BAB V

IMPLEMENTASI DAN PENGUJIAN SISTEM

5.1 Implementasi

Implementasi adalah tahapan penerapan rancangan program yang telah dibuat pada pembahasan Bab IV dan didesain secara rinci, kemudian dilakukan tahap implementasi. sebelumnya 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 perangkat lunak atau sistem ini dapat menghasilkan sistem informasi reservasi yang sesuai dengan yang diharapkan.

5.1.1 Batasan Implementasi

Dalam mengimplementasikan perangkat lunak, batasan sebagai kontrol pengendalian aplikasi ini ada beberapa hal yang perlu dibatasi dalam implementasi sistem ini yaitu :

1. Perangkat lunak yang digunakan dalam pengimplementasian basis data ini adalah *MySQL* yang telah terkompilasi kedalam sebuah perangkat lunak bebas yaitu *apache web server*.
2. Sitem informasi reservasi *online* paket wisata pada Brema *Tour&Travel* ini hanya sebatas pengeolaan pandaftaran agen, reservasi hotel, paket *tours*,
3. Sistem ini dibangun menggunakan bahasa pemrograman PHP.

5.1.2 Kebutuhan Perangkat Lunak

Perangkat lunak *(Software)* digunakan untuk mendukung sistem operasi (SO) dan aplikasi *Database*. Perangkat lunak yang dibutuhkan oleh Komputer server maupun Komputer *client*, adapun kebutuhan perangkat lunak yang digunakan sebagai berikut :

1. Komputer *Server*

   Komputer *server* adalah komputer yang mempunyai kemampuan yang lebih dari komputer *client*, dimana didalamnya tersimpan data-data yang akan didistribusikan. Dimana *software* yang dibutuhkan diantaranya :

   a. Sistem Operasi *Linux*

   b. Modem / jaringan *internet*

   c. Notepad ++

   d. Apache webserver

   e. MySQL.

   f. Css Syle

   g. Menggunakan bahasa *Php, Java Script* dan *Html*

2. Komputer *Client*

   Komputer *Client* adalah komputer yang didalamnya tersimpan data-data yang dapat digunakan secara perorangan, mengumpulkan dan menampilkan data, serta menyimpan data ke *server*. Dimana *software* yang dibutuhkan diantaranya :

   a. Sistem Operasi *Windows 7*

   b. Modem / jaringan *internet*

5.1.3 Kebutuhan Perangkat Keras

Untuk dapat menjalankan aplikasi yang dirancang maka dibutuhkan suatu perangkat keras sebagai penunjangnya. Adapun perangkat kebutuhan perangkat keras yang dibutuhkan tersebut adalah sebagai berikut :

1. *Server* :
   a. *Processor* Minimal *dual core*
   b. Menggunakan minimal RAM 1 Gb.
   c. *Harddisk*500Gb.
   d. *Keyboard*, *Monitor* dan *Mouse.*
   e. *Modem /* jaringan internet.
2. *Client*
   a. *Processor* Minimal *dual core*
   b. Menggunakan minimal RAM 1 Gb.
   c. *Harddisk*500Gb.
   d. *Keyboard*, *Monitor* dan *Mouse.*
   e. *Modem /* jaringan internet.

5.1.4 Implementasi Basis Data (Sintaks SQL)

Implementasi basis data dilakukan dengan menggunakan bahasa *SQL*, dimana aplikasi pemrograman yang digunakan adalah *MySQL*, implementasi basis datanya dalam bahasa *SQL* adalah sebagai berikut :

1. Tabel agen_ads

   SELECT * FROM `agent_ads` WHERE 1

   ‘id’ int(11) NOT NULL AUTO_INCREMENT

‘agen_id int(11) NOT NULL ,

‘Code’ text NOT NULL,

2. agent_level

‘id’ int(11) NOT NULL AUTO_INCREMENT,

‘pack_name’ varchar(255) latin1_swedish_ci NOT NULL,

‘pack description’ text latin1_swedish_ci NOT NULL,

‘img’ varchar(255) latin1_swedish_ci NOT NULL,

‘max_pic’ int(11) NOT NULL,

‘coupons’ int(11) NOT NULL,

‘ads’ int(11) NOT NULL,

‘max_car_rent_pack’ int(11) NOT NULL,

‘max_hotels’ int(11) NOT NULL,

‘max_tours’ int(11) NOT NULL,

‘fea_pack’ int(11) NOT NULL,

‘newsletter’ int(11) NOT NULL,

‘pages’ int(11) NOT NULL,

3. biling_period

‘id’ int(11) NOT NULL AUTO_INCREMENT,

‘pack_id’ int(11) NOT NULL,

‘period’ int(11) NOT NULL,

‘price’ varchar(255) NOT NULL,

‘currency, varchar(255) NOT NULL,

4. Country

   ‘country_id’ int(11) NOT NULL AUTO_INCREMENT,

   ‘country_name’ varchar(100) latin1_swedish_ci NOT NULL,

5. destination

   ‘destination_id’ int(11) NOT NULL AUTO_INCREMENT,

   ‘destination_name’ varchar(150) latin1_swedish_ci NOT NULL,

   ‘country_id’ int(11) NOT NULL,

6. Hotels

   ‘id’ int(11) NOT NULL AUTO_INCREMENT,

   ‘hotel_name’ varchar(255) latin1_swedish_ci,

   ‘agent_id’ int(11) NOT NULL,

   ‘rooms’ int(11) NOT NULL

   ‘bathrooms’ int(11) NOT NULL

   ‘price’int(11) NOT NULL,

   ‘currency’ varchar(255) latin1_swedish_ci NOT NULL,

   ‘other’ text latin1_swedish_ci NOT NULL,

7. Hotel_pic

   ‘id’ int(11) NOT NULL AUTO_INCREMENT,

   ‘hotel_id’ int(11) NOT NULL,

   ‘img’ varchar(255) latin1_swedish_ci NOT NULL,

8. Invoices

   ‘id’ int(11) NOT NULL AUTO_INCREMENT,

   ‘txn’ varchar(255) latin1_swedish_ci NOT NULL,

‘agent_id’ int(11) NOT NULL

‘price’ varchar(255) latin1_swedish_ci NOT NULL,

‘currency’ varchar(255) latin1_swedish_ci NOT NULL,

‘period’ int(11) NOT NULL,

‘level_id’ int(11) NOT NULL,

‘indate’ varchar(255) latin1_swedish_ci NOT NULL,

‘renew’ varchar(255) latin1_swedish_ci NOT NULL,

‘status’ varchar(255) latin1_swedish_ci NOT NULL,

9. News

‘id’ int(11) NOT NULL AUTO_INCREMENT,

‘title’ varchar(128) latin1_swedish_ci NOT NULL,

‘text’ text latin1_swedish_ci NOT NULL,

10. Orders

‘id’ int(50) NOT NULL AUTO_INCREMENT,

‘type’ varchar(255) latin1_swedish_ci NOT NULL,

‘tour_id’ int(11) NOT NULL,

‘pack_id’ int(50) NOT NULL,

‘hotel_id’ int(11) NOT NULL,

‘period’ int(11) NOT NULL,

‘agent_id’ int(11) NOT NULL,

‘client_name’ varchar(250) latin1_swedish_ci NOT NULL,

‘client_email’ varchar(250) latin1_swedish_ci NOT NULL,

‘client_tel’ varchar(250) latin1_swedish_ci NOT NULL,

‘message’ text latin1_swedish_ci NOT NULL,

‘indate’ varchar(255) latin1_swedish_ci NOT NULL,

11. Package_detail

‘package_id’ int(11) NOT NULL AUTO_INCREMENT,

‘agent_id’ int(11) NOT NULL,

‘package_code’ varchar(10) latin1_swedish_ci NOT NULL,

‘hotel_id’ int(11) NOT NULL,

‘tour_id’ int(11) NOT NULL,

‘period’ varchar(255) latin1_swedish_ci NOT NULL,

‘package_name’ varchar(255) latin1_swedish_ci NOT NULL,

‘destination_id’ int(11) NOT NULL,

‘country_id’ int(11) NOT NULL,

‘package_description’ text latin1_swedish_ci YES NULL,

‘package_sumary’ text latin1_swedish_ci NOT NULL,

‘price’ int(11) YES NULL,

‘thumbnail’ varchar(255) latin1_swedish_ci YES NULL,

‘currencuy’ varchar(45) latin1_swedish_ci YES NULL,

‘indate’ varchar(255) latin1_swedish_ci NOT NULL,

‘featured’ int(11) NOT NULL,

‘status’ int(11) NOT NULL,

12. Pages

‘id’ int(11) NOT NULL AUTO_INCREMENT,

‘title’ varchar(250) latin1_swedish_ci NOT NULL,

‘content’ text latin1_swedish_ci NOT NULL,

‘indate’ varchar(250) latin1_swedish_ci NOT NULL,

‘agent_id’ int(11) NOT NULL,

‘level’ int(11) NOT NULL,

‘status’ int(11) NOT NULL,

13. Ratings

‘id’ varchar(11) latin1_swedish_ci NOT NULL,

‘total_votes’ int(11) NOT NULL,

‘total_value’ int(11) NOT NULL,

“used_ips’ longtext latin1_swedish_ci NOT NULL,

‘indate’ varchar(255) latin1_swedish_ci YES NULL,

14. Reviews

‘id’ int(11) NOT NULL AUTO_INCREMENT,

‘type’ varchar(255) latin1_swedish_ci NOT NULL,

‘tour_id’ int(11) NOT NULL,

‘hotel_id’ int(11) NOT NULL,

‘pack_id’ int(11) NOT NULL,

‘agent_id’ int(11) NOT NULL,

‘name’ varchar(255) latin1_swedish_ci NOT NULL,

‘title’ varchar(255) latin1_swedish_ci NOT NULL,

‘description text latin1_swedish_ci NOT NULL,

‘reate’ varchar(225) latin1_swedish_ci NOT NULL,

‘indate’ varchar(255) latin1_swedish_ci NOT NULL,

‘status’ int(11) NOT NULL,

15. Review_pic

‘item_id’ int(11) NOT NULL,

‘review_id’ int(11) NOT NULL,

‘picture’ varchar(255) latin1_swedish_ci NOT NULL,

16. Screenshots

‘id’ int(11) NOT NULL,

‘pack_id’ nit(50) NOT NULL,

‘name’ varchar(255) latin1_swedish_ci NOT NULL,

17. Settings

‘name’ varchar(250) latin1_swedish_ci NOT NULL,

‘value’ text’ latin1_swedish_ci NOT NULL,

18. Subscription

‘id’ int(11) NOT NULL,

‘name’ varchar(255) latin1_swedish_ci NOT NULL,

‘email’ varchar(255) latin1_swedish_ci NOT NULL,

19. Tours

‘id’ int(11) NOT NULL,

‘agent_id’ int(11),

‘tour_name’ varchar(255) latin1_swedish_ci NOT NULL,

‘description’ text latin1_swedish_ci NOT NULL,

20. Tour_pic

‘id’ int(11) NOT NULL,

‘tour_id’ int(11) NOT NULL,

‘img’ varchar(255) latin1_swedish_ci NOT NULL,

21. Useraccounts

‘id’ int(11) NOT NULL AUTO_INCREMENT,

‘username’ varchar(50) latin1_swedish_ci NOT NULL,

‘password’ varchar(35) latin1_swedish_ci NOT NULL,

‘company_name’ varchar(255) latin1_swedish_ci NOT NULL,

‘full_name’ varchar(255) latin1_swedish_ci NOT NULL,

‘email’ varchar(255) latin1_swedish_ci NOT NULL,

‘phone’ varchar(255 latin1_swedish_ci NOT NULL,

‘country’ int(11) NOT NULL,

‘city’ varchar(255) latin1_swedish_ci NOT NULL,

‘address’ text latin1_swedish_ci NOT NULL,

‘zip’ varchar(255) latin1_swedish_ci NOT NULL,

‘join’ varchar(255) latin1_swedish_ci NOT NULL,

‘level’ int(11) NOT NULL,

‘status’ int(11) NOT NULL,

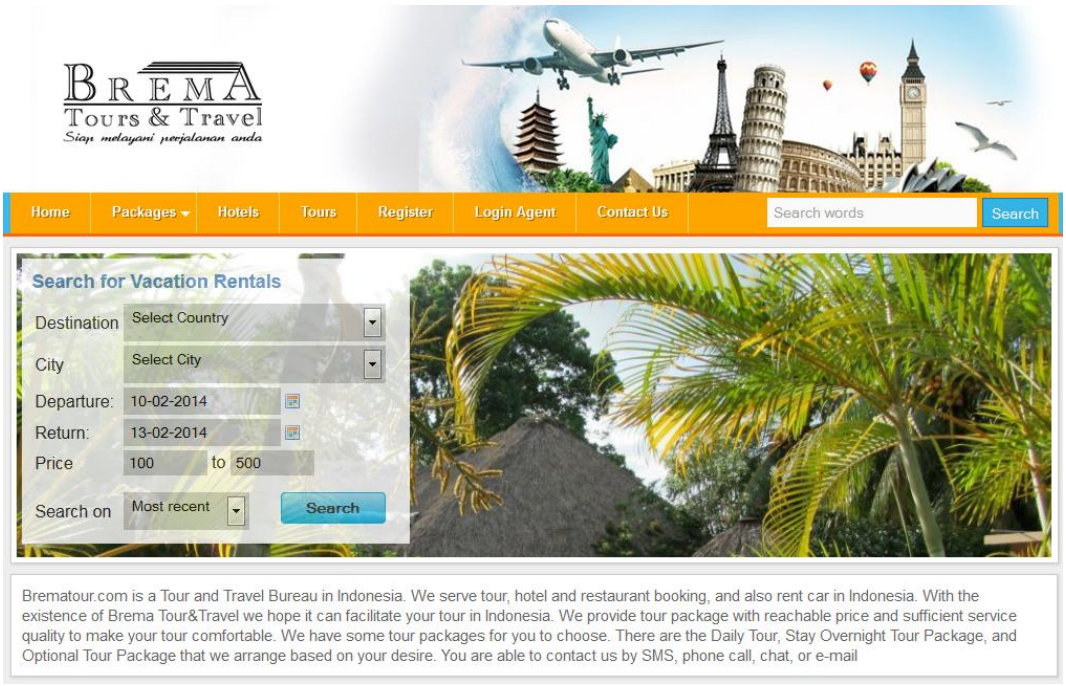
5.1.5 Implementasi Antarmuka

Implementasi antarmuka dilakukan dengan setiap halaman *web* yang dibuat. Implementasi antarmuka yang dibuat untuk semua pengguna, yaitu pengunjung, agen, dan admin.

A. Pengunjung

Dibawah ini merupakan halaman *front end* yang dilihat oleh pengunjung *website*.

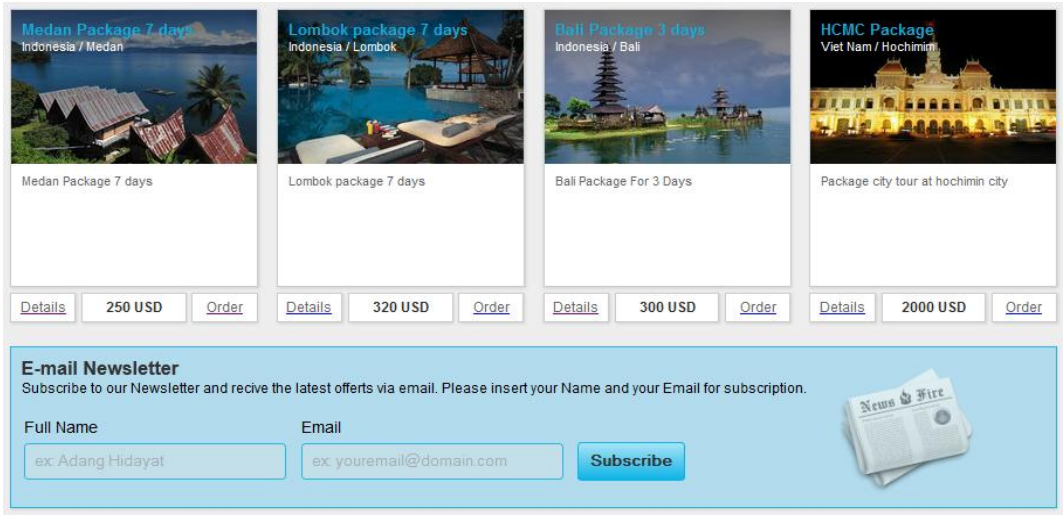
1. Halaman Utama Pengunjung.



Gambar 5.1 Tampilan Halaman Utama

Pada gambar diatas merupakan halaman utama pengunjung dimana awal halaman dibuka saat mengakses *url* brematour.com .

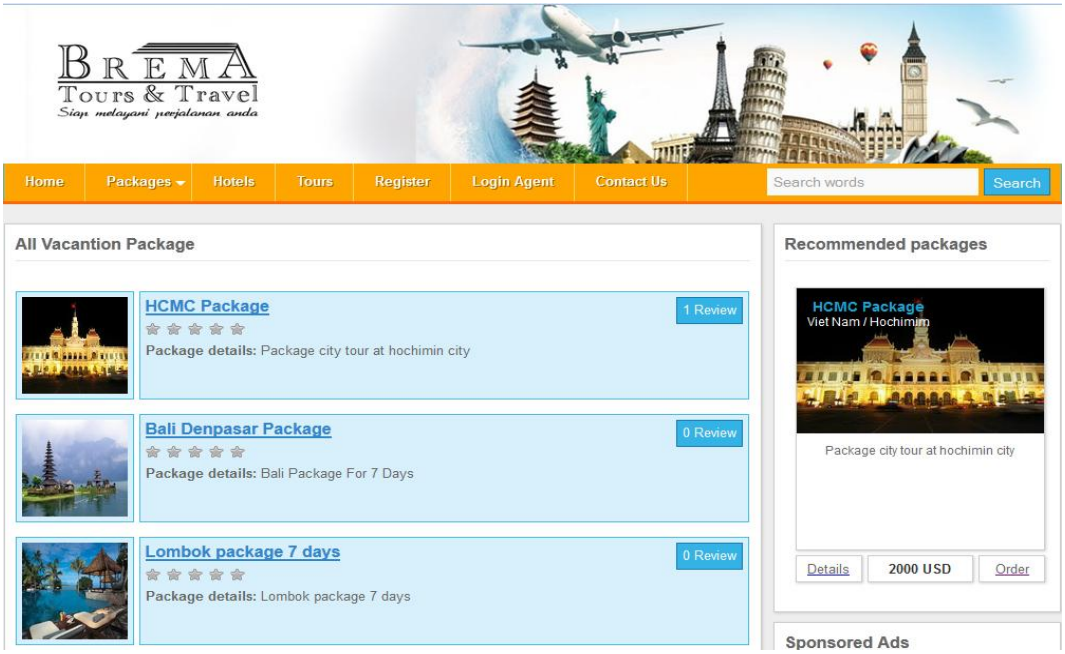
2. Daftar paket wisata untuk pengunjung



Gmabar 5.2 Daftar Paket Wisata

Pada gambar diatas merupakan halaman tampilan daftar paket wisata yang direkomendasikan khusunya bagi para pengunjung.

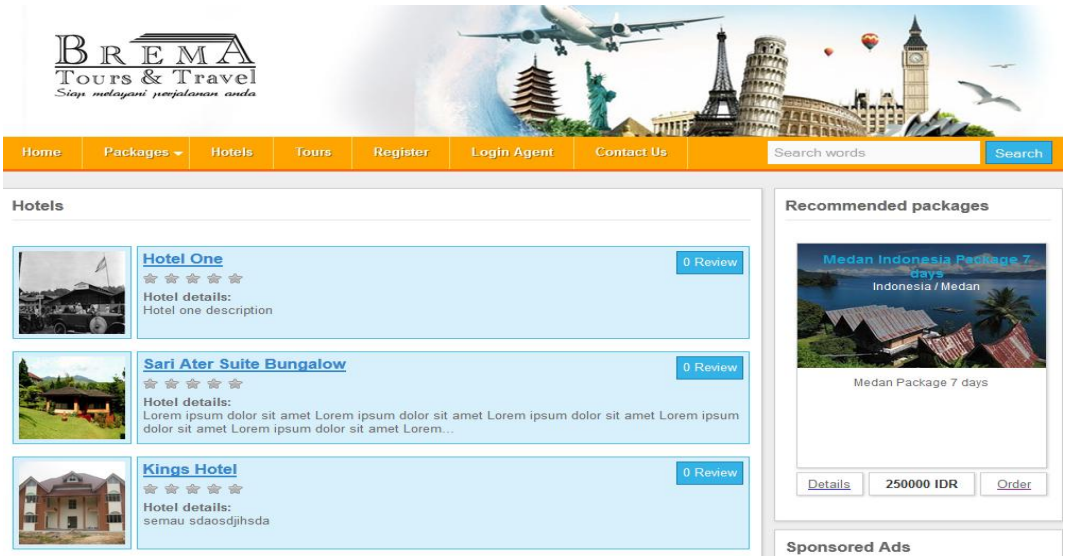
3. Halaman semua paket wisata



Gambar 5.3 Tampilan Semua Paket Wisata

Pada gambar diatas adalh halaman dmna semua *paket Tour&Travel* ada pake *button packages*, beserta detail tentang paket wisata.

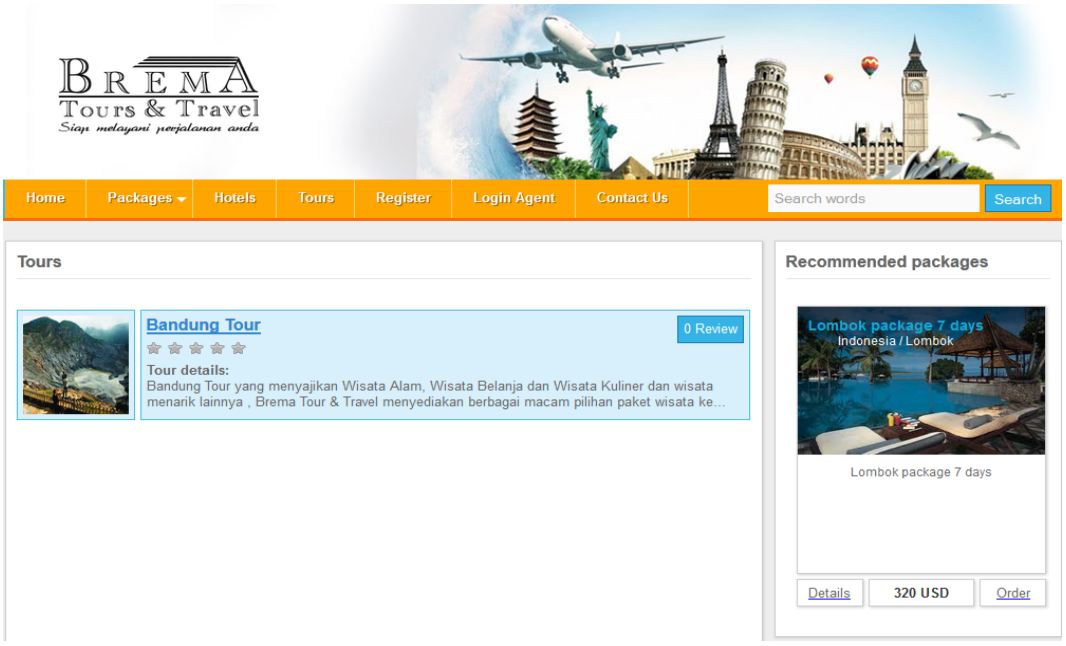
4. Halaman Hotel



Gambar 5.4 Halaman Hotel

Pada gambar diatas *button* Hotel berisi tentang paket hotel yang disediakan oleh agen.

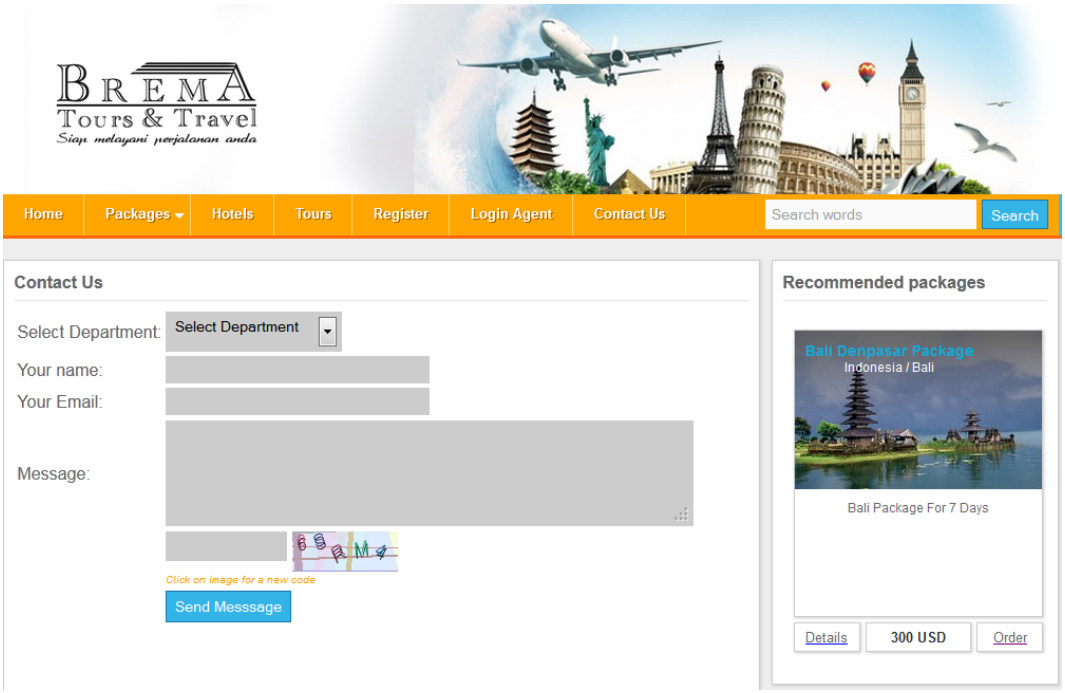
5. Halaman Tour



Gambar 5.5 Halaman Tour

Pada gambaar diatas menunjukan halaman untuk agen yang ingin menjadi member untuk proses bisnis pariwisatanya.

6. Halaman *Contact Us*

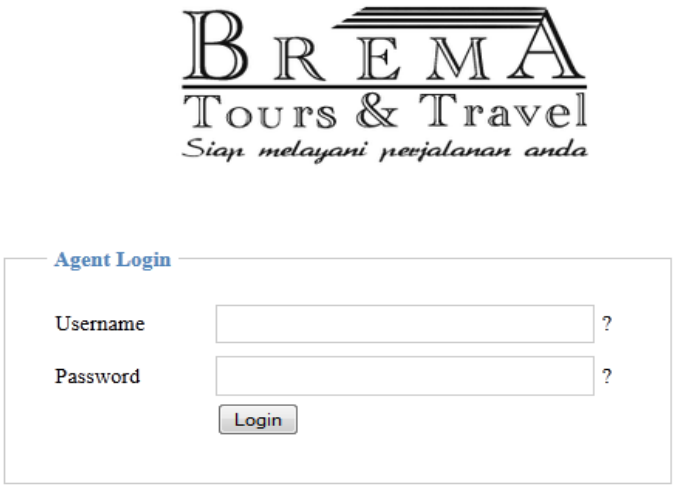


Gambar 5.6 Halaman Contact Us

Pada gambar diatas menunjukan *button Contact Us* yaitu tentang detail perusahaan PT Brema *Tour&Travel*.

B. Agen

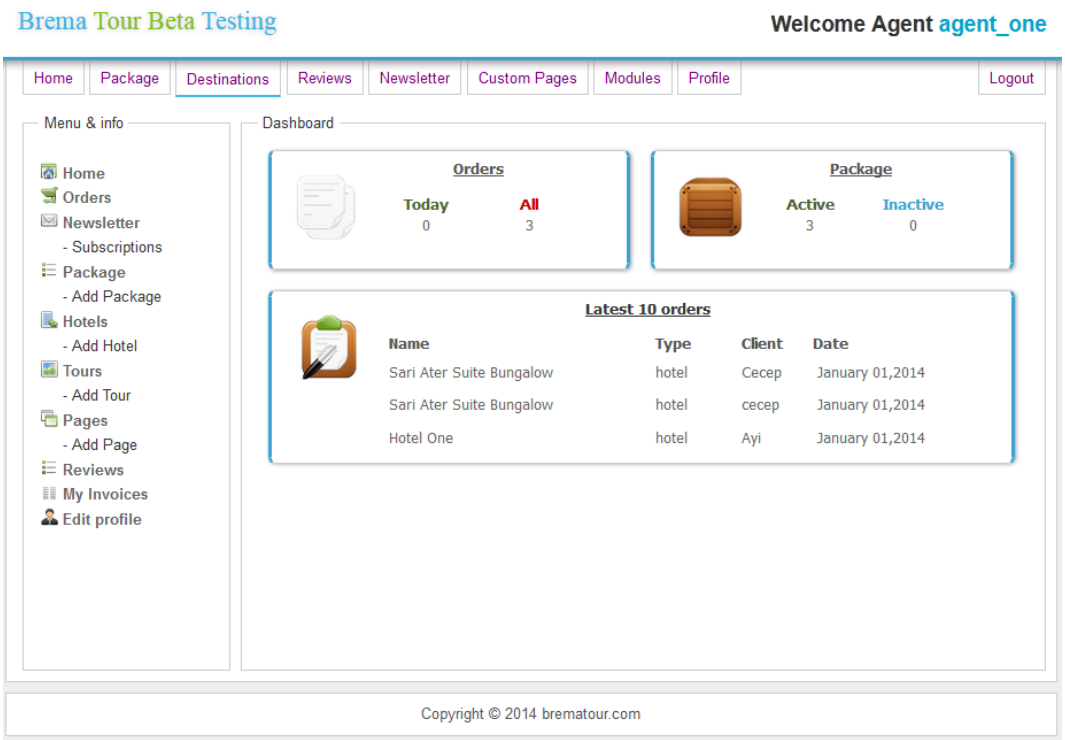
1. Halaman Login Agen



Gambar 5.7 Halaman Login Agen

Gambar diatas merupakan halaman utaman ketika agen melakukan login terlebih dahulu.

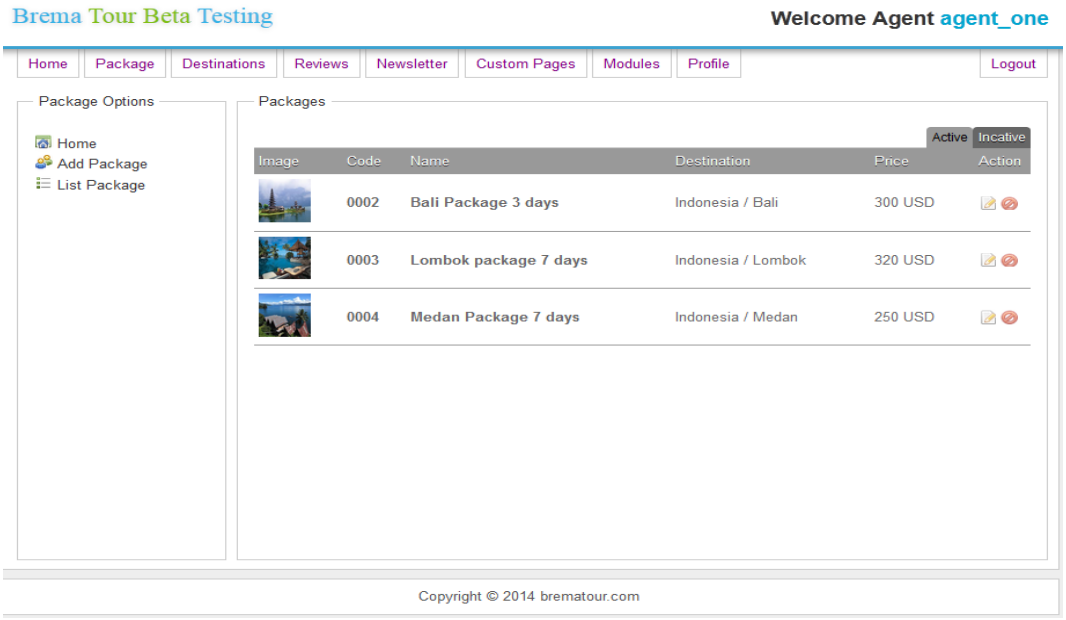
2. Halaman Home agen



Gambar 5.8 Halaman Home Agen

Gambar diatas merupakan halaman *home* ketika agen sudah melewati login pada halaman login.

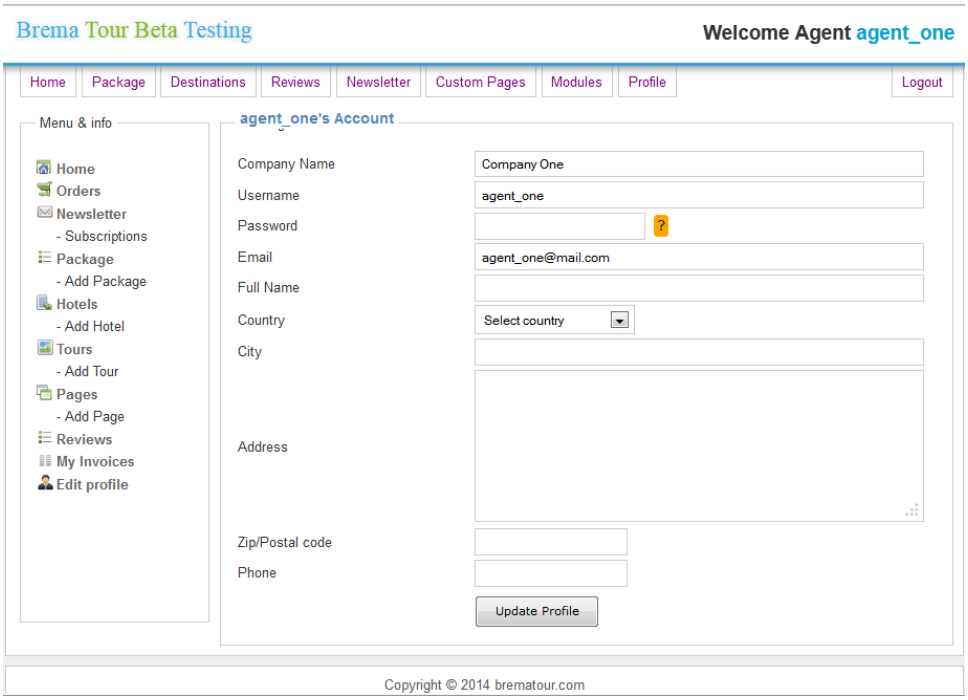
3. Halaman Pesanan Paket



Gambar 5.9 Halaman Pesanan Paket

Pada halaman diatas menunjukan daftar *list* pemesanan paket yang dilakukan oleh pengunjung.

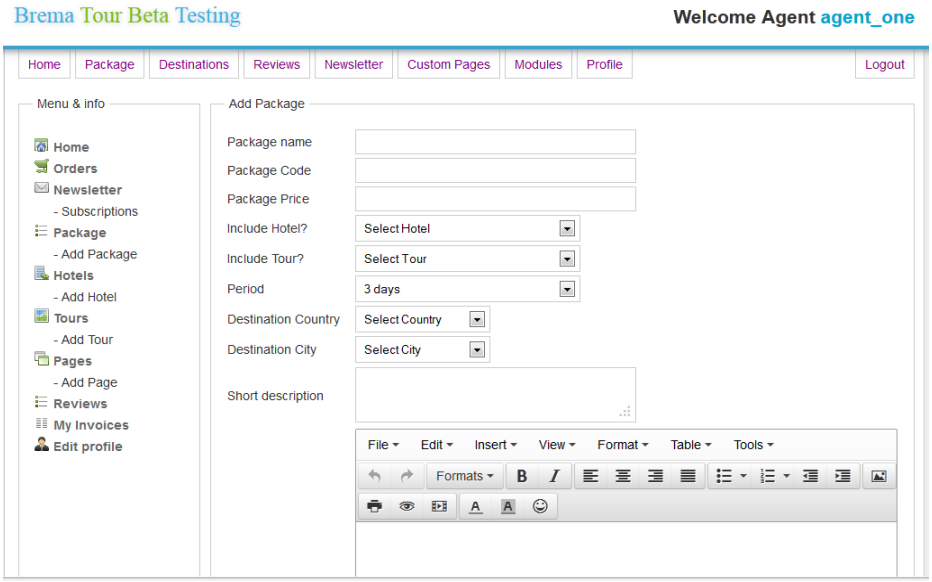
4. Halaman Profil agen



Gambar 5.10 Halaman Profil Agen

Gambar diatas merupakan halaman profil agen ketika sudah menjadi member.

5. Halaman Tambah wisata



Gambar 5.11 Halaman Tambah Wisata

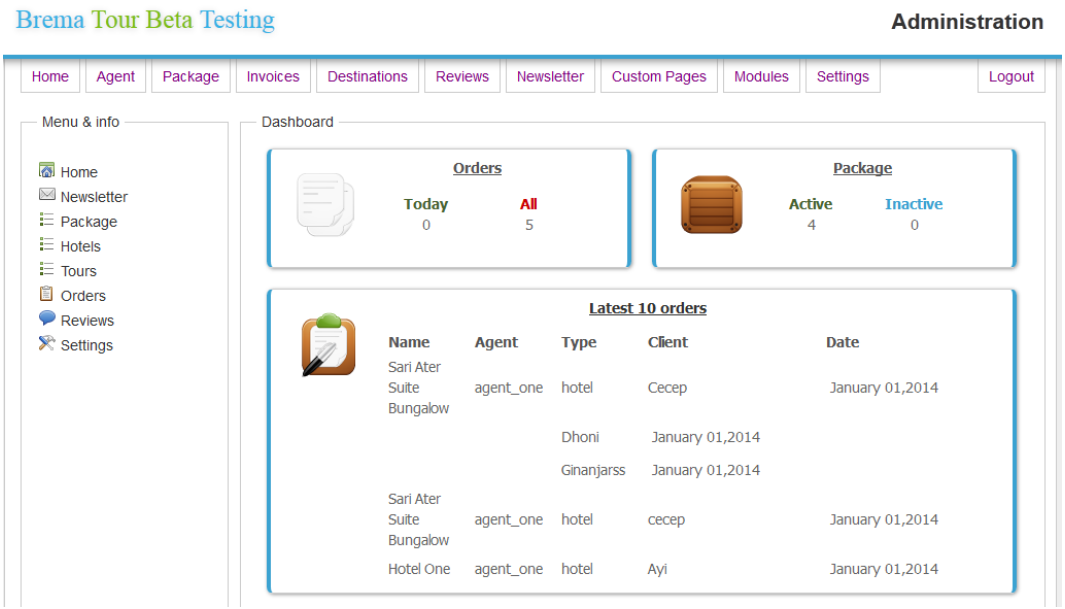
Pada gambar datas merupakan halaman pemanbahan wisata yang dilakukan oleh agen.

c. Admin

1. Halaman Home Admin



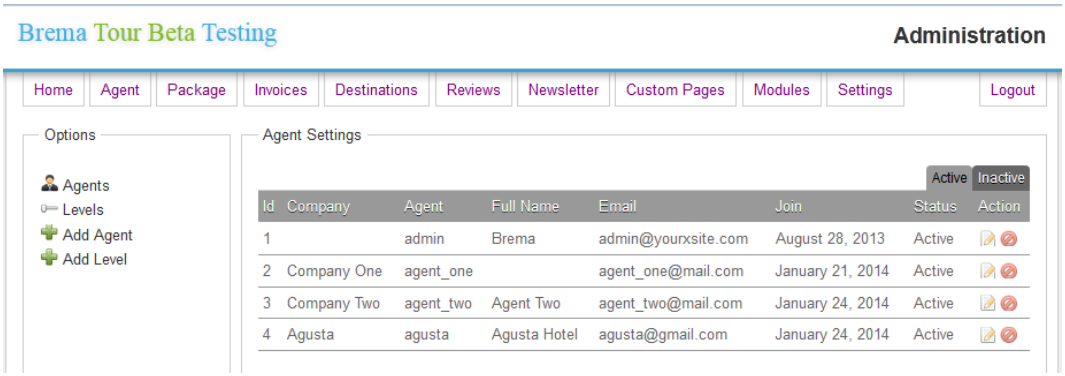
Gambar 5.12 Home Admin



Gambar 5.13 Halaman Home Admin

Pada gambar diatas merupakan tampilan halaman *home* admin ketika sudah melakukan login.

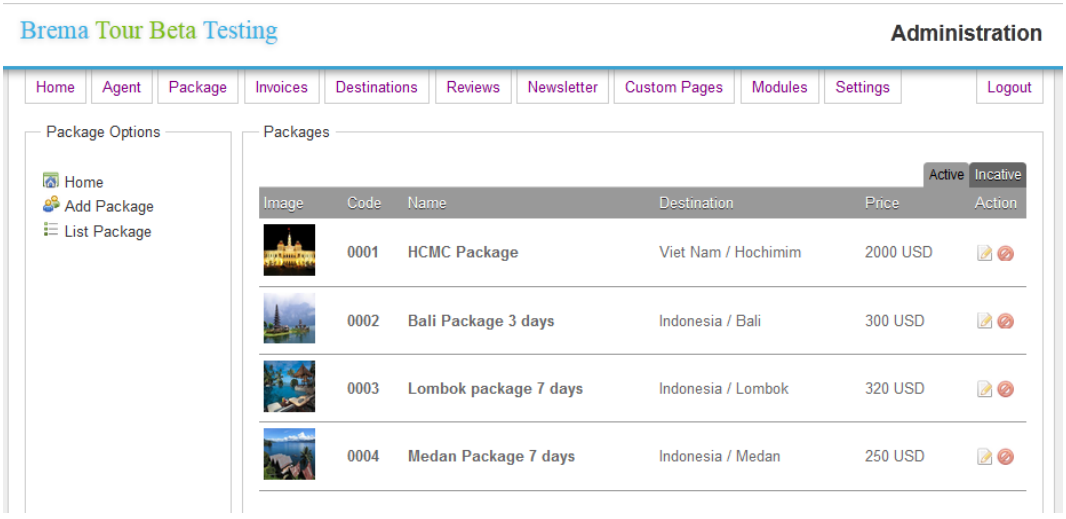
2. Halaman list agen



Gambar 5.14 Halaman List Agen

Gambar diatas merupakan tampilan halaman setting agen yang telah menginputkan iklannya.

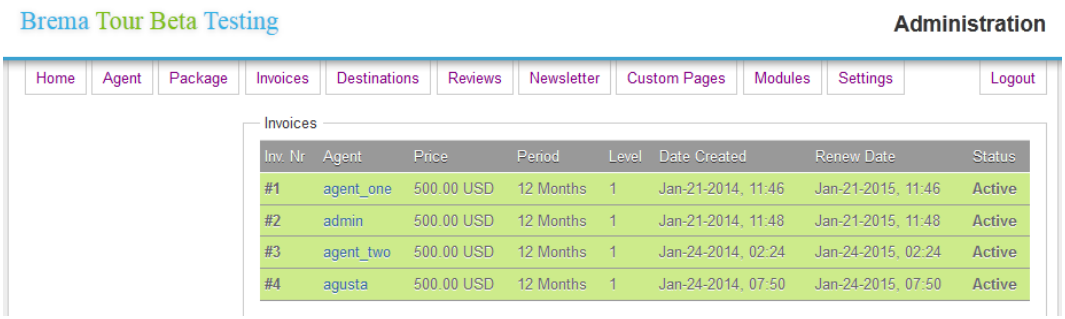
3. Halaman Pesanan Paket



Gambar 5.15 Halaman Pesanan Paket

Gambar diatas merupakan tampilan halaman paket wisata yang diinput oleh semua agen.

4. Halaman Invoice



Gambar 5.16 Halaman Invoice

Gambar diatas merupakan tampilan halaman *invoice* admin ketika agen ingin mengiklankan paket wisatanya.

5.2 Pengujian Sistem

Pengujian merupakan bagian yang penting dalam siklus pengembangan 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 memadai dan sesuai dengan kebutuhan.Pengujian terhadap program itu sendiri bertujuan agar program dapat berjalan dengan baik tanpa mengalami gangguan atau *error*, dan memungkinkan untuk dilakukannya pengembangan sistem lebih lanjut.

Pengujian perangkat lunak ini menggunakan metode pengujian *black box*. Pengujian *black box* ini tidak perlu tahu apa yang sesungguhnya terjadi dalam

sistem atau perangkat lunak, yang diuji adalah masukan serta keluarannya. Berikut ini adalah rencana pengujian input dan output program Sistem Informasi reservasi paket *tour&travel*.

5.2.1 Rencana Pengujian

Rencana pengujian yang akan dilakukan dengan menguji sistem. Pengujian program sistem informasi penilaian berikut menggunakan data uji berupa masukan dari *user* atau pengguna. Rencana pengujian dapat dilihat pada tabel dibawah ini :

1. Pengujian Aplikasi Agen

Tabel 5.1 Rencana Pengujian Agen

Item Uji	Detail Pengujian	Jenis Pengujian
Pengujian input data Register Agen	Menampilkan halaman pengisian member baru, Isi Form pendaftaran Simpan data member baru	*Black box*
Login	Isi data login Verifikasi *Email* Verifikasi *Password.*	*Black box*
Home Agent	Menampilkan List Order	*Black box*
Profile	Menampilkan form profil Ubah profil data member Ganti *password member*	*Black box*
Destinations	Menambah lokasi data paket wisata	*Black box*
Update Paket Tour&Travel	Proses tambah, *update,delete* data paket wisata.	*Black box*

2. Pengujian Aplikasi Admin

Tabel 5.2 Rencana Pengujian Admin

Item Uji	Detail Pengujian	Jenis Pengujian
Login Admin	Isi Login Verifikasi *id* Masukan *Password* salah	*Black box*
Menu & info	*Home* *Newsletter* *Add Paket Tour&Travel* *List Orders* *Reviews* *Settings*	*Black box*

5.2.2 Kasus dan Hasil Pengujian

Berdasarkan rencana pengujian yang telah disusun, maka dapat dilakukan pengujian sebagai berikut.

A. Login Pengujian Registrasi Agent

Tabel 5.3 Pengujian Login

Kasus dan Hasil Uji (Data normal)			
Data Masuk	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Pilih Menu Register	Tampil form registrasi agen	Dapat menampilakn *form* agen	[√] Diterima [ ] Ditolak
Maukan data agen yang telah disediakan Contoh : Nama agen Jenis paket *email*	Jika semua *field* terisi dengan benar, maka pengisian, maka akan muncul pesan kesalahan bahwa field tersebut harus terisi.	*Form* registrasi terisi atau benar dalam dan memberikan pesan “Data tidak *valid*, silahkan coba lagi!”.	[√] Diterima [ ] Ditolak

Kasus dan Hasil Uji (Data salah)			
Data Masuk	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Masukan data kosong	Tampilkan pesan kesalahan dalam pengisian, maka akan muncul pesan kesalahan bahwa *field* tersebut harus terisi.	*User* tidak dapat melakukan *login* dan memberikan pesan "Data tidak *valid*, silahkan coba lagi!". Sesuai yang diharapkan.	[√] Diterima [ ] Ditolak
Data Lengkap harus diisi dan tidak boleh ada field kosong	Tampilkan pesan kesalahan"Data tidak boleh kosong"	Dapat menampilkan Pesan kesalahan "Data tidak boleh kosong"	[√] Diterima [ ] Ditolak

B. Login Admin

Tabel 5.4 Pengujian Login Admin

Kasus dan Hasil Uji (Data normal)			
Data Masuk	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
User name : *user* *Password* : *user*	Dapat *Login* dan masukke halaman utama sesuai hak akses	Dapat *login* dan masuk ke halaman utama sesuai hak aksis	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data salah)			
Data Masuk	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
User name : bebas *Password* : bebas	Tidak dapat *Login* dan kembali ke *home login*	usertidak dapat *login* jika salah *Username* dan atau *Password*	[√] Diterima [ ] Ditolak

C. Pengujian Data Profil Agen

Tabel 5.5 Pengujian Data Profil

Kasus dan Hasil Uji (Data normal)			
Data Masuk	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Klik Menu *Profile* sesuai dengan *field* yang tersedia secara lengkap	Menampilkan *profile member*, *update* data *profile*, dan ganti *password member* Dapat mengisi *field* data yang akan diubah berhasil	Dapat menampilkan *profile member*, *update* data *profile*, dan ganti *password member* Dapat mengisi *field* data yang akan diubah berhasil sesuai yang diharapkan	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data salah/Tidak Lengkap)			
Data Masuk	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Mengosongkan data	Tampilkan pesan kesalahan dalam pengisian, maka akan muncul pesan kesalahan bahwa *field* tersebut harus terisi.	Tampilkan pesan kesalahan dalam pengisian, maka akan muncul pesan kesalahan bahwa *field* tersebut harus terisi.	[√] Diterima [ ] Ditolak

D. Pengujian Pengelolaan Data Paket Wisata

Tabel 5.6 Pengelolaan Data Paket Wisata

Kasus dan Hasil Uji (Data normal)			
Data Masuk	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Klik *add* data paket	Dapat menampilakan *form* data paket wisata	Pengisian data harus sesuai jenis paketnya	[√] Diterima [ ] Ditolak
Isi Data Paket wisata secara *detail*	Data paket harus jelas secara *detail*	Dapat menampilakan *form* paket *tour*	[√] Diterima [ ] Ditolak
Klik *button Add Package*	Penampilkan data paket yang di tambah	Dapat menampilkan data peket yang ditambah	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data salah/Tidak Lengkap)			
Data Masuk	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Form tidak diisi	Muncul pesan kesalahan *field* tersebut harus diisi	Dapat menampilkan pesan kesalahan ”*field* tersebut harus diisi”	[√] Diterima [ ] Ditolak

5.2.3 Kesimpulan Hasil Pengujian

Berdasarkan hasil pengujian dengan kasus *sample* uji yang telah dilakukan memberikan kesimpulan bahwa perangkat lunak ini dapat digunakan dengan baik, namun pengujian tersebut dapat dikatakan belum sempurna, karena hanya dilakukan pada satu sisi pengujian. Dari semua yang telah dilakukan dalam pengujian ini diharapkan mampu mewakili pengujian fungsi yang lain dalam aplikasi Sistem Informasi reservasi ini.

Proses masih memungkinkan untuk terjadinya kesalahan pada sintaks karena penyaringan proses dalam bentuk arahan tampilan paket belum maksimal diciptakan dan ditampilkan, secara fungsional, sistem sudah dapat menghasilkan output yang diharapkan.