

BAB V

IMPLEMENTASI

Implementasi perancangan basis data simlatte dilakukan sesuai dengan *entity relationship diagram*. Implementasi perancangan basis data simlatte menggunakan *Data Definition Language* (DDL). DDL adalah struktur basis data yang menggambarkan desain basis data secara keseluruhan.

5.1 Implementasi perancangan basis data

DDL yang digunakan untuk membentuk basis data db_simlatte adalah sebagai berikut :

```
create database db_simlatte;
```

5.2.1 DDL Untuk Membuat Tabel d_assign

DDL yang digunakan untuk membentuk tabel d_assign adalah sebagai berikut:

```
CREATE TABLE IF NOT EXISTS `d_assign` (  
  `id_assign` int(10) unsigned NOT NULL AUTO_INCREMENT,  
  `id_kurikulum` int(10) unsigned DEFAULT '0',  
  `id_konsentrasi` int(10) unsigned DEFAULT '0',  
  `id_lab` int(10) unsigned DEFAULT '0',  
  `id_praktikum` int(10) unsigned DEFAULT '0',  
  `id_periode` int(10) unsigned DEFAULT '0',  
  `id_user` int(10) unsigned DEFAULT '0',  
  `id_auth_level` int(10) unsigned DEFAULT '0',  
  `flag` enum('Outstanding','Accept','Reject','Delete') DEFAULT  
  'Outstanding',  
  PRIMARY KEY (`id_assign`)  
) ENGINE=MyISAM AUTO_INCREMENT=25 DEFAULT CHARSET=utf8;
```

5.2.2 DDL untuk Membuat Tabel d_assign_lab

DDL yang digunakan untuk membentuk tabel d_assign_lab adalah sebagai berikut:

```
CREATE TABLE IF NOT EXISTS `d_assign_lab` (  
  `id_assign_lab` int(10) unsigned NOT NULL AUTO_INCREMENT,  
  `id_kurikulum` int(10) unsigned DEFAULT '0',  
  `id_lab` int(10) unsigned DEFAULT '0',  
  `id_praktikum` int(10) unsigned DEFAULT '0',  
  PRIMARY KEY (`id_assign_lab`)  
) ENGINE=MyISAM AUTO_INCREMENT=11 DEFAULT CHARSET=utf8;
```

5.2.3 DDL untuk Membuat Tabel `d_auth_menu`

DDL yang digunakan untuk membentuk tabel `d_auth_menu` adalah sebagai berikut:

```
CREATE TABLE IF NOT EXISTS `d_auth_menu` (  
  `id_auth_menu` int(10) unsigned NOT NULL AUTO_INCREMENT,  
  `id_auth_level` int(10) unsigned DEFAULT NULL,  
  `id_menu` int(10) unsigned DEFAULT NULL,  
  PRIMARY KEY (`id_auth_menu`),  
  KEY `d_auth_menu_FKIndex1` (`id_auth_level`),  
  KEY `d_auth_menu_FKIndex2` (`id_menu`)  
) ENGINE=MyISAM AUTO_INCREMENT=159 DEFAULT CHARSET=latin1;
```

5.2.4 DDL untuk Membuat Tabel `d_auth_user`

DDL yang digunakan untuk membentuk tabel `d_auth_user` adalah sebagai berikut :

```
CREATE TABLE IF NOT EXISTS `d_auth_user` (  
  `id_auth_user` bigint(20) unsigned NOT NULL AUTO_INCREMENT,  
  `id_user` int(10) unsigned DEFAULT NULL,  
  `id_auth_level` int(10) unsigned DEFAULT NULL,  
  PRIMARY KEY (`id_auth_user`),  
  KEY `d_auth_user_FKIndex1` (`id_user`),  
  KEY `d_auth_user_FKIndex2` (`id_auth_level`)  
) ENGINE=MyISAM AUTO_INCREMENT=278 DEFAULT CHARSET=latin1;
```

5.2.5 DDL untuk Membuat Tabel `d_bab`

DDL yang digunakan untuk membentuk tabel `d_bab` adalah sebagai berikut :

```
CREATE TABLE IF NOT EXISTS `d_bab` (  
  `id_bab` int(10) unsigned NOT NULL AUTO_INCREMENT,  
  `id_periode` int(10) unsigned DEFAULT '0',  
  `label` varchar(255) DEFAULT '0',  
  `deskripsi` varchar(255) DEFAULT '0',  
  PRIMARY KEY (`id_bab`)  
) ENGINE=MyISAM DEFAULT CHARSET=utf8;
```

5.2.6 DDL untuk Membuat Tabel `d_daftar_postest`

DDL yang digunakan untuk membentuk tabel `d_daftar_postest` adalah sebagai berikut :

```
CREATE TABLE IF NOT EXISTS `d_daftar_postest` (  
  `id_daftar_postest` int(10) unsigned NOT NULL AUTO_INCREMENT,  
  `id_daftar_praktikan` int(10) unsigned NOT NULL DEFAULT '0',  
  `id_dosen` int(10) unsigned NOT NULL DEFAULT '0',  
  `tgl_postest` datetime NOT NULL,  
  `keterangan` text NOT NULL,  
  PRIMARY KEY (`id_daftar_postest`)  
) ENGINE=MyISAM DEFAULT CHARSET=utf8;
```

5.2.7 DDL untuk Membuat Tabel `d_daftar_praktikan`

DDL yang digunakan untuk membentuk tabel `d_daftar_praktikan` adalah sebagai berikut :

```
CREATE TABLE IF NOT EXISTS `d_daftar_praktikan` (  
  `id_daftar_praktikan` mediumint(8) unsigned NOT NULL  
  AUTO_INCREMENT,  
  `id_periode` int(10) unsigned NOT NULL,  
  `id_user` int(10) unsigned DEFAULT NULL,  
  `flag` enum('Draft','Accept','Complete','Delete') DEFAULT 'Draft',  
  PRIMARY KEY (`id_daftar_praktikan`)  
) ENGINE=MyISAM DEFAULT CHARSET=utf8;
```

5.2.8 DDL untuk Membuat Tabel `d_group_praktikum`

DDL yang digunakan untuk membentuk tabel `d_group_praktikum` adalah sebagai berikut :

```
CREATE TABLE IF NOT EXISTS `d_group_praktikum` (  
  `id_d_group_praktikum` int(10) unsigned NOT NULL AUTO_INCREMENT,  
  `id_kurikulum` int(10) unsigned NOT NULL DEFAULT '0',  
  `id_group_praktikum` int(10) unsigned NOT NULL DEFAULT '0',  
  `id_konsentrasi` int(10) unsigned NOT NULL DEFAULT '0',  
  `id_praktikum` int(10) unsigned NOT NULL DEFAULT '0',  
  PRIMARY KEY (`id_d_group_praktikum`)  
) ENGINE=MyISAM AUTO_INCREMENT=62 DEFAULT CHARSET=utf8;
```

5.2.9 DDL untuk Membuat Tabel `d_jadwal_kelompok`

DDL yang digunakan untuk membentuk tabel `d_jadwal_kelompok` adalah sebagai berikut :

```
CREATE TABLE IF NOT EXISTS `d_jadwal_kelompok` (  
  `id_jadwal_kelompok` int(10) unsigned NOT NULL AUTO_INCREMENT,  
  `id_bab` int(10) unsigned NOT NULL DEFAULT '0',  
  `start_date` timestamp NULL DEFAULT CURRENT_TIMESTAMP ON UPDATE  
CURRENT_TIMESTAMP,  
  `stop_date` timestamp NULL DEFAULT NULL,  
  `jumlah_max` tinyint(4) DEFAULT NULL,  
  PRIMARY KEY (`id_jadwal_kelompok`)  
) ENGINE=MyISAM DEFAULT CHARSET=utf8;
```

5.2.10 DDL untuk Membuat Tabel `d_kelompok_praktikum`

DDL yang digunakan untuk membentuk tabel `d_kelompok_praktikum` adalah sebagai berikut :

```
CREATE TABLE IF NOT EXISTS `d_kelompok_praktikum` (  
  `id_kelompok_praktikum` bigint(20) unsigned NOT NULL  
  AUTO_INCREMENT,  
  `id_jadwal_kelompok` int(10) unsigned NOT NULL,  
  `id_daftar_praktikum` int(10) unsigned NOT NULL,  
  PRIMARY KEY (`id_kelompok_praktikum`)  
) ENGINE=MyISAM DEFAULT CHARSET=utf8;
```

5.2.11 DDL untuk Membuat Tabel `d_konsentrasi_praktikum`

DDL yang digunakan untuk membentuk tabel `d_konsentrasi_praktikum` adalah sebagai berikut :

```
CREATE TABLE IF NOT EXISTS `d_konsentrasi_praktikum` (  
  `id_konsentrasi_praktikum` int(10) unsigned NOT NULL  
  AUTO_INCREMENT,  
  `id_kurikulum` int(10) unsigned DEFAULT '0',  
  `id_konsentrasi` int(10) unsigned DEFAULT '0',  
  `id_praktikum` int(10) unsigned DEFAULT '0',  
  PRIMARY KEY (`id_konsentrasi_praktikum`)  
) ENGINE=MyISAM DEFAULT CHARSET=utf8;
```

5.2.12 DDL untuk Membuat Tabel `d_news`

DDL yang digunakan untuk membentuk tabel `d_news` adalah sebagai berikut :

```

CREATE TABLE IF NOT EXISTS `d_news` (
  `id_news` int(10) unsigned NOT NULL AUTO_INCREMENT,
  `id_user` int(10) unsigned NOT NULL DEFAULT '0',
  `title_news` varchar(255) NOT NULL,
  `date_news` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP ON UPDATE
CURRENT_TIMESTAMP,
  `short_news` tinytext NOT NULL,
  `full_news` text NOT NULL,
  `img_path` varchar(255) DEFAULT NULL,
  `id_lab` int(10) unsigned DEFAULT NULL,
  `id_praktikum` int(10) unsigned DEFAULT NULL,
  PRIMARY KEY (`id_news`)
) ENGINE=MyISAM AUTO_INCREMENT=9 DEFAULT CHARSET=utf8;

```

5.2.13 DDL untuk Membuat Tabel `d_nilai`

DDL yang digunakan untuk membentuk tabel `d_nilai` adalah sebagai berikut :

```

CREATE TABLE IF NOT EXISTS `d_nilai` (
  `id_nilai` bigint(20) unsigned NOT NULL AUTO_INCREMENT,
  `id_daftar_praktikan` int(10) unsigned NOT NULL DEFAULT '0',
  `nilai_praktikum` decimal(10,0) unsigned NOT NULL DEFAULT '0'
COMMENT 'nilai praktikum by Assisten',
  `id_assisten` int(10) unsigned NOT NULL DEFAULT '0' COMMENT 'id
Assisten',
  `nilai_dosen` decimal(10,0) unsigned NOT NULL DEFAULT '0' COMMENT
'nilai praktikum by Dosen',
  `id_dosen` int(10) unsigned NOT NULL DEFAULT '0' COMMENT 'id
dosen',
  PRIMARY KEY (`id_nilai`)
) ENGINE=MyISAM DEFAULT CHARSET=utf8;

```

5.2.14 DDL untuk Membuat Tabel `d_paket_user`

DDL yang digunakan untuk membentuk tabel `d_paket_user` adalah sebagai berikut:

```
CREATE TABLE IF NOT EXISTS `d_paket_user` (  
  `id_paket_user` int(10) unsigned NOT NULL AUTO_INCREMENT,  
  `id_konsentrasi` int(10) unsigned DEFAULT '0',  
  `id_user` int(10) unsigned NOT NULL DEFAULT '0',  
  PRIMARY KEY (`id_paket_user`)  
) ENGINE=MyISAM AUTO_INCREMENT=60 DEFAULT CHARSET=utf8;
```

5.2.15 DDL untuk Membuat Tabel `d_periode`

DDL yang digunakan untuk membentuk tabel `d_periode` adalah sebagai berikut :

```
CREATE TABLE IF NOT EXISTS `d_periode` (  
  `id_periode` int(10) unsigned NOT NULL AUTO_INCREMENT,  
  `id_kurikulum` int(10) unsigned DEFAULT NULL,  
  `id_praktikum` int(10) unsigned DEFAULT NULL,  
  `label` varchar(255) DEFAULT NULL,  
  `tgl_start` timestamp NULL DEFAULT CURRENT_TIMESTAMP ON UPDATE  
CURRENT_TIMESTAMP,  
  `tgl_stop` timestamp NULL DEFAULT NULL,  
  `kuota` tinyint(3) unsigned DEFAULT NULL,  
  PRIMARY KEY (`id_periode`),  
  KEY `tgl_start` (`tgl_start`),  
  KEY `tgl_stop` (`tgl_stop`)  
) ENGINE=MyISAM AUTO_INCREMENT=3 DEFAULT CHARSET=utf8;
```

5.2.16 DDL untuk Membuat Tabel `m_auth_level`

DDL yang digunakan untuk membentuk tabel `m_auth_level` adalah sebagai berikut:

```
CREATE TABLE IF NOT EXISTS `m_auth_level` (  
  `id_auth_level` int(10) unsigned NOT NULL AUTO_INCREMENT,  
  `label` varchar(255) DEFAULT NULL,  
  `description` text,  
  PRIMARY KEY (`id_auth_level`)  
) ENGINE=MyISAM AUTO_INCREMENT=103 DEFAULT CHARSET=latin1;
```

5.2.17 DDL untuk Membuat Tabel m_config

DDL yang digunakan untuk membentuk tabel m_config adalah sebagai berikut :

```
CREATE TABLE IF NOT EXISTS `m_config` (  
  `config_var` text,  
  `config_val` text  
) ENGINE=MyISAM DEFAULT CHARSET=latin1;
```

5.2.18 DDL untuk Membuat Tabel m_group_praktikum

DDL yang digunakan untuk membentuk tabel m_group_praktikum adalah sebagai berikut :

```
CREATE TABLE IF NOT EXISTS `m_group_praktikum` (  
  `id_group_praktikum` int(10) unsigned NOT NULL AUTO_INCREMENT,  
  `kode_mk` varchar(255) NOT NULL,  
  `label` varchar(255) NOT NULL,  
  `sks` tinyint(3) unsigned NOT NULL DEFAULT '0',  
  `deskripsi` text,  
  PRIMARY KEY (`id_group_praktikum`)  
) ENGINE=MyISAM AUTO_INCREMENT=15 DEFAULT CHARSET=utf8;
```

5.2.19 DDL untuk Membuat Tabel m_konsentrasi

DDL yang digunakan untuk membentuk tabel m_konsentrasi adalah sebagai berikut :


```
CREATE TABLE IF NOT EXISTS `m_konsentrasi` (  
  `id_konsentrasi` int(10) unsigned NOT NULL AUTO_INCREMENT,  
  `label` varchar(255) COLLATE latin1_general_ci DEFAULT NULL,  
  `kode` varchar(1) COLLATE latin1_general_ci DEFAULT NULL,  
  PRIMARY KEY (`id_konsentrasi`)  
) ENGINE=MyISAM AUTO_INCREMENT=7 DEFAULT CHARSET=latin1  
COLLATE=latin1_general_ci;
```

5.2.20 DDL untuk Membuat Tabel `m_kurikulum`

DDL yang digunakan untuk membentuk tabel `m_kurikulum` adalah sebagai berikut :

```
CREATE TABLE IF NOT EXISTS `m_kurikulum` (  
  `id_kurikulum` int(10) unsigned NOT NULL AUTO_INCREMENT,  
  `label` varchar(255) DEFAULT 'Ganjil / ',  
  PRIMARY KEY (`id_kurikulum`)  
) ENGINE=MyISAM AUTO_INCREMENT=6 DEFAULT CHARSET=utf8;
```

5.2.21 DDL untuk Membuat Tabel `m_lab`

DDL yang digunakan untuk membentuk tabel `m_lab` adalah sebagai berikut :

```
CREATE TABLE IF NOT EXISTS `m_lab` (  
  `id_lab` int(10) unsigned NOT NULL AUTO_INCREMENT,  
  `label` varchar(255) COLLATE latin1_general_ci DEFAULT NULL,  
  PRIMARY KEY (`id_lab`)  
) ENGINE=MyISAM AUTO_INCREMENT=14 DEFAULT CHARSET=latin1  
COLLATE=latin1_general_ci;
```

5.2.22 DDL untuk Membuat Tabel `m_menu`

DDL yang digunakan untuk membentuk tabel `m_menu` adalah sebagai berikut :

```
CREATE TABLE IF NOT EXISTS `m_menu` (  
  `id_menu` int(10) unsigned NOT NULL AUTO_INCREMENT,  
  `id_parent` int(10) unsigned DEFAULT '0',
```

```

`label` varchar(255) DEFAULT NULL,
`path` varchar(255) DEFAULT NULL,
`description` text,
PRIMARY KEY (`id_menu`)
) ENGINE=MyISAM AUTO_INCREMENT=48 DEFAULT CHARSET=latin1;

```

5.2.23 DDL untuk membuat tabel `m_praktikum`

DDL yang digunakan untuk membentuk tabel `m_menu` adalah sebagai berikut :

```

CREATE TABLE IF NOT EXISTS `m_praktikum` (
  `id_praktikum` int(10) unsigned NOT NULL AUTO_INCREMENT,
  `kode_praktikum` varchar(255) NOT NULL,
  `label` varchar(255) NOT NULL,
  `deskripsi` text NOT NULL,
  PRIMARY KEY (`id_praktikum`)
) ENGINE=MyISAM AUTO_INCREMENT=35 DEFAULT CHARSET=utf8;

```

5.2.24 DDL Untuk Membuat Tabel `m_user`

DDL yang digunakan untuk membentuk tabel `m_user` adalah sebagai berikut :

```

CREATE TABLE IF NOT EXISTS `m_user` (
  `id_user` int(10) unsigned NOT NULL AUTO_INCREMENT,
  `user_name` varchar(255) DEFAULT NULL,
  `user_passwd` varchar(255) DEFAULT NULL,
  `user_realname` varchar(255) DEFAULT NULL,
  `mhs` enum('mahasiswa','dosen','karyawan') NOT NULL DEFAULT
'mahasiswa',
  `nim_nip` varchar(255) DEFAULT NULL,
  `description` text,
  `flag` enum('Complete','Delete') DEFAULT 'Complete',
  PRIMARY KEY (`id_user`)
) ENGINE=MyISAM AUTO_INCREMENT=277 DEFAULT CHARSET=latin1;

```

Basis data `simlatte` yang telah diimplementasikan pada basis data MySQL ditunjukkan dalam Gambar 5.2.

Table	Action	Records	Type	Collation	Size
d_assign		21	MyISAM	utf8_general_ci	1.9 KB
d_assign_lab		21	MyISAM	utf8_general_ci	1.2 KB
d_auth_menu		21	MyISAM	latin1_swedish_ci	4.9 KB
d_auth_user		171	MyISAM	latin1_swedish_ci	24.6 KB
d_bab		21	MyISAM	utf8_general_ci	1.4 KB
d_daftar_posttest		9	MyISAM	utf8_general_ci	1.9 KB
d_daftar_praktikum		9	MyISAM	utf8_general_ci	1.9 KB
d_group_praktikum		46	MyISAM	utf8_general_ci	1.4 KB
d_jadwal_kelompok		5	MyISAM	utf8_general_ci	1.3 KB
d_jadwal_mhs		9	MyISAM	utf8_general_ci	1.9 KB
d_jadwal_praktikum		9	MyISAM	utf8_general_ci	1.9 KB
d_kelompok_praktikum		9	MyISAM	utf8_general_ci	1.9 KB
d_konsentrasi_praktikum		9	MyISAM	utf8_general_ci	1.9 KB
d_lab_praktikum		9	MyISAM	utf8_general_ci	1.9 KB
d_lab_group_praktikum		9	MyISAM	utf8_general_ci	1.9 KB
d_news		7	MyISAM	utf8_general_ci	4.3 KB
d_rilis		9	MyISAM	utf8_general_ci	1.9 KB
d_paket_user		19	MyISAM	utf8_general_ci	1.7 KB
d_periode		5	MyISAM	utf8_general_ci	4.1 KB
m_auth_level		21	MyISAM	latin1_swedish_ci	1.7 KB
m_cordig		21	MyISAM	latin1_swedish_ci	1.9 KB
m_group_praktikum		22	MyISAM	utf8_general_ci	1.4 KB
m_konsentrasi		5	MyISAM	latin1_swedish_ci	2.1 KB
m_kurikulum		4	MyISAM	utf8_general_ci	1.1 KB
m_lab		21	MyISAM	latin1_swedish_ci	1.1 KB
m_menu		41	MyISAM	latin1_swedish_ci	5.9 KB
m_praktikum		22	MyISAM	utf8_general_ci	4.6 KB
m_user		171	MyISAM	latin1_swedish_ci	21.7 KB
m_options		192	InnoDB	utf8_general_ci	1.6 KB

Gambar 5.1 Implementasi Basis Data simlatte pada MySQL
Sumber: [implementasi]

5.2 Implementasi perancangan sistem administrator SIMLATTE

Pada implementasi perangkat lunak aplikasi dilakukan implementasi perangkat lunak untuk login, perangkat lunak untuk administrasi admin *management*, administrasi *manage auth*, administrasi *manage menu*, administrasi *manage menu level*, administrasi *user*, administrasi *user level*, administrasi *assign* kajar, administrasi data *login user*.

5.2.1 Implementasi Perangkat Lunak Login SIMLATTE.

User yang akan menggunakan aplikasi pada Sistem Administrator Sistem Informasi Manajemen Laboratorium Teknik Elektro Universitas Brawijaya harus melalui proses otentifikasi *login*. Halaman *login* mempunyai dua kolom untuk *user name* dan *password*. Implementasi fasilitas *login* ditunjukkan dalam Gambar 5.2.

Gambar 5.2 Implementasi antarmuka fasilitas login
Sumber: [implementasi]

Proses login juga melakukan pengecekan terhadap *access level* dari seorang *user*. *User* dapat melakukan berbagai macam proses yang sesuai dengan *access level* yang dimiliki. Apabila proses *login* berhasil, *user* akan langsung masuk ke halaman utama SIMLATTE. Implementasi hasil *login* ditunjukkan dalam Gambar 5.3.



Gambar 5.3 Implementasi antarmuka hasil login
Sumber: [implementasi]

5.2.2 Implementasi perangkat lunak menu admin management.

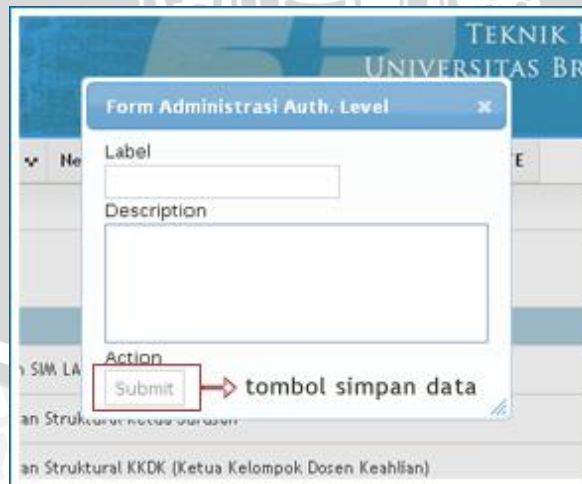
Setelah aktor admin berhasil *login*, aktor admin dapat memiliki fitur menu seperti : “*Manage Management*”, “*Assign Kujur*”, “*News*”. Di dalam menu *Menu “Manage Management”* memiliki *submenu* antara lain : “*Manage Auth*”, “*Manage Menu*”, “*Manage Menu Level*”, “*Manage User*”, “*Manage User Level*”. Apabila admin memilih *menu Admin*

Management – Manage Auth, maka sistem akan menampilkan halaman *Manage Authorisasi*. Implementasi halaman “Manage Authorisasi” ditunjukkan dalam gambar 5.4



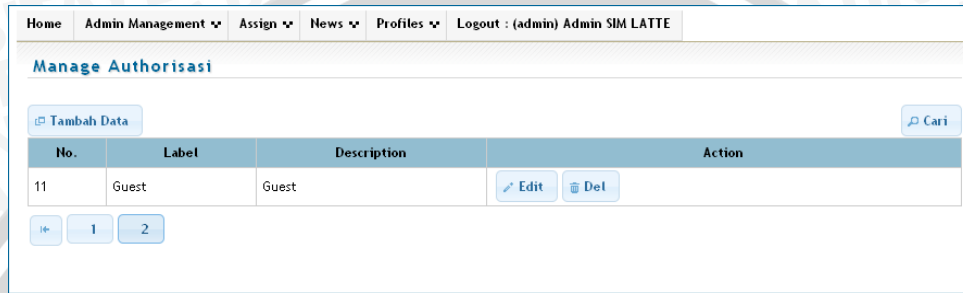
Gambar 5.4 Implementasi antarmuka halaman *manage Authorisasi*
Sumber: [implementasi]

Setelah berhasil masuk ke halaman “Manage Authorisasi”, jika aktor admin memilih tombol “tambah data”, maka sistem akan menampilkan *pop-up* seperti yang ditunjukkan pada gamabar 5.5.



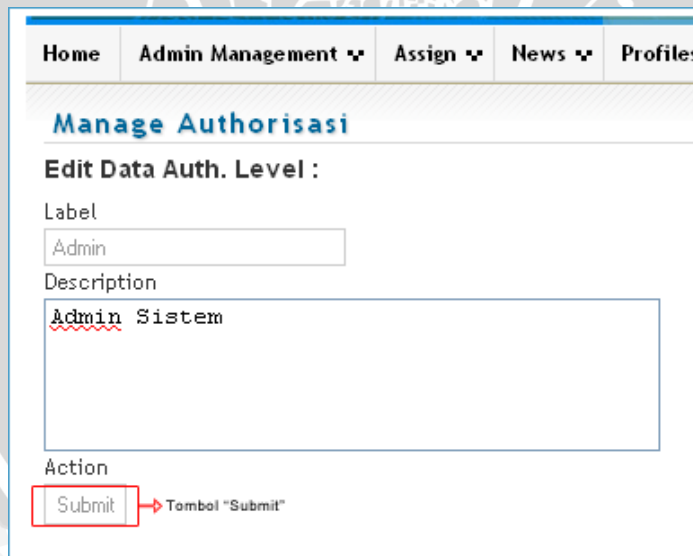
Gambar 5.5 Implementasi antarmuka *pop-up* “Form Administrasi Auth. Level”
Sumber: [implementasi]

Apabila aktor sudah mengisi *input form* data baru, aktor kemudian menekan tombol “*Submit*” untuk melakukan penambahan data. Apabila tidak terjadi kesalahan pada proses penambahan data, pada layar akan ditampilkan pesan yang menyatakan bahwa proses perubahan data sudah berhasil dan secara otomatis kembali ke halaman sebelumnya. Proses tersebut ditunjukkan pada gambar 5.6.



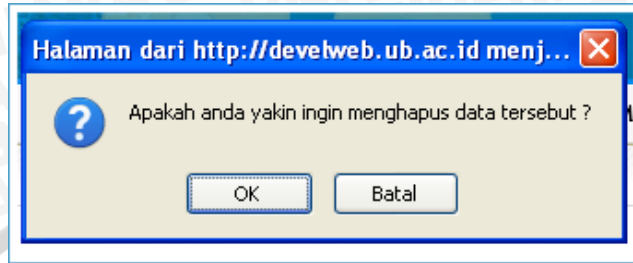
Gambar 5.6 Implementasi antarmuka penambahan data pada *manage autorisasi*
Sumber: [implementasi]

Jika aktor admin memilih tombol “*edit*” pada halaman “*Manage Authorisasi*” maka pada layar kemudian ditampilkan halaman yang berisi *form* yang akan diisi dengan data baru dari aktor admin, yang ditunjukkan pada gambar 5.7



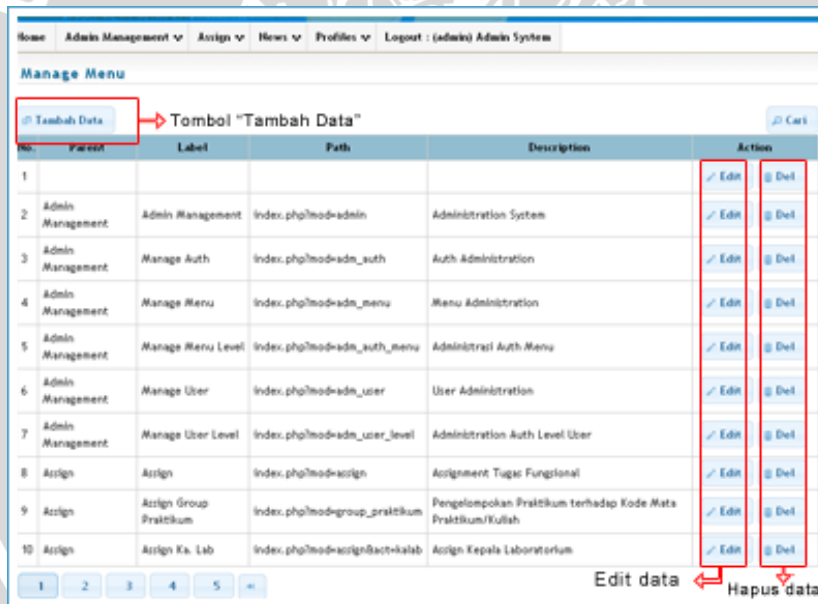
Gambar 5.7 Implementasi antarmuka halaman “*Edit Manage Authorisasi*”
Sumber: [implementasi]

Jika aktor kajar memilih tombol “del” pada halaman “Manage Authorisasi”, maka pada layar kemudian ditampilkan *pop-up* yang berisi peringatan, yang ditunjukkan pada gambar 5.8



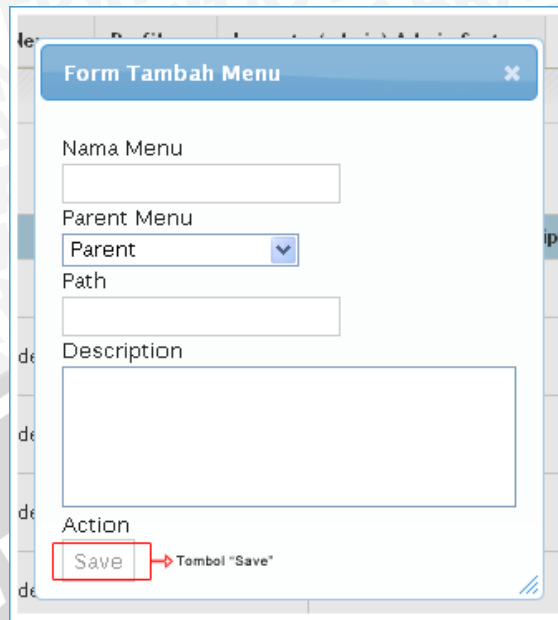
Gambar 5.8 Implementasi antarmuka *pop-up* peringatan hapus data
Sumber: [implementasi]

Apabila admin memilih menu *Admin Management – Manage Menu*, maka sistem akan menampilkan halaman *Manage Menu*. Untuk implementasi halaman “Manage Menu” ditunjukkan dalam gambar 5.9



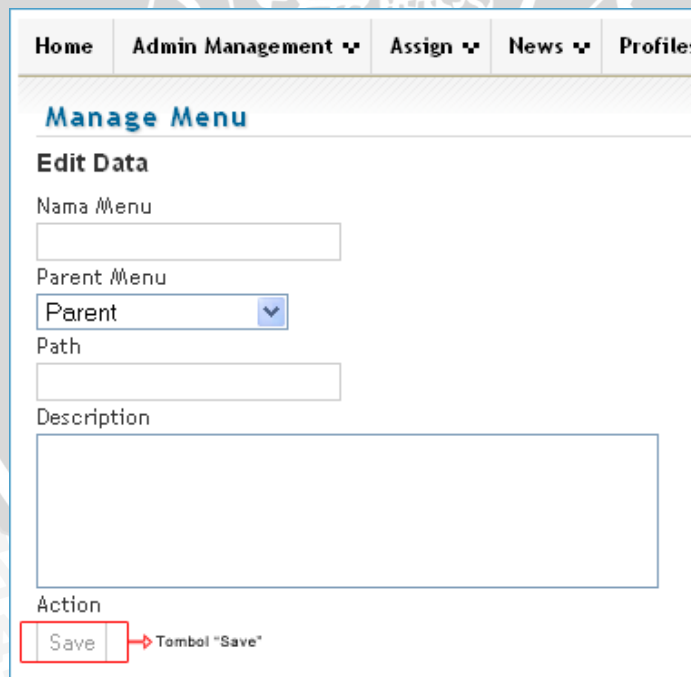
Gambar 5.9 Implementasi antarmuka halaman *Manage Menu*
Sumber: [implementasi]

Pada implementasi antarmuka “Manage Menu” terdapat tombol “Tambah Data”, “Edit Data” dan “Hapus Data”. Fungsi tombol “Tambah Data” adalah untuk menambah data baru pada halaman “Manage Menu”. Tampilan halaman untuk “Tambah Data” ditunjukkan pada gambar 5.10.



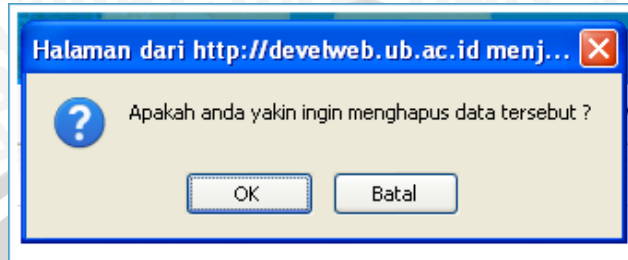
Gambar 5.10 Implementasi antarmuka *pop-up* tambah data, “*Form Manage Menu*”
Sumber: [implementasi]

Jika aktor admin memilih tombol “*edit*” pada halaman “*Manage Menu*” maka pada layar kemudian ditampilkan halaman yang berisi *form* yang akan diisi dengan data baru dari aktor admin, yang ditunjukkan pada gambar 5.11



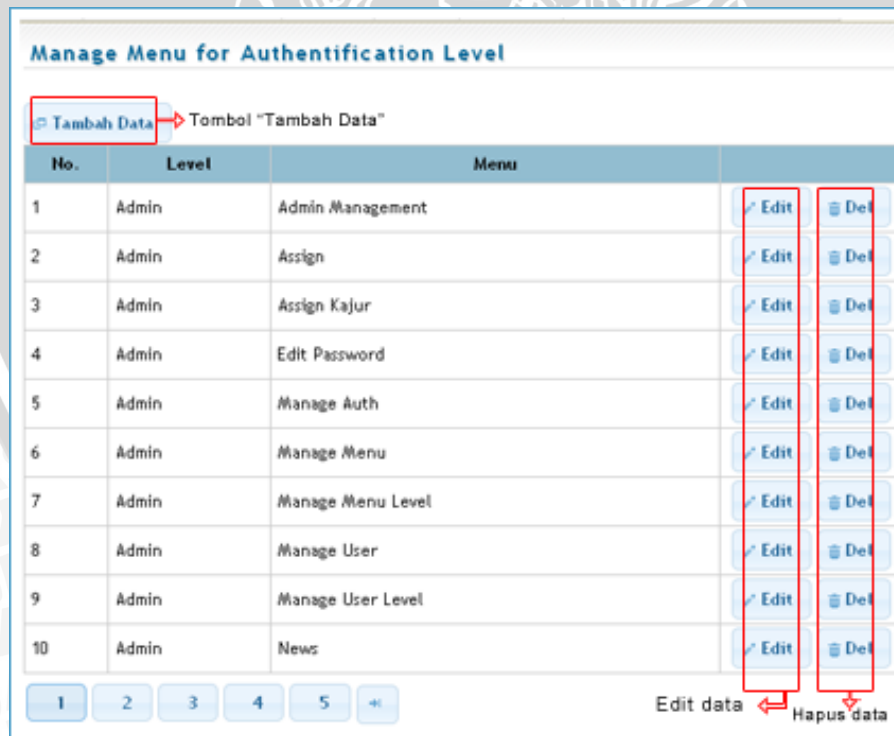
Gambar 5.11 Implementasi antarmuka halaman “*Edit Manage Menu*”
Sumber: [implementasi]

Jika aktor admin memilih tombol “del” pada halaman “Manage Menu”, maka pada layar kemudian ditampilkan *pop-up* yang berisi peringatan, yang ditunjukkan pada gambar 5.12



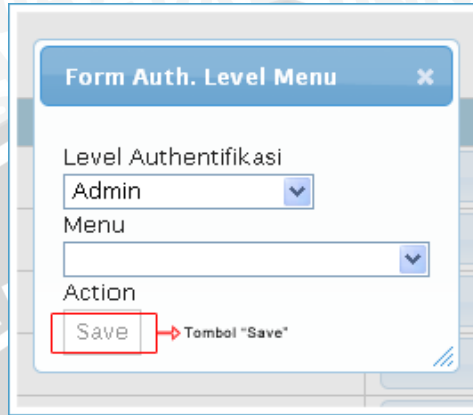
Gambar 5.12 Implementasi antarmuka *pop-up* peringatan hapus data
Sumber: [implementasi]

Apabila admin memilih menu *Admin Management – Manage Menu Level*, maka sistem akan menampilkan halaman “Manage Menu for Authentication Level”. Untuk implementasi halaman “Manage Menu for Authentication Level” ditunjukkan dalam gambar 5.13



Gambar 5.13 Implementasi antarmuka halaman *Manage Menu for Authentication Level*
Sumber: [implementasi]

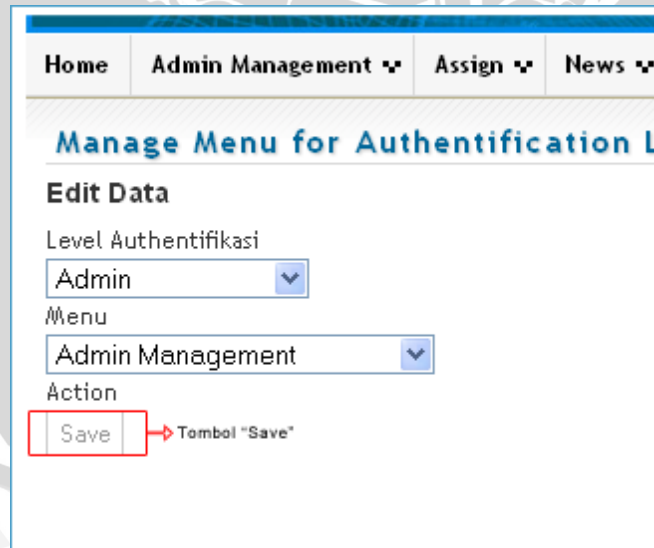
Pada implementasi antarmuka “*Manage Menu for Authentication Level*” terdapat tombol “Tambah Data”, “*Edit Data*” dan “Hapus Data”. Fungsi tombol “Tambah Data” adalah untuk menambah data baru pada halaman “*Manage Menu for Authentication Level*”. Tampilan halaman untuk “Tambah Data” ditunjukkan pada gambar 5.14.



Gambar 5.14 Implementasi antarmuka *pop-up* tambah data, “*Form Manage Menu for Authentication Level*”

Sumber: [implementasi]

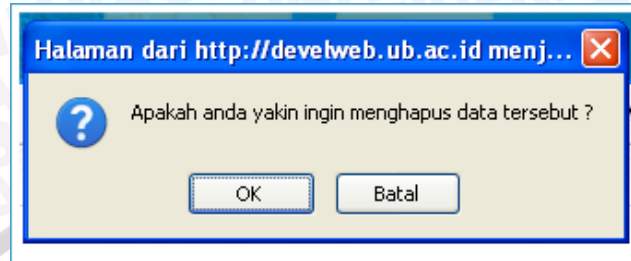
Jika aktor admin memilih tombol “*edit*” pada halaman “*Manage Menu for Authentication Level*” maka pada layar kemudian ditampilkan halaman yang berisi *form* yang akan diisi dengan data baru dari aktor admin, yang ditunjukkan pada gambar 5.15



Gambar 5.15 Implementasi antarmuka halaman “*Edit anage Menu for Authentication Level*”

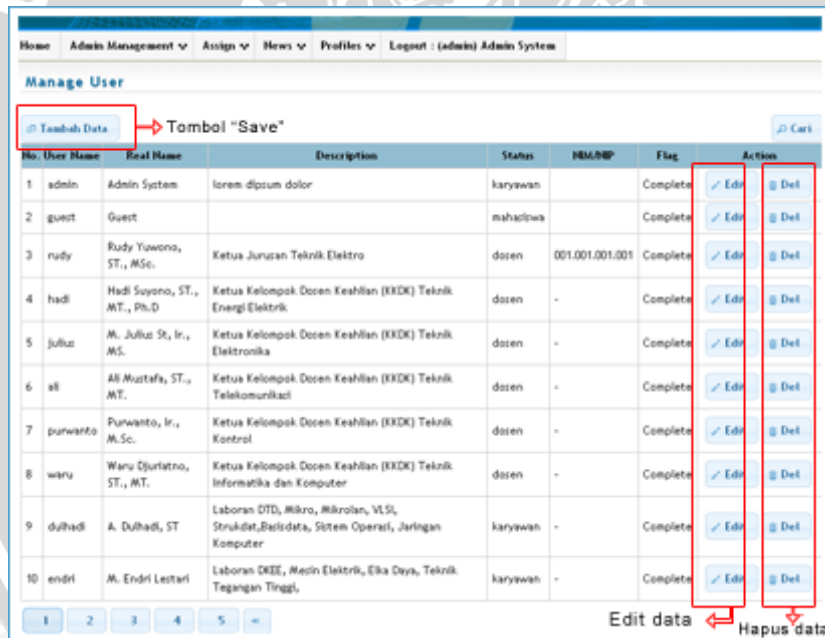
Sumber: [implementasi]

Jika aktor admin memilih tombol “del” pada halaman “*Manage Menu for Authentication Level*”, maka pada layar kemudian ditampilkan *pop-up* yang berisi peringatan, yang ditunjukkan pada gambar 5.16.



Gambar 5.16 Implementasi antarmuka *pop-up* peringatan hapus data
Sumber: [implementasi]

Apabila admin memilih menu *Admin Management – Manage User*, maka sistem akan menampilkan halaman “*Manage User*”. Untuk implementasi halaman “*Manage User*” ditunjukkan dalam gambar 5.17.



Gambar 5.17 Implementasi antarmuka halaman *Manage User*
Sumber: [implementasi]

Pada implementasi antarmuka “*Manage Menu User*” terdapat tombol “Tambah Data”, “*Edit Data*” dan “*Hapus Data*”. Fungsi tombol “Tambah Data” adalah untuk

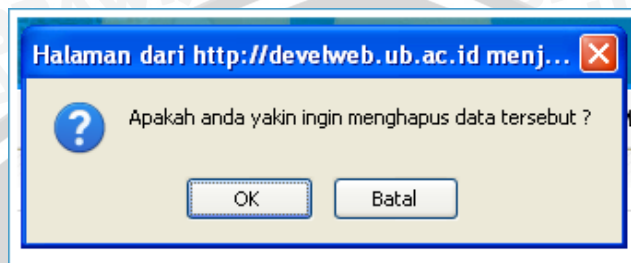
menambah data baru pada halaman “*Manage User*”. Tampilan halaman untuk “Tambah Data” ditunjukkan pada gambar 5.18.

Gambar 5.18 Implementasi antarmuka *pop-up* tambah data, “*Form Manage User*”
Sumber: [implementasi]

Jika aktor admin memilih tombol “*edit*” pada halaman “*Manage User*” maka pada layar kemudian ditampilkan halaman yang berisi *form* yang akan diisi dengan data baru dari aktor admin, yang ditunjukkan pada gambar 5.19.

Gambar 5.19 Implementasi antarmuka halaman “*Edit Manage User*”
Sumber: [implementasi]

Jika aktor admin memilih tombol “*del*” pada halaman “*Manage User*”, maka pada layar kemudian ditampilkan *pop-up* yang berisi peringatan, yang ditunjukkan pada gambar 5.20.



Gambar 5.20 Implementasi antarmuka *pop-up* peringatan hapus data
Sumber: [implementasi]

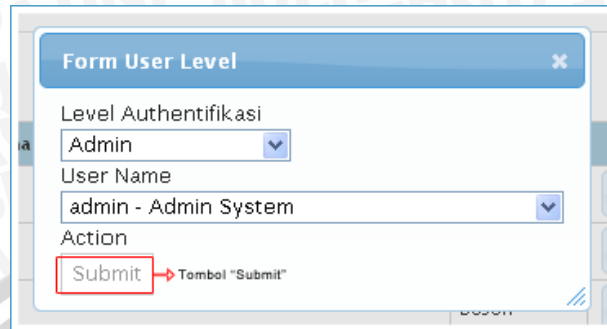
Apabila admin memilih *menu Admin Management – Manage User Level*, maka sistem akan menampilkan halaman “*Manage User Level*”. Untuk implementasi halaman “*Manage User Level*” ditunjukkan dalam gambar 5.21.

No.	(Username) Real Name	Level	Aksi	
1	(admin) Admin System	Admin	Edit	Del
2	(sawin) Ir. Muhammad Sawin, MT.	Dosen	Edit	Del
3	(arief) Ir. Arief Widlatmodjo	Dosen	Edit	Del
4	(tibyani) Tibyani, ST, MT.	Dosen	Edit	Del
5	(right) Ir. Sigit Kusmaryanto	Dosen	Edit	Del
6	(primantara) Ir. Primantara Hari Trinawan	Dosen	Edit	Del
7	(wijono) Ir. Wijono, MT.	Dosen	Edit	Del
8	(bambang) Ir. Bambang Slowojo, MT.	Dosen	Edit	Del
9	(sutrino) Ir. Sutrisno	Dosen	Edit	Del
10	(suprpto) Suprpto, ST, MT.	Dosen	Edit	Del

Gambar 5.21 Implementasi antarmuka halaman *Manage User Level*
Sumber: [implementasi]

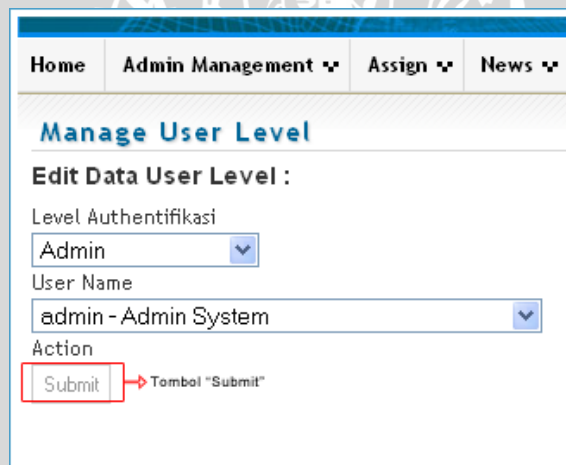
Pada implementasi antarmuka “*Manage User Level*” terdapat tombol “*Tambah Data*”, “*Edit Data*” dan “*Hapus Data*”. Fungsi tombol “*Tambah Data*” adalah untuk

menambah data baru pada halaman “*Manage User Level*”. Tampilan halaman untuk “Tambah Data” ditunjukkan pada gambar 5.22.



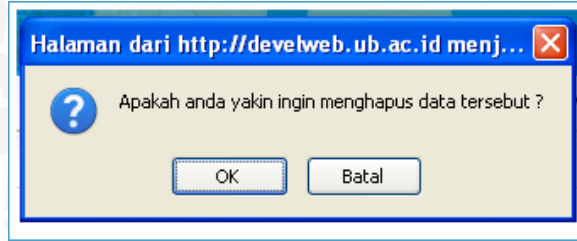
Gambar 5.22 Implementasi antarmuka *pop-up* tambah data, “*Form Manage User Level*”
Sumber: [implementasi]

Jika aktor admin memilih tombol “*edit*” pada halaman “*Manage User Level*” maka pada layar kemudian ditampilkan halaman yang berisi *form* yang akan diisi dengan data baru dari aktor admin, yang ditunjukkan pada gambar 5.23.



Gambar 5.23 Implementasi antarmuka halaman “*Edit Manage User Level*”
Sumber: [implementasi]

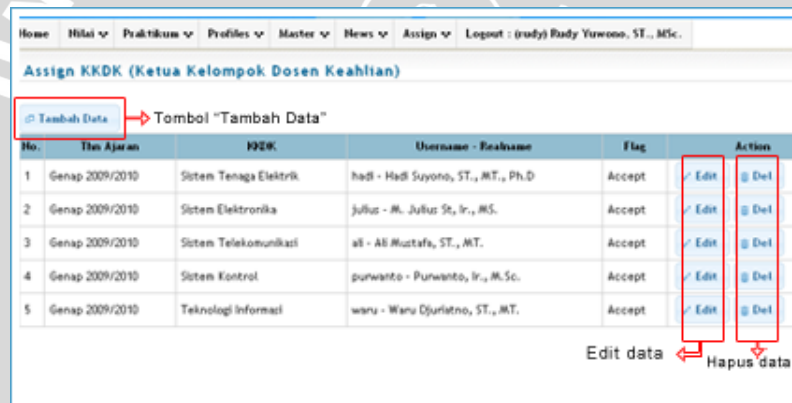
Jika aktor admin memilih tombol “*del*” pada halaman “*Manage User Level*”, maka pada layar kemudian ditampilkan *pop-up* yang berisi peringatan, yang ditunjukkan pada gambar 5.24.



Gambar 5. 24 Implementasi antarmuka *pop-up* peringatan hapus data
 Sumber: [implementasi]

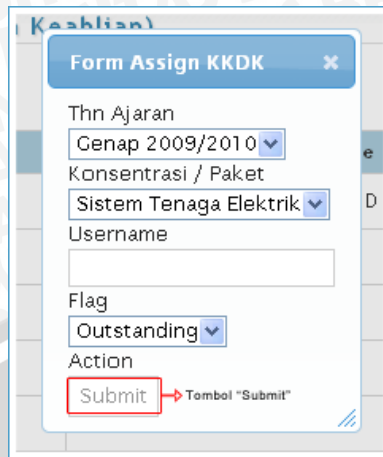
5.3 Implementasi perangkat lunak aktor kajar

Pada implementasi perangkat lunak dengan aktor kajar mempunyai fitur-fitur antara lain: *Assign KKDK*, *Assign Paket Dosen*, *News From Lab*, *Master Group Praktikum*, *Master Kurikulum*, *Master Konsentrasi*. Tampilan halaman untuk *Assign KKDK* ditunjukkan pada gambar 5.25.



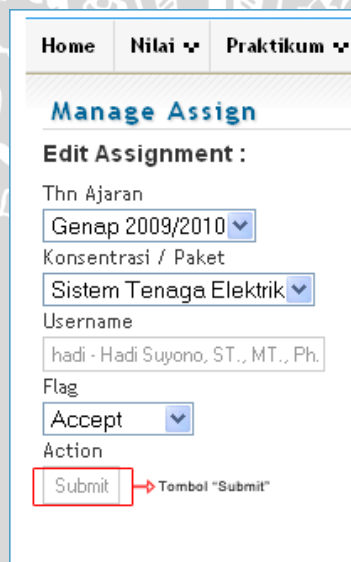
Gambar 5.25 Implementasi antarmuka halaman *Assign KKDK*
 Sumber: [implementasi]

Pada implementasi antarmuka “*Assign KKDK*” terdapat tombol “*Tambah Data*”, “*Edit Data*” dan “*Hapus Data*”. Fungsi tombol “*Tambah Data*” adalah untuk menambah data baru pada halaman “*Assign KKDK*”. Tampilan halaman untuk “*Tambah Data*” ditunjukkan pada gambar 5.26.



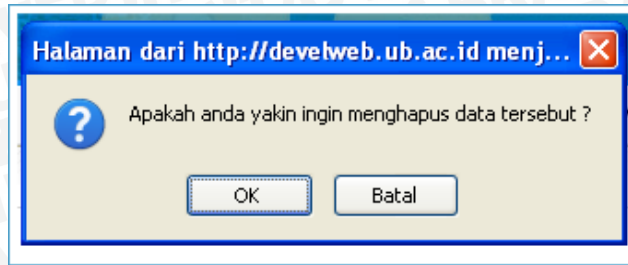
Gambar 5.26 Implementasi antarmuka *pop-up* tambah data, “Form Assign KKDK”
Sumber: [implementasi]

Jika aktor kajar memilih tombol “*edit*” pada halaman “Assign KKDK” maka pada layar kemudian ditampilkan halaman yang berisi *form* yang akan diisi dengan data baru dari aktor kajar, yang ditunjukkan pada gambar 5.27.



Gambar 5.27 Implementasi antarmuka halaman “Edit Assign KKDK”
Sumber: [implementasi]

Jika aktor kajar memilih tombol “*del*” pada halaman “Assign KKDK”, maka pada layar kemudian ditampilkan *pop-up* yang berisi peringatan, yang ditunjukkan pada gambar 5.28.



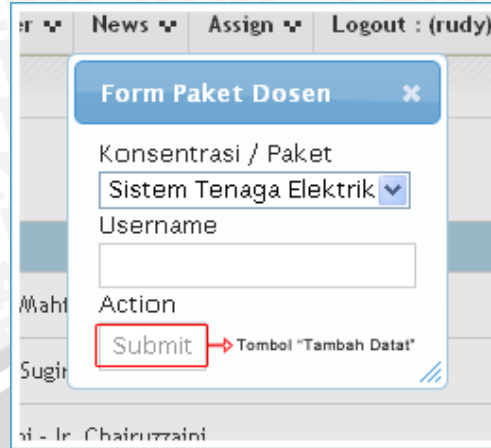
Gambar 5.28 Implementasi antarmuka *pop-up* peringatan hapus data
Sumber: [implementasi]

Pada menu “Assign” terdapat *submenu* ke-2, yaitu “Assign Paket Dosen”. Tampilan halaman untuk “Assign Paket Dosen” ditunjukkan pada gambar 5.29.

No.	Paket/Konsentrasi	Username - Realname	Action
1	Sistem Tenaga Elektrik	mahfudz - Mahfudz Shidiq, Ir., MS.	Edit Del
2	Sistem Tenaga Elektrik	sugiri - Ir. Sugiri	Edit Del
3	Sistem Tenaga Elektrik	chalruzzalni - Ir. Chalruzzalni	Edit Del
4	Sistem Tenaga Elektrik	unggul - Ir. Unggul Wibawa, M.Sc.	Edit Del
5	Sistem Tenaga Elektrik	harry - Dr.Ir. Harry Soekotjo Dachlan, MSc.	Edit Del
6	Sistem Tenaga Elektrik	budiono - Budiono Mlimal, Prof. Ir. MSEE, PhD	Edit Del
7	Sistem Tenaga Elektrik	dhofir - Moch. Dhofir, Drs., Ir., MT.	Edit Del
8	Sistem Tenaga Elektrik	hari - Hari Santoso, Ir., Ms.	Edit Del
9	Sistem Tenaga Elektrik	retnowati - Ir. Retnowati, MT	Edit Del
10	Sistem Tenaga Elektrik	hadi - Hadi Suyono, ST., MT., Ph.D	Edit Del

Gambar 5.29 Implementasi antarmuka halaman “Assign Paket Dosen”
Sumber: [implementasi]

Pada implementasi antarmuka “Assign Paket Dosen” terdapat tombol “Tambah Data”, “Edit Data” dan “Hapus Data”. Fungsi tombol “Tambah Data” adalah untuk menambah data baru pada halaman “Assign Paket Dosen”. Tampilan halaman untuk “Tambah Data” ditunjukkan pada gambar 5.30.



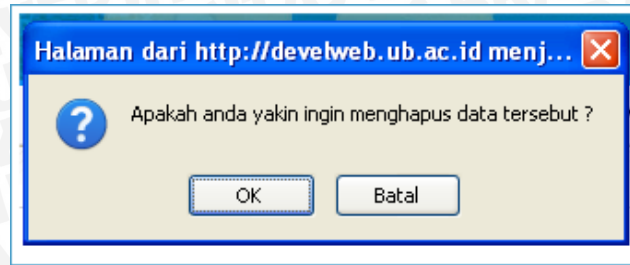
Gambar 5.30 Implementasi antarmuka *pop-up* tambah data, “Form Assign Paket Dosen”
Sumber: [implementasi]

Jika aktor kajar memilih tombol “*edit*” pada halaman “Assign Paket Dosen” maka pada layar kemudian ditampilkan halaman yang berisi *form* yang akan diisi dengan data baru dari aktor kajar, yang ditunjukkan pada gambar 5.31.



Gambar 5.31 Implementasi antarmuka halaman “Edit Assign Paket Dosen”
Sumber: [implementasi]

Jika aktor kajar memilih tombol “*del*” pada halaman “Assign Paket Dosen”, maka pada layar kemudian ditampilkan *pop-up* yang berisi peringatan, yang ditunjukkan pada gambar 5.32.



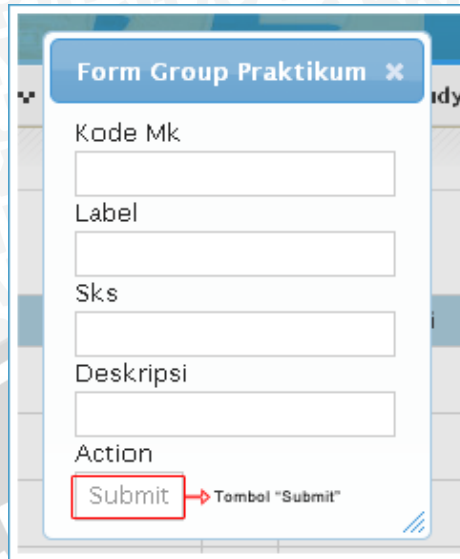
Gambar 5.32. Implementasi antarmuka *pop-up* peringatan hapus data
Sumber: [implementasi]

Pada menu “Master” terdapat 3 submenu, yaitu “Manage Group Praktikum”, “Master Kurikulum”, “Master Konsentrasi”. Tampilan halaman untuk “Manage Group Praktikum” ditunjukkan pada gambar 5.33.

No.	Kode MK	Label	Sks	Deskripsi	Action
1	TRE4138	Praktikum ABCDE-1	1		✓ Edit Del
2	TRE4243	Praktikum ABCDE-2	1		✓ Edit Del
3	TRE4139	Praktikum ABCDE-3	1		✓ Edit Del
4	TRE4242	Praktikum A-1	1		✓ Edit Del
5	TRE4137	Praktikum A-2	1		✓ Edit Del
6	TRE4244	Praktikum B-1	1		✓ Edit Del
7	TRE4140	Praktikum B-2	1		✓ Edit Del
8	TRE4245	Praktikum C-1	1		✓ Edit Del
9	TRE4141	Praktikum C-2	1		✓ Edit Del
10	TRE4246	Praktikum D-1	1		✓ Edit Del

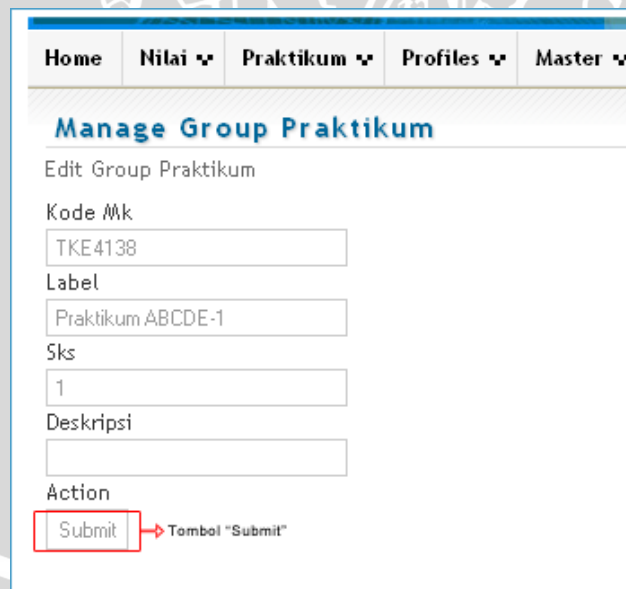
Gambar 5.33 Implementasi antarmuka halaman “Manage Group Praktikum”
Sumber: [implementasi]

Pada implementasi antarmuka “Manage Group Praktikum” terdapat tombol “Tambah Data”, “Edit Data” dan “Hapus Data”. Fungsi tombol “Tambah Data” adalah untuk menambah data baru pada halaman “Manage Group Praktikum”. Tampilan halaman untuk “Tambah Data” ditunjukkan pada gambar 5.34.



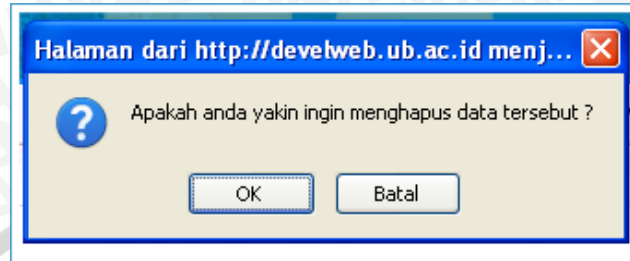
Gambar 5.34 Implementasi antarmuka *pop-up* tambah data, “Form Manage Group Praktikum”
Sumber: [implementasi]

Jika aktor kajar memilih tombol “*edit*” pada halaman “Manage Group Praktikum” maka pada layar kemudian ditampilkan halaman yang berisi *form* yang akan diisi dengan data baru dari aktor kajar, yang ditunjukkan pada gambar 5.35.



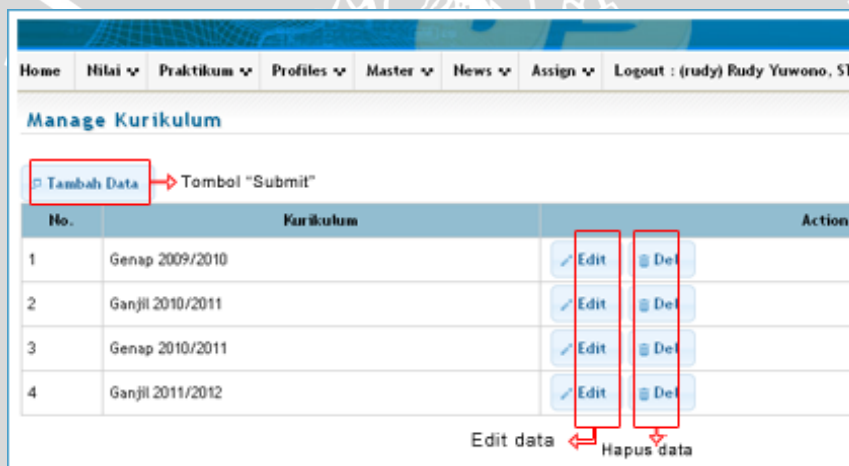
Gambar 5.35 Implementasi antarmuka halaman “Edit Manage Group Praktikum”
Sumber: [implementasi]

Jika aktor kajar memilih tombol “del” pada halaman “*Manage Group Praktikum*”, maka pada layar kemudian ditampilkan *pop-up* yang berisi peringatan, yang ditunjukkan pada gambar 5.36.



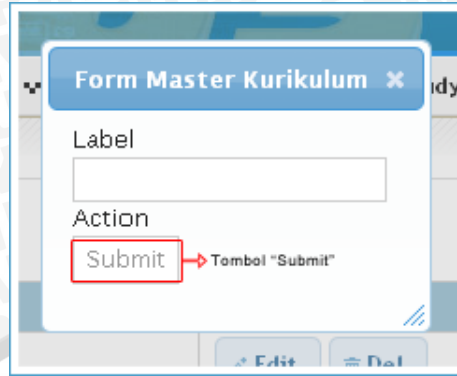
Gambar 5.36 Implementasi antarmuka *pop-up* peringatan hapus data
Sumber: [implementasi]

Untuk *submenu* yang kedua pada *menu* “*Master*” adalah “*Manage Kurikulum*”, tampilan halaman untuk “*Manage Kurikulum*” ditunjukkan pada gambar 5.37.



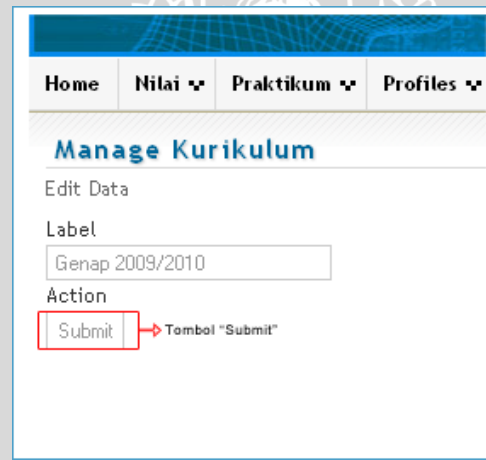
Gambar 5.37 Implementasi antarmuka halaman “*Manage Kurikulum*”
Sumber: [implementasi]

Pada implementasi antarmuka “*Manage Kurikulum*” terdapat tombol “*Tambah Data*”, “*Edit Data*” dan “*Hapus Data*”. Fungsi tombol “*Tambah Data*” adalah untuk menambah data baru pada halaman “*Manage Kurikulum*”. Tampilan halaman untuk “*Tambah Data*” ditunjukkan pada gambar 5.38.



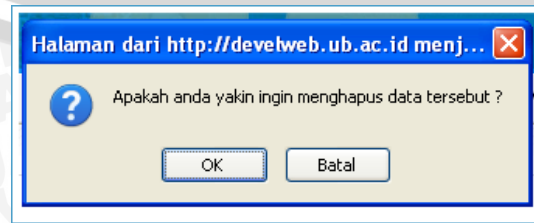
Gambar 5.38 Implementasi antarmuka *pop-up* tambah data, “Form Manage Kurikulum”
Sumber: [implementasi]

Jika aktor kajar memilih tombol “*edit*” pada halaman “*Manage Kurikulum*” maka pada layar kemudian ditampilkan halaman yang berisi *form* yang akan diisi dengan data baru dari aktor kajar, yang ditunjukkan pada gambar 5.39.



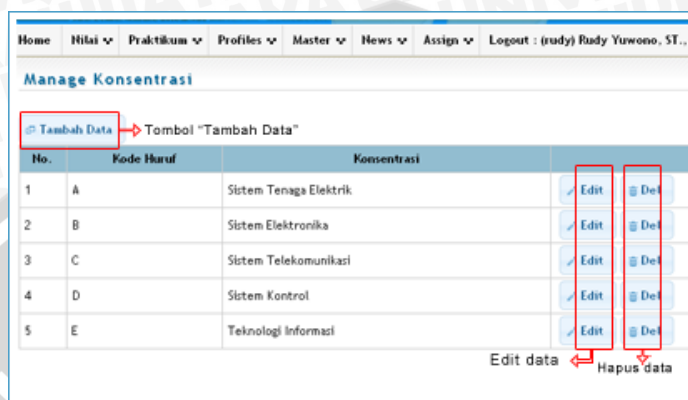
Gambar 5.39 Implementasi antarmuka halaman “*Edit Manage Kurikulum*”
Sumber: [implementasi]

Jika aktor kajar memilih tombol “*del*” pada halaman “*Manage Kurikulum*”, maka pada layar kemudian ditampilkan *pop-up* yang berisi peringatan, yang ditunjukkan pada gambar 5.40.



Gambar 5.40 Implementasi antarmuka *pop-up* peringatan hapus data
Sumber: [implementasi]

Untuk *submenu* yang ketiga pada *menu* “Master” adalah “Manage Konsentrasi”, tampilan halaman untuk “Manage Konsentrasi” ditunjukkan pada gambar 5.41.



Gambar 5.41 Implementasi antarmuka halaman “Manage Konsentrasi”
Sumber: [implementasi]

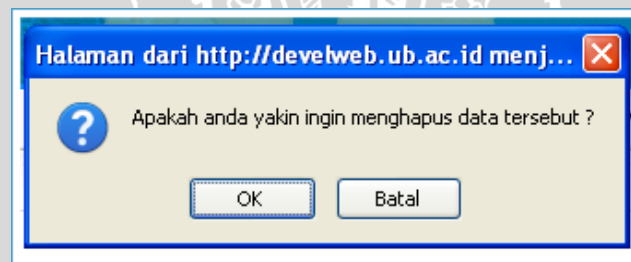
Pada implementasi antarmuka “Manage Konsentrasi” terdapat tombol “Tambah Data”, “Edit Data” dan “Hapus Data”. Fungsi tombol “Tambah Data” adalah untuk menambah data baru pada halaman “Manage Konsentrasi”. Tampilan halaman untuk “Tambah Data” ditunjukkan pada gambar 5.42.

Gambar 5.42 Implementasi antarmuka *pop-up* tambah data, “Form Manage Konsentrasi”
Sumber: [implementasi]

Jika aktor kajar memilih tombol “edit” pada halaman “Manage Konsentrasi” maka pada layar kemudian ditampilkan halaman yang berisi *form* yang akan diisi dengan data baru dari aktor kajar, yang ditunjukkan pada gambar 5.43.

Gambar 5.43 Implementasi antarmuka halaman “*Edit Manage Konsentrasi*”
Sumber: [implementasi]

Jika aktor kajar memilih tombol “*del*” pada halaman “*Manage Konsentrasi*”, maka pada layar kemudian ditampilkan *pop-up* yang berisi peringatan, yang ditunjukkan pada gambar 5.44.



Gambar 5.44 Implementasi antarmuka *pop-up* peringatan hapus data
Sumber: [implementasi]

5.4 Implementasi perangkat lunak aktor kdkk

Pada implementasi perangkat lunak dengan aktor kdkk mempunyai fitur-fitur antara lain: *Assign Ka.lab*, *Assign Laboratorium*, *Assign Group Praktikum*, *Assign Konsentrasi Praktikum*, *news from lab*, *Master Praktikum*, *Master Laboratorium*. Tampilan halaman untuk *Assign Kalab* ditunjukkan pada gambar 5.45.



Gambar 5.45 Implementasi antarmuka *Assign* kalab
Sumber: [implementasi]

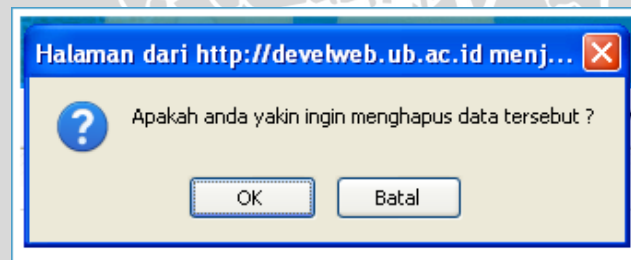
Pada implementasi antarmuka “*Assign Ka.lab*” terdapat tombol tambah data, *edit* data dan hapus data. Fungsi tombol “Tambah Data” adalah untuk menambah data baru pada halaman “*Assign Ka.lab*”. Tampilan halaman untuk “Tambah Data” ditunjukkan pada gambar 5.46.

Gambar 5.46 Implementasi antarmuka *pop-up* “*Form Assign Ka.lab*”
Sumber: [implementasi]

Jika aktor *kkdk* memilih tombol “*edit*” pada halaman “*Assign Ka.lab*” maka pada layar kemudian ditampilkan halaman yang berisi *form* yang akan diisi dengan data baru dari aktor *kkdk*, yang ditunjukkan pada gambar 5.47.

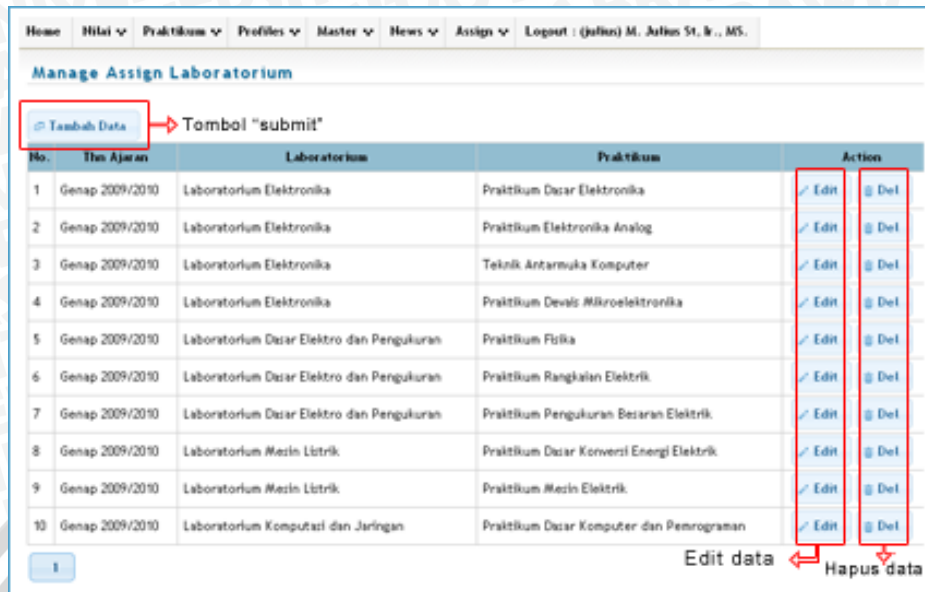
Gambar 5.47 Implementasi antarmuka halaman “*Edit Assign Ka.lab*”
Sumber: [implementasi]

Jika aktor kalab memilih tombol “*del*” pada halaman “*Assign Ka.lab*”, maka pada layar kemudian ditampilkan *pop-up* yang berisi peringatan, yang ditunjukkan pada gambar 5.48.



Gambar 5.48 Implementasi antarmuka *pop-up* peringatan hapus data
Sumber: [implementasi]

Tampilan halaman untuk *Manage Assign* Laboratorium ditunjukkan pada gambar 5.49.



Gambar 5.49 Implementasi antarmuka halaman "Manage Assign Laboratorium"
Sumber: [implementasi]

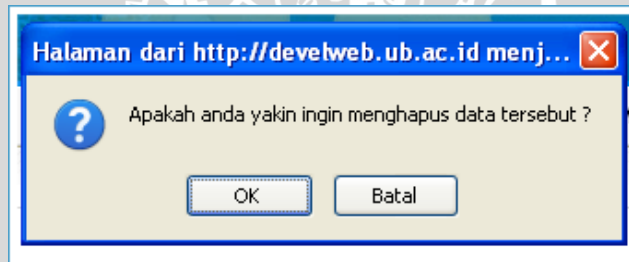
Pada implementasi antarmuka "Manage Assign Laboratorium" terdapat tombol tambah data, *edit* data dan hapus data. Fungsi tombol "Tambah Data" adalah untuk menambah data baru pada halaman "Manage Assign Laboratorium". Tampilan halaman untuk "Tambah Data" ditunjukkan pada gambar 5.50.

Gambar 5.50 Implementasi antarmuka *pop-up* tambah "Assign Group Laboratorium"
Sumber: [implementasi]

Jika aktor *kkdk* memilih tombol "edit" pada halaman "Assign Group Laboratorium" maka pada layar kemudian ditampilkan halaman yang berisi *form* yang akan diisi dengan data baru dari aktor *kkdk*, yang ditunjukkan pada gambar 5.51.

Gambar 5.51 Implementasi antarmuka halaman “*Edit Assign Group Laboratorium*”
Sumber: [implementasi]

Jika aktor kkdk memilih tombol “*del*” pada halaman “*Assign Group Laboratorium*”, maka pada layar kemudian ditampilkan *pop-up* yang berisi peringatan, yang ditunjukkan pada gambar 5.52.



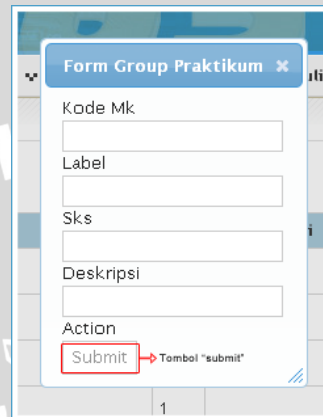
Gambar 5.52 Implementasi antarmuka *pop-up* peringatan hapus data
Sumber: [implementasi]

Tampilan halaman untuk *Manage Group Praktikum* ditunjukkan pada gambar 5.53.

No.	Kode MK	Label	Sks	Deskripsi	Action
1	TXE4138	Praktikum ABCDE-1	1		Edit Del
2	TXE4243	Praktikum ABCDE-2	1		Edit Del
3	TXE4139	Praktikum ABCDE-3	1		Edit Del
4	TXE4242	Praktikum A-1	1		Edit Del
5	TXE4137	Praktikum A-2	1		Edit Del
6	TXE4244	Praktikum B-1	1		Edit Del
7	TXE4140	Praktikum B-2	1		Edit Del
8	TXE4245	Praktikum C-1	1		Edit Del
9	TXE4141	Praktikum C-2	1		Edit Del
10	TXE4246	Praktikum D-1	1		Edit Del

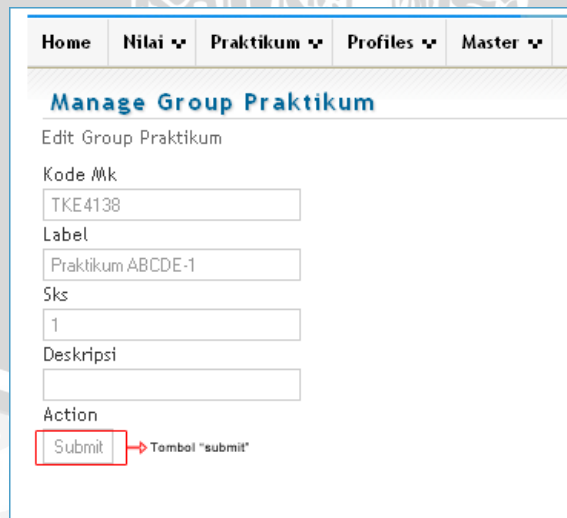
Gambar 5.53 Implementasi antarmuka *Manage Group Praktikum*
Sumber: [implementasi]

Pada implementasi antarmuka “*Manage Group Praktikum*” terdapat tombol tambah data, *edit* data dan hapus data. Fungsi tombol “Tambah Data” adalah untuk menambah data baru pada halaman “*Manage Group Praktikum*”. Tampilan halaman untuk “Tambah Data” ditunjukkan pada gambar 5.54.



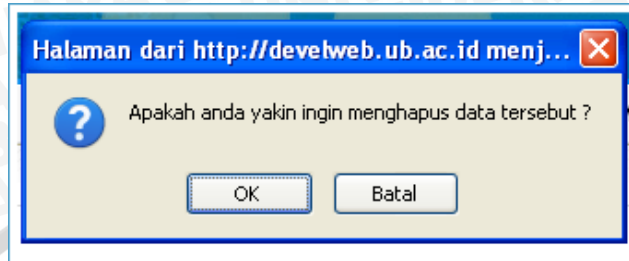
Gambar 5.54 Implementasi antarmuka *pop-up* “*Form Group Praktikum*”
Sumber: [implementasi]

Jika aktor *kkdk* memilih tombol “*edit*” pada halaman “*Manage Group Praktikum*” maka pada layar kemudian ditampilkan halaman yang berisi *form* yang akan diisi dengan data baru dari aktor *kkdk*, yang ditunjukkan pada gambar 5.55.



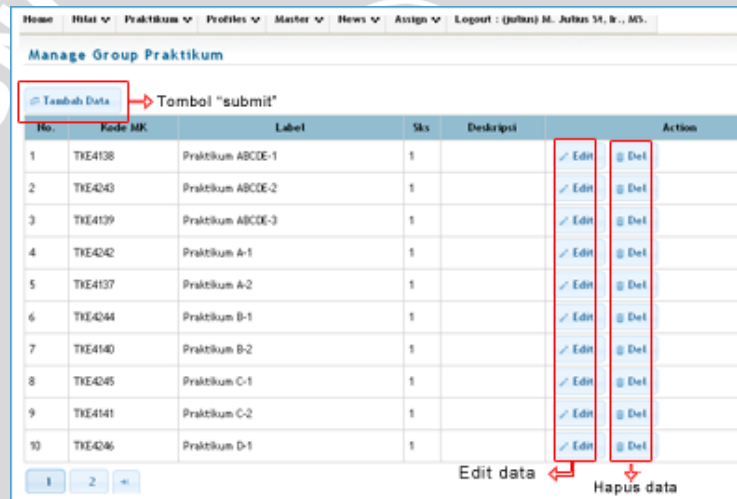
Gambar 5.55 Implementasi antarmuka halaman “*Edit Manage Group Praktikum*”
Sumber: [implementasi]

Jika aktor kalab memilih tombol “del” pada halaman “Manage Group Praktikum”, maka pada layar kemudian ditampilkan *pop-up* yang berisi peringatan, yang ditunjukkan pada gambar 5.56.



Gambar 5.56 Implementasi antarmuka *pop-up* peringatan hapus data
Sumber: [implementasi]

Tampilan halaman untuk *Manage* Konsentrasi Praktikum ditunjukkan pada gambar 5.57.



Gambar 5.57 Implementasi antarmuka halaman *Manage* Konsentrasi Praktikum
Sumber: [implementasi]

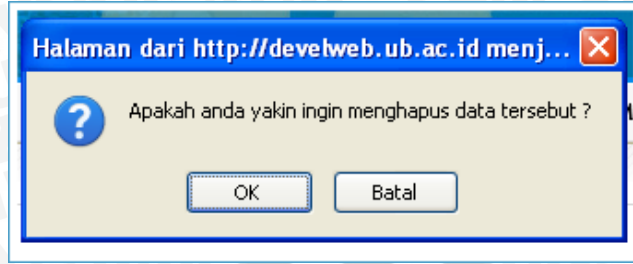
Pada implementasi antarmuka “Manage Konsentrasi Praktikum” terdapat tombol tambah data, *edit* data dan hapus data. Fungsi tombol “Tambah Data” adalah untuk menambah data baru pada halaman “Manage Konsentrasi Praktikum”. Tampilan halaman untuk “Tambah Data” ditunjukkan pada gambar 5.58.

Gambar 5.58 Implementasi antarmuka *pop-up* “Form Konsentrasi Praktikum”
Sumber: [implementasi]

Jika aktor *kkdk* memilih tombol “*edit*” pada halaman “*Manage Konsentrasi Praktikum*” maka pada layar kemudian ditampilkan halaman yang berisi *form* yang akan diisi dengan data baru dari aktor *kkdk*, yang ditunjukkan pada gambar 5.59.

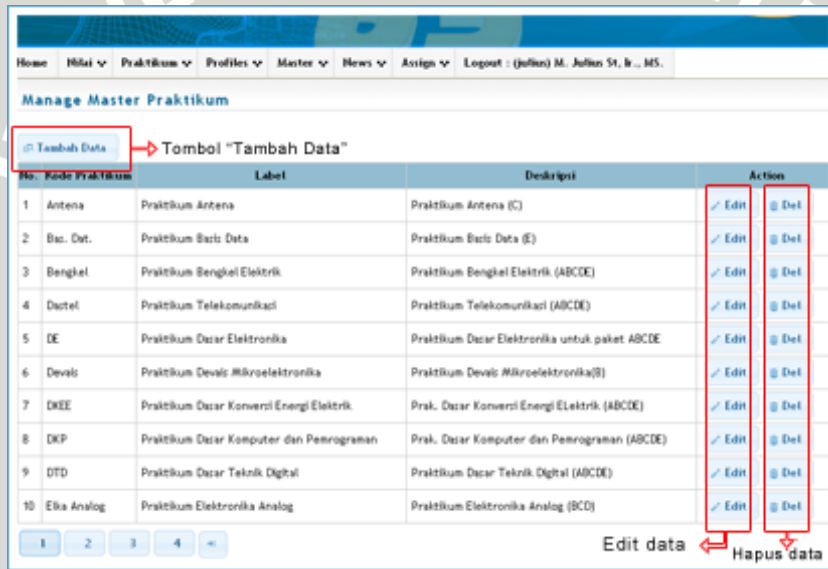
Gambar 5.59 Implementasi antarmuka halaman “*Edit Manage Konsentrasi Praktikum*”
Sumber: [implementasi]

Jika aktor *kkdk* memilih tombol “*del*” pada halaman “*Manage Konsentrasi Praktikum*”, maka pada layar kemudian ditampilkan *pop-up* yang berisi peringatan, yang ditunjukkan pada gambar 5.60.



Gambar 5.60 Implementasi antarmuka *pop-up* peringatan hapus data
Sumber: [implementasi]

Pada menu “Master” memiliki 2 submenu yaitu: menu “Master Praktikum” dan menu “Master Laboratorium”. Untuk tampilan halaman menu master praktikum ditunjukkan dengan gambar 5.61.



Gambar 5.61 Implementasi antarmuka halaman *Manage Master Praktikum*
Sumber: [implementasi]

Pada implementasi antarmuka “Manage Master Praktikum” terdapat tombol tambah data, *edit data* dan *hapus data*. Fungsi tombol “Tambah Data” adalah untuk menambah data baru pada halaman “Manage Master Praktikum”. Tampilan halaman untuk “Tambah Data” ditunjukkan pada gambar 5.62.

Form Auth. Level Menu

Kode Praktikum

Label

Deskripsi

Action

Submit → Tombol "Submit"

Praktikum Elektronika Analog (BCD)

Gambar 5.62 Implementasi antarmuka *pop-up* “Form Master Praktikum”
Sumber: [implementasi]

Jika aktor *kkdk* memilih tombol “*edit*” pada halaman “*Manage Master Praktikum*” maka pada layar kemudian ditampilkan halaman yang berisi *form* yang akan diisi dengan data baru dari aktor *kkdk*, yang ditunjukkan pada gambar 5.63.

Home Nilai Praktikum Profiles Master News

Manage Master Praktikum

Edit Master Praktikum

Kode Praktikum

Antena

Label

Praktikum Antena

Deskripsi

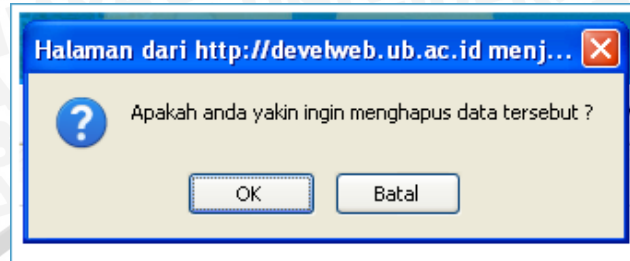
Praktikum Antena (C)

Action

Submit → Tombol "Submit"

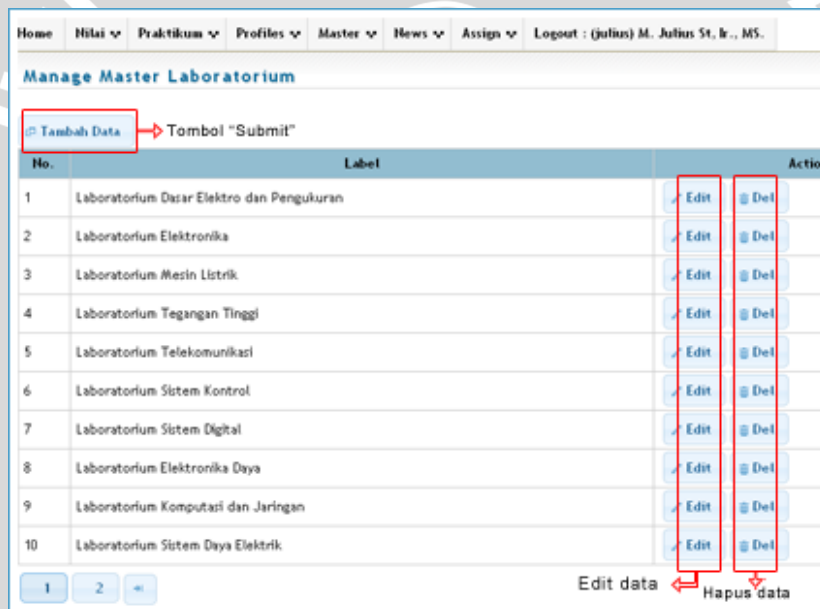
Gambar 5.63 Implementasi antarmuka halaman “*Edit Manage Master Praktikum*”
Sumber: [implementasi]

Jika aktor kkdik memilih tombol “del” pada halaman “*Manage Master Praktikum*”, maka pada layar kemudian ditampilkan *pop-up* yang berisi peringatan, yang ditunjukkan pada gambar 5.64.



Gambar 5.64 Implementasi antarmuka *pop-up* peringatan hapus data
Sumber: [implementasi]

Untuk tampilan halaman *menu* “*Master Laboratorium*” ditunjukkan dengan gambar 5.65.



Gambar 5.65 Implementasi antarmuka halaman *Manage Master Praktikum*
Sumber: [implementasi]

Pada implementasi antarmuka “*Manage Master Laboratorium*” terdapat tombol tambah data, *edit* data dan hapus data. Fungsi tombol “Tambah Data” adalah untuk menambah data baru pada halaman “*Manage Master Laboratorium*”. Tampilan halaman untuk “Tambah Data” ditunjukkan pada gambar 5.66.

Gambar 5.66 Implementasi antarmuka *pop-up* tambah “*Form Master Laboratorium*”
Sumber: [implementasi]

Jika aktor *kkdk* memilih tombol “*edit*” pada halaman “*Manage Master Laboratorium*” maka pada layar kemudian ditampilkan halaman yang berisi *form* yang akan diisi dengan data baru dari aktor *kkdk*, yang ditunjukkan pada gambar 5.67.

Gambar 5.67 Implementasi antarmuka halaman “*Edit Manage Master Laboratorium*”
Sumber: [implementasi]

Jika aktor *kkdk* memilih tombol “*del*” pada halaman “*Manage Master Laboratorium*”, maka pada layar kemudian ditampilkan *pop-up* yang berisi peringatan, yang ditunjukkan pada gambar 5.68.

Gambar 5.68 Implementasi antarmuka *pop-up* peringatan hapus data
Sumber: [implementasi]

5.5 Implementasi perangkat lunak aktor kalab

Pada implementasi perangkat lunak dengan aktor kalab mempunyai fitur-fitur antara lain: *assign* laboran, *news from labs*. Menu *assign* laboran adalah fitur yang hanya bisa digunakan oleh aktor kalab saja, dimana bertujuan untuk mengangkat / menugaskan karyawan yang ada di Teknik Elektro sebagai laboran. Tampilan halaman untuk *assign* laboran ditunjukkan pada gambar 5.69.



Gambar 5.69 Implementasi antarmuka *assign* laboran

Sumber: [implementasi]

Pada implementasi antarmuka “Assign Laboran” terdapat tombol tambah data, *edit* data dan hapus data. Fungsi tombol “Tambah Data” adalah untuk menambah data baru pada halaman “Assign Laboran”. Tampilan halaman untuk “Tambah Data” ditunjukkan pada gambar 5.70.

Gambar 5.70 Implementasi antarmuka *pop-up* “Form Assign Laboran”

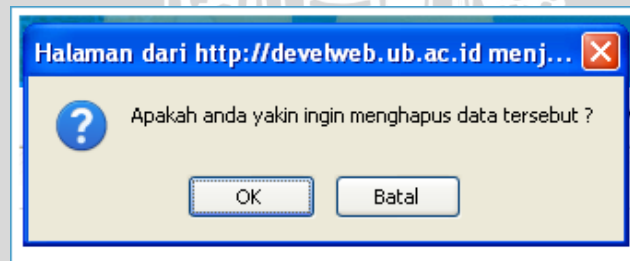
Sumber: [implementasi]

Jika aktor kalab memilih tombol “*edit*” pada halaman “*Assign Laboran*” maka pada layar kemudian ditampilkan halaman yang berisi *form* yang akan diisi dengan data baru dari aktor kalab, yang ditunjukkan pada gambar 5.71

Gambar 5.71 Implementasi antarmuka halaman “*Edit Assign Laboran*”

Sumber: [implementasi]

Jika aktor kalab memilih tombol “*del*” pada halaman “*Assign Laboran*”, maka pada layar kemudian ditampilkan *pop-up* yang berisi peringatan, yang ditunjukkan pada gambar 5.72.



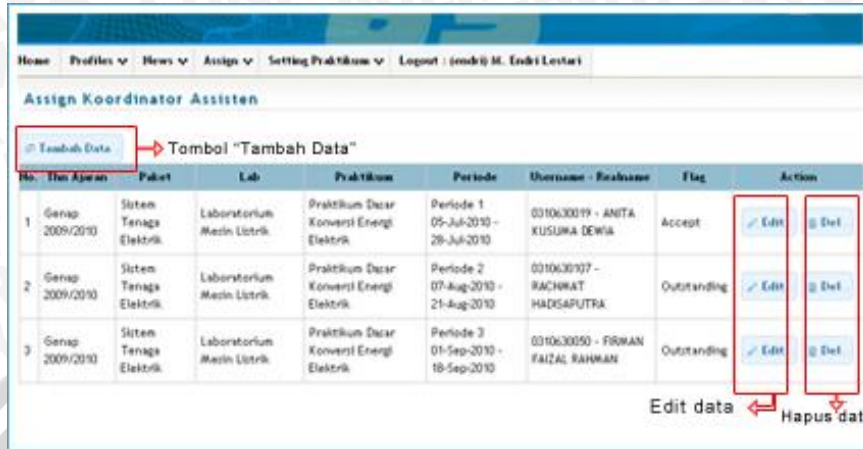
Gambar 5.72 implementasi antarmuka *pop-up* peringatan hapus data

Sumber: [implementasi]

5.6 Implementasi perangkat lunak aktor laboran

Pada implementasi perangkat lunak dengan aktor laboran mempunyai fitur-fitur antara lain: *Assign Ko.Ass*, *Setting Praktikum*, *News*. Pada menu “*Setting Praktikum*” memiliki 2 *submenu*, antara lain : *Setting Kelompok Posstest*, *Setting Periode Praktikum*.

Apabila actor laboran memilih menu “Assign Ko.Ass”, maka tampilan halaman untuk Assign Ko.Ass ditunjukkan pada gambar 5.73.

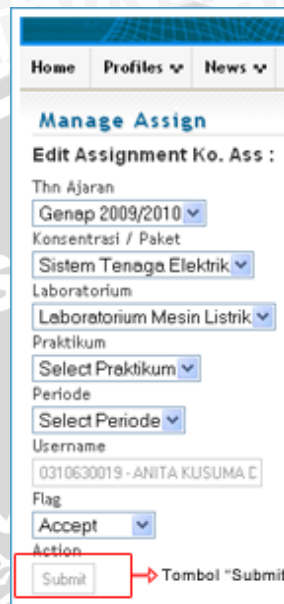


Gambar 5.73 Implementasi antarmuka Assign Koordinator Asisten
Sumber: [implementasi]

Pada implementasi antarmuka “Assign Ko.Ass” terdapat tombol tambah data, edit data dan hapus data. Fungsi tombol “Tambah Data” adalah untuk menambah data baru pada halaman “Assign Ko.Ass”. Tampilan halaman untuk “Tambah Data” ditunjukkan pada gambar 5.74.

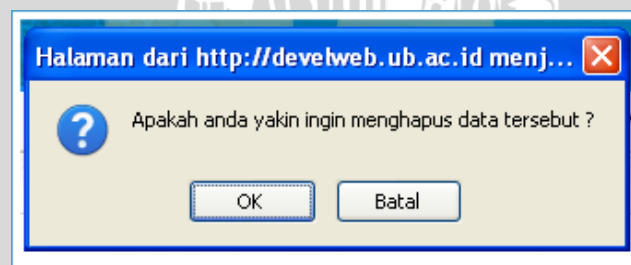
Gambar 5.74 Implementasi antarmuka pop-up “Form Assign Ka.lab”
Sumber: [implementasi]

Jika aktor kkdk memilih tombol “*edit*” pada halaman “*Assign Ko.Ass*” maka pada layar kemudian ditampilkan halaman yang berisi *form* yang akan diisi dengan data baru dari aktor laboran, yang ditunjukkan pada gambar 5.75.



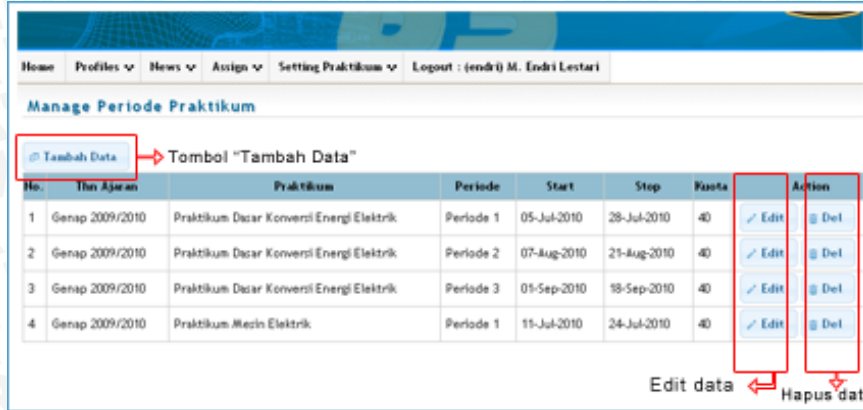
Gambar 5.75 Implementasi antarmuka halaman “*Edit Assign Ko.Ass*”
Sumber: [implementasi]

Jika aktor laboran memilih tombol “*del*” pada halaman “*Assign Ko.Ass*”, maka pada layar kemudian ditampilkan *pop-up* yang berisi peringatan, yang ditunjukkan pada gambar 5.76.



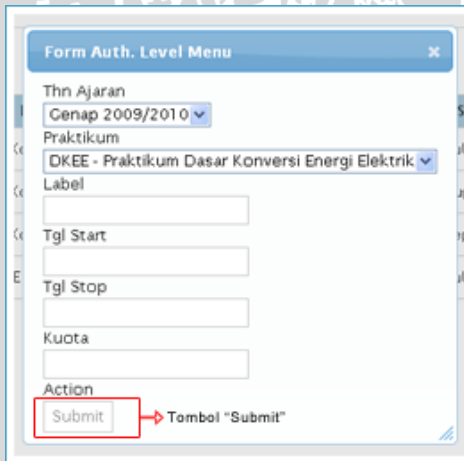
Gambar 5.76 Implementasi antarmuka *pop-up* peringatan hapus data
Sumber: [implementasi]

Apabila aktor laboran memilih menu *Setting* Praktikum kemudian memilih *submenu* “*Setting Periode Praktikum*”, maka tampilan halaman untuk “*Setting Periode Praktikum*” ditunjukkan pada gambar 5.77.



Gambar 5.77 Implementasi antarmuka *Assign Setting Periode Praktikum*
Sumber: [implementasi]

Pada implementasi antarmuka “*Setting Periode Praktikum*” terdapat tombol tambah data, *edit* data dan hapus data. Fungsi tombol “Tambah Data” adalah untuk menambah data baru pada halaman “*Setting Periode Praktikum*”. Tampilan halaman untuk “Tambah Data” ditunjukkan pada gambar 5.78.

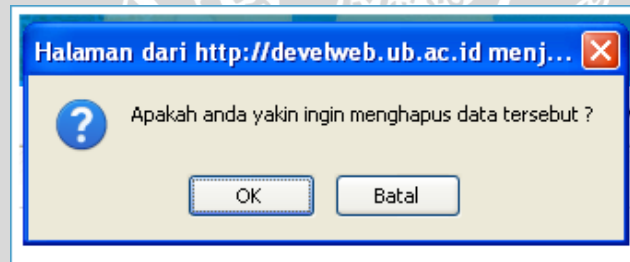


Gambar 5.78 Implementasi antarmuka *pop-up “Form Setting Periode Praktikum”*
Sumber: [implementasi]

Jika aktor *kkdk* memilih tombol “*edit*” pada halaman “*Setting Periode Praktikum*” maka pada layar kemudian ditampilkan halaman yang berisi *form* yang akan diisi dengan data baru dari aktor *laboran*, yang ditunjukkan pada gambar 5.79.

Gambar 5.79 Implementasi antarmuka halaman “*Edit Setting Periode Praktikum*”
Sumber: [implementasi]

Jika aktor laboran memilih tombol “*del*” pada halaman “*Setting Periode Praktikum*”, maka pada layar kemudian ditampilkan *pop-up* yang berisi peringatan, yang ditunjukkan pada gambar 5.80.



Gambar 5.80 Implementasi antarmuka *pop-up* peringatan hapus data
Sumber: [implementasi]

5.7 Implementasi perangkat lunak aktor koass

Pada implementasi perangkat lunak aplikasi dilakukan implementasi perangkat lunak untuk *login*, perangkat lunak untuk aktor koass mempunyai beberapa fitur seperti: *manage setting* bab praktikum, *manage setting* kelompok jadwal, nilai, praktikum. Berikut ini adalah tampilan dari ketika aktor koass berhasil login, yang ditunjukkan pada gambar 5.81.



Gambar 5.81 implementasi antarmuka halaman awal setelah sukses *login* sebagai aktor koass
Sumber: [implementasi]

1. *Menu Manage Setting* Bab Praktikum

Setelah sukses *login* sebagai koass, aktor dapat melakukan perubahan data pada menu manage setting bab praktikum. Proses tersebut dapat dilakukan dengan memilih menu “*Setting Praktikum*” pada halaman utama SIMLATTE yang ditunjukkan pada Gambar 5.82.



Gambar 5.82 Implementasi antarmuka halaman *manage* bab praktikum
Sumber: [implementasi]

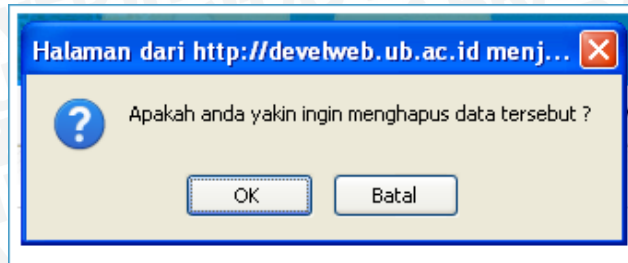
Jika aktor koass memilih tombol tambah data pada halaman *manage* bab praktikum maka akan muncul *pop-up* yang ditunjukkan pada gambar 5.83.

Gambar 5.83 Implementasi antarmuka *pop-up* tambah data *form* bab praktikum
Sumber: [implementasi]

Jika aktor koass memilih tombol “*edit*” pada halaman *manage* bab praktikum maka pada layar kemudian ditampilkan halaman yang berisi *form* yang akan diisi dengan data baru dari aktor koass, yang ditunjukkan pada gambar 5.84

Gambar 5.84 Implementasi antarmuka halaman *edit – manage* bab praktikum
Sumber: [implementasi]

Jika aktor koass memilih tombol “*del*” pada halaman *manage* bab praktikum, maka pada layar kemudian ditampilkan *pop-up* yang berisi peringatan, yang ditunjukkan pada gambar 5.85.



Gambar 5.85 implementasi antarmuka *pop-up* peringatan hapus data
Sumber: [implementasi]

2. Menu Manage Setting Kelompok Jadwal

Untuk memilih *menu setting* kelompok jadwal pada menu “Setting Praktikum”. Setelah itu pada layar ditampilkan halaman yang berisi data lengkap dari masing-masing jadwal setiap bab pada praktikum yang diasistensi oleh aktor yang bersangkutan. Tampilan halaman untuk melihat data masing-masing jadwal setiap bab ditunjukkan pada gambar 5.86.

No.	Praktikum	Periode	Bab	Tanggal Start	Tanggal Stop	Max Praktikum	Action
1	DKEE - Praktikum Dasar Konversi Energi Elektrik	Periode 1	Bab 1	11-Jul-2010 07:00	11-Jul-2010 07:30	7	✓ Edit Del
2	DKEE - Praktikum Dasar Konversi Energi Elektrik	Periode 1	Bab 1	10-Jul-2010 11:15	10-Jul-2010 11:45	7	✓ Edit Del
3	DKEE - Praktikum Dasar Konversi Energi Elektrik	Periode 1	Bab 2	13-Jul-2010 14:30	13-Jul-2010 15:15	7	✓ Edit Del
4	ME - Praktikum Mesin Elektrik	Periode 2	Bab 1	09-Sep-2010 12:15	09-Sep-2010 13:30	7	✓ Edit Del
5	ME - Praktikum Mesin Elektrik	Periode 2	Bab 2	17-Jul-2010 12:30	17-Jul-2010 13:30	7	✓ Edit Del

Gambar 5.86 Implementasi antarmuka halaman *Manage Jadwal Kelompok*
Sumber: [implementasi]

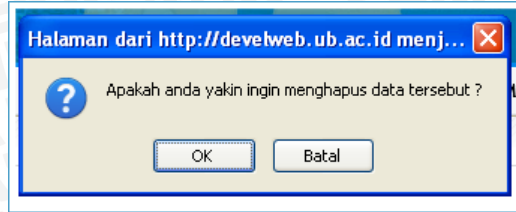
Jika aktor koass memilih tombol tambah data pada halaman *Manage Jadwal Kelompok* maka akan muncul *pop-up* yang ditunjukkan pada gambar 5.87.

Gambar 5.87 Implementasi antarmuka *pop-up* tambah data *Form Jadwal Kelompok*
Sumber: [implementasi]

Jika aktor koass memilih tombol “*edit*” pada halaman *Manage Jadwal Kelompok* maka pada layar kemudian ditampilkan halaman yang berisi *form* yang akan diisi dengan data baru dari aktor koass, yang ditunjukkan pada gambar 5.88.

Gambar 5.88 Implementasi antarmuka halaman *edit* – *Manage Jadwal Kelompok*
Sumber: [implementasi]

Jika aktor koass memilih tombol “*del*” pada halaman *manage* bab praktikum, maka pada layar kemudian ditampilkan *pop-up* yang berisi peringatan, yang ditunjukkan pada gambar 5.89.



Gambar 5.89 implementasi antarmuka *pop-up* peringatan hapus data pada *Manage Jadwal Kelompok*
Sumber: [implementasi]

5.8 Implementasi perangkat lunak aktor praktikan

Pada implementasi perangkat lunak aplikasi dilakukan implementasi perangkat lunak untuk *login*, perangkat lunak untuk aktor koass mempunyai beberapa fitur seperti: praktikum, nilai. Tampilan untuk aktor praktikan ketika berhasil login ditunjukkan oleh gambar 5.90.



Gambar 5.90 implementasi antarmuka halaman awal setelah sukses *login* sebagai aktor praktikan
Sumber: [implementasi]

Setelah sukses *login* sebagai praktikan, aktor dapat melakukan perubahan data pada menu “Praktikum”. Proses tersebut dapat dilakukan dengan memilih menu “Daftar Praktikum” pada halaman utama SIMLATTE yang ditunjukkan pada Gambar 5.91.

No.	Praktikum	Periode / Tanggal	Action
1	Praktikum Dasar Konversi Energi Listrik	Periode 1 : 05-July-2010 s/d 28-July-2010 Periode 2 : 07-August-2010 s/d 21-August-2010 Periode 3 : 05-September-2010 s/d 18-September-2010	Setting
2	Praktikum Dasar Mikroelektronika	Periode 1 : 11-October-2010 s/d 27-October-2010	Daftar
3	Praktikum Elektronika Analog	Periode 1 : 10-September-2010 s/d 24-September-2010	Daftar
4	Praktikum Mesin Listrik	Periode 1 : 11-July-2010 s/d 24-July-2010	Daftar
5	Teknik Antarmuka Komputer	Periode 1 : 18-August-2010 s/d 31-August-2010	Daftar

Gambar 5.91 Implementasi antarmuka halaman *manage* bab praktikum

Sumber: [implementasi]

Jika aktor praktikan memilih tombol “Daftar Praktikum” pada halaman “Daftar Periode” maka akan muncul halaman yang ditunjukkan pada gambar 5.92



Gambar 5.92 Implementasi antarmuka *pop-up* tambah data *form* bab praktikum
 Sumber: [implementasi]

