BAB V

IMPLEMENTASI DAN PENGUJIAN SISTEM

5.1 Implementasi

Pada bab ini akan dilakukan implementasi dan pengujian terhadap aplikasi yang dibangun. Pada tahap pertama implementasi dilakukan kedalam bahasa pemrograman. Selanjutnya dilakukan pengujian terhadap aplikasi, Untuk memastikan aplikasi dapat berjalan sesuai dengan tujuan dan memastikan *software* memiliki kualitas yang kuat dengan keamanan yang baik.

5.1.1 Batasan Implementasi

Dalam implementasi aplikasi ini ada beberapa hal yang menjadi batasan dari implementasi, hal ini bertujuan untuk membatasi ruang lingkup pemahaman. Adapun batasan-batasan dari implementasi tersebut adalah sebagai berikut :

1. User yang telah daftar dan mendaftarkan layanan hanya diijinkan satu user untuk satu layanan yang artinya tidak bisa lebih dari satu layanan yang didaftarkan.
2. Aplikasi memanfaatkan model layanan *Platform as a Service* (PaaS).
3. Aplikasi dikelola oleh administrator untuk mengelola user, layanan, dan mengelola pelaporan yang berfungsi sebagai masukan dari user berupa penambahan layanan atau *feedback* apapun. Administrator harus memiliki pengetahuan mengenai *cloud computing* dan *Version Control System* (VCS) Git sehingga memudahkan melakukan perubahan aplikasi ke *server* maupun *debugging* dan kontrol aplikasi.

5.1.2 Implementasi Perangkat Lunak

Untuk membangun aplikasi diperlukan perangkat lunak yang mendukung dengan spesifikasi berikut:

1. Sistem operasi Linux Ubuntu 12.04

2. Tool pembantu pembangun aplikasi

a. Apache versi 2.2.22

b. PHP versi 5.3.14

c. Zend Framework versi 1.12.0

d. Netbeans IDE versi 7.2.1

3. Alat pembantu untuk berkomunikasi dengan server OpenShift Red Hat.

a. Git versi 1.7.9.5.

b. Ruby dan RubyGem harus terinstal di komputer local.

c. Browser. Google Chrome versi 21.0 dan Mozilla Firefox versi 17.01.

d. Basis data menggunakan MySQL.

e. Manajemen basis data MySQL menggunakan PhpMyAdmin.

f. Model analisis sistem merupakan pemodelan beroreantasi objek dan *tool* yang digunakan adalah UML (*Unified Modeling Languange*).

g. Aplikasi akan dibangun menggunakan bahasa pemrograman PHP. Dan *framework* yang digunakan adalah Zend Framework sebagai model yang berpola MVC (*Model View Controller*).

5.1.3 Implementasi Perangkat Keras

Perangkat keras yang dibutuhkan berdasarkan minimum requirement yang harus dipenuhi dalam penerapannya adalah sebagai berikut :

a. Prosesor Intel Pentium Dual Core atau setara.
b. RAM 512 Mb
c. VGA Card 512 Mb
d. Monitor dengan resolusi 1024 x 768 *pixels*
e. Koneksi *internet*. Kartu jaringan, Modem dan Router atau Switch.

5.1.4 Implementasi Basis Data

Implementasi basis data dilakukan dengan menggunakan bahasa SQL, dimana RDBMS (*Relational Database Management System*) yang digunakan adalah MySQL, implementasi basis datanya dalam bahas SQL adalah sebagai berikut :

```
CREATE TABLE IF NOT EXISTS `tbl_users` (
  `id` int(11) NOT NULL AUTO_INCREMENT,
  `email` varchar(50) NOT NULL,
  `password` varchar(255) NOT NULL,
  `tgl` datetime NOT NULL,
  PRIMARY KEY (`id`),
  UNIQUE KEY `email` (`email`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=3 ;
CREATE TABLE IF NOT EXISTS `tbl_admin` (
  `id` int(11) NOT NULL AUTO_INCREMENT,
```

```
  `email` varchar(100) NOT NULL,
  `password` varchar(100) NOT NULL,
  PRIMARY KEY (`id`),
  UNIQUE KEY `email` (`email`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=2 ;

CREATE TABLE IF NOT EXISTS `tbl_category` (
  `id` int(11) NOT NULL AUTO_INCREMENT,
  `nama` varchar(50) NOT NULL,
  `tgl_publikasi` date NOT NULL,
  `tgl_perubahan` date DEFAULT NULL COMMENT 'Default null digunakan untuk cek jika kategori baru ditambahkan. sehingga status Tidak ada perubahan.',
  PRIMARY KEY (`id`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=10 ;

CREATE TABLE IF NOT EXISTS `tbl_faq` (
  `id` int(11) NOT NULL AUTO_INCREMENT,
  `judul_konten` varchar(50) NOT NULL,
  `deskripsi` text NOT NULL,
  `tgl_publikasi` date NOT NULL,
  `tgl_perubahan` date DEFAULT NULL COMMENT 'Default null digunakan untuk cek jika faq baru ditambahkan. sehingga status Tidak ada perubahan.',
  PRIMARY KEY (`id`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=12 ;
```

```
CREATE TABLE IF NOT EXISTS `tbl_kecamatan` (
  `id` int(11) NOT NULL AUTO_INCREMENT,
  `nama` varchar(50) NOT NULL,
  `tgl_publikasi` date NOT NULL,
  `tgl_perubahan` date DEFAULT NULL COMMENT 'Default null digunakan untuk cek jika kecamatan baru ditambahkan. sehingga status Tidak ada perubahan.',
  PRIMARY KEY (`id`),
  KEY `nama` (`nama`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=31 ;

CREATE TABLE IF NOT EXISTS `tbl_layanan` (
  `id_layanan` int(11) NOT NULL AUTO_INCREMENT,
  `email` varchar(50) NOT NULL,
  `tgl_daftar` datetime NOT NULL,
  `tgl_perubahan` datetime DEFAULT NULL COMMENT 'Default null digunakan untuk cek jika user baru mendaftar. sehingga status Tidak ada perubahan.',
  `nama_layanan` varchar(100) NOT NULL,
  `kategori_layanan` varchar(100) NOT NULL,
  `jenis_layanan` varchar(50) NOT NULL,
  `deskripsi` text,
  `logo` varchar(50) NOT NULL,
```

```
  `daerah` varchar(50) NOT NULL,
  `alamat` text NOT NULL,
  `notelp` varchar(20) NOT NULL,
  `notelp2` varchar(20) DEFAULT NULL,
  `email_layanan` varchar(100) DEFAULT NULL,
  PRIMARY KEY (`id_layanan`),
  UNIQUE KEY `email` (`email`),
  KEY `daerah` (`daerah`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=3 ;

CREATE TABLE IF NOT EXISTS `tbl_pelaporan` (
  `id_pelaporan` int(11) NOT NULL AUTO_INCREMENT,
  `nama` varchar(100) NOT NULL,
  `email_pelaporan` varchar(50) NOT NULL,
  `pesan` text NOT NULL,
  `tgl_pelaporan` datetime NOT NULL,
  PRIMARY KEY (`id_pelaporan`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=139 ;

CREATE TABLE IF NOT EXISTS `tbl_pesan` (
  `id` int(11) NOT NULL AUTO_INCREMENT,
  `penerima_pesan` varchar(50) NOT NULL,
  `pengirim_pesan` varchar(50) NOT NULL,
  `subjek_pesan` varchar(100) NOT NULL,
```

```
  `isi_pesan` text NOT NULL,
  `tgl_pesan` datetime NOT NULL,
  PRIMARY KEY (`id`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=3 ;

ALTER TABLE `tbl_layanan`
  ADD CONSTRAINT `tbl_layanan_ibfk_4` FOREIGN KEY (`id_layanan`) REFERENCES `tbl_users` (`id`) ON DELETE CASCADE ON UPDATE CASCADE;
ALTER TABLE `tbl_pesan`
  ADD CONSTRAINT `tbl_pesan_ibfk_2` FOREIGN KEY (`pengirim_pesan`) REFERENCES `tbl_users` (`email`) ON DELETE CASCADE ON UPDATE CASCADE,
  ADD CONSTRAINT `tbl_pesan_ibfk_1` FOREIGN KEY (`penerima_pesan`) REFERENCES `tbl_users` (`email`) ON DELETE CASCADE ON UPDATE CASCADE;
```

5.1.5 Implementasi Antarmuka

Implementasi dari proses perancangan antar muka, merupakan gambaran mengenai antar muka perangkat lunak yang dibuat. Berikut ini merupakan implementasi antar muka dari perangkat lunak yang dirancang.

1. **Antarmuka Menu Publik.**

   Dibawah ini merupakan implementasi antarmuka dari menu publik.

Tabel 5.1 Implementasi Menu Publik

Menu	Deskripsi	Nama File
Home	Publik menu, menu yang pertama kali masuk kewebsite.	IndexController.php
Layanan	Daftar layanan yang terdafatar	ServiceController.php
FAQ	*Frequently asked questions*, pertanyaan-pertanyaan yang sering diajukan user, dirangkum dalam halaman FAQ.	FaqController.php
Pelaporan	Untuk melakukan pelaporan layanan, kontak dan lain-lain.	ReportingController.php
Menu	**Deskripsi**	**Nama File**
Login	Digunakan untuk login user yang telah mendaftar.	AuthController.php
Daftar	Mendaftar sebagai user member untuk mendapat fasilitas layanan bisnis.	AuthController.php
Layanan Baru	5 (lima) layanan baru yang terdaftar.	ServiceController.php
Kategori Layanan	Kategori yang dibuat untuk membagi setiap layanan yang terdaftar.	SearchController.php
Cari Layanan	Mencari layanan berdasarkan nama layanan yang telah terdaftar. Dan Menampilkan sesuai *query*	SearchController.php

2. Implementasi Menu User

Pada bagian menu user akan ditampilkan jika user telah login. Berikut menu user.

Tabel 5.2 Menu User

Menu	Deskripsi	Nama File
Dashboard	Menu utama user selelah berhasil login	UController.php
Konfigurasi	Ubah password lama dengan password baru.	UController.php
Daftar Layanan / Edit layanan.	Daftar (Jika user belum mendaftar layanan), Ubah dan Hapus layanan user.	UController.php
Menu	**Deskripsi**	**Nama File**
Pesan	Untuk berkomunikasi dengan sesame user member, yang berguna untuk bertanya harga atau detail layanan yang ditawarkan.	UController.php

3. Implementasi Menu IT Admin

Menu IT admin yang digunakan untuk mengelola user, layanan user dan pelaporan.

Tabel 5.3 Menu IT Admin

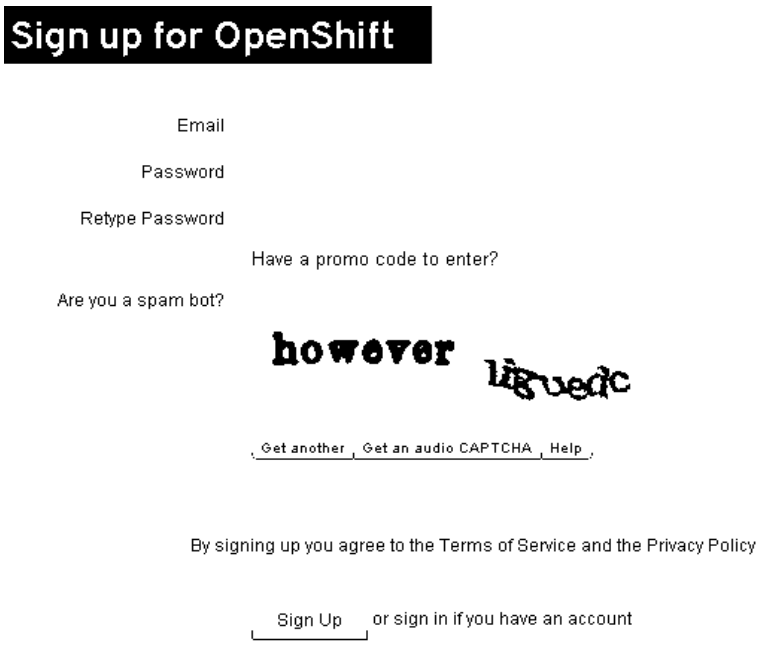
Menu	Deskripsi	Nama File
Home	Menu utama IT Admin jika berhasil login	AcladminController.php
Manajemen User	Untuk melakukan pengelolaan data user	AcladminController.php
Manajemen Layanan	Untuk melakukan pengelolaan data layanan user	AcladminController.php
Manajemen Pelaporan	Untuk melakukan pengelolaan data laporan user member atau bukan member	AcladminController.php
Manajemen FAQ	Mengelola data FAQ (*Frequentlu asked questions*). Menambah,edit dan hapus data FAQ.	AcladminController.php

5.1.6 Implementasi Instalasi Program

Perangkat lunak yang dibangun ini agar dapat berfungsi sesuai tujuan maka diperlukan *uploading* ke *server* OpenShift Red Hat selaku penyedia *Cloud Platform as a Service* (PaaS). Sebelum melakukan *uploading* diperlukan 3 langkah umum yang harus terpenuhi, berikut lengkapnya:

1. Melakukan pendaftaran ke OpenShift

   Buka url (https://openshift.redhat.com/)



Gambar 5.1 Pendaftaran OpenShift

Login dengan akun yang baru didaftar.



Gambar 5.2 Login OpenShift

2. Melakukan instalasi perangkat lunak pada klien (*Client tool*) dan konfigurasi

   Diperlukan instalasi OpenSSH, Ruby, RubyGem dan Git. Dalam hal ini penulis melakukan instalasi pada sistem operasi Linux Ubuntu.

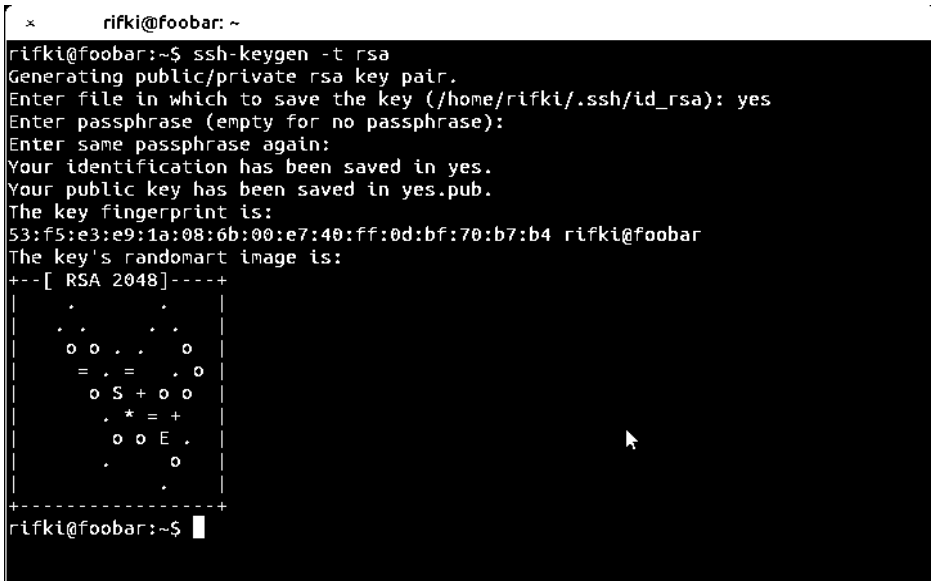
   Instalasi openSSH, ruby dan git pada terminal:

```
sudo apt-get install ruby-full rubygems git-core

sudo gem install rhc
```

Melakukan *generate* RSA untuk di unggah ke *website* OpenShift

```
ssh-keygen -t rsa
```



Gambar 5.3 *Generate* RSA

Salin id_rsa.pub dan *upload* ke *website* OpenShift

```
sudo gedit ~/.ssh/id_rsa.pub
```



Gambar 5.4 Memasukan publik *key*

Membuat nama domain dengan rifki.rhcloud.com

```
$ rhc domain create -n rifki -l rifki@mail.com

Password: **********

RESULT:

Success!

You may now create an application using the 'rhc app create' command
```

Membuat *NameSpace* dengan nama app sehingga hasilnya : app-rifki.rhcloud.com

```
rhc app create app
```

Pilih *cartridge* zend-5.6, mysql-5.1 dan phpmyadmin-3.4.

```
rhc cartridge add zend-5.6

rhc cartridge add mysql-5.1
```

```
rhc cartridge add phpmyadmin-3.4
```

3. *Uploading* ke *server* menggunakan Git (*Version Control System* Git)

   Melakukan cloning dari server ke lokal, Yang berguna untuk *upload* ke *server*.

```
git clone ssh://de78cef3f5e84eecb35a1458657d470b@app
rifki.rhcloud.com/~/git/app.git/
```

Masuk ke direktori app dan salin semua aplikasi yang ada di *webserver* dipindahkan ke direktori app agar aplikasi dapat dilakukan *upload* dari lokal ke *server*.

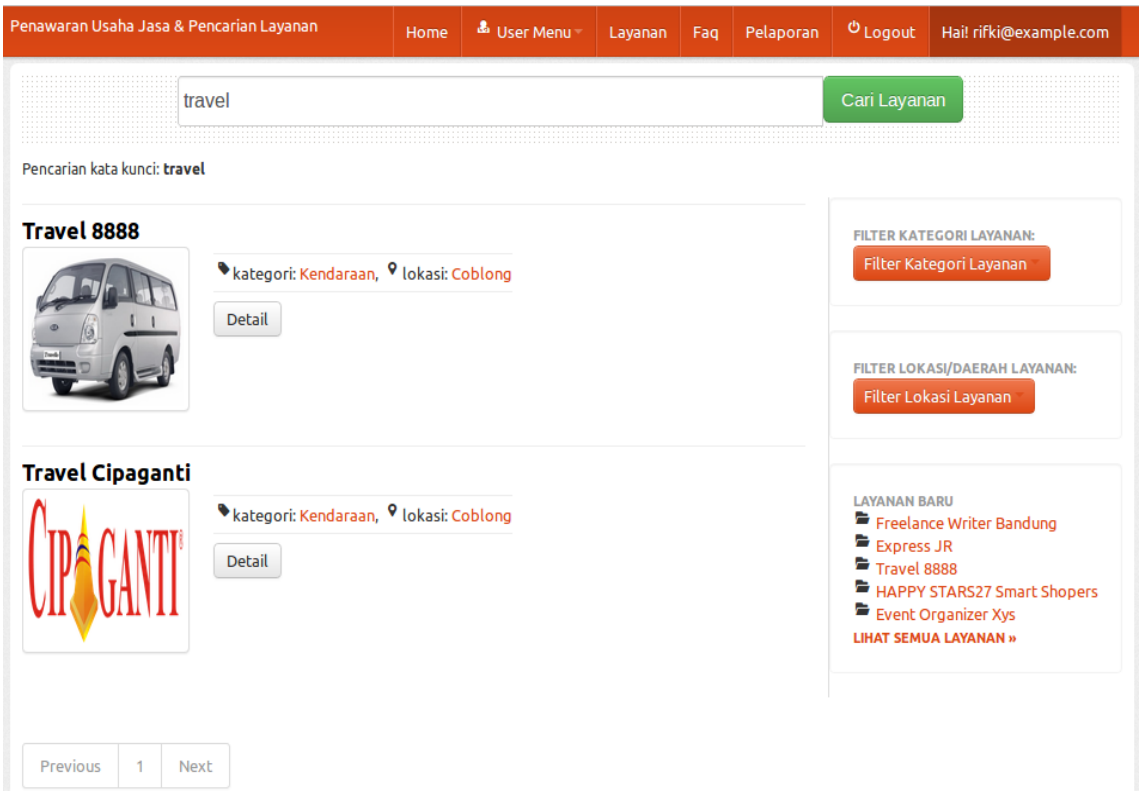
```
git init
git add .
git commit  -m “First Commit”
git push
```

5.1.7 Penggunaan Program

Penggunaan program merupakan langkah-langkah dalam menggunakan dan menjalankan suatu aplikasi yang dilakukan oleh pengguna. Langkah-langkah penggunaan aplikasi menggunakan struktur menu dan penjelasan sebagai berikut:

1. Form Pencarian

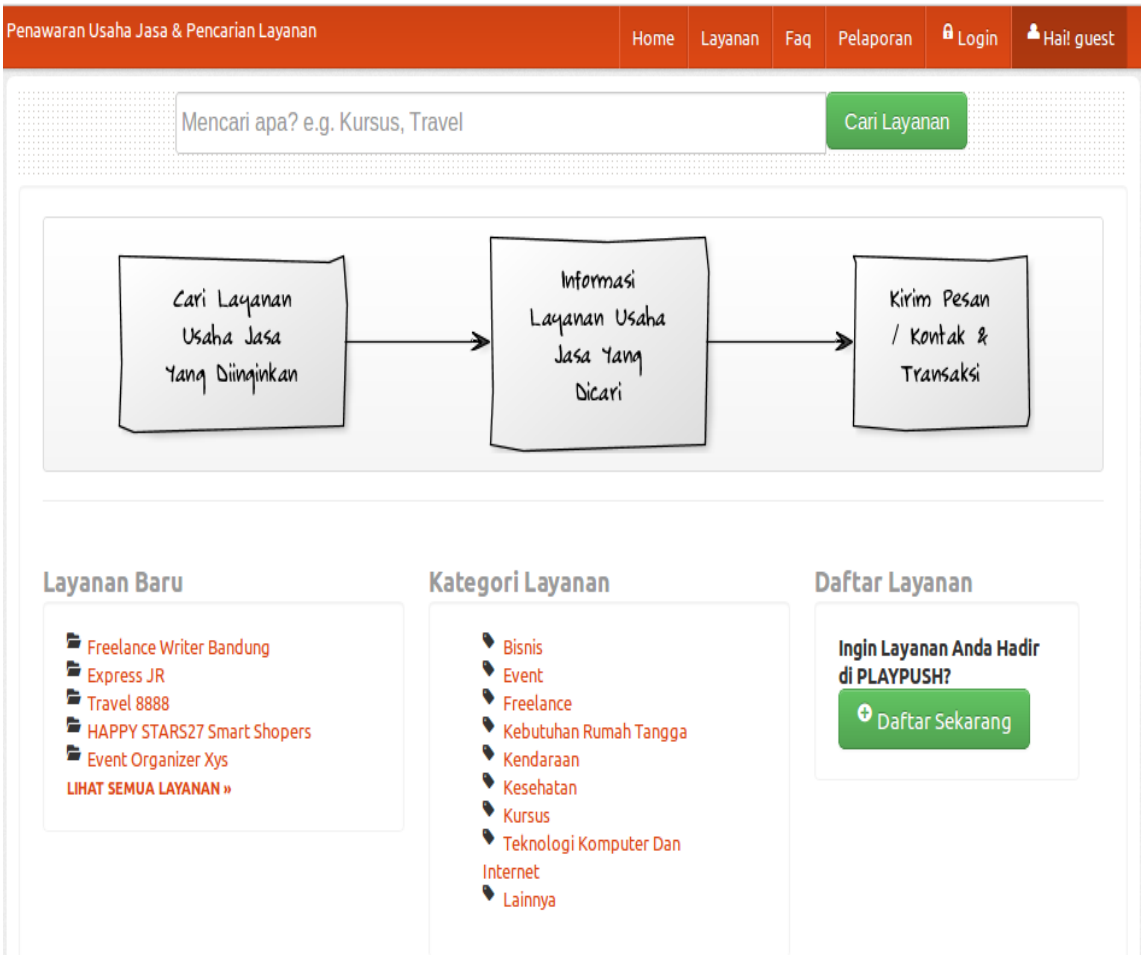
   Digunakan untuk pencarian layanan usaha dan jasa.



Gambar 5.5 Form pencarian

2. Menu Home

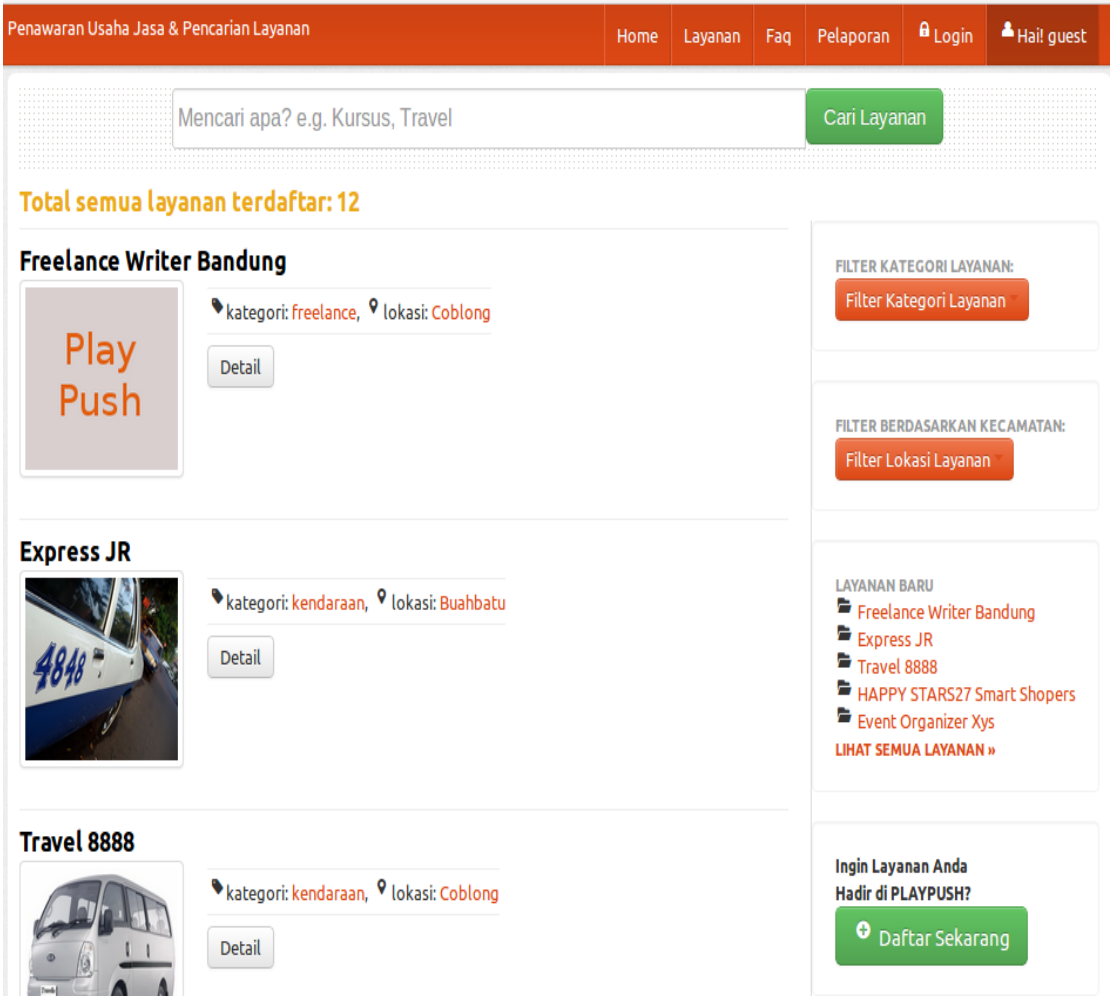
Menu utama pertama kali user mengunjungi situs.



Gambar 5.6 Menu utama

3. Menu Layanan

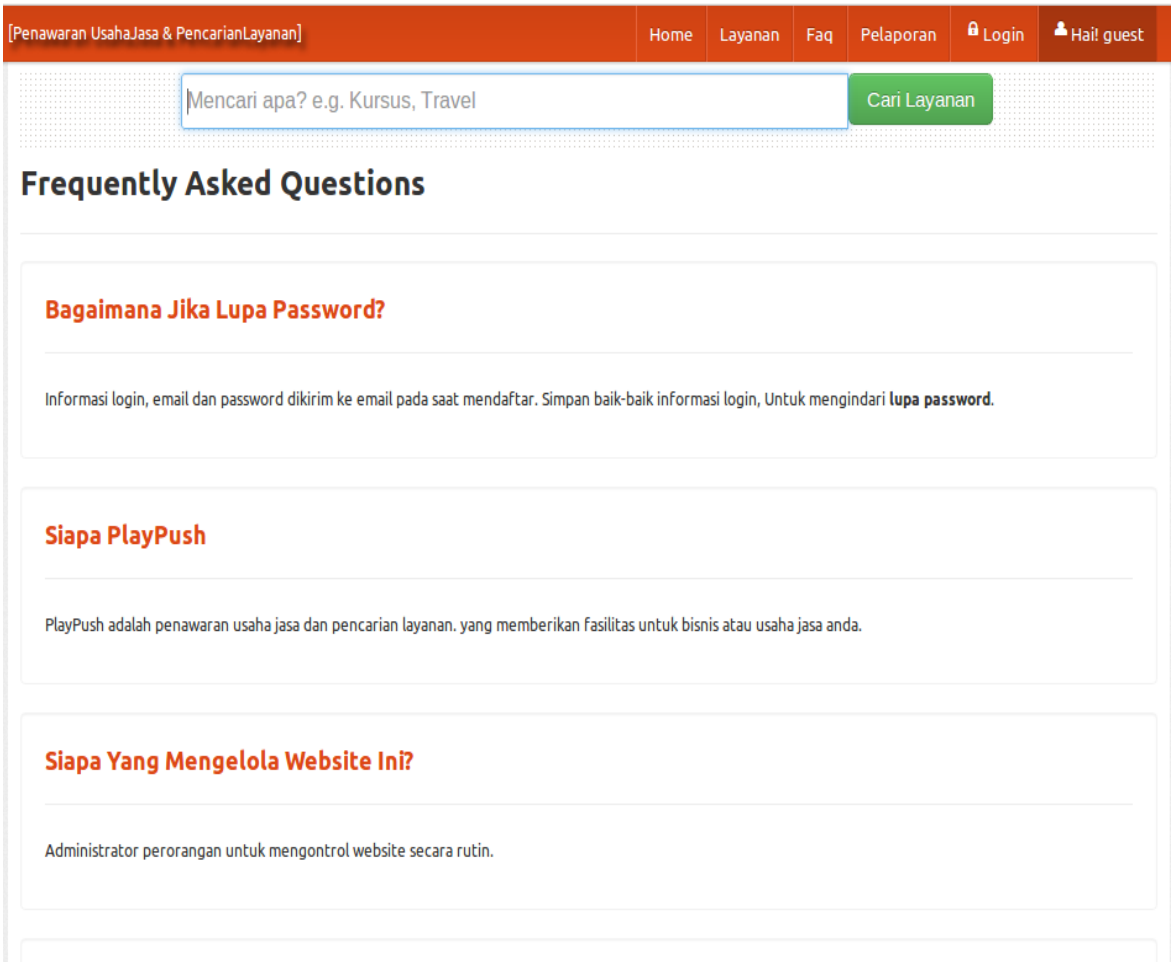
Menampilkan daftar layanan yang terdaftar



Gambar 5.7 Menu layanan

4. Menu Faq

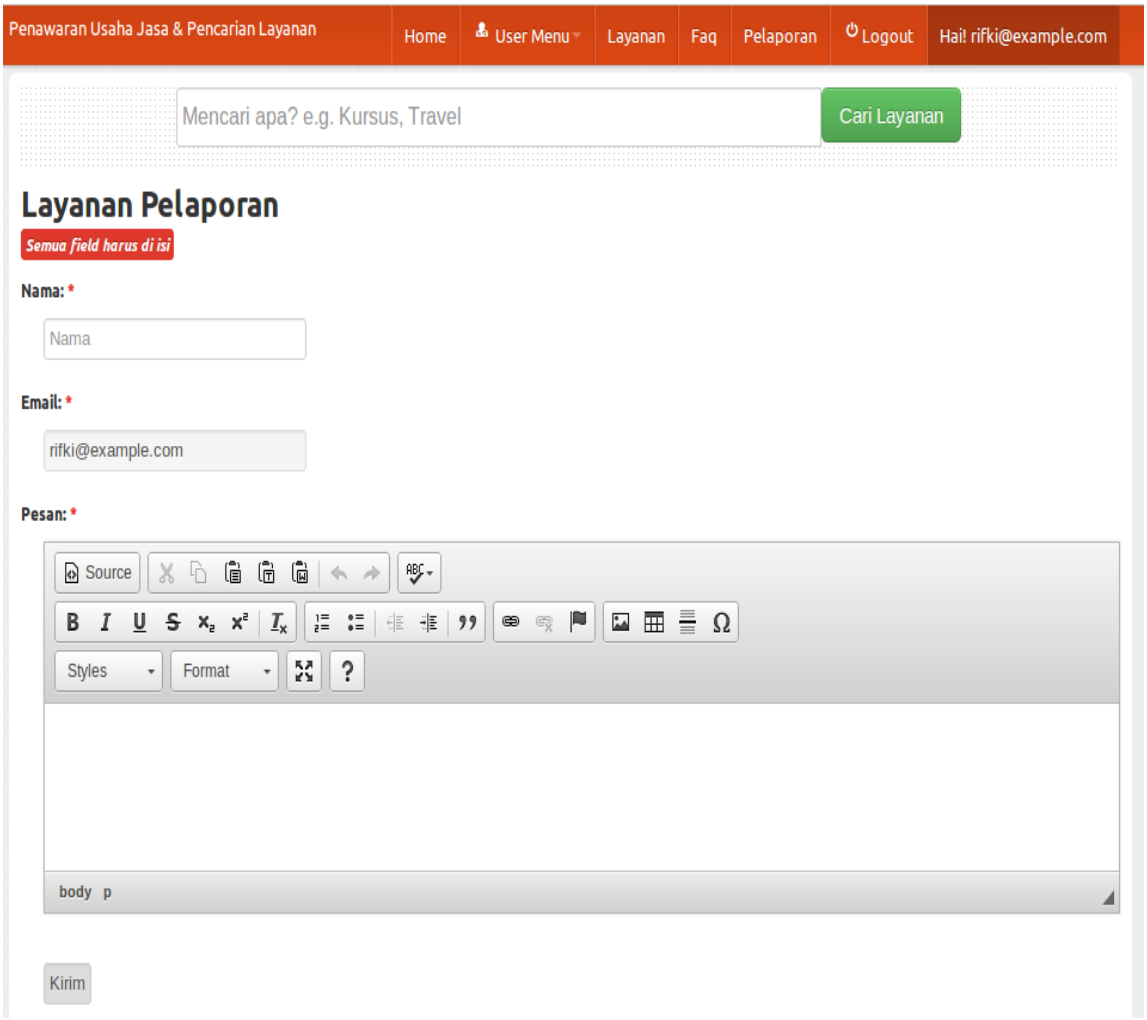
   Menampilkan daftar pertanyaan yang sering diajukan user.



Gambar 5.8 Menu FAQ

5. Menu Pelaporan

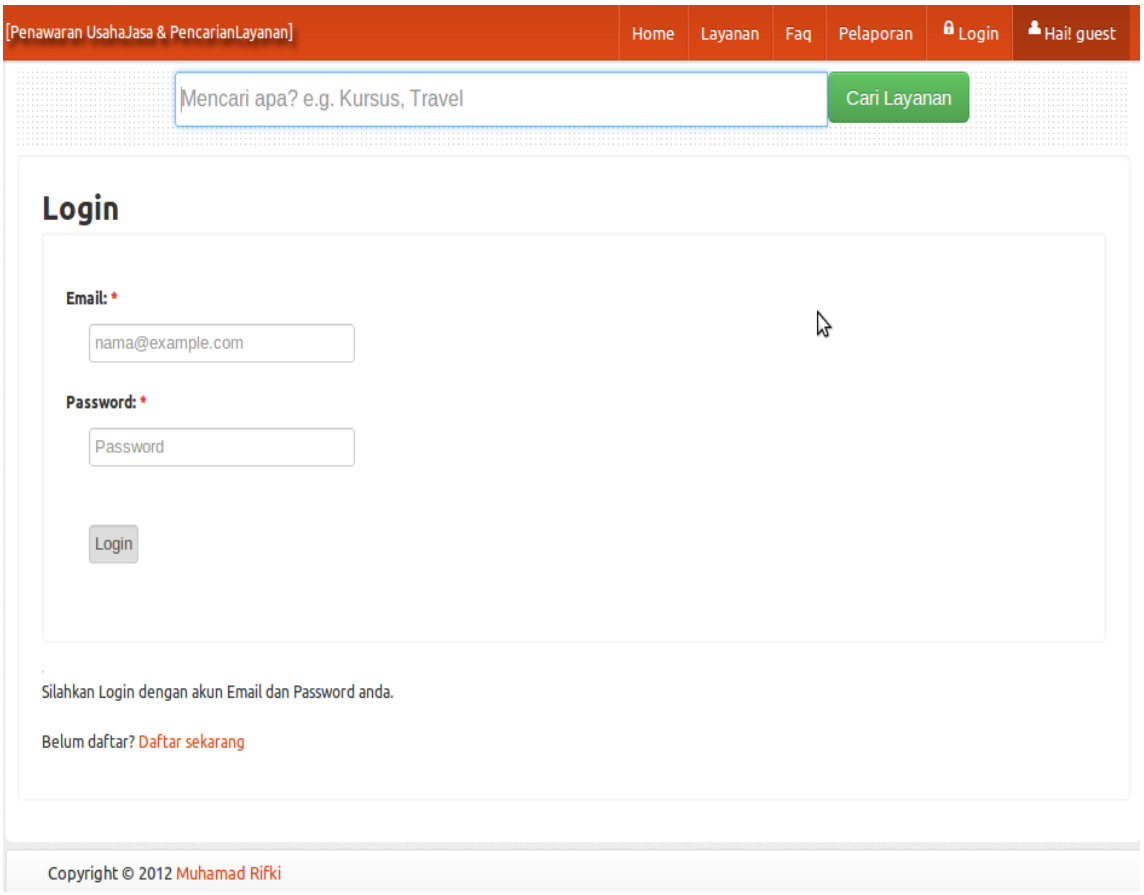
Menu pelaporan dengan mengisi form nama, email dan pesan untuk melaporkan dari user atau tamu ke IT Admin atau administrator



Gambar 5.9 Menu pelaporan

6. Login User

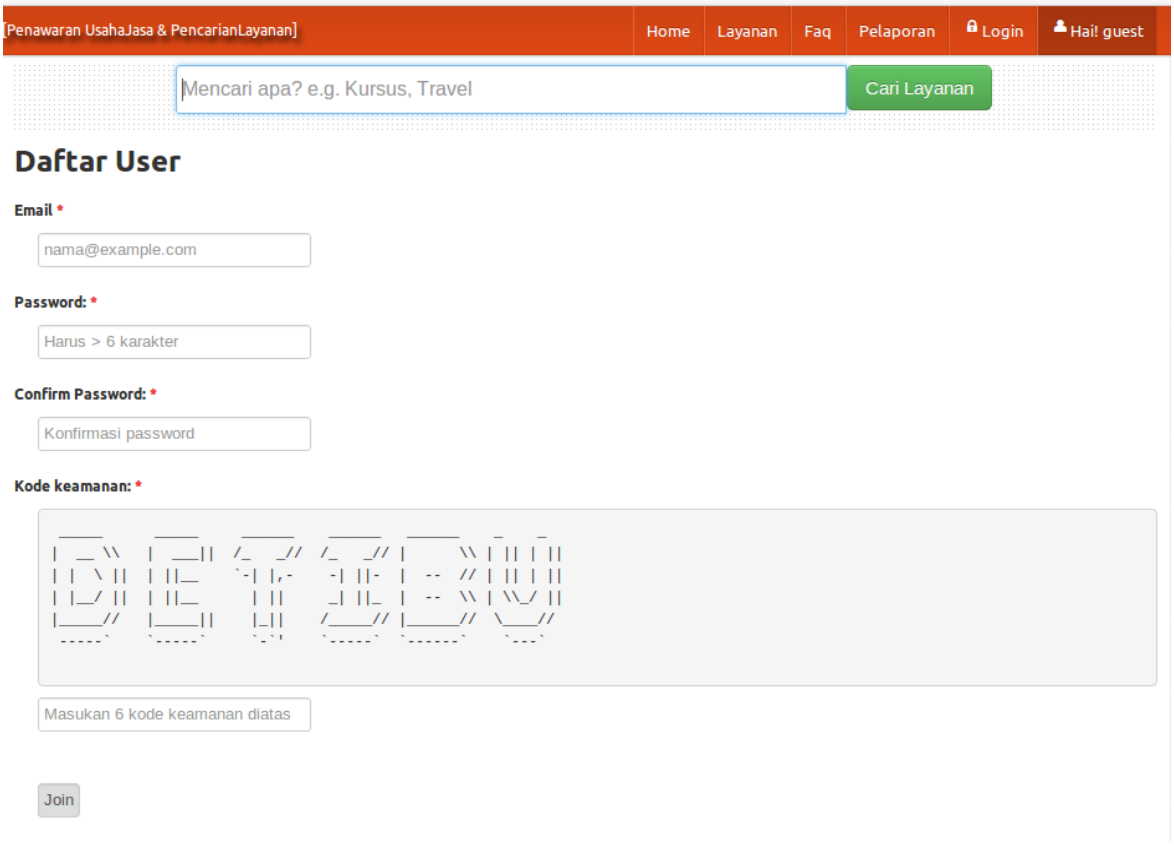
Dengan mengisi form email dan password maka user dapat masuk ke aplikasi utama untuk mendaftarkan layanan.



Gambar 5.10 Login user

7. Daftar User

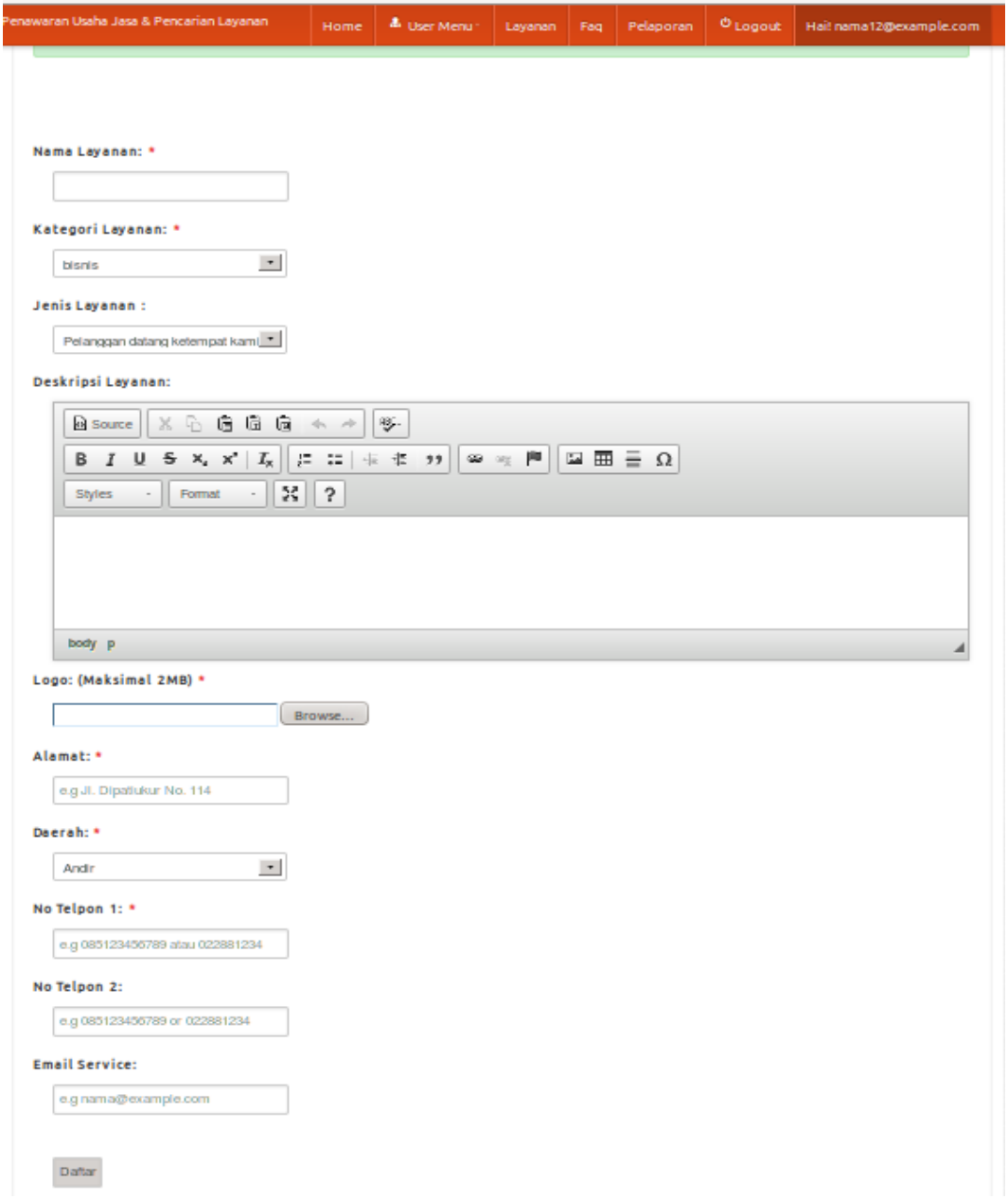
   Mengisi form email, password, confirmasi password dank ode keamanan jika valid user dapat melakukan login



Gambar 5.11 Daftar user

8. Daftar Layanan

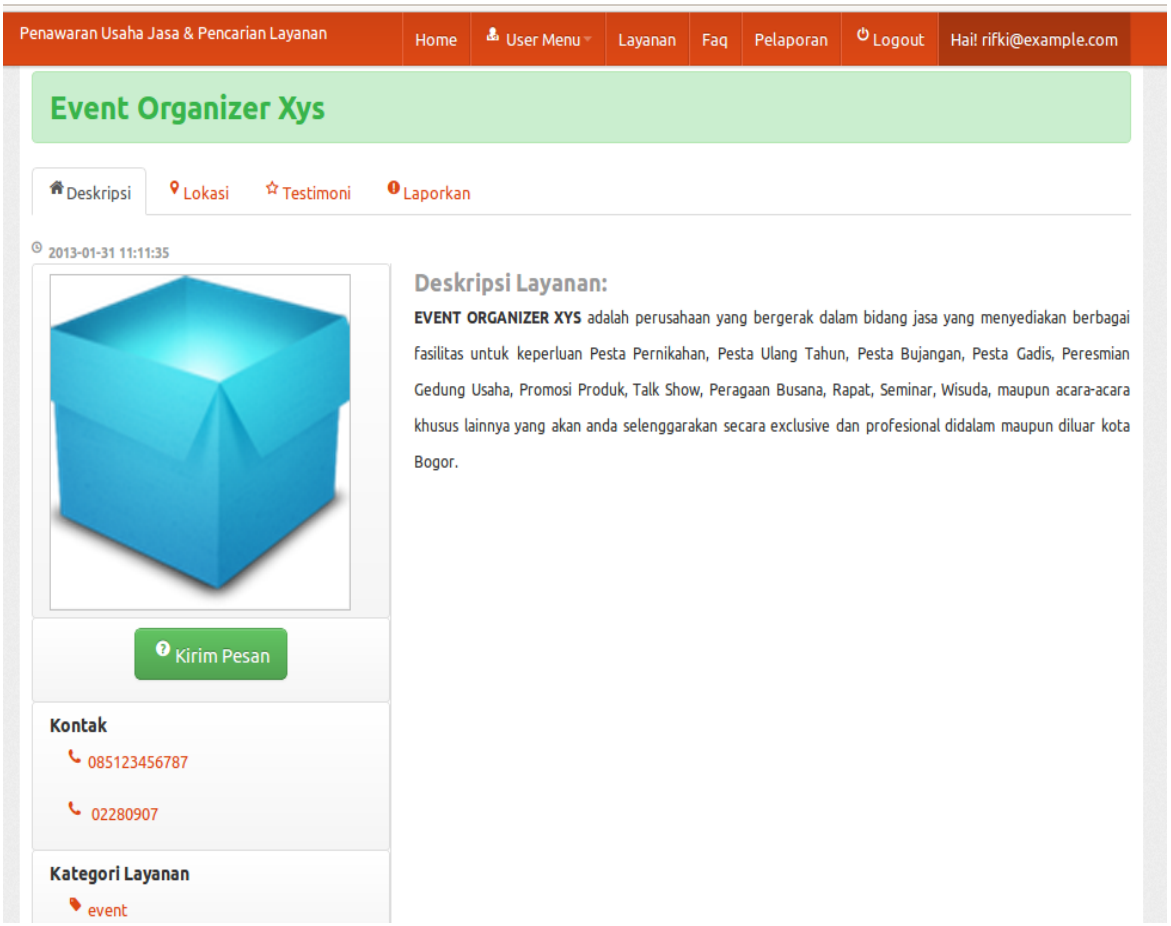
Digunakan untuk mendaftarkan layanan usaha jasa yang meliputi operasi ubah dan hapus layanan.



Gambar 5.12 Daftar layanan

9. Menu Detail Layanan

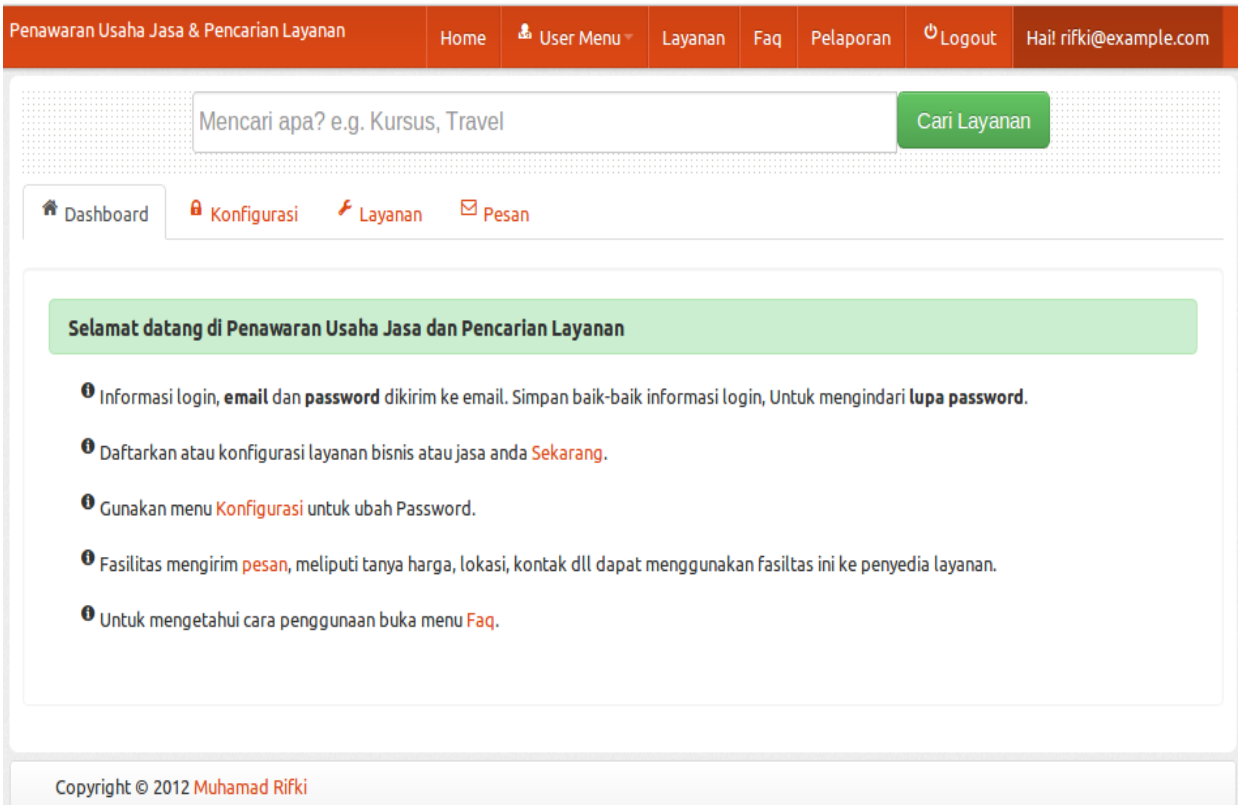
   Menampilkan detail layanan yang telah terdaftar.



Gambar 5.13 Menu detail layanan

10. Menu Dashboard

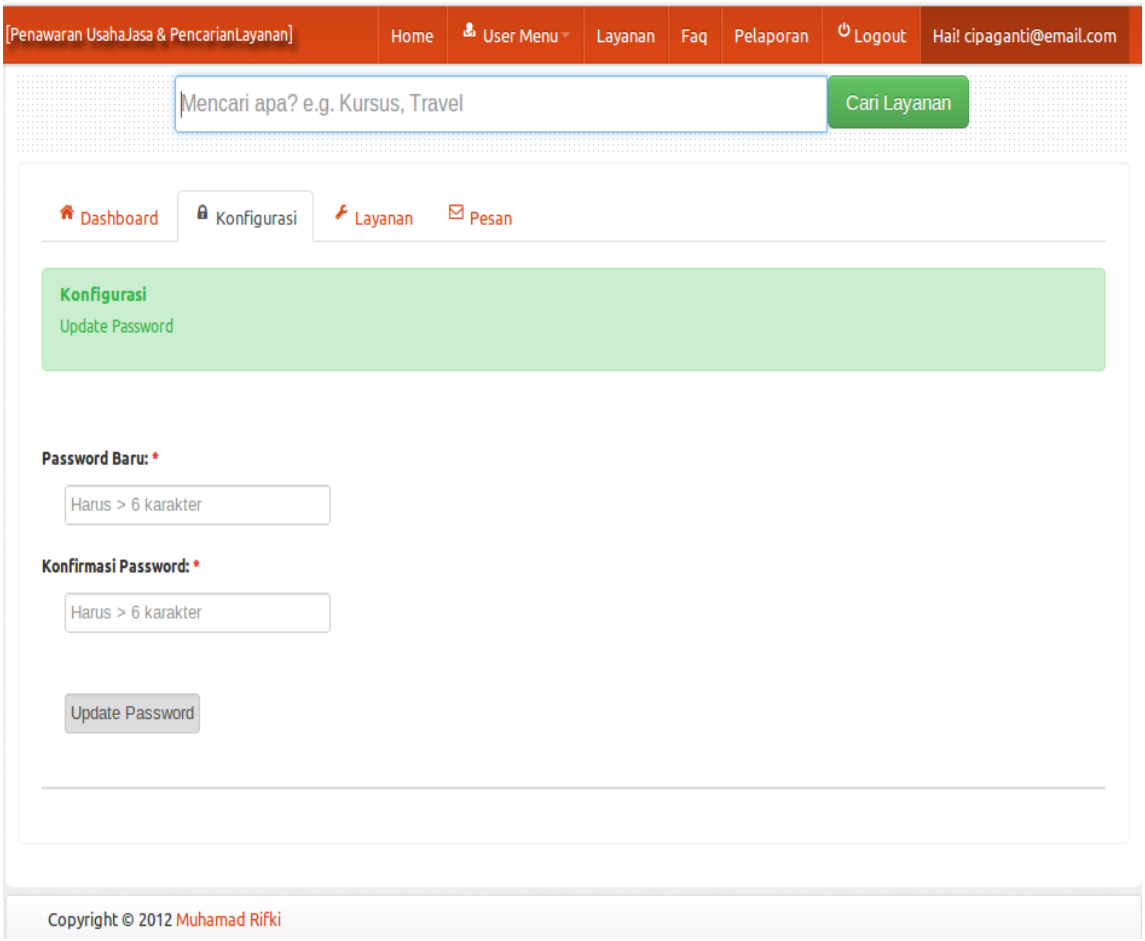
Menu pertama kali user berhasil login atau daftar.



Gambar 5.14 Menu dashboard

11. Menu Konfigurasi

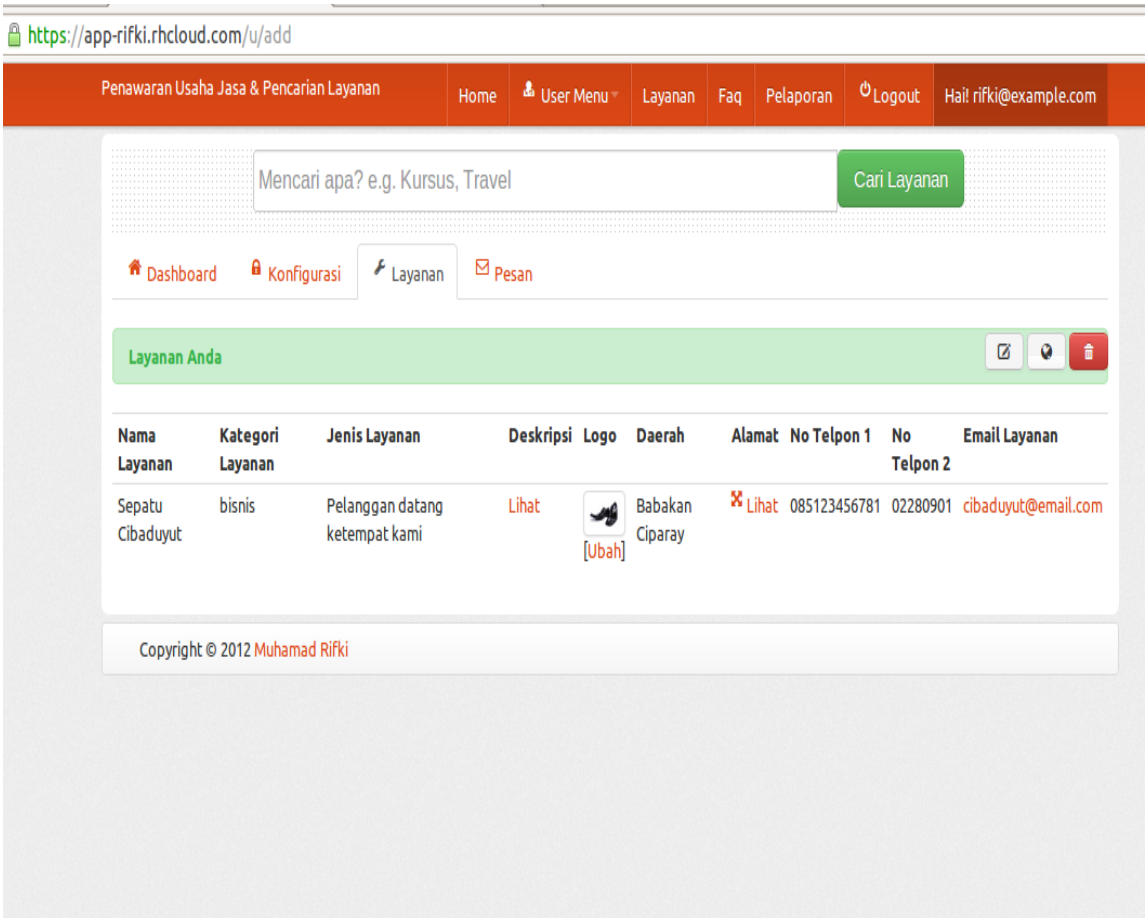
Menu ini muncul jika user sudah melakukan daftar. Digunakan untuk mengubah password dengan mengisi password baru dan konfirmasi password baru



Gambar 5.15 Menu konfigurasi

12. Menu Layanan

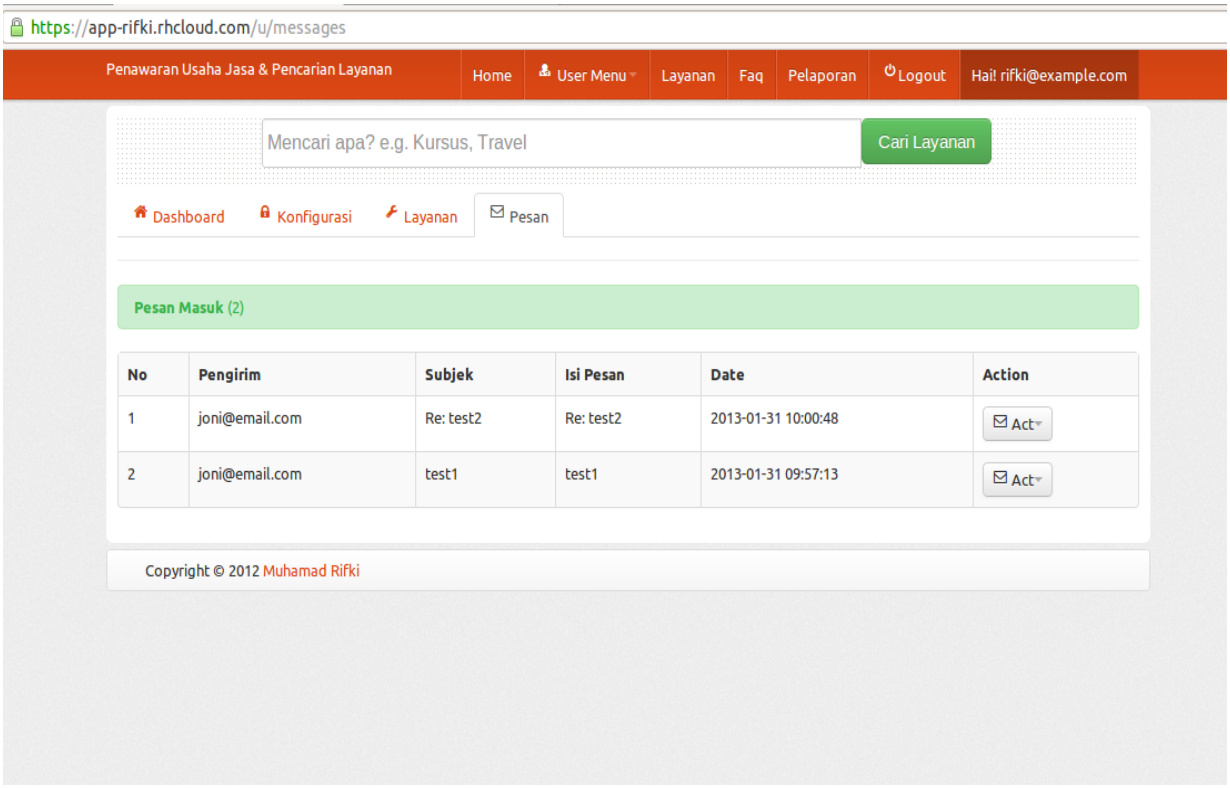
Digunakan untuk mendaftarkan layanan, ubah layanan dan hapus layanan.



Gambar 5.16 Menu layanan

13. Menu Pesan

Mengelola pesan masuk dari user yang meliputi balas pesan dan hapus pesan.



Gambar 5.17 Menu pesan

5.2 Pengujian

Pengujian diperlukan sebagai salah satu tahapan implementasi untuk menguji tingkat minimal kesalahan dan keakuratan perangkat lunak yang dirancang. Pengujian dilakukan dengan metode pengujian *black box*.

5.2.1 Rencana Pengujian

Pengujian perangkat lunak yang digunakan yaitu metode *black box*. Pengujian *black box* berfokus pada persyaratan fungsional perangkat lunak yang dibuat. Rencana pengujian program meliputi :

Tabel 5.4 Rencana Pengujian

Kelas Uji	Detail Uji
Daftar User	Memasukan email, password, konfirmasi password dan kode keamanan.
Login User	Memasukan email dan password user.
Ubah Password User	Memasukan password baru dan memasukan konfirmasi password
Daftar Layanan	Memasukan nama layanan, kategori layanan, jenis layanan, deskripsi, upload logo, daerah, alamat, notelpon1, notelpon2 dan email layanan
Edit Layanan	Mengubah layanan user yang telah terdaftar sebelumnya.
Hapus Layanan	Menghapus Layanan yang telah terdaftar
Pelaporan	Memasukan nama, email dan pesan
Login IT Admin	Memasukan email dan password

5.2.2 Kasus dan Hasil Pengujian

Setelah melaksanakan rencana pengujian maka dilanjutkan dengan kasus dan hasil pengujian. Berikut kasus dan hasil pengujian :

5.2.2.1. Pengujian Daftar User

Tabel 5.5 Pengujian Daftar User

Kasus dan Hasil Uji (Data Normal)			
Data Masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Input email, password dan konfirmasi password	User secara otomatis sistem akan redirect ke halaman login. Untuk melakukan login sesuai dengan akun yang didaftar	Hasil sesuai dengan harapan	[X] diterima [ ] ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Input email atau password < 6 atau confirm password tidak sama	Tidak Dapat melakukan login	Email, password, confirm password tidak valid	[ ] diterima [X] ditolak

5.2.2.2. Pengujian Login User

Tabel 5.6 Pengujian Login User

Kasus dan Hasil Uji (Data Normal)			
Data Masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Input email dan password	User masuk ke menu utama user	Hasil sesuai dengan harapan	[X] diterima [ ] ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Input email benar dan password salah. Dan input email salah dan password benar	Tidak Dapat melakukan login	Email atau password tidak valid	[ ] diterima [X] ditolak

5.2.2.3. Pengujian Ubah Password User

Tabel 5.7 Pengujian Ubah Password User

Kasus dan Hasil Uji (Data Normal)			
Data Masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Input password baru dan konfirmasi password baru	User dapat menggunakan password baru	Hasil sesuai dengan harapan	[X] diterima [ ] ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Input password baru dan konfirmasi password tidak sama	Tidak dapat melakukan *update* password	Password dan *confirm* password tidak sama.	[ ] diterima [X] ditolak

5.2.2.4. Pengujian Mendaftarkan Layanan

Tabel 5.8 Pengujian Mendaftarkan Layanan

Kasus dan Hasil Uji (Data Normal)			
Data Masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Semua data (Yang wajib diisi)	User dapat mendaftarkan layanan	Hasil sesuai dengan harapan	[X] diterima [ ] ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Semua data yang wajib diisi tidak diisi	Tidak dapat melakukan pendaftaran harap dilengkapi	Hasil sesuai dengan harapan	[ ] diterima [X] ditolak

5.2.2.5. Pengujian Edit Layanan

Tabel 5.9 Pengujian Edit Layanan

Kasus dan Hasil Uji (Data Normal)			
Data Masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Ubah data layanan dengan mengisi semua yang wajib diisi	User dapat mengubah data layanan	Hasil sesuai dengan harapan	[X] diterima [ ] ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Ubah data layanan dengan tidak mengisi semua yang wajib diisi	Tidak dapat mengubah layanan dan harap dilengkapi	Hasil sesuai dengan harapan	[ ] diterima [X] ditolak

5.2.2.6. Pengujian Hapus Layanan

Tabel 5.10 Pengujian Hapus Layanan

Kasus dan Hasil Uji (Data Normal)			
Data Masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Hapus layanan user yang terdaftar dan klik button “Ok”	Layanan terhapus	Hasil sesuai dengan harapan	[X] diterima [ ] ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Hapus layanan user dan klik tombol “Cancel”	Tidak dapat menghapus layanan	Hasil sesuai dengan harapan	[ ] diterima [X] ditolak

5.2.2.7. Pengujian Pelaporan

Tabel 5.11 Pengujian Pelaporan

Kasus dan Hasil Uji (Data Normal)			
Data Masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Input data nama, email dan pesan	Memasukan data nama dan email secara otomatis *readonly* (jika user dalam keadaan login). Jika tidak dalam keadaan login/belum daftar user input email secara manual	Hasil sesuai dengan harapan	[X] diterima [ ] ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Input pelaporan tanpa cantumkan email valid	Tidak Dapat melakukan input	Email tidak valid	[ ] diterima [X] ditolak

5.2.2.8. Pengujian Login IT Admin

Tabel 5.12 Pengujian Login IT Admin

Kasus dan Hasil Uji (Data Normal)			
Data Masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Memasukan email dan password	IT Admin masuk ke aplikasi utama	Hasil sesuai dengan harapan	[X] diterima [ ] ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Email atau password tidak diisi. Email dan password diisi tidak sesuai	Tidak dapat melakukan login jika tidak sesuai dengan aksesnya	Hasil sesuai dengan harapan	[ ] diterima [X] ditolak

5.2.3 Kesimpulan Hasil Pengujian

Berdasarkan dari hasil implementasi dan pengujian terhadap hasil dari perancangan, dapat disimpulkan bahwa perangkat lunak yang dibangun telah sesuai dengan kasus - kasus yang diberikan terhadap pengujian yang dilakukan. Prosedur - prosedur perangkat lunak telah berjalan dengan baik dengan tingkat pemahaman pengguna terhadap perangkat lunak yang dibangun dapat dipahami dengan mudah.