BAB IV

IMPLEMENTASI DAN PENGUJIAN

4. 1 Implementasi

Tahap implementasi sistem merupakan tahap penterjemahan perancangan berdasarkan hasil analisis ke dalam suatu bahasa pemrograman tertentu serta penerapan perangkat lunak yang dibangun pada lingkungan yang sesungguhnya. Tahap ini merupakan tahap dimana sistem siap untuk dioperasikan, yang terdiri dari penjelasan mengenai lingkungan implementasi dan implementasi program. Adapun pembahasan implementasi terdiri dari perangkat lunak pembangun, perangkat keras pembangun, dan implementasi antarmuka.

4. 1. 1 Perangkat Lunak Pembangun

Perangkat lunak pembangun menjelaskan perangkat lunak yang digunakan untuk membangun aplikasi *mobile* Bandung Review. Adapun perangkat lunak yang digunakan untuk membangun aplikasi *mobile* Bandung Review adalah sebagai berikut:

1. Windows 7 Ultimate x64 sebagai sistem operasi.
2. Visual Studio 2010 Ultimate sebagai perangkat lunak pembangun.
3. Zune Software.
4. Windows Phone SDK 7.1.

4. 1. 2 Perangkat Keras Pembangun

Perangkat keras pembangun menjelaskan perangkat keras yang digunakan untuk membangun aplikasi *mobile* Bandung Review. Adapun perangkat keras yang

digunakan untuk membangun aplikasi *mobile* Bandung Review adalah sebagai berikut:

1. *Processor* AMD Turion X2 Dual-Core 2.3 GHz
2. RAM 2 GB
3. *Harddisk* 320 GB
4. *Monitor* LCD 15.5”
5. *Keyboard* dan *Mouse*

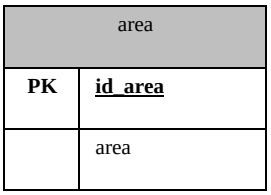
4. 1. 3 Implementasi Basis Data

Basis data yang digunakan oleh BandungReview.com menggunakan DBMS MySQL. Implementasi basis data dalam bahasa SQL seperti tertera pada Tabel 4.1 sampai dengan Tabel 4.15.

3.2.4.1 Tabel area

Berikut *syntax* hasil *generate* tabel area ke dalam basis data MySQL dapat dilihat pada Tabel 4.1.

Tabel 4. 1 Implementasi Tabel area

Tabel	***Syntax* MySQL**
	CREATE TABLE IF NOT EXISTS `area` (`id_area` int(11) NOT NULL AUTO_INCREMENT, `area` varchar(255) DEFAULT NULL, PRIMARY KEY (`id_area`)) ENGINE=InnoDB DEFAULT CHARSET=latin1 AUTO_INCREMENT=22 ;

3.2.4.2 Tabel article_v1

Berikut *syntax* hasil *generate* tabel article_v1 ke dalam basis data MySQL dapat dilihat pada Tabel 4.2.

Tabel 4. 2 Implementasi Tabel article_v1

Tabel	***Syntax* MySQL**
article_v1 **PK** **id** **FK** article_category_id title content article_file_name article_content_type article_file_size article_date **FK** user_id is_published created_at updated_at content_indonesian_version title_indonesian_version is_locked_byadmin mobile_article_file_name mobile_content_type mobile_file_size count_comment	CREATE TABLE IF NOT EXISTS `articles_v1` (`id` bigint(20) NOT NULL AUTO_INCREMENT, `article_category_id` bigint(20) DEFAULT '0', `title` varchar(250) DEFAULT NULL, `content` text, `article_file_name` varchar(255) DEFAULT NULL, `article_content_type` varchar(255) DEFAULT NULL, `article_file_size` int(11) DEFAULT NULL, `article_date` datetime DEFAULT NULL, `user_id` bigint(20) DEFAULT '0', `is_published` tinyint(4) DEFAULT NULL, `created_at` datetime DEFAULT NULL, `updated_at` datetime DEFAULT NULL, `content_indonesian_version` text, `title_indonesian_version` varchar(250) DEFAULT NULL, `is_locked_byadmin` tinyint(4) DEFAULT '0', `mobile_article_file_name` varchar(255) DEFAULT NULL, `mobile_content_type` varchar(255) DEFAULT NULL, `mobile_file_size` int(11) DEFAULT NULL,

	`count_comment` int(11) DEFAULT '0', PRIMARY KEY (`id`), FOREIGN KEY (article_category_id) REFERENCES article_categories_v1(id), FOREIGN KEY (user_id) REFERENCES zf_user(user_id)) ENGINE= InnoDB DEFAULT CHARSET=latin1 AUTO_INCREMENT=365 ;

3.2.4.3 Tabel article_categories_v1

Berikut *syntax* hasil *generate* tabel article_categories_v1 ke dalam basis data MySQL dapat dilihat pada Tabel 4.3.

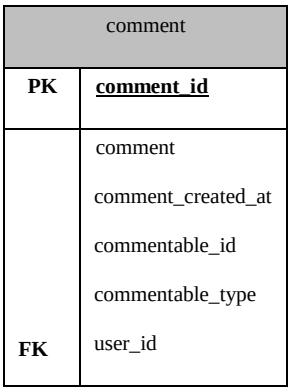
Tabel 4. 3 Implementasi Tabel article_categories_v1

Tabel	***Syntax* MySQL**
article_categories_v1 PK: **id** name label sorted_number is_show	CREATE TABLE IF NOT EXISTS `article_categories_v1` (`id` bigint(20) NOT NULL AUTO_INCREMENT, `name` varchar(250) DEFAULT NULL, `label` varchar(255) DEFAULT NULL, `sorted_number` smallint(6) DEFAULT '1', `is_show` smallint(6) DEFAULT '1', PRIMARY KEY (`id`)) ENGINE= InnoDB DEFAULT CHARSET=latin1 AUTO_INCREMENT=12 ;

3.2.4.4 Tabel comment

Berikut *syntax* hasil *generate* tabel comment ke dalam basis data MySQL dapat dilihat pada Tabel 4.4.

Tabel 4. 4 Implementasi Tabel comment

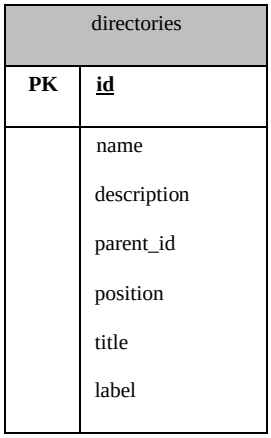
Tabel	***Syntax* MySQL**
	CREATE TABLE IF NOT EXISTS `comments` (`comment_id` bigint(20) NOT NULL AUTO_INCREMENT, `comment` text, `comment_created_at` datetime NOT NULL, `commentable_id` bigint(20) DEFAULT '0', `commentable_type` varchar(15) NOT NULL, `user_id` bigint(20) DEFAULT '0', PRIMARY KEY (`comment_id`), KEY `fk_comments_user` (`user_id`), FOREIGN KEY (user_id) REFERENCES zf_user(user_id)) ENGINE=InnoDB DEFAULT CHARSET=latin1 AUTO_INCREMENT=414 ;

3.2.4.5 Tabel directories

Berikut *syntax* hasil *generate* tabel directories ke dalam basis data MySQL dapat dilihat pada Tabel 4.5.

Tabel 4. 5 Implementasi Tabel directories

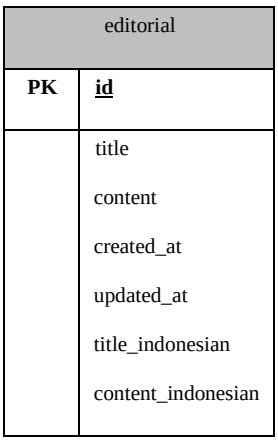
Tabel	***Syntax* MySQL**

	CREATE TABLE IF NOT EXISTS `directories` (`id` bigint(20) NOT NULL AUTO_INCREMENT, `name` varchar(255) DEFAULT NULL, `description` varchar(255) DEFAULT NULL, `parent_id` bigint(20) DEFAULT '0', `position` int(11) DEFAULT NULL, `title` varchar(70) DEFAULT NULL, `label` varchar(50) DEFAULT NULL, PRIMARY KEY (`id`)) ENGINE=InnoDB DEFAULT CHARSET=latin1 AUTO_INCREMENT=46 ;

3.2.4.6 Tabel editorial

Berikut *syntax* hasil *generate* tabel editorial ke dalam basis data MySQL dapat dilihat pada Tabel 4.6.

Tabel 4. 6 Implementasi Tabel editorial

Tabel	***Syntax* MySQL**
	CREATE TABLE IF NOT EXISTS `editorial` (`id` int(11) NOT NULL AUTO_INCREMENT, `title` varchar(255) DEFAULT NULL, `content` text, `created_at` datetime DEFAULT '2010-10-04 00:00:00', `updated_at` datetime DEFAULT NULL, `title_indonesian` varchar(255) DEFAULT NULL, `content_indonesian` text,

	PRIMARY KEY (`id`)) ENGINE=InnoDB DEFAULT CHARSET=latin1 AUTO_INCREMENT=2 ;

3.2.4.7 Tabel events

Berikut *syntax* hasil *generate* tabel event ke dalam basis data MySQL dapat dilihat pada Tabel 4.7.

Tabel 4. 7 Implementasi Tabel event

Tabel	***Syntax* MySQL**

event	
PK	**id**
FK	event_category_id
	title
	content
	event_file_name
	event_content_type
	event_file_size
	event_date
FK	place_id
	is_published
FK	user_id
	created_at
	updated_at
	event_date_end
	place_outside
	title_indonesian_version
	content_indonesia_version
	is_infront
	long
	lat
	is_hot

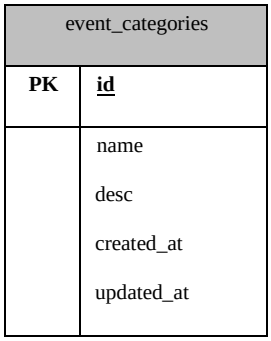
```
CREATE TABLE IF NOT EXISTS `events` (
  `id` bigint(20) NOT NULL
AUTO_INCREMENT,
  `event_category_id` bigint(20) DEFAULT '0',
  `title` varchar(250) DEFAULT NULL,
  `content` text,
  `event_file_name` varchar(255) DEFAULT
NULL,
  `event_content_type` varchar(255) DEFAULT
NULL,
  `event_file_size` int(11) DEFAULT NULL,
  `event_date` datetime DEFAULT NULL,
  `place_id` bigint(20) DEFAULT '0',
  `is_published` tinyint(4) DEFAULT NULL,
  `user_id` bigint(20) DEFAULT '0',
  `created_at` datetime DEFAULT NULL,
  `updated_at` datetime DEFAULT NULL,
  `event_date_end` datetime DEFAULT NULL,
  `place_outside` varchar(255) DEFAULT NULL,
  `title_indonesian_version` varchar(250)
DEFAULT NULL,
  `content_indonesian_version` text,
  `is_infront` tinyint(4) DEFAULT '0',
  `long` varchar(30) DEFAULT '0',
  `lat` varchar(30) DEFAULT '0',
  `is_hot` tinyint(4) DEFAULT '0',
  PRIMARY KEY (`id`),
FOREIGN KEY (event_category_id)
REFERENCES event _categories(id),
```

	FOREIGN KEY (place_id) REFERENCES place(place_id), FOREIGN KEY (user_id) REFERENCES zf_user(user_id)) ENGINE=InnoDB DEFAULT CHARSET=latin1 AUTO_INCREMENT=418 ;

3.2.4.8 Tabel event_categories

Berikut *syntax* hasil *generate* tabel event_categories ke dalam basis data MySQL dapat dilihat pada Tabel 4.8.

Tabel 4. 8 Implementasi Tabel event_categories

Tabel	***Syntax* MySQL**
	CREATE TABLE IF NOT EXISTS `event_categories` (`id` bigint(20) NOT NULL AUTO_INCREMENT, `name` varchar(50) DEFAULT NULL, `desc` varchar(100) DEFAULT NULL, `created_at` datetime DEFAULT NULL, `updated_at` datetime DEFAULT NULL, PRIMARY KEY (`id`)) ENGINE=InnoDB DEFAULT CHARSET=latin1 AUTO_INCREMENT=5 ;

3.2.4.9 Tabel place

Berikut *syntax* hasil *generate* tabel place ke dalam basis data MySQL dapat dilihat pada Tabel 4.9.

Tabel 4. 9 Implementasi Tabel place

Tabel	***Syntax* MySQL**

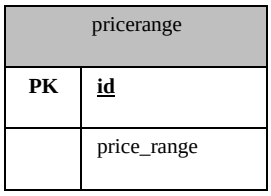
place	
PK	**id**
FK	directory_id
	code
	name
	phone
	website
	email
	street
	city
	state
	zipcode
	long
	lat
	logo_file_name
	logo_content_type
	logo_file_size
	image_file_name
	image_content_type
	image_file_size
	image1_file_name
	image1_content_type
	image1_file_size
	image2_file_name
	image2_content_type
	image2_file_size
	image3_file_name
	image3_content_type
	image3_file_size
	image4_file_name
	image4_content_type
	image4_file_size
	opening_hour
	foodtype

```
CREATE TABLE IF NOT EXISTS `places` (
  `id` bigint(20) NOT NULL
AUTO_INCREMENT,
  `directory_id` bigint(20) DEFAULT '0',
  `code` varchar(10) NOT NULL,
  `name` varchar(255) DEFAULT NULL,
  `phone` varchar(255) DEFAULT NULL,
  `website` varchar(255) DEFAULT NULL,
  `email` varchar(100) DEFAULT NULL,
  `street` varchar(255) DEFAULT NULL,
  `city` varchar(255) DEFAULT NULL,
  `state` varchar(255) DEFAULT NULL,
  `zipcode` varchar(255) DEFAULT NULL,
  `long` varchar(25) DEFAULT NULL,
  `lat` varchar(25) DEFAULT NULL,
  `logo_file_name` varchar(255) DEFAULT
NULL,
  `logo_content_type` varchar(255) DEFAULT
NULL,
  `logo_file_size` int(11) DEFAULT NULL,
  `image_file_name` varchar(255) DEFAULT
NULL,
  `image_content_type` varchar(255) DEFAULT
NULL,
  `image_file_size` int(11) DEFAULT NULL,
  `image1_file_name` varchar(255) DEFAULT
NULL,
  `image1_content_type` varchar(255) DEFAULT
NULL,
  `image1_file_size` int(11) DEFAULT NULL,
  `image2_file_name` varchar(255) DEFAULT
NULL,
  `image2_content_type` varchar(255) DEFAULT
```

3.2.4.10 Tabel pricerange

Berikut *syntax* hasil *generate* tabel pricerange ke dalam basis data MySQL dapat dilihat pada Tabel 4.10.

Tabel 4. 10 Implementasi Tabel pricerange

Tabel	***Syntax* MySQL**
	CREATE TABLE IF NOT EXISTS `pricerange` (`id` bigint(20) NOT NULL AUTO_INCREMENT, `price_range` varchar(70) DEFAULT NULL, PRIMARY KEY (`id`)) ENGINE=InnoDB DEFAULT CHARSET=latin1 AUTO_INCREMENT=6 ;

3.2.4.11 Tabel promotions

Berikut *syntax* hasil *generate* tabel promotions ke dalam basis data MySQL dapat dilihat pada Tabel 4.11.

Tabel 4. 11 Implementasi Tabel promotions

Tabel	***Syntax* MySQL**

promotions	
PK	**id**
	title
	content
	start_date
	end_date
	is_published
FK	place_id
	place_outside
FK	user_id
	created_at
	updated_at
	description
	description_english_version
FK	catid
	promotion_file_name
	promotion_content_type
	promotion_file_size
	is_infront
	lat
	long

```
CREATE TABLE IF NOT EXISTS `promotions`
(
  `id` bigint(20) NOT NULL
AUTO_INCREMENT,
  `title` varchar(250) DEFAULT NULL,
  `content` text,
  `start_date` datetime DEFAULT NULL,
  `end_date` datetime DEFAULT NULL,
  `is_published` tinyint(4) DEFAULT NULL,
  `place_id` bigint(20) DEFAULT '0',
  `place_outside` varchar(40) DEFAULT NULL,
  `user_id` bigint(20) DEFAULT '0',
  `created_at` datetime DEFAULT NULL,
  `updated_at` datetime DEFAULT NULL,
  `description` text,
  `description_english_version` text,
  `catid` smallint(6) DEFAULT '2',
  `promotion_file_name` varchar(255) DEFAULT
NULL,
  `promotion_content_type` varchar(50)
DEFAULT NULL,
  `promotion_file_size` int(11) DEFAULT NULL,
  `is_infront` smallint(6) DEFAULT '0',
  `lat` varchar(30) DEFAULT '0',
  `long` varchar(30) DEFAULT '0',
  PRIMARY KEY (`id`),
FOREIGN KEY (catid) REFERENCES
promotion_category_v1(catid),
FOREIGN KEY (place_id) REFERENCES
```

	place(place_id), FOREIGN KEY (user_id) REFERENCES zf_user(user_id)) ENGINE=InnoDB DEFAULT CHARSET=latin1 AUTO_INCREMENT=395 ;

3.2.4.12 Tabel promotion_category_v1

Berikut *syntax* hasil *generate* tabel promotion_category_v1 ke dalam basis data MySQL dapat dilihat pada Tabel 4.12.

Tabel 4. 12 Implementasi Tabel promotion_category_v1

Tabel	***Syntax* MySQL**
promotion_category_v1 PK: **catid** catname parent_catid label	CREATE TABLE IF NOT EXISTS `promotion_category_v1` (`catid` smallint(11) NOT NULL AUTO_INCREMENT, `catname` varchar(50) DEFAULT NULL, `parent_catid` smallint(6) DEFAULT '0', `label` varchar(50) DEFAULT NULL, PRIMARY KEY (`catid`)) ENGINE=InnoDB DEFAULT CHARSET=latin1 AUTO_INCREMENT=8 ;

3.2.4.13 Tabel userinfo

Berikut *syntax* hasil *generate* tabel userinfo ke dalam basis data MySQL dapat dilihat pada Tabel 4.13.

Tabel 4. 13 Implementasi Tabel userinfo

Tabel	***Syntax* MySQL**

userinfo	
PK	**userinfo_id**
FK	user_id
	firstname
	lastname
	place_birthday
	date_birthday
	sex
	marital_status
	address
	post_code
	city
	province
	region
	home_phone
	office_phone
	fax
	mobile_phone_1
	mobile_phone_2
	email_address_1
	email_address_2
	last_education
	education_place_name
	currency_job
	job_place_name
	job_address
	job_post_code
	job_city
	job_province
	job_region
	revenue_per_year
	location
	aboutme

```
CREATE TABLE IF NOT EXISTS `userinfo` (
  `userinfo_id` bigint(20) NOT NULL AUTO_INCREMENT,
  `user_id` bigint(20) DEFAULT NULL,
  `firstname` varchar(255) DEFAULT NULL,
  `lastname` varchar(255) DEFAULT NULL,
  `place_birthday` text,
  `date_birthday` date DEFAULT NULL,
  `sex` varchar(50) DEFAULT NULL,
  `marrital_status` varchar(50) DEFAULT NULL,
  `address` text,
  `post_code` varchar(6) DEFAULT NULL,
  `city` varchar(50) DEFAULT NULL,
  `province` varchar(50) DEFAULT NULL,
  `region` varchar(50) DEFAULT NULL,
  `home_phone` varchar(50) DEFAULT NULL,
  `office_phone` varchar(50) DEFAULT NULL,
  `fax` varchar(50) DEFAULT NULL,
  `mobile_phone_1` varchar(50) DEFAULT NULL,
  `mobile_phone_2` varchar(50) DEFAULT NULL,
  `email_address_1` varchar(255) DEFAULT NULL,
  `email_address_2` varchar(255) DEFAULT NULL,
  `last_education` varchar(50) DEFAULT NULL,
  `education_place_name` varchar(255) DEFAULT NULL,
```

	`currency_job` varchar(255) DEFAULT NULL, `job_place_name` varchar(255) DEFAULT NULL, `job_address` text, `job_post_code` varchar(6) DEFAULT NULL, `job_city` varchar(50) DEFAULT NULL, `job_province` varchar(50) DEFAULT NULL, `job_region` varchar(50) DEFAULT NULL, `revenue_per_year` varchar(255) DEFAULT NULL, `location` varchar(50) DEFAULT NULL, `aboutme` text, PRIMARY KEY (`userinfo_id`), FOREIGN KEY (user_id) REFERENCES zf_user(user_id)) ENGINE=InnoDB DEFAULT CHARSET=latin1 AUTO_INCREMENT=780 ;

3.2.4.14 Tabel zf_users

Berikut *syntax* hasil *generate* tabel zf_users ke dalam basis data MySQL dapat dilihat pada Tabel 4.14.

Tabel 4. 14 Implementasi Tabel zf_users

Tabel	***Syntax* MySQL**

zf_user	
PK	**user_id**
FK	user_name
	passwd
	information
	is_active
	role_id
	email
	salt
	membership
	last_company
	created_at
	updated_at
	remember_token
	remember_token_expires_at
	activation_code
	activated_at
	password_reset_code
	deleted_at
	updated_by
	photo_file_name
	photo_content_type
	photo_file_size
	has_survey
	ip_address_signup

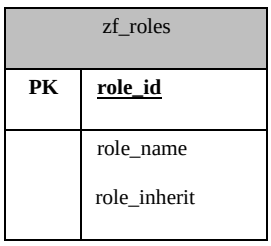
```
CREATE TABLE IF NOT EXISTS `zf_users` (
  `user_id` bigint(20) NOT NULL AUTO_INCREMENT,
  `user_name` varchar(255) DEFAULT NULL,
  `passwd` varchar(255) DEFAULT NULL,
  `information` text,
  `is_active` smallint(6) DEFAULT '0',
  `role_id` bigint(20) DEFAULT NULL,
  `email` varchar(40) DEFAULT NULL,
  `salt` varchar(40) DEFAULT NULL,
  `membership` varchar(10) DEFAULT NULL,
  `last_company` bigint(20) DEFAULT NULL,
  `created_at` datetime DEFAULT NULL,
  `updated_at` datetime DEFAULT NULL,
  `remember_token` varchar(40) DEFAULT NULL,
  `remember_token_expires_at` datetime DEFAULT NULL,
  `activation_code` varchar(255) DEFAULT NULL,
  `activated_at` datetime DEFAULT NULL,
  `password_reset_code` varchar(255) DEFAULT NULL,
  `deleted_at` datetime DEFAULT NULL,
  `updated_by` bigint(20) DEFAULT NULL,
  `photo_file_name` varchar(255) DEFAULT NULL,
  `photo_content_type` varchar(255) DEFAULT NULL,
```

	`photo_file_size` bigint(20) DEFAULT NULL, `has_survey` smallint(6) DEFAULT '0', `ip_address_signup` varchar(255) DEFAULT '-notdetected-', PRIMARY KEY (`user_id`), FOREIGN KEY (role_id) REFERENCES zf_roles (role_id),) ENGINE=InnoDB DEFAULT CHARSET=latin1 AUTO_INCREMENT=1688 ;

3.2.4.15 Tabel zf_roles

Berikut *syntax* hasil *generate* tabel zf_roles ke dalam basis data MySQL dapat dilihat pada Tabel 4.15.

Tabel 4. 15 Implementasi Tabel zf_roles

Tabel	***Syntax* MySQL**
	CREATE TABLE IF NOT EXISTS `zf_roles` (`role_id` bigint(20) NOT NULL AUTO_INCREMENT, `role_name` varchar(70) DEFAULT NULL, `role_inherit` bigint(20) DEFAULT '0', PRIMARY KEY (`role_id`)) ENGINE=InnoDB DEFAULT CHARSET=latin1 AUTO_INCREMENT=6 ;

4. 1. 4 Implementasi Antarmuka

Implementasi antarmuka berisi pemaparan setiap tampilan aplikasi *mobile* Bandung Review yang dibangun beserta *file* .xaml sebagai *layout* dari masing-

masing antarmuka seperti terlihat pada Tabel 4.16. Adapun gambar dari tampilan implementasi antarmuka aplikasi *mobile* Bandung Review tertera pada halaman Lampiran A Implementasi Antarmuka.

Tabel 4. 16 Implementasi Antarmuka Aplikasi *Mobile* Bandung Review

No.	Halaman	Deskripsi	*File* .xaml
1	*Home*	Halaman utama, berisi menu utama, daftar direktori, favorit dan *hot pick*	MainPage
2	Tempat	Halaman yang berisi detail dari suatu tempat	PlacePage
3	Direktori	Halaman yang berisi seluruh daftar direktori pada Bandung Review	DirectoryPage
4	Menu	Halaman yang berisi daftar artikel, daftar *event*, daftar promosi, *advertorial* dan editorial	MenuPage
5	Artikel	Halaman yang berisi detail artikel	ArticlePage
6	*Event*	Halaman yang berisi detail info event	EventPage
7	Promosi	Halaman yang berisi detail info promosi	PromotionPage
8	Daftar Tempat	Halaman yang berisi daftar tempat berdasarkan direktori tertentu	PlaceListPage
9	*Login*	Halaman yang digunakan untuk masuk sebagai *member* Bandung Review	LoginPage
10	*Register*	Halaman yang digunakan untuk pendaftaran akun baru Bandung Review	RegisterPage
11	Profil	Halaman yang berisi info pengguna	ProfilePage
12	Ubah Profil	Halaman yang digunakan untuk mengubah data profil pengguna	EditProfilePage
13	Pencarian	Halaman yang digunakan untuk	SearchPage

		melakukan pencarian tempat	
14	*About*	Halaman yang berisi penjelasan mengenai aplikasi *mobile* Bandung Review	AboutPage
15	Pengaturan	Halaman yang berisi pengaturan penggunaan *location service*	SettingPage

4. 1. 5 Implementasi Kelas

Implementasi kelas menggambarkan struktur *file* (.cs) yang berisi kelas-kelas dalam aplikasi *mobile* Bandung Review. Berikut ini implementasi dari kelas-kelas aplikasi *mobile* Bandung Review seperti terlihat pada Tabel 4.17.

Tabel 4. 17 Implementasi Kelas Aplikasi *Mobile* Bandung Review

No.	Kelas	*File* .cs
1	AboutPage	AboutPage.cs
2	ArticlePage	ArticlePage.cs
3	DirectoryPage	DirectoryPage.cs
4	EditProfilePage	EditProfilePage.cs
5	EventPage	EventPage.cs
6	LoginPage	LoginPage.cs
7	MainPage	MainPage.cs
8	ManuPage	ManuPage.cs
9	PlaceListPage	PlaceListPage.cs
10	PlacePage	PlacePage.cs
11	ProfilePage	ProfilePage.cs
12	PromotionPage	PromotionPage.cs
13	RegisterPage	RegisterPage.cs
14	SearchPage	SearchPage.cs
15	SettingPage	SettingPage.cs
16	Foursquare	Foursquare.cs

17	FoursquareHelper	FoursquareHelper.cs
18	WebBrowserHelper	WebBrowserHelper.cs
19	AppSettings	AppSettings.cs
20	FavouriteItem	FavouriteItem.cs
21	LocationService	LocationService.cs
22	Article	Article.cs
23	ArticleInfo	ArticleInfo.cs
24	Comment	Comment.cs
25	Directory	Directory.cs
26	Editorial	Editorial.cs
27	Event	Event.cs
28	EventInfo	EventInfo.cs
29	Location	Location.cs
30	Place	Place.cs
31	PlaceInfo	PlaceInfo.cs
32	Promotion	Promotion.cs
33	PromotionInfo	PromotionInfo.cs
34	UserInfo	UserInfo.cs
35	AdvertorialViewModel	AdvertorialViewModel.cs
36	ArticleViewModel	ArticleViewModel.cs
37	CommentViewModel	CommentViewModel.cs
38	DirectoryViewModel	DirectoryViewModel.cs
39	EditorialViewModel	EditorialViewModel.cs
40	EventViewModel	EventViewModel.cs
41	HotPickViewModel	HotPickViewModel.cs
42	LocationViewModel	LocationViewModel.cs
43	NearbyViewModel	NearbyViewModel.cs
44	PlaceViewModel	PlaceViewModel.cs
45	PromotionViewModel	PromotionViewModel.cs
46	SearchViewModel	SearchViewModel.cs

47	UserViewModel	UserViewModel.cs

4. 2 Pengujian Perangkat Lunak

Pengujian perangkat lunak merupakan tahapan untuk menemukan kesalahan-kesalahan dan kekurangan-kekurangan pada perangkat lunak yang dibangun sehingga bisa diketahui apakah perangkat lunak tersebut telah memenuhi kreteria sesuai dengan tujuan atau tidak. Adapun metode pengujian yang digunakan pada perangkat lunak ini adalah metode pengujian *black box*. Pengujian *black box* berfokus pada persyaratan fungsional perangkat lunak. Metode pengujian *black box* ini terdiri dari dua tahapan pengujian, yaitu tahapan pengujian alpha dan tahapan pengujian beta.

4. 2. 1 Pengujian *Black Box*

Pengujian *black box* merupakan pengujian fungsional diadakan di lingkungan pembangun oleh sekumpulan pengguna yang akan menggunakan perangkat lunaknya. Pihak pembangun mendampingi serta mencatat kesalahan-kesalahan maupun permasalahan yang dirasakan oleh pengguna. Pengujian *black box* yang dilakukan pada aplikasi *mobile* Bandung Review dibagi menjadi tiga bagian, yaitu skenario pengujian, kasus dan hasil pengujian, dan kesimpulan pengujian.

3.2.4.1 Skenario Pengujian

Skenario pengujian memaparkan urutan dan hal yang diuji pengujian yang dilakukan pada aplikasi *mobile* Bandung Review. Adapun skenario pengujian yang dilakukan dapat dilihat pada Tabel 4.18.

Tabel 4. 18 Skenario Pengujian Aplikasi *Mobile* Bandung Review

Kelas Uji	Butir Uji	Jenis Pengujian
Favorites	Menekan tombol favorit	*Black Box*
	Menekan tombol hapus favorit	*Black Box*
Login	Isi data *login*	*Black Box*
	Validasi *field login*	*Black Box*
	Menekan tombol *login*	*Black Box*
Register	Isi data *register*	*Black Box*
	Validasi *field register*	*Black Box*
	Menekan tombol *register*	*Black Box*
Memberi komentar pada tempat tertentu	Mengisi data komentar	*Black Box*
	Validasi *field* komentar	*Black Box*
	Menekan *keyboard enter*	*Black Box*
Memberi komentar pada artikel tertentu	Mengisi data komentar	*Black Box*
	Validasi *field* komentar	*Black Box*
	Menekan *keyboard enter*	*Black Box*
Menambahkan artikel	Mengisi data artikel	*Black Box*
	Validasi *field* artikel	*Black Box*
	Menekan tombol *publish*	*Black Box*
Pencarian tempat	Mengisi data pencarian	*Black Box*
	Validasi *field* pencarian	*Black Box*
	Menekan *keyboard enter*	*Black Box*
Ubah data profil	Mengisi data profil	*Black Box*
	Validasi *field* profil	*Black Box*
	Menekan tombol *save*	*Black Box*
Ubah data *password*	Mengisi data profil	*Black Box*
	Validasi *field* profil	*Black Box*
	Menekan tombol *save*	*Black Box*

Ubah data artikel	Mengisi data artikel	*Black Box*
	Validasi *field* artikel	*Black Box*
	Menekan tombol *update*	*Black Box*
Setting	Pengaturan *location service on*	*Black Box*
	Pengaturan *location service off*	*Black Box*

3.2.4.2 Kasus dan Hasil Pengujian

Kasus dan hasil pengujian berisi pemaparan dari rencana pengujian yang telah disusun pada skenario pengujian. Pengujian ini dilakukan secara *black box* dengan hanya memperhatikan masukan ke dalam sistem dan keluaran dari masukan tersebut. Berikut ini pemaparan dari setiap butir pengujian yang terdapat pada skenario pengujian:

4.2.1.2.1 Pengujian *Favorites*

Pengujian *favorites* memaparkan pengujian yang dilakukan terhadap aktivitas pengguna saat melakukan *favorites* pada aplikasi *mobile* Bandung Review. Pengujian *favorites* menggunakan pengujian dengan kasus data benar. Berikut ini penguijan *favorites* dengan kasus data benar yang dapat dilihat pada Tabel 4.19.

Tabel 4. 19 Pengujian Favorites Data Benar

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Tekan tombol *Favorite*	Ketika menekan tombol *favorite* maka akan menambahkan data tempat ke menu	Data tempat terdapat pada menu *favorites* dan *button*	Diterima

	favorites dan *button favorite* berubah menjadi *button unfavorite*	*favorite* berubah menjadi *button unfavorite*	
Tekan tombol *Unfavorite*	Ketika menekan tombol *unfavorite* maka akan menghapus data tempat favorit dan *button unfavorite* berubah menjadi *button favorite*	Data tempat terhapus pada menu *favorites* dan *button unfavorite* berubah menjadi *button favorite*	Diterima

4.2.1.2.2 Pengujian *Login*

Pengujian *login* memaparkan pengujian yang dilakukan terhadap aktivitas pengguna saat melakukan *login* pada aplikasi *mobile* Bandung Review. Pengujian *login* terdiri dari dua tipe pengujian, yaitu pengujian dengan kasus data benar dan pengujian dengan kasus data salah. Berikut ini penguijan *login* dengan kasus data benar yang dapat dilihat pada Tabel 4.20.

Tabel 4. 20 Pengujian *Login* Data Benar

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Username: danitaufani8	*Text box username* terisi sesuai dengan *username* yang dimasukkan	Isi dari *text box username* sesuai dengan *username* yang dimasukkan	Diterima
Password: 123456	*Text box password* terisi sesuai dengan	Isi dari *text box password*	Diterima

	password yang dimasukkan	sesuai dengan *password* yang dimasukkan	
Tekan tombol *Login*	Ketika menekan tombol *login* berpindah ke halaman utama dan menu *application bar* berubah	Pindah ke halaman utama dan menu *application bar* berubah	Diterima

Sementara itu untuk pengujian *login* dengan kasus data salah pada aplikasi *mobile* Bandung Review dipaparkan pada Tabel 4.21.

Tabel 4. 21 Pengujian *Login* Data Salah

Kasus dan Hasil Uji (Data Salah)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Username: (Kosong) dan *Password:* (Kosong)	Ketika menekan tombol *login* menampilkan pesan "*Username or password can't be blank*"	Tampil pesan "*Username or password can't be blank*"	Diterima
Username: danitaufani8 dan *Password:* (Kosong)	Ketika menekan tombol *login* menampilkan pesan "*Username or password can't be blank*"	Tampil pesan "*Username or password can't be blank*"	Diterima
Username: (Kosong) dan	Ketika menekan tombol *login* menampilkan pesan	Tampil pesan "*Username or password can't*	Diterima

Password: 123456	"*Username or password can't be blank*"	*be blank*"	
Username: danitaufani8 dan *Password:* XXXXXX	Ketika menekan tombol *login* menampilkan pesan "*ERROR: Invalid username or password*"	Tampil pesan "*ERROR: Invalid username or password*"	Diterima
Username: XXXXXX dan *Password:* 123456	Ketika menekan tombol *login* menampilkan pesan "*ERROR: Invalid username or password*"	Tampil pesan "*ERROR: Invalid username or password*"	Diterima

4.2.1.2.3 Pengujian *Register*

Pengujian *register* memaparkan pengujian yang dilakukan terhadap aktivitas pengguna saat melakukan *register* pada aplikasi *mobile* Bandung Review. Pengujian *register* terdiri dari dua tipe pengujian, yaitu pengujian dengan kasus data benar dan pengujian dengan kasus data salah. Berikut ini penguijan *login* dengan kasus data benar yang dapat dilihat pada Tabel 4.22.

Tabel 4. 22 Pengujian *Register* Data Benar

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
First name: Dani	*Text box first name* terisi sesuai dengan *first name* yang dimasukkan	Isi dari *text box first name* sesuai dengan *first name*	Diterima

		yang dimasukkan	
Last name: Taufani	*Text box first name* terisi sesuai dengan *first name* yang dimasukkan	Isi dari *text box last name* sesuai dengan *last name* yang dimasukkan	Diterima
Username: danitaufani	*Text box username* terisi sesuai dengan *username* yang dimasukkan	Isi dari *text box username* sesuai dengan *username* yang dimasukkan	Diterima
Email: danzgreyta@gmail.com	*Text box email* terisi sesuai dengan *email* yang dimasukkan	Isi dari *text box email* sesuai dengan *email* yang dimasukkan	Diterima
Password: 123456	*Text box password* terisi sesuai dengan *password* yang dimasukkan	Isi dari *text box password* sesuai dengan *password* yang dimasukkan	Diterima
Confirm password: 123456	*Text box confirm password* terisi sesuai dengan *confirm password* yang dimasukkan	Isi dari *text box confirm password* sesuai dengan *confirm password* yang dimasukkan	Diterima
Terms and Conditions:	*Check box terms*	Tombol *sign*	Diterima

Check box dalam kondisi tercentang	*and conditions* Bandung Review dalam kondisi sesuai dengan yang dipilih	*up* dalam kondisi *enable*	
Tekan tombol *sign up*	Ketika menekan tombol *sign up* menampilkan pesan "*You have been registered successfully! Please check your email to activate your account*"	Tampil pesan "*You have been registered successfully! Please check your email to activate your account*"	Diterima

Sementara itu untuk pengujian *register* dengan kasus data salah pada aplikasi *mobile* Bandung Review dipaparkan pada Tabel 4.23.

Tabel 4. 23 Pengujian *Register* Data Salah

Kasus dan Hasil Uji (Data Salah)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Mengkosongkan *text box first name, username, email* dan *password*	Ketika menekan tombol *sign up* menampilkan pesan "*Firstname, username, email and password are required*"	Tampil pesan "*Firstname, username, email and password are required*"	Diterima
Mengisi konfirmasi *password* yang berbeda dengan *password*,	Ketika menekan tombol *sign up* menampilkan pesan "*Password must match!*"	Tampil pesan "*Password must match!*"	Diterima

Contoh *Password*: 123456 *Confirm Pasword*: 654321			
Mengisi *username* atau *password* dengan jumlah karakter kurang dari 6. Contoh *Username:* dani *Password*: 123	Ketika menekan tombol *sign up* menampilkan pesan "*Username or password must contain at least 6 letters*"	Tampil pesan "*Username or password must contain at least 6 letters*"	Diterima
Mengisi *email* yang dengan format tidak valid. Contoh *Email*: dani.taufani.com	Ketika menekan tombol *sign up* menampilkan pesan "*Email is not in a valid format*"	Tampil pesan "*Email is not in a valid format*"	Diterima
Mengisi *username* atau *email* yang sudah digunakan	Ketika menekan tombol *sign up* menampilkan pesan "Username *or* e*mail already registered*"	Tampil pesan "Username *or* e*mail already registered*"	Diterima
Tidak mencentang *check box terms and conditions*	Ketika akan melakukan pendaftaran, tombol *sign up* dalam	Tombol *sign up* dalam kondisi *disable*	Diterima

	kondisi *disable*		

4.2.1.2.4 Pengujian Memberi Komentar Pada Tempat Tertentu

Pengujian memberi komentar pada tempat tertentu memaparkan pengujian untuk aktivitas memberi komentar pada tempat tertentu. Pengujian memberi komentar pada tempat tertentu terdiri dari dua tipe pengujian, yaitu pengujian dengan kasus data benar dan pengujian dengan kasus data salah. Berikut ini penguijan memberi komentar pada tempat tertentu dengan kasus data benar seperti terlihat pada Tabel 4.24.

Tabel 4. 24 Pengujian Memberi Komentar Pada Tempat Tertentu Data Benar

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Komentar: Halo	*Text box* komentar tempat terisi sesuai dengan komentar tempat yang dimasukkan	Isi dari *text box* komentar tempat sesuai dengan komentar yang dimasukkan	Diterima
	Ketika menekan tombol *send* menampilkan komentar baru pada *list box*	Muncul komentar baru pada *list box*	Diterima

Sementara itu untuk pengujian memberi komentar pada tempat tertentu dengan kasus data salah pada aplikasi *mobile* Bandung Review dipaparkan pada Tabel 4.25.

Tabel 4. 25 Pengujian Memberi Komentar Pada Tempat Tertentu Data Salah

Kasus dan Hasil Uji (Data Salah)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Komentar: (Kosong)	Ketika menekan tombol *send* tidak melakukan proses tambah komentar tempat	Proses tambah komentar tempat tidak dieksekusi	Diterima

4.2.1.2.5 Pengujian Memberi Komentar Pada Artikel Tertentu

Pengujian memberi komentar pada artikel tertentu memaparkan pengujian untuk aktivitas memberi komentar pada artikel tertentu. Pengujian memberi komentar pada artikel tertentu terdiri dari dua tipe pengujian, yaitu pengujian dengan kasus data benar dan pengujian dengan kasus data salah. Berikut ini penguijan memberi komentar pada artikel tertentu dengan kasus data benar seperti terlihat pada Tabel 4.26.

Tabel 4. 26 Pengujian Memberi Komentar Pada Artikel Tertentu Data Benar

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Komentar: Halo	*Text box* komentar artikel terisi sesuai dengan komentar artikel yang dimasukkan	Isi dari *text box* komentar artikel sesuai dengan	Diterima

		komentar yang dimasukkan	
	Ketika menekan tombol *send* menampilkan komentar baru pada *list box*	Muncul komentar baru pada *list box*	Diterima

Sementara itu untuk pengujian memberi komentar pada artikel tertentu dengan kasus data salah pada aplikasi *mobile* Bandung Review dipaparkan pada Tabel 4.27.

Tabel 4. 27 Pengujian Memberi Komentar Pada Artikel Tertentu Data Salah

Kasus dan Hasil Uji (Data Salah)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Komentar: (Kosong)	Ketika menekan tombol *send* tidak melakukan proses tambah komentar artikel	Proses tambah komentar artikel tidak dieksekusi	Diterima

4.2.1.2.6 Pengujian Menambahkan Artikel

Pengujian menambahkan artikel memaparkan pengujian untuk aktivitas menambahkan artikel. Pengujian menambahkan artikel terdiri dari dua tipe pengujian, yaitu pengujian dengan kasus data benar dan pengujian dengan kasus data salah. Berikut ini penguijan memberi komentar pada artikel tertentu dengan kasus data benar seperti terlihat pada Tabel 4.28.

Tabel 4. 28 Pengujian Menambahkan Artikel Data Benar

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**

Title: Halo Bandung	*Text box title* artikel terisi sesuai dengan judul artikel yang dimasukkan	Isi dari *text box* judul artikel sesuai dengan komentar yang dimasukkan	Diterima
Content: Ini deskripsi artikel mengenai Bandung	*Text box content* artikel terisi sesuai dengan konten artikel yang dimasukkan	Isi dari *text box* konten artikel sesuai dengan konten yang dimasukkan	Diterima
Tekan tombol *publish*	Ketika menekan tombol *publish* menampilkan artikel baru pada *list box* artikel pengguna	Muncul artikel baru pada *list box* artikel pengguna	Diterima

Sementara itu untuk pengujian menambahkan artikel dengan kasus data salah pada aplikasi *mobile* Bandung Review dipaparkan pada Tabel 4.29.

Tabel 4. 29 Pengujian Menambahkan Artikel Data Salah

Kasus dan Hasil Uji (Data Salah)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Mengkosongkan *text box title* dan *content*	Ketika menekan tombol *publish* menampilkan pesan "*Title and content are required*"	Tampil pesan "*Title and content are required*"	Diterima

4.2.1.2.7 Pengujian Pencarian Tempat

Pengujian pencarian tempat memaparkan pengujian yang dilakukan untuk aktivitas pencarian tempat pada Bandung Review. Pengujian pencarian tempat

terdiri dari dua tipe pengujian, yaitu pengujian dengan kasus data benar dan pengujian dengan kasus data salah. Berikut ini penguijan pencarian tempat dengan kasus data benar seperti terlihat pada Tabel 4.30.

Tabel 4. 30 Pengujian Pencarian Tempat Data Benar

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Keyword: soto	*Text box keyword* terisi sesuai dengan *keyword* yang dimasukkan	Isi dari *text box keyword* sesuai dengan *keyword* yang dimasukkan	Diterima
Category: all	*List picker category* terisi sesuai dengan *category* yang dipilih	Isi dari *list picker category* sesuai dengan *category* yang dipilih	Diterima
	Ketika menekan *enter* pada *keyboard* menampilkan data tempat yang dicari	Tampil data pengguna yang dicari	Diterima

Sementara itu untuk pengujian pencarian tempat dengan kasus data salah pada aplikasi *mobile* Bandung Review dipaparkan pada Tabel 4.31.

Tabel 4. 31 Pengujian Pencarian Tempat Data Salah

Kasus dan Hasil Uji (Data Salah)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Keyword: (Kosong)	Ketika menekan *enter* pada *keyboard* tidak melakukan proses pencarian tempat	Proses pencarian tempat tidak dieksekusi	Diterima

4.2.1.2.8 Pengujian Ubah Data Profil

Pengujian ubah data profil memaparkan pengujian yang dilakukan untuk aktivitas ubah data profil. Pengujian ubah data profil terdiri dari dua tipe pengujian, yaitu pengujian dengan kasus data benar dan pengujian dengan kasus data salah. Berikut ini penguijan ubah data profil dengan kasus data benar seperti terlihat pada Tabel 4.32.

Tabel 4. 32 Pengujian Ubah Data Profil Data Benar

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
First name: Dani	*Text box first name* terisi sesuai dengan *first name* yang dimasukkan	Isi dari *text box first name* sesuai dengan *first name* yang dimasukkan	Diterima
Last name: Taufani	*Text box last name* terisi sesuai dengan *last name* yang dimasukkan	Isi dari *text box last name* sesuai dengan *last name* yang dimasukkan	Diterima
Email: dani@mugi.or.id	*Text box email* terisi sesuai dengan *email* yang dimasukkan	Isi dari *text box email* sesuai dengan *email* yang dimasukkan	Diterima
Gender: Laki-laki	*List picker gender* terisi sesuai dengan *gender* yang dipilih	Isi dari *list picker gender* sesuai dengan *gender* yang	Diterima

		dipilih	
Birthday: 11/11/1901	*Date picker birthday* terisi sesuai dengan *birthday* yang dipilih	Isi dari *date picker birthday* sesuai dengan *birthday* yang dipilih	Diterima
Location: Bandung	*Text box location* terisi sesuai dengan *location* yang dimasukkan	Isi dari *text box location* sesuai dengan *location* yang dimasukkan	Diterima
About me: Mahasiswa Tingkat Akhir Unikom	*Text box about me* terisi sesuai dengan *about me* yang dimasukkan	Isi dari *text box about me* sesuai dengan *about me* yang dimasukkan	Diterima
	Ketika menekan tombol *save* menampilkan pesan "*Your profile has been successfully changed*"	Tampil pesan "*Your profile has been successfully changed*"	Diterima

Sementara itu untuk pengujian ubah data profil dengan kasus data salah pada aplikasi *mobile* Bandung Review dipaparkan pada Tabel 4.33.

Tabel 4. 33 Pengujian Ubah Data Profil Data Salah

Kasus dan Hasil Uji (Data Salah)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Mengkosongkan *text box first name, gender* dan	Ketika menekan tombol *save* menampilkan pesan	Tampil pesan "*First name, gender and*	Diterima

birthday	"*First name, gender and date of birth are required*"	*date of birth are required*"	
Mengisi *location* dengan jumlah karakter kurang dari 6.	Ketika menekan tombol *save* menampilkan pesan "*Location must contain at least 6 letters*"	Tampil pesan "*Location must contain at least 6 letters*"	Diterima
Mengisi *email* yang dengan format tidak valid. Contoh *Email*: dani.taufani.com	Ketika menekan tombol *save* menampilkan pesan "*Email is not in a valid format*"	Tampil pesan "*Email is not in a valid format*"	Diterima
Mengisi *email* yang sudah digunakan	Ketika menekan tombol *save* menampilkan pesan "*Email already registered*"	Tampil pesan "*Email already registered*"	Diterima

4.2.1.2.9 Pengujian Ubah Data *Password*

Pengujian ubah data *password* memaparkan pengujian yang dilakukan untuk aktivitas ubah data *password*. Pengujian ubah data *password* terdiri dari dua tipe pengujian, yaitu pengujian dengan kasus data benar dan pengujian dengan kasus data salah. Berikut ini penguijan ubah data *password* dengan kasus data benar seperti terlihat pada Tabel 4.34.

Tabel 4. 34 Pengujian Ubah Data *Password* Data Benar

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Old password: 123456	*Text box old password* terisi sesuai dengan *old password* yang dimasukkan	Isi dari *text box old password* sesuai dengan *old password* yang dimasukkan	Diterima
New password: 654321	*Text box new password* terisi sesuai dengan *new password* yang dimasukkan	Isi dari *text box new password* sesuai dengan *new password* yang dimasukkan	Diterima
Confirm password: 654321	*Text box confirm password* terisi sesuai dengan *confirm password* yang dimasukkan	Isi dari *text box confirm password* sesuai dengan *confirm password* yang dimasukkan	Diterima
	Ketika menekan tombol *save* menampilkan pesan "*Password successfully changed*"	Tampil pesan "*Password successfully changed*"	Diterima

Sementara itu untuk pengujian memberi ubah data *password* dengan kasus data salah pada aplikasi *mobile* Bandung Review dipaparkan pada Tabel 4.35.

Tabel 4. 35 Pengujian Ubah Data *Password* Data Salah

Kasus dan Hasil Uji (Data Salah)

Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Mengkosongkan *text box old password, new password* dan *confirm password*	Ketika menekan tombol *save* menampilkan pesan "*All fields are required*"	Tampil pesan "*All fields are required*"	Diterima
Mengisi *new password* dengan jumlah karakter kurang dari 6. Contoh *Password*: 123	Ketika menekan tombol *save* menampilkan pesan "*New password must contain at least 6 letters*"	Tampil pesan "*New password must contain at least 6 letters*"	Diterima
Mengisi konfirmasi *password* yang berbeda dengan *password*, Contoh *Password*: 123456 *Confirm Pasword*: 654321	Ketika menekan tombol *sign up* menampilkan pesan "*Password must match!*"	Tampil pesan "*Password must match!*"	Diterima
Mengisi *old password* yang berbeda	Ketika menekan tombol *sign up* menampilkan pesan "*ERROR: Current password doesn't match!*"	Tampil pesan "*ERROR: Current password doesn't match!!*"	Diterima

4.2.1.2.10 Pengujian Ubah Data Artikel

Pengujian ubah data artikel memaparkan pengujian yang dilakukan untuk aktivitas ubah data artikel. Pengujian ubah data artikel terdiri dari dua tipe pengujian, yaitu pengujian dengan kasus data benar dan pengujian dengan kasus data salah. Berikut ini penguijan ubah data artikel dengan kasus data benar seperti terlihat pada Tabel 4.36.

Tabel 4. 36 Pengujian Ubah DataArtikel Data Benar

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Title: Halo-halo Bandung	*Text box title* artikel terisi sesuai dengan judul artikel yang dimasukkan	Isi dari *text box* judul artikel sesuai dengan komentar yang dimasukkan	Diterima
Content: Ini deskripsi artikel mengenai halo-halo Bandung	*Text box content* artikel terisi sesuai dengan konten artikel yang dimasukkan	Isi dari *text box* konten artikel sesuai dengan konten yang dimasukkan	Diterima
Tekan tombol *publish*	Ketika menekan tombol *publish* menampilkan artikel baru pada *list box* artikel pengguna	Muncul artikel baru pada *list box* artikel pengguna	Diterima

Sementara itu untuk pengujian memberi ubah data artikel dengan kasus data salah pada aplikasi *mobile* Bandung Review dipaparkan pada Tabel 4.37.

Tabel 4. 37 Pengujian Ubah Data Artikel Data Salah

Kasus dan Hasil Uji (Data Salah)

Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Mengkosongkan *text box title* dan *content*	Ketika menekan tombol *publish* menampilkan pesan "*Title and content are required*"	Tampil pesan "*Title and content are required*"	Diterima

4.2.1.2.11 Pengujian *Setting*

Pengujian *setting* memaparkan pengujian yang dilakukan terhadap aktivitas pengguna saat melakukan *setting* pada aplikasi *mobile* Bandung Review. Pengujian *setting* menggunakan pengujian dengan kasus data benar. Berikut ini penguijan *setting* dengan kasus data benar yang dapat dilihat pada Tabel 4.38.

Tabel 4. 38 Pengujian *Setting* Data Benar

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Toggle Location service On	Ketika *toggle location service* dalam kondisi on maka lokasi pengguna akan dikirimkan ke *web service*	Titik lokasi pengguna akan dikirimkan ke *web service*	Diterima
Toggle Location service Off	Ketika *toggle location service* dalam kondisi off maka lokasi pengguna tidak akan dikirimkan ke web service dan akan	Titik lokasi pengguna akan dikirimkan ke *web service* dan menampilkan pesan "*location services are*	Diterima

	menampilkan pesan "*location services are turned off*"	*turned off*"	

3.2.4.3 Kesimpulan Pengujian *Black Box*

Berdasarkan hasil pengujian aplikasi yang dilakukan maka dapat disimpulkan bahwa aplikasi *mobile* Bandung Review berbasis Windows Phone yang dibangun sudah berjalan sesuai dengan yang diharapkan baik itu dari segi validasi maupun proses penanganan kesalahan.

4. 2. 2 Pengujian Performansi

Pengujian performansi merupakan pengujian untuk mendapatkan tingkat performa dari sebuah metode yang digunakan pada perangkat lunaknya. Pengujian perfomansi ini bertujuan untuk menguji penggunaan metode *haversine* dalam menentukan jarak suatu tempat dari lokasi pengguna. Pengujian performansi yang dilakukan pada aplikasi *mobile* Bandung Review dibagi menjadi tiga bagian, yaitu skenario pengujian, kasus dan hasil pengujian, dan kesimpulan pengujian

4.2.2.1 Skenario Pengujian

Skenario pengujian memaparkan urutan dan hal yang diuji pengujian yang dilakukan pada aplikasi *mobile* Bandung Review. Adapun skenario pengujian dilakukan pada 3 buah sample data, dapat dilihat pada Tabel 4.39.

Tabel 4. 39 Skenario Pengujian Performansi

Nama Tempat	Alamat	*Latitude*	*Longitude*
Episode FO	Jl. Ir. H. Juanda No. 125	-6.886698	107.613616

Sasana Budaya Ganesha (Sabuga)	Jl. Tamansari No. 73	-6.890746	107.607640
Mc Donald's (Simpang Dago)	Jl. Ir. H. Juanda No.181	-6.884951	107.614302

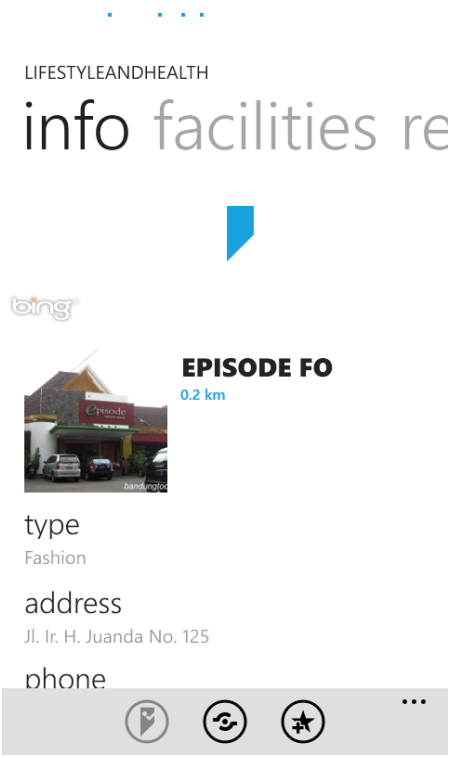
4.2.2.2 Kasus dan Hasil Pengujian

Kasus dan hasil pengujian berisi pemaparan dari rencana pengujian yang telah disusun pada skenario pengujian. Adapun detail dari kasus dan hasil pengujian performansi dapat dilihat pada Tabel 4.40.

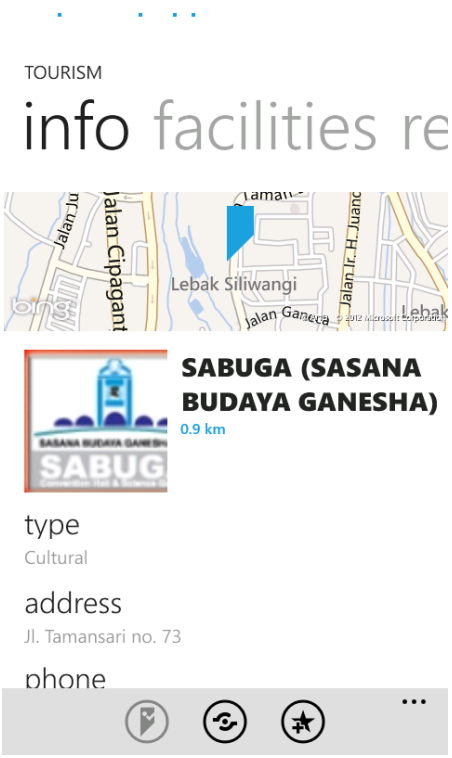
Tabel 4. 40 Hasil Pengujian Performansi

Nama Tempat	Titik Koordinat Pengguna		Titik Koordinat Tempat Tujuan		Jarak (km)	Waktu Akses (ms)
	Lat	Long	Lat	Long		
Episode FO	-6.886309	107.613639	-6.886698	107.613616	0.2	852
Sasana Budaya Ganesha (Sabuga)			-6.890746	107.607640	0.9	985
Mc Donald's (Simpang Dago)			-6.884951	107.614302	0.3	796

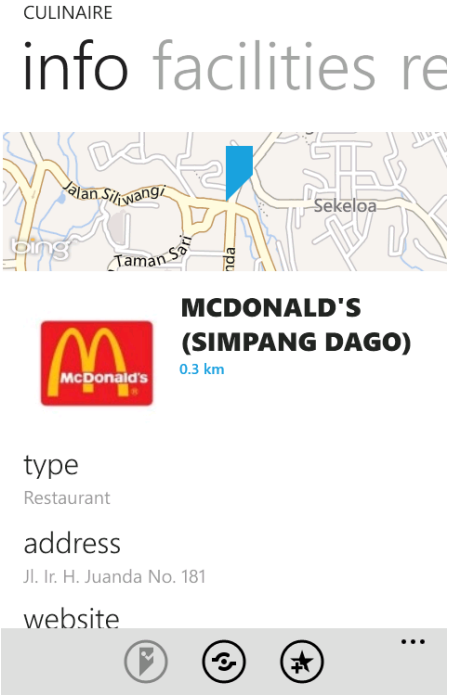
Adapun gambar dari tampilan hasil pengujian performansi yang telah dilakukan seperti terlihat pada Gambar 4.1 sampai dengan Gambar 4.3.



Gambar 4. 1 Tampilan Episode FO



Gambar 4. 2 Tampilan Sabuga



Gambar 4. 3 Tampilan McDonald's (Simpang Dago)

4.2.2.3 Kesimpulan Pengujian Performansi

Berdasarkan hasil pengujian performansi kalkulasi jarak yang dilakukan maka dapat disimpulkan bahwa aplikasi *mobile* Bandung Review berbasis Windows Phone yang dibangun sudah memenuhi untuk membantu pengguna dalam memperoleh jarak ke suatu tempat dan menentukan tempat-tempat terdekat.

4. 2. 3 Pengujian Beta

Pengujian beta dilakukan di lingkungan pengguna tanpa kehadiran pihak pembangun aplikasi. Pengujian ini merupakan pengujian yang bersifat langsung di lingkungan yang sebenarnya. Pengguna melakukan penilaian terhadap aplikasi dengan menggunakan media kuesioner. Dari hasil kuesioner tersebut maka dapat ditarik kesimpulan apakah aplikasi yang dibangun telah sesuai dengan tujuan atau tidak.

4.2.3.1 Kuesioner Pengujian Beta

Kuesioner pengujian beta merupakan media yang digunakan pengguna Bandung Review untuk memberikan penilaian terhadap aplikasi yang dibangun. kuesioner ini disebarkan menggunakan teknik sampling yaitu *Simple Random Sampling* yang disebarkan kepada banyak pengguna. Dari hasil kuesioner tersebut akan dilakukan perhitungan agar dapat diambil kesimpulan terhadap penilaian penerapan aplikasi yang dibangun. Kuisioner ini terdiri dari 5 pertanyaan dengan menggunakan skala *likert* dari skala 1 sampai 5. Adapun pertanyaan dari kuesioner pengujian beta adalah sebagai berikut:

1. Apakah fasilitas melihat tempat-tempat wisata dengan mengimplementasikan teknologi *Global Positioning System* (GPS) yang telah dibuat sudah baik?

A. Sangat Baik
B. Baik
C. Cukup Baik
D. Kurang Baik
E. Tidak Baik

2. Apakah secara umum aplikasi *mobile* Bandung Review berbasis Windows Phone ini telah mewakili fitur-fitur yang terdapat pada BandungReview.com?

A. Sangat Mewakili
B. Mewakili
C. Cukup Mewakili
D. Kurang Mewakili
E. Tidak Mewakili

3. Apakah aplikasi *mobile* Bandung Review berbasis Windows Phone ini membantu Anda dalam mengakses BandungReview.com?

A. Sangat Membantu
B. Membantu
C. Cukup Membantu
D. Kurang Membantu
E. Tidak Membantu

4. Apakah antarmuka aplikasi *mobile* Bandung Review berbasis Windows Phone ini mudah dimengerti?

A. Sangat Mudah
B. Mudah
C. Cukup Mudah
D. Kurang Mudah
E. Tidak Mudah

5. Apakah antarmuka aplikasi *mobile* Bandung Review berbasis Windows Phone ini mudah digunakan?

A. Sangat Mudah
B. Mudah
C. Cukup Mudah
D. Kurang Mudah
E. Tidak Mudah

Berdasarkan data hasil kuisioner tersebut, dapat dicari presentasi masing-masing jawaban dengan menggunakan rumus : Y = P/Q * 100%

Keterangan :

P = Banyaknya jawaban responden tiap soal.

Q = Jumlah responden

Y = Nilai presentasi

Berikut ini adalah hasil presentasi masing-masing nilai jawaban kuisioner yang diujikan kepada 30 responden dan telah dihitung menggunakan rumus diatas.

1. Apakah fasilitas melihat tempat-tempat wisata dengan mengimplementasikan teknologi *Global Positioning System* (GPS) yang telah dibuat sudah baik?

Tabel 4. 41 Hasil Pengujian Kuisioner Soal Nomor 1

Pertanyaan	No.	Keterangan	Responden	Persentasi (%)
1	1	Sangat Baik	4	13,33
	2	Baik	23	76,67
	3	Cukup Baik	3	10
	4	Kurang Baik	0	0
	5	Tidak Baik	0	0
Jumlah			**30**	**100**

Berdasarkan hasil persentasi pada Tabel 4.41 maka dapat disimpulkan bahwa sebanyak 4 orang atau 13,33% mengatakan bahwa fasilitas melihat tempat-tempat wisata dengan mengimplementasikan teknologi GPS sangat baik, 23 orang atau 76,67% mengatakan bahwa fasilitas melihat tempat-tempat wisata dengan mengimplementasikan teknologi GPS baik dan 3 orang atau 10% mengatakan

bahwa fasilitas melihat tempat-tempat wisata dengan mengimplementasikan teknologi GPS cukup baik. Sehingga dapat disimpulkan bahwa aplikasi *mobile* Bandung Review telah memiliki fasilitas melihat tempat-tempat wisata dengan mengimplementasikan teknologi GPS dengan baik.

2. Apakah secara umum aplikasi *mobile* Bandung Review berbasis Windows Phone ini telah mewakili fitur-fitur yang terdapat pada BandungReview.com?

Tabel 4. 42 Hasil Pengujian Kuisioner Soal Nomor 2

Pertanyaan	No.	Keterangan	Responden	Persentasi (%)
2	1	Sangat Mewakili	5	16,67
	2	Mewakili	18	60
	3	Cukup Mewakili	7	23,33
	4	Kurang Mewakili	0	0
	5	Tidak Mewakili	0	0
Jumlah			**30**	**100**

Berdasarkan hasil persentasi pada Tabel 4.42 maka dapat disimpulkan bahwa sebanyak 5 orang atau 16,67% mengatakan bahwa aplikasi *mobile* Bandung Review berbasis Windows Phone sangat mewakili fitur-fitur yang terdapat pada BandungReview.com, 18 orang atau 60% mengatakan bahwa aplikasi *mobile* Bandung Review berbasis Windows Phone mewakili fitur-fitur yang terdapat pada BandungReview.com dan 7 orang atau 23,33% mengatakan bahwa aplikasi *mobile* Bandung Review berbasis Windows Phone cukup mewakili fitur-fitur yang terdapat pada BandungReview.com. Sehingga dapat disimpulkan bahwa aplikasi *mobile* Bandung Review telah mewakili fitur-fitur yang terdapat pada BandungReview.com.

3. Apakah aplikasi *mobile* Bandung Review berbasis Windows Phone ini membantu Anda dalam mengakses BandungReview.com?

Tabel 4. 43 Hasil Pengujian Kuisioner Soal Nomor 3

Pertanyaan	No.	Keterangan	Responden	Persentasi (%)
3	1	Sangat Membantu	7	23,33
	2	Membantu	17	56,67
	3	Cukup Membantu	6	20
	4	Kurang Membantu	0	0
	5	Tidak Membantu	0	0
Jumlah			**30**	**100**

Berdasarkan hasil persentasi pada Tabel 4.43 maka dapat disimpulkan bahwa sebanyak 7 orang atau 23,33% mengatakan bahwa aplikasi *mobile* Bandung Review berbasis Windows Phone sangat membantu dalam mengakses BandungReview.com, 17 orang atau 56,67% mengatakan bahwa aplikasi *mobile* Bandung Review berbasis Windows Phone membantu dalam mengakses BandungReview.com dan 6 orang atau 20% mengatakan bahwa aplikasi *mobile* Bandung Review berbasis Windows Phone cukup membantu dalam mengakses BandungReview.com. Sehingga dapat disimpulkan bahwa aplikasi *mobile* Bandung Review membantu dalam pengaksesan BandungReview.com dari perangkat *mobile*.

4. Apakah antarmuka aplikasi *mobile* Bandung Review berbasis Windows Phone ini mudah dimengerti?

Tabel 4. 44 Hasil Pengujian Kuisioner Soal Nomor 4

Pertanyaan	No.	Keterangan	Responden	Persentasi (%)
4	1	Sangat Mudah	4	13,33

	2	Mudah	17	56,67
	3	Cukup Mudah	7	23,33
	4	Kurang Mudah	2	6.67
	5	Tidak Mudah	0	0
Jumlah			**30**	**100**

Berdasarkan hasil persentasi pada Tabel 4.44 maka dapat disimpulkan bahwa sebanyak 4 orang atau 13,33% mengatakan bahwa sangat mudah dimengerti antarmuka aplikasi *mobile* Bandung Review berbasis Windows Phone, 17 orang atau 56,67% mengatakan bahwa antarmuka aplikasi *mobile* Bandung Review berbasis Windows Phone mudah untuk dimengerti, 7 orang atau 23,33% mengatakan bahwa antarmuka aplikasi *mobile* Bandung Review berbasis Windows Phone cukup mudah untuk dimengerti dan 2 orang atau 6,67% mengatakan bahwa antarmuka aplikasi *mobile* Bandung Review berbasis Windows Phone kurang mudah untuk dimengerti. Sehingga dapat disimpulkan bahwa aplikasi *mobile* Bandung Review mempunyai antarmuka yang mudah dimengerti oleh pengguna.

5. Apakah antarmuka aplikasi *mobile* Bandung Review berbasis Windows Phone ini mudah digunakan?

Tabel 4. 45 Hasil Pengujian Kuisioner Soal Nomor 5

Pertanyaan	No.	Keterangan	Responden	Persentasi (%)
5	1	Sangat Mudah	4	13,33
	2	Mudah	17	56,67
	3	Cukup Mudah	9	30
	4	Kurang Mudah	0	0

	5	Tidak Mudah	0	0
Jumlah			**30**	**100**

Berdasarkan hasil persentasi pada Tabel 4.45 maka dapat disimpulkan bahwa sebanyak 4 orang atau 13,33% mengatakan bahwa sangat mudah menggunakan antarmuka aplikasi *mobile* Bandung Review berbasis Windows Phone, 17 orang atau 56,67% mengatakan bahwa antarmuka aplikasi *mobile* Bandung Review berbasis Windows Phone mudah untuk digunakan dan 9 orang atau 30% mengatakan bahwa antarmuka aplikasi *mobile* Bandung Review berbasis Windows Phone cukup mudah untuk digunakan. Sehingga dapat disimpulkan bahwa aplikasi *mobile* Bandung Review mempunyai antarmuka yang mudah digunakan.

4.2.3.2 Kesimpulan Pengujian Beta

Berdasarkan hasil persentasi hasil perhitungan pengujian beta pengguna aplikasi *mobile* Bandung Review maka dapat disimpulkan bahwa pembangunan aplikasi *mobile* Bandung Review berbasis Windows Phone ini sudah sesuai dengan tujuan, yaitu memberikan fitur-fitur utama yang terdapat pada Bandung Review.