**RANCANG BANGUN SISTEM PENGELOLAAN NILAI DAN ABSENSI SEKOLAH MENENGAH PERTAMA MUARA INDONESIA BERBASIS WEB MENGGUNAKAN METODE AGILE DEVELOPMENT**

**SKRIPSI**

**Disusun untuk memenuhi sebagian persyaratan**

**mencapai derajat Sarjana**

****

**Disusun Oleh:**

Ahmad Fajar

2000018225

**PROGRAM STUDI S1 INFORMATIKA**

**FAKULTAS TEKNOLOGI INDUSTRI**

**UNIVERSITAS AHMAD DAHLAN**

**2025**

# KATA PENGANTAR

Puji syukur penulis panjatkan ke hadirat Tuhan Yang Maha Esa atas segala rahmat, karunia, dan hidayah-Nya sehingga penulis dapat menyelesaikan penyusunan skripsi yang berjudul “Rancang Bangun Sistem Informasi Akademik Sekolah Menengah Pertama Muara Indonesia Berbasis Web Menggunakan Metode Agile” ini dengan baik dan lancar.

Skripsi ini disusun sebagai salah satu syarat untuk memperoleh gelar Sarjana pada program studi [nama program studi], Fakultas [nama fakultas], [nama universitas]. Penulisan skripsi ini bertujuan untuk memberikan solusi terhadap permasalahan dalam pengelolaan data akademik di SMP Muara Indonesia, yang selama ini masih dilakukan secara manual, melalui pengembangan sistem informasi berbasis web dengan metode Agile.

Dalam proses penyusunan skripsi ini, penulis menyadari bahwa keberhasilan penyusunan tidak lepas dari bantuan, bimbingan, serta dukungan dari berbagai pihak. Oleh karena itu, penulis menyampaikan ucapan terima kasih yang sebesar-besarnya kepada:

1. Bapak/Ibu [Nama Rektor], selaku dosen pembimbing I yang telah memberikan bimbingan, arahan, dan motivasi selama penyusunan skripsi ini.
2. Bapak/Ibu [Nama Dekan], selaku dosen pembimbing II yang turut memberikan masukan yang sangat berharga dalam proses penulisan.
3. Bapak/Ibu dosen pembimbing dan staf pengajar di [nama fakultas/universitas] yang telah memberikan ilmu pengetahuan selama masa perkuliahan.
4. Kepala Sekolah, guru, dan staf SMP Muara Indonesia yang telah memberikan izin serta data yang diperlukan dalam penelitian ini.
5. Kedua orang tua tercinta dan keluarga besar yang selalu memberikan doa, semangat, dan dukungan moral maupun material.
6. Teman-teman seperjuangan dan semua pihak yang tidak dapat penulis sebutkan satu per satu, yang telah membantu dan mendukung dalam proses penyusunan skripsi ini.

Penulis menyadari bahwa skripsi ini masih jauh dari sempurna. Oleh karena itu, penulis sangat mengharapkan kritik dan saran yang membangun demi kesempurnaan karya tulis ini. Semoga skripsi ini dapat memberikan manfaat bagi pembaca serta dapat menjadi referensi untuk pengembangan sistem informasi akademik di lingkungan sekolah.

Yogyakarta, 15 Mei 2025

Ahmad Fajar

# DAFTAR ISI

[KATA PENGANTAR ii](#_Toc198770900)

[DAFTAR ISI iii](#_Toc198770901)

[DAFTAR GAMBAR iv](#_Toc198770902)

[DAFTAR TABEL v](#_Toc198770903)

[DAFTAR KODE PROGRAM vi](#_Toc198770904)

[ABSTRAK vii](#_Toc198770905)

[BAB II Pendahuluan 1](#_Toc198770906)

[1.1. Latar Belakang Masalah 1](#_Toc198770907)

[1.2. Batasan Masalah Penelitian 3](#_Toc198770908)

[1.3. Rumusan Masalah 3](#_Toc198770909)

[1.4. Tujuan Penelitian 3](#_Toc198770910)

[1.5. Manfaat Penelitian 4](#_Toc198770911)

[BAB III Tinjauan Pustaka 5](#_Toc198770912)

[2.1. Kajian Penelitian Terdahulu 5](#_Toc198770913)

[2.2. Landasan Teori 9](#_Toc198770914)

[BAB IV METODOLOGI PENELITIAN 17](#_Toc198770915)

[3.1. Pengumpulan Data 17](#_Toc198770916)

[3.2. *Software* dan *Hardware* 18](#_Toc198770917)

[3.3. Tahapan Penelitian 19](#_Toc198770918)

[DAFTAR PUSTAKA 50](#_Toc198770919)

# DAFTAR GAMBAR

[Gambar 2. 1 Tahapan Metode Agile 11](#_Toc198770975)

[Gambar 2. 2 Rumus Menghitung Skor SUS 16](#_Toc198770976)

[Gambar 2. 3 Skala Penilaian SUS 16](#_Toc198770977)

[Gambar 3. 1 Tahapan Penelitian 19](#_Toc213965532)

[Gambar 4. 1 Use Case Diagram 25](#_Toc213965543)

[Gambar 4. 2 Activity diagram login sistem 27](#_Toc213965544)

[Gambar 4. 3 Activity diagram tambah data siswa 27](#_Toc213965545)

[Gambar 4. 4 Activity diagram edit data siswa 28](#_Toc213965546)

[Gambar 4. 5 Activity diagram tambah data tahun ajaran 29](#_Toc213965547)

[Gambar 4. 6 Activity diagram edit data tahun ajaran 30](#_Toc213965548)

[Gambar 4. 7 Activity diagram tambah data kelas 30](#_Toc213965549)

[Gambar 4. 8 Activity diagram edit data kelas 31](#_Toc213965550)

[Gambar 4. 9 Activity diagram tambah data mata pelajaran 32](#_Toc213965551)

[Gambar 4. 10 Activity diagram edit data mata pelajaran 33](#_Toc213965552)

[Gambar 4. 11 Activity diagram tambah data presensi siswa 33](#_Toc213965553)

[Gambar 4. 12 Activity diagram edit data presensi siswa 34](#_Toc213965554)

[Gambar 4. 13 Activity diagram tambah data penilaian siswa 34](#_Toc213965555)

[Gambar 4. 14 Activity diagram edit data penilaian siswa 35](#_Toc213965556)

[Gambar 4. 15 Acitivity diagram tambah data kejuaraan siswa 36](#_Toc213965557)

[Gambar 4. 16 Activity diagram edit data kejuaraan siswa 37](#_Toc213965558)

[Gambar 4. 17 Entity Relationship Diagram 37](#_Toc213965559)

[Gambar 4. 18 Wireframe daftar tahun ajaran 43](#_Toc213965560)

[Gambar 4. 19 Wireframe buat tahun ajaran 44](#_Toc213965561)

[Gambar 4. 20 Wireframe edit tahun ajaran 45](#_Toc213965562)

[Gambar 4. 21 Wireframe detail tahun ajaran 46](#_Toc213965563)

[Gambar 4. 22 Wireframe daftar kelas 47](#_Toc213965564)

[Gambar 4. 23 Wireframe buat kelas 48](#_Toc213965565)

[Gambar 4. 24 Wireframe edit kelas 49](#_Toc213965566)

[Gambar 4. 25 Wireframe detail kelas 50](#_Toc213965567)

[Gambar 4. 26 Wireframe daftar presensi 51](#_Toc213965568)

[Gambar 4. 27 Wireframe buat presensi 52](#_Toc213965569)

[Gambar 4. 28 Wireframe edit presensi siswa 53](#_Toc213965570)

[Gambar 4. 29 Wireframe detail presensi siswa 54](#_Toc213965571)

[Gambar 4. 30 Wireframe daftar mata pelajaran 55](#_Toc213965572)

[Gambar 4. 31 Wireframe buat mata pelajaran 56](#_Toc213965573)

[Gambar 4. 32 Wireframe edit mata pelajaran 57](#_Toc213965574)

[Gambar 4. 33 Wireframe detail mata pelajaran 58](#_Toc213965575)

[Gambar 4. 34 Wireframe daftar admin 59](#_Toc213965576)

[Gambar 4. 35 Wireframe buat admin 60](#_Toc213965577)

[Gambar 4. 36 Wireframe edit admin 61](#_Toc213965578)

[Gambar 4. 37 Wireframe detail admin 62](#_Toc213965579)

[Gambar 4. 38 Wireframe daftar siswa 63](#_Toc213965580)

[Gambar 4. 39 Wireframe buat siswa 64](#_Toc213965581)

[Gambar 4. 40 Wireframe edit siswa 65](#_Toc213965582)

[Gambar 4. 41 Wireframe detail siswa 66](#_Toc213965583)

[Gambar 4. 42 Wireframe daftar guru 67](#_Toc213965584)

[Gambar 4. 43 Wireframe buat guru 68](#_Toc213965585)

[Gambar 4. 44 Wireframe edit guru 69](#_Toc213965586)

[Gambar 4. 45 Wireframe detail guru 70](#_Toc213965587)

[Gambar 4. 46 Wireframe daftar penilaian 71](#_Toc213965588)

[Gambar 4. 47 Wireframe tambah penilaian 72](#_Toc213965589)

[Gambar 4. 48 Wireframe edit penilaian 73](#_Toc213965590)

[Gambar 4. 49 Wirefram detail penilaian 74](#_Toc213965591)

[Gambar 4. 50 Wireframe daftar kejuaraan 75](#_Toc213965592)

[Gambar 4. 51 Wireframe tambah kejuaraan 76](#_Toc213965593)

[Gambar 4. 52 Wireframe edit kejuaraan 77](#_Toc213965594)

[Gambar 4. 53 Wireframe detail kejuaraan 78](#_Toc213965595)

# DAFTAR TABEL

[Tabel 2. 1 Perbandingan beberapa penelitian terdahulu mengenai klasifikasi 8](#_Toc213965451)

[Tabel 2. 2 Notasi Use Case Diagram 12](#_Toc213965452)

[Tabel 2. 3 Notasi Activity Diagram 13](#_Toc213965453)

[Tabel 2. 4 Form System Usability Scale (SUS) 15](#_Toc213965454)

[Tabel 4. 1 Kebutuhan Fungsional 23](#_Toc213965475)

[Tabel 4. 2 Kebutuhan Non-Fungsional 24](#_Toc213965476)

[Tabel 4. 3 Struktur tabel users 38](#_Toc213965477)

[Tabel 4. 4 Struktur tabel academic year 38](#_Toc213965478)

[Tabel 4. 5 Struktur tabel classroom 39](#_Toc213965479)

[Tabel 4. 6 Struktur tabel student classroom 39](#_Toc213965480)

[Tabel 4. 7 Struktur tabel presences 40](#_Toc213965481)

[Tabel 4. 8 Struktur tabel subjects 40](#_Toc213965482)

[Tabel 4. 9 Struktur tabel assesments 40](#_Toc213965483)

[Tabel 4. 10 Struktur tabel student assesments 41](#_Toc213965484)

[Tabel 4. 11 Struktur tabel student presences 41](#_Toc213965485)

[Tabel 4. 12 Struktur tabel championships 42](#_Toc213965486)

# DAFTAR KODE PROGRAM

[Kode Program 4.1. Menampilkan tulisan “Hello World!” **Error! Bookmark not defined.**](#_Toc97531850)

# ABSTRAK

Teknologi informasi saat ini memiliki peran krusial sebagai sarana untuk menyebarkan informasi dengan cara yang efisien dan efektif. Salah satu bentuk teknologi yang dapat memfasilitasi komunikasi dan distribusi informasi merupakan website. Website merupakan platform yang dapat diakses secara daring oleh pengguna untuk memperoleh informasi yang diperlukan tanpa terikat oleh lokasi dan waktu, asalkan perangkat terhubung ke internet.

SMP Muara Indonesia, yang merupakan salah satu lembaga pendidikan tingkat menengah pertama, saat ini menghadapi tantangan dalam pengelolaan data akademik yang masih dilakukan secara manual. Proses pencatatan absensi, nilai ujian, dan prestasi siswa umumnya menggunakan media kertas atau spreadsheet yang tidak terintegrasi, sehingga rentan terhadap kehilangan data, kesalahan pencatatan, dan rendahnya efisiensi.

Masalah dalam pengelolaan data akademik ini juga berdampak pada keterbatasan akses informasi, baik bagi siswa maupun orang tua. Hal ini sering kali menghambat keterlibatan mereka dalam memantau perkembangan akademik siswa, yang pada akhirnya dapat mempengaruhi kualitas proses belajar mengajar di sekolah. Selain itu, pihak manajemen sekolah mengalami kesulitan dalam menganalisis data untuk pengambilan keputusan strategis terkait peningkatan kualitas pendidikan.

Untuk mengatasi permasalahan tersebut, diperlukan sebuah sistem informasi akademik berbasis web yang dapat mengotomatisasi proses pengelolaan data akademik, seperti absensi, nilai ujian, prestasi siswa, hingga pembuatan laporan akhir atau rekapitulasi data. Dengan menggunakan metode agile yang bersifat dinamis menyesuaikan pada kebutuhan, keterlibatan langsung dari pengguna guna memastikan kebutuhan bisnis terpenuhi, serta pemberian nilai tambah secara berkala diharapkan dapat mempermudah pengguna dalam menggunakan website secara baik, untuk memperoleh informasi yang relevan.

Kata kunci: Pemerograman Web, Sistem Informasi Akademik, Rancang Bangun, Agile Development

# PENDAHULUAN

## Latar Belakang Masalah

Teknologi informasi saat ini memiliki peran krusial sebagai sarana untuk menyebarkan informasi dengan cara yang efisien dan efektif[1]. Salah satu bentuk teknologi yang dapat memfasilitasi komunikasi dan distribusi informasi merupakan website. Website merupakan platform yang dapat diakses secara daring oleh pengguna untuk memperoleh informasi yang diperlukan tanpa terikat oleh lokasi dan waktu, asalkan perangkat terhubung ke internet[2]. Keunggulan ini menjadikan website sebagai solusi yang ideal dalam pengembangan sistem informasi yang mampu mengolah dan menyampaikan data dengan cepat dan akurat[2].

SMP Muara Indonesia, yang merupakan salah satu lembaga pendidikan tingkat menengah pertama, saat ini menghadapi tantangan dalam pengelolaan data akademik yang masih dilakukan secara manual. Proses pencatatan absensi, nilai ujian, dan prestasi siswa umumnya menggunakan media kertas atau spreadsheet yang tidak terintegrasi, sehingga rentan terhadap kehilangan data, kesalahan pencatatan, dan rendahnya efisiensi. Dengan jumlah siswa yang cukup besar dan masa pendidikan yang berlangsung selama tiga tahun, dari kelas 7 hingga kelas 9, pengelolaan data akademik yang efektif dan efisien menjadi kebutuhan yang mendesak[3].

Masalah dalam pengelolaan data akademik ini juga berdampak pada keterbatasan akses informasi, baik bagi siswa maupun orang tua. Hal ini sering kali menghambat keterlibatan mereka dalam memantau perkembangan akademik siswa, yang pada akhirnya dapat mempengaruhi kualitas proses belajar mengajar di sekolah. Selain itu, pihak manajemen sekolah mengalami kesulitan dalam menganalisis data untuk pengambilan keputusan strategis terkait peningkatan kualitas pendidikan[3].

Berdasarkan observasi dan wawancara yang dilakukan di SMP Muara Indonesia, ditemukan berbagai permasalahan dalam pengelolaan data akademik. Permasalahan tersebut meliputi pendataan siswa yang masih manual menggunakan buku induk atau file Excel terpisah yang menyebabkan pencarian dan pembaruan data membutuhkan waktu lama. Pencatatan nilai oleh guru di media terpisah menyulitkan integrasi dan perhitungan nilai akhir siswa. Sistem absensi manual yang digunakan saat ini rentan terhadap kesalahan dan tidak mudah diakses oleh orang tua. Pembuatan laporan akademik memerlukan waktu lama karena harus mengumpulkan data dari berbagai sumber yang berbeda. Akses informasi bagi siswa dan orang tua terhadap data nilai, absensi, dan pengumuman sekolah masih terbatas. Selain itu, keamanan data akademik juga menjadi perhatian karena penyimpanan dalam bentuk fisik atau file elektronik yang tidak terlindungi dengan baik rentan terhadap kehilangan atau kerusakan[3].

Untuk mengatasi permasalahan tersebut, diperlukan sebuah sistem informasi akademik berbasis web yang dapat mengotomatisasi proses pengelolaan data akademik, seperti absensi, nilai ujian, prestasi siswa, hingga pembuatan laporan akhir atau rekapitulasi data. Sistem ini diharapkan dapat mempermudah guru dalam mencatat dan memantau data akademik, serta memberikan akses kepada siswa dan orang tua untuk memperoleh informasi yang relevan5.

Pengembangan sistem informasi akademik ini akan menggunakan metode Agile Development, yaitu sebuah pendekatan pengembangan perangkat lunak yang bersifat iteratif dan kolaboratif. Metode ini dipilih karena memungkinkan pengembangan sistem secara bertahap dengan fleksibilitas untuk menyesuaikan kebutuhan yang dinamis selama proses berjalan. Dengan Agile Development, pengembangan sistem informasi akademik ini akan dilakukan melalui siklus pendek yang meliputi perencanaan, desain, pengkodean, pengujian, dan evaluasi secara berulang, sehingga setiap iterasi menghasilkan bagian sistem yang dapat diuji dan disempurnakan berdasarkan masukan dari pengguna. Pendekatan ini diharapkan dapat meningkatkan komunikasi antara tim pengembang dan pengguna (guru, siswa, serta orang tua), mempercepat penyampaian nilai fungsionalitas, serta memastikan sistem informasi akademik berbasis web yang dibangun dapat memenuhi kebutuhan secara tepat dan aman di SMP Muara Indonesia[3].

## Batasan Masalah Penelitian

Untuk menjaga agar penelitian ini tetap fokus dan terarah, maka ditentukan beberapa batasan masalah sebagai berikut:

1. Sistem hanya mengelola absensi, nilai siswa, dan laporan akademik, tidak mencakup keuangan, inventaris, atau layanan non-akademik lainnya.
2. Akses Hanya untuk Internal Sekolah Pengguna sistem terbatas pada admin, guru, siswa, dan orang tua, tanpa integrasi dengan pihak eksternal seperti Dinas Pendidikan.
3. Metode pengembangan sistem menerapkan metode agile

## Rumusan Masalah

Berdasarkan latar belakang yang telah diuraikan, maka rumusan masalah dalam penelitian ini merupakan:

1. Bagaimana merancang dan membangun sistem pengelolaan nilai dan absensi siswa berbasis web di SMP Muara Indonesia menggunakan metode Agile Development?
2. Bagaimana sistem yang dibangun dapat mempermudah proses pencatatan, pemantauan, dan penyajian data nilai serta absensi siswa secara terintegrasi?
3. Bagaimana melakukan pengujian terhadap sistem pengelolaan nilai dan absensi berbasis web yang telah dikembangkan?

## Tujuan Penelitian

Tujuan dari penelitian ini merupakan:

1. Mengembangkan sistem pengelolaan nilai dan absensi siswa berbasis web di SMP Muara Indonesia dengan menggunakan metode Agile Development.
2. Memberikan kemudahan bagi guru dalam proses pencatatan dan pemantauan nilai serta absensi siswa secara efisien dan terintegrasi.
3. Melakukan pengujian terhadap sistem yang dibangun untuk memastikan fungsionalitas dan kualitas sistem sesuai dengan kebutuhan pengguna.

## Manfaat Penelitian

Beberapa manfaat yang dapat diperoleh dari penelitian ini merupakan:

1. Memberikan solusi dalam bentuk sistem berbasis web yang dapat membantu sekolah mengelola data nilai dan absensi siswa secara lebih efisien dan terintegrasi dengan baik
2. Memudahkan guru dalam proses pencatatan, perhitungan, dan pemantauan nilai siswa serta pencatatan absensi harian, sehingga dapat mengurangi kesalahan dan menghemat waktu dalam administrasi akademik.
3. Memberikan kemudahan akses informasi, sepert nilai dan kehadiran siswa, secara real-time dan transparan melalui sistem berbasis web, sehingga komunikasi antara sekolah dan orang tua menjadi lebih efektif.

# TINJAUAN PUSTAKA



## Kajian Penelitian Terdahulu

Kajian ilmiah dalam penelitian ini menggunakan artikel ilmiah yang memiliki kesamaan topik penelitian. Hasil kajian lima artikel ilmiah terdahulu yaitu:

Penelitian yang berjudul "Rancang Bangun Sistem Informasi Akademik Website untuk Madrasah Diniyah Awaliyah Ubudiyah" dilakukan oleh Muhammad Junaedi, I Gede Putu Krisna Juliharta, dan Eka Grana Aristyana Dewi pada tahun 2023. Penelitian ini bertujuan mengatasi permasalahan administrasi di MDA Ubudiyah yang masih menggunakan sistem berbasis kertas, yang menyebabkan kerusakan data, kesulitan pencarian informasi, serta lambatnya proses pengelolaan. Untuk itu, peneliti merancang dan membangun sistem informasi akademik berbasis website guna mempermudah pengelolaan data siswa, guru, kelas, dan keuangan (SPP), serta menyediakan media informasi kepada masyarakat. Metode yang digunakan merupakan studi kasus dengan pendekatan Agile dalam pengembangan perangkat lunak, pengumpulan data dilakukan melalui wawancara tidak terstruktur dan observasi, perancangan sistem menggunakan UML dan ERD, bahasa pemrograman PHP, dan database MySQL, serta pengujian sistem menggunakan metode black-box testing. Hasilnya, sistem yang dikembangkan mampu memenuhi kebutuhan pengguna dan mempermudah proses administrasi, serta mendapatkan tanggapan positif dari pihak MDA Ubudiyah. Kesimpulannya, sistem informasi ini efektif sebagai solusi atas permasalahan yang dihadapi dan mampu meningkatkan efisiensi kerja pengurus MDA Ubudiyah[1].

Penelitian yang berjudul "Rancang Bangun Sistem Informasi Pengumuman Kelulusan Siswa Berbasis Web Menggunakan Metode Agile" yang dilakukan oleh Fandi Rahmat Halim, Razanul Al-Kiramy, Della Oktoriani, Shalfa Vernia, Dwi Erlangga, dan Muhammad Luthfi Hamzah pada tahun 2023 ini mengangkat permasalahan lambatnya penyampaian informasi kelulusan kepada siswa di SMK Negeri 2 Pekanbaru akibat error sistem saat beberapa siswa melakukan pengecekan secara bersamaan. Tujuan dari penelitian ini merupakan untuk merancang ulang sistem pengumuman kelulusan berbasis web yang dapat meminimalisir kesalahan serta mempercepat proses penyampaian informasi. Penelitian ini menggunakan metode Agile dengan tahapan perencanaan, implementasi, pengujian perangkat lunak, dokumentasi, deployment, dan pemeliharaan. Hasil dari penelitian menunjukkan bahwa sistem yang dikembangkan berjalan dengan baik, memperbarui sistem lama, menambah fitur baru, dan mempermudah siswa dalam mengakses serta mencetak informasi kelulusan. Kesimpulannya, sistem informasi kelulusan berbasis web ini berhasil meningkatkan efisiensi penyampaian informasi dan memberikan kemudahan bagi pihak sekolah maupun siswa[4].

Penelitian berjudul "Implementasi Agile Software Development dalam Perancangan Sistem Pengelolaan Limbah Sampah" oleh Imam Saputra, Pristi Sukmasetya, dan Ardhin Primadewi pada tahun 2023 ini mengangkat permasalahan proses pengelolaan limbah sampah di Bank Sampah Cempaka Dusun Karang yang masih dilakukan secara manual, menyebabkan risiko kesalahan pencatatan, keterlambatan pencarian data, serta kurangnya transparansi kepada masyarakat; tujuan penelitian ini merupakan membangun sistem pengelolaan limbah sampah berbasis web untuk mempermudah proses pengolahan dan memperluas jangkauan layanan; metode yang digunakan merupakan Agile Software Development dengan teknik pengumpulan data melalui observasi, wawancara, dan studi pustaka; hasil penelitian menunjukkan sistem yang dikembangkan berjalan baik dan semua fitur berfungsi sesuai harapan berdasarkan pengujian black-box kesimpulannya, sistem ini terbukti membantu mempercepat proses, meningkatkan transparansi, serta memenuhi kebutuhan pengguna[5].

Penelitian berjudul *"Perancangan Sistem Informasi Akademik Berbasis Website (Studi Kasus: SMK Negeri 5 Kota Bekasi)"* oleh Alwan Abiyu Putra dan Popong Setiawati pada tahun 2024 ini mengangkat permasalahan pengelolaan data akademik yang masih dilakukan secara manual di SMK Negeri 5 Kota Bekasi yang menyebabkan risiko kehilangan data, kesalahan input, dan lambatnya proses administrasi. Tujuan dari penelitian ini merupakan merancang sistem informasi akademik berbasis website untuk memudahkan kepala sekolah, guru, staf tata usaha, dan siswa dalam mengakses data akademik secara efisien. Metode yang digunakan meliputi observasi, wawancara, studi pustaka, dan pengembangan sistem dengan pendekatan Agile serta pengujian menggunakan black-box testing dan System Usability Scale (SUS). Hasil penelitian menunjukkan bahwa sistem yang dirancang berjalan sesuai dengan kebutuhan pengguna dengan nilai rata-rata SUS sebesar 81 yang tergolong grade A. Kesimpulannya, sistem informasi akademik berbasis web ini berhasil meningkatkan efisiensi, akurasi, dan kemudahan akses informasi akademik di SMK Negeri 5 Kota Bekasi[6].

Penelitian berjudul \*"Rancang Bangun Sistem Informasi Sekolah Berbasis Web Dengan Menggunakan Metode Agile Pada SDN 056001 Karang Rejo" oleh Muhammad Prayoga Indra Surya dan Heri Kurniawan pada tahun 2024 ini mengangkat permasalahan keterbatasan akses informasi di SDN 056001 Karang Rejo karena belum adanya sistem informasi berbasis web sehingga penyampaian informasi masih bergantung pada brosur dan kunjungan langsung yang memakan waktu dan biaya. Tujuan penelitian ini merupakan membangun sistem informasi sekolah berbasis web yang dapat menyajikan data guru, siswa, absensi, nilai, dan jadwal pelajaran secara cepat dan akurat serta memudahkan akses informasi bagi masyarakat, siswa, alumni, dan orang tua. Metode yang digunakan merupakan Agile dengan tahapan observasi, wawancara, studi pustaka, implementasi sistem berbasis PHP dan MySQL, pengujian perangkat lunak, dokumentasi, deployment, dan pemeliharaan. Hasil penelitian menunjukkan bahwa sistem berhasil dibangun sesuai kebutuhan pengguna dan mempermudah penyampaian serta pembaruan informasi sekolah. Kesimpulannya, sistem informasi berbasis web ini mampu meningkatkan efisiensi, transparansi, dan kredibilitas sekolah dalam pengelolaan informasi pendidikan[7].

Tabel 2. 1 Perbandingan beberapa penelitian terdahulu mengenai klasifikasi

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Peneliti (Sitasi) \* | Teknologi | Database | Metode | Hasil\* |
| **Muhammad Junaedi, I Gede Putu Krisna Juliharta, Eka Grana Aristyana Dewi**[1] | Web | Mysql | Agile | Hasil penelitian menunjukkan bahwa penerapan metode Agile dalam pengembangan sistem informasi kelulusan siswa di SMK Negeri 2 Pekanbaru berjalan dengan baik. Sistem yang dikembangkan berhasil memperbarui sistem lama yang sebelumnya mengalami error saat banyak siswa mengakses secara bersamaan, sehingga menyebabkan keterlambatan informasi. Sistem baru dilengkapi fitur login siswa menggunakan NISN, tampilan pengumuman kelulusan, dan fitur cetak kelulusan. Pengujian menggunakan black-box testing menunjukkan bahwa seluruh fitur system mulai dari login, pengelolaan data siswa dan admin, tampilan pengumuman, hingga cetak hasil berjalan dengan valid dan sesuai fungsi yang diharapkan. Secara keseluruhan, sistem ini dinilai berhasil meningkatkan kecepatan, efisiensi, dan keamanan dalam penyampaian informasi kelulusan kepada siswa. |
| **Fandi Rahmat Halim, Razanul Al-Kiramy, Della Oktoriani, Shalfa Vernia, Dwi Erlangga, Muhammad Luthfi Hamzah**[4] | Web | Myqsl | Agile | Hasil penelitian menunjukkan bahwa penerapan metode Agile dalam pengembangan sistem informasi pengumuman kelulusan siswa di SMK Negeri 2 Pekanbaru berjalan dengan baik, terbukti dari sistem yang berhasil memperbarui sistem lama, menambahkan fitur baru, meningkatkan keamanan data, mempercepat akses informasi, serta memudahkan siswa dalam memperoleh informasi kelulusan secara efisien dan efektif. |
| **Imam Saputra, Pristi Sukmasetya, Ardhin Primadewi**[5] | Web | Mysql | Agile | Hasil penelitian menunjukkan bahwa penerapan metode Agile Software Development dalam pembangunan sistem pengelolaan limbah sampah berbasis web di Bank Sampah Cempaka Dusun Karang berhasil dengan baik, terbukti sistem dapat membantu proses pengelolaan limbah secara efisien, meningkatkan transparansi data, mempercepat pencatatan dan pencarian data, mempermudah proses jual beli sampah antara masyarakat dan pengepul, serta seluruh fitur sistem berfungsi dengan baik berdasarkan pengujian black-box testing yang menunjukkan sistem berjalan sesuai harapan dan kebutuhan pengguna. |
| **Alwan Abiyu Putra, Popong Setiawati**[6] | Web | Mysql | Agile | Hasil penelitian menunjukkan bahwa perancangan sistem informasi akademik berbasis website di SMK Negeri 5 Kota Bekasi berhasil diimplementasikan menggunakan metode Agile dengan dukungan framework Laravel dan pengujian menggunakan System Usability Scale (SUS) yang menghasilkan skor rata-rata 81 atau grade A, yang membuktikan bahwa sistem ini dapat diterima dengan baik oleh pengguna karena mampu menyelesaikan masalah pengelolaan data akademik secara manual menjadi digital yang lebih efisien, akurat, dan terintegrasi. |
| **Muhammad Prayoga Indra Surya, Heri Kurniawan**[7] | Web | Mysql | Agile | Hasil penelitian menunjukkan bahwa perancangan dan pembangunan sistem informasi sekolah berbasis website di SDN 056001 Karang Rejo berhasil dilakukan menggunakan metode Agile, yang menghasilkan sistem informasi yang mampu menyampaikan informasi sekolah seperti data guru, siswa, nilai, absensi, dan jadwal pelajaran dengan lebih cepat, efisien, dan akurat, serta memudahkan akses informasi bagi siswa, guru, orang tua, dan masyarakat luas secara daring kapan saja dan di mana saja. |

Tabel 2.1 merupakan kajian terhadap penelitian-penelitian terdahulu, terdapat benang merah yang menunjukkan bahwa pengembangan sistem informasi akademik berbasis web dengan metode Agile telah terbukti efektif dalam mengatasi permasalahan pengelolaan data akademik secara manual di berbagai institusi pendidikan. Penelitian-penelitian tersebut secara umum berhasil meningkatkan efisiensi, akurasi, dan aksesibilitas data akademik, serta mendapat respons positif dari pengguna. Namun demikian, beberapa keterbatasan masih ditemukan, seperti tampilan antarmuka yang berbeda pada tiap browser (seperti pada penelitian Ratri Purwaningtyas) dan belum optimalnya keterlibatan orang tua dalam sistem informasi. Oleh karena itu, penelitian ini akan mengadopsi keunggulan integrasi data akademik dan kemudahan akses pengguna dari penelitian sebelumnya, serta mengembangkan lebih lanjut dengan menambahkan fitur akses informasi bagi orang tua melalui akun berbasis NISN (Nomor Induk Siswa Nasional), guna meningkatkan transparansi dan partisipasi dalam pemantauan perkembangan akademik siswa.

## Landasan Teori

1. **Sistem Informasi Akademik**

Sistem informasi akademik merupakan sistem yang dirancang untuk menyediakan informasi terkait sekolah. Secara umum, sistem ini mengelola berbagai data dalam lingkup akademik, seperti data pengajar, data siswa, jadwal pelajaran, jadwal ujian, serta data lain yang diperlukan sesuai kebutuhan masingmasing sekolah (Syamsul, Mangesa and Muliadi, 2021).

Berdasarkan definisi tersebut, penulis menyimpulkan bahwa sistem informasi akademik merupakan sebuah sistem yang digunakan oleh lembaga pendidikan untuk mengelola data akademik sekolah, sehingga memudahkan dalam melaksanakan kegiatan akademik[8].

1. **Website**

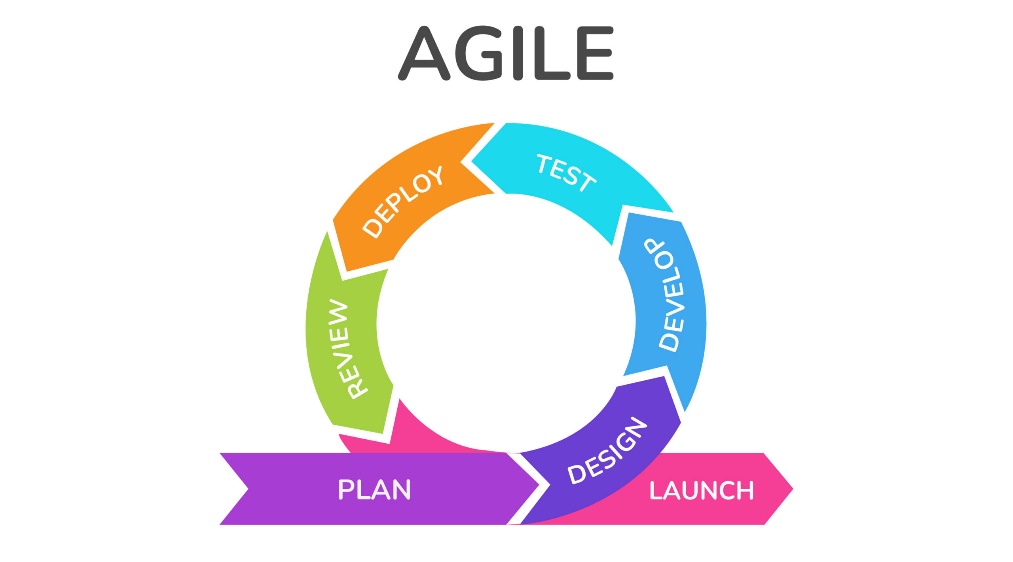
Sekumpulan halaman yang saling berhubungan yang umumnya berada pada peladen yang sama berisikan kumpulan informasi yang disediakan secara perorangan, kelompok, ataupun organisasi bisa dikatakan sebagai website.

Menurut Elgamar website merupakan sebuah media yang memiliki banyak halaman yang saling terhubung (hyperlink), dimana website memiliki fungsi dalam memberikan informasi berupa teks, gambar, video, suara dan animasi[9].

1. **Metode Agile Development**

Agile development merupakan sebuah metodologi dalam pengembangan perangkat lunak yang mengandalkan proses berulang, berdasarkan pada aturan dan solusi yang telah disepakati bersama. Pelaksanaannya dilakukan secara kolaboratif antara tim pengembang, pemangku kepentingan, dan pengguna akhir guna mengidentifikasi kebutuhan dan solusi secara bertahap selama proses pengembangan perangkat lunak.

Keunggulan dari metode ini merupakan kemampuannya untuk beradaptasi terhadap perubahan kebutuhan proyek yang bersifat dinamis, keterlibatan langsung dari pelanggan guna memastikan kebutuhan bisnis terpenuhi, serta pemberian nilai tambah secara berkala[10].



Gambar 2. 1 Tahapan Metode Agile

Gambar 2.1 merupakan gambaran tahapan metode agile development sebagai berikut:

* 1. Tahap plan merupakan proses asesmen terhadap pengguna guna memahami secara mendalam kebutuhan mereka terhadap perangkat lunak.
  2. Tahap *Design* mencakup perancangan arsitektur sistem, alur proses bisnis, serta struktur basis data.
  3. Tahap Develop merupakan proses penerapan dari hasil perancangan aplikasi, yang mencakup pengkodean dan pengelolaan basis data.
  4. Tahap Test merupakan proses untuk mengidentifikasi apakah sistem yang telah dibangun sudah sesuai dengan hasil analisis dan perancangan yang telah dibuat sebelumnya.
  5. Tahap Deploy merupakan proses mengunggah aplikasi ke layanan web hosting agar dapat diakses oleh pengguna melalui jaringan internet.
  6. Tahap Review merupakan proses evaluasi terhadap hasil yang telah dicapai serta pemberian masukan atau feedback kepada pengembang.
  7. Tahap Launch merupakan tahap akhir yang menandakan bahwa proyek telah selesai dan siap untuk digunakan oleh pengguna[11].

1. **UML (Unified Modelling Language)**

UML (Unified Modeling Language) merupakan salah satu bahasa pemodelan standar yang banyak digunakan dalam dunia bisnis untuk menggambarkan kebutuhan, analisis, dan desain suatu sistem[1]. UML menyediakan berbagai jenis diagram visual yang dapat digunakan untuk merepresentasikan berbagai aspek dalam sebuah system. Beberapa contoh diagram visual yang terdapat dalam UML antara lain:

* + 1. Use Case Diagram

Use case diagram merupakan salah satu jenis diagram visual yang digunakan untuk memvisualisasikan interaksi antara sistem dan aktor. Diagram ini menggambarkan fungsionalitas sistem dari perspektif pengguna, dengan menampilkan bagaimana aktor (pengguna atau sistem eksternal) berinteraksi dengan fitur-fitur yang disediakan oleh sistem[1]. Notasi yang digunakan dalam use case diagram dapat dilihat pada tabel 2.2.

Tabel 2. 2 Notasi Use Case Diagram

|  |  |
| --- | --- |
| Simbol | Keterangan |
|  | Aktor menggambarkan entitas seperti individu, proses, atau sistem eksternal yang berinteraksi dengan sistem yang sedang dikembangkan. |
|  | Use case merepresentasikan serangkaian aktivitas atau tindakan yang dilakukan oleh aktor dalam berinteraksi dengan sistem. |
|  | Relasi, Hubungan antara use case dan aktivitas berkaitan dengan bagaimana setiap kasus penggunaan dikaitkan dengan rangkaian aktivitas yang dilakukan untuk mencapainya. |
|  | Hubungan *include* menunjukkan bahwa suatu *use case* selalu memanfaatkan fungsionalitas yang disediakan oleh *use case* lainnya. |
|  | Relasi *extend* memungkinkan suatu *use case* secara opsional memanfaatkan fungsionalitas yang disediakan oleh *use case* lain. |

* + 1. Activity Diagram

Activity Diagram digunakan untuk memvisualisasikan urutan aktivitas dan proses bisnis dalam suatu kegiatan yang sedang berlangsung, dengan alur proses digambarkan secara vertikal. Diagram ini berfungsi sebagai alat bantu dalam memahami keseluruhan proses bisnis[1]. Notasi dalam diagram aktivitas ditampilkan dalam tabel 2.3.

Tabel 2. 3 Notasi Activity Diagram

|  |  |
| --- | --- |
| Simbol | Keterangan |
|  | Status awal, menunjukkan dimulainya aktivitas |
|  | Status akhir, menunjukkan berakhirnya aktivitas |
|  | Aktivitas, menunjukkan kegiatan yang dilakukan oleh sistem |
|  | Decision, menggambarkan tindakan atau keputusan yang harus diambil |

1. **ERD (Entity Relationship Diagram)**

ERD (Entity Relationship Diagram) atau diagram hubungan entitas merupakan sebuah diagram yang digunakan dalam proses perancangan database, yang menggambarkan hubungan antara entitas serta atribut-atribut yang dimilikinya secara rinci[1].

1. **Pengujian Black Box**

Pengujian Black Box merupakan metode pengujian perangkat lunak yang berfokus pada pemeriksaan keluaran berdasarkan input yang diberikan, tanpa melihat proses internal sistem. Tujuannya merupakan untuk memastikan bahwa fungsi-fungsi aplikasi berjalan sesuai dengan spesifikasi atau kebutuhan yang telah ditetapkan. Pengujian ini menitikberatkan pada tampilan antarmuka serta alur fungsi yang digunakan oleh pengguna.

pengujian Black Box hanya menguji sisi eksternal dari perangkat lunak tanpa memperhatikan struktur kode di dalamnya. Teknik ini berlandaskan pada spesifikasi fungsional sistem dan mencakup berbagai metode pengujian, seperti equivalence partitioning, boundary value analysis, cause-effect graphing, comparison testing, random data selection, feature testing, all-pair testing, fuzzing, orthogonal array testing, sample testing, robustness testing, behavior testing, performance testing, endurance testing, serta metode lainnya[12].

1. **Pengujian SUS (System Usability Scale)**

System Usability Scale (SUS) pertama kali diperkenalkan oleh John Brooke sebagai alat untuk melakukan pengujian kegunaan (usability testing). Pengujian ini menggunakan skala Likert, di mana hasil akhirnya berupa nilai rata-rata yang merepresentasikan tingkat kemudahan penggunaan suatu sistem serta tingkat kepuasan pengguna, dengan rentang skor antara 0 hingga 100.

SUS terdiri dari 10 pertanyaan yang diberikan kepada pengguna setelah mereka menyelesaikan pengujian terhadap sistem. Setiap pertanyaan dinilai menggunakan skala Likert dari 1 hingga 5, dengan skala 1 berarti "Sangat Tidak Setuju", skala 2 "Tidak Setuju", skala 3 "Netral", skala 4 "Setuju", dan skala 5 "Sangat Setuju". Dari sepuluh pertanyaan tersebut, lima di antaranya bersifat positif dan ditempatkan pada nomor ganjil (1, 3, 5, 7, dan 9), sedangkan lima lainnya bersifat negatif dan berada pada nomor genap (2, 4, 6, 8, dan 10)[13]. Rincian pertanyaan dapat dilihat pada Tabel 2.4.

Tabel 2. 4 Form System Usability Scale (SUS)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Pertanyaan | STS | TS | N | S | SS |
| Saya sepertinya akan sering menggunakan sistem ini |  |  |  |  |  |
| Saya menemukan fitur yang sulit dimengerti pada sistem ini |  |  |  |  |  |
| Saya merasa mudah dalam menggunakan sistem ini |  |  |  |  |  |
| Saya pikir perlu membutuhkan bantuan orang lain dalam menggunakan sistem ini |  |  |  |  |  |
| Saya merasa fitur pada sistem ini dibuat sudah terintegrasi dengan baik |  |  |  |  |  |
| Saya menemukan ketidakkonsistenan terhadap beberapa fitur pada sistem ini |  |  |  |  |  |
| Saya yakin akan lebih produktif dan efisien ketika menggunakan sistem ini |  |  |  |  |  |
| Saya kesulitan dalam menggunakan sistem ini |  |  |  |  |  |
| Saya tidak memiliki hambatan dalam menggunakan sistem ini |  |  |  |  |  |
| Saya perlu belajar sebelum menggunakan sistem ini |  |  |  |  |  |

Dalam menghitung skor untuk pertanyaan ganjil dan genap pada SUS, digunakan langkah-langkah berikut:

1. Untuk pertanyaan dengan nomor ganjil, skor setiap responden dihitung dengan rumus xi - 1, yaitu nilai jawaban dikurangi 1.
2. Untuk pertanyaan dengan nomor genap, skor dihitung dengan rumus 5 - xi, yakni 5 dikurangi nilai jawaban responden.
3. Total skor SUS diperoleh dengan menjumlahkan seluruh skor dari 10 pertanyaan, kemudian hasilnya dikalikan dengan 2,5

Nilai akhir dari perhitungan ini berada dalam rentang 0 hingga 100. Adapun rumus untuk menghitung skor SUS seperti yang terlihat pada Gambar 2.2:



Gambar 2. 2 Rumus Menghitung Skor SUS

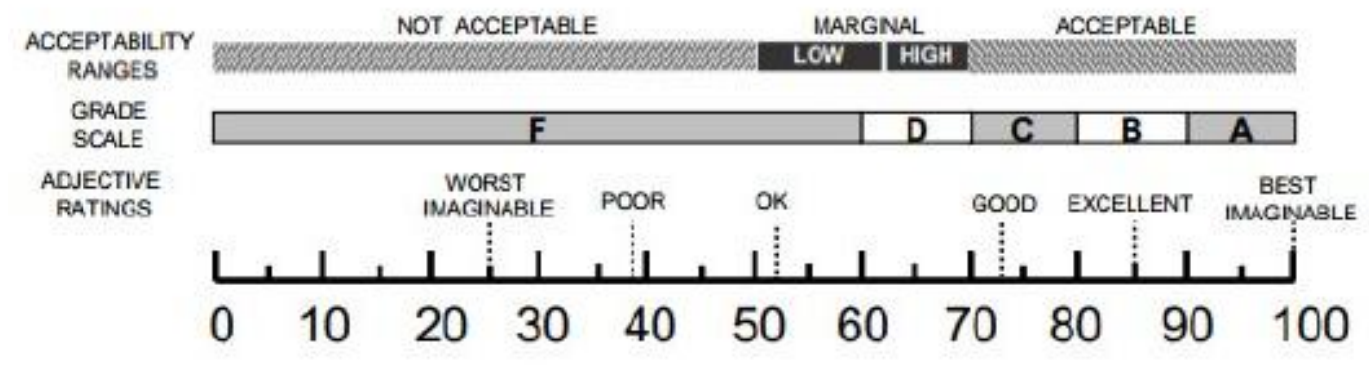
Keterangan:

𝑥̅ = nilai rata-rata skor

∑𝑥 = total skor SUS yang diperoleh

𝑛 = jumlah responden

Berdasarkan nilai rata-rata skor SUS, dapat ditentukan tingkat usability dari sistem yang dikembangkan. Penilaian usability ini diklasifikasikan ke dalam tiga kategori, yaitu Acceptability Ranges, Grade Scale, dan Adjective Ratings[13]. Skala penilaian SUS ditampilkan pada Gambar 2.3.



Gambar 2. 3 Skala Penilaian SUS

Ketentuan untuk pemberian nilai huruf (letter grades) merupakan sebagai berikut:

1. Grade A: skor SUS ≥ 90
2. Grade B: 81 ≤ skor SUS ≤ 89
3. Grade C: 70 ≤ skor SUS ≤ 80
4. Grade D: 60 ≤ skor SUS < 69
5. Grade F: skor SUS ≤ 59

# METODOLOGI PENELITIAN



## Pengumpulan Data

Penelitian memerlukan data sebagai unsur pendukung yang penting dalam menunjang kelancaran proses pelaksanaannya. Adapun metode yang digunakan dalam penelitian ini merupakan sebagai berikut:

1. **Metode Wawancara**

Metode wawancara merupakan salah satu teknik pengumpulan data yang dilakukan melalui komunikasi langsung dengan narasumber yang relevan terhadap permasalahan yang diteliti. Dalam penelitian ini, wawancara dilakukan dengan kepala sekolah yang memiliki tanggung jawab dalam pengelolaan kegiatan akademik di SMP Muara Indonesia. Wawancara tersebut menggambarkan bahwa proses akademik seperti pencatatan absensi, penilaian ujian, dan dokumentasi prestasi siswa masih dilakukan secara manual menggunakan media kertas. Cara ini dinilai rawan mengalami kesalahan, kerusakan, atau kehilangan data karena sifatnya yang tidak tahan lama.

1. **Metode Observasi**

Metode observasi merupakan salah satu cara untuk mengumpulkan data dengan melakukan pengamatan langsung terhadap aktivitas atau proses yang terjadi pada objek penelitian. Dalam konteks penelitian ini, observasi dilakukan dengan memperhatikan secara langsung kegiatan akademik di lingkungan sekolah, terutama pada interaksi antara guru dan siswa dalam hal pencatatan kehadiran, pengisian nilai, dan pencatatan prestasi siswa. Dari hasil pengamatan, diketahui bahwa seluruh aktivitas akademik tersebut masih dilakukan secara manual, belum menggunakan sistem digital, dan masih bergantung pada media pencatatan berbasis kertas.

1. **Metode Studi Literatur**

Metode studi literatur merupakan teknik pengumpulan data dengan menelusuri dan mengumpulkan informasi dari berbagai sumber tertulis, seperti jurnal ilmiah, buku, dan artikel yang memiliki keterkaitan dengan topik penelitian yang sedang dikaji.

## *Software* dan *Hardware*

Dalam pelaksanaan penelitian mengenai perancangan dan pembangunan sistem informasi akademik di SMP Muara Indonesia, dibutuhkan berbagai peralatan pendukung. Peralatan tersebut terbagi menjadi dua kategori utama, yaitu perangkat lunak (software) dan perangkat keras (hardware). Adapun rincian dari peralatan yang digunakan dijelaskan sebagai berikut:

1. **Perangkat lunak (Software)**

Pada pengembangan sistem akademik sekolah ini digunakan berbagai jenis perangkat lunak, diantaranya:

* 1. Visual Studio Code Version 1.99
  2. XAMPP v3.3.0
  3. Google Chrome, Firefox, Microsoft Edge
  4. Figma Browser
  5. Draw.io
  6. Microsoft Office (Word, Excel)

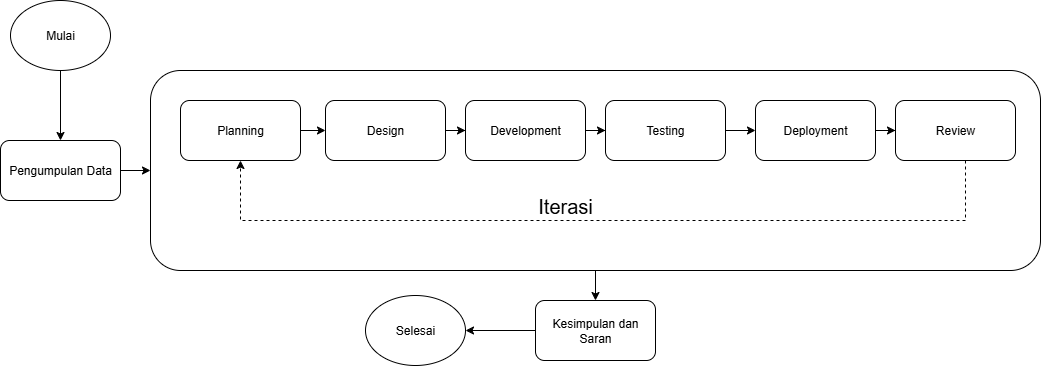
1. **Perangkat Keras (Hardware)**

Untuk pengembangan sistem akademik sekolah ini digunakan peralatan keras berupa dengan spesifikasi sebagai berikut:

* 1. Asus Rog Strix G513QR
  2. Processor AMD Ryzen 7 5800H
  3. SSD 1TB
  4. RAM 32GB

## Tahapan Penelitian

Penelitian ini menerapkan dan mengikuti beberapa tahapan yang terdapat dalam metode agile sebagaimana telah dijelaskan sebelumnya. Adapun tahapan-tahapan dalam penelitian ini dapat dilihat pada Gambar 3.1.



Gambar 3. 1 Tahapan Penelitian

1. **Pengumpulan Data**

Tahap pengumpulan data dilakukan untuk memperoleh informasi kebutuhan pengguna dan memahami proses akademik yang berjalan di sekolah. Metode yang digunakan meliputi observasi sistem yang sedang berjalan, wawancara dengan pihak sekolah, guru, dan staf, studi literatur mengenai sistem akademik. Hasil dari tahap ini adalah daftar kebutuhan sistem (requirement) yang akan menjadi acuan dalam perencanaan sprint pada tahap berikutnya.

1. **Planning**

Pada tahap perencanaan, dilakukan identifikasi kebutuhan sistem berdasarkan hasil pengumpulan data. Kegiatan meliputi penyusunan backlog, prioritas fitur, serta perencanaan sprint yang akan dilaksanakan. Tujuan tahap ini adalah memastikan seluruh kebutuhan pengguna tercatat dengan jelas sehingga proses pengembangan dapat terarah.

1. **Design**

Tahap perancangan mencakup pembuatan rancangan sistem baik dari sisi arsitektur, alur proses, maupun rancangan antarmuka pengguna (user interface). Perancangan dilakukan menggunakan prinsip UI/UX agar sistem mudah digunakan dan sesuai dengan kebutuhan pengguna. Hasil dari tahap ini berupa prototipe antarmuka serta model sistem yang siap diimplementasikan.

1. **Development**

Tahap pengembangan merupakan proses implementasi kode program sesuai rancangan yang telah dibuat. Proses pengembangan dilakukan secara bertahap sesuai sprint, sehingga fitur dapat diuji dan dievaluasi lebih cepat. Tahap ini menghasilkan aplikasi yang dapat dijalankan sesuai fungsionalitas yang telah direncanakan.

1. **Testing**

Pengujian dilakukan untuk memastikan bahwa sistem berfungsi sesuai kebutuhan. Pengujian meliputi:

1. Pengujian fungsional (blackbox testing), yang bertujuan menguji apakah setiap fitur berjalan sesuai dengan spesifikasi.
2. Pengujian usability dengan System Usability Scale (SUS), yang digunakan untuk mengukur tingkat kemudahan penggunaan serta kepuasan pengguna terhadap sistem. Instrumen SUS terdiri dari 10 butir pertanyaan dengan skala Likert 1–5, dan hasilnya diolah menjadi skor 0–100. Nilai SUS kemudian diinterpretasikan untuk menentukan tingkat kelayakan dan penerimaan sistem oleh pengguna.
3. **Deployment**

Tahap implementasi atau deployment dilakukan dengan menempatkan aplikasi pada server sehingga dapat diakses dan digunakan secara nyata oleh pihak sekolah. Pada tahap ini juga dilakukan konfigurasi serta penyesuaian agar sistem dapat berjalan stabil di lingkungan pengguna.

1. **Review**

Tahap tinjauan atau review bertujuan untuk melakukan evaluasi hasil implementasi sistem. Pada tahap ini dikumpulkan umpan balik dari pengguna dan dilakukan analisis terhadap kekurangan maupun perbaikan yang diperlukan. Apabila terdapat hal yang harus diperbaiki, maka proses kembali pada tahap planning untuk dilakukan iterasi berikutnya.

1. **Kesimpulan dan Saran**

Tahap ini berisi penyusunan kesimpulan dari hasil penelitian yang telah dilakukan serta pemberian saran untuk pengembangan lebih lanjut.

**HASIL DAN PEMBAHASAN**

1. **Hasil Pengumpulan Data**

Tahap pengumpulan data dilakukan untuk memperoleh informasi yang akurat mengenai kebutuhan pengguna serta memahami proses akademik yang berjalan di sekolah. Kegiatan ini bertujuan untuk mengetahui permasalahan yang dihadapi dalam pengelolaan data akademik dan menentukan kebutuhan sistem yang akan dikembangkan. Melalui pengumpulan data, diperoleh gambaran menyeluruh mengenai alur kerja, pihak-pihak yang terlibat, serta proses administrasi yang berlangsung di lingkungan sekolah.

Metode yang digunakan dalam tahap ini meliputi observasi, wawancara, dan studi literatur. Observasi dilakukan dengan mengamati secara langsung kegiatan akademik di sekolah, seperti proses pendaftaran siswa, pengelolaan nilai, serta pembuatan laporan akademik. Wawancara dilakukan dengan kepala sekolah, guru, dan staf administrasi untuk mendapatkan informasi mendalam terkait kebutuhan dan kendala yang dihadapi dalam pengelolaan data akademik. Sementara itu, studi literatur dilakukan dengan menelaah buku, jurnal, artikel ilmiah, serta referensi lain yang berkaitan dengan sistem informasi akademik untuk memperkuat landasan teoritis dan memperoleh pemahaman yang lebih luas.

Berdasarkan hasil pengumpulan data, diketahui bahwa proses akademik di sekolah masih dilakukan secara manual sehingga membutuhkan waktu lama dan berpotensi menimbulkan kesalahan dalam pengolahan data. Selain itu, data akademik seperti nilai, jadwal, dan laporan siswa belum terintegrasi dengan baik. Oleh karena itu, pihak sekolah memerlukan sistem informasi akademik yang mampu mengelola data secara terstruktur, cepat, dan efisien. Hasil dari tahap ini berupa daftar kebutuhan sistem (requirement) yang akan menjadi dasar dalam proses perencanaan dan pengembangan sistem pada tahap berikutnya.

1. Daftar Kebutuhan Sistem (Requirement)

Berdasarkan hasil dari kegiatan observasi, wawancara, dan studi literatur yang telah dilakukan, diperoleh daftar kebutuhan sistem yang akan dijadikan dasar dalam proses perancangan dan pengembangan sistem informasi akademik. Kebutuhan sistem ini dibagi menjadi dua kategori, yaitu kebutuhan fungsional dan kebutuhan non-fungsional.

1. Kebutuhan Fungsional

Kebutuhan fungsional merupakan fungsi-fungsi utama yang harus dimiliki oleh sistem agar dapat mendukung proses akademik di sekolah secara optimal. Adapun kebutuhan fungsional yang diidentifikasi dapat dilihat pada Tabel berikut

Tabel 4. 1 Kebutuhan Fungsional

|  |  |  |
| --- | --- | --- |
| No. | Modul | Fitur |
| 1 | ­­Autentikasi | Login dengan email dan password |
| CRUD guru oleh admin |
| CRUD siswa oleh admin |
| 2 | Modul Utama | Presensi siswa oleh guru |
| CRUD kejuaraan oleh guru |
| Create dan Update data tahun ajaran oleh admin |
| Create dan Update data kelas oleh admin |
| Create dan Update data mata pelajaran oleh admin |
| Manajemen konten website oleh admin |
| CRUD data pengguna oleh admin |
| Cetak laporan presensi |

1. Kebutuhan Non-Fungsional

Kebutuhan non-fungsional mencakup aspek-aspek kualitas yang harus dimiliki oleh sistem agar dapat berjalan dengan baik, efisien, dan mudah digunakan. Adapun kebutuhan non-fungsional sistem dapat dilihat pada Tabel berikut

Tabel 4. 2 Kebutuhan Non-Fungsional

|  |  |  |
| --- | --- | --- |
| No. | Modul | Fitur |
| 1 | Keamanan | Menggunakan autentikasi fitur login untuk dapat mengakses dashboard sistem akademik |
| 2 | Skalabilitas | Menggunakan fitur multi pengguna dan autorisasi hak akses yang dinamis untuk setiap pengguna |
| 3 | Kemudahan pengguna | UI interaktif, tabel yang memiliki filter dan sorting, responsif |
| 4 | Pemeliharaan | Menggunakan struktur kode yang mudah dibaca oleh pengembang lainnya |
| 5 | Kompatibilitas | Mendukung berbagai jenis web browser seperti Chrome, Edge, dan Firefox |
| 6 | Kinerja | Menggunakan fitur caching data sehingga mampu meningkatkan kecepatan aliran data |

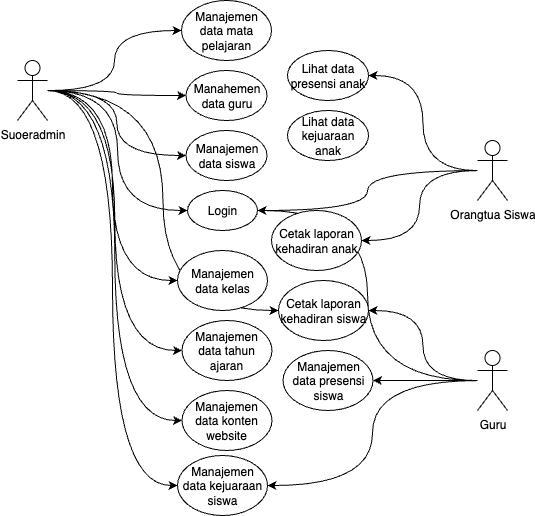
1. **Hasil Perencanaan (Design)**

Tahap perancangan (design) merupakan lanjutan dari tahap perencanaan dalam metode Agile Development. Pada tahap ini dilakukan proses perancangan sistem yang meliputi rancangan arsitektur, rancangan basis data, serta rancangan antarmuka pengguna (user interface). Tujuan dari tahap ini adalah untuk menerjemahkan kebutuhan sistem yang telah dirumuskan pada tahap planning menjadi model yang dapat dijadikan acuan dalam proses pengembangan perangkat lunak.

Perancangan dilakukan dengan menggunakan beberapa alat bantu pemodelan seperti Use Case Diagram, Activity Diagram, dan Entity Relationship Diagram (ERD) untuk menggambarkan hubungan antar komponen sistem. Selain itu, dilakukan pula perancangan tampilan antarmuka pengguna yang bersifat interaktif, sederhana, dan mudah dipahami agar dapat digunakan oleh berbagai kalangan pengguna di lingkungan sekolah.

1. Perancangan Use Case Diagram

Use case diagram digunakan untuk menunjukkan bagaimana pengguna berinteraksi dengan suatu sistem. Diagram ini memudahkan pemahaman fungsi utama sistem dan pihak-pihak (aktor) yang terlibat dalam interaksi tersebut.



Gambar 4. 1 Use Case Diagram

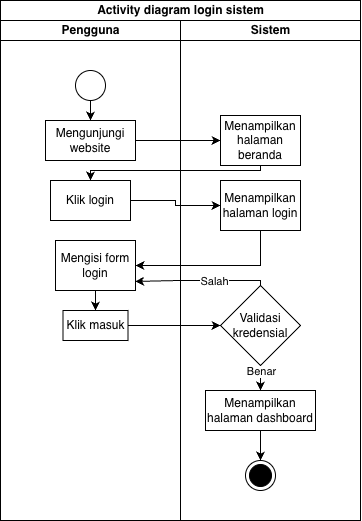
Gambar (…) menunjukkan interaksi antar aktor yaitu Admin, Guru, dan Orangtua Siswa. Admin dapat melakukan manajemen berbagai data seperti mengedit, menambah, menghapus, dan melihat berbagai macam data di dalam sistem. Kemudian Guru dapat melakukan tambah data presensi siswa dan membuatkan presensi untuk setiap siswa. Selain itu Guru juga dapat menambahkan data kejuaraan siswa. Sedangkan Orangtua Siswa hanya diberikan akses untuk melihat dan mencetak data kehadiran dan kejuaraan anaknya.

1. Perancangan Activity Diagram

Diagram aktivitas atau activity diagram dapat digunakan untuk memvisualkan alur fitur dalam sebuah sistem. Diagram ini juga digunakan untuk memahami proses yang terjadi pada setiap use case. Berikut merupakan rancangan Activity diagram yang digunakan

1. Login

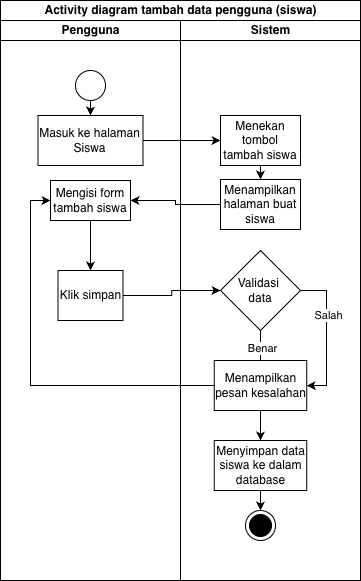
Sebelum memasuki dashboard, maka pengguna wajib melakukan login untuk memastikan pengguna memiliki akses. Gambar 4.2 berikut menunjukkan activity diagram untuk login ke dalam sistem.



Gambar 4. 2 Activity diagram login sistem

1. Tambah data siswa

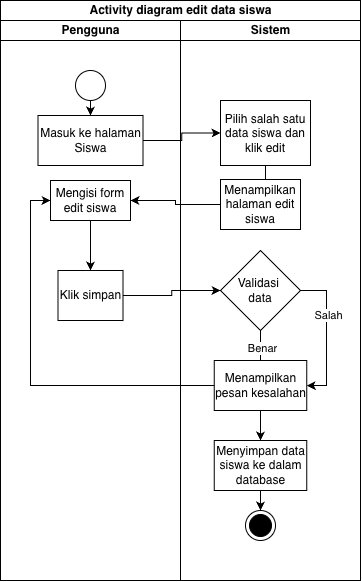
Halaman tambah data siswa dapat digunakan untuk menambahkan siswa baru dan menyimpannya ke dalam database sistem. Gambar 4.3 berikut menunjukkan activity diagram untuk melakukan tambah data siswa.



Gambar 4. 3 Activity diagram tambah data siswa

1. Edit data siswa

Halaman edit data siswa dapat digunakan untuk melakukan perubahan pada data siswa yang sudah dibuat. Untuk melakukan perubahan data siswa dapat dilihat alurnya pada Gambar 4.4 activity diagram berikut.



Gambar 4. 4 Activity diagram edit data siswa

1. Tambah data tahun akademik

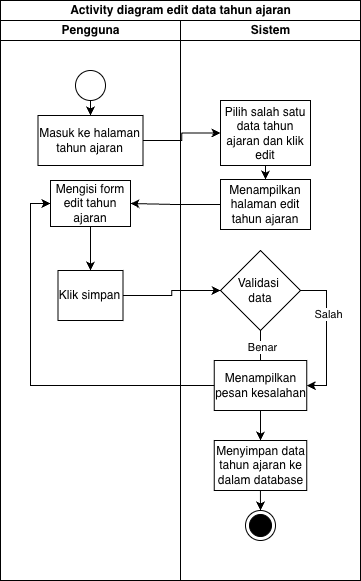
Data tahun ajaran berguna dalam beberapa modul seperti kelas dan siswa. Data tahun ajaran dapat digunakan untuk menyimpan data di setiap tahunnya. Untuk menambah data tahun ajaran dapat dilihat activity diagram nya pada Gambar 4.5 berikut.



Gambar 4. 5 Activity diagram tambah data tahun ajaran

1. Edit data tahun ajaran

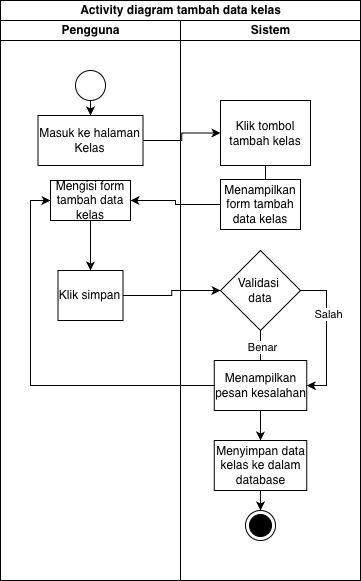
Halaman edit data tahun ajaran dapat digunakan untuk melakukan perubahan pada data tahun ajaran seperti mengubah status atau mengubah kesalahan input nama tahun ajaran. Untuk melakukan perubahan pada data tahun ajaran Gambar 4.6 berikut menunjukkan activity diagram untuk edit data tahun ajaran.



Gambar 4. 6 Activity diagram edit data tahun ajaran

1. Tambah data kelas

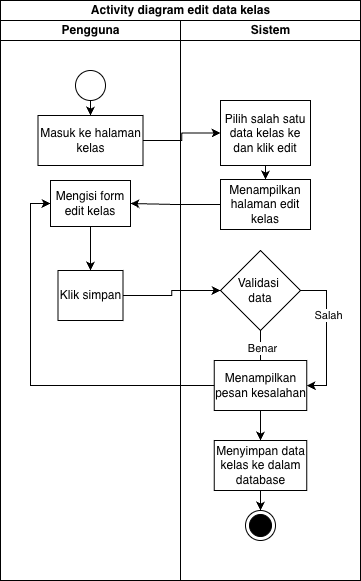
Halaman tambah data kelas dapat digunakan untuk menambahkan kelas ke dalam sistem. Gambar 4.7 menunjukkan activity diagram tambah data kelas sebagai berikut.



Gambar 4. 7 Activity diagram tambah data kelas

1. Edit data kelas

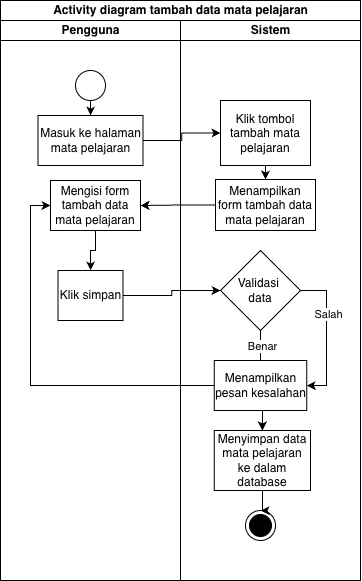
Halaman edit data kelas dapat digunakan untuk melakukan perubahan pada data kelas apabila terdapat kesalahan pada saat menambah data kelas. Gambar 4.8 berikut menunjukkan activity diagram untuk edit data kelas.



Gambar 4. 8 Activity diagram edit data kelas

1. Tambah data mata pelajaran

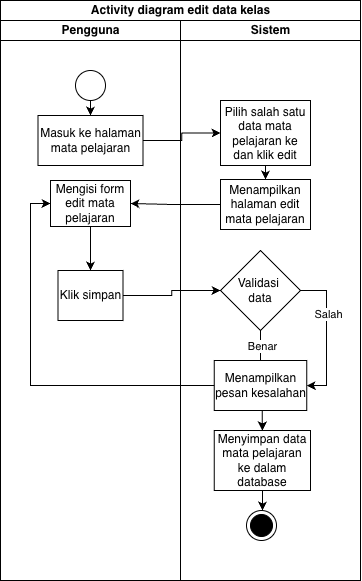
Halaman tambah mata pelajaran digunakan untuk menambahkan data mata pelajaran yang di ajarkan di sekolah dalam sistem akademik SMP Muara Indonesia. Gambar 4.9 berikut menunjukkan activity diagram untuk menambahkan mata pelajaran baru.



Gambar 4. 9 Activity diagram tambah data mata pelajaran

1. Edit data mata pelajaran

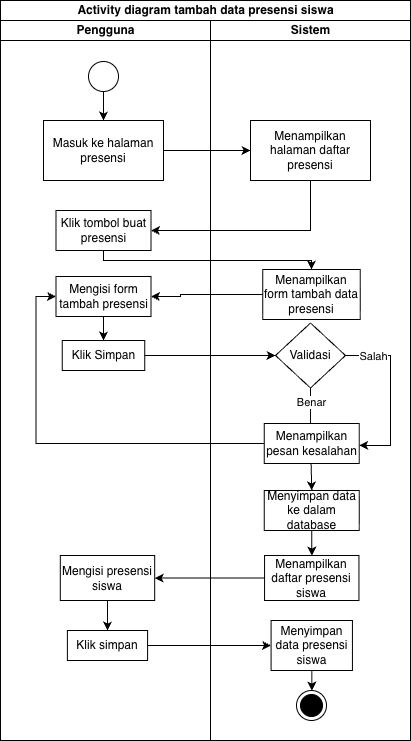
Halaman edit data mata pelajaran digunakan untuk melakukan perubahan pada data mata pelajaran yang apabila terjadi kesalah input data ketika melakukan penambahan data mata pelajaran. Activity diagram untuk edit data mata pelajaran ditunjukkan pada Gambar 4.10 berikut.



Gambar 4. 10 Activity diagram edit data mata pelajaran

1. Tambah data presensi siswa

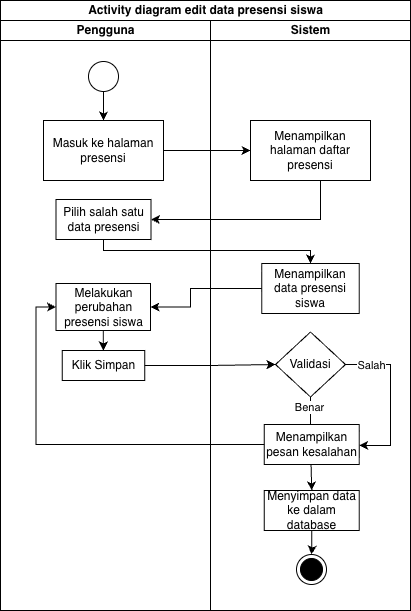
Halaman tambah data presensi siswa digunakan untuk menyimpan status kehadiran siswa berdasarkan hari. Presensi siswa diinputkan oleh guru ataupun admin. Untuk menambahkan data presensi, pengguna dapat melihat Gambar 4.11 activity diagram tambah data presensi siswa berikut.



Gambar 4. 11 Activity diagram tambah data presensi siswa

1. Edit data presensi siswa

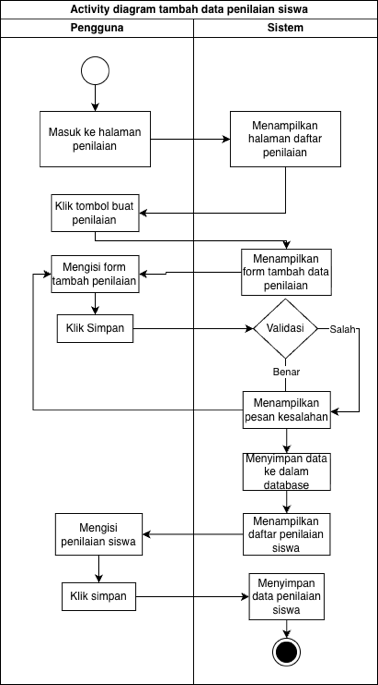
Halaman edit data presensin dapat digunakan untuk melakukan perubahan pada data presensi siswa apabila terdapat kesalahan input data pada saat menambah data presensi. Gambar 4.12 dibawah ini menunjukkan activity diagram edit data presensi siswa.



Gambar 4. 12 Activity diagram edit data presensi siswa

1. Tambah data penilaian siswa

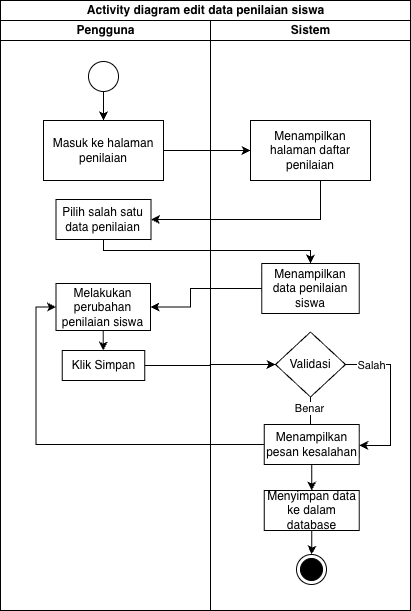
Halaman penilaian siswa dapat digunakan untuk menyimpan hasil studi atau nilai nilai dari setiap tugas siswa. Untuk menambah data penilaian siswa dapat dilihat pada Gambar 4.13 berikut.



Gambar 4. 13 Activity diagram tambah data penilaian siswa

1. Edit data penilaian siswa

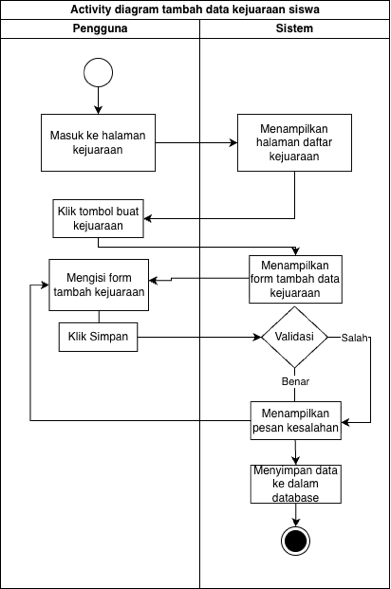
Halaman pengeditan data penilaian siswa memungkinkan Anda melakukan perubahan atau pembaruan pada data penilaian siswa jika terjadi kesalahan pada proses entri data sebelumnya. Gambar 4.14 menunjukkan diagram aktivitas yang mengilustrasikan proses penambahan data penilaian siswa.



Gambar 4. 14 Activity diagram edit data penilaian siswa

1. Tambah data kejuaraan siswa

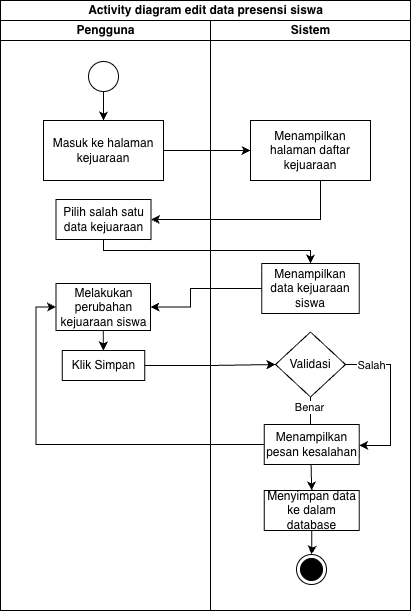
Halaman Tambah Data Kejuaraan Siswa digunakan untuk menyimpan informasi tentang prestasi atau kejuaraan siswa. Gambar 4.15 menampilkan diagram aktivitas yang mengilustrasikan proses penambahan data kejuaraan siswa.



Gambar 4. 15 Acitivity diagram tambah data kejuaraan siswa

1. Edit data kejuaraan siswa

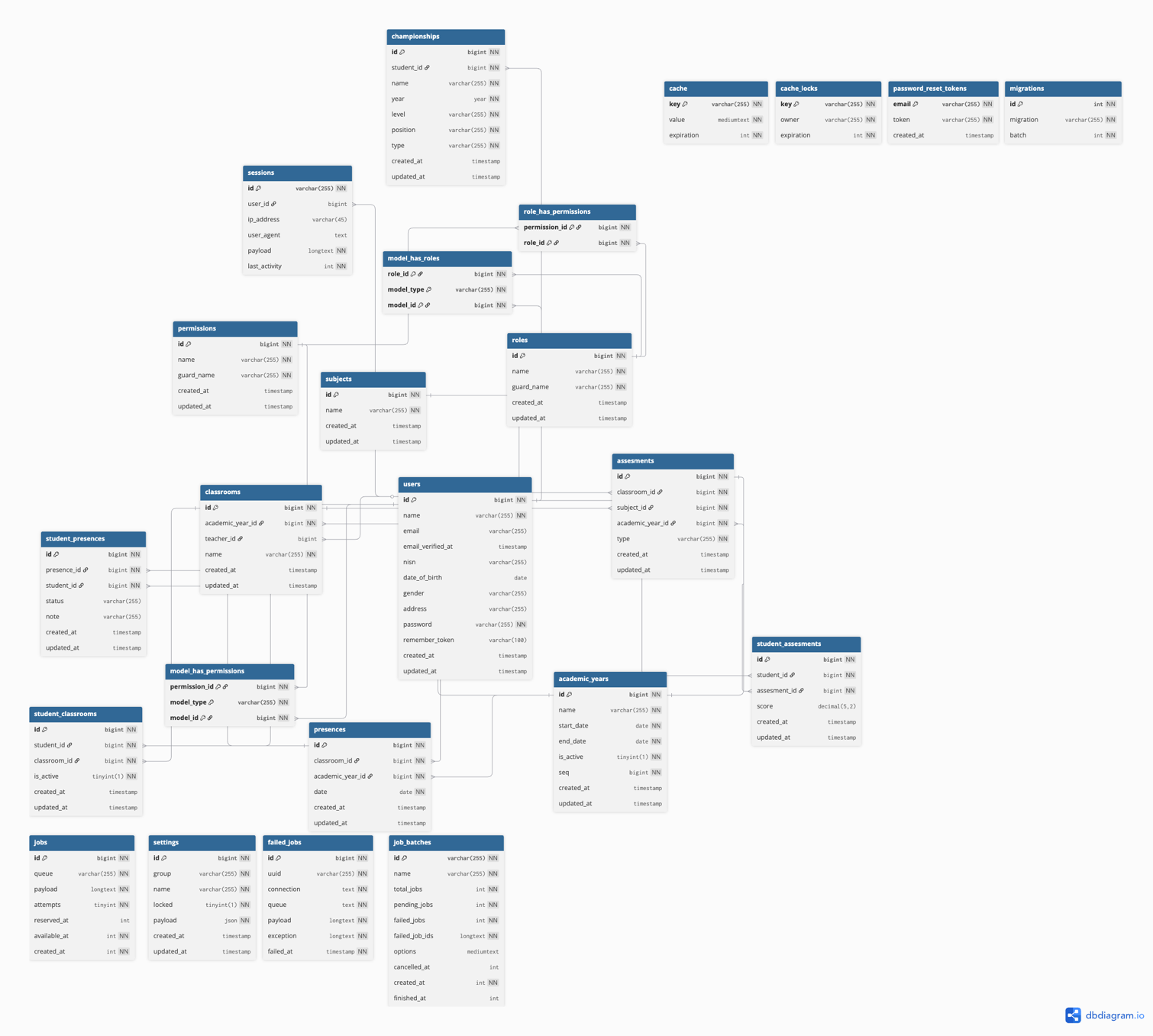
Halaman pengeditan data kejuaraan siswa memungkinkan Anda melakukan perubahan atau pembaruan pada data kejuaraan yang telah disimpan sebelumnya jika terdapat kesalahan atau data yang perlu diperbaiki. Gambar 4.16 berikut menunjukkan diagram aktivitas yang mengilustrasikan proses pengeditan data kejuaraan siswa.



Gambar 4. 16 Activity diagram edit data kejuaraan siswa

1. Perancangan Basis Data (ERD)

Entity Relationship Diagram (ERD) digunakan untuk memvisualkan atau memodelkan struktur data serta hubungan antar entitas dalam sebuah sistem.



Gambar 4. 17 Entity Relationship Diagram

Gambar 4.17 menunjukkan visualisasi diagram hubungan antar entitas dalam sebuah sistem yang selanjutnya dapat dijadikan acuan dalam merancang basis data. Berikut merupakan struktur tabel yang akan digunakan dalam sistem yang akan dirancang bangun.

* + - * 1. Struktur tabel users

Tabel pengguna berisi data akun pengguna yang dapat digunakan untuk login sistem dashboard admin. Adapun struktur tabel pengguna dapat dilihat pada Tabel 4.3 berikut.

Tabel 4. 3 Struktur tabel users

|  |  |  |  |
| --- | --- | --- | --- |
| Atribut | Tipe data | Panjang | Keterangan |
| id | integer | 11 | Primary key |
| name | varchar | 50 |  |
| email | varchar | 50 |  |
| password | varchar | 15 |  |
| nisn | varchar | 50 |  |
| Email\_verified\_at | timestamp |  |  |
| Date\_of\_birth | date |  |  |
| gender | varchar |  |  |
| address | text |  |  |

* + - * 1. Struktur tabel academic year

Tabel academic year atau tahun ajaran digunakan untuk menyimpan data tahun ajaran yang dapat digunakan untuk mengelompokkan beberapa data seperti data kelas. Adapaun struktur tabel tahun ajaran atau academic year dapat dilihat pada Tabel 4.4 berikut.

Tabel 4. 4 Struktur tabel academic year

|  |  |  |  |
| --- | --- | --- | --- |
| Atribut | Tipe data | Panjang | Keterangan |
| id | integer | 11 | Primary key |
| name | varchar | 50 |  |
| start\_date | date |  |  |
| end\_date | date |  |  |
| is\_active | boolean |  |  |
| seq | integer | 11 |  |

* + - * 1. Struktur tabel classroom

Tabel kelas digunakan untuk menyimpan informasi tentang data kelas dalam sistem akademik. Data yang tercatat meliputi tahun akademik, guru yang bertanggung jawab, dan nama kelas. Struktur tabel lengkap dapat dilihat pada Tabel 4.5

Tabel 4. 5 Struktur tabel classroom

|  |  |  |  |
| --- | --- | --- | --- |
| Atribut | Tipe data | Panjang | Keterangan |
| id | integer | 11 | Primary key |
| academic\_year\_id | integer | 11 | Foreign key |
| teacher\_id | integer | 11 |  |
| name | varchar | 50 |  |

* + - * 1. Struktur tabel student classroom

Tabel student\_classroom digunakan untuk mencatat hubungan antara mahasiswa dan kelas mereka dalam sistem akademik. Informasi yang tercatat meliputi ID mahasiswa, ID kelas, dan status aktivitas mahasiswa di kelas tersebut. Struktur tabel selengkapnya dapat dilihat pada Tabel 4.6.

Tabel 4. 6 Struktur tabel student classroom

|  |  |  |  |
| --- | --- | --- | --- |
| Atribut | Tipe data | Panjang | Keterangan |
| id | integer | 11 | Primary key |
| student\_id | integer | 11 | Foreign key |
| classroom\_id | integer | 11 | Foreign key |
| is\_active | boolean |  |  |

* + - * 1. Struktur tabel presences

Tabel kehadiran digunakan untuk menyimpan data kehadiran setiap kelas pada tanggal tertentu dalam sistem akademik. Informasi yang dicatat meliputi tahun akademik, kelas yang bersangkutan, dan tanggal ketidakhadiran. Struktur tabel lengkap dapat dilihat pada Tabel 4.7.

Tabel 4. 7 Struktur tabel presences

|  |  |  |  |
| --- | --- | --- | --- |
| Atribut | Tipe data | Panjang | Keterangan |
| id | integer | 11 | Primary key |
| academic\_year\_id | integer | 11 | Foreign key |
| classroom\_id | integer | 11 | Foreign key |
| date | date |  |  |

* + - * 1. Struktur tabel subjects

Tabel mata kuliah digunakan untuk menyimpan informasi tentang mata kuliah yang tersedia dalam sistem akademik. Data yang tercatat meliputi identitas dan nama mata kuliah. Struktur tabel lengkap dapat dilihat pada Tabel 4.8.

Tabel 4. 8 Struktur tabel subjects

|  |  |  |  |
| --- | --- | --- | --- |
| Atribut | Tipe data | Panjang | Keterangan |
| id | integer | 11 | Primary key |
| name | varchar | 50 | Foreign key |

* + - * 1. Struktur tabel assesments

Tabel penilaian digunakan untuk menyimpan informasi tentang penilaian setiap mata pelajaran dalam tahun ajaran dan kelas tertentu. Data yang tercatat meliputi tahun ajaran, kelas, mata pelajaran yang dinilai, dan jenis penilaian. Struktur tabel lengkap dapat dilihat pada Tabel 4.9.

Tabel 4. 9 Struktur tabel assesments

|  |  |  |  |
| --- | --- | --- | --- |
| Atribut | Tipe data | Panjang | Keterangan |
| id | integer | 11 | Primary key |
| academic\_year\_id | integer | 11 | Foreign key |
| classroom\_id | integer | 11 | Foreign key |
| Subject\_id | integer | 11 | Foreign key |
| type | varchar | 10 |  |

* + - * 1. Sturktur tabel student assesments

Tabel student\_assessments digunakan untuk menyimpan hasil penilaian yang diperoleh setiap siswa pada komponen penilaian tertentu. Informasi yang tercatat meliputi ID penilaian, ID siswa, dan nilai yang diberikan. Struktur tabel lengkap dapat dilihat pada Tabel 4.10.

Tabel 4. 10 Struktur tabel student assesments

|  |  |  |  |
| --- | --- | --- | --- |
| Atribut | Tipe data | Panjang | Keterangan |
| id | integer | 11 | Primary key |
| assesment\_id | integer | 11 | Foreign key |
| student\_id | integer | 11 | Foreign key |
| score | double |  |  |

* + - * 1. Struktur tabel student presences

Tabel student\_presences digunakan untuk menyimpan informasi kehadiran setiap siswa untuk aktivitas kehadiran yang telah direkam sebelumnya. Data yang direkam meliputi ID kehadiran, ID siswa, status kehadiran, dan catatan tambahan jika diperlukan. Struktur tabel lengkap dapat dilihat pada Tabel 4.11.

Tabel 4. 11 Struktur tabel student presences

|  |  |  |  |
| --- | --- | --- | --- |
| Atribut | Tipe data | Panjang | Keterangan |
| id | integer | 11 | Primary key |
| presence\_id | integer | 11 | Foreign key |
| student\_id | integer | 11 | Foreign key |
| status | varchar | 10 |  |
| notes | varchar | 200 |  |

* + - * 1. Struktur tabel championships

Tabel kejuaraan digunakan untuk menyimpan data kejuaraan yang telah diikuti siswa. Informasi yang tercatat meliputi identitas siswa, nama kejuaraan, tahun kompetisi, tingkat kejuaraan, posisi yang diraih, dan jenis kejuaraan. Struktur tabel selengkapnya dapat dilihat pada Tabel 4.12.

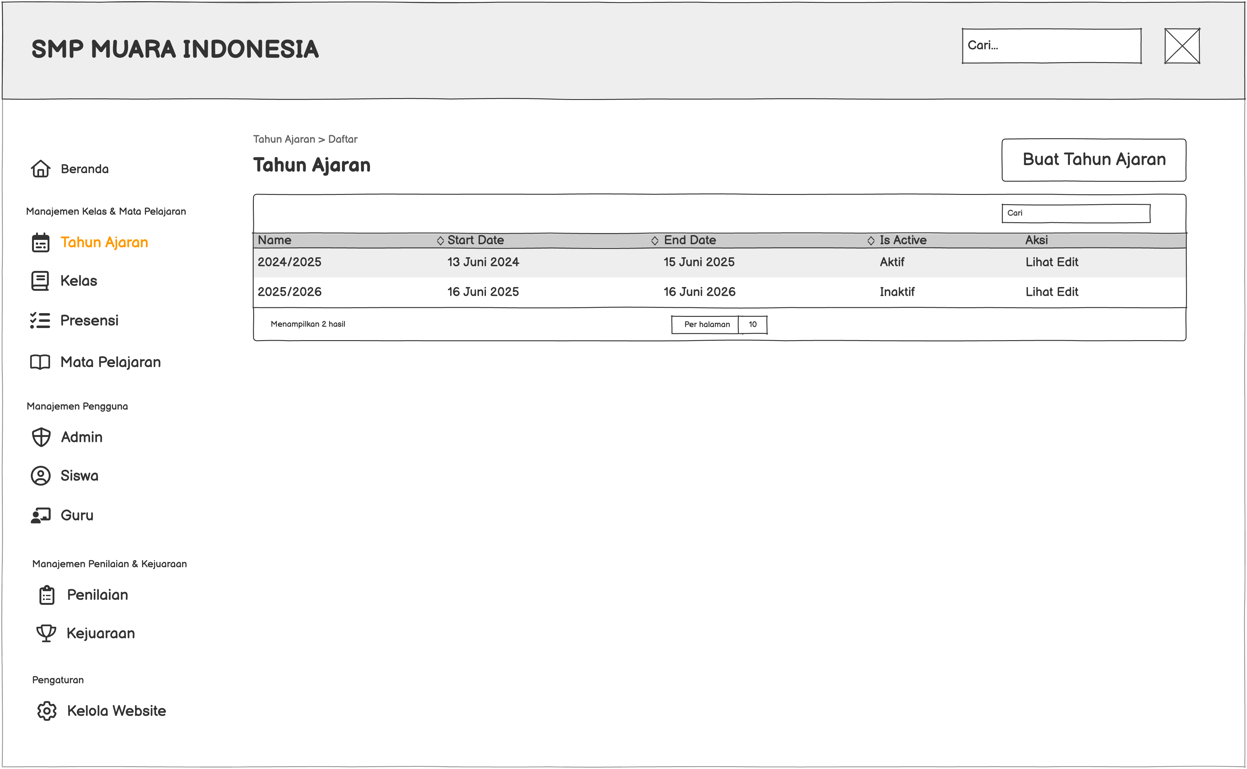
Tabel 4. 12 Struktur tabel championships

|  |  |  |  |
| --- | --- | --- | --- |
| Atribut | Tipe data | Panjang | Keterangan |
| id | integer | 11 | Primary key |
| student\_id | integer | 11 | Foreign key |
| name | integer | 11 | Foreign key |
| year | date |  |  |
| level | varchar | 10 |  |
| position | varchar | 50 |  |
| type | varchar | 10 |  |

1. Perancangan Antarmuka (Wireframe)

Aplikasi Balsamiq digunakan untuk membuat wireframe antarmuka pengguna. Balsamiq adalah alat desain wireframe yang digunakan untuk menggambarkan struktur dan tata letak awal tampilan aplikasi, baik untuk platform seluler, desktop, maupun situs web. Dengan Balsamiq, desain sistem dapat divisualisasikan secara sederhana namun jelas, membantu calon pengguna dan pengembang memahami alur dan fungsionalitas sistem sebelum beralih ke desain visual yang lebih detail.

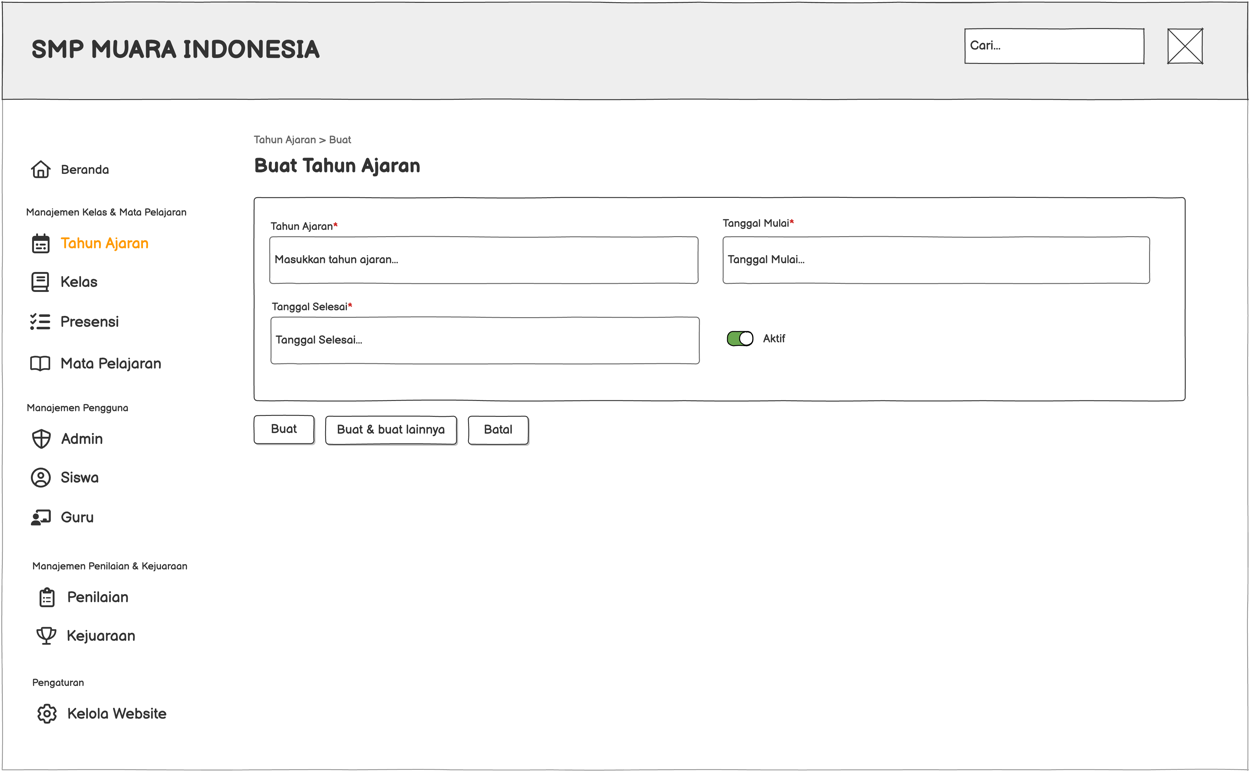
* + 1. Daftar tahun ajaran



Gambar 4. 18 Wireframe daftar tahun ajaran

Gambar 4.18 di atas menunjukkan halaman daftar tahun ajaran. Halaman ini menampilkan data tahun ajaran yang telah dibuat dan menyediakan tombol untuk menambah, melihat, dan mengedit data tahun ajaran.

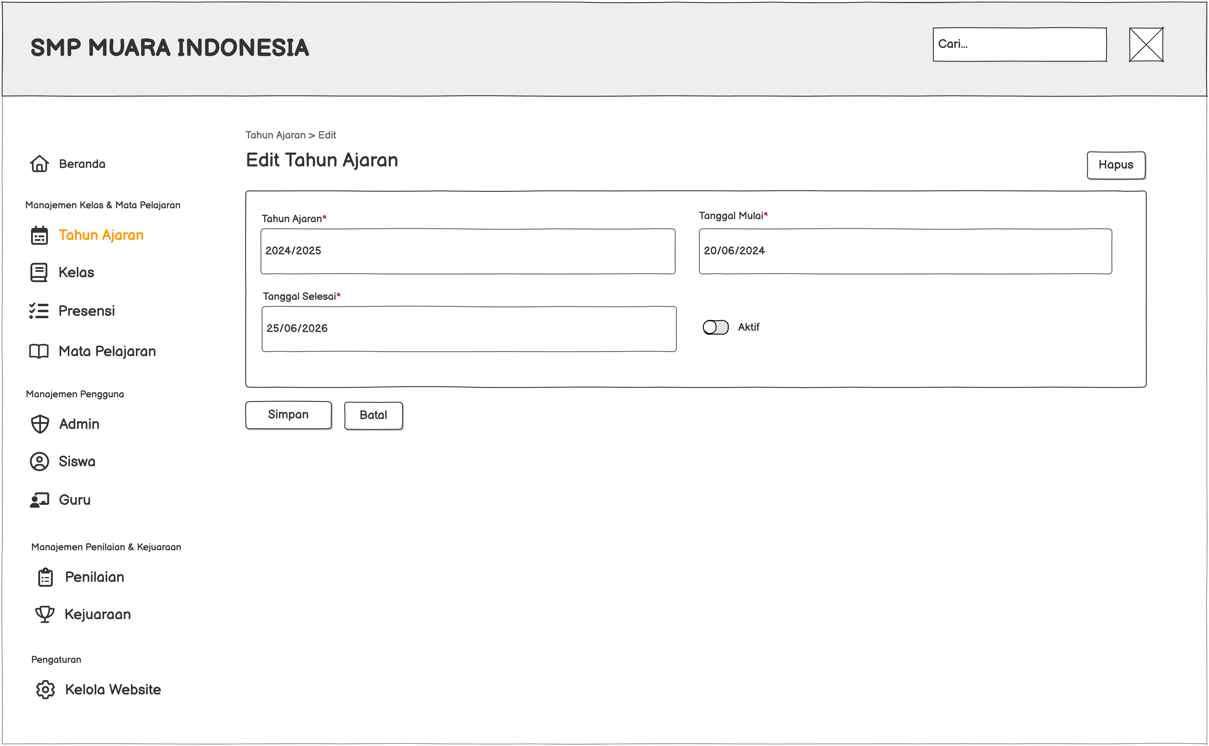
* + 1. Tambah data tahun ajaran



Gambar 4. 19 Wireframe buat tahun ajaran

Gambar 4.19 di atas menunjukkan halaman untuk membuat tahun ajaran baru. Halaman ini digunakan untuk menambahkan data tahun ajaran baru dengan memasukkan informasi seperti nama tahun, tanggal mulai, tanggal berakhir, dan status aktif. Terdapat juga tombol Buat, Buat & Buat Ulang, dan Batal untuk mengelola proses penyimpanan data.

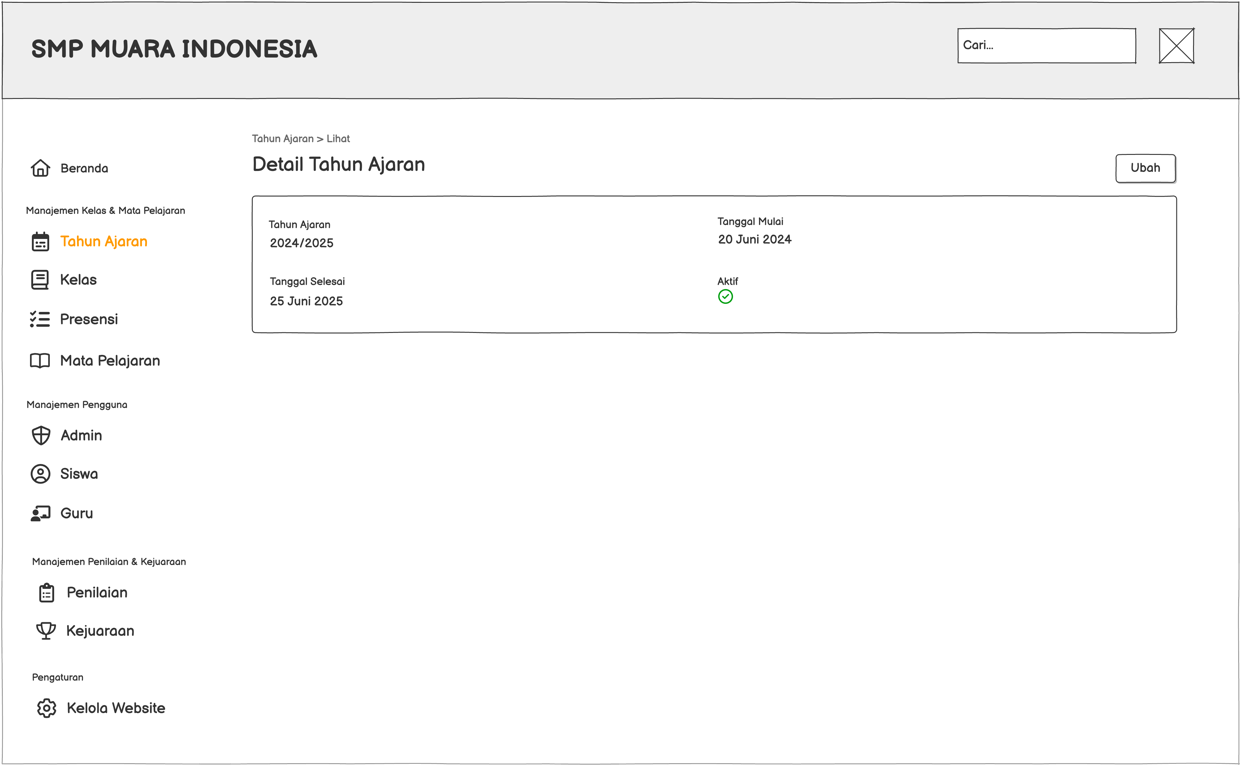
* + 1. Edit data tahun ajaran



Gambar 4. 20 Wireframe edit tahun ajaran

Gambar 4.20 menampilkan halaman edit tahun ajaran. Di halaman ini, pengguna dapat memperbarui informasi tahun ajaran yang tersimpan, termasuk nama tahun ajaran, tanggal mulai, tanggal berakhir, dan status aktif. Selain itu, terdapat tombol untuk menyimpan perubahan, Batal untuk membatalkan perubahan, dan Hapus untuk menghapus data tahun ajaran.

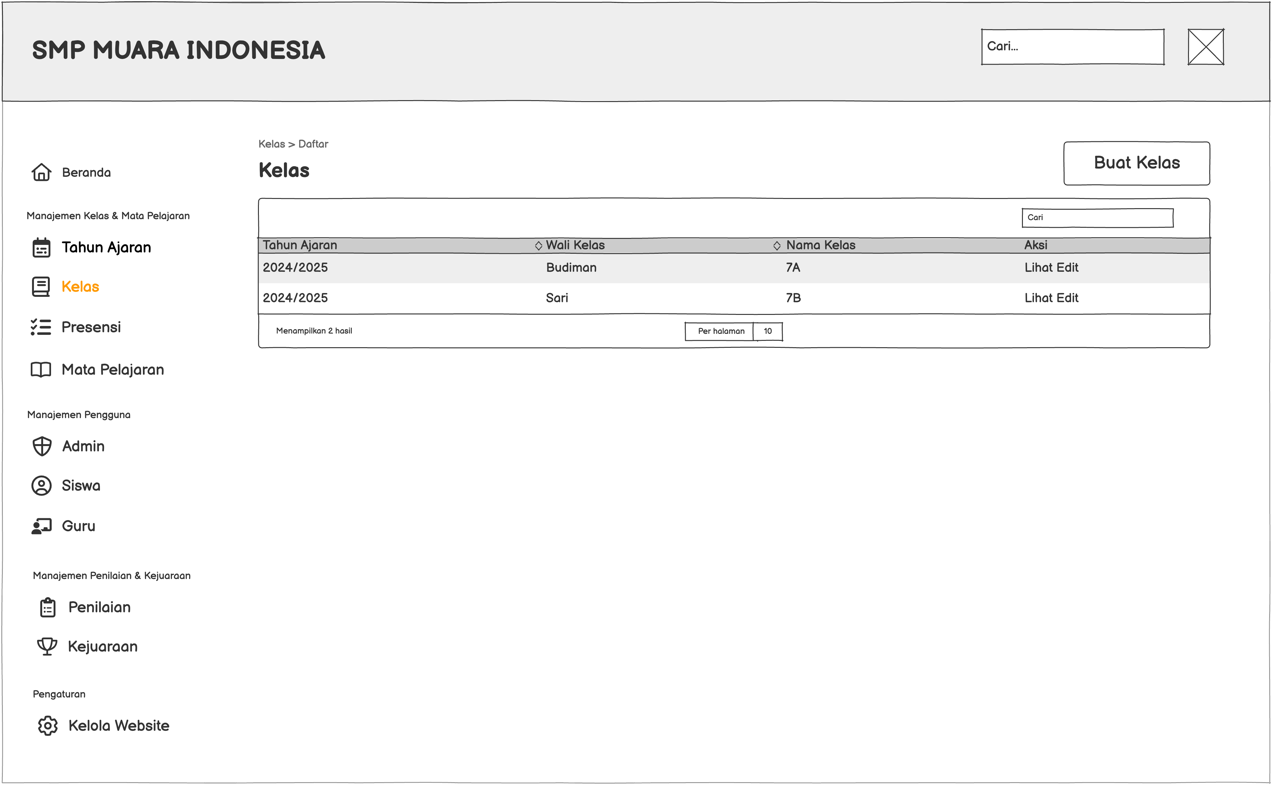
* + 1. Detail tahun ajaran



Gambar 4. 21 Wireframe detail tahun ajaran

Gambar 4.21 menunjukkan halaman detail tahun akademik. Halaman ini menampilkan informasi lengkap tentang tahun akademik yang dipilih, seperti nama tahun akademik, tanggal mulai, tanggal berakhir, dan status aktif. Tombol Ubah tersedia untuk mengarahkan pengguna ke halaman edit jika diperlukan pembaruan data.

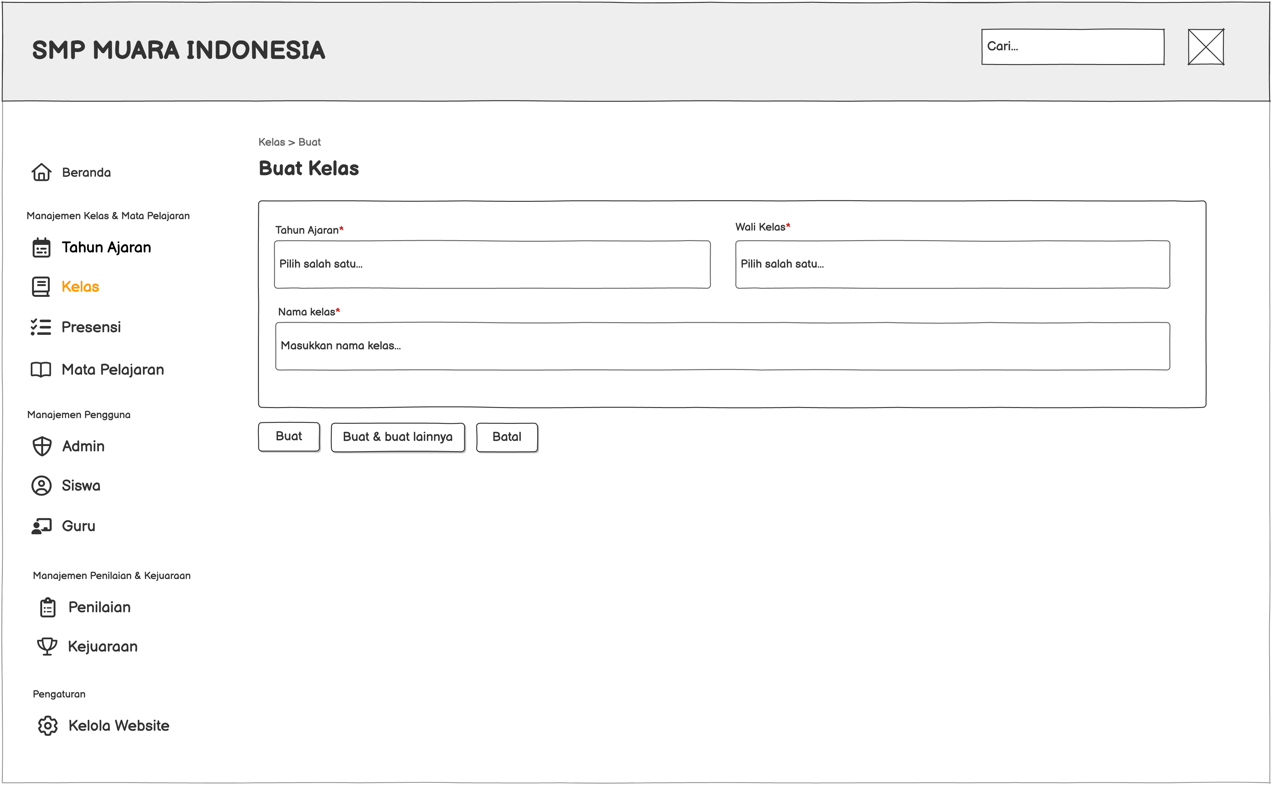
* + 1. Daftar kelas



Gambar 4. 22 Wireframe daftar kelas

Gambar 4.22 menampilkan halaman daftar kelas. Di halaman ini, pengguna dapat melihat daftar kelas terdaftar berdasarkan tahun ajaran, beserta informasi wali kelas dan nama kelas. Tombol Lihat, Edit, dan Buat Kelas juga tersedia untuk memudahkan pengelolaan data kelas.

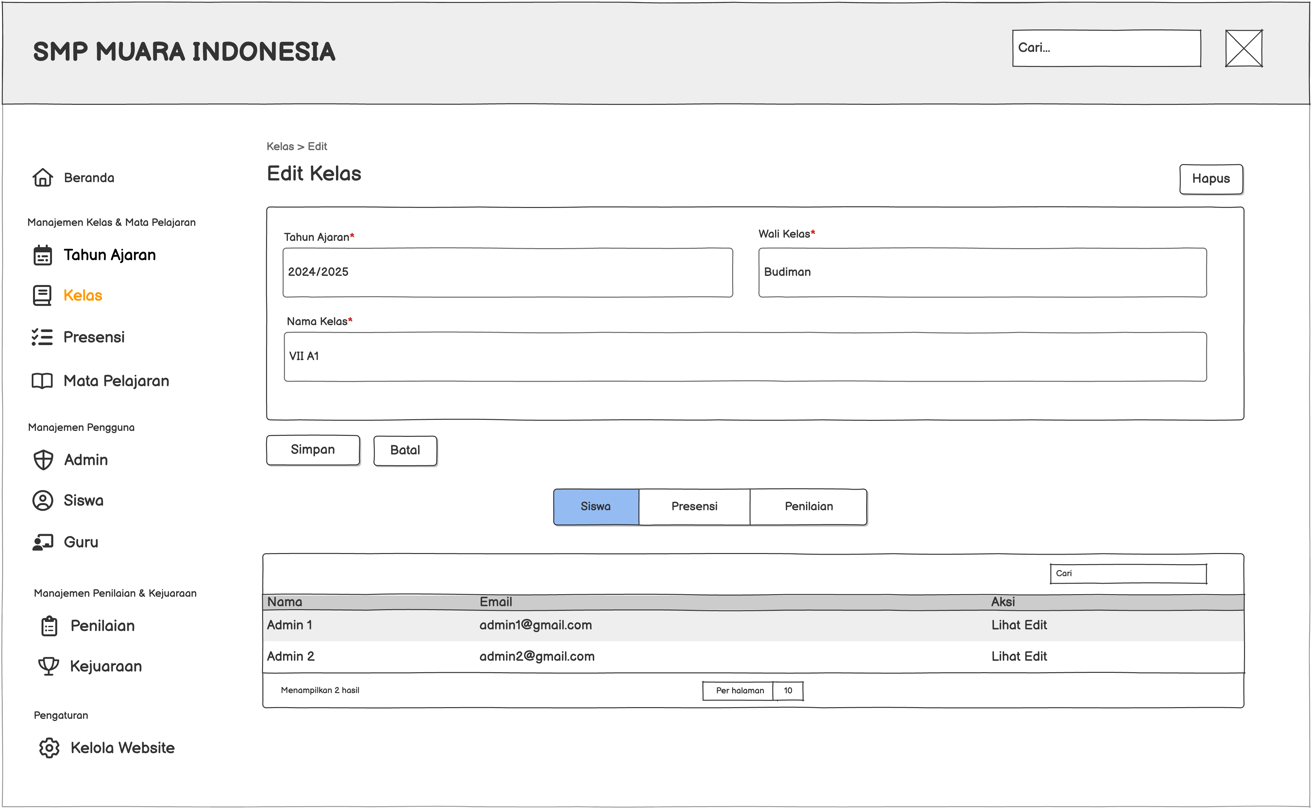
* + 1. Tambah data kelas



Gambar 4. 23 Wireframe buat kelas

Gambar 4.23 menunjukkan halaman pembuatan kelas. Halaman ini digunakan untuk menambahkan data kelas baru dengan mengisi informasi seperti tahun ajaran, wali kelas, dan nama kelas. Terdapat tombol Buat, Buat & Buat Ulang, dan Batal untuk mengelola proses penyimpanan data kelas.

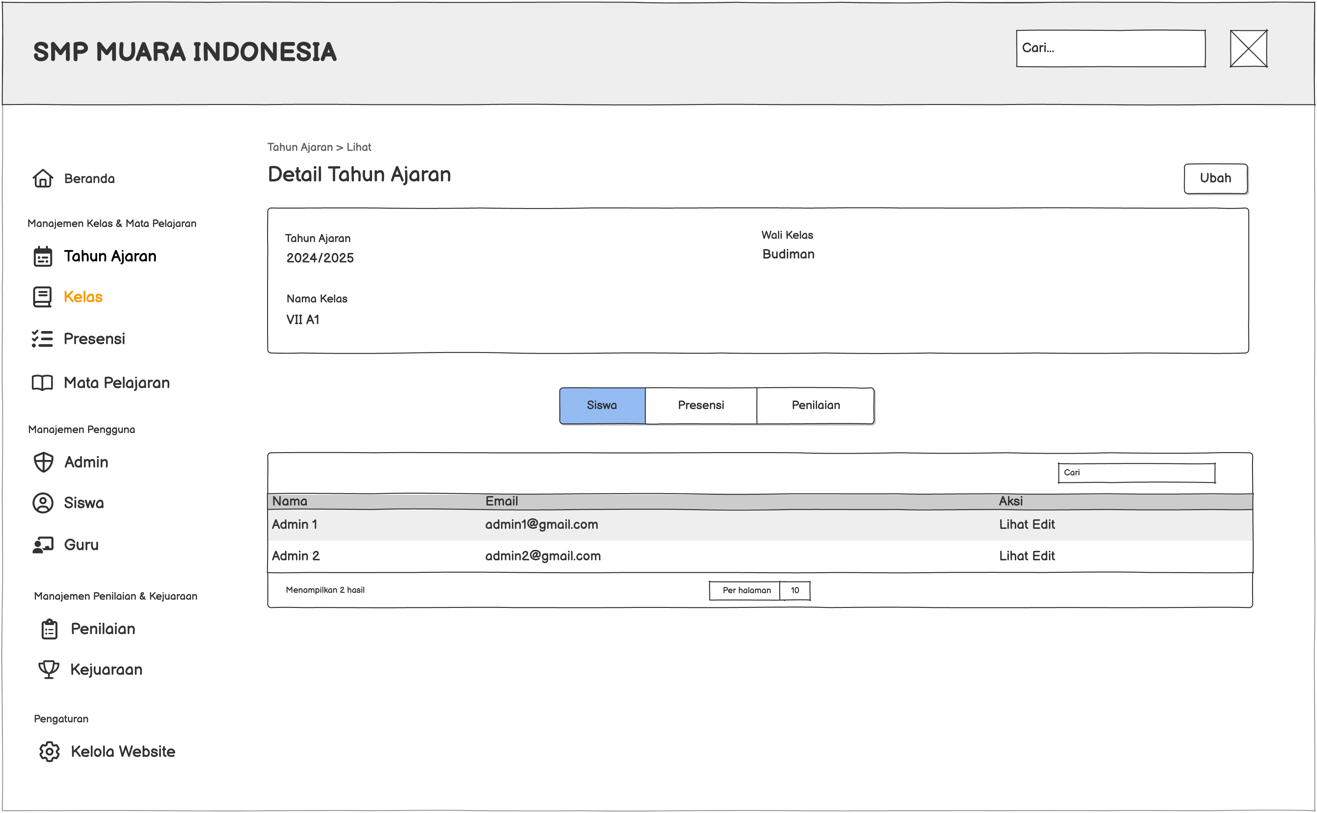
* + 1. Edit data kelas



Gambar 4. 24 Wireframe edit kelas

Gambar 4.24 menunjukkan halaman edit kelas. Halaman ini digunakan untuk memperbarui data kelas yang ada, seperti tahun ajaran, wali kelas, dan nama kelas. Halaman ini juga menampilkan daftar siswa, kehadiran, dan penilaian yang terkait dengan kelas tersebut. Tombol Simpan, Batal, dan Hapus tersedia untuk mengelola perubahan data.

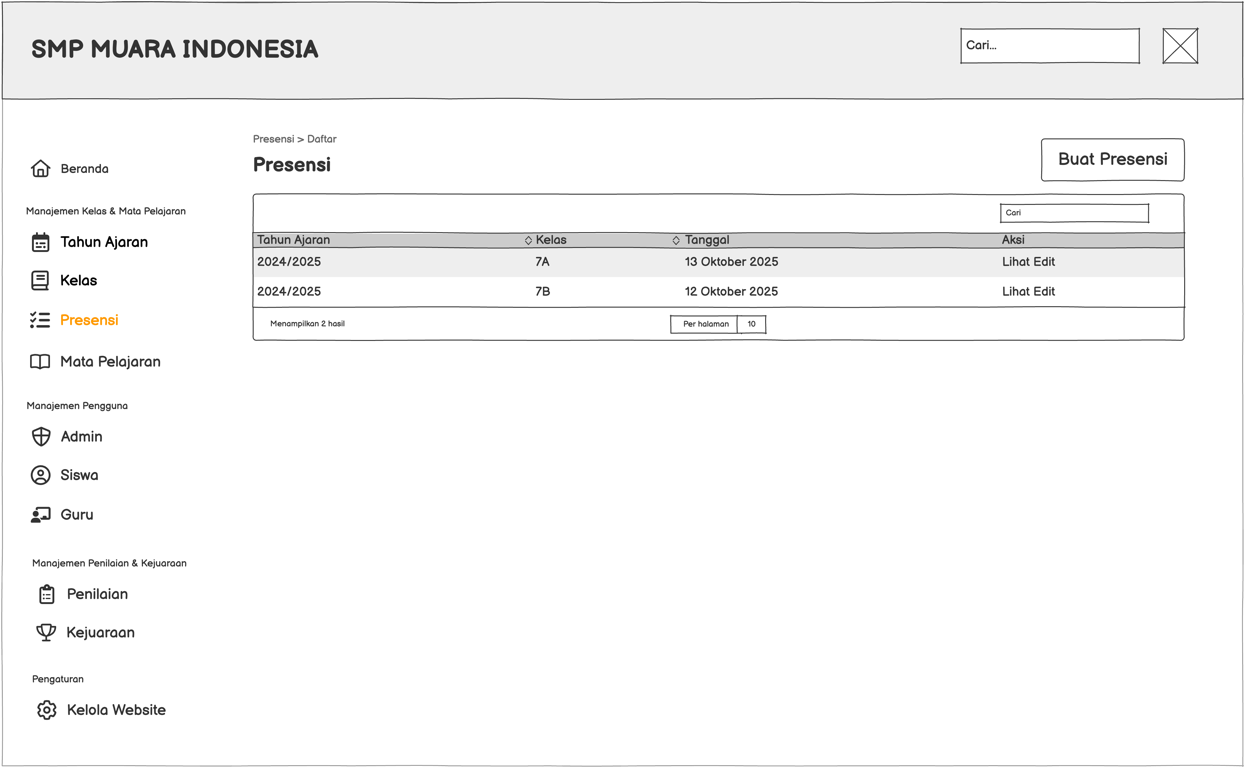
* + 1. Detail kelas



Gambar 4. 25 Wireframe detail kelas

Gambar 4.25 menunjukkan halaman detail kelas. Halaman ini menampilkan informasi lengkap tentang kelas yang dipilih, seperti tahun ajaran, wali kelas, dan nama kelas. Halaman ini juga mencakup bagian untuk melihat data siswa, kehadiran, dan penilaian terkait kelas tersebut, serta tombol Edit untuk memperbarui data.

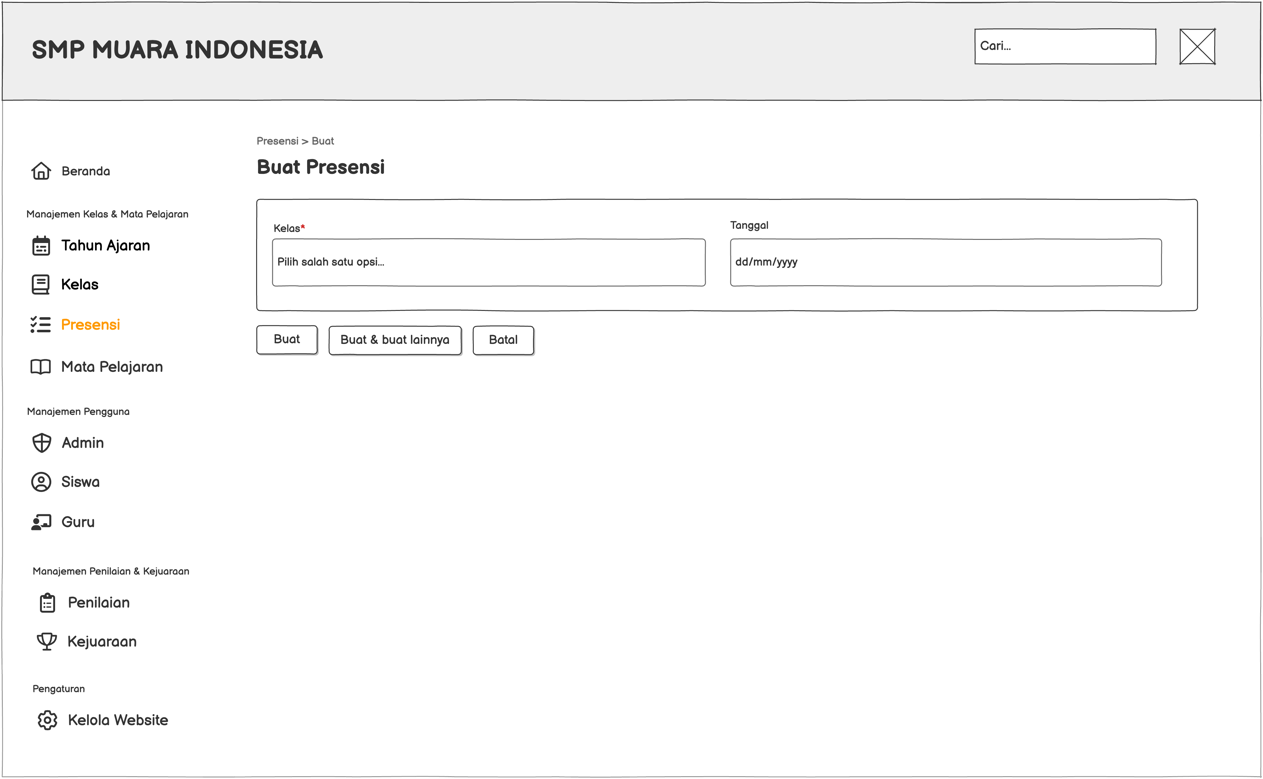
* + 1. Daftar presensi



Gambar 4. 26 Wireframe daftar presensi

Gambar 4.26 menunjukkan halaman daftar kehadiran. Halaman ini menampilkan data kehadiran untuk setiap kelas berdasarkan tahun ajaran, nama kelas, dan tanggal. Pengguna dapat melakukan tindakan seperti melihat, mengedit, atau menambahkan data kehadiran menggunakan tombol Buat Kehadiran.

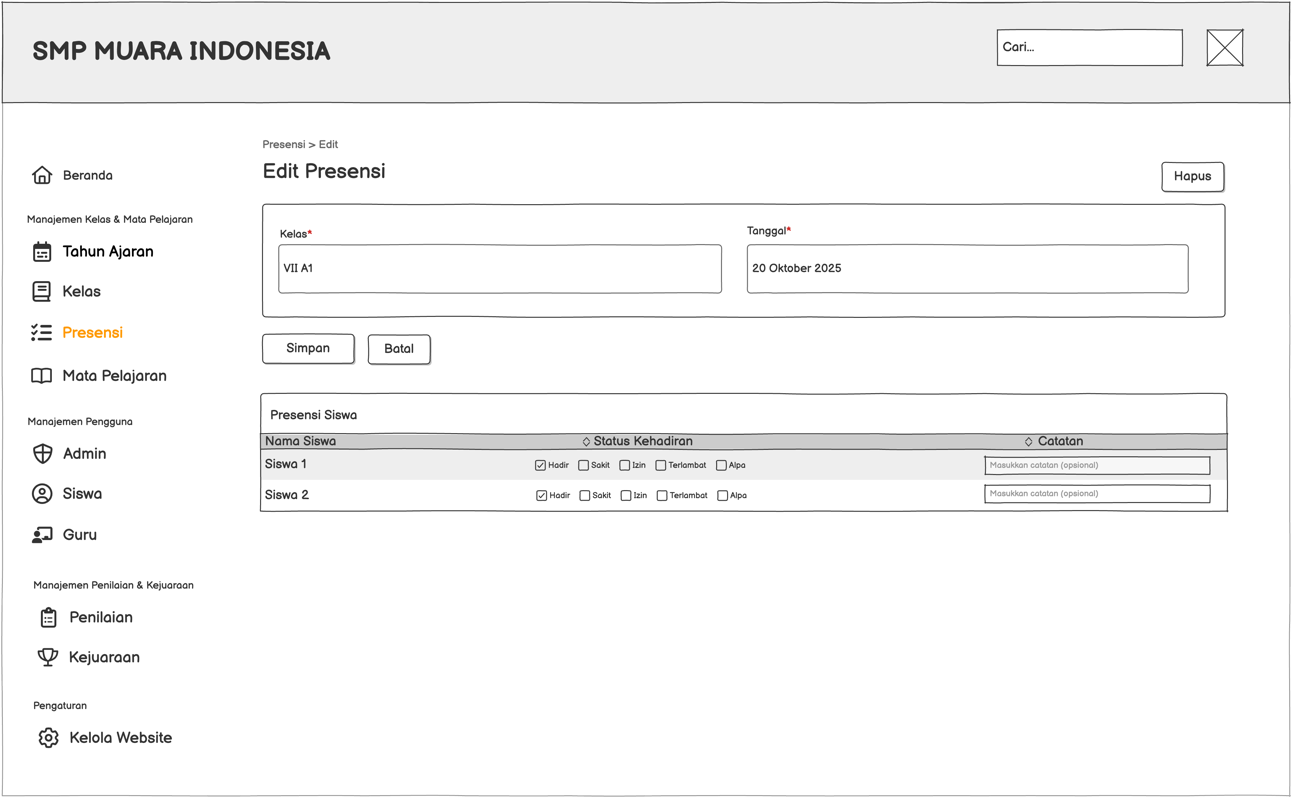
* + 1. Tambah data presensi



Gambar 4. 27 Wireframe buat presensi

Gambar 4.27 menunjukkan halaman Buat Kehadiran. Halaman ini digunakan untuk menambahkan data kehadiran baru dengan memilih kelas dan menentukan tanggal. Terdapat tombol Buat, Buat & Buat Ulang, dan Batal untuk mengelola proses penyimpanan data kehadiran.

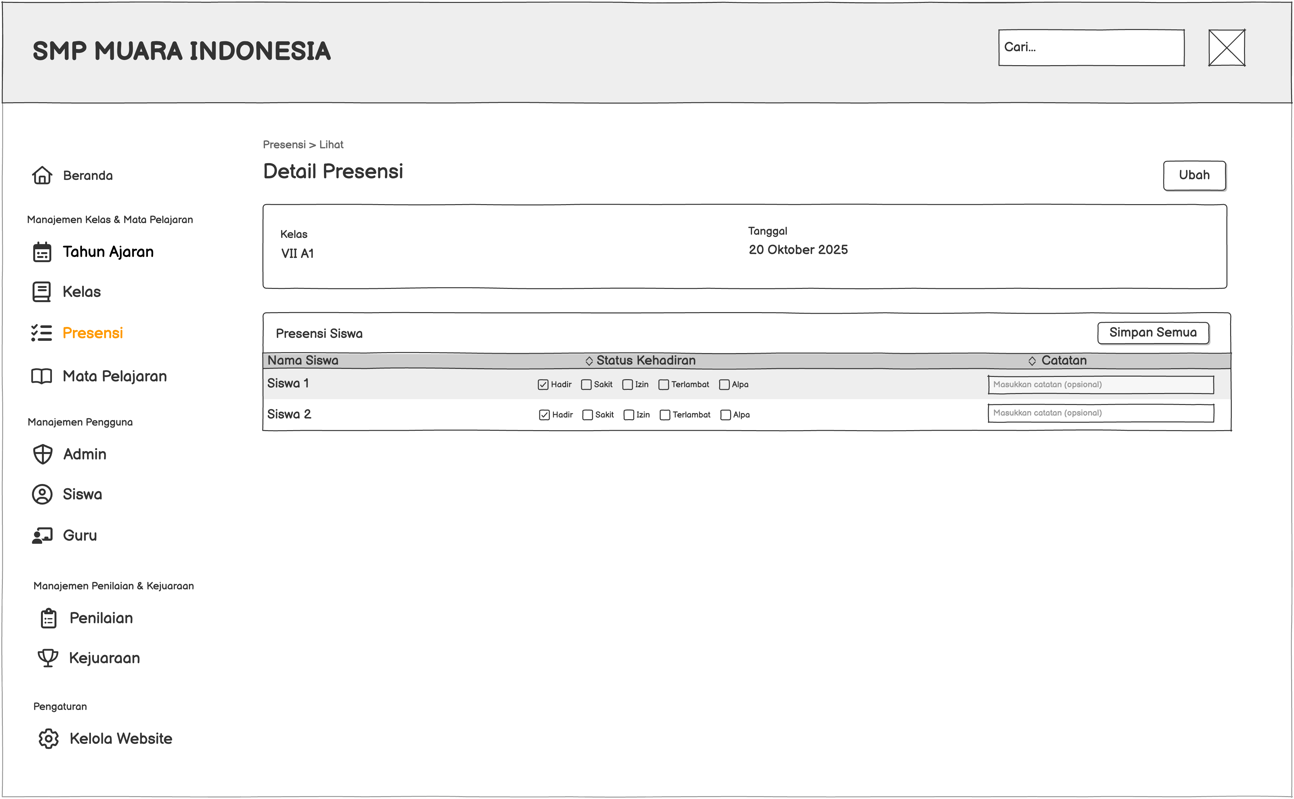
* + 1. Edit data presensi



Gambar 4. 28 Wireframe edit presensi siswa

Gambar 4.28 menunjukkan halaman edit kehadiran. Halaman ini digunakan untuk memperbarui data kehadiran siswa untuk kelas tertentu dengan menentukan status kehadiran seperti hadir, sakit, izin, terlambat, atau tidak hadir. Pengguna juga dapat menambahkan catatan tambahan untuk setiap siswa. Tombol Simpan, Batal, dan Hapus tersedia untuk mengelola data kehadiran.

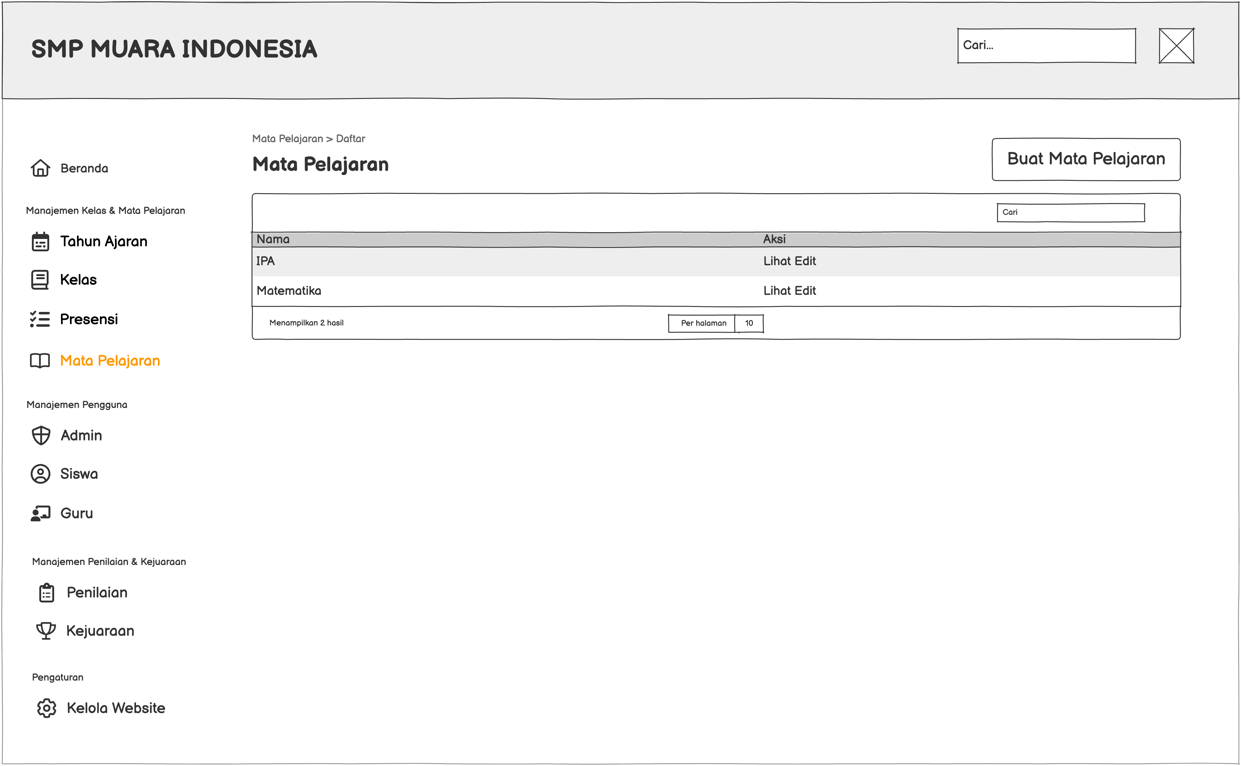
* + 1. Detail presensi siswa



Gambar 4. 29 Wireframe detail presensi siswa

Gambar 4.xx menunjukkan halaman detail kehadiran. Halaman ini menampilkan data kehadiran siswa untuk kelas dan tanggal tertentu. Informasi yang ditampilkan meliputi nama siswa, status kehadiran, dan catatan tambahan lainnya. Terdapat tombol Edit untuk memperbarui data dan Simpan Semua untuk menyimpan perubahan pada semua catatan kehadiran siswa.

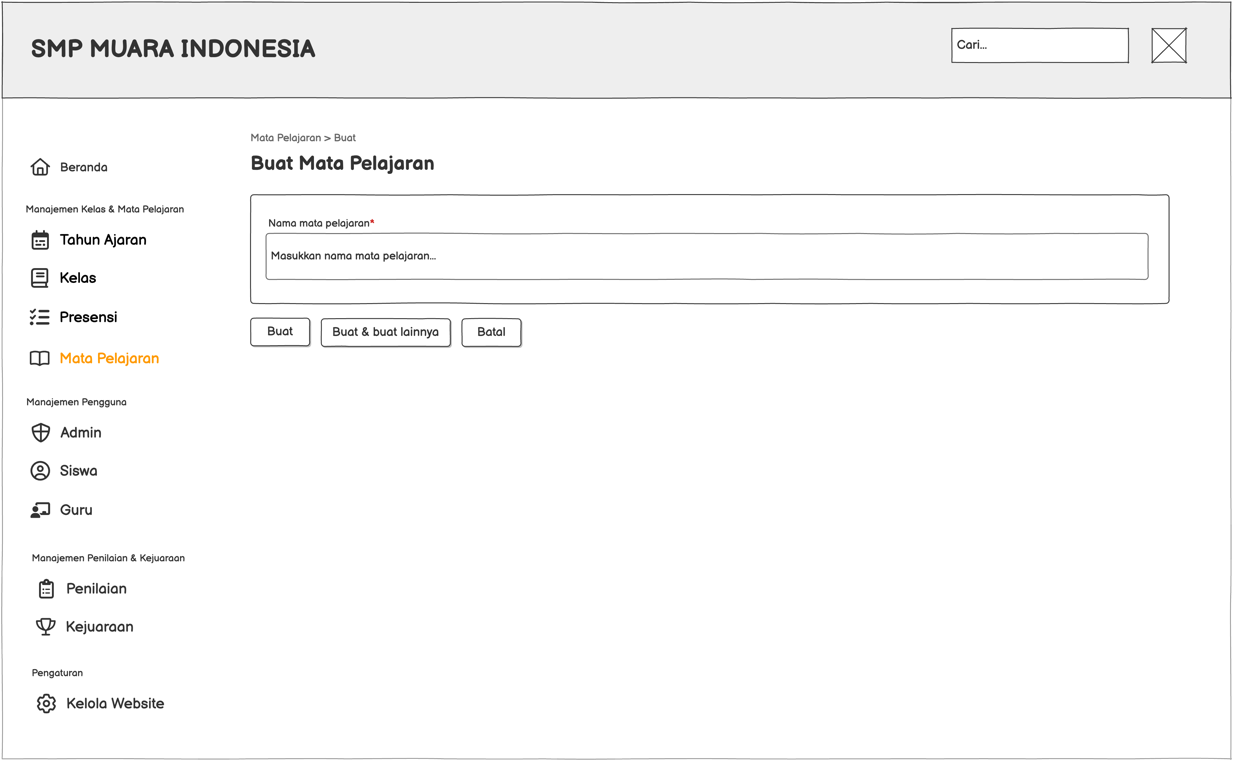
* + 1. Daftar mata pelajaran



Gambar 4. 30 Wireframe daftar mata pelajaran

Gambar 4.30 menunjukkan halaman daftar subjek. Halaman ini menampilkan semua data subjek yang tersedia di sistem. Pengguna dapat melakukan tindakan seperti melihat, mengedit, atau menambahkan subjek baru menggunakan tombol Buat Subjek.

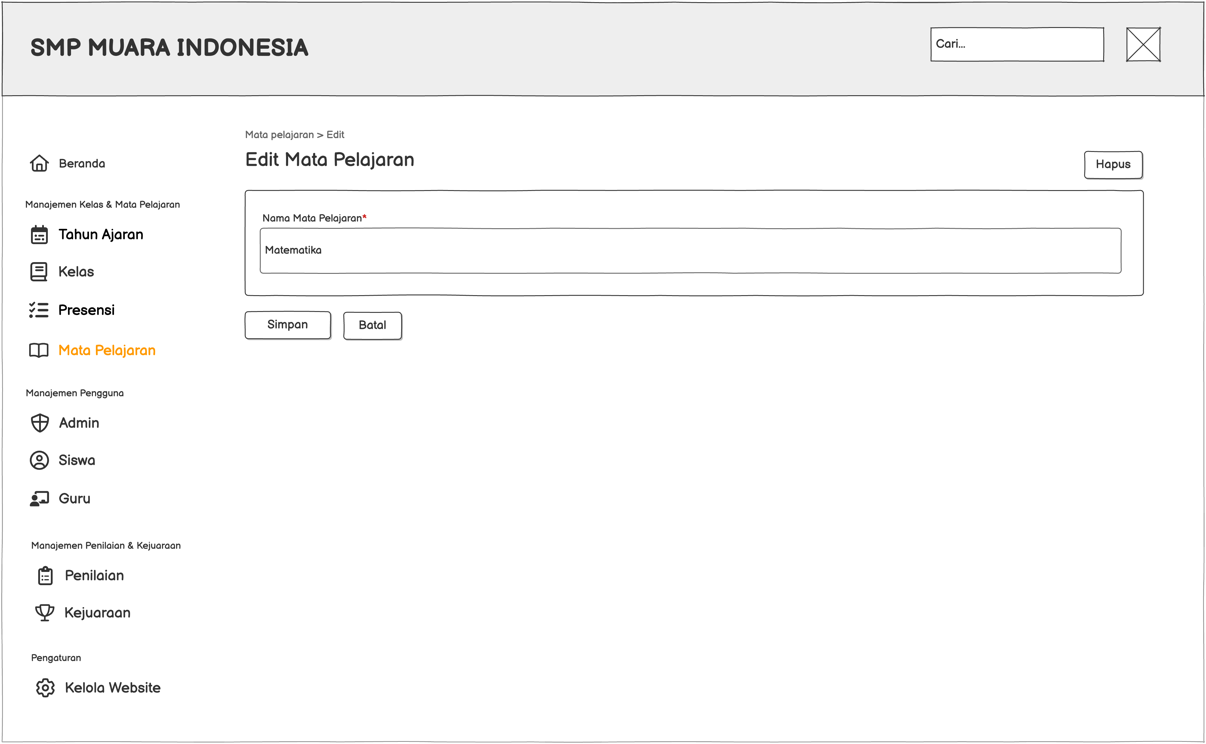
* + 1. Tambah data mata pelajaran



Gambar 4. 31 Wireframe buat mata pelajaran

Gambar 4.31 menunjukkan halaman pembuatan mata pelajaran. Halaman ini digunakan untuk menambahkan data mata pelajaran baru dengan memasukkan nama mata pelajaran. Terdapat tombol Buat, Buat & Buat Ulang, dan Batal untuk mengelola proses penyimpanan data.

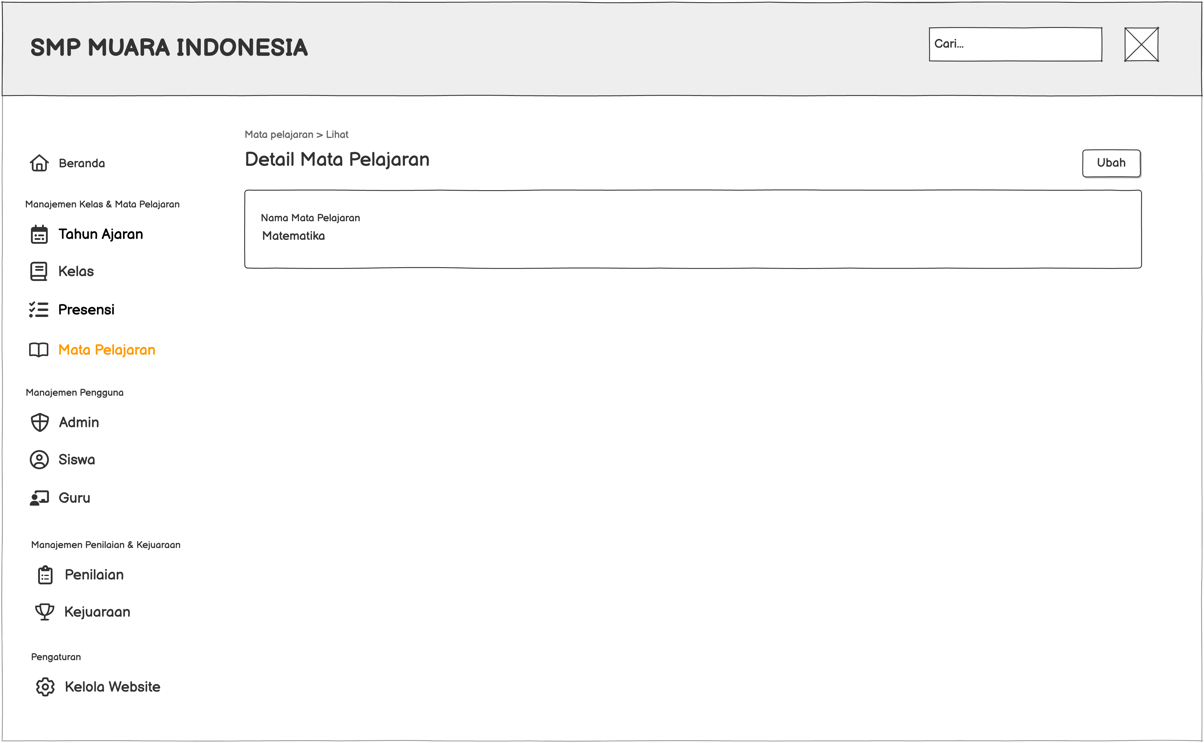
* + 1. Edit data mata pelajaran



Gambar 4. 32 Wireframe edit mata pelajaran

Gambar 4.32 menunjukkan halaman edit mata pelajaran. Halaman ini digunakan untuk memperbarui data mata pelajaran yang ada dengan mengubah nama mata pelajaran. Tombol Simpan, Batal, dan Hapus tersedia untuk mengelola perubahan data.

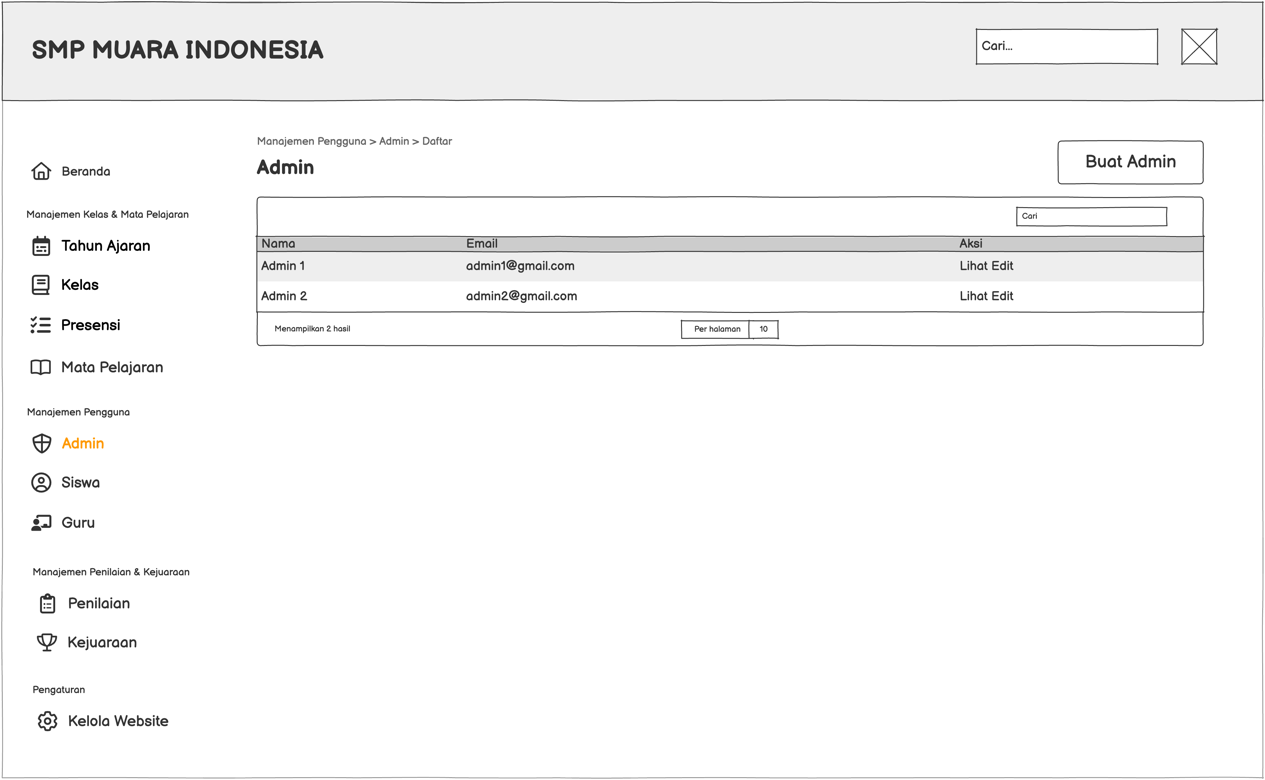
* + 1. Detail mata pelajaran



Gambar 4. 33 Wireframe detail mata pelajaran

Gambar 4.33 menunjukkan halaman detail mata pelajaran. Di halaman ini, pengguna dapat melihat informasi tentang mata pelajaran yang terdaftar dalam sistem. Tombol Edit tersedia untuk mengedit data mata pelajaran jika diperlukan.

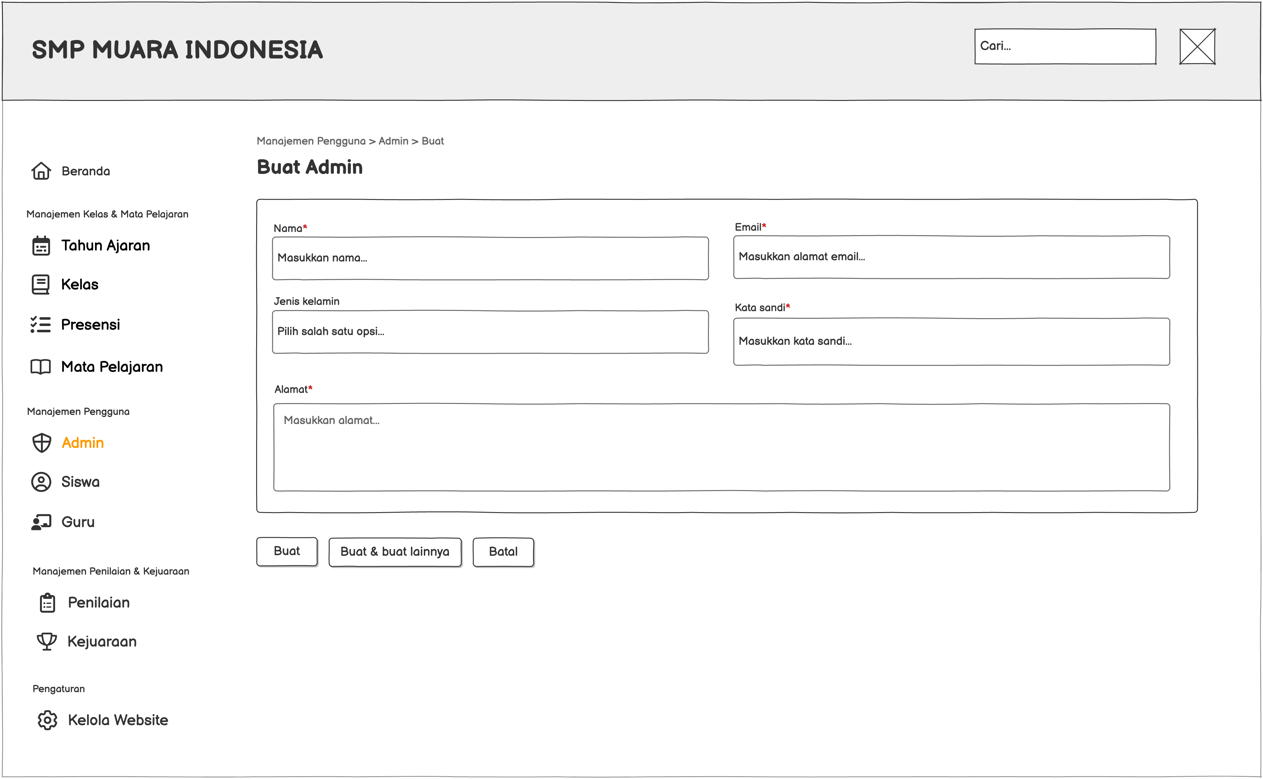
* + 1. Daftar admin



Gambar 4. 34 Wireframe daftar admin

Gambar 4.34 menunjukkan halaman daftar admin, yang menampilkan data pengguna dengan peran admin, termasuk nama dan alamat email. Halaman ini juga menyediakan fitur pencarian, serta tombol Lihat, Edit, dan Buat Admin untuk menyederhanakan pengelolaan data admin dalam sistem.

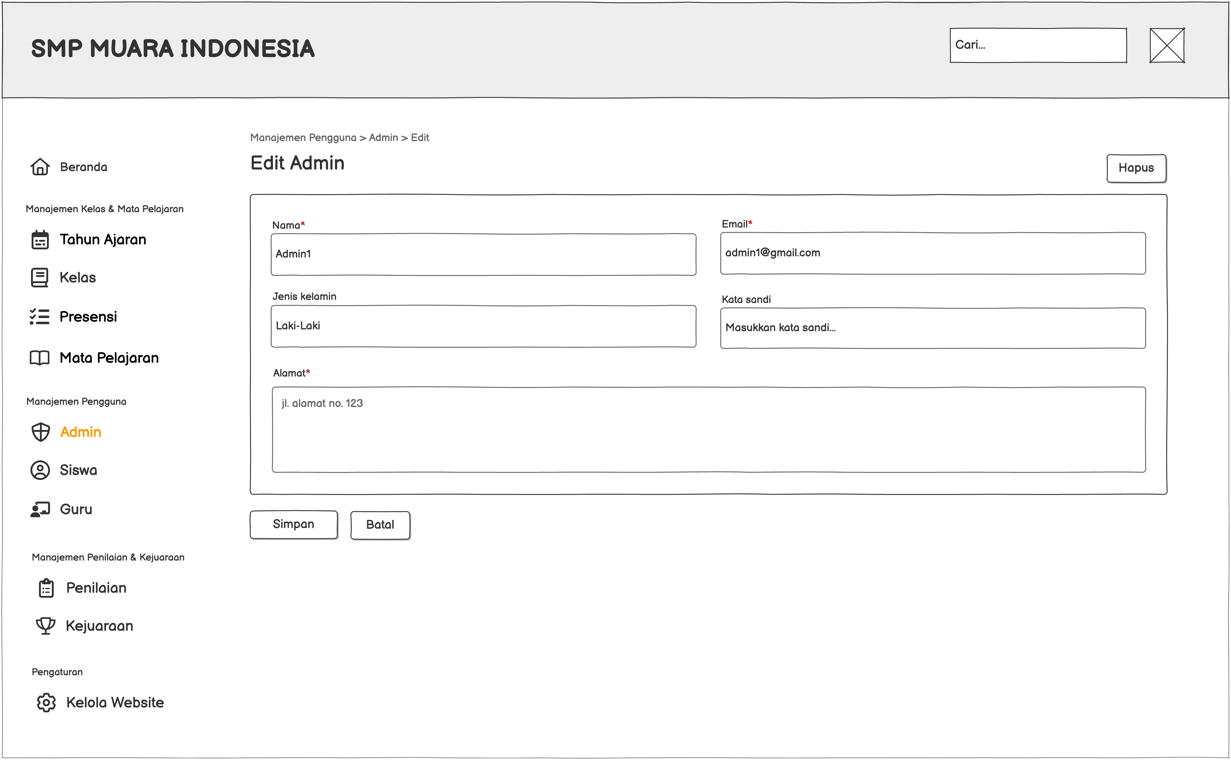
* + 1. Tambah data admin



Gambar 4. 35 Wireframe buat admin

Gambar 4.35 menunjukkan halaman pembuatan admin. Halaman ini digunakan untuk menambahkan data admin baru dengan mengisi informasi seperti nama, email, jenis kelamin, kata sandi, dan alamat. Terdapat tombol Buat, Buat & Buat Ulang, dan Batal untuk mengelola proses penyimpanan data admin.

* + 1. Edit data admin



Gambar 4. 36 Wireframe edit admin

Halaman edit admin pada Gambar 4.36 digunakan untuk memperbarui data admin yang sudah tersimpan di sistem. Melalui halaman ini, pengguna dapat mengubah informasi seperti nama, email, jenis kelamin, kata sandi, dan alamat. Selain itu, terdapat tombol untuk menyimpan perubahan, tombol Batal untuk membatalkan proses, dan tombol Hapus untuk menghapus data admin.

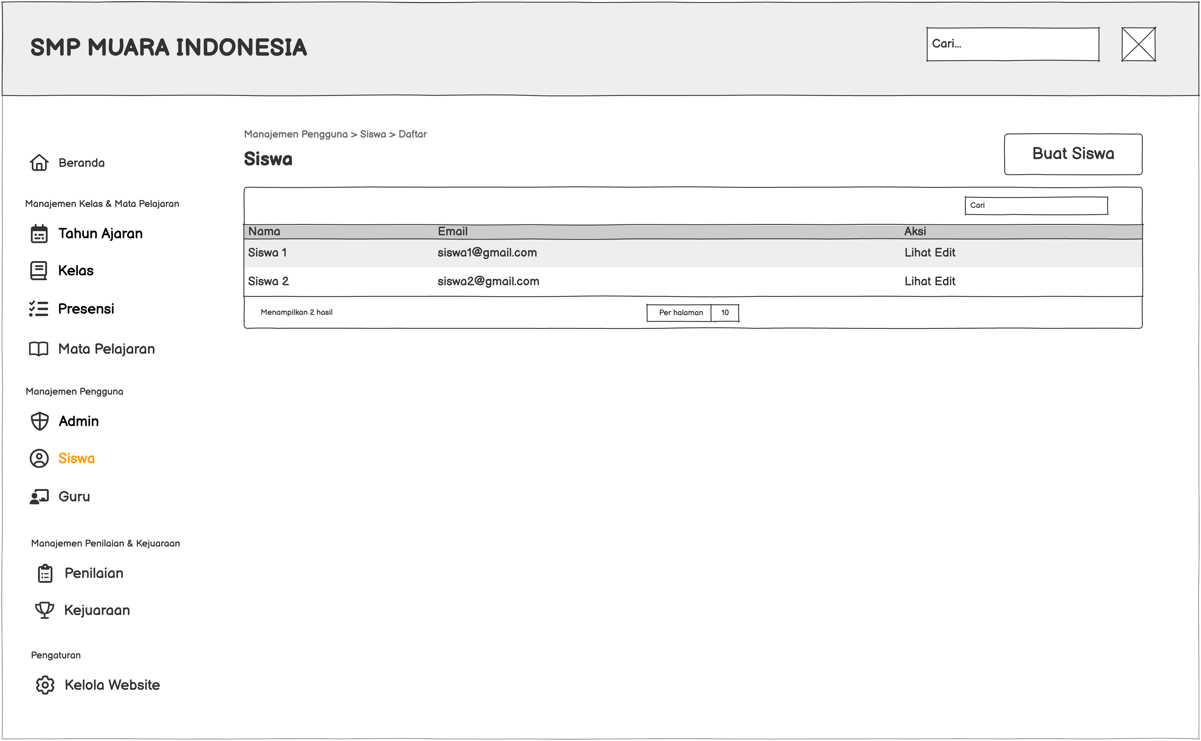
* + 1. Detail admin



Gambar 4. 37 Wireframe detail admin

Gambar 4.37 menampilkan halaman detail admin, yang menampilkan informasi lengkap tentang data admin yang tersimpan dalam sistem. Informasi yang ditampilkan meliputi nama, alamat email, jenis kelamin, dan alamat. Terdapat juga tombol Ubah untuk mengarahkan pengguna ke halaman edit jika perlu memperbarui data.

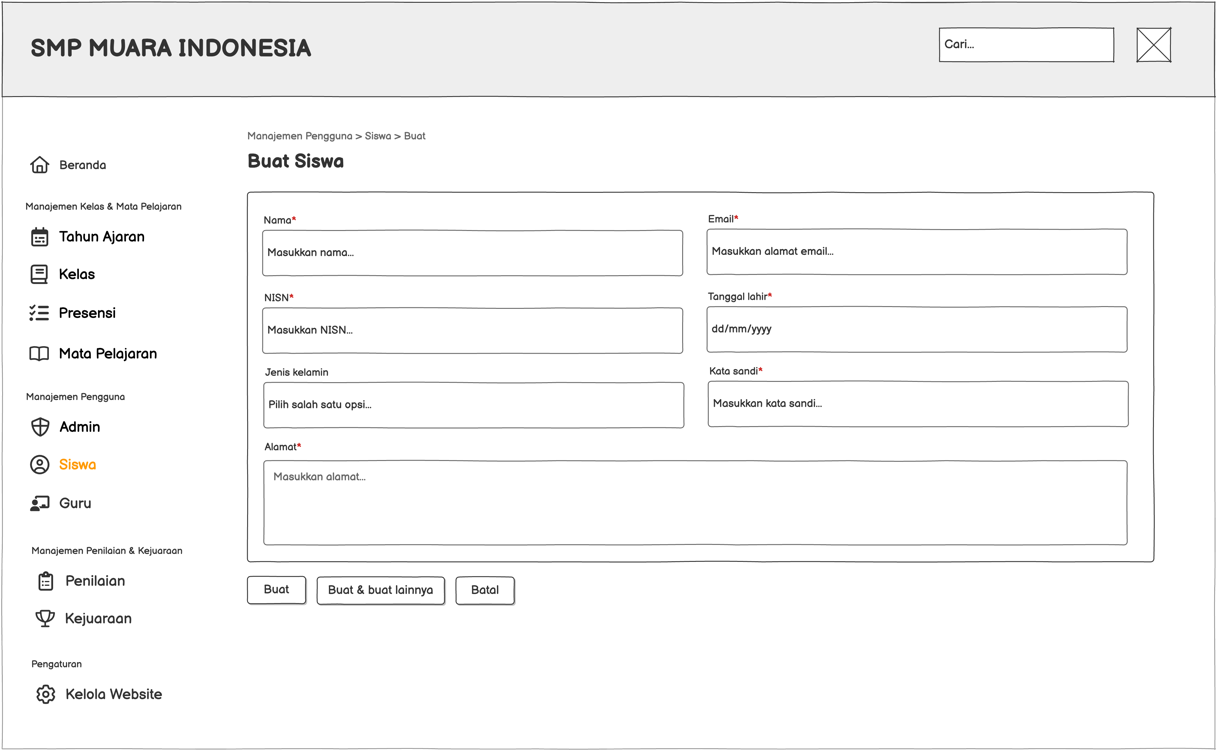
* + 1. Daftar Siswa



Gambar 4. 38 Wireframe daftar siswa

Halaman daftar siswa, seperti yang terlihat pada Gambar 4.38, menampilkan data semua siswa yang terdaftar dalam sistem. Informasi yang ditampilkan meliputi nama dan alamat email siswa. Pengguna dapat melakukan tindakan seperti melihat, mengedit, atau menambahkan data baru menggunakan tombol Buat Siswa.

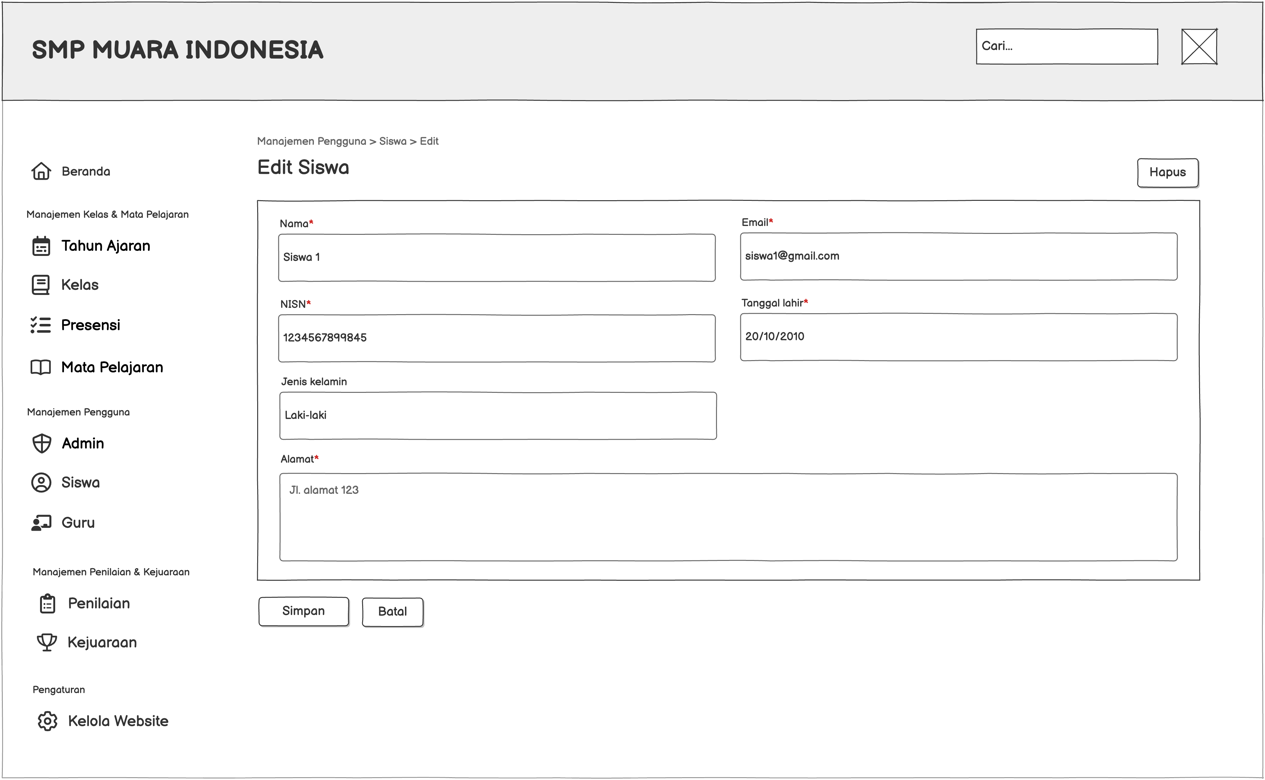
* + 1. Tambah data siswa



Gambar 4. 39 Wireframe buat siswa

Gambar 4.39 menunjukkan halaman pembuatan data siswa, yang digunakan untuk menambahkan data siswa baru ke dalam sistem. Di halaman ini, pengguna dapat memasukkan data penting seperti nama, email, NISN, tanggal lahir, jenis kelamin, kata sandi, dan alamat. Terdapat tombol untuk membuat, membuat & membuat ulang, dan membatalkan untuk memproses atau membatalkan penyimpanan data.

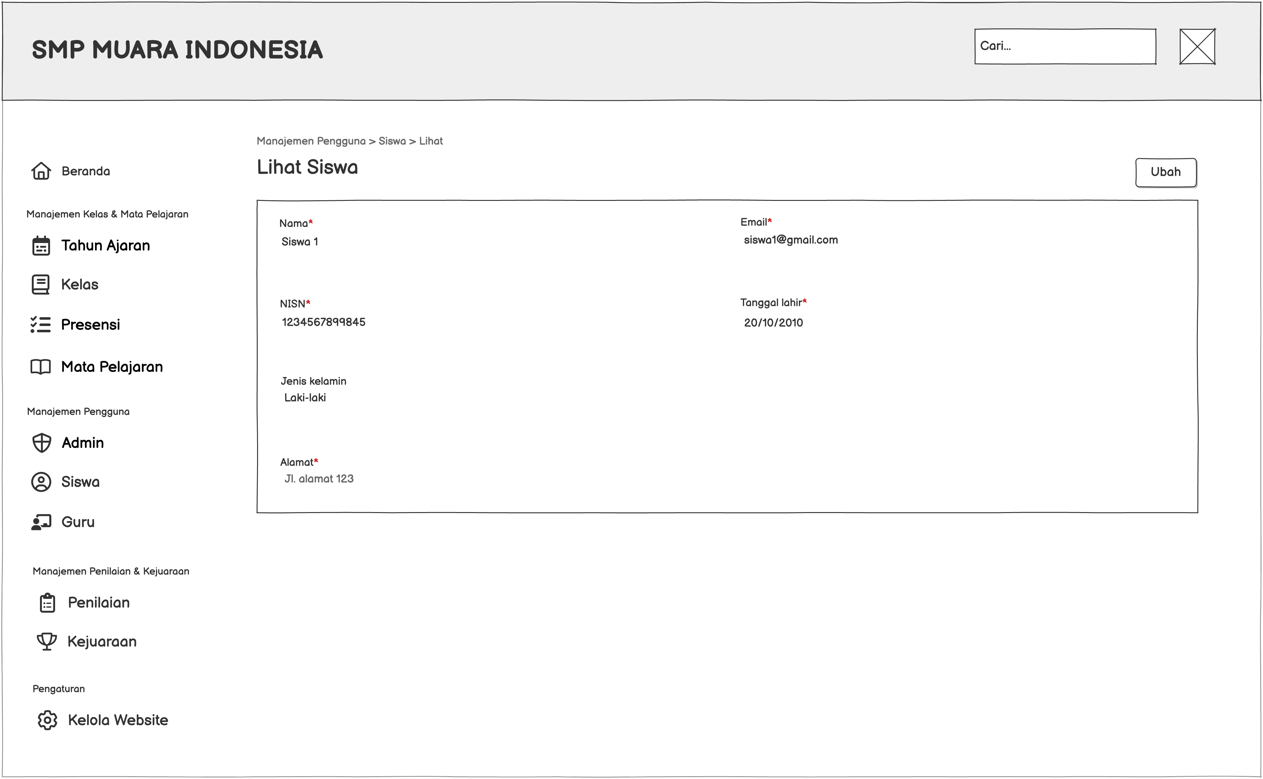
* + 1. Edit data siswa



Gambar 4. 40 Wireframe edit siswa

Halaman edit siswa pada Gambar 4.40 digunakan untuk memperbarui data siswa yang sudah terdaftar di sistem. Pengguna dapat mengubah informasi seperti nama, alamat email, Nomor Induk Siswa Nasional (NISN), tanggal lahir, jenis kelamin, dan alamat. Selain itu, terdapat tombol Simpan, Batal, dan Hapus untuk mengelola perubahan data siswa sesuai kebutuhan.

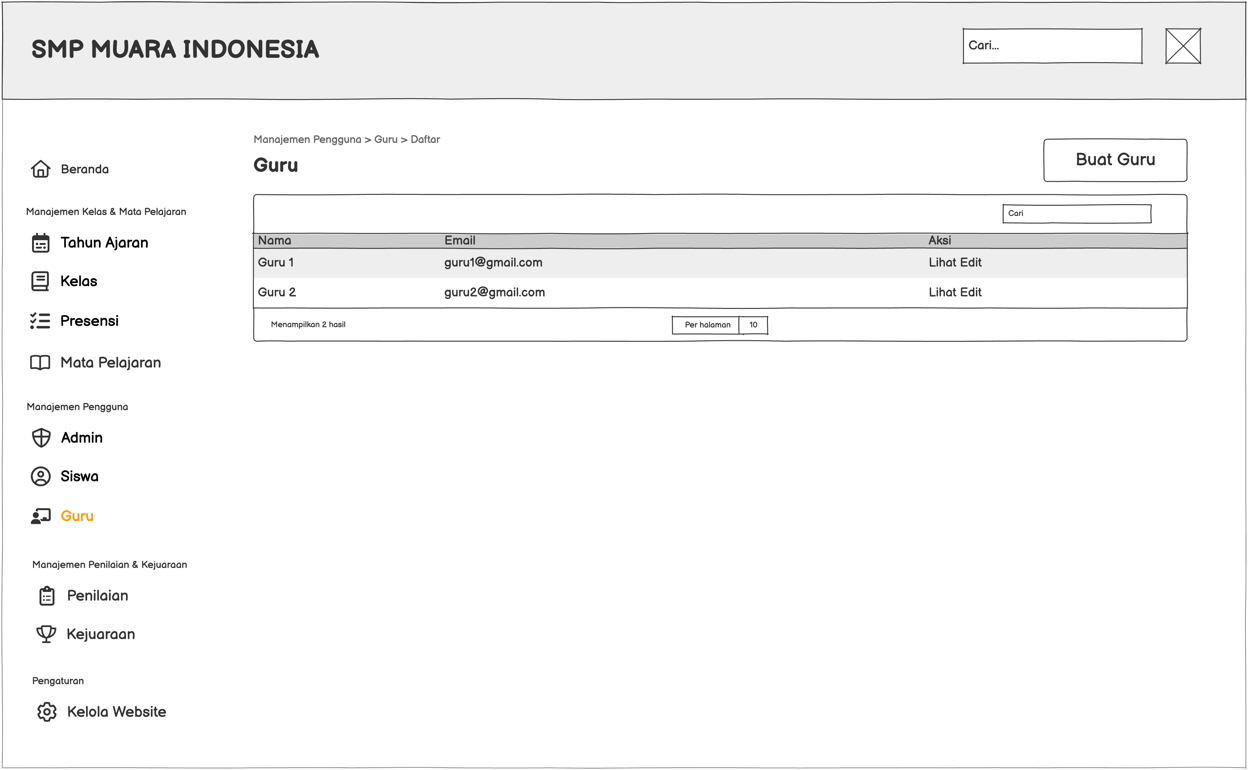
* + 1. Detail siswa



Gambar 4. 41 Wireframe detail siswa

Gambar 4.41 menampilkan halaman tampilan siswa, yang menampilkan data siswa lengkap yang tersimpan dalam sistem. Informasi yang ditampilkan meliputi nama, alamat email, Nomor Induk Siswa Nasional (NISN), tanggal lahir, jenis kelamin, dan alamat. Tombol Ubah tersedia untuk mengarahkan pengguna ke halaman edit jika diperlukan pembaruan data.

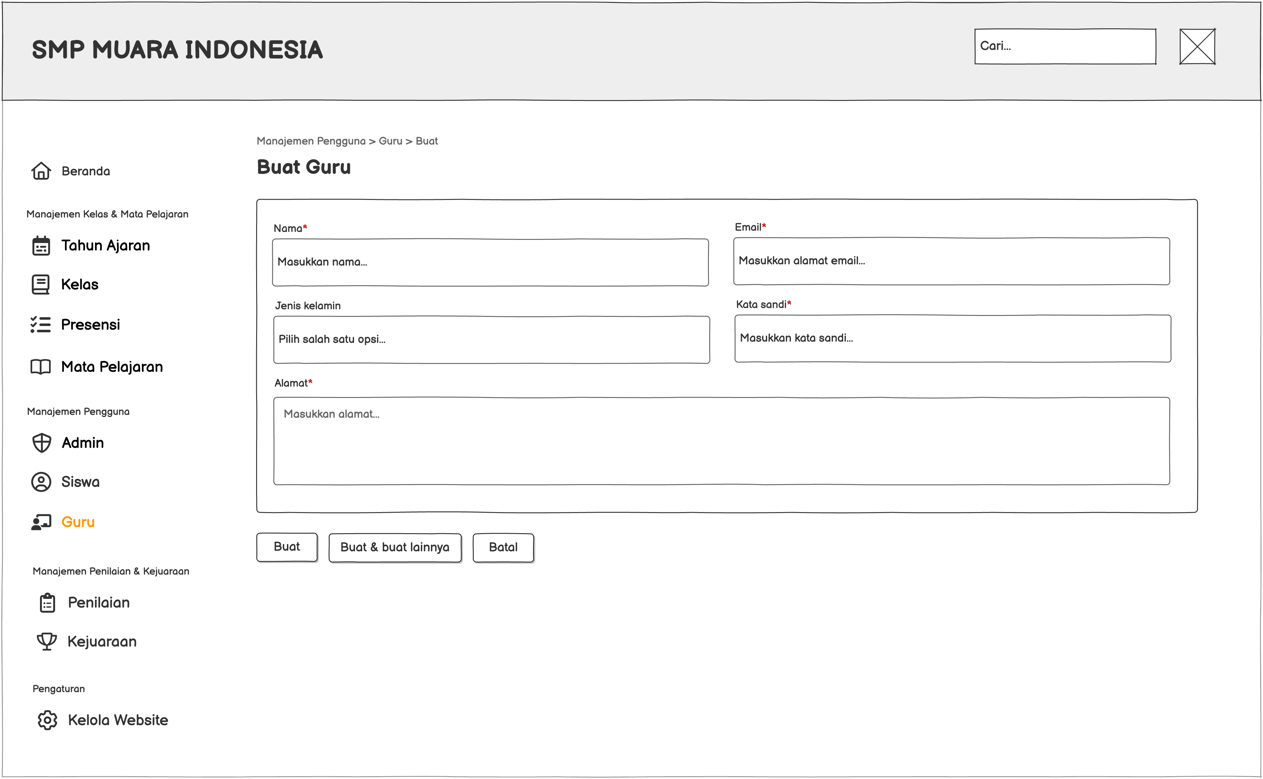
* + 1. Daftar guru



Gambar 4. 42 Wireframe daftar guru

Halaman daftar guru, seperti yang terlihat pada Gambar 4.42, menampilkan data guru yang terdaftar dalam sistem, termasuk nama dan alamat email setiap guru. Pengguna dapat melakukan tindakan seperti melihat, mengedit, atau menambahkan data baru menggunakan tombol Buat Guru yang terletak di kanan atas halaman.

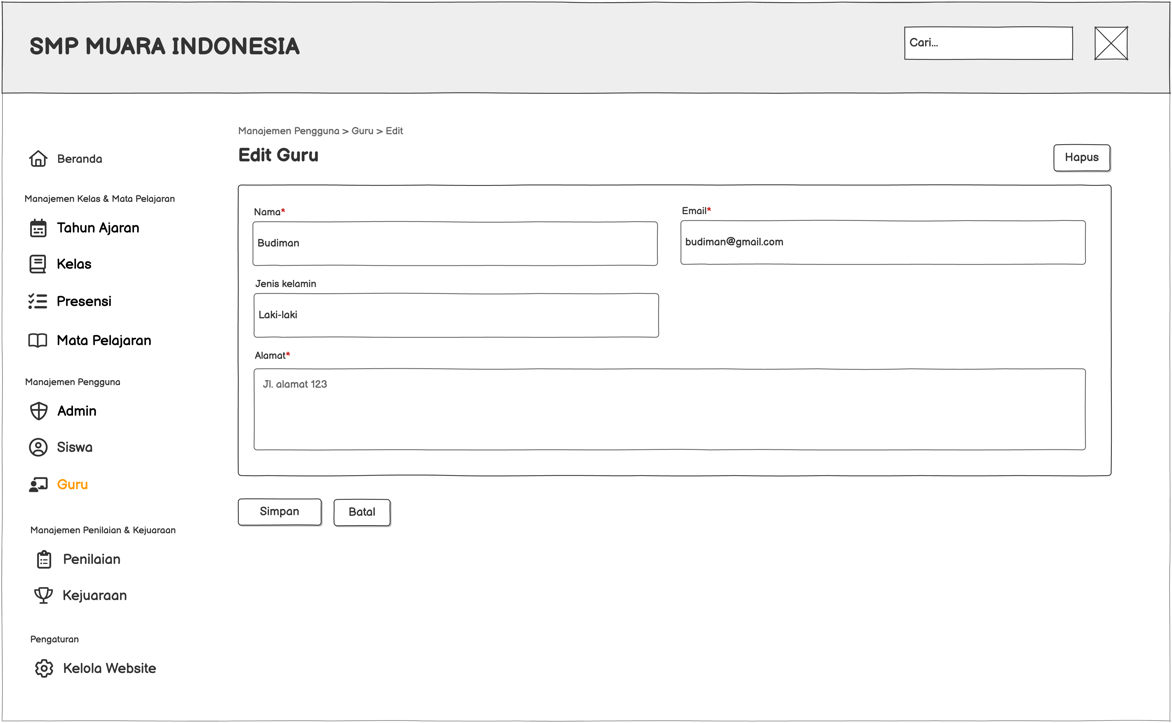
* + 1. Tambah data guru



Gambar 4. 43 Wireframe buat guru

Gambar 4.43 menampilkan halaman Buat Guru, yang memungkinkan pengguna menambahkan data guru baru ke sistem. Pengguna dapat memasukkan informasi seperti nama, alamat email, jenis kelamin, kata sandi, dan alamat. Halaman ini juga menyediakan tombol Buat, Buat & buat lagi, dan Batal untuk mengelola proses penyimpanan data guru.

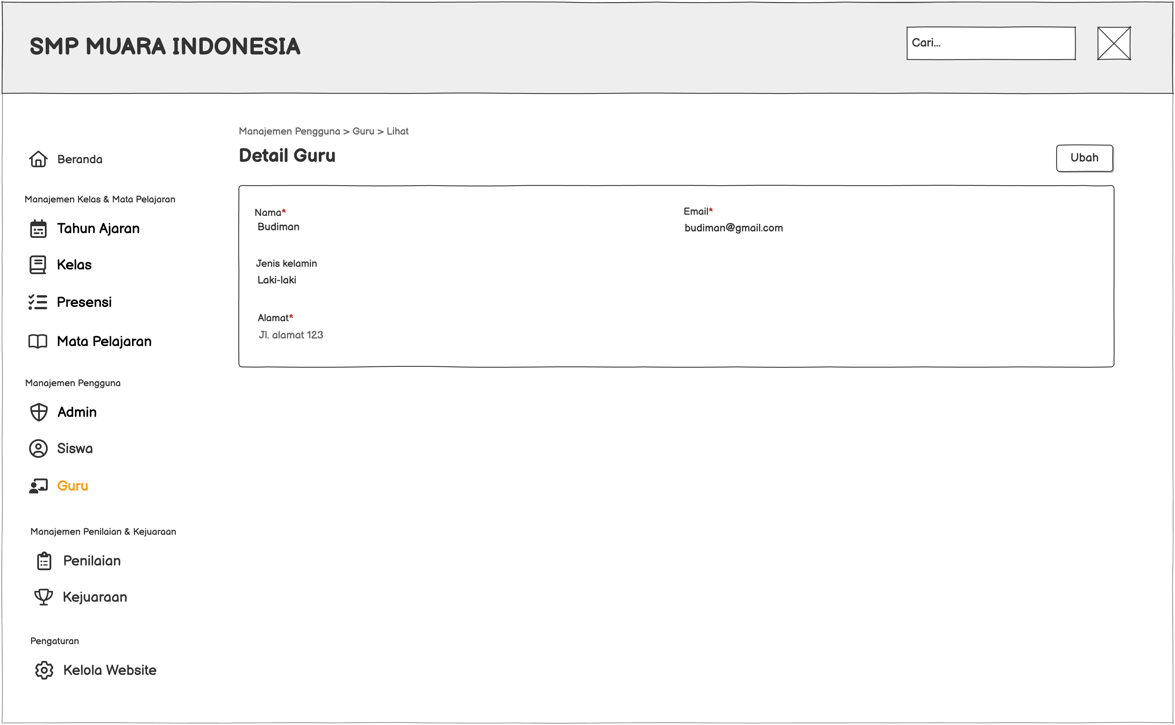
* + 1. Edit data guru



Gambar 4. 44 Wireframe edit guru

Gambar 4.44 menunjukkan wireframe halaman edit guru, yang digunakan untuk memperbarui data guru, seperti nama, alamat email, jenis kelamin, dan alamat. Halaman ini juga dilengkapi tombol Simpan, Batal, dan Hapus untuk mengelola perubahan data guru.

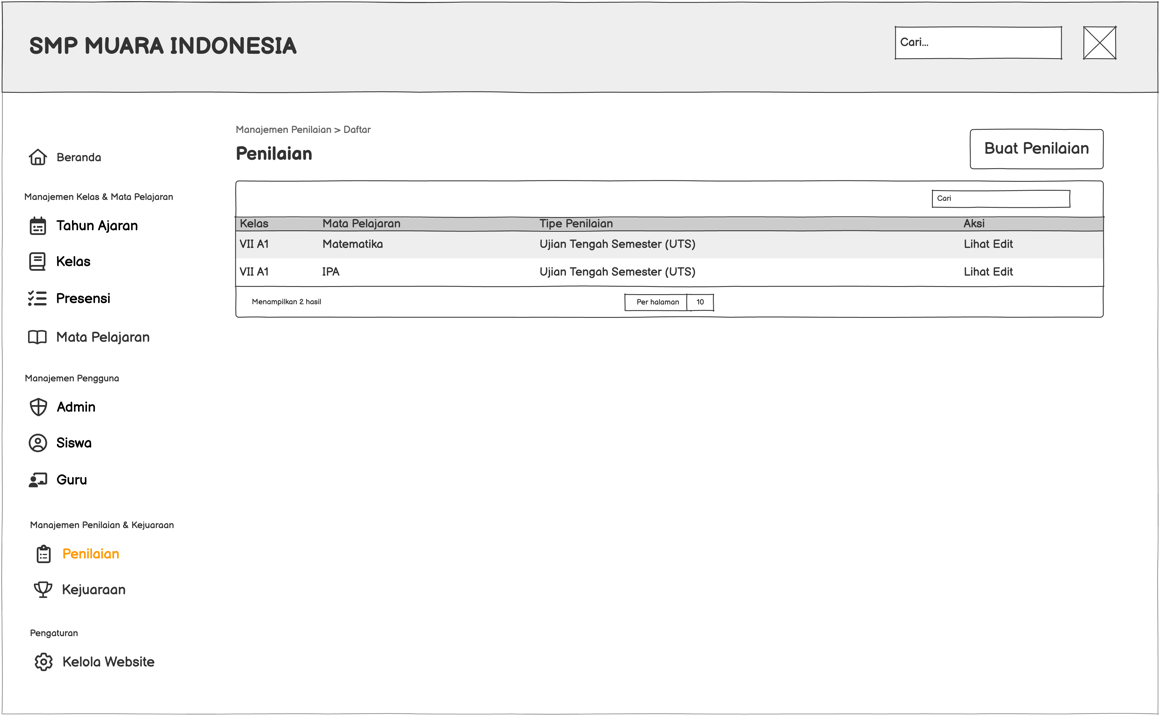
* + 1. Detail guru



Gambar 4. 45 Wireframe detail guru

Gambar 4.45 menunjukkan wireframe halaman detail guru, yang menampilkan informasi lengkap guru, seperti nama, alamat email, jenis kelamin, dan alamat. Halaman ini menyediakan tombol Edit yang memungkinkan pengguna menavigasi ke halaman edit jika diperlukan pembaruan data.

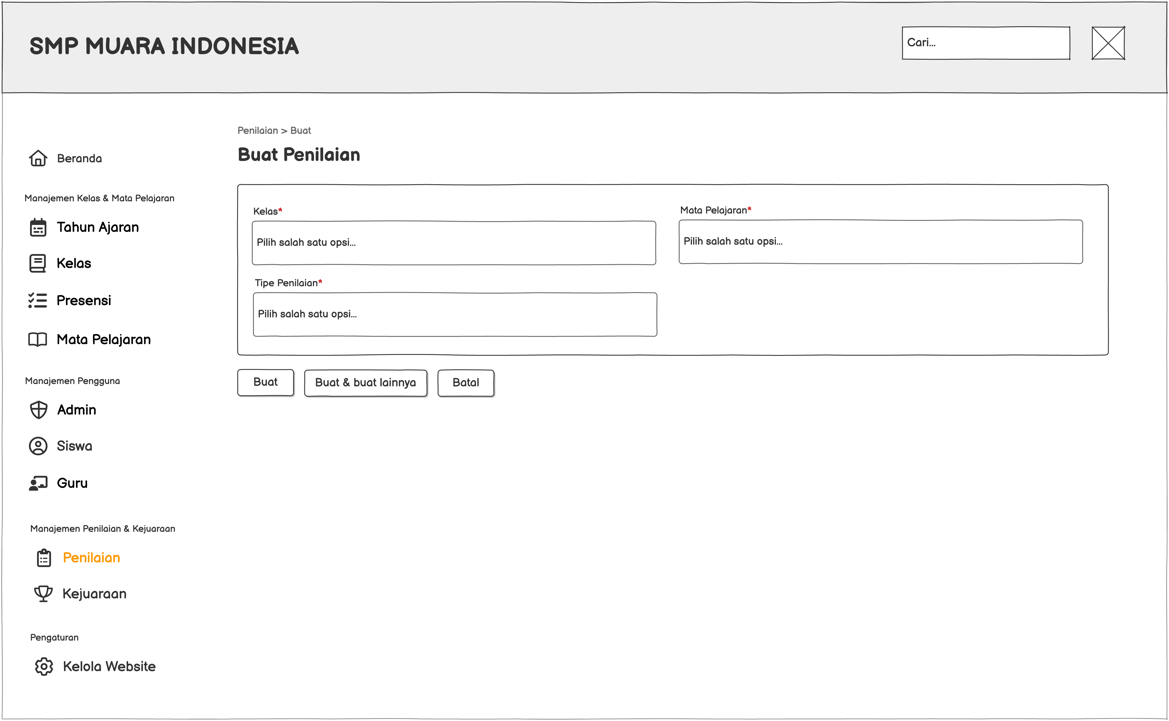
* + 1. Daftar Penilaian



Gambar 4. 46 Wireframe daftar penilaian

Gambar 4.46 menunjukkan wireframe halaman daftar penilaian, yang menampilkan data penilaian berdasarkan jenjang, mata pelajaran, dan jenis penilaian. Halaman ini dilengkapi fitur pencarian, navigasi halaman, dan tombol Buat Penilaian untuk menambahkan data penilaian baru. Setiap baris penilaian menyediakan opsi Lihat dan Edit untuk mengelola informasi penilaian yang tercatat.

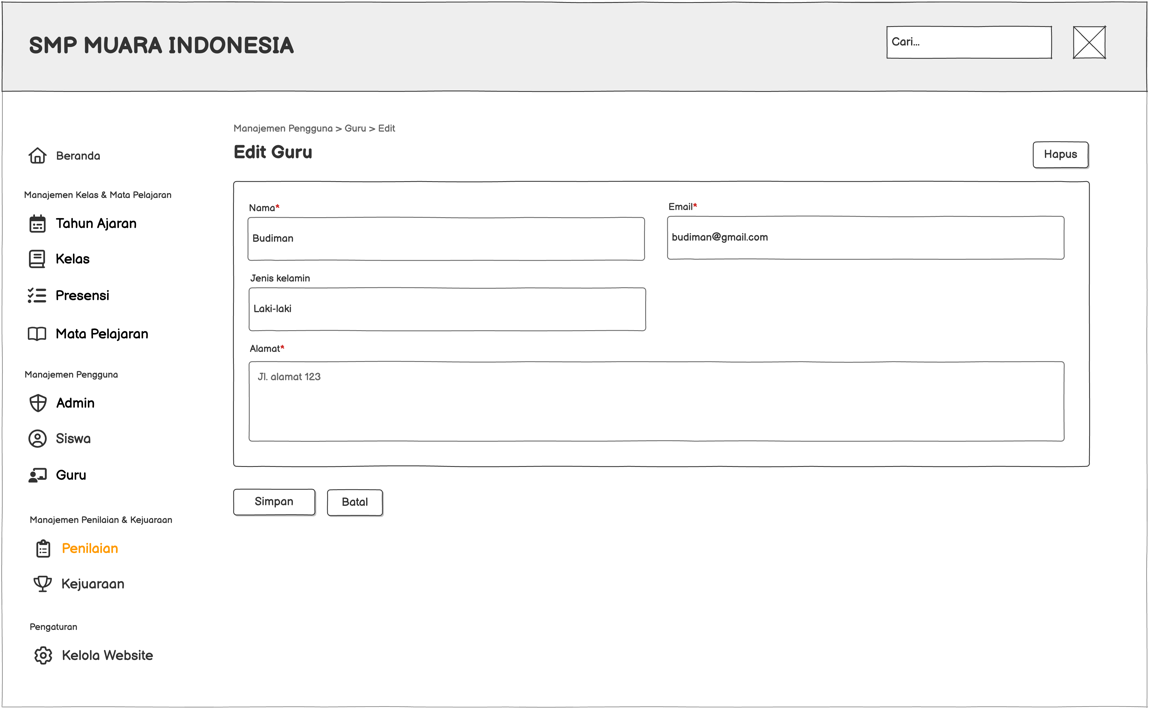
* + 1. Tambah Penilaian



Gambar 4. 47 Wireframe tambah penilaian

Gambar 4.47 menunjukkan wireframe halaman Tambah Penilaian, yang digunakan untuk membuat data penilaian baru berdasarkan nilai, mata pelajaran, dan jenis penilaian. Formulir di halaman ini menyediakan opsi nilai, mata pelajaran, dan jenis penilaian yang harus diisi sebelum menyimpan. Terdapat juga tombol Buat, Buat & Buat Ulang, dan Batal untuk mengelola proses pembuatan penilaian.

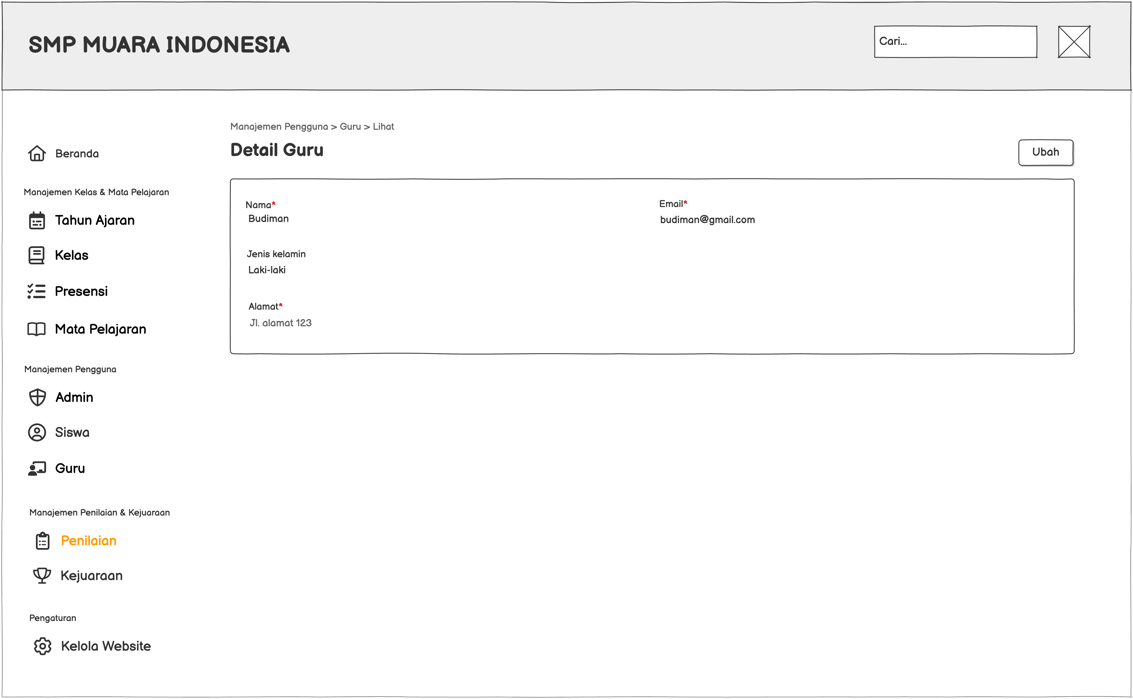
* + 1. Edit Penilaian



Gambar 4. 48 Wireframe edit penilaian

Gambar 4.48 menunjukkan garis besar halaman edit penilaian yang digunakan untuk memperbarui data penilaian yang telah dibuat sebelumnya. Di halaman ini, pengguna dapat mengubah informasi penilaian, seperti nilai, mata pelajaran, dan jenis penilaian. Formulir edit dilengkapi tombol Simpan untuk menyimpan perubahan dan tombol Batal untuk membatalkan proses edit.

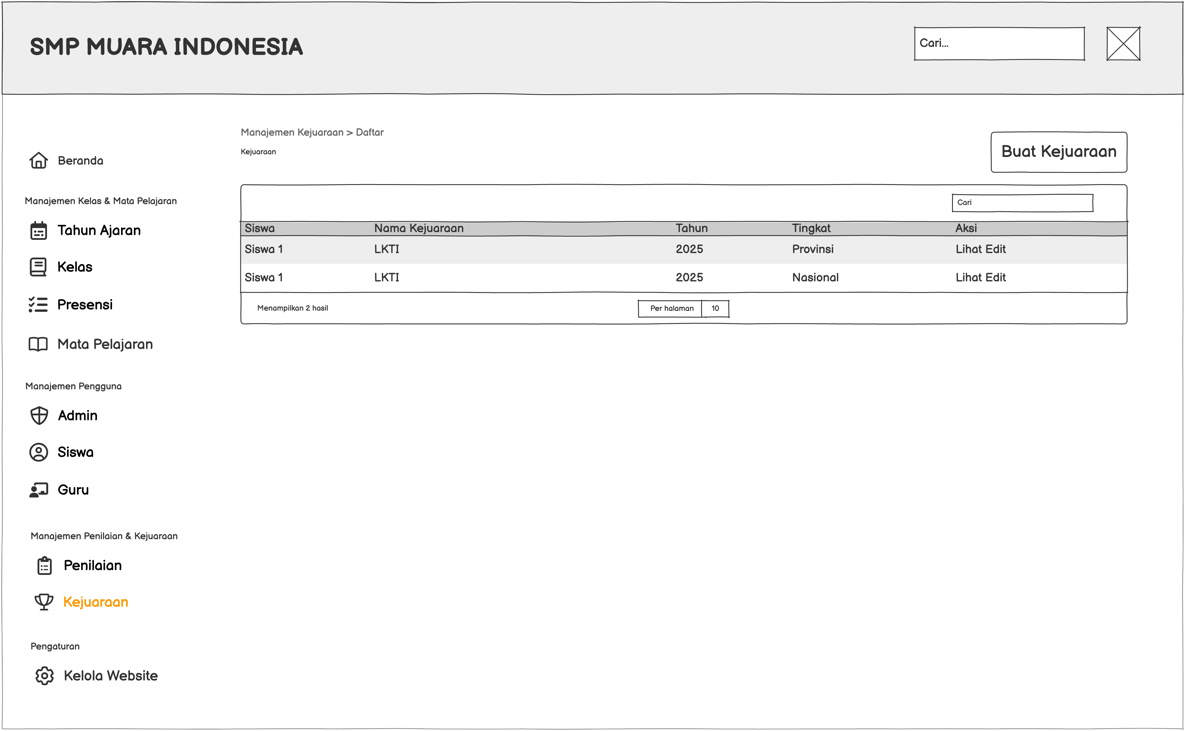
* + 1. Detail Penilaian



Gambar 4. 49 Wirefram detail penilaian

Gambar 4.49 menunjukkan wireframe halaman detail penilaian, yang digunakan untuk menampilkan informasi lengkap tentang suatu penilaian. Informasi yang ditampilkan meliputi nilai, mata pelajaran, dan jenis penilaian yang diberikan. Halaman ini juga menyediakan tombol Edit untuk mengarahkan pengguna ke halaman edit jika diperlukan perubahan pada data penilaian.

* + 1. Daftar kejuaraan



Gambar 4. 50 Wireframe daftar kejuaraan

Gambar 4.50 menunjukkan wireframe halaman daftar kejuaraan yang digunakan untuk menampilkan data kejuaraan yang telah diikuti siswa. Informasi yang ditampilkan meliputi nama siswa, nama kejuaraan, tahun kompetisi, dan tingkat kejuaraan. Halaman ini dilengkapi fitur pencarian, navigasi halaman, dan tombol Buat Kejuaraan untuk menambahkan data kejuaraan baru. Setiap entri menyediakan tindakan Lihat dan Edit untuk memudahkan pengelolaan data kejuaraan.

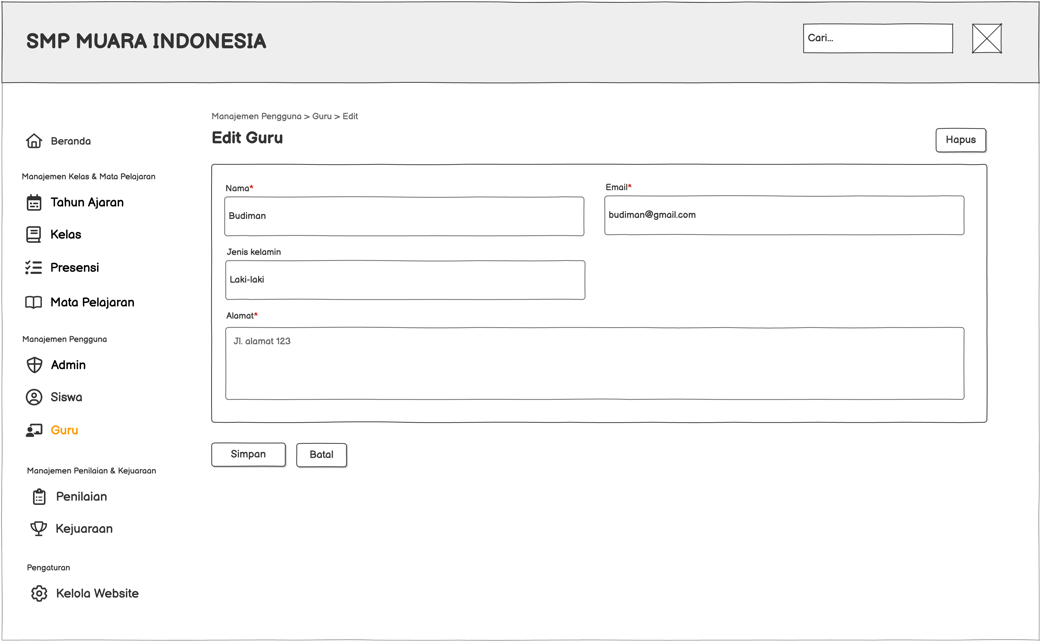
* + 1. Tambah kejuaraan



Gambar 4. 51 Wireframe tambah kejuaraan

Gambar 4.51 menunjukkan wireframe halaman Tambah Kejuaraan, yang digunakan untuk menambahkan data kejuaraan baru bagi mahasiswa. Formulir ini memungkinkan pengguna untuk memasukkan informasi seperti nama mahasiswa, nama kejuaraan, tahun kompetisi, tingkat kejuaraan, dan posisi yang diraih. Di bagian bawah halaman terdapat tombol Buat, Buat & Buat Ulang, dan Batal untuk mengelola proses penambahan data kejuaraan.

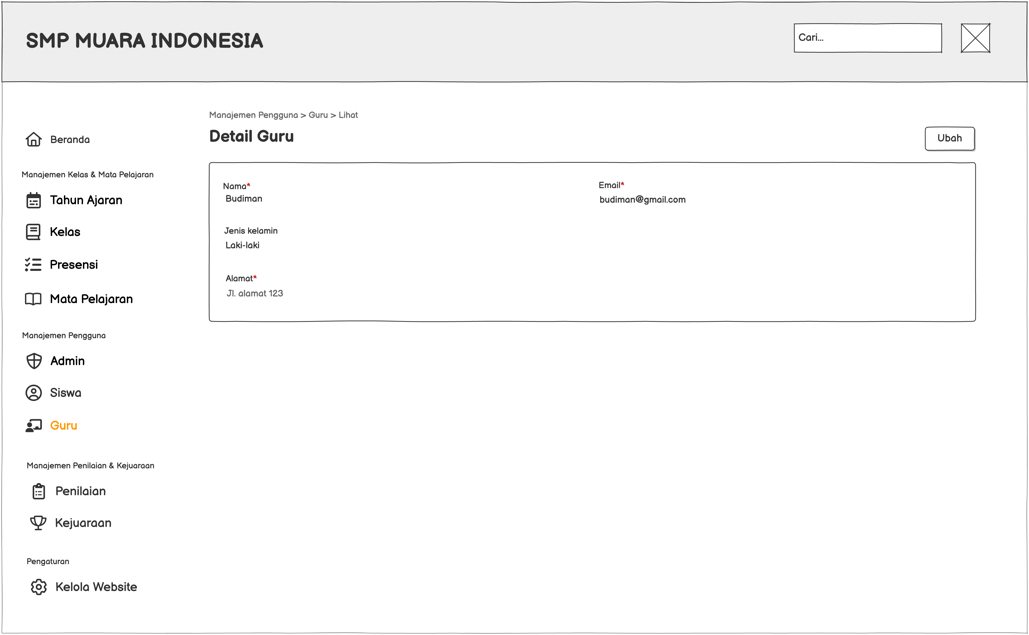
* + 1. Edit kejuaraan



Gambar 4. 52 Wireframe edit kejuaraan

Gambar 4.52 menunjukkan wireframe halaman edit kejuaraan, yang digunakan untuk memperbarui data kejuaraan yang telah direkam sebelumnya. Di halaman ini, pengguna dapat mengedit informasi kejuaraan seperti nama siswa, nama kejuaraan, tahun, tingkat kejuaraan, dan posisi yang diraih. Terdapat tombol Simpan untuk menyimpan perubahan dan tombol Batal untuk membatalkan proses pengeditan.

* + 1. Detail kejuaraan



Gambar 4. 53 Wireframe detail kejuaraan

Gambar 4.53 menunjukkan wireframe halaman detail kejuaraan, yang digunakan untuk menyajikan informasi lengkap tentang kejuaraan yang telah diikuti siswa. Informasi yang ditampilkan meliputi nama siswa, nama kejuaraan, tahun kompetisi, tingkat kejuaraan, dan posisi yang diraih. Selain itu, terdapat tombol Edit yang memungkinkan pengguna untuk mengedit data jika diperlukan.

1. **Hasil Pengembangan Sistem (Development)**

Tahap pengembangan (development) merupakan tahapan implementasi rancangan sistem menjadi bentuk aplikasi nyata yang dapat dijalankan. Tahap ini dilakukan berdasarkan hasil perencanaan dan rancangan yang telah disusun pada tahap sebelumnya. Seluruh fitur yang telah ditentukan dalam product backlog dikembangkan secara bertahap sesuai dengan prioritas dan jadwal sprint yang telah ditetapkan dalam tahap planning.

Pada metode Agile Development, proses pengembangan dilakukan secara iteratif dan inkremental. Setiap iterasi menghasilkan versi sistem yang dapat diuji dan dievaluasi secara langsung oleh pengguna untuk memastikan kesesuaian dengan kebutuhan yang sebenarnya. Dengan pendekatan ini, pengembang dapat menyesuaikan fitur berdasarkan umpan balik dari pengguna pada setiap sprint review

1. **Hasil Pengujian Sistem (Testing)**

Tahap pengujian (testing) merupakan tahapan penting dalam proses pengembangan sistem yang bertujuan untuk memastikan bahwa seluruh fitur dan fungsi yang telah dibangun berjalan sesuai dengan kebutuhan pengguna. Pengujian dilakukan untuk mendeteksi kesalahan (error), memastikan sistem berfungsi dengan benar, dan menilai tingkat kemudahan penggunaan (usability) dari sistem pengelolaan nilai dan absensi berbasis web di SMP Muara Indonesia.

Pada penelitian ini, pengujian dilakukan dengan dua metode utama, yaitu pengujian fungsional menggunakan metode Black Box Testing dan pengujian usability menggunakan System Usability Scale (SUS).

* + 1. Pengujian Fungsional (Black Box Testing)
    2. Pengujian Kegunaan (System Usability Scale - SUS)

1. **Hasil Implementasi (Deployment) dan Tinjauan (Review)**

Tahap implementasi (deployment) dan tinjauan (review) merupakan tahap akhir dalam proses pengembangan sistem menggunakan metode Agile Development. Pada tahap ini, sistem yang telah melalui proses pengujian dinyatakan layak untuk diterapkan di lingkungan pengguna sesungguhnya, dalam hal ini adalah SMP Muara Indonesia.

Tahap ini juga mencakup proses evaluasi akhir dari pengguna untuk memastikan bahwa sistem telah berjalan stabil, memenuhi kebutuhan fungsional, serta memberikan kemudahan dalam pengelolaan data akademik sekolah.

* + 1. Implementasi Sistem (Deployment)
    2. Tinjauan Sistem (Review)
    3. Evaluasi Akhir
    4. Hubungan dengan Tahapan Agile

**KESIMPULAN DAN SARAN**

1. **Kesimpulan**
2. **Saran**

# DAFTAR PUSTAKA

[1] J. K. D. A. Junaedi M, “RANCANG BANGUN SISTEM INFORMASI AKADEMIK WEBSITE UNTUK MADRASAH DINIYAH AWALIYAH UBUDIYAH,” *RANCANG BANGUN SISTEM INFORMASI AKADEMIK WEBSITE UNTUK MADRASAH DINIYAH AWALIYAH UBUDIYAH*, vol. 5, no. 1, pp. 29–38, 2023.

[2] M. Akmal, A. Abdilah, A. Nugroho, and R. Fitri, “RANCANG BANGUN LEARNING MANAGEMENT SYSTEM DI MTS MIFTAHUL HUDA,” 2025.

[3] S. Kaka *et al.*, “PENERAPAN SISTEM INFORMASI AKADEMIK PENGOLAHAN DATA PRESTASI PEMBELAJARAN SISWA SDN WERY BERBASIS WEBSITE,” 2023.

[4] F. Rahmat Halim *et al.*, “RANCANG BANGUN SISTEM INFORMASI PENGUMUMAN KELULUSAN SISWA BERBASIS WEB MENGGUNAKAN METODE AGILE WEB-BASED STUDENT GRADUATION ANNOUNCEMENT INFORMATION SYSTEM DESIGN USING THE AGILE METHOD.”

[5] I. Saputra, P. Sukmasetya, and A. Primadewi, “KLIK: Kajian Ilmiah Informatika dan Komputer Implementasi Agile Software Development dalam Perancangan Sistem Pengelolaan Limbah Sampah,” *Media Online*, vol. 4, no. 3, 2023, doi: 10.30865/klik.v4i3.1379.

[6] A. Abiyu Putra and P. Setiawati, “Perancangan Sistem Informasi Akademik Berbasis Website (Studi Kasus : SMK Negri 5 Kota Bekasi),” *Popong Setiawati INNOVATIVE: Journal Of Social Science Research*, vol. 4, pp. 9334–9347.

[7] M. Prayoga, I. Surya, and H. Kurniawan, “Rancang Bangun Sistem Informasi Sekolah Berbasis Web Dengan Menggunakan Metode Agile Pada Sdn 056001 Karang Rejo,” *Jurnal Minfo Polgan*, vol. 13, no. 1, 2024, doi: 10.33395/jmp.v13i1.14017.

[8] U. Brawijaya, M. Y. Pamungkas, A. Rachmadi, and A. D. Herlambang, “Fakultas Ilmu Komputer Rancang Bangun Sistem Informasi Akademik Berbasis Web pada Madrasah Tsanawiyah Riyadlatul Fallah,” 2017. [Online]. Available: http://j-ptiik.ub.ac.id

[9] T. Putri Rahmadani, A. Siswanto, H. Yani, and F. Ilmu Komputer, “Jurnal Informatika Dan Rekayasa Komputer (JAKAKOM) Perancangan Sistem Informasi Akademik Berbasis Web Pada SMP N 1 Muaro Jambi.”

[10] P. A. Zahid M, “RANCANG BANGUN WEBSITE ABSENSI MENGGUNAKAN RFID DAN WHATSAPP UNTUK MENINGKATKAN KEDISIPLINAN DAN NILAI RAPOR SISWA,” *Jurnal IT-EDU*, vol. 08, no. 02, pp. 9–16, 2023.

[11] A. Raya Suhari, A. Faqih, and F. M. Basysyar, “Sistem Informasi Kepegawaian Mengunakan Metode Agile Development di CV,” *Jurnal Teknologi dan Informasi*, doi: 10.34010/jati.v12i1.

[12] M. Mintarsih, “Pengujian Black Box Dengan Teknik Transition Pada Sistem Informasi Perpustakaan Berbasis Web Dengan Metode Waterfall Pada SMC Foundation,” *Jurnal Teknologi Dan Sistem Informasi Bisnis*, vol. 5, no. 1, pp. 33–35, Feb. 2023, doi: 10.47233/jteksis.v5i1.727.

[13] A. Fatmawati, “Evaluasi Usability pada Learning Management System OpenLearning Menggunakan System Usability Scale,” vol. 6, no. 1, p. 2021, [Online]. Available: https://ums.ucm.ac.id/.