

**RANCANGAN DAN IMPLEMENTASI SISTEM INFORMASI PRESENSI
DAN KETERTIBAN SISWA DI SMAN 1 SURAKARTA**



DAFFA MUDIA ATHIFAH
V3420022

**PROGRAM STUDI DIPLOMA III TEKNIK INFORMATIKA
SEKOLAH VOKASI
UNIVERSITAS SEBELAS MARET
SURAKARTA**

2023

PERSETUJUAN

**RANCANGAN DAN IMPLEMENTASI SISTEM INFORMASI PRESENSI
DAN KETERTIBAN SISWA DI SMAN 1 SURAKARTA**

Disusun Oleh :
DAFFA MUDIA ATHIFAH
V3420022

Disetujui untuk dipertahankan dihadapan Panitia Ujian Tugas Akhir
Program Studi Diploma III Teknik Informatika
Sekolah Vokasi
Universitas Sebelas Maret Surakarta

Surakarta, 27 Juni 2023

Pembimbing,

Eko Harry Pratisto, S.T., M .Info.Tech., Ph.D.
NIK. 1981112420130201

PENGESAHAN

RANCANGAN DAN IMPLEMENTASI SISTEM INFORMASI PRESENSI DAN KETERTIBAN SISWA DI SMAN 1 SURAKARTA

Tugas Akhir ini telah diuji dan disahkan oleh Panitia Ujian Tugas Akhir
Program Studi Diploma III Teknik Informatika
Sekolah Vokasi
Universitas Sebelas Maret Surakarta

Pada Hari :
Tanggal :

Disusun Oleh :
DAFFA MUDIA ATHIFAH
V3420022

Panitia Ujian Tugas Akhir	Nama	Tanda Tangan
Pembimbing	Eko Harry Pratisto, S.T., M.Info.Tech., Ph.D NIP. 1981112420130201	
Pengaji	Rudi Hartono, S.Si., M.Eng NIP. 1984122620160601	
Pengaji	Nurul Firdaus, S.Kom., M.Info.Tech NIP. 1991092320200801	

Mengetahui,

Dekan
Sekolah Vokasi

Kepala Program Studi
D3 Teknik Informatika

Drs. Santoso Tri Hananto, M.Acc., Ak. Eko Harry Pratisto, S.T., M .Info.Tech., Ph.D.
NIP. 196909241994021001 NIK. 1981112420130201

PERNYATAAN KEASLIAN KARYA TUGAS AKHIR

Dengan ini saya menyatakan bahwa dalam Laporan Tugas Akhir ini tidak terdapat karya yang pernah diajukan untuk memperoleh gelar akademik di suatu Perguruan Tinggi, dan sepanjang pengetahuan saya juga tidak terdapat karya atau pendapat yang pernah ditulis atau diterbitkan oleh orang lain, kecuali yang secara tertulis diacu dalam naskah ini dan disebutkan dalam Daftar Pustaka.

Surakarta,

DAFFA MUDIA ATHIFAH
NIM. V3420022

PERNYATAAN PERSETUJUAN PUBLIKASI UNTUK KEPENTINGAN AKADEMIS

Sebagai bagian dari sivitas akademika Universitas Sebelas Maret, saya yang bertanda tangan di bawah ini:

Nama : DAFFA MUDIA ATHIFAH

NIM : V3420022

Program Studi : Diploma 3 Teknik Informatika

menyetujui untuk memberikan Hak Bebas Royalti Non-Eksklusif (*Non-exclusive Royalty Free Right*) kepada Universitas Sebelas Maret atas karya Tugas Akhir saya yang berjudul

“RANCANGAN DAN IMPLEMENTASI SISTEM INFORMASI PRESENSI DAN KETERTIBAN SISWA DI SMAN 1 SURAKARTA”

Dengan Hak Bebas Royalti Noneksklusif ini, Universitas Sebelas Maret berhak menyimpan, mengalihmedia/formatkan, mengelola dalam bentuk pangkalan data (*database*), merawat dan memublikasikan Tugas Akhir saya, selama tetap mencantumkan nama saya sebagai penulis/pencipta dan sebagai pemilik Hak Cipta. Demikian pernyataan ini saya buat dengan sebenar-benarnya.

Surakarta,

Materei
Rp.
10.000

DAFFA MUDIA ATHIFAH

NIM. V3420022

BAB I

PENDAHULUAN

1.1 Latar Belakang

Lembaga pendidikan di Surakarta berupaya melakukan semua kegiatan akademik secara maksimal baik pada pembelajaran sekolah maupun pengembangan prestasi siswa. Sehingga Kota Surakarta juga sangat dikenal dengan prestasi pelajarannya yang amat membanggakan. Seperti yang tercatat dalam Open Data Kota Surakarta terdapat kenaikan jumlah prestasi Pendidikan dari 131 prestasi ditahun 2019 menjadi 457 prestasi ditahun 2021 (Open Data Surakarta, 2021).

Namun demikian selain fokus pada pembelajaran dan pengembangan prestasi siswa, Pemerintah Kota Surakarta harus fokus pada tata tertib sekolah seperti yang dijelaskan pada Peraturan Pemerintah Republik Indonesia Nomor 57 tahun 2021 pasal 4 ayat 1 bahwa “Standar kompetensi lulusan merupakan kriteria minimal tentang kesatuan sikap, keterampilan, dan pengetahuan yang menunjukkan capaian kemampuan Peserta Didik dari hasil pembelajarannya pada akhir Jenjang Pendidikan.” Pada peraturan pemerintah tersebut jelas bahwa sikap dalam sekolah juga perlu diperhatikan yakni dengan menaati adanya tata tertib sekolah. Selain itu dalam pengelolaan sistem data ketertiban siswa disekolah perlu diperhatikan juga, seperti pada Peraturan Pemerintah Republik Indonesia Nomor 17 tahun 2010 pasal 3 tentang Pengelolaan dan Penyelenggaraan Pendidikan bahwa “pengelolaan pendidikan ditujukan untuk menjamin :

- a. akses masyarakat atas pelayanan pendidikan yang mencukupi, merata, dan terjangkau
- b. mutu dan daya saing pendidikan serta relevansinya dengan kebutuhan dan/atau kondisi masyarakat dan
- c. efektivitas, efisiensi dan akuntabilitas pengelolaan pendidikan.

SMAN 1 Surakarta merupakan salah satu sekolah unggulan di Surakarta yang masih menggunakan cara manual dalam ketertiban masuk dan pulang sekolah serta pada proses pengolahan data ketertiban siswa, sehingga mengakibatkan beberapa kendala yang dihadapi oleh pihak sekolah dalam urusan pencatatan data ketertiban sekolah.

1.2 Tujuan

Tujuan dari tugas akhir ini yaitu membangun sistem informasi presensi dan ketertiban siswa untuk mengontrol ketertiban presensi masuk dan pulang siswa serta juga dapat mengelola data ketertiban siswa di SMAN 1 Surakarta.

1.3 Manfaat Produk

Manfaat adanya Sistem Informasi Presensi dan Ketertiban Siswa berbasis Website di SMAN 1 Surakarta antara lain:

a. Bagi Admin:

- 1) Memudahkan dalam mengontrol ketertiban siswa pada jam masuk dan pulang sekolah
- 2) Memudahkan proses pengelolaan data ketertiban siswa secara cepat dan efisien waktu
- 3) Menghindari terjadinya input data yang berulang karena berkas secara manual/lembaran yang hilang.

b. Bagi Guru:

- 1) Memudahkan untuk menginputkan data siswa yang melakukan pelanggaran tata tertib sekolah.
- 2) Guru dapat melihat daftar data pelanggaran siswa yang ada di SMAN 1 Surakarta.

- c. Bagi Siswa:
 - 1) Memudahkan untuk melakukan presensi masuk dan pulang sekolah.
 - 2) Memudahkan dalam melihat data ketertiban masing - masing di sekolah

1.4 Metode Pengembangan

Metode pengembangan dalam perancangan dan pembangunan Sistem Informasi Presensi dan Ketertiban Siswa di SMAN 1 Surakarta ini menerapkan metode agile. Menurut Sutiono, 2021 Metode Agile adalah sekumpulan metode yang berperan dalam pengembangan perangkat lunak yang mana prosesnya terus diulang dengan aturan dan solusi yang sudah disetujui dan disepakati dalam tim yang telah terstruktur dan terorganisir. Dalam pengembangan perangkat lunak dengan Metode Agile, terdapat beberapa tahapan yang harus dilalui antara lain:

- a. Perencanaan

Metode perencanaan merupakan Langkah dimana tim pengembang dan juga klien merancang apa saja yang dibutuhkan dalam suatu perangkat lunak yang hendak dibuat. Pada metode ini penulis melakukan wawancara di SMAN 1 Surakarta bersama dengan pihak pihak yang bersangkutan dan paham mengenai sistem yang diharapkan. Kemudian penulis merancang kebutuhan fungsional dari sistem yang akan dibangun, penulis juga merancang desain database dan tampilan yang akan digunakan dalam pembuatan sistem.

- b. Implementasi

Metode implementasi merupakan tahapan yang dimana para tim pemrograman melakukan pengkodean pada suatu perangkat lunak.

- c. Tes Perangkat Lunak

Pada metode tes perangkat lunak ini , perangkat lunak yang telah diprosuksi akan dites atau dicek, yang menjadi tanggung jawab bagian control kualitas supaya bug yang masih ditemukan dapat

langsung diperbaiki agar kualitas perangkat lunak tersebut tetap terjaga

d. Dokumentasi

Pada metode ini penulis melakukan proses dokumentasi perangkat lunak untuk memberi kemudahan terhadap proses pemeliharaan atau maintenance ke depannya.

e. Deployment Metode

Deployment merupakan tahap yang dilakukan untuk menjamin kualitas perangkat lunak yang diciptakan dengan menguji kualitas sistem. Jika sistem yang diproduksi telah memenuhi syarat, perangkat lunak tersebut nantinya sudah siap untuk dikembangkan

f. Pemeliharaan Metode pemeliharaan merupakan tahapan implementasi dan pemeliharaan system yang sudah lolos pengujian. Pemeliharaan system ini bertujuan untuk meningkatkan kualitas dari system serta perbaikan atau pengembangan dari system.

1.5 Definisi dan Istilah

Beberapa istilah dan definisi yang digunakan dalam penulisan naskah tugas akhir dapat dilihat pada tabel 1.1

Table 1.1 Definisi dan Istilah

No	Istilah/Singkatan	Keterangan
1.	Sistem Informasi	Sistem Informasi adalah sistem yang menggabungkan aktivitas manusia dengan penggunaan teknologi untuk mendukung aktivitas manajemen dan operasional. Hubungan tersebut dibuat berdasarkan interaksi manusia, data, informasi, teknologi, dan algoritma. (Ridwanti, 2023)

2.	Website	Website adalah halaman atau kumpulan halaman pada sebuah domain di internet yang dibuat dengan tujuan tertentu. Website umumnya berisi tampilan halaman berupa teks, gambar, animasi, audio, video atau gabungan satu dengan lainnya. (Putra, 2022)
3.	Perl Hypertext Preprocessor (PHP)	PHP singkatan dari Hypertext Preprocessor. Sedangkan pengertian dari PHP adalah bahasa pemrograman server-side yang digunakan dalam pembuatan website bersama dengan CSS dan HTML. PHP merubah website dari statis menjadi lebih dinamis dan mengubah konten serta fungsi website yang lebih interaktif untuk keperluan user. (Choiri, 2020)
4.	My Structure Query Language (MySQL)	MySQL adalah sebuah DBMS (<i>Database Management System</i>) menggunakan perintah SQL (<i>Structured Query Language</i>) yang banyak digunakan saat ini dalam pembuatan aplikasi berbasis website. MySQL termasuk ke dalam RDBMS (<i>Relational Database Management System</i>). Sehingga, menggunakan tabel, kolom, baris, di dalam struktur database -nya. Jadi, dalam proses pengambilan data menggunakan metode relational database serta menjadi penghubung antara perangkat lunak dan database server. (Adani, 2020)
5.	XAMPP	XAMPP adalah salah satu aplikasi yang berfungsi sebagai <i>web server</i> di <i>localhost</i> . Dengan menggunakan XAMPP, kamu dapat

		mengelola <i>database</i> yang ada di localhost tanpa memerlukan akses internet. (Maksum, 2022)
6.	Hyper Text Markup Language (HTML)	HTML atau <i>Hypertext Markup Language</i> adalah bahasa markup yang digunakan untuk membuat struktur halaman website agar dapat ditampilkan pada <i>web browser</i> . Jadi, HTML dapat dianalogikan sebagai pondasi awal dalam menyusun kerangka halaman web secara terstruktur sebelum membahas terkait tampilan desain dan sisi fungsionalitas. (Adani, 2021)
7.	Entity Relationship Diagram (ERD)	ERD adalah pemodelan data atau sistem dalam database, Fungsi ERD adalah untuk memodelkan struktur dan hubungan antar data yang relatif kompleks. (Sulthon, 2022)
8.	Use Case Diagram	Use Case Diagram sendiri adalah proses penggambaran yang dilakukan untuk menunjukkan hubungan antara pengguna dengan sistem yang dirancang. Hasil representasi dari skema tersebut dibuat secara sederhana dan bertujuan untuk memudahkan <i>user</i> dalam membaca informasi yang diberikan. (Adani, 2021)
9.	Activity Diagram	Activity Diagram adalah pemodelan yang dilakukan pada suatu sistem dan menggambarkan aktivitas sistem berjalan. Activity diagram di gunakan sebagai penjelasan aktivitas program tanpa melihat koding atau tampilan. Activity diagram di

		gambarkan dengan simbol-simbol yang setiap simbolnya memiliki makna dan tujuan. (Novitasari, 2023)
10.	Laravel	Laravel adalah framework berbasis bahasa pemrograman PHP yang bisa digunakan untuk membantu proses pengembangan sebuah website agar lebih maksimal. Dengan menggunakan Laravel, website yang dihasilkan akan lebih dinamis. (Maksum, 2022)
11.	Mockup	<i>Mockup</i> adalah rancangan desain yang akan diimplementasikan pada produk dan terdiri dari beberapa elemen warna, tipografi, <i>layout</i> , spasi dan navigasi (Amalia, 2021)

1.6 Referensi

Referensi yang digunakan sebagai referensi pada pembuatan sistem informasi presensi dan ketertiban siswa ini yang pertama yakni jurnal dari Kurniawan Andy 2016 dengan judul Rancan Bangun Sistem Informasi Penilaian dan Presensi Siswa berbasis Web pada SMAN 12 Surabaya. Tujuan penelitian tersebut untuk mempermudah rekapitulasi presensi siswa yang dapat diakses oleh wali murid melalui web.

Kemudian referensi produk yang kedua yakni referensi jurnal dari Syarah, Sri Rahayu, dan Arie Fiyani 2011 dengan judul Perancangan Sistem Informasi Absensi Staff dan Guru pada SMA Bina Insani berbasis Web. Tujuan dari sistem ini adalah memberikan kemudahan baik kepada staff saat melakukan absensi sampai pembuatan laporan kepada kepala sekolah.

Referensi yang ketiga yakni referensi jurnal dari Paramitha, Risnasari, dan Saputro 2018 dengan judul Pengembangan Sistem Informasi absensi Berbasis Java Desktop di SMA Darul Kholil Bangkalan. Tujuan dari sistem ini untuk mempermudah mencatat dan menyimpan data, lebih teliti dalam hal pengolahan data dan mengurangi redundansi data. Sistem ini dikembangkan menggunakan program java

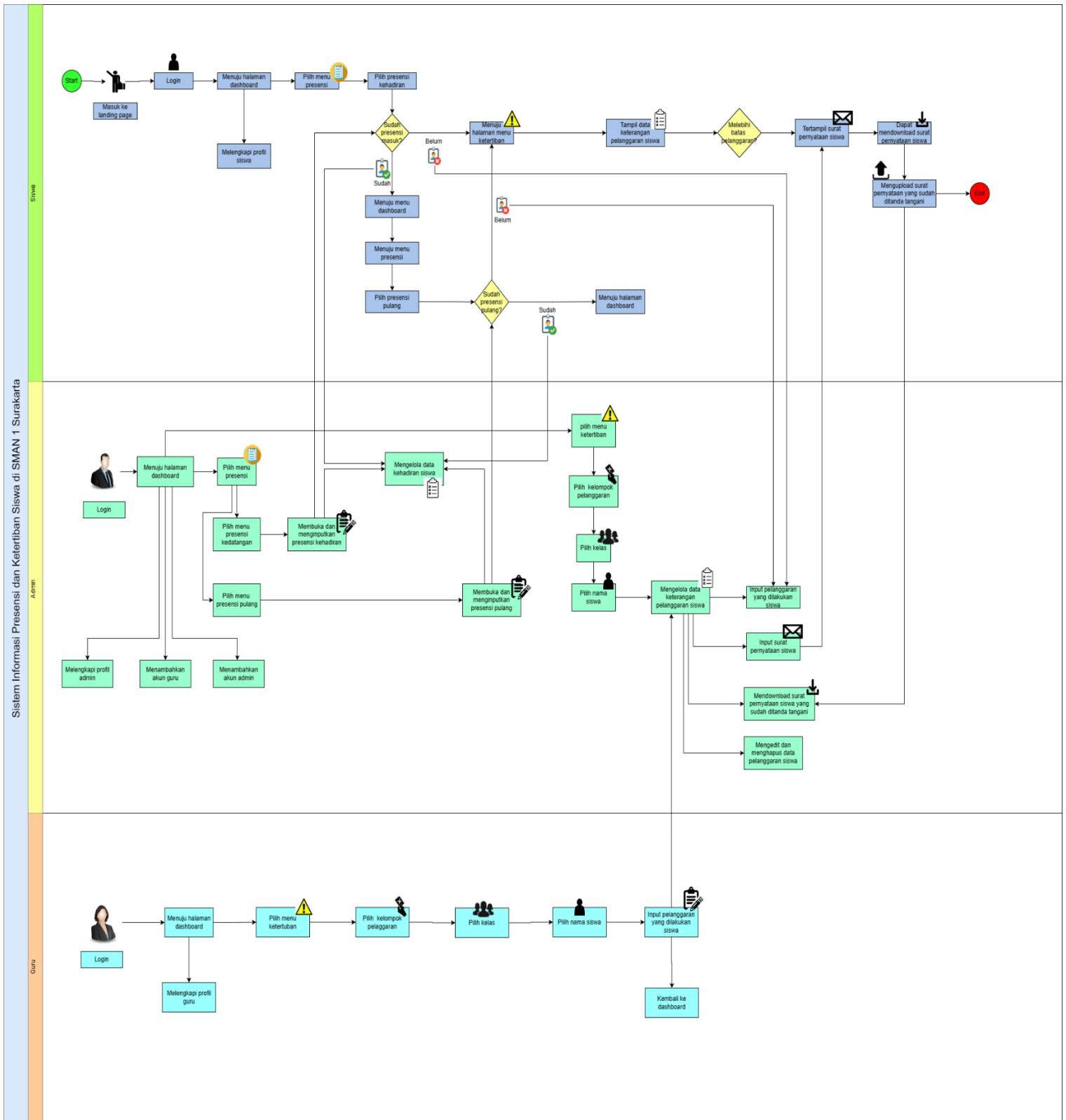
Dengan beberapa penelitian yang dijelaskan diatas, sistem yang akan dibuat oleh penulis yaitu “Sistem Infomasi Presensi dan Ketertiban Siswa di SMAN 1 Surakarta” dengan fitur tak hanya dapat melakukan presensi dan mengontrol presensi bagi admin, tetapi juga mengelola data ketertiban siswa (Kurniawan, 2016).

BAB II

DESKRIPSI PRODUK

2.1 Deskripsi Produk

Produk yang akan dihasilkan dari tugas akhir ini yaitu Sistem Informasi Presensi dan Ketertiban Siswa di SMAN 1 Surakarta. Pada sistem ini terdapat beberapa fitur yang akan ditawarkan berdasarkan permintaan dari SMAN 1 Surakarta seperti presensi pada saat masuk dan pulang sekolah serta juga pengelolaan data ketertiban siswa. Pada sistem ini terdapat 3 aktor, yaitu Bagian admin yakni tim tata tertib sekolah, Guru, dan Siswa. Admin mempunyai kemampuan membuka dan menutup presensi siswa serta memanajemen data ketertiban siswa. Guru dapat mengisi data siswa yang melakukan pelanggaran diSekolah. Siswa dapat melakukan presensi dan melihat data pelanggaran siswa itu sendiri. Proses bisnis dari Sistem Informasi Presensi dan Ketertiban Siswa di SMAN 1 Surakarta dapat dilihat pada gambar 2.1



Gambar 2.1 Proses Bisnis

2.2 Fungsional Produk

Kebutuhan fungsionalitas dari “Rancangan dan Implementasi Sistem Informasi Presensi dan Ketertiban Siswa di SMAN 1 Surakarta” dapat dilihat pada table 2.1.

Tabel 2.1 Fungsional Produk

KODE	KEBUTUHAN FUNCTIONAL	AKTOR
FR001	Sistem dapat menampilkan halaman form login	Admin, guru, siswa
FR002	Sistem dapat menampilkan halaman dashboard	Admin, guru, siswa
FR003	Sistem dapat menampilkan halaman menu ketertiban pada admin	Admin
FR004	Sistem dapat menambahkan pelanggaran siswa	Admin, guru
FR005	Sistem dapat mengedit pelanggaran siswa	Admin
FR006	Sistem dapat menghapus pelanggaran siswa	Admin
FR007	Sistem dapat mengunggah surat pernyataan pelanggaran untuk siswa	Admin
FR008	Sistem dapat mendownload surat pernyataan pelanggaran siswa yang sudah ditanda tangani orang tua	Admin
FR009	Sistem dapat melakukan presensi datang	Siswa
FR010	Sistem dapat melakukan presensi pulang	Siswa
FR011	Sistem dapat menampilkan daftar data ketertiban siswa itu sendiri	Siswa
FR012	Sistem dapat mendownload surat pernyataan pelanggaran	Siswa

FR013	Sistem dapat mengupload surat pernyataan pelanggaran yang sudah ditanda tangani orang tua	Siswa
FR014	Sistem dapat menambahkan data siswa	Admin
FR015	Sistem dapat mengubah data siswa	Admin
FR016	Sistem dapat menghapus data siswa	Admin
FR017	Sistem dapat menambahkan data guru	Admin
FR018	Sistem dapat mengubah data guru	Admin
FR019	Sistem dapat menghapus data guru	Admin
FR020	Sistem dapat menampilkan data presensi	Admin
FR021	Sistem dapat mencari nama pada menu ketertiban siswa	Admin
FR022	Sistem dapat mencari nama pada menu data siswa	Admin

2.3 Penggolongan dan Karakteristik Pengguna

Penggolongan dan Karakteristik Pengguna dari “Rancangan dan Implementasi sistem informasi presensi dan ketertiban siswa di SMAN 1 Surakarta” dapat dilihat pada table 2.2.

Tabel 2.2 Penggolongan Karakteristik Pengguna

No	Penggolongan	Tugas	Hak Akses
1.	Admin	<ul style="list-style-type: none"> - Login - Mengatur data presensi dating 	<ul style="list-style-type: none"> - Dapat melihat data presensi kedatangan siswa

		<ul style="list-style-type: none"> - Mengatur data presensi pulang - Menambahkan data pelanggaran siswa - Mengedit data pelanggaran siswa - Menghapus data pelanggaran siswa - Mengupload surat pernyataan pelanggaran siswa - Menambahkan data guru - Menambahkan data siswa - logout 	<ul style="list-style-type: none"> - Dapat melihat data presensi kepulangan siswa - Dapat melihat data pelanggaran siswa - Dapat mendownload surat pernyataan siswa yang sudah ditanda tangani orang tua - Dapat mengelola data presensi siswa - Dapat mengelola data pelanggaran siswa - Dapat mengelola data guru - Dapat mengelola data siswa
2.	Siswa	<ul style="list-style-type: none"> - Login - Melakukan presensi kedatangan - Melakukan presensi kepulangan - Mendownload surat pernyataan pelanggaran siswa - Mengupload surat pernyataan pelanggaran yang 	<ul style="list-style-type: none"> - Dapat melihat data ketertiban milik siswa itu sendiri - Dapat melihat surat pernyataan pelanggaran siswa - Dapat melihat data presensi milik siswa itu sendiri

		<ul style="list-style-type: none"> - sudah ditanda tangani orang tua - Logout 	
3.	Guru	<ul style="list-style-type: none"> - Login - Menambahkan pelanggaran yang dilakukan siswa - Logout 	<ul style="list-style-type: none"> - Dapat melihat data pelanggaran yang ditambahkan oleh guru itu sendiri.

2.4 Lingkungan Operasi

Perangkat lunak pada sisi server yang digunakan pada “Rancangan dan Implementasi Sistem Informasi Presensi dan Ketertiban Siswa di SMAN 1 Surakarta” dapat dilihat pada table 2.3.

Tabel 2.3 Lingkungan Operasi

Text Editor	Visual Studio Code 1.74.2
Scripting Language	PHP, CSS, HTML, JavaScript
Framework	Laravel 9
Database Management System	MySQL
Web Browser	Microsoft Edge, Google Chrome

Kemudian untuk spesifikasi perangkat keras yang digunakan dapat dilihat pada tabel 2.4.

Tabel 2.4 Spesifikasi Perangkat Lunak

Perangkat	Laptop
Processor Type	Intel® Core™ i5-8250U

Processor	Intel® Core™ i5-8250U CPU @ 1.60GHz 1.80 GHz
Memory	8,00 GB RAM
Graphics	Intel® UHD Graphics
Operating System	Microsoft Windows 10 (64-bit)

2.5 Batasan Desain dan Implementasi

Batasan desain dan implementasi pada “Rancangan dan Implementasi Sistem Informasi Presensi dan Ketertiban Siswa di SMAN 1 Surakarta” sebagai berikut :

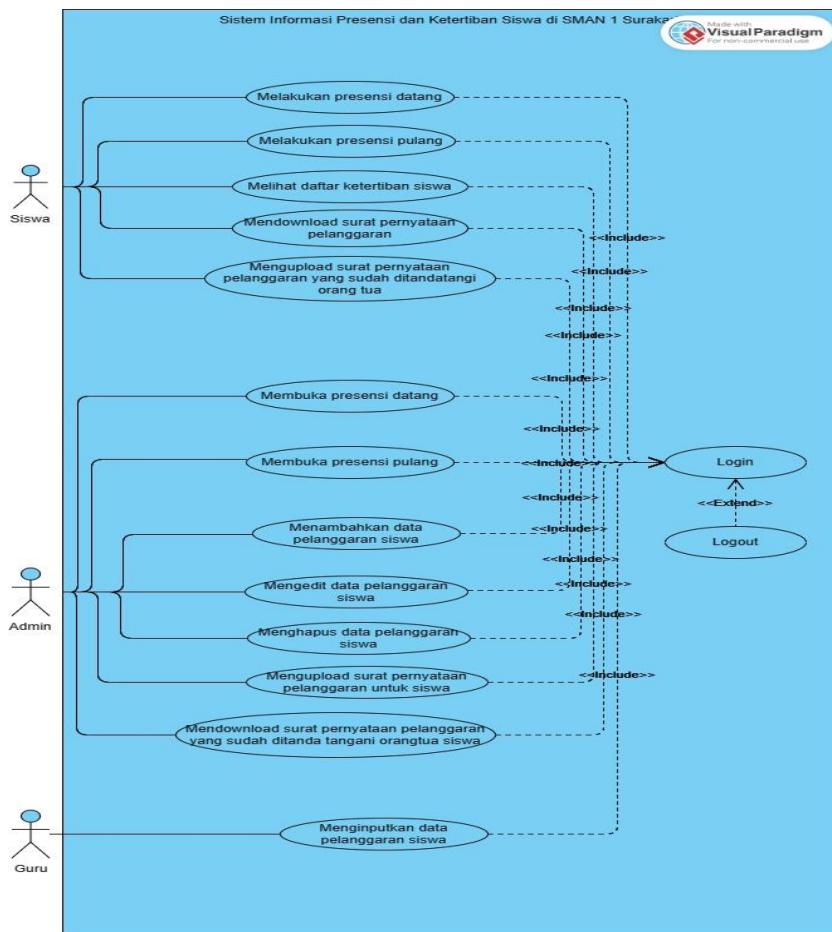
- Perancangan yang dibuat terdiri dari data presensi datang dan pulang siswa, serta data ketertiban siswa yang berisikan keterangan pelanggaran yang dilakukan siswa.
- Pada Tugas Akhir ini penulis membuat website dari rancangan yang telah dibuat yakni sistem informasi presensi dan ketertiban siswa.
- Pada implementasi nanti menggunakan database MySQL, scripting language PHP,CSS,HTML, dan menggunakan framework Laravel.
- Untuk permasalahan mengenai jaringan tidak termasuk dalam cakupan penggerjaan Tugas Akhir ini.

BAB III

Perancangan Produk

1.1 Use Case

Use case diagram “Rancangan dan Implementasi Sistem Informasi presensi dan Ketertiban Siswa di SMAN 1 Surakarta” dapat dilihat pada gambar 3.1.



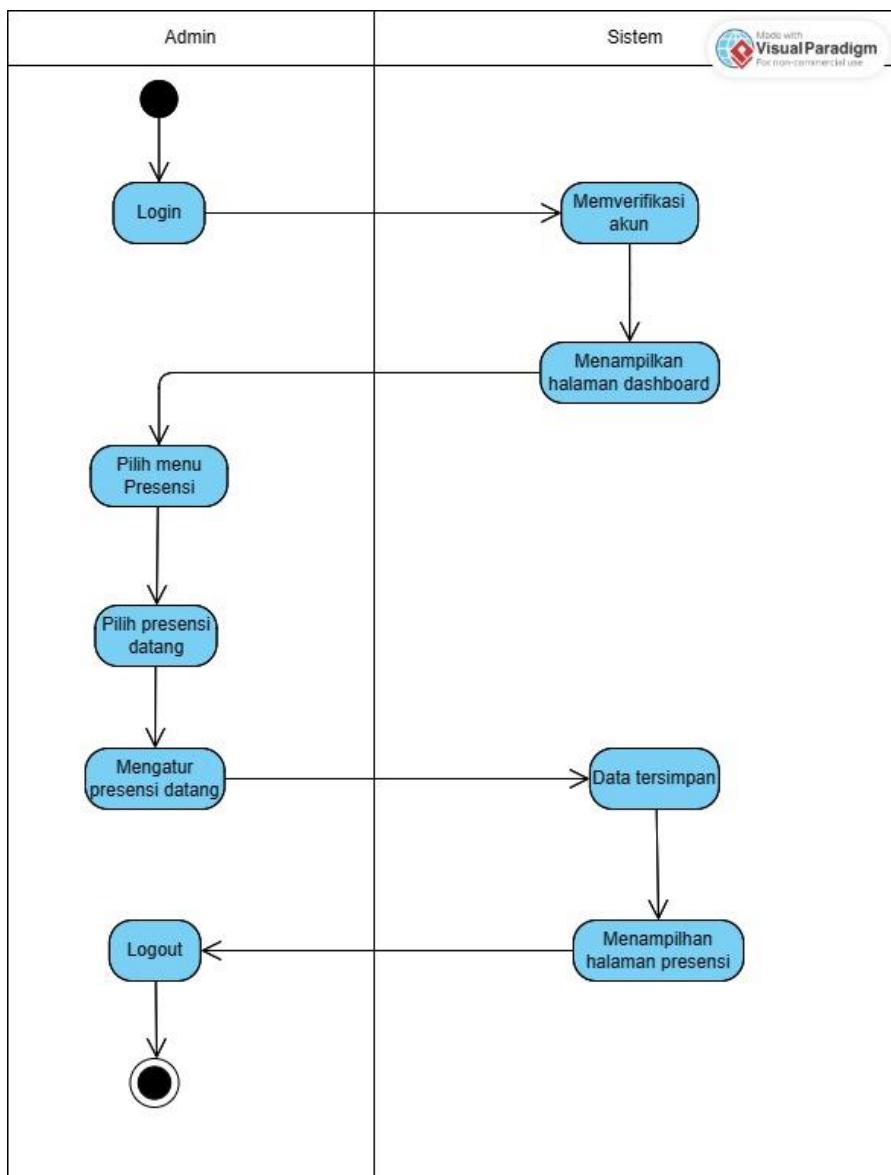
Gambar 3.1 Use Case Diagram

Di luar use case terdapat alur dimana setelah siswa mencetak surat pernyataan maka siswa harus memintakan tanda tangan kepada orangtua kemudian baru bisa mengupload lagi surat pernyataan yang sudah ditanda tangani orang tua.

1.2 Activity Diagram

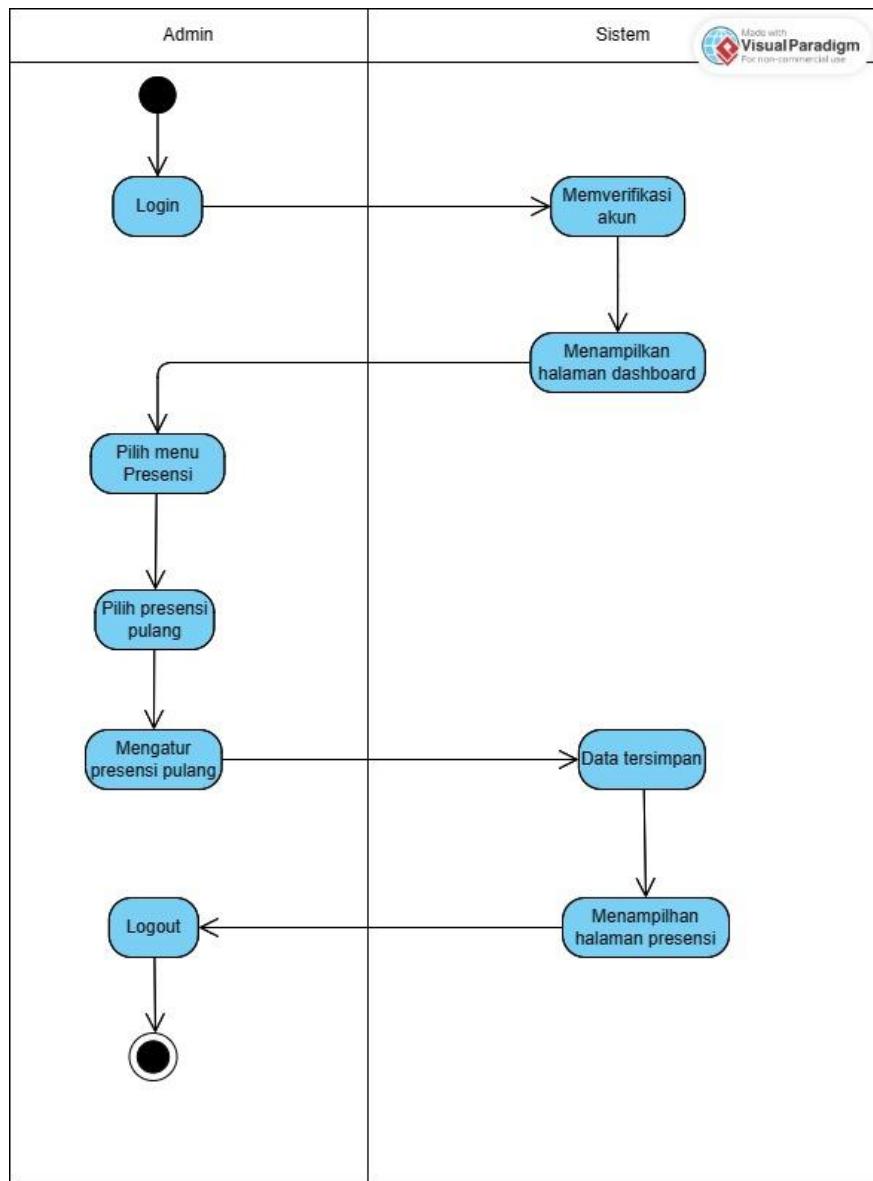
Activity diagram pada “Rancangan dan Implementasi Sistem Informasi presensi dan Ketertiban Siswa di SMAN 1 Surakarta” sebagai berikut :

1. Activity diagram admin mengatur presensi dapat dilihat pada gambar 3.2



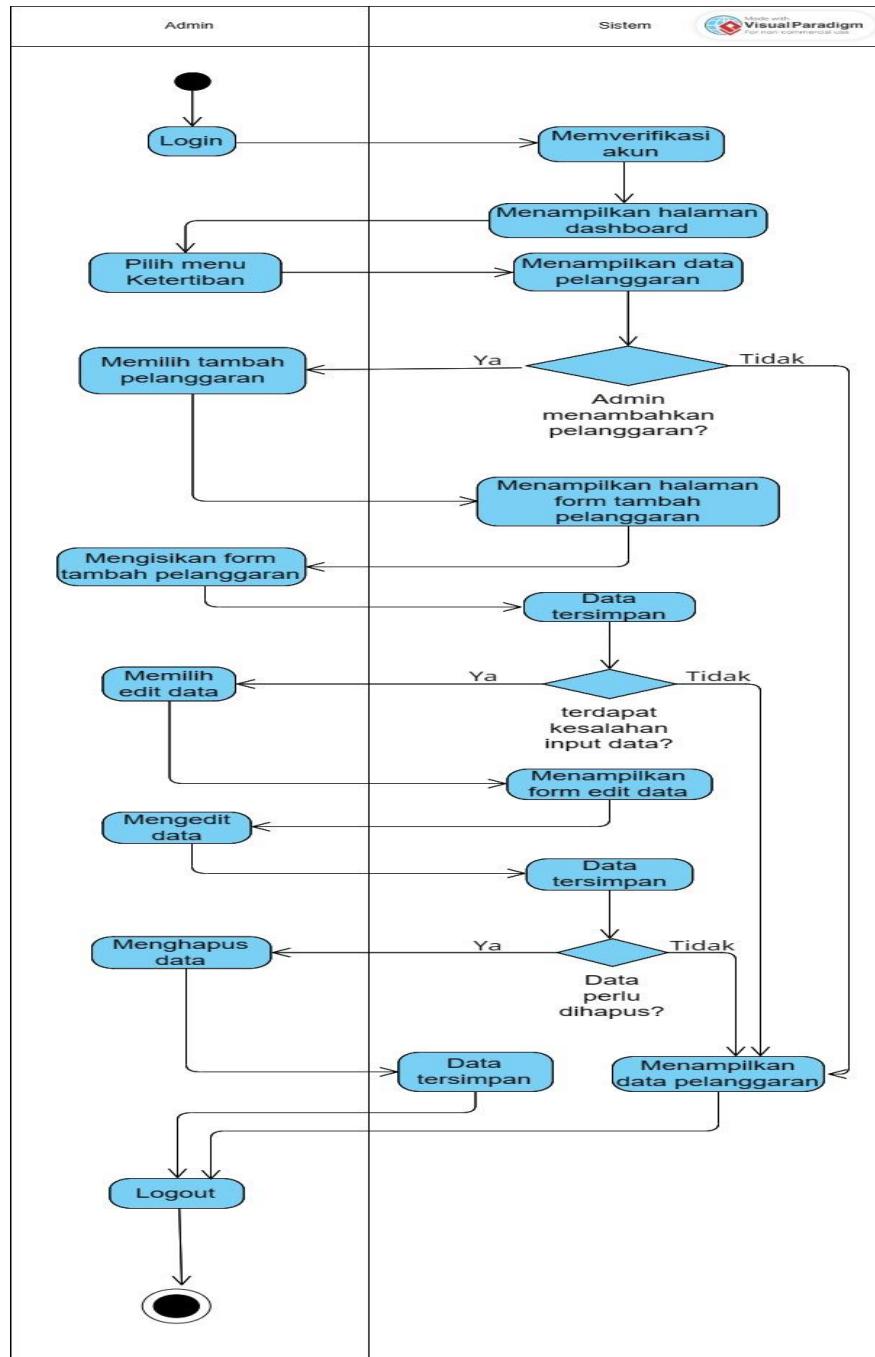
Gambar3.2 Activity Diagram Admin mengatur presensi datang

2. Activity diagram admin mengatur presensi pulang dapat dilihat pada gambar 3.3



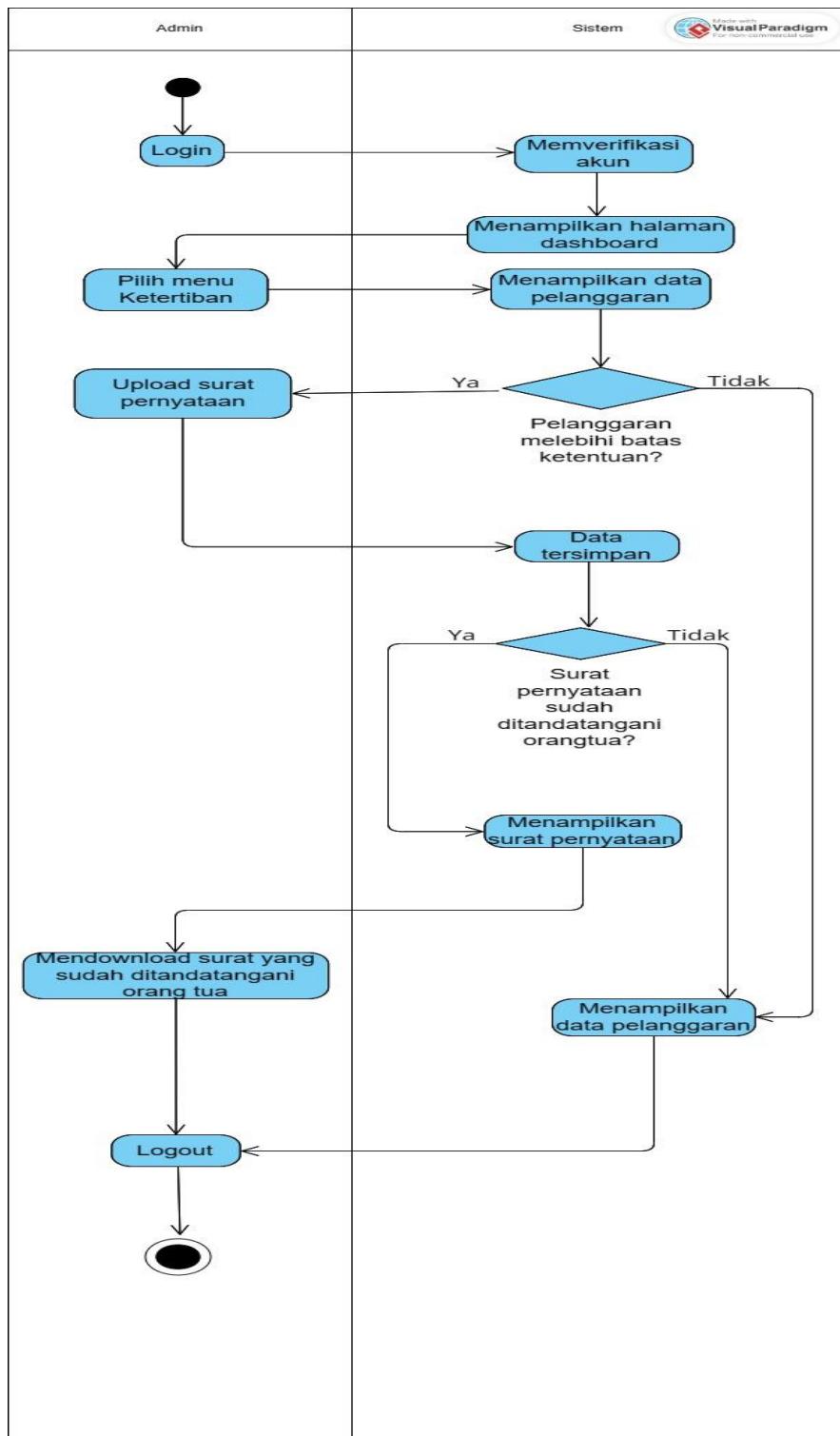
Gambar 3.3 Activity Diagram Admin mengatur presensi pulang

3. Activity diagram admin mengelola menu ketertiban dapat dilihat pada gambar 3.4



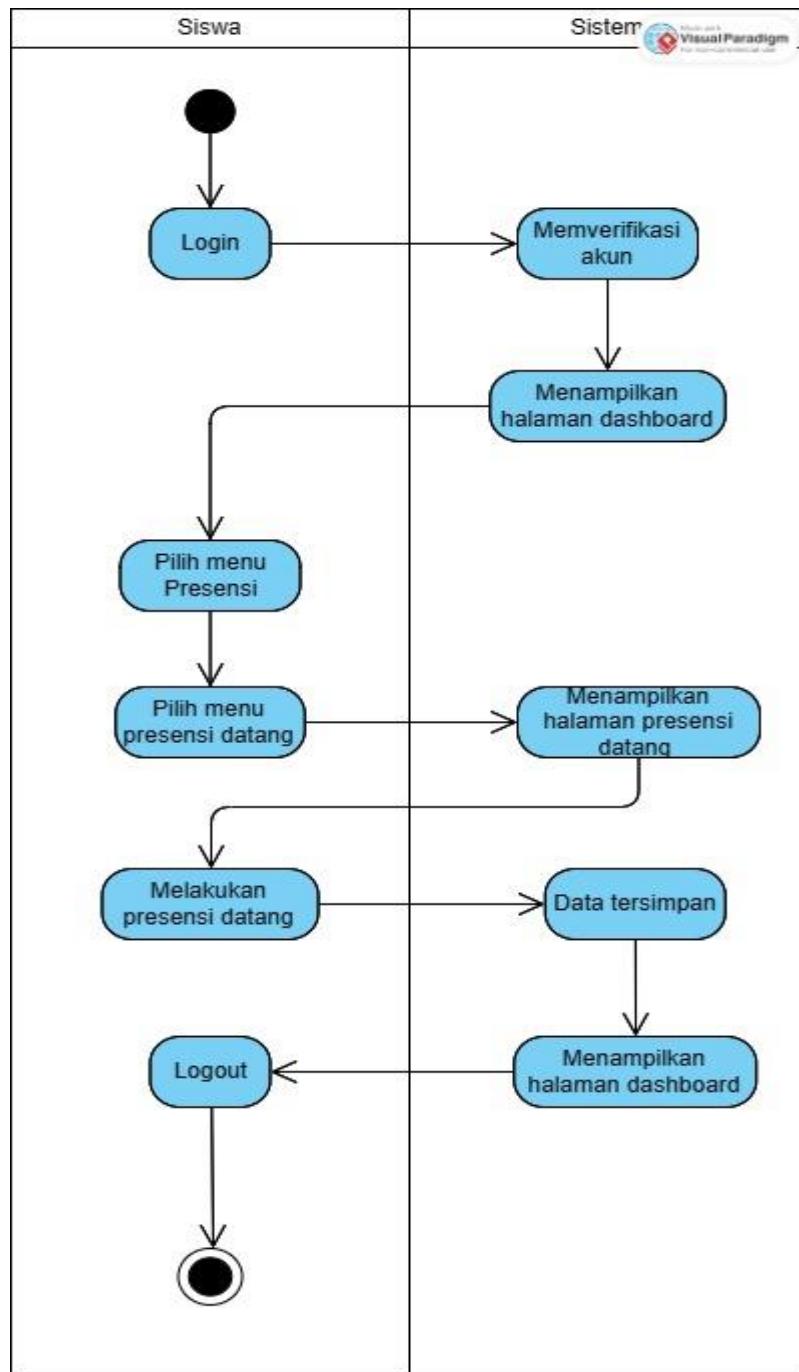
Gambar 3.4 Activity Diagram Admin mengelola menu ketertiban

4. Activity diagram admin untuk mengunggah surat pernyataan dapat dilihat pada gambar 3.5



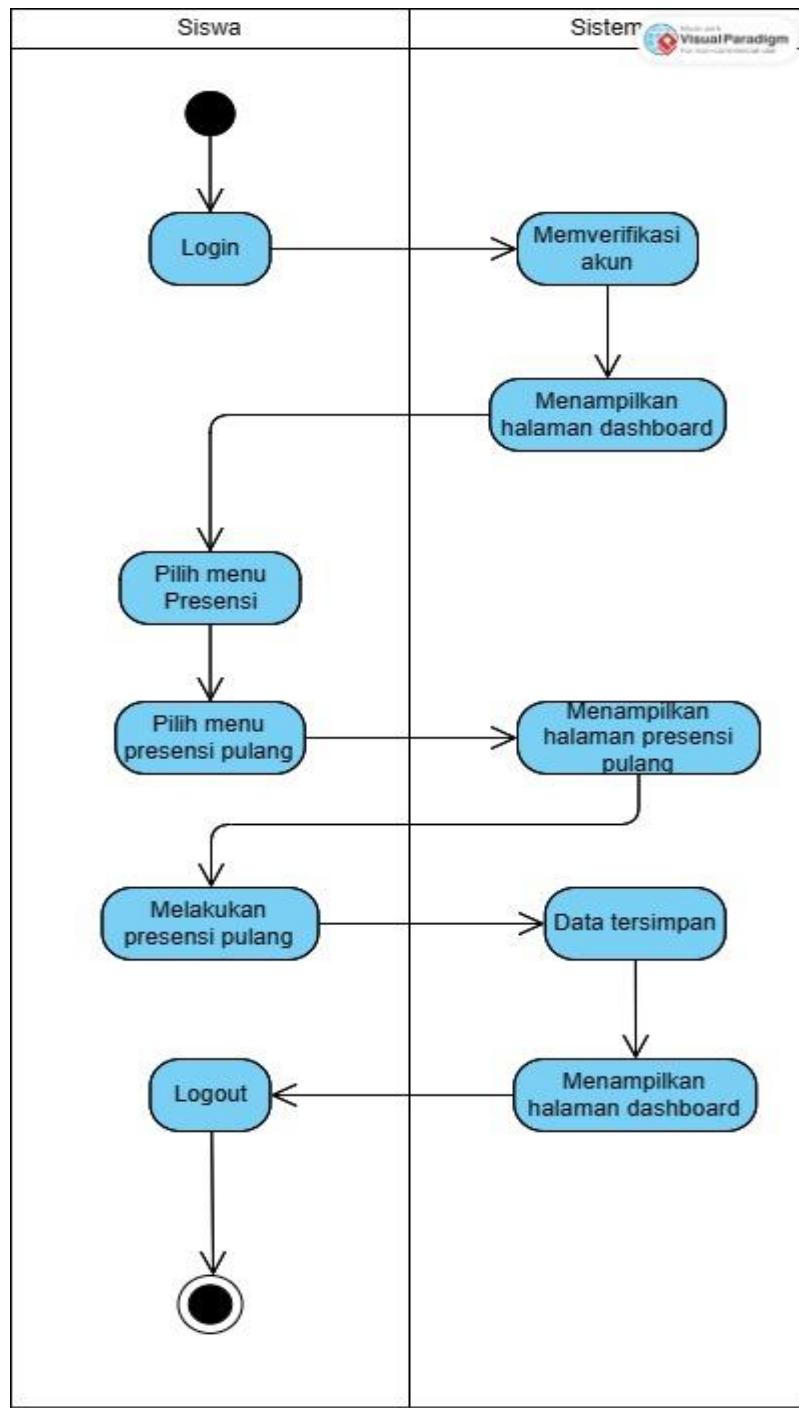
Gambar 3.5 Activity Diagram admin mengunggah surat pernyataan

5. Activity diagram siswa presensi kedatangan dapat dilihat pada gambar 3.5



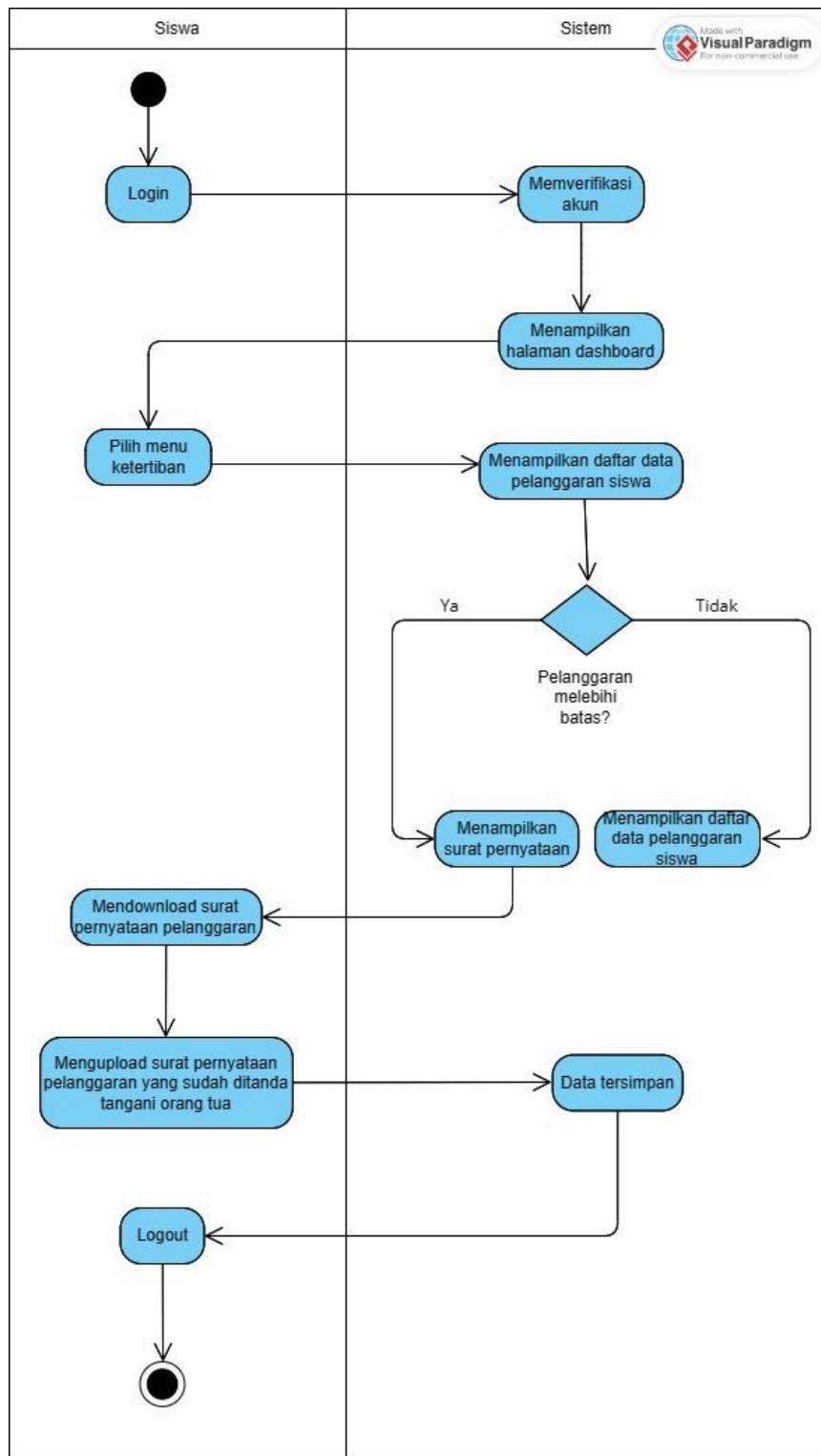
Gambar 3.5 Activity Diagram siswa melakukan presensi datang

6. Activity diagram siswa presensi pulang dapat dilihat pada gambar 3.6



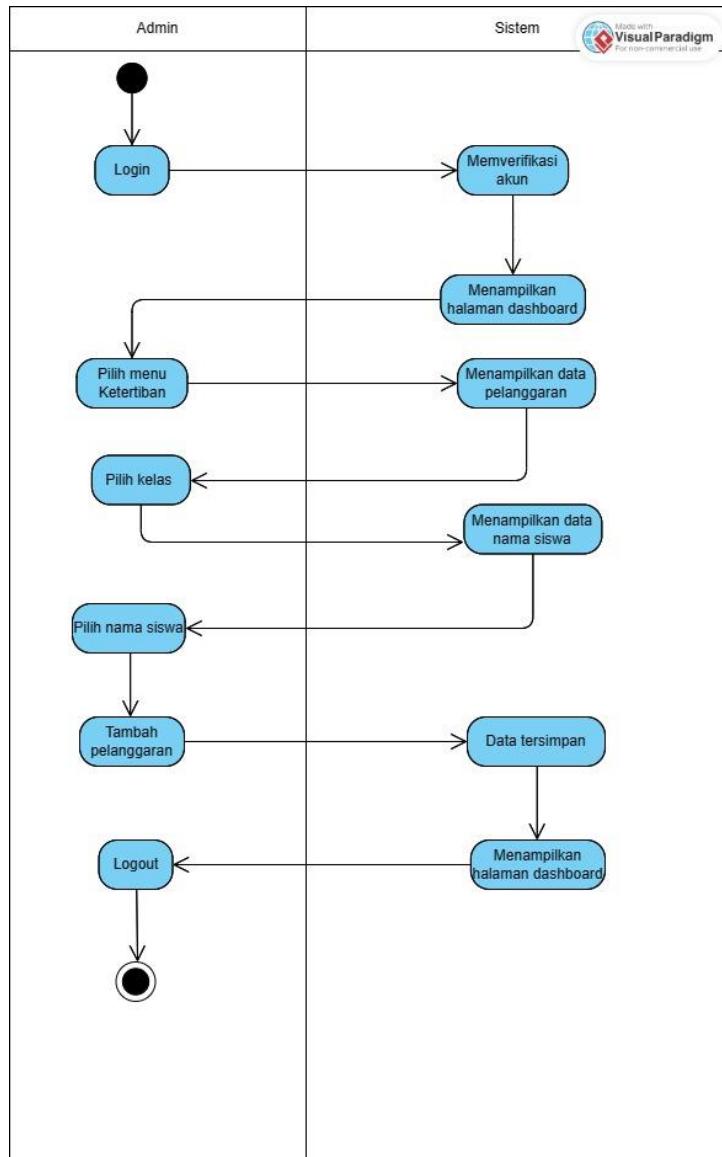
Gambar 3.6 Activity Diagram siswa melakukan presensi pulang

7. Activity diagram siswa menu ketertiban dapat dilihat pada gambar 3.7



Gambar 3.7 Activity Diagram siswa pada menu ketertiban

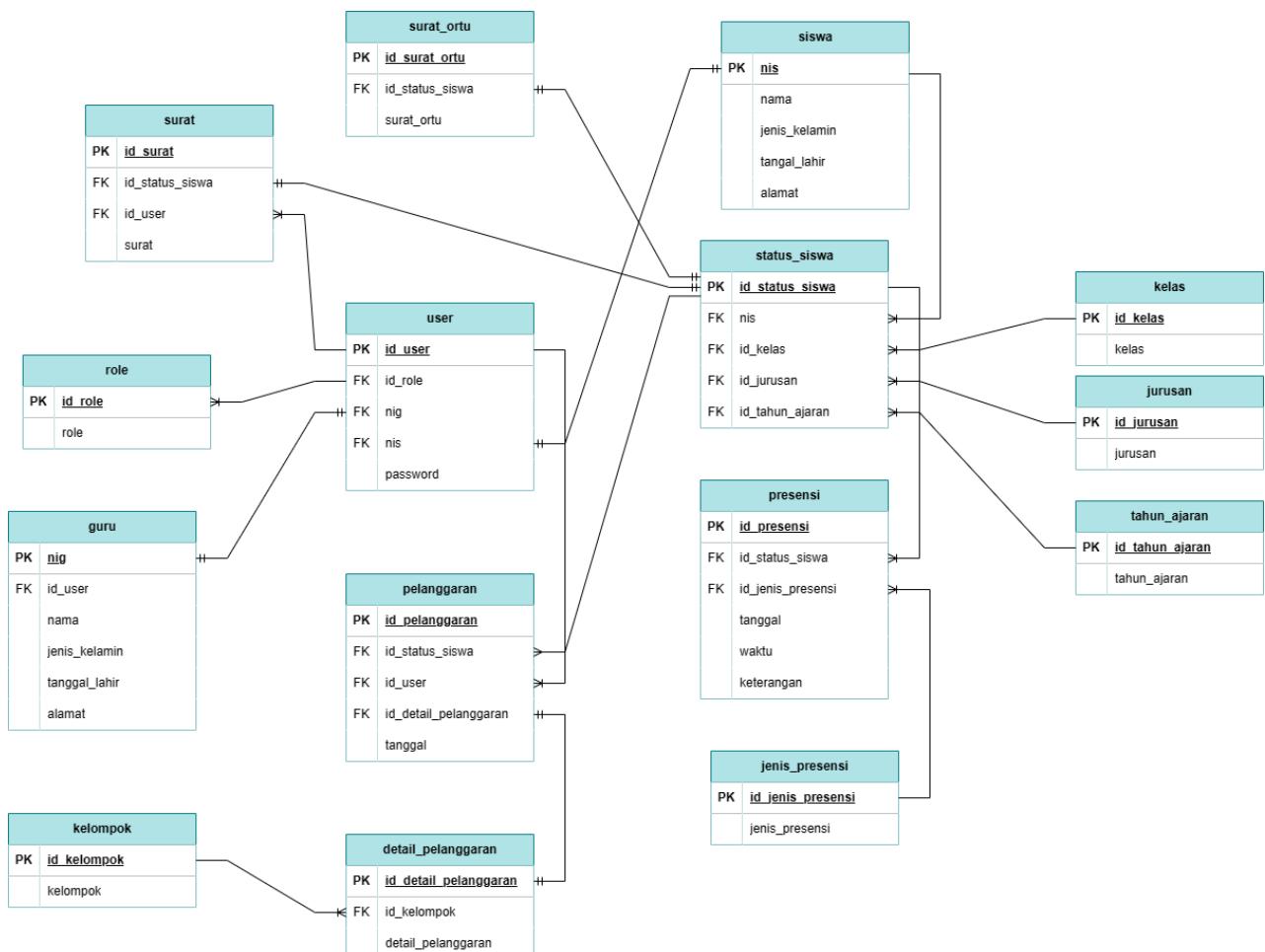
8. Activity diagram guru menambahkan data pelanggaran siswa dapat dilihat pada gambar 3.8



Gambar 3.8 Activity Diagram Guru menambahkan data pelanggaran

1.3 Desain database (ERD)

ERD “Rancangan dan Implementasi Sistem Informasi presensi dan Ketertiban Siswa di SMAN 1 Surakarta” seperti pada gambar 3.9

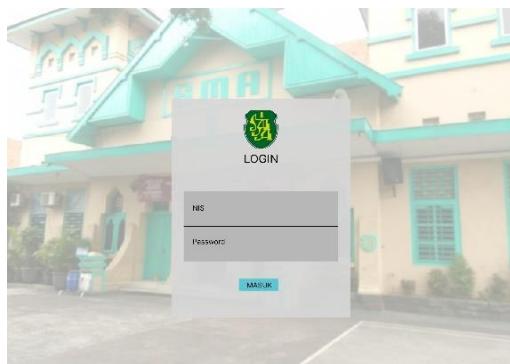


Gambar 3.9 ERD

3.4 Mock up

Mockup pada “Rancangan dan Implementasi Sistem Informasi dan ketertiban Siswa di SMAN 1 Surakarta” sebagai berikut:

1. Admin



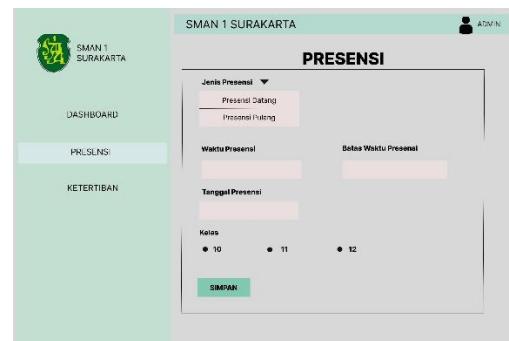
Gambar 3.10 Login Admin



Gambar 3.11 Dashboard Admin



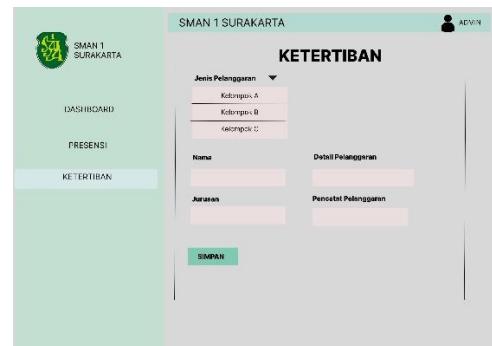
Gambar 3.12 Admin Data Presensi



Gambar 3.13 Admin Atur Presensi



Gambar 3.13 Admin menu Ketertiban



Gambar 3.14 Form pelanggaran

2. Siswa



Gambar 3.15 Login Siswa

Gambar 3.16 Dashboard siswa



Gambar 3.17 Presensi siswa

Gambar 3.18 Siswa sudah presensi



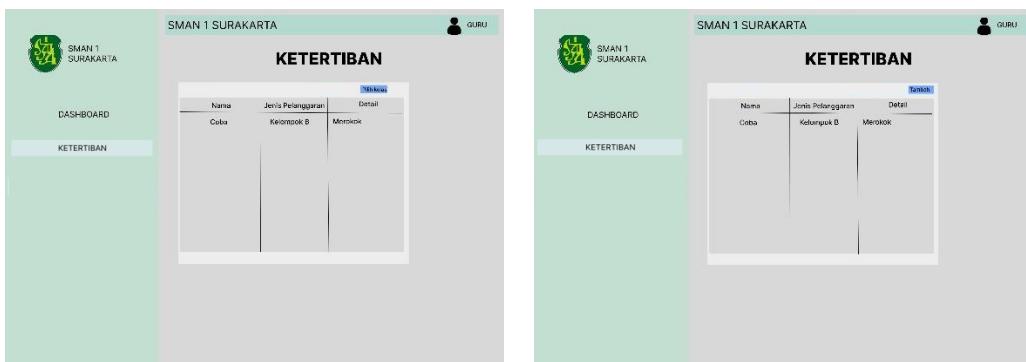
Gambar 3.19 Ketertiban Siswa

3. Guru



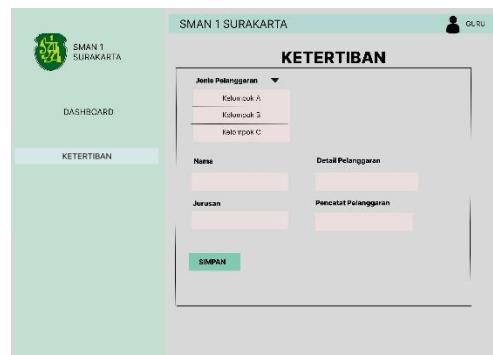
Gambar 3.20 Login Guru

Gambar 3.21 Dashboard Siswa



Gambar 3.22 Menu Ketertiban

Gambar 3.23 Menu Ketertiban kelas



Gambar 3.24 Form Ketertiban

Dokumentasi Pengguna

The screenshot shows the main dashboard of the SMAN 1 Surakarta application. On the left, there is a green sidebar with the school logo and navigation links: Dashboard, Presensi, and Ketertiban. The main content area has a header "SELAMAT DATANG". Below it, there are two sections: "PROFIL SMAN 1 SURAKARTA" featuring a photo of the school building and its address; and "PRESTASI SMAN 1 SURAKARTA" featuring a photo of students receiving awards. A search bar and a "Search" button are at the top of the main content area. The top of the screen shows browser tabs for "SMAN 1 Surakarta" and "localhost / 127.0.0.1 / smansa /". The bottom of the screen shows a Windows taskbar with various icons and system status.

The screenshot shows the "PRESENSI" (Attendance) section of the dashboard. It features a table with columns for "Nama" (Name), "Tanggal Presensi" (Attendance Date), "Jam Presensi" (Attendance Time), and "Action". There are buttons for "Tambah Presensi" (Add Attendance) and "Export". The sidebar on the left remains the same as the previous screenshot. The top and bottom of the screen are consistent with the first screenshot.

The screenshot shows a web application interface for managing student discipline cases. The left sidebar has a green background with the school logo and navigation items: Dashboard, Presensi, and Ketertiban. The main content area has a white background with a header bar containing a search input, a 'Search' button, and user information (DAFFA M ADMIN). Below the header is a section titled 'KETERTIBAN' with a table header row containing columns for Name, Group of Offense, Infraction Details, Statement Letter, Signature Letter, and Action. There are also 'Tambah Pelanggaran' and 'Export' buttons. At the bottom of the page is a footer with the text '2023 © Hyper - Coderthemes.com'.

The screenshot shows a web application interface for adding a new student infraction case. The left sidebar has a green background with the school logo and navigation items: Dashboard, Presensi, and Ketertiban. The main content area has a white background with a header bar containing a search input, a 'Search' button, and user information (DAFFA M ADMIN). Below the header is a section titled 'DATA KETERTIBAN' with a sub-section titled 'Pelanggaran'. It includes fields for selecting the offense group ('Pilih Kelompok Pelanggaran') and choosing the student ('Pilih Nama Siswa'), both with dropdown menus. There is also a date input field ('Masukkan Tanggal') with the value '05/24/2023' and a 'Simpan' (Save) button. At the bottom of the page is a footer with the text '2023 © Hyper - Coderthemes.com'.

BAB IV

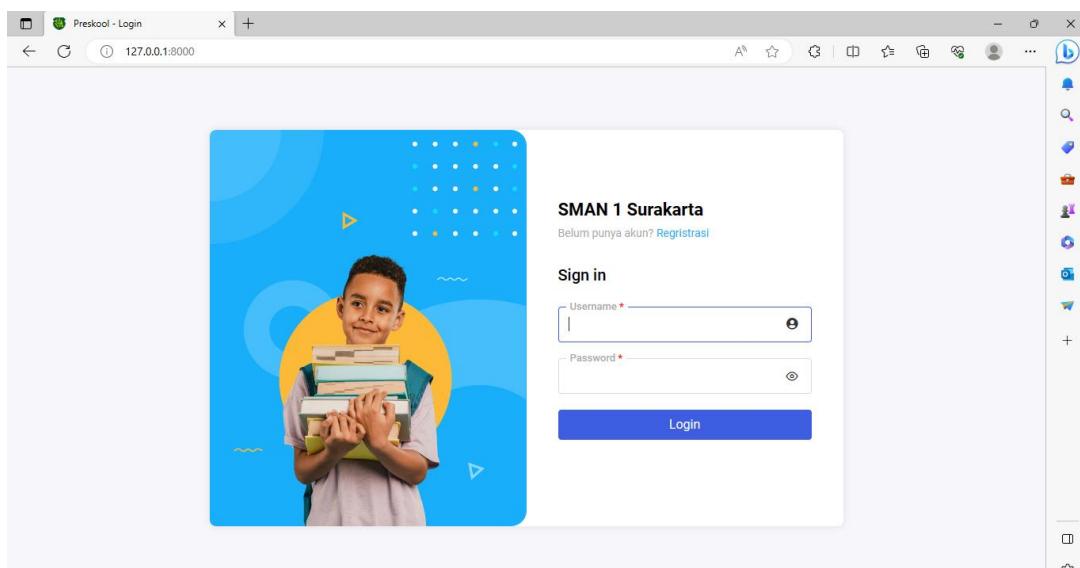
Implementasi

4.1 Implementasi Halaman Admin

Pada halaman Admin ini nanti akan dioperasikan oleh tim tata tertib SMAN 1 Surakarta untuk mengelola data ketertiban, mengelola data guru, serta mengelola data siswa,

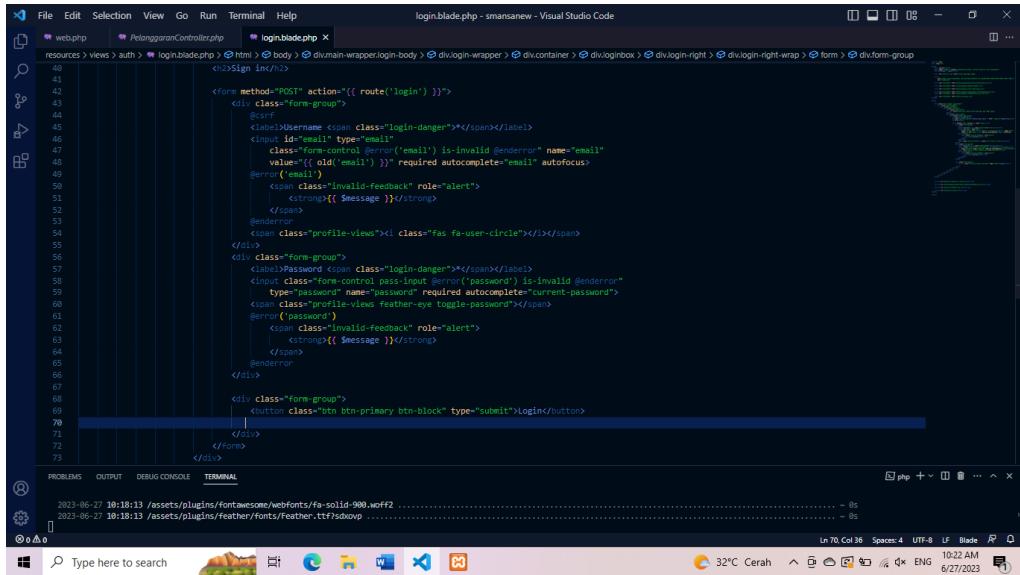
4.1.1 Halaman Login

Halaman login ini dapat diakses oleh semua role pada sistem yang telah memiliki akun.



Gambar 4.1 Gambar Tampilan Halaman Login

Source Code Halaman Login

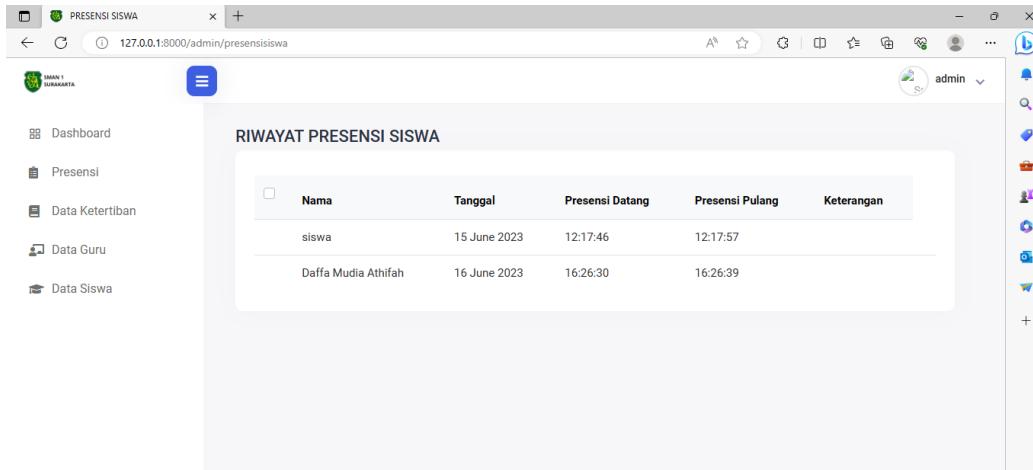


```
<form method="POST" action="{{ route('login') }}>
    @csrf
    <div class="form-group">
        <label>Username </label>
        <input id="email" type="email" class="form-control" name="email" value="old('email')" required autocomplete="email" autofocus>
        @error('email')
            <span class="invalid-feedback" role="alert">
                {{ $message }}</span>
        @enderror
        <span class="profile-view"><i class="fas fa-user-circle"></i></span>
    </div>
    <div class="form-group">
        <label>Password </label>
        <input class="form-control" type="password" name="password" required autocomplete="current-password" value="old('password')"/>
        <span class="profile-view feather-eye toggle-password"></span>
        @error('password')
            <span class="invalid-feedback" role="alert">
                {{ $message }}</span>
        @enderror
    </div>
    <div class="form-group">
        <button class="btn btn-primary btn-block" type="submit">Login</button>
    </div>
</form>
```

Gambar 4.2 Gambar Source Code Halaman Login

4.1.4 Halaman Data Presensi

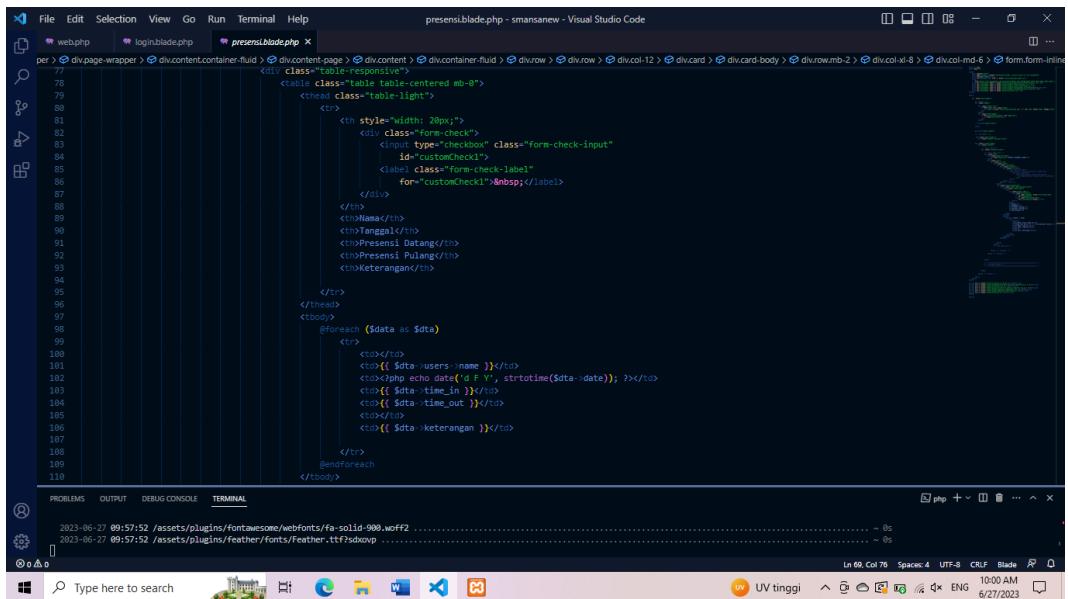
Halaman presensi ini akan menampilkan data hasil dari presensi siswa pada tiap harinya.



	Nama	Tanggal	Presensi Datang	Presensi Pulang	Keterangan
	siswa	15 June 2023	12:17:46	12:17:57	
	Daffa Mudia Athifah	16 June 2023	16:26:30	16:26:39	

Gambar 4.3 Gambar Tampilan Halaman Presensi

Source Code Halaman Presensi



```
per @ div.page-wrapper > @ div.content.container-fluid > @ div.content-page > @ div.container-fluid > @ div.row > @ div.col-12 > @ div.card > @ div.card-body > @ div.row.mb-2 > @ div.col-md-6 > @ form.form-inline
    77     <div class="table-responsive">
    78         <table class="table table-centered mb-0">
    79             <thead class="table-light">
    80                 <tr>
    81                     <th style="width: 20px;">
    82                         <div class="form-check">
    83                             <input type="checkbox" class="form-check-input" id="customCheck1">
    84                             <label class="form-check-label" for="customCheck1">&nbsp;</label>
    85                         </div>
    86                     </th>
    87                     <th>Nama</th>
    88                     <th>Tanggal</th>
    89                     <th>Presensi Datang</th>
    90                     <th>Presensi Pulang</th>
    91                     <th>Keterangan</th>
    92                 </tr>
    93             </thead>
    94             <tbody>
    95                 @foreach ($data as $dta)
    96                     <tr>
    97                         <td>
    98                             <td>{{ $dta->users->name }}</td>
    99                             <td>{{ $dta->date('d F Y', strtotime($dta->date)) }}</td>
   100                             <td>{{ $dta->time_in }}</td>
   101                             <td>{{ $dta->time_out }}</td>
   102                             <td>{{ $dta->keterangan }}</td>
   103                         </td>
   104                     </tr>
   105                 @endforeach
   106             </tbody>
   107         </table>
   108     </div>
   109 
```

Gambar 4.4 Gambar Source Code Halaman Presensi Pada Blade Presensi

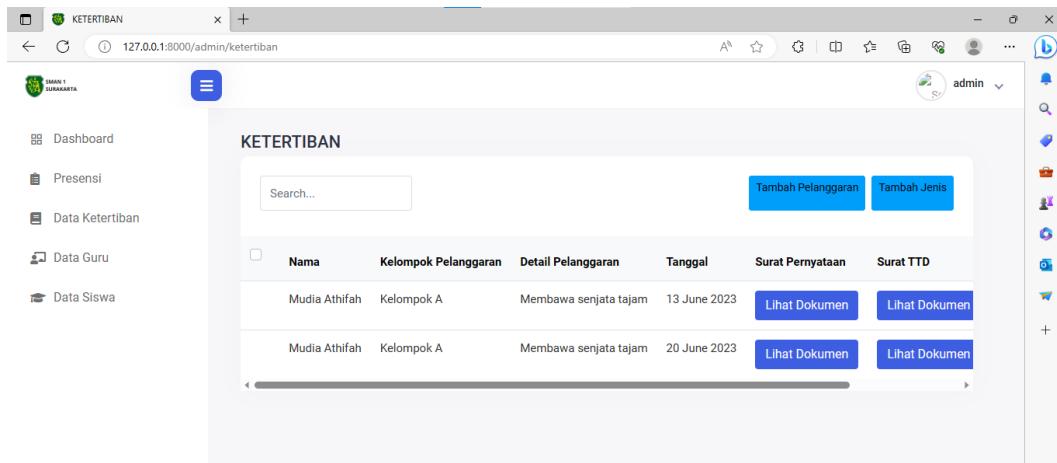


```
public function presensiadmin()
{
    $dateNow = Carbon::now();
    $date = $dateNow->toDateString();
    $data = Presensi::all();
    $users = User::all();
    return view('admin.presensi', compact('data', 'users', 'date'));
}
```

Gambar 4.5 Gambar Source Code Halaman Presensi

4.1.5 Halaman Data Ketertiban

Halaman data ketertiban ini akan menampilkan data pelanggaran dari semua siswa.



Gambar 4.6 Gambar Tampilan Halaman Data Ketertiban

Source Code Halaman Ketertiban

A screenshot of Visual Studio Code showing the source code for 'ketertiban_admin.blade.php'. The code is a Blade template with PHP logic. It includes a table structure with columns for action, name, group, detail, date, and documents. Buttons for viewing documents are included. The code uses Blade's @foreach loop to iterate over \$pelanggarans or \$pelanggaran.

Gambar 4.7 Gambar Source Code Halaman Data Ketertiban Pada Blade Ketertiban

```

public function index()
{
    $siswas = Siswa::all();
    $kelompoks = Kelompok::all();
    $pelanggarans = Pelanggaran::with(['detail_pelanggaran', 'detail_pelanggaran.kelompok'])->get();
    if ($request->search) {
        $pelanggarans = $pelanggarans->join('siswa', 'pelanggaran.siswa_id', 'siswa.id')->where('siswa.nama', 'LIKE', '%' . $request->search . '%')->get();
    }
    return view('admin.ketertiban_admin', compact('pelanggarans', 'kelompoks', 'siswas'));
}

```

Gambar 4.8 Gambar Source Code Halaman Data Ketertiban

4.1.6 Halaman Tambah Data Pelanggaran

Pada halaman ini admin dapat menambahkan pelanggaran dengan mengisi form sesuai dengan pelanggaran yang ada, admin akan mengisikan jenis pelanggaran, nama siswa, dan tanggal pengisian pelanggaran.

The screenshot shows a web application interface titled 'Form Pelanggaran'. On the left, there's a sidebar with navigation links: Dashboard, Presensi, Data Ketertiban, Data Guru, and Data Siswa. The main content area has a title 'DATA KETERTIBAN' and a sub-section 'Pelanggaran' with the instruction 'Silahkan masukkan data pelanggaran siswa dengan mengisi kolom dibawah ini'. Below this, there are two input fields: 'Pilih Pelanggaran' (dropdown menu showing 'Membawa senjata tajam') and 'Pilih Nama Siswa' (dropdown menu showing 'Mudia Athifah 12345'). There are also two text input fields: 'Nama Pencatatan' (containing 'admin') and 'Masukkan Tanggal' (containing 'mm/dd/yyyy'). At the bottom is a green 'Simpan' button.

Gambar 4.9 Gambar Halaman Tambah Data Pelanggaran

Source Code Form Tambah Pelanggaran

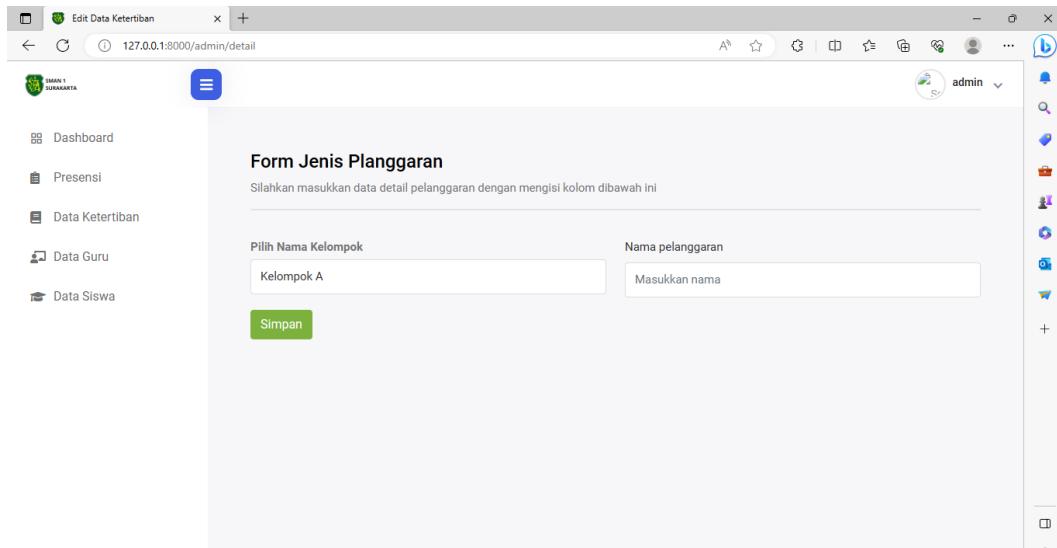
Gambar 4.10 Gambar Source Code Tambah Data Pelanggaran Pada Blade Ketertiban

```
public function store(Request $request,)  
{  
    $pel = new Pelanggaran();  
    $pel->detail_pelanggaran_id = $request->detail_pelanggaran_id;  
    $pel->siswa_id = $request->siswa_id;  
    $pel->nama_pencatat = $request->nama_pencatat;  
    $pel->tanggal = $request->tanggal;  
    $pel->save();  
    // dd($pel);  
    return redirect('/admin/ketertiban');  
    // $request->all();  
}
```

Gambar 4.11 Gambar Source Code Tambah Data Pelanggaran

4.1.7 Halaman Tambah Jenis Detail Pelanggaran

Pada halaman ini admin dapat menambahkan jenis data pelanggaran apabila adanya penambahan dari aturan buku saku sekolah.



Gambar 4.12 Gambar Halaman Tambah Jenis Pelanggaran

Source Code Halaman Tambah Jenis pelanggaran

```
File Edit Selection View Go Run Terminal Help
resources > views > admin > formdetail.blade.php - smansanew - Visual Studio Code
resources > views > admin > formdetail.blade.php
formdetail.blade.php
resources > views > admin > formdetail.blade.php < html > body > div.main-wrapper > div.page-wrapper > div.content.container-fluid > div.row > div.col-12
resources > views > admin > formdetail.blade.php <div class="col-12">
  <div class="card-body">
    <h4 class="header-title">Form Jenis Planggaran</h4>
    <p class="text-muted font-14">Silahkan masukkan data detail pelanggaran dengan mengisi kolom dibawah ini</p>
    <ul class="nav nav-tabs nav-bordered mb-3">
      </ul> <!-- end nav-->
    <form action="/admin/kirje" method="POST">
      @csrf
      <div class="tab-pane show active" id="select2-preview">
        <div class="row">
          <div class="col-6">
            <div class="mb-1 fw-bold text-muted">Pilih Nama Kelompok</div>
            <select name="kelompok_id" class="form-control select2" data-toggle="select2">
              @foreach ($kelompoks as $kelompok)
                <option value="{{ $kelompok->id }}>{{ $kelompok->name_kelompok }}</option>
              @endforeach
            </select>
          </div> <!-- end col -->
          <div class="col-6">
            <label class="form-label">Nama pelanggaran</label>
            <input type="text" name="name_detail" class="form-control" data-provide="typeahead" id="the-basics" placeholder="Masukkan nama">
          </div>
        </div>
        <div class="col-md-6">
          <button type="submit" class="btn btn-success">Simpan</button>
        </div>
      </div>
    </form>
  </div>
</div>
```

The screenshot shows the Visual Studio Code interface with the file 'formdetail.blade.php' open. The code is a Blade template for a form. It includes a header section with a title and a note. Below this is a navigation bar with tabs. The main content is a form with two columns. The first column contains a dropdown labeled 'Pilih Nama Kelompok' populated with data from '\$kelompoks' (looped through \$kelompok). The second column contains a text input labeled 'Nama pelanggaran' with a placeholder 'Masukkan nama'. At the bottom of the form is a green 'Simpan' button. The code uses Bootstrap classes like 'col-12', 'card-body', 'nav-tabs', 'form-control', and 'typeahead'.

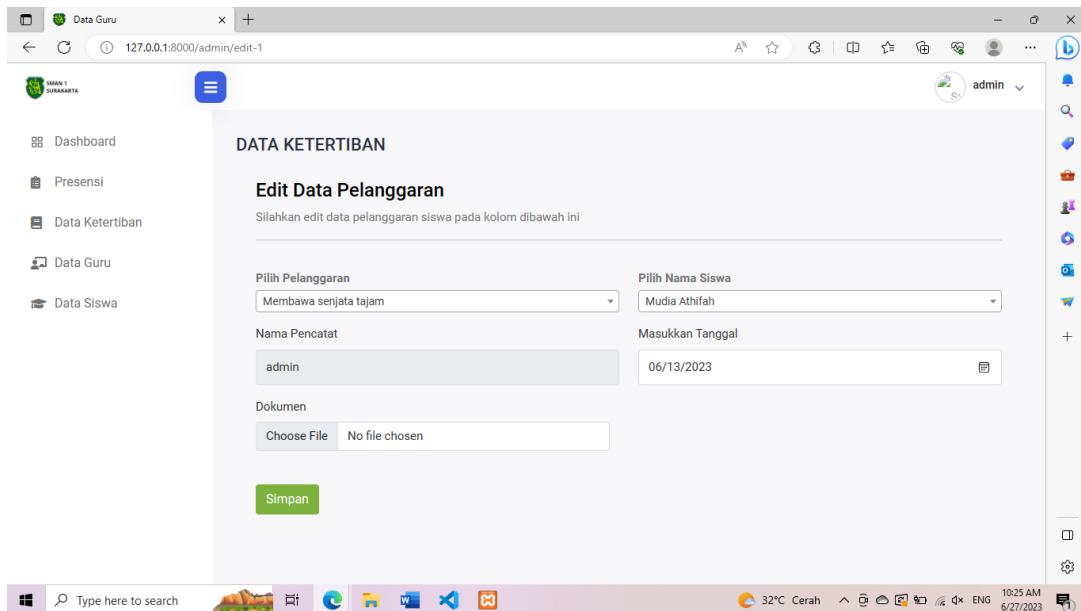
Gambar 4.13 Gambar Source Code Tambah Jenis Pelanggaran pada Bladenya

```
public function store(Request $request)
{
    $del = new DetailPelanggaran();
    $del->kelompok_id = $request->kelompok_id;
    $del->nama_detail = $request->nama_detail;
    $del->save();
    // dd($pel);
    return redirect('/admin/ketertiban');
    // $request->all();
}
```

Gambar 4.14 Gambar Source Code Tambah Jenis Pelanggaran

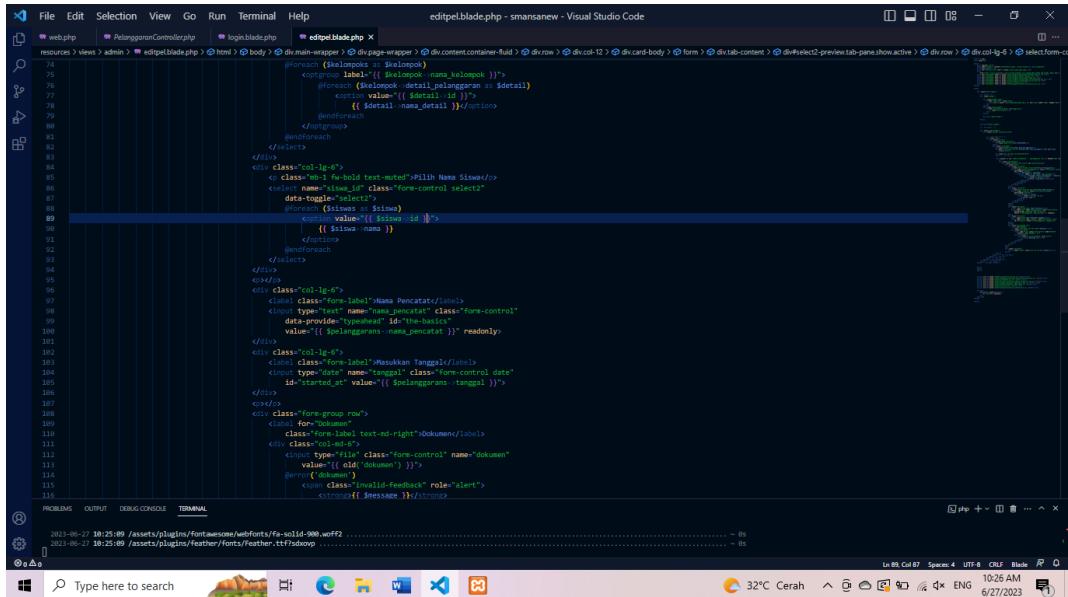
4.1.8 Halaman Edit Pelanggaran

Pada halaman ini admin dapat mengedit pelanggaran yang terdapat dan juga dapat mengunggah surat pernyataan untuk pelanggaran siswa.



Gambar 4.15 Gambar Tampilan Halaman Edit Pelanggaran

Source Code Halaman Edit Pelanggaran

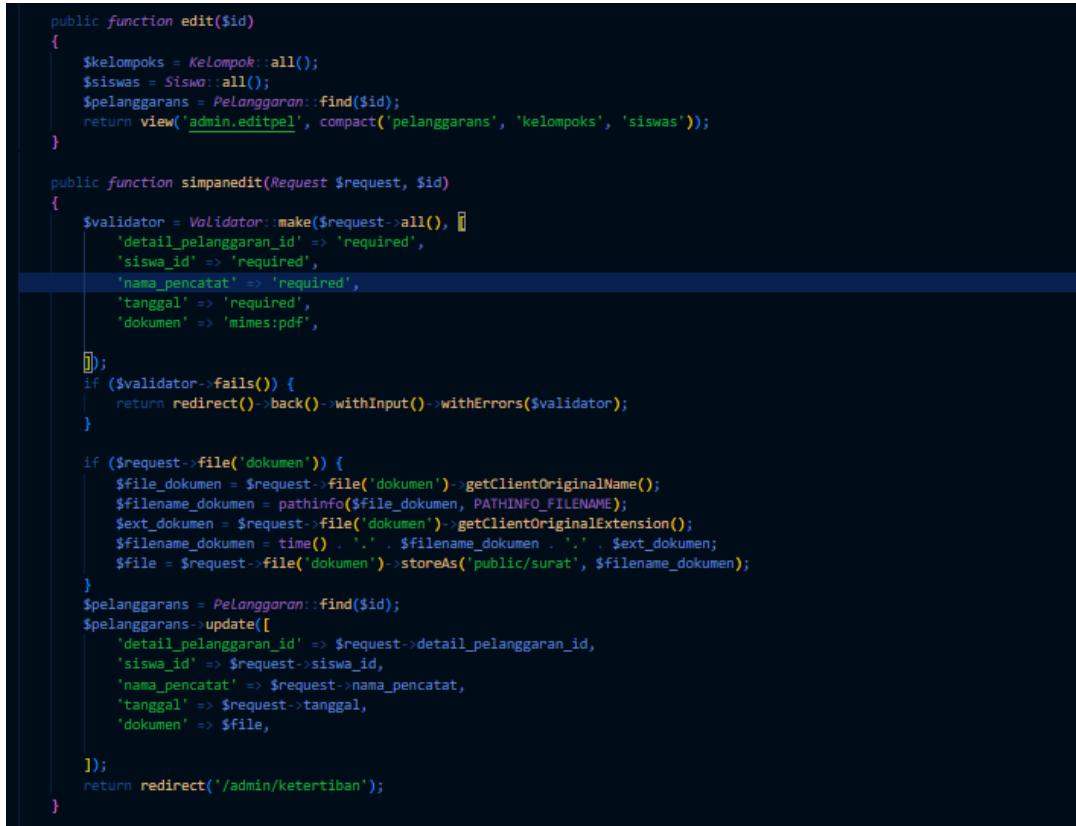


```

resources > views > admin > editpel.blade.php > html > body > div.main-wrapper > div.page-wrapper > div.content.container-fluid > div.row > div.col-12 > div.card-body > form > div.tab-content > div#select2-preview.tab-pane.show.active > div.row > div.col-lg-6 > select.form-control
    74
    75
    76
    77
    78
    79
    80
    81
    82
    83
    84
    85
    86
    87
    88
    89
    90
    91
    92
    93
    94
    95
    96
    97
    98
    99
    100
    101
    102
    103
    104
    105
    106
    107
    108
    109
    110
    111
    112
    113
    114
    115
    116
    117
    118
    119
    120
    121
    122
    123
    124
    125
    126
    127
    128
    129
    130
    131
    132
    133
    134
    135
    136
    137
    138
    139
    140
    141
    142
    143
    144
    145
    146
    147
    148
    149
    150
    151
    152
    153
    154
    155
    156
    157
    158
    159
    160
    161
    162
    163
    164
    165
    166
    167
    168
    169
    170
    171
    172
    173
    174
    175
    176
    177
    178
    179
    180
    181
    182
    183
    184
    185
    186
    187
    188
    189
    190
    191
    192
    193
    194
    195
    196
    197
    198
    199
    200
    201
    202
    203
    204
    205
    206
    207
    208
    209
    210
    211
    212
    213
    214
    215
    216
    217
    218
    219
    220
    221
    222
    223
    224
    225
    226
    227
    228
    229
    230
    231
    232
    233
    234
    235
    236
    237
    238
    239
    240
    241
    242
    243
    244
    245
    246
    247
    248
    249
    250
    251
    252
    253
    254
    255
    256
    257
    258
    259
    260
    261
    262
    263
    264
    265
    266
    267
    268
    269
    270
    271
    272
    273
    274
    275
    276
    277
    278
    279
    280
    281
    282
    283
    284
    285
    286
    287
    288
    289
    290
    291
    292
    293
    294
    295
    296
    297
    298
    299
    300
    301
    302
    303
    304
    305
    306
    307
    308
    309
    310
    311
    312
    313
    314
    315
    316
    317
    318
    319
    320
    321
    322
    323
    324
    325
    326
    327
    328
    329
    330
    331
    332
    333
    334
    335
    336
    337
    338
    339
    340
    341
    342
    343
    344
    345
    346
    347
    348
    349
    350
    351
    352
    353
    354
    355
    356
    357
    358
    359
    360
    361
    362
    363
    364
    365
    366
    367
    368
    369
    370
    371
    372
    373
    374
    375
    376
    377
    378
    379
    380
    381
    382
    383
    384
    385
    386
    387
    388
    389
    390
    391
    392
    393
    394
    395
    396
    397
    398
    399
    400
    401
    402
    403
    404
    405
    406
    407
    408
    409
    410
    411
    412
    413
    414
    415
    416
    417
    418
    419
    420
    421
    422
    423
    424
    425
    426
    427
    428
    429
    430
    431
    432
    433
    434
    435
    436
    437
    438
    439
    440
    441
    442
    443
    444
    445
    446
    447
    448
    449
    450
    451
    452
    453
    454
    455
    456
    457
    458
    459
    460
    461
    462
    463
    464
    465
    466
    467
    468
    469
    470
    471
    472
    473
    474
    475
    476
    477
    478
    479
    480
    481
    482
    483
    484
    485
    486
    487
    488
    489
    490
    491
    492
    493
    494
    495
    496
    497
    498
    499
    500
    501
    502
    503
    504
    505
    506
    507
    508
    509
    510
    511
    512
    513
    514
    515
    516
    517
    518
    519
    520
    521
    522
    523
    524
    525
    526
    527
    528
    529
    530
    531
    532
    533
    534
    535
    536
    537
    538
    539
    540
    541
    542
    543
    544
    545
    546
    547
    548
    549
    550
    551
    552
    553
    554
    555
    556
    557
    558
    559
    560
    561
    562
    563
    564
    565
    566
    567
    568
    569
    570
    571
    572
    573
    574
    575
    576
    577
    578
    579
    580
    581
    582
    583
    584
    585
    586
    587
    588
    589
    590
    591
    592
    593
    594
    595
    596
    597
    598
    599
    600
    601
    602
    603
    604
    605
    606
    607
    608
    609
    610
    611
    612
    613
    614
    615
    616
    617
    618
    619
    620
    621
    622
    623
    624
    625
    626
    627
    628
    629
    630
    631
    632
    633
    634
    635
    636
    637
    638
    639
    640
    641
    642
    643
    644
    645
    646
    647
    648
    649
    650
    651
    652
    653
    654
    655
    656
    657
    658
    659
    660
    661
    662
    663
    664
    665
    666
    667
    668
    669
    670
    671
    672
    673
    674
    675
    676
    677
    678
    679
    680
    681
    682
    683
    684
    685
    686
    687
    688
    689
    690
    691
    692
    693
    694
    695
    696
    697
    698
    699
    700
    701
    702
    703
    704
    705
    706
    707
    708
    709
    710
    711
    712
    713
    714
    715
    716
    717
    718
    719
    720
    721
    722
    723
    724
    725
    726
    727
    728
    729
    730
    731
    732
    733
    734
    735
    736
    737
    738
    739
    740
    741
    742
    743
    744
    745
    746
    747
    748
    749
    750
    751
    752
    753
    754
    755
    756
    757
    758
    759
    760
    761
    762
    763
    764
    765
    766
    767
    768
    769
    770
    771
    772
    773
    774
    775
    776
    777
    778
    779
    780
    781
    782
    783
    784
    785
    786
    787
    788
    789
    790
    791
    792
    793
    794
    795
    796
    797
    798
    799
    800
    801
    802
    803
    804
    805
    806
    807
    808
    809
    810
    811
    812
    813
    814
    815
    816
    817
    818
    819
    820
    821
    822
    823
    824
    825
    826
    827
    828
    829
    830
    831
    832
    833
    834
    835
    836
    837
    838
    839
    840
    841
    842
    843
    844
    845
    846
    847
    848
    849
    850
    851
    852
    853
    854
    855
    856
    857
    858
    859
    860
    861
    862
    863
    864
    865
    866
    867
    868
    869
    870
    871
    872
    873
    874
    875
    876
    877
    878
    879
    880
    881
    882
    883
    884
    885
    886
    887
    888
    889
    890
    891
    892
    893
    894
    895
    896
    897
    898
    899
    900
    901
    902
    903
    904
    905
    906
    907
    908
    909
    910
    911
    912
    913
    914
    915
    916
    917
    918
    919
    920
    921
    922
    923
    924
    925
    926
    927
    928
    929
    930
    931
    932
    933
    934
    935
    936
    937
    938
    939
    940
    941
    942
    943
    944
    945
    946
    947
    948
    949
    950
    951
    952
    953
    954
    955
    956
    957
    958
    959
    960
    961
    962
    963
    964
    965
    966
    967
    968
    969
    970
    971
    972
    973
    974
    975
    976
    977
    978
    979
    980
    981
    982
    983
    984
    985
    986
    987
    988
    989
    990
    991
    992
    993
    994
    995
    996
    997
    998
    999
    999

```

Gambar 4.16 Gambar Source Code Halaman Edit Pelanggaran Pada Bladanya



```

public function edit($id)
{
    $kelompoks = Kelompok::all();
    $siswas = Siswa::all();
    $pelanggarans = Pelanggaran::find($id);
    return view('admin.editpel', compact('pelanggarans', 'kelompoks', 'siswas'));
}

public function simpanedit(Request $request, $id)
{
    $validator = Validator::make($request->all(), [
        'detail_pelanggaran_id' => 'required',
        'siswa_id' => 'required',
        'nama_pencatatan' => 'required',
        'tanggal' => 'required',
        'dokumen' => 'mimes:pdf',
    ]);
    if ($validator->fails()) {
        return redirect()->back()->withInput()->withErrors($validator);
    }

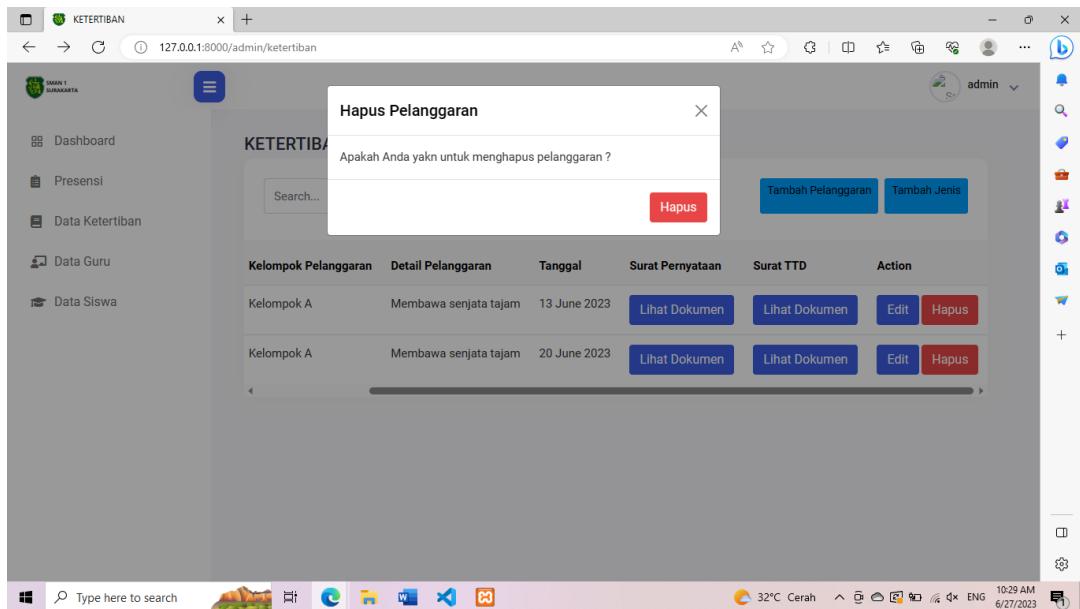
    if ($request->file('dokumen')) {
        $file_dokumen = $request->file('dokumen')->getClientOriginalName();
        $filename_dokumen = pathinfo($file_dokumen, PATHINFO_FILENAME);
        $ext_dokumen = $request->file('dokumen')->getClientOriginalExtension();
        $filename_dokumen = time() . '.' . $filename_dokumen . '.' . $ext_dokumen;
        $file = $request->file('dokumen')->storeAs('public/surat', $filename_dokumen);
    }
    $pelanggarans = Pelanggaran::find($id);
    $pelanggarans->update([
        'detail_pelanggaran_id' => $request->detail_pelanggaran_id,
        'siswa_id' => $request->siswa_id,
        'nama_pencatatan' => $request->nama_pencatatan,
        'tanggal' => $request->tanggal,
        'dokumen' => $file,
    ]);
}

```

Gambar 4.17 Gambar Source Code Halaman Edit Pelanggaran

4.1.9 Halaman Hapus Data Pelanggaran

Pada halaman ini admin dapat menghapus data pelanggaran



Gambar 4.18 Gambar Tampilan Halaman Hapus Data Pelanggaran

Source Code Hapus Data Pelanggaran

```
<button type="button" class="btn btn-danger"
       data-bs-toggle="modal"
       data-bs-target="#hapus({ $pelanggaran->id })">
    Hapus
</button>
</td>
</tr>
@endforeach
</tbody>
</table>

</div>
<!-- Modal -->
@foreach ($pelanggarans as $pelanggaran)
    <div class="modal fade" id="hapus({ $pelanggaran->id })"
        data-bs-backdrop="static" data-bs-keyboard="false" tabindex="-1"
        aria-labelledby="staticBackdropLabel" aria-hidden="true">
        <div class="modal-dialog">
            <div class="modal-content">
                <div class="modal-header">
                    <h1 class="modal-title fs-5" id="staticBackdropLabel">
                        Hapus
                        Pelanggaran</h1>
                    <button type="button" class="btn-close"
                           data-bs-dismiss="modal"
                           aria-label="Close"></button>
                </div>
                <div class="modal-body">
                    Apakah Anda yakin untuk menghapus pelanggaran ?
                </div>
                <div class="modal-footer br">
                    <a href="{{ url('/admin/hapus/' . $pelanggaran->id) }}"
                       class="btn btn-danger">Hapus</a>
                </div>
            </div>
        </div>
    </div>

```

Gambar 4.19 Gambar Source Code Halaman Hapus Data Pelanggaran

```
public function destroy($id)
{
    Pelanggaran::destroy($id);
    return back();
}
```

Gambar 4.20 Gambar Source Code Controller Hapus Data Pelanggaran

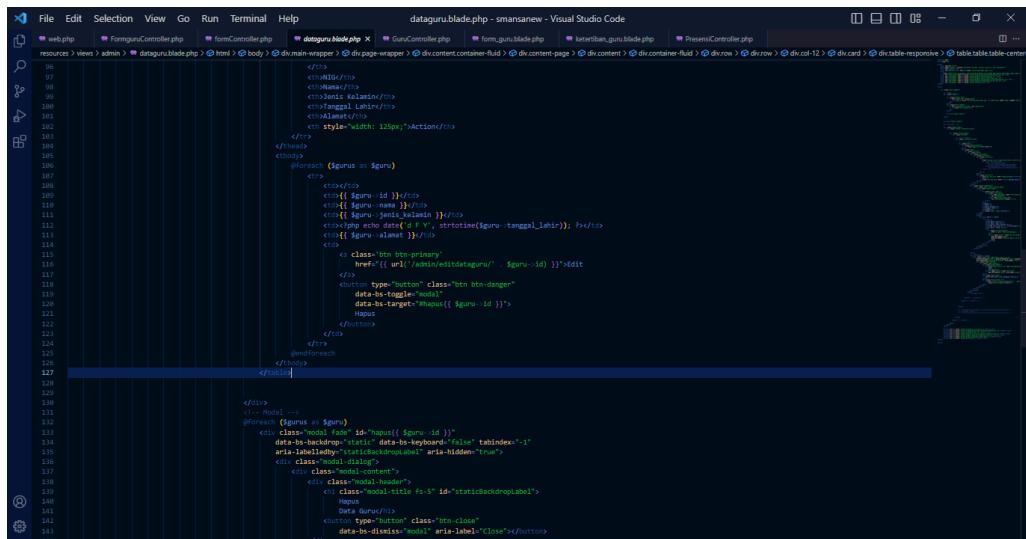
4.1.10 Halaman Data Guru

Pada halaman ini admin dapat melihat daftar guru yang sudah telah ditambahkan.

	NIG	Nama	Jenis Kelamin	Tanggal Lahir	Alamat	Action
<input type="checkbox"/>	1234567890123456	Guru Mudia	Perempuan	09 June 2023	Solo	<button>Edit</button> <button>Hapus</button>
<input type="checkbox"/>	9876543219876543	Sherly Munica	Perempuan	14 June 2023	Solo	<button>Edit</button> <button>Hapus</button>

Gambar 4.21 Gambar tampilan halaman data guru

Source Code Halaman Data Guru



```
resources > views > admin > dataguru.blade.php > HTML > body > div.main-wrapper > div.page-wrapper > div.content.container-fluid > div.content-page > div.content > div.container-fluid > div.row > div.col-12 > div.card > div.table-responsive > table.table.table-centered
 96
 97
 98
 99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
```

Gambar 4.22 Gambar source code blade data guru

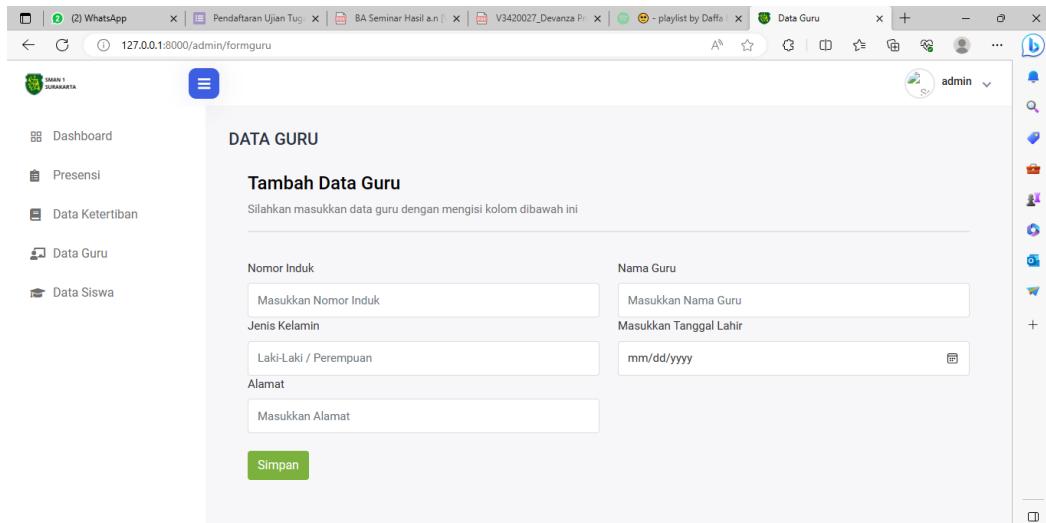
```
public function index()
{
    $gurus = Guru::all();
    return view('admin.dataguru', compact('gurus'));

}
```

Gambar 4.23 Gambar source code controller data guru

4.1.11 Halaman Tambah Data Guru

Pada halaman ini admin dapat menambahkan data guru dengan mengisi form dengan data yang ada.



Gambar 4.24 Gambar Tampilan Halaman tambah data guru

Source Code Halaman Tambah Guru

```
resources > views > admin > formdataguru.blade.php
```

```
<?php web.php <?> formdataguru.blade.php <?> file <?> Selection <?> View <?> Go <?> Run <?> Terminal <?> Help
```

```
formdataguru.blade.php - smansanew - Visual Studio Code
```

```
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
```

```
<div class="header-title">Tambah Data Guru</div>
<div class="text-muted font-14">Silahkan masukkan data guru dengan mengisi kolom dibawah ini</div>
<div class="nav nav-tabs nav-bordered mb-3">
</div> <!-- end nav -->
<form action="/admin/simpanFormguru" method="POST">
@csrf
<div class="tab-content">
<div class="tab-pane show active" id="select2-preview">
<div class="row">
<div class="col-lg-6">
<label class="form-label">Nomor Induk</label>
<input type="text" name="nomorInduk" class="form-control" data-provide="typeahead" id="the-basic" placeholder="Masukkan Nomor Induk">
</div>
<div class="col-lg-6">
<label class="form-label">Nama Guru</label>
<input type="text" name="nama" class="form-control" data-provide="typeahead" id="the-basic" placeholder="Masukkan Nama Guru">
</div>
<div class="col-lg-6">
<label class="form-label">Jenis Kelamin</label>
<input type="text" name="jenis_kelamin" class="form-control" data-provide="typeahead" id="the-basic" placeholder="Laki-Laki / Perempuan">
</div>
<div class="col-lg-6">
<label class="form-label">Masukkan Tanggal Lahir</label>
<input type="date" name="tanggal_lahir" class="form-control date" id="started_at" value="{{ old('started_at') }}"/>
</div>
<div class="col-lg-6">
<label class="form-label">Alamat</label>
<input type="text" name="alamat" class="form-control" data-provide="typeahead" id="the-basic" placeholder="Masukkan Alamat">
</div>
</div>
<div class="col-md-6">
<button type="submit" class="btn btn-success">Simpan</button>
</div>
</div>
```

Gambar 4.25 Gambar source code form tambah guru

```

public function store(Request $request)
{
    Guru::create([
        'id' => $request->id,
        'nama' => $request->nama,
        'jenis_kelamin' => $request->jenis_kelamin,
        'tanggal_lahir' => $request->tanggal_lahir,
        'alamat' => $request->alamat,
    ]);
    return redirect('/admin/dataguru');
}

```

Gambar 4.26 Gambar source code controller form guru

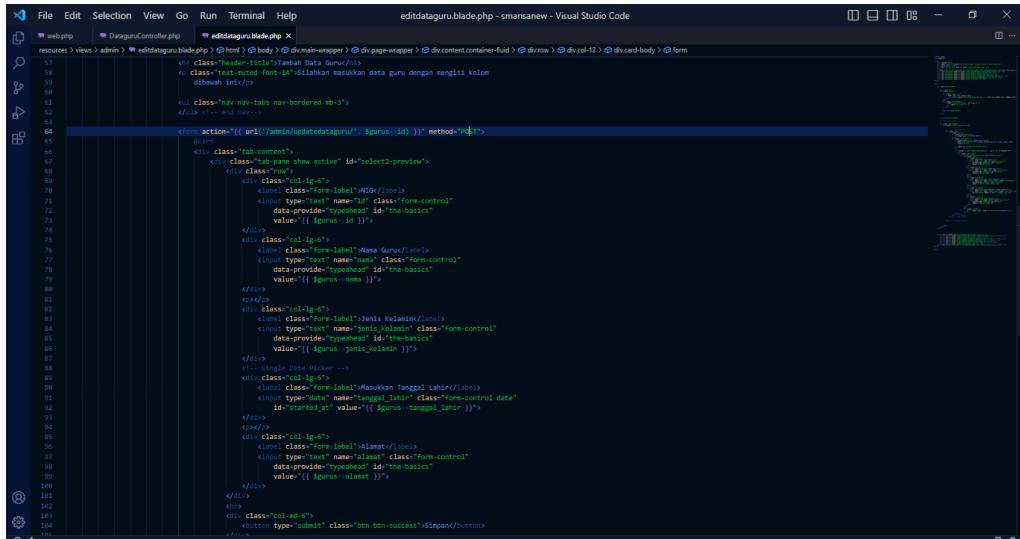
4.1.12 Halaman Edit Data Guru

Pada halaman ini admin dapat mengubah data guru dengan data yang baru

The screenshot shows a web application interface for managing teacher data. The top navigation bar includes links for WhatsApp, Pendaftaran Ujian Tug, BA Seminar Hasil a.n, V3420027_Devanza Pi, playlist by Daffa, Data Guru, and other system-related links. The main content area is titled 'DATA GURU' and 'Tambah Data Guru'. It contains input fields for 'NIG' (with value 1234567890123456), 'Nama Guru' (Guru Mudia), 'Jenis Kelamin' (Perempuan), 'Masukkan Tanggal Lahir' (06/09/2023), and 'Alamat' (Solo). A green 'Simpan' button is located at the bottom of the form.

Gambar 4.27 Gambar tampilan halaman edit data guru

Source Code Edit Data Guru



```
<!-- action="{{ url('/admin/updatadataguru/' . $gurus->id) }}" method="POST"-->
<div class="tab-content">
    <div class="tab-pane show active" id="select2-preview">
        <div class="form-group row">
            <div class="col-lg-4">
                <label class="form-label">Nama Guru</label>
                <input type="text" name="nama" class="form-control" value="{{ $gurus->nama }}"/>
            </div>
            <div class="col-lg-4">
                <label class="form-label">Jenis Kelamin</label>
                <select class="form-select" name="jenis_kelamin" id="the-basics" data-provide="typeahead" data-value="{{ $gurus->jenis_kelamin }}"/>
            </div>
            <div class="col-lg-4">
                <label class="form-label">Tanggal Lahir</label>
                <input type="date" name="tanggal_lahir" class="form-control" id="selected_at" value="{{ $gurus->tanggal_lahir }}"/>
            </div>
            <div class="col-lg-4">
                <label class="form-label">Alamat</label>
                <input type="text" name="alamat" class="form-control" value="{{ $gurus->alamat }}"/>
            </div>
        </div>
        <div class="col-md-6">
            <button type="submit" class="btn btn-success">Simpan</button>
        </div>
    </div>

```

Gambar 4.28 Gambar Source code edit data guru



```
public function updatedataguru(Request $request, $id)
{
    $validator = Validator::make($request->all(), [
        'id' => 'required',
        'nama' => 'required',
        'jenis_kelamin' => 'required',
        'tanggal_lahir' => 'required',
        'alamat' => 'required',
    ]);

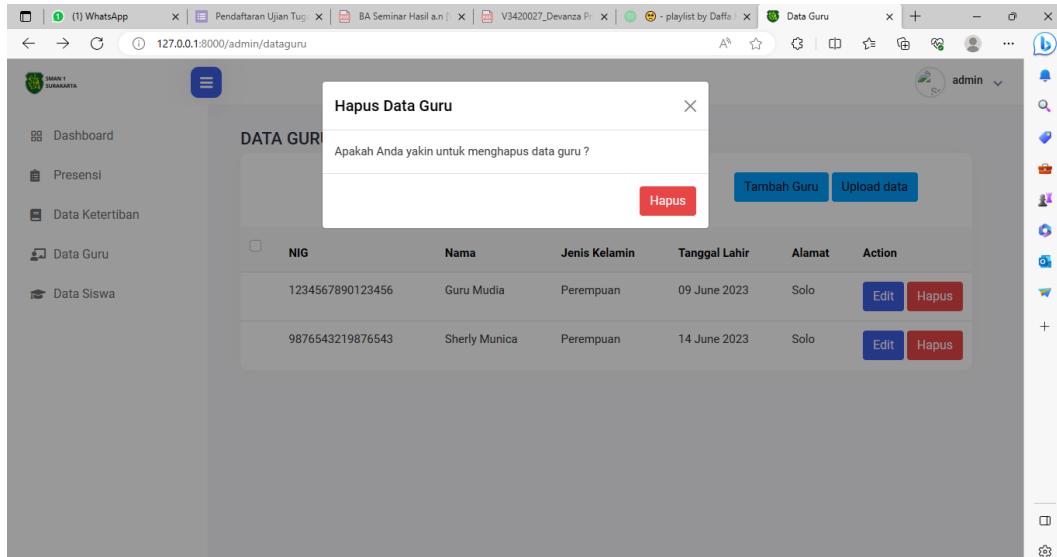
    if ($validator->fails()) {
        return redirect()->back()->withInput()->withErrors($validator);
    }

    $gurus = Guru::find($id);
    $gurus->update([
        'id' => request('id'),
        'nama' => request('nama'),
        'jenis_kelamin' => request('jenis_kelamin'),
        'tanggal_lahir' => request('tanggal_lahir'),
        'alamat' => request('alamat'),
    ]);
    return redirect('/admin/dataguru');
}
```

Gambar 4.29 Gambar Source Code controller edit data guru

4.1.13 Halaman Hapus Data Guru

Pada halaman ini admin bisa menghapus data guru yang telah diinputkan.



Gambar 4.30 Gambar tampilan halaman hapus data guru

Source Code Halaman Hapus Data Guru

```
<button type="button" class="btn btn-danger"
       data-bs-toggle="modal"
       data-bs-target="#hapus{{ $guru->id }}">
    Hapus
</button>
</td>
</tr>
@endforeach
</tbody>
</table>

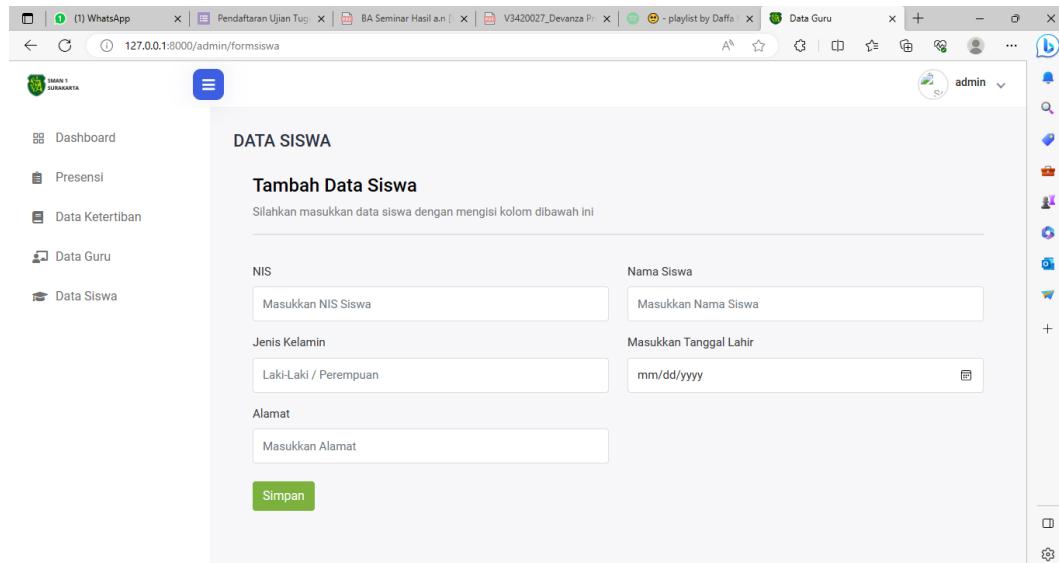
</div>

@foreach ($gurus as $guru)
    <div class="modal fade" id="hapus{{ $guru->id }}"
        data-bs-backdrop="static" data-bs-keyboard="false" tabindex="-1"
        aria-labelledby="staticBackdropLabel" aria-hidden="true">
        <div class="modal-dialog">
            <div class="modal-content">
                <div class="modal-header">
                    <h1 class="modal-title fs-5" id="staticBackdropLabel">
                        Hapus
                        Data Guru</h1>
                    <button type="button" class="btn-close"
                           data-bs-dismiss="modal" aria-label="Close"></button>
                </div>
                <div class="modal-body">
                    Apakah Anda yakin untuk menghapus data guru ?
                </div>
                <div class="modal-footer br">
                    <a href="{{ url('/admin/hapusdataguru/' . $guru->id) }}"
                       class="btn btn-danger">Hapus</a>
                </div>
            </div>
        @endforeach
    </div>
</div>
```

Gambar 4.31 Gambar source code hapus data

4.1.14 Halaman Tambah Data Siswa

Pada halaman ini admin dapat menambahkan data siswa dengan mengisi form tambah siswa



Gambar 4.32 Gambar tampilan halaman tambah data siswa

Source Code Halaman Tambah Data Siswa

```
<form action="/admin/simpansiswa" method="POST">
    @csrf
    <div class="tab-content">
        <div class="tab-pane show active" id="select2-preview">
            <div class="row">
                <div class="col-lg-6">
                    <label class="form-label">NIS</label>
                    <input type="text" name="nis" class="form-control" data-provide="typeahead" id="the-basics" placeholder="Masukkan NIS Siswa" data-select2="true" data-select2-id="1" data-select2-type="text">
                </div>
                <div class="col-lg-6">
                    <label class="form-label">Nama Siswa</label>
                    <input type="text" name="name" class="form-control" data-provide="typeahead" id="the-basics" placeholder="Masukkan Nama Siswa" data-select2="true" data-select2-id="2" data-select2-type="text">
                </div>
            </div>
            <div class="row">
                <div class="col-lg-6">
                    <label class="form-label">Jenis Kelamin</label>
                    <input type="text" name="jenis_kelamin" class="form-control" data-provide="typeahead" id="the-basics" placeholder="Laki-Laki / Perempuan" data-select2="true" data-select2-id="3" data-select2-type="text">
                </div>
                <div class="col-lg-6">
                    <label class="form-label">Masukkan Tanggal Lahir</label>
                    <input type="date" name="tanggal_lahir" class="form-control date" id="startdate_at" value="{{ old('startdate_at') }}">
                </div>
            </div>
            <div class="row">
                <div class="col-lg-6">
                    <label class="form-label">Alamat</label>
                    <input type="text" name="alamat" class="form-control" data-provide="typeahead" id="the-basics" placeholder="Masukkan Alamat" data-select2="true" data-select2-id="4" data-select2-type="text">
                </div>
            </div>
        </div>
    </div>
    <div class="col-md-12">
        <button type="submit" class="btn btn-success">Simpan</button>
    </div>
</form>
```

Gambar 4.33 Gambar Source Code halaman tambah data siswa

```

    public function store(Request $request)
    {
        Siswa::create([
            'id' => $request->id,
            'nama' => $request->nama,
            'jenis_kelamin' => $request->jenis_kelamin,
            'tanggal_lahir' => $request->tanggal_lahir,
            'alamat' => $request->alamat,
        ]);
        return redirect('/admin/datasiswa');
    }
}

```

Gambar 4.34 Gambar Source code controller tambah siswa

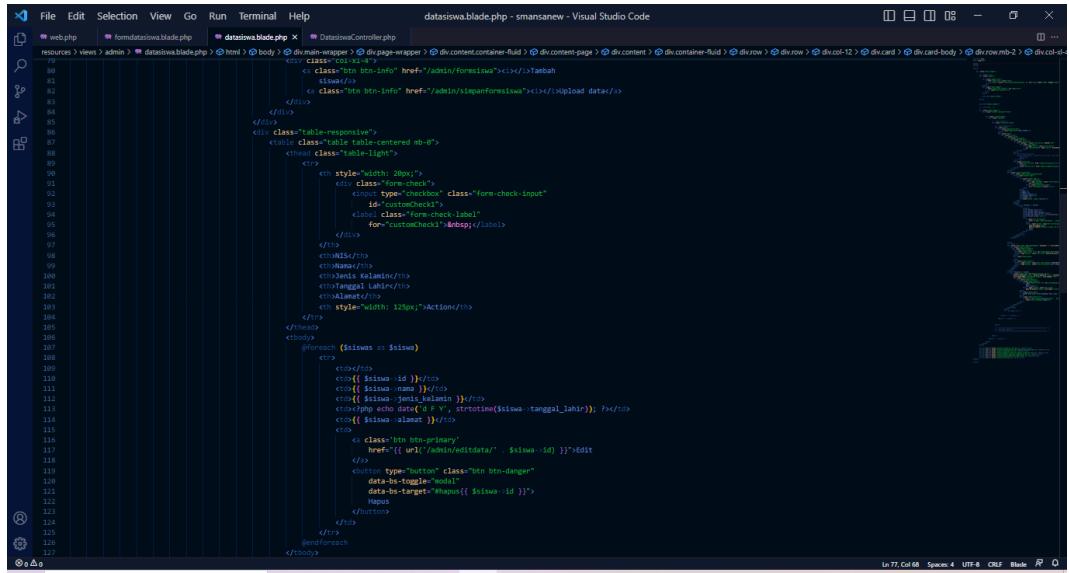
4.1.15 Halaman Data Siswa

Pada halaman ini admin dapat melihat data siswa yang telah tertambahkan.

	NIS	Nama	Jenis Kelamin	Tanggal Lahir	Alamat	Action
<input type="checkbox"/>	12345	Mudia Athifah	Perempuan	08 June 2023	Solo	<button>Edit</button> <button>Hapus</button>
<input type="checkbox"/>	23456	Diana	Perempuan	09 June 2023	Solo	<button>Edit</button> <button>Hapus</button>
<input type="checkbox"/>	45678	Diana	Perempuan	22 June 2023	Solo	<button>Edit</button> <button>Hapus</button>

Gambar 4.35 Gambar tampilan halaman data siswa

Source Code Halaman Data Siswa



The screenshot shows the Visual Studio Code interface with the file 'datisiswa.blade.php' open. The code is a Blade template for displaying student data. It includes a header section with a back button and a search bar. Below this is a table with columns for id, name, address, and a date column. Each row contains edit and delete buttons. The code uses PHP and Blade syntax, including loops and conditionals.

```
<div>
    <a href="#"><--> Back
    <input type="text" placeholder="Search" />
</div>

<div>
    <div>
        <div>
            <div>
                <div>
                    <div>
                        <div>
                            <div>
                                <div>
                                    <div>
                                        <div>
                                            <div>
                                                <div>
                                                    <div>
                                                        <div>
                                                            <div>
                                                                <div>
                                                                    <div>
                                                                        <div>
                                                                            <div>
                                                                                <div>
                                                                                    <div>
                                                                                        <div>
                                                                                            <div>
                                                                                                <div>
                                                                                                    <div>
                                                                                                        <div>
                                                                                                            <div>
                                                                                                                <div>
                                                                                                                    <div>
                                                                                                                        <div>
                                                                                                                            <div>
                                                                                                                                <div>
                                                                                                                                    <div>
                                                                                                                                        <div>
                                                                                                                                            <div>
                                                                                                                                                <div>
                                                                                                                                                    <div>
................................................................
```

Gambar 4.36 Gambar Source Code Halaman Data Siswa



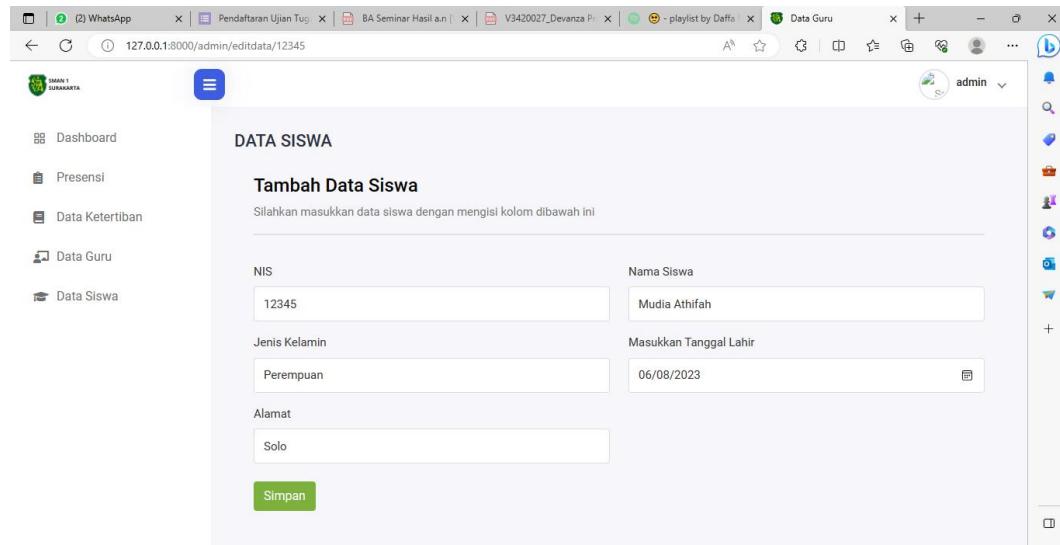
The screenshot shows the Visual Studio Code interface with the file 'DatiSiswaController.php' open. The code defines an 'index' method. It starts by querying all students from the 'Siswa' model. If a search term is provided via the request, it filters the results using the 'where' method. Finally, it returns a view named 'admin.datisiswa' with the filtered data as a compacted variable.

```
public function index()
{
    $siswas = Siswa::all();
    if(request('search')){
        $siswas = Siswa::where('nama', 'LIKE', '%' . request('search') . '%')->get();
    }
    return view('admin.datisiswa', compact('siswas'));
}
```

Gambar 4.37 Gambar Source Code Controller Halaman Data siswa

4.1.16 Halaman Edit Data Siswa

Pada halaman ini admin dapat mengubah data siswa dengan data yang benar.



Gambar 4.38 Gambar Tampilan edit data siswa

Source Code Halaman Edit Data Siswa

```
<div class="header-title">Tambah Data Siswa</div>
<div><p>Silahkan masukkan data siswa dengan mengisi kolom dibawah ini</p>
</div>

<div class="nav nav-tabs nav-bordered mb-3">
</div> <!-- end nav -->

<form action="{{ url('/admin/update/' . $siswa->id) }}" method="POST">
    @csrf
    <div class="tab-content">
        <div class="tab-pane show active" id="select2-preview">
            <div class="col-lg-6">
                <label class="form-label">NIS</label>
                <input type="text" name="nis" class="form-control" data-provide="typeahead" id="the-nis" value="{{ $siswa->nis }}"/>
            </div>
            <div class="col-lg-6">
                <label class="form-label">Nama Siswa</label>
                <input type="text" name="nama" class="form-control" data-provide="typeahead" id="the-nama" value="{{ $siswa->nama }}"/>
            </div>
        </div>
        <div class="col-lg-6">
            <label class="form-label">Jenis Kelamin</label>
            <input type="text" name="jenis_kelamin" class="form-control" data-provide="typeahead" id="the-basic" value="{{ $siswa->jenis_kelamin }}"/>
        </div>
        <div class="col-lg-6">
            <label class="form-label">Masukkan Tanggal Lahir</label>
            <input type="date" name="tanggal_lahir" class="form-control" id="started_at" value="{{ $siswa->tanggal_lahir }}"/>
        </div>
    </div>
    <div class="col-lg-6">
        <label class="form-label">Alamat</label>
        <input type="text" name="alamat" class="form-control" data-provide="typeahead" id="the-alamat" value="{{ $siswa->alamat }}"/>
    </div>
    <br>
    <div class="col-md-6" style="text-align: right;">
        <button type="submit" class="btn btn-success">Simpan</button>
    </div>
</form>
```

Gambar 4.39 Gambar Source Code edit data siswa

```

public function editdatasiswa($id)
{
    $siswas= Siswa::find($id);
    return view('admin.editdatasiswa', compact('siswas'));
}

public function updatedatasiswa(Request $request, $id)
{
    $validator = Validator::make($request->all(), [
        'id' => 'required',
        'nama' => 'required',
        'jenis_kelamin' => 'required',
        'tanggal_lahir' => 'required',
        'alamat' => 'required',
    ]);

    if ($validator->fails()) {
        return redirect()->back()->withInput()->withErrors($validator);
    }

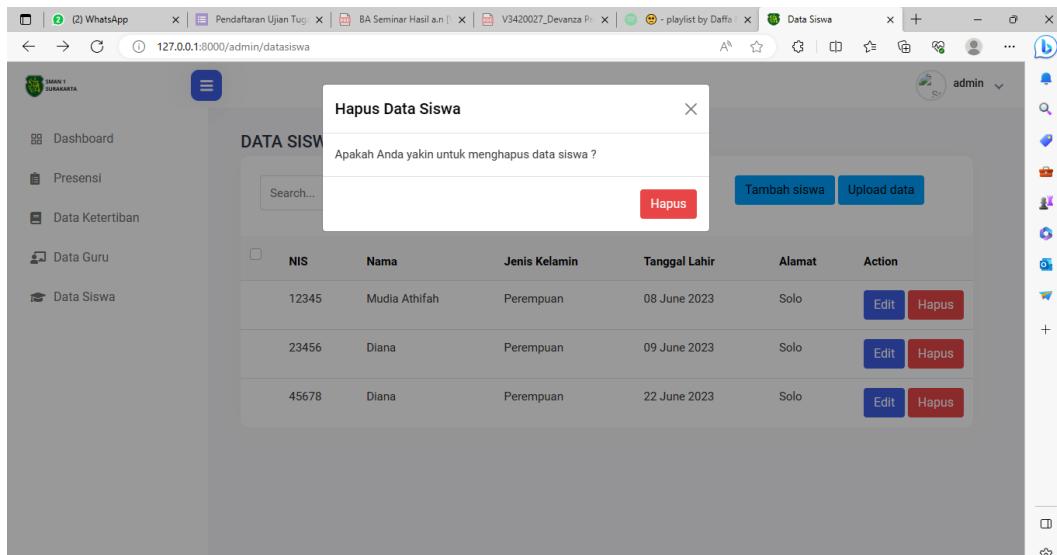
    $siswas = Siswa::find($id);
    $siswas->update([
        'id' => request('id'),
        'nama' => request('nama'),
        'jenis_kelamin' => request('jenis_kelamin'),
        'tanggal_lahir' => request('tanggal_lahir'),
        'alamat' => request('alamat'),
    ]);
    return redirect('/admin/datasiswa');
}

```

Gambar 4.40 Gambar controller edit data siswa

4.1.17 Halaman Hapus Data Siswa

Pada halaman ini admin dapat menghapus data siswa yang telah ditambahkan sebelumnya.



Gambar 4.41 Gambar tampilan halaman hapus data siswa

Source Code Halaman Hapus Data Siswa

```
</td>
</tr>
@endforeach
</tbody>
</table>

</div>
<!-- Modal -->
@foreach ($siswas as $siswa)
    <div class="modal fade" id="hapus{{ $siswa->id }}"
        data-bs-backdrop="static" data-bs-keyboard="false" tabindex="-1"
        aria-labelledby="staticBackdropLabel" aria-hidden="true">
        <div class="modal-dialog">
            <div class="modal-content">
                <div class="modal-header">
                    <h1 class="modal-title fs-5" id="staticBackdropLabel">
                        Hapus
                        Data Siswa</h1>
                    <button type="button" class="btn-close"
                        data-bs-dismiss="modal" aria-label="Close"></button>
                </div>
                <div class="modal-body">
                    Apakah Anda yakin untuk menghapus data siswa ?
                </div>
                <div class="modal-footer br">
                    <a href="{{ url('/admin/hapusdatasiswa/' . $siswa->id) }}"
                        class="btn btn-danger">Hapus</a>
                </div>
            </div>
        </div>
    </div>
@endforeach
```

Gambar 4.42 Gambar source code halaman hapus data

```
public function destroy($id)
{
    Siswa::destroy($id);
    return back();
}
```

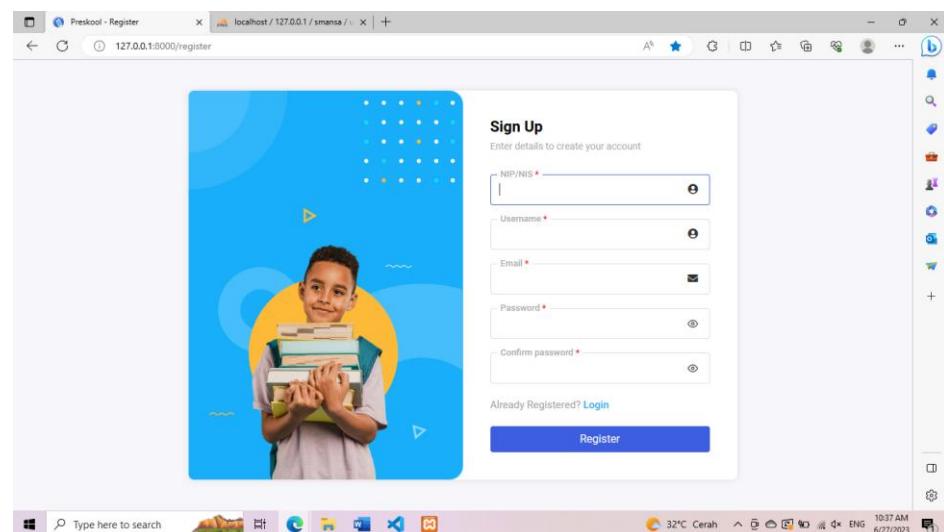
Gambar 4.43 gambar source code controller halaman hapus data

4.2 Implementasi Halaman Siswa

Pada halaman siswa ini akan digunakan siswa untuk melakukan presensi dan melihat data ketertiban masing masing siswa tersebut.

4.2.1 Halaman Register

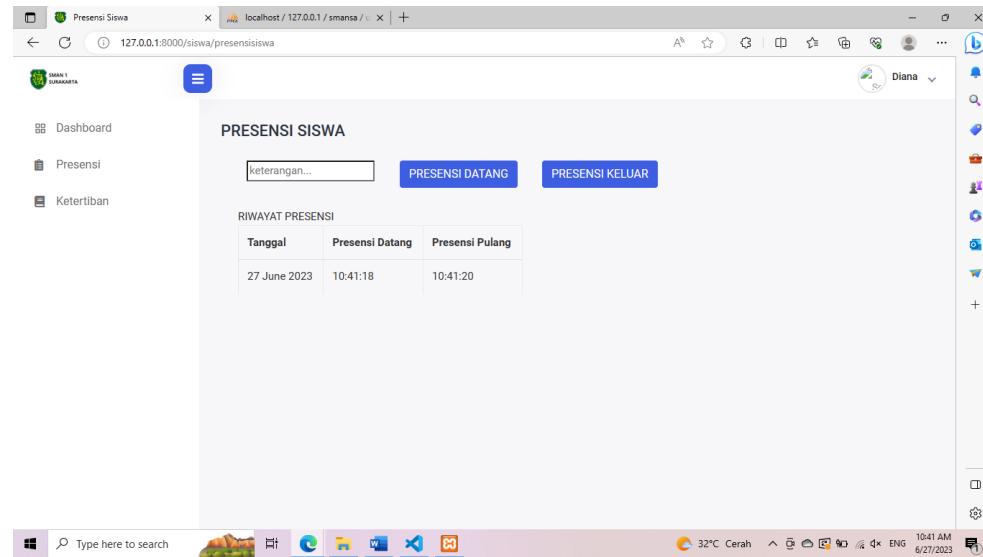
Pada halaman ini siswa dapat melakukan registrasi dengan memasukkan NIS, nama, email, dan password.



Gambar 4.44 Gambar Tampilan Halaman Register

4.2.2 Halaman Presensi

Pada halaman ini siswa melakukan presensi datang dan presensi pulang, kemudian Riwayat presensi akan tertampil pada tabel pada halaman presensi ini.



Gambar 4.45 Gambar Tampilan Halaman Presensi

Source Code Halaman Presensi

```
<div class="row">
  <div class="col-md-8 col-md-offset-2">
    <div class="panel panel-default">
      <div class="page-title-box">
        <h4 class="page-title">PRESENSI SISWA</h4>
      </div>
      <p></p>
      <div class="container-fluid">
        <div class="panel-body">
          <table class="table table-responsive">
            <form action="/siswa/presensi" method="POST">
              {{ csrf_field() }}
              <tr>
                <td>
                  <input type="text" class="form-control"
                         placeholder="keterangan..." name="keterangan">
                </td>
                <td>
                  <button type="submit" class="btn btn-flat btn-primary"
                         value="true" name="btnIn">
                      PRESENSI DATANG
                  </button>
                </td>
                <td>
                  <button type="submit" class="btn btn-flat btn-primary"
                         value="true" name="btnOut">
                      PRESENSI KELUAR
                  </button>
                </td>
              </tr>
            </form>
          </table>
        </div>
      </div>
    </div>
  </div>
```

Gambar 4.46 Gambar Source Code Halaman Presensi

```


<div class="col-md-8 col-md-offset-2">
        <div class="panel panel-default">
            <div class="panel-heading">RIWAYAT PRESENSI</div>
            <div class="panel-body">
                <table
                    class="table table-responsive table-bordered table-hover">
                    <thead>
                        <tr>
                            <th>Tanggal</th>
                            <th>Presensi Datang</th>
                            <th>Presensi Pulang</th>
                        </tr>
                    </thead>
                    <tbody>
                        @foreach ($data as $dta)
                            <tr>
                                <td><?php echo date('d F Y', strtotime($dta->date)); ?></td>
                                <td>{{ $dta->time_in }}</td>
                                <td>{{ $dta->time_out }}</td>
                            </tr>
                        @endforeach
                    </tbody>
                </table>
            </div>
        </div>
    </div>


```

Gambar 4.47 Gambar Source Code Halaman Presensi 2

```

public function presensi(Request $request)
{
    $users_id = Auth::user()->id;
    $dateNow = Carbon::now();
    $date = $dateNow->toDateString();
    $time = $dateNow->toTimeString();
    $keterangan = $request->keterangan;

    $presensi = new ModelsPresensi;
    $dataPresensi = ModelsPresensi::where(['users_id' => $users_id])->first() ?? null;
    $sudahPresensi = $dataPresensi->date ?? null;
    $sudahPresensiKeluar = $dataPresensi->time_out ?? null;

    //presensi masuk
    if (isset($request->btnIn)) {
        if($sudahPresensi && $date == $sudahPresensi ){
            return back();
        }

        $presensi->create([
            'users_id' => $users_id,
            'date' => $date,
            'time_in' => $time,
            'keterangan' => $keterangan,
        ]);

        return redirect()->back();
    } elseif (isset($request->btnOut)) {
        if($sudahPresensiKeluar){
            return back();
        }

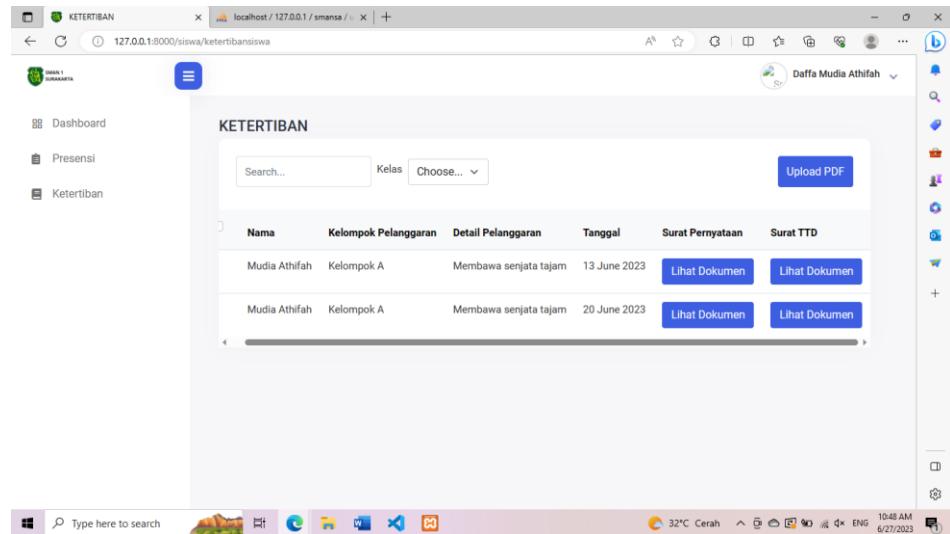
        $presensi->where(['date' => $date, 'users_id' => $users_id])->update([
            'time_out' => $time,
            'keterangan' => $keterangan,
        ]);
        return redirect()->back();
    }
    // return $request->all();
}

```

Gambar 4.48 Gambar Source Code Controller Halaman Presensi

4.2.3 Halaman Ketertiban Siswa

Pada halaman ini siswa dapat melihat data ketertiban dari siswa itu sendiri.



Gambar 4.49 Gambar Tampilan Halaman Ketertiban Siswa

Source Code Halaman Ketertiban

```
<div class="table-responsive">
    <table class="table table-centered mb-0">
        <thead class="table-light">
            <tr>
                <th style="width: 20px;">
                    <div class="form-check">
                        <input type="checkbox" class="form-check-input"
                            id="customCheck1">
                        <label class="form-check-label"
                            for="customCheck1">&ampnbsp</label>
                    </div>
                </th>
                <th>Nama</th>
                <th>Kelompok Pelanggaran</th>
                <th>Detail Pelanggaran</th>
                <th>Tanggal</th>
                <th>Surat Pernyataan</th>
                <th>Surat TTD</th>
            </tr>
        </thead>
        <tbody>
            @foreach ($pelanggarans as $pelanggaran)
            <tr>
                <td></td>
                <td>{{ $pelanggaran->siswa->nama }}</td>
                <td>{{ $pelanggaran->detail_pelanggaran->kelompok->nama_kelompok }}</td>
                <td>{{ $pelanggaran->detail_pelanggaran->nama_detail }}</td>
                <td>?php echo date('d F Y', strtotime($pelanggaran->tanggal)); ?></td>
                <td><button type="button" class="btn btn-primary"
                    data-bs-toggle="modal"
                    data-bs-target="#modalPDFPernyataan">
                    Lihat Dokumen
                </button></td>
                <td><button type="button" class="btn btn-primary"
                    data-bs-toggle="modal"
                    data-bs-target="#modalPDFPernyataanttd">
                    Lihat Dokumen
                </button></td>
            </tr>
        @endforeach
    </tbody>
```

Gambar 4.50 Gambar Source Code Halaman Ketertiban Siswa

```
public function indexSiswa()
{
    $pelanggarans = Pelanggaran::where('siswa_id', Auth::user()->nis)->get();
    return view('siswa.ketertiban_siswa', compact('pelanggarans'));
}
```

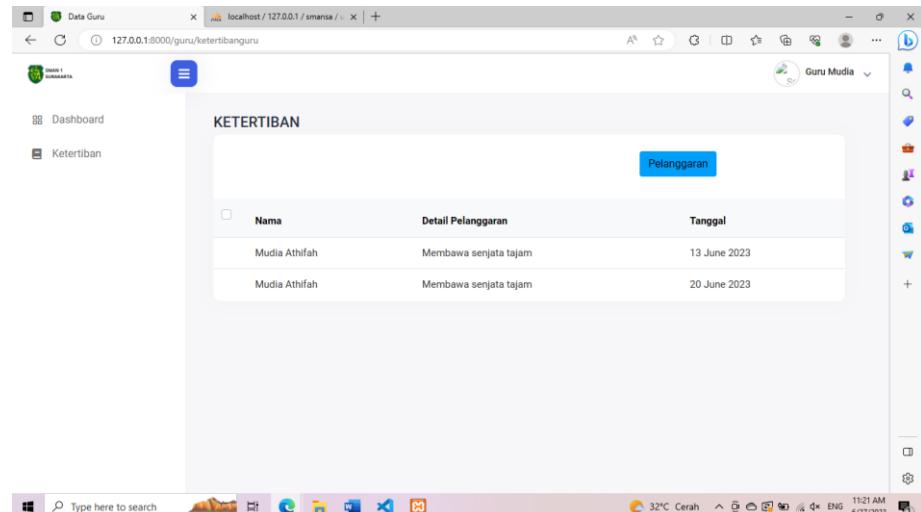
Gambar 4.51 Gambar Source Code Controller Ketertiban Siswa

4.3 Implementasi Halaman Guru

Pada halaman ini nanti guru dapat menambahkan pelanggaran dengan mengisi form yang ada

4.3.1 Halaman Ketertiban

Pada halaman ini akan menampilkan daftar data ketertiban siswa



Gambar 4.52 Gambar Tampilan Halaman Ketertiban pada Guru

Source Code Halaman Ketertiban

```
<div class="table-responsive">
    <table class="table table-centered mb-0">
        <thead class="table-light">
            <tr>
                <th style="width: 20px;">
                    <div class="form-check">
                        <input type="checkbox" class="form-check-input"
                               id="customCheck1">
                        <label class="form-check-label"
                               for="customCheck1">&ampnbsp</label>
                    </div>
                </th>
                <th>Nama</th>
                {{-- <th>Kelompok Pelanggaran</th> --}}
                <th>Detail Pelanggaran</th>
                <th>Tanggal</th>
            </tr>
        </thead>
        <tbody>
            @foreach ($pelanggarans as $pelanggaran)
            <tr>
                <td></td>
                <td>{{ $pelanggaran->siswa->nama }}</td>
                <td>{{ $pelanggaran->detail_pelanggaran->nama_detail }}</td>
                {{-- <td>{{ $pelanggaran->tanggal }}</td> --}}
                <td>?>{{php echo date('d F Y', strtotime($pelanggaran->tanggal)); ?>}</td>
            </tr>
        @endforeach
    </tbody>
</table>
```

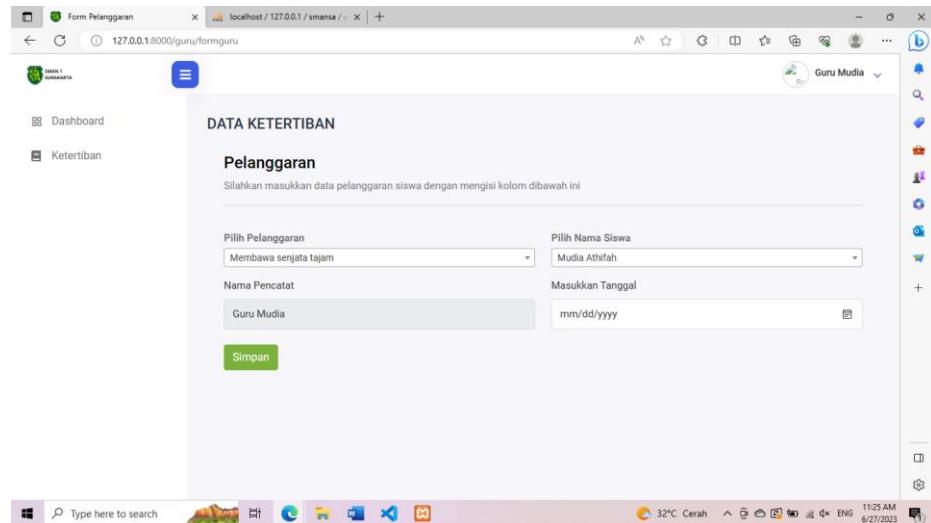
Gambar 4.53 Gambar Source Code Halaman Ketertiban pada Guru

```
public function index()
{
    $pelanggarans = Pelanggaran::all();
    return view('guru.ketertiban_guru', compact('pelanggarans'));
}
```

Gambar 4.54 Gambar Source Code Controller Halaman Ketertiban pada Guru

4.3.2 Halaman Form Tambah Pelanggaran

Pada halaman ini guru akan mengisi form untuk menambahkan pelanggaran siswa.



Gambar 4.55 Gambar Tampilan Halaman Tambah Pelanggaran pada Guru
Source Code Halaman Form Tambah Pelanggaran

```
<form action="/guru/simpanguru" method="POST">
    @csrf
    <div class="tab-content">
        <div class="tab-pane show active" id="select2-preview">
            <div class="row">
                <div class="col-lg-6">
                    <p class="mb-1 fw-bold text-muted">Pilih Pelanggaran</p>
                    <select name="detail_pelanggaran_id" class="form-control select2"
                           data-toggle="select2">
                        @foreach ($kelompoks as $kelompok)
                            <optgroup label="{{ $kelompok->nama_kelompok }}>
                                @foreach ($kelompok->detail_pelanggaran as $detail)
                                    <option value="{{ $detail->id }}>
                                        {{ $detail->nama_detail }}</option>
                                @endforeach
                            </optgroup>
                        @endforeach
                    </select>
                </div> <!-- end col -->
                <div class="col-lg-6">
                    <p class="mb-1 fw-bold text-muted">Pilih Nama Siswa</p>
                    <select name="siswa_id" class="form-control select2"
                           data-toggle="select2">
                        @foreach ($siswas as $siswa)
                            <option value="{{ $siswa->id }}>
                                {{ $siswa->nama }}</option>
                        @endforeach
                    </select>
                </div>
            </div></div>
            <div class="col-lg-6">
                <label class="form-label">Nama Pencatat</label>
                <input value="{{ Auth::user()->name }}" name="nama_pencatat" class="form-control"
                       data-provide="typeahead" id="the-basics" readonly>
            </div>
            <!-- Single Date Picker -->
            <div class="col-lg-6">
                <label class="form-label">Masukkan Tanggal</label>
                <input type="date" name="tanggal" class="form-control date"
                       id="started_at" value="{{ old('started_at') }}>
            </div>
        
```

Gambar 4.56 Gambar Source Code Halaman Tambah Pelanggaran Guru

```
class FormguruController extends Controller
{
    public function index ()
    {
        $pelanggarans = Pelanggaran::all();
        $siswas = Siswa::all();
        $kelompoks = Kelompok::get();
        return view('guru.form_guru', compact('kelompoks', 'pelanggarans', 'siswas'));
    }
}
```

Gambar 4.57 Gambar Source Code Controller Halaman Tambah
Pelanggaran Guru

BAB V

PENGUJIAN, HASIL, DAN KESIMPULAN

5.1 Metode Pengujian

Pada pengujian sistem ini penulis menggunakan metode blackbox. Menurut Thabroni blackbox testing atau pengujian kotak hitam menurapkansatu metode pengujian terhadap perangkat lunak. Metode pengujian sendiri adalah cara atau teknik untuk menguji perangkat lunak, mempunyai mekanisme untuk menentukan data uji yang dapat menguji perangkat lunak secara lengkap dan mempunyai kemungkinan tinggi untuk menemukan kesalahan.

5.2 Lingkungan Pengujian

Dalam pengujian ini menggunakan komponen pendukung perangkat keras dan lunak dengan spesifikasi sebagai berikut:

- a. Perangkat keras yang digunakan

Tabel 5.1 Tabel Perangkat Keras

Perangkat	Laptop ASUS X441UB
Processor Type	Intel® Core™ i5-8250U
Processor	Intel® Core™ i5-8250U CPU @ 1.60GHz 1.80 GHz
Memory	8,00 GB RAM
Graphics	Intel® UHD Graphics

- b. Perangkat lunak yang dibutuhkan

Tabel 5.2 Tabel Perangkat Lunak

Sistem Operasi	Windows 10
Web Server	XAMPP
Browser	Microsoft Edge

Bahasa Pemrograman	PHP
Database	PHP My Admin
Framework	Laravel

5.3 Prosedur Pengujian

Tabel 5.3 tabel prosedur pengujian

ID	Object	Butir Uji	Butir Uji	Jadwal
UJI001	Melakukan login sistem	Melakukan pengisian data menggunakan data user yang sudah ada	Blackbox	27 Juni 2023
UJI002	Melakukan register oleh siswa	Melakukan pengisian data dengan menginputkan data siswa untuk mendaftar akun	Blackbox	27 Juni 2023
UJI003	Melakukan presensi	Penambahan data presensi pada halaman presensi	Blackbox	27 Juni 2023
UJI004	Melihat data presensi dari halaman siswa	Menampilkan data presensi dari siswa itu sendiri	Blackbox	27 Juni 2023
UJI005	Mengelola data ketertiban siswa	Penambahan data pelanggaran siswa pada halaman ketertiban Menampilkan data presensi dari semua siswa yang telah melakukan presensi Mengubah data ketertiban yang telah ada dengan data baru Menghapus data ketertiban siswa	Blackbox	27 Juni 2023

		<p>yang telah ada pada table</p> <p>Penambahan dokumen pada halaman tambah dokumen menggunakan dokumen pdf surat pernyataan</p>		
UJI006	Melihat data presensi dari oleh admin	Menampilkan data presensi dari semua siswa yang telah melakukan presensi	Blackbox	27 Juni 2023
UJI007	Melihat data ketertiban siswa oleh siswa	Menampilkan data pelanggaran dari siswa itu sendiri pada halaman ketertiban	Blackbox	27 Juni 2023
UJI008	Mengunggah dokumen surat pernyataan dari siswa	Penambahan dokumen pada halaman tambah dokumen menggunakan dokumen pdf surat pernyataan	Blackbox	27 Juni 2023
UJI009	Input data ketertiban oleh guru	Penambahan data pelanggaran pada halaman ketertiban siswa dengan data baru oleh guru	Blackbox	27 Juni 2023

UJI010	Mengelola data siswa	Penambahan data siswa pada halaman form tambah siswa menggunakan data baru Mengubah data siswa yang telah ada menggunakan data baru Menghapus data siswa yang telah ada pada table Menampilkan data siswa	Blackbox	27 Juni 2023
UJI011	Mengelola data guru	Penambahan data guru pada halaman form tambah guru menggunakan data baru Mengubah data guru yang telah ada menggunakan data baru Menghapus data guru yang telah ada pada table Menampilkan data guru	Blackbox	27 Juni 2023

5.4 Hasil Uji dan Kesimpulan

Tabel 5.4 tabel uji UJI001

ID	UJI001
Nama Butir Uji	Melakukan pengisian data menggunakan data user yang sudah ada
Deskripsi	Memeriksa apakah data akun valid atau ada pada database
Kondisi Awal	Pengguna berada di halaman login
Tanggal Pengujian	27 Juni 2023
Penguji	Daffa Mudia
Skenario	
1. Isi form yang tersedia dengan data: - Username : admin@gmail.com - Password : password	
2. Klik button “Login”	Hasil
Yang diharapkan	Hasil Pengamatan
Data akun tervalidasi dan lanjut ke halaman dashboard	Field uasernama dan password terisi Button bisa diklik
Kesimpulan	
Tes ini berjalan dengan baik apabila data Account sudah sesuai dengan scenario dan valid.	

Tabel 5.4 tabel uji UJI002

ID	UJI002
Nama Butir Uji	Melakukan pengisian data dengan menginputkan data siswa untuk mendaftar akun
Deskripsi	Memeriksa apakah data bisa tersimpan pada sistem ataupun database
Kondisi Awal	Pengguna berada di halaman register
Tanggal Pengujian	27 Juni 2023
Penguji	Daffa Mudia
Skenario	
3. Isi form yang tersedia dengan data: - NIS : 23456 - Username : daffa mudia - Email : daffamudiaa@gmail.com - Password : 12345678	
4. Klik button “register”	
Hasil	
Yang diharapkan	Hasil Pengamatan
Data akun tervalidasi dan lanjut ke halaman login	Field email memiliki sifat unique sehingga tidak bisa duplikasi data Button bisa diklik
Kesimpulan	
Tes ini berjalan dengan baik karena data sudah masuk sesuai dengan skenario	

Tabel 5.5 Tabel uji UJI003

ID	UJI003
Nama Butir Uji	Penambahan data presensi pada halaman presensi
Deskripsi	Memeriksa apakah data bisa tersimpan pada sistem ataupun database
Kondisi Awal	Pengguna berada di halaman presensi
Tanggal Pengujian	27 Juni 2023
Penguji	Daffa Mudia
Skenario	
<ol style="list-style-type: none"> 1. Klik button “presensi datang” disaat masuk sekolah 2. Klik button “presensi pulang” disaat pulang sekolah 	
Hasil	
Yang diharapkan	Hasil Pengamatan
Berhasil melakukan presensi dan Kembali ke halaman presensi	Button bisa diklik
Kesimpulan	
Tes ini berjalan dengan baik karena sesuai dengan skenario	

Tabel 5.6 tabel uji UJI004

ID	UJI004
Nama Butir Uji	Menampilkan data presensi dari siswa itu sendiri
Deskripsi	Memeriksa apakah sistem dapat menampilkan data pada halaman presensi
Kondisi Awal	Pengguna berada di halaman presensi
Tanggal Pengujian	27 Juni 2023
Penguji	Daffa Mudia
Skenario	
1. Setelah klik button presensi maka akan tertampil Riwayat presensi	
Hasil	
Yang diharapkan	Hasil Pengamatan
Data presensi tervalidasi dan lanjut ke halaman presensi untuk melihat Riwayat presensi	Data yang ditampilkan sesuai dengan yg dikirim dan di database
Kesimpulan	
Tes ini berjalan dengan baik karena sesuai dengan skenario	

Tabel 5.7 tabel uji UJI005

ID	UJI005
Nama Butir Uji	<p>Penambahan data pelanggaran siswa pada halaman ketertiban</p> <p>Menampilkan data presensi dari semua siswa yang telah melakukan presensi</p> <p>Mengubah data ketertiban yang telah ada dengan data baru</p> <p>Menghapus data ketertiban siswa yang telah ada pada table</p> <p>Penambahan dokumen pada halaman tambah dokumen menggunakan dokumen pdf surat pernyataan</p>
Deskripsi	Memeriksa apakah sistem dapat menambahkan, menampilkan, mengedit, dan menghapus data ketertiban siswa
Kondisi Awal	Pengguna berada di halaman ketertiban
Tanggal Pengujian	27 Juni 2023
Penguji	Daffa Mudia
Skenario	
<ul style="list-style-type: none"> a. Menambahkan data pelanggaran siswa <ul style="list-style-type: none"> 1. Klik button tambah pelanggaran 2. Mengisi form pelanggaran sesuai dengan data yang ada 3. Klik submit b. Menampilkan data ketertiban siswa <ul style="list-style-type: none"> 1. Klik menu ketertiban c. Melakukan pengecekan data ketertiban kembali <ul style="list-style-type: none"> 1. Klik button edit 2. Pilih data yang akan direvisi 3. Upload dokumen surat pernyataan 4. Klik submit d. Menghapus data ketertiban <ul style="list-style-type: none"> 1. Klik button hapus 	
Hasil	

Yang diharapkan	Hasil Pengamatan
Sistem mampu menampilkan data ketertiban	Sistem dapat menambahkan data ketertiban siswa
Sistem mampu merubah data ketertiban	Sistem dapat menampilkan data ketertiban siswa Sistem berhasil merubah data ketertiban siswa
Sistem dapat menghapus data ketertiban	Sistem berhasil menghapus data ketertiban siswa
Kesimpulan	
Tes ini berjalan dengan baik karena sesuai dengan skenario	

Tabel 5.8 tabel uji UJI006

ID	UJI006
Nama Butir Uji	Menampilkan data presensi dari semua siswa yang telah melakukan presensi
Deskripsi	Memeriksa apakah sistem dapat menampilkan data pada halaman presensi
Kondisi Awal	Pengguna berada di halaman presensi
Tanggal Pengujian	27 Juni 2023
Penguji	Daffa Mudia

Skenario	
1. Setelah klik button presensi maka akan tertampil Riwayat presensi dari seluruh siswa	
Hasil	
Yang diharapkan	Hasil Pengamatan
Data presensi tervalidasi dan lanjut ke halaman presensi untuk melihat Riwayat presensi	Data yang ditampilkan sesuai dengan yg dikirim dan di database
Kesimpulan	
Tes ini berjalan dengan baik karena sesuai dengan skenario	

Tabel 5.9 tabel uji UJI007

ID	UJI007
Nama Butir Uji	Menampilkan data pelanggaran dari siswa itu sendiri pada halaman ketertiban
Deskripsi	Memeriksa apakah sistem dapat menampilkan data pada halaman ketertiban
Kondisi Awal	Pengguna berada di halaman ketertiban
Tanggal Pengujian	27 Juni 2023

Penguji	Daffa Mudia
Skenario	
Setelah klik sidebar ketertiban maka akan tertampil data pelanggaran di halaman ketertiban	
Hasil	
Yang diharapkan	Hasil Pengamatan
Data pelanggaran tervalidasi dan lanjut ke halaman ketertiban untuk melihat data ketertiban	Data yang ditampilkan sesuai dengan yg dikirim dan di database
Kesimpulan	
Tes ini berjalan dengan baik karena sesuai dengan skenario	

Tabel 5.10 tabel uji UJI008

ID	UJI008
Nama Butir Uji	Penambahan dokumen pada halaman tambah dokumen menggunakan dokumen pdf surat pernyataan
Deskripsi	Memeriksa apakah sistem dapat menambahkan dokumen surat pernyataan
Kondisi Awal	Pengguna berada di halaman ketertiban
Tanggal Pengujian	27 Juni 2023

Penguji	Daffa Mudia
Skenario	
1. Klik button upload dan modal akan tertampil 2. Masukkan dokumen surat pertanyaan 3. Submit	
Hasil	
Yang diharapkan	Hasil Pengamatan
Dokumen tervalidasi dan lanjut ke halaman ketertiban untuk melihat surat pernyataan	Data yang akan tersimpan sesuai dengan yang didatabase
Kesimpulan	
Tes ini berjalan dengan baik karena sesuai dengan skenario	

Tabel 5.11 tabel uji UJI009

ID	UJ009
Nama Butir Uji	Penambahan data pelanggaran pada halaman ketertiban siswa dengan data baru oleh guru
Deskripsi	Memeriksa apakah sistem dapat menambahkan data ketertiban siswa
Kondisi Awal	Pengguna berada di halaman dashboard
Tanggal Pengujian	27 Juni 2023

Penguji	Daffa Mudia
Skenario	
1. Klik sidebar ketertiban 2. Klik button tambah pelanggaran 3. Mengisi form pelanggaran dengan data pelanggaran yang ada 4. Submit	
Hasil	
Yang diharapkan	Hasil Pengamatan
Data tervalidasi dan lanjut ke halaman ketertiban	Data yang akan tersimpan sesuai dengan yang didatabase
Kesimpulan	
Tes ini berjalan dengan baik karena sesuai dengan skenario	

Tabel 5.12 tabel uji UJI010

ID	UJI010
Nama Butir Uji	Penambahan data siswa pada halaman form tambah siswa menggunakan data baru Mengubah data siswa yang telah ada menggunakan data baru Menghapus data siswa yang telah ada pada table Menampilkan data siswa
Deskripsi	Memeriksa apakah sistem dapat menambahkan, menampilkan, mengedit, dan menghapus data siswa
Kondisi Awal	Pengguna berada di halaman siswa
Tanggal Pengujian	27 Juni 2023

Penguji	Daffa Mudia
Skenario	
<ul style="list-style-type: none"> a. Menambahkan data siswa <ul style="list-style-type: none"> 1. Klik button tambah siswa 2. Mengisi form sesuai dengan data siswa yang ada 3. Klik submit b. Menampilkan data siswa <ul style="list-style-type: none"> 1. Klik menu ketertiban c. Melakukan pengecekan data ketertiban kembali <ul style="list-style-type: none"> 1. Klik button edit 2. Pilih data yang akan direvisi 3. Klik submit d. Menghapus data ketertiban <ul style="list-style-type: none"> 1. Klik button hapus 	
Hasil	
Yang diharapkan	Hasil Pengamatan
Sistem mampu menambahkan data siswa	Sistem berhasil menambahkan data siswa
Sistem mampu menampilkan data siswa	Sistem berhasil menampilkan data ketertiban siswa
Sistem mampu merubah data siswa	Sistem berhasil merubah data ketertiban siswa
Sistem dapat menghapus data siswa	Sistem berhasil menghapus data ketertiban siswa
Kesimpulan	
Tes ini berjalan dengan baik karena sesuai dengan skenario	

Tabel 5.13 tabel uji UJI011

ID	UJI011
Nama Butir Uji	Penambahan data siswa pada halaman form tambah guru menggunakan data baru Mengubah data guru yang telah ada menggunakan data baru Menghapus data guru yang telah ada pada table Menampilkan data guru
Deskripsi	Memeriksa apakah sistem dapat menambahkan, menampilkan, mengedit, dan menghapus data guru
Kondisi Awal	Pengguna berada di halaman data guru
Tanggal Pengujian	27 Juni 2023
Penguji	Daffa Mudia
Skenario	
a. Menambahkan data guru 1. Klik button tambah guru b. Mengisi form sesuai dengan data guru yang ada 1. Klik submit 2. Menampilkan data guru 3. Klik menu ketertiban c. Melakukan pengecekan data ketertiban kembali 1. Klik button edit 2. Pilih data yang akan direvisi 3. Klik submit d. Menghapus data ketertiban 1. Klik button hapus	
Hasil	
Yang diharapkan	Hasil Pengamatan

Sistem mampu menambahkan data guru	Sistem berhasil menambahkan data guru Sistem berhasil menampilkan data guru
Sistem mampu menampilkan data guru	Sistem berhasil merubah data guru Sistem berhasil menghapus data guru
Sistem mampu merubah data guru	
Sistem dapat menghapus data guru	
Kesimpulan	
Tes ini berjalan dengan baik karena sesuai dengan skenario	

DAFTAR PUSTAKA

- Adani, M. R., 2020. *Mengenal MySQL: Pengertian, Fungsi, serta Kelebihannya*. [Online]
Available at: <https://www.sekawanmedia.co.id/blog/pengertian-mysql/>
- Adani, M. R., 2021. *Mengulas Penggunaan HTML dan Cara Kerjanya pada Web Development*. [Online]
Available at: <https://www.sekawanmedia.co.id/blog/pengertian-html/>
- Adani, M. R., 2021. *Use Case Diagram: Pengertian, Fungsi, dan Contoh penerapan*.
[Online]
Available at: <https://www.sekawanmedia.co.id/blog/use-case-diagram/#:~:text=Use%20case%20diagram%20adalah%20teknik%20yang%20biasa%20digunakan,hubungan%20antara%20pengguna%20dengan%20sistem%20yang%20telah%20dirancang.>
- Amalia, A. F., 2021. *Mockup: Pengertian, Fungsi, 7 Contoh Tools, dan Cara Membuat*.
[Online]
Available at: <https://sekawanstudio.com/blog/apa-itu-mockup/>
- Choiri, E. O., 2020. *Pengertian PHP dan Fungsinya Dalam Pemrograman Web*. [Online]
Available at: <https://qwords.com/blog/pengertian-php/>
- Kurniawan, A., 2016. *TA : Rancang Bangun Sistem Informasi Penilaian dan Presensi Siswa Berbasis Web pada SMA Negeri 12 Surabaya*. [Online]
Available at: <https://repository.dinamika.ac.id/id/eprint/2112/>
- Maksum, M. A., 2022. *Apa itu Laravel? Pengertian, Fitur dan Kelebihannya*. [Online]
Available at: <https://www.dewaweb.com/blog/apa-itu-laravel/>
- Maksum, M. A., 2022. *Pengertian XAMPP, Fungsi, dan Cara Menggunakannya*. [Online]
Available at: <https://www.dewaweb.com/blog/apa-itu-xampp/#:~:text=XAMPP%20adalah%20salah%20satu%20aplikasi%20yang%20berfungsi%20sebagai,internet.%20XAMPP%20juga%20digunakan%20dalam%20proses%20development%20aplikasi.>
- Myrna Dwi Rahmatya, M. F., 2020. Pengembangan Sistem Informasi Akademik Berbasis Web pada SMA Pasundan 2 Bandung. *Jurnal Manajemen Informatika (JAMIKA)*.
- Novitasari, C., 2023. *Pengertian Activity Diagram dan Simbol-Simbolnya - Pelajarindo.com*. [Online]
Available at: <https://pelajarindo.com/pengertian-activity-diagram-simbol/>

Putra, 2022. *PENGERTIAN WEBSITE: Fungsi, Sejarah, Kegunaan, Jenis jenis & Contoh Web*. [Online]
Available at: <https://salamadian.com/pengertian-website/>

Ridwanti, A. E., 2023. *Pengertian Sistem Informasi - Sejarah, Fungsi, Tujuan, Kelebihan, Kelemahan, dll.* [Online]
Available at: <https://dianisa.com/pengertian-sistem-informasi/>

Sulthon, A., 2022. *Pengertian Entity Relationship Diagram (ERD): Simbol, Entitas, Atribut.* [Online]
Available at: <https://www.domainesia.com/berita/pengertian-erd-adalah/>