BAB 4
IMPLEMENTASI DAN PENGUJIAN

4.1 Implementasi Sistem

Tahapan implementasi merupakan tahap penciptaan perangkat lunak, tahap kelanjutan dari kegiatan perancangan sistem. Tahap ini merupakan tahap dimana sistem siap untuk dioperasikan yang terdiri dari penjelasan mengenai lingkungan implementasi dan implementasi program.

4.1.1 Implementasi Perangkat Keras

Spesifikasi perangkat keras minimal yang dibutuhkan untuk mengimplementasikan perangkat keras antara lain terdapat pada tabel sebagai berikut.

Tabel 4.1 Implementasi Perangkat Keras *Back end*

No	Perangkat Keras	Spesifikasi
1.	Processor	Intel Pentium dual core 2 GHz
2.	Monitor	Monitor17”
3.	Hardisk	250 GB
4.	Memori	2 GB
5.	Keyboard dan Mouse	Standar

Tabel 4.2 Implementasi Perangkat Keras *Front end*

No	Perangkat Keras	Spesifikasi
1.	Processor	800Mhz
2.	Monitor	320x240 px
3.	Memori	512Mb Internal
5.	Jaringan	HSDPA 3.6Mbps

4.1.2 Implementasi Perangkat Lunak

Spesifikasi perangkat lunak yang dibutuhkan untuk mengimplementasikan perangkat lunak antara lain sebagai berikut.

Tabel 4.3 Implementasi Perangkat Lunak *Back end*

No	Perangkat Lunak	Keterangan
1	Sistem Operasi	Windows XP
2	*Web Browser*	Google Chrome
3	*Code Editor*	Dreamweaver CS3
4	DBMS	Mysql 5.1
5	*Web Server*	Wamp server 2.0
6	Analisis pemodelan	Star UML

Tabel 4.4 Implementasi Perangkat Lunak *Front end*

No	Perangkat Lunak	Keterangan
1	Sistem Operasi	Android 4.0.4 ICS
2	*Code Editor*	ADT Eclipse bundle for windows
3	Analisis pemodelan	Star UML

4.1.3 Implementasi Basis Data

Implementasi basis data sistem peringatan dini yang dibangun menggunakan DBMS MySQL. Adapun implementasi basis data yang dibangun adalah sebagai berikut :

Tabel 4.5 Implementasi *Database*

No	**Nama Tabel**	**Perintah SQL**
1	user	CREATE TABLE IF NOT EXISTS `user` (`user_id` int(11) NOT NULL AUTO_INCREMENT, `user_nama` varchar(100) NOT NULL, `user_username` varchar(100) NOT NULL, `user_password` varchar(100) NOT NULL, `email` varchar(200) NOT NULL, `user_level` varchar(8) NOT NULL DEFAULT 'operator', PRIMARY KEY (`user_id`), UNIQUE KEY `user_nama` (`user_nama`), KEY `user_id` (`user_id`), KEY `user_level` (`user_level`)

		) ENGINE=InnoDB DEFAULT CHARSET=latin1 AUTO_INCREMENT=0 ;
2	gunung	CREATE TABLE IF NOT EXISTS `gunung` (`id_gunung` int(8) NOT NULL AUTO_INCREMENT, `user_id` int(4) DEFAULT NULL, `nama_gunung` varchar(35) NOT NULL, `deskripsi_gunung` text, `alamat` varchar(50) NOT NULL, `lat` decimal(65,6) NOT NULL, `long` decimal(65,6) NOT NULL, `status_gunung` varchar(10) NOT NULL, `pesan_rekomendasi` text, PRIMARY KEY (`id_gunung`), UNIQUE KEY `nama_gunung` (`nama_gunung`), KEY `id_gunung` (`id_gunung`), KEY `status_gunung` (`status_gunung`), KEY `user_id` (`user_id`)) ENGINE=InnoDB DEFAULT CHARSET=latin1 AUTO_INCREMENT=0 ;
3	polygon	CREATE TABLE IF NOT EXISTS `polygon` (`id_polygon_krb` int(5) unsigned NOT NULL AUTO_INCREMENT, `id_gunung` int(8) DEFAULT NULL, `koordinat` polygon NOT NULL, PRIMARY KEY (`id_polygon_krb`), KEY `id_gunung` (`id_gunung`)) ENGINE=InnoDB DEFAULT CHARSET=latin1 AUTO_INCREMENT=0;
4	gcm_users	CREATE TABLE IF NOT EXISTS `gcm_users` (`id` int(11) NOT NULL AUTO_INCREMENT, `gcm_regid` text, `name` varchar(50) NOT NULL, `email` varchar(255) NOT NULL, `created_at` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP, PRIMARY KEY (`id`)) ENGINE=InnoDB DEFAULT CHARSET=latin1 AUTO_INCREMENT=0 ;
5	menu	CREATE TABLE IF NOT EXISTS `menu` (`menu_id` int(11) NOT NULL AUTO_INCREMENT, `menu_nama` varchar(100) NOT NULL, `menu_uri` varchar(100) NOT NULL, `menu_allowed` varchar(100) NOT

		NULL, PRIMARY KEY (`menu_id`), KEY `menu_id` (`menu_id`)) ENGINE=InnoDB DEFAULT CHARSET=utf8 AUTO_INCREMENT=0 ;
6	Constraints for table `gunung`	ALTER TABLE `gunung` ADD CONSTRAINT `gunung_ibfk_4` FOREIGN KEY (`user_id`) REFERENCES `user` (`user_id`);
7	Constraints for table `polygon`	ALTER TABLE `polygon` ADD CONSTRAINT `polygon_ibfk_1` FOREIGN KEY (`id_gunung`) REFERENCES `gunung` (`id_gunung`);

Tabel 4.6 Implementasi Data Aplikasi *Front end*

No	Nama Tabel	Perintah SQL

1	table_gunung	onCreate(SQLiteDatabase db) {
		String CREATE_GUNUNG_TABLE =
		"CREATE TABLE " + *TABLE_GUNUNG* + "("
		+ *KEY_ID* + " INTEGER PRIMARY KEY," + *KEY_SQL* + " TEXT,"
		+ " TEXT" + ")";
		db.execSQL(CREATE_GUNUNG_TABLE);
		}

4.1.4 Implementasi Antarmuka

Implementasi antarmuka merupakan tampilan dari aplikasi yang dibangun. Aplikasi pengolahan data proses memilki beberapa bagian tampilan menu dapat dilihat pada tabel berikut :

1. Implementasi Antarmuka *Back end*

Tabel 4.7 Implementasi Antarmuka *Back end*

Menu	Deskripsi	Nama File
Login	Digunakan Sebagai Halaman Login Admin dan User(Operator)	login_form.php
Lupa *Password*	Digunakan Jika Admin atau operator lupa password atau username	lupa_password.php

Tampilkan Daftar *User*	Digunakan untuk menampilkan semua daftar operator gunung yang terdaftar	tabel_user.php
Tambah *User*	Digunakan untuk menambahkan operator gunung	form_input_user.php
Edit *User*	Digunakan untuk melakukan edit user	form_edit_user.php
Tampilkan Daftar Gunung	Menampilkan daftar seluruh gunung yang terdaftar	tabel_gunung.php
Tambah Gunung	Menampilkan form tambah gunung	form_input_gunung.php
Edit Gunung	Menampilkan Form Edit gunung	form_edit_gunung.php
Input KRB	Menampilkan form input KRB	form_input_krb_gunung.php
Edit KRB	Menampilkan Edit KRB	edit_krb_gunung.php

2. Implementasi Antarmuka *Front end*

Tabel 4.8 Implementasi Antarmuka *Front end*

Menu	Deskripsi	Nama File
Tampilkan Info Gunung	Digunakan untuk menampilkan daftar gunung	list_gunung.xml
Pilih gunung yang dipantau	Digunakan untuk form pilih gunung yang dipantau	gunung_add_layout.xml
Tampilkan detail gunung	Digunakan untuk menampilkan detail informasi gunung	gunung_detail.xml
Registrasi	Digunakan untuk menampilkan form registrasi pengguna	gcm_activity_register.xml

4.1 Pengujian Sistem

Pengujian sistem yang dibangun menggunakan metode pengujian *black box*. Pengujian *black box* fokus pada persyaratan fungsional perangkat lunak.

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 4.9 Rencana Pengujian *Back end*

Pengujian Aplikasi Backend		
Item Uji	Detail Uji	Jenis Uji
Login	Menampilkan form login	*Black box*
	Mengisi form login	
	Verifikasi username	
	Verifikasi password	
	Menampilkan halaman kelola informasi	
Tambah info gunung	Menampilkan form tambah gunung	*Black box*
	Mengisi form tambah gunung	
	Verifikasi data masukan	
	Memasukan data ke dalam database	
Ubah info gunung	Menampilkan form edit data gunung	*Black box*
	Mengubah data gunung	
	Verifikasi masukan	
	Menyimpan data kedalam database	
Hapus info gunung	Menampilkan pesan konfirmasi	*Black box*
	Menyetujui pesan konfirmasi	
	Menghapus data dalam database	
Tambah *user* (operator)	Menampilkan form tambah user	*Black box*
	Mengisi form tambah user	
	Verifikasi data masukan	
	Memasukan data kedalam database	
Ubah *user*	Menampilkan form edit user	*Black box*
	Mengubah data user	
	Verifikasi masukan	
	Menyimpan data kedalam database	
Hapus *user*	Menampilkan pesan konfirmasi	*Black box*
	Menyetujui pesan konfirmasi	
	Menghapus data dalam database	
Tambah Kawasan Rawan Bencana (KRB)	Tampilkan peta	*Black box*
	Mengisi data KRB	
	Memasukan data kedalam database	
Ubah KRB	Tampilakan form edit	*Black box*
	Mengubah data KRB	

	Menyimpan data kedalam database	
Hapus KRB	Menampilkan pesan konfirmasi	*Black box*
	Menyetujui pesan konfirmasi	
	Menghapus data dalam database	
Lupa password	Menampilkan form input email	*Black box*
	Mengirim pesan konfirmasi	
	Reset Password	
	Menampilkan Password baru	

Tabel 4.10 Rencana Pengujian Frontend

Pengujian Aplikasi Frontend		
Item Uji	Detail Uji	Jenis Uji
Pilih gunung yang dipantau	Tampilkan form registrasi pengguna	Black box
	Mangisi form registrasi	
	Memasukan data kedalam database	
Tampilkan Info gunung	Nama gunung	Black box
	Status gunung	
Tampilkan detail informasi	Status gunung	Black box
	Deskripsi gunung	
	Pesan Rekomendasi	
Tampilkan peta Kawasan Rawan Bencana	Tampilkan peta	Black box
	Tampilkan letak geografis gunung	
	Tampilkan lokasi pengguna	
	Tampilkan kawasan rawan bencana	
Registrasi pengguna	Tampilkan form registrasi	Black box
	Menyimpan data	
Tampilkan Notifikasi	Tampilkan Notifikasi Status gunung	Black box

4.2.2 Pengujian Black Box

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

1. Pengujian *Login*

Tabel 4.11 Pengujian *Login*

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	Data yang diharapkan	Pengamatan	Kesimpulan
Masukan data login, contoh	Mengisi data login dan field yang	Dapat mengisi data login dan	[√] Diterima [ ] Ditolak

masukan : Username : admin Password : admin	tersedia dan berhasil masuk kedalam halaman kelola informasi	dapat masuk ke halaman kelola informasi	
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	Data yang diharapkan	Pengamatan	Kesimpulan
Username : wqeqe Password : dfvsdf	Tampil pesan kesalahan "Maaf, username dan password salah!"	Dapat menampilkan pesan kesalahan "Maaf, username dan password salah!"	[√] Diterima [ ] Ditolak

2. Pengujian Tambah Info Gunung

Tabel 4.12 Pengujian Tambah Gunung

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	Data yang diharapkan	Pengamatan	Kesimpulan
Masukan data gunug, contoh masukan : Nama gunung : Tangkuban parahu Alamat : Lembang, Kab. Bandung Latitude : -7.322159 Longitude : 107.728407 Deskripsi: Gunungapi tipe A Pesan Rekomendasi:	Mengisi data gunung dan field yang tersedia dan berhasil menyimpan kedalam database, serta menampilkan pesan "sukses, data sudah tersimpan !"	Dapat menyimpan kedalam database, dan menampilkan pesan : "sukses, data sudah tersimpan !"	[√] Diterima [ ] Ditolak

Waspada gas racun jika berada di puncak gunung Status: normal			
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	Data yang diharapkan	Pengamatan	Kesimpulan
Nama gunung : (kosong) Alamat : Lembang, Kab. Bandung Latitude : -7.322159 Longitude : 107.728407 Deskripsi: Gunungapi tipe A Pesan Rekomendasi: Waspada gas racun jika berada di puncak gunung Status: normal	Tampil pesan kesalahan: "Field tidak boleh kosong" dan data tidak ditambahkan kedalam database	Dapat menampilkan pesan kesalahan: "Nama gunung tidak boleh kosong" dan data tidak ditambahkan kedalam database	[√] Diterima [ ] Ditolak

3. Pengujian Ubah Info Gunung

Tabel 4.13 Pengujian Ubah Info

Kasus dan Hasil Uji			
Data Masukan	Data yang diharapkan	Pengamatan	Kesimpulan
Masukan data gunug, contoh masukan :	Mengubah data berdasarkan field yang tersedia,	Dapat mengisi form ubah data gunung lalu dapat	[√] Diterima [ ] Ditolak

Pesan rekomendasi : Dihimbau untuk tidak melakukan aktivitas di sekitar kawah Status : waspada	berhasil memasukan ke dalam database dan penampilkan pesan "Sukses, data berhasil diubah!"	mengubah data dalam database dan menampilkan pesan "Sukses, data berhasil diubah!"	

4. Pengujian Hapus Gunung

Tabel 4.14 Pengujian Hapus Info

Kasus dan Hasil Uji			
Data Masukan	Data yang diharapkan	Pengamatan	Kesimpulan
Memilih data yang akan dihapus	Menampilkan pesan konfirmasi hapus, menghapus data dalam database dan penampilkan pesan "**sukses, data** berhasil dihapus !"	Dapat menampilkan konfirmasi hapus dan menampilkan pesan "**sukses, data** berhasil dihapus !"	[√] Diterima [ ] Ditolak

5. Pengujian Input Kawasan Rawan Bencana

Tabel 4.15 Pengujian Input KRB

Kasus dan Hasil Uji			
Data Masukan	Data yang diharapkan	Pengamatan	Kesimpulan
Mengisi titik koordinat kawasan rawan bencan gunungapi pada peta yang ditampilkan	Membuat polygon pada peta yang ditampilkan, Menyimpan koordinat kedalam database dan menampilkan pesan "Sukses, KRB sudah disimpan!"	Dapat mengisi koordinat polygon pada peta yang ditampilkan, Menyimpan koordinat kedalam database dan menampilkan	[√] Diterima [ ] Ditolak

		pesan “Sukses, KRB sudah disimpan!”	
Kasus dan Hasil Uji			
Data Masukan	Data yang diharapkan	Pengamatan	Kesimpulan
Input Koordinat : (kosong)	Tidak dapat menyimpan data kedalam database	Button simpan tidak berfungsi	[√] Diterima [ ] Ditolak

6. Pengujian Edit Kawasan Rawan Bencana

Tabel 4.16 Pengujian Ubah KRB

Kasus dan Hasil Uji			
Data Masukan	Data yang diharapkan	Pengamatan	Kesimpulan
Mengubah titik koordinat kawasan rawan bencana gunungapi dengan cara menggeser titik koordinat.	Mengubah polygon dengan mengubah koordinat yang dimasukan pada peta dan berhasil memasukan ke dalam database serta menampilkan pesan “Sukses, KRB berhasil diubah!”	Dapat mengubah polygon dengan mengubah koordinat yang dimasukan pada peta dan berhasil memasukan ke dalam database serta menampilkan pesan “Sukses, KRB berhasil diubah!”	[√] Diterima [ ] Ditolak

7. Pengujian Hapus Kawasan Rawan Bencana

Tabel 4.17 Pengujian Hapus KRB

Kasus dan Hasil Uji		
Data yang diharapkan	Pengamatan	Kesimpulan
Menampilkan pesan konfirmasi hapus dan menghapus koordinat polygon dalam database	Dapat Menampilkan pesan konfirmasi hapus dan menghapus data dalam database	[√] Diterima [ ] Ditolak

8. Pengujian Tambah User

Tabel 4.18 Pengujian Tambah Operator

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	Data yang diharapkan	Pengamatan	Kesimpulan
Masukan data user, contoh masukan : Nama gunung : pos parahu username : parahu password : parahu gunung yang dipantau : tangkuban parahu	Mengisi data user dan field yang tersedia dan berhasil menyimpan kedalam database, serta menampilkan pesan "sukses, data sudah tersimpan !"	Dapat mengisi data user dan tersimpan dalam database	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	Data yang diharapkan	Pengamatan	Kesimpulan
Nama gunung : pos parahu username : parahu password : parahu gunung yang dipantau : tangkuban parahu	Tampil pesan kesalahan "field tidak boleh kosong" dan data tidak ditambahkan kedalam database	Dapat menampilkan pesan kesalahan "field tidak boleh kosong" Dan data tidak ditambahkan kedalam database	[√] Diterima [ ] Ditolak

9. Pengujian Ubah *User*

Tabel 4.19 Pengujian Ubah Operator

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	Data yang diharapkan	Pengamatan	Kesimpulan
Masukan data gunug, contoh masukan :	Mengubah data akun berdasarkan field yang tersedia, berhasil memasukan	Dapat mengisi form ubah data gunung lalu dapat mengubah	[√] Diterima [ ] Ditolak

Pesan Rekomendasi: Awas Status:	ke dalam database dan penampilkan pesan "Sukses, data berhasil diubah!"	data dalam database dan menampilkan pesan "Sukses, data berhasil diubah!"	
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	Data yang diharapkan	Pengamatan	Kesimpulan
(kosong)	Tampil pesan kesalahan "field tidak boleh kosong" dan data dalam database tidak berubah	Dapat menampilkan pesan kesalahan "field tidak boleh kosong" dan data dalam database tidak berubah	[√] Diterima [ ] Ditolak

10. Pengujian Hapus *User*

Tabel 4.20 Pengujian Hapus Operator

Kasus dan Hasil Uji			
Data Masukan	Data yang diharapkan	Pengamatan	Kesimpulan
Memilih data yang akan dihapus	Menampilkan pesan konfirmasi hapus, menghapus data dalam database dan penampilkan pesan "**sukses, data** berhasil dihapus !"	Dapat menampilkan konfirmasi hapus dan menampilkan pesan "**sukses, data** berhasil dihapus !"	[√] Diterima [ ] Ditolak

11. Pengujian Pilih Gunung

Tabel 4.21 Pengujian Pilih Gunung

Kasus dan Hasil Uji			
Data Masukan	Data yang diharapkan	Pengamatan	Kesimpulan
Memilih gunung yang akan dipantau, Contoh masukan :	Menyimpan gunung yang ingin dipantau	Dapat memilih gunung yang akan dipantau	[√] Diterima [ ] Ditolak

-gunung merapi -gunung guntur			

12. Pengujian Tampil Info Gunung

Tabel 4.22 Pengujian Tampilkan Informasi

Kasus dan Hasil Uji		
Data yang diharapkan	Pengamatan	Kesimpulan
Menampilkan informasi status gunung	Dapat menampilkan menampilkan informasi status gunung	[√] Diterima [ ] Ditolak

13. Pengujian Tampilkan Detail Gunung

Tabel 4.23 Pengujian Tampilkan Detail Informasi

Kasus dan Hasil Uji		
Data yang diharapkan	Pengamatan	Kesimpulan
Menampilkan Informasi : Status gunung Pesan rekomendasi	Dapat menampilkan informasi berupa Status Gunung, pesan rekomendasi, dan deskripsi gunung	[√] Diterima [ ] Ditolak

14. Pengujian Tampilkan Peta KRB

Tabel 4.24 Pengujian Tampilkan Peta KRB

Kasus dan Hasil Uji		
Data yang diharapkan	Pengamatan	Kesimpulan
Menampilkan Peta, menampilkan letak geografis gunung, kawasan rawan bencana gunung dan lokasi perangkat.	Dapat menampilkan Peta, menampilkan letak geografis gunung, kawasan rawan bencana gunung dan lokasi perangkat.	[√] Diterima [ ] Ditolak

15. Pengujian Tampilkan Notifikasi

Tabel 4.25 Pengujian Notifikasi

Kasus dan Hasil Uji		
Data yang diharapkan	Pengamatan	Kesimpulan
Menampilkan notifikasi jika ada perubahan status gunung	Dapat menampilkan notifikasi jika ada perubahan status gunung	[√] Diterima [ ] Ditolak

4.2.3 Kesimpulan Hasil Pengujian Black Box

Berdasarkan hasil pengujian dengan kasus sampel uji yang telah dilakukan memberikan kesimpulan bahwa secara fungsional sistem sudah dapat menghasilkan *output* yang diharapkan.

4.2.4 Pengujian Beta

Pengujian beta merupakan pengujian yang dilakukan secara objektif dimana diuji secara langsung kepada pihak yang bersangkutan mengenai kepuasan user dengan kandungan poin yaitu pemenuhan kebutuhan dari tujuan awal pembangunan sistem peringatan dini bencan gunungapi pada aplikasi mobile. Pengujian beta dilakukan melalui dua teknik pengambilan data, yaitu melalui kuesioner dan wawancara.

Pengujian beta kuesioner dilakukan secara random terhadap mahasiswa ataupun warga pengguna smartphone untuk dapat menyimpulkan bahwa sistem ini sudah dapat diterima dan dapat dipublikasikan. Sedangkan pengujian beta melalui wawancara dilakukan untuk halaman admin pada aplikasi *backend* sistem yang dibangun.

4.2.4.1 Wawancara

Wawancara merupakan teknik pengujian beta sistem peringatan dini bencana gunung api yang dilakukan dengan petugas dari pihak Pusat Vulkanologi dan Mitigasi Bencana Geologi (PVMBG), berikut ini daftar pertanyaan dan hasil wawancara yang diberikan kepada administrator ialah :

1. Apakah aplikasi ini mempermudah penyampaian informasi tentang gunungapi kepada masyarakat, sehingga dapat mengoptimalkan upaya mitigasi bencana?
   Jawaban :

   aplikasi ini bisa mempermudah penyampaian informasi tentang gunungapi, karena masyarakat akan lebih mudah dalam mengakses serta mendapatkan informasi resmi seputar aktifitas gunungapi, sehingga bisa

dijadikan sarana penyampaian informasi lain untuk melakukan upaya mitigasi bencana.

2. Apakah aplikasi ini dapat mempercepat penyebaran informasi baik di masyarakat, media cetak maupun elektronik?

   Jawaban :

   Aplikasi ini cukup mempercepat penyebaran informasi tentang gunungapi, karena semakin banyak yang mengetahui informasi maka akan semakin cepat informasi tersebut menyebar di masyarakat umum baik melalui media cetak maupun elektronik yang saat ini sedang popular, sehingga tidak hanya pengguna aplikasi yang mengetahui informasi.

3. Apakah aplikasi ini cukup mudah untuk digunakan?

   Jawaban :

   aplikasi ini bisa dikatakan cukup mudah untuk digunakan, karena fungsionalitasnya tidak terlalu banyak dan cara pengoperasian yang cukup jelas.

4. Apakah aplikasi ini sudah sesuai dengan kebutuhan pihak PVMBG sebagai pihak yang memberikan informasi ?

   Jawaban :

   Aplikasi ini sesuai dengan kebutuhan pihak PVMBG, karena dirasa cukup membantu pihak PVMBG, karena jika sistem ini dijalankan dengan benar akan memberikan informasi tentang aktifitas gunungapi secara cepata kepada masyarakat.

5. Apakah aplikasi ini sudah layak untuk digunakan?

   Jawaban :

   Melihat fungsionalitasnya aplikasi ini cukup layak untuk digunakan sebagai peringatan dini. Namun masih perlu pengembangan dan pengujian untuk bisa dijalankan dengan benar.

4.2.4.2 Kuesioner

Kuesioner Pengujian merupakan teknik pengolahan data yang merupakan langkah dalam mengolah data yang telah didapat untuk dijadikan hasil penelitian

sehinggan dapat ditarik kesimpulan. Berikut ini adalah tahapan dalam Kuesioner Pengujian.

1. Pertanyaan Kuesioner

1. Apakah anda setuju aplikasi ini dapat mempercepat dalam mendapatkan informasi gunungapi?
2. Apakah anda setuju aplikasi ini mempermudah dalam mendapatkan informasi kawasan rawan bencana gunungapi ?
3. Apakah anda setuju aplikasi ini mudah untuk digunakan?
4. Apakah anda setuju tampilan antarmuka aplikasi ini menarik?
5. Apakah anda setuju aplikasi ini layak untuk digunakan?

Berdasarkan data yang dihasilkan dari kuesioner, dilakukan perhitungan menggunakan skala likert. Skala Likert adalah metode perhitungan yang digunakan untuk keperluan riset atas jawaban setuju atau tidaknya seorang responden terhadap suatu pernyataan. Untuk menghitung skor maksimum tiap jawaban, dengan mengalikan skor dengan jumlah keseluruhan responden, yaitu skor dikali 30 responden.

Tabel 4.26 Skor Maksimum

Jawaban	Skor	Skor Maksimum (Skor * Jumlah Koresponden)
Sangat Setuju	5	150
Setuju	4	120
Cukup Setuju	3	90
Kurang Setuju	2	60
Tidak Setuju	1	30

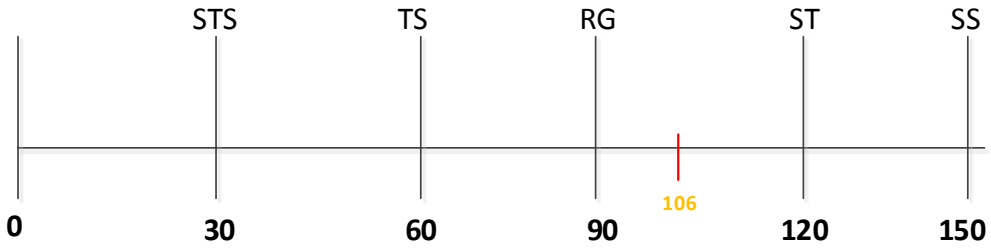
4.2.4.3 Skenario Pengujian Kuesioner

Berikut ini adalah hasil persentase masing – masing jawaban yang sudah dihitung nilainya. Kuesioner ini telah diujikan kepada 30 orang calon pengguna aplikasi peringatan dini bencana gunungapi. Hasil perhitungan persentase masing – masing jawaban dapat dilihat pada tabel berikut :

1. Apakah anda setuju aplikasi ini dapat mempercepat dalam mendapatkan informasi gunungapi ?

Tabel 4.27 Hasil Kuesioner Pertanyaan 1

Pertanyaan	Jawaban	Skor	Responden	Jumlah Skor	Nilai Presentase (%)
1	Sangat Setuju	5	0	0	(106:150)x100= 70,6%
	Setuju	4	16	64	
	Cukup Setuju	3	14	42	
	Kurang Setuju	2	0	0	
	Tidak Setuju	1	0	0	
Jumlah			30	106	

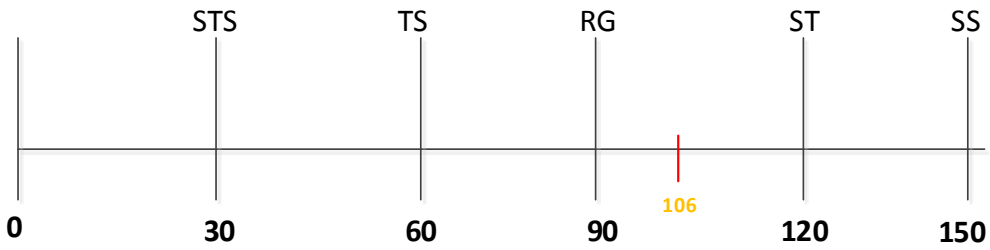


Jadi berdasarkan pengolahan data yang diperoleh dari 30 responden, maka rata-rata 106 terletak pada kriteria setuju.

2. Apakah aplikasi ini mempermudah dalam mendapatkan informasi letak geografis gunungapi?

Tabel 4.28 Hasil Kuesioner Pertanyaan 2

Pertanyaan	Jawaban	Skor	Responden	Jumlah Skor	Nilai Presentase (%)
2	Sangat Setuju	5	3	15	(106:150)x100= 70,6%
	Setuju	4	10	40	
	Cukup Setuju	3	17	51	
	Kurang Setuju	2	0	0	
	Tidak Setuju	1	0	0	
Jumlah			30	106	



Jadi berdasarkan pengolahan data yang diperoleh dari 30 responden, maka rata-rata 106 terletak pada kriteria setuju.

3. Apakah anda setuju aplikasi ini mudah untuk digunakan?

Tabel 4.29 Hasil Kuesioner Pertanyaan 3

Pertanyaan	Jawaban	Skor	Responden	Jumlah Skor	Nilai Presentase (%)
3	Sangat Setuju	5	3	30	(110:150)x100= 73,3%
	Setuju	4	9	36	
	Cukup Setuju	3	18	54	
	Kurang Setuju	2	0	0	
	Tidak Setuju	1	0	0	
Jumlah			30	110	

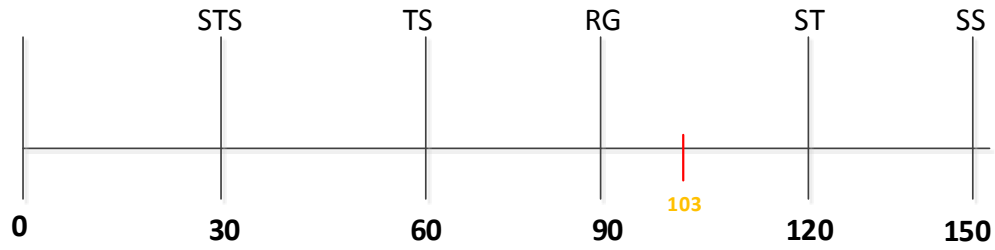


Jadi berdasarkan pengolahan data yang diperoleh dari 30 responden, maka rata-rata 110 terletak pada kriteria setuju.

4. Apakah anda setuju tampilan antarmuka aplikasi ini menarik?

Tabel 4.30 Hasil Kuesioner Pertanyaan 4

Pertanyaan	Jawaban	Skor	Responden	Jumlah Skor	Nilai Presentase (%)
4	Sangat Setuju	5	2	10	(103:150)x100= 68,6%
	Setuju	4	9	36	
	Cukup Setuju	3	19	57	
	Kurang Setuju	2	0	0	
	Tidak Setuju	1	0	0	
Jumlah			30	103	

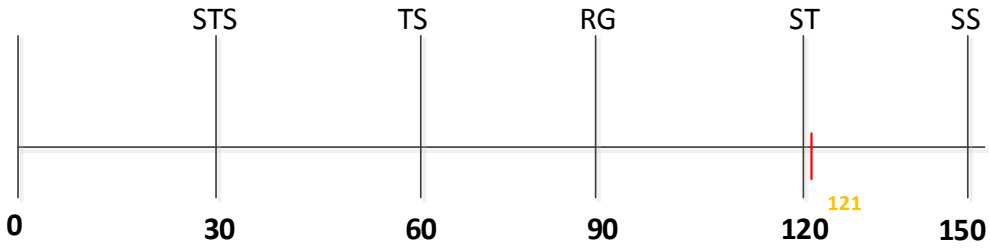


Jadi berdasarkan pengolahan data yang diperoleh dari 30 responden, maka rata-rata 103 terletak pada kriteria setuju.

5. Apakah anda setuju aplikasi ini layak untuk digunakan?

Tabel 4.31 Hasil Kuesioner Pertanyaan 5

Pertanyaan	Jawaban	Skor	Responden	Jumlah Skor	Nilai Presentase (%)
5	Sangat Setuju	5	5	25	(121:150)x100= 80,6 %
	Setuju	4	21	84	
	Cukup Setuju	3	4	12	
	Kurang Setuju	2	0	0	
	Tidak Setuju	1	0	0	
Jumlah			30	121	



Jadi berdasarkan pengolahan data yang diperoleh dari 30 responden, maka rata-rata 121 terletak pada kriteria sangat setuju.

4.2.5 Kesimpulan Hasil Pengujian Beta

Berdasarkan pengujian beta diatas, dapat diambil kesimpulan bahwa:

1. Aplikasi ini bisa mempercepat pengguna dalam mendapatkan informasi gunungapi.
2. Aplikasi ini dapat mempermudah pengguna dalam mendapatkan informasi kawasan rawan bencana gunungapi.
3. Aplikasi ini dapat mempermudah pihak PVMBG dalam menyampaikan informasi kepada masyarakat.
4. Aplikasi ini layak untuk digunakan oleh pengguna.