

BAB IV

ANALISIS DAN PERANCANGAN

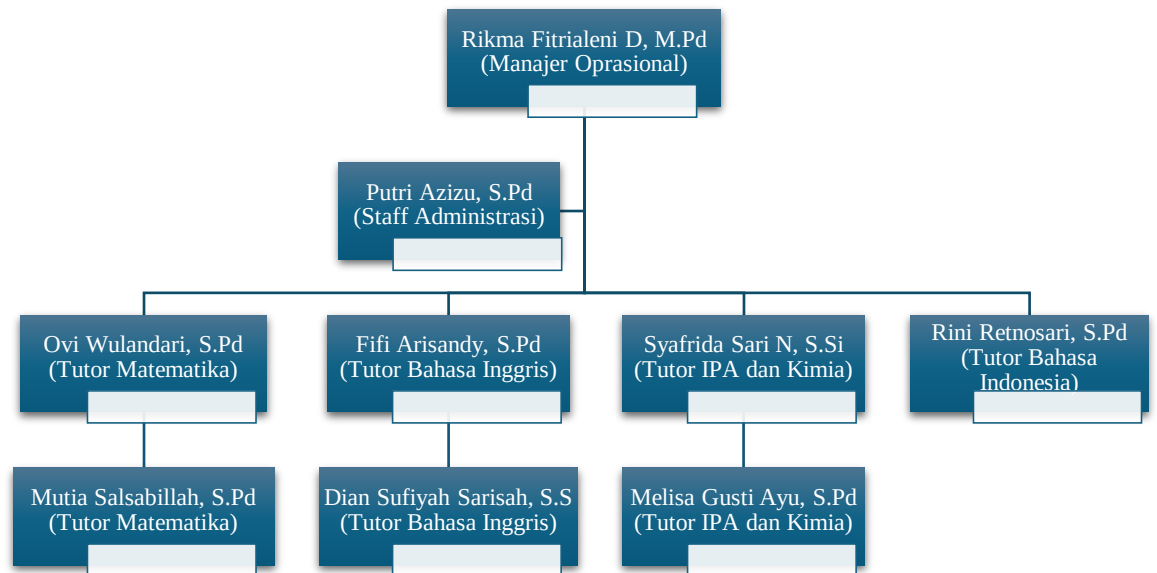
4.1 ANALISIS SISTEM

4.1.1 Gambaran Umum Bimbel Quadrant Jambi

Bimbingan Belajar (Bimbel) Quadrant merupakan salah satu lembaga pendidikan non-formal yang bergerak di bidang bimbingan belajar dan berlokasi di Kota Jambi. Sebagai lembaga yang menyelenggarakan program pembelajaran tambahan bagi siswa, Bimbel Quadrant secara rutin mengadakan evaluasi pembelajaran melalui ujian atau try-out untuk mengukur tingkat pemahaman siswa terhadap materi yang telah diajarkan.

Bimbel Quadrant dipimpin oleh Ibu Rikma Fitrialeni yang berperan sebagai Manajer sekaligus Owner. Dengan jumlah siswa yang terus bertambah dan program pembelajaran yang beragam, Bimbel Quadrant memerlukan sistem evaluasi yang efektif, efisien, dan mampu memberikan umpan balik yang cepat kepada siswa maupun pengajar. Meskipun telah memiliki fasilitas yang memadai untuk mendukung pembelajaran, pelaksanaan ujian di bimbel ini masih menghadapi berbagai kendala yang perlu segera diatasi untuk meningkatkan kualitas layanan pendidikan.

Adapun struktur organisasi dari Bimbingan Belajar Quadrant dapat dilihat pada :



Gambar 4.1 Struktur Organisasi Bimbingan Belajar Quadrant

4.1.2 Analisis Sistem Yang Sedang Berjalan

Dalam pengamatan yang dilakukan penulis berdasarkan hasil wawancara dengan Rikma Fitrialeni selaku Manajer dan Owner Bimbel Quadrant, pelaksanaan ujian di lembaga ini saat ini masih menggunakan *platform* Google Forms yang menghadapi berbagai keterbatasan teknis dan fungsional. Permasalahan utama yang diidentifikasi meliputi:

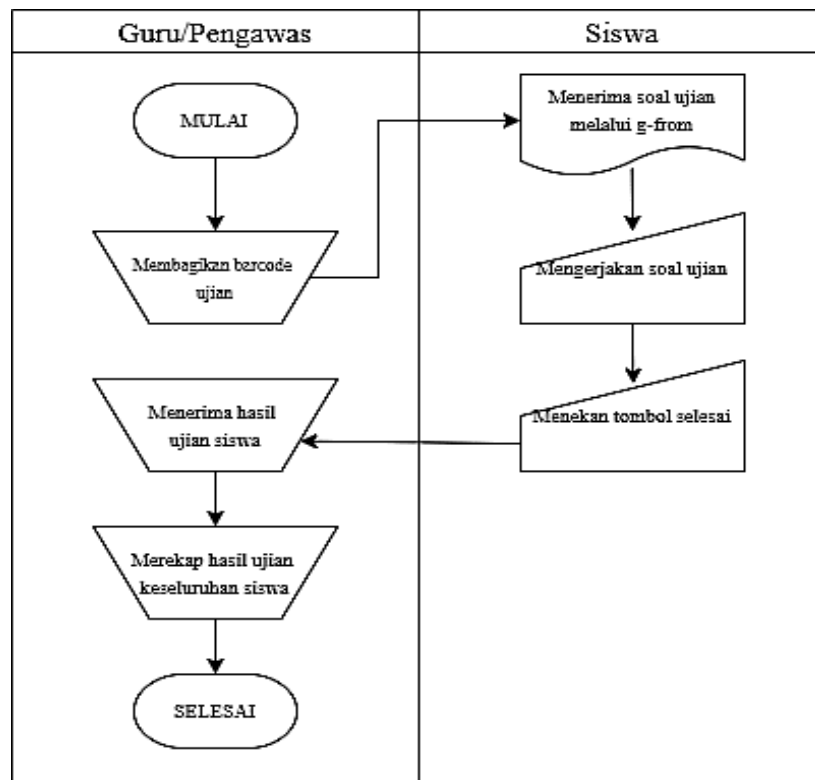
1. Ketiadaan fitur timer otomatis yang menyebabkan durasi pengerjaan ujian tidak dapat dikontrol secara sistematis
2. Link ujian yang kadang mengalami error atau tidak dapat diakses dengan baik

3. Keterbatasan dalam menampilkan format soal yang kompleks seperti rumus matematika, simbol ilmiah, atau notasi kimia dan fisika
4. Keterbatasan dalam menyajikan soal interaktif yang melibatkan multimedia seperti gambar berkualitas tinggi, video pembelajaran, atau animasi
5. Proses pengelolaan data hasil ujian yang masih dilakukan secara manual
Secara jelas sistem yang berjalan saat ini dapat diuraikan.

Berikut ini adalah proses pelaksanaan ujian yang terjadi pada Bimbingan

Belajar Quadrant yaitu:

1. Guru membuat soal ujian di Google Forms secara manual
2. Guru membagikan link Google Forms kepada siswa melalui grup WhatsApp atau media komunikasi lainnya
3. Siswa mengakses link Google Forms dan mengerjakan ujian online
4. Siswa mensubmit jawaban melalui Google Forms
5. Hasil ujian dari Google Forms disimpan ke Google Spreadsheet
6. Guru mengambil data hasil ujian dari Google Spreadsheet
7. Guru mengurutkan nilai secara manual karena tidak tersedia fitur ranking otomatis
8. Guru mengolah data secara manual menggunakan Microsoft Excel untuk mendapatkan analisis nilai, persentase jawaban benar, serta rekap hasil ujian keseluruhan
9. Laporan nilai ujian telah siap



Gambar 4.2 Flowchart ujian yang sedang berjalan pada Bimbel Quadrant

Dari sistem yang berjalan diatas, maka penulis memperoleh kesimpulan bahwa proses ujian online pada Bimbingan Belajar Quadrant saat ini masih terdapat beberapa kendala yaitu:

1. Keterbatasan sistem dalam menampilkan format soal kompleks seperti rumus matematika, simbol ilmiah, notasi kimia dan fisika, serta soal interaktif dengan multimedia berkualitas tinggi
2. Hasil ujian harus diolah lagi untuk menentukan rangking
3. Ketiadaan fitur pelaporan hasil secara real-time membuat siswa tidak dapat segera mengetahui hasil ujian mereka
4. Perbandingan history data-data sebelumnya

4.1.3 Solusi Pemecahan Masalah

Berdasarkan permasalahan yang telah diidentifikasi pada sistem ujian online Bimbingan Belajar Quadrant, penulis mengusulkan solusi dengan merancang sebuah sistem aplikasi ujian online berbasis web yang lebih komprehensif dan terintegrasi. Sistem ini akan dikembangkan menggunakan framework Laravel dengan database MySQL sebagai media penyimpanan data. Tujuan utama dari perancangan sistem ini adalah untuk mengubah sistem yang lama menjadi lebih efektif dan efisien dalam mendukung kegiatan evaluasi pembelajaran di Bimbingan Belajar Quadrant.

1. Permasalahan pertama yang dihadapi adalah keterbatasan sistem dalam menampilkan format soal kompleks seperti rumus matematika, simbol ilmiah, serta notasi kimia dan fisika. Untuk mengatasi hal tersebut, penulis merancang sistem yang terintegrasi dengan library MathJax atau KaTeX sehingga mampu menampilkan berbagai rumus dan notasi ilmiah dengan akurat. Sistem juga akan dilengkapi dengan editor soal yang mendukung format LaTeX agar guru dapat membuat soal-soal eksakta dengan tampilan yang profesional dan mudah dibaca oleh siswa. Selain itu, penulis juga menambahkan fitur upload media berupa gambar, audio, dan video untuk mendukung pembuatan soal-soal interaktif berbasis multimedia. Dengan adanya manajemen bank soal yang terstruktur, guru dapat menyimpan, mengelola, dan menggunakan kembali soal-soal ujian dengan berbagai format kompleks secara lebih praktis.

2. Permasalahan kedua berkaitan dengan proses pengolahan hasil ujian yang masih harus diolah secara manual untuk menentukan ranking siswa. Penulis mengatasi permasalahan ini dengan mengembangkan modul koreksi otomatis yang mampu menilai hasil ujian dengan cepat dan akurat tanpa memerlukan intervensi manual dari pihak guru maupun admin. Sistem juga dilengkapi dengan algoritma perhitungan ranking otomatis berdasarkan nilai yang diperoleh setiap siswa, sehingga peringkat dapat langsung ditampilkan begitu ujian selesai dikerjakan. Seluruh data yang berkaitan dengan informasi ujian, data siswa, data guru, dan hasil ujian akan tersimpan dalam database MySQL yang saling terintegrasi dengan baik. Dengan demikian, proses pengolahan data dapat dilakukan secara sistematis dan data tidak rentan hilang karena tersimpan secara terpusat dalam server.
3. Permasalahan ketiga adalah ketiadaan fitur pelaporan hasil ujian secara real-time yang menyebabkan siswa tidak dapat segera mengetahui hasil ujian mereka. Untuk menjawab permasalahan ini, penulis merancang dashboard real-time yang akan menampilkan nilai beserta peringkat siswa segera setelah mereka menyelesaikan ujian. Sistem juga akan memberikan notifikasi otomatis kepada siswa mengenai hasil ujian yang telah dikerjakan. Laporan hasil ujian dapat diakses langsung oleh siswa, guru, maupun admin sesuai dengan hak akses yang dimiliki masing-masing pengguna. Penulis juga menyediakan fitur export laporan dalam format PDF atau Excel untuk keperluan dokumentasi dan pelaporan

kepada orang tua siswa, sehingga transparansi hasil evaluasi pembelajaran dapat terjaga dengan baik.

4. Permasalahan keempat adalah kesulitan dalam melakukan perbandingan history data ujian sebelumnya. Penulis mengatasi hal ini dengan mengembangkan modul history ujian yang mampu menyimpan seluruh rekam jejak hasil ujian siswa dalam database secara terstruktur. Sistem akan menampilkan analisis perkembangan nilai siswa dalam bentuk grafik sehingga guru dan siswa dapat memvisualisasikan progres belajar dari waktu ke waktu dengan lebih mudah. Fitur perbandingan nilai antar periode ujian juga disediakan untuk memudahkan evaluasi kinerja akademik siswa secara berkala. Selain itu, sistem akan menyajikan laporan statistik yang mencakup informasi rata-rata nilai, nilai tertinggi, nilai terendah, dan standar deviasi pada setiap periode ujian sebagai bahan analisis bagi pihak Bimbingan Belajar Quadrant.

Sebagai penunjang keseluruhan sistem, penulis juga merancang beberapa fitur pendukung tambahan. Siswa Bimbel Quadrant nantinya dapat mengakses sistem ujian online melalui website dengan menggunakan akun login yang telah terdaftar tanpa harus bergantung pada link Google Forms yang sering mengalami error. Sistem dilengkapi dengan timer otomatis yang dapat mengatur dan mengontrol durasi pengerjaan ujian secara sistematis, serta fitur pengacakan soal untuk mengurangi tingkat kecurangan siswa dalam pelaksanaan ujian. Penulis juga menyediakan dashboard dengan tampilan yang user-friendly untuk admin,

guru, dan siswa sehingga memudahkan setiap pengguna dalam mengoperasikan sistem sesuai dengan peran dan hak akses masing-masing.

Dengan implementasi sistem yang dirancang tersebut, diharapkan efisiensi waktu dalam pelaksanaan ujian dan pengolahan nilai akan meningkat secara signifikan. Akurasi data hasil ujian juga dapat terjamin dengan baik karena seluruh proses berjalan secara otomatis dan terintegrasi dalam satu sistem yang terpusat. Pada akhirnya, sistem ini diharapkan mampu memberikan kontribusi positif bagi peningkatan kualitas layanan evaluasi pembelajaran di Bimbingan Belajar Quadrant.

4.2 ANALISIS KEBUTUHAN SISTEM

Dalam pengembangan perancangan sistem aplikasi ujian online berbasis web ini, terdapat beberapa hal yang dibutuhkan untuk menjadikan sistem ini lebih baik. Kebutuhan tersebut dibagi menjadi kebutuhan fungsional dan kebutuhan non-fungsional.

4.2.1 Kebutuhan Fungsional Sistem

Kebutuhan fungsional sistem menggambarkan proses atau fungsi yang harus dilakukan oleh sistem untuk melayani kebutuhan pengguna. Berdasarkan analisis yang telah dilakukan, kebutuhan fungsi utama yang harus ada pada sistem adalah sebagai berikut:

A. Kebutuhan Fungsional Admin

1. *Login*

Digunakan oleh admin di Bimbel Quadrant untuk memasukkan nama pengguna dan kata sandi agar dapat masuk ke dalam sistem aplikasi ujian online

2. Home

Berfungsi untuk menampilkan halaman utama aplikasi yang berisi informasi ringkasan sistem

3. Menu Data Siswa

Digunakan admin untuk mengelola data siswa, termasuk menambah, mengubah, melihat, dan menghapus data siswa

4. Menu Data Guru

Digunakan admin untuk mengelola data guru, termasuk menambah, mengubah, melihat, dan menghapus data guru

5. Menu Kelas

Digunakan admin untuk mengelola data kelas yang tersedia di Bimbel Quadrant

6. Menu Mata Pelajaran

Digunakan admin untuk mengelola data mata pelajaran yang diajarkan di Bimbel Quadrant

7. Menu Bank Soal

Digunakan admin untuk mengelola data soal ujian yang tersimpan dalam sistem

8. Menu Jadwal Ujian

Digunakan admin untuk mengelola jadwal pelaksanaan ujian, termasuk mengatur waktu mulai dan berakhirnya ujian

9. Menu Hasil Ujian

Digunakan admin untuk melihat dan mengelola data nilai hasil ujian siswa

10. Menu Laporan

Digunakan admin untuk menghasilkan berbagai laporan terkait ujian dan hasil belajar siswa

11. Menu Ganti Kata Sandi

Digunakan untuk mengubah atau mengganti kata sandi untuk masuk ke dalam aplikasi

12. Menu Keluar

Berfungsi untuk keluar dari sistem aplikasi ujian online

B. Kebutuhan Fungsional Guru

1. Login

Digunakan oleh guru di Bimbel Quadrant untuk memasukkan nama pengguna dan kata sandi agar dapat masuk ke dalam sistem aplikasi ujian online

2. Halaman Utama

Berfungsi untuk menampilkan halaman utama aplikasi yang berisi informasi ringkasan untuk guru

3. Menu Bank Soal

Digunakan guru untuk mengelola data soal, termasuk menambah, mengubah, melihat, dan menghapus soal ujian

4. Menu Ujian

Digunakan guru untuk membuat ujian baru dan mengelola sistem ujian, termasuk mengatur durasi waktu dan jumlah soal

5. Menu Hasil Ujian

Berisi nilai-nilai ujian dari siswa yang sudah dikerjakan, guru dapat melihat detail jawaban siswa dan analisis hasil ujian

6. Menu Laporan

Digunakan guru untuk melihat dan mencetak laporan hasil ujian siswa

7. Menu Ganti Kata Sandi

Digunakan untuk mengubah atau mengganti kata sandi untuk masuk ke dalam aplikasi

8. Menu Keluar

Berfungsi untuk keluar dari sistem aplikasi ujian online

C . Kebutuhan Fungsional Siswa

1. Login

Digunakan oleh siswa di Bimbel Quadrant untuk memasukkan nama pengguna dan kata sandi agar dapat masuk ke dalam sistem aplikasi ujian online sebelum melakukan ujian

2. Halaman Utama

Berfungsi untuk menampilkan halaman utama aplikasi yang berisi informasi ujian yang tersedia untuk siswa

3. Menu Ujian

Digunakan oleh siswa untuk mengerjakan ujian yang telah dijadwalkan dengan fitur pengatur waktu otomatis

4. Menu Hasil Ujian

Digunakan siswa untuk melihat nilai ujian yang telah dikerjakan beserta peringkatnya

5. Menu Ganti Kata Sandi

Digunakan siswa untuk mengubah atau mengganti kata sandi aplikasi ujian online

6. Menu Keluar

Berfungsi untuk keluar dari sistem aplikasi ujian online

4.2.2 Kebutuhan Non-Fungsional Sistem

Analisis kebutuhan non-fungsional dilakukan untuk mengetahui spesifikasi kebutuhan sistem secara teknis. Sistem yang dirancang harus memiliki karakteristik sebagai berikut:

1. Perangkat Keras

Dapat diakses menggunakan komputer, laptop, atau perangkat mobile dengan memori RAM minimal 2 GB atau lebih

2. Perangkat Lunak

Sistem dapat diakses melalui web browser modern seperti Google Chrome, Mozilla Firefox, atau Microsoft Edge

3. Antarmuka Pengguna

Sistem menyediakan antarmuka pengguna yang sederhana, intuitif, dan mudah digunakan oleh semua tingkat pengguna

4. Kemudahan Penggunaan

Sistem dapat dioperasikan dengan mudah oleh pengguna tanpa memerlukan pelatihan khusus yang intensif

5. Keamanan

Sistem hanya dapat diakses oleh pengguna yang memiliki hak akses (siswa, guru, dan admin) di Bimbel Quadrant dengan menggunakan autentikasi nama pengguna dan kata sandi

6. Kinerja

Sistem harus dapat merespons dengan cepat dan mampu menangani banyak pengguna yang mengakses secara bersamaan

7. Ketersediaan

Sistem harus dapat diakses kapan saja selama terhubung dengan jaringan internet

8. Keandalan

Sistem harus stabil dan tidak mudah mengalami gangguan atau kesalahan selama proses ujian berlangsung

9. Portabilitas

Sistem berbasis web yang dapat diakses dari berbagai perangkat dan platform sistem operasi

4.3 PERANCANGAN SISTEM

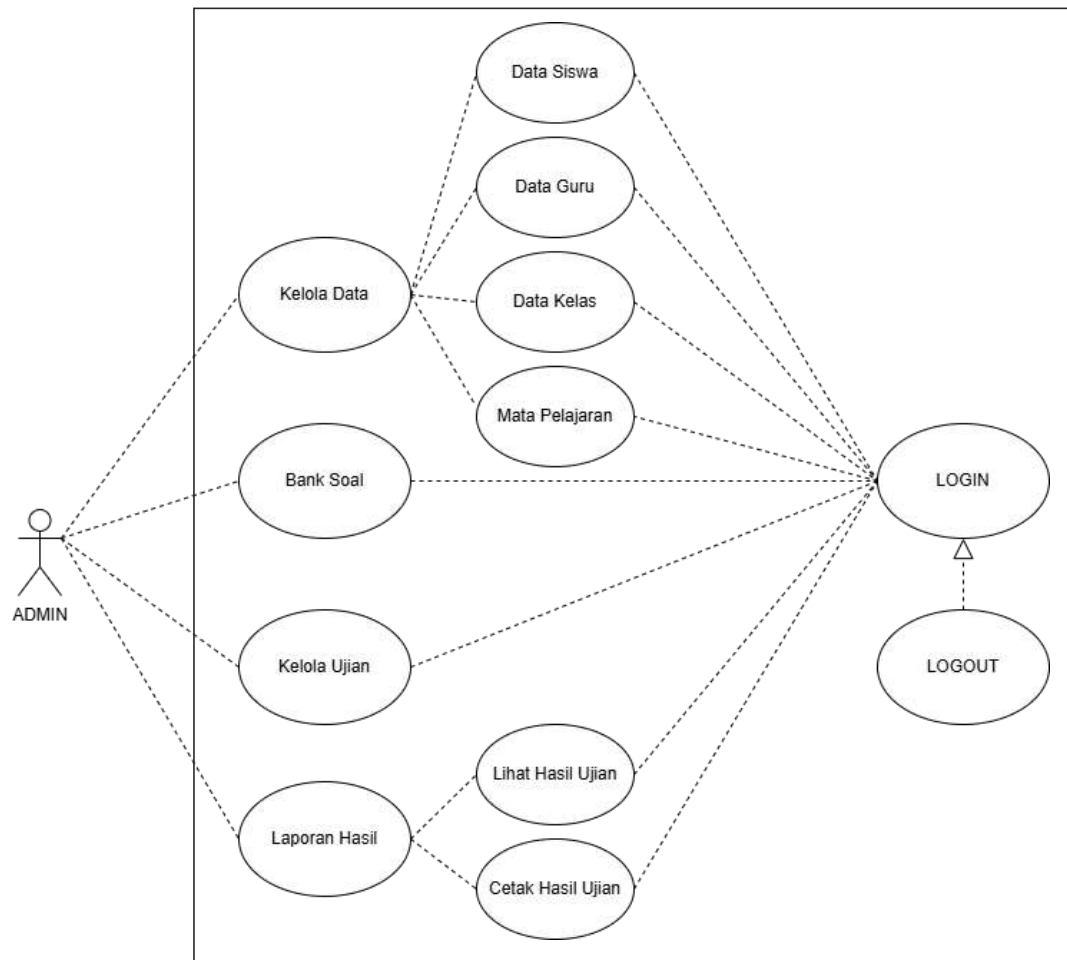
Tahap perancangan merupakan tahap awal dari perancangan perangkat lunak. Pada perancangan ini, akan dijelaskan tahapan perancangan sistem, cara kerja program aplikasi, dan gambaran arsitektur sistem mengenai aplikasi ujian online di Bimbel Quadrant. Untuk mencapai hasil yang sesuai dengan kebutuhan sistem, diperlukan beberapa tahap perancangan.

Analisis fungsi yang dibutuhkan pada aplikasi dapat digambarkan menggunakan metode perancangan berorientasi objek dengan menggunakan diagram UML (*Unified Modeling Language*), yaitu *use case diagram*, *activity diagram*, dan *class diagram* untuk menentukan visualisasi dan mendokumentasikan informasi yang digunakan atau dihasilkan dalam suatu proses pembuatan sistem.

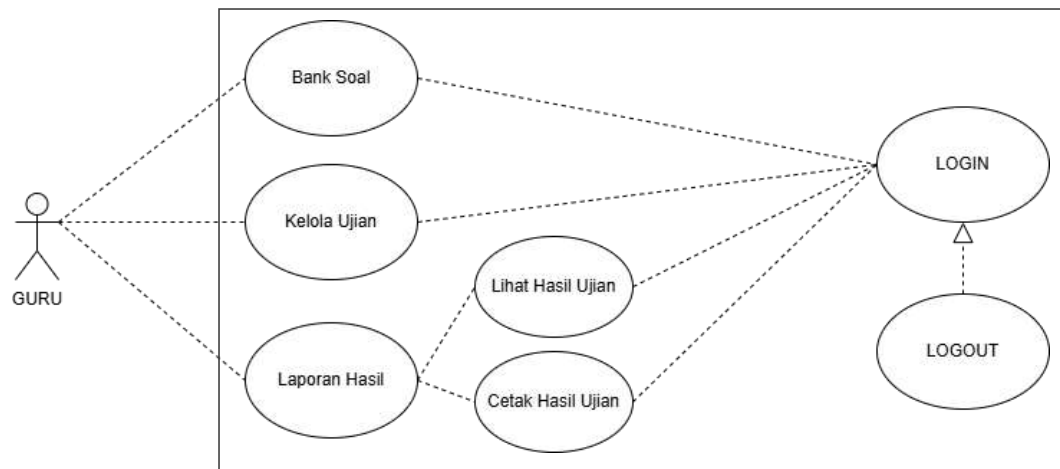
Berdasarkan analisis yang telah dilakukan dalam pembuatan aplikasi ini, dapat diketahui metode yang digunakan serta antarmuka sistem yang dirancang sesuai dengan yang diharapkan. Hasil perancangan berupa gambaran umum mengenai aplikasi secara keseluruhan yang akan menjelaskan proses implementasi sistem.

4.3.1 Perancangan Use Case Diagram

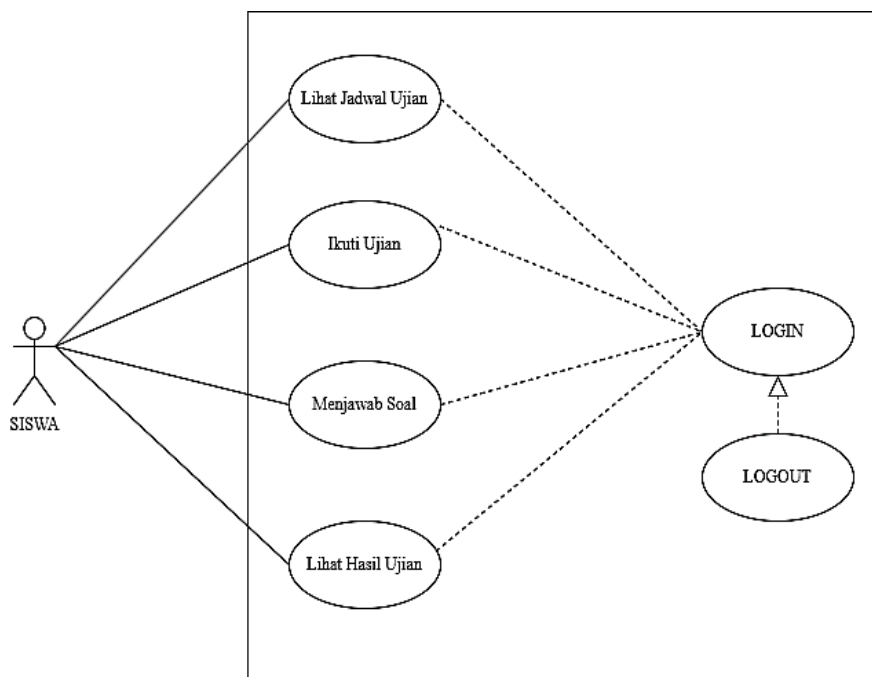
Use case diagram menggambarkan interaksi antara pengguna (aktor) dengan sistem yang akan dikembangkan. Diagram ini menunjukkan fungsionalitas sistem dari sudut pandang pengguna dan bagaimana sistem merespons terhadap aksi yang dilakukan oleh pengguna. Seperti contoh dibawah.



Gambar 4.3 Use Case Diagram Admin



Gambar 4.4 Use Case Diagram Guru



Gambar 4.5 Use Case Diagram Siswa

4.3.2 Deskripsi Use Case Diagram

Deskripsi use case merupakan penjelasan atau skenario untuk melakukan interaksi antara aktor dengan sistem yang dirancang untuk masing-masing use case yang terdapat pada use case diagram aplikasi, dan deskripsi use case antara lain:

1. Deskripsi *Use Case* Login Admin, Guru, Dan Siswa

Deskripsi use case login admin, guru, dan siswa ini menjelaskan mengenai bagaimana sistem menginput username dan password dari admin, guru, dan siswa agar berhasil masuk ke aplikasi. Untuk lebih jelasnya, deskripsi use case login dapat dilihat pada tabel 4.1.

Tabel 4.1 Deskripsi *Use Case* Login Admin, Guru, Dan Siswa

Nama	Login		
Aktor	Admin, Guru, Dan Siswa		
Deskripsi	Aktor melakukan Login untuk otentikasi dalam pengecekan dan akses sistem		
Exception	Proses Login gagal, dikarenakan username atau password tidak sesuai, kosong atau belum terdata sebelumnya		
Pre condition	Username dan Password telah tersimpan dalam database		
Aktor		Sistem	
Skenario Normal			
1.	Aktor membuka aplikasi ujian online		
		2.	Sistem menampilkan halaman <i>login</i>
3.	Aktor menginputkan Username dan Password		
4.	Aktor mengklik tombol Login		
		5.	Sistem memverifikasi Username dan Password dengan database

		6.	Data cocok, sistem menampilkan Halaman menu utama sesuai role aktor
Skenario Alternative			
Aktor batal masuk			
4.a	Aktor memilih batal masuk dengan menekan tombol Cancel		
		4.b	Sistem kembali ke halaman login
Login Gagal			
		5.a	Sistem memverifikasi dan data tidak cocok
		5.b	Sistem menampilkan pesan error "Username atau Password salah"
		5.c	Sistem kembali menampilkan form login
Post condition		Aktor berhasil masuk ke sistem sesuai hak akses	

2. Deskripsi *Use Case* Admin Mengelola Data Siswa

Deskripsi *use case* Admin mengelola data siswa ini menjelaskan mengenai bagaimana sistem mengelola data siswa dengan cara admin menekan menu pilihan Data Siswa. Untuk lebih jelasnya, deskripsi *use case* mengelola data siswa dapat dilihat pada tabel 4.2.

Tabel 4.2 Deskripsi *Use Case* Mengelola Data Siswa

Nama		Mengelola data siswa	
Aktor		Admin	
Deskripsi		Aktor dapat menambah, mengubah, melihat, dan menghapus data Siswa	
Exception		Pengolahan data tidak dapat dilakukan jika belum login	
Pre condition		Aktor telah dalam keadaan login	
Aktor		Sistem	
Skenario Normal			
1.	Aktor memilih menu Data Siswa		

		2.	Sistem menampilkan halaman Data Siswa dengan daftar siswa
3.	Kondisi Jika aktor memilih : Tombol tambah maka S1 berlaku. Tombol edit maka S2 berlaku. Tombol hapus maka S3 berlaku.		
S1 – fungsi tambah			
1.	Aktor memilih tombol Tambah		
		2.	Sistem menampilkan form tambah data siswa
3.	Aktor menginputkan data siswa (NIS, Nama, Kelas, Username, Password)		
4.	Aktor mengklik tombol Simpan		
		5.	Sistem memvalidasi kelengkapan data
		6.	Data valid, sistem menyimpan data ke database
		7.	Sistem menampilkan pesan "Data berhasil disimpan"
S2 – fungsi edit			
1.	Aktor memilih data siswa yang akan diedit		
2.	Aktor mengklik tombol Edit		
		3.	Sistem menampilkan form edit dengan data siswa terpilih
4.	Aktor memodifikasi data siswa		
5.	Aktor mengklik tombol Update		
		6.	Sistem memvalidasi kelengkapan data
		7.	Data valid, sistem menyimpan perubahan ke database
		8	Sistem menampilkan pesan "Data berhasil diupdate"
S3 – fungsi hapus			
1.	Aktor memilih data siswa yang akan dihapus		
2.	Aktor mengklik tombol Hapus		
		3.	Sistem menampilkan konfirmasi "Apakah Anda yakin ingin menghapus data ini?"
4.	Aktor mengklik tombol Ya		

		5.	Sistem menghapus data dari database
		6.	Sistem menampilkan pesan "Data berhasil dihapus"
Skenario alternative			
Data tambah tidak lengkap			
S1-5a	Saat validasi, data yang diinput belum lengkap		
S1-5b	Sistem menampilkan pesan error dan kembali ke form input		
Data edit tidak lengkap			
S2-6a	Saat validasi, data yang dimodifikasi belum lengkap		
S2-6b	Sistem menampilkan pesan error dan kembali ke form edit		
Data hapus dibatalkan			
S3-3a	Aktor memilih tombol Tidak pada konfirmasi		
S3-3b	Sistem membatalkan proses hapus dan kembali ke daftar siswa		
Post condition		Aktor berhasil mengelola data siswa	

3. Deskripsi *Use Case* Admin Mengelola Data Guru

Deskripsi *use case* mengelola data guru ini menjelaskan bagaimana sistem mengelola data guru dengan cara admin menekan menu pilihan Data Guru.

Untuk lebih jelasnya, deskripsi *use case* mengelola data guru dapat dilihat pada tabel 4.3.

Tabel 4.3 Deskripsi *Use Case* Mengelola Data Guru

Nama	Mengelola Data Guru
Aktor	Admin
Deskripsi	Aktor dapat menambah, mengubah, melihat, dan menghapus data Guru
Exception	Pengolahan data tidak dapat dilakukan jika belum login
Pre condition	Aktor telah dalam keadaan login

Aktor		Sistem	
Skenario Normal			
1.	Aktor memilih menu Data Guru		
		2.	Sistem menampilkan halaman Data Guru dengan daftar guru
3.	Kondisi Jika aktor memilih : Tombol tambah maka S1 berlaku. Tombol edit maka S2 berlaku. Tombol hapus maka S3 berlaku.		
S1 – fungsi tambah			
1.	Aktor memilih tombol Tambah		
		2.	Sistem menampilkan form tambah data guru
3.	Aktor menginputkan data guru (NIP, Nama, Mata Pelajaran, Username, Password)		
4.	Aktor mengklik tombol Simpan		
		5.	Sistem memvalidasi kelengkapan data
		6.	Data valid, sistem menyimpan data ke database
		7.	Sistem menampilkan pesan "Data berhasil disimpan"
S2 – fungsi edit			
1.	Aktor memilih data guru yang akan diedit		
2.	Aktor mengklik tombol Edit		
		3.	Sistem menampilkan form edit dengan data guru terpilih
4.	Aktor memodifikasi data guru		
5.	Aktor mengklik tombol Update		
		6.	Sistem memvalidasi kelengkapan data
		7.	Data valid, sistem menyimpan perubahan ke database
		8.	Sistem menampilkan pesan "Data berhasil diupdate"
S3 – fungsi hapus			
1.	Aktor memilih data guru yang akan dihapus		

2.	Aktor mengklik tombol Hapus		
		3.	Sistem menampilkan konfirmasi "Apakah Anda yakin ingin menghapus data ini?"
4.	Aktor mengklik tombol Ya		
		5.	Sistem menghapus data dari database
		6.	Sistem menampilkan pesan "Data berhasil dihapus"
Skenario alternative			
Data tambah tidak lengkap			
S1-5a	Saat validasi, data yang diinput belum lengkap		
S1-5b	Sistem menampilkan pesan error dan kembali ke form input		
Data edit tidak lengkap			
S2-6a	Saat validasi, data yang dimodifikasi belum lengkap		
S2-6b	Sistem menampilkan pesan error dan kembali ke form edit		
Data hapus dibatalkan			
S3-3a	Aktor memilih tombol Tidak pada konfirmasi		
S3-3b	Sistem membatalkan proses hapus dan kembali ke daftar guru		
Post condition		Aktor berhasil mengelola data guru	

4. Deskripsi *Use Case* Admin Mengelola Data Kelas

Deskripsi *use case* mengelola data kelas ini menjelaskan bagaimana sistem mengolah data kelas dengan cara admin menekan menu pilihan Kelas. Untuk lebih jelasnya, deskripsi *use case* mengelola data kelas dapat dilihat pada tabel 4.4.

Tabel 4.4 Deskripsi Use Case Mengelola Data Kelas

Nama	Mengelola Data Kelas		
Aktor	Admin		
Deskripsi	Aktor dapat menambah, mengubah, melihat, dan menghapus data Kelas		
Exception	Pengolahan data tidak dapat dilakukan jika belum login		
Pre condition	Aktor telah dalam keadaan login		
Aktor		Sistem	
Skenario Normal			
1.	Aktor memilih menu Kelas		
		2.	Sistem menampilkan halaman Kelas dengan daftar kelas
3.	Kondisi Jika aktor memilih : Tombol tambah maka S1 berlaku. Tombol edit maka S2 berlaku. Tombol hapus maka S3 berlaku.		
S1 – fungsi tambah			
1.	Aktor memilih tombol Tambah		
		2.	Sistem menampilkan form tambah data kelas
3.	Aktor menginputkan data kelas (Nama Kelas)		
4.	Aktor mengklik tombol Simpan		
		5.	Sistem memvalidasi kelengkapan data
		6.	Data valid, sistem menyimpan data ke database
		7.	Sistem menampilkan pesan "Data berhasil disimpan"
S2 – fungsi edit			
1.	Aktor memilih data kelas yang akan diedit		
2.	Aktor mengklik tombol Edit		
		3.	Sistem menampilkan form edit dengan data kelas terpilih
4.	Aktor memodifikasi data kelas		
5.	Aktor mengklik tombol Update		

		6.	Sistem memvalidasi kelengkapan data
		7.	Data valid, sistem menyimpan perubahan ke database
		8.	Sistem menampilkan pesan "Data berhasil diupdate"
S3 – fungsi hapus			
1.	Aktor memilih data kelas yang akan dihapus		
2.	Aktor mengklik tombol Hapus		
		3.	Sistem menampilkan konfirmasi "Apakah Anda yakin ingin menghapus data ini?"
4.	Aktor mengklik tombol Ya		
		5.	Sistem menghapus data dari database
		6.	Sistem menampilkan pesan "Data berhasil dihapus"
Skenario alternative			
Data tambah tidak lengkap			
S1-5a	Saat validasi, data yang diinput belum lengkap		
S1-5b	Sistem menampilkan pesan error dan kembali ke form input		
Data edit tidak lengkap			
S2-6a	Saat validasi, data yang dimodifikasi belum lengkap		
S2-6b	Sistem menampilkan pesan error dan kembali ke form edit		
Data hapus dibatalkan			
S3-3a	Aktor memilih tombol Tidak pada konfirmasi		
S3-3b	Sistem membatalkan proses hapus dan kembali ke daftar kelas		
Post condition		Aktor berhasil mengelola data kelas	

5. Deskripsi *Use Case* Admin Mengelola Mata Pelajaran

Deskripsi *use case* mengelola mata pelajaran ini menjelaskan bagaimana sistem mengelola data mata pelajaran dengan cara admin menekan menu pilihan Mata Pelajaran. Untuk lebih jelasnya, deskripsi *use case* mengelola mata pelajaran dapat dilihat pada tabel 4.5.

Tabel 4.5 Deskripsi *Use Case* Mengelola Mata Pelajaran

Nama	Mengelola Mata Pelajaran		
Aktor	Admin		
Deskripsi	Aktor dapat menambah, mengubah, melihat, dan menghapus data Mata Pelajaran		
Exception	Pengolahan data tidak dapat dilakukan jika belum login		
Pre condition	Aktor telah dalam keadaan login		
Aktor		Sistem	
Skenario Normal			
1.	Aktor memilih menu Mata Pelajaran		
		2.	Sistem menampilkan halaman Mata Pelajaran dengan daftar mata pelajaran
3.	Kondisi Jika aktor memilih : Tombol tambah maka S1 berlaku. Tombol edit maka S2 berlaku. Tombol hapus maka S3 berlaku.		
S1 – fungsi tambah			
1.	Aktor memilih tombol Tambah		
		2.	Sistem menampilkan form tambah data mata pelajaran
3.	Aktor menginputkan data mata pelajaran (Nama Mata Pelajaran, Kode)		
4.	Aktor mengklik tombol Simpan		
		5.	Sistem memvalidasi kelengkapan data
		6.	Data valid, sistem menyimpan data ke database

		7.	Sistem menampilkan pesan "Data berhasil disimpan"
S2 – fungsi edit			
1.	Aktor memilih data mata pelajaran yang akan diedit		
2.	Aktor mengklik tombol Edit		
		3.	Sistem menampilkan form edit dengan data mata pelajaran terpilih
4.	Aktor memodifikasi data mata pelajaran		
5.	Aktor mengklik tombol Update		
		6.	Sistem memvalidasi kelengkapan data
		7.	Data valid, sistem menyimpan perubahan ke database
		8.	Sistem menampilkan pesan "Data berhasil diupdate"
S3 – fungsi hapus			
1.	Aktor memilih data mata pelajaran yang akan dihapus		
2.	Aktor mengklik tombol Hapus		
		3.	Sistem menampilkan konfirmasi "Apakah Anda yakin ingin menghapus data ini?"
4.	Aktor mengklik tombol Ya		
		5.	Sistem menghapus data dari database
		6.	Sistem menampilkan pesan "Data berhasil dihapus"
Skenario alternative			
Data tambah tidak lengkap			
S1-5a	Saat validasi, data yang diinput belum lengkap		
S1-5b	Sistem menampilkan pesan error dan kembali ke form input		
Data edit tidak lengkap			
S2-6a	Saat validasi, data yang dimodifikasi belum lengkap		
S2-6b	Sistem menampilkan pesan error dan kembali ke form edit		

Data hapus dibatalkan	
S3-3a	Aktor memilih tombol Tidak pada konfirmasi
S3-3b	Sistem membatalkan proses hapus dan kembali ke daftar mata pelajaran
Post condition	Aktor berhasil mengelola data kelas

6. Deskripsi *Use Case* Admin dan Guru Mengelola Bank Soal

Deskripsi *use case* mengelola bank soal ini menjelaskan mengenai bagaimana sistem mengelola data soal dengan cara Admin atau Guru menekan pilihan Bank Soal. Untuk lebih jelasnya, deskripsi *use case* bank soal dapat dilihat pada tabel 4.6.

Tabel 4.6 Deskripsi *Use Case* Mengelola Bank Soal

Nama	Mengelola Bank Soal		
Aktor	Admin dan Guru		
Deskripsi	Aktor dapat menambah, mengubah, melihat, dan menghapus data Soal		
Exception	Pengolahan data tidak dapat dilakukan jika belum login		
Pre condition	Aktor telah dalam keadaan login		
Aktor		Sistem	
Skenario Normal			
1.	Aktor memilih menu Bank Soal		
		2.	Sistem menampilkan halaman Bank Soal dengan daftar soal
3.	Kondisi Jika aktor memilih : Tombol tambah maka S1 berlaku. Tombol edit maka S2 berlaku. Tombol hapus maka S3 berlaku.		
S1 – fungsi tambah			
1.	Aktor memilih tombol Tambah		
		2.	Sistem menampilkan form tambah soal

3.	Aktor menginputkan data soal (Mata Pelajaran, Pertanyaan, Pilihan A, B, C, D, Jawaban Benar, Bobot)		
4.	Aktor dapat menambahkan gambar atau file pendukung jika diperlukan		
5.	Aktor mengklik tombol Simpan		
		6.	Sistem memvalidasi kelengkapan data
		7.	Data valid, sistem menyimpan data soal ke database
		8.	Sistem menampilkan pesan "Soal berhasil disimpan"
S2 – fungsi edit			
1.	Aktor memilih soal yang akan diedit		
2.	Aktor mengklik tombol Edit		
		3.	Sistem menampilkan form edit dengan data soal terpilih
4.	Aktor memodifikasi data soal		
5.	Aktor mengklik tombol Update		
		6.	Sistem memvalidasi kelengkapan data
		7.	Data valid, sistem menyimpan perubahan ke database
		8.	Sistem menampilkan pesan "Soal berhasil diupdate"
S3 – fungsi hapus			
1.	Aktor memilih soal yang akan dihapus		
2.	Aktor mengklik tombol Hapus		
		3.	Sistem menampilkan konfirmasi "Apakah Anda yakin ingin menghapus soal ini?"
4.	Aktor mengklik tombol Ya		
		5.	Sistem menghapus soal dari database
		6.	Sistem menampilkan pesan "soal berhasil dihapus"
Skenario alternative			
Data tambah tidak lengkap			

S1-6a	Saat validasi, data yang diinput belum lengkap
S1-6b	Sistem menampilkan pesan error "Lengkapi semua field yang wajib diisi"
Data edit tidak lengkap	
S2-6a	Saat validasi, data yang dimodifikasi belum lengkap
S2-6b	Sistem menampilkan pesan error dan kembali ke form edit
Data hapus dibatalkan	
S3-3a	Aktor memilih tombol Tidak pada konfirmasi
S3-3b	Sistem membatalkan proses hapus dan kembali ke daftar mata pelajaran
Post condition	Aktor berhasil mengelola bank soal

7. Deskripsi Use Case Guru Membuat dan Mengelola Ujian

Deskripsi *use case* guru membuat ujian ini menjelaskan mengenai bagaimana sistem membuat ujian baru dengan cara guru memilih menu Ujian kemudian menekan pilihan Buat Ujian Baru. Untuk lebih jelasnya, deskripsi *use case* guru membuat ujian dapat dilihat pada tabel 4.7.

Tabel 4.7 Deskripsi Use Case Guru Membuat Ujian

Nama	Membuat dan Mengelola Ujian		
Aktor	Guru		
Deskripsi	Guru membuat ujian baru dan mengatur parameter ujian		
Exception	Tidak dapat membuat ujian jika bank soal kosong		
Pre condition	Guru telah login dan bank soal tersedia		
Aktor		Sistem	
Skenario Normal			
1.	Guru memilih menu Ujian		
		2.	Sistem menampilkan halaman Ujian dengan daftar ujian yang pernah dibuat

3.	Guru mengklik tombol Buat Ujian Baru		
		4.	Sistem menampilkan form pengaturan ujian
5.	Guru menginputkan data ujian: a. Nama Ujian b. Mata Pelajaran c. Kelas d. Jumlah Soal e. Durasi Waktu (menit) f. Tanggal Mulai g. Tanggal Berakhir h. Pengacakan Soal (Ya/Tidak)		
6.	Guru mengklik tombol Simpan		
		7.	Sistem memvalidasi kelengkapan data
		8.	Sistem membuat ujian dan generate token ujian
		9.	Sistem menampilkan pesan "Ujian berhasil dibuat" dan menampilkan token ujian
		10.	Sistem menyimpan data ujian ke database
S1 – fungsi tambah			
1.	Aktor memilih tombol Tambah		
		2.	Sistem menampilkan form tambah soal
3.	Aktor menginputkan data soal (Mata Pelajaran, Pertanyaan, Pilihan A, B, C, D, Jawaban Benar, Bobot)		
4.	Aktor dapat menambahkan gambar atau file pendukung jika diperlukan		
5.	Aktor mengklik tombol Simpan		
		6.	Sistem memvalidasi kelengkapan data
		7.	Data valid, sistem menyimpan data soal ke database
		8.	Sistem menampilkan pesan "Soal berhasil disimpan"
S2 – fungsi edit			
1.	Aktor memilih soal yang akan diedit		
2.	Aktor mengklik tombol Edit		

		3.	Sistem menampilkan form edit dengan data soal terpilih
4.	Aktor memodifikasi data soal		
5.	Aktor mengklik tombol Update		
		6.	Sistem memvalidasi kelengkapan data
		7.	Data valid, sistem menyimpan perubahan ke database
		8.	Sistem menampilkan pesan "Soal berhasil diupdate"
S3 – fungsi hapus			
1.	Aktor memilih soal yang akan dihapus		
2.	Aktor mengklik tombol Hapus		
		3.	Sistem menampilkan konfirmasi "Apakah Anda yakin ingin menghapus soal ini?"
4.	Aktor mengklik tombol Ya		
		5.	Sistem menghapus soal dari database
		6.	Sistem menampilkan pesan "soal berhasil dihapus"
Skenario alternative			
Data tidak lengkap			
7.a	Data tidak lengkap		
7.b	Sistem menampilkan pesan error "Lengkapi semua field yang diperlukan"		
7.c	Sistem kembali ke form pengaturan ujian		
Soal melebihi ketersediaan			
8.a	Jumlah soal yang diminta melebihi soal yang tersedia di bank soal		
8.b	Sistem menampilkan pesan "Jumlah soal tidak mencukupi, tersedia hanya X soal"		
Post condition		Ujian berhasil dibuat dan siap diikuti oleh siswa	

8. Deskripsi *Use Case* Siswa Mengerjakan Ujian

Deskripsi *use case* mengerjakan ujian ini menjelaskan mengenai bagaimana sistem menampilkan soal-soal ujian berdasarkan mata pelajaran yang

sedang dilaksanakan di Bimbel Quadrant dengan cara siswa memilih menu Ujian. Untuk lebih jelasnya, deskripsi *use case* mengerjakan ujian dapat dilihat pada tabel 4.8.

Tabel 4.8 Deskripsi Use Case Mengerjakan Ujian

Nama		Mengerjakan Ujian	
Aktor		Siswa	
Deskripsi		Siswa mengerjakan soal ujian online	
Exception		Token salah, Ujian belum dibuka, Ujian sudah ditutup	
Pre condition		Siswa telah login dan memiliki token ujian	
Aktor		Sistem	
Skenario Normal			
1.	Siswa memilih menu Ujian		
		2.	Sistem menampilkan halaman daftar ujian yang tersedia
3.	Siswa mengklik ujian yang akan diikuti		
		4.	Sistem menampilkan form input token ujian
5.	Siswa memasukkan token ujian		
6.	Siswa mengklik tombol Mulai Ujian		
		7.	Sistem memverifikasi token
		8.	Token valid, sistem menampilkan halaman konfirmasi
		9.	Sistem menampilkan informasi ujian (nama ujian, jumlah soal, durasi)
		10.	Sistem menyimpan data ujian ke database
11.	Siswa mengklik tombol Mulai Mengerjakan		
		12.	Sistem menampilkan soal ujian dengan timer berjalan
		13.	Sistem mengacak urutan soal jika pengacakan diaktifkan

14.	Siswa membaca soal dan memilih jawaban		
15.	Siswa mengklik tombol Selanjutnya untuk soal berikutnya		
16.	Siswa mengulang langkah 13-14 hingga semua soal terjawab		
17.	Siswa mengklik tombol Selesai		
		18.	Sistem menampilkan konfirmasi "Yakin ingin mengakhiri ujian?"
19.	Siswa mengklik tombol Ya		
		20.	Sistem menghentikan timer
		21	Sistem menghitung nilai secara otomatis
		22	Sistem menyimpan hasil ujian ke database
		23	Sistem menampilkan nilai dan peringkat siswa
Skenario alternative			
Data tidak valid			
7.a	Token tidak valid		
7.b	Sistem menampilkan pesan "Token salah, silakan coba lagi"		
7.c	Sistem kembali ke form input token		
Timer waktu telah habis			
11.a	Timer habis sebelum siswa menyelesaikan ujian		
11.b	Sistem otomatis mengakhiri ujian		
11.c	Sistem menghitung nilai berdasarkan jawaban yang sudah dijawab		
Siswa Membatalkan penyelesaian			
17.a	Siswa mengklik tombol Tidak		
17.b	Sistem membatalkan proses selesai dan kembali ke halaman ujian		
Post condition		Siswa berhasil mengerjakan ujian dan nilai tersimpan	

9. Deskripsi Use Case Melihat Hasil Ujian

Deskripsi use case melihat hasil ujian ini menjelaskan mengenai bagaimana sistem menampilkan nilai hasil ujian dengan cara admin, guru, atau siswa

memilih menu Hasil Ujian. Untuk lebih jelasnya, deskripsi *use case* melihat hasil ujian dapat dilihat pada tabel 4.9.

Tabel 4.9 Deskripsi Use Case Melihat Hasil Ujian

Nama	Melihat Hasil Ujian		
Aktor	Admin, Guru, Siswa		
Deskripsi	Aktor melihat hasil ujian dan analisis nilai		
Exception	-		
Pre condition	Aktor telah login dan ujian telah dilaksanakan		
Aktor		Sistem	
Skenario Normal			
1.	Aktor memilih menu Hasil Ujian		
		2.	Sistem menampilkan daftar ujian yang telah dilaksanakan
3.	Aktor memilih ujian yang ingin dilihat hasilnya		
		4.	Sistem menampilkan hasil ujian: a. Daftar siswa yang mengikuti ujian (untuk Admin & Guru) b. Nilai masing-masing siswa c. Peringkat siswa d. Nilai tertinggi, terendah, dan rata-rata (untuk Admin & Guru) e. Detail jawaban siswa (untuk Siswa: hanya miliknya sendiri)
5.	Aktor dapat mengklik nama siswa untuk melihat detail		
		6.	Sistem menampilkan detail jawaban siswa: a. Soal yang dijawab benar (warna hijau) b. Soal yang dijawab salah (warna merah) c. Kunci jawaban yang benar

7.	Aktor dapat mengklik tombol Export untuk mencetak laporan (Admin & Guru)		
		8.	Sistem generate laporan dalam format PDF
		9.	Sistem menampilkan informasi ujian (nama ujian, jumlah soal, durasi)
		10.	Sistem menyimpan data ujian ke database
11.	Siswa mengklik tombol Mulai Mengerjakan		
		12.	Sistem menampilkan soal ujian dengan timer berjalan
		13.	Sistem mengacak urutan soal jika pengacakan diaktifkan
14.	Siswa membaca soal dan memilih jawaban		
15.	Siswa mengklik tombol Selanjutnya untuk soal berikutnya		
16.	Siswa mengulang langkah 13-14 hingga semua soal terjawab		
17.	Siswa mengklik tombol Selesai		
		18.	Sistem menampilkan konfirmasi "Yakin ingin mengakhiri ujian?"
19.	Siswa mengklik tombol Ya		
		20.	Sistem menghentikan timer
		21.	Sistem menghitung nilai secara otomatis
		22.	Sistem menyimpan hasil ujian ke database
		23.	Sistem menampilkan nilai dan peringkat siswa
Post condition		Aktor berhasil melihat hasil ujian	

10. Deskripsi *Use Case* Ganti Password

Deskripsi *use case* ganti password ini menjelaskan mengenai bagaimana sistem mengubah password dengan cara aktor memilih menu Ganti

Password. Untuk lebih jelasnya, deskripsi *use case* ganti password dapat dilihat pada tabel 4.10.

Tabel 4.10 Deskripsi Use Case Ganti Password

Nama		Ganti Password	
Aktor		Admin, Guru, Siswa	
Deskripsi		Aktor mengubah password akun	
Exception		Password lama tidak sesuai	
Pre condition		Aktor telah login	
Aktor		Sistem	
Skenario Normal			
1.	Aktor memilih menu Ganti Password		
		2.	Sistem menampilkan form ganti password
3.	Aktor menginputkan: a. Password Lama b. Password Baru c. Konfirmasi Password Baru		
4.	Aktor mengklik tombol Simpan		
		5.	Sistem memvalidasi password lama
		6.	Sistem memvalidasi kecocokan password baru dengan konfirmasi
		7.	Validasi berhasil, sistem mengenkripsi password baru
		8.	Sistem menyimpan password baru ke database
		9.	Sistem menampilkan pesan "Password berhasil diubah"
Skenario alternative			
Password lama tidak sesuai			
5.a	Password lama tidak sesuai		
5.b	Sistem menampilkan pesan "Password lama salah"		
Password baru tidak sesuai			

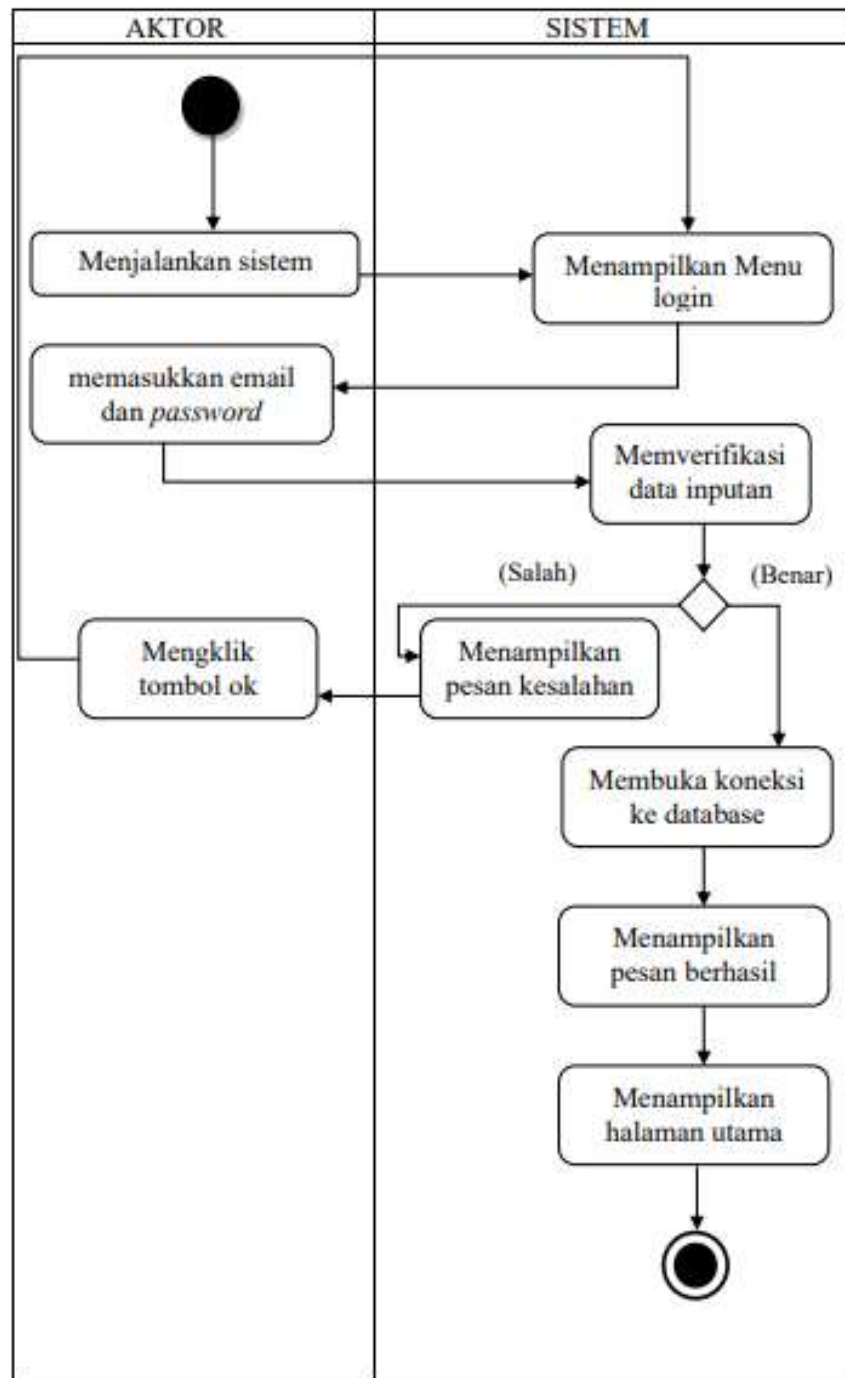
6.a	Password baru dan konfirmasi tidak cocok		
6.b	Sistem menampilkan pesan "Password baru dan konfirmasi tidak cocok"		
6.c	Sistem kembali ke form ganti password		
Post condition		Password berhasil diubah	

4.3.3 Activity Diagram

Activity diagram merupakan gambar proses bisnis dan urutan aktivitas dalam sebuah proses. Diagram ini menggambarkan aliran kerja dari satu aktivitas ke aktivitas lainnya dalam sistem. Berikut ini adalah *activity* diagram aplikasi ujian online Bimbel Quadrant.

1. Activity Diagram Menu Utama Admin, Guru dan Siswa

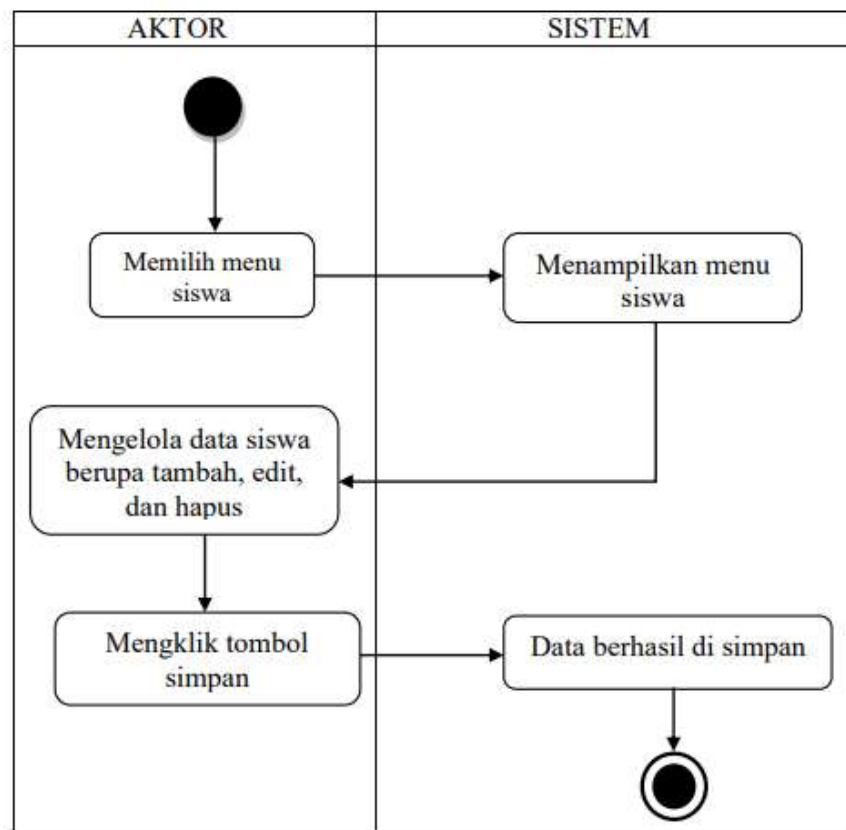
Activity diagram login menggambarkan aktivitas antara aktor (Admin, Guru, Siswa) dengan sistem pada saat melakukan login. Pada diagram ini dijelaskan bahwa aktor menjalankan sistem, kemudian sistem menampilkan halaman *login*. Aktor memasukkan *username* dan *password*, kemudian sistem memverifikasi data inputan. Jika data salah, sistem menampilkan pesan kesalahan dan kembali ke halaman login. Jika data benar, sistem menampilkan halaman utama sesuai dengan role pengguna. Untuk lebih jelasnya dapat dilihat pada Gambar 4.5.



Gambar 4.6 *Activity* Diagram Login Admin, Guru

2. Activity Diagram Admin Mengelola Data Siswa

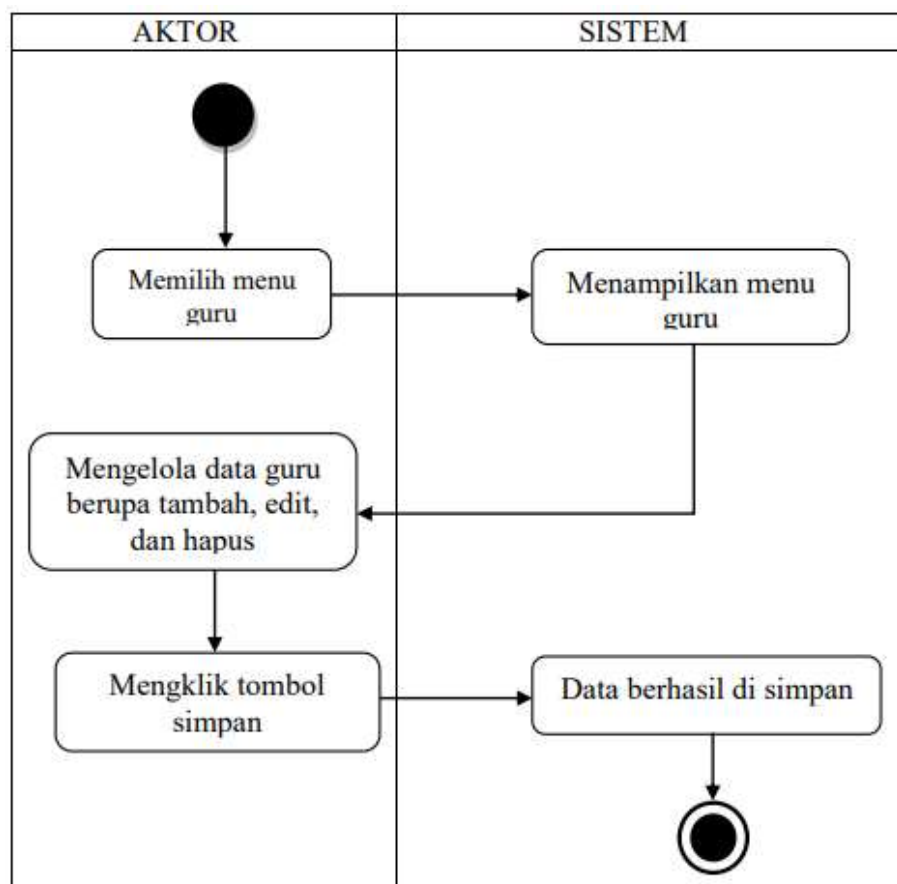
Activity diagram mengelola data siswa menggambarkan aktivitas antara admin dan sistem pada saat memilih menu data siswa yang tersedia pada aplikasi. Pada diagram ini dijelaskan bahwa *admin* menjalankan program dan memilih menu data siswa pada menu utama, kemudian *admin* akan mengelola data siswa yang berupa tambah, edit, dan hapus yang ada pada menu data siswa dan sistem akan menampilkan menu untuk mengelola data siswa yang akan diolah oleh *admin*. Untuk lebih jelasnya dapat dilihat pada Gambar 4.6.



Gambar 4.7 Activity Diagram Mengelola Data Siswa

3. *Activity* Diagram Admin Mengelola Data Guru

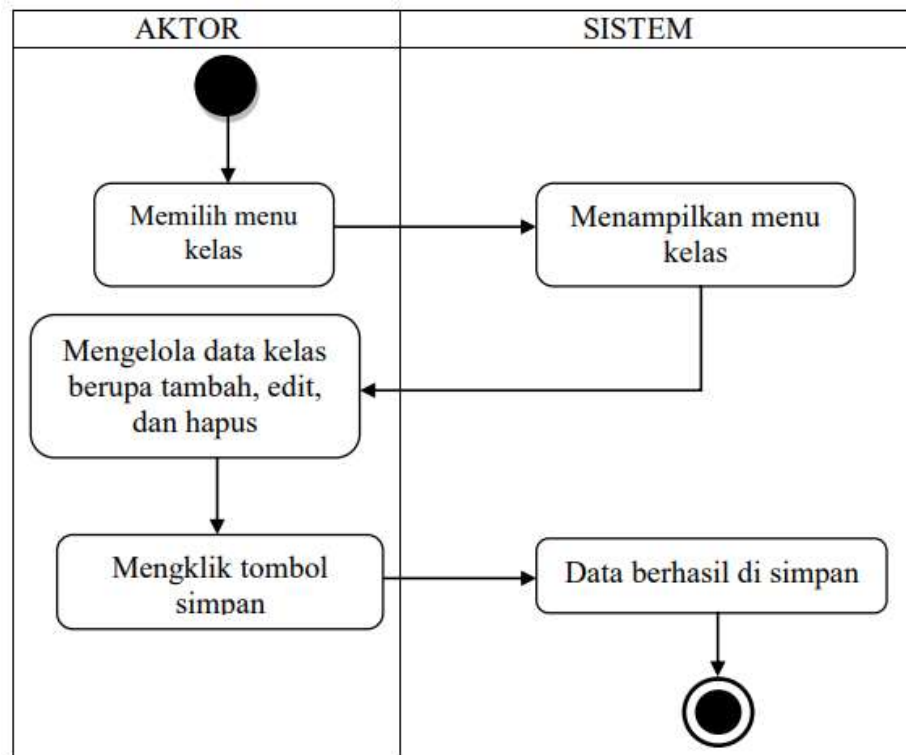
Activity diagram mengelola data guru menggambarkan aktivitas antara admin dan sistem pada saat memilih menu data guru yang tersedia pada aplikasi. Pada diagram ini dijelaskan bahwa admin menjalankan program dan memilih menu data guru pada menu utama, kemudian *admin* akan mengelola data guru yang berupa tambah, edit, dan hapus yang ada pada menu data guru dan sistem akan menampilkan menu untuk mengelola data guru yang akan diolah oleh *admin*. Untuk lebih jelasnya dapat dilihat pada Gambar 4.7.



Gambar 4.8 *Activity* Diagram Mengelola Data Guru

4. *Activity* Diagram Admin Mengelola Data Kelas

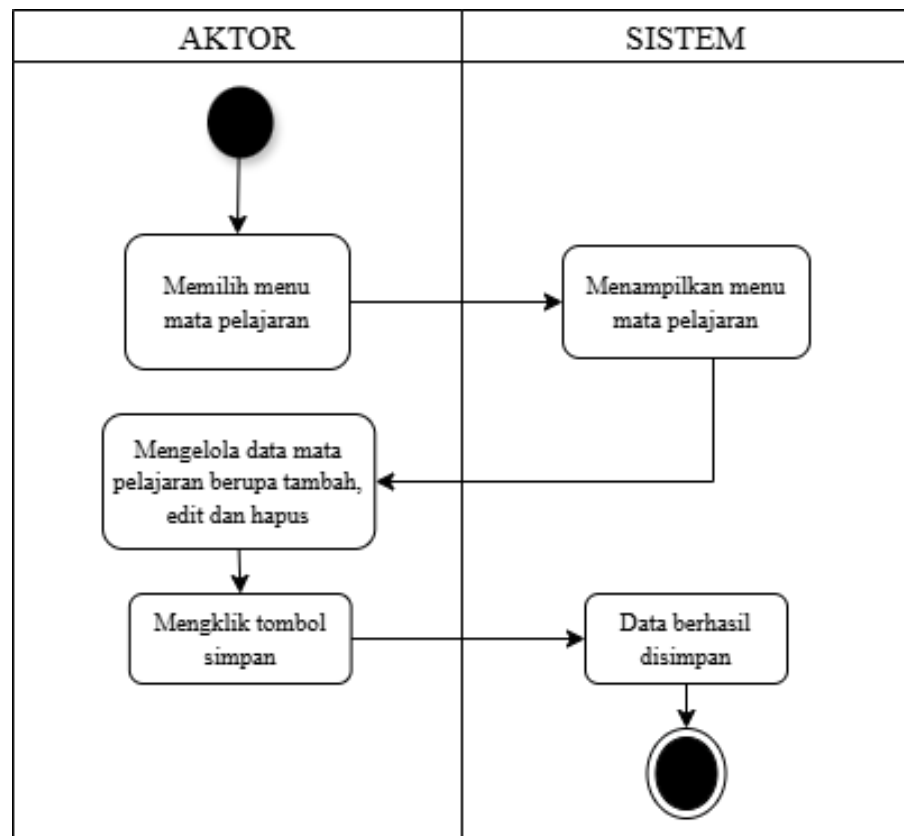
Activity diagram mengelola data kelas menggambarkan aktivitas antara admin dan sistem pada saat memilih menu data kelas yang tersedia pada aplikasi. Pada diagram ini dijelaskan bahwa admin menjalankan program dan memilih menu data kelas pada menu utama, kemudian admin akan mengelola data kelas yang berupa tambah, edit, dan hapus yang ada pada menu data kelas dan sistem akan menampilkan menu untuk mengelola data kelas yang akan diolah oleh admin. Untuk lebih jelasnya dapat dilihat pada Gambar 4.8.



Gambar 4.9 *Activity* Diagram Mengelola Data Kelas

5. *Activity* Diagram Admin Mengelola Mata Pelajaran

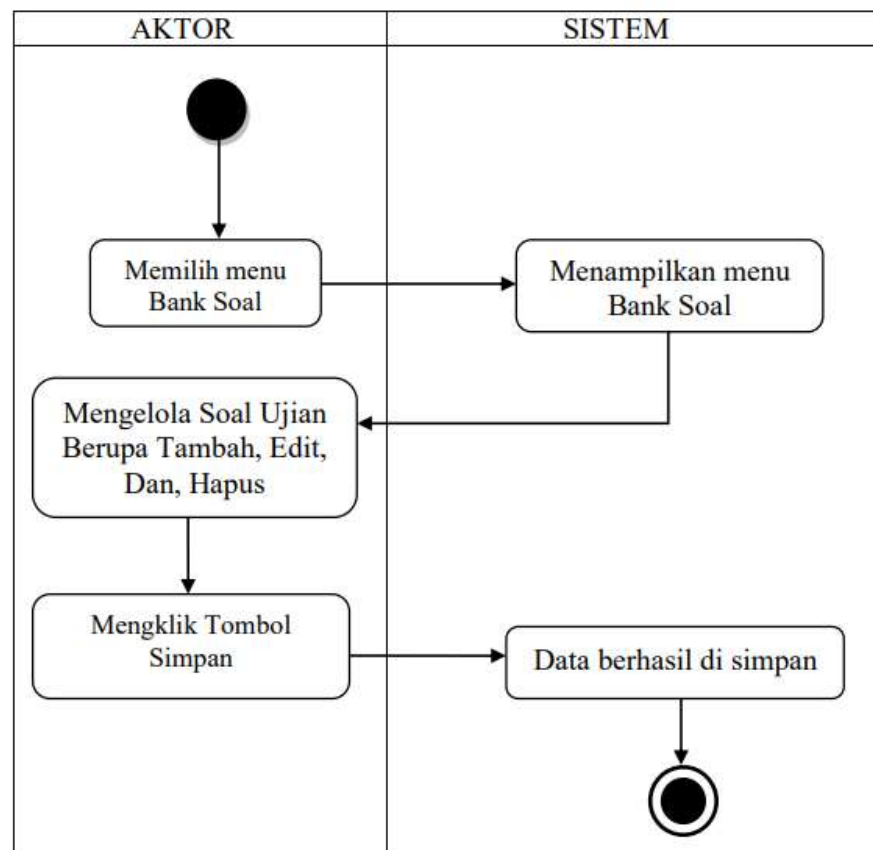
Activity diagram mengelola mata pelajaran menggambarkan aktivitas antara admin dan sistem pada saat memilih menu mata pelajaran yang tersedia pada aplikasi. Pada diagram ini dijelaskan bahwa admin menjalankan program dan memilih menu mata pelajaran pada menu utama, kemudian admin akan mengelola data mata pelajaran yang berupa tambah, edit, dan hapus yang ada pada menu mata pelajaran. Untuk lebih jelasnya dapat dilihat pada Gambar 4.9.



Gambar 4.10 *Activity* Diagram Mengelola Mata Pelajaran

6. *Activity Diagram Admin dan Guru Mengelola Bank Soal*

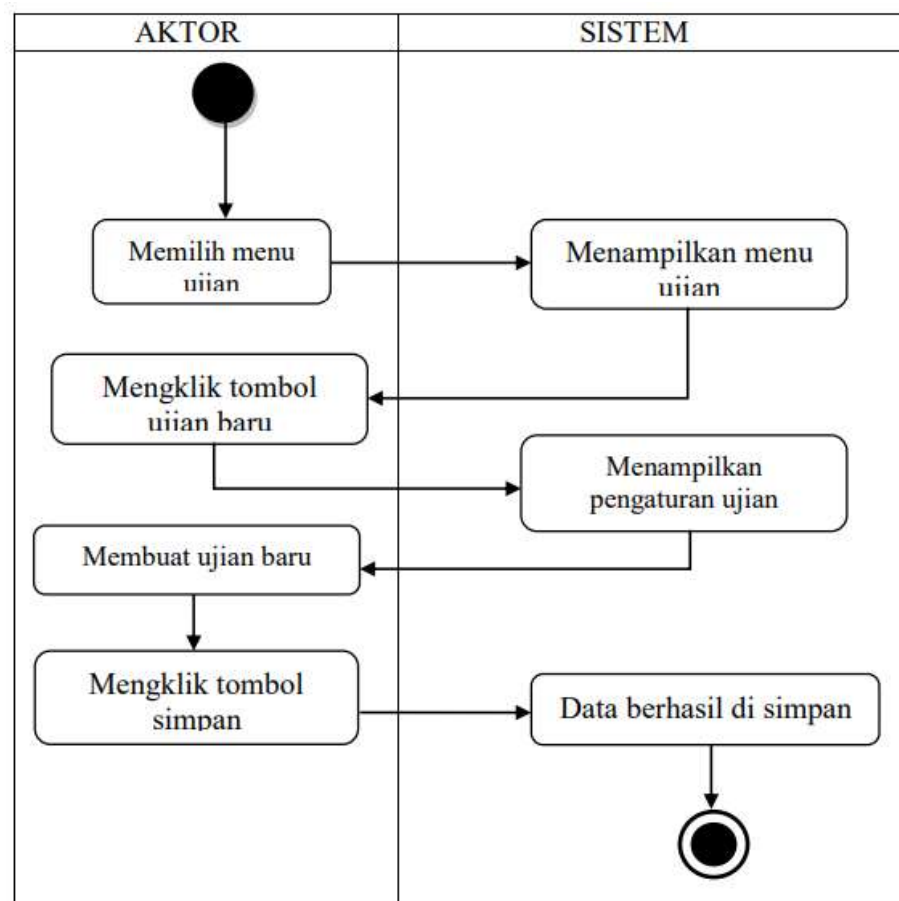
Activity diagram mengelola bank soal menggambarkan aktivitas antara admin atau guru dan sistem pada saat memilih menu bank soal yang tersedia pada aplikasi. Pada diagram ini dijelaskan bahwa admin atau guru menjalankan program dan memilih menu bank soal pada menu utama, kemudian admin atau guru akan mengelola soal ujian yang ada pada menu bank soal. Untuk lebih jelasnya dapat dilihat pada Gambar 4.10.



Gambar 4.11 *Activity Diagram* Mengelola Bank Soal

7. Activity Diagram Guru Membuat Ujian

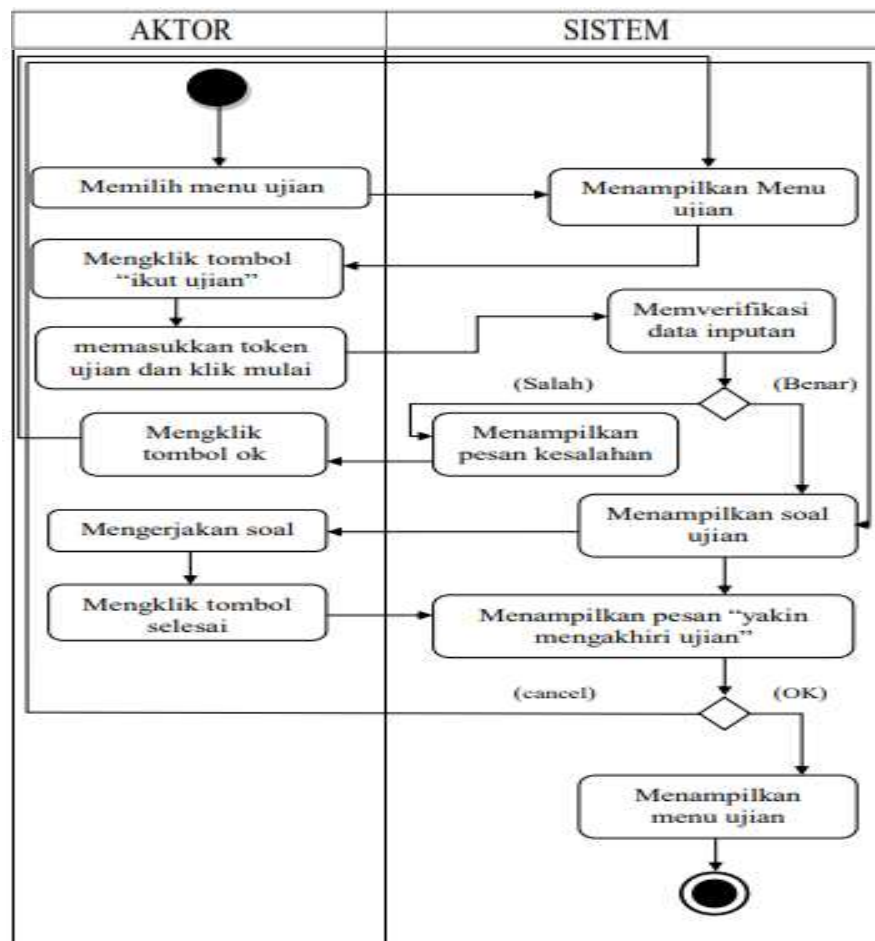
Activity diagram guru membuat ujian menggambarkan aktivitas guru dan sistem pada saat memilih menu membuat ujian yang tersedia pada aplikasi. Pada diagram ini dijelaskan bahwa guru menjalankan program dan memilih menu ujian pada menu utama, kemudian guru akan membuat ujian baru yang ada pada menu ujian dan sistem akan menampilkan pengaturan ujian dari menu ujian yang dipilih oleh guru. Untuk lebih jelasnya dapat dilihat pada Gambar 4.11.



Gambar 4.12 Activity Diagram Guru Membuat Ujian

8. *Activity* Diagram Siswa Mengerjakan Ujian

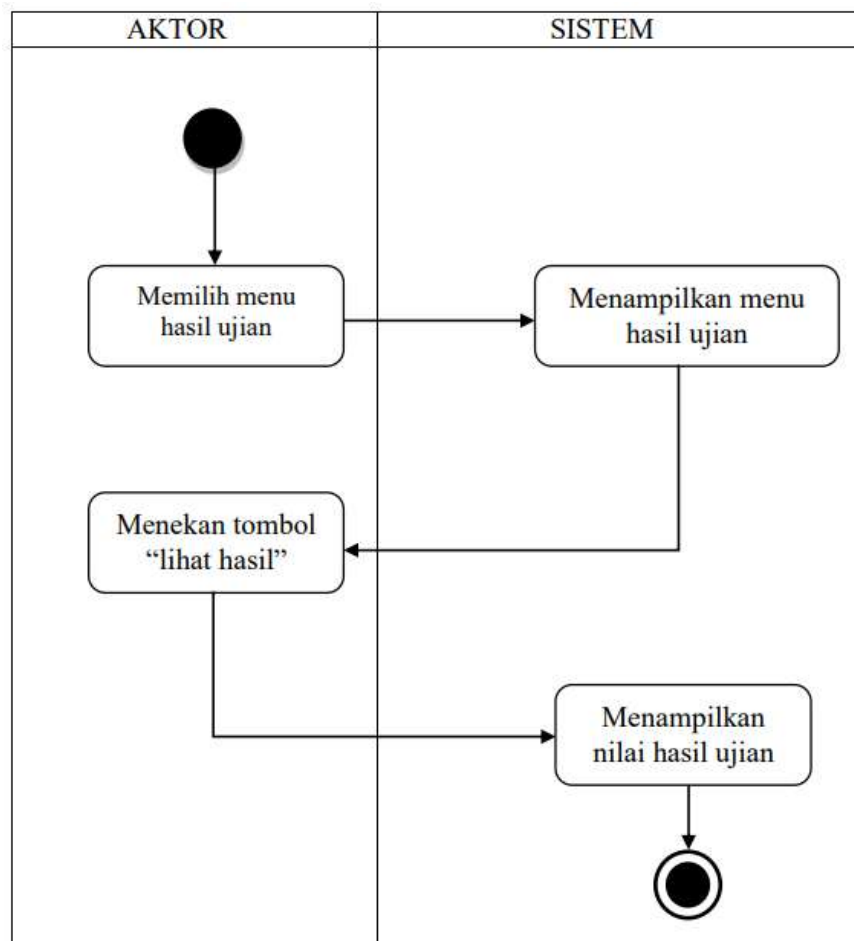
Activity diagram mengerjakan ujian menggambarkan aktivitas antara siswa dan sistem pada saat memilih menu ujian yang tersedia pada aplikasi. Pada diagram ini dijelaskan bahwa siswa menjalankan program dan memilih menu ujian pada menu utama, kemudian siswa akan memilih menu ujian yang ada pada menu utama dan sistem akan menampilkan pilihan soal ujian dari menu ujian yang dipilih oleh siswa. Untuk lebih jelasnya dapat dilihat pada Gambar 4.12.



Gambar 4.13 *Activity* Diagram Mengerjakan Ujian

9. *Activity* Diagram Melihat Hasil Ujian

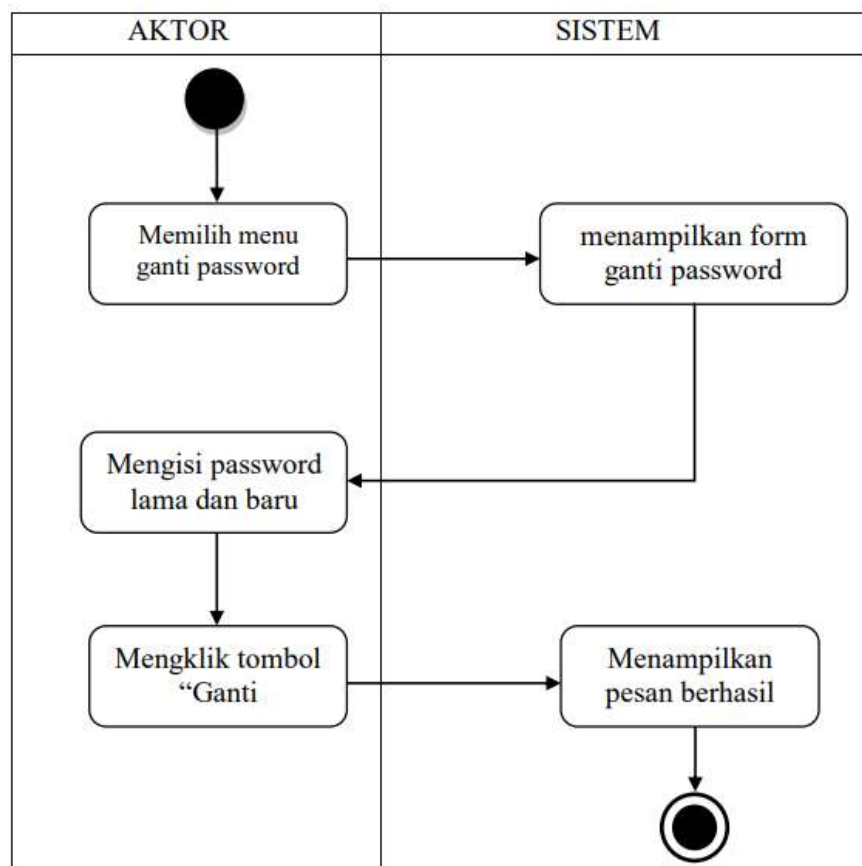
Activity diagram melihat hasil ujian menggambarkan aktivitas antara admin, guru, atau siswa dan sistem pada saat memilih menu hasil ujian yang tersedia pada aplikasi. Pada diagram ini dijelaskan bahwa aktor menjalankan program dan memilih menu hasil ujian pada menu utama, kemudian sistem akan menampilkan nilai ujian siswa beserta analisis hasilnya. Untuk lebih jelasnya dapat dilihat pada Gambar 4.13.



Gambar 4.14 *Activity* Diagram Melihat Hasil Ujian

10. Activity Diagram Mengganti Password

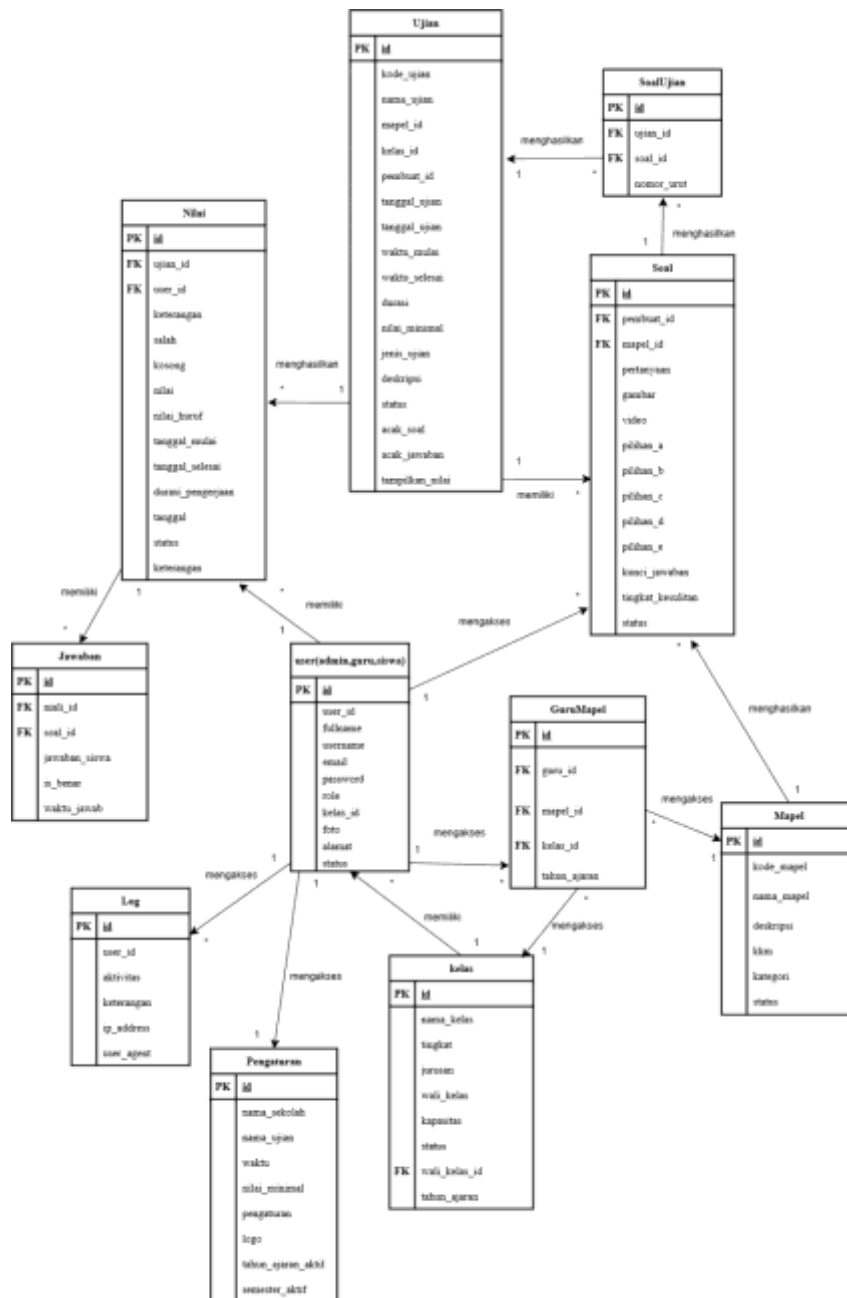
Activity diagram mengganti password menggambarkan aktivitas antara user (Admin, Guru, Siswa) dan sistem pada saat mengganti password yang tersedia pada aplikasi. Pada diagram ini dijelaskan bahwa user menjalankan program dan memilih menu ganti password pada menu utama, kemudian user akan memilih menu ganti password yang ada dan sistem akan menampilkan form ganti password. User mengisi password lama dan password baru kemudian menekan tombol 'Ganti Password'. Untuk lebih jelasnya dapat dilihat pada Gambar 4.15.



Gambar 4.15 Activity Diagram Mengganti Password

4.4 ANALISIS KEBUTUHAN DATA

Untuk menggambarkan kebutuhan data untuk perangkat lunak yang dihasilkan penulis menggunakan alat bantu class diagram seperti yang terlihat pada gambar dibawah ini:



Gambar 4.16 Class Diagram

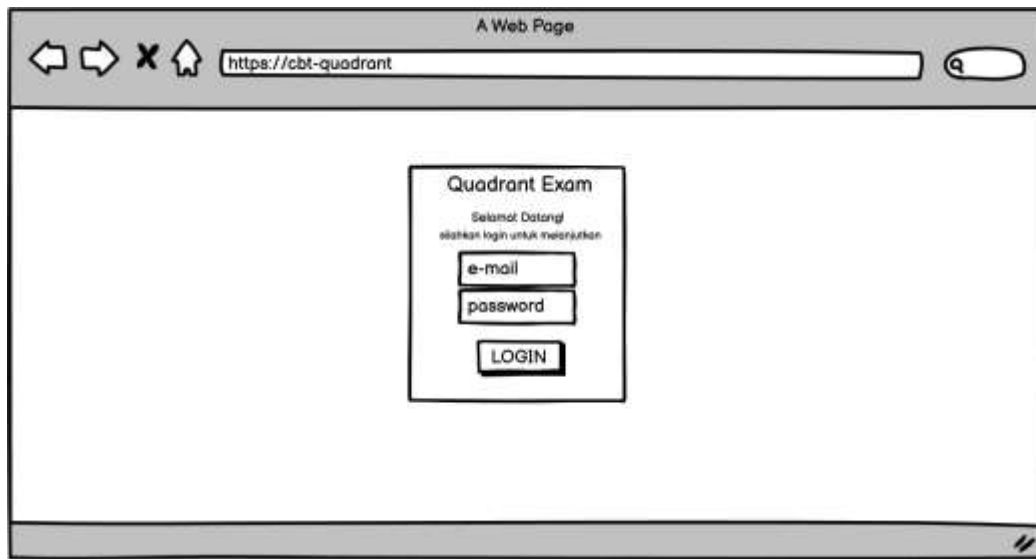
4.5 PERANCANGAN PERANGKAT LUNAK

4.5.1 Rancangan Input

Rancangan input merupakan desain antarmuka yang digunakan oleh pengguna untuk memasukkan data ke dalam sistem. Tampilan ini umumnya tersedia pada halaman admin, guru, maupun siswa sesuai dengan kebutuhan masing-masing pengguna. Adanya rancangan input yang baik akan mempermudah proses pengelolaan data dan membantu pengguna dalam mengoperasikan sistem secara efektif.

1. Rancangan Input Halaman Login

Halaman *login* merupakan antarmuka pertama yang ditampilkan ketika pengguna hendak mengakses sistem aplikasi ujian online. Pada halaman ini, pengguna dapat masuk ke dalam sistem sesuai dengan peran masing-masing, baik sebagai admin, guru, maupun siswa. Proses autentikasi dilakukan dengan cara memasukkan username dan password pada kolom yang telah disediakan, kemudian menekan tombol "*Login*" untuk memvalidasi data dan masuk ke halaman utama sistem. Adapun rancangan tampilan halaman login dapat dilihat pada Gambar 4.17.



Gambar 4.17 Rancangan *Input* Halaman *Login*

2. Rancangan Input Tambah Data Siswa

Rancangan *input tambah data siswa* berfungsi sebagai antarmuka untuk menginputkan informasi siswa baru ke dalam sistem. Melalui halaman ini, admin dapat memasukkan data-data siswa seperti NIS, nama lengkap, kelas, username, dan password. Data yang telah dimasukkan akan tersimpan dalam database dan dapat digunakan untuk keperluan autentikasi serta manajemen data siswa. Rancangan tampilan halaman input tambah data siswa ditunjukkan pada Gambar 4.18.

The screenshot shows a web browser window with the address bar displaying 'https://cbit-quadrant'. The page title is 'A Web Page'. The main content area is titled 'QUADRANT EXAM' and shows the user 'Admin, Administrator'. On the left is a sidebar menu with options: Dashboard, Kelola Data Master, Bank Soal, Kelola Ujian, Hasil Ujian, Pengaturan, and Ganti Password. The main content area is titled 'Tambah Siswa Baru' and contains a form with two columns of input fields. At the bottom of the form are two buttons: 'Simpan Data Siswa' and 'Batal'. A link '< Kembali' is also visible in the top right corner of the main content area.

Gambar 4.18 Rancangan *Input* Tambah Data Siswa

3. Rancangan Input Tambah Data Guru

Rancangan *input tambah data guru* dirancang untuk memfasilitasi proses penambahan informasi guru ke dalam sistem. Admin dapat menggunakan halaman ini untuk memasukkan data guru meliputi NIP, nama lengkap, mata pelajaran yang diampu, username, dan password. Informasi tersebut akan digunakan untuk keperluan login guru serta pengelolaan data pengajar di Bimbingan Belajar Quadrant. Visualisasi rancangan input tambah data guru dapat diamati pada Gambar 4.19.

The image shows a web browser window with the address bar displaying 'https://cbit-quadrant'. The page title is 'QUADRANT EXAM' and the user is logged in as 'Admin, Administrator'. On the left is a sidebar menu with the following items: Dashboard, Kelola Data Master, Bank Soal, Kelola Ujian, Hasil Ujian, Pengaturan, and Ganti Password. The main content area is titled 'Tambah Guru Baru' and includes a '< Kembali' link. The form consists of two columns of input fields. The first column has four text input fields, and the second column has four text input fields. At the bottom of the form, there are two buttons: 'Simpan Data Guru' (Save Teacher Data) and 'Batal' (Cancel).

Gambar 4.19 Rancangan *Input* Tambah Data Guru

4. Rancangan Input Tambah Data Kelas

Rancangan *input tambah data kelas* digunakan untuk memasukkan informasi kelas baru yang tersedia di Bimbingan Belajar Quadrant. Melalui form ini, admin dapat menambahkan data kelas beserta keterangan yang diperlukan. Data kelas yang telah diinputkan akan digunakan sebagai referensi dalam pengelompokan siswa serta pengaturan jadwal ujian. Rancangan tampilan halaman input tambah data kelas dapat dilihat pada Gambar 4.20.

The image shows a web application interface for 'QUADRANT EXAM'. A modal window titled 'Tambah Kelas' is open, allowing users to add new classes. The modal contains two text input fields and two buttons: 'Batal' (Cancel) and 'Simpan' (Save). In the background, the main interface is visible, featuring a sidebar menu with the following items: Dashboard, Kelola Data Master, Bank Soal, Kelola Ujian, Hasil Ujian, Pengaturan, and Ganti Password. The main content area displays a table with the following headers: No, Tingkat, Nama Kelas, and Aksi. The user is logged in as 'Admin, Administrator', and the URL in the browser is 'https://cbt-quadrant'.

Gambar 4.20 Rancangan *Input* Tambah Data Kelas

5. Rancangan Input Tambah Data Mata Pelajaran

Rancangan *form tambah data mata pelajaran* berfungsi untuk menginputkan informasi mata pelajaran yang diajarkan di Bimbingan Belajar Quadrant. Admin dapat memasukkan kode mata pelajaran beserta nama mata pelajaran melalui form yang telah disediakan. Data mata pelajaran ini akan menjadi acuan dalam pembuatan soal ujian dan pengelompokan bank soal berdasarkan mata pelajaran tertentu. Rancangan tampilan halaman input tambah data mata pelajaran ditampilkan pada Gambar 4.21.

The screenshot displays the 'QUADRANT EXAM' web application. At the top, there is a browser address bar showing 'https://cbit-quadrant'. Below the address bar, the page title 'QUADRANT EXAM' is visible. On the left side, there is a sidebar menu with the following items: Dashboard, Kelola Data Master, Bank Soal, Kelola Ujian, Hasil Ujian, Pengaturan, and Ganti Password. The main content area is titled 'Data Mata Pelajaran' with the subtitle 'Kelola Data Mapel'. In the top right corner, the user is identified as 'Admin, Administrator'. A modal form titled 'Tambah Mapel' is open in the center, containing two input fields and two buttons: 'Batal' and 'Simpan'. To the right of the modal, there is a button labeled '+ Tambah Mapel'. Below the modal, there is a table with the following headers: 'No', 'Kode', 'Mata Pelajaran', 'Jumlah Soal', 'Jumlah Ujian', and 'Aksi'. The table body is currently empty.

Gambar 4.21 Rancangan *Input* Tambah Data Mata Pelajaran

6. Rancangan Input Tambah Data Soal

Rancangan *form tambah data soal* merupakan antarmuka yang digunakan untuk memasukkan soal-soal ujian ke dalam bank soal. Guru atau admin dapat mengisi form ini dengan data soal yang meliputi pemilihan mata pelajaran, penulisan pertanyaan, pengisian pilihan jawaban (A, B, C, D, E), penentuan jawaban yang benar, serta pengaturan bobot nilai. Sistem juga menyediakan fitur untuk mengunggah gambar atau file pendukung apabila soal memerlukan media tambahan. Rancangan tampilan halaman input tambah data soal dapat diamati pada Gambar 4.22.

The image shows a web browser window with the address bar displaying 'https://cbit-quadrant'. The page title is 'QUADRANT EXAM'. The user is logged in as 'Admin, Administrator'. On the left is a sidebar menu with the following items: Dashboard, Kelola Data Master, Bank Soal, Kelola Ujian, Hasil Ujian, Pengaturan, and Ganti Password. The main content area is titled 'Tambah Soal' and contains a form with the following elements: a text input field, a dropdown menu labeled 'Mapel', two more text input fields, and two buttons at the bottom: 'Simpan Soal' (Save Question) and 'Batal' (Cancel). A 'Kembali' (Back) link is also present in the top right of the main area.

Gambar 4.22 Rancangan *Input* Tambah Data Soal

7. Rancangan Input Buat Ujian

Rancangan *form buat ujian* dirancang untuk memfasilitasi guru dalam membuat dan mengatur ujian baru. Melalui form ini, guru dapat mengonfigurasi berbagai parameter ujian seperti nama ujian, mata pelajaran yang diujikan, kelas peserta, jumlah soal yang akan ditampilkan, durasi waktu pengerjaan, jadwal pelaksanaan ujian (tanggal mulai dan berakhir), serta opsi pengacakan urutan soal. Setelah ujian dibuat, sistem akan menghasilkan token unik yang digunakan siswa untuk mengakses ujian tersebut. Rancangan tampilan halaman input buat ujian ditunjukkan pada Gambar 4.23.

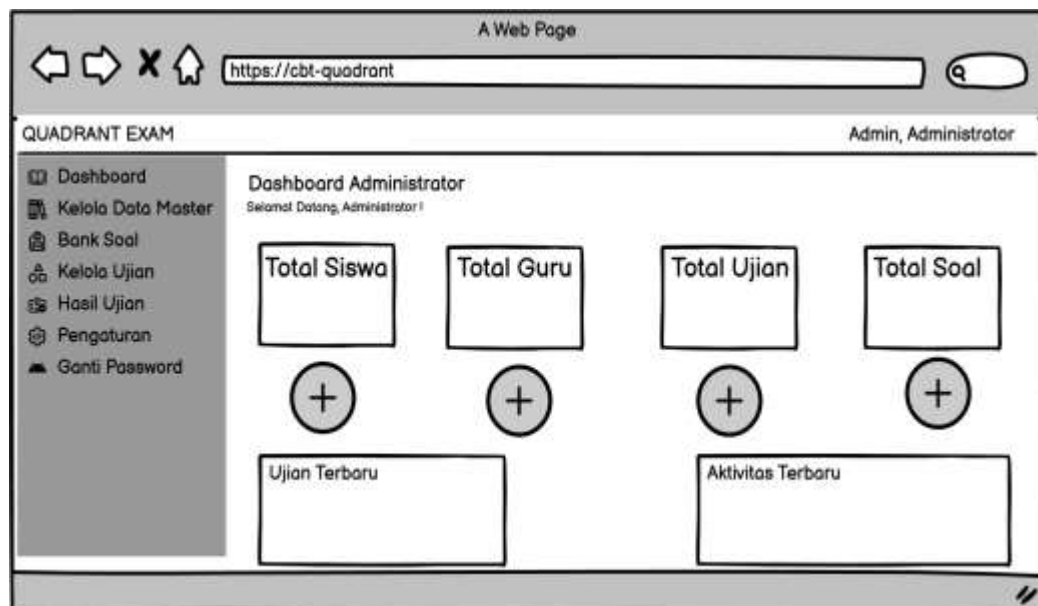
Gambar 4.23 Rancangan *Input* Buat Ujian

4.5.2 Rancangan Output

Rancangan output merupakan desain tampilan yang menampilkan hasil pengolahan data dari sistem kepada pengguna. Berikut ini adalah rancangan-rancangan tampilan output dari sistem aplikasi ujian online yang akan dibangun:

1. Rancangan Output Halaman Dashboard

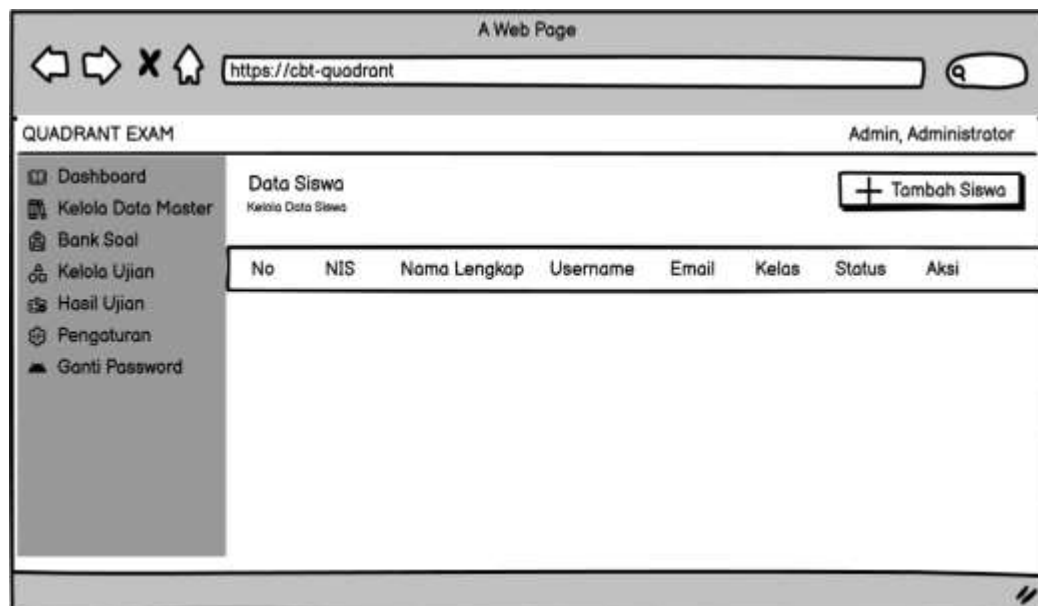
Halaman dashboard merupakan tampilan utama yang akan muncul setelah pengguna berhasil melakukan proses login ke dalam sistem. Halaman ini menyajikan ringkasan informasi penting yang relevan dengan masing-masing peran pengguna. Bagi admin, dashboard menampilkan statistik keseluruhan sistem seperti jumlah siswa, guru, dan ujian yang terdaftar. Sementara bagi guru dan siswa, dashboard menampilkan informasi yang disesuaikan dengan kebutuhan mereka masing-masing. Visualisasi rancangan output halaman dashboard dapat dilihat pada Gambar 4.24.



Gambar 4.24 Rancangan *Output* Halaman Dashboard

2. Rancangan Output Data Siswa

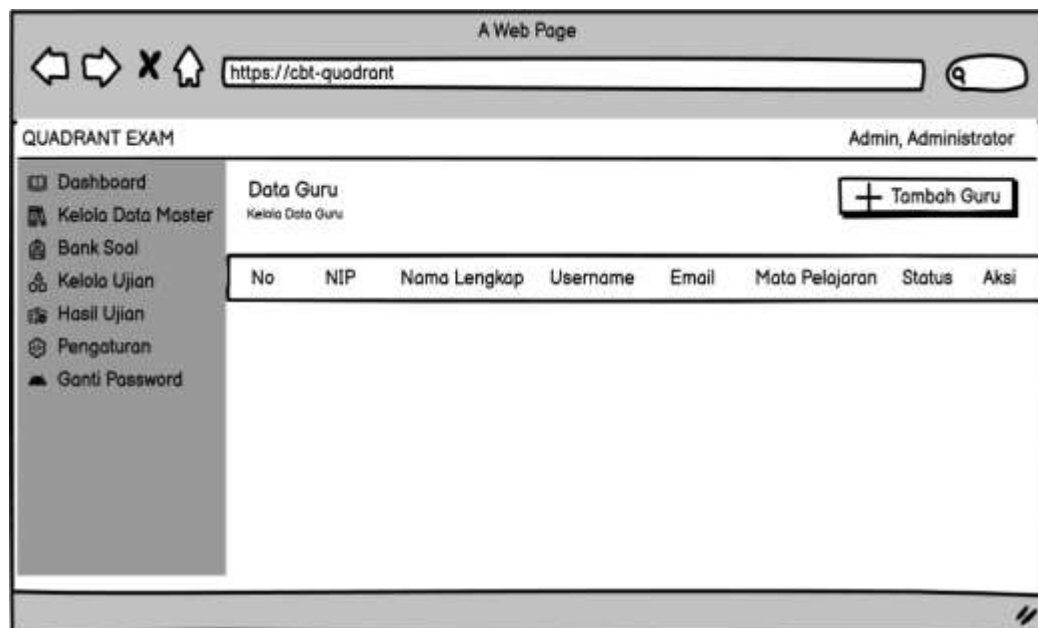
Halaman output data siswa akan ditampilkan ketika admin mengakses menu data siswa pada sistem. Halaman ini menyajikan daftar seluruh siswa yang terdaftar dalam bentuk tabel yang memuat informasi seperti nomor urut, NIS, nama siswa, kelas, dan username. Selain itu, tersedia juga tombol aksi untuk melakukan pengelolaan data seperti edit, hapus, dan reset password. Rancangan tampilan output data siswa dapat diamati pada Gambar 4.25.



Gambar 4.25 Rancangan *Output* Data Siswa

3. Rancangan Output Data Guru

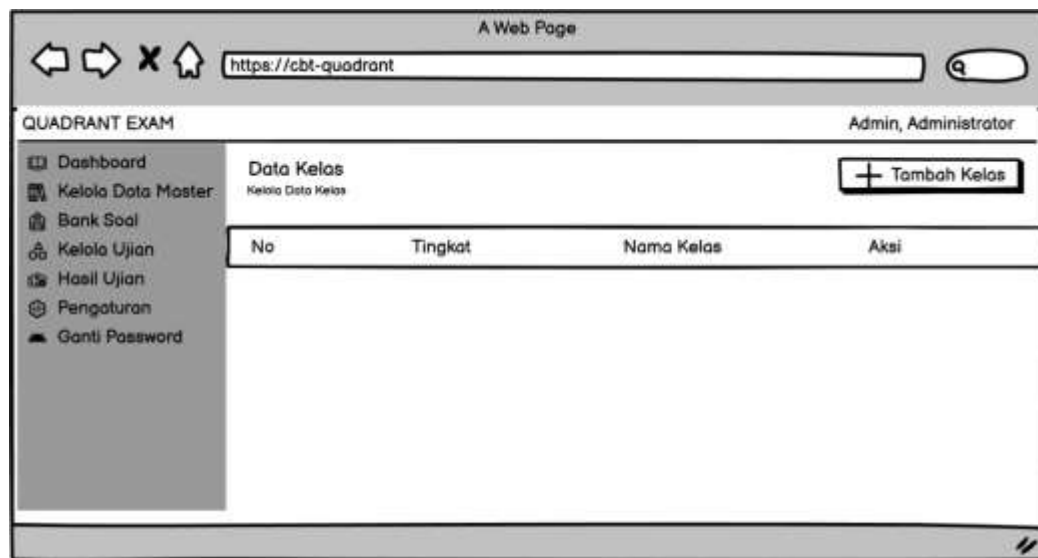
Halaman output data guru akan tampil saat admin membuka menu pengelolaan data guru. Tampilan ini menampilkan daftar seluruh guru yang terdaftar dalam sistem beserta informasi terkait seperti NIP, nama guru, mata pelajaran yang diampu, dan username. Admin dapat melakukan berbagai aksi pengelolaan data melalui tombol yang tersedia pada setiap baris data guru. Rancangan tampilan output data guru ditunjukkan pada Gambar 4.26.



Gambar 4.26 Rancangan *Output* Data Guru

4. Rancangan Output Data Kelas

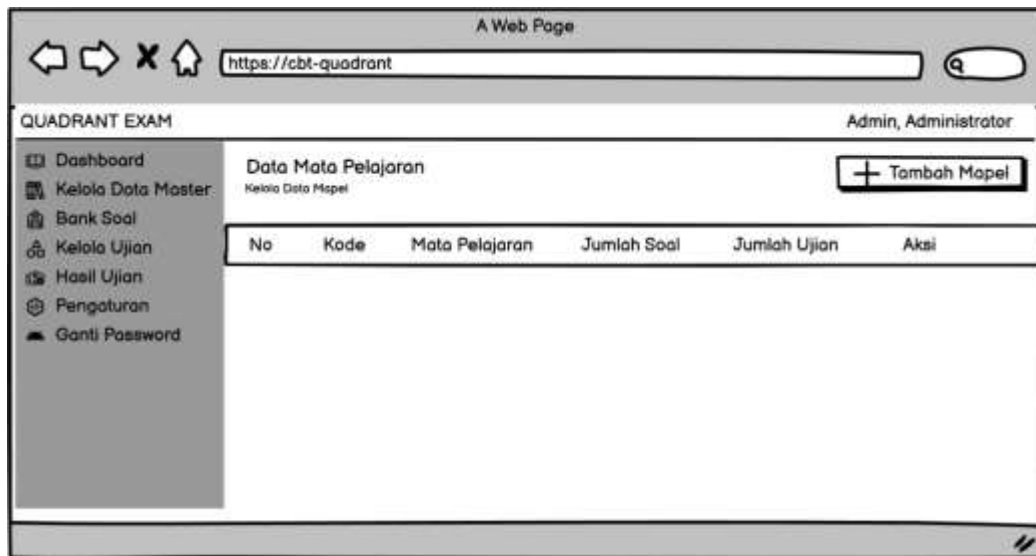
Halaman output data kelas akan ditampilkan ketika admin mengakses menu data kelas. Tampilan ini menyajikan daftar seluruh kelas yang tersedia di Bimbingan Belajar Quadrant dalam format tabel yang mudah dibaca. Admin dapat melakukan penambahan kelas baru, mengubah data kelas yang sudah ada, atau menghapus kelas yang tidak diperlukan lagi. Rancangan tampilan output data kelas dapat dilihat pada Gambar 4.27.



Gambar 4.27 Rancangan *Output* Data Kelas

5. Rancangan Output Data Mata Pelajaran

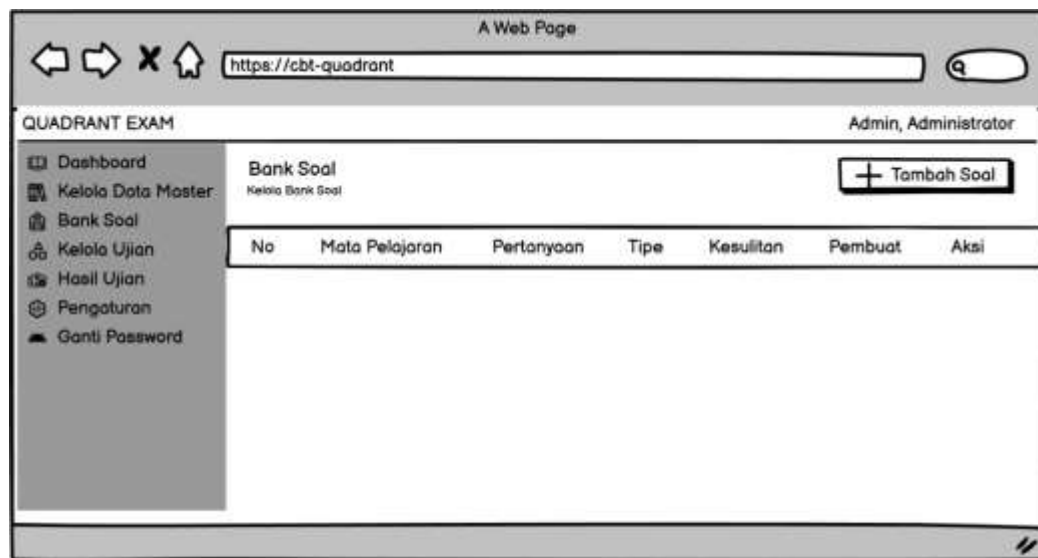
Halaman output data mata pelajaran akan muncul saat admin membuka menu pengelolaan mata pelajaran. Halaman ini menampilkan daftar seluruh mata pelajaran yang tersedia dalam sistem beserta kode dan nama mata pelajaran. Fitur pengelolaan data seperti tambah, edit, dan hapus juga disediakan untuk memudahkan admin dalam mengatur data mata pelajaran. Rancangan tampilan output data mata pelajaran ditampilkan pada Gambar 4.28.



Gambar 4.28 Rancangan *Output* Data Mata Pelajaran

6. Rancangan Output Data Bank Soal

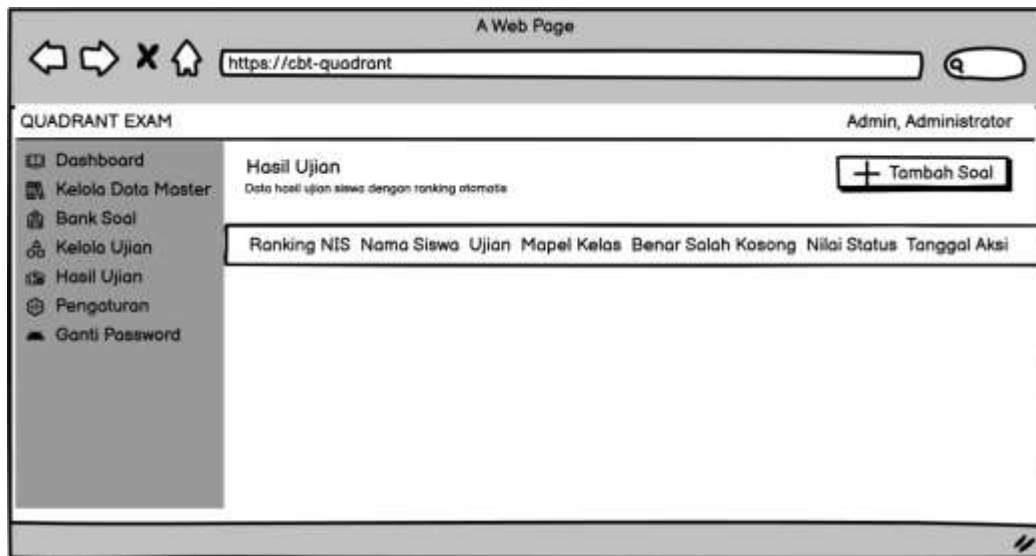
Halaman output data bank soal akan ditampilkan ketika admin atau guru mengakses menu bank soal. Tampilan ini menyajikan daftar seluruh soal yang telah diinputkan ke dalam sistem beserta informasi terkait seperti mata pelajaran, nama guru pembuat, dan statistik penggunaan soal dalam ujian. Guru dan admin dapat melakukan pengelolaan soal melalui fitur tambah, edit, dan hapus yang tersedia. Rancangan tampilan output bank soal dapat diamati pada Gambar 4.29.



Gambar 4.29 Rancangan *Output* Data Bank Soal

7. Rancangan Output Hasil Ujian

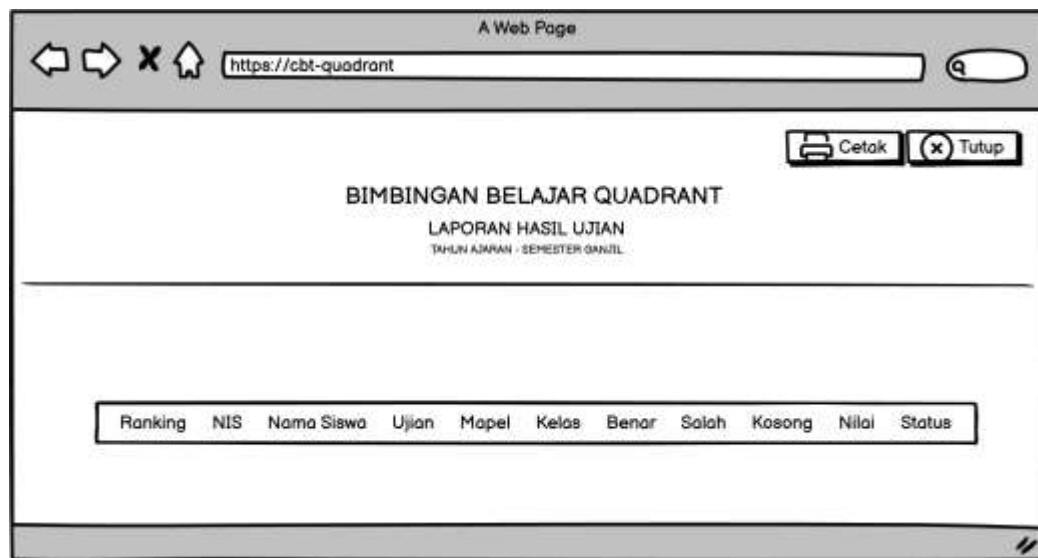
Halaman output hasil ujian akan tampil saat pengguna mengakses menu hasil ujian. Tampilan ini menyajikan informasi lengkap mengenai hasil ujian yang telah dilaksanakan, meliputi data peserta, nilai yang diperoleh, jumlah jawaban benar dan salah, peringkat siswa, serta analisis statistik seperti nilai tertinggi, nilai terendah, dan rata-rata kelas. Fitur cetak laporan juga disediakan untuk keperluan dokumentasi dan pelaporan. Rancangan tampilan output hasil ujian ditunjukkan pada Gambar 4.30.



Gambar 4.30 Rancangan *Output* Hasil Ujian

8. Rancangan Output Laporan

Halaman output laporan merupakan tampilan yang akan muncul ketika admin atau guru mengakses fitur cetak laporan. Tampilan ini menyajikan laporan hasil ujian dalam format yang siap cetak, memuat informasi lengkap seperti identitas ujian, daftar peserta beserta nilainya, dan ringkasan statistik. Laporan dapat diekspor dalam format PDF untuk keperluan dokumentasi, pelaporan kepada pihak manajemen bimbingan, maupun pemberian informasi kepada orang tua siswa. Rancangan tampilan output laporan dapat dilihat pada Gambar 4.31.



Gambar 4.31 Rancangan *Output* Laporan

4.6 RANCANGAN STRUKTUR DATA

Rancangan struktur data menjelaskan struktur tabel-tabel yang digunakan dalam database sistem aplikasi ujian online. Database yang digunakan adalah MySQL dengan framework Laravel. Berikut adalah struktur tabel-tabel dalam sistem.

4.6.1 Tabel Kelas

Tabel kelas digunakan untuk menyimpan data-data kelas dalam *database*, Nama tabel yang digunakan dalam database adalah kelas :

Tabel 4.11 Struktur Tabel Kelas

Nama Field	Tipe Data	Panjang	Keterangan
id	BIGINT	20	Primary Key, Auto Increment
nama_kelas	VARCHAR	50	Nama kelas

tingkat	VARCHAR	20	Tingkat kelas (X, XI, XII)
jurusan	VARCHAR	100	Jurusan (opsional)
keterangan	TEXT	-	Keterangan tambahan
status	ENUM	-	Status kelas (aktif/nonaktif)
created_at	TIMESTAMP	-	Waktu dibuat
updated_at	TIMESTAMP	-	Waktu diperbarui

4.6.2 Table User

Tabel user digunakan untuk menyimpan data-data user dalam *database*,

Nama tabel yang digunakan dalam database adalah user :

Tabel 4.12 Struktur Tabel Users

Nama Field	Tipe Data	Panjang	Keterangan
id	BIGINT	20	Primary Key, Auto Increment
user_id	VARCHAR	50	NIS/NIP/ID Unik, Unique
fullname	VARCHAR	100	Nama lengkap pengguna
username	VARCHAR	50	Username untuk login, Unique
email	VARCHAR	100	Email pengguna, Unique
password	VARCHAR	255	Password terenkripsi
role	TINYINT	1	Role (1=Admin, 2=Siswa, 3=Guru)
kelas_id	BIGINT	20	Foreign Key ke tabel kelas
status	ENUM	-	Status (aktif/nonaktif)
foto	VARCHAR	255	Nama file foto profil
remember_token	VARCHAR	100	Token untuk remember me
created_at	TIMESTAMP	-	Waktu dibuat

updated_at	TIMESTAMP	-	Waktu diperbarui
------------	-----------	---	------------------

4.6.3 Tabel Mapel

Tabel mapel digunakan untuk menyimpan data-data mapel dalam *database*,

Nama tabel yang digunakan dalam database adalah mapel :

Tabel 4.13 Struktur Tabel Mapel

Nama Field	Tipe Data	Panjang	Keterangan
id	BIGINT	20	Primary Key, Auto Increment
kode_mapel	VARCHAR	20	Kode mata pelajaran, Unique
nama_mapel	VARCHAR	100	Nama mata pelajaran
deskripsi	TEXT	-	Deskripsi mata pelajaran
guru_id	BIGINT	20	Foreign Key ke tabel users (guru)
status	ENUM	-	Status (aktif/nonaktif)
created_at	TIMESTAMP	-	Waktu dibuat
updated_at	TIMESTAMP	-	Waktu diperbarui

4.6.4 Tabel Soal

Tabel soal digunakan untuk menyimpan data-data soal dalam *database*,

Nama tabel yang digunakan dalam database adalah soal :

Tabel 4.14 Struktur Tabel Soal

Nama Field	Tipe Data	Panjang	Keterangan
id	BIGINT	20	Primary Key, Auto Increment
mapel_id	BIGINT	20	Foreign Key ke tabel mapel
pembuat_id	BIGINT	20	Foreign Key ke tabel users
pertanyaan	TEXT	-	Isi pertanyaan soal
gambar	VARCHAR	255	Nama file gambar (opsional)
video	VARCHAR	255	Nama file/URL video (opsional)
pilihan_a	TEXT	-	Pilihan jawaban A
pilihan_b	TEXT	-	Pilihan jawaban B
pilihan_c	TEXT	-	Pilihan jawaban C
pilihan_d	TEXT	-	Pilihan jawaban D
pilihan_e	TEXT	-	Pilihan jawaban E (opsional)
jawaban_benar	CHAR	1	Jawaban benar (A/B/C/D/E)
pembahasan	TEXT	-	Pembahasan jawaban
bobot	INT	11	Bobot nilai soal
tingkat_kesulitan	ENUM	-	Tingkat (mudah/sedang/sulit)
status	ENUM	-	Status (aktif/nonaktif)
created_at	TIMESTAMP	-	Waktu dibuat
updated_at	TIMESTAMP	-	Waktu diperbarui

4.6.5 Tabel Ujian

Tabel ujian digunakan untuk menyimpan data-data ujian dalam *database*,

Nama tabel yang digunakan dalam database adalah ujian :

Tabel 4.15 Struktur Tabel Ujian

Nama Field	Tipe Data	Panjang	Keterangan
id	BIGINT	20	Primary Key, Auto Increment
kode_ujian	VARCHAR	50	Kode ujian, Unique
nama_ujian	VARCHAR	200	Nama ujian
mapel_id	BIGINT	20	Foreign Key ke tabel mapel
kelas_id	BIGINT	20	Foreign Key ke tabel kelas
pembuat_id	BIGINT	20	Foreign Key ke tabel users
jenis_ujian	ENUM	-	Jenis (UH/UTS/UAS/Latihan/Try Out)
tanggal_ujian	DATE	-	Tanggal pelaksanaan ujian
waktu_mulai	TIME	-	Waktu mulai ujian
waktu_selesai	TIME	-	Waktu selesai ujian
durasi	INT	11	Durasi ujian dalam menit
kkm	INT	11	Kriteria Ketuntasan Minimal
acak_soal	BOOLEAN	-	Opsi pengacakan urutan soal
acak_jawaban	BOOLEAN	-	Opsi pengacakan pilihan jawaban

tampilkan_nilai	BOOLEAN	-	Tampilkan nilai setelah ujian
tampilkan_pembahasan	BOOLEAN	-	Tampilkan pembahasan
petunjuk	TEXT	-	Petunjuk pengerjaan ujian
status	ENUM	-	Status (draft/aktif/selesai/dibatalkan)
created_at	TIMESTAMP	-	Waktu dibuat
updated_at	TIMESTAMP	-	Waktu diperbarui

1.6.6 Tabel Soal Ujian

Tabel soal ujian digunakan untuk menyimpan data-data soal ujian dalam *database*, Nama tabel yang digunakan dalam *database* adalah soal ujian :

Tabel 4.16 Struktur Tabel Soal Ujian

Nama Field	Tipe Data	Panjang	Keterangan
id	BIGINT	20	Primary Key, Auto Increment
ujian_id	BIGINT	20	Foreign Key ke tabel ujian
soal_id	BIGINT	20	Foreign Key ke tabel soal
nomor_urut	INT	11	Nomor urut soal dalam ujian
created_at	TIMESTAMP	-	Waktu dibuat
updated_at	TIMESTAMP	-	Waktu diperbarui

1.6.7 Tabel Nilai

Tabel nilai digunakan untuk menyimpan data-data nilai dalam *database*,

Nama tabel yang digunakan dalam *database* adalah soal nilai :

Tabel 4.17 Struktur Tabel Nilai

Nama Field	Tipe Data	Panjang	Keterangan
id	BIGINT	20	Primary Key, Auto Increment
ujian_id	BIGINT	20	Foreign Key ke tabel ujian
user_id	BIGINT	20	Foreign Key ke tabel users (siswa)
total_benar	INT	11	Jumlah jawaban benar
total_salah	INT	11	Jumlah jawaban salah
tidak_dijawab	INT	11	Jumlah soal tidak dijawab
nilai	DECIMAL	5,2	Nilai akhir ujian
tanggal_mulai	DATETIME	-	Waktu mulai mengerjakan
tanggal_selesai	DATETIME	-	Waktu selesai mengerjakan
durasi_pengerjaan	INT	11	Durasi pengerjaan dalam detik
status	ENUM	-	Status (BELUM_SELESAI/SELESAI/TIMEOUT)
created_at	TIMESTAMP	-	Waktu dibuat
updated_at	TIMESTAMP	-	Waktu diperbarui

4.6.8 Tabel Jawaban

Tabel jawaban digunakan untuk menyimpan data-data jawaban dalam *database*, Nama tabel yang digunakan dalam *database* adalah jawaban :

Tabel 4.18 Struktur Tabel Jawaban

Nama Field	Tipe Data	Panjang	Keterangan
id	BIGINT	20	Primary Key, Auto Increment
nilai_id	BIGINT	20	Foreign Key ke tabel nilai
soal_id	BIGINT	20	Foreign Key ke tabel soal
jawaban	CHAR	1	Jawaban siswa (A/B/C/D/E)
is_benar	BOOLEAN	-	Status kebenaran jawaban
created_at	TIMESTAMP	-	Waktu dibuat
updated_at	TIMESTAMP	-	Waktu diperbarui

4.6.9 Tabel Log Aktivitas

Tabel log aktivitas digunakan untuk menyimpan data-data log aktivitas dalam *database*, Nama tabel yang digunakan dalam *database* adalah log aktivitas :

Tabel 4.19 Struktur Tabel Log Aktivitas

Nama Field	Tipe Data	Panjang	Keterangan
id	BIGINT	20	Primary Key, Auto Increment
user_id	BIGINT	20	Foreign Key ke tabel users
aktivitas	VARCHAR	100	Jenis aktivitas yang dilakukan

keterangan	TEXT	-	Detail keterangan aktivitas
ip_address	VARCHAR	45	Alamat IP pengguna
user_agent	TEXT	-	Informasi browser pengguna
created_at	TIMESTAMP	-	Waktu aktivitas
updated_at	TIMESTAMP	-	Waktu diperbarui

1.6.10 Tabel Pengaturan

Tabel pengaturan digunakan untuk menyimpan data-data pengaturan dalam *database*, Nama table digunakan dalam database adalah pengaturan :

Tabel 4.20 Struktur Tabel Pengaturan

Nama Field	Tipe Data	Panjang	Keterangan
id	BIGINT	20	Primary Key, Auto Increment
key	VARCHAR	100	Kunci pengaturan, Unique
value	TEXT	-	Nilai pengaturan
group	VARCHAR	50	Grup pengaturan
description	TEXT	-	Deskripsi pengaturan
created_at	TIMESTAMP	-	Waktu dibuat
updated_at	TIMESTAMP	-	Waktu diperbarui

