

BAB IV

HASIL DAN PEMBAHASAN

4.1. Perancangan Aplikasi

Perancangan Aplikasi pada pengembangan Sistem Informasi Pendaftaran Peserta Kursus di UPT Balai Pelatihan Kerja (BLK) Kabupaten Kudus menggunakan metode RAD (*Rapid Application Development*) dengan alur atau siklus yang antara lain dapat dikategorikan sebagai beberapa tahapan yang antara lain adalah *Requirement Planning*, *System Design*, dan *Implementation* yang akan dijabarkan pada penjelasan berikut :

4.1.1. Requirement Planning (Perencanaan Kebutuhan)

Pada Tahap ini, dilaksanakan suatu analisa kebutuhan masalah yang digunakan untuk pengembangan atau pembuatan aplikasi yang antara lain memerlukan analisa kebutuhan yang akan dijabarkan pada penjelasan berikut:

a. Analisa Kebutuhan Masalah

Sistem pendaftaran merupakan suatu sistem yang digunakan untuk pencatatan ataupun pendataan data pendaftar. Semakin berkembangnya teknologi membuat beberapa sistem mendapat suatu adopsi terhadap ilmu pengetahuan dan teknologi yang mana dapat berupa suatu aplikasi atau suatu sistem yang berjalan pada suatu perangkat. Hal ini merupakan suatu peralihan sistem dari suatu sistem yang dapat dianggap sebagai konvensional menuju ke sistem yang dapat disebut berupa aplikasi pada perangkat.

Pada aplikasi atau sistem yang diterapkan pada suatu BLK (Balai Latihan Kerja) yang dalam ruang lingkup pembahasan ini adalah UPT BLK Kudus, memerlukan suatu kesesuaian yang berhubungan dengan suatu lingkup yang berhubungan dengan proses pendaftaran lama sehingga pengembangan sistem ke wujud sistem aplikasi perangkat terdapat suatu kesesuaian.

b. Analisa Kebutuhan Alat

Analisa terhadap kebutuhan alat yang digunakan pada pengembangan atau perancangan sistem yang digunakan oleh peneliti antara lain adalah sebagai berikut :

1. Perangkat Keras (Software)

Perangkat keras yang digunakan pada pengembangan atau perancangan adalah Laptop dengan rincian sebagai berikut :

- a. AMD E2 Dual Core.
- b. RAM 4 GB DDR3.
- c. Hardisk 320 GB.

2. Perangkat Lunak (Hardware)

Perangkat lunak yang digunakan pada pengembangan atau perancangan adalah sebagai berikut :

- a. UML Diagram *Software* seperti *Star UML*, *Argo UML*, ataupun *Enterprise Arsitect* yang digunakan untuk pembuatan diagram UML.
- b. *Xampp* yang digunakan sebagai *web server (apache)* dan *database mySQL*.
- c. *Text Editor (Notepad++ atau Visual Studio Code)* untuk menulis *script* (kode pemrograman) yang digunakan pada pengembangan atau perancangan aplikasi.
- d. *Browser (Firefox, Chrome, Opera, dll)* yang digunakan untuk menampilkan hasil dari *script* dan *xampp*.

c. Analisa Kebutuhan Fungsional Sistem

Analisa kebutuhan fungsional dari sistem yang dikembangkan antara lain adalah sebagai berikut :

1. Aplikasi atau sistem yang dikembangkan dapat digunakan untuk pendataan peserta.
2. Admin dapat mengelola data yang berhubungan dengan aplikasi atau sistem yang dikembangkan seperti kelola laporan dan kejuruan.

d. Analisa Kebutuhan Non Fungsional Sistem

Kebutuhan non-fungsional dari sistem merupakan kebutuhan yang tidak berpengaruh secara langsung dengan spesifikasi yang dibutuhkan untuk sistem. Kebutuhan ini berhubungan dengan komponen sistem yang mungkin muncul setelah sistem selesai. Hal tersebut antara lain seperti respon sistem disaat digunakan, media penyimpanan (*storage*) yang diperlukan, maupun keefektifan sistem. Kebutuhan non-fungsional aplikasi atau sistem yang dikembangkan atau dirancang antara lain adalah :

1. Aplikasi atau sistem ini dapat digunakan pada laptop atau komputer dengan processor minimal AMD E2 atau sejenisnya.
2. Aplikasi atau sistem ini dapat digunakan secara optimal pada laptop atau komputer dengan RAM minimal 2 GB dan Hardisk 320 GB.

e. Analisa Data

Analisa data yang diperlukan terhadap aplikasi atau sistem antara lain adalah :

1. Data yang berisi identitas pendaftar.

Data ini mencakup hal seperti nama pendaftar, nik pendaftar, alamat, pendaftar, tanggal lahir pendaftar, maupun hal lainnya yang digunakan pada pendataan pendaftar.

2. Data yang berkaitan dengan kejuruan.

Pada Data ini, hal yang dimaksud merupakan data kejuruan seperti kejuruan otomotif, tata boga, perkomputeran, desain grafis, maupun yang lainnya. Data tersebut dimasukkan ke suatu kelompok yang disebut kejuruan.

4.1.2. System Design (Desain System)

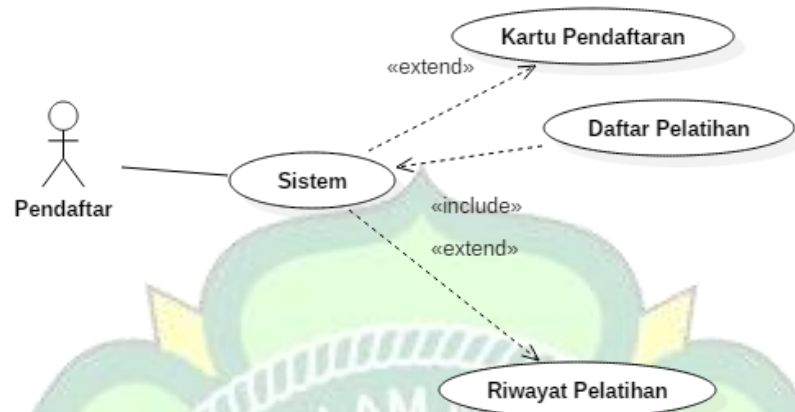
System Design atau yang dapat disebut juga desain dari suatu sistem. Pada pembahasan yang dibahas peneliti kali ini, peneliti

menggunakan desain sistem yang berupa UML (*Unified Modelling Language*) atau yang dapat dikenal juga sebagai merupakan kumpulan diagram - diagram yang didalamnya telah memiliki standar untuk membangun suatu perangkat lunak berbasis objek [6]. Yang akan dijabarkan pada beberapa penjelasan yang antara lain adalah sebagai berikut :

a. *Use Case Diagram*

Untuk *use case* yang akan peneliti jabarkan antara lain adalah sebagai berikut :

1. *Use Case Pendaftar*



Gambar 4. 1 *Use case* pendaftar

Sedangkan untuk penjelasannya antara lain adalah sebagai berikut:

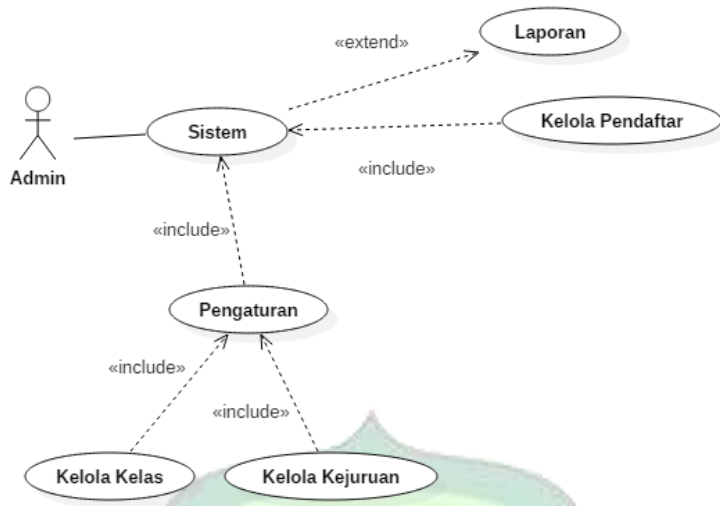
Tabel 4. 1 Tabel Penjelasan *use case* pendaftar

<p>Aktor Utama : Peserta</p> <p>Nama Use Case : <i>Usecase</i> Diagram peserta.</p> <p>Tujuan Use Case : <i>Usecase</i> ini bertujuan untuk membuat suatu prakira tentang hal yang dapat diaktivitaskan peserta terhadap aplikasi atau sistem.</p> <p>Kondisi Awal : Peserta</p>	<p>Optimistik flow :</p> <ol style="list-style-type: none"> 1. Peserta dapat melaksanakan pendaftaran dan memilih pelatihan kejuruan yang dipilih. 2. Data pendaftar masuk ke sistem atau aplikasi. <p>Pesimistik flow : Koneksi data tidak bagus</p>
--	---

melaksanakan *registrasi* atau pendaftaran akun.

Kondisi Akhir : Peserta telah terdaftar pada aplikasi atau sistem.

2. Use Case Admin



Gambar 4. 2 Use case admin

Sedangkan untuk penjelasannya antara lain adalah sebagai berikut:

Tabel 4. 2 Penjelasan use case admin

<p>Aktor Utama : admin</p> <p>Nama Use Case : Usecase Diagram admin.</p> <p>Tujuan Use Case : Usecase ini digunakan dalam membuat prakira tentang apa saja yang berhubungan dengan admin pada sistem.</p> <p>Kondisi Awal : admin masuk ke halaman admin.</p> <p>Kondisi Akhir : Data yang ada terkelola dan terpantau oleh admin.</p>	<p>Optimistik flow :</p> <ol style="list-style-type: none"> Admin dapat mengelola data yang telah tampil pada halaman admin. Admin masuk ke interface admin. <p>Pesimistik flow :</p> <p>-</p>
---	--

b. *Activity Diagram*

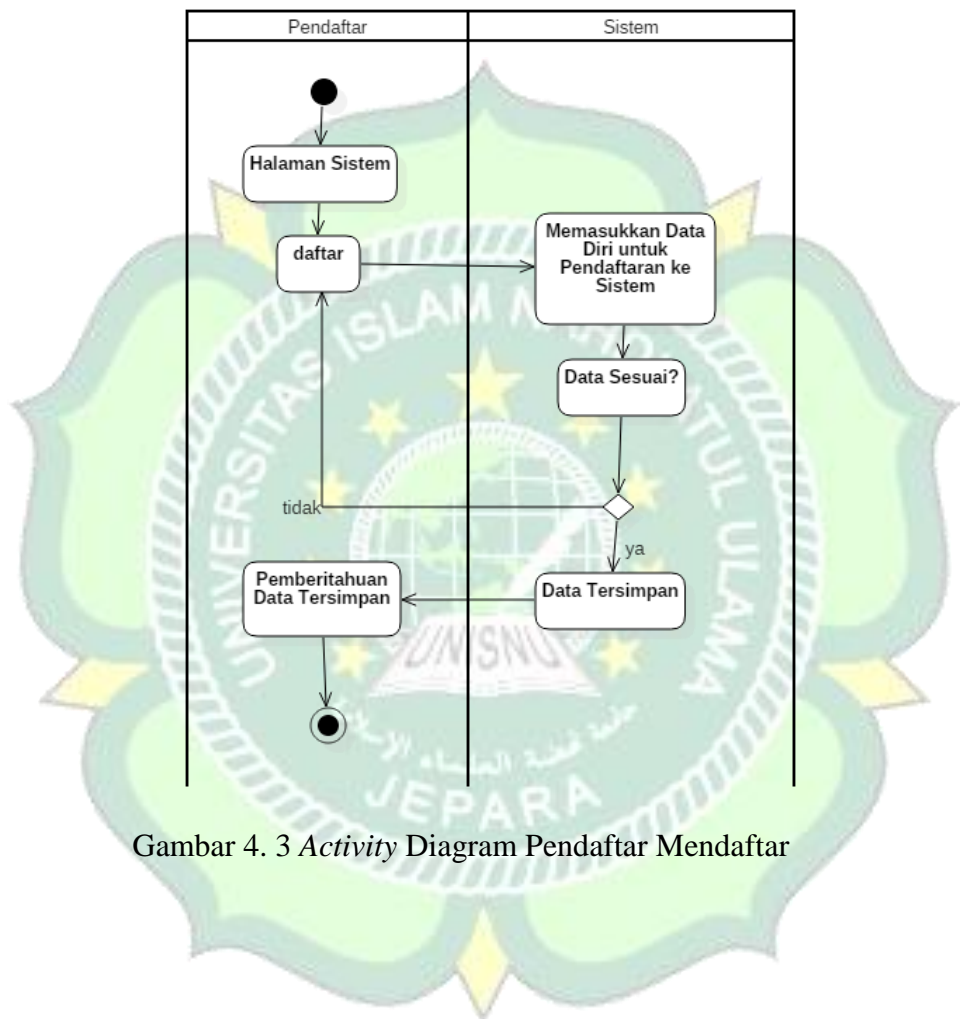
Pada *activity diagram*, akan dijabarkan atau diperlihatkan tentang suatu proses yang tergolong pada UML tipe *activity diagram* yang antara lain adalah sebagai berikut :

a. *Activity Diagram Pendaftar*

Untuk *activity diagram* dari pendaftar memiliki alur yang sebagai berikut:

1. *Activity Diagram Pendaftar Mendaftar*

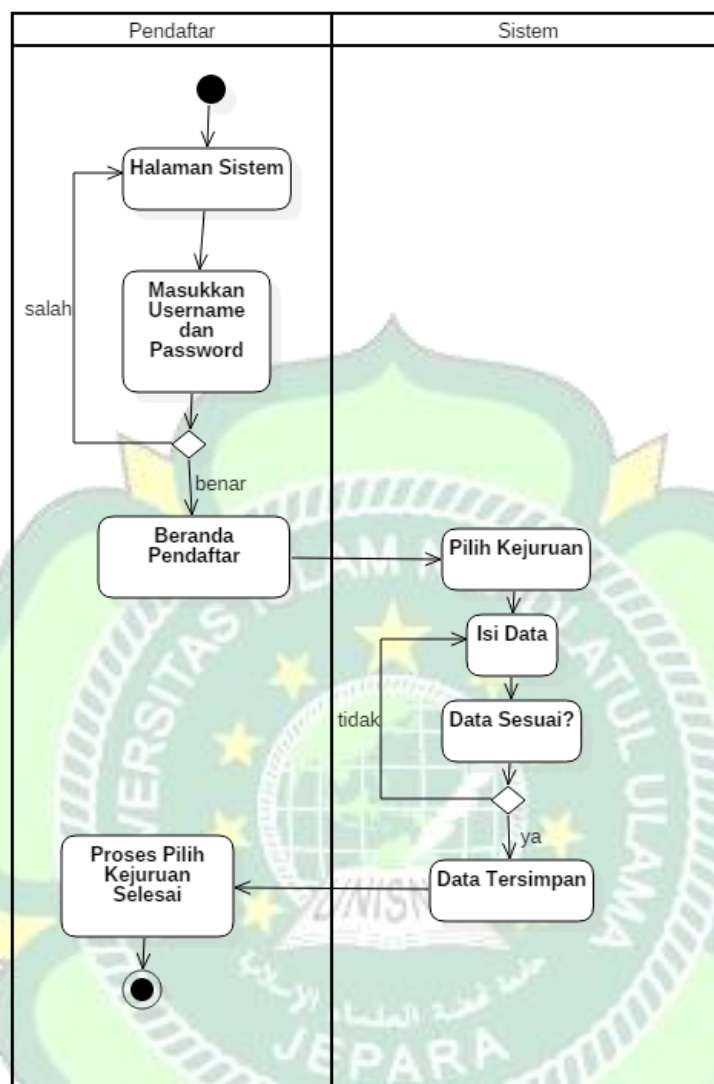
Merupakan suatu diagram *activity* yang memperlihatkan suatu alur dimana pendaftar melaksanakan suatu proses pendaftaran. Proses tersebut antara lain dapat dilihat pada gambar dibawah ini.



Gambar 4. 3 *Activity Diagram Pendaftar Mendaftar*

2. Activity Diagram Pendaftar Kejuruan

Merupakan suatu diagram *activity* yang memperlihatkan suatu alur dimana pendaftar melaksanakan suatu proses memilih kejuruan. Pada alur ini pendaftar memilih beberapa kejuruan yang telah tersedia untuk dipilih sehingga dapat terdaftar sebagai pendaftar pada kejuruan tersebut. Proses tersebut antara lain dapat dilihat dan diperhatikan pada gambar dibawah ini.



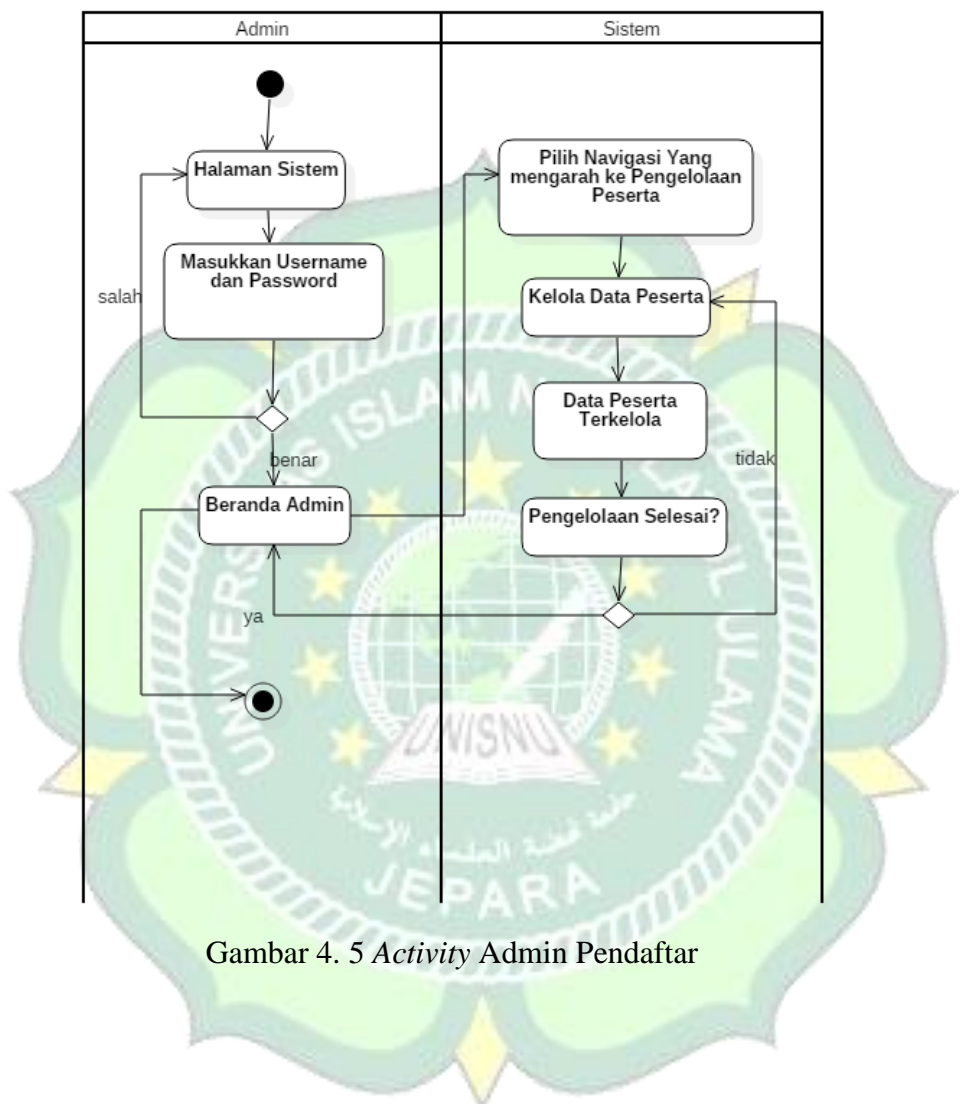
Gambar 4. 4 Activity Diagram Pendaftar Kejuruan

b. Activity Diagram Admin

Activity Diagram Admin terbagi menjadi beberapa macam kategori yangmana memiliki suatu penjelasan tersendiri. Activity diagram tersebut terkategori dalam beberapa macam penjelasan dalam suatu ketegori yang dibagi atau dikategori tersebut yang antara lain adalah sebagai berikut :

1. Activity Admin Pendaftar

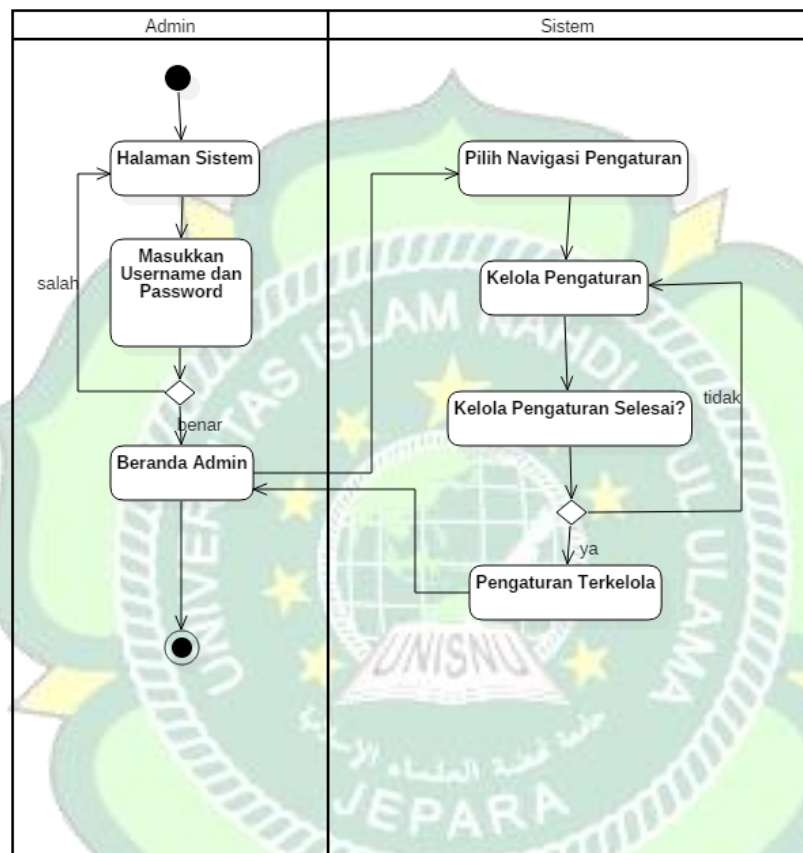
Pada activity diagram ini, menggambarkan tentang arus atau alur admin mengelola pendaftar yang diantaranya admin perlu masuk ke halaman admin terlebih dahulu. Yang akan diperlihatkan pada gambar dibawah ini.



Gambar 4. 5 Activity Admin Pendaftar

2. Activity Admin Pengaturan

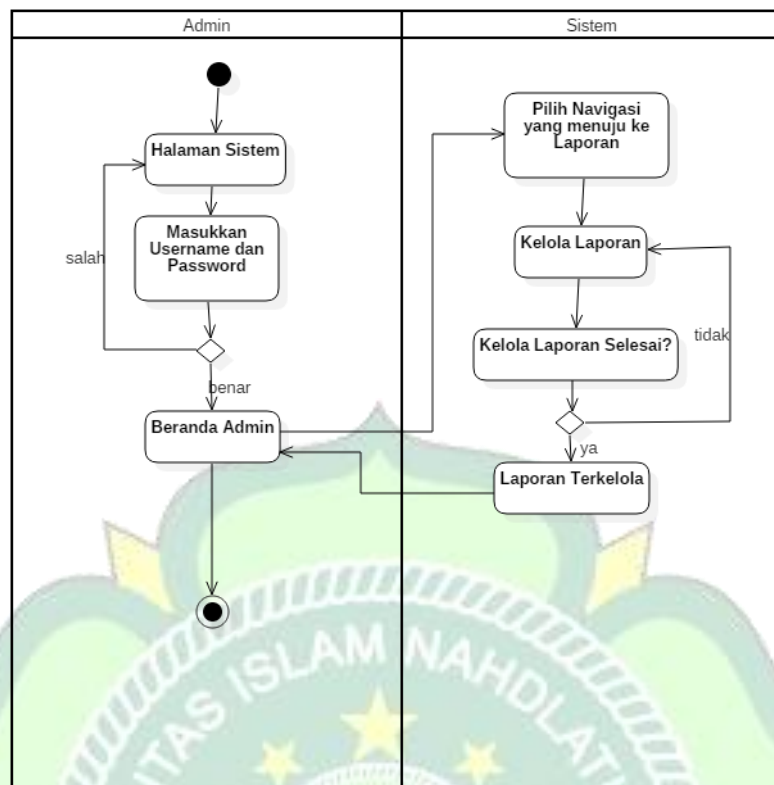
Pada *activity* diagram ini, menggambarkan tentang arus atau alur admin mengelola pengaturan yang diperlukan oleh pihak admin dalam suatu kegiatan pengelolaan pengaturan yang berhubungan dengan suatu proses pendaftaran oleh pendaftar sehingga admin dapat memantau dan mengatur jalannya pengaturan sesuai dengan sistem atau tata arus atau alur dari sistem sebelumnya (agar sesuai). Sehingga sistem atau aplikasi (pengaplikasian) mirip dengan sistem sebelumnya secara arus. Sehingga kegiatan admin dalam hal pengaturan, admin sedikit faham melalui penjelasan dan mudah terbiasa. Hal tersebut akan diperlihatkan pada gambar dibawah ini.



Gambar 4. 6 Activity Admin Pengaturan

3. Activity Admin Laporan

Pada *activity* diagram ini, menggambarkan tentang arus atau alur admin mengelola laporan merupakan suatu diagram atau alura (arus) kegiatan admin menuju ke pengelolaan laporan yang diperlukan oleh pihak admin. *Activity* diagram tersebut diperlihatkan pada gambar dibawah ini.

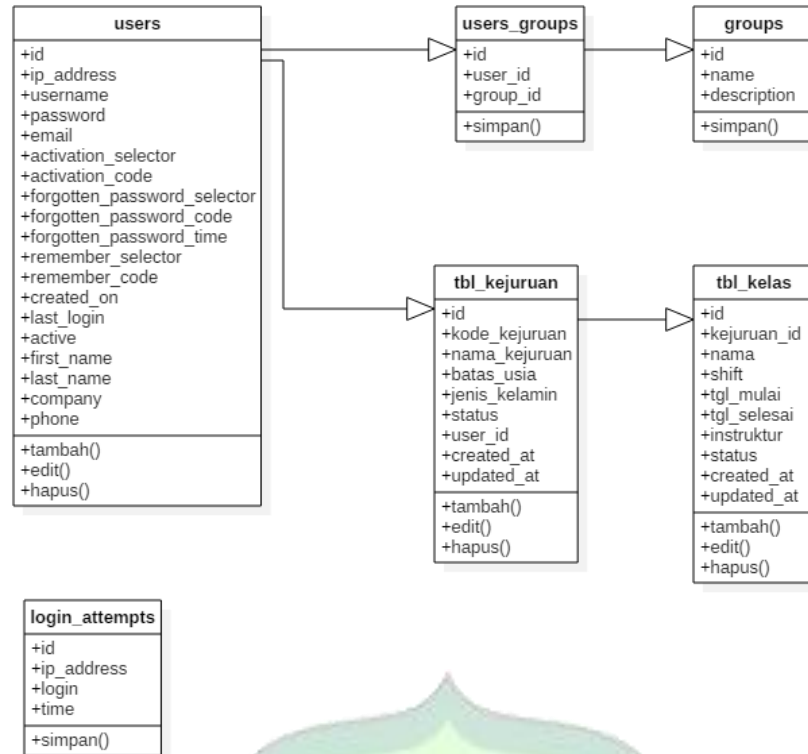


Gambar 4. 7 Activity Diagram Admin Laporan

c. Class Diagram

Pada *class* diagram akan dijabarkan suatu tabel – tabel yang terlibat terhadap aplikasi atau sistem. Tabel – tabel tersebut berada pada suatu database (sql). Yang merupakan suatu hal yang dapat disebut sebagai wadah atau tempat bernaung tabel – tabel tersebut yangmana diperlukan suatu aplikasi atau sistem yang perlu atau memerlukan tempat yang mewadahi arus informasi yang

mempengaruhi jalannya aplikasi atau sistem. Hal tersebut antara lain dapat diperhatikan melalui gambar dibawah ini.



Gambar 4. 8 Class Diagram aplikasi atau sistem

d. Sequence Diagram

Sequence diagram untuk aplikasi atau sistem yang akan dijabarkan dibagi menjadi beberapa macam yang mana pada hal tersebut objek memiliki interaksi dengan aktor dengan suatu penyebutan tersendiri sehingga timbul beberapa penjelasan. Hal tersebut akan disebutkan atau dijelaskan pada penjelasan – penjelasan berikut ini :

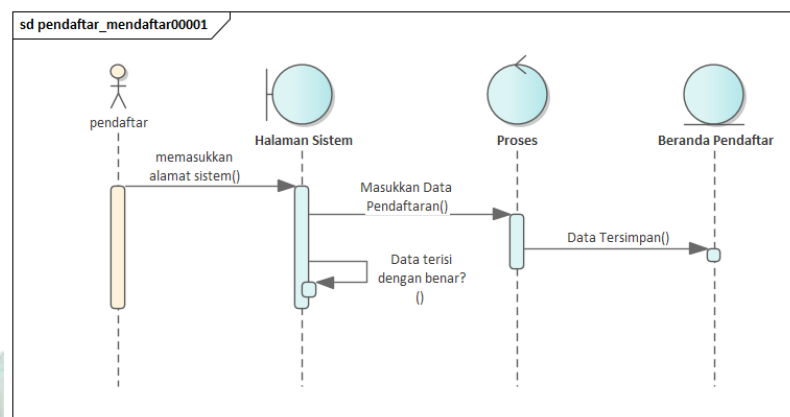
1. Sequence Diagram Pendaftar

Sequence diagram pendaftar terbagi menjadi hal – hal yang bersinggungan atau yang dapat disebut juga berinteraksi dengan hal antara pendaftar maupun objek lain yang termasuk pada suatu aplikasi atau sistem. Sehingga aktifitas tersebut dihasilkan

dari interaksi pendaftar atau suatu objek dengan aplikasi pada suatu fungsi atau halaman(tampilan di aplikasi/sistem). Hal tersebut antara lain adalah sebagai berikut :

a. *Sequence* Diagram Pendaftar Mendaftar

Sequence diagram pendaftar mendaftar, merupakan suatu *sequence* yang menggambarkan tentang arus interaksi pendaftar terhadap aplikasi atau sistem dalam hal pendaftaran. Dalam hal ini, pendaftar mengetik link yang berhubungan atau terhubung dengan aplikasi atau sistem, lalu setelah hal tersebut pendaftar melaksanakan proses pendaftaran melalui suatu halaman setelah halaman login atau bagian awal page. Pada gambar yang akan diperlihatkan proses langsung ke halaman pendaftaran. Hal tersebut dapat dilihat pada gambar dibawah ini.

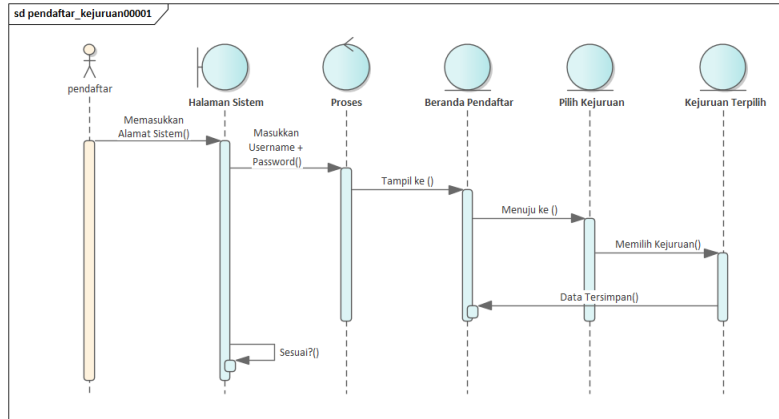


Gambar 4. 9 *Sequence* diagram pendaftar mendaftar

b. *Sequence* Diagram Pendaftar Kejuruan

Sequence diagram pendaftar kejuruan, merupakan suatu *sequence* yang menggambarkan tentang arus interaksi pendaftar terhadap aplikasi atau sistem dalam hal kejuruan. Kejuruan dapat dipilih ketika pendaftar telah mengetik dan membukan aplikasi atau sistem melalui suatu alamat atau

yang disebut juga sebagai *link*. Hal tersebut dapat dilihat pada gambar dibawah ini.



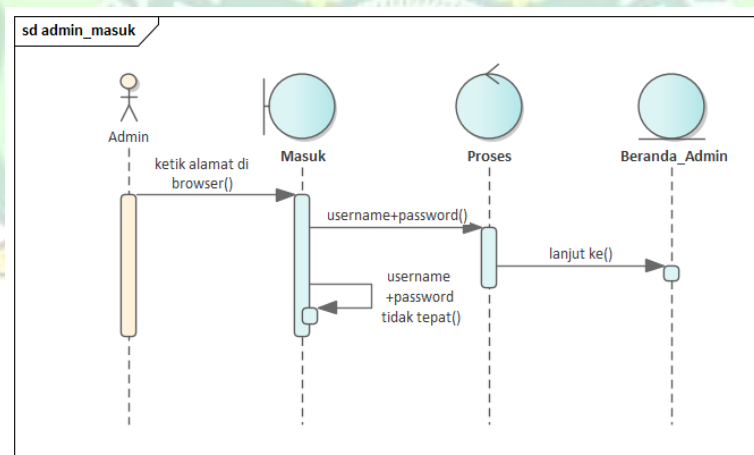
Gambar 4. 10 *Sequence* diagram pendafar kejuruan

2. *Sequence* Diagram Admin

Sequence diagram admin terbagi menjadi beberapa macam dikarenakan admin berinteraksi lebih dari satu fungsi atau dapat disebut objek dalam suatu aplikasi atau sistem. Hal tersebut dapat dilihat pada penjelasan – penjelasan berikut :

a. *Sequence* Diagram Admin Masuk

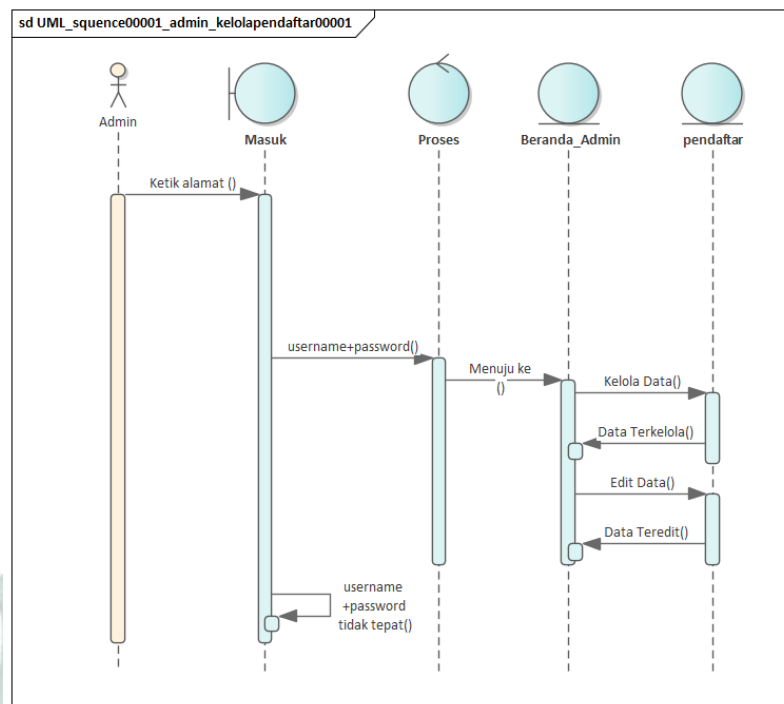
Sequence diagram admin masuk, merupakan *sequence* yang menggambarkan arus atau alur admin masuk ke aplikasi atau sistem. Hal tersebut dapat dilihat pada gambar dibawah ini.



Gambar 4. 11 *Sequence* diagram admin masuk

b. *Sequence* Diagram Admin Kelola Pendaftar

Sequence diagram admin kelola pendaftar, merupakan *sequence* yang menggambarkan arus atau alur admin mengelola pendaftar melalui aplikasi atau sistem. Penggambaran yang ada pada gambar meliputi admin perlu masuk ke sistem dan didalam ada suatu proses ke halaman atau tampilan admin setelah itu menuju ke kelola pendaftar. Yang dalam hal ini, data tersebut dapat lihat ataupun dikelola atau diedit oleh admin. Hal tersebut dapat dilihat pada gambar dibawah ini.

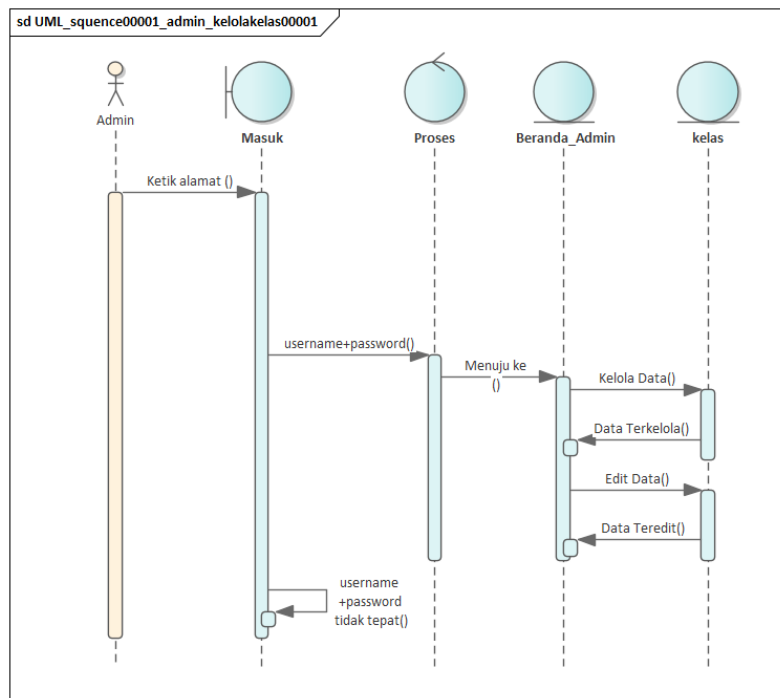


Gambar 4. 12 *Sequence* diagram admin kelola pendaftar

c. *Sequence* Diagram Admin Kelola Kelas

Sequence diagram admin kelola kelas, merupakan *sequence* yang menggambarkan arus atau alur admin mengelola pendaftar melalui aplikasi atau sistem. Penggambaran yang ada pada gambar meliputi admin perlu masuk ke sistem dan didalam ada suatu proses ke halaman atau tampilan admin

setelah itu menuju ke kelola kelas. Yangmana merupakan suatu fungsi atau manajemen fungsi yang digunakan admin terhadap kelas. Dalam hal ini, data tersebut dapat dilihat ataupun dikelola atau diedit oleh admin. Hal tersebut dapat dilihat pada gambar dibawah ini.

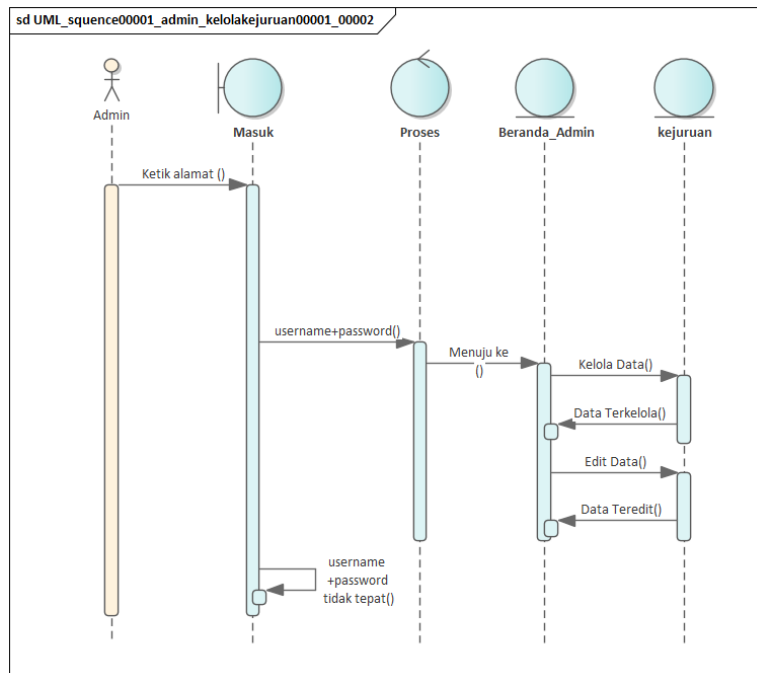


Gambar 4. 13 *Sequence* diagram admin kelola kelas

d. *Sequence* Diagram Admin Kelola Kejuruan

Sequence diagram admin kelola kejuruan, merupakan *sequence* yang menggambarkan arus atau alur admin mengelola kejuruan melalui aplikasi atau sistem. Penggambaran yang ada pada gambar meliputi admin perlu masuk ke sistem dan didalam ada suatu proses ke halaman atau tampilan admin setelah itu menuju ke kelola kejuruan. Dalam hal ini, data tersebut dapat dilihat ataupun dikelola atau diedit oleh admin. Sehingga dapat dikatakan admin dapat mengelola data tersebut.

Hal tersebut dapat dilihat pada gambar dibawah ini.

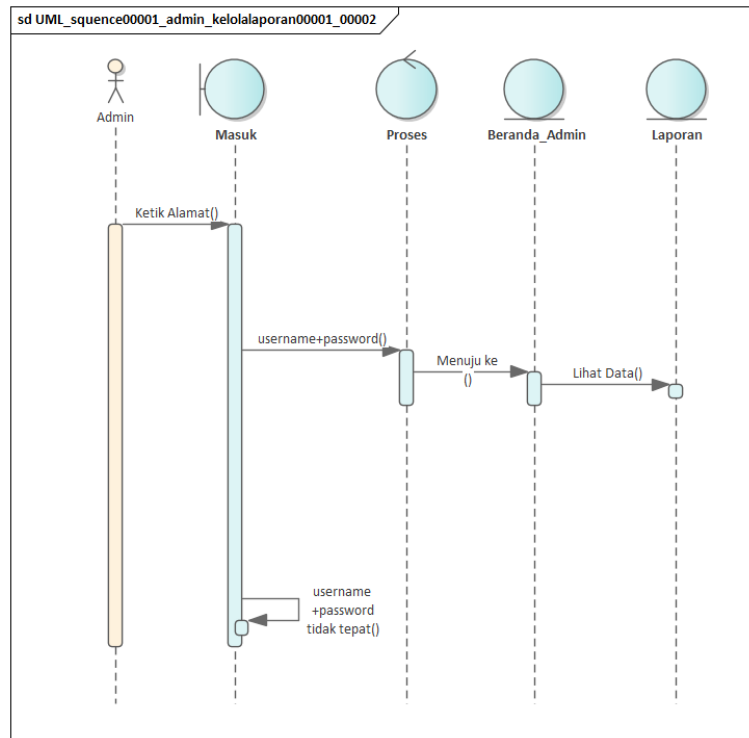


Gambar 4. 14 *Sequence* diagram admin kelola kejuruan

e. *Sequence* Diagram Admin Kelola Laporan

Sequence diagram admin kelola laporan, merupakan *sequence* yang menggambarkan arus atau alur admin mengelola kejuruan melalui aplikasi atau sistem. Penggambaran yang ada pada gambar meliputi admin perlu masuk ke sistem dan didalam ada suatu proses ke halaman atau tampilan admin setelah itu menuju ke kelola laporan. Pada hal ini, admin akan dapat melihat data – data atau laporan dari data pendaftar maupun data peserta yang telah tersimpan ataupun tersusun oleh aplikasi atau sistem. Admin dapat mengecek data – data tersebut pada fungsional atau halaman laporan ini. Sehingga data – data dari beberapa pendaftar dapat dipantau oleh admin.

Dalam Hal tersebut akan dapat dilihat atau diperhatikan pada pembahasan yang berupa gambar dibawah ini.



Gambar 4. 15 *Sequence* diagram admin kelola laporan

4.1.3. *Build System* (Membangun Sistem)

Pada pembangunan sistem atau aplikasi, merupakan suatu kegiatan atau aktivitas yang tergolong ke dalam suatu tahap pembangunan aplikasi atau sistem yang akan dikerjakan. Hal tersebut antara lain terbagi menjadi beberapa macam penjelasan. Dalam hal tersebut, penjelasan - penjelasannya antara lain adalah sebagai berikut :

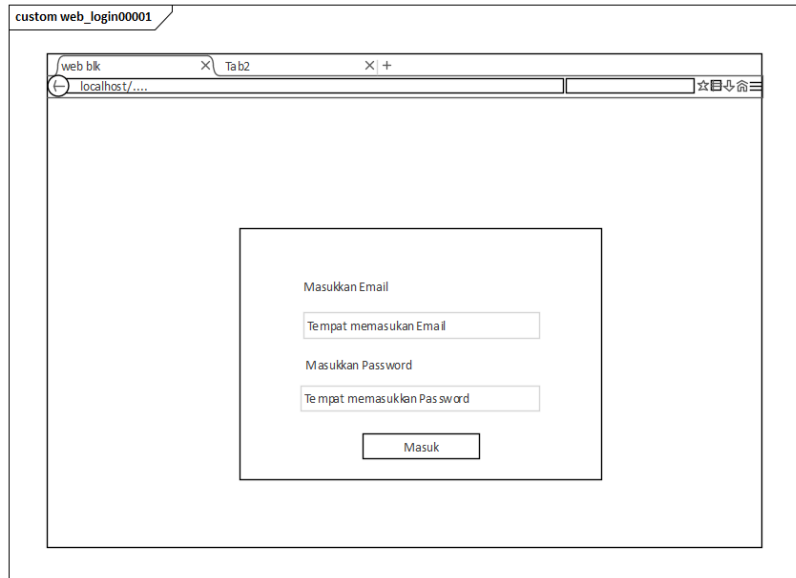
a. Perancangan *Interface* (Tampilan)

Pada perancangan *interface* (tampilan) dari aplikasi merupakan suatu proses (tahap) dengan maksud memperkirakan bentuk aplikasi (sistem) yang akan dibuat. Hal tersebut antara lain adalah :

1. Perancangan Tampilan *Login*

Perancangan tampilan *login*, merupakan perancangan bentuk kasar suatu tampilan yang dimaksudkan untuk memberi suatu fungsi pembeda antar pengguna yang akan

menggunakan aplikasi atau sistem. Perancangan tampilan *login* (masuk) dapat dilihat pada gambar berikut.

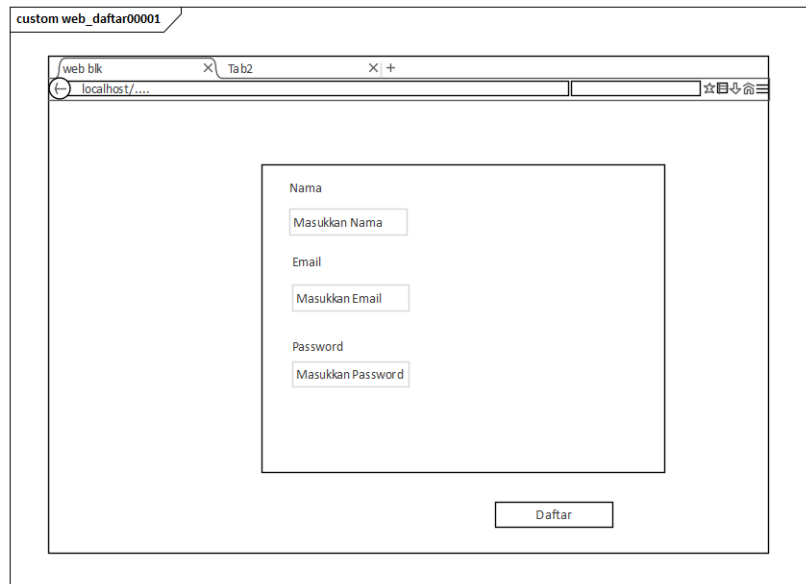


Gambar 4. 16 Perancangan Tampilan *login* (masuk)

2. Perancangan Tampilan Daftar

Perancangan tampilan daftar, merupakan perancangan bentuk kasar suatu tampilan yang dimaksudkan untuk memberi suatu bentuk tampilan yang memiliki suatu isian yang didalam hal tersebut perlu diisi hal – hal yang diperlukan untuk mendaftar ke aplikasi atau sistem. Hal – hal tersebut dapat berupa nama seseorang, nama pengguna, alamat pesan (*e-mail*), alamat seseorang, tanggal lahir seseorang, kata sandi (*password*) yang dipilih atau dipikirkan oleh orang yang mendaftar, maupun isian – isian lainnya yang masih terhubung dalam ruang lingkup data yang diperlukan dalam pendaftaran ke aplikasi atau sistem. Dalam hal tersebut tentunya ada beberapa macam bentuk. Salah satu bentuk tersebut akan dijelaskan pada suatu penggambaran.

Berikut ini merupakan contoh gambaran kasar dari perancangan tampilan daftar yang akan diperlihatkan pada gambar berikut.

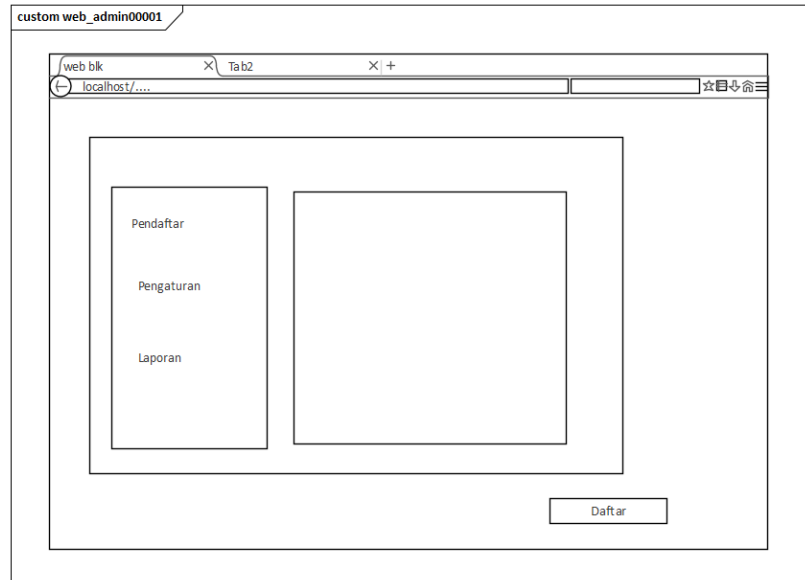


Gambar 4. 17 Perancangan tampilan daftar

3. Perancangan Tampilan Admin

Perancangan tampilan daftar, merupakan perancangan bentuk kasar suatu tampilan yang dimaksudkan untuk membuat rancangan suatu bentuk tampilan yang ada pada *interface* atau halaman yang dapat dimasuki oleh admin. Seperti yang secara umum ketahui, pengguna yang tergolong pihak admin atau yang dapat disebut juga pihak administrator (admin) merupakan suatu pihak yang dapat bertugas mengelola atau yang dapat disebut juga mengatur suatu aplikasi atau sistem yang bersangkutan sesuai dengan fitur atau fungsi yang dimiliki atau didapatkan oleh pihak admin terhadap aplikasi atau sistem yang bersangkutan. Sehingga melalui perancangan penggambaran halaman (tampilan) admin, peneliti memiliki suatu gambaran seperti apa tampilan atau halaman (*interface*) admin yang diteliti oleh peneliti. Perancangan tampilan admin

ini antara lain dapat dilihat dan sisimak pada gambar berikut ini.



Gambar 4. 18 Perancangan tampilan daftar

b. Perancangan Database

Database merupakan suatu tempat yang mencakup atau dapat disebut sebagai sesuatu yang mawadahi tabel – tabel yang dapat menyimpan suatu data yang terhubung dengan suatu aplikasi atau sistem. Pada suatu database terdapat tabel – tabel, dalam hal ini tabel – tabel yang ada pada aplikasi atau sistem yang dibahas peneliti antara lain adalah :

1. Tabel *Groups*

Nama Tabel : groups

Tabel *groups* memiliki struktur tabel yang antara lain adalah sebagai berikut.

Tabel 4. 3 Tabel *groups*

No	Nama	Jenis
1	id	mediumint(8)
2	name	varchar(20)
3	description	varchar(100)

2. Tabel *Login_attempts*

Nama Tabel : *login_attempts*

Tabel *login_attempts* memiliki struktur tabel yang antara lain adalah sebagai berikut.

Tabel 4. 4 Tabel *login_attempts*

No	Nama	Jenis
1	id	int(11)
2	ip_address	varchar(45)
3	login	varchar(100)
4	time	int(11)

3. Tabel Kejuruan

Nama Tabel : *tbl_kejuruan*

Tabel *tbl_kejuruan* memiliki struktur tabel yang antara lain adalah sebagai berikut.

Tabel 4. 5 Tabel *tbl_kejuruan*

No	Nama	Jenis
1	id	int(11)
2	kode_kejuruan	varchar(255)
3	nama_kejuruan	varchar(255)
4	batas_usia	int(11)
5	jenis_kelamin	varchar(1)
6	status	varchar(12)
7	user_id	int(10)

8	created_at	datetime
9	updated_at	datetime

4. Tabel Kelas

Nama Tabel : tbl_kelas

Tabel tbl_kelas memiliki struktur tabel yang antara lain adalah sebagai berikut.

Tabel 4. 6 Tabel tbl_kelas

No	Nama	Jenis
1	id	int(11)
2	kejuruan_id	int(11)
3	nama	varchar(255)
4	shift	varchar(255)
5	tgl_mulai	date
6	tgl_selesai	date
7	instruktur	varchar(255)
8	status	varchar(255)
9	created_at	datetime
10	update_at	datetime

5. Tabel *Users*

Nama Tabel : *users*

Tabel *users* memiliki struktur tabel yang antara lain adalah sebagai berikut.

Tabel 4. 7 Tabel *users*

No	Nama	Jenis
1	id	int(11)
2	ip_address	varchar(45)
3	username	varchar(100)
4	password	varchar(255)

5	email	varchar(254)
6	activation_selector	varchar(255)
7	activation_code	varchar(255)
8	forgotten_password_selector	varchar(255)
9	forgotten_password_code	varchar(255)
10	forgotten_password_time	int(11)
11	remember_selector	varchar(255)
12	remember_code	varchar(255)
13	created_on	int(11)
14	last_login	int(11)
15	active	tinyint(1)
16	first_name	varchar(50)
17	last_name	varchar(50)
18	company	varchar(100)
19	phone	varchar(20)

6. Tabel *Users_groups*

Nama Tabel : *users_groups*

Tabel *users_groups* memiliki struktur tabel yang antara lain adalah sebagai berikut.

Tabel 4. 8 Tabel *users_groups*

No	Nama	Jenis
1	id	int(11)
2	user_id	int(11)
3	group_id	mediumint(8)

c. *Script* Aplikasi (Sistem)

Script atau yang dapat disebut juga *source code* merupakan suatu karakter yang ada pada bahasa pemrograman dan digunakan pada fungsionalitas atau fungsi pada suatu aplikasi atau sistem. Sehingga

aplikasi atau sistem memiliki suatu fungsi. Selain sebagai fungsi *script* juga merupakan suatu karakter yang dapat membangun suatu tampilan seperti yang ada pada pemrograman dengan basis seperti *web*, *php*, *html*, ataupun sejenisnya.

Script yang perlu dipergunakan pada aplikasi atau sistem yang akan dijabarkan menjadi beberapa macam yang antara lain adalah sebagai berikut:

1. *Script login*

Script login atau yang dapat disebut juga *source code* yang digunakan untuk menghasilkan suatu fungsi login pada aplikasi atau sistem, merupakan suatu karakter yang diperlukan agar fungsi masuk ke aplikasi atau sistem dapat dijalankan.

Script login yang terdapat pada aplikasi atau sistem yang dibahas peneliti antara lain akan diperlihatkan melalui gambar berikut ini.

```
8 public function index()
9 {
10     if ($this->input->post()) {
11         $this->load->library('form_validation');
12         $this->form_validation->set_rules('email', 'Email', 'required');
13         $this->form_validation->set_rules('remember', 'Remember me', 'integer');
14         if ($this->form_validation->run()==TRUE) {
15             $remember = (bool) $this->input->post('remember');
16             if ($this->ion_auth->login($this->input->post('email'), $this->input->post('password'), $remember)) {
17                 redirect('admin', 'refresh');
18             } else {
19                 $this->session->flashdata('message', $this->ion_auth->errors());
20                 redirect('/', 'refresh');
21             }
22         }
23     }
24     if (!$this->ion_auth->logged_in()) {
25         $data = array('title'=>$this->config->item('app_name'),
26                     'content'=>'login');
27         return $this->load->view('layouts/front', $data, FALSE);
28     } else {
29         if ($this->ion_auth->in_group('admin')) {
30             redirect('admin', 'refresh');
31         } elseif ($this->ion_auth->in_group('members')) {
32             redirect('members', 'refresh');
33         } else {
34             redirect('login', 'refresh');
35         }
36     }
37 }
38 }
```

Gambar 4. 19 *Script login*

2. *Script daftar*

Script daftar merupakan suatu *script* atau *source code* yang memiliki peran terhadap fitur daftar atau penyimpanan data pendaftar. *Script* tersebut merupakan suatu kumpulan beberapa karakter yang tergolong kedalam suatu bahasa pemrograman

yang memiliki fungsi sistem data ke tabel yang berada dalam suatu database yang terhubung ke suatu aplikasi atau sistem.

Script daftar tersebut dapat dilihat melalui gambar berikut ini.

```
10 if ($this->input->post()) {
11     $this->load->library('form_validation');
12     $this->load->helper(array('security'));
13     $this->form_validation->set_rules('first_name', 'Nama Depan', 'trim|required|xss_clean');
14     $this->form_validation->set_rules('last_name', 'Nama Belakang', 'trim|required|xss_clean');
15     $this->form_validation->set_rules('phone', 'No Tlp', 'trim|required|xss_clean');
16     $this->form_validation->set_rules('email', 'Email', 'trim|required|xss_clean');
17     $this->form_validation->set_rules('password', 'Password', 'trim|required|xss_clean');
18     if ($this->form_validation->run() == false) {
19         $data = array('content' => 'home/formregistrasi',
20                     'message' => validation_errors());
21         $this->load->view('templates/template-home', $data);
22     } else {
23         $password = $this->input->post('password');
24         $email = $this->input->post('email');
25         $additional_data = array(
26             'first_name' => $this->input->post('first_name'),
27             'last_name' => $this->input->post('last_name'),
28             'phone' => $this->input->post('phone')
29         );
30         if (!$this->ion_auth->email_check($email)) //cek email sudah terdaftar apa belum
31         {
32             $group = array('2'); // Sets user ke grup member
33             if ($this->ion_auth->register($username, $password, $email, $additional_data, $group)) {
34                 redirect('login');
35             } else {
36                 $data = array('content' => 'home/formregistrasi',
37                             'message' => 'Proses registrasi gagal diproses');
38                 $this->load->view('templates/template-home', $data);
39             }
40         } else {
41             $data = array('content' => 'home/formregistrasi',
42                         'message' => 'Email sudah terdaftar');
43             $this->load->view('templates/template-home', $data);
44         }
45     }
46 }
```

Gambar 4. 20 *Script* daftar

3. *Script* Tambah Data

Script tambah data merupakan suatu *script* atau *source code* yang memiliki fungsi dalam menambah atau menyimpan data ke tabel yang ada di database. *Script* tambah data tersebut dapat dilihat melalui gambar berikut ini.

```
// Add a new item
public function add()
{
    if (!$this->input->is_ajax_request()) {
        show_404();
    } else {
        // validasi input
        $this->form_validation->set_rules('kode_kejuruan', 'Kode Kejuruan', 'trim|required');
        $this->form_validation->set_rules('nama_kejuruan', 'Nama Kejuruan', 'trim|required');
        $this->form_validation->set_rules('batas_usia', 'Batas Usia', 'trim|required');
        $this->form_validation->set_rules('jenis_kelamin', 'Jenis Kelamin', 'trim|required');
        if ($this->form_validation->run() == false) // cek kalo validasi gagal
        {
            $status = 'error';
            $msg = validation_errors();
        } else {
            if ($this->kejuruan->cekkode($this->input->post('kode_kejuruan')) > 0) {
                // cek kode kejuruan, kalo sudah ada maka data tidak diinput
                $status = 'error';
                $msg = 'Kode kejuruan sudah ada, coba yang lain';
            } else {
                if ($this->kejuruan->setkejuruan()) // cek input kejuruan baru
                {
                    $status = 'success';
                    $msg = 'Input kejuruan berhasil';
                } else // kalo gagal input kejuruan baru
                {
                    $status = 'error';
                    $msg = 'Input kejuruan gagal';
                }
            }
        }
    }
    $this->output->set_content_type('application/json')->set_output(json_encode(['status' => $status, 'msg' => $msg]));
}
```

Gambar 4. 21 *Script* tambah data

4. Script Tampil Data

Script tampil data merupakan suatu *script* atau *source code* dengan fungsi menampilkan data yang tersimpan di tabel yang ada di database. *Script* tampil data tersebut dapat dilihat melalui gambar berikut ini.

```
71
72 public function show($id = NULL) {
73     if (!$this->input->is_ajax_request()) {
74         show_404();
75     } else {
76         $data = $this->kejuruan->getbyid($id)->row();// data kejuruan
77         if (isset($data)) {
78             $status = 'success';
79             $msg = 'Data kejuruan ditemukan';
80         } else {
81             $status = 'error';
82             $msg = 'Data kejuruan tidak ditemukan';
83         }
84         // $data = $id;
85         $respon = [
86             'status' => $status,
87             'msg' => $msg,
88             'content' => $data
89         ];
90         $this->output->set_content_type('application/json')->set_output(json_encode($respon));
91     }
92 }
93
```

Gambar 4. 22 Script tampil data

5. Script Edit Data

Script edit data, merupakan suatu *script* (source code) yang berfungsi pada pengeditan data yang ada ditabel *database*. *Script* edit data tersebut dapat dilihat melalui gambar berikut ini.

```
94 //Update one item
95 public function update($id = NULL) {
96     if (!$this->input->is_ajax_request()) {
97         show_404();
98     } else {
99         // validasi input
100         $this->form_validation->set_rules('kode_Kejuruan', 'Kode Kejuruan', 'trim|required');
101         $this->form_validation->set_rules('nama_Kejuruan', 'Nama Kejuruan', 'trim|required');
102         $this->form_validation->set_rules('batas_usia', 'Batas Usia', 'trim|required');
103         $this->form_validation->set_rules('jenis_kelamin', 'Jenis Kelamin', 'trim|required');
104         if ($this->form_validation->run() == false) { // cek kalo validasi gagal
105             $status = 'error';
106             $msg = validation_errors();
107         } else {
108             if ($this->kejuruan->cekkodeupdate($this->input->post('kode_Kejuruan'), $id) > 0) {
109                 // cek kode kejuruan, kalo sudah ada maka data tidak diupdate
110                 $status = 'error';
111                 $msg = 'Kode kejuruan sudah ada, coba yang lain!';
112             } else {
113                 if ($this->kejuruan->updatebyid($id) { // update data kejuruan
114                     $status = 'success';
115                     $msg = 'Kejuruan berhasil diupdate!';
116                 } else { // kalo gagal update kejuruan baru
117                     $status = 'error';
118                     $msg = 'Kejuruan gagal diupdate!';
119                 }
120             }
121         }
122         $this->output->set_content_type('application/json')->set_output(json_encode(['status' => $status, 'msg' => $msg]));
123     }
124 }
125
126
```

Gambar 4. 23 Script edit data

6. Script Hapus Data

Script hapus data merupakan suatu *script* (source code) yang berfungsi pada penghapusan data data yang ada ditabel

database. Script hapus data tersebut dapat dilihat melalui gambar berikut ini.

```
127 //Delete one item
128 public function delete()
129 {
130     if (!$this->input->is_ajax_request()) {
131         show_404();
132     } else {
133         if ($this->kejuruan->kelaskejuruan($this->input->post('id')) > 0) {
134             // cek kalo ada kelas yang menggunakan kejuruan id ini
135             $status = 'error';
136             $msg = 'Hapus dulu kelas pada kejuruan ini';
137         } else {
138             if ($this->kejuruan->deletebyid($this->input->post('id')) // hapus data kejuruan
139                 $status = 'success';
140                 $msg = 'Kejuruan berhasil dihapus';
141             } else // kalo gagal hapus kejuruan baru
142                 $status = 'error';
143                 $msg = 'Kejuruan gagal dihapus';
144         }
145     }
146     $this->output->set_content_type('application/json')->set_output(json_encode(['status' => $status, 'msg' => $msg]));
147 }
148 }
149 }
150 }
```

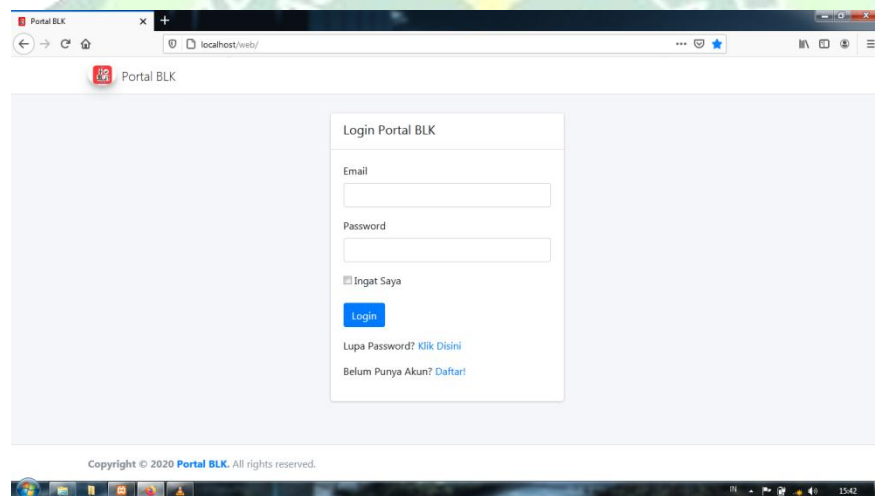
Gambar 4. 24 Script hapus data

4.1.4. Implementation (Implementasi)

Pada Implementasi sistem atau aplikasi, akan ditampilkan suatu gambar yang memperlihatkan aplikasi atau sistem yang dimaksud oleh peneliti yang antara adalah sebagai berikut :

a. *Login Aplikasi (Sistem)*

Halaman *login* atau halaman masuk yang ada pada aplikasi atau sistem merupakan suatu halaman yang menampilkan suatu karakter – karakter seperti *username* dan *password* yang digunakan untuk masuk menggunakan tombol masuk (*login*). Halaman tersebut diperlihatkan antara lain pada gambar berikut ini.



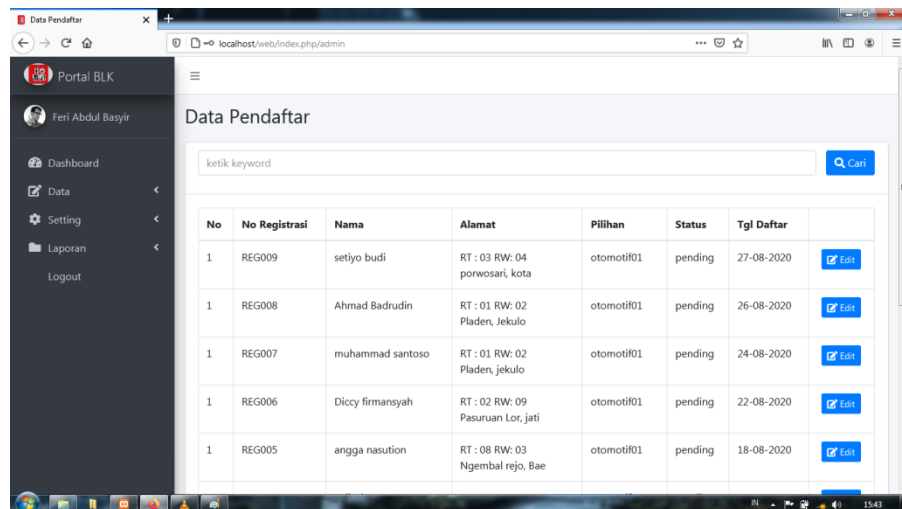
Gambar 4. 25 Halaman login aplikasi atau sistem

b. Halaman Admin

Halaman admin merupakan halaman yang digunakan oleh admin untuk melaksanakan pengelolaan aplikasi atau sistem. Halaman ini terbagi menjadi beberapa penjelasan yang antara lain adalah sebagai berikut :

1. Halaman Data Pendaftar

Halaman data pendaftar merupakan suatu halaman yang menampilkan suatu halaman yang digunakan dalam mengelola pendaftar. Yangmana halaman tersebut berarti merupakan halaman yang digunakan oleh admin untuk melaksanakan suatu tugas pengelolaan data – data pendaftar yang telah mendaftar atau mendaftar di aplikasi atau sistem. Halaman tersebut dapat dilihat pada gambar berikut.



No	No Registrasi	Nama	Alamat	Pilihan	Status	Tgl Daftar	
1	REG009	setyo budi	RT : 03 RW: 04 porwosari, kota	otomotif01	pending	27-08-2020	Edit
1	REG008	Ahmad Badrudin	RT : 01 RW: 02 Pladen, Jekulo	otomotif01	pending	26-08-2020	Edit
1	REG007	muhammad santoso	RT : 01 RW: 02 Pladen, jekulo	otomotif01	pending	24-08-2020	Edit
1	REG006	Dicky firmansyah	RT : 02 RW: 09 Pasuruan Lor, jati	otomotif01	pending	22-08-2020	Edit
1	REG005	angga nasution	RT : 08 RW: 03 Ngemal rejo, Bae	otomotif01	pending	18-08-2020	Edit

Gambar 4. 26 Halaman data pendaftar

2. Halaman Data Seleksi Pendaftar

Halaman data seleksi pendaftar merupakan suatu halaman yang menampilkan suatu halaman yang digunakan dalam mengelola pendaftar yang terseleksi. Pada halaman ini, admin

dapat mengelola atau mengatur data – data peserta yang terseleksi. Halaman tersebut dapat dilihat pada gambar berikut.

No	No Registrasi	Nama	Alamat	Pilihan	Status
1	REG009	setiyo budi	RT : 03 RW: 04 porwosari, kota	otomotif01 - otomotif 01	lolos
1	REG007	muhammad santoso	RT : 01 RW: 02 Pladen, jekulo	otomotif01 - otomotif 01	lolos
1	REG006	Diccy firmansyah	RT : 02 RW: 09 Pasuruan Lor, jati	otomotif01 - otomotif 01	lolos
1	REG005	angga nasution	RT : 08 RW: 03 Ngembal rejo, Bae	otomotif01 - otomotif 01	lolos
1	REG004	yulisukarno	RT : 03 RW: 02 Papringan, kaliwungu	otomotif01 - otomotif 01	lolos

Gambar 4. 27 Halaman data seleksi pendaftar

3. Halaman Data Peserta Pelatihan

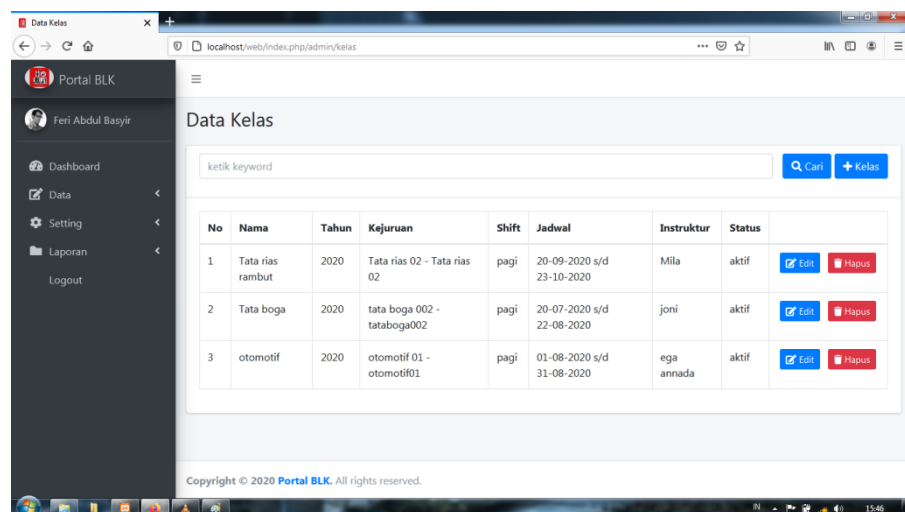
Halaman data peserta pelatihan merupakan suatu halaman yang menampilkan suatu halaman yang digunakan dalam mengelola peserta pelatihan. Pada halaman ini, admin dapat mengelola atau mengatur data – data peserta pelatihan yang sudah melewati proses seleksi. Halaman tersebut dapat dilihat pada gambar berikut.

No	No Registrasi	Nama	Alamat	Kelas	Kejuruan	Status
1	REG009	setiyo budi	RT/RW: 03/04 porwosari, kota	otomotif 01-08-2020 s/d 31-08-2020	otomotif 01 (otomotif01)	pending
1	REG007	muhammad santoso	RT/RW: 01/02 Pladen, jekulo	otomotif 01-08-2020 s/d 31-08-2020	otomotif 01 (otomotif01)	pending
1	REG006	Diccy firmansyah	RT/RW: 02/09 Pasuruan Lor, jati	otomotif 01-08-2020 s/d 31-08-2020	otomotif 01 (otomotif01)	pending
1	REG005	angga nasution	RT/RW: 08/03 Ngembal rejo,	otomotif 01-08-2020 s/d	otomotif 01 (otomotif01)	pending

Gambar 4. 28 Halaman data peserta pelatihan

4. Halaman *Setting* Kelas

Halaman *setting* kelas merupakan suatu halaman yang menampilkan suatu halaman yang digunakan dalam mengelola penyetaraan (*setting*) terhadap kelas. Halaman ini dapat diakses oleh admin, setelah admin masuk ke aplikasi menggunakan halaman masuk. Setelah masuk, admin menekan pilihan *setting*, lalu pilih data kelas (*kelas*) agar halaman *setting* data kelas tampil. Setelah itu admin dapat mengelola atau mengatur data kelas agar sesuai. Halaman tersebut dapat dilihat pada gambar berikut.



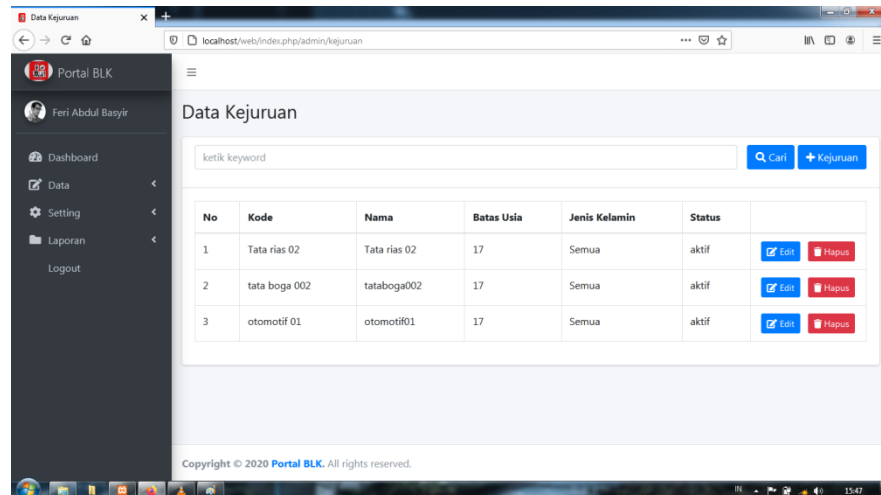
No	Nama	Tahun	Kejuruan	Shift	Jadwal	Instruktur	Status	
1	Tata rias rambut	2020	Tata rias 02 - Tata rias 02	pagi	20-09-2020 s/d 23-10-2020	Mila	aktif	Edit Hapus
2	Tata boga	2020	tata boga 002 - tataboga002	pagi	20-07-2020 s/d 22-08-2020	joni	aktif	Edit Hapus
3	otomotif	2020	otomotif 01 - otomotif01	pagi	01-08-2020 s/d 31-08-2020	ega annada	aktif	Edit Hapus

Gambar 4. 29 Halaman *setting* kelas

5. Halaman *Setting* Kejuruan

Halaman *setting* kejuruan merupakan suatu halaman yang menampilkan suatu halaman yang digunakan dalam mengelola penyetaraan (*setting*) terhadap kejuruan. Halaman ini dapat diakses oleh admin, setelah admin masuk ke aplikasi menggunakan halaman masuk. Setelah masuk, admin menekan pilihan *setting*, lalu pilih data kejuruan (*kejuruan*) agar halaman *setting* data kejuruan tampil. Setelah itu admin dapat mengelola

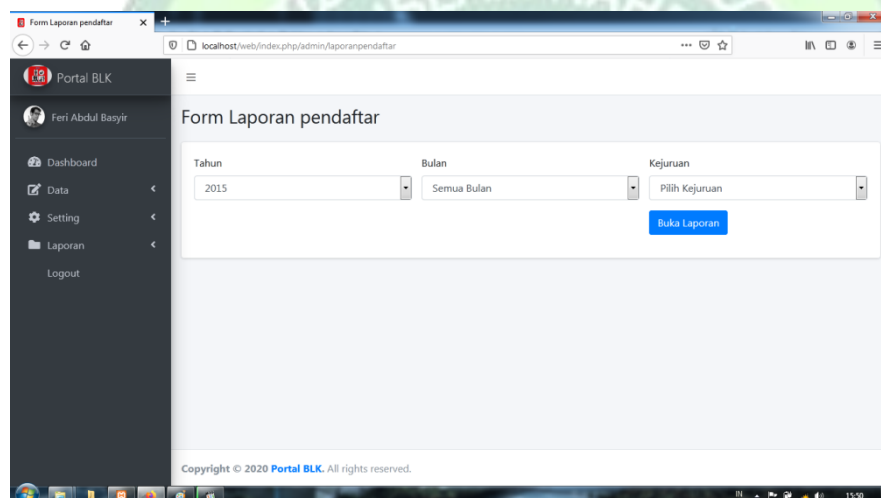
atau mengatur data kejuruan agar sesuai. Halaman tersebut dapat dilihat pada gambar berikut.



Gambar 4. 30 Halaman *setting* kejuruan

6. Halaman Laporan Pendaftar

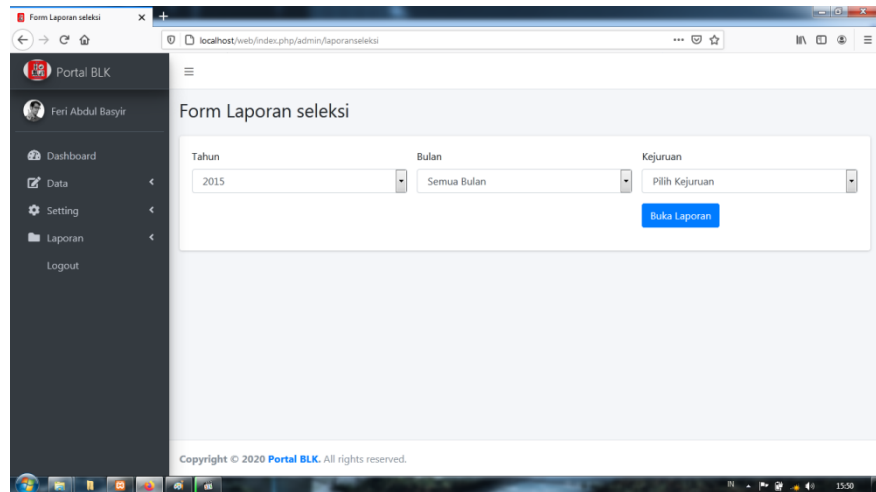
Halaman laporan pendaftar merupakan suatu halaman yang menampilkan suatu halaman yang digunakan dalam mengelola laporan pendaftar. Disini admin dapat memantau atau melihat laporan pendaftar yang terdaftar. Halaman tersebut dapat dilihat pada gambar berikut.



Gambar 4. 31 Halaman laporan pendaftar

7. Halaman Laporan Seleksi Pendaftar

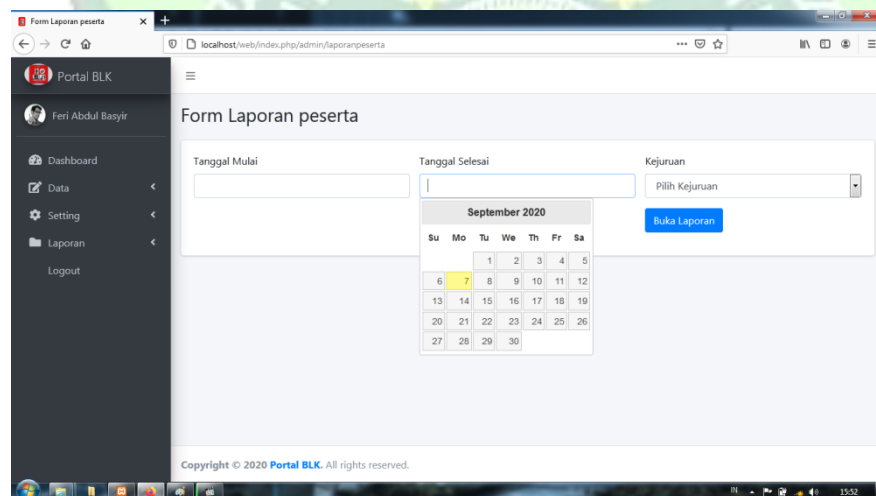
Halaman laporan seleksi pendaftar merupakan suatu halaman yang menampilkan suatu halaman yang digunakan dalam mengelola laporan seleksi pendaftar. Halaman tersebut dapat dilihat pada gambar berikut.



Gambar 4. 32 Halaman laporan seleksi

8. Halaman Laporan Peserta Pelatihan

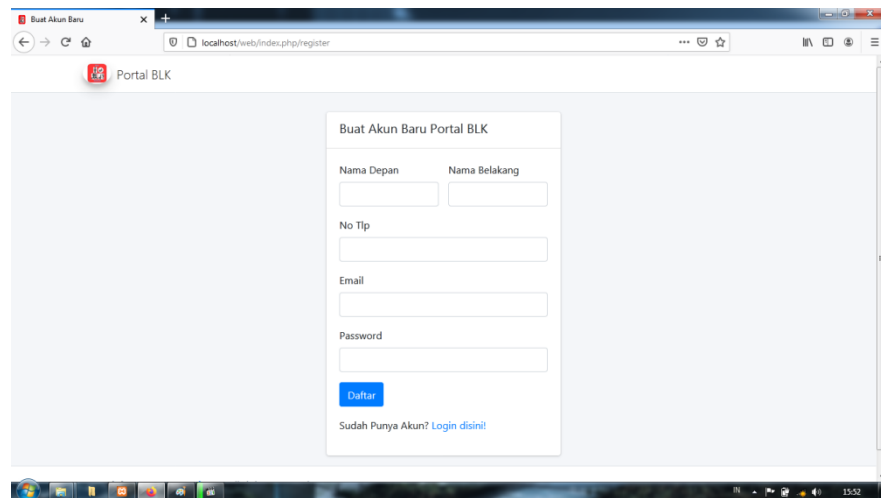
Halaman laporan peserta pelatihan merupakan suatu halaman yang menampilkan suatu halaman yang digunakan dalam mengelola laporan peserta pelatihan. Halaman tersebut dapat dilihat pada gambar berikut.



Gambar 4. 33 Halaman laporan peserta pelatihan

c. Halaman Pendaftaran

Halaman pendaftaran merupakan halaman yang digunakan oleh pendaftar untuk melaksanakan suatu proses pendaftaran ke aplikasi atau sistem. Halaman tersebut diperlihatkan antara lain pada gambar berikut ini.



Gambar 4. 34 Halaman pendaftaran

4.2. Pengujian Metode

Pengujian Metode atau yang dapat disebut juga metode pengujian. Merupakan metode yang dipergunakan atau digunakan dalam mengecek (mengetes) suatu aplikasi atau sistem. Pengujian (pengetesan) aplikasi atau sistem yang dilaksanakan antara lain adalah :

a. *BlackBox Testing*

Black box testing merupakan metode pengujian dimana pengetesan program dilaksanakan dengan melihat secara langsung pada aplikasinya disaat aplikasi tersebut dijalankan tanpa mengetahui struktur programnya. Sehingga *blackbox testing* adalah suatu uji coba atau pengecekan ketika aplikasi atau sistem dijalankan.

Pada *blackbox testing*, terdapat suatu tabel pengecekan yang dimana setelah menjalankan aplikasi (sistem), tabel tersebut digunakan sebagai salah satu penentu apa saja indikator yang di cek

pada aplikasi atau sistem yang dijalankan. Untuk tabelnya akan diperlihatkan pada tabel berikut.

Tabel 4. 9 *Blackbox Testing* Aplikasi

No	Indikator	Hasil Yang diharapkan	Hasil	Keterangan
1	Tampilan menyesuaikan terhadap penggunaan dan kinerja yang diperlukan	Tampilan sesuai dengan fungsi	sesuai	Dapat dilihat saat menjalankan aplikasi
2	Tampilan tidak memberatkan aplikasi (sistem)	Aplikasi atau sistem berjalan dengan sedikit <i>lag</i>	sesuai	Perlu memperhatikan spesifikasi perangkat
3	Pengguna mudah memahami dan menggunakan aplikasi	Pengguna dapat menjalankan aplikasi atau sistem	sesuai	Perlu suatu njeabaran atau mungkin sosialisasi ke pengguna
4	Fungsi Pendaftaran pada aplikasi (sistem) dapat digunakan	Data yang dimasukkan tersimpan	sesuai	Data yang dimasukkan sesuai dengan yang dicantumkan
5	Fungsi CRUD (<i>create, read, update, delete</i>) pada aplikasi atau	Fungsi CRUD pada aplikasi (ssitem) seperti tambah, edit,	sesuai	Data yang tampil menyesuaikan dengan data

	sistem	hapus, dan tampil data berfungsi		yang tersimpan ditabel <i>database</i>
6	Tombol pada aplikasi (sistem) berfungsi sesuai dengan perintah	Tombol dapat digunakan dan berfungsi	sesuai	Dapat di cek ketika aplikasi (sistem) dijalankan
7	Informasi tersimpan di data aplikasi (sistem)	Informasi yang tampil sesuai	sesuai	Menyesuaikan dengan <i>database</i> yang terhubung dengan aplikasi (sistem)
8	Penampil data menampilkan data yang sesuai dengan data yang tersimpan	Data Dapat ditampilkan	sesuai	Menyesuaikan dengan koneksi aplikasi dengan <i>database</i> ataupun tabel didalamnya
9	Fungsi masuk dan keluar (akun pengguna) di aplikasi (sistem)	Pengguna dapat masuk dan keluar dari aplikasi (sistem)	sesuai	Jika <i>username</i> (<i>e-mail</i>) yang digunakan sesuai dengan yang tersimpan di <i>database</i>
10	Bahasa yang digunakan adalah bahasa formal	Bahasa yang digunakan bahasa Baku atau Standar yang ada	sesuai	Dapat dilihat saat aplikasi (sistem) dijalankan

b. Pengujian BLK

Pengujian BLK yang dimaksud antara lain adalah suatu pengecekan aplikasi yang dilaksanakan atau dilakukan oleh pihak BLK (Bali Latihan Kerja). Pada pengecekan tersebut, suatu tabel digunakan untuk merangkum hasilnya. Sehingga hasil cek atau pengujian mendapat suatu rangkuman lain dalam bentuk suatu tabel. Tabel tersebut anatar lain adalah sebagai berikut.

Tabel 4. 10 Rangkum Pengujian BLK

No	Indikator	Nilai
A	Aspek Tampilan	
1	Tampilan mudah difahami	5
2	Tampilan tidak memberatkan aplikasi (sistem)	4
3	Tampilan menyesuaikan dengan penggunaan	5
B	Aspek Fungsi	
4	Fungsi akun pada aplikasi / sistem	5
5	Fungsi masuk / keluar aplikasi	5
6	Tombol di aplikasi (sistem) berfungsi seperti yang diharapkan	4
7	Data dapat tersimpan dan ditampilkan	5
8	Data yang disimpan dan ditampilkan menyesuaikan namun sama	4
C	Aspek Kinerja Aplikasi (Sistem)	
9	Waktu yang diperlukan pada pemrosesan sistem	4
D	Aspek Bahasa Aplikasi (Sistem)	
10	Penggunaan Bahasa Formal	5
	Jumlah	46

Untuk kategori dari skor yang dimasukkan oleh penguji, penjelasannya antara lain adalah sebagai berikut.

Tabel 4. 11 Penjelasan kategori untuk penguji

Kategori	Skor
Sangat Setuju	5
Setuju	4
Cukup Setuju	3
Tidak Setuju	2
Sangat Tidak Setuju	1

Untuk kategori dari persentase yang didapat dari penguji, penjelasannya antara lain adalah sebagai berikut.

Tabel 4. 12 Penjelasan kategori persentase penguji

No	Kategori	Skor dalam Persentase
1	Sangat layak	81% - 100%
2	Layak	61% - 80%
3	Cukup	41% - 60%
4	Tidak layak	21% - 40%
5	Sangat tidak layak	-21%

Sedangkan skala digunakan untuk menghitung menggunakan suatu rumus hasil rangkuman dan perhitungan tersebut yang antara lain adalah sebagai berikut.

$$\text{Persentase} = \frac{\text{Nilai (skor) yang didapat}}{\text{Nilai (skor) maksimal}} \times 100\%$$

$$\text{Persentase} = \frac{46}{50} \times 100\%$$

$$= 92\%$$

Dari beberapa proses sebelumnya, didapatkan suatu hasil atau yang dapat disebut juga rangkuman yang berupa tabel dengan penjelasan rangkuman yang antara lain sebagai berikut.

Tabel 4. 13 Rangkuman Hasil Pengujian BLK

No.	Penguji	Indikator	Skor Maksimal	Skor Didapat	Persentase (%)	Kategori	Keterangan Ahli
1.	Pihak BLK	10	50	46	92%	Sangat Layak	Aplikasi mudah difahami dalam melakukan pendaftaran

Dari Tabel diatas dapat disimpulkan bahwa menurut penguji, aplikasi (sistem) mendapat kategori sangat layak.

c. Pengujian Masyarakat (Respondensi Masyarakat)

Pengujian masyarakat adalah suatu pengecekan aplikasi yang dilaksanakan atau dilakukan oleh masyarakat sebagai responden. Pada pengecekan tersebut, suatu tabel digunakan untuk merangkum hasilnya. Sehingga hasil cek atau pengujian mendapat suatu rangkuman lain dalam bentuk suatu tabel. Tabel tersebut anatar lain adalah sebagai berikut.

Tabel 4. 14 Rangkuman pengujian masyarakat

No	Indikator	Total Nilai
A	Aspek Tampilan	
1	Tampilan mudah difahami	73
2	Tampilan tidak memberatkan aplikasi (sistem)	71
3	Tampilan menyesuaikan dengan penggunaan	67
B	Aspek Fungsi	
4	Fungsi akun pada aplikasi / sistem	69
5	Fungsi masuk / keluar aplikasi	69
6	Tombol di aplikasi (sistem) berfungsi seperti yang diharapkan	72
7	Data dapat tersimpan dan ditampilkan	73
8	Data yang disimpan dan ditampilkan menyesuaikan namun sama	72
C	Aspek Kinerja Aplikasi (Sistem)	
9	Waktu yang diperlukan pada pemrosesan sistem	65
D	Aspek Bahasa Aplikasi (Sistem)	
10	Penggunaan Bahasa Formal	68
	Jumlah	699

Untuk kategori dari skor yang dimasukkan oleh penguji, penjelasannya antara lain adalah sebagai berikut.

Tabel 4. 15 Penjelasan kategori untuk responden

Kategori	Skor
Sangat Setuju	5
Setuju	4
Cukup Setuju	3
Tidak Setuju	2
Sangat Tidak Setuju	1

Untuk kategori dari persentase yang didapat dari pengujian, penjelasannya antara lain adalah sebagai berikut.

Tabel 4. 16 Penjelasan kategori persentase responden

No	Kategori	Skor dalam Persentase
1	Sangat layak	81% - 100%
2	Layak	61% - 80%
3	Cukup	41% - 60%
4	Tidak layak	21% - 40%
5	Sangat tidak layak	-21%

Sedangkan skala *likert* untuk menghitung suatu rumus hasil rangkuman tersebut yang antara lain adalah sebagai berikut.

$$\text{Persentase} = \frac{\text{Nilai(skor) yang didapat}}{\text{Nilai (skor) maksimal}} \times 100\%$$

$$\begin{aligned} \text{Persentase} &= \frac{699}{750} \times 100\% \\ &= 93,2\% \end{aligned}$$

Dari beberapa proses sebelumnya, didapatkan suatu hasil atau yang dapat disebut juga rangkuman yang berupa tabel dengan penjelasan rangkuman yang antara lain sebagai berikut.

Tabel 4. 17 Rangkuman hasil pengujian masyarakat

No.	Total Responden	Indikator	Skor Maksimal	Skor Didapat	Persentase (%)	Kategori	Keterangan Ahli
1.	Masyarakat (15 orang)	10	750	699	93,2%	Sangat Layak	Aplikasi berjalan dengan baik

Dari Tabel diatas dapat disimpulkan bahwa menurut penguji, aplikasi (sistem) mendapat kategori sangat layak.

d. Pengujian Ahli Media

Pengujian ahli media merupakan suatu pengecekan aplikasi yang dilaksanakan atau dilakukan oleh pihak khusus yang menguji sistem atau aplikasi sebagai suatu media. Pada pengecekan tersebut, suatu tabel digunakan untuk merangkum hasilnya. Sehingga hasil cek atau pengujian mendapat suatu rangkuman lain dalam bentuk suatu tabel. Tabel tersebut anatar lain adalah sebagai berikut.

Tabel 4. 18 Rangkum pengujian Ahli Media

No	Indikator	Nilai
A	Aspek Tampilan	
1	Tampilan mudah difahami	5
2	Tampilan tidak memberatkan aplikasi (sistem)	4
3	Tampilan menyesuaikan dengan penggunaan	5
B	Aspek Fungsi	
4	Fungsi akun pada aplikasi / sistem	5
5	Fungsi masuk / keluar aplikasi	5

6	Tombol di aplikasi (sistem) berfungsi seperti yang diharapkan	4
7	Data dapat tersimpan dan ditampilkan	5
8	Data yang disimpan dan ditampilkan menyesuaikan namun sama	4
C	Aspek Kinerja Aplikasi (Sistem)	
9	Waktu yang diperlukan pada pemrosesan sistem	4
D	Aspek Bahasa Aplikasi (Sistem)	
10	Penggunaan Bahasa Formal	5
	Jumlah	46

Untuk kategori dari skor yang dimasukkan oleh penguji, penjelasannya antara lain adalah sebagai berikut.

Tabel 4. 19 Penjelasan kategori keterangan Ahli Media

Kategori	Skor
Sangat Setuju	5
Setuju	4
Cukup Setuju	3
Tidak Setuju	2
Sangat Tidak Setuju	1

Untuk kategori dari persentase yang didapat dari penguji, penjelasannya antara lain adalah sebagai berikut.

Tabel 4. 20 Penjelasan kategori persentase keterangan Ahli Media

No	Kategori	Skor dalam Persentase
1	Sangat layak	81% - 100%
2	Layak	61% - 80%

3	Cukup	41% - 60%
4	Tidak layak	21% - 40%
5	Sangat tidak layak	-21%

Sedangkan skala digunakan untuk menghitung menggunakan suatu rumus hasil rangkuman dan perhitungan tersebut yang antara lain adalah sebagai berikut.

$$\text{Persentase} = \frac{\text{Nilai (skor) yang didapat}}{\text{Nilai (skor) maksimal}} \times 100\%$$

$$\begin{aligned} \text{Persentase} &= \frac{42}{50} \times 100\% \\ &= 84\% \end{aligned}$$

Dari beberapa proses sebelumnya, didapatkan suatu hasil atau yang dapat disebut juga rangkuman yang berupa tabel dengan penjelasan rangkuman yang antara lain sebagai berikut.

Tabel 4. 21 Hasil Pengujian Ahli Media

No.	Penguji	Indikator	Skor Maksimal	Skor Didapat	Persentase (%)	Kategori	Keterangan Ahli
1.	Ahli Media	10	50	42	84%	Sangat Layak	Pengembangan aplikasi tetap perlu diperhatikan

Dari Tabel tersebut dapat disimpulkan bahwa menurut penguji, aplikasi (sistem) mendapat kategori sangat layak.

4.3. Pembahasan

Setelah melihat penjelasan – penjelasan sebelumnya. Sekiranya perlu dibahas tentang aplikasi atau sistem yang diteliti oleh peneliti. Peneliti menyadari setiap pengembangan aplikasi memiliki sifat atau tipe

sendiri – sendiri dalam pengembangannya. Alangkah baiknya jika pengembangan yang dilaksanakan masih sesuai dengan dasarnya yang antara lain adalah alur atau proses sistem yang digunakan sebelumnya. Dengan kata lain, maksudnya adalah pengembangan yang dilaksanakan tidak melenceng jauh dari sistem awal.

Sedangkan menurut dari respon pengujian di BLK dan masyarakat, dari hasil mendapat respon yang baik.

