akhir	Informasi citra LJK dan hasil koreksi berhasil disimpan
Skenario	<ol style="list-style-type: none">1. Sistem memeriksa informasi dari citra LJK2. Sistem melakukan validasi data yang akan disimpan3. Sistem menampilkan informasi presentase proses penyimpanan data4. Sistem berhasil menyimpan informasi citra LJK dan hasil koreksi
Skenario Alternatif	-

Tabel 3.18 merupakan skenario bagaimana sistem menyimpan informasi yang telah diperiksa ke dalam database sehingga data tersebut dapat diolah lebih lanjut.

Tabel 3. 19 Usecase Skenario Mencari Data Try Out

Nama Use Case	Menampilkan data try out
Deskripsi	Sistem menampilkan data peserta beserta hasil nilai try out yang telah dikoreksi
Kondisi Awal	Data hasil koreksi telah tersimpan
Kondisi Akhir	Sistem dapat menampilkan data peserta beserta hasil nilai try out
Skenario	<ol style="list-style-type: none"> 1. Sistem menampilkan data peserta beserta nilai try out keseluruhan secara auto ketika gui terbuka 2. Admin memasukkan kata kunci nama atau jurusan 3. Sistem menampilkan data peserta beserta hasil nilai try out siswa
Skenario Alternatif	3a. [Data try out tidak ditemukan] tabel data try out dikosongkan.

Tabel 3.19 menjelaskan apabila pengguna mencari data try out dengan data yang benar sistem akan berjalan sesuai dengan skenario sedangkan jika pengguna mencari data try out yang salah maka skenario alternatif akan berjalan.

Tabel 3. 20 Usecase Skenario Mencetak Data Hasil Try Out

Nama Use Case	Mencetak data hasil try out
Deskripsi	Sistem dapat mencetak hasil data try out dalam format PDF
Kondisi Awal	Sistem telah menampilkan data yang sebelumnya telah dicari
Kondisi Akhir	Sistem menampilkan data dalam bentuk file pdf yang siap cetak
Skenario	<ol style="list-style-type: none">1. Sistem menampilkan data yang sebelumnya dicari2. Sistem memasukkan data dalam format file pdf3. Sistem menampilkan data dalam bentuk file pdf yang siap cetak
Skenario Alternatif	1a. [Data try out tidak ditemukan] menampilkan file pdf dengan data kosong.

Tabel 3.20 skenario mencetak data hasil try out yang mana data tersebut harus dicari terlebih dahulu kemudian data dikonversi dalam bentuk data siap cetak. Apabila data tidak ditemukan maka scenario alternatif akan berjalan.

Tabel 3. 21 Usecase Skenario Mencari Jumlah Jawaban Siswa Salah

Nama Use Case	Mencari jumlah jawaban siswa salah
Deskripsi	Sistem dapat menampilkan jumlah siswa salah dan benar pada setiap soal
Kondisi Awal	data peserta, jawaban serta nilai sudah tersimpan di database
Kondisi Akhir	Sistem dapat menampilkan jumlah siswa yang salah dan jumlah siswa yang benar.
Skenario	<ol style="list-style-type: none">1. Sistem menampilkan jumlah siswa salah dan jumlah siswa benar secara keseluruhan ketika gui dibuka2. Admin memasukkan kata kunci kode paket dan periode3. Sistem menampilkan data jumlah siswa salah dan jumlah siswa benar berdasarkan kata kunci
Skenario Alternatif	3a. [Data jumlah siswa salah tidak ditemukan] tabel data try out dikosongkan.

Tabel 3.21 menjelaskan apabila pengguna mencari data jumlah siswa salah dengan data yang benar sistem akan berjalan sesuai dengan skenario

sedangkan jika pengguna mencari data jumlah siswa salah dengan kata kunci yang salah / tidak sesuai maka skenario alternatif akan berjalan.

Tabel 3. 22 Usecase Skenario Menampilkan Grafik Jumlah Siswa Salah

Nama Use Case	Menampilkan grafik jumlah siswa salah
Deskripsi	Sistem menampilkan grafik jumlah siswa salah berdasarkan kode paket dan periode pada setiap soal
Aktor	Admin
Kondisi Awal	Admin telah <i>login</i> dan GUI JSS
Kondisi Akhir	Sistem berhasil menampilkan grafik jumlah siswa salah
Skenario	<ol style="list-style-type: none"> 1. Sistem menampilkan semua jumlah siswa salah berdasarkan kode paket dan periode 2. Admin memilih kode paket soal dan memasukkan periode ujian 3. Sistem melakukan pencarian data sesuai dengan kode paket dan periode yang dimasukkan admin 4. Sistem menampilkan data jumlah siswa salah pada setiap soal dalam bentuk grafik
Skenario Alternatif	4a. [Data tidak ditemukan] sistem menampilkan pesan data tidak ditemukan.

Tabel 3.22 menjelaskan apabila pengguna mencari data jumlah siswa salah dengan data yang benar sistem akan berjalan sesuai dengan skenario sedangkan jika pengguna mencari data jumlah siswa salah dengan kata kunci yang salah / tidak ditemukan maka skenario alternatif akan berjalan.

Tabel 3. 23 Usecase Skenario Mengeksport Data Jumlah Siswa Salah

Nama Use Case	mengeksport data jumlah siswa salah
Deskripsi	Sistem dapat mengeksport data jumlah siswa salah dalam bentuk file excel untuk dokumentasi.
Aktor	Admin
Kondisi Awal	Admin telah <i>login</i> dan GUI JSS telah terbuka
Kondisi Akhir	Sistem berhasil mengeksport data dalam bentuk file excel
Skenario	<ol style="list-style-type: none"> 1. Sistem menampilkan semua jumlah siswa salah berdasarkan kode paket dan periode 2. Admin memilih kode paket soal dan memasukkan periode ujian 3. Sistem melakukan pencarian data sesuai dengan

	kode paket dan periode yang dimasukkan admin 4. Sistem mengekspor data dalam bentuk format xls (file excel)
Skenario Alternatif	-

Tabel 3.13 merupakan skenario jalannya sistem ketika pengguna ingin mengekspor data jumlah siswa salah dalam bentuk file .xls. Dimana sistem tersebut akan mencari data yang diinginkan pengguna. Apabila data tidak ditemukan maka file excel yang diexport akan kosong.

Tabel 3. 24 Usecase Skenerio Menambah Data Mapel

Nama Use Case	Menambah data mapel
Deskripsi	Sistem dapat menambah data mapel yang akan diujikan kepada peserta
Aktor	Admin
Kondisi Awal	Admin telah <i>login</i> dan membuka GUI mapel
Kondisi Akhir	Sistem berhasil menambahkan data mapel
Skenario	<ol style="list-style-type: none"> 1. Admin memasukkan data mata pelajaran 2. Sistem memverifikasi inputan data mata pelajaran dan memeriksa kelengkapan data 3. Sistem menyimpan data mata pelajaran 4. Sistem menampilkan pesan data mata pelajaran berhasil ditambahkan 5. Sistem menampilkan kembali seluruh daftar data mata pelajaran
Skenario Alternatif	<ol style="list-style-type: none"> 2a. [Inputan data mapel tidak valid] menampilkan pesan inputan data tidak valid, penambahan data mata pelajaran gagal, kembali ke langkah 1. 2b. [Data mata pelajaran tidak lengkap] menampilkan pesan data mata pelajaran tidak lengkap, penambahan data mata pelajaran gagal, kembali ke langkah 1.

Tabel 3.24 menjelaskan apabila pengguna menambah data mapel dengan prosedur yang benar sistem akan berjalan sesuai dengan skenario sedangkan jika pengguna menambah data mapel dengan prosedur yang salah maka skenario alternatif akan berjalan.

Tabel 3. 25 Usecase Mengubah Data Mata Pelajaran

Nama Use Case	Mengubah data mata pelajaran
Deskripsi	Sistem dapat mengubah semua data mata pelajaran kecuali kode mata pelajaran
Aktor	Admin
Kondisi Awal	Admin telah <i>login</i> dan membuka GUI mata pelajaran
Kondisi Akhir	Sistem berhasil mengubah data mata pelajaran
Skenario	<ol style="list-style-type: none">1. Sistem menampilkan seluruh data mata pelajaran yang telah terdaftar2. Admin memasukkan kata kunci pencarian (nama mata pelajaran)3. Sistem menampilkan data yang dimaksud4. Admin memilih data yang akan diubah5. Admin memasukkan perubahan data6. Sistem memverifikasi perubahan data mata pelajaran7. Sistem menyimpan perubahan data yang dipilih admin8. Sistem menampilkan pesan data mata pelajaran berhasil diubah9. Sistem menampilkan kembali seluruh data mata pelajaran
Skenario Alternatif	4a. [Data mata pelajaran tidak lengkap] menampilkan pesan data mata pelajaran tidak lengkap, perubahan data mata pelajaran gagal, kembali ke langkah 1.

Tabel 3.25 menjelaskan apabila pengguna mengubah data mata pelajaran dengan prosedur yang benar sistem akan berjalan sesuai dengan skenario sedangkan jika pengguna mengubah data mata pelajaran dengan prosedur yang salah maka skenario alternatif akan berjalan.

Tabel 3. 26 Usecase Skenario Mencari Data Mata Pelajaran

Nama Use Case	Mencari data mata pelajaran
Deskripsi	Sistem dapat menampilkan data peserta secara keseluruhan maupun secara spesifik sesuai dengan instruksi admin
Aktor	Admin
Kondisi Awal	Admin telah <i>login</i> dan GUI mata pelajaran telah menampilkan seluruh data mata pelajaran
Kondisi Akhir	Sistem berhasil menampilkan data mata pelajaran yang dicari

Skenario	<ol style="list-style-type: none"> 1. Sistem menampilkan data mata pelajaran secara keseluruhan 2. Admin memasukkan kata kunci nama mata pelajaran untuk mencari data mata pelajaran 3. Sistem menampilkan data mata pelajaran yang telah ditemukan sesuai dengan kata kunci yang dimasukkan
Skenario Alternatif	3a [Data mata pelajaran tidak ditemukan] sistem tidak menampilkan data yang dicari dan tabel mata pelajaran dikosongkan

Tabel 3.26 menjelaskan apabila pengguna mencari data mata pelajaran dengan data yang benar sistem akan berjalan sesuai dengan skenario sedangkan jika pengguna mencari data mata pelajaran yang salah maka skenario alternatif akan berjalan.

Tabel 3. 27 Usecase Skenerio Menambah Data Jurusan

Nama Use Case	Menambah data jurusan
Deskripsi	Sistem dapat menambah data jurusan sehingga jurusan yang diambil siswa akan lebih fleksibel
Aktor	Admin
Kondisi Awal	Admin telah <i>login</i> dan membuka GUI jurusan
Kondisi Akhir	Sistem berhasil menambahkan data jurusan
Skenario	<ol style="list-style-type: none"> 1. Admin memasukkan data jurusan 2. Sistem memverifikasi inputan data jurusan dan memeriksa apakah kode jurusan sudah terdaftar atau belum 3. Sistem menyimpan data jurusan 4. Sistem menampilkan pesan data jurusan berhasil ditambahkan 5. Sistem menampilkan kembali seluruh daftar data jurusan
Skenario Alternatif	<ol style="list-style-type: none"> 2a. [Nama jurusan telah terdaftar] menampilkan jurusan telah terdaftar, penambahan data jurusan gagal, kembali ke langkah 1. 2b. [Data jurusan tidak lengkap] menampilkan pesan data jurusan tidak lengkap, penambahan data jurusan gagal, kembali ke langkah 1.

Tabel 3.29 menjelaskan apabila pengguna menambah data jurusan dengan prosedur yang benar sistem akan berjalan sesuai dengan skenario sedangkan jika pengguna menambah data jurusan dengan prosedur yang salah maka skenario alternatif akan berjalan.

Tabel 3. 28 Usecase Skenario Mengubah Data Jurusan

Nama Use Case	Mengubah data jurusan
Deskripsi	Admin dapat mengubah semua data jurusan
Aktor	Admin
Kondisi Awal	Admin telah <i>login</i> dan membuka GUI jurusan
Kondisi Akhir	Sistem berhasil mengubah data jurusan
Skenario	<ol style="list-style-type: none"> 1. Sistem menampilkan seluruh data jurusan yang telah terdaftar 2. Admin mencari data jurusan sesuai dengan kode jurusan 3. Sistem menampilkan data yang dicari 4. Admin memilih data yang akan diubah 5. Memasukkan perubahan data jurusan 6. Sistem memverifikasi perubahan data jurusan 7. Sistem menyimpan perubahan data jurusan 8. Sistem menampilkan pesan data jurusan berhasil diubah 9. Sistem menampilkan kembali seluruh data jurusan
Skenario Alternatif	7a. [Data jurusan tidak lengkap] menampilkan pesan data jurusan tidak lengkap, perubahan data jurusan gagal, kembali ke langkah 1.

Tabel 3.28 menjelaskan apabila pengguna mengubah data jurusan dengan prosedur yang benar sistem akan berjalan sesuai dengan skenario sedangkan jika pengguna mengubah data jurusan dengan prosedur yang salah maka skenario alternatif akan berjalan.

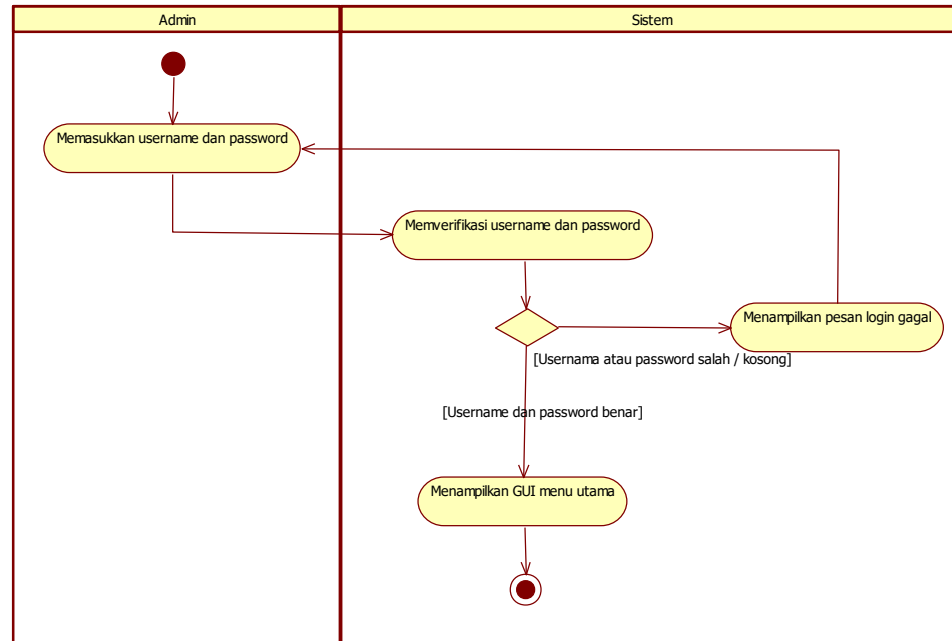
Tabel 3. 29 Usecase Skenario Mencari Data Jurusan

Nama Use Case	Mencari data jurusan
Deskripsi	Sistem dapat menampilkan jurusan secara keseluruhan maupun secara spesifik sesuai dengan instruksi admin
Aktor	Admin
Kondisi Awal	Admin telah <i>login</i> dan GUI jurusan telah menampilkan

	seluruh data jurusan
Kondisi Akhir	Sistem berhasil menampilkan data jurusan yang dicari
Skenario	<ol style="list-style-type: none"> 1. Sistem menampilkan data jurusan secara keseluruhan 2. Admin memasukkan kata kunci kode jurusan untuk mencari data jurusan 3. Sistem menampilkan data jurusan yang telah ditemukan
Skenario Alternatif	3a [Data jurusan tidak ditemukan] sistem tidak menampilkan data yang dicari dan tabel dikosongkan.

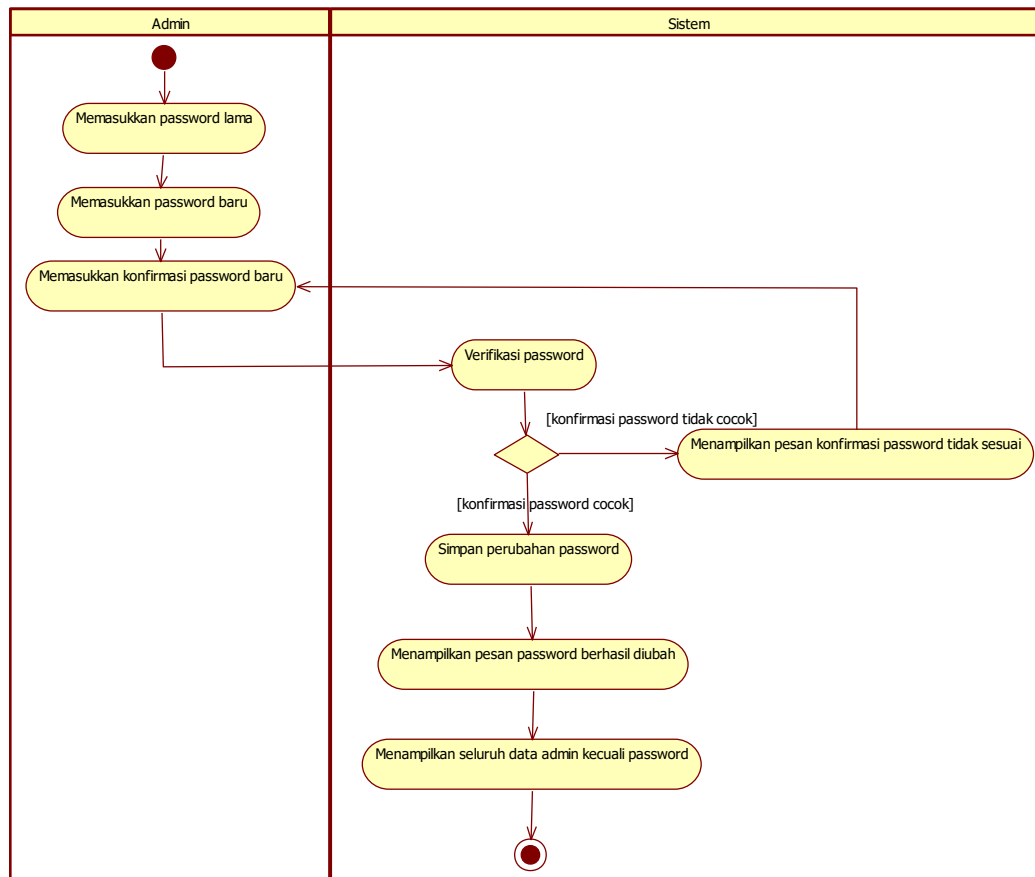
Tabel 3.29 menjelaskan apabila pengguna mencari data jurusan dengan data yang benar sistem akan berjalan sesuai dengan skenario sedangkan jika pengguna mencari data jurusan yang salah maka skenario alternatif akan berjalan.

3.7.3 Activity Diagram



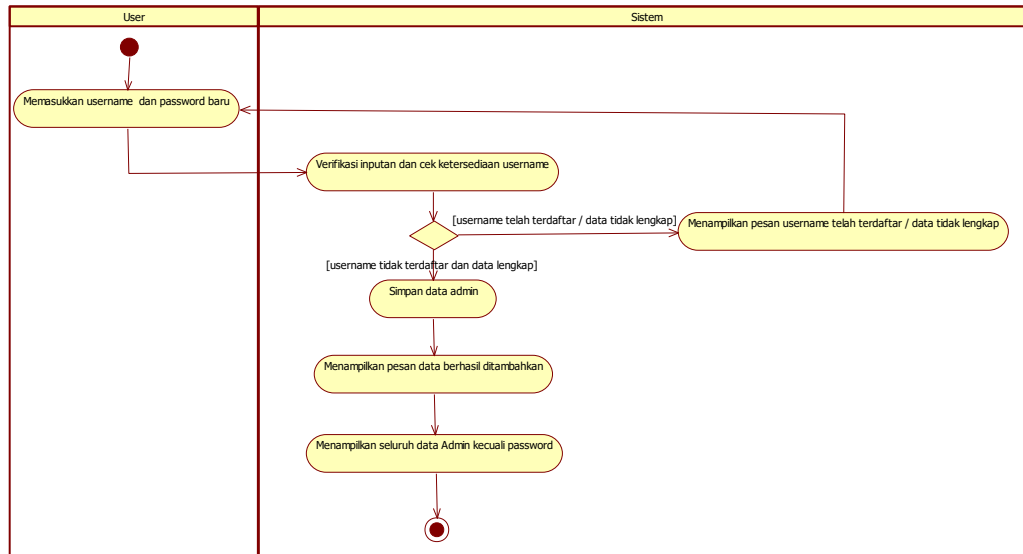
Gambar 3. 21 Activity Diagram Login

Pada gambar 3.21 menunjukkan alur aktivitas pada saat login yaitu pertama pengguna memasukkan *username* dan *password* kemudian sistem akan memverifikasi inputan pengguna apabila pengguna terdaftar sebagai admin maka sistem akan membuka GUI menu utama.



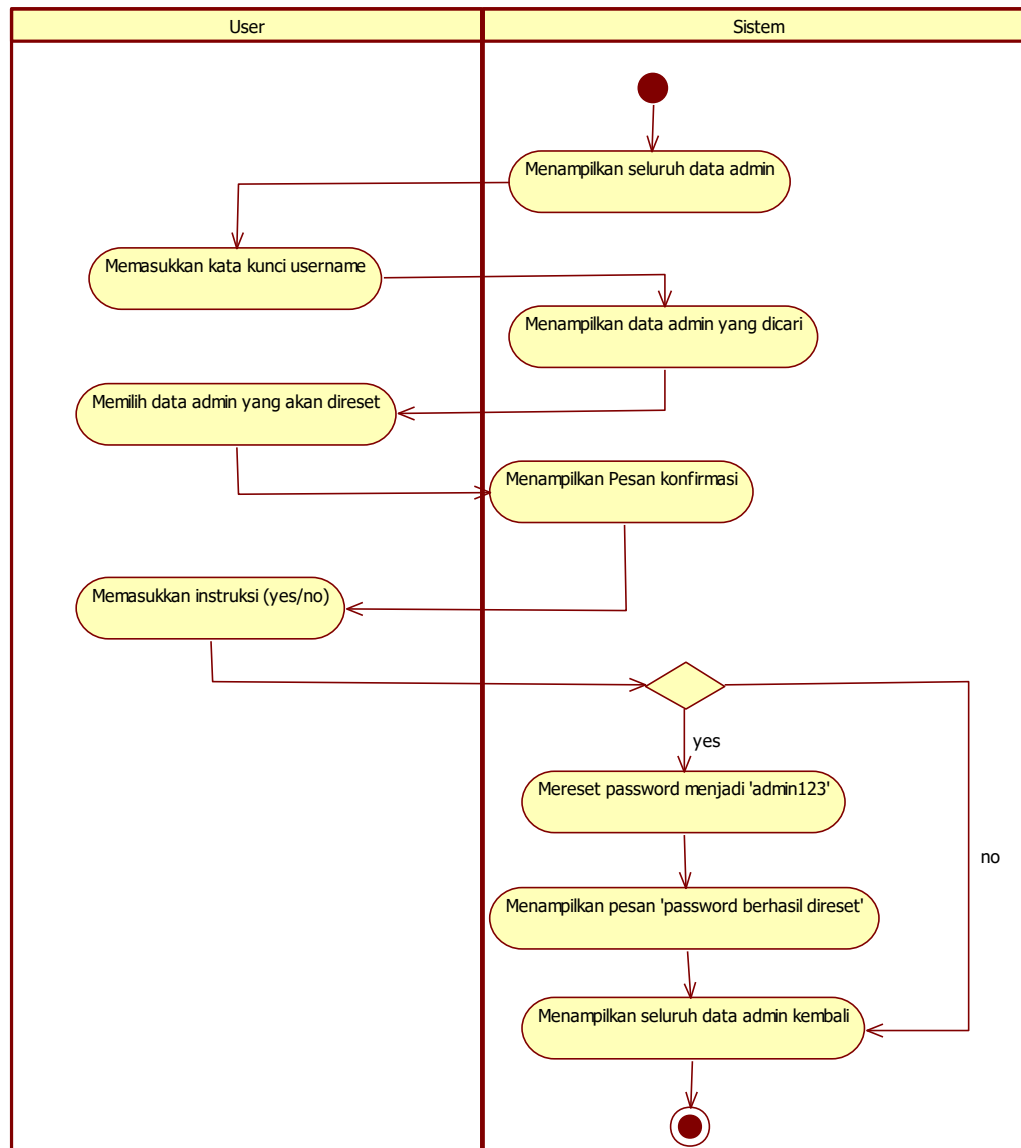
Gambar 3. 22 Activity Diagram Mengubah *Password* Admin

Pada gambar 3.22 merupakan alur aktivitas mengubah *password* admin (*password* pribadi) langkah pertama admin akan memasukkan *password* lama kemudian admin harus memasukkan *password* baru beserta konfirmasi *password* yang sama dengan *password* yang baru. Selanjutnya sistem akan memverifikasi *password* yang telah dimasukkan apabila tidak sesuai maka sistem akan menampilkan pesan konfirmasi *password* tidak cocok agar admin dapat menggantinya. Jika konfirmasi *password* sudah sesuai maka sistem akan menyimpan perubahan *password* dan menampilkan pesan *password* berhasil diubah. Terakhir sistem menampilkan seluruh data admin kecuali *password* digunakan untuk *refresh* data admin.



Gambar 3. 23 Activity Diagram Menambah Admin

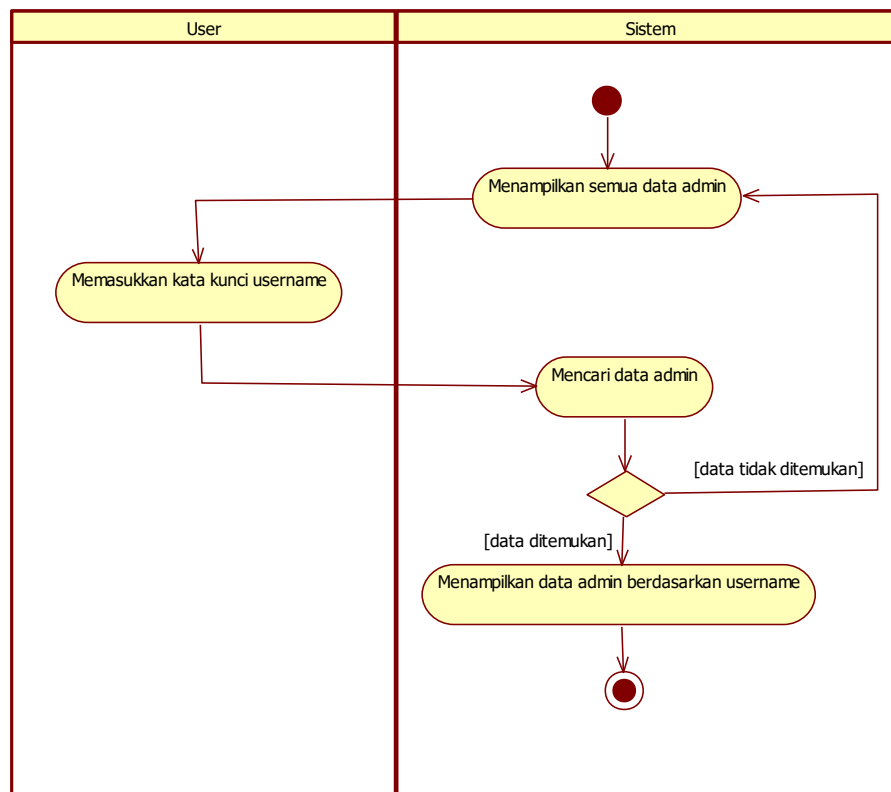
Pada gambar 3.23 admin dapat menambah anggota baru agar dapat membantu meringankan pekerjaan apabila dia sedang berhalangan. Langkah pertama dengan memasukkan *username* dan *password* baru kemudian sistem memverifikasi inputan serta cek ketersediaan *username*. Apabila *username* telah terdaftar maka sistem akan menampilkan pesan bahwa *username* sudah tersedia sehingga pengguna dapat mengganti *username* dengan nama lain. Apabila *username* belum ada maka sistem akan menyimpan data admin baru kemudian sistem menampilkan pesan data admin berhasil ditambah dan langkah terakhir sistem *refresh* data admin dan menampilkannya.



Gambar 3. 24 Activity Diagram Mereset *Password*

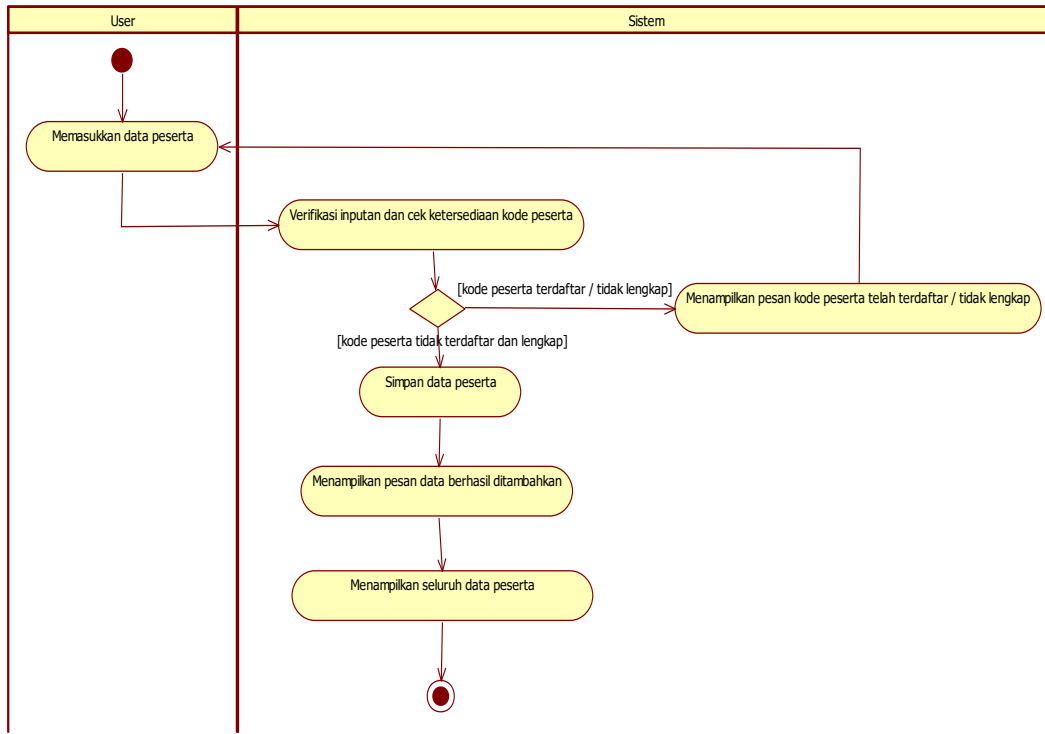
Pada gambar 3.24 merupakan alur aktivitas yang akan dijalankan ketika akan mereset *password* admin. Pertama sistem akan menampilkan seluruh data admin kemudian untuk mempercepat menemukan data data admin maka pengguna dapat mencari dengan memasukkan username admin yang dicari. Sistem akan menampilkan data admin yang dicari didalam tabel daftar admin. Selanjutnya admin

dapat memilih pengguna yang akan direset *password*. Kemudian sistem akan menampilkan pesan konfirmasi untuk meyakinkan pengguna. Jika pengguna / admin memberi instruksi *yes* maka *password* akan direset dan sistem akan menampilkan pesan bahwa *password* berhasil direset jika *no* maka sistem akan menutup pesan konfirmasi dan menampilkan seluruh data admin kembali / *refresh* data.



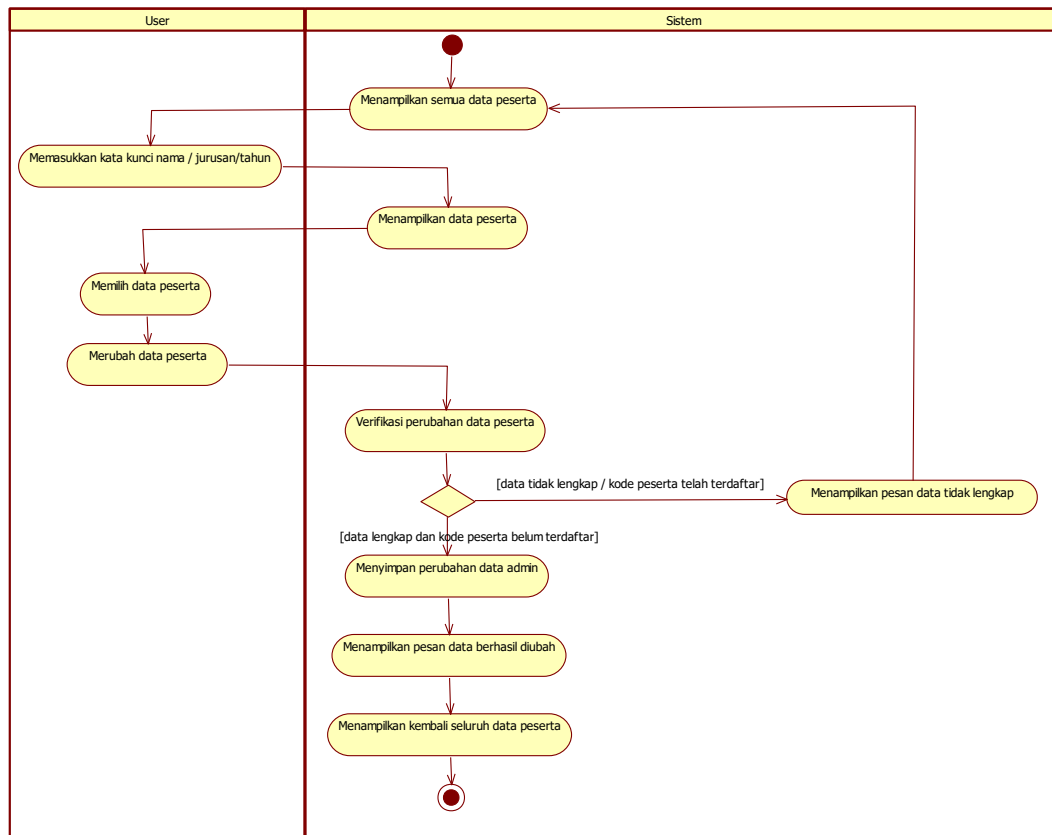
Gambar 3. 25 Activity Diagram Mencari Data Admin

Pada gambar 3.25 merupakan alur aktifitas mencari data admin. Pertama sistem menampilkan seluruh data admin dalam tabel daftar admin kemudian admin / pengguna dapat memasukkan kata kunci *username* agar dapat mencari data admin lebih cepat. Sistem akan melakukan pencarian data admin yang dimaksud apabila data tidak ditemukan maka tabel akan dikosongkan dan kembali ke langkah awal. Dan apabila data admin ditemukan maka akan ditampilkan pada tabel daftar admin yang tersedia.



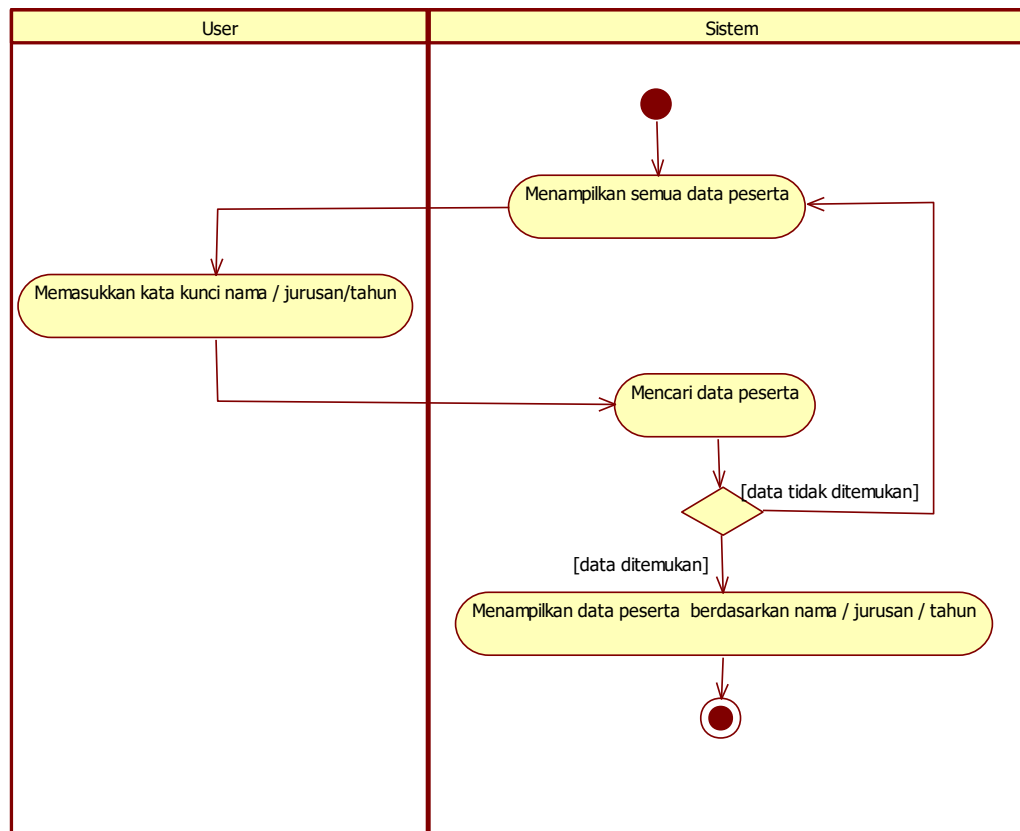
Gambar 3. 26 Activity Diagram Menambah Data Peserta

Gambar 3.26 merupakan alur aktifitas menambah data peserta baru. Pertama pengguna / admin memasukkan data peserta kemudian sistem akan memverifikasi inputan pengguna sekaligus memeriksa apakah kode peserta sudah terdaftar atau belum dan cek keterlengkapan data yang dimasukkan apabila kode peserta sudah terdaftar / ada salah satu data tidak lengkap maka sistem akan menampilkan pesan peringatan dan kembali ke langkah awal. Jika kode peserta belum terdaftar dan data yang dimasukkan lengkap maka sistem akan langsung menyimpan data peserta baru dan menampilkan pesan data berhasil ditambahkan kemudian sistem akan refresh data peserta dan menampilkan semua data peserta.



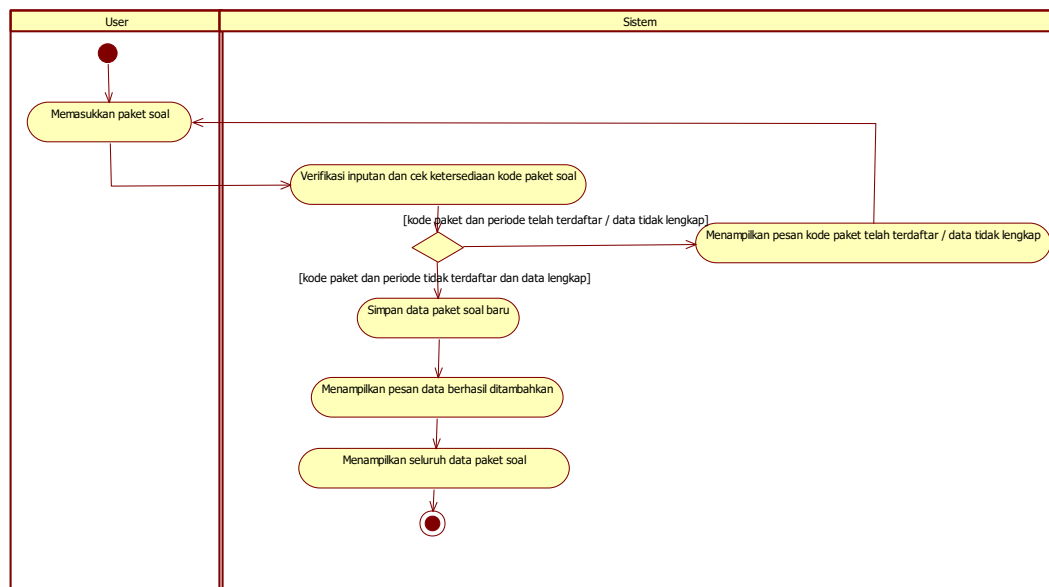
Gambar 3. 27 Activity Diagram Mengubah Data Peserta

Pada gambar 3.27 merupakan diagram aktifitas mengubah data peserta. Secara default sistem akan menampilkan seluruh daftar peserta try out kemudian untuk mempercepat menemukan data admin dapat melakukan pencarian dengan memasukkan kata kunci berupa nama / jurusan / tahun. Sistem akan menampilkan data yang ditemukan kemudian admin / pengguna dapat memilih data peserta yang akan dirubah. Selanjutnya admin dapat langsung merubah data peserta yang dipilih. Setelah dirubah sistem akan memverifikasi perubahan data sekaligus memeriksa keterlengkapan data. Apabila data tidak lengkap sistem akan menampilkan pesan bahwa data tidak lengkap. Jika data lengkap maka sistem akan menyimpan perubahan data menampilkan pesan data berhasil diubah sekaligus *refresh* data peserta.



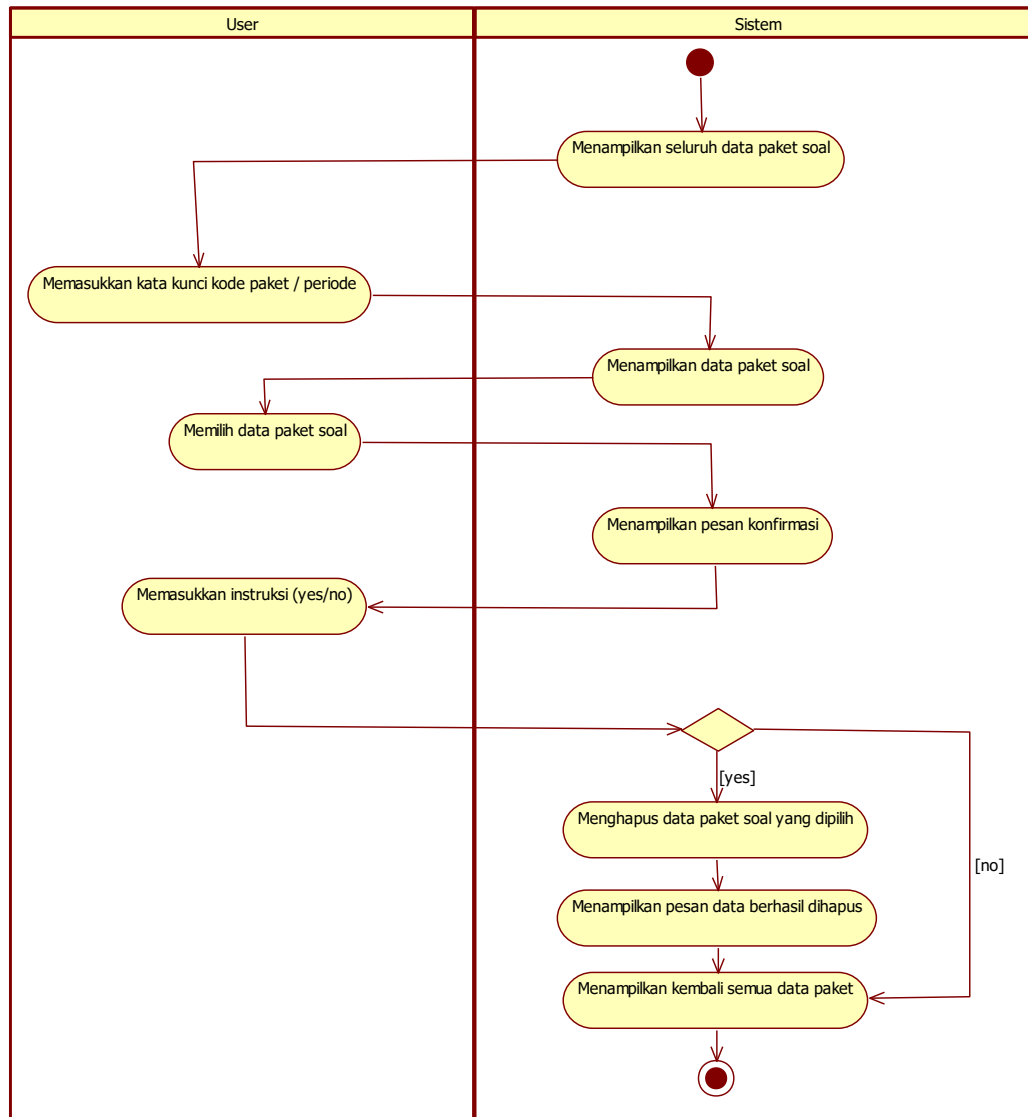
Gambar 3. 28 Activity Diagram Mencari Data Peserta

Pada gambar 3.28 merupakan alur aktifitas mencari data peserta. Pertama sistem menampilkan seluruh data peserta dalam tabel daftar peserta kemudian admin / pengguna dapat memasukkan kata kunci nama / jurusan / tahun agar dapat mencari data peserta lebih cepat. Sistem akan melakukan pencarian data peserta yang dimaksud apabila data tidak ditemukan maka tabel akan dikosongkan dan kembali ke langkah awal. Dan apabila data peserta ditemukan maka akan ditampilkan pada tabel daftar peserta yang tersedia.



Gambar 3. 29 Activity Diagram Menambah Data Paket Soal

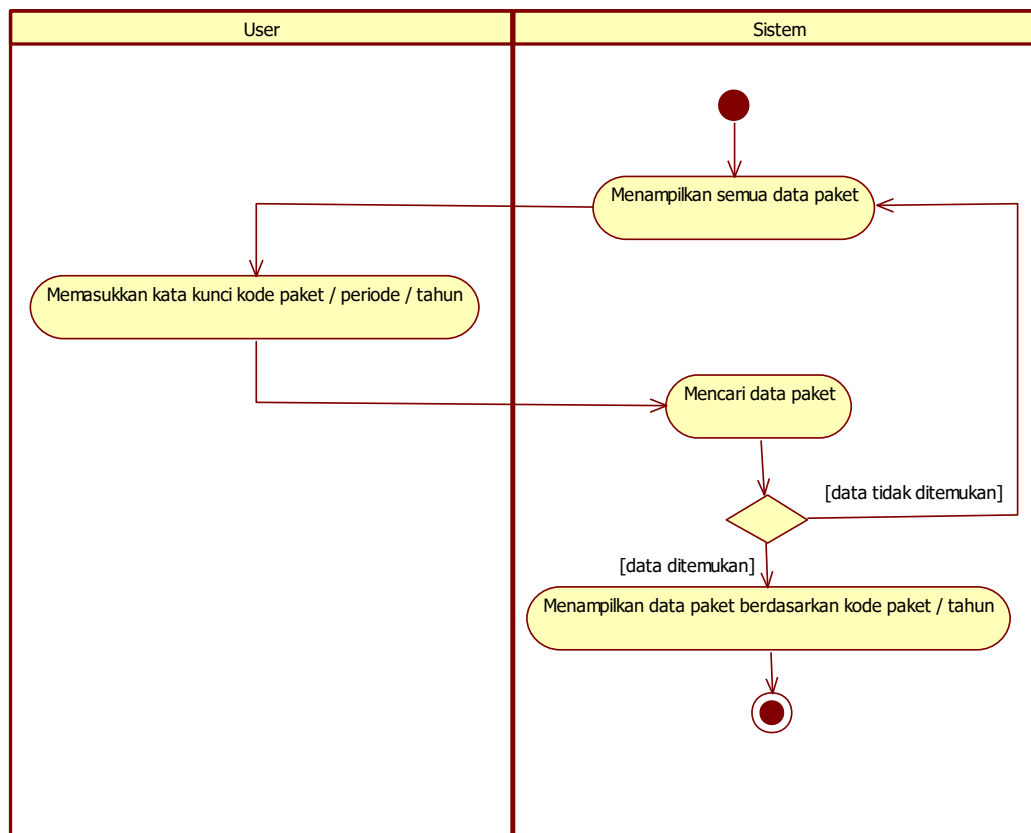
Gambar 3.29 merupakan alur aktifitas menambah data paket soal baru. Pertama pengguna / admin memasukkan data paket kemudian sistem akan memverifikasi inputan pengguna sekaligus memeriksa apakah kode paket sudah terdaftar atau belum dan cek keterlengkapan data yang dimasukkan apabila kode paket dengan periode yang sama sudah terdaftar / ada salah satu data tidak lengkap maka sistem akan menampilkan pesan peringatan dan kembali ke langkah awal. Jika kode paket dan periode belum terdaftar dan data yang dimasukkan lengkap maka sistem akan langsung menyimpan data paket baru dan menampilkan pesan data berhasil ditambahkan kemudian sistem akan refresh data paket dan menampilkan semua data paket sehingga paket yang digunakan sebagai acuan kunci jawaban dapat bersifat fleksibel.



Gambar 3.30 Activity Diagram Menghapus Data Paket Soal

Pada gambar 3.30 merupakan alur aktivitas yang akan dijalankan ketika akan menghapus data paket. Pertama sistem akan menampilkan seluruh data paket kemudian untuk mempercepat menemukan data admin maka pengguna / admin dapat mencari dengan memasukkan kode paket dan periode yang dicari. Sistem akan menampilkan data paket yang dicari didalam tabel daftar paket. Selanjutnya admin dapat memilih paket yang akan dihapus. Kemudian sistem akan menampilkan pesan

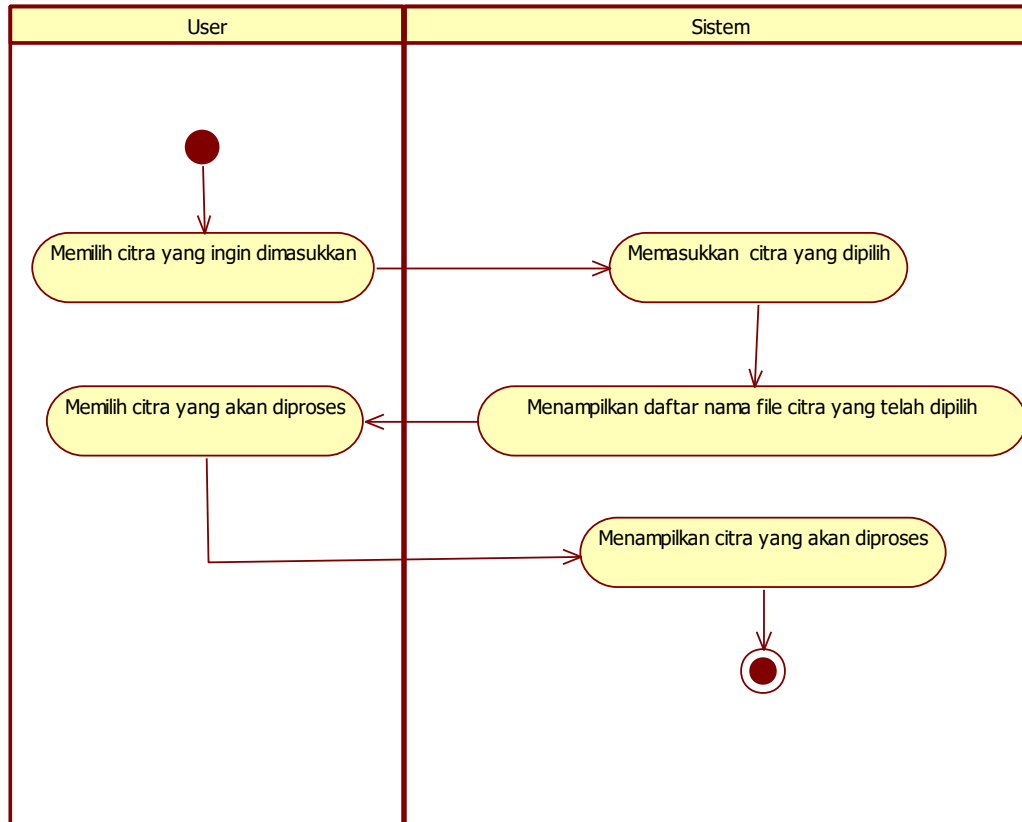
konfirmasi untuk meyakinkan pengguna apakah ingin benar-benar menghapus data yang dimaksud. Jika pengguna / admin memberi instruksi *yes* maka data paket tersebut akan dihapus dan sistem akan menampilkan pesan bahwa data paket berhasil dihapus jika *no* maka sistem akan menutup pesan konfirmasi dan menampilkan seluruh data paket kembali didalam tabel daftar paket yang tersedia / *refresh* data paket.



Gambar 3. 31 Activity Diagram Mencari Data Paket Soal

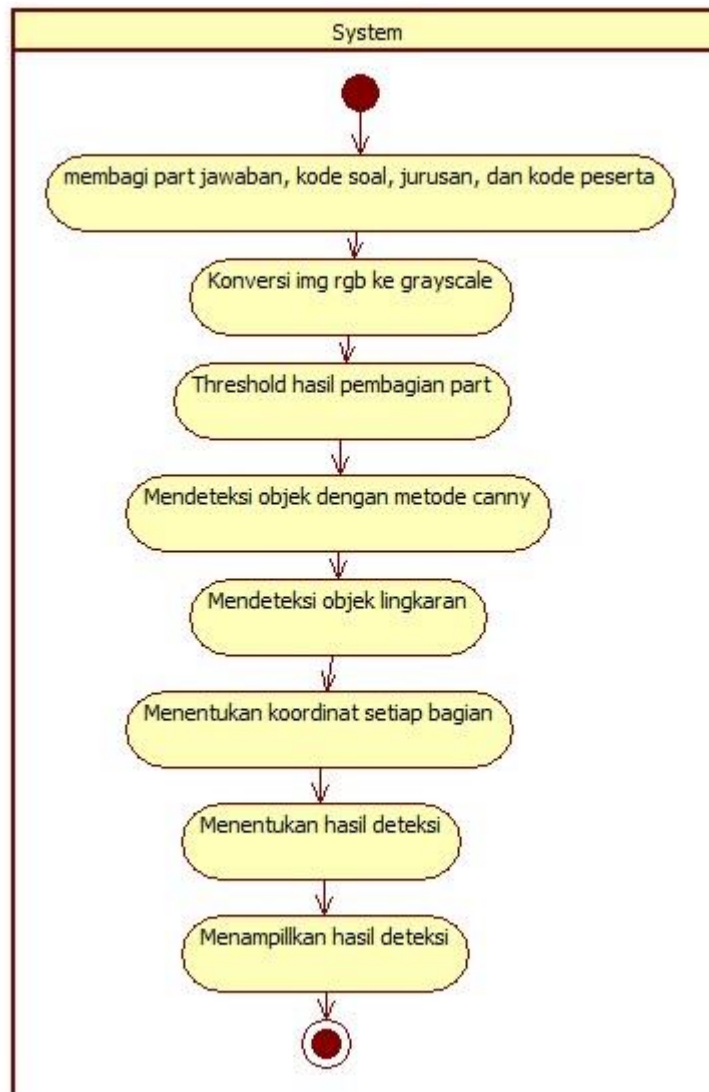
Pada gambar 3.31 merupakan alur aktifitas mencari data paket soal. Pertama sistem menampilkan seluruh data paket dalam tabel daftar paket kemudian admin / pengguna dapat memasukkan kata kunci kode paket / periode / tahun agar dapat mencari data paket lebih cepat. Sistem akan melakukan pencarian data paket yang dimaksud apabila data tidak ditemukan maka tabel akan dikosongkan dan kembali ke

langkah awal. Dan apabila data paket ditemukan maka akan ditampilkan pada tabel daftar paket yang tersedia.



Gambar 3. 32 Activity Diagram Memasukkan Citra LJK

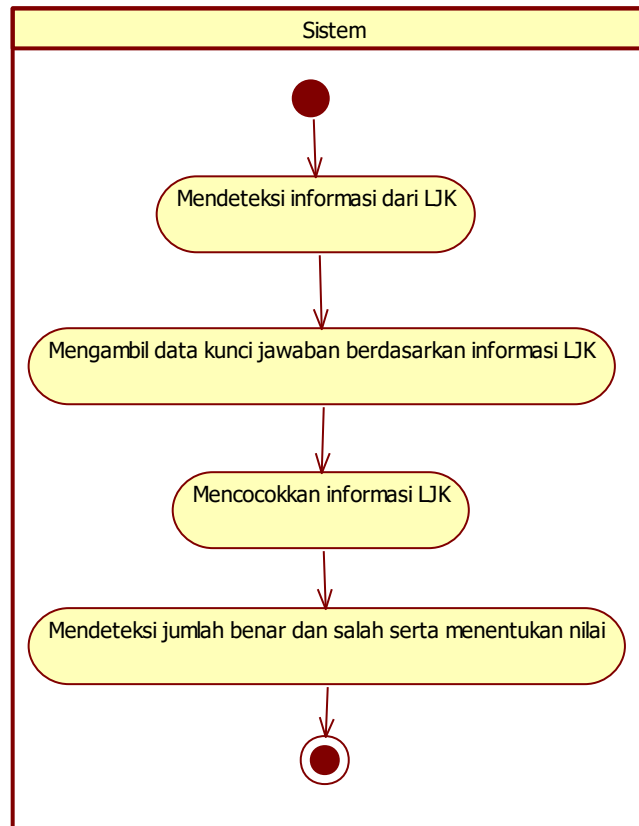
Gambar 3.32 merupakan diagram aktifitas memasukkan citra LJK / *import* citra. Pertama pengguna / admin memilih beberapa citra yang akan dimasukkan ke aplikasi pemilihan ini akan dilakukan ketika sistem telah menampilkan windows explorer dan mensetting format citra yang boleh di masukkan. Kemudian sistem memasukkan citra yang telah dipilih oleh admin / pengguna. Selanjutnya sistem menampilkan daftar nama citra yang dipilih admin. Admin selanjutnya akan memilih citra yang akan diproses dan sistem menampilkan citra yang akan proses.



Gambar 3. 33 Activity Digram Mendeteksi Informasi LJK

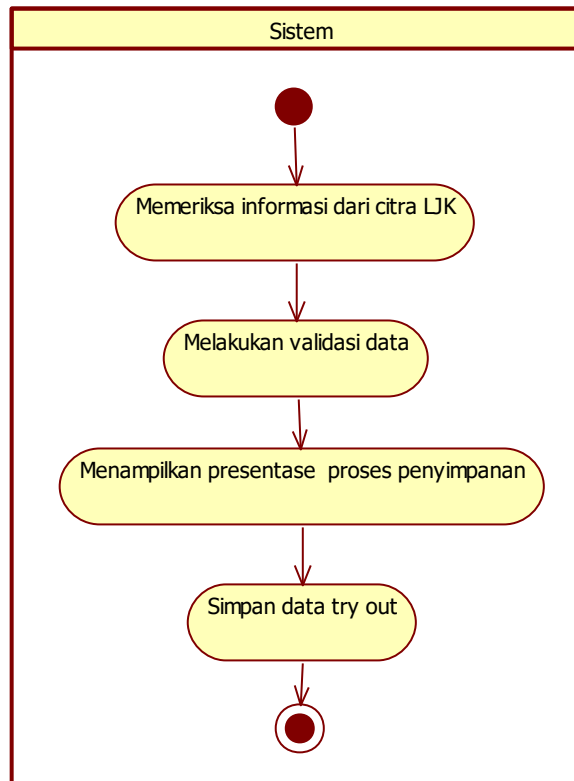
Pada gambar 3.33 merupakan *activity diagram* mendeteksi informasi dari LJK seperti jawaban, kode peserta, kode paket, jurusan. Proses awalnya sistem akan membagi setiap bagian dengan membatasi pixel yang akan diproses. Kemudian sistem akan mengkonversi gambar dari rgb ke grayscale. Selanjutnya sistem akan melakukan threshold pada masing-masing bagian yang sudah dibagi. Kemudian sistem akan mendeteksi objek dengan deteksi tepi canny dan mendeteksi objek lingkaran. Tahap

selanjutnya menentukan koordinat dari masing-masing objek dan mengelompokkannya. Berdasarkan kelompok tersebut maka sistem dapat menentukan hasil dan menampilkannya pada textfield yang tersedia sesuai dengan bagian yang sudah ditentukan.



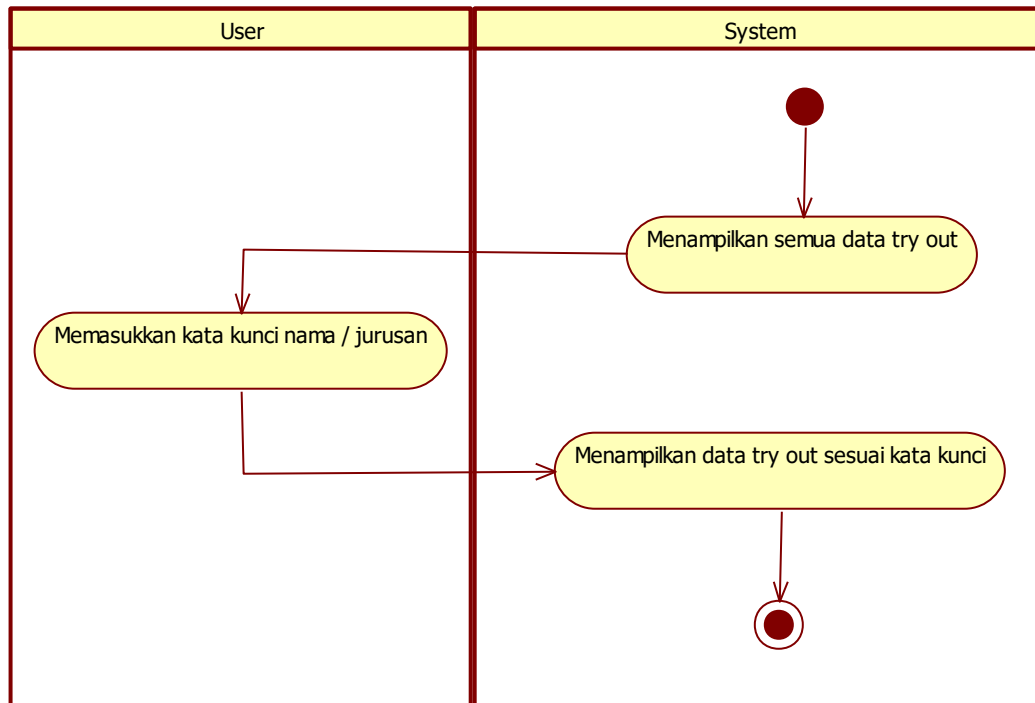
Gambar 3. 34 Activity Diagram Memeriksa Jawaban *Try Out* Siswa

Pada gambar 3.34 merupakan *activity* diagram memeriksa jawaban *try out* siswa berdasarkan informasi yang cocok dari LJK yang dimasukkan. Sebelum memeriksa sistem akan terlebih dahulu melakukan proses mendeteksi informasi dari LJK baik itu kode peserta, kode paket, maupun jawaban. Selanjutnya sistem akan mengambil kunci jawaban yang cocok dengan informasi LJK yang didapat kemudian mencocokkannya. Setelah dicocokkan maka sistem akan mendapatkan hasil jumlah salah, jumlah benar, dan dapat menentukan nilai / hasil *try out* siswa.



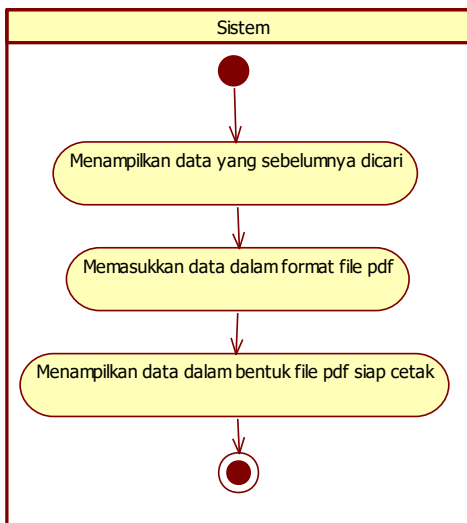
Gambar 3. 35 Activity Diagram Menyimpan Data Try Out Siswa

Pada gambar 3.35 menunjukkan diagram activity menyimpan data *try out* siswa untuk tahap pertama sistem terlebih dahulu melakukan pemeriksaan terhadap data kemudian sistem memvalidasi hasil pemeriksaan dan menampilkan presentase proses penyimpanan. Tahap terakhir sistem menyimpan data yang telah didapat pada database.



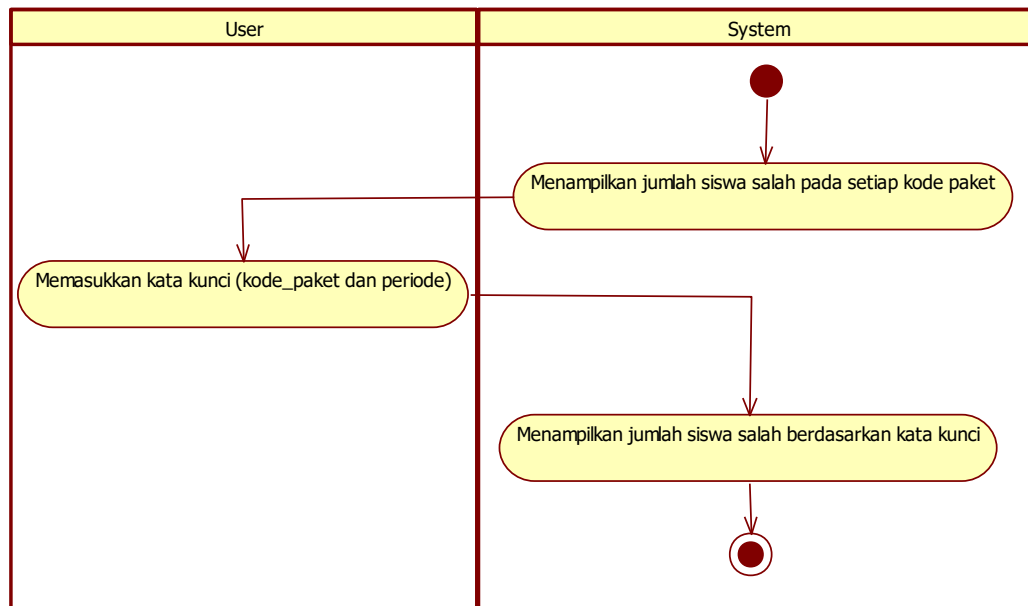
Gambar 3. 36 Activity Diagram Mencari Hasil Try Out Siswa

Pada gambar 3.36 menunjukkan diagram activity mencari hasil *try out* dengan tahap awal sistem akan menampilkan seluruh data *try out* kemudian pengguna/ admin dapat melakukan pencarian dengan memasukkan kata kunci baik itu nama ataupun jurusan. Selanjutnya sistem akan menampilkan data *try out* sesuai dengan kata kunci yang dicari.



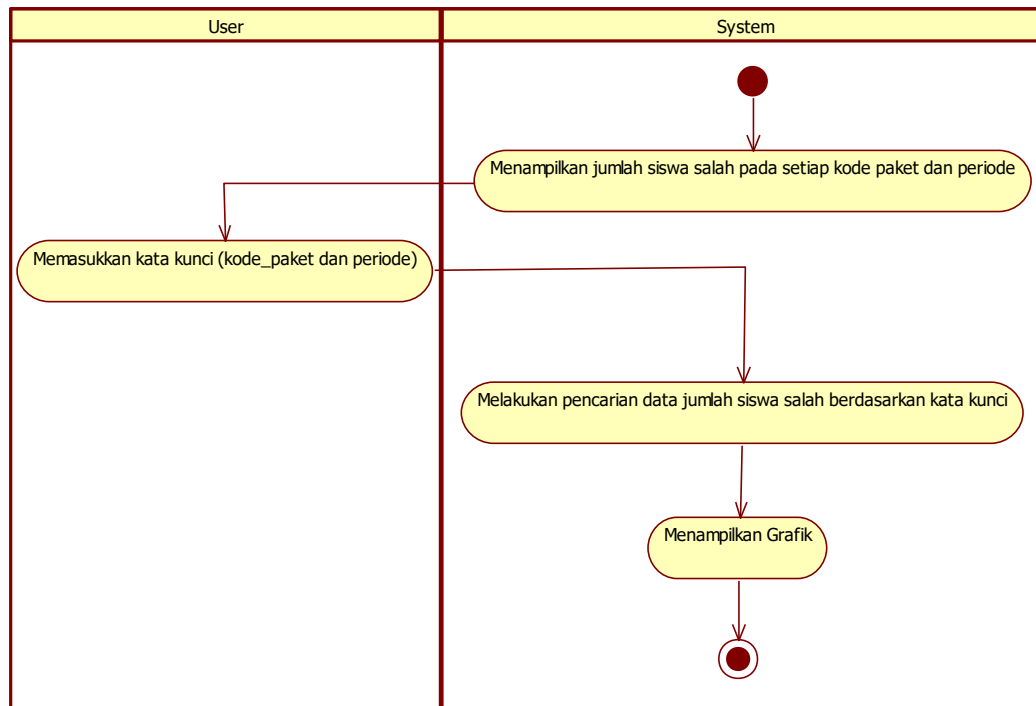
Gambar 3. 37 Activity Diagram Mencetak Data Hasil Try Out

Pada gambar 3.37 menunjukkan *activity* diagram mencetak data hasil try out dimana untuk proses pertama adalah mencari data yang akan dicetak kemudian ditampilkan terlebih dahulu. Proses selanjutnya data yang dipanggil akan dimasukkan dalam file dengan format pdf. Dan proses terakhir sistem menampilkan file pdf tersebut dalam bentuk file siap cetak.



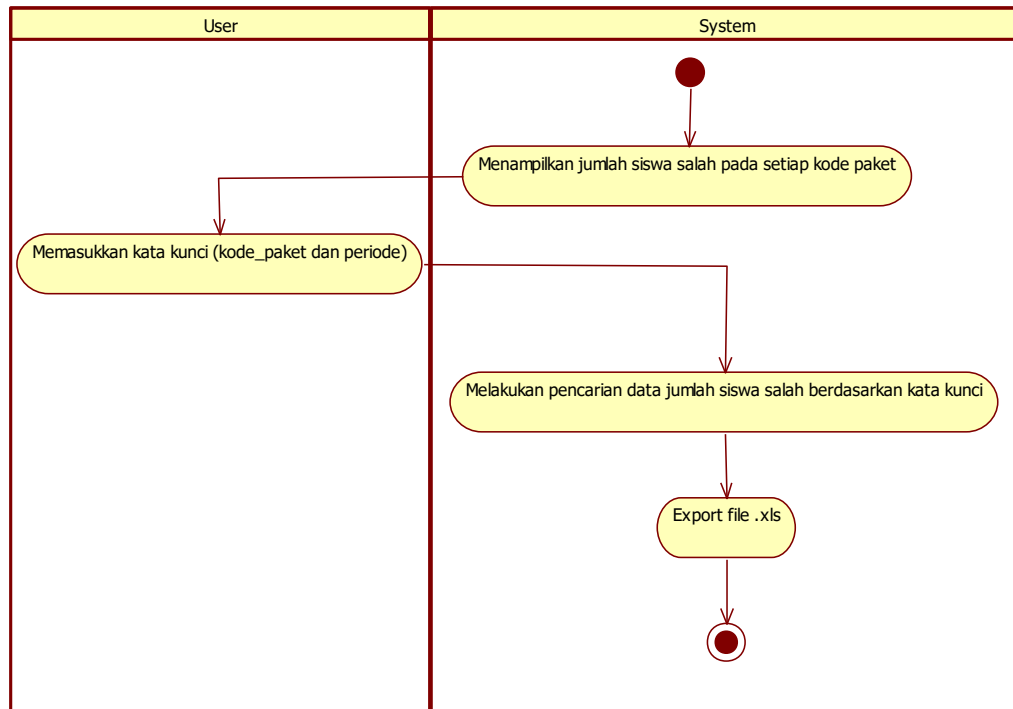
Gambar 3. 38 Activity Diagram Mencari Jumlah Siswa Salah

Pada gambar 3.38 menunjukkan diagram *activity* mencari jumlah siswa salah dimana proses pertama yang dilakukan sistem adalah menampilkan seluruh daftar jumlah siswa salah berdasarkan kode paket dan periode kemudian pengguna / admin dapat memasukkan kata kunci berupa kode paket dan periode proses selanjutnya sistem akan menampilkan data jumlah siswa salah berdasarkan kode paket dan periode yang dicari.



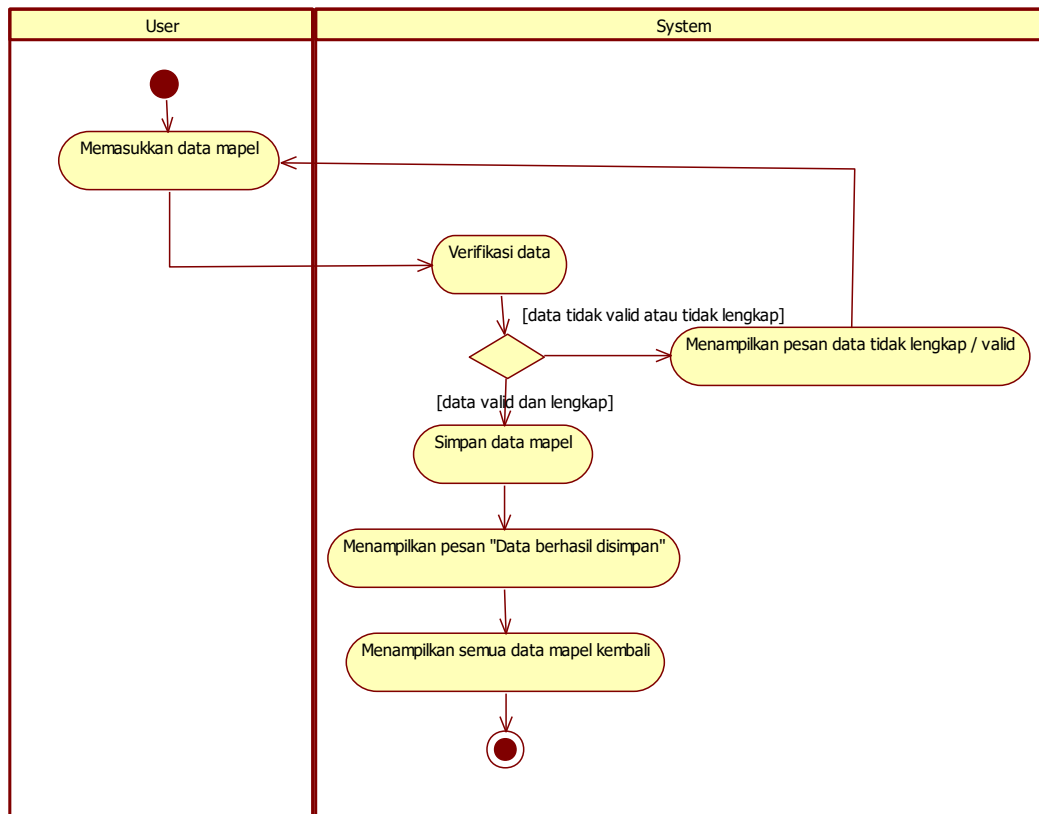
Gambar 3. 39 Activity Diagram Menampilkan Grafik Jumlah Siswa salah

Pada gambar 3.39 menunjukkan diagram *activity* menampilkan grafik jumlah siswa salah pada setiap soal berdasarkan kode paket dan periode. Dimana tahap pertama adalah proses pencarian data jumlah siswa salah berdasarkan kode paket dan periode. Secara default sistem akan menampilkan semua data jumlah siswa salah kemudian admin dapat memasukkan kata kunci berupa kode paket dan periode. Selanjutnya sistem akan melakukan pencarian data. Tahap terakhir ketika data telah ditemukan sistem akan menampilkan grafik sesuai dengan kata kunci yang dimasukkan admin.



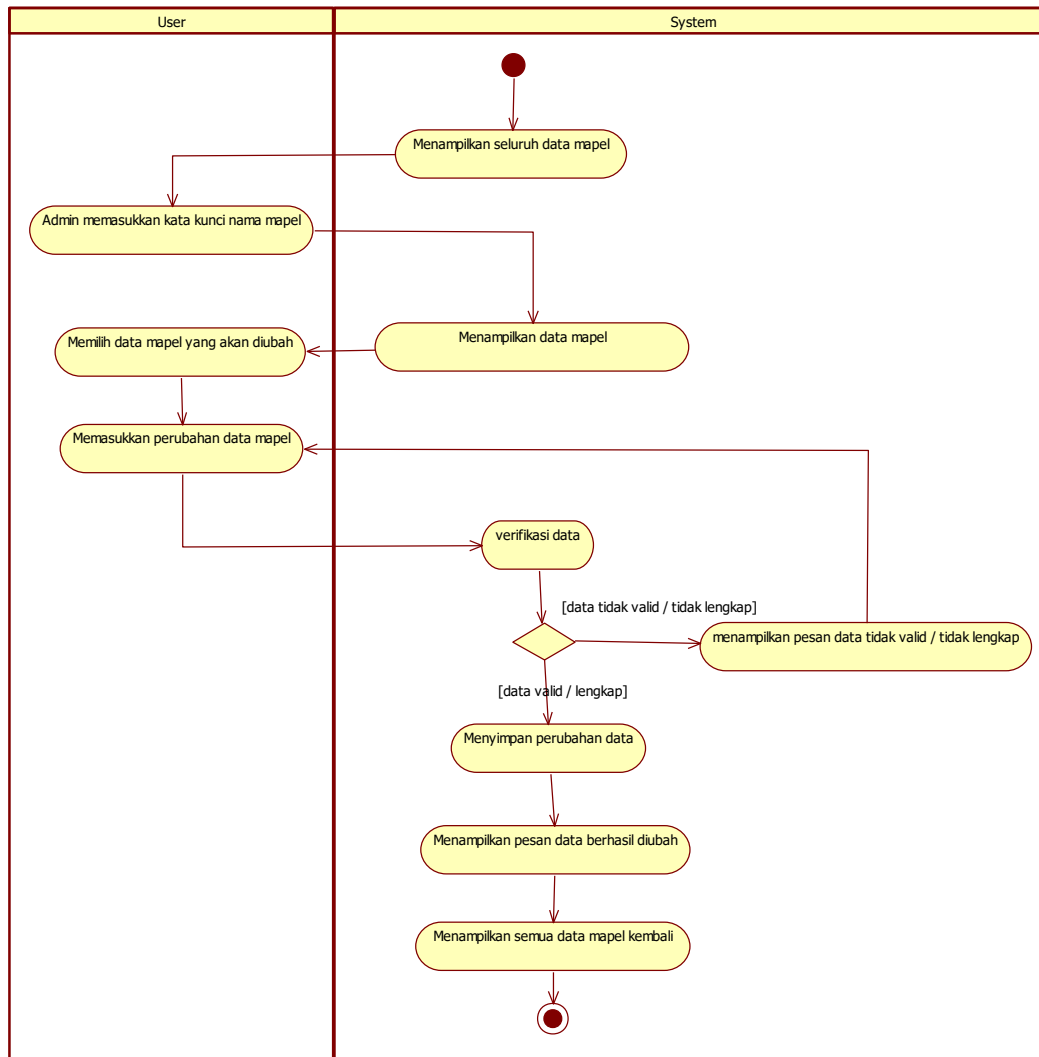
Gambar 3. 40 Activity Diagram Mengeksport Data Jumlah Siswa Salah

Pada gambar 3.40 menunjukkan diagram *activity* mengeksport data jumlah siswa salah pada setiap soal berdasarkan kode paket dan periode. Dimana tahap pertama adalah proses pencarian data jumlah siswa salah berdasarkan kode paket dan periode. Secara default sistem akan menampilkan semua data jumlah siswa salah kemudian admin dapat memasukkan kata kunci berupa kode paket dan periode. Selanjutnya sistem akan melakukan pencarian data. Tahap terakhir ketika data telah ditemukan sistem akan mengeksport data dalam format .xls sesuai dengan kata kunci yang dimasukkan admin.



Gambar 3. 41 Activity Diagram Menambah Data Mapel

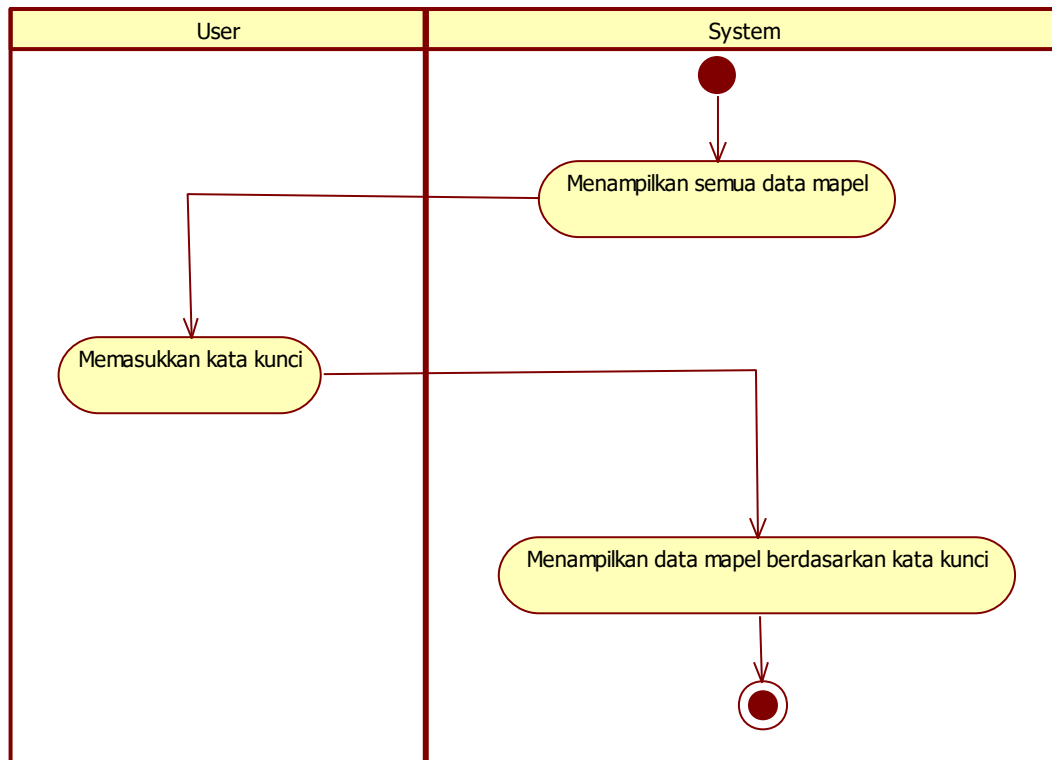
Gambar 3.41 merupakan alur aktifitas menambah data mata pelajaran baru. Pertama pengguna / admin memasukkan data mata pelajaran kemudian sistem akan memverifikasi inputan pengguna sekaligus memeriksa inputan pengguna apakah sudah valid atau belum dan cek keterlengkapan data yang dimasukkan apabila inputan tidak valid / ada salah satu data tidak lengkap maka sistem akan menampilkan pesan peringatan dan kembali ke langkah awal. Jika inputan data sudah valid dan data yang dimasukkan lengkap maka sistem akan langsung menyimpan data mata pelajaran dan menampilkan pesan data berhasil ditambahkan kemudian sistem akan refresh data mata pelajaran dan menampilkan semua data mata pelajaran.



Gambar 3.42 Activity Diagram Mengubah Data Mapel

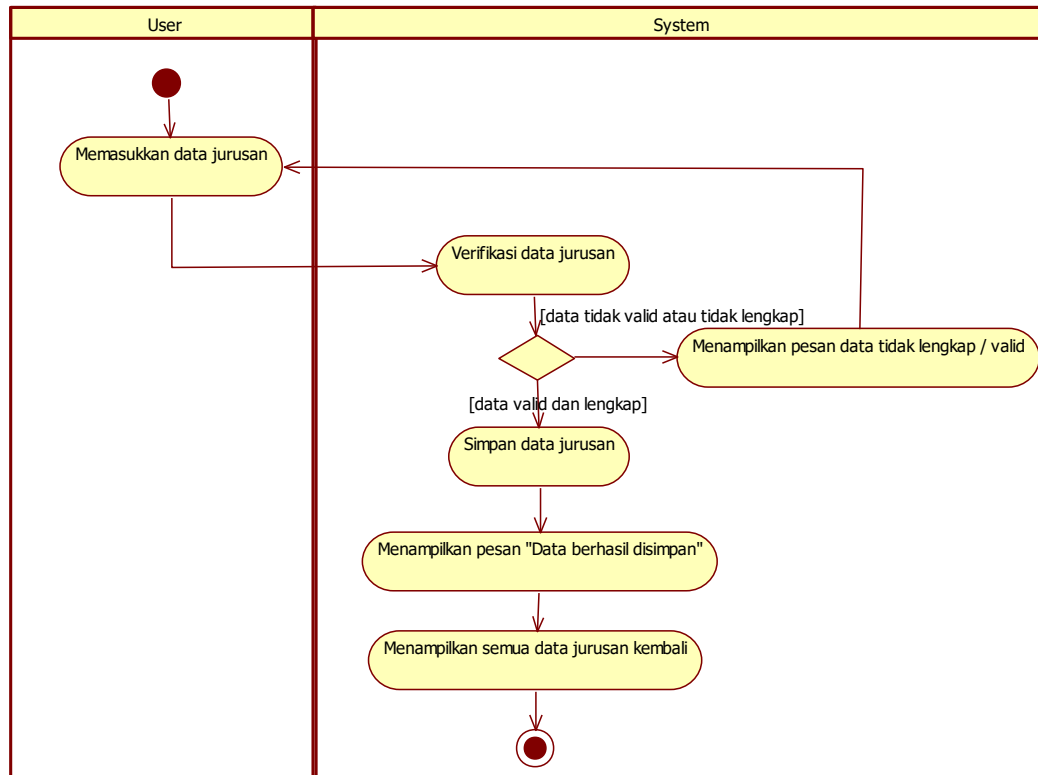
Gambar 3.42 menunjukkan *activity* diagram mengubah data mata pelajaran yang mana proses pertamanya sistem menampilkan data mapel secara keseluruhan kemudian untuk mempercepat pencarian data admin dapat memasukkan kata kunci data memilih data yang akan diubah. Selanjutnya admin memasukkan perubahan data mata pelajaran. Kemudian sistem memverifikasi inputan dan memeriksa keterlengkapan data jika data tidak lengkap / inputan tidak valid maka sistem akan menampilkan pesan data tidak valid / data tidak lengkap. Jika semua perubahan data

sesuai maka sistem akan menyimpan perubahan data dan menampilkan pesan data berhasil diubah sekaligus *refresh* data / menampilkan data mata pelajaran kembali.



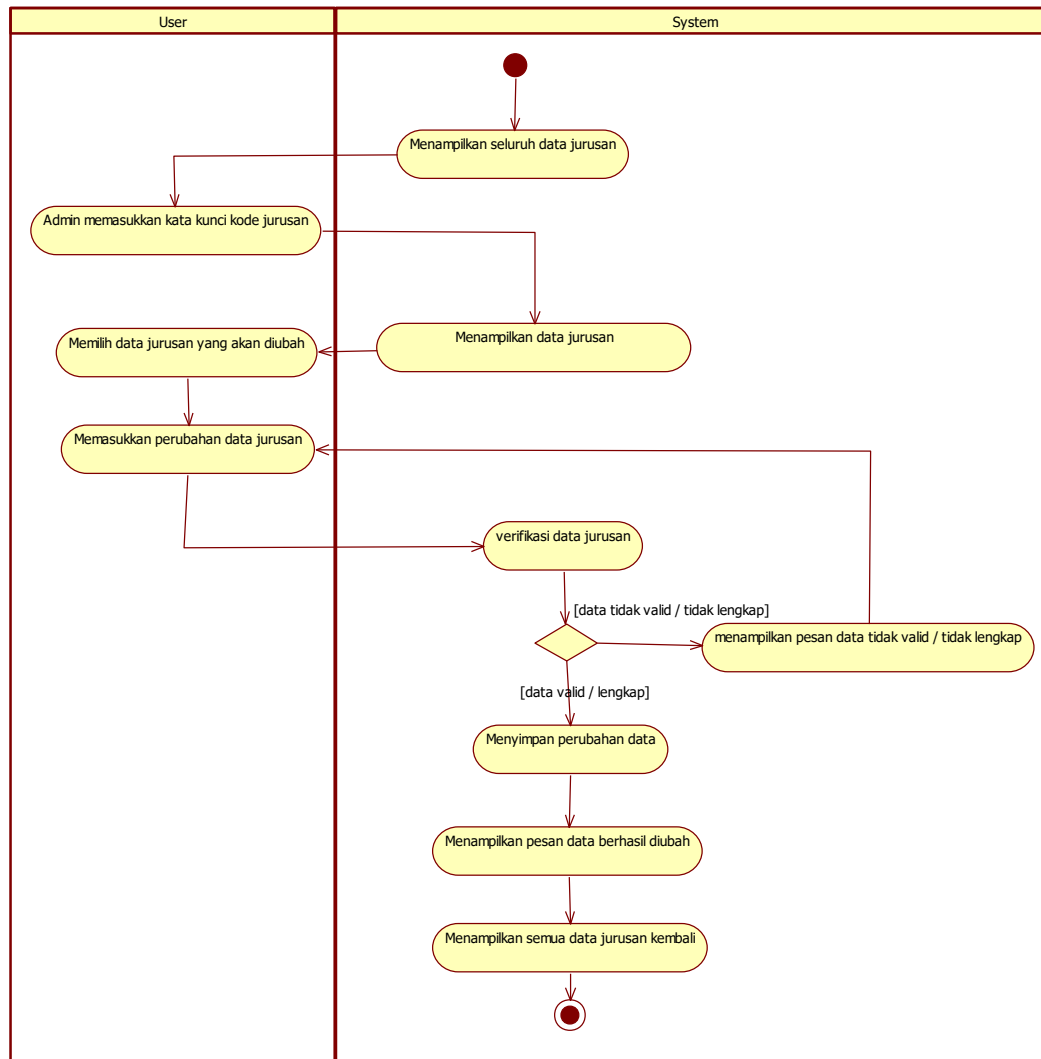
Gambar 3. 43 Activity Diagram Mencari Data Mata Pelajaran

Pada gambar 3.43 menunjukkan *activity* diagram mencari data mata pelajaran. Secara *default* sistem akan menampilkan semua data mata pelajaran kemudian admin dapat memasukkan kata kunci berupa nama mata pelajaran kemudian sistem akan mencari sekaligus menampilkan data mata pelajaran yang sesuai dengan kata kunci yang diinputkan admin / pengguna.



Gambar 3. 44 Activity Diagram Menambah Data Jurusan

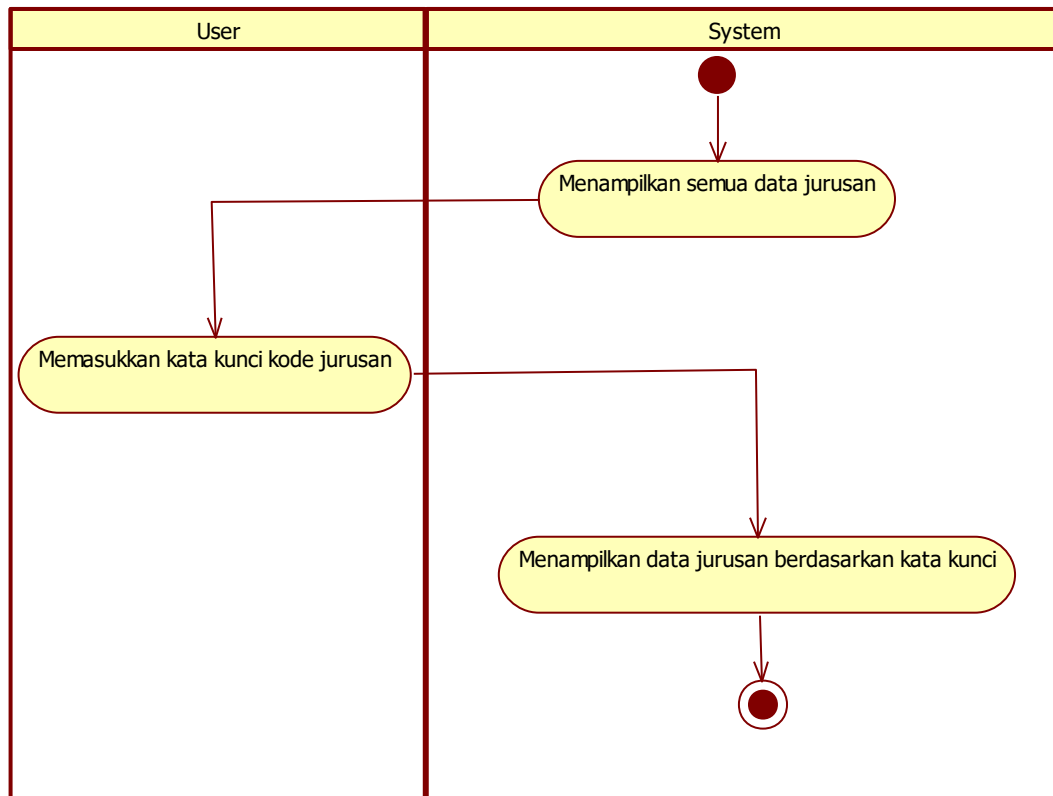
Gambar 3.44 merupakan alur aktifitas menambah data jurusan baru. Pertama pengguna / admin memasukkan data jurusan kemudian sistem akan memverifikasi inputan pengguna sekaligus memeriksa inputan pengguna apakah sudah valid atau belum dan cek keterlengkapan data yang dimasukkan apabila inputan tidak valid / ada salah satu data tidak lengkap maka sistem akan menampilkan pesan peringatan dan kembali ke langkah awal. Jika inputan data sudah valid dan data yang dimasukkan lengkap maka sistem akan langsung menyimpan data jurusan dan menampilkan pesan data berhasil ditambahkan kemudian sistem akan refresh data jurusan dan menampilkan semua data jurusan kembali.



Gambar 3. 45 Activity Diagram Mengubah Data Jurusan

Gambar 3.45 menunjukkan *activity* diagram mengubah data jurusan yang mana proses pertamanya sistem menampilkan data jurusan secara keseluruhan kemudian untuk mempercepat pencarian data, admin dapat memasukkan kata kunci kode jurusan dan memilih data yang akan diubah. Selanjutnya admin memasukkan perubahan data jurusan. Kemudian sistem memverifikasi inputan dan memeriksa keterlengkapan data jika data tidak lengkap / inputan tidak valid maka sistem akan menampilkan pesan data tidak valid / data tidak lengkap. Jika semua perubahan data

sesuai maka sistem akan menyimpan perubahan data dan menampilkan pesan data berhasil diubah sekaligus *refresh* data / menampilkan data jurusan kembali.

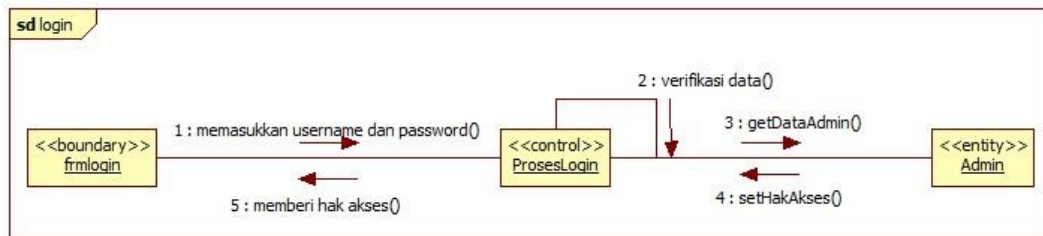


Gambar 3. 46 Activity Diagram Mencari Data Jurusan

Pada gambar 3.46 menunjukkan *activity* diagram mencari data jurusan. Secara *default* sistem akan menampilkan semua data jurusan kemudian admin dapat memasukkan kata kunci berupa kode jurusan kemudian sistem akan mencari sekaligus menampilkan data mata jurusan yang sesuai dengan kata kunci yang diinputkan admin / pengguna.

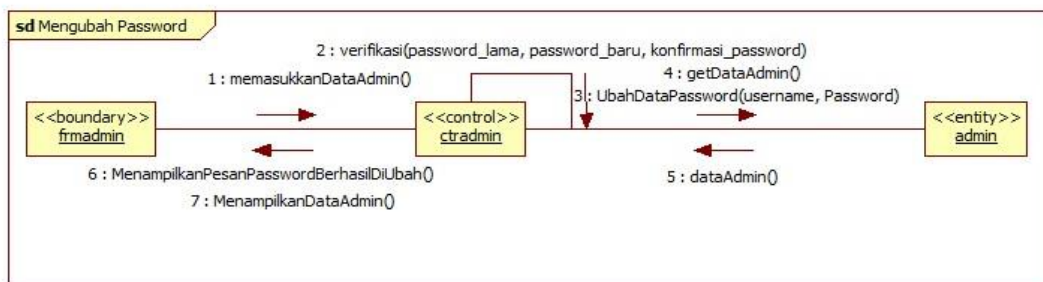
3.7.4 Communication Diagram

Berikut merupakan communication diagram dari aplikasi yang akan dibangun.



Gambar 3. 47 Communication Diagram Login

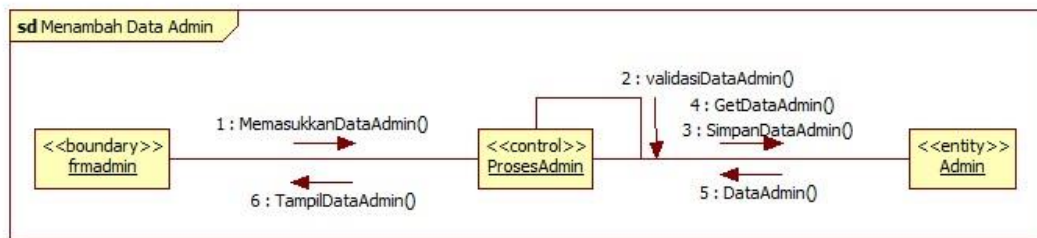
Gambar 3.47 merupakan communication diagram login dimana proses pertama adalah dimulai dari pengguna memasukkan username dan password melalui frmlogin kemudian nilai uasername dan password diproses / diverifikasi. Selanjutnya sistem akan memanggil data dari database untuk dicocokkan dengan username dan password. Jika username dan password terdaftar maka sistem akan memberikan hak akses kepada pengguna dan menutup form login untuk masuk ke frmmenuutama.



Gambar 3. 48 Communication Diagram Mengubah Password

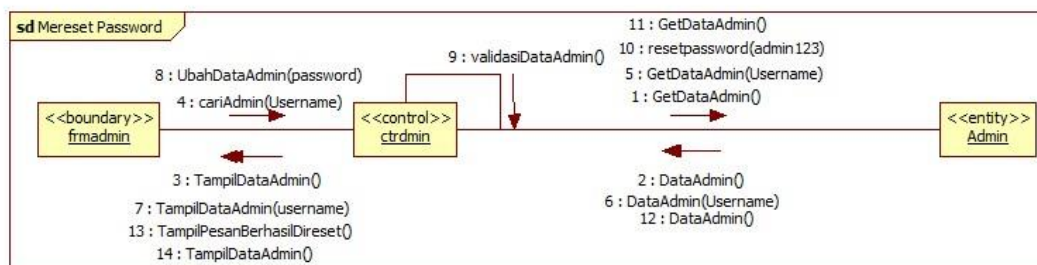
Pada gambar 3.48 menunjukkan communication diagram mengubah password. Proses pertamanya sistem akan menampilkan data session pengguna aplikasi. Kemudian pengguna memasukkan data admin berupa data password lama, password baru, dan konfirmasi password. Selanjutnya sistem akan memverifikasi inputan pengguna jika sudah sesuai maka sistem akan menyimpan perubahan data password ke database.

Kemudian sistem akan merequest data admin kembali dan database akan mengirimkan data yang direquest. Selanjutnya sistem akan menampilkan pesan password berhasil diubah serta menampilkan data kembali pada frmadmin.



Gambar 3. 49 Communication Diagram Menambah Data Admin

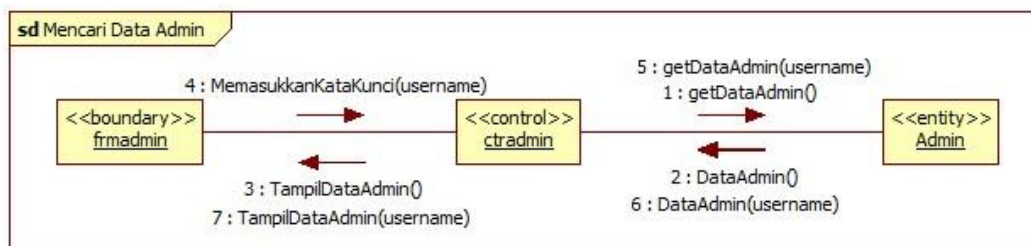
Pada gambar 3.49 menunjukkan communication diagram menambah data admin. Untuk proses pertama user akan memasukkan data yang dibutuhkan pada form yang tersedia di frmadmin kemudian sistem akan memvalidasi data inputan pengguna / user jika data tersebut sudah sesuai maka data tersebut akan disimpan ke database. Kemudian sistem akan merequest data admin secara keseluruhan. Selanjutnya database akan merespon request data dari sistem dengan mengirim data admin yang diminta secara keseluruhan. Proses akhir data admin kemudian ditampilkan kembali secara keseluruhan pada frmadmin.



Gambar 3. 50 Communication Diagram Mereset Password

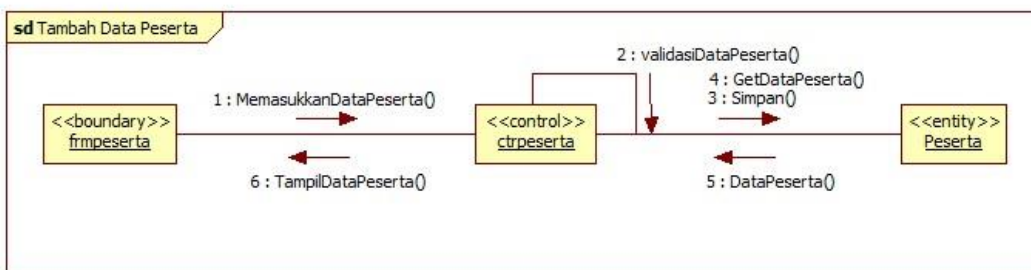
Pada gambar 3.50 menunjukkan communication diagram mereset password. Proses pertamanya sistem akan merequest data pada database kemudian database merespon request data dari sistem yang terhubung dengan mengirimkan data admin secara

keseluruhan. Untuk mempercepat pencarian pengguna dapat mengisi form pencarian. Kemudian pengguna memilih data admin yang akan direset passwordnya dan kemudian sistem menampilkan data yang dipilih pada form yang digunakan untuk mereset password admin menjadi admin123. Setelah data password diubah sistem akan mevalidasi inputan kemudian jika sudah sesuai maka sistem akan menyimpan perubahan data password dan menampilkannya kembali pada frmadmin.



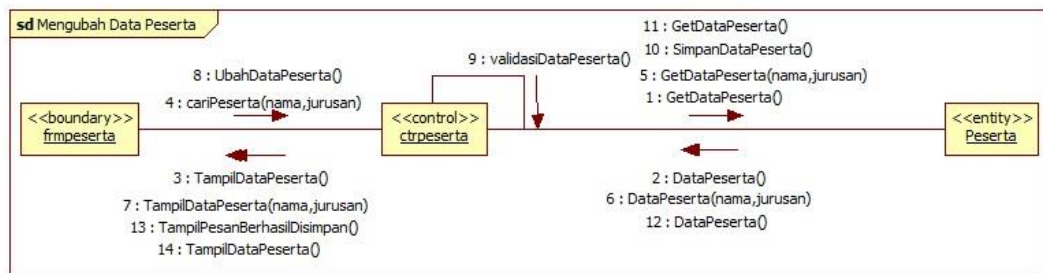
Gambar 3. 51 Communication Diagram Mencari Data Admin

Gambar 3.51 merupakan communication Diagram mencari data admin dimana proses pertamanya sistem akan request data ke database untuk mendapatkan data admin kemudian database akan mengirim data admin yang diminta secara keseluruhan. Kemudian sistem akan menampilkan data tersebut secara keseluruhan. Setelah itu user dapat mengisi form pencarian data sesuai kata kunci. Kemudian sistem akan merequest kembali data admin sesuai dengan kata kunci yang dimasukkan. Database akan merespon dengan mengirim kembali data admin yang sesuai dengan kata kunci yang dimasukkan.



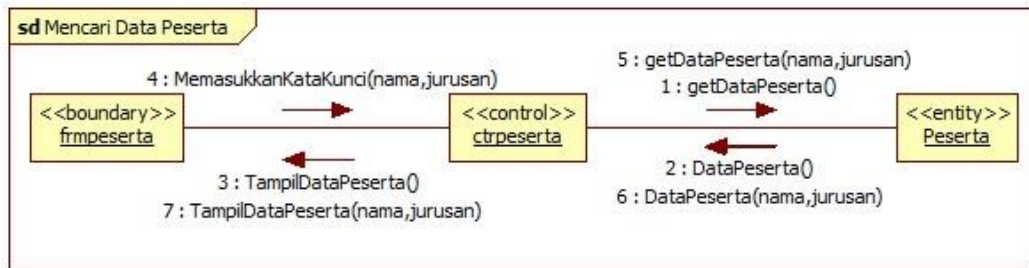
Gambar 3. 52 Communication Diagram Menambah Data Peserta

Pada gambar 3.52 menunjukkan communication diagram menambah data peserta. Untuk proses pertama user akan memasukkan data yang dibutuhkan pada form yang tersedia di frmpeserta kemudian sistem akan memvalidasi data inputan pengguna / user jika data tersebut sudah sesuai maka data tersebut akan disimpan ke database. Kemudian sistem akan merequest data peserta secara keseluruhan. Selanjutnya database akan merespon request data dari sistem dengan mengirim data peserta yang diminta secara keseluruhan. Proses akhir data peserta kemudian ditampilkan kembali secara keseluruhan pada frmpeserta.



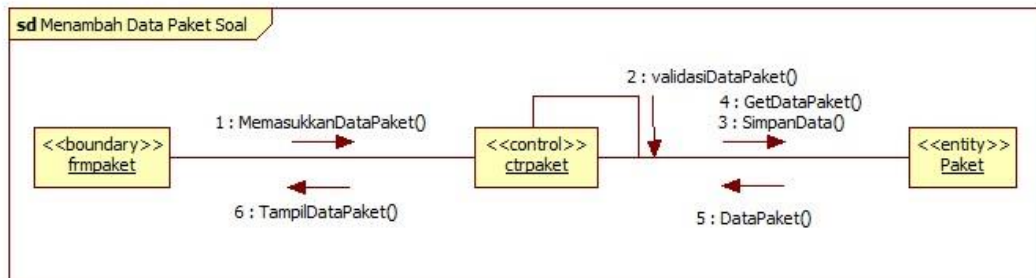
Gambar 3. 53 Communication Diagram Mengubah Data Peserta

Pada gambar 3.53 menunjukkan communication diagram mengubah data peserta. Proses pertamanya sistem akan merequest data pada database kemudian database merespon request data dari sistem yang terhubung dengan mengirimkan data peserta secara keseluruhan. Untuk mempercepat pencarian pengguna dapat mengisi form pencarian. Kemudian pengguna memilih data yang akan diubah dan kemudian sistem menampilkan data yang dipilih pada form yang digunakan untuk merubah data peserta. Setelah data diubah sistem akan mevalidasi inputan kemudian jika sudah sesuai maka sistem akan menyimpan perubahan data dan menampilkannya kembali pada frmpeserta.



Gambar 3. 54 Communication Diagram Mencari Data Peserta

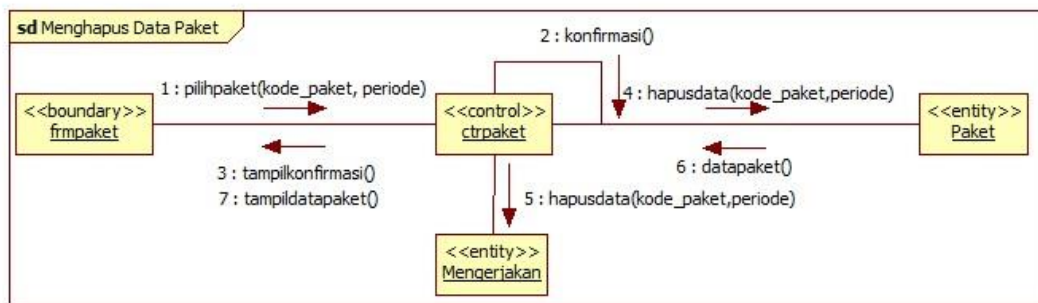
Gambar 3.54 merupakan communication Diagram mencari data peserta dimana proses pertamanya sistem akan request data ke database untuk mendapatkan data peserta kemudian database akan mengirim data peserta yang diminta secara keseluruhan. Kemudian sistem akan menampilkan data tersebut secara keseluruhan. Setelah itu user dapat mengisi form pencarian data sesuai kata kunci. Kemudian sistem akan merequest kembali data peserta sesuai dengan kata kunci yang dimasukkan. Database akan merespon dengan mengirim kembali data peserta yang sesuai dengan kata kunci yang dimasukkan.



Gambar 3. 55 Communication Diagram Menambah Data Paket Soal

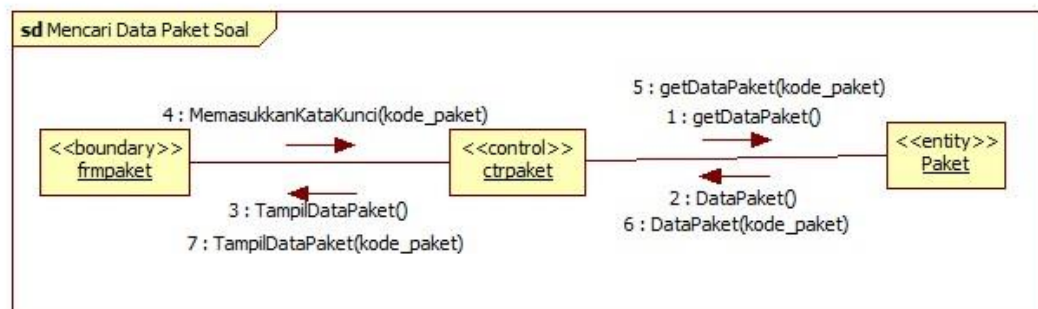
Pada gambar 3.55 menunjukkan communication diagram menambah data paket soal. Untuk proses pertama user akan memasukkan data yang dibutuhkan pada form yang tersedia di frmpaket kemudian sistem akan memvalidasi data inputan pengguna / user jika data tersebut sudah sesuai maka data tersebut akan disimpan ke database. Kemudian sistem akan merequest data paket soal secara keseluruhan.

Selanjutnya database akan merespon request data dari sistem dengan mengirim data paket soal yang diminta secara keseluruhan. Proses akhir data paket soal kemudian ditampilkan kembali secara keseluruhan pada frmpaket.



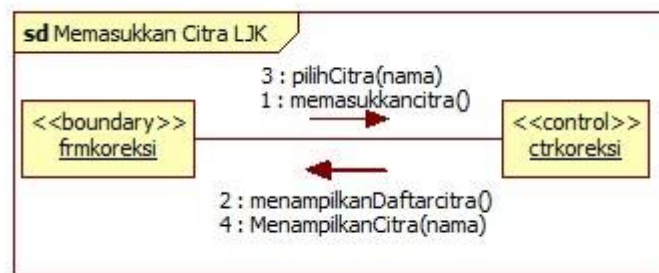
Gambar 3. 56 Communication Diagram Menghapus Data Paket Soal

Gambar 3.56 menunjukkan communication diagram menghapus dimana proses pertama pengguna dapat memilih kode paket soal try out yang dihapus kemudian sistem akan menampilkan pesan konfirmasi untuk meyakinkan pengguna apakah data tersebut akan benar-benar dihapus atau tidak. Apabila yakin akan menghapus maka sistem akan menghapus data yang berhubungan dengan kode paket dan periode yang telah dipilih oleh pengguna baik itu di entity paket maupun di entity mengerjakan. Proses selanjutnya sistem merequest data ke database. Dan database akan mengirimkan data yang diminta. Kemudian proses terakhir sistem akan merrefresh data paket dan menampilkan data paket kembali.



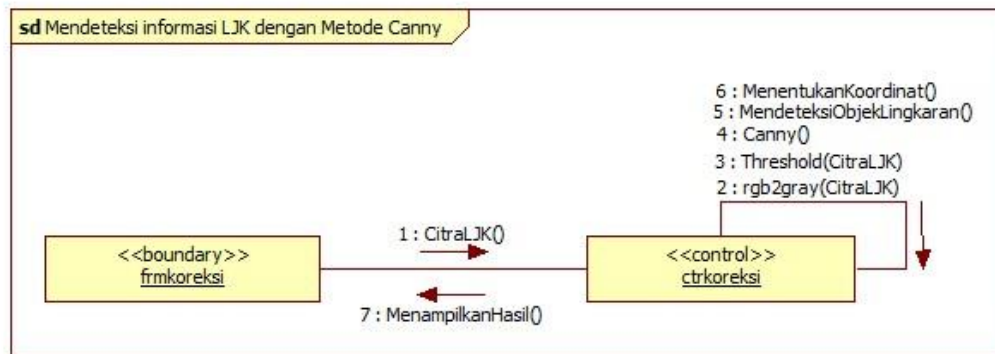
Gambar 3. 57 Communication Diagram Mencari Data Paket Soal

Gambar 3.57 merupakan communication diagram mencari data paket soal dimana proses pertamanya sistem akan request data ke database untuk mendapatkan data paket soal kemudian database akan mengirim data paket soal yang diminta secara keseluruhan. Kemudian sistem akan menampilkan data tersebut secara keseluruhan. Setelah itu user dapat mengisi form pencarian data sesuai kata kunci. Kemudian sistem akan merequest kembali data paket soal sesuai dengan kata kunci yang dimasukkan. Database akan merespon dengan mengirim kembali data paket soal yang sesuai dengan kata kunci yang dimasukkan.



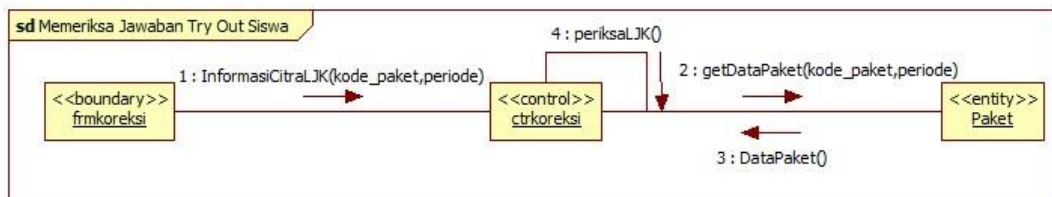
Gambar 3. 58 Communication Diagram Memasukkan Citra LJK

Pada gambar 3.58 menunjukkan Communication diagram memasukkan citra LJK yang proses utaman user memasukkan beberapa citra LJK kedalam sistem kemudian sistem akan menampilkan daftar list nama citra LJK yang telah dimasukkan oleh pengguna. Selanjutnya pengguna memilih citra yang akan diolah terlebih dahulu. Kemudian sistem menampilkan citra yang akan diolah pada gui koreksi.



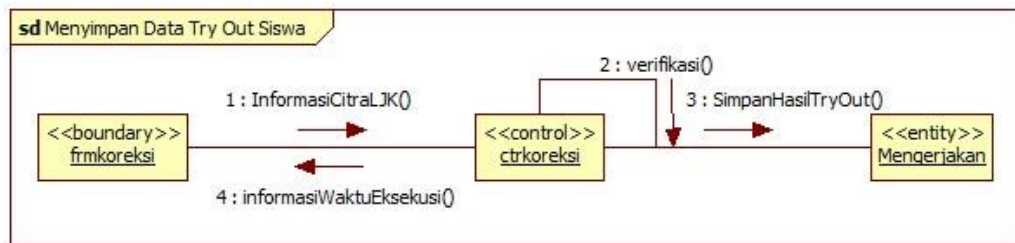
Gambar 3. 59 Communication Diagram Mendeteksi informasi LJK

Gambar 3.59 merupakan communication diagram mendeteksi informasi dari LJK dimana tahapan pertama user telah memilih image yang akan diproses kemudian sistem akan memproses image tersebut dengan beberapa fungsi seperti konversi citra, threshold, canny, deteksi objek lingkaran dan menentukan koordinat objek lingkaran. Tahap terakhir sistem menampilkan informasi hasil deteksi.



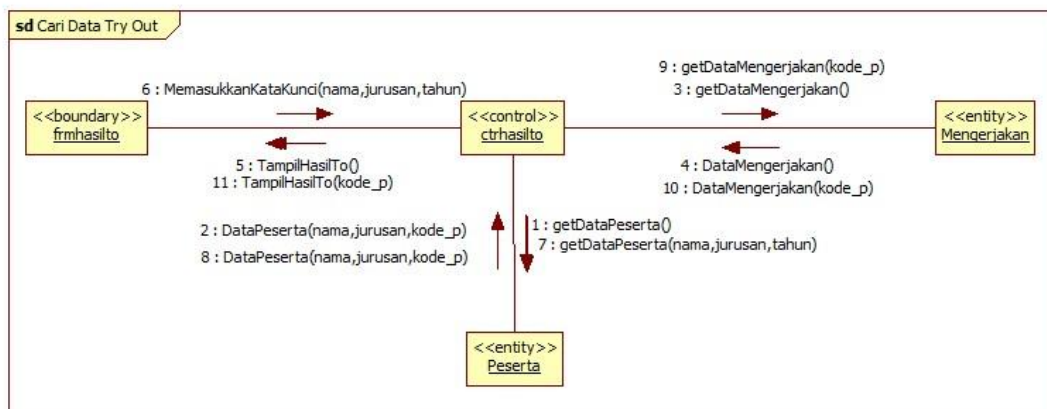
Gambar 3. 60 Communication Diagram Memeriksa Jawaban Try Out Siswa

Gambar 3.60 menunjukkan communication diagram memeriksa jawaban try out. Proses ini dilakukan setelah proses deteksi informasi LJK. Informasi yang didapat berupa jawaban, kode peserta, jurusan, kode mata pelajaran, kode soal, periode, tanggal ujian. Kemudian informasi tersebut akan dijadikan acuan dalam mencari kunci jawaban yang sesuai pada entity paket. Selanjutnya sistem akan mencocokkan antara kunci jawaban dengan jawaban dari LJK sehingga menghasilkan jumlah salah, jumlah benar dan nilai untuk disimpan pada tabel mengerjakan.



Gambar 3. 61 Communication Diagram Menyimpan Data Try Out Siswa

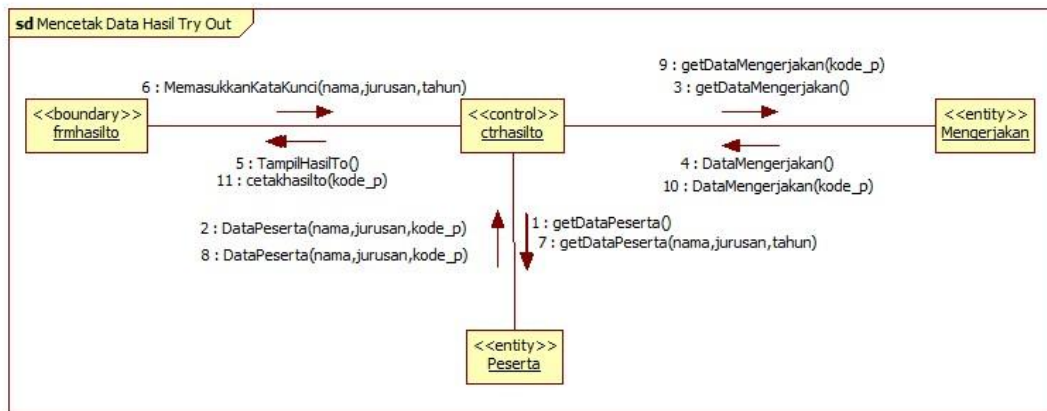
Gambar 3.61 menunjukkan communication diagram menyimpan data try out siswa. Proses penyimpanan dilakukan setelah proses deteksi informasi dari citra LJK dan memeriksa dengan kunci jawaban. Kemudian semua informasi tersebut diverifikasi jika sudah sesuai maka informasi tersebut akan disimpan ke database. Selanjutnya sistem menampilkan lama waktu eksekusi program mulai dari deteksi hingga menyimpan data ke database.



Gambar 3. 62 Communication Diagram Cari Data Try Out

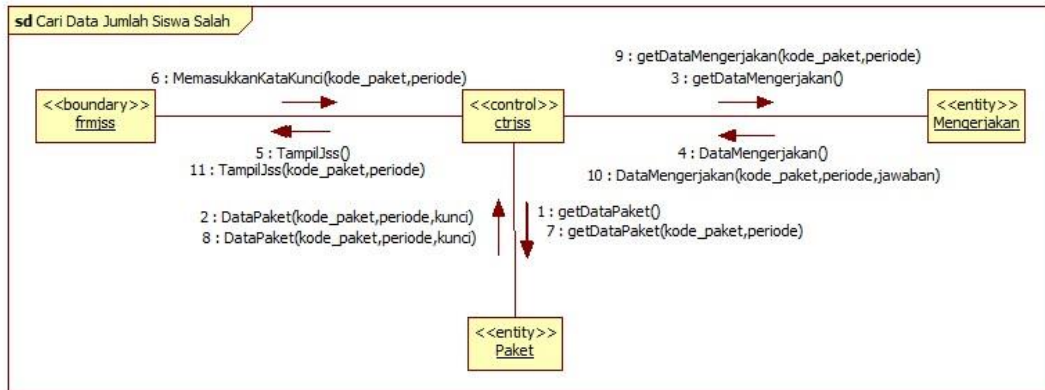
Gambar 3.62 merupakan communication diagram mencari data try out dimana proses pertamanya sistem akan request data ke database untuk mendapatkan data peserta dan data hasil try out kemudian database akan mengirim data peserta dan data hasil try out yang diminta secara keseluruhan. Kemudian sistem akan menampilkan data tersebut secara keseluruhan. Setelah itu user dapat mengisi form pencarian data sesuai kata kunci. Kemudian sistem akan merequest kembali data

peserta dan data hasil try out sesuai dengan kata kunci yang dimasukkan. Database akan merespon dengan mengirim kembali data peserta dan data hasil mengerjakan yang sesuai dengan kata kunci yang dimasukkan.



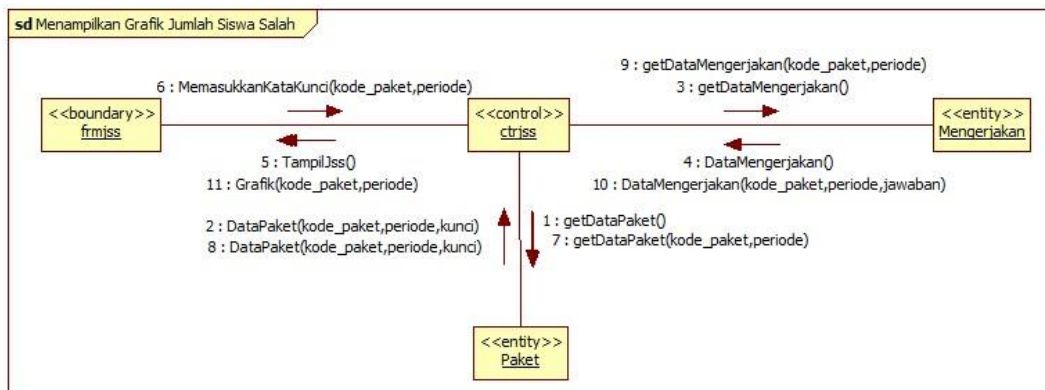
Gambar 3. 63 Communication Diagram Mencetak Data Hasil Try Out

Gambar 3.63 merupakan communication diagram mencari data try out dimana proses pertamanya sistem akan request data ke database untuk mendapatkan data peserta dan data hasil try out kemudian database akan mengirim data peserta dan data hasil try out yang diminta secara keseluruhan. Kemudian sistem akan menampilkan data tersebut secara keseluruhan. Setelah itu user dapat mengisi form pencarian data sesuai kata kunci. Kemudian sistem akan merequest kembali data peserta dan data hasil try out sesuai dengan kata kunci yang dimasukkan. Database akan merespon dengan mengirim kembali data peserta dan data hasil mengerjakan yang sesuai dengan kata kunci yang dimasukkan kemudian data tersebut ditampilkan dalam bentuk report pdf yang siap cetak.



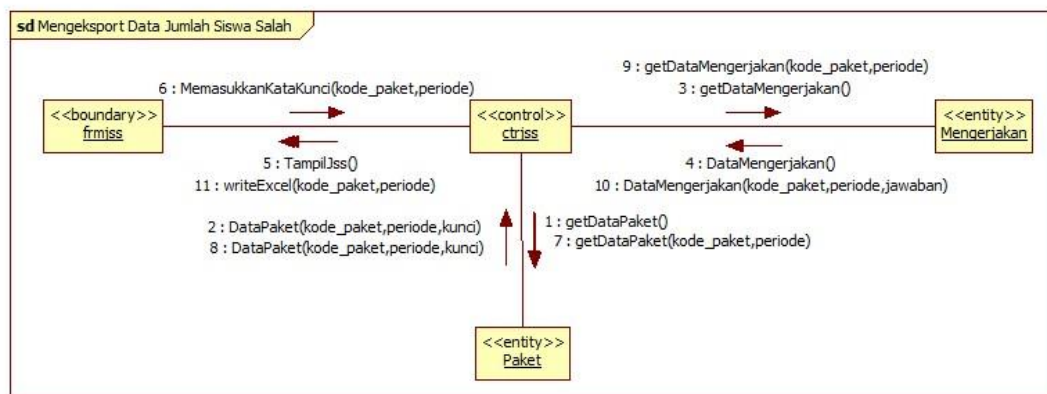
Gambar 3. 64 Communication Diagram Cari Data Jumlah Siswa Salah

Gambar 3.64 merupakan communication diagram cari data jumlah siswa salah dimana proses pertamanya sistem akan request data ke database untuk mendapatkan data paket dan data hasil try out kemudian database akan mengirim data paket dan data hasil try out yang diminta secara keseluruhan. Kemudian sistem akan menampilkan data tersebut secara keseluruhan. Setelah itu user dapat mengisi form pencarian data sesuai kata kunci. Kemudian sistem akan merequest kembali data paket dan data hasil try out sesuai dengan kata kunci yang dimasukkan. Database akan merespon dengan mengirim kembali data paket dan data hasil mengerjakan yang sesuai dengan kata kunci yang dimasukkan.



Gambar 3. 65 Communication Diagram Menampilkan Grafik JSS

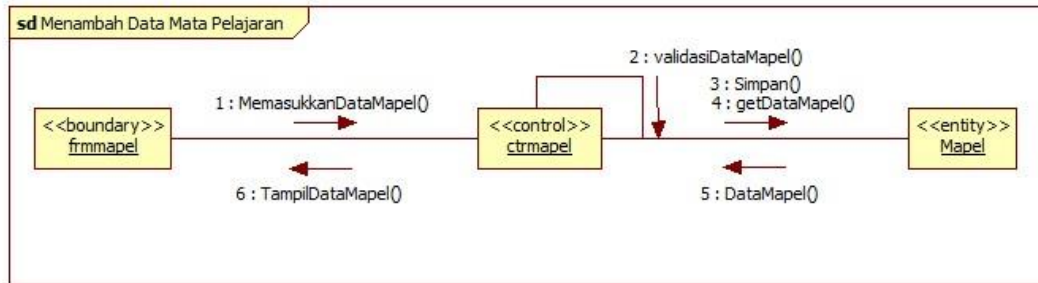
Gambar 3.65 merupakan communication diagram menampilkan grafik jumlah siswa salah dimana proses pertamanya sistem akan request data ke database untuk mendapatkan data paket dan data hasil try out kemudian database akan mengirim data paket dan data hasil try out yang diminta secara keseluruhan. Kemudian sistem akan menampilkan data tersebut secara keseluruhan. Setelah itu user dapat mengisi form pencarian data sesuai kata kunci. Kemudian sistem akan merequest kembali data paket dan data hasil try out sesuai dengan kata kunci yang dimasukkan. Database akan merespon dengan mengirim kembali data paket dan data hasil mengerjakan yang sesuai dengan kata kunci yang dimasukkan kemudian data tersebut ditampilkan dalam bentuk grafik.



Gambar 3. 66 Communication Diagram Mengeksport Data Jumlah Siswa Salah

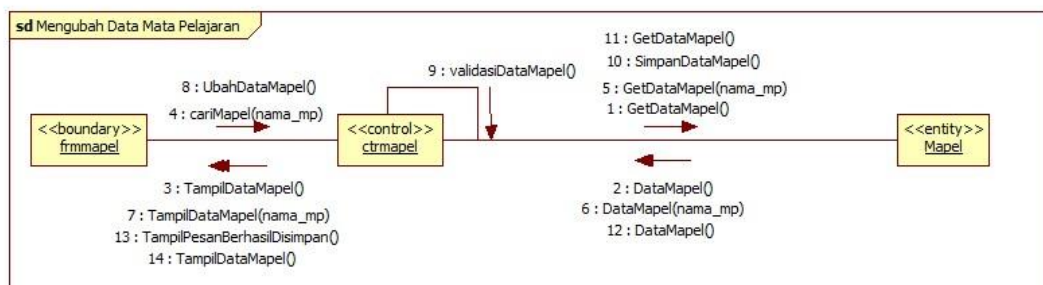
Gambar 3.66 merupakan communication diagram menampilkan grafik jumlah siswa salah dimana proses pertamanya sistem akan request data ke database untuk mendapatkan data paket dan data hasil try out kemudian database akan mengirim data paket dan data hasil try out yang diminta secara keseluruhan. Kemudian sistem akan menampilkan data tersebut secara keseluruhan. Setelah itu user dapat mengisi form pencarian data sesuai kata kunci. Kemudian sistem akan merequest kembali data paket dan data hasil try out sesuai dengan kata kunci yang dimasukkan. Database akan merespon dengan mengirim kembali data paket dan data hasil

mengerjakan yang sesuai dengan kata kunci yang dimasukkan kemudian data tersebut diekspor dalam bentuk file excel / .xls.



Gambar 3. 67 Communication Diagram Menambah Data Mata Pelajaran

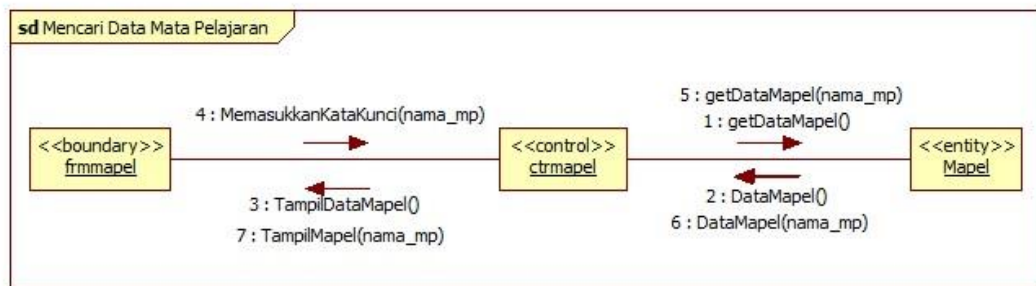
Pada gambar 3.67 menunjukkan communication diagram menambah data mata pelajaran. Untuk proses pertama user akan memasukkan data yang dibutuhkan pada form yang tersedia di frmmapel kemudian sistem akan memvalidasi data inputan pengguna / user jika data tersebut sudah sesuai maka data tersebut akan disimpan ke database. Kemudian sistem akan merequest data mata pelajaran secara keseluruhan. Selanjutnya database akan merespon request data dari sistem dengan mengirim data mata pelajaran yang diminta secara keseluruhan. Proses akhir data mata pelajaran kemudian ditampilkan kembali secara keseluruhan pada frmmapel.



Gambar 3. 68 Communication Diagram Mengubah Data Mata Pelajaran

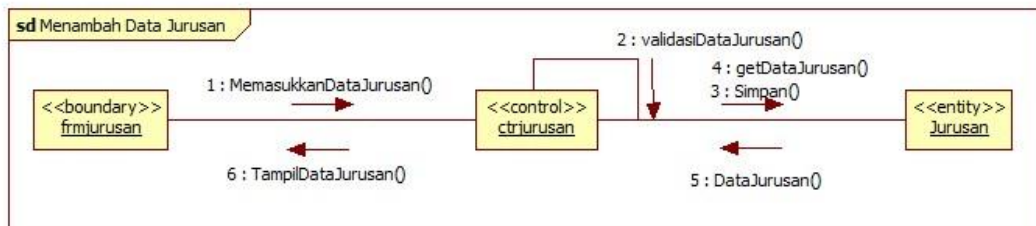
Pada gambar 3.68 menunjukkan communication diagram mengubah data mata pelajaran. Proses pertamanya sistem akan merequest data pada database kemudian database merespon request data dari sistem yang terhubung dengan mengirimkan data

mata pelajaran secara keseluruhan. Untuk mempercepat pencarian pengguna dapat mengisi form pencarian. Kemudian pengguna memilih data yang akan diubah dan kemudian sistem menampilkan data yang dipilih pada form yang digunakan untuk merubah data mata pelajaran. Setelah data diubah sistem akan mevalidasi inputan kemudian jika sudah sesuai maka sistem akan menyimpan perubahan data dan menampilkannya kembali pada frmmapel.



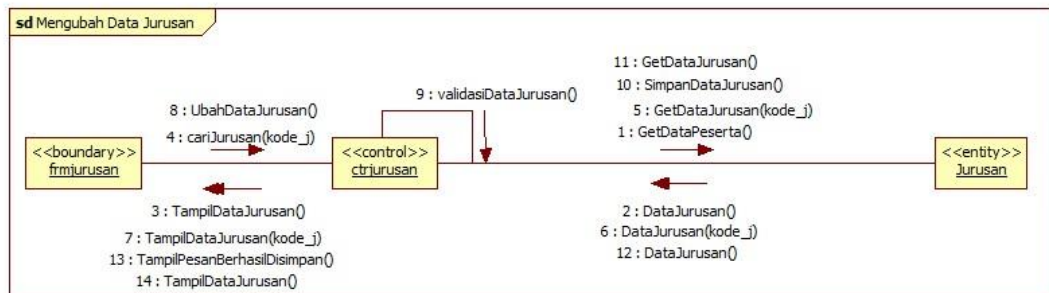
Gambar 3. 69 Communication Diagram Mencari Data Mata Pelajaran

Gambar 3.69 merupakan communication diagram mencari data mata pelajaran dimana proses pertamanya sistem akan request data ke database untuk mendapatkan data mata pelajaran kemudian database akan mengirim data mata pelajaran yang diminta secara keseluruhan. Kemudian sistem akan menampilkan data tersebut secara keseluruhan. Setelah itu user dapat mengisi form pencarian data sesuai kata kunci. Kemudian sistem akan merequest kembali data mata pelajaran sesuai dengan kata kunci yang dimasukkan. Database akan merespon dengan mengirim kembali data mata pelajaran yang sesuai dengan kata kunci yang dimasukkan.



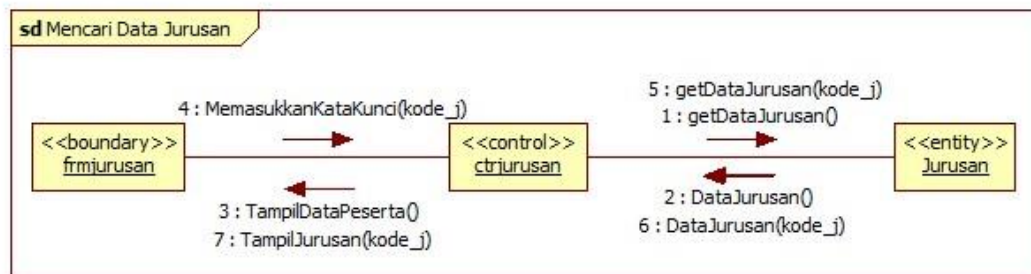
Gambar 3. 70 Communication Diagram Menambah Data Jurusan

Pada gambar 3.70 menunjukkan communication diagram menambah data jurusan. Untuk proses pertama user akan memasukkan data yang dibutuhkan pada form yang tersedia di frmjurusan kemudian sistem akan memvalidasi data inputan pengguna / user jika data tersebut sudah sesuai maka data tersebut akan disimpan ke database. Kemudian sistem akan merequest data jurusan secara keseluruhan. Selanjutnya database akan merespon request data dari sistem dengan mengirim data jurusan yang diminta secara keseluruhan. Proses akhir data jurusan kemudian ditampilkan kembali secara keseluruhan pada frmjurusan.



Gambar 3. 71 Communication Diagram Mengubah Data Jurusan

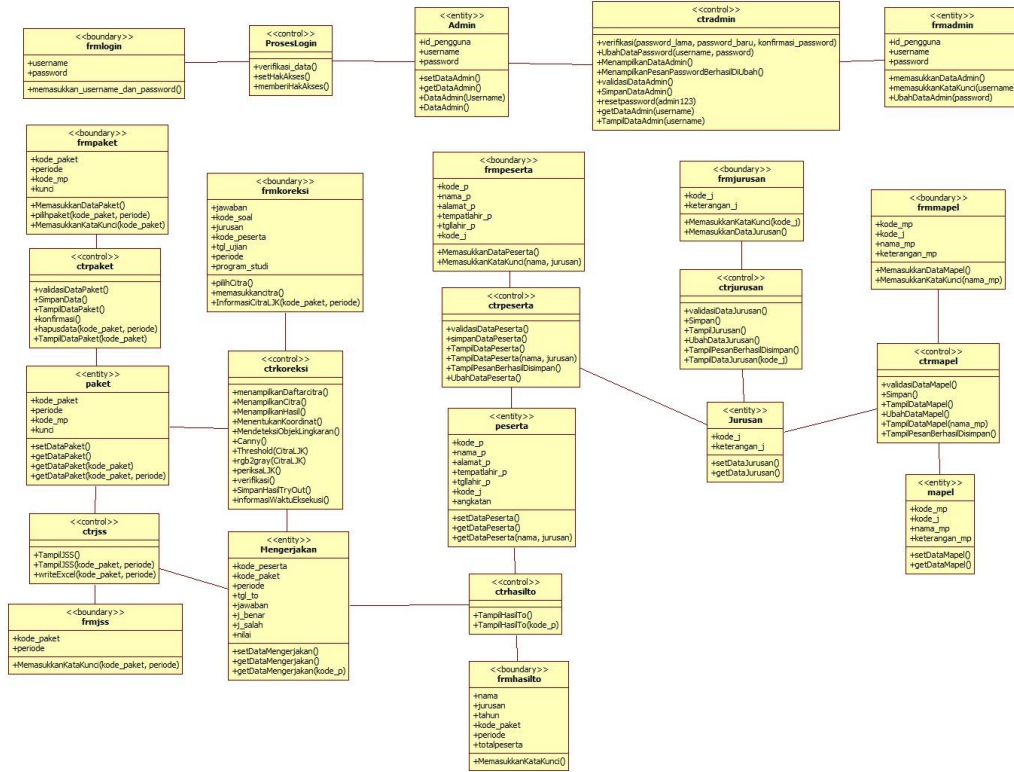
Pada gambar 3.71 menunjukkan communication diagram mengubah data jurusan. Proses pertamanya sistem akan merequest data pada database kemudian database merespon request data dari sistem yang terhubung dengan mengirimkan data jurusan secara keseluruhan. Untuk mempercepat pencarian pengguna dapat mengisi form pencarian. Kemudian pengguna memilih data yang akan diubah dan kemudian sistem menampilkan data yang dipilih pada form yang digunakan untuk merubah data jurusan. Setelah data diubah sistem akan mevalidasi inputan kemudian jika sudah sesuai maka sistem akan menyimpan perubahan data dan menampilkannya kembali pada frmjurusan.



Gambar 3. 72 Communiacion Diagram Mencari Data Jurusan

Gambar 3.72 merupakan communication diagram mencari data jurusan dimana proses pertamanya sistem akan request data ke database untuk mendapatkan data jurusan kemudian database akan mengirim data jurusan yang diminta secara keseluruhan. Kemudian sistem akan menampilkan data tersebut secara keseluruhan. Setelah itu user dapat mengisi form pencarian data sesuai kata kunci. Kemudian sistem akan merequest kembali data jurusan sesuai dengan kata kunci yang dimasukkan. Database akan merespon dengan mengirim kembali data jurusan yang sesuai dengan kata kunci yang dimasukkan.

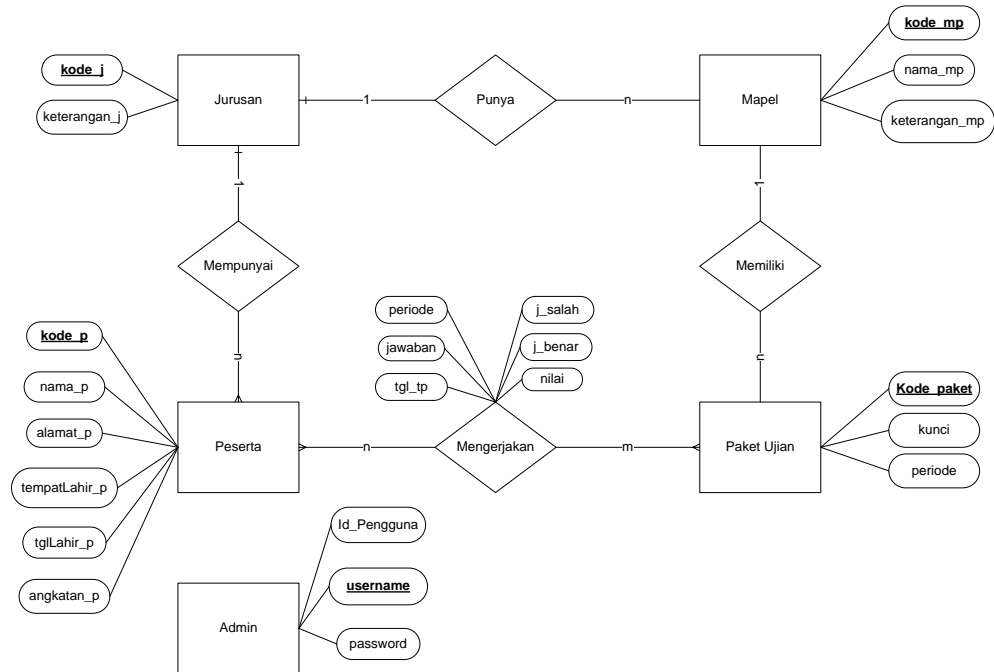
3.7.5 Class Diagram



Gambar 3. 73 Class Diagram

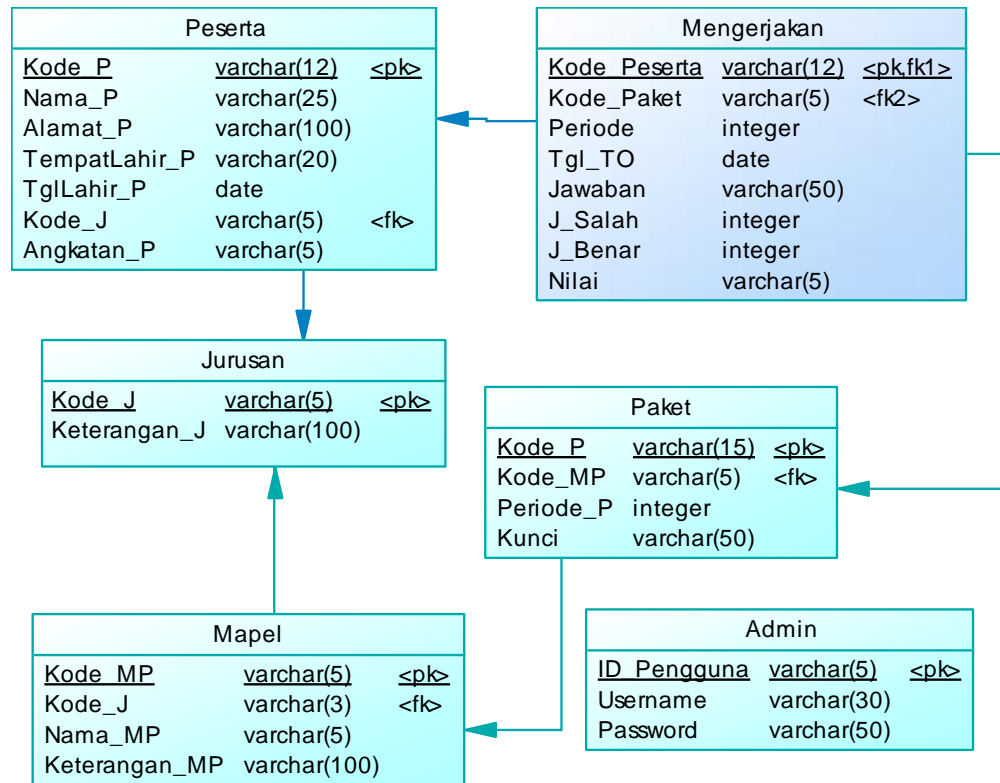
Pada gambar 3.73 menggambarkan sebuah *class* diagram aplikasi yang dibangun. *Class* diagram merupakan model statis yang menggambarkan struktur dan deskripsi class serta hubungannya antara class. Pada class diagram diatas mendeskripsikan hubungan antar class peserta, mengerjakan, paket, jurusan, dan mapel. Masing-masing dari class mempunyai multiplicity dan method masing-masing.

3.7.6 Perancangan Basis Data



Gambar 3. 74 Entity Relational Diagram

Pada gambar 3.74 menggambarkan ERD yang menjelaskan setiap siswa mempunyai jurusan dan setiap jurusan mempunyai mata pelajaran. Sekaligus setiap mata pelajaran mempunyai paket try out sesuai dengan ketentuan. Rancangan ERD diatas setiap relasinya mempunyai cardinalitas seperti setiap siswa mempunyai 1 jurusan dan 1 jurusan mempunyai bisa lebih dari 1 mata pelajaran untuk try out.

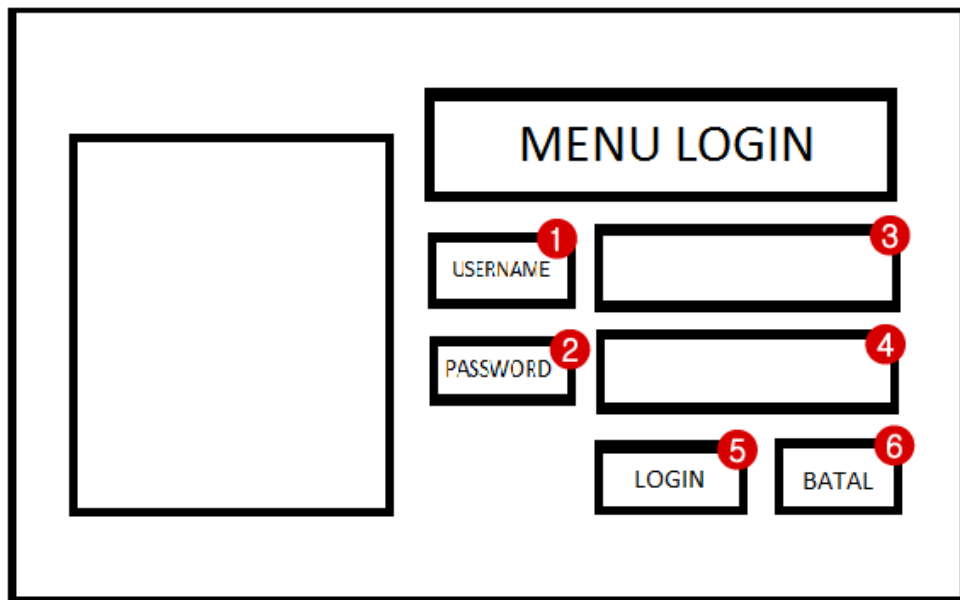


Gambar 3. 75 Physical Data Model

Gambar 3.75 menunjukkan gambar physical data model dari penelitian yang dikerjakan hal ini bertujuan untuk memberikan informasi attribute dari table yang digunakan beserta relasi antar table. PDM ini juga menjadi acuan dalam implementasi database yang akan dibuat.

3.8 Perancangan Antarmuka

Adapun rancangan antarmuka dari sistem pendukung keputusan penerimaan siswa dengan menggunakan metode saw adalah sebagai berikut :



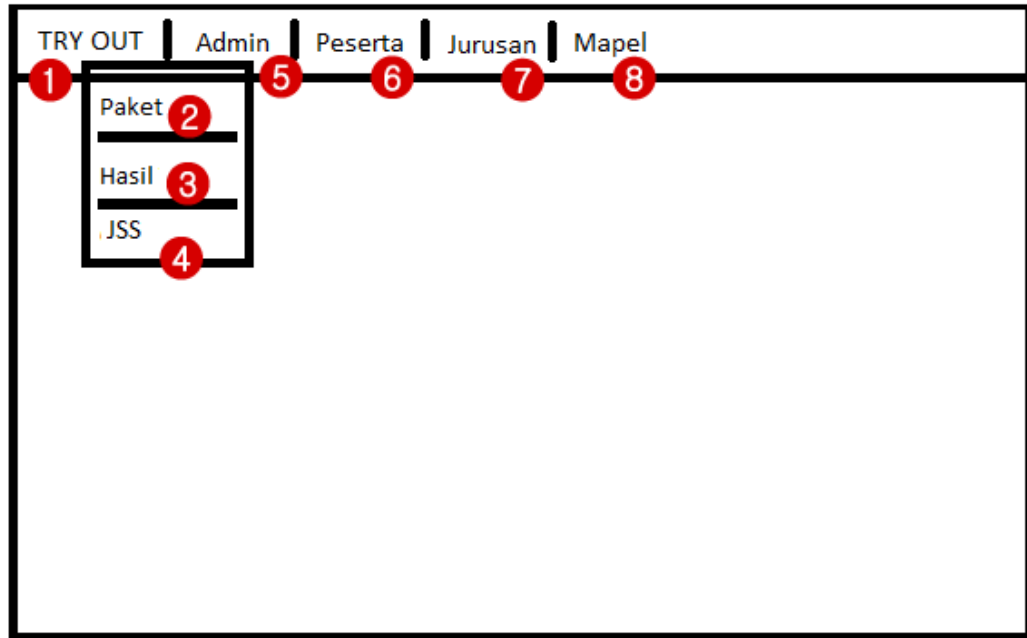
Gambar 3. 76 User Interface Login

Berikut ialah penjelasan dari *user interface* login :

Tabel 3. 30 Penjelasan User Interface Login

No	ID Objek	Jenis	Nama	Keterangan
1	Username	JLabel	Jlable1	Sebagai text keterangan username
2	Password	JLabel	Jlable2	Sebagi text keterangan password
3	Textfield1	JTextField	txtusername	Untuk memasukkan data username
4	Textfield2	JTextField	txtpassword	Untuk memasukkan data password di form login
5	Login	JButton	btnlogin	Untuk masuk ke aplikasi

No	ID Objek	Jenis	Nama	Keterangan
6	Reset	JButton	btnreset	Untuk mereset textfield



Gambar 3. 77 User Interface Menu Utama

Berikut ialah penjelasan dari *user interface* Menu Utama :

Tabel 3. 31 Penjelasan User Interface Menu Utama

No	ID Objek	Jenis	Nama	Keterangan
1	Try Out	JMenu	menuto	Sebagai text keterangan username
2	Paket	JMenuItem	JmenuItem1	Sebagai text keterangan password
3	Hasil	JMenuItem	JmenuItem2	Untuk memasukkan data username
4	JSS	JMenuItem	mnjss	Untuk memasukkan data

No	ID Objek	Jenis	Nama	Keterangan
				password di form login
5	Admin	JMenu	mnadmin	Untuk masuk ke aplikasi
6	Peserta	JMenu	mnpeserta	Untuk menampilkan form peserta
7	Jurusan	JMenu	mnjurusan	Untuk Menampilkan Form Jurusan
8	Mapel	JMenu	mnmapel	Untuk Menampilkan Form mapel

The screenshot shows a user interface for creating a question package. It includes several input fields and buttons:

- 1:** Jurusan (dropdown menu)
- 2:** Mata Pelajaran (dropdown menu)
- 3:** Tahun (text input)
- 4:** Periode (text input)
- 5:** Kode soal (dropdown menu)
- 6:** A grid of 50 question number input fields (1-50).
- 7:** Cari (text input)
- 8:** Jumlah (text input)
- 9:** Tambah (button)
- 10:** Simpan (button)
- 11:** Hapus (button)
- 12:** Batal (button)

Gambar 3. 78 User Interface Paket Soal

Berikut ialah penjelasan dari *user interface* Paket Soal :

Tabel 3. 32 Penjelasan User Interface Paket Soal

No	ID Objek	Jenis	Nama	Keterangan
1	Jurusan	JComboBox	cbojurusan	Untuk menampilkan

No	ID Objek	Jenis	Nama	Keterangan
				daftar jurusan yang telah terdaftar pada tabel jurusan.
2	Mata Pelajaran	JComboBox	cbomapel	Untuk menampilkan daftar mapel yang telah terdaftar pada tabel mapel sesuai dengan jurusan yang dipilih
3	Tahun	JTextField	txttahun	Untuk memasukkan data tahun kunci jawaban
4	Periode	JTextField	txtperiode	Untuk memasukkan data periode try out apabila dalam 1 tahun ada beberapa kali try out
5	Kode Soal	JComboBox	cbokodesoal	Untuk menampilkan jumlah kode soal dengan maksimal kode soal perkelas adalah 20 kode soal.
6	No 1-50	JTextfield	txtno(1-50)	Untuk menampung kunci jawaban soal try out
7	Cari	JTextfield	txtcari	Digunakan untuk menampung variable kata kunci yang sebagai acuan pencarian data agar lebih cepat
8	jumlah	JLabel	lbljumlah	Untuk menampilkan

No	ID Objek	Jenis	Nama	Keterangan
				jumlah data secara keseluruhan
9	Tambah	JButton	btntambah	Untuk mengaktifkan form agar pengguna dapat memasukka data baru
10	Simpan	JButton	btnsimpan	Untuk menyimpan data yang telah diinputkan ke database yang sebelumnya data akan divalidasi terlebih dahulu
11	Hapus	JButton	btnhapus	Untuk menghapus data paket
12	Batal	JButton	btnbatal	Untuk mereset form dalam kondisi awal
13	Kunci	JButton	btnkunci	Untuk mengaktifkan form kunci sesuai dengan mata pelajaran yang dipilih.

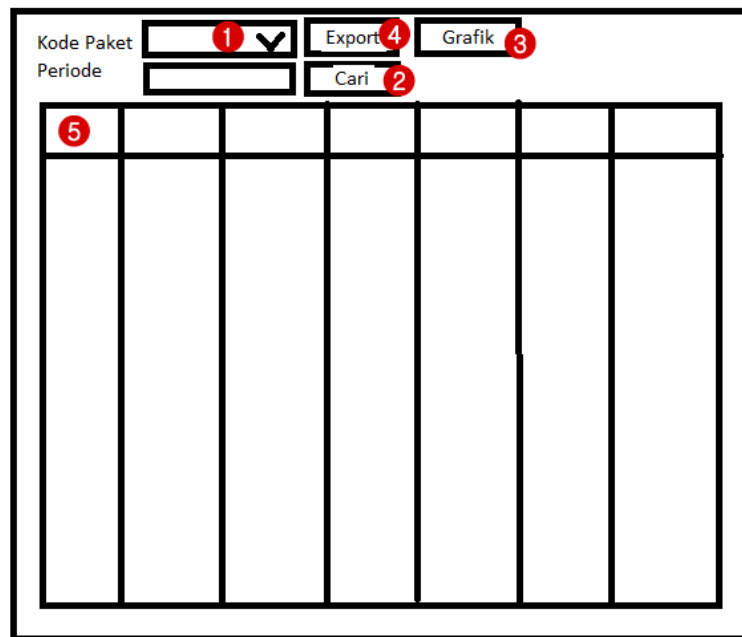
Gambar 3. 79 User Interface

Berikut ialah penjelasan dari *user interface* Hasil Try Out :

Tabel 3. 33 Penjelasan User Interface Hasil Try Out

No	ID Objek	Jenis	Nama	Keterangan
1	Jurusan	JComboBox	Cbojurusan	Untuk mnampilkan tabel jurusan
2	Tahun	JLabel	txttahun	Untuk menampilkan jumlah tabel yang akan di cari
3	Cari	JButton	btncari	Tombol yang digunakan untuk mencari tahun
4	Menu	JButton	btnkoreksi	Untuk menghubungkan program java dengan matlab (program matlab digunakan untuk deteksi jawaban LJK)

No	ID Objek	Jenis	Nama	Keterangan
5	Tabel	JTable	Tbldaftar	Untuk menampilkan daftar hasil tryout
6	Reset	JButton	btnreset	Untuk kembali mengosongkan field yang sudah di isi
7	Cetak	JButton	Btncetak	Untuk menampilkan hasil tryout berupa file pdf yang siap cetak



Gambar 3. 80 User Interface Menampilkan Jumlah Siswa Salah

Berikut ialah penjelasan dari *user interface* Jumlah Siswa Salah :

Tabel 3. 34 Penjelasan User Interface Jumlah Siswa Salah

No	ID Objek	Jenis	Nama	Keterangan
1	Kode Paket	Combo box	Cbokodepaket	Untuk memilih paket apa

No	ID Objek	Jenis	Nama	Keterangan
				saja yang ada didalam soal tryout
2	Cari	Button	Btncari	Untuk menampilkan kode paket yang dipilih
3	Grafik	Button	Btngrafik	Menampilkan hasil jawaban siswa yang salah dari soal tryout berupa grafik
4	Export	Button	Btnexport	Untuk mengexport data jawaban siswa yang salah kedalam data excel
5	Tabel	Table	jScrollPane1	Menampilkan jawaban soal siswa yang salah dalam bentuk tabel

The image shows a user interface for managing admin data. It consists of two main parts: a form for adding or updating admin users and a table for displaying the data.

Form Section:

- 1:** ID-Pengguna input field
- 2:** Username input field
- 3:** Password input field
- 4:** Password Lama input field
- 5:** Password Baru input field
- 6:** Konfirm Pwd input field
- 7:** Simpan button
- 8:** Batel button
- 9:** Tambah Admin button
- 10:** Ubah Password button
- 11:** Reset Password button

Table Section:

- 12:** Table with columns: Id Pengguna, Username
- Search bar: Cari: [input field]
- Jumlah: [input field]

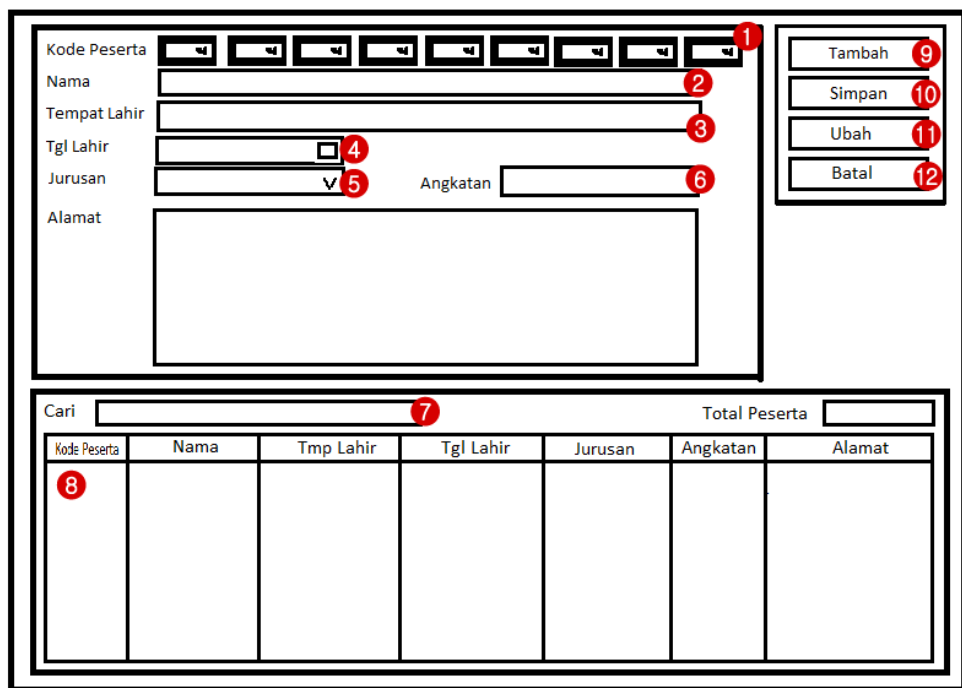
Gambar 3. 81 User Interface Mengelola Data Admin

Berikut ialah penjelasan dari *user interface* Mengelola Data Admin :

Tabel 3. 35 Penjelasan *User Interface* Mengelola Data Admin

No	ID Objek	Jenis	Nama	Keterangan
1	ID	Jlabel	Lblidpg	Untuk memasukkan Id Pengguna
2	Username	JtextField	Txtusername	Untuk memasukkan username pengguna
3	Password	JPasswordField	Txtpassword	Untuk memasukkan data password yang diinputkan dalam database
4	PasswordLama	JPasswordField	Txtpasslama	Untuk mengganti kata sandi yang sudah ada
5	PasswordBaru	JPasswordField	Txtpassbaru	Untuk mengganti kata sandi yang baru
6	Konfirmasi Pwd	JPasswordField	Txtkonfirmasi	Untuk mengkonfirmasi password baru yang ingin diganti
7	Simpan	JButton	Btnsimpan	Untuk menyimpan perubahan data
8	Batal	JButton	Btnbatal	Untuk membatalkan proses pergantian password yang ingin dibuat
9	Tambah	JButton	Btntambahadmin	Untuk menambah data admin yang ingin ditambahkan
10	Ubah Password	JButton	Btubahpass	Untuk mengubah

No	ID Objek	Jenis	Nama	Keterangan
				password lama dan password baru
11	Reset Password	JButton	jButton1	Untuk mereset field data password yang akan di isi.
12	Tabel Daftar Admin	JTable	Tbldaftaradmin	Untuk menampilkan daftar para admin yang akan mengelola aplikasi



Gambar 3. 82 User Interface Mengelola Data Peserta

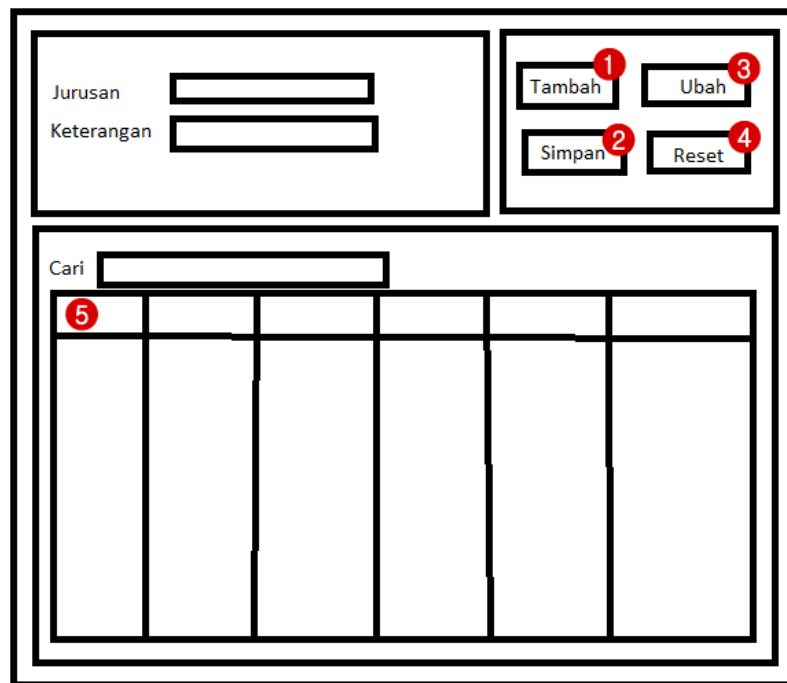
Berikut ialah penjelasan dari *user interface* Mengelola Data Peserta :

Tabel 3. 36 User Interface Mengelola Data Peserta

No	ID Objek	Jenis	Nama	Keterangan
1	Kode Peserta	JText field	Kd1	Untuk mengisi kode

No	ID Objek	Jenis	Nama	Keterangan
				kode dari setiap peserta
2	Nama	JText field	Txtnama	Untuk mengisi data nama peserta yang mengikuti tryout
3	Tempat lahir	JText field	Txttmptlahir	Untuk memasukkan data tempat lahir dari peserta tryout
4	Tgl lahir	Jcalender	Jdctgllahir	Untuk memasukkan tanggal lahir dari peserta tryout
5	Jurusan	JCombo box	Cbojurusan	Untuk menampilkan data jurusan yang ada
6	Alamat	JField	Txtalamat	Untuk memasukkan alamat dari data peserta tryout
7	Cari	JTextfield	Txtcari	Untuk memasukkan nama data peserta yang ingin dicari
8	Tabel	JTable	Tbldaftarpeserta	Untuk menampilkan data data tabel peserta yang sudah masuk dalam
9	Tambah	JButton	Btntambah	Untuk menambah data peserts
10	Simpan	JButton	Btnsimpan	Untuk menyimpan data yang akan di tambahkan
11	Ubsh	JButton	Btnubah	Untuk menguubah data peserta yang sudah ada

No	ID Objek	Jenis	Nama	Keterangan
				sebelumnya
12	Batal	JButton	Btnbatal	Untuk membatalkan perubahan yang akan dilakukan



Gambar 3. 83 User Interface Mengelola Jurusan

Berikut ialah penjelasan dari *user interface* Mengelola Jurusan :

Tabel 3. 37 User Interface Mengelola Jurusan

No	ID Objek	Jenis	Nama	Keterangan
1	Tambah	JButton	Btntambah	Untuk menambah data jurusan yang baru
2	Simpan	JButton	Btnsimpan	Untuk menyimpan data jurusan yang baru

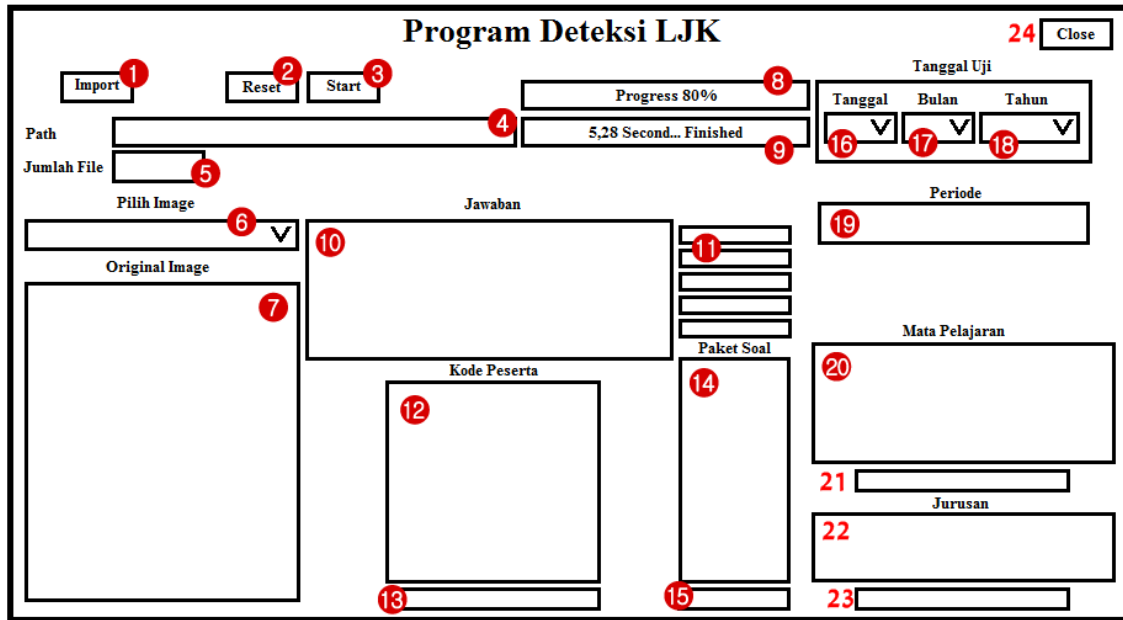
No	ID Objek	Jenis	Nama	Keterangan
				ditambahkan
3	Ubah	JButton	Btnubah	Untuk mengubah data jurusan yang sudah ada
4	Reset	JButton	Btnriset	Untuk mereset data jurusan yang sebelumnya
5	Tabel	JTable	Tbldaftarjurusan	Untuk menampilkan data-data jurusan yang sudah ada

Gambar 3. 84 User Interface Mengelola Mata Pelajaran

Berikut ialah penjelasan dari *user interface* Mengelola Mata Pelajaran :

Tabel 3. 38 User Interface Mengelola Mata Pelajaran

No	ID Objek	Jenis	Nama	Keterangan
1	Jurusan	JCombobox	Cbojurusan	Untuk menampilkan pilihan jurusan dari peserta tryout
2	Kode MP	Jlabel	Lblkode_mp	Untuk mengisi kode mata pelajaran
3	Tambah	JButton	Btntambah	Untuk menambah data mata pelajaran yang baru
4	Simpan	JButton	btnsimpan	Untuk menyimpan data mata pelajaran yang baru ditambahkan
5	Ubah	JButton	Btnubah	Untuk mengubah data mata pelajaran yang sudah ada
6	Riset	JButton	Btnriset	Untuk mereset data mata pelajaran yang sebelumnya
7	Tabel	JTable	Tbldaftarmapel	Untuk menampilkan data-data mata pelajaran yang sudah ada



Gambar 3. 85 User Interface Koreksi Citra LJK

Berikut ialah penjelasan dari *user interface* Koreksi Citra LJK :

Tabel 3. 39 User Interface Koreksi Citra LJK

No	ID Objek	Jenis	Nama	Keterangan
1	Import	Button	import	Untuk memasukkan citra LJK pada sistem
2	Reset	Button	reset	Untuk mereset form agar kembali ke kondisi awal
3	Start	Button	start_scan	Untuk memproses LJK mulai dari deteksi informasi, memeriksa jawaban, kemudian simpan jawaban.
4	Path	TextBox	pathname_string	Untuk menampilkan lokasi citra yang diolah
5	Jumlah File	TextBox	jum_scan	Untuk menampilkan jumlah citra yang dimasukkan

No	ID Objek	Jenis	Nama	Keterangan
6	Pilih Image	SelectBox	pilih_image	Untuk menampilkan list nama citra yang akan diolah
7	Original Image	Axes	original_image	Untuk menampilkan citra LJK yang akan diolah
8	Progress	Label	loading	Untuk menampilkan presentase proses citra yang diolah
9	Time finised	Label	time	Untuk menampilkan waktu dalam mengelola citra LJK
10	Jawaban	Axes	jawaban	Untuk menampilkan citra hasil kelola pada bagian jawaban.
11	Jawaban	TextBox	jawaban_text1 jawaban_text2 jawaban_text3 jawaban_text4 jawaban_text5	Untuk menampilkan informasi jawaban hasil deteksi
12	Kode Peserta	Axes	kode_peserta	Untuk menampilkan citra hasil kelola pada bagian kode peserta.
13	Kode Peserta	TextBox	kode_peserta_text	Untuk menampilkan informasi kode peserta hasil deteksi.
14	Kode paket	Axes	paket_soal	Untuk menampilkan citra hasil kelola pada bagian kode paket.
15	Kode paket	TextBox	paket_soal_text	Untuk menampilkan

No	ID Objek	Jenis	Nama	Keterangan
				informasi kode paket hasil deteksi.
16	Tanggal	SelectBox	tanggal	Untuk menampilkan list tanggal try out
17	Bulan	SelectBox	bulan	Untuk menampilkan list bulan try out
18	Tahun	SelectBox	tahun	Untuk menampilkan list tahun try out
19	Periode	TextBox	periode	Untuk mengisi data periode try out
20	Mata Pelajaran	Axes	program_studi	Untuk menampilkan citra hasil kelola pada bagian program_studi/mata pelajaran.
21	Mata Pelajaran	TextBox	program_studi_text	Untuk menampilkan informasi mata pelajaran hasil deteksi.
22	Jurusan	Axes	jurusan	Untuk menampilkan citra hasil kelola pada bagian jurusan.
23	Jurusan	TextBox	jurusan_text	Untuk menampilkan informasi jurusan hasil deteksi.
24	Close	Button	close	Untuk menutup form koreksi

BAB IV IMPLEMENTASI DAN PENGUJIAN

4.1 Implementasi Basis Data

4.1.1 Tabel Admin

Berikut ialah penjelasan dari Implementasi Basis Data Tabel Admin :

Tabel 4. 1 Basis Data Admin

Field	Type	Length	Key
ID_pengguna	varchar	5	
Username	varchar	30	Primary key
Password	varchar	50	

4.1.2 Tabel Jurusan

Berikut ialah penjelasan dari Implementasi Basis Data Tabel Jurusan :

Tabel 4. 2 Basis Data Jurusan

Field	Type	Length	Key
Kode_j	Varchar	5	Primary key
Keterangan_j	varchar	100	

4.1.3 Tabel Mapel

Berikut ialah penjelasan dari Implementasi Basis Data Tabel Mapel :

Tabel 4. 3 Basis Data Mapel

Field	Type	Length	Key
Kode_mp	varchar	5	Primary Key

Field	Type	Length	Key
Kode_j	varchar	3	Foreign Key
Nama_mp	varchar	5	
Keterangan_mp	varchar	100	

4.1.4 Tabel Mengerjakan

Berikut ialah penjelasan dari Implementasi Basis Data Tabel Mengerjakan :

Tabel 4. 4 Basis Data Mengerjakan

Field	Type	Length	Key
Kode_peserta	Varchar	12	Foreign Key
Kode_paket	Varchar	12	Foreign Key
Periode	Integer	2	Foreign Key
Tgl_to	Date		
Jawaban	Varchar	50	
J_salah	Integer	2	
J_benar	Integer	2	
Nilai	Integer	3	

4.1.5 Tabel Paket

Berikut ialah penjelasan dari Implementasi Basis Data Tabel Paket :

Tabel 4. 5 Basis Data Paket

Field	Type	Length	Key
Kode_paket	Varchar	12	Primary Key
Periode	Varchar	2	Primary Key

Field	Type	Length	Key
Kode_mp	Varchar	5	Foreign Key
Kunci	Varchar	50	

4.1.6 Tabel Peserta

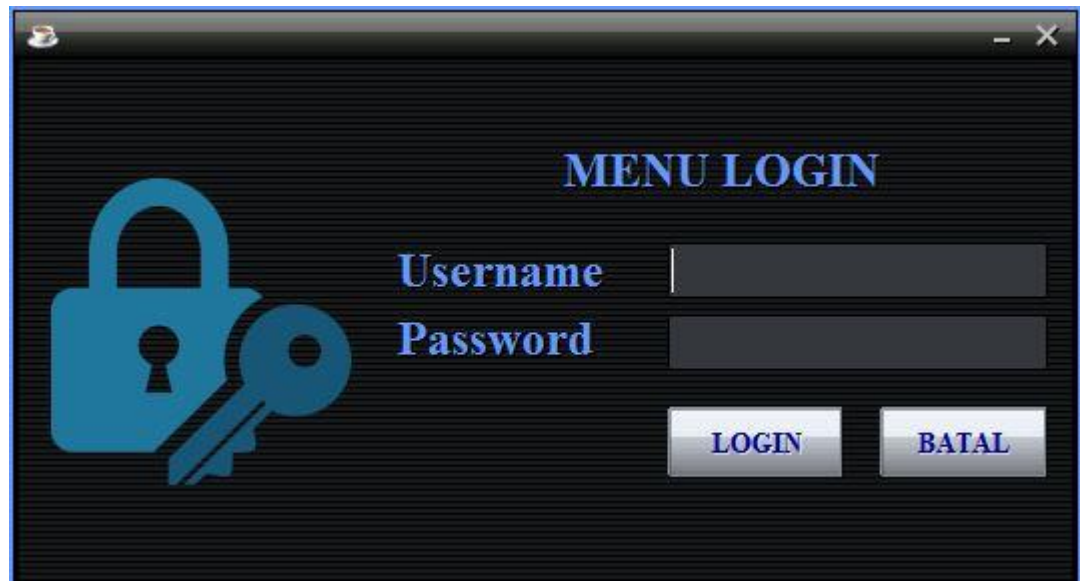
Berikut ialah penjelasan dari Implementasi Basis Data Tabel Peserta :

Tabel 4. 6 Basis Data Peserta

Field	Type	Length	Key
Kode_p	Varchar	12	Primary Key
Nama_p	Varchar	25	
Alamat_p	Varchar	100	
TempatLahir_p	Varchar	20	
TglLahir_p	Date		
Jurusan	Varchar	3	Foreign Key
Angkatan_p	Varchar	5	

4.2 Implementasi Antarmuka

4.1.1 Login



Gambar 4. 1 Implementasi Antarmuka Login

Pada gambar 4.1 merupakan design implementasi dari form login program yang dibangun dimana pada gui ini didalamnya terdapat beberapa fungsi untuk login, reset form untuk kembali dalam kondisi awal, serta menyimpan session agar dapat dikelola didalam aplikasi. Berikut screenshot source code program login:

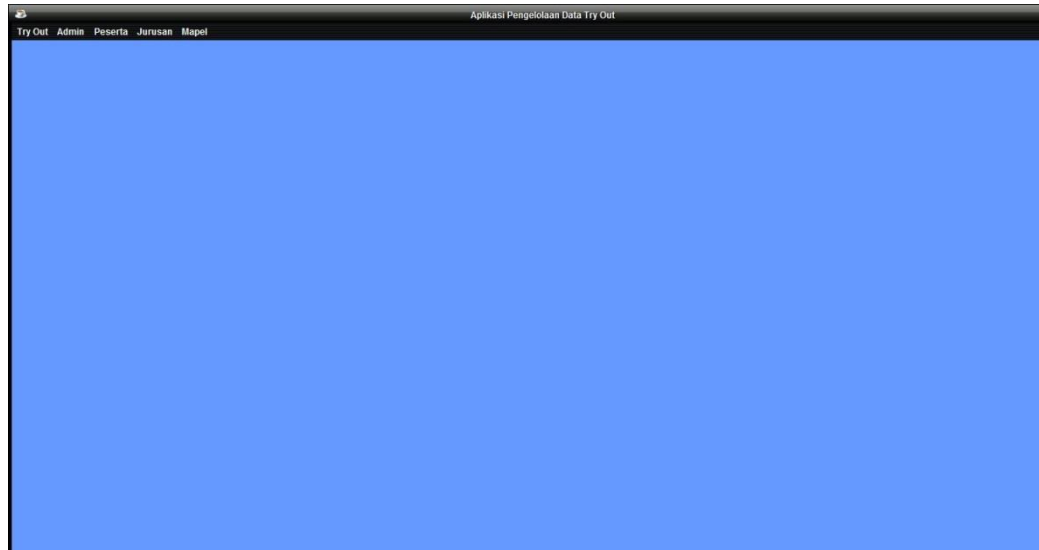
```
private void btnloginActionPerformed(java.awt.event.ActionEvent evt) {  
    if(!txtusername.getText().equals("") && !txtpassword.getText().equals("")){  
        try {  
            java.sql.Connection conn = (java.sql.Connection) koneksiDatabase.koneksiDB();  
            java.sql.Statement st = conn.createStatement();  
            java.sql.ResultSet sql = st.executeQuery("select * from admin where username='"+txtusername.getText()+"' "  
                + "AND password='"+txtpassword.getText()+"'");  
            if(sql.next()){  
                this.dispose();  
                frmmainmenu mnutama=new frmmainmenu();  
                mnutama.setVisible(true);  
                sessionuser.setUserLogin(txtusername.getText());  
                sessionuser.setPassLogin(txtpassword.getText());  
            }else{  
                JOptionPane.showMessageDialog(this, "Cek Kembali Username dan Password Anda");  
                txtpassword.setText("");  
                txtpassword.requestFocus();  
            }  
        } catch (Exception e) {  
            JOptionPane.showMessageDialog(null,"terjadi kesalahan");  
        }  
    } else {  
        JOptionPane.showMessageDialog(this,"Data Tidak Lengkap");  
    }  
}
```

Berikut source code untuk menampung session user:

```
/**
 *
 * @author ASUS A550L
 */
public class sessionuser {
private static String Username, password;

public static void setUserLogin(String Username) {
sessionuser.Username = Username;
}
public static String getUserLogin() {
return Username;
}
public static void setPassLogin(String pass){
    sessionuser.password = pass;
}
public static String getPass(){
    return password;
}
}
```

4.1.2 Menu Utama



Gambar 4. 2 Implementasi Antarmuka Menu Utama

Gambar 4.2 merupakan menu utama (parent form) dimana didalam form inilah masing2 child form akan dipanggil. Sebagai tools pengelolaan data try out. Berikut source code pemanggilan child form melalui form utama:

```
private void mnpesertaMouseClicked(java.awt.event.MouseEvent evt) {  
    frmpeserta pst = new frmpeserta();  
    mainmenu.add(pst);  
    Dimension parentSize = mainmenu.getSize();  
    Dimension childSize = pst.getSize();  
  
    //set center screen  
    pst.setLocation((parentSize.width - childSize.width)/2, (parentSize.height - childSize.height)/2);  
    pst.setVisible(true);  
}
```

4.1.3 Menu Admin



The screenshot displays a web application window titled "Menu Admin". It is divided into two main sections: "Data Admin" and "Daftar Admin".

Data Admin Section:

- ID:** A text input field containing "P-001".
- Username:** A text input field containing "admin".
- Password:** A password input field with masked characters (dots).
- Buttons:** Three buttons are located to the right of the input fields: "Tambah Admin", "Ubah Password", and "Reset Password".

Daftar Admin Section:

- Cari:** A search input field.
- Jumlah:** A label indicating the number of results, currently "2".
- Table:** A table with two columns: "ID PM" and "USERNAME".

ID PM	USERNAME
P-001	admin
P-002	ela

Gambar 4. 3 Implementasi Antarmuka Menu Admin

Gambar 4.3 merupakan form pengelolaan data admin dimana di form ini pengguna dapat mengelola data admin seperti menambah data baru admin, mereset password dan mengubah password pribadi. Berikut screenshot sebagian source code yang terdapat pada form admin:

```

public void simpan(){
    try{
        java.sql.Connection conn = (java.sql.Connection) koneksiDatabase.koneksiDB();
        String lihat="select count(*) as jumlah from admin where username = '"+adm.getTextUsername().getText()+"'";
        try(PreparedStatement lht = conn.prepareStatement(lihat)){
            ResultSet rs = lht.executeQuery();
            if(rs.next()){
                if(rs.getString("jumlah").equals("0")){
                    try{
                        java.sql.Connection cn = (java.sql.Connection) koneksiDatabase.koneksiDB();
                        String upd="insert into admin values(?, ?, ?)";
                        try(PreparedStatement ps = cn.prepareStatement(upd)){
                            ps.setString(1, adm.getLbidpg().getText());
                            ps.setString(2, adm.getTextUsername().getText());
                            ps.setString(3, adm.getTextPassword().getText());
                            ps.executeUpdate();
                            ps.close();
                            JOptionPane.showMessageDialog(null, "SUCCESS!!!");
                        }
                    }
                }
            }
        }catch(Exception r){
            JOptionPane.showMessageDialog(null, r);
        }
        }else{
            JOptionPane.showMessageDialog(null, rs.getString("jumlah"));
        }
    }
    lht.close();
}
}catch(Exception x){
    JOptionPane.showMessageDialog(null, "Terjadi Kesalahan");
}
}

```

4.1.4 Menu Peserta

The screenshot shows a Java Swing window titled "MENU PESERTA". It is divided into two main sections. The top section, "Form Pengisian Data Peserta", contains several input fields: "Kode Peserta" (a text box with a grid of characters), "Nama", "Tempat Lahir", "Tgl Lahir" (with a date picker set to Jan 4, 2017), "Jurusan" (a dropdown menu showing "--Pilih Jurusan--"), "Angkatan" (a text box), and "Alamat" (a large text area). To the right of these fields is a vertical stack of four buttons: "TAMBAH", "SIMPAN", "UBAH", and "BATAL". The bottom section, "Daftar Peserta", includes a search bar labeled "Cari" and a "Total Peserta: 1" indicator. Below this is a table with the following data:

KODE PESERTA	NAMA	TMP LAHIR	TGL LAHIR	JURUSAN	ANGKATAN	ALAMAT
02 002 027 6	JAKSEN ANG	aaaaaaa	2016-08-03	IPA	2012	Belakang Pad...

Gambar 4. 4 Implementasi Antarmuka Menu Peserta

Gambar 4.4 merupakan form peserta yang didapat digunakan oleh pengguna untuk mengelola data peserta seperti menambah data peserta, mengubah data peserta dan mencari data peserta. Berikut sebagian source code yang terdapat pada form peserta. (Simpan data peserta baru) :

4.1.5 Menu Jurusan

The screenshot shows a web application window with a blue background. The window is divided into two main sections: 'Data Jurusan' and 'Opsi'. The 'Data Jurusan' section contains two input fields: 'Jurusan' with an example 'IPA' and 'Keterangan' with an example 'Ilmu Pengetahuan Alam'. The 'Opsi' section contains four buttons: 'Tambah', 'Ubah', 'Simpan', and 'Reset'. Below these sections is a 'Daftar Jurusan' section with a search field 'Cari:' and a table. The table has two columns: 'Kode Jurusan' and 'Keterangan Jurusan'. The table contains two rows of data: IPA (Ilmu Pengetahuan Alam) and IPS (Ilmu Pengetahuan Sosial).

Kode Jurusan	Keterangan Jurusan
IPA	Ilmu Pengetahuan Alam
IPS	Ilmu Pengetahuan Sosial

Gambar 4. 5 Implementasi Antarmuka Menu Jurusan

Gambar 4.5 merupakan form yang digunakan untuk mengelola data jurusan seperti menginputkan data jurusan baru, mengubah data jurusan, mereset form, serta mencari data jurusan. Berikut sebagian source code fungsi yang terdapat pada form jurusan:

```

public void simpan() {
    try {
        java.sql.Connection conn = (java.sql.Connection) koneksiDatabase.koneksiDB();
        String lihat="select count(*) as jumlah from jurusan where kode_j = '"+jx.getTxtJurusan().getText()+"'";
        try(PreparedStatement lht = conn.prepareStatement(lihat)){
            ResultSet rs = lht.executeQuery();
            if(rs.next()){
                if(rs.getString("jumlah").equals("0")){
                    try {
                        java.sql.Connection cn = (java.sql.Connection) koneksiDatabase.koneksiDB();
                        String sql = "insert into jurusan values(?,?)";
                        try (PreparedStatement ps = cn.prepareStatement(sql)) {
                            ps.setString(1, jx.getTxtJurusan().getText());
                            ps.setString(2, jx.getTxtKeterangan().getText());
                            ps.executeUpdate();
                            ps.close();
                        }
                        JOptionPane.showMessageDialog(null, "Data Berhasil Disimpan");
                    } catch (Exception e) {
                        JOptionPane.showMessageDialog(null, e);
                    }
                } else {
                    JOptionPane.showMessageDialog(null, "Data Sudah Ada");
                    reset();
                    isiTabelSemua();
                    btnNonActive();
                    jx.getBtnTambah().setEnabled(true);
                }
            }
        }
    } catch (Exception a) {
        JOptionPane.showMessageDialog(null, a);
    }
}

```

4.1.5 Menu Mapel

Kode MP	Jurusan	Nama MP	Keterangan MP
M-001	IPA	BIN	Bahasa Indonesia
M-002	IPS	BIN	Bahasa Indonesia
M-003	IPA	BIG	Bahasa Inggris
M-004	IPS	BIG	Bahasa Inggris
M-005	IPA	MAT	Matematika
M-006	IPS	MAT	Matematika

Gambar 4. 6 Implementasi Antarmuka Menu Mapel

Gambar 4.6 merupakan form yang digunakan untuk mengelola data mapel seperti menginputkan data mapel baru, mengubah data mapel, mereset form, serta

mencari data mapel. Berikut sebagian source code fungsi yang terdapat pada form mapel:

```
public void simpan(){
    try{
        java.sql.Connection cn = (java.sql.Connection) koneksidatabase.koneksiDB();
        String sql = "insert into mapel values(?, ?, ?, ?)";
        try (PreparedStatement ps = cn.prepareStatement(sql)) {
            ps.setString(1, mp.getLblkode_mp().getText());
            ps.setString(2, mp.getCbojurusan().getSelectedItem().toString());
            ps.setString(3, mp.getTxtnama_mp().getText());
            ps.setString(4, mp.getTxtketerangan_mp().getText());
            ps.executeUpdate();
            ps.close();

            JOptionPane.showMessageDialog(null, "Data Berhasil Disimpan");
        } catch (Exception e) {
            JOptionPane.showMessageDialog(null, e);
        }
    } catch (Exception er){
        JOptionPane.showMessageDialog(null, er);
    }
}
}
```

4.1.6 Menu Paket

The screenshot shows the 'Menu Paket' application window. It contains the following elements:

- Data Paket:** Fields for 'Jurusan' (dropdown), 'Mapel' (dropdown), 'Tahun' (text), 'Periode' (text), and 'Kode Soal' (dropdown). A 'Kunci' button is next to the Jurusan field.
- Opsi:** Buttons for 'Tambah', 'Simpan', 'Hapus', and 'Batal'.
- Kunci Jawaban:** A grid of 20 numbered items, each with a small input field.
- Daftar Paket:** A search bar 'Cari Kode Paket' and a table with 10 rows. The table has columns: KODE PAKET, KODE MAPEL, KUNCI, and PERIODE.

KODE PAKET	KODE MAPEL	KUNCI	PERIODE
IPAMAT201501	M-005	dcbbbeaddbdb...	1
IPAMAT201502	M-005	dccbbeaeaeccc...	1
IPAMAT201503	M-005	acbbebdaccdd...	1
IPAMAT201504	M-005	cadcbdddbcbcc...	1
IPAMAT201505	M-005	dadabbbeadead...	1
IPSMAT201501	M-006	ddcadceeebcb...	1
IPSMAT201502	M-006	dcdaeabdbeacb...	1
IPSMAT201503	M-006	ecbdeacebcded...	1
IPSMAT201504	M-006	dcadcaaeaeec...	1
IPSMAT201505	M-006	cbacadcabaaeb...	1

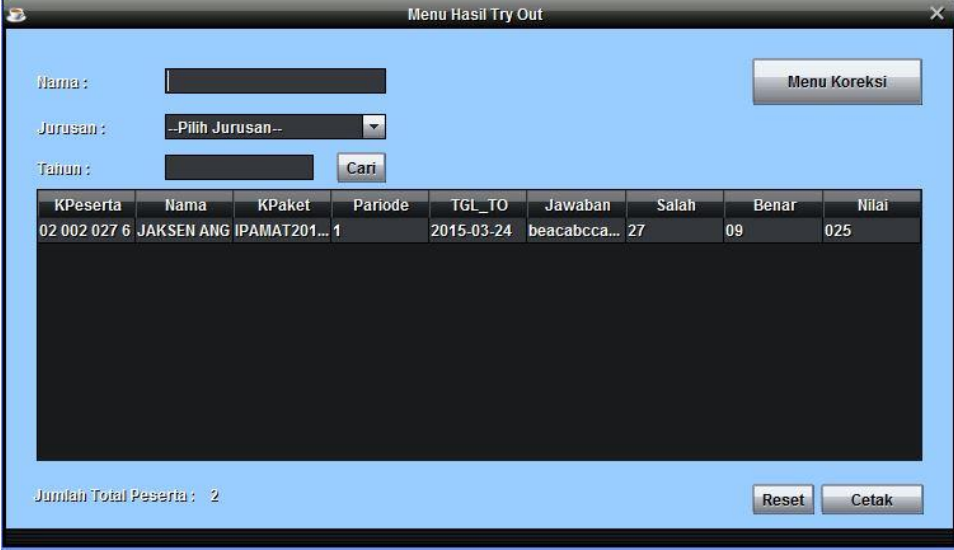
Gambar 4. 7 Implementasi Antarmuka Menu Paket

Gambar 4.7 merupakan form yang digunakan untuk mengelola data paket seperti menginputkan data paket baru, menghapus data paket serta mencari data

paket. Berikut sebagian source code fungsi yang terdapat pada form paket tepatnya source code simpan data paket:

```
        pkt.getTxtno32().getText()+pkt.getTxtno33().getText()+pkt.getTxtno34().getText()+pkt.getTxtno35().getText()+pkt.getTx
        pkt.getTxtno37().getText()+pkt.getTxtno38().getText()+pkt.getTxtno39().getText()+pkt.getTxtno40().getText();
    }
    String kode_paket = pkt.getCbojurusan().getSelectedItem().toString()+pkt.getCbomapel().getSelectedItem().toString()+
        pkt.getTtxtahun().getText()+pkt.getCbokode_soal().getSelectedItem().toString();
    String sql = "select count(*) as cek from paket where kode_paket = '"+kode_paket+"'";
    try(PreparedStatement ps = conn.prepareStatement(sql)){
        ResultSet rs = ps.executeQuery();
        if(rs.next()){
            if(rs.getString("cek").equals("0")){
                try{
                    String simpan = "insert into paket values(?, ?, ?, ?)";
                    try(PreparedStatement pss = conn.prepareStatement(simpan)){
                        pss.setString(1, kode_paket);
                        pss.setString(2, pkt.getTxperiode().getText());
                        pss.setString(3, pkt.getLblkode_mp().getText());
                        pss.setString(4, kode.toLowerCase());
                        pss.executeUpdate();
                        pss.close();
                    }catch(Exception x){
                        JOptionPane.showMessageDialog(null, x);
                    }
                }catch(Exception er){
                    JOptionPane.showMessageDialog(null, er);
                }
            }else{
                JOptionPane.showMessageDialog(null, "Kode Paket Sudah Ada");
            }
        }
    }catch(Exception err){
```

4.1.7 Menu Hasil Try Out



KPeserta	Nama	KPaket	Pariode	TGL_TO	Jawaban	Salah	Benar	Nilai
02 002 027 6	JAKSEN ANG	IPAMAT201...	1	2015-03-24	beacabcca...	27	09	025

Jumlah Total Peserta : 2

Gambar 4. 8 Implementasi Antarmuka Menu Hasil Try Out

Gambar 4.8 merupakan form yang digunakan untuk menampilkan data hasil try out pada form tersebut terdapat button koreksi digunakan untuk membuka form koreksi. Berikut sebagian source code fungsi yang terdapat pada form hasil try out:

dapat dianalisa lebih lanjut. Selain itu pada form ini juga terdapat fitur untuk menampilkan data dalam bentuk grafik maupun ekspor dalam bentuk file excel / .xls. Berikut sebagian source code fungsi yang terdapat pada form JSS tepatnya source code untuk ekspor data dalam bentuk excel / .xls:

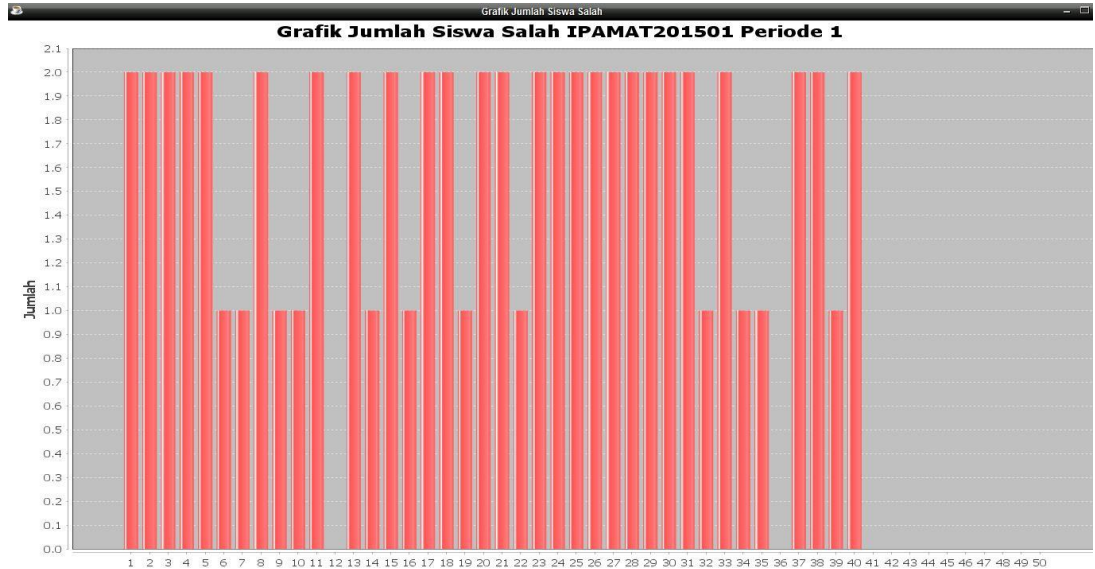
```
public void writeExcel(){
    try{
        String filename="D:/data_JSS.xls" ;
        HSSFWorkbook hwb=new HSSFWorkbook();
        HSSFSheet sheet = hwb.createSheet("new sheet");

        HSSFRow rowhead= sheet.createRow((short)0);
        rowhead.createCell((short) 0).setCellValue("Kode Paket");
        rowhead.createCell((short) 1).setCellValue("Periode");
        rowhead.createCell((short) 2).setCellValue("No.1");
        rowhead.createCell((short) 3).setCellValue("No.2");
        rowhead.createCell((short) 4).setCellValue("No.3");
        rowhead.createCell((short) 5).setCellValue("No.4");
        rowhead.createCell((short) 6).setCellValue("No.5");
        rowhead.createCell((short) 7).setCellValue("No.6");
        rowhead.createCell((short) 8).setCellValue("No.7");
        rowhead.createCell((short) 9).setCellValue("No.8");
        rowhead.createCell((short) 10).setCellValue("No.9");
        rowhead.createCell((short) 11).setCellValue("No.10");
        rowhead.createCell((short) 12).setCellValue("No.11");
        rowhead.createCell((short) 13).setCellValue("No.12");
        rowhead.createCell((short) 14).setCellValue("No.13");
        rowhead.createCell((short) 15).setCellValue("No.14");
        rowhead.createCell((short) 16).setCellValue("No.15");
        rowhead.createCell((short) 17).setCellValue("No.16");
        rowhead.createCell((short) 18).setCellValue("No.17");
        rowhead.createCell((short) 19).setCellValue("No.18");
        rowhead.createCell((short) 20).setCellValue("No.19");
    }
}
```

Hasil ekspor dalam file excel:

	A1	fx Kode Paket										
	A	B	C	D	E	F	G	H	I	J	K	L
1	Kode Paket	Periode	No.1	No.2	No.3	No.4	No.5	No.6	No.7	No.8	No.9	No.10
2	IPAMAT201501	1	3	4	2	3	3	1	4	3	1	2
3	IPAMAT201502	1	0	2	0	0	0	2	2	0	2	0

4.1.9 Form Grafik



Gambar 4. 10 Implementasi Antarmuka Grafik

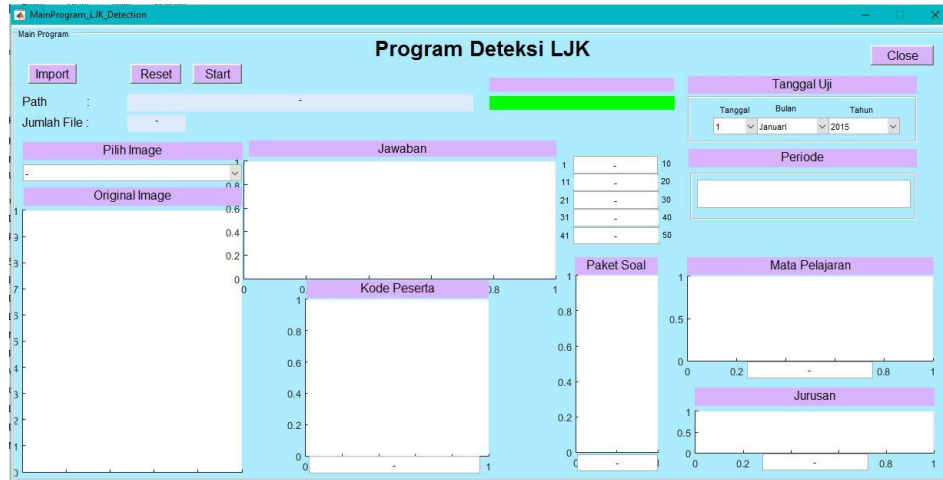
Gambar 4.10 merupakan form yang menampilkan grafik berdasarkan data dari form jss hal tersebut bertujuan agar pengguna data mengetahui jumlah siswa yang menjawab salah pada nomor tertentu sehingga dapat dianalisa lebih lanjut. Berikut sebagian source code fungsi yang terdapat pada form JSS untuk menampilkan grafik:

```
"\n" +
"FROM paket p inner join pengerjakan m on\n" +
" p.kode_paket=m.kode_paket AND p.periode=m.periode where p.kode_paket = '"+fj.getCbkodepaket().getSelectedItem().toString()+"' and p.pe
try(PreparedStatement ps = cn.prepareStatement(sql)){
ResultSet rs = ps.executeQuery();
while(rs.next()){
dcd.setValue(Integer.parseInt(rs.getString("no1")), "Jumlah Siswa", "1");
dcd.setValue(Integer.parseInt(rs.getString("no2")), "Jumlah Siswa", "2");
dcd.setValue(Integer.parseInt(rs.getString("no3")), "Jumlah Siswa", "3");
dcd.setValue(Integer.parseInt(rs.getString("no4")), "Jumlah Siswa", "4");
dcd.setValue(Integer.parseInt(rs.getString("no5")), "Jumlah Siswa", "5");
dcd.setValue(Integer.parseInt(rs.getString("no6")), "Jumlah Siswa", "6");
dcd.setValue(Integer.parseInt(rs.getString("no7")), "Jumlah Siswa", "7");
dcd.setValue(Integer.parseInt(rs.getString("no8")), "Jumlah Siswa", "8");
dcd.setValue(Integer.parseInt(rs.getString("no9")), "Jumlah Siswa", "9");
dcd.setValue(Integer.parseInt(rs.getString("no10")), "Jumlah Siswa", "10");

dcd.setValue(Integer.parseInt(rs.getString("no11")), "Jumlah Siswa", "11");
dcd.setValue(Integer.parseInt(rs.getString("no12")), "Jumlah Siswa", "12");
dcd.setValue(Integer.parseInt(rs.getString("no13")), "Jumlah Siswa", "13");
dcd.setValue(Integer.parseInt(rs.getString("no14")), "Jumlah Siswa", "14");
dcd.setValue(Integer.parseInt(rs.getString("no15")), "Jumlah Siswa", "15");
dcd.setValue(Integer.parseInt(rs.getString("no16")), "Jumlah Siswa", "16");
dcd.setValue(Integer.parseInt(rs.getString("no17")), "Jumlah Siswa", "17");
dcd.setValue(Integer.parseInt(rs.getString("no18")), "Jumlah Siswa", "18");
dcd.setValue(Integer.parseInt(rs.getString("no19")), "Jumlah Siswa", "19");
dcd.setValue(Integer.parseInt(rs.getString("no20")), "Jumlah Siswa", "20");

dcd.setValue(Integer.parseInt(rs.getString("no21")), "Jumlah Siswa", "21");
dcd.setValue(Integer.parseInt(rs.getString("no22")), "Jumlah Siswa", "22");
```

4.1.10 Menu Koreksi



Gambar 4. 11 Implementasi Antarmuka Grafik

Gambar 4.11 merupakan form koreksi untuk mengelola citra LJK yang dimasukkan agar hasil dari pengelolaan citra tersebut dapat dijadikan data yang diharapkan dapat diteliti lebih lanjut. Berikut sebagian source code fungsi yang terdapat pada form koreksi :

```

1040 - conn=database('DBTAVI','',''); % connect to database
1041 - % data_paket = exec(conn,'INSERT INTO mengerjakan kode_peserta,kode_paket,periode,tgl_to,jawaban VALUES kode_p
1042 - tampil =['select kunci from paket where kode_paket = "',kode_paket_fix,'" and periode = "',num2str(periode_fix),'"]
1043 - ex=fetch(conn,tampil);
1044 - konv = cell2mat(ex);
1045 - if isempty(ex)
1046 -     msgbox(['Kunci Paket ',kode_paket_fix, ' Periode ',num2str(periode_fix),' Tidak Ditemukan!']);
1047 - else
1048 -     jum = length(konv);
1049 -     jb = 0;
1050 -     js = 0;
1051 -     nilai = 0;
1052 -     for i=1:jum
1053 -         if konv(i) == jawaban_fix(i)
1054 -             jb=jb+1;
1055 -             js = jum - jb;
1056 -             nilai = (jb/jum)*100;
1057 -             nilai1st=num2str(nilai);
1058 -         end
1059 -     end
1060 -     exdata(1,1) = {kode_peserta_fix};
1061 -     exdata(1,2) = {kode_paket_fix};
1062 -     exdata(1,3) = {periode_fix};
1063 -     exdata(1,4) = {tanggal_uji_fix};
1064 -     exdata(1,5) = {jawaban_fix};
1065 -     exdata(1,6) = {js};
1066 -     exdata(1,7) = {jb};
1067 -     exdata(1,8) = {nilai1st};
1068 -     fastinsert(conn, 'mengerjakan', {'kode_peserta','kode_paket', 'periode', 'tgl_to', 'jawaban','j_salah','j_benar',
1069 -     end

```

4.3 Pengujian

4.1.1 Deskripsi Pengujian

Aplikasi ini diuji menggunakan metode blackbox, dimana metode pengujian blackbox ini dilakukan dengan mengamati hasil eksekusi melalui data uji dan memeriksa fungsionalitas suatu sistem secara lengkap. Terdapat beberapa fungsionalitas yang akan diuji, antara lain :

1. Login
2. Mengelola data admin
3. Mengelola data jurusan
4. Mengelola data mata pelajaran
5. Mengelola data peserta
6. Mengelola data paket
7. Menampilkan hasil try out siswa
8. Menampilkan data jumlah siswa salah dengan grafik
9. Mengekspor data jumlah siswa salah dalam bentuk excel
10. Mendeteksi informasi citra LJK

4.1.2 Hasil Pengujian

Pengujian dengan metode blackbox telah dilaksanakan sesuai dengan deskripsi masing-masing fungsi. Hasil pengujian menunjukkan bahwa aplikasi ini berjalan dengan cukup baik. Untuk hasil pengujian tiap fungsional dapat dilihat pada Lampiran I.

BAB V

KESIMPULAN DAN SARAN

5.1 Kesimpulan

Berdasarkan tahapan analisis, perancangan, serta implementasi pada sistem yang telah dibangun, maka dapat ditarik kesimpulan bahwa

1. Aplikasi berhasil mendeteksi jawaban, kode soal, kode peserta, dan jurusan dari citra LJK
2. Aplikasi telah berhasil menampilkan hasil koreksi berupa nilai serta data siswa (peserta)
3. Aplikasi telah berhasil menampilkan rekapitulasi hasil koreksi jawaban yang benar dan salah

Berdasarkan hasil pengujian yang telah terlampir pada lampiran II maka dapat disimpulkan tingkat keakuratan aplikasi yang dibangun telah mencapai 89,1% dimana dari 285 butir jawaban terdapat 254 butir yang dapat terdeteksi dengan benar sedangkan sisanya terdeteksi salah.

5.2 Saran

Untuk pengembangan lebih lanjut, diharapkan pengembang selanjutnya dapat meningkatkan akurasi dari aplikasi ini dengan metode yang sama atau menggunakan metode lain yang dapat digunakan dengan baik dan menambahkan fitur :

1. Dapat mengenali tulisan tangan untuk tanggal ujian.
2. Dapat mengenali tanda tangan peserta.
3. Dapat mendeteksi tingkat kemiringan kertas dan memperbaiki posisi kertas menjadi tegak lurus sehingga informasi yang diperoleh lebih akurat.
4. Tampilan lebih interaktif dan menarik.
5. Dapat mengimport data peserta dari excel.

6. Inputan citra LJK lebih fleksibel dengan berbagai format gambar dan berbagai ukuran pixel.

DAFTAR PUSTAKA

- A. Johar, dkk, “Aplikasi Pengelolaan Citra Digital untuk Pendeteksi Jawaban pada Lembar Jawaban Komputer Menggunakan Algoritma Sobel (Studi Kasus SMP N 2 Bengkulu)”, Jurnal Informatika, Bengkulu, Indonesia, 60-67, 2014.
- B. Amaliah, dkk, “Penggunaan Persamaan Difusi Nonlinier dan Deteksi Tepi Canny untuk Segmentasi Citra Melanoma”, Seminar Sistem Informasi Indonesia (SESINDO2010) – ITS, Surabaya, Indonesia, 55-60, 2010.
- E. J. Mohammad, dkk, “Study Sobel Edge Detection Effect on the ImageEdges Using MATLAB”, International Journal of Innovative Research in Science, Engineering and Technology”, Baghdad, Iraq, 10408-10415, 2014.
- Ermundari, “Perancangan Aplikasi Pemindai Citra Lembar Jawaban Komputer (LJK) Dengan Teknik Smart Scan”, Jurnal Informatika , Tanjungpura, Indonesia, 2013.
- F. Suryanisih, “Komparasi Algoritma Deteksi Tepi (EDGE DETECTION) untuk Deteksi Tumor Hepar”, Jurnal Perangkat Nuklir, Yogyakarta, Indonesia, 26-32, 2012.
- I. Munandar, dkk, “Implementasi Algoritma Deteksi Tepi Untuk Menentukan Kualitas Surface pada Mutiara Laut dengan Metode Sobel (Study Kasus PT. AUTORE Pearl Farm Show Room)”, Seminar Teknik Informatika UMM, Malang, Indonesia, 1-13, 2015.
- Lifwarda, dkk, “Unjuk Kerja Kombinasi Metode Deteksi Tepi Pada Citra Untuk Menghasilkan Tepi Objek Yang Lebih Jelas” , Poli Rekayasa, Padang, Indonesia, 14-20, 2013

- P. T. K. Putra dan N. K. A. Wirdiani, “Pengolahan Citra Digital Deteksi Tepi Untuk Membandingkan Metode *Sobel*, *Robert* dan *Canny*”, Merpati, Bali, Indonesia, 253-261, 2014.
- Riwinoto,S.T,M.Kom, “Penggunaan Algoritma Hough Tranforms Untuk Deteksi Bentuk Lingkaran pada Ruang 2D”, KKT Game dan Multimedia, Batam, Indonesia, 1-4,2012.
- S. E. Indraani, dkk, “ Implementasi Edge Detection Pada Citra Grayscale dengan Metode Operator Prewitt dan Operator Sobel”, Jurnal Informatika, Sumatra Utara, Indonesia, 1-5, 2014.
- S. Pribadi, dkk, “Metode Pendeteksian Tepi Sobel dan Canny untuk Pengukuran Diameter Serat *Nonwoven Polycarbosilane*”, Seminar Nasional XI SDM Teknologi Nuklir, Yogyakarta, Indonesia, 248-252, 2015.

LAMPIRAN I HASIL PENGUJIAN

LAMPIRAN II DATA UJI LJK