BAB IV

IMPLEMENTASI DAN PENGUJIAN SISTEM

Pada bab ini akan dijelaskan secara rinci penerjemahan kebutuhan pembangunan aplikasi ke dalam representasi perangkat lunak sesuai dengan hasil analisis yang telah dilakukan. Setelah implementasi maka dilakukan pengujian sistem yang baru dimana akan dilihat kekurangan-kekurangan pada aplikasi yang baru untuk selanjutnya diadakan pengembangan sistem.

4.1 Implementasi

Tujuan implementasi sistem adalah untuk menjelaskan tentang manual modul kepada semua *user* yang akan menggunakan sistem. Sehingga user tersebut dapat merespon apa yang ditampilkan di sistem dan memberikan masukan kepada pembuat sistem untuk dilakukan perbaikan agar sistem lebih baik lagi.

4.1.1 Perangkat Lunak Pembangun

Perangkat lunak yang digunakan pada sistem komputer yang digunakan untuk membangun Aplikasi Pemandu Turis adalah sebagai berikut:

1. Sistem Operasi *Windows 7 Enterprise*.
2. *WampServer* untuk *local* internet dan penyimpanan *database*.
3. *Macromedia Dreamweaver* 8 untuk penulisan kode program PHP.
4. *Eclipse Indigo* untuk penulisan kode program android.
5. *Google Map API Android 2.0.*
6. *www.hostinggokil.com* sebagai web hosting aplikasi *backend*.

7. *Mozilla Firefox* atau *Google Chrome* sebagai *browser*.

4.1.2 Perangkat Keras Pembangun

Kebutuhan minimum perangkat keras (*hardware*) yang diperlukan untuk mengimplementasikan program aplikasi yang dibuat adalah perangkat keras komputer PC dan *handphone kompatibel* dengan spesifikasi yang disebutkan dibawah ini. Semakin tinggi spesifikasi komputer yang digunakan untuk menjalankan aplikasi, akan semakin baik. Kebutuhan minimumnya yaitu :

1. Processor : Dengan Kecepatan Minimum 2.0 GHZ
2. Memory / RAM : Minimum 512MB
3. Hardisk : Minimum Kapasitas 20 GB
4. VGA : Dengan Kecepatan Minimum 32 MB
5. *Modem*
6. Handphone yang digunakan berbasis Android dengan spefikasi OS minimal Android 2.1(Eclair)
7. Handphone yang digunakan minimum processor 600 MHZ.

4.1.3 Kebutuhan *Web Hosting*

Kebutuhan web hosting pada pada penelitian ini digunakan untuk keperluan aplikasi *backend* yaitu web administrator dan web service, oleh karena itu agar sistem dapat berjalan maka perlu di upload di *web hosting*. Adapun spesifikasi *web hosting* dan *nama domain* (situs) yang dipakai adalah :

Nama Domain / Situs : *http://bandunginterest.com*

Tabel 4.1 Spesifikasi *web hosting*

Disk space	500 MB
Bandwidth	Unlimited
Depelovment Feature	1. *PHP version 5.0.X with Zend Optimizer* 2. *PERL version 5.8.3* 3. *MySQL 5.x* 4. *PostgreSQL (IIX Only)* 5. *Private CGI-BIN* 6. *CGI Script Library* 7. *Server Side Includes (SSI)* 8. *WAP/WML/Wireless* 9. *Java* 10. *GD* 11. *cPanel* 12. *phpMyAdmin* 13. *SMTP/IPETA/Webmail* 14. *Ruby On Rails* 15. Instant Backups

4.1.4 Implementasi Basis Data (*Database*)

Implementasi basis data diambil berdasarkan perancangan basis data yang dibuat sebelumnya, Secara fisik, implementasi basis data diimplementasikan menggunakan perangkat lunak MySQL Server 5.6, Tabel-tabel berikut menggambarkan struktur tabel yang diimplementasikan pada basis data.

1. Tabel login

```
CREATE TABLE IF NOT EXISTS `login` (
  `username` varchar(20) NOT NULL,
  `userpass` varchar(30) NOT NULL,
  `nama` varchar(50) NOT NULL,
```

```
  PRIMARY KEY (`username `)
) ENGINE= InnoDB DEFAULT CHARSET=latin1;
```

2. Tabel category

```
DROP TABLE IF EXISTS `category `;
CREATE TABLE ` category ` (
  `id_cat` int(11) NOT NULL AUTO_INCREMENT,
  `nama_cat ` varchar(20) NOT NULL,
  `icon_category ` varchar(25) DEFAULT NULL,
  PRIMARY KEY (`id_cat`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

3. Tabel poi

```
DROP TABLE IF EXISTS `poi`;
CREATE TABLE `poi ` (
  `id_poi` int(11) NOT NULL AUTO_INCREMENT,
  `nama_poi` varchar(50) NOT NULL,
  `telpon` varchar(20) DEFAULT NULL,
  `deskripsi_s` text NOT NULL,
  `deksiprsi_l` text NOT NULL,
  `icon` varchar(20) NOT NULL,
  `id_cat` int(11) NOT NULL,
  `id_location` int(11) NOT NULL,
```

```
  `id_trans` int(11) NOT NULL,
  PRIMARY KEY (`id_poi`),
  KEY `FK_category` (`id_cat`),
  KEY `FK_location` (`id_location`),
  KEY `FK_location` (`id_trans`),
  CONSTRAINT `FK_category` FOREIGN KEY (`id_cat`)
REFERENCES `category` (`id_cat`) ON DELETE NO
ACTION ON UPDATE NO ACTION,
  CONSTRAINT `FK_location` FOREIGN KEY
(`id_location `) REFERENCES `location` (`id_
location`) ON DELETE NO ACTION ON UPDATE NO ACTION,
  CONSTRAINT `FK_trans` FOREIGN KEY (`id_trans `)
REFERENCES `trans` (`id_ trans`) ON DELETE NO
ACTION ON UPDATE NO ACTION
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

4. Tabel *location*

```
DROP TABLE IF EXISTS `location`;
CREATE TABLE `location` (
  `id_location` int(11) NOT NULL,
  `alamat` varchar(50) NOT NULL,
  `latitude` double DEFAULT NULL,
  `longitude` double DEFAULT NULL,
```

```
  PRIMARY KEY (`id_location`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

5. Tabel trans

```
DROP TABLE IF EXISTS `trans`;
CREATE TABLE `trans ` (
  `id_trans` int(11) NOT NULL AUTO_INCREMENT,
  `nama_angkot` varchar(50) NOT NULL,
  `warna` varchar(30) DEFAULT NULL,
  `rute` text NOT NULL,
  `gambar_a` varchar(20) NOT NULL,
  `id_cat` int(11) NOT NULL,
  PRIMARY KEY (`id_trans`),
  KEY `FK_category` (`id_cat`),
  CONSTRAINT `FK_category` FOREIGN KEY (`id_cat`)
REFERENCES `category` (`id_cat`) ON DELETE NO
ACTION ON UPDATE NO ACTION
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

4.1.5 Implementasi *Class*

Implementasi *class* pada Aplikasi pemandu turis merupakan implementasi dari perancangan *class* diagram yang terbentuk dari *use case* diagram kedalam bentuk fisik file-file dengan ekstensi *.java* maupun ekstensi *.php* yang digunakan

dalam pembuatan dan jalannya proses dalam sistem. Berikut beberapa implementasi *class* yang telah terdefinisi, sedangkan untuk *sourcodenya* dilampirkan.

Tabel 4.2 Implementasi *class*

No	Nama Kelas	Nama File Fisik
1.	Main	Main.java
2.	POIMarker	POIMarker.java
3.	Marker	Marker.java
4.	MixView	MixView.java
5.	PhsycalPlace	PhsycalPlace.java
6.	DownloadManager	DownloadManager.java
7.	JSON	JSON. java
8.	Matrix	Matrix.java
9.	DataHandler	DataHandler.java
10.	Trans	Trans.java
11.	DeskripsiTrans	DeskripsiTrans.java
12.	PaintScreen	PaintScreen.java
13.	Camera	Camera.java
14.	MyGmap	MyGmap.java
15.	GoogleMapsDirection	GoogleMapsDirection.java
16.	DirectionOverlay	DirectionOverlay.java
17.	ScreenLine	ScreenLine.javaa
18.	TextObj	TextObj.java

19.	MixUtils	MixUtils.java
20.	RadarPoints	RadarPoints.java

4.1.6 Implementasi Antarmuka

Implementasi antarmuka dilakukan dengan setiap halaman yang dibuat baik aplikasi *frontend* yang dibangun di perangkat *mobile* maupun aplikasi *backend* yang dibangun menggunakan web. Berikut ini beberapa tampilan antar muka yang telah di implementasikan.

4.1.6.1 Antarmuka Aplikasi *Frontend*

Berikut ini implementasi antarmuka dari aplikasi *frontend* Bandung *tour guide* beserta *file* xml sebagai *layout* dari masing-masing antarmuka yang terlihat pada Tabel 4.3. Adapun tampilan implementasi antarmuka aplikasi *mobile* Bandung *tour guide* tertera pada halaman Lampiran A Tampilan Antarmuka.

Tabel 4.3 Implementasi antarmuka aplikasi *frontend*

No	Halaman	Deskripsi	Nama File
1.	Menu utama	Digunakan oleh *user* untuk memilih kategori	main.xml
2.	POI *augmented reality*	Digunakan oleh *user* untuk masuk ke halaman kategori POI dengan *augmented reality*	mainar.xml
3.	*Detail marker*	Digunakan oleh *user* untuk masuk ke halaman *detail marker* ketika salah satu *marker augmented reality* dipilih	marker_detail.xml
4.	Navigasi map lokasi	Digunakan oleh *user* untuk masuk ke halaman google *map direction*	map.xml
5.	Transportasi	Digunakan oleh *user* untuk masuk ke halaman kategori	list.xml

		transportasi	
6.	*Detail* transportasi	Digunakan oleh *user* untuk masuk ke halaman *detail* transportasi	deskripsi.xml

4.1.6.2 Antarmuka Aplikasi *Backend*

Berikut ini implementasi antarmuka dari aplikasi *backend* Bandung *tour guide* beserta *file* program sebagai tampilan dari masing-masing antarmuka yang terlihat pada Tabel 4.4. Adapun tampilan implementasi antarmuka aplikasi *backend* Bandung *tour guide* tertera bersama antarmuka aplikasi *frontend* pada halaman Lampiran A Tampilan Antarmuka.

Tabel 4.4 Implementasi antarmuka aplikasi *backend*

No	Halaman	Deskripsi	Nama File
1.	Login	Digunakan oleh *admin* sebagai halaman login administrator	Index.php
2.	Menu Utama	Digunakan oleh *admin* untuk masuk ke halaman menu utama	Home.php
3.	*Olah Kategori*	Digunakan oleh *admin* untuk melihat, menambah dan merubah data kategori	FormCategory.php
4.	Olah POI	Digunakan oleh *admin* untuk melihat, menambah dan merubah data kategori POI(restoran, penginapan, perbelanjaan dan wisata)	FormPOI.php
5.	Olah transportasi	Digunakan oleh *admin* untuk melihat, menambah dan merubah data kategori transportasi	FormTrans.php

4.2 Pengujian

Dalam penelitian ini pengujian yang dilakukan terhadap sistem yaitu pengujian secara fungsional *(alpha)* dan betha. Metode yang digunakan dalam pengujian ini adalah pengujian *blackbox* yang berfokus pada persyaratan fungsional dari sistem yang dibangun.

4.2.1 Rencana Pengujian

Rencana pengujian adalah pengujian terhadap fungsi-fungsi yang ada di dalam sistem, apakah fungsional dari aplikasi tersebut berfungsi sesuai yang diharapkan atau tidak. Berikut ini tabel rencana pengujian dari sistem yang dibangun :

Tabel 4.5 Rencana pengujian

Pengujian **Aplikasi *Frontend***		
Item Uji	**Detail Uji**	**Jenis Uji**
Kategori	Menu Gambar Kategori	*Black box*
Marker AR	Tampil *Marker* AR	*Black box*
	Tampil *Radar Point*	*Black box*
Radius Lokasi	Tampil Radius Lokasi	*Black box*
Detail Marker AR	Tampil *Detail Marker* AR	*Black box*
Button Navigasi Lokasi pada *Detail Marker* AR	Tampil Map	*Black box*
	Tampil Navigasi Lokasi	*Black box*
Button Telepon pada *Detail Marker* AR	Tampil Nomor Lokasi dalam mode *Phone View*	*Black box*
Button Keluar pada *Detail Marker* AR	Tampil *Marker* AR	*Black box*
Transportasi	Tampil List Jurusan Transportasi	*Black box*

Search pada *List* Transportasi	Tampil Hasil *Search*	*Black box*
List Jurusan Transportasi	Tampil *Detail* Jurusan Transportasi	*Black box*
Pengujian Aplikasi *Backend*		
Item Uji	**Detail Uji**	**Jenis Uji**
Login Admin	Username	*Black box*
	Password	*Black box*
Tambah Kategori	Nama Kategori	*Black box*
	Gambar Kategori	*Black box*
Ubah Kategori	Nama Kategori	*Black box*
	Gambar Kategori	*Black box*
Hapus Kategori	Hapus	*Black box*
Tambah POI	Category	*Black box*
	Nama POI	*Black box*
	Telepon	*Black box*
	Deskripsi Singkat	*Black box*
	Deskripsi Lengkap	*Black box*
	Icon Lokasi	*Black box*
	Gambar pendukung	*Black box*
	Alamat	*Black box*
	Latitude	*Black box*
	Longitude	*Black box*
Ubah POI	Nama POI	*Black box*
	Telepon	*Black box*
	Deskripsi Singkat	*Black box*
	Deskripsi Lengkap	*Black box*
	Icon Lokasi	*Black box*
	Gambar pendukung	*Black box*
	Alamat	*Black box*

	Latitude	*Black box*
	Longitude	*Black box*
Hapus POI	Nama POI	*Black box*
	Telepon	*Black box*
	Deskripsi Singkat	*Black box*
	Deskripsi Lengkap	*Black box*
	Icon Lokasi	*Black box*
	Gambar pendukung	*Black box*
	Alamat	*Black box*
	Latitude	*Black box*
	Longitude	*Black box*
Tambah Transportasi	Category	*Black box*
	Jurusan Angkot	*Black box*
	Warna Angkot	*Black box*
	Rute Angkot	*Black box*
	Gambar Angkot	*Black box*
Ubah Transportasi	Jurusan Angkot	*Black box*
	Warna Angkot	*Black box*
	Rute Angkot	*Black box*
	Gambar Angkot	*Black box*
Hapus Transportasi	Jurusan Angkot	*Black box*
	Warna Angkot	*Black box*
	Rute Angkot	*Black box*
	Gambar Angkot	*Black box*

4.2.2 Kasus dan Hasil Pengujian Apha

Berdasarkan rencana pengujian, maka dapat dilakukan pengujian *Alpha* pada Aplikasi Pemandu Turis sebagai berikut:

4.2.2.1 Pengujian Pilih Kategori

Pengujian pilih kategori merupakan pengujian fungsionalitas dengan cara memilih menu gambar kategori.

Tabel 4.6 Pengujian kategori

Kasus dan hasil uji (data normal)			
Data masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Kategori : "*Restoran*"	Menampilkan data sesuai kategori yang dipilih	Data kategori Restoran ditampilkan	[√] diterima [ ] ditolak

4.2.2.2 Pengujian *Marker* AR

Pengujian pilih *marker* AR merupakan pengujian fungsionalitas ketika salah kategori POI dipilih.

Tabel 4.7 Pengujian *marker* AR

Kasus dan hasil uji (data normal)			
Data masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Kategori : POI(restoran, wisata, penginapan dan perbelanjaan)	Menampilkan mode tampilan *augmented reality* beserta visualisasi *marker* AR	*Marker augmented reality* beserta informasinya berhasil ditampilkan	[√] diterima [ ] ditolak
	Menampilkan mode tampilan *augmented reality* beserta visualisasi radar di pojok kiri atas layar	Tampilan radar pada mode AR beserta titik lokasi berhasil ditampilkan	

4.2.2.3 Pengujian Radius Lokasi

Pengujian pilih radius lokasi merupakan pengujian fungsionalitas dengan cara menekan tombol menu pada *handset* ketika berada dalam mode *augmented reality*.

Tabel 4.8 Pengujian radius lokasi

Kasus dan hasil uji (data normal)			
Data masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Radius Lokasi : 10KM	Menampilkan hasil *marker* POI sesuai pengaturan radius lokasi	Radius Lokasi berhasil mengatur jauh dekatnya lokasi pengguna dengan lokasi tujuan	[√] diterima [ ] ditolak

4.2.2.4 Pengujian *Detail Marker* AR

Pengujian pilih detail *marker* AR merupakan pengujian fungsionalitas dengan cara memilih salah satu *marker* AR ketika berada dalam mode *augmented reality*.

Tabel 4.9 Pengujian *detail marker* AR

Kasus dan hasil uji (data normal)			
Data masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
POI : "*Roemah Keboen*"	Menampilkan detail marker AR POI beserta informasinya	Detail marker AR POI beserta informasinya berhasil ditampilkan	[√] diterima [ ] ditolak

4.2.2.5 Pengujian *Button* Navigasi Lokasi pada *Detail Marker* AR

Pengujian pilih *button* navigasi lokasi pada *detail marker* AR merupakan pengujian fungsionalitas dengan cara memilih *button* navigasi lokasi.

Tabel 4.10 Pengujian *button* navigasi lokasi pada *detail marker* AR

Kasus dan hasil uji (data normal)			
Data masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Kategori *: "Restoran"* *Button* Navigasi Lokasi	Menampilkan peta	Peta berhasil ditampilkan	[√] diterima [ ] ditolak
	Menampilkan navigasi lokasi dengan pengguna	Map navigasi lokasi dengan pengguna berhasil ditampilkan	

4.2.2.6 Pengujian *Button* Telepon pada *Detail Marker* AR

Pengujian pilih *button* telepon pada *detail marker* AR merupakan pengujian fungsionalitas dengan cara memilih *button* telepon.

Tabel 4.11 Pengujian *button* telepon pada *detail marker* AR

Kasus dan hasil uji (data normal)			
Data masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Kategori *: "Restoran"* *Button* Telepon	Menampilkan nomor telepon lokasi dalam mode *phone view*	Nomor telepon lokasi dalam mode *phone view* berhasil ditampilkan	[√] diterima [ ] ditolak

4.2.2.7 Pengujian Pilih *Button* Keluar pada *Detail Marker* AR

Pengujian pilih *button* keluar pada *detail marker* AR merupakan pengujian fungsionalitas dengan cara memilih *button* keluar.

Tabel 4.12 Pengujian *button* keluar pada *detail marker* AR

Kasus dan hasil uji (data normal)			
Data masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Kategori *:* *"Restoran"* *Button* Kembali	Kembali ke tampilan mode *augmented reality*	Berhasil kembali ke tampilan mode *augmented reality*	[√] diterima [ ] ditolak

4.2.2.8 Pengujian Transportasi

Pengujian pilih transportasi merupakan pengujian fungsionalitas dengan cara memilih kategori transportasi pada menu utama.

Tabel 4.13 Pengujian transportasi

Kasus dan hasil uji (data normal)			
Data masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Kategori : "*Transportasi* "	Menampilkan *list* jurusan transportasi	*List* jurusan transportasi berhasil ditampilkan	[√] diterima [ ] ditolak

4.2.2.9 Pengujian *Search* pada *List* Transportasi

Pengujian pilih *search* pada *list* transportasi merupakan pengujian fungsionalitas dengan cara memasukkan *keyword*(kata kunci) pada kolom *search*.

Tabel 4.14 Pengujian pilih *search* pada *list* transportasi

Kasus dan hasil uji (data normal)			
Data masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Keyword : "Dago"	Menampilkan *list* data transportasi sesuai *keyword*	Data yang dicari berdasarkan *keyword* berhasil ditampilkan	[√] diterima [ ] ditolak

4.2.2.10 Pengujian *List* Jurusan Transportasi

Pengujian pilih *list* jurusan transportasi merupakan pengujian fungsionalitas dengan cara memilih salah satu list jusuran transportasi.

Tabel 4.15 Pengujian pilih *list* jurusan transportasi

Kasus dan hasil uji (data normal)			
Data masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
List jurusan transportasi : "*ABD. MUIS – DAGO*"	Menampilkan *detail* transportasi sesuai dengan jurusan yang dipilih	*Detail* transportasi beserta informasi berhasil ditampilkan	[√] diterima [ ] ditolak

4.2.2.11 Pengujian *Login Admin*

Pengujian login admin terbagi atas dua bagian diantaranya verifikasi *username* dan verifikasi *password*.

Tabel 4.16 Pengujian *login admin*

Kasus dan hasil uji (data normal)			
Data masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Username : "*admin*" Password : "*rahasia*"	Dapat terisi pada *textfield username* dan *textfield password*	Dapat mengisi nama *usename* dan *password* Sesuai yang diharapakan	[√] diterima [ ] ditolak
Kasus dan hasil uji (data salah)			
Data masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Username : "*admin*" Password : "*admin*"	Tidak dapat login dan tampilan pesan "Maaf, Username Dan Password Anda	*User* tidak dapat login	[√] diterima [ ] ditolak

	Salah.”		

4.2.2.12 Pengujian Tambah Kategori

Pengujian tambah kategori merupakan fungsionalitas sistem yang terdiri dari pengisian data-data kategori.

Tabel 4.17 Pengujian tambah kategori

Kasus dan hasil uji (data normal)			
Data masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Nama kategori : “*Restoran* “ Gambar kategori : “*Resto.png*”	Dapat terisi pada semua *field* sesuai data masukan	Dapat mengisi semua *field* dengan data yang diharapkan	[√] diterima [ ] ditolak
Kasus dan hasil uji (data salah)			
Data masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Nama kategori : “*Restoran* “ Gambar kategori : “ ”	Menampilkan pesan di setiap *field* yang tidak terisi bahwa data tersebut harus diisi atau data tidak boleh kosong	Fungsi berjalan sesuai dengan yang diharapkan	[√] diterima [ ] ditolak

4.2.2.13 Pengujian Ubah Kategori

Pengujian ubah kategori merupakan fungsionalitas sistem yang dilakukan dengan cara mengganti data-data kategori baru.

Tabel 4.18 Pengujian ubah kategori

Kasus dan hasil uji (data normal)			
Data masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Nama kategori : “*Wisata*” Gambar kategori :	Data kategori baru dapat terisi pada	Dapat mengisi semua *field*	[√] diterima [ ] ditolak

"*wisata.png*"	semua *field*	dengan data yang diharapkan	

4.2.2.14 Pengujian Hapus Kategori

Pengujian hapus kategori merupakan pengujian fungsionalitas dengan cara memilih kategori yang dipilih.

Tabel 4.19 Pengujian hapus kategori

Kasus dan hasil uji (data normal)			
Data masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Kategori : "*Bank*"	Menampilkan konfirmasi data ingin dihapus dan data terhapus.	Fungsi berjalan sesuai dengan yang diharapkan	[√] diterima [ ] ditolak

4.2.2.15 Pengujian Tambah POI

Pengujian tambah POI merupakan fungsionalitas sistem yang terdiri dari pengisian data-data POI.

Tabel 4.20 Pengujian tambah POI

Kasus dan hasil uji (data normal)			
Data masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Nama : ""*Roemah Keboen*" Alamat : "*Jl. Riau No.210"* Deskripsi singkat : *"Tipe Makanan : Indonesian Food "* Deskripsi lengkap : *"Menu Andalan*	Dapat terisi pada semua atau beberapa *field* sesuai data masukan	Dapat mengisi *field* sesuai yang diharapkan	[√] diterima [ ] ditolak

: Nasi Bimbal” Icon : “*rbk.jpg*” Latitude : -6.89706 Longitude : 100.2988			
Kasus dan hasil uji (data salah)			
Data masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Nama : ““*Roemah Keboen*” Alamat : “ ” Deskripsi singkat : *“Tipe Makanan : Indonesian Food “* Deskripsi lengkap : *“Menu Andalan : Nasi Bimbal”* Icon : “*rbk.jpg*” Latitude : -6.89706 Longitude : 100.2988	Menampilkan pesan di setiap *field* yang tidak terisi bahwa data tersebut harus diisi atau data tidak boleh kosong	Fungsi berjalan sesuai dengan yang diharapkan	[√] diterima [ ] ditolak

4.2.2.16 Pengujian Ubah POI

Pengujian ubah POI merupakan fungsionalitas sistem yang dilakukan dengan cara mengubah data-data POI.

Tabel 4.21 Pengujian ubah POI

Kasus dan hasil uji (data normal)			
Data masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Nama : ““*Roemah Keboen*” Alamat : “*Jl. Dipatiukur No.114”* Deskripsi singkat : *“Tipe Makanan : Indonesian Food “* Deskripsi lengkap : *“Menu Andalan : Nasi Bimbal”* Icon : “*rbk.jpg*” Latitude : -6.89706 Longitude : 100.2988	Data poi baru dapat terisi pada semua atau beberapa *field*	Dapat mengisi *field* sesuai yang diharapkan	[√] diterima [ ] ditolak

4.2.2.17 Pengujian Hapus POI

Pengujian hapus POI terdiri dari data-data POI yang akan di hapus.

Tabel 4.22 Pengujian hapus POI

Kasus dan hasil uji (data normal)			
Data masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
POI : "*Roemah Keboen*"	Menampilkan konfirmasi data ingin dihapus dan data terhapus.	Fungsi berjalan sesuai dengan yang diharapkan	[√] diterima [ ] ditolak

4.2.2.18 Pengujian Tambah Transportasi

Pengujian tambah transportasi merupakan fungsionalitas sistem yang terdiri dari pengisian data-data transrportasi.

Tabel 4.23 Pengujian tambah transportasi

Kasus dan hasil uji (data normal)			
Data masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Nama Angkot : "*Dago - St.Hall*" Warna : *"Putih – Hijau"* Rute Angkot : *"Riung – Binong – Kiara Condong – Siliwangi – Dipatiukur – Dago"* Gambar : "*wp11.jpg*"	Dapat terisi pada semua atau beberapa *field* sesuai data masukan	Dapat mengisi *field* sesuai yang diharapkan	[√] diterima [ ] ditolak
Kasus dan hasil uji (data salah)			
Data masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**

Nama Angkot : "*Dago - St.Hall*" Warna : *" "* Rute Angkot : *"Riung – Binong – Kiara Condong – Siliwangi – Dipatiukur – Dago"* Gambar : "*wp11.jpg*"	Menampilkan pesan di setiap *field* yang tidak terisi bahwa data tersebut harus diisi atau data tidak boleh kosong	Fungsi berjalan sesuai dengan yang diharapkan	[√] diterima [ ] ditolak

4.2.2.19 Pengujian Ubah Transportasi

Pengujian ubah transportasi terdiri dari data-data transportasi yang akan diubah.

Tabel 4.24 Pengujian ubah transportasi

Kasus dan hasil uji (data normal)			
Data masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Nama Angkot : "*Dago - St.Hall*" Warna : *"Putih – Merah"* Rute Angkot : *"Riung – Binong – Kiara Condong – Siliwangi – Dipatiukur – Dago"* Gambar : "*wp11.jpg*"	Data transportasi baru dapat terisi pada semua atau beberapa *field*	Dapat mengisi *field* sesuai yang diharapkan	[√] diterima [ ] ditolak

4.2.2.20 Pengujian Hapus Transportasi

Pengujian hapus transportasi terdiri dari data-data transportasi yang akan di hapus.

Tabel 4.25 Pengujian hapus transportasi

Kasus dan hasil uji (data normal)			
Data masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Rute : "*Dago - St.Hall*"	Menampilkan konfirmasi data ingin dihapus dan data terhapus.	Fungsi berjalan sesuai dengan yang diharapkan	[√] diterima [ ] ditolak

4.2.3 Kesimpulan Hasil Uji *Alpha*

Berdasarkan hasil pengujian *Alpha* yang telah dilakukan dapat ditarik kesimpulan bahwa aplikasi sudah berjalan cukup maksimal, tetapi tidak menutup kemungkinan dapat terjadi kesalahan suatu saat, pada saat aplikasi digunakan, sehingga membutuhkan proses *maintenance* untuk lebih mengetahui kekurangan dari aplikasi.

4.2.4 Kasus dan Hasil Pengujian *Betha*

Pengujian *beta* merupakan pengujian yang dilakukan dengan tujuan untuk mengetahui sejauh mana kualitas sistem, apakah sudah memenuhi harapan atau belum. Untuk itu dalam pengujian *betha* dilakukan penelitian terhadap responden atau calon pengguna sistem dengan melakukan pengumpulan data menggunakan kuesioner atau angket. Metode yang digunakan adalah dalam penelitian ini adalah metode kuantitatif.

Pada penelitian ini pengujian *betha* dilakukan pada aplikasi *frontend* dan *backend*. Pengujian pada aplikasi *backend* dilakukan dengan melakukan *interview* terhadap seorang responden yang ditunjuk sebagai admin, sedangkan pengujian aplikasi *frontend* menggunakan kuesioner. Kusioner yang diberikan menggunakan Google Docs sehingga tidak terbatas jumlah pengguna yang dapat mengisinya. Didapat data terakhir sebanyak 10 orang yang telah menguji aplikasi *frontend*.

Berdasarkan hasil wawancara dengan responden, aplikasi *backend* (web admin) telah sesuai harapan baik secara fungsionalitas-fungsionalitasnya maupun tampilan aplikasi secara keseluruhan.

Kuesioner untuk pengguna sistem *frontend* (*enduser*) terdiri 6 pertanyaan dan menggunakan Skala Likert jawaban 1 sampai 5. Adapun pertanyaan, jawaban dan hasil dari kuesioner pengujian beta adalah sebagai berikut :

a. Pertanyaan

Berikut ini adalah pertanyaan yang diajukan kepada calon *user* dalam melakukan pengujian aplikasi ini :

1. Apakah tampilan aplikasi Bandung *Tour Guide* ini menarik ?
2. Apakah aplikasi Bandung *Tour Guide* ini mudah untuk digunakan?
3. Apakah informasi POI(restoran, penginapan, perbelanjaan, penginapan) yang ditampilkan cukup lengkap?
4. Apakah teknologi augmented reality yang ada pada aplikasi memudahkan pengguna dalam memperoleh informasi POI yang akan dituju?

5. Apakah informasi transportasi yang ditampilkan cukup lengkap dan memudahkan pengguna?
6. Apakah aplikasi Bandung *Tour Guide* ini dapat bermanfaat bagi pariwisata?

b. Jawaban

Berikut adalah jawaban yang disediakan dalam pengisian kuesioner dengan menggunakan skala Likert dari skala 1 - 5.

Tabel 4.26 Skala jawaban likert

Nilai	Keterangan
1	Sangat Setuju
2	Setuju
3	Netral
4	Tidak setuju
5	Sangat tidak setuju

Skala jawaban Likert pada tabel 4.26 tidak harus sepenuhnya selalu seperti itu, skala jawaban dapat disesuaikan dengan kebutuhan *aplikasi* agar dapat ditarik kesimpulan persepsi, sikap dan pendapat seseorang atau kelompok. Maka skala jawaban yang sesuai dengan calon *user* aplikasi Bandung Tour Guide dapat dilihat pada tabel 4.27 :

Tabel 4.27 Skala jawaban

Nilai	Keterangan
5	Sangat Setuju
4	Setuju
3	Cukup setuju
2	Kurang setuju
1	Sangat tidak setuju

Berdasarkan data hasil kuesioner tersebut, dicari prosentase masing-masing jawaban dengan menggunakan rumus :

$$H = \frac{Z}{J} X\ 100\%$$

Keterangan :

H = Hasil perhitungan

Z = Jumlah skala jawaban

J = Nilai tertinggi dikalikan dengan jumlah responden

Agar hasil perhitungan dapat diterjemahan kembali menjadi hasil skala jawaban maka diperlukan kriteria interpretasi sebagai berikut :

- 0 – 20% = Sangat tidak setuju
- 20 – 40% = Kurang setuju
- 40 – 60% = Cukup Setuju
- 60 – 80% = Setuju
- 80 – 100% = Sangat Setuju

Berikut ini hasil perhitungan prosentase dari jawaban hasil kuesioner yang didapat dari 10 responden yang telah menguji penelitian Aplikasi Bandung *Tour Guide* ini.

1. Pengolahan untuk pertanyaan pertama

Tabel 4.27 Hasil pengujian pertanyaan pertama

Kategori jawaban	Sangat Setuju	Setuju	Cukup Setuju	Kurang Setuju	Sangat Tidak Setuju	Jumlah
Responden	3	6	1	0	0	10
Nilai Frekuensi	15	24	3	0	0	42

$$H = \frac{42}{50}\ X\ 100\% = 84\%$$

Berdasarkan hasil prosentase tabel 4.18, maka dapat disimpulkan 84% menyatakan sangat setuju bahwa tampilan aplikasi Bandung Tour Guide sudah menarik.

2. Pengolahan untuk pertanyaan kedua

Tabel 4.28 Hasil pengujian pertanyaan kedua

Kategori jawaban	Sangat Setuju	Setuju	Cukup Setuju	Kurang Setuju	Sangat Tidak Setuju	Jumlah
Responden	4	6	0	0	0	10
Hasil Perhitungan	20	24	0	0	0	44

$$H = \frac{44}{50}\ X\ 100\% = 88\%$$

Berdasarkan hasil prosentase tabel 4.19 diatas, dapat disimpulkan bahwa sebanyak 88% menyatakan sangat setuju. Dengan hasil tersebut dapat disimpulkan bahwa Aplikasi Bandung *Tour Guide* ini mudah sekali digunakan.

3. Pengolahan untuk pertanyaan ketiga

Tabel 4.29 Hasil pengujian pertanyaan ketiga

Kategori jawaban	Sangat Setuju	Setuju	Cukup Setuju	Kurang Setuju	Sangat Tidak Setuju	Jumlah
Responden	2	4	4	0	0	10
Hasil Perhitungan	10	16	12	0	0	38

$$H = \frac{38}{50}\ X\ 100\% = 84\%$$

Berdasarkan hasil prosentase tabel 4.20, dapat dilihat bahwa 84% menyatakan sangat setuju, dengan demikian dapat disimpulkan bahwa informasi POI yang ditampilkan dalam Aplikasi Bandung *Tour Guide* ini sudah lengkap.

4. Pengolahan untuk pertanyaan keempat

Tabel 4.30 Hasil pengujian pertanyaan keempat

Kategori jawaban	Sangat Setuju	Setuju	Cukup Setuju	Kurang Setuju	Sangat Tidak Setuju	Jumlah
Responden	0	6	3	1	0	10
Hasil Perhitungan	0	24	9	2	0	35

$$H = \frac{35}{50}\ X\ 100\% = 70\%$$

Berdasarkan hasil prosentase tabel 4.21, dapat dilihat bahwa sebanyak 6 orang atau 70% menyatakan setuju. Dengan hasil tersebut dapat disimpulkan bahwa teknologi yang ada pada Aplikasi Bandung *Tour Guide* ini dapat memudahkan pengguna dalam memperoleh informasi POI yang akan dituju.

5. Pengolahan untuk pertanyaan kelima

Tabel 4.31 Hasil pengujian pertanyaan kelima

Kategori jawaban	Sangat Setuju	Setuju	Cukup Setuju	Kurang Setuju	Sangat Tidak Setuju	Jumlah
Responden	1	7	1	1	0	10
Hasil Perhitungan	5	28	3	2	0	38

$$H = \frac{32}{50}\ X\ 100\% = 84\%$$

Berdasarkan hasil prosentase tabel 4.22, dapat dilihat bahwa 84% menyatakan sangat setuju. Dengan hasil tersebut dapat disimpulkan bahwa informasi transportasi yang ditampilkan sudah cukup lengkap dan mudah diperoleh.

6. Pengolahan untuk pertanyaan keenam

Tabel 4.32 Hasil pengujian pertanyaan keenam

Kategori jawaban	Sangat Setuju	Setuju	Cukup Setuju	Kurang Setuju	Sangat Tidak Setuju	Jumlah
Responden	4	4	2	0	0	10
Hasil Perhitungan	20	16	6	0	0	4

$$H = \frac{42}{50}\ X\ 100\% = 84\%$$

Berdasarkan hasil prosentase tabel 4.22, didapat hasil 84% menyatakan sangat setuju. Dengan hasil tersebut dapat disimpulkan bahwa Aplikasi Bandung *Tour Guide* ini dapat bermanfaat bagi pariwisata di kota bandung.

4.3 Kesimpulan Hasil Pengujian

Berdasarkan hasil pengujian *alpha* dengan kasus uji *sample* diatas dapat ditarik kesimpulan bahwa sistem bebas dari kesalahan sintaks dan secara fungsional mengeluarkan hasil yang sesuai dengan yang diharapkan.

Sedangkan dalam pengujian *betha* dapat ditarik kesimpulan bahwa sistem yang dibangun sudah *user friendly*, mudah digunakan, sangat bermanfaat dan dapat membantu pengguna dalam memandu tempat-tempat yang menjadi objek persinggahan para turis, ini dapat dilihat dari presentase jawaban setiap pengguna atau responden terhadap pertanyaan yang diajukan.