

LAPORAN PERANCANGAN SISTEM INFORMASI AGENDA MURID GURU KEPADA MURID

Disusun oleh:

ELLY GOU

NIS. 22161017

Jurusan Rekayasa Perangkat Lunak



SEKOLAH MENENGAH KEJURUAN (SMK)

SEKOLAH PERMATA HARAPAN

BATAM

2024

PERANCANGAN

1. Context Diagram

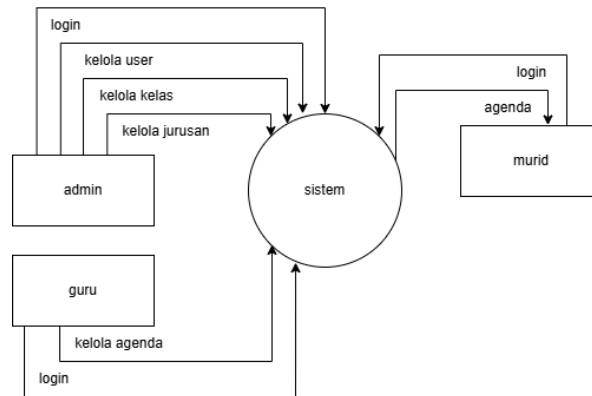


Diagram *context* di atas menggambarkan hubungan antara aktor dan sistem. Aktor-aktor yang terlibat meliputi admin, guru, dan murid. Setiap aktor harus melalui proses *login* untuk dapat mengakses sistem. Setelah *login*, admin memiliki akses ke fitur *kelola user*, *kelola kelas*, dan *kelola jurusan*, yang memungkinkan mereka untuk mengelola pengguna, kelas serta jurusan dalam sistem. Guru dapat mengakses fitur *kelola agenda*, di mana mereka dapat membuat agenda yang kemudian dapat diakses murid. Kemudian terdapat murid yang dapat mengakses agenda yang diberikan oleh guru setelah berhasil melakukan *login*.

2. DFD Level 1

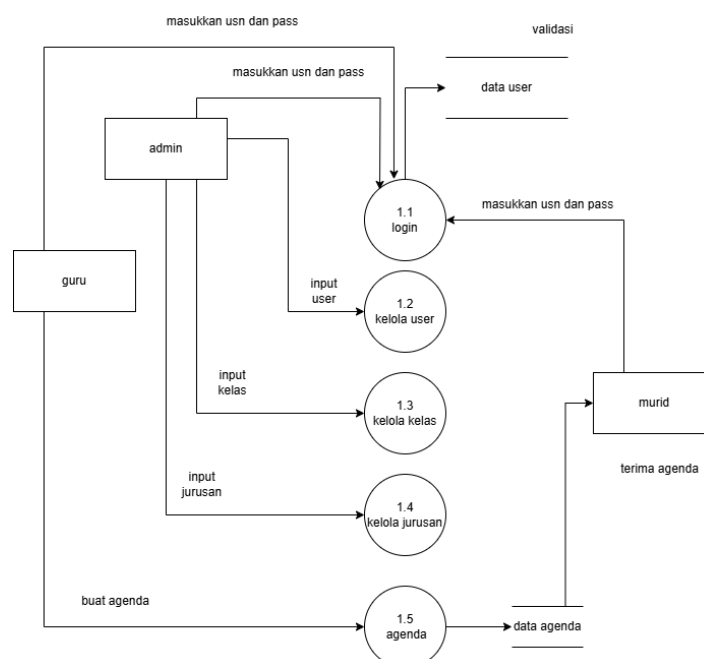


Diagram DFD level 1 di atas menggambarkan aliran data dalam sistem, terutama terkait *login*, pengelolaan *user*, pengelolaan kelas, pengelolaan jurusan, dan agenda. Proses dimulai ketika aktor-aktor seperti admin, guru, dan muird memasukkan *username* dan *password* mereka untuk *login*. Data *login* ini kemudian divalidasi oleh sistem. Setelah berhasil *login*, admin memiliki akses untuk mengelola pengguna, termasuk menambah, mengubah, atau menghapus data pengguna yang ada dalam sistem. Selain itu, admin juga dapat mengelola kelas dan jurusan, dengan kemampuan menambahkan, mengubah, atau menghapus data kelas dan jurusan. Di sisi lain, guru dapat membuat agenda yang kemudian dapat diakses oleh murid. Agenda yang dibuat akan disimpan dalam sistem sebagai data agenda dan diterima oleh muird.

3. DFD Level 2

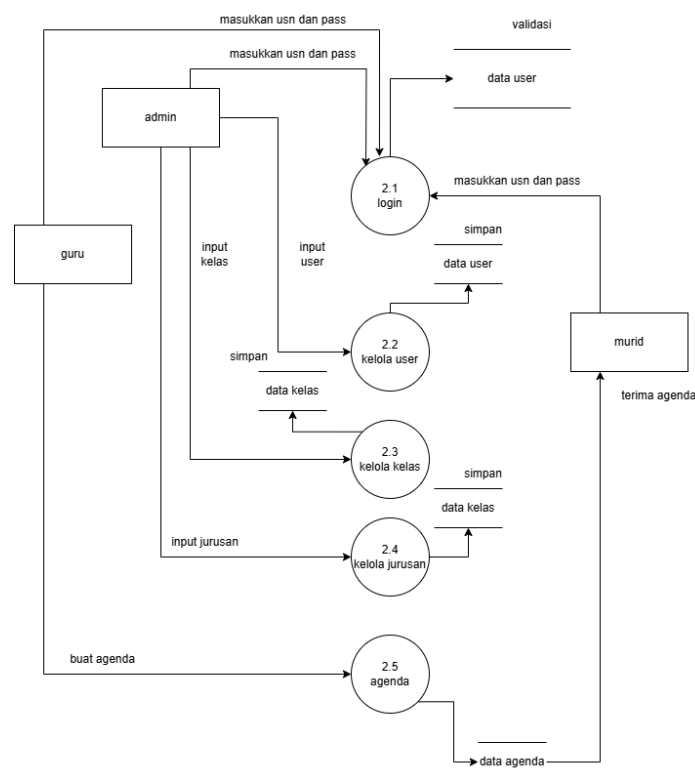


Diagram DFD level 2 ini menggambarkan alur pengelolaan data dalam sistem, khususnya terkait *login*, pengelolaan pengguna, kelas, jurusan, dan agenda. Proses dimulai dari pengguna seperti admin, guru, dan muird yang memasukkan *username* dan *password* untuk *login*. Data *login* ini kemudian divalidasi dengan mencocokkannya pada data *user* yang tersimpan dalam sistem. Jika valid, pengguna

akan mendapatkan akses ke sistem. Setelah *login*, admin bisa mengelola data pengguna, misalnya menambahkan, mengubah, atau memperbarui informasi pengguna. Data yang diinput ini disimpan kembali dalam sistem sebagai data pengguna. Selain itu, admin juga bisa mengelola data kelas dan jurusan, termasuk menambah kelas dan jurusan baru atau memperbarui informasi kelas dan jurusan yang sudah ada. Setelah diinput, data kelas dan jurusan ini juga akan disimpan di dalam sistem. Pada proses agenda, guru bisa membuat agenda yang nantinya dapat oleh murid. Agenda ini disimpan dalam sistem dan kemudian ditampilkan kepada murid.

4. Use Case Diagram

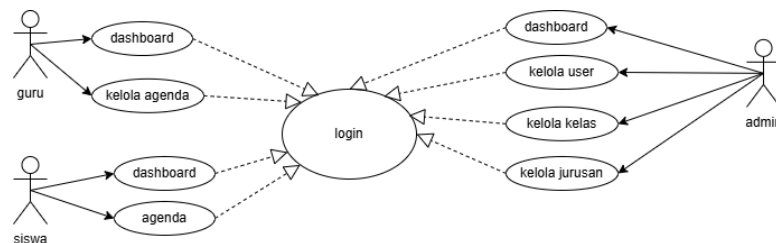
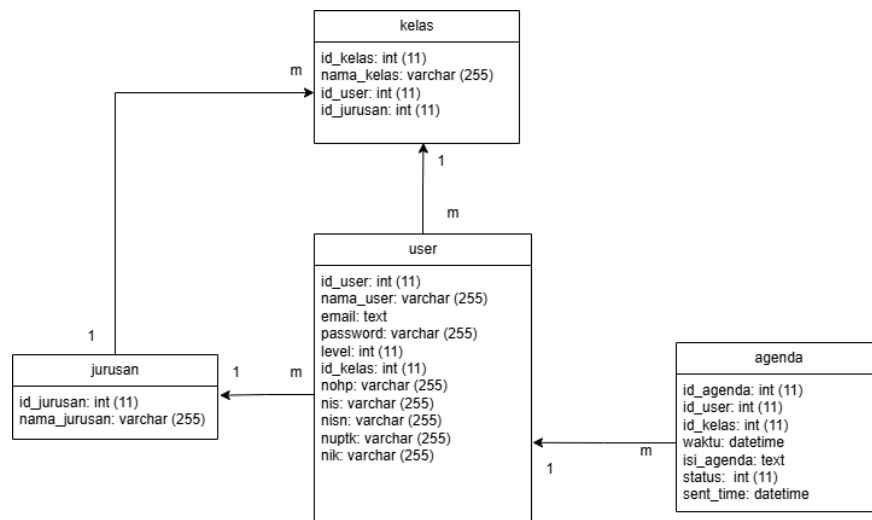


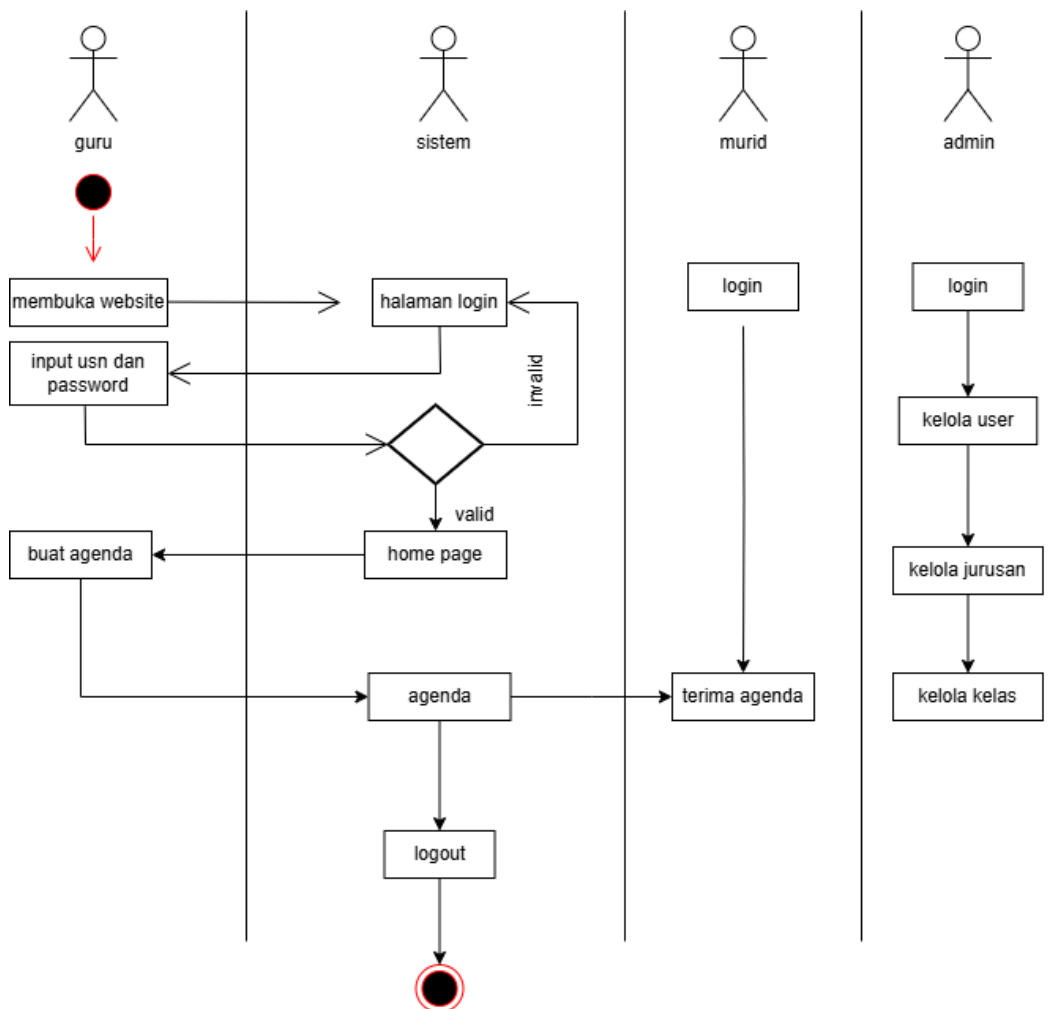
Diagram *use case* di atas menggambarkan interaksi antara beberapa aktor dengan sistem. Semua aktor, seperti admin, guru, dan murid harus melalui proses *login* untuk mengakses fitur-fitur yang tersedia. Setelah *login*, admin, guru, dan murid dapat mengakses *dashboard*. Aktor guru memiliki akses untuk mengelola agenda, termasuk membuat, mengedit, dan menghapus agenda. Setelah itu siswa dapat mengakses agenda tersebut. Selain itu, admin, memiliki akses untuk mengelola pengguna, jurusan, dan kelas, memungkinkan mereka untuk mengatur data pengguna, jurusan dan kelas.

5. Class Diagram



Class diagram di atas menggambarkan empat entitas utama yaitu kelas, jurusan, user, dan agenda, serta hubungan antar entitas tersebut. Entitas kelas memiliki atribut seperti `id_kelas`, dan `nama_kelas`, yang menunjukkan identitas dan deskripsi kelas. Setiap kelas memiliki hubungan *one to many* dengan user, artinya satu kelas bisa memiliki banyak user (siswa atau guru), tapi setiap user hanya dapat tergabung di satu kelas. Entitas jurusan memiliki atribut `id_jurusan` dan `nama_jurusan`. Setiap jurusan memiliki hubungan *one to many* dengan user dan kelas, karena satu jurusan dapat dimiliki oleh banyak user dan kelas. Pada entitas user, terdapat berbagai atribut seperti `id_user`, `nama_user`, `email`, `password`, dan atribut lainnya yang menjelaskan data pribadi user baik sebagai siswa maupun guru. Setiap user juga memiliki *foreign key* yang terhubung dengan entitas kelas, yaitu `id_kelas`, menunjukkan bahwa user tersebut tergabung dalam suatu kelas tertentu. Selain itu, entitas user juga memiliki hubungan *one to many* dengan entitas agenda, yang berarti satu user bisa membuat banyak agenda, namun satu agenda hanya dibuat oleh satu user. Di entitas agenda sendiri, terdapat atribut seperti `id_agenda`, `id_kelas`, `id_user`, `waktu`, `isi_agenda`, `status`, dan `sent_time`, yang menjelaskan detail dari agenda tersebut. Hubungan terakhir antara user dan agenda adalah *one to many*, di mana satu user dapat membuat banyak agenda, dan satu agenda hanya tertuju kepada salah satu user atau salah satu kelas tertentu.

6. Activity Diagram



Activity diagram di atas menggambarkan alur aktivitas sistem agendasekolah, dimulai dari proses *login* hingga pembuatan agenda. Pertama, pengguna seperti admin, guru, dan murid membuka *website* sistem dan diarahkan ke halaman *login*. Pada halaman *login* akan diminta untuk memasukkan *username* dan *password*. Jika data *login* yang dimasukkan valid, pengguna diarahkan ke halaman utama, sedangkan jika tidak valid, sistem akan meminta untuk mengulangi proses *login*. Setelah berhasil *login*, pengguna dapat membuat agenda. Agenda yang telah dibuat akan ditampilkan kepada murid, murid dapat mengakses agenda setelah berhasil melakukan *login*. Pada akhir proses, pengguna dapat melakukan *logout* untuk keluar dari sistem. Selain itu, terdapat peran admin yang memiliki akses untuk *login*

ke sistem dan mengelola data pengguna, kelas, dan jurusan melalui fitur kelola *user*, kelas, dan jurusan.

7. ERD

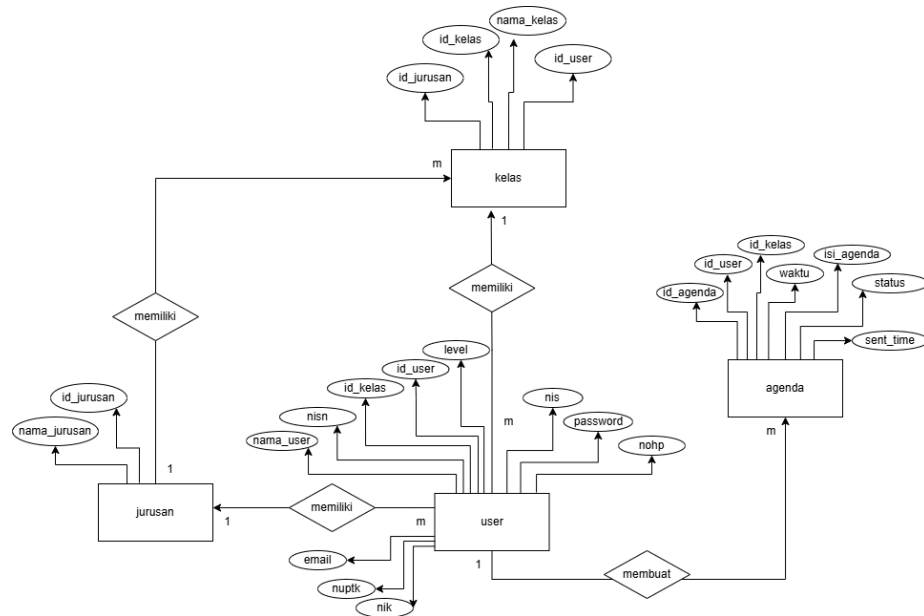


Diagram ERD di atas menggambarkan empat entitas utama, yaitu kelas, *user*, jurusan, dan agenda, serta relasi antar entitas tersebut. Entitas kelas memiliki hubungan *one to many* dengan *user*, di mana satu kelas dapat memiliki banyak *user*, namun setiap *user* hanya dapat terdaftar di satu kelas. Relasi ini ditandai dengan atribut seperti id_kelas di kedua entitas, yang menghubungkan mereka. Selain itu, entitas *user* juga memiliki relasi *one to many* dengan entitas agenda. Hal ini berarti seorang *user* dapat mengelola atau membuat beberapa agenda, namun setiap agenda dikelola oleh satu *user* tertentu, yang diidentifikasi melalui atribut id_user dan id_agenda. Entitas jurusan memiliki hubungan *one to many* dengan *user* dan kelas, di mana satu jurusan dapat dimiliki oleh banyak *user* dan kelas. Terakhir, ada relasi *many to one* antara agenda dan *user*, yang menunjukkan bahwa satu *user* dapat membuat banyak agenda, dan satu agenda hanya tertuju kepada salah satu *user* atau salah satu kelas tertentu.

8. Sequence Diagram

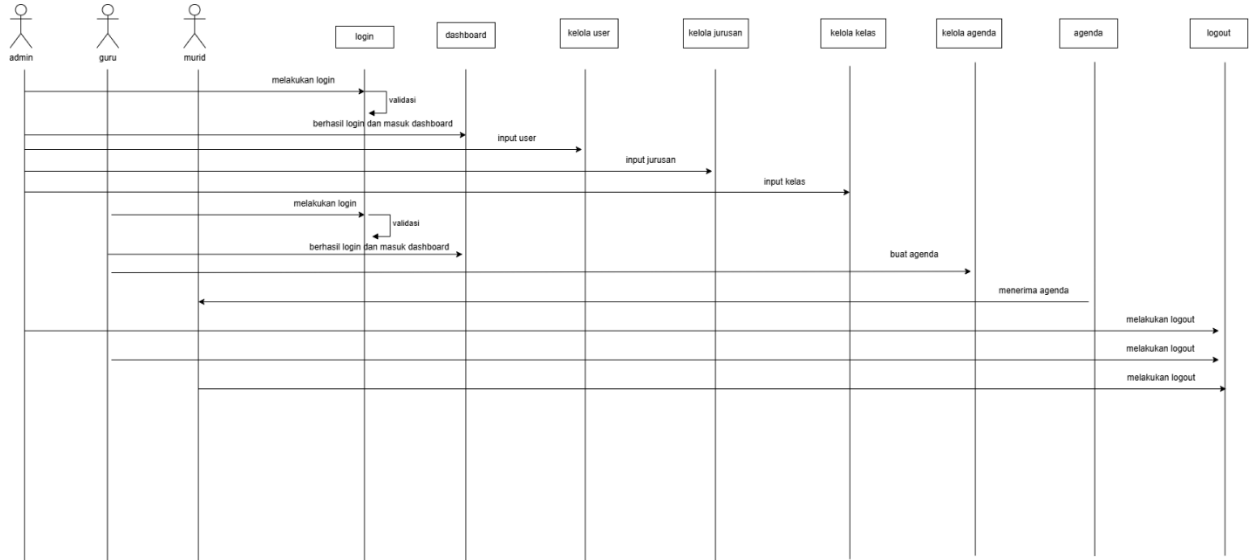
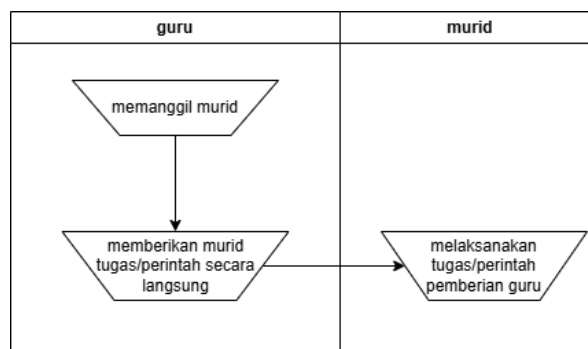


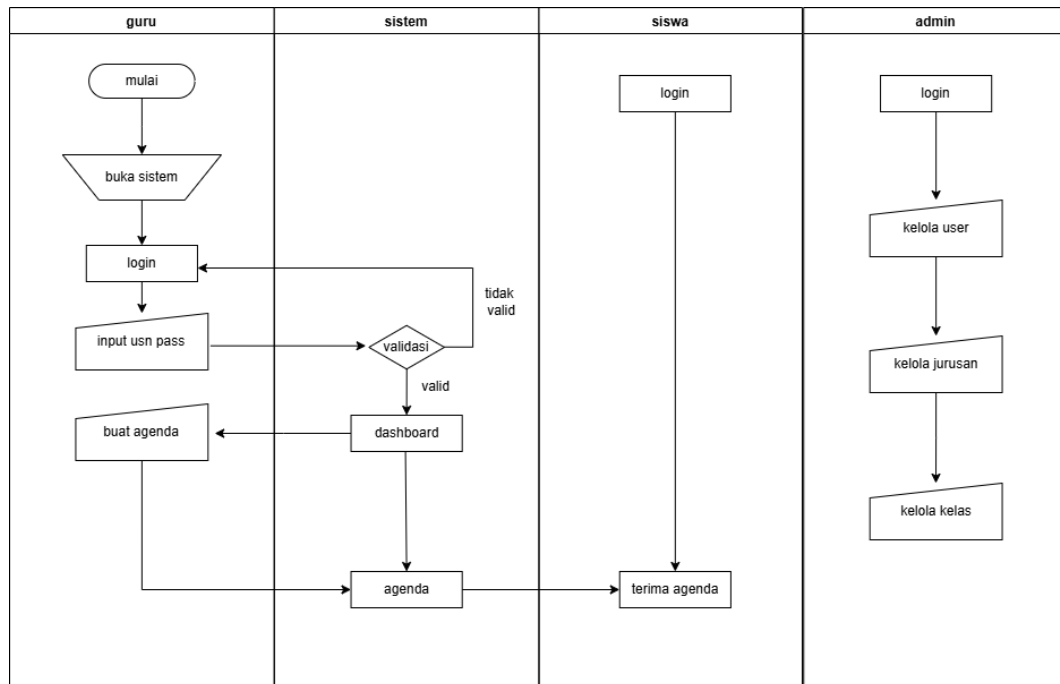
Diagram *sequence* di atas menggambarkan alur interaksi antara berbagai aktor dalam sistem, yaitu admin, guru, dan murid. Proses dimulai ketika admin, guru, dan murid melakukan *login* ke sistem dengan memasukkan kredensial mereka. Setelah itu, sistem akan memvalidasi kredensial tersebut. Jika valid, mereka akan diarahkan ke *dashboard*. Admin kemudian dapat mengakses fitur kelola *user*, kelola jurusan dan kelola kelas untuk menginput atau memperbarui data yang diperlukan. Setelah *login*, guru dapat membuat agenda menggunakan fitur kelola agenda. agenda yang dibuat akan ditampilkan oleh murid. Setelah menyelesaikan tugas masing-masing, baik admin, kesiswaan, wali kelas, kepsek, wakepsek, maupun siswa, orang tua dapat melakukan *logout* untuk mengakhiri sesi mereka.

9. Flowmap Sistem Berjalan



Flowmap sistem berjalan di atas menggambarkan alur kerja secara langsung, dimulai dari guru yang akan memanggil murid yang akan diberikan perintah atau tugas, kemudian murid akan melaksanakan tugas atau perintah yang diberikan guru.

10. *Flowmap* Sistem Diusulkan



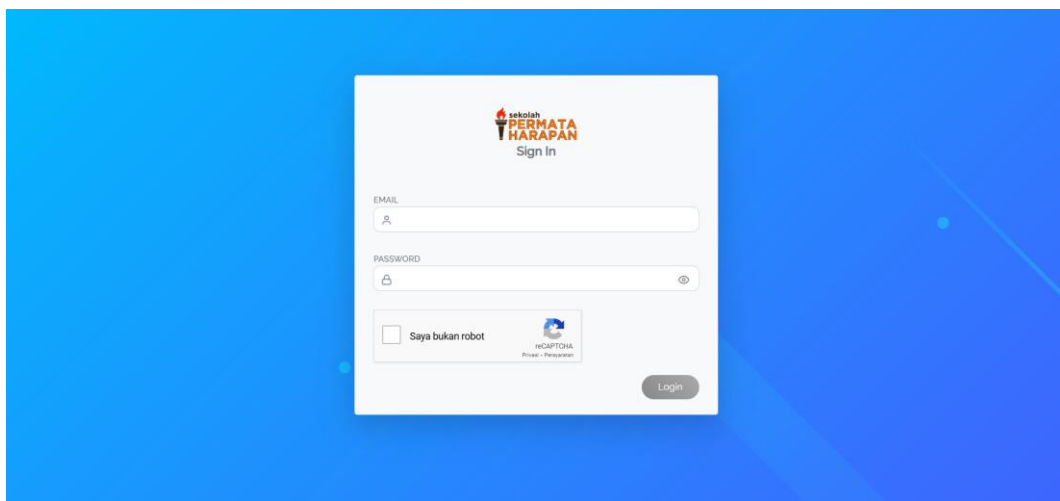
Flowmap sistem yang diusulkan menggambarkan alur kerja sistem yang melibatkan beberapa aktor, yaitu admin, guru, dan murid. Proses dimulai dari guru yang membuka sistem dan melakukan *login* dengan memasukkan *username* dan *password*. Sistem kemudian melakukan validasi terhadap informasi *login* tersebut. Jika *login* tidak valid, pengguna akan diminta memasukkan kembali informasi yang benar. Jika validasi berhasil, pengguna diarahkan ke *dashboard*, di mana mereka dapat membuat agenda. Setelah agenda dibuat dan tersimpan, agenda akan ditampilkan dan dapat diakses oleh murid setelah berhasil melakukan *login*. Di sisi lain, admin juga memiliki peran penting dalam sistem ini. Setelah *login*, admin dapat mengelola pengguna, jurusan, dan kelas, seperti menambah, menghapus, atau memperbarui data terkait pengguna, jurusan, dan kelas.

HASIL RANCANGAN

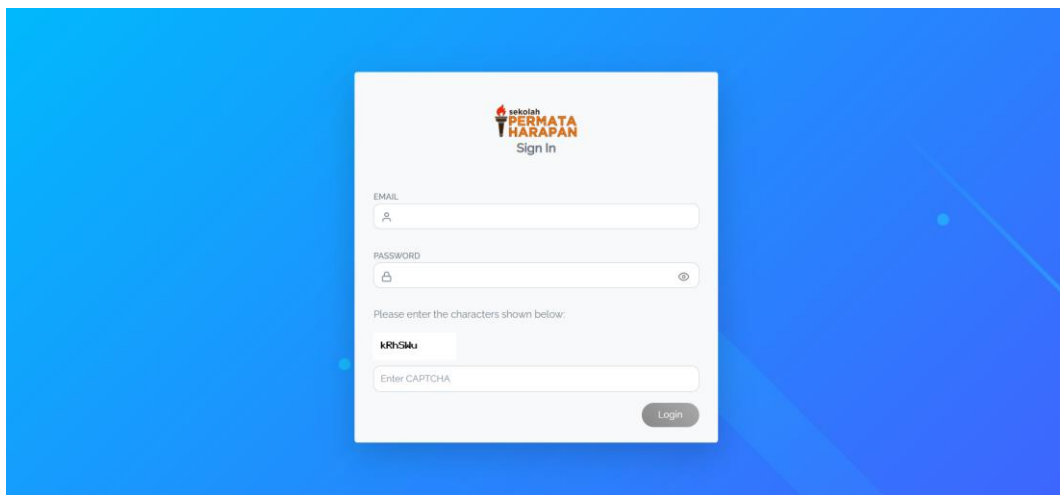
1. Halaman *Login*

Tampilan di bawah merupakan halaman *login* untuk pengguna, pengguna dapat melakukan *login* dengan cara memasukkan email dan *password* yang telah terdaftar dan mengeklik tombol *login*, kemudian terdapat 2 jenis captcha yang akan ditampilkan tergantung koneksi internet pengguna.

Tampilan halaman *login* terkoneksi internet.



Tampilan halaman *login* tidak terkoneksi internet.

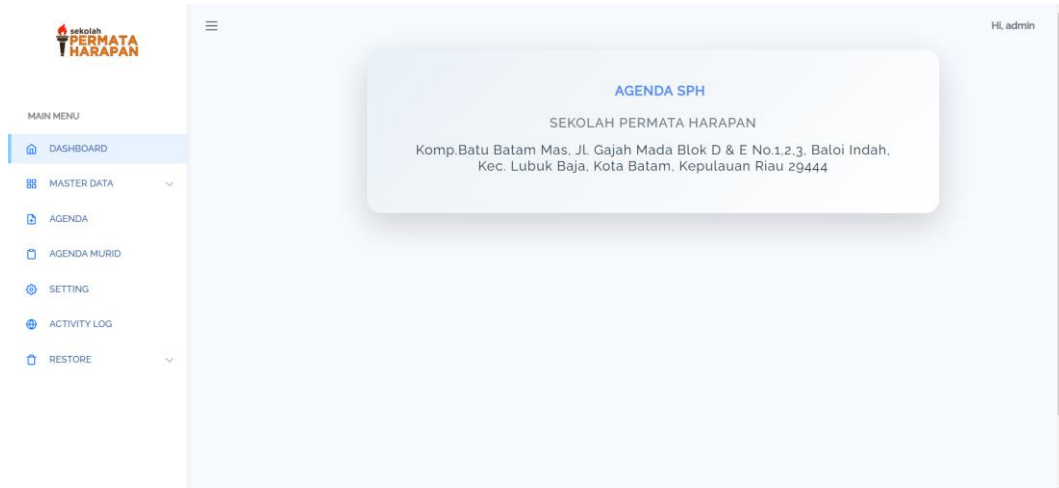


2. Menu *Dashboard*

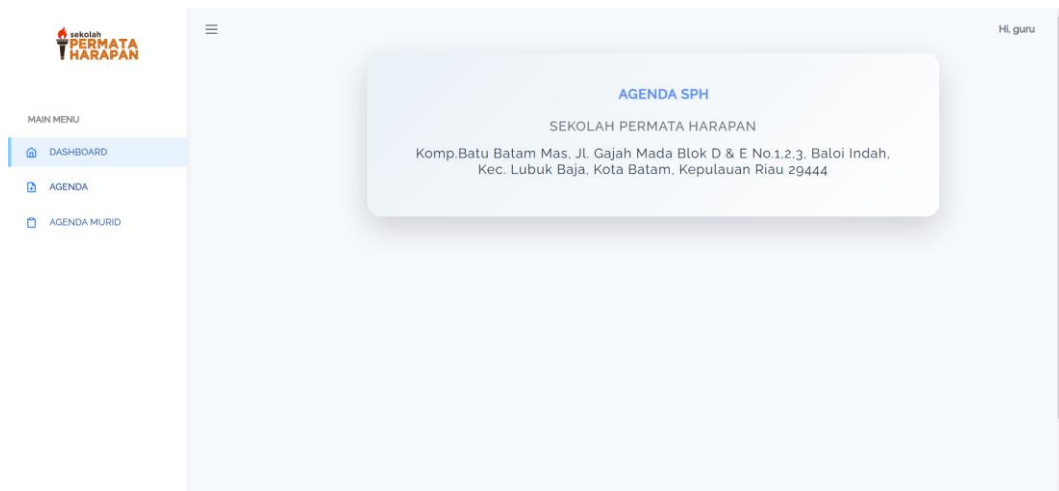
Menu *dashboard* merupakan tampilan utama yang ditampilkan kepada pengguna setelah berhasil melakukan *login* ke dalam sistem, kemudian pada pojok

kanan atas terdapat nama pengguna yang *login* dan saat diklik terdapat *dropdown* untuk *account* serta *logout*.

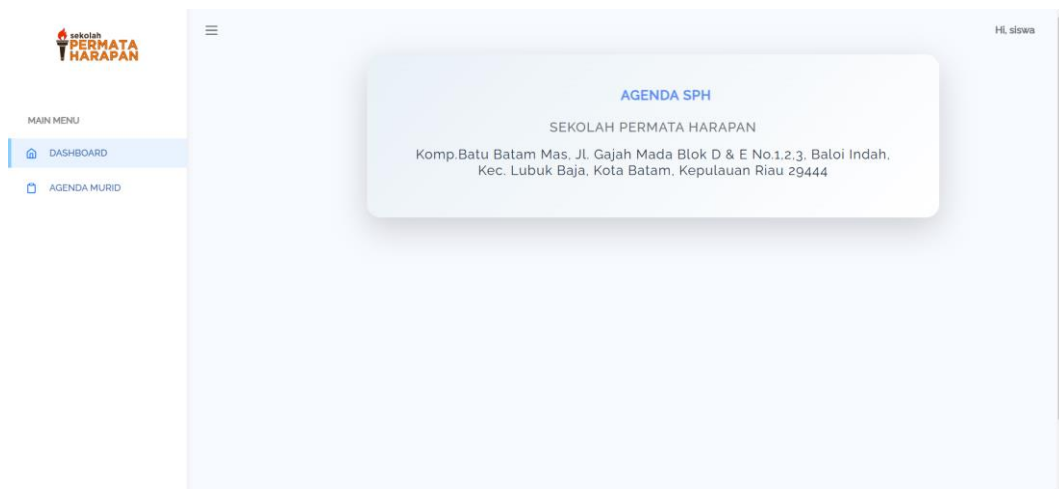
Menu admin, pengguna dengan level admin dapat mengakses semua menu pada sistem, yaitu menu *dashboard*, *user*, jurusan, kelas, agenda, agenda murid, *setting*, *activity log* dan *restore*.



Menu guru. Pengguna dengan guru dapat mengakses menu *dashboard*, agenda, dan agenda murid.

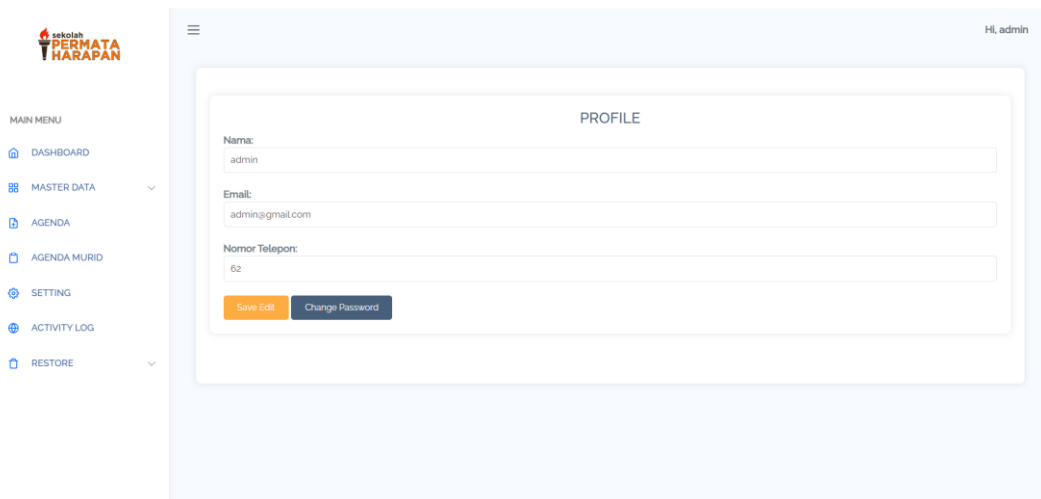


Menu murid. Pengguna dengan level siswa dapat mengakses menu *dashboard* dan agenda murid.



3. Menu *Account*

Menu *account* akan menampilkan informasi akun sesuai pengguna yang *login*, pengguna dapat mengubah informasi akun tersebut kemudian klik tombol *save edit* untuk simpan.



Pada menu *account* terdapat fungsi untuk mengubah *password* pengguna, ketika mengeklik tombol *change password* maka *form* untuk *change password* akan ditampilkan, pengguna harus memasukkan *password* lama yang sesuai, sebelum mengubah *password* dengan *password* yang baru, kemudian klik *confirm*.

sekolah PERMATA HARAPAN

Hi, admin

MAIN MENU

- DASHBOARD
- MASTER DATA
- AGENDA
- AGENDA MURID
- SETTING
- ACTIVITY LOG
- RESTORE

PROFILE

Name: admin

Email: admin@gmail.com

Nomor Telepon: 62

Save Edit Change Password

4. Menu *User*

Pada menu *user* terdapat tabel yang berisi data-data *user*, dan pengguna dapat menambah data *user*, mengubah, menghapus, mereset *password user*, serta *undo* edit data *user* yang telah diubah dengan mengeklik tombol yang sesuai. *Dropdown* edit, hapus, reset *password*, dan *undo* edit akan muncul saat pengguna mengeklik tombol aksi. Tombol *undo* edit hanya muncul saat data tersebut terdapat perubahan atau sudah pernah diubah.

sekolah PERMATA HARAPAN

Hi, admin

MAIN MENU

- DASHBOARD
- MASTER DATA
- AGENDA
- AGENDA MURID
- SETTING
- ACTIVITY LOG
- RESTORE

USER

Cari Cari Tambah

No	Nama	Level	Aksi
1	siswa	Murid	Aksi +
2	guru	Guru	Aksi +
3	admin	Admin	Aksi +

Berikut adalah tampilan *form* untuk menambah *user*, pengguna dapat memasukkan data *user* sesuai dalam *form*, dan mengeklik tombol *submit*.

Berikut adalah tampilan *form edit user* yang berfungsi untuk ubah data *user*, untuk edit *user* pengguna hanya perlu mengubah data yang ingin diubah kemudian klik tombol *submit*.

Berikut adalah tampilan *form undo edit user* yang berfungsi untuk mengembalikan data *user* yang telah diubah, pengguna dapat mengembalikan data tersebut dengan mengeklik tombol *undo edit*.

sekolah PERMATA HARAPAN

Hi, admin

MAIN MENU

- DASHBOARD
- MASTER DATA
- AGENDA
- AGENDA MURID
- SETTING
- ACTIVITY LOG
- RESTORE

Undo Edit

Level: Murid

Nama User: siswa Email: siswa@gmail.com

Nomor Telepon: 6282171810404 Kelas: XII A - REKAYASA PERANGKAT LUNAK

NIS: 22161017 NISN: 1

NIK: NIK NUPTK: NUPTK

Back Undo

5. Menu Kelas

Pada menu kelas terdapat tabel yang berisi data-data kelas, pengguna dapat menambah, mengubah, dan menghapus data kelas, serta mengembalikan data kelas yang telah diubah dengan mengeklik tombol yang sesuai. *Dropdown* edit, hapus, dan *undo* edit akan muncul saat pengguna mengeklik tombol aksi. Tombol *undo* edit hanya muncul saat data tersebut terdapat perubahan atau sudah pernah diubah.

sekolah PERMATA HARAPAN

Hi, admin

MAIN MENU

- DASHBOARD
- MASTER DATA
- AGENDA
- AGENDA MURID
- SETTING
- ACTIVITY LOG
- RESTORE

KELAS

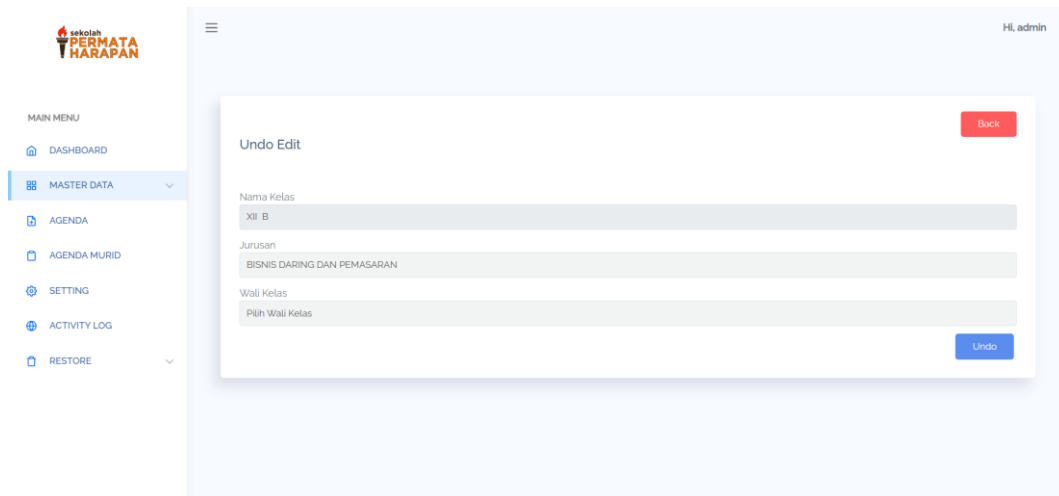
Cari Cari Tambah

No	Kelas	Jurusan	Wali Kelas	Aksi
1	XII B	BISNIS DARING DAN PEMASARAN	guru	Aksi -
2	XII A	REKAYASA PERANGKAT LUNAK	guru	Aksi -

Berikut adalah tampilan *form* untuk menambah kelas, pengguna dapat memasukkan data kelas sesuai dalam *form*, dan mengeklik tombol *submit*.

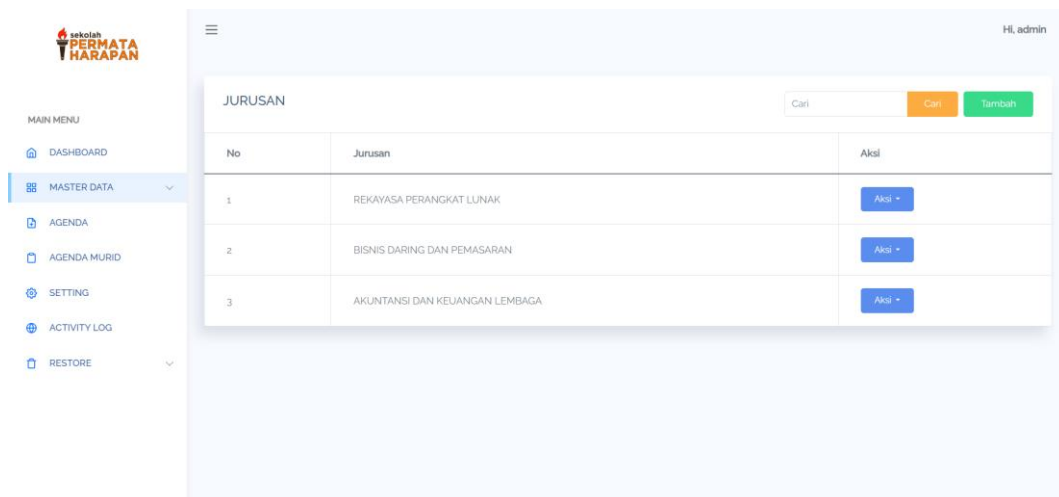
Berikut adalah tampilan *form* edit kelas yang berfungsi untuk ubah data kelas, untuk edit kelas pengguna hanya perlu mengubah data yang ingin diubah kemudian klik tombol *submit*.

Berikut adalah tampilan *form undo* edit kelas yang berfungsi untuk mengembalikan data kelas yang telah diubah, pengguna dapat mengembalikan data tersebut dengan mengeklik tombol *undo* edit.



6. Menu Jurusan

Pada menu jurusan terdapat tabel yang berisi data-data jurusan, pengguna dapat menambah, mengubah, dan menghapus data jurusan, serta mengembalikan data jurusan yang telah diubah dengan mengeklik tombol yang sesuai. *Dropdown* edit, hapus, dan *undo* edit akan muncul saat pengguna mengeklik tombol aksi. Tombol *undo* edit hanya muncul saat data tersebut terdapat perubahan atau sudah pernah diubah.



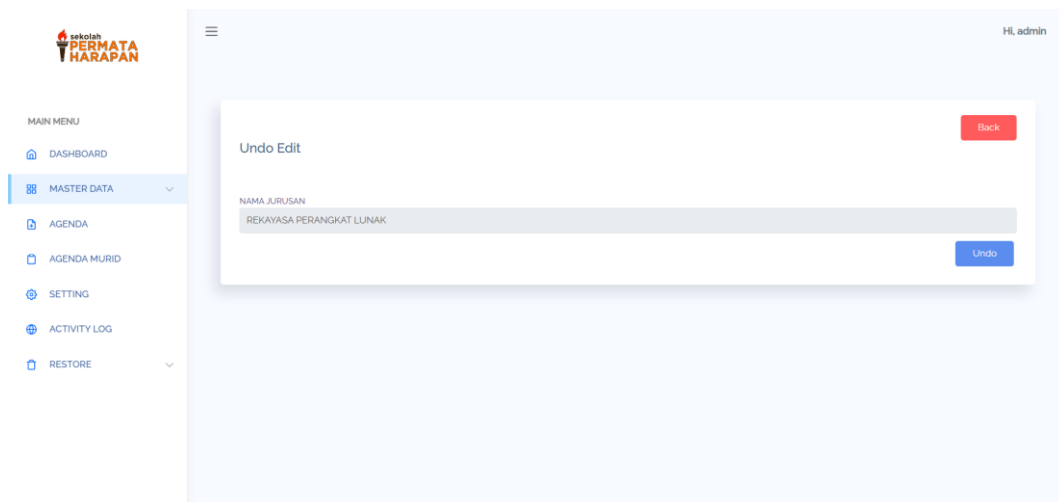
Berikut adalah tampilan *form* untuk menambah jurusan, pengguna dapat memasukkan data jurusan sesuai dalam *form*, dan mengeklik tombol *submit*.

The screenshot shows a web application interface for 'sekolah PERMATA HARAPAN'. On the left is a 'MAIN MENU' sidebar with options: DASHBOARD, MASTER DATA (selected), AGENDA, AGENDA MURID, SETTING, ACTIVITY LOG, and RESTORE. The main content area displays a 'Tambah' (Add) form for 'NAMA JURUSAN'. The form has a red 'Back' button at the top right, a text input field containing 'Nama Jurusan', and 'Submit' and 'Reset' buttons at the bottom right.

Berikut adalah tampilan *form* edit jurusan yang berfungsi untuk ubah data jurusan, untuk edit jurusan pengguna hanya perlu mengubah data yang ingin diubah kemudian klik tombol *submit*.

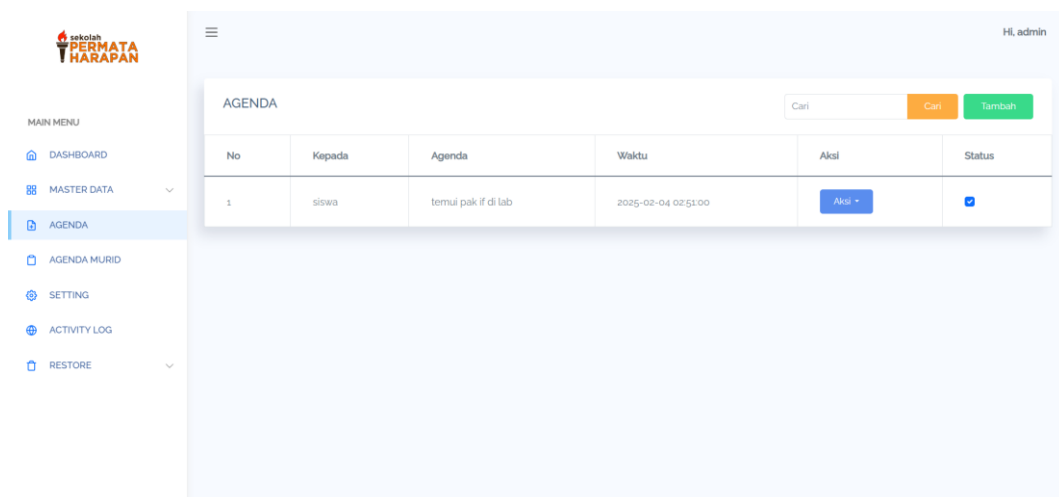
The screenshot shows the same web application interface. The 'MAIN MENU' sidebar remains the same. The main content area displays an 'Edit' form for 'NAMA JURUSAN'. The form has a red 'Back' button at the top right, a text input field containing 'REKAYASA PERANGKAT LUNAK', and a blue 'Submit' button at the bottom right.

Berikut adalah tampilan *form undo* edit jurusan yang berfungsi untuk mengembalikan data jurusan yang telah diubah, pengguna dapat mengembalikan data tersebut dengan mengeklik tombol *undo* edit.



7. Menu Agenda

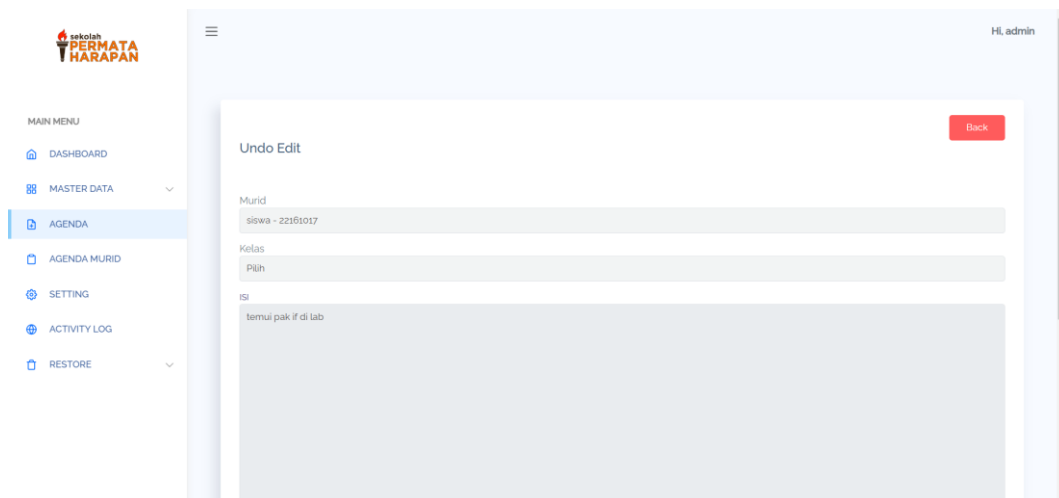
Pada menu agenda terdapat tabel yang berisi data-data agenda, pengguna dapat menambah, mengubah, dan menghapus data agenda, serta mengembalikan data agenda yang telah diubah dengan mengeklik tombol yang sesuai. *Dropdown* edit, hapus, dan *undo* edit akan muncul saat pengguna mengeklik tombol aksi. Tombol *undo* edit hanya muncul saat data tersebut terdapat perubahan atau sudah pernah diubah. Kemudian, terdapat kolom status untuk menceklis agenda yang telah diselesaikan oleh murid.



Berikut adalah tampilan *form* untuk menambah agenda, pengguna dapat memasukkan data agenda sesuai dalam *form*, dan mengeklik tombol *submit*.

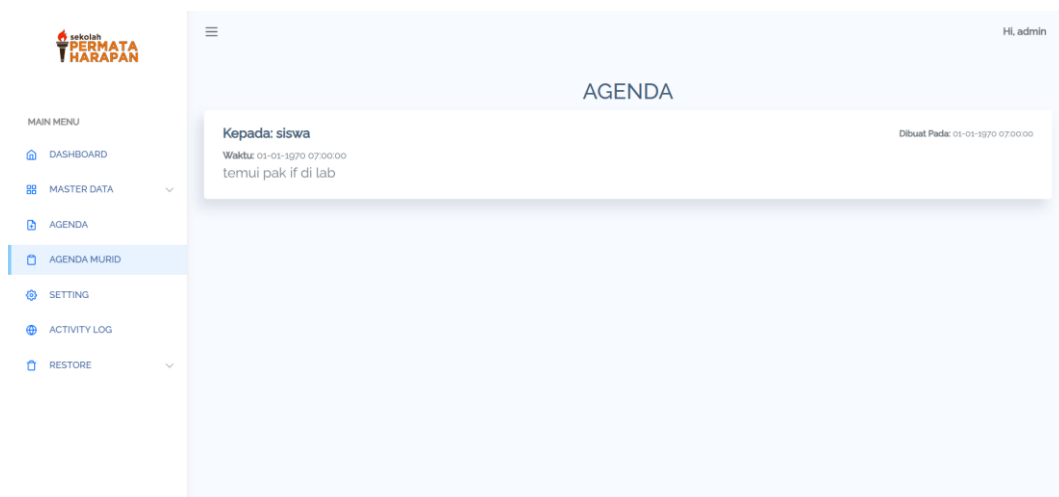
Berikut adalah tampilan *form* edit agenda yang berfungsi untuk ubah data agenda, untuk edit agenda pengguna hanya perlu mengubah data yang ingin diubah kemudian klik tombol *submit*.

Berikut adalah tampilan *form undo* edit agenda yang berfungsi untuk mengembalikan data agenda yang telah diubah, pengguna dapat mengembalikan data tersebut dengan mengeklik tombol *undo* edit.



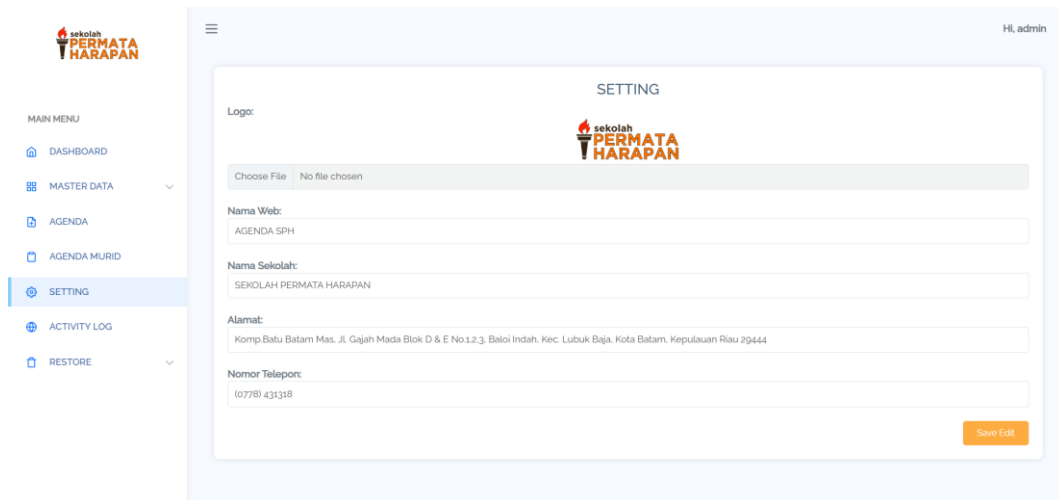
8. Menu Agenda Murid

Pada menu agenda murid, akan ditampilkan data dari agenda-agenda yang telah dibuat yang akan diberikan kepada murid.



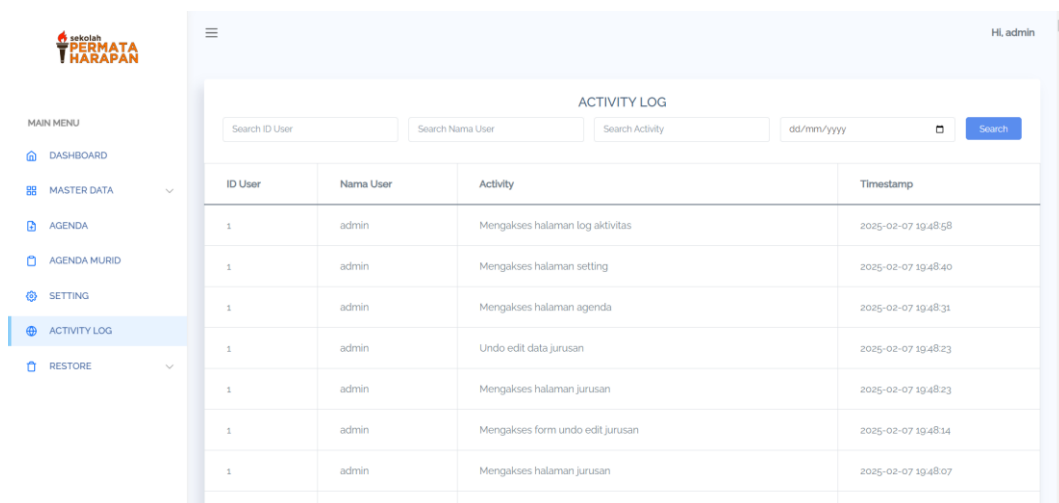
9. Menu Setting

Pada menu *setting* akan ditampilkan beberapa informasi dari sistem, pengguna dapat mengubah informasi tersebut dengan cara mengubah data dalam *form* dan klik tombol *save edit*.



10. Menu *Activity Log*


Pada menu *log activity*, akan ditampilkan aktivitas yang dilakukan oleh pengguna.



11. Menu *Restore*

Menu *restore* merupakan halaman yang menampilkan data-data yang telah dihapus oleh pengguna, pada halaman ini pengguna dapat mengembalikan data-data yang telah dihapus dengan cara mengeklik tombol *restore*.

Berikut adalah tampilan halaman *restore user*.




MAIN MENU

- DASHBOARD
- MASTER DATA
- AGENDA
- AGENDA MURID
- SETTING
- ACTIVITY LOG
- RESTORE

Hi, admin

RESTORE			
No	Nama	Level	Aksi
1	siswa	Murid	Detail Restore

Berikut adalah tampilan halaman *restore* kelas.




MAIN MENU

- DASHBOARD
- MASTER DATA
- AGENDA
- AGENDA MURID
- SETTING
- ACTIVITY LOG
- RESTORE

Hi, admin

RESTORE				
No	Kelas	Jurusan	Wali Kelas	Aksi
1	XII B	BISNIS DARING DAN PEMASARAN	guru	Restore

Berikut adalah tampilan halaman *restore* jurusan.



MAIN MENU

- DASHBOARD
- MASTER DATA
- AGENDA
- AGENDA MURID
- SETTING
- ACTIVITY LOG
- RESTORE

Hi, admin

RESTORE		
No	Jurusan	Aksi
1	AKUNTANSI DAN KEUANGAN LEMBAGA	Restore

Berikut adalah tampilan halaman *restore* agenda.

Logo: sekolah PERMATA HARAPAN

MAIN MENU

- DASHBOARD
- MASTER DATA
- AGENDA
- AGENDA MURID
- SETTING
- ACTIVITY LOG
- RESTORE

Hi, admin

RESTORE

No	Kepada	Waktu	Aksi
1	Siswa	2025-02-04 02:51:00	Detail Restore