BAB V

IMPLEMENTASI DAN PENGUJIAN SISTEM

5.1 Implementasi

Implementasi adalah proses penerapan rancangan program yang telah dibuat pada bab sebelumnya atau aplikasi dalam melaksanakan sistem informasi pemrograman yang telah dibuat, hasil dari tahapan implementasi ini adalah suatu sistem pengolahan data yang sudah dapat berjalan dengan baik. Dengan demikian dapat diketahui apakah aplikasi ini dapat menghasilkan sistem informasi penjadwalan *event* pada PT. Mitra Bandung Consortium sesuai dengan tujuan yang diharapkan.

5.1.1 Batasan Implementasi

1. Tidak semua kebutuhan sistem yang ada di PT. Mitra Bandung Consortium dapat dikerjakan yang dikarenakan keterbatasan waktu dan keterbatasan sumber daya, akan tetapi hanya sistem yang mendukung pada proses penjadwalan *event* pada bagian marketing saja.
2. Aplikasi yang dibangun difokuskan untuk pengelolaan dan penjadwalan *event* dimana pengguna *event* dapat memasukkan proposal *event* dan mengelola proposal atau *event* yang telah dimasukkan.
3. Penggunaan fasilitas-fasilitas yang tidak berhubungan dengan sistem yang terdapat pada halaman administrator hanya sekedar menampilkan informasi pada website utama.

4. Aplikasi yang dibangun ini masih merupakan prototype.
5. Bahasa yang digunakan pada aplikasi website ini yaitu bahasa Inggris.

5.1.2 Implementasi Perangkat Lunak

Untuk mendukung kelancaran sistem informasi ini selain diperlukan *hardware* juga diperlukan perangkat lunak (*software*) yang terdiri dari sistem operasi dan aplikasi *database*. Adapun perangkat lunak yang digunakan untuk membangun sistem informasi penjadwalan *event* adalah sebagai berikut:

1. Sistem Operasi menggunakan Apple Mac OS X 10.8.4
2. Bahasa pemograman menggunakan PHP, Javascript, jQuery dan Ajax
3. *Script* editor menggunakan Adobe Dreamweaver CS5
4. Web *server* menggunakan Apache (Xampp)
5. GUI *database* menggunakan PhpMyAdmin (Xampp)
6. *Database Management System* (DBMS) menggunakan MySQL
7. Aplikasi *browser* menggunakan Mozilla Firefox dan Safari

5.1.3 Implementasi Perangkat Keras

Perangkat keras yang dibutuhkan berdasarkan kebutuhan minimal yang harus terpenuhi antara lain:

1. *Processor* yang digunakan (Intel Pentium 4 2.3GHz) atau setara
2. RAM terpasang 1 GB
3. *Harddisk* 80Gb
4. *Mouse*, *keyboard*, *monitor* sebagai peralatan antarmuka
5. Koneksi *internet*

5.1.4 Implementasi Basis Data (Sintaks SQL)

Pembuatan basis data dilakukan dengan menggunakan bahasa SQL, dimana perangkat lunak yang digunakan adalah MySQL dan PhpMyAdmin sebagai GUI basis datanya. Implementasi basis datanya dalam urutan tampilan desain tabel SQL adalah sebagai berikut :

1. Tabel Admin

```
CREATE TABLE IF NOT EXISTS `admin` (
  `id` int(10) NOT NULL AUTO_INCREMENT,
  `id_level` int(1) NOT NULL,
  `username` varchar(30) NOT NULL,
  `nama_lengkap` varchar(100) NOT NULL,
  `password` varchar(100) NOT NULL,
  `email` varchar(100) NOT NULL,
  `telpon` varchar(50) NOT NULL,
  `keterangan` varchar(100) NOT NULL,
  `foto` varchar(100) NOT NULL,
  `aktif` enum('Y','N') NOT NULL,
  PRIMARY KEY (`id`)
) ENGINE=MyISAM DEFAULT CHARSET=latin1
AUTO_INCREMENT=6 ;
```

2. Tabel Album

```
CREATE TABLE IF NOT EXISTS `album` (
  `id` int(10) NOT NULL AUTO_INCREMENT,
```

```
 `id_user` int(10) NOT NULL,
 `album` varchar(80) NOT NULL,
 `tanggal` date NOT NULL,
 `keterangan` text NOT NULL,
 `cover` varchar(100) NOT NULL,
 PRIMARY KEY (`id`)
) ENGINE=MyISAM  DEFAULT CHARSET=latin1
AUTO_INCREMENT=17 ;
```

3. Tabel Berita

```
CREATE TABLE IF NOT EXISTS `berita` (
 `id` int(11) NOT NULL AUTO_INCREMENT,
 `edit` varchar(100) NOT NULL,
 `tanggal` date NOT NULL,
 `isi` text NOT NULL,
 `gambar` varchar(100) NOT NULL,
 `aktif` enum('Y','N') NOT NULL,
 PRIMARY KEY (`id`)
) ENGINE=MyISAM  DEFAULT CHARSET=latin1
AUTO_INCREMENT=2 ;
```

4. Tabel Chat

```
CREATE TABLE IF NOT EXISTS `chat` (
 `id_chat` int(10) NOT NULL AUTO_INCREMENT,
 `username` varchar(100) COLLATE latin1_general_ci NOT NULL,
 `pesan` text COLLATE latin1_general_ci NOT NULL,
```

```
 `jam` datetime NOT NULL,
 PRIMARY KEY (`id_chat`)
) ENGINE=MyISAM  DEFAULT CHARSET=latin1
COLLATE=latin1_general_ci AUTO_INCREMENT=115 ;
```

5. Tabel Menu Admin

```
CREATE TABLE IF NOT EXISTS `ddadmin` (
 `id` int(10) NOT NULL AUTO_INCREMENT,
 `parent_id` int(10) NOT NULL,
 `judul` varchar(100) NOT NULL,
 `url` varchar(100) NOT NULL,
 `menu_order` int(10) NOT NULL,
 `keterangan` varchar(100) NOT NULL,
 `aktif` enum('Y','N') NOT NULL,
 PRIMARY KEY (`id`)
) ENGINE=MyISAM  DEFAULT CHARSET=latin1
AUTO_INCREMENT=34 ;
```

6. Tabel Menu Awal

```
CREATE TABLE IF NOT EXISTS `ddawal` (
 `id` int(3) NOT NULL AUTO_INCREMENT,
 `parent_id` int(3) NOT NULL,
 `judul` varchar(100) NOT NULL,
 `url` varchar(100) NOT NULL,
 `menu_order` int(3) NOT NULL,
 `aktif` enum('Y','N') NOT NULL,
```

```
 PRIMARY KEY (`id`)
) ENGINE=MyISAM  DEFAULT CHARSET=latin1
AUTO_INCREMENT=16 ;
```

7. Tabel Menu *Event Organizer*

```
CREATE TABLE IF NOT EXISTS `ddeo` (
 `id` int(10) NOT NULL AUTO_INCREMENT,
 `parent_id` int(10) NOT NULL,
 `judul` varchar(100) NOT NULL,
 `url` varchar(100) NOT NULL,
 `menu_order` int(10) NOT NULL,
 `keterangan` varchar(100) NOT NULL,
 `aktif` enum('Y','N') NOT NULL,
 PRIMARY KEY (`id`)
) ENGINE=MyISAM  DEFAULT CHARSET=latin1
AUTO_INCREMENT=12 ;
```

8. Tabel *Event Organizer*

```
CREATE TABLE IF NOT EXISTS `eo` (
 `id` int(10) NOT NULL AUTO_INCREMENT,
 `id_level` int(1) NOT NULL,
 `username` varchar(30) NOT NULL,
 `password` varchar(100) NOT NULL,
 `nama_lengkap` varchar(100) NOT NULL,
 `jenis_kelamin` varchar(100) NOT NULL,
 `no_ktp` varchar(100) NOT NULL,
```

```
  `perusahaan` varchar(100) NOT NULL,
  `jabatan` varchar(100) NOT NULL,
  `alamat` varchar(250) NOT NULL,
  `telpon` varchar(100) NOT NULL,
  `handphone` varchar(100) NOT NULL,
  `pin` varchar(100) NOT NULL,
  `email` varchar(100) NOT NULL,
  `twitter` varchar(100) NOT NULL,
  `bank` varchar(100) NOT NULL,
  `no_rek` varchar(100) NOT NULL,
  `atas_nama` varchar(100) NOT NULL,
  `keterangan` varchar(100) NOT NULL,
  `foto` varchar(100) NOT NULL,
  `aktif` enum('Y','N') NOT NULL,
  PRIMARY KEY (`id`)
) ENGINE=MyISAM  DEFAULT CHARSET=latin1
AUTO_INCREMENT=5 ;
```

9. Tabel Level

```
CREATE TABLE IF NOT EXISTS `level` (
  `id_level` int(10) NOT NULL,
  `level` varchar(100) NOT NULL
) ENGINE=MyISAM DEFAULT CHARSET=latin1;
```

10. Tabel *News*

```
CREATE TABLE IF NOT EXISTS `news` (
```

```
 `id` int(11) NOT NULL AUTO_INCREMENT,
 `tanggal` date NOT NULL,
 `judul` varchar(100) NOT NULL,
 `isi` text NOT NULL,
 `gambar` varchar(100) NOT NULL,
 `aktif` enum('Y','N') NOT NULL,
 `user` varchar(100) NOT NULL,
 PRIMARY KEY (`id`)
) ENGINE=MyISAM  DEFAULT CHARSET=latin1
AUTO_INCREMENT=22 ;
```

11. Tabel *Resident*

```
CREATE TABLE IF NOT EXISTS `resident` (
 `id` int(10) NOT NULL AUTO_INCREMENT,
 `nama` varchar(100) NOT NULL,
 `tipe` varchar(50) NOT NULL,
 `kontak` varchar(50) NOT NULL,
 `keterangan` text NOT NULL,
 `foto` varchar(100) NOT NULL,
 `aktif` enum('Y','N') NOT NULL,
 PRIMARY KEY (`id`)
) ENGINE=MyISAM  DEFAULT CHARSET=latin1
AUTO_INCREMENT=6 ;
```

12. Tabel *Resident Type*

```
CREATE TABLE IF NOT EXISTS `res_type` (
```

```
 `id` int(10) NOT NULL AUTO_INCREMENT,
 `type` varchar(50) NOT NULL,
 PRIMARY KEY (`id`)
) ENGINE=MyISAM  DEFAULT CHARSET=latin1
AUTO_INCREMENT=6 ;
```

13. Tabel *View* Semua *User*

```
CREATE ALGORITHM=UNDEFINED DEFINER=`root`@`localhost`
SQL SECURITY DEFINER VIEW `mbc-db`.`v_alluser` AS select `mbc-
db`.`admin`.`id` AS `id`,`mbc-db`.`admin`.`id_level` AS `id_level`,`mbc-
db`.`admin`.`username` AS `username`,`mbc-db`.`admin`.`nama_lengkap`
AS `nama_lengkap`,`mbc-db`.`admin`.`telpon` AS `contact`,`mbc-
db`.`admin`.`email` AS `email`,`mbc-db`.`admin`.`aktif` AS `aktif` from
`mbc-db`.`admin` union select `mbc-db`.`es`.`id` AS `id`,`mbc-
db`.`es`.`id_level` AS `id_level`,`mbc-db`.`es`.`username` AS
`username`,`mbc-db`.`es`.`nama_lengkap` AS `nama_lengkap`,`mbc-
db`.`es`.`handphone` AS `contact`,`mbc-db`.`es`.`email` AS `email`,`mbc-
db`.`es`.`aktif` AS `aktif` from `mbc-db`.`es` union select `mbc-
db`.`eo`.`id` AS `id`,`mbc-db`.`eo`.`id_level` AS `id_level`,`mbc-
db`.`eo`.`username` AS `username`,`mbc-db`.`eo`.`nama_lengkap` AS
`nama_lengkap`,`mbc-db`.`eo`.`handphone` AS `contact`,`mbc-
db`.`eo`.`email` AS `email`,`mbc-db`.`eo`.`aktif` AS `aktif` from `mbc-
db`.`eo` union select `mbc-db`.`user`.`id` AS `id`,`mbc-
db`.`user`.`id_level` AS `id_level`,`mbc-db`.`user`.`username` AS
`username`,`mbc-db`.`user`.`nama_lengkap` AS `nama_lengkap`,`mbc-
db`.`user`.`telpon` AS `contact`,`mbc-db`.`user`.`email` AS `email`,`mbc-
db`.`user`.`aktif` AS `aktif` from `mbc-db`.`user`;
```

14. Tabel Proposal

```
CREATE TABLE IF NOT EXISTS `propose` (
```

```
`id` int(10) NOT NULL AUTO_INCREMENT,
`id_user` int(10) NOT NULL,
`id_level` int(10) NOT NULL,
`user` varchar(100) NOT NULL,
`submit_date` datetime NOT NULL,
`propose_date` date NOT NULL,
`propose_by` varchar(100) NOT NULL,
`e_tittle` varchar(100) NOT NULL,
`e_concept` varchar(100) NOT NULL,
`theme` varchar(100) NOT NULL,
`dj` text NOT NULL,
`musisi` text NOT NULL,
`dancer` varchar(100) NOT NULL,
`vj` varchar(100) NOT NULL,
`mc` varchar(100) NOT NULL,
`others` varchar(100) NOT NULL,
`decor` varchar(100) NOT NULL,
`gimmick` varchar(100) NOT NULL,
`sponsor` varchar(100) NOT NULL,
`promo` text NOT NULL,
`gl` text NOT NULL,
`status` varchar(100) NOT NULL,
`msg` text NOT NULL,
`foto` varchar(100) NOT NULL,
```

```
 PRIMARY KEY (`id`)
) ENGINE=MyISAM  DEFAULT CHARSET=latin1
AUTO_INCREMENT=23 ;
```

15. Tabel *View Event*

```
CREATE ALGORITHM=UNDEFINED DEFINER=`root`@`localhost`
SQL SECURITY DEFINER VIEW `mbc-db`.`event` AS select `mbc-
db`.`propose`.`id` AS `id`,`mbc-db`.`propose`.`id_user` AS
`id_user`,`mbc-db`.`propose`.`id_level` AS `id_level`,`mbc-
db`.`propose`.`user` AS `user`,`mbc-db`.`propose`.`submit_date` AS
`submit_date`,`mbc-db`.`propose`.`propose_date` AS
`propose_date`,`mbc-db`.`propose`.`propose_by` AS `propose_by`,`mbc-
db`.`propose`.`e_tittle` AS `e_tittle`,`mbc-db`.`propose`.`e_concept` AS
`e_concept`,`mbc-db`.`propose`.`theme` AS `theme`,`mbc-
db`.`propose`.`dj` AS `dj`,`mbc-db`.`propose`.`musisi` AS `musisi`,`mbc-
db`.`propose`.`dancer` AS `dancer`,`mbc-db`.`propose`.`vj` AS `vj`,`mbc-
db`.`propose`.`mc` AS `mc`,`mbc-db`.`propose`.`others` AS
`others`,`mbc-db`.`propose`.`decor` AS `decor`,`mbc-
db`.`propose`.`gimmick` AS `gimmick`,`mbc-db`.`propose`.`sponsor` AS
`sponsor`,`mbc-db`.`propose`.`promo` AS `promo`,`mbc-
db`.`propose`.`gl` AS `gl`,`mbc-db`.`propose`.`status` AS `status`,`mbc-
db`.`propose`.`msg` AS `msg`,`mbc-db`.`propose`.`foto` AS `foto` from
`mbc-db`.`propose` where (`mbc-db`.`propose`.`status` = 'Fixed');
```

16. Tabel Laporan *Event*

```
CREATE TABLE IF NOT EXISTS `e_report` (
 `id` int(10) NOT NULL AUTO_INCREMENT,
 `id_user` int(10) NOT NULL,
```

```
  `id_level` int(10) NOT NULL,
  `id_event` int(10) NOT NULL,
  `fdc` int(10) NOT NULL,
  `hc` int(10) NOT NULL,
  `gl_eo_m` int(10) NOT NULL,
  `gl_eo_f` int(10) NOT NULL,
  `gl_mbc_m` int(10) NOT NULL,
  `gl_mbc_f` int(10) NOT NULL,
  `gl_reject` int(10) NOT NULL,
  `gl_note` text NOT NULL,
  `ck_m` int(10) NOT NULL,
  `ck_f` int(10) NOT NULL,
  `tb_wi` int(10) NOT NULL,
  `tb_ots` int(10) NOT NULL,
  `tb_rsvp` int(10) NOT NULL,
  `tb_note` text NOT NULL,
  `bev` int(50) NOT NULL,
  `cig` int(50) NOT NULL,
  `food` int(50) NOT NULL,
  `nett` int(50) NOT NULL,
  `rev` int(50) NOT NULL,
  PRIMARY KEY (`id`)
) ENGINE=MyISAM  DEFAULT CHARSET=latin1
AUTO_INCREMENT=5 ;
```

17. Tabel *View* Kontrak

```
CREATE ALGORITHM=UNDEFINED DEFINER=`root`@`localhost`
SQL SECURITY DEFINER VIEW `mbc-db`.`kontrak` AS select
`event`.`id` AS `id`,`event`.`id_user` AS `id_user`,`event`.`id_level` AS
`id_level`,`event`.`user` AS `user`,`event`.`submit_date` AS
`submit_date`,`event`.`propose_date` AS
`propose_date`,`event`.`propose_by` AS `propose_by`,`event`.`e_tittle`
AS `e_tittle`,`event`.`e_concept` AS `e_concept`,`event`.`theme` AS
`theme`,`event`.`dj` AS `dj`,`event`.`musisi` AS `musisi`,`event`.`dancer`
AS `dancer`,`event`.`vj` AS `vj`,`event`.`mc` AS `mc`,`event`.`others` AS
`others`,`event`.`decor` AS `decor`,`event`.`gimmick` AS
`gimmick`,`event`.`sponsor` AS `sponsor`,`event`.`promo` AS
`promo`,`event`.`gl` AS `gl`,`event`.`status` AS `status`,`event`.`msg` AS
`msg`,`event`.`foto` AS `foto`,`mbc-db`.`eo`.`id` AS `id_eo`,`mbc-
db`.`eo`.`username` AS `username`,`mbc-db`.`eo`.`nama_lengkap` AS
`nama_lengkap`,`mbc-db`.`eo`.`no_ktp` AS `no_ktp`,`mbc-
db`.`eo`.`perusahaan` AS `perusahaan`,`mbc-db`.`eo`.`jabatan` AS
`jabatan`,`mbc-db`.`eo`.`alamat` AS `alamat`,`mbc-db`.`eo`.`telpon` AS
`telpon`,`mbc-db`.`eo`.`handphone` AS `handphone`,`mbc-
db`.`eo`.`email` AS `email`,`mbc-db`.`eo`.`bank` AS `bank`,`mbc-
db`.`eo`.`no_rek` AS `no_rek`,`mbc-db`.`eo`.`atas_nama` AS `atas_nama`
from (`mbc-db`.`event` join `mbc-db`.`eo` on((`event`.`id_user` = `mbc-
db`.`eo`.`id`))) where (`event`.`id_level` = 3);
```

5.1.5 Implementasi Antar Muka

Implementasi antarmuka merupakan tahapan dalam memenuhi kebutuhan pengguna dalam berinteraksi dengan sistem. Fasilitas antar muka yang baik sangat membantu pemakai dalam memahami proses yang sedang dilakukan oleh sistem

tersebut dan dapat meningkatkan kinerja sistem. Berikut ini adalah implementasi antar muka yang di rancang.

5.1.5.1 Implementasi Halaman Utama

Implementasi halaman utama yang dibuat menggunakan program PHP dengan pemanggilan modular dari dokumen PHP, dimana pemanggilan modular dilakukan dari alamat URL (*uniform resource locator*) dengan menggunakan metode *gets* ($_GET).

Namun dikarenakan beberapa alasan keamanan dan kemudahan pengguna maka pada halaman utama atau halaman awal dimanipulasi pemanggilan modular dengan pemanggilan file HTML *virtual*. Manipulasi pemanggilan file HTML *virtual* ini dilakukan dengan menggunakan file .htaccess.

Menu	Deskripsi	Halaman
Home	Halaman awal (Beranda)	home.html
About	Halaman informasi tentang alamat peta tempat diselenggarakan *event*	about.html
Club	Halaman informasi tentang tempat diselenggarakan *event*	club.html
Connect with us	Halaman informasi tentang media sosial tempat penyelenggaraan *event*	contact.html
Event Album	Halaman multimedia dokumentasi *event* yang telah diselengarakan	album.html
Gallery	Halaman informasi galeri multimedia tempat diselenggarakan *event*	gallery.html

Resident	Halaman informasi *Resident Vanue*	resident.html
News	Halaman informasi berita	news.html
Sign In	Halaman untuk melakukan login dan melakukan registrasi	login.html

Tabel 5.1 *Implementasi Halaman Utama*

5.1.5.2 Implementasi Halaman *Event Organizer*

Implementasi halaman untuk *event organizer* yang dibuat menggunakan program PHP yang dimanipulasi menjadi file HTML yang dapat dilihat pada tabel dibawah ini :

Menu	Deskripsi	Halaman
Home	Halaman awal sistem (Beranda)	home.html
Profile	Halaman untuk memperbaharui profil	editprofil.html
Change Picture	Halaman untuk memperbaharui foto profil	changeava.html
Change Password	Halaman untuk memperbaharui *password*	changepwd.html
Propose	Halaman informasi proposal yang telah dimasukkan	vpropose.html
Propose ACC	Halaman informasi proposal yang telah disetujui	accpropose.html
Submit Event	Halaman untuk memasukkan proposal *event*	propose.html
Event	Halaman informasi *event* yang sudah atau akan diselenggarakan	event.html
RSVP	Halaman untuk melakukan reservasi *online*	rsvp.html

Sign Out	Halaman untuk keluar dari sistem	logout.html

Tabel 5.2 *Implementasi Halaman Event Organizer*

5.1.5.3 Implementasi Halaman Administrator

Implementasi halaman untuk admin yang dibuat menggunakan program PHP yang dimanipulasi menjadi file HTML yang dapat dilihat pada tabel dibawah ini :

Menu	Deskripsi	Halaman
Home	Halaman awal sistem (Beranda)	home.html
Profile	Halaman untuk memperbaharui profil	editprofil.html
Change Picture	Halaman untuk memperbaharui foto profil	changeava.html
Change Password	Halaman untuk memperbaharui *password*	changepwd.html
Reg *Admin*	Halaman untuk melakukan registrasi admin	regadmin.html
Reg *Staff*	Halaman untuk melakukan registrasi *staff*	regis.html
Reset Password	Halaman untuk mereset password	resetpwd.html
Propose	Halaman untuk mengelola proposal yang telah dimasukkan	vpropose.html
Propose ACC	Halaman informasi proposal yang telah disetujui	accpropose.html
Submit Event	Halaman untuk memasukkan proposal *event*	propose.html
Event	Halaman informasi event yang sudah atau akan diselenggarakan	event.html
Submit Event	Halaman untuk memasukkan laporan event	ereport.html

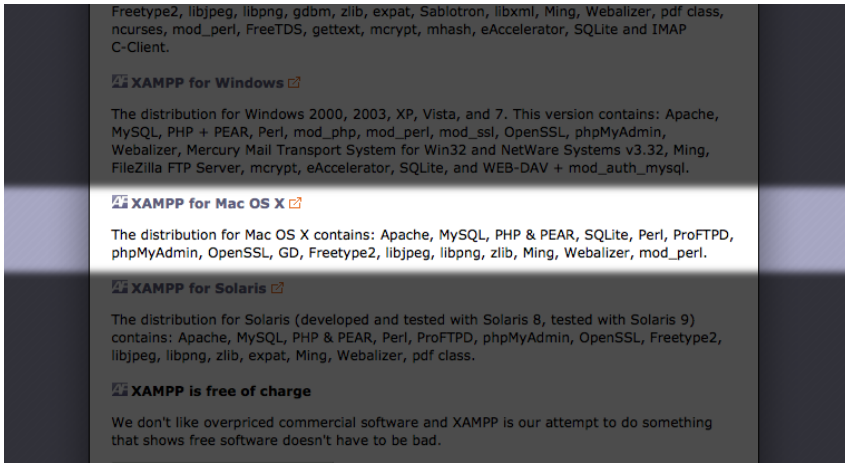
Report		
Event Report	Halaman untuk mengelola laporan event	ereview.html
Reservation	Halaman untuk mengelola reservasi yang telah dimasukkan	rsvp.html
Submit RSVP	Halaman untuk memasukkan reservasi	srsvp.html
All User	Halaman informasi semua pengguna yang telah melakukan registrasi	vuall.html
Embassy Staff	Halaman untuk mengelola pengguna dengan level 2 (*embassy staff)*	ves.html
Event Organizer	Halaman untuk mengelola pengguna dengan level 3 (*event organizer*)	veo.html
Member	Halaman untuk mengelola pengguna dengan level 4 (*member*)	vuser.html
Headline	Halaman untuk mengedit *headline* pada halaman utama	headline.html
News	Halaman untuk mengelola berita pada halaman utama	vnews.html
Submit News	Halaman untuk memasukkan berita terbaru	news.html
Resident	Halaman untuk mengelola *Resident* pada halaman utama	vres.html
Submit Resident	Halaman untuk memasukkan *Resident* baru	res.html
Gallery	Halaman untuk mengelola galeri album	gallery.html
Create Album	Halaman untuk membuat album baru untuk galeri album	album.html
Sign Out	Halaman untuk keluar dari sistem	logout.html

Tabel 5.3 *Implementasi Halaman Administrator*

5.1.6 Implementasi Instalasi Program

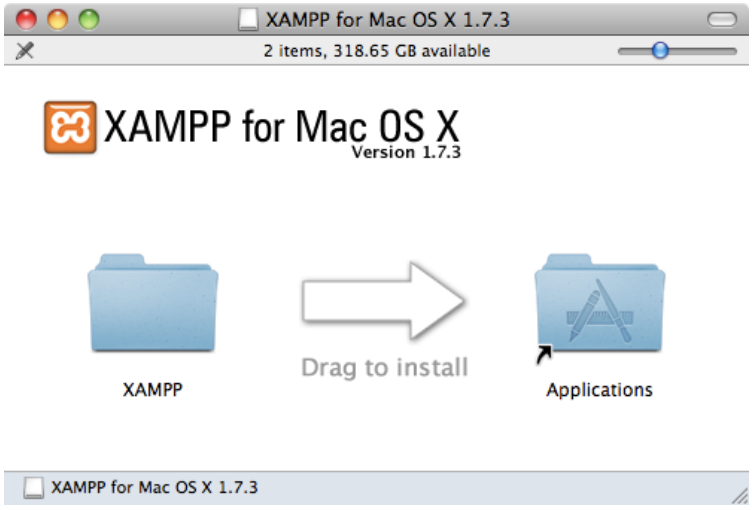
Adapun implementasi instalasi program bagaimana cara menggunakan program sistem informasi penjadwalan *event* berbasis web pada PT. Mitra Bandung Consortium adalah sebagai berikut :

1. Download paket aplikasi XAMPP pada halaman www.apachefriends.org



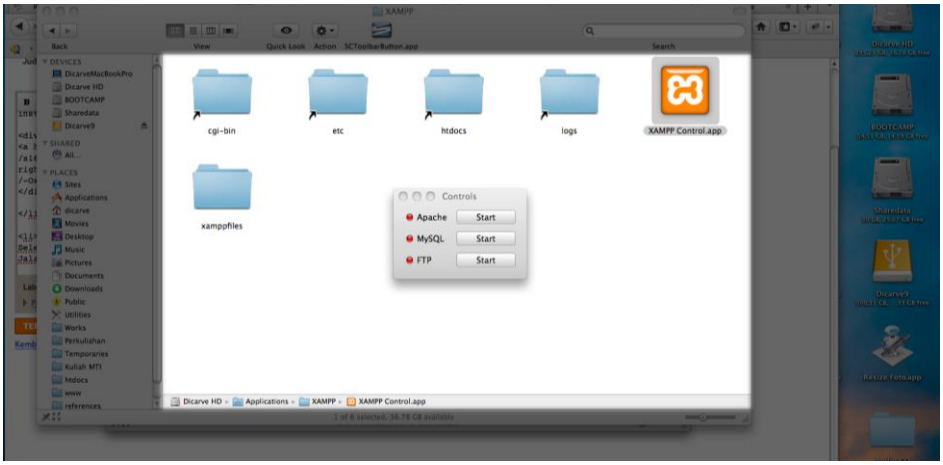
Gambar 5.1 *Download Aplikasi XAMPP*

2. Setelah selesai di-*download*, klik dua kali pada *file image* .dmg XAMPP untuk membuka *installer*-nya. *Drag* folder XAMPP ke folder *Applications* untuk meng*install*.



Gambar 5.2 *Instalasi XAMPP*

3. Masuk ke folder *Applications* dan cari folder XAMPP. Jalankan program "XAMPP *Control*". Akan ada jendela aplikasi kecil yang berisi tombol untuk menjalan Apache, MySQL dan FTP.



Gambar 5.3 *Buka Aplikasi XAMPP*

4. Jalankan service aplikasi XAMPP dengan mengklik tombol start, dan lampu indikator akan berubah menjadi hijau.



Gambar 5.4 *Menjalankan Service XAMPP*

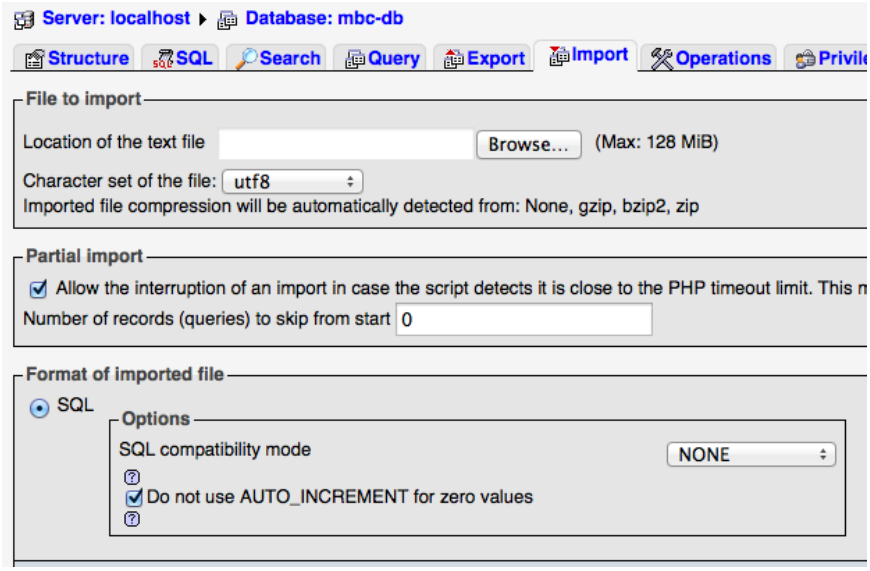
5. Buka aplikasi *browser* seperti Mozilla Firefox, Google Chrome atau Safari dan masukkan alamat URL http://localhost. Jika muncul halaman dari XAMPP, maka instalasi XAMPP telah berhasil.

6. Kemudian masukkan alamat URL http://localhost/phpmyadmin untuk akses *database* GUI, dan buatlah sebuah *database* dengan nama ‘mbc-db’.



Gambar 5.5 *Membuat Database*

7. *Import database* dengan mengklik menu import pada PhpMyAdmin. Kemudian klik *browse* dan carilah nama *file* mbc-db.sql. Klik *go* untuk mengimport *database* tersebut.



Gambar 5.6 *Import Database*

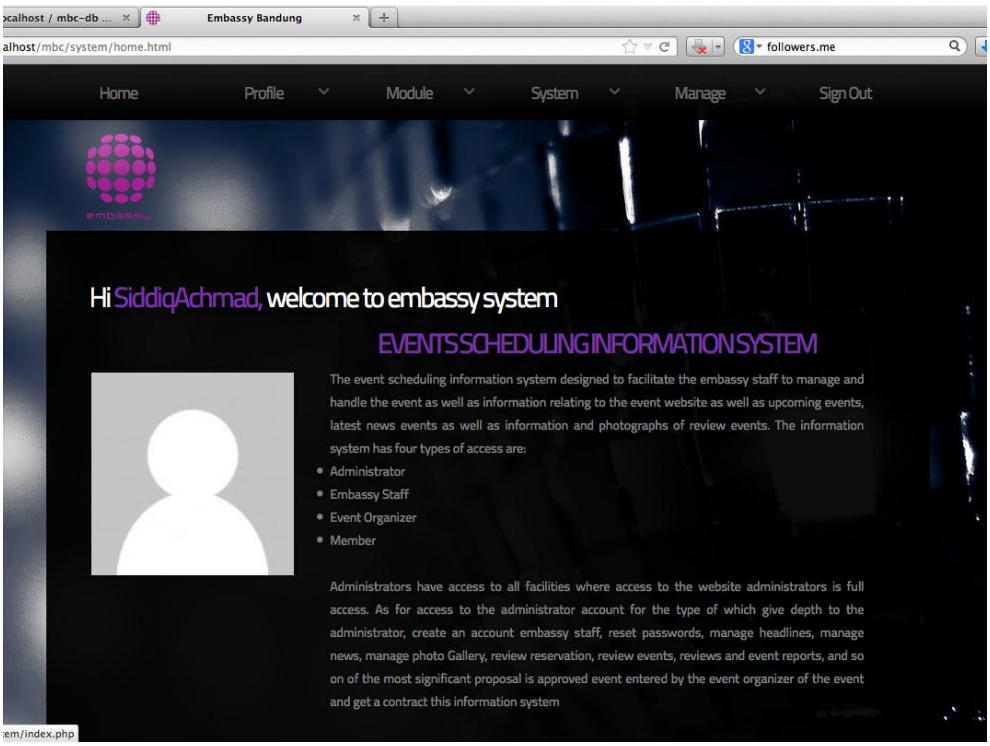
8. Proses instalasi selesai, dan aplikasi siap untuk digunakan.

5.1.7 Penggunaan Program Aplikasi

Penggunaan program aplikasi ini bertujuan untuk menjelaskan tahapan dalam penggunaan program serta tampilan dari program yang berguna sebagai panduan penggunaan program. Berikut ini dijelaskan cara penggunaan program beserta tampilannya :

5.1.7.1 Halaman *Home* Sistem

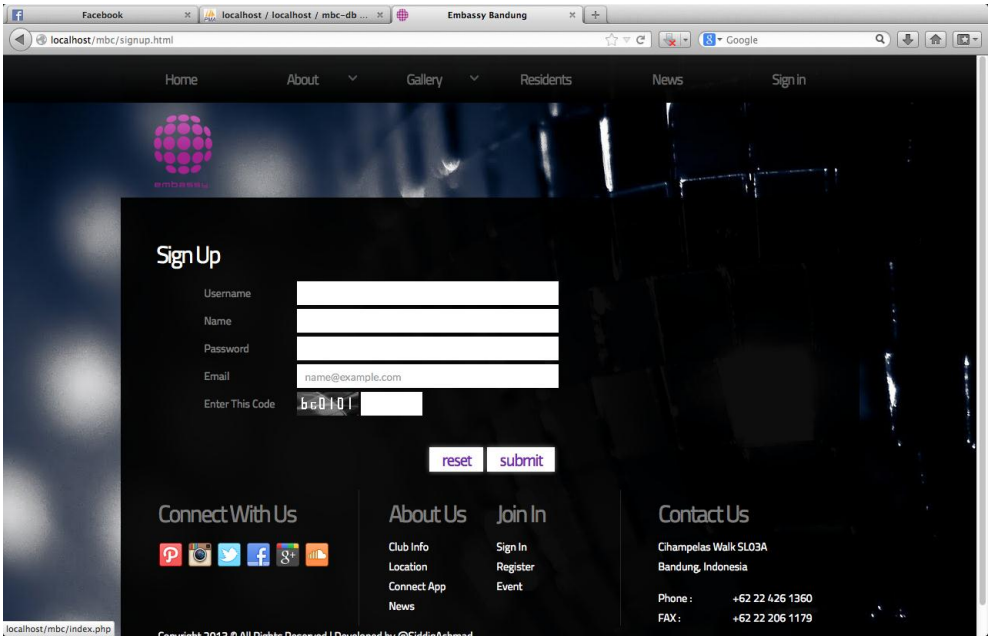
Pada halaman *home* sistem *website* ini berisikan informasi tentang sistem penjadwalan *event* dan penggunaanya. Pada halaman *home* ini juga terdapat informasi tentang user yang sedang digunakan dan juga terdapat foto user yang sedang digunakan.



Gambar 5.7 *Tampilan Halaman Home Sistem*

5.1.7.2 Halaman Registrasi

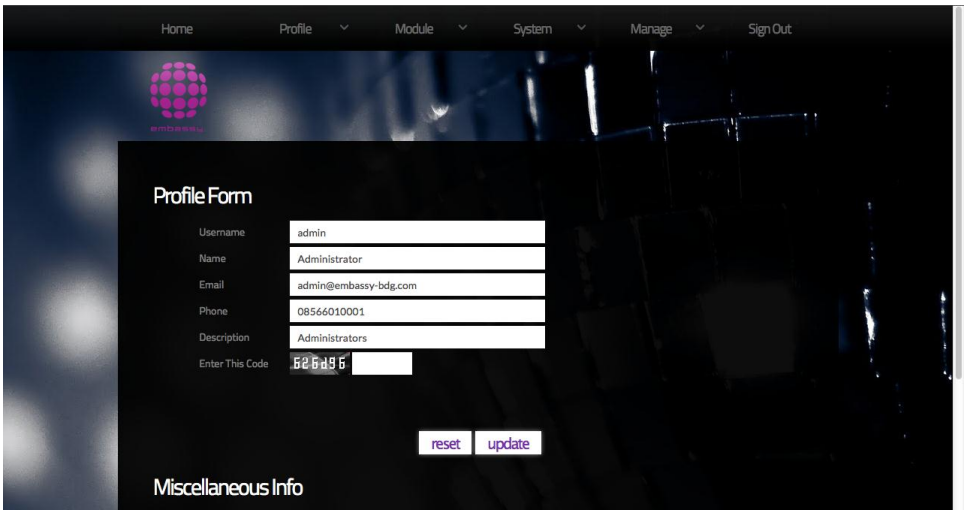
Pada halaman registrasi ini digunakan untuk melakukan pendaftaran pengguna baru. Adapun tampilan form registrasi adalah sebagai berikut:



Gambar 5.8 *Tampilan Form Registrasi*

5.1.7.3 Halaman *Edit* Profil

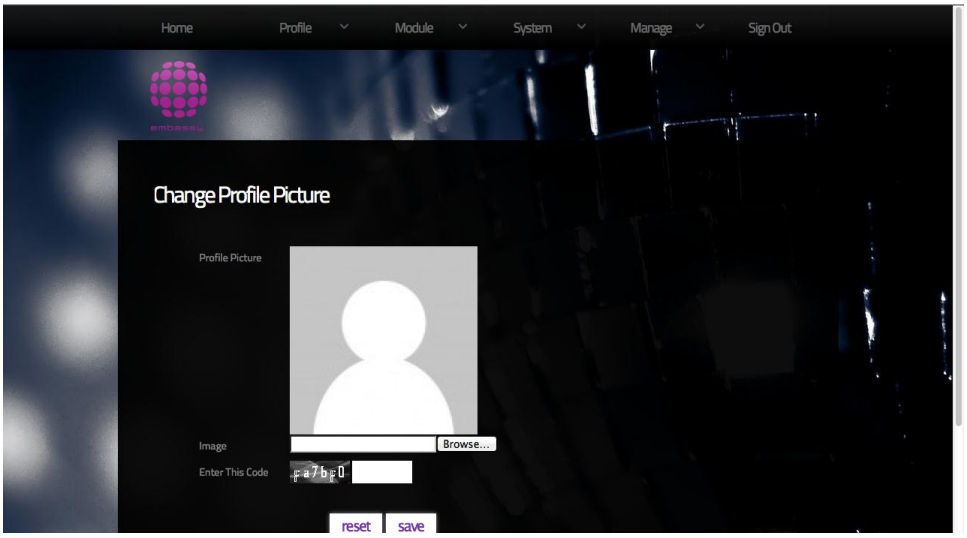
Pada halaman *edit* profil ini digunakan untuk melakukan pembaharuan data pengguna. Adapun tampilan form *edit* profil adalah sebagai berikut:



Gambar 5.9 *Tampilan Form Edit Profil*

5.1.7.4 Halaman Ganti Foto Profil

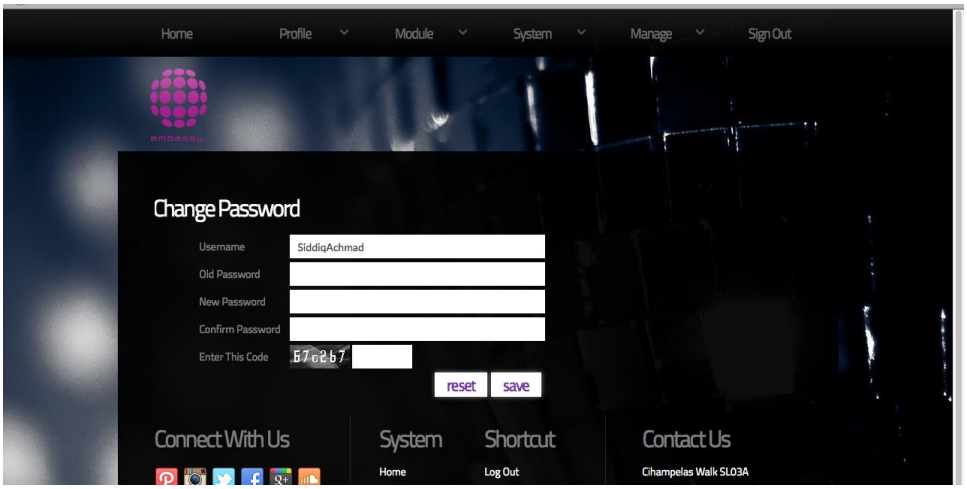
Pada halaman ganti foto profil ini digunakan untuk melakukan pembaharuan data pengguna. Adapun tampilan form ganti foto profil adalah sebagai berikut:



Gambar 5.10 *Tampilan Form Ganti Foto Profil*

5.1.7.5 Halaman Ganti *Password*

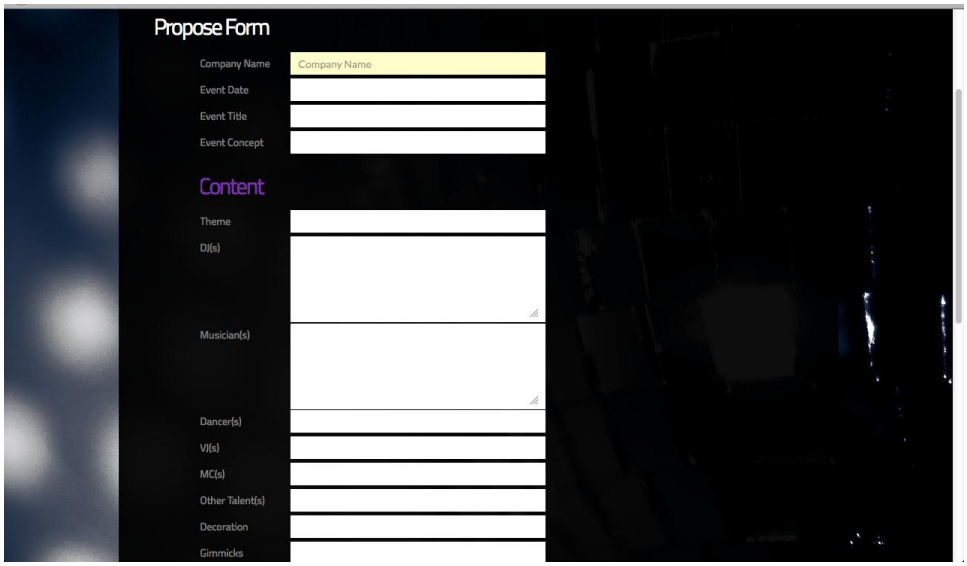
Pada halaman ganti password ini digunakan untuk melakukan pembaharuan password pengguna. Adapun tampilan halaman ganti password profil adalah sebagai berikut:



Gambar 5.11 *Tampilan Form Ganti Password*

5.1.7.6 Halaman *Submit* Proposal

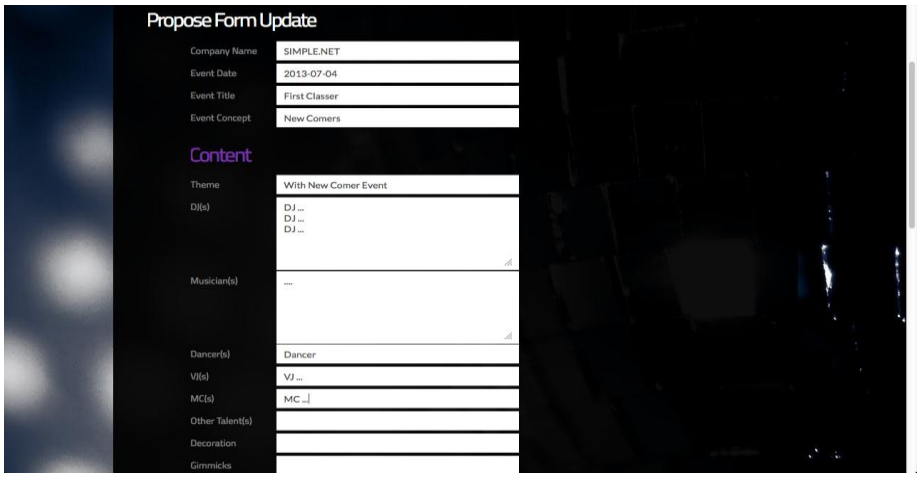
Pada halaman *submit* proposal ini digunakan untuk memasukkan proposal *event*. Adapun tampilan form *submit* proposal adalah sebagai berikut:



Gambar 5.12 *Tampilan Form Submit Proposal*

5.1.7.7 Halaman *Edit* Proposal

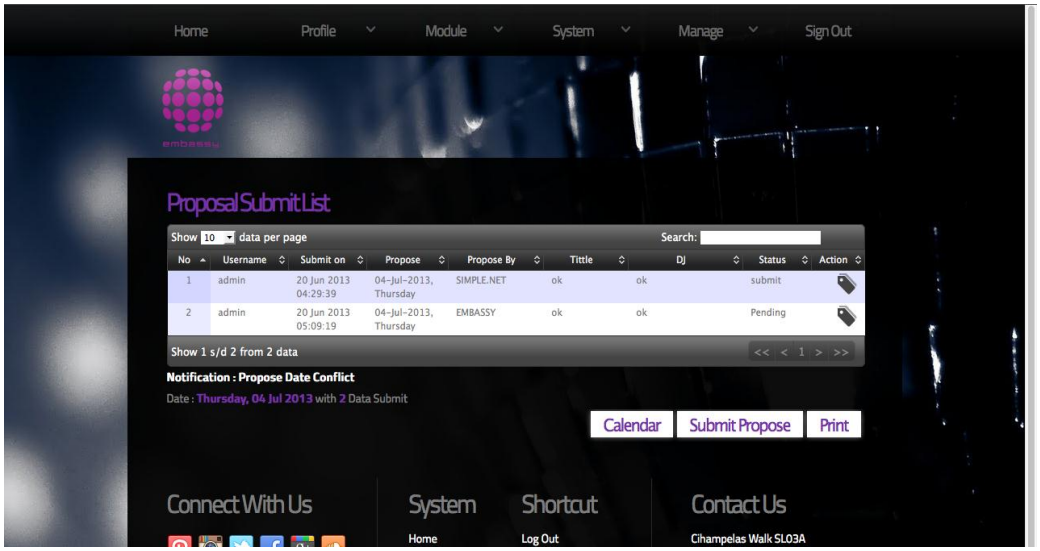
Pada halaman *edit* proposal ini digunakan untuk melakukan pembaharuan data proposal *event*. Adapun tampilan halaman *edit* proposal adalah sebagai berikut:



Gambar 5.13 *Tampilan Form Edit Proposal*

5.1.7.8 Halaman *View* Proposal

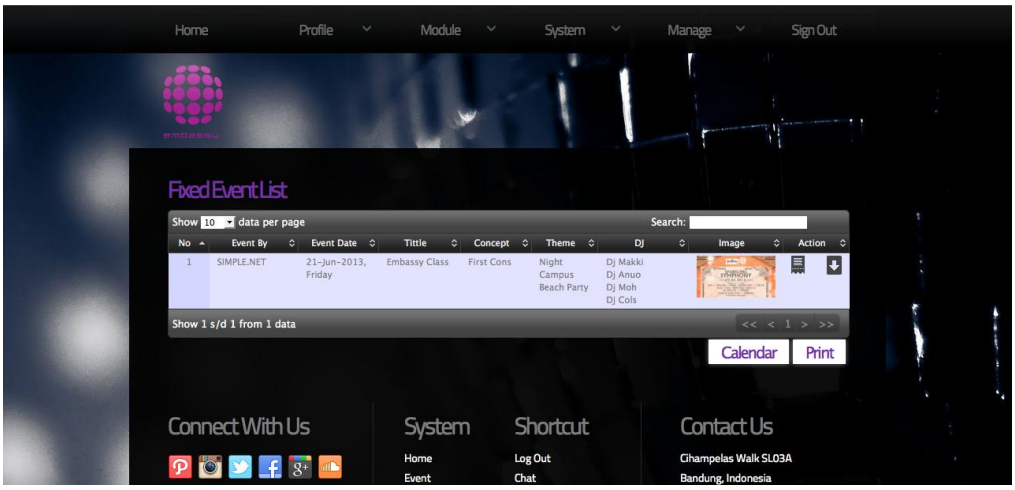
Pada halaman *view* proposal ini digunakan untuk melihat proposal *event*. Adapun tampilan *view* proposal adalah sebagai berikut:



Gambar 5.14 *Tampilan View Proposal*

5.1.7.9 Halaman *View Event*

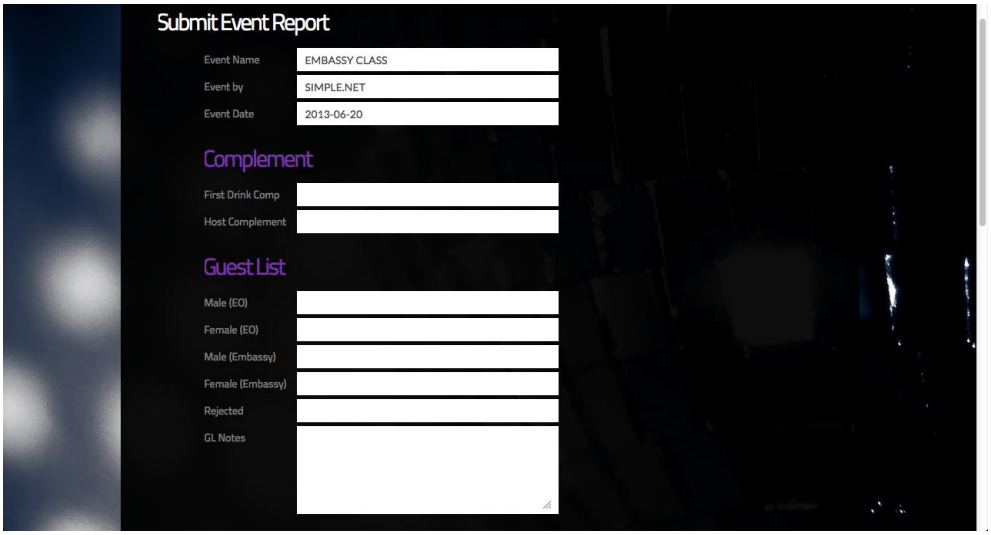
Pada halaman *view event* ini digunakan untuk melakukan melihat data *event*. Adapun tampilan halaman *view event* adalah sebagai berikut:



Gambar 5.15 *Tampilan View Event*

5.1.7.10 Halaman *Submit* Laporan *Event*

Pada halaman *submit* laporan *event* ini digunakan untuk memasukkan laporan *event*. Adapun tampilan *submit* laporan *event* adalah sebagai berikut:



Gambar 5.16 *Tampilan Form Laporan Event*

5.1.7.11 Halaman *View* Laporan *Event*

Pada halaman *view* laporan *event* ini digunakan untuk melakukan melihat laporan *event*. Adapun tampilan halaman *view* laporan *event* adalah sebagai berikut:



Gambar 5.17 *Tampilan View Laporan Event*

5.2 Pengujian

Pengujian merupakan bagian yang penting dalam siklus pembangunan perangkat lunak. Pengujian dilakukan untuk menjamin kualitas dan juga mengetahui kelemahan dari perangkat lunak. Tujuan dari pengujian ini adalah untuk menjamin bahwa perangkat lunak yang dibangun memiliki kualitas yang handal, yaitu merepresentasikan kajian pokok dari spesifikasi, analis perancangan dan pengkodean dari perangkat lunak itu sendiri.

5.2.1 Rencana Pengujian

Rencana pengujian pengelolaan sistem informasi penjadwalan *event* berikut menggunakan data uji berupa data masukan dari pengguna pada aplikasi yang telah dibuat.

Kelas Uji	Butir Uji	Tingkat Pengujian	Jenis Pengujian
Login	*Login* Admin	Integrasi	*Black box*
	Login Staff	Integrasi	*Black box*
	Login Event Organizer	Integrasi	*Black box*
	Login Member	Integrasi	*Black box*
Registrasi	Registrasi Admin	Integrasi	*Black box*
	Registrasi *Staff*	Integrasi	*Black box*
	Registrasi *Event Organizer*	Integrasi	*Black box*
	Registrasi *Member*	Integrasi	*Black box*
Input	*Submit* Proposal	Integrasi	*Black box*
	Submit Laporan *Event*	Integrasi	*Black box*
Edit	*Edit* Proposal	Integrasi	*Black box*
	Edit Pengguna	Integrasi	*Black box*
View	*View Event*	Integrasi	*Black Box*

	View Pengguna	Integrasi	*Black Box*
Validasi	Validasi *Login*	Integrasi	*Black box*
	Validasi Registrasi	Integrasi	*Black box*
	Validasi *Submit*	Integrasi	*Black box*
	Validasi *Edit*	Integrasi	*Black box*

Tabel 5.6 *Tabel Rencana Pengujian*

5.2.2 Kasus dan Hasil Pengujian

5.2.2.1 Pengujian *Login*

Pengujian *login* terbagi menjadi empat bagian, yaitu *login* admin, *login staff, login event organizer, login member*. Namun pada kasus login yang berbeda akses ini menggunakan sistem pengecekan validasi dan verifikasi yang sama pada tiap bagiannya. Dengan sistem pengecekan validasi dan verifikasi yang sama, maka tidak semua akses pengguna perlu diuji.

Beberapa hasil pengujian pada kasus pengujian *login* yang telah dilakukan dengan tiga variabel yaitu *username, password* dan *level* baik dengan data normal maupun data yang salah dengan beberapa variasi pengujian yang dapat dilihat pada tabel berikut :

Kasus dan Hasil Uji *Login* (Data Normal)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Username : admin *Password* : admin *level* : Administrator	Dapat masuk ke halaman utama sistem	Dapat masuk kedalam halama utama sistem seperti yang diharapkan	[X] diterima [ ] ditolak

Username : siddiq *Password* : siddiq *level* : Embassy *Staff*	Dapat masuk ke halaman utama sistem	Dapat masuk kedalam halama utama sistem seperti yang diharapkan	[X] diterima [ ] ditolak
Kasus dan Hasil Uji *Login* (Data Salah)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Username : minda *Password* : minda *level* : Administrator	Tidak dapat masuk kedalam halaman sistem	Menampilkan pesan login gagal	[X] diterima [ ] ditolak
Username : siddiq *Password* : siddiq *level* : (dikosongkan)	Tidak dapat masuk kedalam halaman sistem dan tidak akan diproses	Tidak ada proses dan perubahan apapun	[X] diterima [ ] ditolak

Tabel 5.7 *Kasus dan Hasil Uji Login*

5.2.2.2 Pengujian Registrasi

Adapun kasus dan hasil pengujian registrasi pengguna dapat dilihat pada tabel berikut :

Kasus dan Hasil Uji Registrasi (Data Normal)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Mengisikan data registrasi dengan lengkap dan benar	Data dapat tersimpan kedalam *database*	Data tersimpan kedalam *database* dan muncul pesan berhasil registrasi	[X] diterima [ ] ditolak
Kasus dan Hasil Uji Registrasi (Data Salah)			

Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Mengisikan data registrasi dengan tidak lengkap	Data tidak tersimpan kedalam *database*	Data tidak tersimpan kedalam *database* dan *textbox* yang tidak lengkap akan berubah warna	[X] diterima [ ] ditolak
Mengisikan data registrasi dengan lengkap namun dengan captcha yang salah	Data tidak tersimpan kedalam *database*	Data tidak tersimpan kedalam *database* dan muncul pesan kesalahan captcha	[X] diterima [ ] ditolak

Tabel 5.8 *Kasus dan Hasil Uji Resigtrasi*

5.2.2.3 Pengujian *Input* Data

Adapun kasus dan hasil pengujian *input* data dimana pada kasus ini diambil sampel pengujian *input* data proposal yang dapat dilihat pada tabel berikut :

Kasus dan Hasil Uji *Input* Data (Data Normal)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Mengisikan data dengan lengkap dan benar	Data dapat tersimpan kedalam *database*	Data tersimpan kedalam *database* dan muncul pesan berhasil meng*input* data	[X] diterima [ ] ditolak
Kasus dan Hasil Uji *Input* Data (Data Salah)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Mengisikan data	Data tidak	Data tidak	[X] diterima

dengan tidak lengkap	tersimpan kedalam *database*	tersimpan kedalam *database* dan *textbox* yang tidak lengkap akan berubah warna	[ ] ditolak
Mengisikan data dengan lengkap namun dengan captcha yang salah	Data tidak tersimpan kedalam *database*	Data tidak tersimpan kedalam *database* dan muncul pesan kesalahan captcha	[X] diterima [ ] ditolak

Tabel 5.9 *Kasus dan Hasil Uji Input Data*

5.2.2.4 Pengujian *Edit* Data

Adapun kasus dan hasil pengujian *edit* data dimana pada kasus ini diambil sampel pengujian *edit* data proposal yang dapat dilihat pada tabel berikut :

Kasus dan Hasil Uji *Edit* Data (Data Normal)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Memperbaharui data dengan lengkap dan benar	Data dapat tersimpan dengan pembaharuan	Data tersimpan kedalam *database* dan diperbaharui	[X] diterima [ ] ditolak
Kasus dan Hasil Uji *Edit* Data (Data Salah)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Memperbaharui data dengan *textbox* yang masih tidak lengkap	Data tidak tersimpan dengan pembaharuan	Data tidak diperbaharui dan *textbox* yang tidak lengkap akan berubah warna	[X] diterima [ ] ditolak
Memperbaharui	Data tidak	Data tidak	[X] diterima

data dengan lengkap namun dengan captcha yang salah	tersimpan kedalam *database*	diperbaharui dan muncul pesan kesalahan captcha	[ ] ditolak

Tabel 5.10 *Kasus dan Hasil Uji Edit Data*

5.2.2.5 Pengujian Validasi Data

Adapun kasus dan hasil pengujian validasi data dimana pada kasus ini diambil sampel pengujian validasi data registrasi yang dapat dilihat pada tabel berikut :

Kasus dan Hasil Uji Validasi Data (Data Normal)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Mengisikan data dengan lengkap, benar dan valid	Data dapat tersimpan kedalam *database*	Data tersimpan kedalam *database* dan muncul pesan berhasil melakukan registrasi	[X] diterima [ ] ditolak
Kasus dan Hasil Uji *Input* Data (Data Salah)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Mengisikan data dengan lengkap namun data telpon atau email yang tidak valid	Data tidak tersimpan kedalam *database*	Data tidak tersimpan kedalam *database* dan *textbox* yang tidak valid akan berubah warna	[X] diterima [ ] ditolak
Mengisikan data dengan lengkap	Data tidak tersimpan kedalam	Data tidak tersimpan kedalam	[X] diterima [ ] ditolak

namun dengan *username* yang sudah ada	*database*	*database* dan muncul pesan bahwa *username* sudah digunakan	

Tabel 5.11 *Kasus dan Hasil Uji Validasi Data*