

BAB III

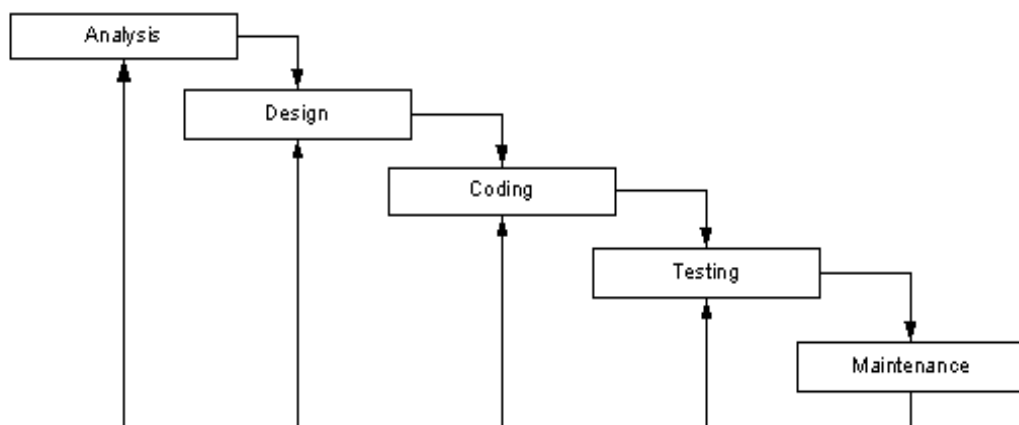
METODOLOGI PENELITIAN

3.1 Alur Penelitian

Dalam merancang dan membangun sistem, penulis menggunakan model penelitian SLDC (*Software Development Life Cycle*). Dalam SDLC terdapat beberapa metode yang dapat digunakan dalam mengembangkan sebuah *software* seperti *Waterfall*, *Prototype*, *RAD (Rapid Application Development)*, dan *Agile Software Development*. SDLC merupakan tahapan-tahapan pekerjaan yang dilakukan oleh analis sistem dan *programmer* dalam membangun sebuah sistem informasi. Dalam proses merancang dan membangun aplikasi ini penulis menggunakan metode *Waterfall*.

Disebut dengan *Waterfall* karena tahap demi tahap yang dilalui harus menunggu selesainya tahap sebelumnya dan berjalan berurutan antara lain *Analysis*, *Design*, *Coding*, *Testing*, dan *Maintenance*.

Tahap-tahap yang digunakan dapat dilihat pada gambar 3.1.



Gambar 3.1 Metode Waterfall

3.2 Alat dan Bahan

Dalam merancang dan membangun sebuah sistem informasi dibutuhkan alat dan bahan yang dapat menunjang proses tersebut.

3.2.1 Alat

Berikut alat yang digunakan dalam pembuatan aplikasi berupa perangkat keras dan perangkat lunak.

a. Perangkat Keras (*Hardware*)

Dalam proses pembuatan aplikasi dibutuhkan perangkat keras yang mampu menunjang kegiatan tersebut, perangkat keras yang digunakan penulis dapat dilihat pada tabel 3.1.

Tabel 3.1 Spesifikasi Perangkat Keras

Nama Perangkat	Spesifikasi
<i>Processor</i>	AMD A10-7300 Redeon R6, 10 Compute Cores 4C+6G 1.90GHz
RAM	8.00 GB
HDD	1000 GB
<i>Operating System</i>	Windows 10 Pro 64-bit

b. Perangkat Lunak (*Software*)

Selain perangkat keras yang digunakan, dibutuhkan juga perangkat lunak yang berfungsi sebagai media untuk pembuatan aplikasi, perangkat lunak yang digunakan penulis dapat dilihat pada tabel 3.2.

Tabel 3.2 Spesifikasi Perangkat Lunak

Nama Perangkat	Spesifikasi
<i>Operating System</i>	Windows 10 Pro 64-bit
<i>Tools yang digunakan</i>	Microsoft Visual Studio 2015 Microsoft SQL Server 2014 Google Chrome

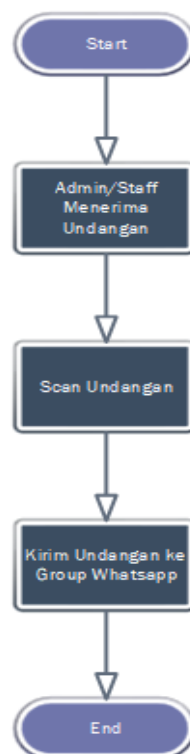
3.2.2 Bahan

Dalam proses merancang dan membuat aplikasi, penulis menggunakan bahan-bahan sebagai berikut:

- a. Contoh data user seperti nik, nama, *email*, dan *password*.
- b. Contoh data undangan.
- c. *Email* aktif yang digunakan untuk mengirim informasi undangan.

3.3 Analisis Kebutuhan

Dalam perancangan dan pembangunan sebuah sistem terdapat tahap analisis kebutuhan fungsional. Kebutuhan fungsional adalah kebutuhan pada sistem yang merupakan layanan dalam aplikasi, serta gambaran proses dari sistem terhadap masukan yang akan dikerjakan oleh sistem. Proses bisnis dalam penyampaian undangan di prodi Teknologi Informasi sebelumnya berdasarkan tanya jawab dengan *staff* TI UMY terkait adalah seperti pada gambar 3.2.



Gambar 3.2 Flowchart Proses Bisnis Penyampaian Undangan

Berdasarkan proses bisnis seperti yang ditunjukkan pada gambar 3.2, maka penulis menganalisa kebutuhan fungsional yang akan diterapkan pada aplikasi ini diantaranya sebagai berikut:

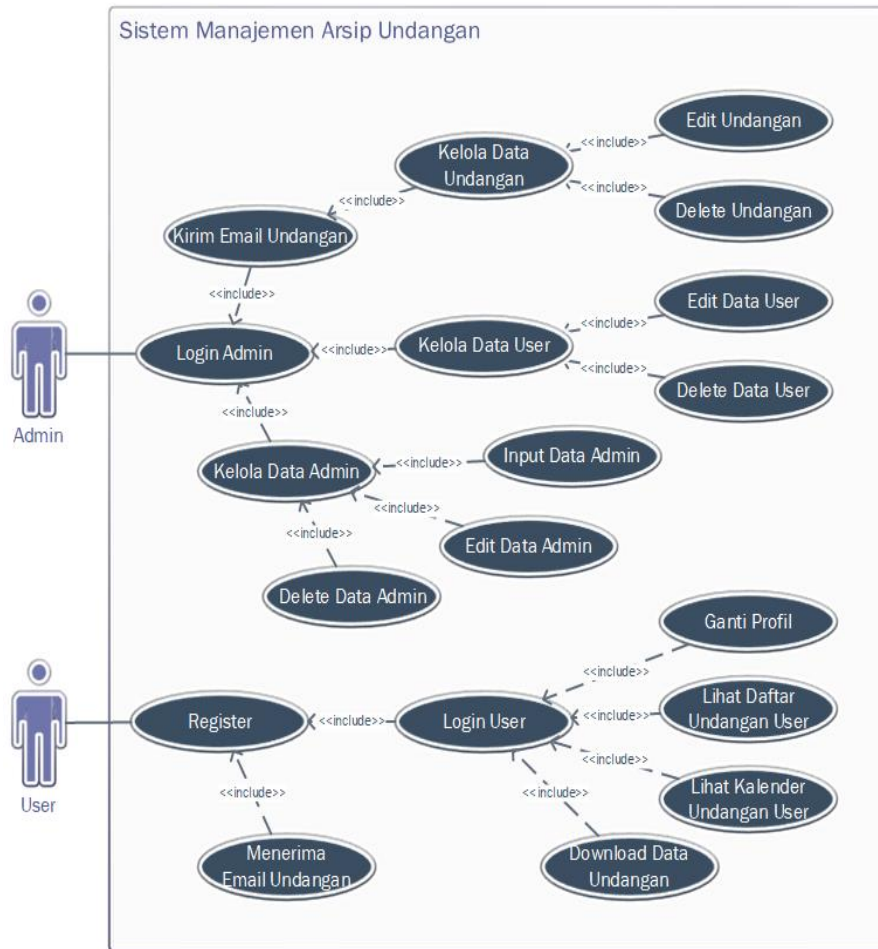
- a. Halaman untuk *login* sebagai tahap awal guna membedakan antara *admin* yang dapat mengelola data dengan *user* sebagai pengguna.
- b. Halaman untuk registrasi data *user*.
- c. Halaman untuk menampilkan serta melakukan proses *input, edit, delete,* dan *search* data undangan, data *user*, dan data *admin*.
- d. Halaman untuk mengirim informasi undangan melalui *email* ke *user*.
- e. Halaman untuk menampilkan informasi undangan, kalender undangan milik *user*, dan mengunduh undangan dalam bentuk dokumen.

3.4 Perancangan (*Design*)

Perancangan merupakan bagian dari metodologi pengembangan perangkat lunak yang dilakukan setelah tahapan analisis kebutuhan untuk memberikan gambaran secara terperinci terhadap sistem yang akan dibuat. Model perancangan yang digunakan dalam pembuatan Sistem Manajemen Arsip Undangan adalah sebagai berikut:

3.4.1 *Use Case Diagram*

Use Case Diagram menggambarkan aktivitas-aktivitas dan interaksi yang dilakukan oleh suatu sistem dari sudut pandang pengamatan luar. Pada Sistem Manajemen Arsip Undangan *Use Case Diagram* memiliki 2 aktor yaitu *Admin* yang memiliki 12 *use case* dan *User* yang memiliki 7 *use case*, seperti yang terlihat pada gambar 3.3 berikut.

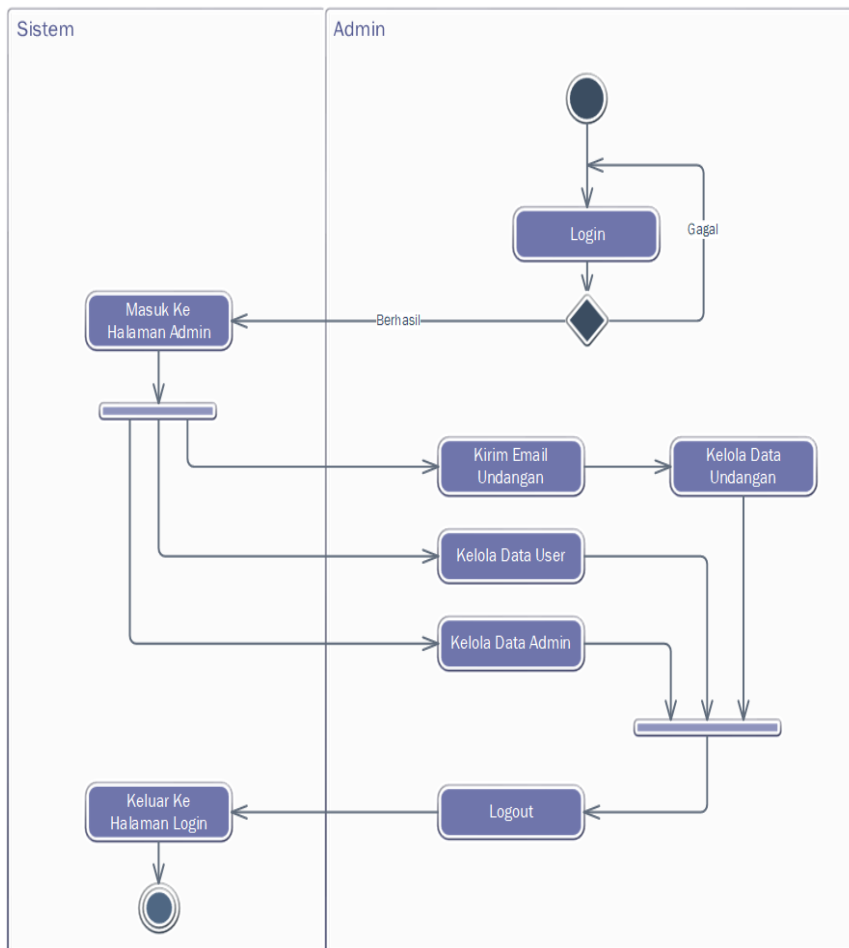


Gambar 3.3 Use Case Diagram

3.4.2 Activity Diagram

Activity diagram menggambarkan aliran aktivitas dalam sebuah sistem yang sedang dirancang, dari awal aliran pada sebuah sistem, decision yang mungkin terjadi, dan akhir dari aliran pada sistem tersebut. *Activity* Diagram juga dapat menggambarkan proses paralel yang mungkin terjadi pada beberapa aktivitas. *Activity* Diagram dari Sistem Manajemen Arsip Undangan adalah sebagai berikut:

1. Activity Diagram Admin



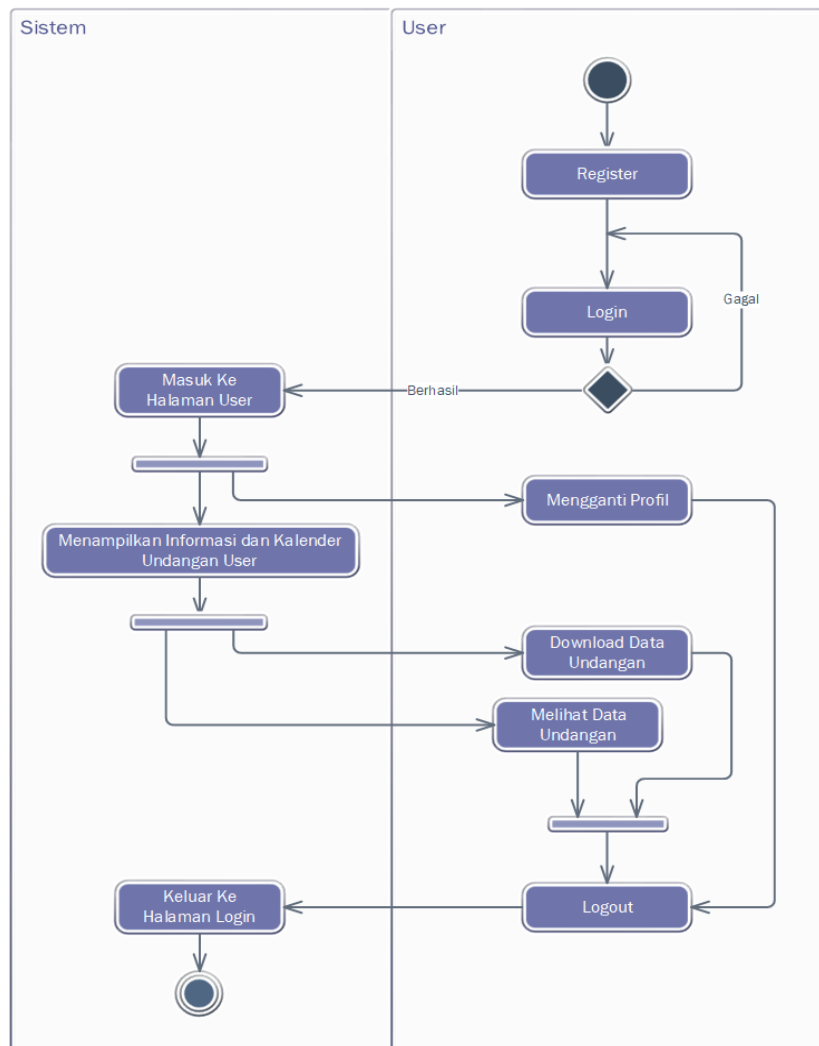
Gambar 3.4 Activity Diagram Admin

Keterangan gambar 3.4:

- Admin* memasukkan *email* dan *password* untuk melakukan proses *login*.
- Jika *login* gagal maka sistem akan menampilkan pesan *error* dan kembali ke *step* pertama
- Jika *login* berhasil maka sistem akan masuk ke halaman *admin*.
- Admin* mengirim *email* undangan dan data akan tersimpan dalam *database*.
- Admin* melakukan kelola data undangan seperti *edit* dan *delete*.

- f. *Admin* melakukan kelola data *user* seperti *edit* dan *delete*.
- g. *Admin* melakukan kelola data *admin* seperti *input*, *edit* dan *delete*.
- h. *Admin* melakukan *logout*.
- i. Sistem keluar ke halaman *login*.

2. Activity Diagram User



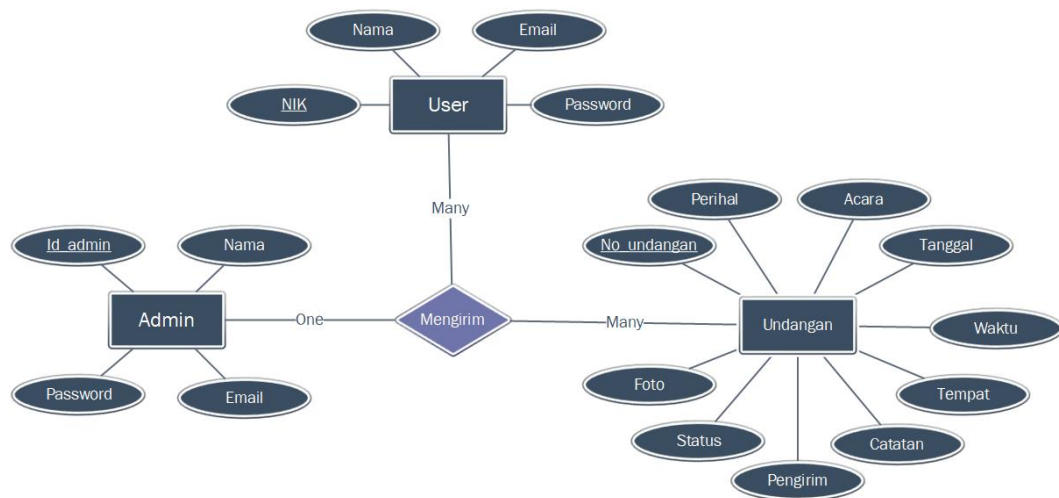
Gambar 3.5 Activity Diagram User

Keterangan gambar 3.5:

- a. *User* melakukan *register* akun.
- b. *User* memasukkan *email* dan *password* untuk melakukan proses *login*.
- c. Jika *login* gagal maka sistem akan menampilkan pesan *error* dan tetap pada *step* untuk *login*.
- d. Jika *login* berhasil maka sistem akan masuk ke halaman *user*.
- e. *User* melakukan ganti profil *user*.
- f. Sistem menampilkan daftar informasi undangan sesuai dengan yang terkirim ke *email user*.
- g. Sistem menampilkan kalender undangan sesuai dengan yang terkirim ke *email user*.
- h. *User* melihat informasi undangan.
- i. *User* melakukan *download* data undangan dalam bentuk file pdf.
- j. *User* melakukan *logout*.
- k. Sistem keluar ke halaman *login*.

3.4.3 *Entity Relationship Diagram* (ERD)

Entity Relationship Diagram merupakan suatu rancangan sistem yang berhubungan pada suatu kegiatan yang digambarkan dengan hubungan antar entitas dimana setiap entitas terdiri dari beberapa atribut yang memiliki fungsi dalam proses tersebut. ERD dari sitem ini dapat dilihat pada gambar 3.6.



Gambar 3.6 Entity Relationship Diagram

Pada gambar 3.6 dapat dilihat bahwa sistem yang dirancang memiliki 3 entitas yaitu *Admin*, *Users*, dan *Undangan* dan memiliki relasi mengirim yang akan menghasilkan satu tabel *Log_undangan*. Berikut ini merupakan penjelasan dari setiap entitas pada ER-Diagram Sistem Manajemen Arsip Undangan:

1. Entitas *Admin* yang memiliki atribut *Id_admin* (*primary key* dari tabel *Admin*), *Nama*, *Email*, dan *Password* memiliki relasi dengan entitas *Undangan* dan entitas *User*. Relasi yang terbentuk adalah *one-to-many* karena satu *admin* dapat mengirim lebih dari satu undangan dan kepada lebih dari satu *user*.
2. Entitas *User* yang memiliki atribut *NIK* (*primary key* dari tabel *User*), *Nama*, *Email*, dan *Password* memiliki relasi dengan entitas *Admin*. Relasi yang terbentuk adalah *one-to-many* karena satu *admin* dapat mengirim undangan kepada lebih dari satu *user*.
3. Entitas *Undangan* memiliki atribut *No_undangan* (*primary key* dari tabel *Undangan*), *Perihal*, *Acara*, *Tanggal*, *Waktu*, *Tempat*, *Catatan*, *Status*, *Pengirim* dan *Foto* dan memiliki relasi dengan entitas *Admin*. Relasi yang terbentuk adalah *one to many* karena *admin* dapat mengirim lebih dari satu undangan kepada *user*.
4. Relasi mengirim yang terjadi antara entitas *Admin* dengan entitas *Undangan* dan entitas *User* menghasilkan satu tabel baru yaitu tabel

Log_undangan. Tabel Log_undangan memiliki Id_log (*primary key* dari tabel Log_undangan), No_undangan (*foreign key* No_Undangan dari tabel undangan), Email (*foreign key email* dari tabel users), Status, serta Tgl_kirim dan Tgl_baca.

3.4.4 Kamus Database

Berikut ini merupakan tabel *database* ManajemenUndangan dari Sistem Manajemen Arsip Undangan.

Tabel 3.3 Tabel Admin

Nama Kolom	Tipe Data	Panjang	Key	Keterangan
<i>Id_admin</i>	<i>int</i>	<i>Auto increment</i>	<i>Primary Key</i>	ID dari tabel <i>Admin</i>
Nama	<i>varchar</i>	50		Nama pengguna sebagai <i>admin</i>
<i>Email</i>	<i>varchar</i>	50		Alamat <i>email admin</i> sebagai <i>username</i> ketika <i>login</i>
<i>Password</i>	<i>varchar</i>	15		Kata sandi pengguna sebagai <i>admin</i>

Tabel *Admin* seperti ditunjukkan pada tabel 3.3 berisi data-data *admin*. Pada tabel ini terdapat *Id_admin* yang merupakan *primary key* dengan tipe data *integer* dengan panjang data otomatis yang terisi secara berurutan, Nama dengan tipe data *varchar* dan panjang data 50, *Email* dengan tipe data *varchar* yang memiliki panjang 50, serta *Password* dengan tipe data *varchar* yang memiliki panjang 15.

Tabel 3.4 Tabel Users

Nama Kolom	Tipe Data	Panjang	Key	Keterangan
NIK	<i>Varchar</i>	25	<i>Primary Key</i>	Nomor Induk Kepegawaian atau ID dari <i>User</i>
Nama	<i>Varchar</i>	50		Nama pengguna sebagai <i>user</i>
<i>Email</i>	<i>Varchar</i>	50		Alamat <i>email user</i> sebagai <i>username</i> ketika <i>login</i> dan alamat tujuan pengiriman <i>email undangan</i>
<i>Password</i>	<i>Varchar</i>	15		Kata sandi pengguna sebagai <i>user</i>

Tabel *Users* seperti ditunjukkan pada tabel 3.4 berisi data-data *user* yaitu dosen, *staff*, dan karyawan di jurusan Teknologi Informasi. Pada tabel ini terdapat NIK dengan tipe data *varchar* dan panjang data 20, Nama dengan tipe data *varchar* dan panjang data 50, serta *Email* yang merupakan *primary key* dengan tipe data *varchar* yang memiliki panjang data 50, serta *Password* dengan tipe data *varchar* yang memiliki panjang 15.

Tabel 3.5 Tabel Undangan

Nama Kolom	Tipe Data	Panjang	Key	Keterangan
No_undangan	<i>varchar</i>	50	<i>Primary Key</i>	Nomor dari Undangan
Perihal	<i>varchar</i>	50		Hal dari undangan tersebut dibuat

Nama Kolom	Tipe Data	Panjang	Key	Keterangan
Acara	<i>varchar</i>	50		Kegiatan atau acara dari undangan
Tanggal	<i>varchar</i>	30		Tanggal undangan dilaksanakan
Waktu	<i>varchar</i>	30		Waktu undangan pada tanggal tersebut dilaksanakan
Tempat	<i>varchar</i>	50		Tempat atau alamat undangan dilaksanakan
Catatan	<i>varchar</i>	100		Berisi catatan terkait undangan
Pengirim	<i>varchar</i>	50		Nama pengirim undangan
Status	<i>varchar</i>	15		Status undangan tersebut terkirim atau belum terkirim ke <i>user</i>
Foto	<i>image</i>			Foto dari undangan asli

Tabel Undangan seperti ditunjukkan pada tabel 3.5 berisi data-data undangan. Pada tabel ini terdapat *No_undangan* yang merupakan *primary key* dengan tipe data *varchar* dan panjang data 50, *Perihal* dengan tipe data *varchar* dan panjang data 50, *Acara* dengan tipe data *varchar* dan panjang data 50, *Tanggal* yang dengan tipe data *varchar* dan panjang data 30 yang sudah diberi format pada aplikasi, *Waktu* yang dengan tipe data *varchar* dan panjang data 30, *Tempat* dengan tipe data *varchar* dan panjang data 50, *Catatan* dengan tipe data *varchar* dan panjang data 100, *Pengirim* dengan tipe data *varchar* dan panjang data 50, *Status* dengan tipe data *varchar* dan panjang data 15, serta *Foto* dengan tipe data *image*.

Tabel 3.6 Tabel Log_undangan

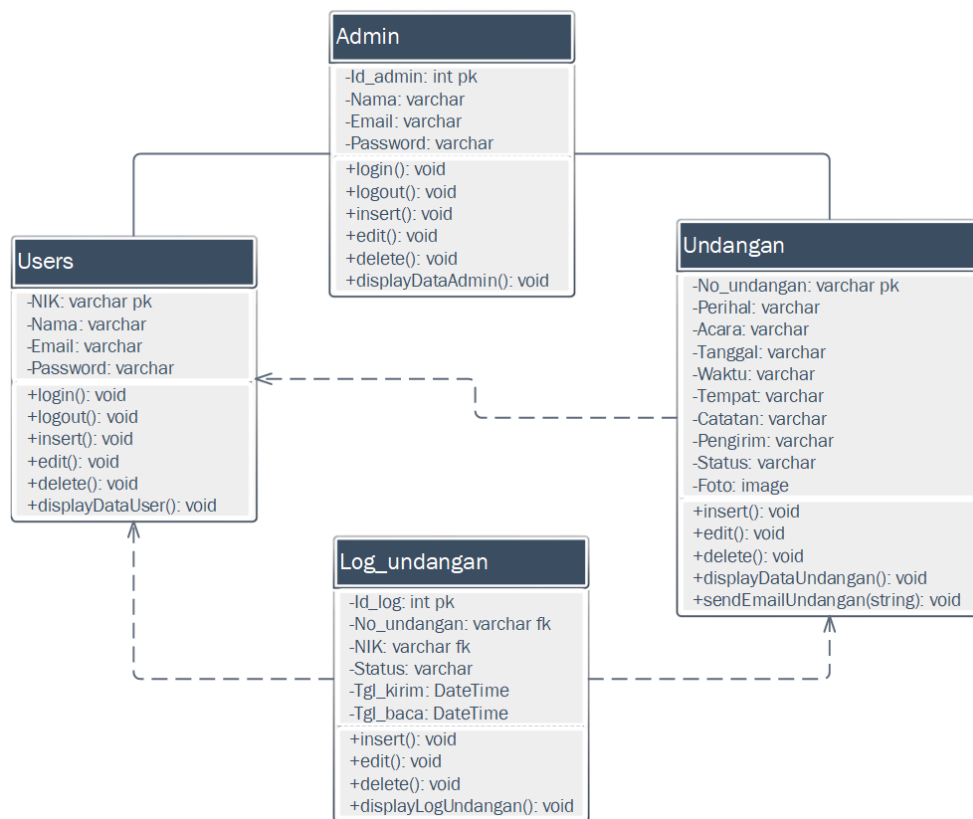
Nama Kolom	Tipe Data	Panjang	Key	Keterangan
Id_log	<i>int</i>	<i>Auto increment</i>	<i>Primary Key</i>	ID dari tabel Log_undangan
No_undangan	<i>varchar</i>	50	<i>Foreign key</i>	Nomor dari Undangan
NIK	<i>varchar</i>	25	<i>Foreign key</i>	Nomor Induk Kepegawaian atau ID dari <i>User</i>
Status	<i>varchar</i>	8		Status undangan tersebut diterima atau belum diterima oleh <i>user</i>
Tgl_kirim	<i>DateTime</i>			Tanggal dimana undangan dikirimkan kepada <i>user</i>
Tgl_baca	<i>DateTime</i>			Tanggal dimana undangan telah dibaca oleh user

Tabel Log_UndanganUser seperti ditunjukkan pada tabel 3.6 berisi data-data undangan yang memiliki fungsi untuk mencatat semua data undangan yang terkirim ke *user* melalui *email* dan akan ditampilkan pada halaman *user* nantinya dimana data yang ditampilkan hanya data yang sesuai dengan data undangan yang ditujukan kepada masing-masing *user*. Pada tabel ini terdapat Id_log yang merupakan *primary key* dengan tipe *integer* dengan panjang data otomatis yang terisi secara berurutan, No_undangan yang merupakan *foreign key* dari tabel Undangan dengan tipe data *varchar* dan panjang data 50, serta

NIK yang merupakan *foreign key* dan berelasi dengan NIK sebagai *primary key* dari tabel Users dengan tipe data *varchar* dan panjang data 25, Status dengan tipe data *varchar* dan panjang data 8, serta Tgl_kirim dan Tgl_baca dengan tipe data *DateTime*.

3.4.5 Class Diagram

Class Diagram memberikan pandangan secara luas dari suatu sistem dengan menunjukkan kelas-kelasnya dan hubungan antar kelas tersebut. *Class Diagram* dari Sistem Manajemen Arsip Undangan dapat dilihat pada gambar 3.7.



Gambar 3.7 Class Diagram

Pada sebuah *class* pada *class diagram* memiliki nama dari sebuah kelas, variabel yang merupakan properti dari sebuah kelas, serta operasi atau method yang merupakan hal yang bisa dilakukan oleh sebuah kelas terhadap kelas lain. Berikut ini merupakan penjelasan dari gambar 3.7 *class diagram* Sistem Manajemen Arsip Undangan:

1. *Class Admin*

Class Admin mempunyai variabel *Id_admin* sebagai *primary key*, Nama, serta *Email* dan *Password*. Pada *Class Admin* terdapat *method* atau fungsi, seperti *login* untuk masuk ke halaman *admin*, *logout* untuk keluar dari halaman *admin*, *input* data, *edit* data, *delete* data, *displayDataAdmin* untuk menampilkan data Admin.

2. *Class Users*

Class Users mempunyai variabel NIK sebagai *primary key*, Nama, serta *Email* dan *Password*. Pada *Class User* terdapat *method* atau fungsi seperti *login* untuk masuk ke halaman *user*, *logout* untuk keluar dari halaman *user*, *insert* untuk menambah data *user* baru, *edit* data, *delete* data, dan *displayDataUser* untuk menampilkan semua data *user* yang terdapat dalam *database*.

3. *Class Undangan*

Class Undangan mempunyai variabel *No_undangan* sebagai *primary key*, Perihal, Acara, Tanggal, Waktu, Tempat, Catatan, Pengirim, Status, dan Foto sebagai gambar dari undangan yang asli. *Class Undangan* memiliki *method* atau fungsi seperti *input* data, *edit* data, *delete* data, serta *displayDataUndangan* untuk menampilkan semua undangan yang terdapat dalam *database*, dan *sendEmailUndangan* untuk mengirim data undangan kepada *user*. *Class Undangan* memiliki relasi dengan *class Users* yaitu dimana class undangan membutuhkan variabel *Email* dari *class Users* sebagai tujuan pengiriman undangan.

4. *Class Log_UndanganUser*

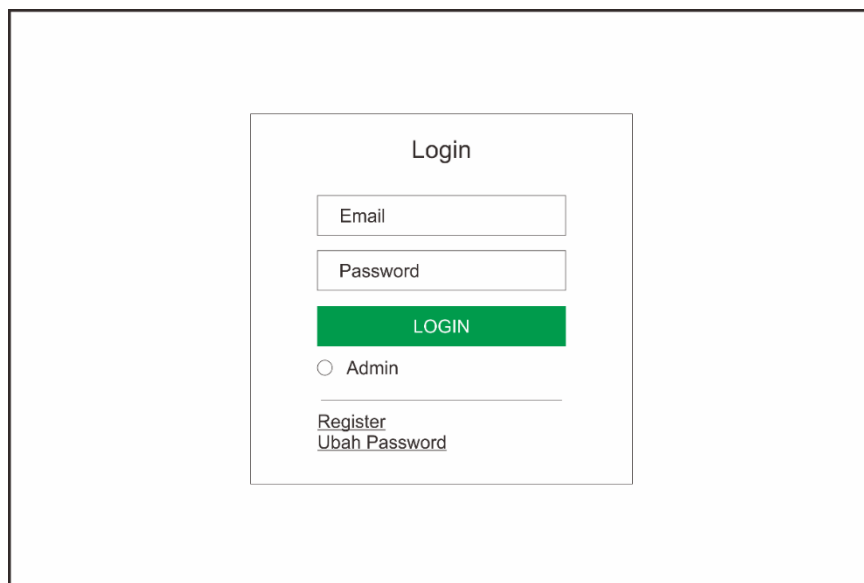
Class Log_UndanganUser mempunyai variabel *Id_log* sebagai *primary key*, *No_undangan* sebagai *foreign key* dari *class Undangan*, NIK sebagai *foreign key* dari *class Users*, Status sebagai tanda sudah dibaca atau tidaknya undangan tersebut, serta *Tgl_kirim* dan *Tgl_baca* sebagai waktu dari dikirim dan dibacanya undangan. *Class Log_UndanganUser* memiliki relasi *dependency* dengan *class Undangan* dan *class Users*

yang memerlukan data dari *class* Undangan dan data dari *class* Users sebagai penerima undangan.

3.4.6 Perancangan *Interface*

1. *Interface* Halaman *Login*

Berikut ini merupakan perancangan *interface* dari halaman *login* yang dapat dilihat pada gambar 3.8.



The image shows a login form with the following elements:

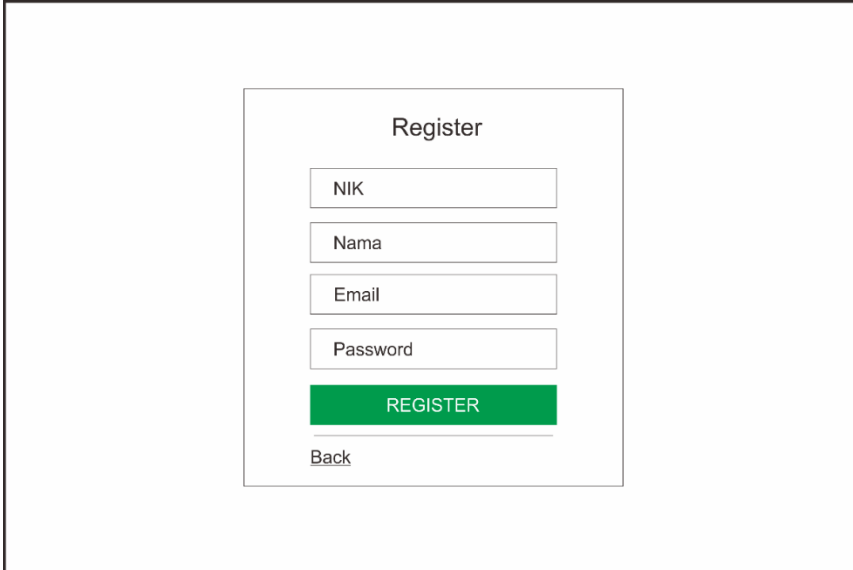
- Title: Login
- Input field: Email
- Input field: Password
- Button: LOGIN (green background)
- Radio button: Admin
- Links: Register, Ubah Password

Gambar 3.8 Perancangan Halaman Login

Pada gambar 3.8 terdapat 2 *textbox*, 1 *radio button* untuk membedakan antara *admin* atau *user*, dan 1 *button* untuk proses *login* ke halaman *admin* dan *user*, serta 2 link untuk menuju ke halaman *register user* dan lupa *password*.

2. *Interface* Halaman *Register User*

Berikut ini merupakan perancangan *interface* dari halaman *register user* yang dapat dilihat pada gambar 3.9.



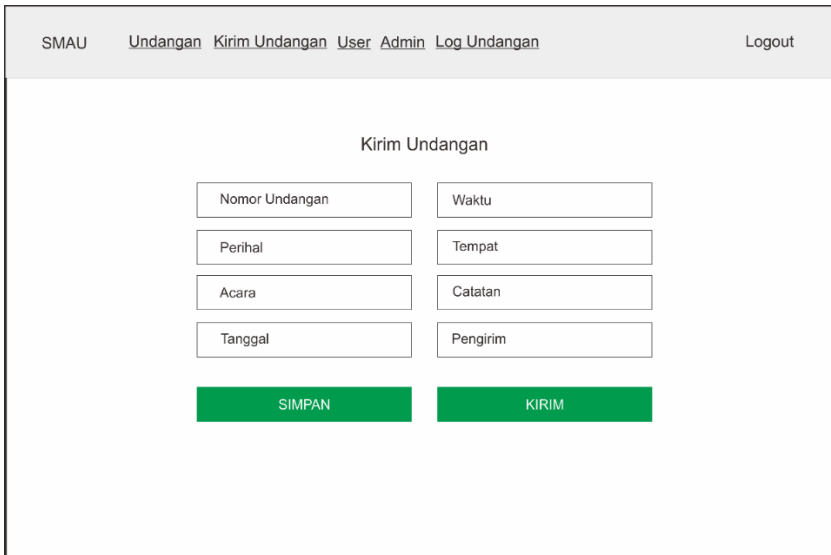
The image shows a registration form titled "Register". It contains four text input fields: "NIK", "Nama", "Email", and "Password". Below these fields is a green button labeled "REGISTER". At the bottom left of the form area, there is a "Back" link.

Gambar 3.9 Perancangan Halaman Register

Pada gambar 3.9 terdapat 4 *textbox*, dan 1 *button* untuk proses *registrasi*, serta *link* untuk kembali ke halaman *login*.

3. *Interface* Halaman Kirim Undangan

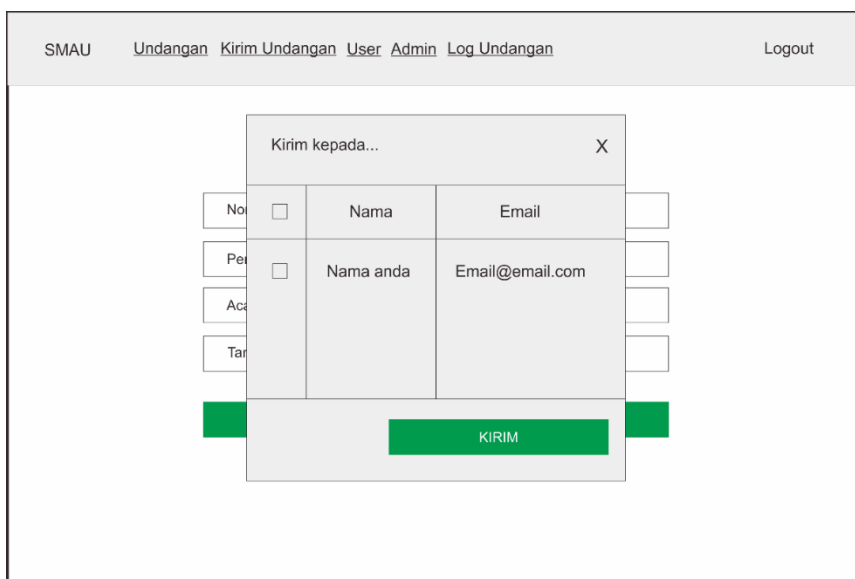
Berikut ini merupakan perancangan *interface* dari halaman kirim undangan.



The image shows a form titled "Kirim Undangan" (Send Invitation). At the top, there is a navigation bar with links: SMAU, Undangan, Kirim Undangan, User, Admin, Log Undangan, and Logout. The form contains eight text input fields arranged in two columns: "Nomor Undangan", "Waktu", "Perihal", "Tempat", "Acara", "Catatan", "Tanggal", and "Pengirim". Below the fields are two green buttons: "SIMPAN" (Save) and "KIRIM" (Send).

Gambar 3.10 Perancangan Halaman Kirim Undangan

Pada gambar 3.10 terdapat 8 *textbox*, dan 2 *button* untuk proses menyimpan dan mengirim undangan, serta navigasi untuk menuju ke halaman undangan, kirim undangan, *user*, *admin*, atau *Log Undangan*. Ketika semua *textbox* terisi dan tombol simpan diklik maka data akan masuk ke tabel Undangan dalam *database* dengan status belum terkirim, ketika tombol kirim diklik maka akan muncul sebuah *pop up* untuk memilih kepada siapa undangan tersebut akan dikirimkan seperti pada gambar 3.11.



Gambar 3.11 Perancangan Fitur Kirim Kepada

Pada gambar 3.11 terdapat terdapat tabel daftar nama serta *email user* dan *checkbox* untuk memilih kepada siapa undangan akan dikirimkan. Ketika tombol kirim diklik maka sistem akan mengirim undangan melalui *email* dan data akan otomatis tersimpan pada *database* tabel Undangan dengan status terkirim dan tersimpan pada tabel *Log_undangan* dengan penerima sesuai dengan alamat *email user* yang telah dikirimkan *email* undangan.

4. *Interface* Halaman Undangan

Berikut ini merupakan perancangan *interface* dari halaman undangan yang dapat dilihat pada gambar 3.12.

Undangan

Cari...

No Und	Perihal	Acara	Tanggal	Waktu	Tempat	Pengirim	Status	
								Detail Delete

Nomor Undangan Waktu

Perihal Tempat

Acara Catatan

Tanggal Pengirim

UBAH
KIRIM ULANG

Gambar 3.12 Perancangan Halaman Undangan

Pada gambar 3.12 terdapat 1 tabel untuk menampilkan data undangan dimana pada setiap data terdapat 2 pilihan detail untuk menampilkan data pada *textbox* dibawah tabel dan *delete* untuk menghapus data, 8 *textbox* dan 2 *button* untuk proses mengubah dan mengirim kembali undangan, serta navigasi untuk menuju ke halaman undangan, kirim undangan, *user*, *admin*, atau Log Undangan. 1 *textbox* untuk mencari data undangan pada tabel.

5. *Interface* Halaman *User*

Berikut ini merupakan perancangan *interface* dari halaman *user* yang dapat dilihat pada gambar 3.13.

SMAU Undangan Kirim Undangan User Admin Log Undangan Logout

User

Cari...

NIK	Nama	Email	
			Detail Delete

SIMPAN UBAH

Gambar 3.13 Perancangan Halaman User

Pada gambar 3.13 terdapat 1 tabel untuk menampilkan data *user* dimana pada setiap data terdapat 2 pilihan detail untuk menampilkan data pada *textbox* dibawah tabel dan *delete* untuk menghapus data dan 1 *textbox* untuk mencari data *user* pada tabel. Terdapat 4 *textbox* serta 2 *button* untuk menyimpan dan mengubah data *user*.

6. *Interface* Halaman Admin

Berikut ini merupakan perancangan *interface* dari halaman *admin* yang dapat dilihat pada gambar 3.14.

The screenshot shows an Admin interface with the following components:

- Header:** SMAU, Undangan, Kirim Undangan, User, Admin, Log Undangan, Logout
- Title:** Admin
- Search:** Cari...
- Table:**

Id Admin	Nama	Email	
			Detail Delete
- Form Fields:** ID Admin, Nama, Email, Password
- Buttons:** SIMPAN, UBAH

Gambar 3.14 Perancangan Halaman Admin

Pada gambar 3.14 terdapat 1 tabel untuk menampilkan data *admin* dimana pada setiap data terdapat 2 pilihan detail untuk menampilkan data pada *textbox* dibawah tabel dan *delete* untuk menghapus data dan 1 *textbox* untuk mencari data admin pada tabel. Terdapat 4 *textbox* serta 1 *button* untuk mengubah data *admin*.

7. *Interface* Halaman Log Undangan

Berikut ini merupakan perancangan *interface* dari halaman *log* undangan yang dapat dilihat pada gambar 3.15.

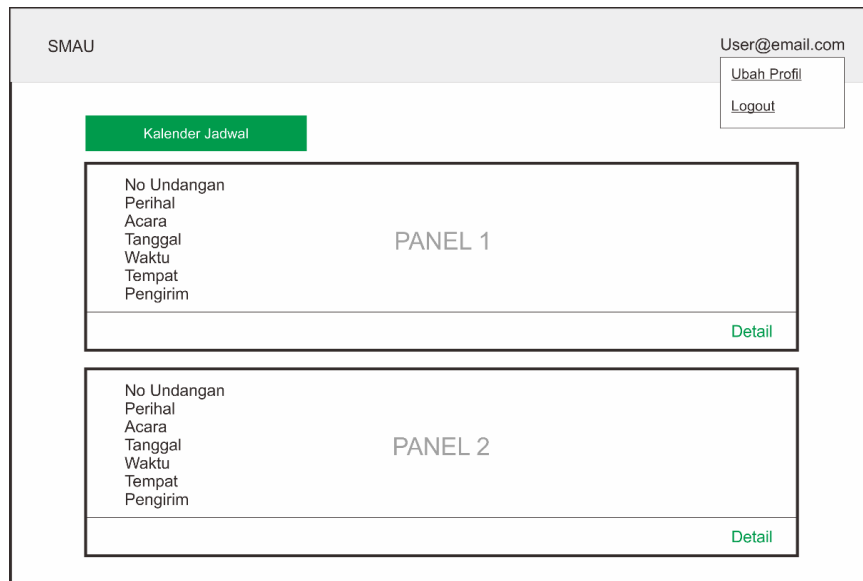
ID Log	No Undangan	NIK	Nama	Status

Gambar 3.15 Perancangan Halaman Log Undangan

Pada gambar 3.15 terdapat 1 tabel untuk menampilkan data undangan semua semua user yang telah terkirim. Pada halaman ini juga terdapat 1 *textbox* untuk mencari data undangan berdasarkan nama *user* pada tabel.

8. *Interface* Halaman Undangan *User*

Berikut ini merupakan perancangan *interface* dari halaman undangan *user* yang dapat dilihat pada gambar 3.16.

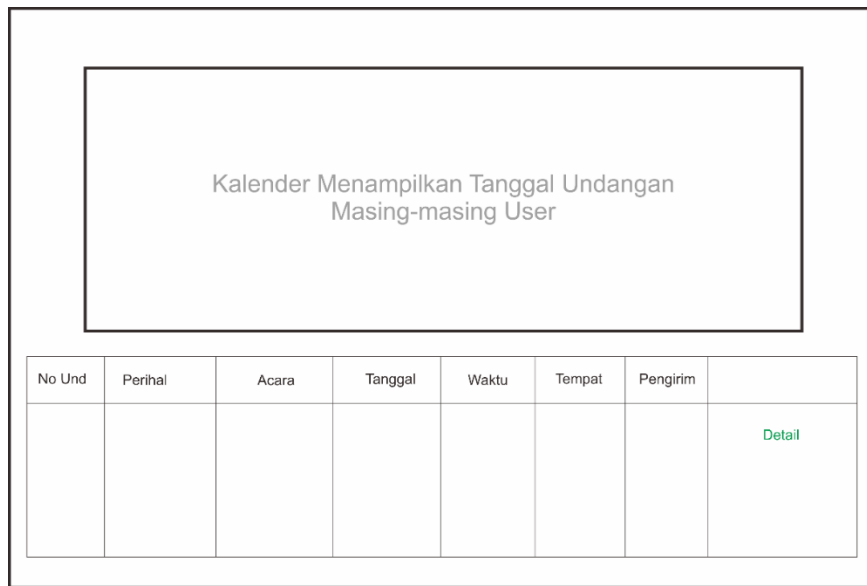


Gambar 3.16 Perancangan Halaman Undangan User

Pada gambar 3.16 terdapat panel-panel untuk menampilkan data undangan masing-masing *user* dari *database* dan setiap panel terdapat *link* detail yang akan menuju ke halaman detail undangan *user*. 1 *button* yang digunakan untuk menuju ke halaman kalender undangan *user*. Pada halaman ini juga terdapat menu yang dapat digunakan untuk mengubah profil *user* dan untuk *logout*.

9. *Interface* Halaman Kalender Undangan *User*

Berikut ini merupakan perancangan *interface* dari halaman detail undangan *user* yang dapat dilihat pada gambar 3.17.



Kalender Menampilkan Tanggal Undangan Masing-masing User

No Und	Perihal	Acara	Tanggal	Waktu	Tempat	Pengirim	
							Detail

Gambar 3.17 Perancangan Kalender Undangan User

Pada gambar 3.17 terdapat 1 kalender untuk menampilkan tanggal undangan masing-masing *user* dan 1 tabel untuk menampilkan informasi undangan berdasarkan tanggal yang dipilih oleh *user* dan setiap tabel terdapat *link* detail yang akan menuju ke halaman detail undangan *user*.

10. *Interface* Halaman Detail Undangan *User*

Berikut ini merupakan perancangan *interface* dari halaman detail undangan *user* yang dapat dilihat pada gambar 3.16.

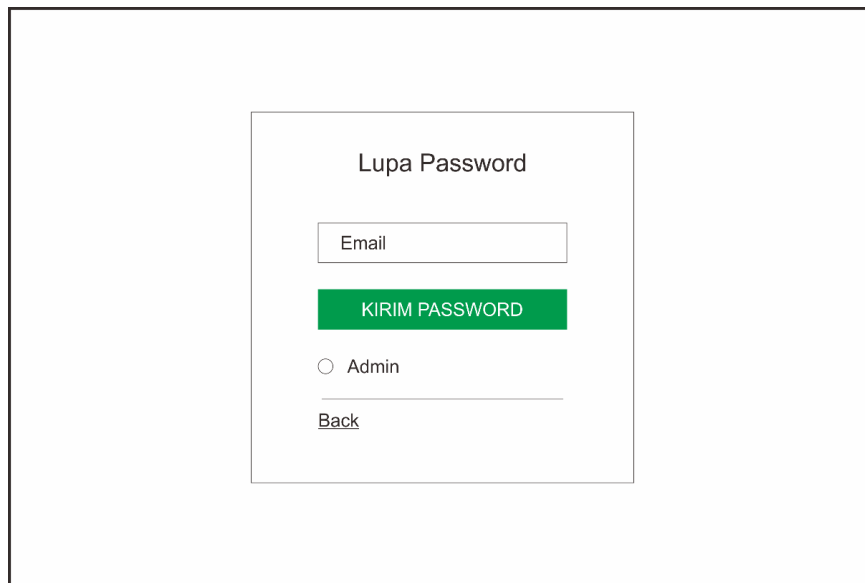


Gambar 3.18 Perancangan Halaman Detail Undangan User

Pada gambar 3.18 terdapat 1 panel untuk menampilkan data undangan masing-masing *user* berbentuk surat dan 1 *button* untuk mengunduh undangan ke bentuk *file* pdf.

11. *Interface* Halaman Lupa *Password*

Berikut ini merupakan perancangan *interface* dari halaman lupa *password* yang dapat dilihat pada gambar 3.19.



Lupa Password

Email

KIRIM PASSWORD

Admin

[Back](#)

Gambar 3.19 Perancangan Halaman Lupa Password

Pada gambar 3.19 terdapat 1 *textbox* untuk menginputkan *email*, 1 *button* untuk mengirim *password user* melalui *email user* serta 1 *radio button* untuk membedakan antara *admin* atau *user*, dan 1 *link* untuk kembali ke halaman *login*.

3.5 Metode Pengujian

Metode pengujian yang digunakan pada aplikasi ini adalah Metode *Black Box Testing* yaitu metode *testing* yang berfokus pada spesifikasi fungsional dari sebuah perangkat lunak. Pusat perhatian dalam pengujian Sistem Manajemen Arsip Undangan ada beberapa hal sebagai berikut :

1. Halaman *Login Admin* dan *Login User*
 - a. Jika *email* atau *password* salah, maka sistem akan menampilkan pesan *error*.
 - b. Jika *email* dan *password* benar, maka sistem akan masuk ke halaman *user*.
 - c. Jika *email* dan *password* benar dan *radio button admin* dipilih, maka sistem akan masuk ke halaman *admin*.

2. Halaman *Register User*
 - a. Aplikasi dapat menyimpan data *user* sesuai dengan kolom yang diisi oleh *user*.
 - b. Jika *email* yang diisi oleh *user* sudah ada pada *database*, maka sistem tidak akan menyimpan data dan akan menampilkan pesan *error*.
3. Halaman Kirim Undangan
 - a. Aplikasi dapat mengirim data undangan sesuai dengan alamat *email user* yang dipilih.
 - b. Aplikasi dapat menyimpan data undangan sesuai dengan kolom yang diisi oleh *admin*.
4. Halaman Undangan
 - a. Aplikasi dapat menampilkan data undangan dalam bentuk tabel.
 - b. Aplikasi dapat mengubah data undangan untuk disimpan atau untuk dikirim kembali kepada *user* melalui *email*.
 - c. Aplikasi dapat menghapus data undangan.
 - d. Aplikasi dapat mencari data undangan berdasarkan perihal undangan.
5. Halaman *User*
 - a. Aplikasi dapat menampilkan data *user* dalam bentuk tabel.
 - b. Aplikasi dapat mengubah data *user*.
 - c. Aplikasi dapat menghapus data *user*.
 - d. Aplikasi dapat mencari data *user* berdasarkan nama *user*.
6. Halaman *Admin*
 - a. Aplikasi dapat menyimpan data *admin*.
 - b. Aplikasi dapat menampilkan data *admin* dalam bentuk tabel.
 - c. Aplikasi dapat mengubah data *admin*.
 - d. Aplikasi dapat menghapus data *admin*.
 - e. Aplikasi dapat mencari data *user* berdasarkan nama *admin*.

7. Halaman *Log Undangan*

- a. Aplikasi dapat menampilkan data undangan yang telah terkirim ke *user* dalam bentuk tabel.
- b. Aplikasi dapat mencari data undangan berdasarkan nama *user*.

8. Halaman Undangan *User*

- a. Aplikasi dapat menampilkan data undangan masing-masing *user*.
- b. Aplikasi dapat mengubah profil dari *user*.

9. Halaman Kalender Undangan *User*

- a. Aplikasi dapat menampilkan jadwal undangan *user* dalam bentuk kalender.
- b. Aplikasi dapat menampilkan data undangan masing-masing *user* pada sebuah tabel berdasarkan tanggal yang dipilih pada kalender.

10. Halaman Detail Undangan *User*

- a. Aplikasi dapat menampilkan detail undangan *user* berbentuk surat.
- b. Aplikasi dapat mengunduh data undangan *user* ke bentuk *file pdf*.

11. Halaman Lupa *Password*

- a. Aplikasi dapat *mengirim* password sesuai *email* masing-masing *user* melalui pesan *email*.