

SISTEM PENGUJIAN APLIKASI *FLASH* DENGAN METODE *EXPLORATORY TESTING*

TUGAS AKHIR

Diajukan Sebagai Salah Satu Syarat
Untuk Memperoleh Gelar Sarjana
Teknik Informatika



Oleh :

Nama : Herlina

NIM : 10523502

**JURUSAN TEKNIK INFORMATIKA
FAKULTAS TEKNOLOGI INDUSTRI
UNIVERSITAS ISLAM INDONESIA**

2014

HALAMAN PENGESAHAN DOSEN PEMBIMBING

LEMBAR PENGESAHAN DOSEN PEMBIMBING

SISTEM PENGUJIAN APLIKASI *FLASH* DENGAN
METODE *EXPLORATORY TESTING*

TUGAS AKHIR

Oleh:

Nama : Herlina

NIM : 10523502

Yogyakarta, Mei 2014

Pembimbing


Beni Suranto, S.T, M.SoftEng.

HALAMAN PENGESAHAN DOSEN PENGUJI

HALAMAN PENGESAHAN DOSEN PENGUJI

SISTEM PENGUJIAN APLIKASI *FLASH* DENGAN METODE *EXPLORATORY TESTING*

TUGAS AKHIR

Oleh:

Nama : Herlina

NIM : 10523502

Telah Dipertahankan di Depan Sidang Penguji sebagai Salah Satu Syarat
untuk Memperoleh Gelar Sarjana Jurusan Teknik Informatika
Fakultas Teknologi Industri Universitas Islam Indonesia

Yogyakarta, 10 Juni 2014

Tim Penguji

Beni Suranto, S.T., M.SoftEng

Ketua

Yudi Prayudi, S.Si., M.Kom.

Anggota 1

Affan Mahtarami, S.Kom., MT.

Anggota 2

Mengetahui,

Ketua Jurusan Teknik Informatika

Universitas Islam Indonesia



(Yudi Prayudi, S.Si., M.Kom.)

LEMBAR PERNYATAAN KEASLIAN TUGAS

Yang bertanda tangan di bawah ini:

Nama : Herlina

NIM : 10523502

Tugas akhir dengan judul:

SISTEM PENGUJIAN APLIKASI *FLASH* DENGAN METODE *EXPLORATORY TESTING*

Menyatakan dengan sesungguhnya bahwa dalam tugas akhir ini tidak terdapat keseluruhan tulisan atau karya yang diambil dengan menyalin, meniru dalam bentuk rangkaian kalimat atau simbol atau algoritma atau program yang menunjukkan gagasan atau pendapat atau pemikiran orang lain, yang diakui seolah-olah tulisan atau karya sendiri.

Apabila di kemudian hari terbukti di beberapa bagian dari karya ini adalah bukan hasil karya sendiri, tugas akhir yang diajukan sebagai hasil karya sendiri ini siap ditarik kembali dan siap menanggung resiko dan konsekuensi apapun.

Demikian surat pernyataan ini dibuat, semoga dapat dipergunakan sebagaimana mestinya.

Yogyakarta, Mei 2014

Yang membuat pernyataan



(Herlina)

HALAMAN PERSEMBAHAN

Ku persembahkan tugas akhir ini untuk . . .

Mama ku yang kuat dan **papa** ku yang aku banggakan, Nelyati dan Sirjon, yang senantiasa mendoakan, memberi dukungan dan berjuang untuk putrinya ini. Terima kasih atas kasih sayang yang mungkin aku tidak dapat membalasnya.

Abang ku tersayang dan satu-satunya, Nico Bryan Franata. Terima kasih telah memberi dukungan dan tauladan bagi adikmu ini.

Adik-adikku yang kakak sayangi, M. Feri Saputra, Melisa Rezki Puspita Sari, Feni Rahma Puspita Sari, dan Nova Hafizaniyallah. Terima kasih adik-adikku, kehadiran kalian mewajibkan kakak untuk melakukan hal terbaik agar menjadi contoh yang baik bagi kalian.

HALAMAN MOTTO

“Allah selalu menjagamu, Dia tidak pernah membiarkanmu berjalan sendiri”

“In tanshurullah Yanshurkumullah”

“Adhu-du’afaa-u yatafawwaquuna ‘alal adzkiyaa-I jiddan wa tawakkulan”

“I have no special talents, I am only passionately curious” – Albert Einstein

“When you give up, it means you never wanted it” - @RealTalk

“Be the best but don’t feel the best”

KATA PENGANTAR



Assalaamu 'alaikum wa rahmatullah wa barakaatuh

Segala puji bagi Allah SWT yang telah memberikan segala nikmat dan hidayahNya yang dapat dirasakan hingga saat ini. Shalawat dan salam dipanjatkan kepada nabi dan rasul Allah, Muhammad SAW yang menjadi suri tauladan bagi semua umat manusia.

Dengan izin Allah, tugas akhir yang berjudul “Sistem Pengujian Aplikasi *Flash* dengan Metode *Exploratory Testing*” dapat diselesaikan. Semua ini tentunya dapat terselesaikan atas dukungan dan bantuan dari pihak-pihak lain. Untuk itu penulis ucapkan terima kasih sebesar-besarnya kepada:

1. Allah SWT yang selalu menjaga diri ini dan menjadi tempat bersandar.
2. Mama, papa, abang dan adik-adikku atas doa, dukungan dan kasih sayangnya.
3. Dekan Fakultas Teknologi Industri.
4. Ketua jurusan teknik informatika.
5. Dosen pembimbing, bapak Beni Suranto S.T, M.SoftEng atas ilmu dan waktu yang diberikan.
6. Dosen-dosen di teknik informatika UII atas ilmu yang diberikan.
7. Tutor dalam pengerjaan tugas akhir, Imam Kusuma, Ekky, Imam S. Rifkan, Indra.
8. Sahabat-sahabat yang selalu mendukung, Panda alias April, Sorijeks alias Ayuk, Ijul alias Zulmi, Nuka alias Nur, Mprit alias Fitri, Niiwul alias Wulan, Maya casin alias Maya Chasindayo, Maia nano alias Dina Kusmayanti, Mita Buwana dan masih banyak lagi yang tidak dapat disebutkan satu-persatu.
9. Teman-teman dari Laboratorium Komputasi dan Sistem Cerdas.
10. Dan pihak-pihak lain yang tidak dapat disebutkan satu-persatu.

Terima kasih atas semua pihak yang telah mendukung dan mendoakan. Semoga selalu dilimpahkan kebaikan.

Dalam pengerjaan tugas akhir ini tentunya masih terdapat kekurangan, untuk itu penulis memohon maaf atas kekhilafan. Saran dan kritik yang membangun diharapkan untuk memperbaiki di tahap pengerjaan sejenis berikutnya.

Akhir kata, semoga tugas akhir ini bermanfaat bagi orang banyak. Aamiin Ya Rabbal 'alamiin.

Wassalaamu'alaikum wa rahmatullah wa barakaatuh

Yogyakarta, Mei 2014

Herlina

SARI

Pengujian merupakan suatu tahapan penting dalam membangun perangkat lunak. Melalui tahap pengujian, dapat diketahui kekurangan dari perangkat lunak. Salah satu jenis pengujian adalah pengujian *usability*. Pada pengujian ini dilakukan pengujian terhadap aspek-aspek usability pada perangkat lunak. Penelitian ini melakukan pengujian *usability* terhadap aplikasi *flash*. Aplikasi *flash* merupakan aplikasi yang banyak digunakan oleh mahasiswa teknik informatika UII untuk dijadikan sebagai tugas akhir khususnya aplikasi *flash* untuk *game* dan alat bantu ajar.

Pada penelitian ini dikembangkan sistem pengujian aplikasi *flash* menggunakan *framework CodeIgniter* dengan bahasa pemrograman PHP dan basisdata MySQL. Aspek *usability* yang diterapkan untuk pengujian terdiri dari *learnability*, *memorability*, *efficiency*, *minimal error rate*, dan *high user satisfaction*.

Simpulan yang didapat berupa sistem yang dapat menguji aplikasi *flash* dengan menampilkan aplikasi dan *form* pertanyaan pengujian yang akan diisi oleh *tester*. Aplikasi yang telah diuji dapat diketahui *score* hasil pengujian sehingga *developer* dapat mengetahui aspek mana yang kurang diterapkan dalam membangun aplikasi. Melalui pengujian ini *developer* dapat memperbaiki kekurangan yang ada. Manfaat bagi *developer* lainnya, dapat mengetahui contoh aplikasi yang dapat dijadikan rujukan dalam membangun aplikasi *flash*.

Kata kunci:

Sistem pengujian aplikasi *flash*, pengujian *usability*, *flash*, *game*, alat bantu ajar

TAKARIR

<i>Aplikasi alat bantu ajar</i>	Aplikasi yang dikembangkan untuk mendukung pembelajaran
<i>Database</i>	Basisdata
<i>Developer</i>	Pengembang perangkat lunak
<i>Edit</i>	Ubah
<i>Email</i>	Surat elektronik
<i>Exploratory testing</i>	Pengujian yang memperkenankan penguji untuk berselancar menggunakan aplikasi
<i>File</i>	Dokumen
<i>Flash</i>	Perangkat lunak yang digunakan untuk membuat animasi
<i>Form</i>	Formulir
<i>Input</i>	Masukan
<i>Output</i>	Keluaran
<i>Password</i>	Kata sandi
<i>Reset</i>	Atur ulang
<i>Score</i>	Nilai
<i>Tester</i>	Penguji perangkat lunak
<i>Upload</i>	Unggah
<i>Usability</i>	Kebergunaan

DAFTAR ISI

COVER	i
HALAMAN PENGESAHAN DOSEN PEMBIMBING.....	ii
HALAMAN PENGESAHAN DOSEN PENGUJI.....	iii
LEMBAR PERNYATAAN KEASLIAN TUGAS.....	Error! Bookmark not defined.
HALAMAN PERSEMBAHAN	v
HALAMAN MOTTO	vi
KATA PENGANTAR	vii
SARI	ix
TAKARIR.....	x
DAFTAR ISI.....	xi
DAFTAR GAMBAR	xiv
DAFTAR TABEL.....	xix
BAB I	
PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	2
1.3 Batasan Masalah.....	2
1.4 Tujuan Penelitian.....	2
1.5 Manfaat Penelitian.....	3
1.6 Metodologi	3
BAB II.....	6
LANDASAN TEORI.....	6
2.1 Rekayasa Perangkat Lunak	6
2.2 Pengujian Perangkat Lunak.....	10

2.3	Konsep Usabilitas.....	12
2.4	Metode Exploratory Testing.....	16
BAB III		
METODOLOGI.....		18
3.1	Analisis Kebutuhan	19
3.1.1	Analisis kebutuhan <i>input</i>	19
3.1.2	Analisis kebutuhan proses.....	20
3.1.3	Analisis kebutuhan <i>output</i>	21
3.2	Perancangan Sistem.....	22
3.2.1	Kebutuhan Sistem	22
3.2.2	Perilaku sistem	24
3.2.3	Perancangan pertanyaan.....	33
3.2.4	Perancangan basis data.....	33
3.2.5	Perancangan antarmuka	36
3.3	Perancangan Pengujian Aplikasi	61
BAB IV		64
IMPLEMENTASI DAN PENGUJIAN		64
4.1	Implementasi Perangkat lunak	64
4.1.9	Halaman manajemen pertanyaan	69
4.1.9	Halaman change password.....	71
4.1.10	Halaman home developer.....	71
4.1.11	Halaman <i>my profil</i>	72
4.1.12	Halaman <i>my application</i>	73
4.1.13	Halaman <i>about us</i>	73
4.1.14	Halaman <i>upload</i>	74

4.1.15	Halaman contact admin.....	74
4.1.16	Halaman <i>received message</i>	74
4.1.17	Halaman <i>sent message</i>	75
4.1.18	Halaman <i>home tester</i>	76
4.1.19	Halaman <i>test</i> aplikasi	76
4.1.20	Halaman <i>list</i> aplikasi <i>game</i>	79
4.1.21	Halaman <i>list</i> aplikasi alat bantu ajar	80
4.1.22	Halaman <i>about us</i>	80
4.1.23	Halaman contact us	81
4.2	Pengujian	82
4.2.1	Pengujian masukan dan keluaran pada perangkat lunak	82
4.2.2	Penanganan kesalahan.....	93
4.2.3	Pengujian URL (Uniform Resource Locator) Injection.....	97
4.2.4	Pengujian SQL (Structured Query language) Injection	98
4.2.5	Pengujian XSS (Cross Side Scripting).....	98
4.2.6	Pengujian kepuasan oleh pengguna	99
4.2.7	Hasil Analisis	99
4.2.8	Kelebihan dan Kekurangan.....	99
BAB V		
KESIMPULAN DAN SARAN.....		100
5.1	Kesimpulan.....	100
5.2	Saran.....	101
DAFTAR PUSTAKA		101
LAMPIRAN		113

DAFTAR GAMBAR

Gambar 2.1 Model waterfall	8
Gambar 2.2 Model <i>spiral boehm</i>	9
Gambar 2.3 <i>Incremental model</i>	9
Gambar 2.4 White-box testing.....	11
Gambar 2.6 <i>Learnability</i>	12
Gambar 2.7 <i>Memorability</i>	14
Gambar 2.8 <i>Efficiency</i>	14
Gambar 2.9 <i>Error</i>	15
Gambar 3.1 Gambaran umum.....	18
Gambar 3.2 <i>Use case</i>	25
Gambar 3.3 Diagram <i>activity</i> proses registrasi	26
Gambar 3.4 Diagram <i>activity</i> proses login.....	27
Gambar 3.5 Diagram <i>activity upload</i> aplikasi	28
Gambar 3.6 Diagram <i>activity edit</i> aplikasi.....	28
Gambar 3.7 Diagram <i>activity</i> hapus aplikasi	29
Gambar 3.8 Diagram <i>activity edit</i> profil	30
Gambar 3.9 Diagram <i>activity edit password</i>	30
Gambar 3.10 Diagram <i>activity reset password</i>	31
Gambar 3.11 Diagram <i>activity</i> kirim pesan	32
Gambar 3.12 Diagram uji aplikasi	32
Gambar 3.13 ERD.....	36
Gambar 3.14 Rancangan antarmuka <i>login</i>	37
Gambar 3.15 Rancangan antarmuka <i>home admin</i>	38
Gambar 3.16 Rancangan antarmuka <i>management developer</i>	38

Gambar 3.17 Rancangan antarmuka <i>view developer</i>	39
Gambar 3.18 Rancangan antarmuka <i>edit developer</i>	39
Gambar 3.19 Rancangan antarmuka <i>management</i> aplikasi <i>game</i>	40
Gambar 3.20 Rancangan antarmuka <i>management</i> aplikasi alat bantu ajar.....	41
Gambar 3.21 Rancangan antarmuka <i>view</i> aplikasi	41
Gambar 3.22 Rancangan antarmuka <i>edit</i> aplikasi.....	42
Gambar 3.23 Rancangan antarmuka <i>view statistik</i> aplikasi <i>game</i>	43
Gambar 3.24 Rancangan antarmuka <i>view statistik</i> aplikasi alat bantu ajar	44
Gambar 3.25 Rancangan antarmuka <i>management</i> pertanyaan	45
Gambar 3.26 Rancangan antarmuka <i>edit</i> pertanyaan.....	45
Gambar 3.27 Rancangan antarmuka <i>change password</i>	45
Gambar 3.28 Rancangan antarmuka <i>home</i>	46
Gambar 3.30 Rancangan antarmuka <i>my profil</i>	47
Gambar 3.31 Rancangan antarmuka <i>edit profil</i>	48
Gambar 3.33 Rancangan antarmuka <i>my application</i>	49
Gambar 3.35 Rancangan antarmuka <i>list</i> aplikasi <i>game</i>	50
Gambar 3.36 Rancangan antarmuka <i>list</i> aplikasi alat bantu ajar	51
Gambar 3.37 Rancangan antarmuka <i>view statistik game</i>	51
Gambar 3.38 Rancangan antarmuka <i>view statistik</i> alat bantu ajar.....	52
Gambar 3.39 Rancangan antarmuka <i>view detail</i> aplikasi	53
Gambar 3.40 Rancangan antarmuka <i>about us</i>	53
Gambar 3.41 Rancangan antarmuka <i>home</i>	54
Gambar 3.43 Rancangan antarmuka <i>test</i> aplikasi	56
Gambar 3.44 Rancangan antarmuka <i>form testing</i>	57
Gambar 3.46 Rancangan antarmuka <i>list</i> aplikasi <i>game</i>	58
Gambar 3.47 Rancangan antarmuka <i>list</i> aplikasi alat bantu ajar	58
Gambar 3.48 Rancangan antarmuka <i>view statistik</i> aplikasi <i>game</i>	59

Gambar 3.49 Rancangan antarmuka <i>view statistik</i> aplikasi alat bantu ajar	60
Gambar 3.50 Rancangan antarmuka <i>about us</i>	60
Gambar 3.51 Rancangan antarmuka <i>contact us</i>	61
Gambar 4.1 Halaman <i>login</i>	64
Gambar 4.2 Halaman <i>home admin</i>	65
Gambar 4.3 Halaman manajemen <i>developer</i>	65
Gambar 4.4 Halaman <i>view developer</i>	66
Gambar 4.5 Halaman <i>view developer</i>	66
Gambar 4.6 Halaman manajemen aplikasi <i>game</i>	67
Gambar 4.7 Halaman aplikasi alat bantu ajar	68
Gambar 4.8 Halaman <i>view aplikasi</i>	68
Gambar 4.9 Halaman <i>edit aplikasi</i>	69
Gambar 4.10 Halaman <i>view statistik</i>	69
Gambar 4.11 Halaman manajemen pertanyaan	70
Gambar 4.12 Halaman <i>add pertanyaan</i>	70
Gambar 4.13 Halaman <i>edit pertanyaan</i>	71
Gambar 4.14 Halaman <i>change password</i>	71
Gambar 4.15 Halaman <i>home developer</i>	72
Gambar 4.17 Halaman <i>my application</i>	73
Gambar 4.18 Halaman <i>about us</i>	73
Gambar 4.19 Halaman <i>upload</i>	74
Gambar 4.20 Halaman <i>contact admin</i>	74
Gambar 4.21 Halaman <i>received message</i>	75
Gambar 4.22 Halaman <i>sent message</i>	75
Gambar 4.23 Halaman <i>home</i>	76
Gambar 4.24 Halaman <i>test aplikasi</i>	77
Gambar 4.25 Halaman <i>form uji aplikasi</i>	78

Gambar 4.26 Halaman <i>view</i> aplikasi.....	79
Gambar 4.27 Halaman <i>list</i> aplikasi <i>game</i>	80
Gambar 4.28 Halaman <i>list</i> aplikasi alat bantu ajar	80
Gambar 4.29 Halaman <i>about us</i>	81
Gambar 4.30 Halaman <i>contact us</i>	82
Gambar 4.31 Halaman masukan <i>sign up</i>	83
Gambar 4.32 Halaman keluaran <i>sign up</i>	83
Gambar 4.33 Masukan <i>login</i>	84
Gambar 4.34 Keluaran <i>login</i>	84
Gambar 4.35 Masukan <i>edit profil</i>	85
Gambar 4.36 Keluaran <i>edit profil</i>	85
Gambar 4.37 Masukan <i>edit password</i>	85
Gambar 4.38 Keluaran <i>edit password</i>	86
Gambar 4.39 Masukan <i>upload</i>	86
Gambar 4.40 Keluaran <i>upload</i>	87
Gambar 4.41 Masukan <i>edit</i> aplikasi.....	87
Gambar 4.42 Keluaran <i>edit</i> aplikasi.....	88
Gambar 4.43 Masukan data diri <i>tester</i>	88
Gambar 4.44 Masukan data pengujian.....	89
Gambar 4.45 Keluaran hasil pengujian.....	90
Gambar 4.46 Masukan kirim pesan <i>tester</i>	90
Gambar 4.47 Keluaran kirim pesan <i>tester</i>	91
Gambar 4.48 Masukan <i>contact us</i>	91
Gambar 4.49 Keluaran <i>contact us</i>	92
Gambar 4.50 Masukan balas pesan.....	92
Gambar 4.51 Keluaran balas pesan.....	93
Gambar 4.52 Penanganan kesalahan <i>login</i>	94

Gambar 4.53 Penanganan kesalahan <i>email</i> terdaftar	95
Gambar 4.54 Penanganan kesalahan <i>email</i> tidak valid.....	95
Gambar 4.55 Penanganan kesalahan jumlah karakter <i>password</i>	95
Gambar 4.56 Penanganan kesalahan validasi <i>password</i>	96
Gambar 4.57 Penanganan kesalahan <i>password</i> konfirmasi	96
Gambar 4.58 Penanganan kesalahan <i>upload</i>	97
Gambar 4.59 Penanganan kesalahan data kosong	97
Gambar 4.60 Pengujian URL <i>injection</i>	97
Gambar 4.61 Pengujian SQL <i>injection</i>	98
Gambar 4.62 Pengujian XSS <i>injection</i>	99

DAFTAR TABEL

Tabel 3.1 Tabel <i>aplikasi</i>	33
Tabel 3.2 Tabel <i>aspek</i>	33
Tabel 3.3 Tabel <i>user</i>	34
Tabel 3.4 Tabel <i>hasil</i>	34
Tabel 3.5 Tabel <i>jenis</i>	35
Tabel 3.6 Tabel <i>pertanyaan</i>	35
Tabel 3.7 Tabel <i>pesan</i>	35
Tabel 3.8 Tabel <i>tester</i>	36

BAB I

PENDAHULUAN

1.1 Latar Belakang

Dalam ilmu rekayasa perangkat lunak, terdapat beberapa langkah yang dilakukan untuk membangun suatu aplikasi. Beberapa di antaranya yaitu perancangan, implementasi dan pengujian perangkat lunak. Perangkat lunak dibangun sesuai rancangan, kemudian dilakukan pengujian. Pengujian merupakan tahapan penting dalam mengembangkan perangkat lunak. Pengembang perangkat lunak bisa jadi merasa sudah membangun sistem atau aplikasi sesuai yang diinginkan pengguna, tetapi ketika digunakan masih ditemukan beberapa kekurangan, untuk itu diperlukan tahap pengujian. Pengujian berguna untuk mengetahui apakah sistem yang sudah dibuat sudah sesuai dengan apa yang diharapkan atau belum.

Untuk mengukur kepuasan pengguna pada aplikasi *flash* yang digunakan, perlu dilakukan pengujian usabilitas oleh para pengguna terhadap aplikasi tersebut. Pengujian ini berguna untuk mendapat respon pengguna terhadap performa aplikasi yang digunakan. Beberapa hal yang diuji seperti kemudahan penggunaan, penyajian informasi, alur penggunaan aplikasi dan lainnya.

Salah satu metode pengujian usabilitas adalah metode *exploratory testing*. Pada pengujian ini pengguna diminta untuk mengeksplorasi atau menggunakan aplikasi yang diuji. Selanjutnya pengguna diharapkan dapat memberikan respon berupa saran dan kritik sebagai evaluasi aplikasi.

Pada penelitian ini pengujian dilakukan terhadap aplikasi *flash*. Aplikasi *flash* merupakan salah satu aplikasi yang semakin banyak digunakan pada setiap tingkatan umur. Misalnya saja, pada aplikasi *game*. Aplikasi *game* merupakan salah satu aplikasi yang banyak digemari, baik dari kalangan anak-anak hingga kalangan dewasa. Contoh lainnya terdapat pada alat bantu ajar. Aplikasi *flash* digunakan untuk membantu para pelajar agar lebih mudah dalam memahami apa yang dipelajari. Penggunaan *flash* dalam beberapa aplikasi dikarenakan

penampilannya yang menarik dan interaktif sehingga lebih mudah digunakan oleh para pengguna.

Jenis aplikasi *flash* yang diuji dalam penelitian ini adalah aplikasi *game* dan alat bantu ajar. Kedua jenis aplikasi tersebut merupakan jenis yang banyak digunakan oleh mahasiswa teknik informatika untuk menyelesaikan tugas akhir di Universitas Islam Indonesia.

1.2 Rumusan Masalah

Rumusan masalah pada penelitian ini adalah bagaimana mengembangkan sistem pengujian aplikasi *flash* dengan metode *exploratory testing* yang dapat dimanfaatkan untuk melakukan pengujian terhadap aplikasi *flash* secara efektif, efisien dan fleksibel.

1.3 Batasan Masalah

Adapun batasan masalah pada penelitian ini adalah:

1. Sistem ditujukan untuk pengujian aplikasi *flash* yang berupa alat bantu ajar dan aplikasi *game*.
2. Sistem akan digunakan untuk proses pengujian dengan metode *exploratory testing* secara terpandu.
3. Pengujian fokus pada aspek usabilitas.
4. Sistem hanya ditujukan untuk aplikasi dengan format *single file*.

1.4 Tujuan Penelitian

Tujuan dari dibuatnya sistem pengujian aplikasi *flash* ini adalah:

1. Membuat sistem yang memudahkan *tester* dalam pengujian aplikasi *flash*.
2. Membuat sistem yang memudahkan perancang aplikasi *flash* untuk mendapatkan hasil pengujian terhadap aplikasi yang dibuat.
3. Merancang sistem yang dapat menampilkan *list* aplikasi *flash* berdasarkan hasil uji.

1.5 Manfaat Penelitian

Penelitian ini diharapkan bermanfaat bagi:

1. Peneliti

Peneliti dapat memahami dan membangun sistem pengujian terhadap aspek usabilitas pada aplikasi *flash* dengan metode *exploratory testing*.

2. Mahasiswa

Mahasiswa mendapat contoh aplikasi *flash* yang dapat digunakan sebagai bahan rujukan dalam tugas akhir. Kekurangan pada aplikasi *flash* yang telah diuji diharapkan dapat dihindari oleh mahasiswa dalam membangun aplikasi *flash* dan kelebihan sistem diharapkan dapat menjadi bahan rujukan yang dapat dicontoh atau ditingkatkan.

3. Pengembang sistem:

- a. Meningkatkan efisiensi kepada *tester* dalam pengujian aplikasi *flash* dengan cara melakukan .
- b. Pengembang aplikasi mendapat respon yang lebih cepat mengenai aplikasi yang dibuat.
- c. Para pengembang aplikasi *flash* mengetahui contoh-contoh aplikasi *flash* yang lebih mudah digunakan oleh pengguna.

1.6 Metodologi

Adapun metodologi yang dilakukan dalam penelitian ini adalah sebagai berikut:

1. Tinjauan pustaka

Tinjauan pustaka dilakukan untuk memperoleh data yang diperlukan melalui penelitian. Melalui tinjauan pustaka didapat landasan teori yang menjadi rujukan dalam penelitian.

2. Pengembangan sistem

Pengembangan sistem dilakukan dengan beberapa tahapan. Adapun tahap dalam pengembangan sistem berupa:

a. Analisis kebutuhan

Pada tahapan ini dilakukan analisis untuk mengetahui kebutuhan sistem. Kebutuhan dapat berupa kebutuhan *input*, *output*, proses hingga fungsionalitas yang dibutuhkan sistem.

b. Perancangan sistem

Pada tahapan ini dilakukan perancangan terhadap sistem yang akan dibuat. Perancangan berupa rancangan basisdata dan perancangan antarmuka.

c. Implementasi sistem

Berdasarkan rancangan sistem yang telah dibuat, maka dilakukan implementasi program. Implementasi merupakan tahapan terpenting dalam pembangunan sistem, dimana dilakukan penulisan kode program sesuai sistem yang akan dibangun.

d. Pengujian sistem

Sistem yang telah dibuat, kemudian diuji. Pengujian dilakukan untuk mengetahui apakah sistem yang dibuat telah sesuai dengan apa yang diharapkan atau belum.

3. Penyusunan laporan

Penyusunan laporan merupakan merupakan tahap akhir dalam penelitian ini. Laporan berisi pembahasan mengenai penelitian yang dilakukan. Adapun sistematika laporan pada penelitian ini berisi:

BAB I PENDAHULUAN

Berisi mengenai latar belakang masalah, rumusan masalah, batasan masalah, tujuan masalah, manfaat masalah, metodologi, hingga sistematika penulisan laporan tugas akhir.

BAB II LANDASAN TEORI

Pada bab ini dibahas beberapa teori yang dijadikan rujukan dalam penelitian. Beberapa diantaranya yaitu mengenai pengujian perangkat lunak, *usability*, dan *exploratory testing*.

BAB III METODOLOGI

Pada bab ini membahas mengenai analisis kebutuhan sistem dan perancangan sistem. Berdasarkan hasil analisis dan perancangan, selanjutnya akan dibangun sistem yang dibutuhkan.

BAB IV HASIL DAN PEMBAHASAN

Berisi mengenai pembahasan mengenai sistem yang telah dibuat. Hasil implementasi sistem dijelaskan dalam bentuk gambar dan tulisan sehingga memudahkan pembaca dalam memahami hasil sistem.

BAB V PENUTUP

Bab ini berisi kesimpulan dari laporan tugas akhir. Bab ini juga berisi saran yang diharapkan dapat diwujudkan pada penelitian selanjutnya.

BAB II

LANDASAN TEORI

2.1 Rekayasa Perangkat Lunak

Perangkat lunak (*software*) adalah instruksi (program komputer) yang ketika dijalankan menyediakan fungsi dan tampilan yang diinginkan, struktur data yang memberikan program untuk memanipulasi informasi dan dokumen yang mendeskripsikan operasi dan penggunaan program (Kristanto, 2004 : 1). Untuk menghasilkan perangkat lunak yang baik, maka dikembangkanlah suatu disiplin ilmu, yaitu rekayasa perangkat lunak.

Istilah *software engineering*, pertama kali digunakan pada akhir tahun 1950-an dan sekitar awal 1960-an. Pada tahun 1968, NATO menyelenggarakan konferensi tentang *software engineering* di Jerman dan kemudian dilanjutkan pada tahun 1969. Meski penggunaan istilah *software engineering* dalam konferensi tersebut menimbulkan perdebatan tajam tentang aspek *engineering* dari pengembangan perangkat lunak, ada banyak pihak yang menganggap bahwa konferensi tersebutlah yang menjadi awal tumbuhnya profesi rekayasa perangkat lunak.(Simarmata, 2004 : 10).

Ada beberapa pengertian rekayasa perangkat lunak menurut para ahli di bidangnya. Pengertian dari rekayasa perangkat lunak menurut para ahli adalah sebagai berikut (Kristanto, 2004 : 7)

1. Menurut Sphen R. Schach

Rekayasa perangkat lunak adalah disiplin dimana dalam menghasilkan perangkat lunak bebas dari kesalahan dan dalam pengiriman anggaran tepat waktu serta memuaskan keinginan pemakai.

2. Menurut Fritz Bauer

Rekayasa perangkat lunak adalah penetapan dan penggunaan prinsip rekayasa dalam rangka memperoleh perangkat lunak yang dapat dipercaya dan dapat bekerja secara efisien pada mesin nyata.

3. Menurut IEEE 610.12

Rekayasa perangkat lunak adalah sebuah studi pendekatan dan aplikasi secara sistematis, disiplin pengembangan operasi dan pemeliharaan perangkat lunak yang kesemuanya itu merupakan aplikasi rekayasa yang berkaitan dengan perangkat lunak.

Dari ketiga pengertian tersebut, arti yang diberikan Fritz Bauer sangat sesuai dengan tujuan dan sasaran dari rekayasa perangkat lunak. Pernyataan Stephen R.Schach terbatas pada mengaggulangi kesalahan yang terjadi jika tidak menerapkan rekayasa perangkat lunak. Pengertian yang diberikan IEEE 160.12 paling baik dalam menyampaikan wujud dari rekayasa perangkat lunak (Kristanto, 2004 : 8).

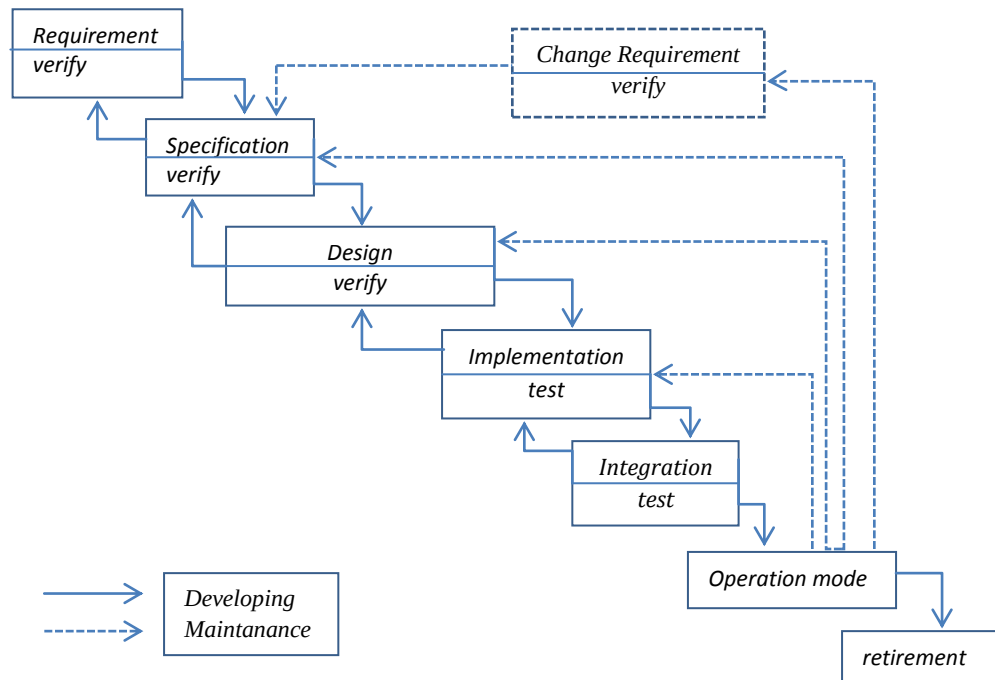
Pada rekayasa perangkat lunak, terdapat beberapa model pengembangan yang digunakan, diantaranya :

1. Model *waterfall*

Adapun tahap pengembangan perangkat lunak pada model ini adalah:

- a. *Requirement* (kebutuhan sistem)
- b. *Specification* (spesifikasi kebutuhan perangkat lunak)
- c. *Design* (desain perangkat lunak)
- d. *Implementation* (implementasi perangkat lunak)
- e. *Integration* (uji mesin Formal integrasi perangkat lunak)
- f. *Operation mode and retirement* (operasi dan perawatan)

Tahapan model *waterfall* dapat dilihat pada gambar 2.1.



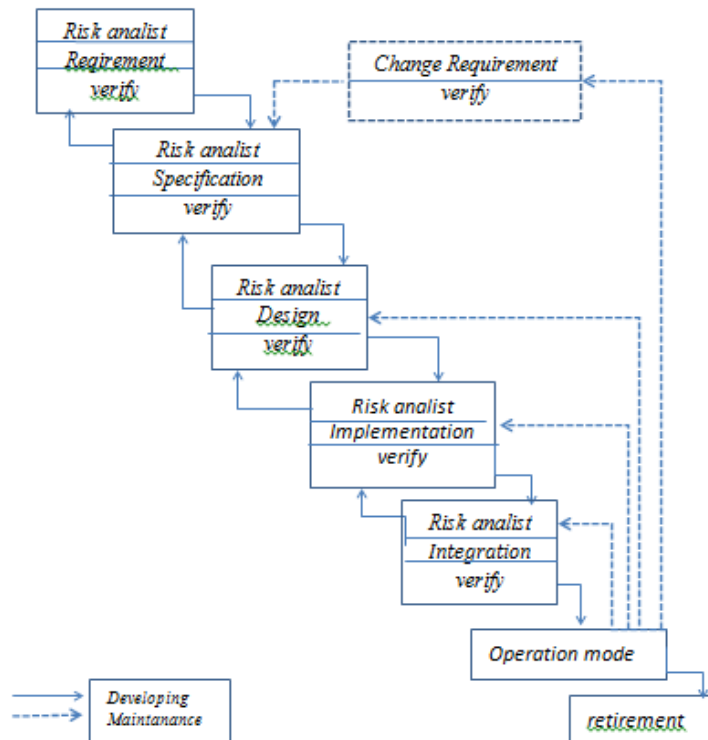
Gambar 2.1 Model waterfall

2. Model *spiral boehm*

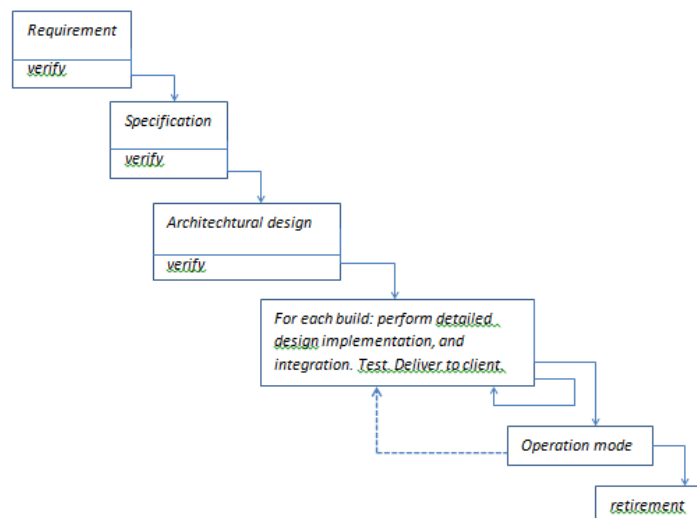
Merupakan model yang berbentuk spiral dan merupakan kombinasi dari *prototyping* model dengan *waterfall* model. Setiap tahapan model ini selalu dilakukan *risk analysis* dan verifikasi atau *testing*. Pada gambar 2.2 digambarkan tahapan model *spiral boehm*.

3. Model *incremental*

Tahapan *incremental model* adalah: *requirement*, *specification*, *architecture design*, kemudian tahapan membangun tiap bagian secara berurutan. Setiap bagian yang sudah selesai dilakukan *testing*, dikirim ke pengguna untuk dapat digunakan. Pada model ini, tiga tahapan awal harus diselesaikan terlebih dahulu sebelum tahap membangun tiap modul. Adapun gambar dari model ini dapat dilihat pada gambar 2.3.



Gambar 2.2 Model spiral boehm



Gambar 2.3 Incremental model

2.2 Pengujian Perangkat Lunak

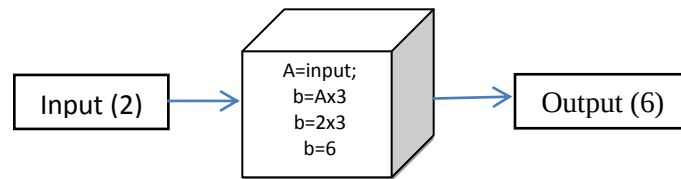
Pada model rekayasa perangkat lunak yang telah dijelaskan pada subbab sebelumnya, terdapat tahapan yang secara umum digunakan dalam pembangunan perangkat lunak. Tahapan-tahapn itu mencakup tahapan analisis kebutuhan, perancangan, implementasi sistem dan pengujian.

Pengujian adalah tahapan pengembangan perangkat lunak yang dilakukan setelah proses implementasi. Berdasarkan standar IEEE, pengujian perangkat lunak memiliki pengertian aktivitas yang dilakukan untuk mengevaluasi kualitas suatu produk dan untuk mengembangkannya dengan mengidentifikasi kelemahan dan permasalahan yang terjadi (Simarmata, 2004 : 303).

Pengujian perangkat lunak adalah proses untuk menjalankan sebuah program atau sistem untuk mencari kesalahan. Atau, pengujian perangkat lunak melibatkan semua kegiatan yang bertujuan untuk mengevaluasi atribut atau kemampuan dari sebuah program atau sistem dan menentukan bahwa pengujian perangkat lunak bertemu dengan hasil yang diperlukan. Perangkat lunak ini tidak berbeda dengan proses fisik lainnya yang diterima, dengan masukan yang diterima dan keluaran yang dihasilkan (Simarmata, 2004 : 304).

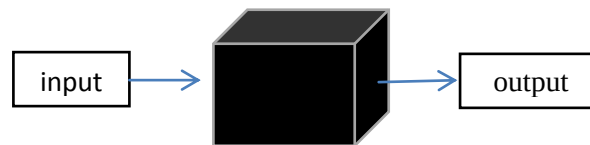
Terdapat dua jenis pengujian berdasarkan hal yang diuji, yaitu *white-box testing* dan *black-box testing*. Definisi masing-masing jenis berdasarkan IEEE adalah sebagai berikut (wicaksono, 2009) :

1. *White-box testing. Testing that takes into account the internal mechanism of a system or component.* Pengujian dilakukan pada mekanisme internal dari sebuah sistem atau komponen. Gambar 2.4 mengilustrasikan pengujian dengan *white-box testing*.



Gambar 2.4 White-box testing

2. *Black-box testing*. Testing that ignores the internal mechanism of the system or component and focuses solely on the outputs in response to selected inputs and execution conditions. Pengujian ini fokus pada apa yang ditampilkan oleh sistem untuk merespon masukan yang diberikan pengguna dan eksekusi terhadap kondisi tertentu. Ilustrasi untuk *black-box testing* dapat dilihat pada gambar 2.5.



Gambar 2.5 Black-box testing

Sedangkan berdasarkan tempat pengujian, pengujian dibagi menjadi dua, yaitu:

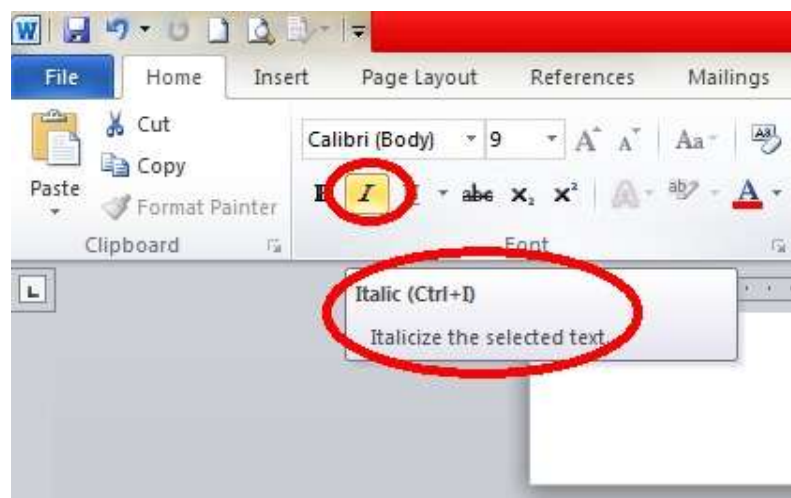
1. Pengujian alfa. Pada pengujian ini pengguna melakukan pengujian di tempat pengembang perangkat lunak, sedangkan pengembang mengawasi secara instens.
2. Pengujian beta. Pengguna melakukan pengujian di tempat pengguna, pengembang perangkat lunak tidak ikut serta dalam pengujian ini. Pengguna menggunakan aplikasi seakan sedang menggunakan sehingga tidak perlu kontribusi dari pengembang. Selanjutnya, pengguna melaporkan hasil pengujian kepada pengembang perangkat lunak.

2.3 Konsep Usabilitas

Usability berasal dari kata *usable* yang berarti dapat digunakan dengan baik (Rizky, 2007 : 61). Definisi *usability* sendiri, menurut ISO (*International Standard Organization*) adalah *Usability is the extends to which a product can be used by specified users to achieve specified goals with effectiveness, efficiency and satisfaction in a specified context of use.* *Usability* adalah seberapa jauh sebuah produk atau sistem bisa digunakan oleh *user* yang spesifik untuk mencapai tujuan yang spesifik pula secara efektif, efisien dan memuaskan pada konteks penggunaan tertentu. Usabilitas merupakan suatu aspek yang menggambarkan tingkat kenyamanan dan penerimaan pengguna terhadap sistem yang digunakan. Terdapat beberapa faktor yang digunakan untuk mengukur tingkat usabilitas suatu sistem, yaitu:

1. *Learnability* (mudah dipelajari)

Menunjukkan kualitas sistem agar mudah dipelajari, terutama bagi pengguna baru (Santoso, 2010 : 75). Pengguna dapat menguasai sistem dengan mudah sehingga dapat menggunakan sistem dengan waktu singkat serta tidak bingung dalam menggunakannya.



Gambar 2.6 *Learnability*

Pada gambar 2.6 terlihat bahwa *Microsoft Office* menyediakan keterangan mengenai fungsi dari symbol 'I' agar pengguna lebih mudah mempelajari fungsi dari simbol ini.

2. *Memorability* (mudah diingat)

Sistem diberi petunjuk atau diberi tampilan tertentu agar pengguna mudah mengingat fitur yang disediakan sistem. Pengguna juga diharapkan dapat dengan mudah mengingat langkah-langkah yang harus dilakukan untuk mengerjakan suatu *task*.

Gambar 2.7 menggambarkan prinsip usability pada tokobagus.com. pada halaman yang dikunjungi diberi hirarki menu yang sedang diakses oleh pengguna, sehingga pengguna mengetahui menu yang diakses dan kemudian mudah untuk kembali ke menu pada hirarki tertentu.

3. *Efficiency* (efisien)

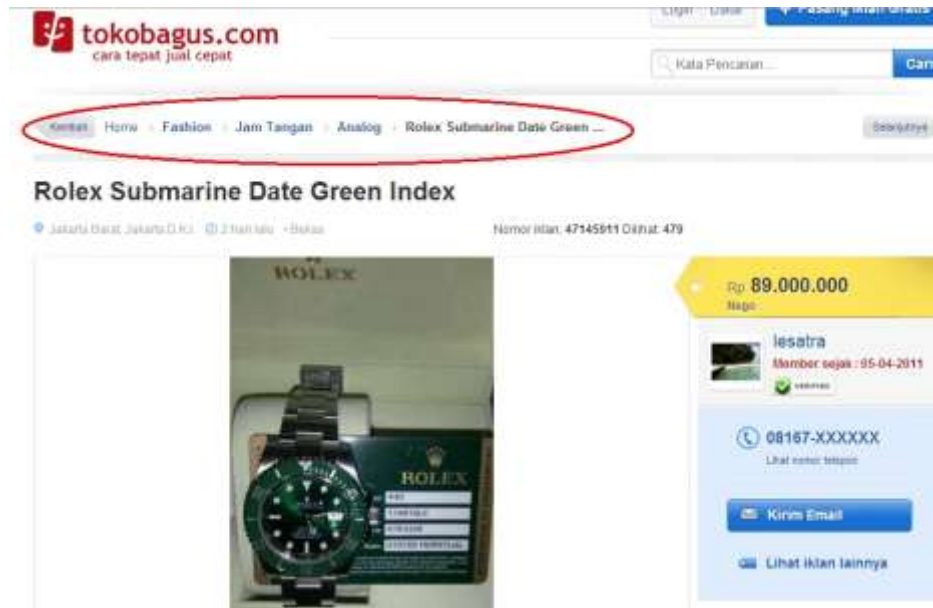
Merupakan faktor yang dilakukan sistem untuk mendukung pengguna dalam menyelesaikan pekerjaannya (Santoso, 2010 : 76). Melalui langkah operational yang sedikit dan sederhana sistem dapat melakukan hasil yang diinginkan.

Gambar 2.8 memperlihatkan penggunaan prinsip efisiensi pada aplikasi twitter. Saat pertama kali mengakses aplikasi, pengguna dapat menuliskan status twitter dengan cara yang sederhana. Pengguna cukup menekan tombol bersimbol kertas dan pensil yang terdapat pada sudut atas aplikasi, menuliskan isi twitter, lalu menekan tombol 'tweet'.

4. *Minimal error rate* (minim kesalahan)

Pengembang sistem hendaknya memperhatikan hal ini. sistem dirancang sedemikian rupa sehingga pengguna terhindar dari kesalahan. Jika terjadi kesalahan pun, sistem dapat membantu pengguna untuk mengatasi, sehingga tidak menyebabkan pengguna kehilangan data yang disebabkan oleh kesalahan yang dilakukan. Sistem sebaiknya dilengkapi dengan fitur

‘informasi kesalahan’ yang berguna untuk menunjukkan kesalahan yang dilakukan pengguna, kemudian memberi solusi untuk menyelesaikannya.



Gambar 2.7 Memorability



Gambar 2.8 Efficiency



Gambar 2.9 Error

Pada gambar 2.9 merupakan contoh dari informasi yang disampaikan ketika pengguna salah memasukkan *user id* atau *password*. Sistem Unisys menunjukkan letak kesalahan pengguna lalu memberi saran untuk pengguna agar tidak terjadi lagi kesalahan dalam penggunaan sistem.

5. *High user satisfaction* (kepuasan)

Menunjukkan tingkat kepuasan yang dirasakan oleh pengguna setelah menggunakan sistem. Disediakkannya fitur yang memudahkan interaksi pengguna terhadap user merupakan salah satu faktor yang mendukung kepuasan pengguna.

Penerapan prinsip usability merupakan hal penting dalam pengembangan aplikasi *flash*. Hal ini dikarenakan hal utama yang menarik dalam penggunaan aplikasi *flash* terdapat pada faktor visualnya. Faktor visual ini sendiri harus menerapkan aspek tertentu untuk dapat menarik pengguna terhadap aplikasi. Aspek-aspek itu terdapat pada konsep usability, seperti *memorability* dan *learnability*.

2.4 Metode Exploratory Testing

Pengembang aplikasi hendaknya memenuhi prinsip usability dalam membangun suatu aplikasi. Kemudian aplikasi yang dibangun, diuji tingkat usabilitynya melalui *usability testing* atau pengujian usability. Pengujian ini adalah proses yang bekerja dengan pengguna akhir secara langsung untuk menilai bagaimana pengguna merasakan paket perangkat lunak dan bagaimana mereka berinteraksi dengannya (Simarmata, 2004 : 317). Pengujian eksploratori merupakan salah satu dari 3 jenis pengujian usability. Dua jenis uji kebergunaan lainnya yaitu uji *threshold* dan uji perbandingan.

Uji *threshold* digunakan untuk mengukur kinerja sistem terhadap sejumlah sasaran yang ditentukan terlebih dahulu. Uji ini merupakan uji lolos atau gagalnya sistem yang dikembangkan (Santoso, 2010 : 79). Lain pula dengan uji perbandingan, pengujian ini bertujuan untuk mengukur karakteristik kebergunaan dari dua pendekatan atau rancangan untuk menentukan rancangan yang lebih cocok bagi pengguna (Santoso, 2010 : 79).

Uji eksploratori yang termasuk dalam kategori *black-box testing* yang merupakan pengujian yang berfokus pada keluaran sistem. Pengujian eksploratori bertujuan untuk menguji sebuah sistem dan mencari titik-titik di mana pengguna mengalami kebingungan, kesalahan, atau unjuk kerjanya melambat. Pengujian ini dilakukan tanpa melihat dimana persoalan-persoalan terjadi atau bentuknya. Tujuan akhir dari uji eksploratori adalah daftar persoalan yang perlu ditinjau lebih lanjut. Uji eksploratori dapat digunakan pada setiap tahap pengembangan sistem, tetapi paling efektif apabila diimplementasikan sejak awal dan sering dilakukan (Santoso, 2010 : 79).

Pada pengujian eksploratori pengguna dapat melakukan pengujian tanpa persiapan dan penjadwalan khusus sebelumnya. Pengujian ini tidak memerlukan dokumen khusus yang biasa digunakan untuk menentukan langkah-langkah tertentu yang harus dilakukan pada pengujian. Pada pengujian eksploratori pengguna dapat

mempelajari aplikasi dengan cara menjalankan aplikasi yang tersedia. Selanjutnya pengguna dapat melakukan pengujian sesuai panduan sistem.

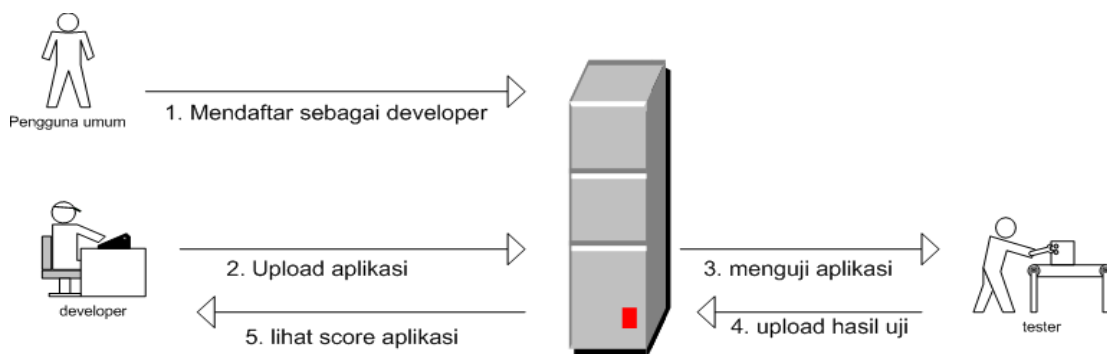
Penggunaan metode *exploratory testing* dapat membantu pengembang dalam melakukan pengujian terhadap aplikasi *flash* yang dikembangkan. *Tester* atau penguji dapat menguji aplikasi tanpa harus memasang aplikasi yang akan diuji pada komputer pribadinya. Sehingga pengujian dapat dilakukan oleh banyak penguji dalam satu waktu. Penguji juga diarahkan mengenai item yang harus diuji sehingga memudahkan dalam pelaporan hasil uji.

BAB III METODOLOGI

Pada penelitian ini akan dikembangkan sistem yang ditujukan untuk pengujian aplikasi flash yang dinamakan dengan TIMEFLASH. Aplikasi flash yang diuji berupa game dan alat bantu ajar dengan format single file.

Pengujian dilakukan dengan metode exploratory testing. Dimana tester dapat menguji aplikasi dengan bebas menggunakannya terlebih dahulu. Kemudian sistem akan menyediakan pertanyaan yang dapat diisi oleh tester. Pengujian aplikasi flash dengan metode eksploratori menggunakan sistem ini akan lebih mudah dilakukan dibandingkan dengan penerapan metode eksploratori secara manual. Secara manual tester harus mengunduh aplikasi flash terlebih dahulu kemudian mengisikan hasil pengujian pada form berbeda. Sedangkan pada sistem ini tester tidak perlu mengunduh aplikasi flash yang akan diuji dan kemudian tester dapat mengisikan hasil pengujian pada halaman yang sama dengan ketika menjalankan aplikasi sehingga memudahkan tester dalam memberi penilaian.

Terdapat banyak kebutuhan dalam pengembangan sistem. Kebutuhan disesuaikan dengan tujuan pengembangan sistem. Sebelum membahas mengenai kebutuhan dan perancangan sistem, akan dijabarkan mengenai gambaran umum untuk sistem. Gambaran umum sistem dapat dilihat pada gambar 3.1.



Gambar 3.1 Gambaran umum

Adapun penjelasan untuk gambar 3.1 adalah:

1. Pengguna umum mendaftarkan diri untuk menjadi anggota *developer* pada sistem timeflash.
2. Pengguna umum yang sudah mendaftar mempunyai hak akses di sistem sebagai *developer*. *Developer* dapat mengunggah aplikasi yang akan diuji.
3. Aplikasi yang telah diunggah, kemudian diuji oleh *tester*.
4. Setelah melakukan pengujian, *tester* mengunggah hasil pengujian ke sistem timeflash.
5. Hasil pengujian yang telah diolah akan ditampilkan kepada *developer*.
Hasil pengujian ini juga dapat dilihat oleh semua pengguna .

3.1 Analisis Kebutuhan

Analisis kebutuhan merupakan idenfikasi hal-hal yang diperlukan untuk menjalankan sistem. Analisis kebutuhan dibagi menjadi beberapa bagian, yaitu analisis kebutuhan *input*, proses dan *output*.

3.1.1 Analisis kebutuhan *input*

Kebutuhan input merupakan data yang dibutuhkan oleh sistem untuk menyelesaikan suatu pekerjaan. Adapun data *input* yang dibutuhkan oleh sistem ini dibagi menjadi 3 bagian berdasarkan pengguna sistem yaitu administrator, *developer* dan *tester*. Administrator merupakan pengguna yang menjadi penanggung jawab untuk mengurus sistem. *Developer* merupakan pengembang aplikasi *flash*. *Tester* merupakan penguji aplikasi. Adapun data *input* yang dibutuhkan sistem adalah:

1. Data *developer*
 - a. Pada saat pendaftaran data yang dibutuhkan dari *developer* adalah nama, alamat *email*, instansi, *password*, tanggal lahir, dan foto.
 - b. Data yang dibutuhkan dari *developer* pada proses *upload* aplikasi adalah Judul aplikasi, jenis dan *file* yang akan diunggah.
 - c. Data yang juga dibutuhkan dari *developer* adalah isi pesan, data ini dibutuhkan ketika *developer* ingin mengirim pesan kepada administrator.

2. Data administrator
 - a. Data yang dibutuhkan dari administrator berupa pertanyaan. Pertanyaan ini akan ditampilkan pada proses pengujian aplikasi.
 - b. Isi pesan merupakan data yang dibutuhkan dari administrator untuk memberi pesan kepada *developer*.
3. Data *tester*
 - a. Data yang dibutuhkan dari *tester* berupa nama, *email* dan jawaban hasil uji aplikasi dari pertanyaan pada proses pengujian.
 - b. Pada saat mengirim pesan, *tester* perlu memberikan data berupa nama, alamat *email* dan isi pesan.

3.1.2 Analisis kebutuhan proses

Kebutuhan proses merupakan suatu aktifitas yang perlu dilakukan sistem untuk mencapai suatu keluaran. Adapun kebutuhan proses pada sistem ini adalah:

1. Proses pendaftaran

Proses pendaftaran merupakan proses untuk melakukan registrasi bagi *tester* untuk bisa terdaftar sebagai *developer* di sistem timeflash.
2. Proses *login*

Proses *login* dilakukan oleh admin dan *developer* untuk dapat mengakses halaman timeflash.
3. Proses unggah aplikasi

Dibutuhkan proses unggah aplikasi untuk mengunggah aplikasi *flash* yang akan diuji. proses ini dilakukan oleh *developer*.
4. Proses pengujian aplikasi

Proses pengujian aplikasi merupakan proses yang dilakukan *tester* dalam menguji aplikasi *flash*.
5. Proses penyimpanan hasil uji aplikasi

Proses penyimpanan aplikasi merupakan proses yang dilakukan oleh sistem untuk menyimpan data hasil uji aplikasi *flash*.

6. Proses menampilkan hasil uji aplikasi

Proses menampilkan hasil uji aplikasi adalah proses yang dibutuhkan untuk menampilkan hasil dari pengujian aplikasi *flash*. Proses ini mencakup proses perhitungan total nilai aplikasi berdasarkan hasil penilaian *tester*.

7. Proses menampilkan statistik aplikasi

Aplikasi yang telah diuji dapat dibandingkan nilainya dengan aplikasi lainnya. Perbandingan ini akan lebih mudah dilihat dalam bentuk diagram yang ditampilkan melalui proses menampilkan statistik aplikasi.

3.1.3 Analisis kebutuhan *output*

Kebutuhan *output* merupakan informasi yang diharapkan dapat ditampilkan oleh sistem. Adapun kebutuhan *output* yang dibutuhkan adalah:

1. Daftar aplikasi *developer*

Output berupa daftar dari semua aplikasi yang telah diunggah oleh *developer*.

2. Daftar semua aplikasi

Semua pengguna sistem dapat melihat seluruh aplikasi yang telah diunggah pada sistem pengujian ini.

3. Detail aplikasi

Dari daftar aplikasi yang ditampilkan dapat dilihat detail atau keterangan lebih rinci dari aplikasi.

4. *Score* hasil uji aplikasi

Output ini menampilkan *score* yang didapat oleh aplikasi setelah dilakukan pengujian.

5. Statistik hasil uji aplikasi

Output ini menampilkan *score* dari aplikasi dalam bentuk statistik sehingga lebih terlihat perbedaan *score* dari beberapa aplikasi.

6. Profil *developer*

Semua pengguna sistem dapat melihat informasi mengenai *developer* pada profil masing-masing *developer*.

3.2 Perancangan Sistem

3.2.1 Kebutuhan Sistem

1. Kebutuhan fungsionalitas

Fungsionalitas yang dilakukan oleh sistem ini adalah:

a. Tambah *developer*

Fungsionalitas ini digunakan oleh *tester* melalui proses *sign up* atau pendaftaran. *Tester* akan terdaftar sebagai *developer* setelah mendaftarkan diri di timeflash.

b. *Edit data developer*

Fungsionalitas yang berfungsi untuk mengubah data *developer* ini dapat digunakan oleh admin ataupun *developer*.

c. *Edit password*

Fungsionalitas ini dapat digunakan oleh admin ataupun *developer*. Fungsionalitas ini digunakan untuk mengubah *password*.

d. *Reset password developer*

Fungsionalitas ini digunakan oleh admin untuk mengatur ulang *password developer*.

e. Hapus data *developer*

Fungsionalitas ini digunakan untuk menghapus data *developer* yang dapat digunakan oleh admin ataupun *developer*.

f. *Upload aplikasi*

Fungsionalitas ini digunakan oleh *developer* untuk mengupload aplikasi yang akan diuji.

g. *Edit data aplikasi*

Fungsionalitas ini dapat digunakan oleh admin ataupun *developer*.
Fungsionalitas ini digunakan untuk mengubah data aplikasi.

h. Hapus aplikasi

Fungsionalitas ini digunakan oleh admin dan *developer* untuk menghapus data aplikasi.

i. Lihat detail aplikasi

Fungsionalitas ini digunakan oleh admin, *developer* dan *tester* untuk melihat aplikasi. Fungsionalitas ini menampilkan aplikasi beserta informasi mengenai aplikasi tersebut.

j. Lihat statistik aplikasi

Fungsionalitas ini menampilkan statistik aplikasi berdasarkan *score* yang didapat. Digunakan oleh admin, *developer* dan *tester*.

k. Uji aplikasi

Fungsionalitas ini digunakan oleh *tester* untuk menguji aplikasi.

l. Kirim pesan

Fungsionalitas ini dapat digunakan oleh admin, *developer* dan *tester* untuk mengirim pesan.

m. Lihat pesan

Fungsionalitas ini dapat digunakan oleh admin dan *developer* untuk \ melihat pesan yang ditujukan kepada mereka.

n. Sistem melakukan autentifikasi saat pengguna mengakses halaman admin dan *developer*.

2. Kebutuhan non-fungsionalitas

a. Sistem menampilkan notifikasi setelah pengguna berhasil melakukan suatu *task*.

b. Sistem memberi notifikasi terhadap kesalahan pengguna dalam memasukkan data.

3.2.2 Perilaku sistem

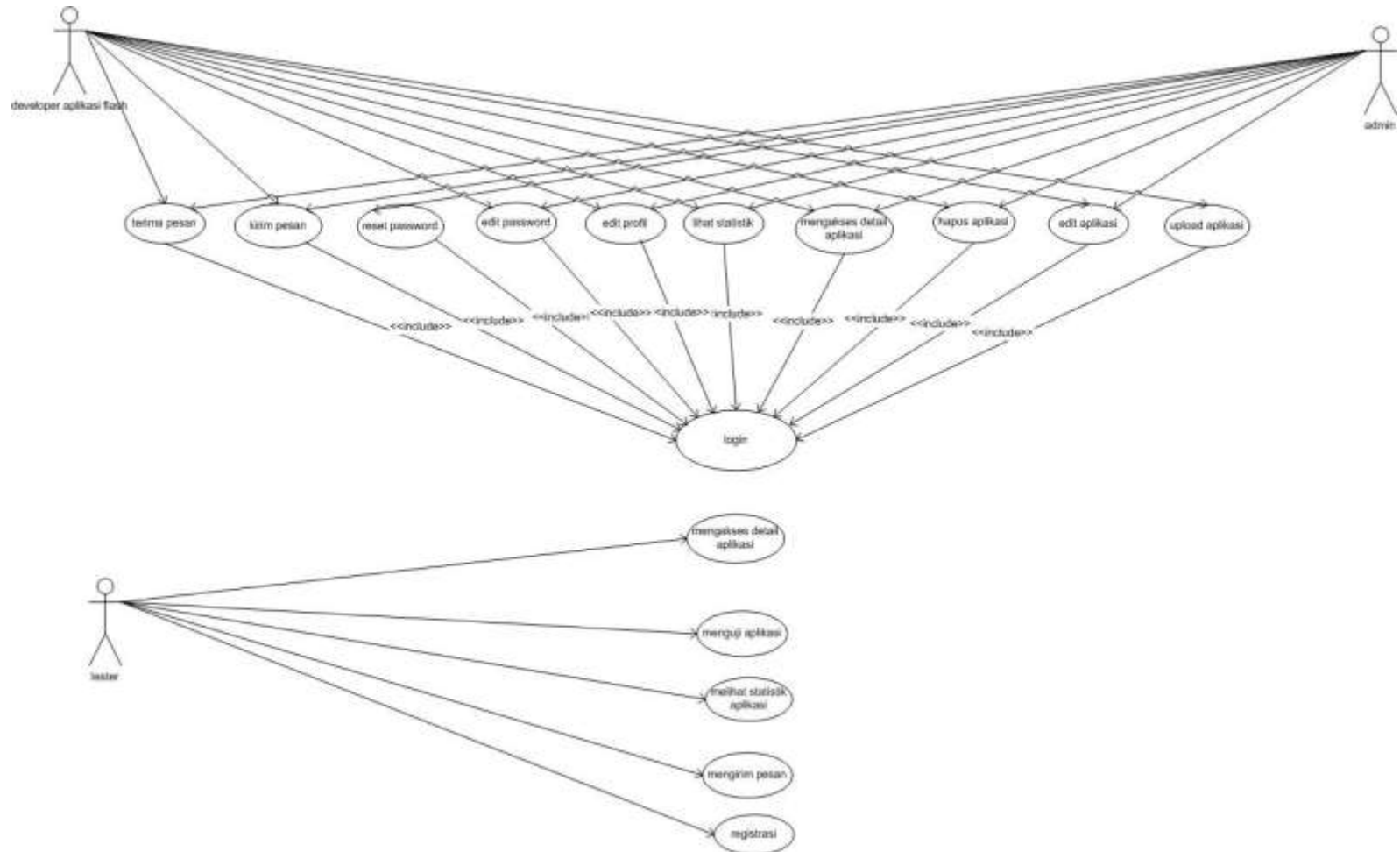
1. Diagram use case

Use case merupakan diagram yang menggambarkan interaksi antara sistem dan pengguna. Pada diagram *use case* dapat diketahui apa yang dapat dilakukan pengguna terhadap sistem. Gambar 3.2 Merupakan *use case* yang terdiri dari 3 aktor yaitu *developer* aplikasi flash, admin dan *tester*.

Admin dapat melakukan beberapa aktivitas yaitu menghapus aplikasi, *edit* aplikasi, mengakses semua aplikasi, melihat statistik aplikasi, *edit* profil *developer*, *reset password developer*, mengirim pesan kepada *developer*, dan menerima pesan. Semua aktivitas yang dilakukan oleh admin diharuskan untuk melakukan proses *login* terlebih dahulu.

Developer aplikasi *flash* dapat melakukan aktivitas berikut, yaitu *upload* aplikasi, mengubah dan menghapus aplikasi yang telah diunggah, menjalankan dan mengakses semua aplikasi, melihat statistik semua aplikasi, mengubah profil, mengubah *password*, mengirim dan menerima pesan. Semua aktivitas yang dilakukan oleh *developer* aplikasi *flash* diharuskan untuk melakukan proses *login* terlebih dahulu.

Sedangkan pengguna umum dapat melakukan pendaftaran, mengakses detail semua aplikasi, menguji aplikasi, melihat statistik aplikasi dan mengirim pesan.



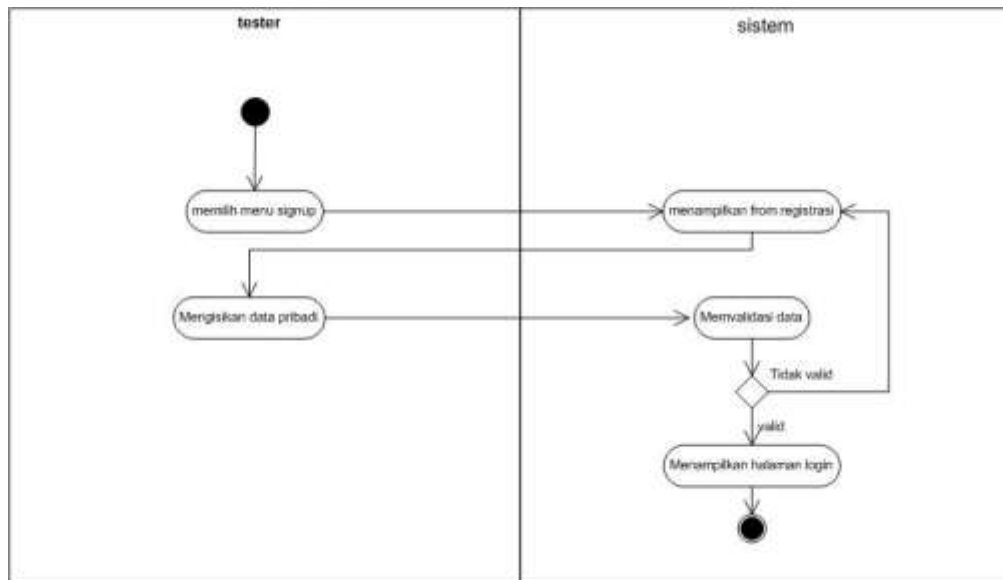
Gambar 3.2 *Use case*

2. Diagram Activity

Diagram *activity* merupakan diagram yang menggambarkan alur kerja sistem. Pada diagram *activity* dapat diketahui urutan kerja sistem untuk melakukan suatu pekerjaan.

a. Diagram *activity* proses registrasi

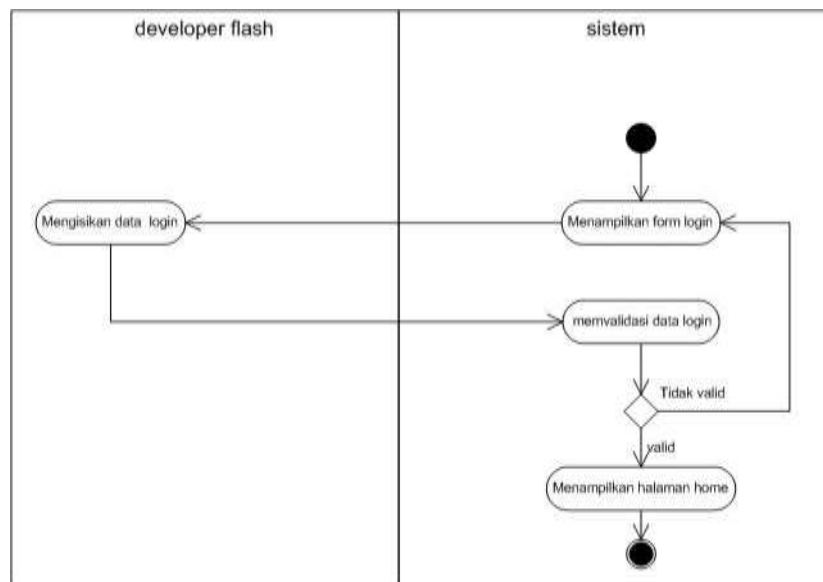
Registrasi merupakan langkah awal *developer* untuk mendapat akses sebagai *developer*. *Tester* mendaftarkan diri, setelah terdaftar *tester* akan mendapat akses sebagai seorang *developer*. Adapun diagram *activity* untuk proses registrasi dapat dilihat pada gambar 3. 3



Gambar 3.3 Diagram *activity* proses registrasi

b. Diagram *activity* proses login

Developer ataupun admin harus melakukan proses login untuk masuk ke akun timeflash. Proses login meminta pengguna untuk memasukkan alamat *email* dan *password*. Gambar 3.4 menjelaskan diagram *activity* untuk proses login.



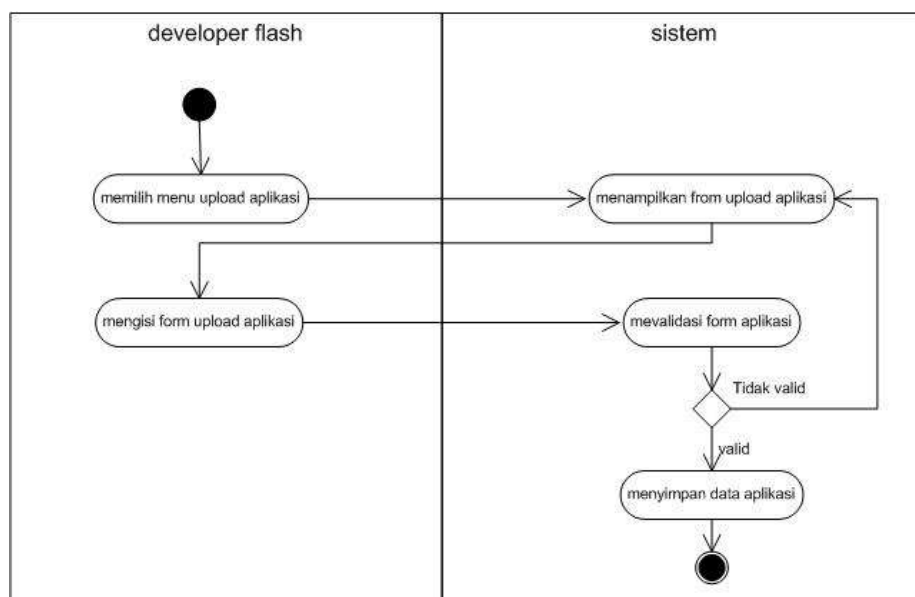
Gambar 3.4 Diagram *activity* proses login

c. Diagram *activity* upload aplikasi

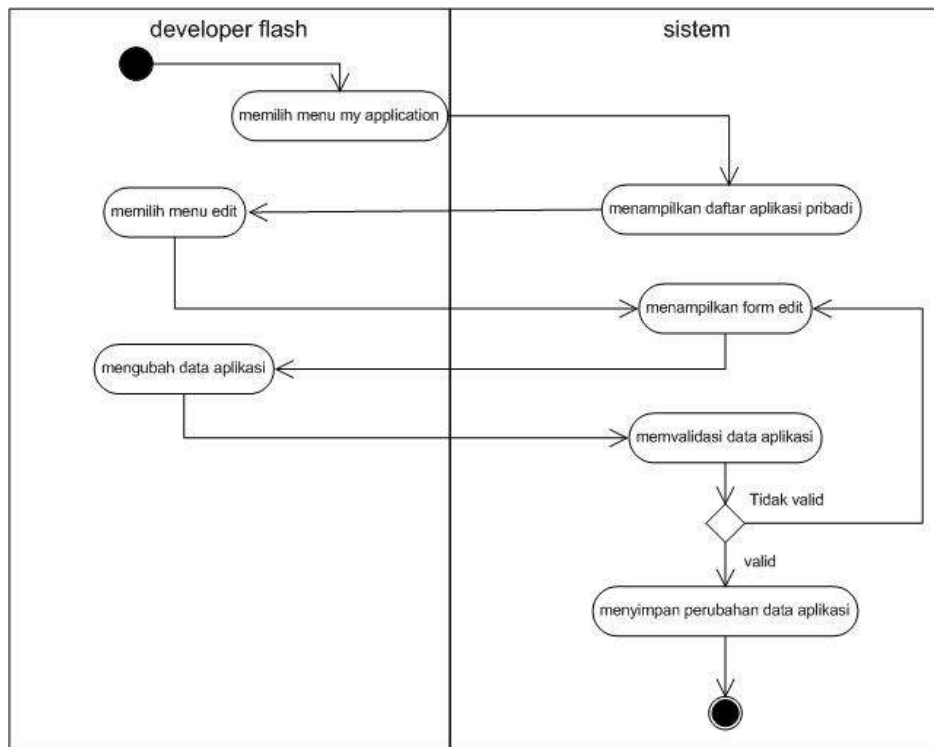
Developer yang telah melakukan proses login dapat mengunggah aplikasi *flash* yang ingin diuji pada sistem timeflash. Adapun alur untuk *upload* aplikasi dapat dilihat pada gambar 3.5.

d. Diagram *activity* edit aplikasi

Beberapa data terkait aplikasi yang telah diunggah oleh *developer* dapat diubah kembali. Aktivitas ini dapat dilakukan setelah pengguna melakukan proses login. Adapun diagram *activity* untuk *edit* aplikasi dapat dilihat pada gambar 3.6.



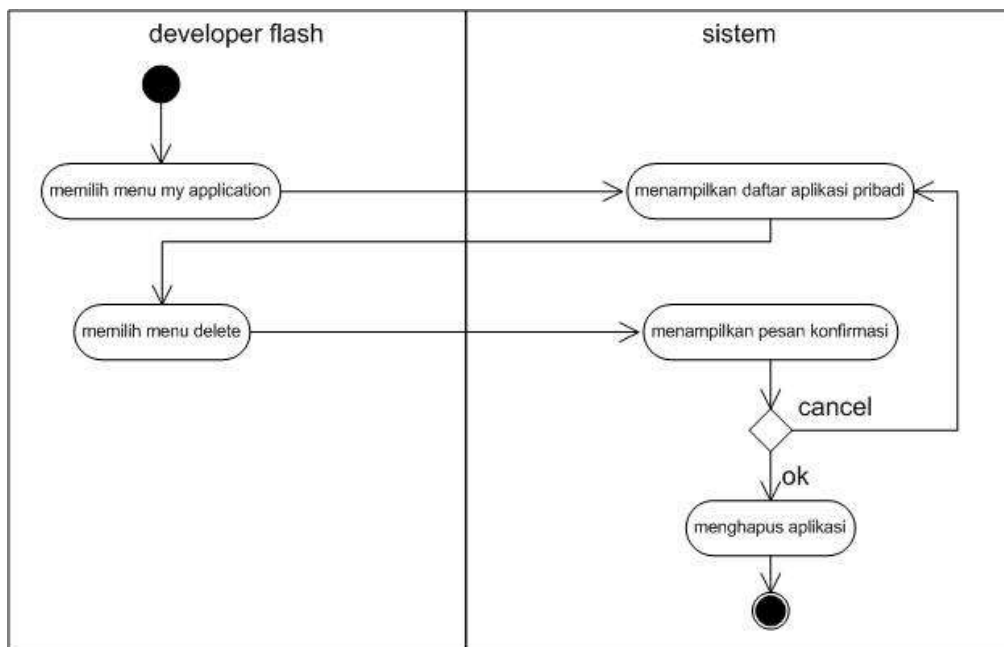
Gambar 3.5 Diagram *activity upload* aplikasi



Gambar 3.6 Diagram *activity edit* aplikasi

e. Diagram *activity* hapus aplikasi

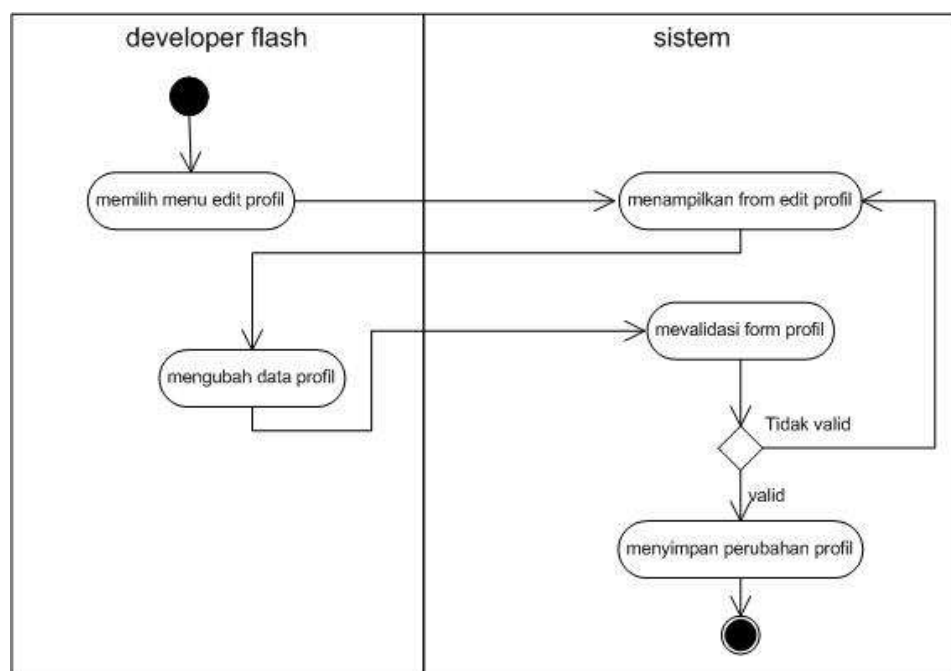
Aplikasi yang telah diunggah juga dapat dihapus jika diinginkan. Aktivitas ini dapat dilakukan setelah pengguna melakukan proses login. Adapun proses hapus aplikasi dapat dilihat pada gambar 3.7.



Gambar 3.7 Diagram *activity* hapus aplikasi

f. Diagram *activity* edit profil

Pada diagram *activity edit* profil, *developer* dapat mengubah data pribadinya seperti nama dan instansi. Aktivitas ini dapat dilakukan setelah pengguna melakukan proses *login*. Gambar lebih detail untuk proses *edit* profil dapat dilihat pada gambar 3.8.

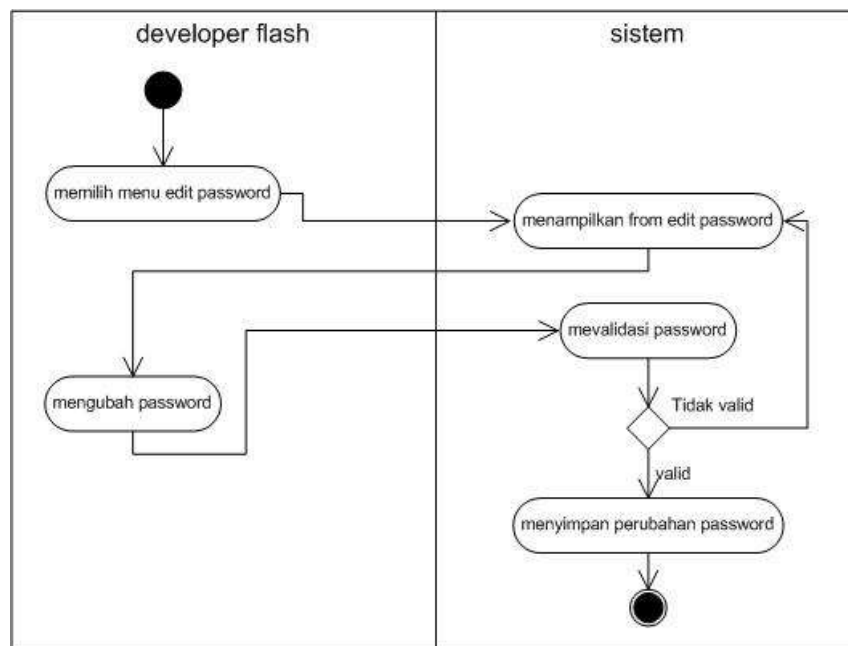


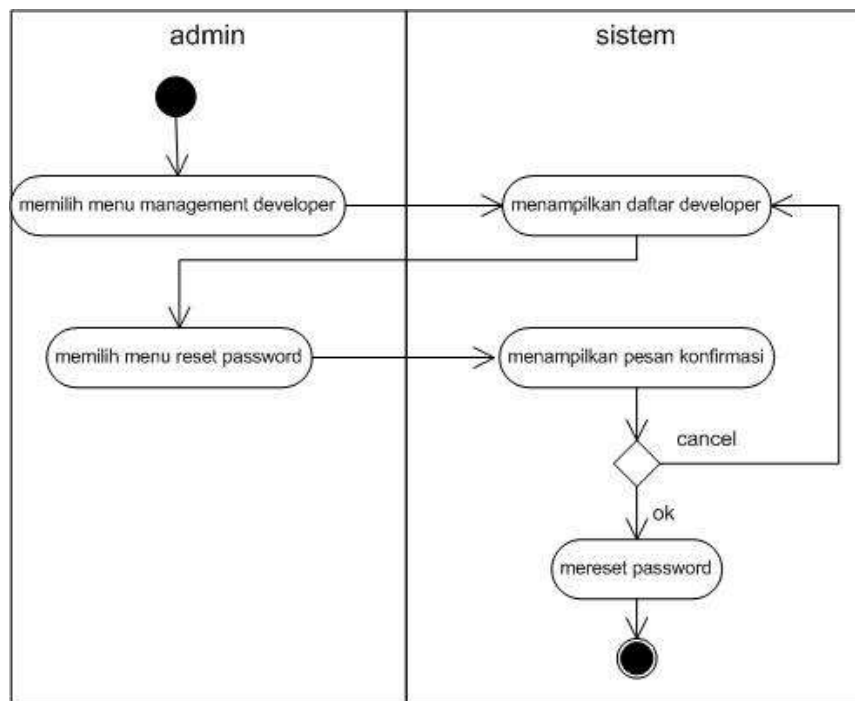
Gambar 3.8 Diagram *activity edit profil***g. Diagram *activity edit password***

Pada diagram *activity edit password*, *developer* dapat mebugbah *password* dengan memasukkan *password* lama, *password* baru kemudian mengetik ulang *password* baru. Aktifitas ini dapat dilakukan setelah pengguna melakukan proses login .Diagram *activity edit password* dapat dilihat pada gambar 3.9.

h. Diagram *activity reset password*

Aktifitas ini dapat dilakukan setelah pengguna melakukan proses login. Admin dapat melakukan *reset password developer*. Diagram *activity* untuk *reset password* terdapat pada gambar 3.10.

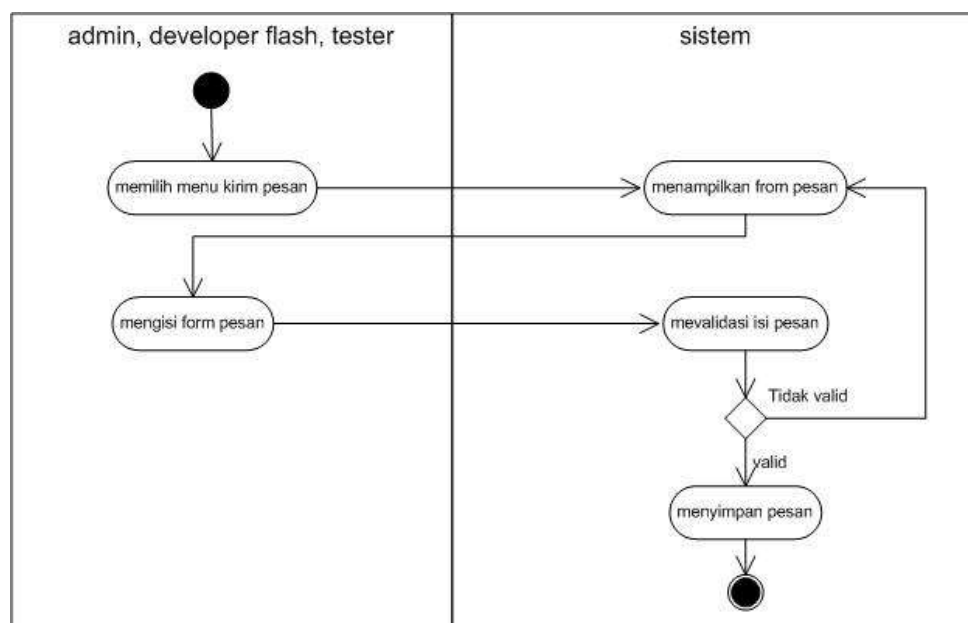
**Gambar 3.9** Diagram *activity edit password*



Gambar 3.10 Diagram *activity reset password*

i. Diagram *activity* kirim pesan

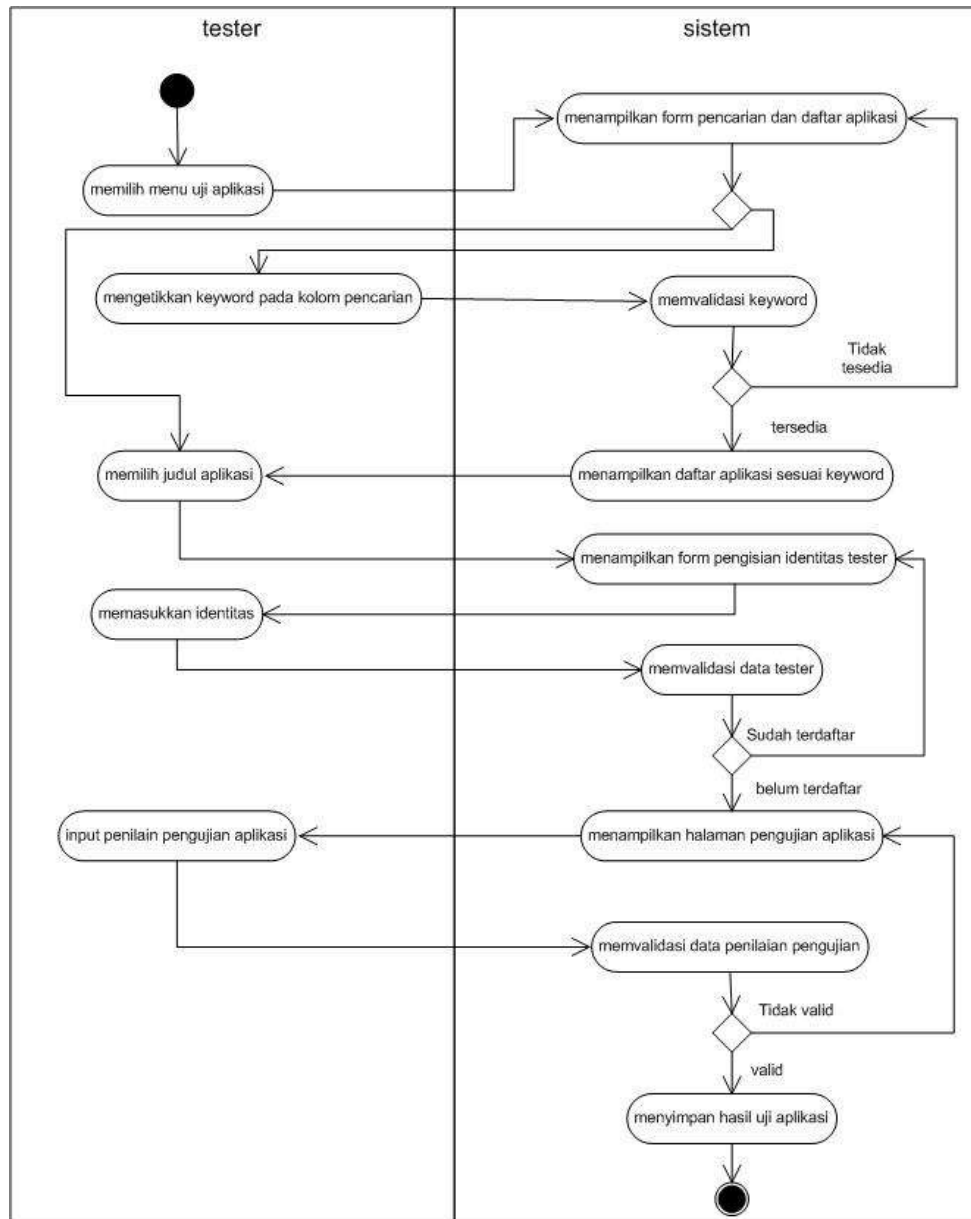
Pada aktivitas kirim pesan, baik admin, *developer* atau *tester* dapat mengirim pesan. Urutan aktivitas untuk kirim pesan dapat dilihat pada gambar 3.11.



Gambar 3.11 Diagram *activity* kirim pesan

j. Diagram *activity* uji aplikasi

Aktivitas ini dilakukan oleh *tester*. *Tester* menguji aplikasi dengan menjalankannya lalu menjawab pertanyaan terkait aplikasi yang telah dijalankan. Jawaban *tester* akan dihitung sebagai *score* untuk aplikasi yang diuji. Alur lebih lanjut mengenai uji aplikasi dapat dilihat pada gambar 3.12.



Gambar 3.12 Diagram uji aplikasi

3.2.3 Perancangan pertanyaan

3.2.4 Perancangan basis data

Sistem membutuhkan tempat untuk menyimpan data. Perancangan basis data merupakan perancangan tabel dan relasinya yang digunakan untuk menyimpan data pada sistem.

1. Struktur tabel

Pada sistem timeflash dibutuhkan 8 tabel untuk menyimpan data. Masing-masing tabel mempunyai atribut yang berbeda sesuai kebutuhan.

a. Tabel *aplikasi*

Tabel *aplikasi* digunakan untuk menyimpan data aplikasi. Tabel *aplikasi* berisi *id_app*, *judul*, *tgl_upload*, *app*, *id_jenis* dan *id_user*. Detail untuk struktur tabel *aplikasi* dapat dilihat pada tabel 3.1.

Tabel 3.1 Tabel *aplikasi*

Nama	Type	Keterangan
<i>Id_app</i>	<i>Int (3)</i>	<i>Primary key</i>
<i>Judul</i>	<i>Varchar (65)</i>	
<i>Tgl_upload</i>	<i>Date</i>	
<i>App</i>	<i>Varchar (33)</i>	
<i>Id_jenis</i>	<i>Int (1)</i>	<i>Foreign key</i>
<i>Id_user</i>	<i>Int (3)</i>	<i>Foreign key</i>

b. Tabel *aspek*

Tabel *aspek* digunakan untuk menyimpan data aspek-aspek *usability* untuk pertanyaan pengujian. Tabel *aspek* berisi *id_aspek*, *nama_aspek* dan *bobot_aspek*. Detail untuk struktur tabel *aspek* dapat dilihat pada tabel 3.2

Tabel 3.2 Tabel *aspek*

Nama	Type	Keterangan
<i>Id_aspek</i>	<i>Int (1)</i>	<i>Primary key</i>
<i>Nama_aspek</i>	<i>Varchar (15)</i>	
<i>Bobot_aspek</i>	<i>float</i>	

c. Tabel *user*

Tabel *user* digunakan untuk menyimpan data pengguna. Tabel *user* berisi *id_user*, *nama*, *email*, *instansi*, *password*, *tgl_lahir*, *status* dan *foto*. Detail untuk struktur tabel *user* dapat dilihat pada tabel 3.3

Tabel 3.3 Tabel *user*

Nama	Type	Keterangan
<i>Id_user</i>	<i>Int (3)</i>	<i>Primary key</i>
<i>nama</i>	<i>Varchar (33)</i>	
<i>Email</i>	<i>Varchar (33)</i>	
<i>Instansi</i>	<i>Varchar (55)</i>	
<i>Password</i>	<i>Varchar (20)</i>	
<i>Tgl_lahir</i>	<i>Date</i>	
<i>status</i>	<i>Varchar (9)</i>	
<i>foto</i>	<i>Varchar (30)</i>	

d. Tabel *hasil*

Tabel *hasil* digunakan untuk menyimpan data hasil pengujian. Tabel *hasil* berisi *id_hasil*, *id_pertanyaan*, *id_aplikasi*, *score* dan *id_tester*. Detail untuk struktur tabel *hasil* dapat dilihat pada tabel 3.4

Tabel 3.4 Tabel *hasil*

Nama	Type	Keterangan
<i>Id_hasil</i>	<i>Int(5)</i>	<i>Primary key</i>
<i>Id_pertanyaan</i>	<i>Int(3)</i>	<i>Foreign key</i>
<i>Id_aplikasi</i>	<i>Int(3)</i>	<i>Foreign key</i>
<i>Score</i>	<i>Float</i>	
<i>Id_tester</i>	<i>Int(4)</i>	<i>Foreign key</i>

e. Tabel *jenis*

Tabel *jenis* digunakan untuk menyimpan data berupa jenis dari aplikasi dan pertanyaan. Tabel *jenis* berisi *id_jenis* dan *nama_jenis*. Detail untuk struktur tabel *jenis* dapat dilihat pada tabel 3.5

Tabel 3.5 Tabel *jenis*

Nama	Type	Keterangan
<i>Id_jenis</i>	<i>Int(2)</i>	<i>Primary key</i>
<i>Nama_jenis</i>	<i>Varchar (15)</i>	

f. Tabel pertanyaan

Tabel pertanyaan digunakan untuk menyimpan data terkait pertanyaan pengujian. Tabel *pertanyaan* berisi *id_pertanyaan*, *isi_pertanyaan*, *id_aspek* dan *id_jenis*. Detail untuk struktur tabel *pertanyaan* dapat dilihat pada tabel 3.6.

Tabel 3.6 Tabel *pertanyaan*

Nama	Type	Keterangan
<i>Id_pertanyaan</i>	<i>Int(3)</i>	<i>Primary key</i>
<i>Isi_pertanyaan</i>	<i>Varchar (50)</i>	
<i>Id_aspek</i>	<i>Int (1)</i>	<i>Foreign key</i>
<i>Id_jenis</i>	<i>Int (1)</i>	<i>Foreign key</i>

g. Tabel pesan

Tabel *pesan* digunakan untuk menyimpan data pesan. Tabel *pesan* berisi *id_pesan*, *isi_pesan*, *pengirim*, *tujuan*, *email*, *status*, dan *tgl_kirim*. Detail untuk struktur tabel *pesan* dapat dilihat pada tabel 3.7

Tabel 3.7 Tabel *pesan*

Nama	Type	Keterangan
<i>Id_pesan</i>	<i>Int(3)</i>	<i>Primary key</i>
<i>Isi_pesan</i>	<i>Text</i>	
<i>Pengirim</i>	<i>Varchar(33)</i>	
<i>Tujuan</i>	<i>Varchar (33)</i>	
<i>Status</i>	<i>Int(1)</i>	
<i>Email</i>	<i>varchar</i>	
<i>Tgl_kirim</i>	<i>date</i>	

h. Tabel tester

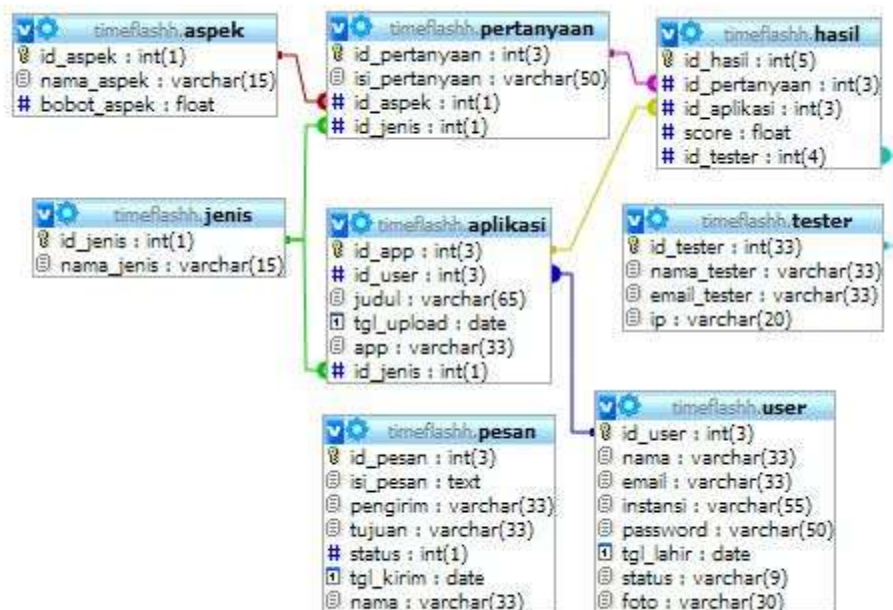
Tabel *tester* digunakan untuk menyimpan data *tester*. Tabel *tester* berisi *id_tester*, *nama_tester*, *email_tester* dan *ip*. Detail untuk struktur tabel *tester* dapat dilihat pada tabel 3.8.

Tabel 3.8 Tabel *tester*

Nama	Type	Keterangan
<i>Id_tester</i>	<i>Int(5)</i>	<i>Primary key</i>
<i>Nama_tester</i>	<i>Varchar(33)</i>	
<i>Email_tester</i>	<i>Varchar (33)</i>	
<i>ip</i>	<i>Varchar (20)</i>	

2. Entity Relationship Diagram (ERD)

ERD merupakan diagram yang menggambarkan relasi antar entitas. ERD untuk sistem timeflash dapat dilihat pada gambar 3.13.



Gambar 3.13 ERD

3.2.5 Perancangan antarmuka

Rancangan antarmuka merupakan gambaran dari tampilan halaman *website* yang akan digunakan. Berikut ini merupakan rancangan antarmuka untuk sistem timeflash.

1. Rancangan antarmuka untuk admin dan *developer*

a. Rancangan antarmuka halaman *login*

Halaman *login* merupakan halaman utama bagi admin dan *developer* untuk dapat mengakses halaman lainnya. Sebelum masuk ke halaman *home* misalnya, admin atau *developer* harus melakukan proses *login* terlebih dahulu. Gambar 3.14 Merupakan rancangan antarmuka untuk halaman *login*.

Gambar 3.14 Rancangan antarmuka *login*

2. Rancangan antarmuka untuk admin

a. Rancangan antarmuka *home*

Halaman *home* merupakan halaman yang pertama tampil setelah admin melakukan proses *login*. Gambar untuk rancangan antarmuka halaman *login* dapat dilihat pada gambar 3.15.

Logo	pesan Hai, admin
Home	Location
Management developer	Home
Management aplikasi	Logo timeflash Paragraf 1 ini adalah paragraf pertama Paragraf 1 ini adalah paragraf pertama Totol aplikasi yang telah diunggah di timeflash
View statistik	
Management pertanyaan	
Change password	
Logout	

Gambar 3.15 Rancangan antarmuka *home admin***b. Rancangan antarmuka *management developer***

Rancangan antarmuka untuk *management developer* berisi informasi mengenai *developer* aplikasi *flash*. Informasi yang ditampilkan berupa nama, *email*, instansi, dan total aplikasi yang telah diunggah. Pada halaman ini juga disediakan fitur untuk melihat detail informasi *developer*, *edit profil developer*, *reset password* dan *delete developer*. Gambar untuk rancangan antarmuka halaman *management developer* dapat dilihat pada gambar 3.16.

Logo	 																																																													
Home	Location																																																													
Management developer	Management developer																																																													
Management aplikasi	<table border="1"> <thead> <tr> <th>No</th> <th>nama</th> <th>email</th> <th>instansi</th> <th>Total aplikasi</th> <th>view</th> <th>edit</th> <th>Reset password</th> <th>delete</th> </tr> </thead> <tbody> <tr><td> </td><td> </td><td> </td><td> </td><td> </td><td> </td><td> </td><td> </td><td> </td></tr> <tr><td> </td><td> </td><td> </td><td> </td><td> </td><td> </td><td> </td><td> </td><td> </td></tr> <tr><td> </td><td> </td><td> </td><td> </td><td> </td><td> </td><td> </td><td> </td><td> </td></tr> <tr><td> </td><td> </td><td> </td><td> </td><td> </td><td> </td><td> </td><td> </td><td> </td></tr> <tr><td> </td><td> </td><td> </td><td> </td><td> </td><td> </td><td> </td><td> </td><td> </td></tr> </tbody> </table>								No	nama	email	instansi	Total aplikasi	view	edit	Reset password	delete																																													
No									nama	email	instansi	Total aplikasi	view	edit	Reset password	delete																																														
View statistik:																																																														
Management pertanyaan																																																														
Change password																																																														
Logout																																																														

Gambar 3.16 Rancangan antarmuka *management developer***c. Rancangan antarmuka *view developer***

Rancangan antarmuka *view developer* berisi informasi mengenai *developer*. Informasi berupa foto, biodata dan total aplikasi yang telah diunggah *developer*. Gambar 3.17 merupakan gambar untuk rancangan antarmuka *view developer*.

Logo	pesan Hal, admin
Home	Location
Management developer	Profil
Management aplikasi	foto Biodata developer Total aplikasi
View statistik	
Management pertanyaan	
Change password	
Logout	

Gambar 3.17 Rancangan antarmuka *view developer*

d. Rancangan antarmuka *edit developer*

Gambar 3.18 merupakan rancangan antarmuka untuk *edit developer*. Data *developer* yang dapat diubah adalah nama, *email*, instansi dan foto profil.

Logo	pesan Hal, admin
Home	Location
Management developer	edit profil
Management aplikasi	nama email instansi foto save cancel
View statistik	
Management pertanyaan	
Change password	
Logout	

Gambar 3.18 Rancangan antarmuka *edit developer*

e. Rancangan antarmuka *management aplikasi game*

Rancangan antarmuka *management* aplikasi *game* berisi informasi tentang daftar aplikasi dengan jenis *game*. Informasi yang ditampilkan adalah nama aplikasi, tanggal *upload*, nama

developer, instansi dan total score hasil pengujian. Rancangan antarmuka untuk *management* aplikasi *game* dapat dilihat pada gambar 3.19.

Logo	pesan Hai, admin																																																																																																		
Home	Location																																																																																																		
Management developer	List aplikasi game																																																																																																		
Management aplikasi	cari																																																																																																		
Aplikasi game	<table border="1"> <thead> <tr> <th>No</th> <th>name</th> <th>Tanggal upload</th> <th>developer</th> <th>instansi</th> <th>Total score</th> <th>view</th> <th>edit</th> <th>delete</th> </tr> </thead> <tbody> <tr><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr> <tr><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr> <tr><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr> <tr><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr> <tr><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr> <tr><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr> <tr><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr> <tr><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr> <tr><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr> </tbody> </table>									No	name	Tanggal upload	developer	instansi	Total score	view	edit	delete																																																																																	
No	name	Tanggal upload	developer	instansi	Total score	view	edit	delete																																																																																											
Aplikasi alat bantu ajar																																																																																																			
View statistik																																																																																																			
Management pertanyaan																																																																																																			
Change password																																																																																																			
Logout																																																																																																			

Gambar 3.19 Rancangan antarmuka *management* aplikasi *game*

f. Rancangan antarmuka *management* aplikasi alat bantu ajar

Rancangan antarmuka *management* aplikasi alat bantu ajar berisi informasi tentang daftar aplikasi dengan jenis alat bantu ajar. Informasi yang ditampilkan adalah nama aplikasi, tanggal *upload*, nama *developer*, instansi dan total score hasil pengujian. Rancangan antarmuka untuk *management* aplikasi alat bantu ajar dapat dilihat pada gambar 3.20.

Logo	pesan Hai, admin																																																																																																		
Home	Location																																																																																																		
Management developer	List aplikasi alat bantu ajar																																																																																																		
Management aplikasi	cari																																																																																																		
Aplikasi game	<table border="1"> <thead> <tr> <th>No</th> <th>name</th> <th>Tanggal upload</th> <th>developer</th> <th>instansi</th> <th>Total score</th> <th>view</th> <th>edit</th> <th>delete</th> </tr> </thead> <tbody> <tr><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr> <tr><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr> <tr><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr> <tr><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr> <tr><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr> <tr><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr> <tr><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr> <tr><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr> <tr><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr> </tbody> </table>									No	name	Tanggal upload	developer	instansi	Total score	view	edit	delete																																																																																	
No	name	Tanggal upload	developer	instansi	Total score	view	edit	delete																																																																																											
Aplikasi alat bantu ajar																																																																																																			
View statistik																																																																																																			
Management pertanyaan																																																																																																			
Change password																																																																																																			
Logout																																																																																																			

Gambar 3.20 Rancangan antarmuka *management* aplikasi alat bantu ajar

g. Rancangan antarmuka *view* aplikasi

Rancangan antarmuka *view* aplikasi menampilkan aplikasi, detail aplikasi dan *score* yang didapat aplikasi dari hasil pengujian. Gambar 3.21 merupakan rancangan antarmuka untuk *view* aplikasi.

h. Rancangan antarmuka *edit* aplikasi

Gambar 3.22 merupakan rancangan untuk antarmuka *edit* aplikasi. Halaman ini menampilkan *form* untuk mengubah data aplikasi.

Logo	pesan Hai, admin
Home	Location
Management developer	Detail aplikasi
Management aplikasi	aplikasi Detail aplikasi Score aplikasi
View statistik	
Management pertanyaan	
Change password	
Logout	

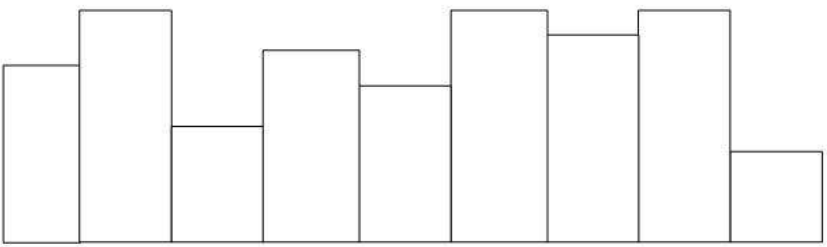
Gambar 3.21 Rancangan antarmuka *view* aplikasi

Logo	pesan Hai, admin
Home	Location
Management developer	edit aplikasi
Management aplikasi	nama Jenis aplikasi save cancel
View statistik	
Management pertanyaan	
Change password	
Logout	

Gambar 3.22 Rancangan antarmuka *edit* aplikasi

i. Rancangan antarmuka *view statistik aplikasi game*

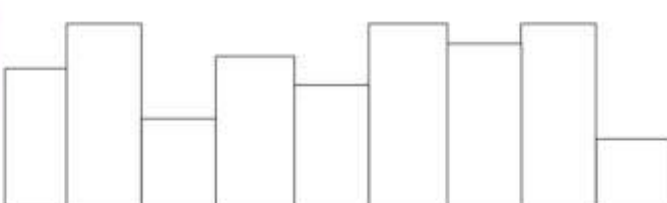
Rancangan antarmuka *view statistik* aplikasi *game* menampilkan statistik dari hasil pengujian aplikasi *game* dalam bentuk diagram batang. Gambar 3.23 merupakan rancangan antarmuka untuk *view statistik* aplikasi *game*.

Logo	pesan Hai, admin
Home	Location
Management developer	Statistik game
Management aplikasi	
View statistik	
Statistik game	
Statistik alat bantu ajar	
Management pertanyaan	
Change password	
Logout	

Gambar 3.23 Rancangan antarmuka *view statistik* aplikasi *game*

j. Rancangan antarmuka *view statistik* aplikasi alat bantu ajar

Rancangan antarmuka *view statistik* aplikasi alat bantu ajar menampilkan statistik dari hasil pengujian aplikasi alat bantu ajar dalam bentuk diagram batang. Gambar 3.24 merupakan rancangan antarmuka untuk *view statistik* aplikasi alat bantu ajar.

Logo	pesan Hai, admin
Home	Location
Management developer	Statistik alat bantu ajar
Management aplikasi	
View statistik	
Statistik game	
Statistik alat bantu ajar	
Management pertanyaan	
Change password	
Logout	

Gambar 3.24 Rancangan antarmuka *view statistik* aplikasi alat bantu ajar

k. Rancangan antarmuka *management* pertanyaan

Pada rancangan antarmuka untuk *management* pertanyaan menampilkan pertanyaan dan jenis pertanyaan. Di halaman ini disediakan fitur *edit*, *add* dan *delete* pertanyaan. Gambar 3.25 Menunjukkan rancangan antarmuka untuk *management* pertanyaan.

l. Rancangan antarmuka *edit* pertanyaan

Rancangan antarmuka *edit* pertanyaan menampilkan *form* untuk mengubah data pertanyaan. Gambar 3.26 merupakan rancangan antarmuka *edit* pertanyaan.

Logo	pesan Hai, admin																
Home	Location																
Management developer	Management pertanyaan																
Management aplikasi	Add pertanyaan																
View statistik	cari																
Management pertanyaan	<table border="1"> <thead> <tr> <th>No</th> <th>isi</th> <th>aspek</th> <th>jenis</th> <th>edit</th> <th>delete</th> </tr> </thead> <tbody> <tr> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> </tr> </tbody> </table>					No	isi	aspek	jenis	edit	delete						
No	isi	aspek	jenis	edit	delete												
Change password																	
Logout																	

Gambar 3.25 Rancangan antarmuka *management* pertanyaan

Logo	pesan Hai, admin
Home	Location
Management developer	edit pertanyaan
Management aplikasi	iai Aspek usability Jenis aplikasi save cancel
View statistik	
Management pertanyaan	
Change password	
Logout	

Gambar 3.26 Rancangan antarmuka *edit* pertanyaan**m. Rancangan antarmuka *change password***

Rancangan antarmuka *change password* menampilkan *form* untuk mengganti *password*. Admin diminta untuk mengisi *password* lama, *password* baru dan mengetikkan ulang *password* baru. Rancangan antarmuka untuk *change password* dapat dilihat pada gambar 3.27.

Logo	pesan Hai, admin
Home	Location
Management developer	Change password
Management aplikasi	Password lama Password baru Confirm password save cancel
View statistik	
Management pertanyaan	
Change password	
Logout	

Gambar 3.27 Rancangan antarmuka *change password*

3. Rancangan antarmuka untuk *developer*

a. Rancangan antarmuka *home*

Halaman *home* merupakan halaman yang pertama tampil setelah *developer* melakukan proses *login*. Rancangan untuk antarmuka halaman *home* dapat dilihat pada gambar 3.28.

Logo:	pesan Upload app Hai, nama
Home	Location:
My account	Home:
List aplikasi	Content home
View statistik	
About us	
Logout	

Gambar 3.28 Rancangan antarmuka *home*

b. Rancangan antarmuka *upload aplikasi*

Rancangan antarmuka *upload* aplikasi menampilkan *form* untuk mengunduh aplikasi. *Developer* diminta untuk mengisi data terkait aplikasi seperti judul aplikasi, jenis dan *file* aplikasi. Gambar 3.29 merupakan rancangan antarmuka untuk *upload* aplikasi.

Logo	pesan Upload app Hai, nama
Home	Location
My account	Upload aplikasi
List aplikasi	judul jenis file save cancel
View statistik	
About us	
Logout	

Gambar 3.29 Rancangan antarmuka *upload* aplikasi

c. Rancangan antarmuka *my profil*

Rancangan antarmuka *my profil* menampilkan informasi berupa biodata diri *developer*. Informasi yang ditampilkan berupa nama, instansi, tanggal lahir dan foto. Pada antarmuka ini juga ditampilkan total aplikasi yang telah diunggah oleh *developer*. Gambar 3.30 merupakan rancangan untuk antarmuka *my profil*.

Logo	pesan Upload app Hai, nama
Home	Location
My account	Profil
My profil	foto Biodata developer nama: example Tanggal lahir : 23/2/2013 email : ex@gmail.com Total aplikasi yang telah diunggah
My application	
List aplikasi	
View statistik	
About us	
Logout	

Gambar 3.30 Rancangan antarmuka *my profil*

d. Rancangan antarmuka *edit profil*

Rancangan antarmuka untuk *edit profil* menampilkan *form* untuk mengubah data diri *developer*. Gambar 3.31 merupakan rancangan untuk antarmuka *edit profil*.

Logo	pesan Upload app Hai, nama
Home	Location
My account	Edit profil
List aplikasi	nama email instansi foto save cancel
View statistik	
About us	
Logout	

Gambar 3.31 Rancangan antarmuka *edit profil*

e. Rancangan antarmuka *edit password*

Rancangan antarmuka untuk *edit password* menampilkan *form* untuk mengubah *password developer*. Gambar 3.32 merupakan rancangan untuk antarmuka *edit password*.

Logo	pesan Upload app Hai, nama
Home	Location
My account	Edit password
List aplikasi	Password lama Password baru Confirm pasword save cancel
View statistik	
About us	
Logout	

Gambar 3.32 Rancangan antarmuka *edit password*

f. Rancangan antarmuka *my application*

Rancangan antarmuka *my application* menampilkan daftar aplikasi yang telah diunggah oleh *developer*. Data aplikasi yang ditampilkan berupa nama, jenis, tanggal *upload*, dan total *score*. Gambar 3.33 merupakan rancangan untuk antarmuka *my application*.

Logo	pesan Upload app Hal, nama																
Home	Location																
My account	Profil																
My profil	carl																
My application	<table border="1"> <thead> <tr> <th>No</th> <th>Nama</th> <th>Jenis</th> <th>Tanggal</th> <th>Total score</th> <th>View</th> <th>Edit</th> <th>Delete</th> </tr> </thead> <tbody> <tr> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> </tr> </tbody> </table>	No	Nama	Jenis	Tanggal	Total score	View	Edit	Delete								
No	Nama	Jenis	Tanggal	Total score	View	Edit	Delete										
List aplikasi																	
View statistik																	
About us																	
Logout																	

Gambar 3.33 Rancangan antarmuka *my application*

g. Rancangan antarmuka *edit* aplikasi

Rancangan antarmuka untuk *edit* aplikasi menampilkan *form* untuk mengubah data aplikasi.

Gambar 3.34 merupakan rancangan untuk antarmuka *edit* aplikasi.

Logo	pesan Upload app Hal, nama
Home	Location
My account	Edit aplikasi
List aplikasi	judul jenis save cancel
View statistik	
About us	
Logout	

Gambar 3.34 Rancangan antarmuka *edit* aplikasi

h. Rancangan antarmuka *list* aplikasi *game*

Rancangan antarmuka *list* aplikasi *game* berisi informasi tentang daftar aplikasi dengan jenis *game*. Informasi yang ditampilkan adalah nama aplikasi, tanggal *upload*, nama *developer*,

Logo

pesan

Upload app

Hai, nama

Home

Location

My account

List aplikasi alat bantu ajar

List aplikasi

cari

Aplikasi game

No	Nama	Tanggal Upload	Developer	Instansi	Total score	View

Aplikasi alat bantu ajar

View statistik

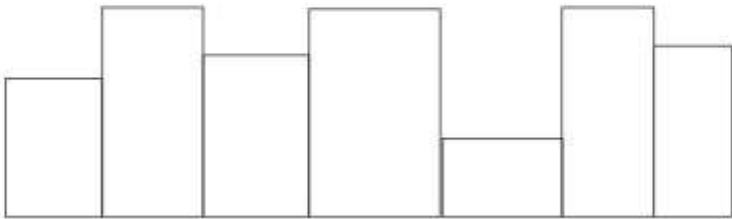
About us

Logout

Gambar 3.36 Rancangan antarmuka *list* aplikasi alat bantu ajar

j. Rancangan antarmuka *view statistik game*

Rancangan antarmuka *view statistik* aplikasi *game* menampilkan statistik dari hasil pengujian aplikasi *game* dalam bentuk diagram batang. Gambar 3.37 merupakan rancangan antarmuka untuk *view statistik* aplikasi *game*.

Logo	pesan Upload app Hai, nama
Home	Location
My account	Statistik game
List aplikasi	
View statistik	
Statistik game	
Statistik alat bantu ajar	
About us	
Logout	

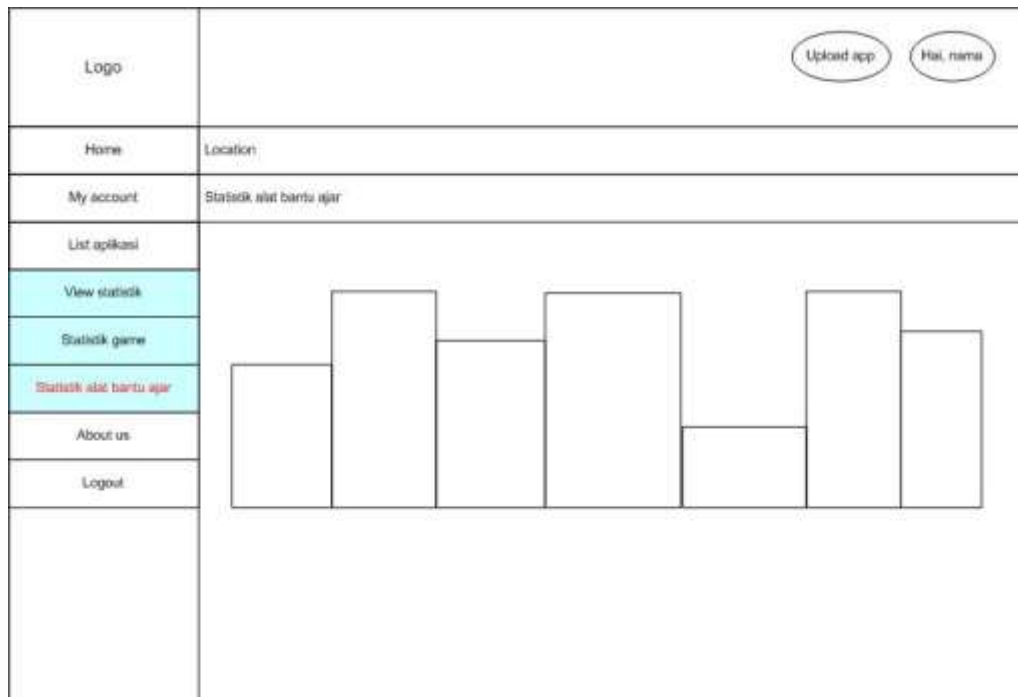
Gambar 3.37 Rancangan antarmuka *view statistik game*

k. Rancangan antarmuka *view statistik alat bantu ajar*

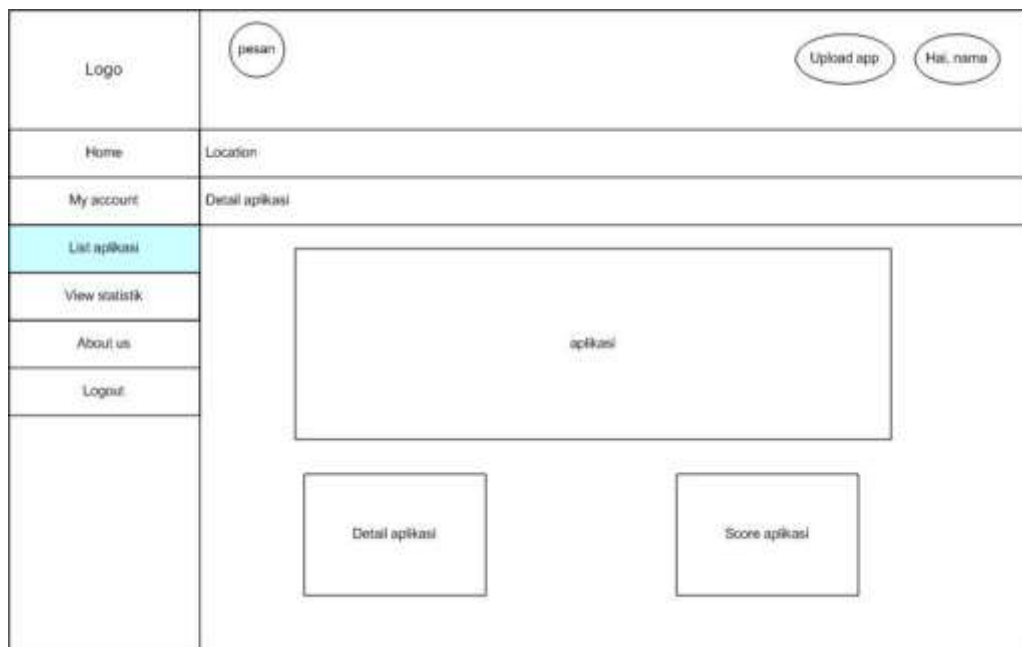
Rancangan antarmuka *view statistik* aplikasi alat bantu ajar menampilkan statistik dari hasil pengujian aplikasi alat bantu ajar dalam bentuk diagram batang. Gambar 3.38 merupakan rancangan antarmuka untuk *view statistik* aplikasi alat bantu ajar.

l. Rancangan antarmuka *view detail aplikasi*

Rancangan antarmuka *view detail* aplikasi menampilkan aplikasi, detail aplikasi dan *score* yang diperoleh aplikasi berdasarkan hasil pengujian. Gambar 3.39 merupakan rancangan antarmuka untuk *view detail* aplikasi.



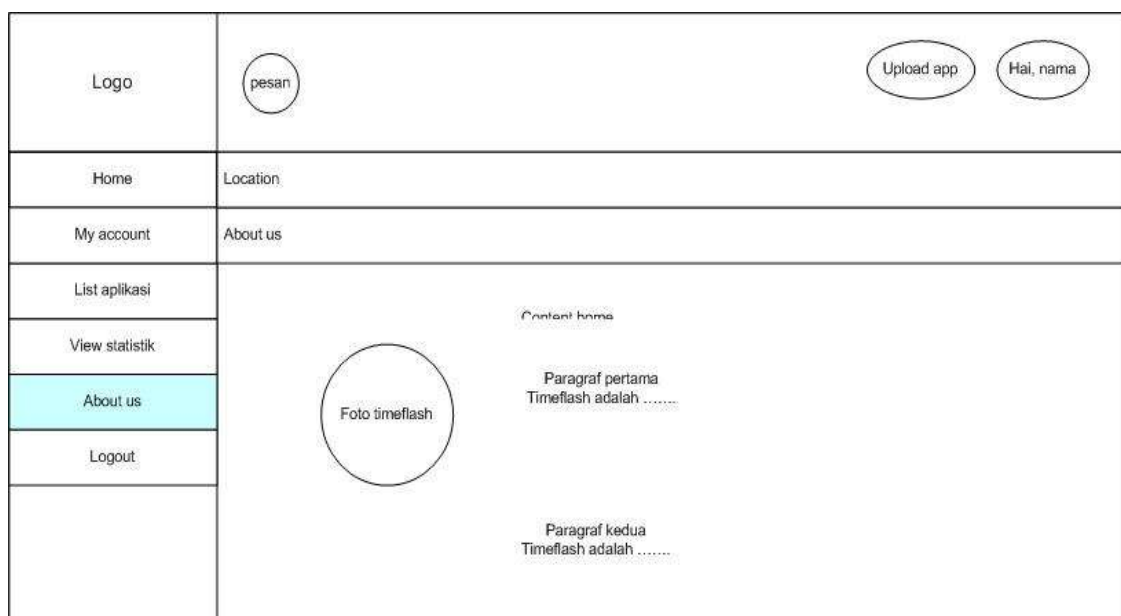
Gambar 3.38 Rancangan antarmuka *view statistik alat bantu ajar*



Gambar 3.39 Rancangan antarmuka view detail aplikasi

m. Rancangan antarmuka *about us*

Rancangan antarmuka *about us* menampilkan informasi mengenai timeflash. Gambar 3.40 merupakan rancangan antarmuka untuk *about us*.



Gambar 3.40 Rancangan antarmuka *about us*

4. Rancangan antarmuka untuk *tester*

a. Rancangan antarmuka *home*

Rancangan antarmuka *home* merupakan tampilan utama yang ditampilkan kepada *tester*. Gambar 3.41 merupakan rancangan antarmuka untuk halaman *home*.

b. Rancangan antarmuka *sign up*

Rancangan antarmuka *sign up* menampilkan *form* untuk *tester* agar dapat mendaftarkan diri sebagai *developer* di akun timeflash. Pada halaman ini, *tester* diminta untuk mengisi data diri. Gambar 3.42 merupakan rancangan antarmuka untuk halaman *sign up*.

logo						
home	Test aplikasi	List aplikasi	Statistik aplikasi	About us	Contact us	
Home		location			Timeflash account Login Sign up	
Content home						
footer						

Gambar 3.41 Rancangan antarmuka *home*

The wireframe shows a web page layout for a sign-up form. At the top is a header bar containing a 'logo' placeholder. Below the header is a navigation menu with links: 'home' (highlighted in cyan), 'Test aplikasi', 'List aplikasi', 'Statistik aplikasi', 'About us', and 'Contact us'. The main content area is titled 'Sign up' and includes a 'location' input field. The form fields are arranged vertically: 'nama', 'Tanggal lahir', 'email', 'instansi', 'password', 'Confirm password', and 'foto'. The 'foto' field has a 'Choose file' button. At the bottom of the form are 'Sign up' and 'reset' buttons. A footer bar at the very bottom contains the word 'footer'.

Gambar 3.42 Rancangan antarmuka *sign up*

c. Rancangan antarmuka *test* aplikasi

Rancangan antarmuka *test* aplikasi menampilkan daftar aplikasi yang dapat dipilih untuk diuji. Pada halaman ini juga disediakan kolom pencarian agar *tester* lebih mudah dalam memilih aplikasi yang akan diuji. Gambar 3.43 merupakan rancangan antarmuka untuk halaman *test* aplikasi.

no	judul	Tanggal upload	developer	intansi	Total score	view	Uji aplikasi

Gambar 3.43 Rancangan antarmuka *test* aplikasi

d. Rancangan antarmuka *form testing*

Pada rancangan antarmuka *form testing* ditampilkan aplikasi yang akan diuji. ditampilkan pula pertanyaan yang digunakan untuk pengujian. Gambar 3.44 merupakan rancangan antarmuka untuk halaman *form testing*.

e. Rancangan antarmuka *view* aplikasi

Rancangan antarmuka *view* aplikasi menampilkan informasi detail terkait aplikasi yang dipilih. Informasi yang ditampilkan berupa aplikasi, detail aplikasi dan *score* aplikasi. Gambar 3.45 merupakan rancangan antarmuka untuk halaman *view* aplikasi.

Gambar 3.44 Rancangan antarmuka *form testing*

Gambar 3.45 Rancangan antarmuka *view aplikasi*

f. Rancangan antarmuka *list aplikasi game*

Rancangan antarmuka *list aplikasi game* berisi informasi tentang daftar aplikasi dengan jenis *game*. Informasi yang ditampilkan adalah nama aplikasi, tanggal *upload*, nama *developer*, instansi dan total *score* hasil pengujian. Rancangan antarmuka untuk *list aplikasi game* dapat dilihat pada gambar 3.46.

logo						
home	Test aplikasi	List aplikasi	Statistik aplikasi	About us	Contact us	
		Aplikasi game				
		Aplikasi alat bantu ajar				
List aplikasi game						location
no	judul	Tanggal upload	developer	instansi	Total score	view
footer						

Gambar 3.46 Rancangan antarmuka *list* aplikasi *game*

g. Rancangan antarmuka *list* aplikasi alat bantu ajar

Rancangan antarmuka *list* aplikasi alat bantu ajar berisi informasi tentang daftar aplikasi dengan jenis alat bantu ajar. Informasi yang ditampilkan adalah nama aplikasi, tanggal *upload*, nama *developer*, instansi dan total *score* hasil pengujian. Rancangan antarmuka untuk *list* aplikasi alat bantu ajar dapat dilihat pada gambar 3.47.

logo						
home	Test aplikasi	List aplikasi	Statistik aplikasi	About us	Contact us	
		Aplikasi game				
		Aplikasi alat bantu ajar				
List aplikasi alat bantu ajar						location
no	judul	Tanggal upload	developer	instansi	Total score	view
footer						

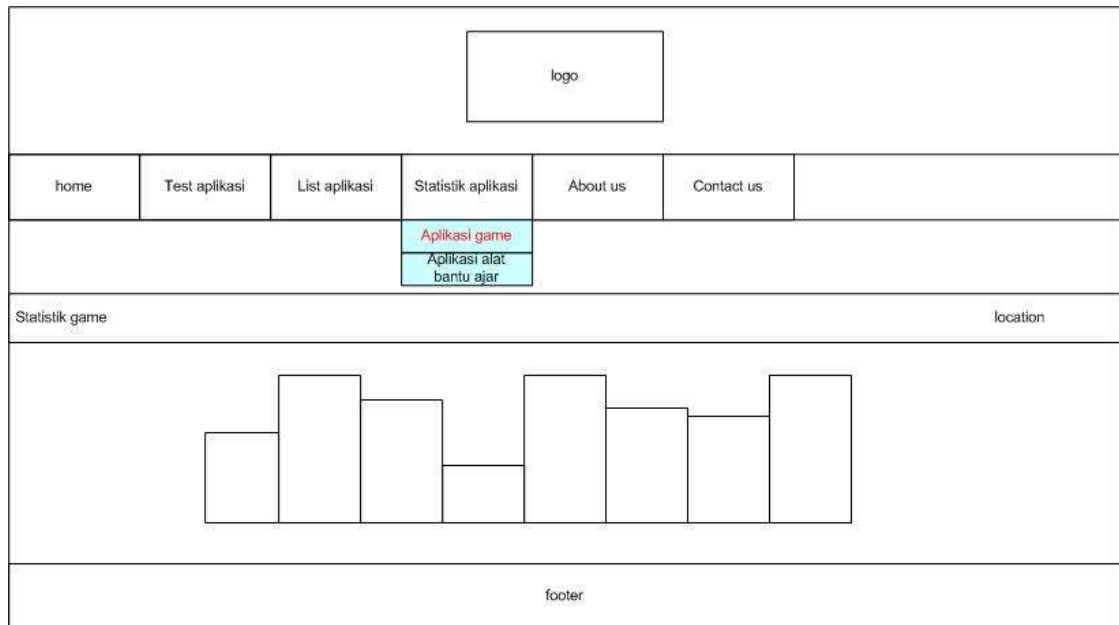
Gambar 3.47 Rancangan antarmuka *list* aplikasi alat bantu ajar

h. Rancangan antarmuka statistik *game*

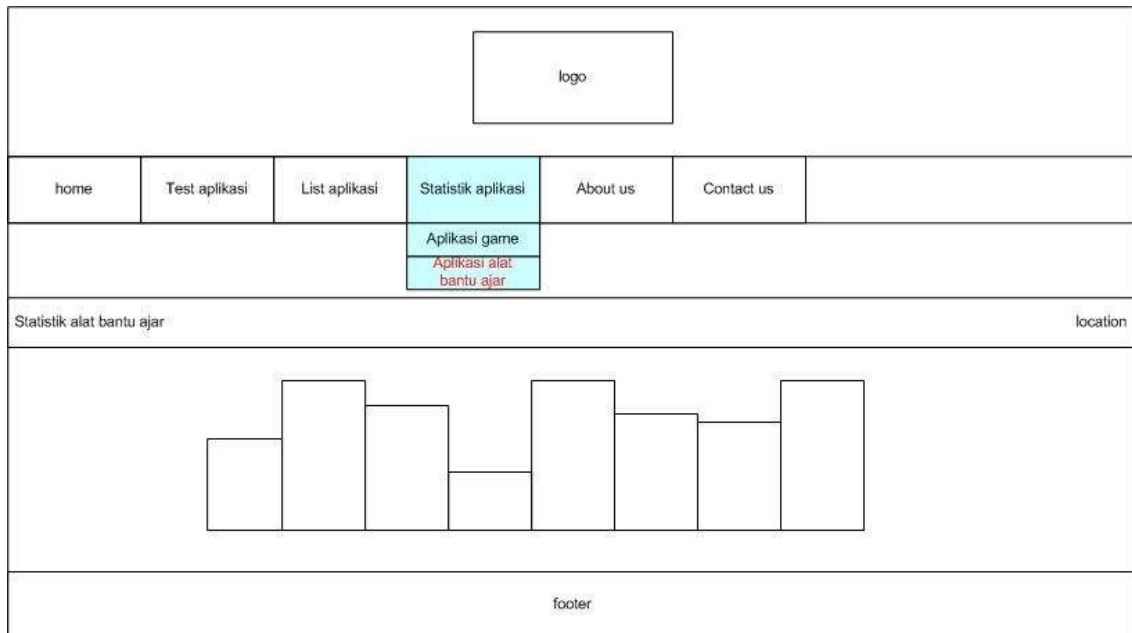
Rancangan antarmuka *view* statistik aplikasi *game* menampilkan statistik dari hasil pengujian aplikasi *game* dalam bentuk diagram batang. Gambar 3.48 merupakan rancangan antarmuka untuk *view* statistik aplikasi *game*.

i. Rancangan antarmuka statistik alat bantu ajar

Rancangan antarmuka *view* statistik aplikasi alat bantu ajar menampilkan statistik dari hasil pengujian aplikasi alat bantu ajar dalam bentuk diagram batang. Gambar 3.49 merupakan rancangan antarmuka untuk *view* statistik aplikasi alat bantu ajar.



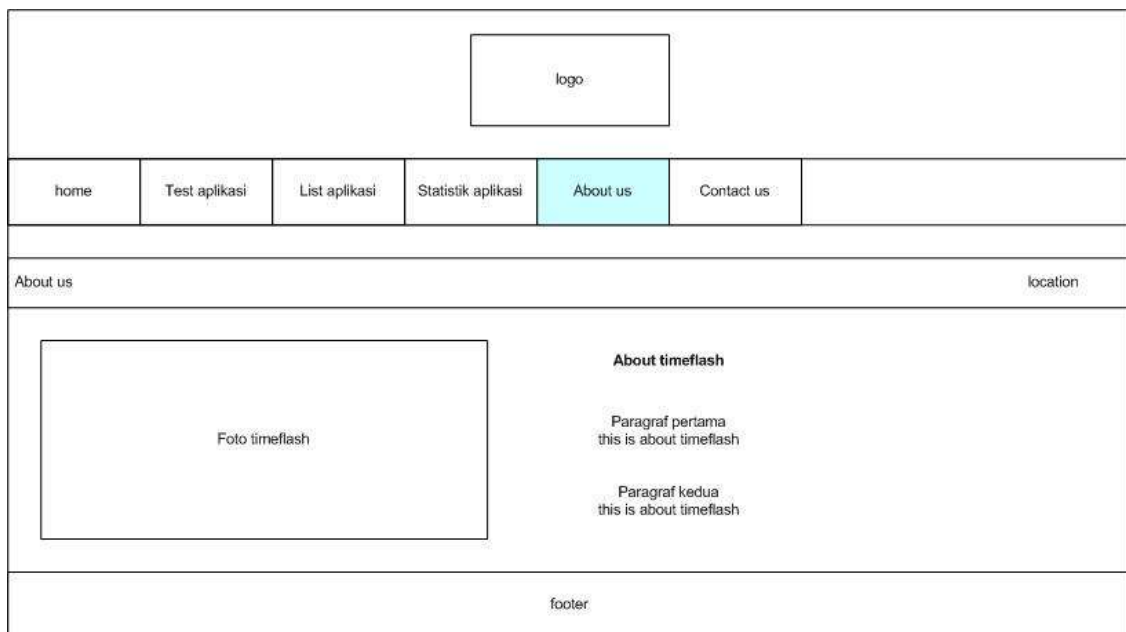
Gambar 3.48 Rancangan antarmuka *view* statistik aplikasi *game*



Gambar 3.49 Rancangan antarmuka *view statistik aplikasi alat bantu ajar*

j. Rancangan antarmuka *about us*

Rancangan antarmuka *about us* menampilkan informasi mengenai timeflash. Gambar 3.50 merupakan rancangan antarmuka untuk *about us*.



Gambar 3.50 Rancangan antarmuka *about us*

k. Rancangan antarmuka *contact us*

Rancangan antarmuka *contact us* menampilkan *form* yang dapat diisi oleh *tester* jika ingin memberi pesan kepada administrator. *Tester* diminta untuk mengisikan data berupa nama, *email* dan komentar. Gambar 3.51 merupakan rancangan antarmuka untuk *contact us*.

Gambar 3.51 Rancangan antarmuka *contact us*

3.3 Perancangan Pengujian Aplikasi

3.3.1 Pertanyaan Pengujian

Pada tahap pengujian aplikasi *flash*, sistem menampilkan pertanyaan pengujian yang akan diisi oleh *tester*. Pertanyaan ditampilkan sesuai jenis aplikasi yang diuji. Pertanyaan terdiri dari 5 aspek *usability* yaitu *learnability*, *memorability*, *efficiency*, *minimal error rate* dan *high user satisfaction*. Adapun pertanyaan untuk aplikasi *game* adalah:

1. Aspek *learnability*
 - a. Alur permainan *game* ini mudah dipelajari
 - b. *Game* ini memberikan petunjuk penggunaan yang jelas
 - c. Mekanisme penghitungan skor atau poin pada *game* ini mudah dipahami
 - d. Anda mengerti cara menghitung *score* yang anda peroleh.
2. Aspek *memorability*
 - a. Anda dengan mudah menghafal alur *game*.

- b. Anda dengan mudah menghafal lokasi dan fungsi elemen antarmuka *game*
- 3. Aspek *efficiency*
 - a. Anda dengan mudah dan cepat mampu mengakses fungsionalitas yang disediakan *game*
 - b. *Game* dapat dimainkan secara baik tanpa membutuhkan spesifikasi *hardware* yang tinggi
 - c. Informasi yang disediakan *game* ditampilkan dengan format yang ringkas dan informatif
- 4. Aspek *minimal error rate*
 - a. Anda tidak menemukan kesalahan teknis pada *game* ini.
 - b. Penanganan kesalahan di *game* ini sudah baik
 - c. Petunjuk dan informasi yang disediakan *game* mampu membantu Anda untuk menghindari kesalahan
- 5. Aspek *high user satisfaction*
 - a. *Game* ini memberikan hiburan yang sesuai dengan ekspektasi Anda
 - b. *Game* ini sangat menarik untuk sering Anda mainkan
 - c. Tampilan *game* sangat menarik dengan desain antarmuka yang sesuai kebutuhan Anda
 - d. Perbedaan tingkat kesulitan di tiap level pada *game* sesuai dengan ekspektasi Anda

Adapun pertanyaan pengujian untuk alat bantu ajar adalah sebagai berikut:

- 1. Aspek *learnability*
 - a. Anda dapat dengan mudah memahami tahap pembelajaran pada alat bantu ajar ini.
 - b. Anda dapat memahami kegunaan dari fungsionalitas yang disediakan pada alat bantu ajar ini
 - c. Alat bantu ajar ini memberikan petunjuk penggunaan yang jelas.
- 2. Aspek *memorability*
 - a. Anda dengan mudah menghafal tahapan pembelajaran pada alat bantu ajar ini.
 - b. Anda dengan mudah menghafal lokasi dan fungsi elemen antarmuka alat bantu ajar.
- 3. Aspek *efficiency*
 - a. Anda dengan mudah dan cepat mampu mengakses fungsionalitas yang disediakan oleh alat bantu ajar ini.
 - b. Dengan menggunakan alat bantu ajar ini, anda membutuhkan waktu yang lebih singkat dalam memahami materi.
 - c. Alat bantu ajar ini membantu anda untuk memahami materi dengan tahapan pembelajaran yang ringkas dan mudah dipahami.
- 4. Aspek *minimal error rate*

- a. Anda tidak menemukan kesalahan teknis pada alat bantu ajar ini.
 - b. Penanganan kesalahan pada alat bantu ajar sudah baik.
 - c. Petunjuk dan informasi pada alat bantu ajar mampu membantu anda untuk menghindari kesalahan.
5. Aspek *high user satisfaction*
- a. Alat bantu ajar ini sangat membantu anda dalam memahami materi yang diajarkan.
 - b. Anda lebih memahami materi yang disampaikan melalui alat bantu ajar ini dibandingkan dengan materi yang disampaikan secara konvensional.
 - c. Tampilan alat bantu ajar sangat menarik dan dapat menambah minat anda untuk menggunakannya.
 - d. Alat bantu ajar ini menyajikan materi dengan cara yang interaktif dan mudah dipahami.

3.3.2 Rumus Penghitungan Score

Aplikasi yang akan dihitung *score* yang didapat berdasarkan hasil jawaban *tester* yang dilakukan pada proses pengujian. Adapun rumus untuk menghitung *score* aplikasi yaitu:

$$\text{nilai aspek} = \frac{(\text{nilai yang didapat} + \text{jumlah tester})}{(5 \times \text{jumlah pertanyaan})} \times \text{bobot aspek}$$

Rumus diatas adalah rumus untuk mencari nilai satu aspek dari satu aplikasi. Nilai yang didapat merupakan nilai yang diperoleh aplikasi dari jawaban *tester* pada tahap pengujian aplikasi. Jumlah *tester* merupakan jumlah penguji pada satu aplikasi *flash*. Nilai 5 yang dimaksud adalah nilai maksimal yang dapat diperoleh aplikasi untuk 1 pertanyaan. Nilai 5 disesuaikan dengan nilai untuk skala likert pada umumnya. Jumlah pertanyaan merupakan jumlah pertanyaan yang diujikan pada tahap pengujian. Bobot aspek merupakan nilai untuk satu aspek *usability*. Adapun cara mendapatkan nilai total pada 1 aplikasi yaitu,

$$\text{total} = \text{nilai aspek 1} + \text{nilai aspek 2} + \text{nilai aspek 3} + \text{nilai aspek 4} + \text{nilai aspek 5}$$

Atau dapat diringkas dengan:

$$\text{total} = \sum \text{nilai aspek}$$

Berdasarkan cara diatas, untuk mendapat nilai total dari semua aspek pada suatu aplikasi yaitu dengan menambahkan semua nilai aspek pada aplikasi itu. Dengan begitu didapatlah nilai total untuk satu aplikasi.

BAB IV

IMPLEMENTASI DAN PENGUJIAN

4.1 Implementasi Perangkat lunak

Implementasi merupakan tahapan yang dilakukan setelah tahapan perancangan. Pada tahapan implementasi, sistem dijalankan dan digunakan pada kasus tertentu sesuai dengan tujuan perancangan sistem. Berikut ini merupakan implementasi dari TIMEFLASH, sistem pengujian aplikasi *flash*.

4.1.1 Halaman *login*

Halaman *login* digunakan oleh admin dan *tester* untuk dapat mengakses halaman di TIMEFLASH. Pada halaman *login*, admin dan *developer* diminta untuk mengisi alat *email* dan *password*. Tampilan untuk halaman *login* dapat dilihat pada gambar 4.1.



Gambar 4.1 Halaman *login*

4.1.2 Halaman *home* admin

Halaman *home* merupakan halaman yang pertama ditampilkan kepada admin setelah melakukan proses *login*. Halaman *home* menampilkan logo TIMEFLASH, informasi terkait *timeflash* dan total aplikasi yang telah diunggah di sistem TIMEFLASH. Gambar 4.2 merupakan gambar untuk tampilan *home*.



Gambar 4.2 Halaman *home* admin

4.1.3 Halaman manajemen *developer*

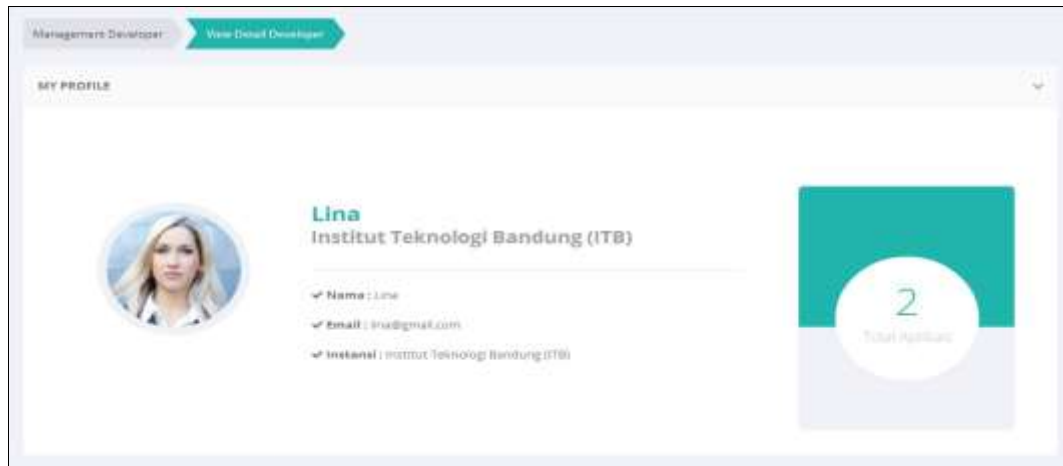
Halaman manajemen *developer* menampilkan informasi mengenai *developer* yang terdaftar di TIMEFLASH. Informasi yang ditampilkan berupa nama, *email*, instansi dan total aplikasi yang telah diunggah oleh *developer* tersebut. Halaman untuk manajemen *developer* dapat dilihat pada gambar 4.3.

MANAGEMENT DEVELOPER								
5	records per page		Search:					
No	Nama	Email	Instansi	Total Aplikasi	View	Edit	Reset Password	Delete
1	Admin	hertua@gmail.com	ui	3				
2	akhy	hsaa@gmail.com	Universitas Islam Indonesia	0				
3	femp	fem@gmail.com	pondok	0				
4	Pery	pery@gmail.com	umt	1				
5	far	faruhana@gmail.com	Universitas Islam Indonesia	0				

Gambar 4.3 Halaman manajemen *developer*

1. Halaman *view developer*

Halaman *view developer* merupakan halaman yang digunakan untuk melihat detail informasi *developer*. Informasi yang ditampilkan berupa foto, nama, *email*, instansi dan total aplikasi yang telah diunggah. Gambar 4.4 merupakan tampilan untuk halaman *view developer*.



Gambar 4.4 Halaman *view developer*

2. Halaman *edit developer*

Halaman *edit developer* merupakan halaman yang digunakan untuk mengubah data *developer*. Halaman ini menampilkan *form* terkait data *developer*. Tampilan untuk halaman *edit developer* dapat dilihat pada gambar 4.5.

Gambar 4.5 Halaman *view developer*

4.1.4 Halaman manajemen aplikasi *game*

Terdapat beberapa informasi yang ditampilkan di halaman manajemen aplikasi *game* yaitu nama aplikasi *game*, tanggal *upload*, *developer* aplikasi, instansi *developer*, dan total score aplikasi. Gambar 4.6 merupakan tampilan untuk halaman manajemen aplikasi *game*.

No	Nama	Tanggal Upload	Developer	Instansi	Total Score	View	Edit	Delete
1	Bermari borsana	2014-05-31	Lina	Institut Teknologi Bandung (ITB)	0			
2	Permainan Seru Asik	2014-05-06	Admin	ua	1.6			

Gambar 4.6 Halaman manajemen aplikasi *game*

4.1.5 Halaman manajemen aplikasi alat bantu ajar

Informasi yang ditampilkan di halaman manajemen aplikasi alat bantu ajar yaitu nama aplikasi alat bantu ajar, tanggal *upload*, *developer* aplikasi, instansi *developer*, dan total score aplikasi. Gambar 4.7 merupakan tampilan untuk halaman manajemen aplikasi alat bantu ajar.

No	Nama	Tanggal Upload	Developer	Instansi	Total Score	View	Edit	Delete
1	Alat bantu Ajar Belajar Membaca	2014-05-15	Nico Bryan Pranata	Universitas Islam Indonesia	55.6			
2	Belajar mengaji	2014-05-01	Fery	um	0			
3	Main belajar dengan aplikasi seru	2014-05-05	Admin	ua	0			
4	Pelajaran matematika menyenangkan	2014-05-02	Admin	ua	0			
5	Hukum Newton	2014-05-15	Lina	Institut Teknologi Bandung (ITB)	0			

Gambar 4.7 Halaman aplikasi alat bantu ajar

4.1.6 Halaman *view* aplikasi

Halaman *view* aplikasi menampilkan detail informasi aplikasi. Pada halaman ini juga ditampilkan aplikasi *flash* yang dapat dijalankan. Halaman ini juga menampilkan detail *score* yang diperoleh aplikasi. Gambar 4.8 merupakan tampilan untuk halaman *view* aplikasi.



Gambar 4.8 Halaman *view* aplikasi

4.1.7 Halaman *edit* aplikasi

Halaman *edit* aplikasi menampilkan *form* untuk mengubah data aplikasi. Data yang dapat diubah yaitu nama dan jenis aplikasi. Tampilan untuk *edit* aplikasi dapat dilihat pada gambar 4.9.

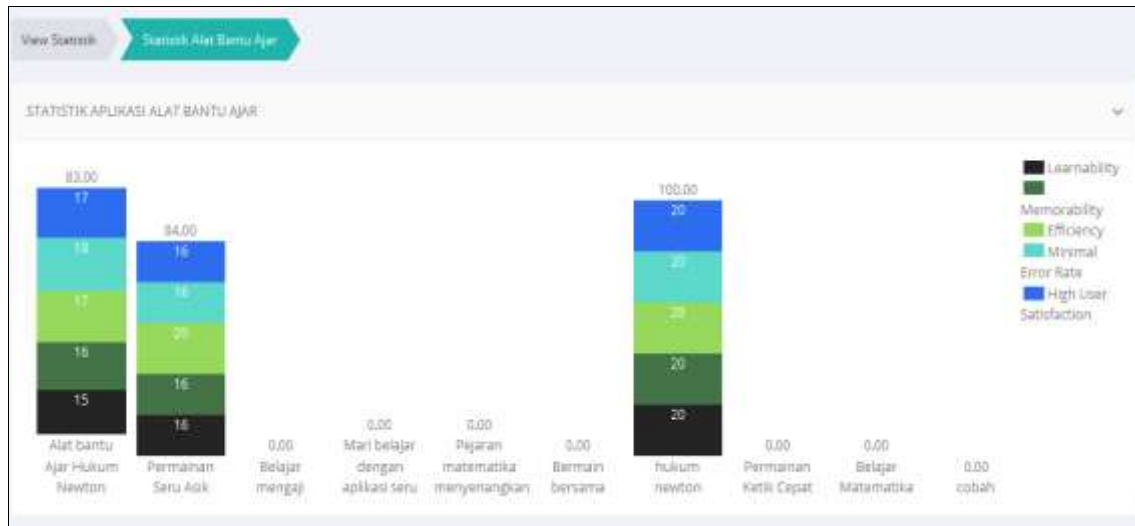
FORM EDIT APLIKASI

Nama:

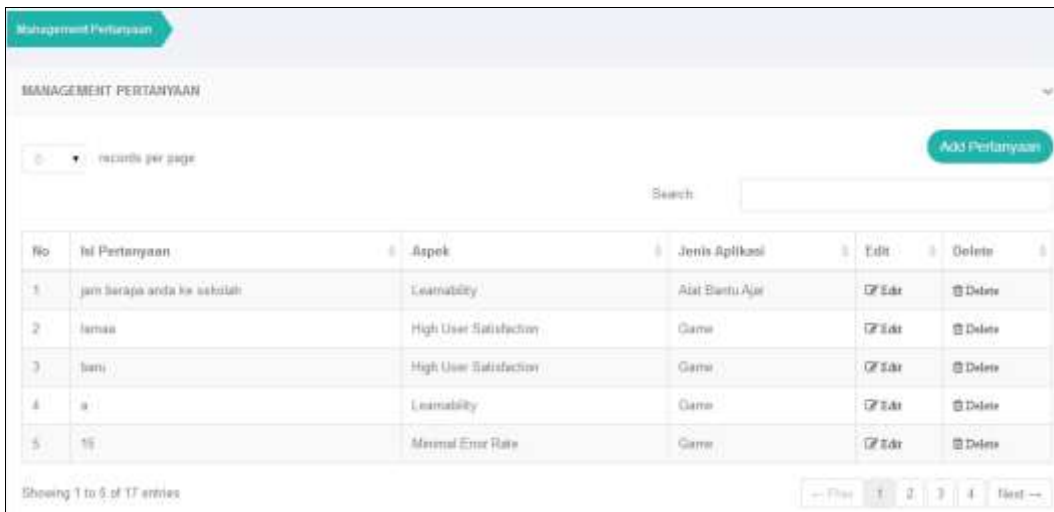
Jenis Aplikasi:

Gambar 4.9 Halaman *edit* aplikasi**4.1.8 Halaman view statistik**

Halaman *view* statistik alat bantu ajar menampilkan diagram yang dibentuk berdasarkan *score* yang didapat aplikasi dari hasil pengujian. Tampilan untuk halaman statistik dapat dilihat pada gambar 4.10.

**Gambar 4.10** Halaman *view* statistik**4.1.9 Halaman manajemen pertanyaan**

Halaman manajemen pertanyaan menampilkan daftar pertanyaan beserta aspek dan jenis pertanyaan. Jenis pertanyaan mengindikasikan untuk aplikasi jenis apa pertanyaan ini ditampilkan. Pada halaman ini admin dapat mengelola data pertanyaan. Admin dapat mengubah atau menghapus pertanyaan. Gambar 4.11 merupakan tampilan untuk halaman manajemen pertanyaan.



No	Isi Pertanyaan	Aspek	Jenis Aplikasi	Edit	Delete
1	jam berapa anda ke sekolah	Learnability	Ajar Bantu Ajar	Edit	Delete
2	nama	High User Satisfaction	Game	Edit	Delete
3	nama	High User Satisfaction	Game	Edit	Delete
4	a	Learnability	Game	Edit	Delete
5	1%	Minimal Error Rate	Game	Edit	Delete

Showing 1 to 5 of 17 entries

Gambar 4.11 Halaman manajemen pertanyaan

1. *Add* pertanyaan

Halaman *add* pertanyaan menampilkan *form* untuk menambah pertanyaan. Tampilan untuk halaman *add* pertanyaan dapat dilihat pada gambar 4.12.



ADD PERTANYAAN

Isi Pertanyaan:

Aspek Usability:

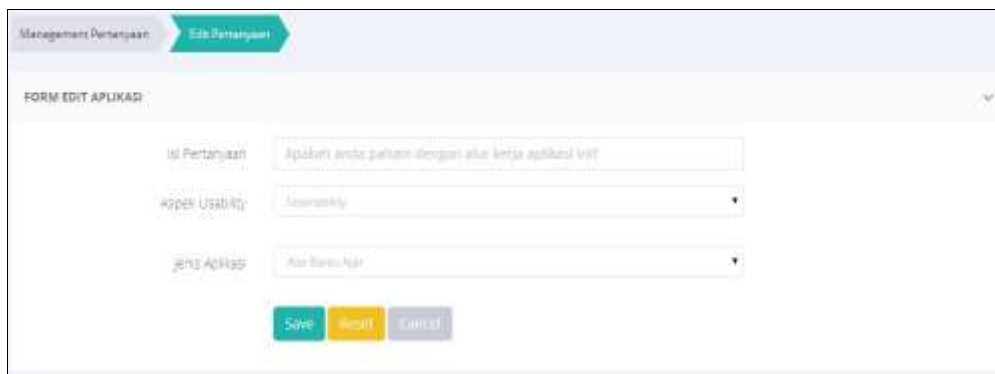
Jenis Aplikasi:

[Save](#) [Reset](#) [Cancel](#)

Gambar 4.12 Halaman *add* pertanyaan

2. *Edit* pertanyaan

Halaman *edit* pertanyaan menampilkan *form* yang digunakan untuk mengubah data pertanyaan. Data yang diubah berupa isi pertanyaan, aspek dan jenis pertanyaan. Tampilan untuk halaman *edit* pertanyaan dapat dilihat pada gambar 4.13.



Gambar 4.13 Halaman *edit* pertanyaan

4.1.9 Halaman *change password*

Halaman *change password* menampilkan *form* yang digunakan admin untuk mengubah *password*. Admin diminta untuk memasukkan *password* lama, *password* baru dan *password* konfirmasi agar dapat mengubah *password*. Gambar 4.14 merupakan tampilan untuk halaman *change password*.



Gambar 4.14 Halaman *change password*

4.1.10 Halaman *home developer*

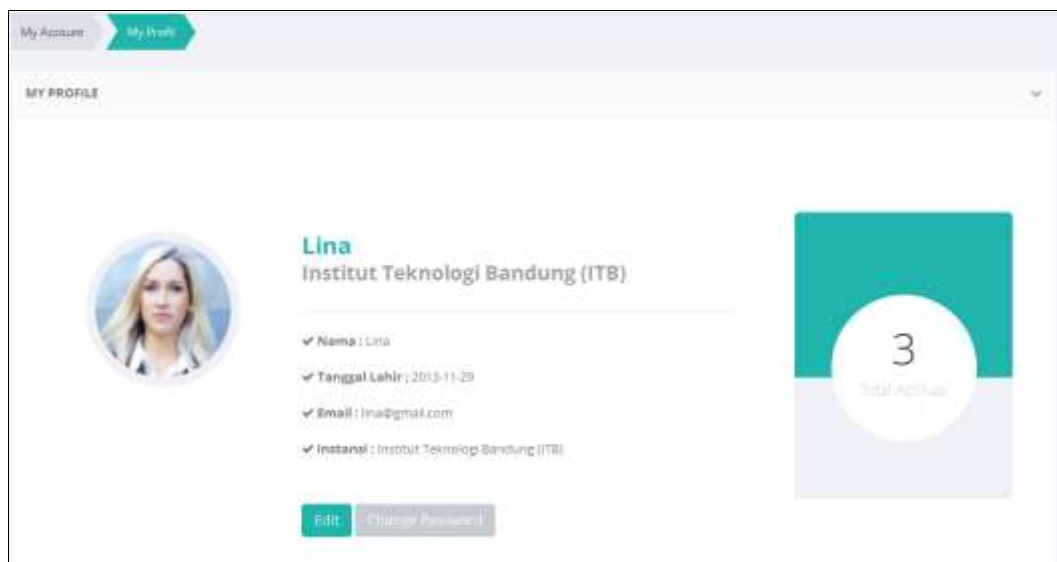
Halaman *home* merupakan halaman yang pertama ditampilkan kepada *developer* setelah melakukan proses *login*. Gambar 4.15 merupakan tampilan untuk halaman *home* pada *developer*.



Gambar 4.15 Halaman *home developer*

4.1.11 Halaman *my profil*

Halaman *my profil* merupakan submenu dari menu *my account* yang menampilkan data diri *developer*. Data diri yang ditampilkan berupa foto, nama, instansi, *email* dan total aplikasi yang telah diunggah. Halaman *my profil* dapat dilihat pada gambar 4.16.



Gambar 4.16 Halaman *my profil*

4.1.12 Halaman *my application*

Halaman *my application* digunakan oleh *developer*. halaman ini digunakan untuk menampilkan informasi berupa daftar aplikasi yang telah diunggah oleh *developer*. Gambar 4.17 merupakan tampilan untuk halaman *my application*.



No	Nama	Jenis	Tanggal	Total Score	View	Edit	Delete
1	Berman bersama	2	2014-05-31	0	View	Edit	Delete
2	hukum newton	2	2014-05-15	0	View	Edit	Delete

Gambar 4.17 Halaman *my application*

4.1.13 Halaman *about us*

Halaman *about us* menampilkan informasi tentang TIMEFLASH. Tampilan untuk halaman *about us* dapat dilihat pada gambar 4.18.



Timeflash
Universitas Islam Indonesia

Timeflash merupakan sistem online yang dibuat untuk mendukung pengujian pada aplikasi flash berupa game dan alat bantu ajar. Pengujian dilakukan pada aspek usability. Terdapat 5 aspek usability yaitu Learnability, memorability, efficiency, minimal error rate dan high user satisfaction.

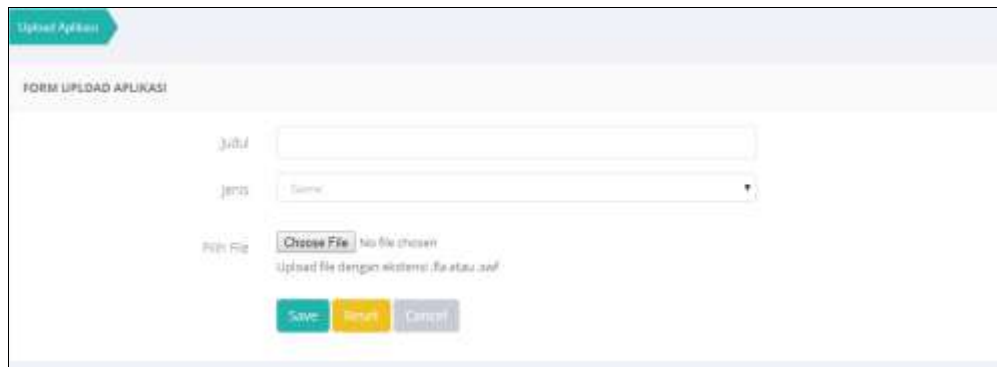
Diharapkan dengan sistem ini, para developer aplikasi flash khususnya di bidang game dan alat bantu ajar dapat mengetahui kelebihan dan kekurangan dari aplikasi yang dibuat pada aspek-aspek tertentu. Disediakan juga daftar aplikasi yang telah diunggah oleh developer lain yang dapat dijadikan acuan ketika seorang developer akan mengembangkan aplikasi flash.

CONTACTS
Email: Timeflash@gmail.com

Gambar 4.18 Halaman *about us*

4.1.14 Halaman *upload*

Halaman *upload* aplikasi digunakan oleh *developer* untuk mengunggah aplikasi yang ingin diuji. *developer* diminta untuk memasukkan beberapa data terkait aplikasi yang akan diunggah. Gambar 4.19 merupakan tampilan untuk halaman *upload*



Gambar 4.19 Halaman *upload*

4.1.15 Halaman *contact admin*

Halaman *contact admin* digunakan oleh *developer* untuk menulis pesan kepada admin. Halaman ini menampilkan *form* untuk menuliskan isi pesan. Tampilan untuk *contact admin* dapat dilihat pada gambar 4.20.

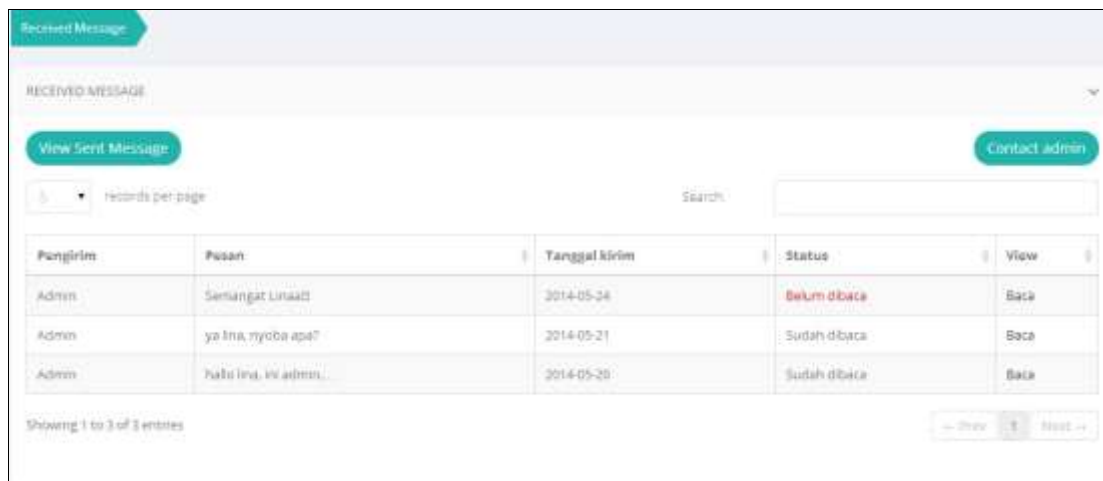


Gambar 4.20 Halaman *contact admin*

4.1.16 Halaman *received message*

Halaman *received message* menampilkan daftar pesan masuk. Halaman ini digunakan oleh admin dan *developer*. Tombol *contact admin* hanya ditampilkan ketika digunakan oleh *developer*. Sedangkan pada

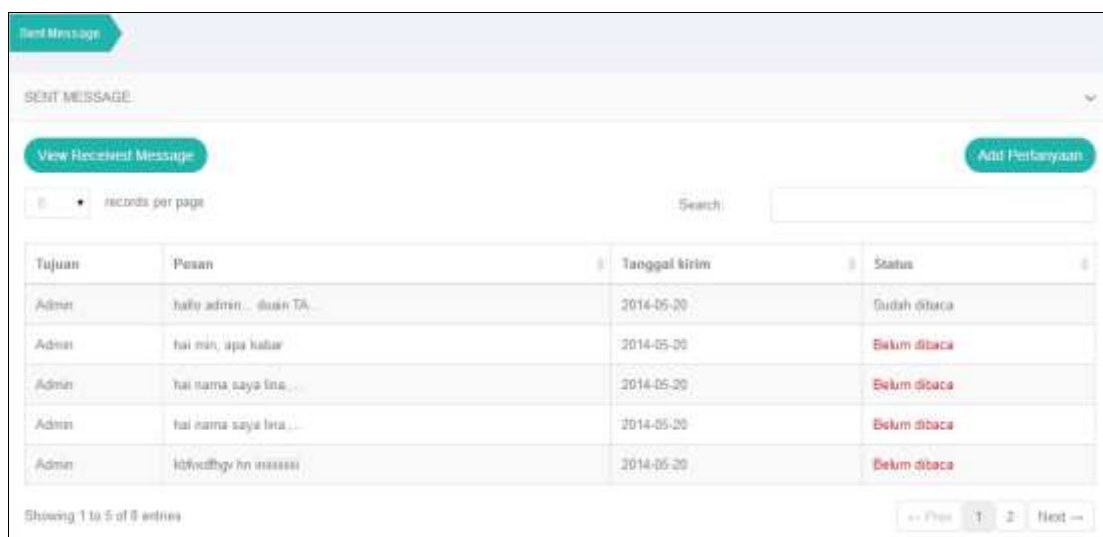
halaman admin tombol ini tidak ditampilkan. Gambar 4.21 merupakan tampilan untuk halaman *received message*.



Gambar 4.21 Halaman *received message*

4.1.17 Halaman *sent message*

Halaman *sent message* menampilkan daftar pesan yang telah dikirim. Halaman ini digunakan oleh admin dan *developer*. Tombol *contact admin* hanya ditampilkan ketika digunakan oleh *developer*. Sedangkan pada halaman admin tombol ini tidak ditampilkan. Gambar 4.22 merupakan tampilan untuk halaman *sent message*.



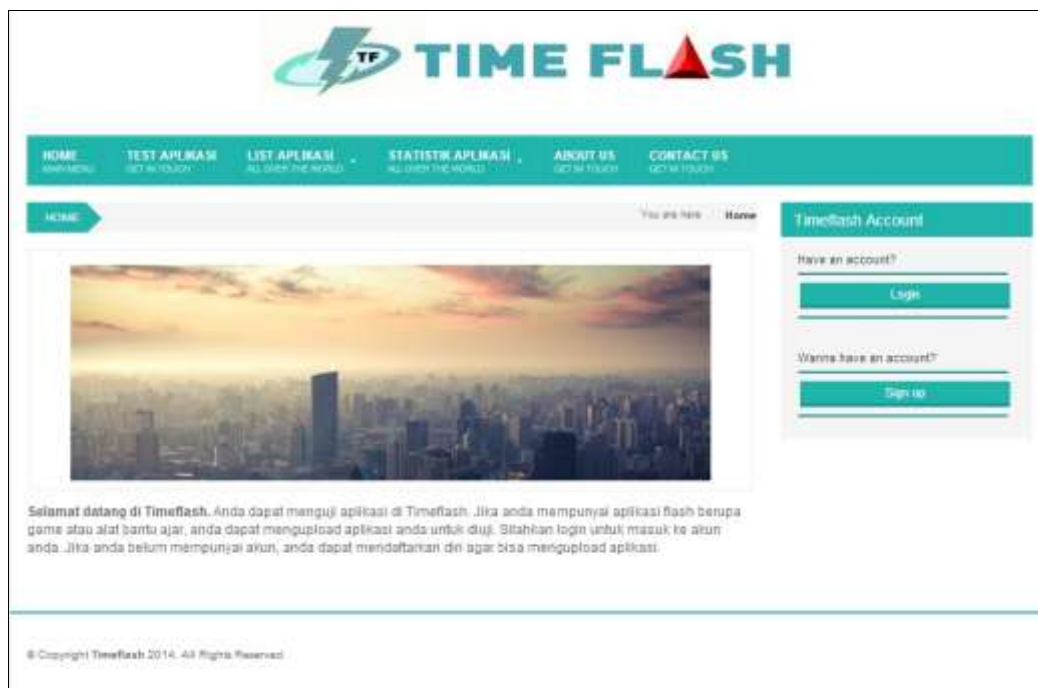
Gambar 4.22 Halaman *sent message*

4.1.18 Halaman *home tester*

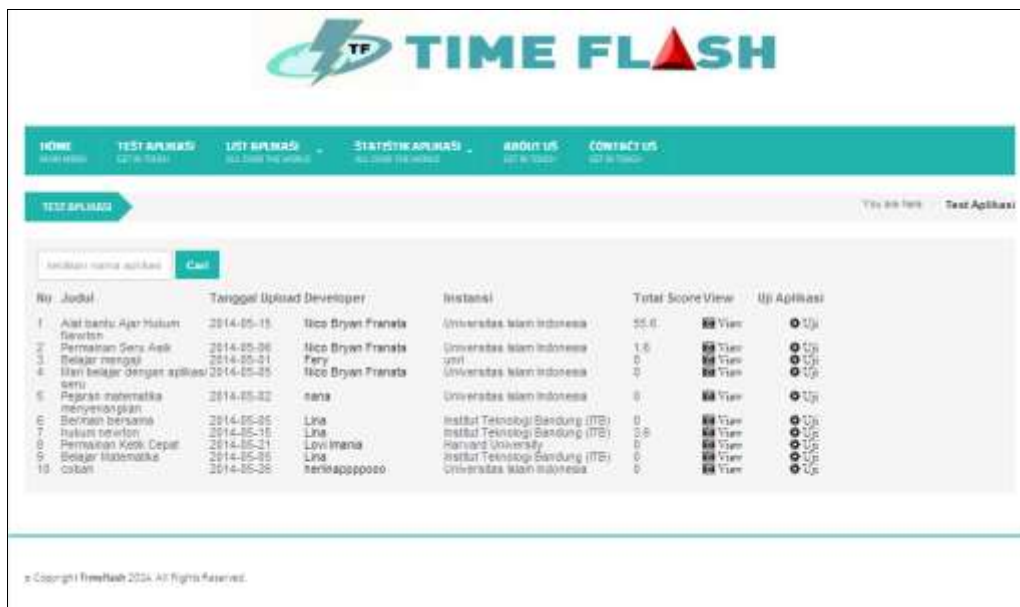
Halaman yang pertama kali ditampilkan kepada pengguna adalah halaman *home*. Pada halaman ini disediakan *link* untuk melakukan pendaftaran dan proses *login* untuk *developer*. Tampilan untuk halaman *home* dapat dilihat pada gambar 4.23.

4.1.19 Halaman *test aplikasi*

Halaman *test* aplikasi menampilkan daftar aplikasi di TIMEFLASH yang dapat diuji oleh *tester*. Di halaman ini juga disediakan kolom pencarian untuk memudahkan *tester* dalam mencari aplikasi yang ingin diuji. gambar 4.24 merupakan tampilan untuk menu *test* aplikasi.



Gambar 4.23 Halaman *home*



No	Judul	Tanggal Upload	Developer	Instansi	Total Score	View	Uji Aplikasi
1	Aksi bantu Ajar Islam	2014-05-15	Nico Bryan Fransis	Universitas Islam Indonesia	55.0	View	Uji
2	Pernikahan Gens Arah	2014-05-26	Nico Bryan Fransis	Universitas Islam Indonesia	1.5	View	Uji
3	Belajar mengaji	2014-05-21	Perry	UIN	0	View	Uji
4	Man belajar dengan aplikasi	2014-05-25	Nico Bryan Fransis	Universitas Islam Indonesia	0	View	Uji
5	Pengantar matematika	2014-05-22	nana	Universitas Islam Indonesia	0	View	Uji
6	Belajar matematika	2014-05-25	Lina	Institut Teknologi Bandung (ITB)	0	View	Uji
7	Belajar matematika	2014-05-16	Lina	Institut Teknologi Bandung (ITB)	3.8	View	Uji
8	Pemahaman Katak Cepat	2014-05-21	Lovi Irena	Rahmad University	0	View	Uji
9	Belajar Matematika	2014-05-25	Lina	Institut Teknologi Bandung (ITB)	0	View	Uji
10	Belajar Matematika	2014-05-25	senoappoo	Universitas Islam Indonesia	0	View	Uji

Gambar 4.24 Halaman *test* aplikasi

1. Halaman *form* uji aplikasi

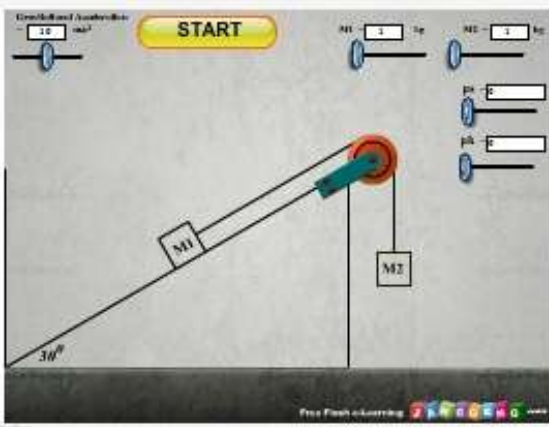
Tester akan diarahkan menuju halaman *form* uji aplikasi setelah memilih tombol uji aplikasi pada aplikasi yang dipilih. *Form* ini juga menampilkan aplikasi yang akan diuji kemudian diikuti dengan pengisian data uji aplikasi. Gambar 4.25 merupakan gambar untuk *form* uji aplikasi.

2. Halaman *view* aplikasi

Halaman *view* aplikasi menampilkan detail informasi terkait aplikasi. Halaman ini juga menampilkan aplikasi *flash* yang dapat dijalankan. Tampilan untuk *view* aplikasi dapat dilihat pada gambar 4.26.

FORM TESTING

You are here: Form Testing



Aspek baru: Aspek Mekanika Newton

Name:

Email:

Aspek Learnability

6

☐ Sangat Tidak Sesuai
 ☐ Tidak Sesuai
 ☐ Sesuai
 ☐ Sangat Sesuai

Aspek Memorability

7

☐ Sangat Tidak Sesuai
 ☐ Tidak Sesuai
 ☐ Sesuai
 ☐ Sangat Sesuai

Aspek Memorability

8

☐ Sangat Tidak Sesuai
 ☐ Tidak Sesuai
 ☐ Sesuai
 ☐ Sangat Sesuai

Aspek Efficiency

9

☐ Sangat Tidak Sesuai
 ☐ Tidak Sesuai
 ☐ Sesuai
 ☐ Sangat Sesuai

Aspek Efficiency

10

☐ Sangat Tidak Sesuai
 ☐ Tidak Sesuai
 ☐ Sesuai
 ☐ Sangat Sesuai

Aspek Minimal Error Rate

11

☐ Sangat Tidak Sesuai
 ☐ Tidak Sesuai
 ☐ Sesuai
 ☐ Sangat Sesuai

Aspek Minimal Error Rate

12

☐ Sangat Tidak Sesuai
 ☐ Tidak Sesuai
 ☐ Sesuai
 ☐ Sangat Sesuai

Aspek High User Satisfaction

baru

☐ Sangat Tidak Sesuai
 ☐ Tidak Sesuai
 ☐ Sesuai
 ☐ Sangat Sesuai

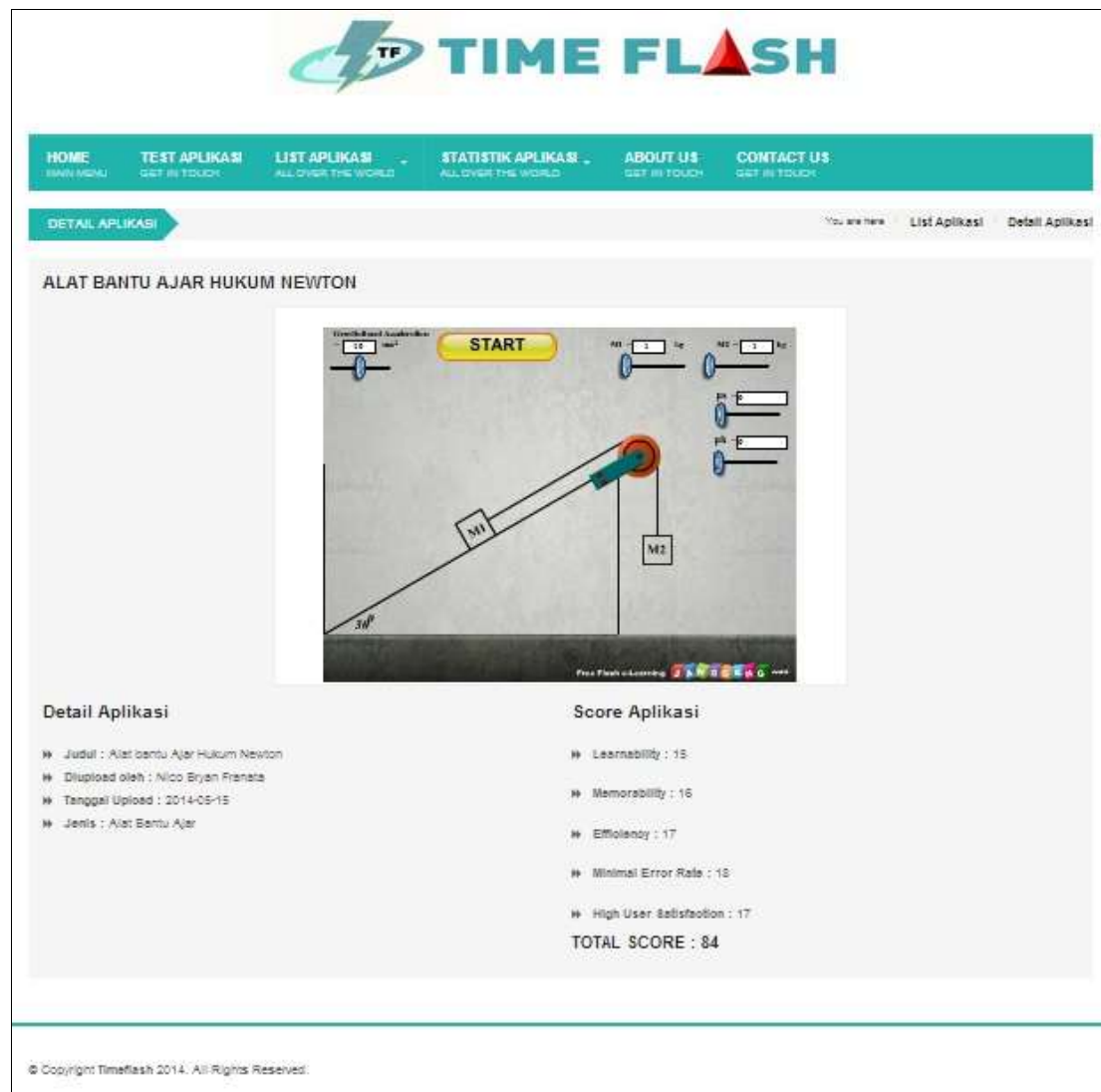
Aspek Learnability

Apakah anda paham dengan alur kerja aplikasi ini?

☐ Sangat Tidak Sesuai
 ☐ Tidak Sesuai
 ☐ Sesuai
 ☐ Sangat Sesuai

Submit Reset

Gambar 4.25 Halaman *form* uji aplikasi



Gambar 4.26 Halaman view aplikasi

4.1.20 Halaman *list* aplikasi *game*

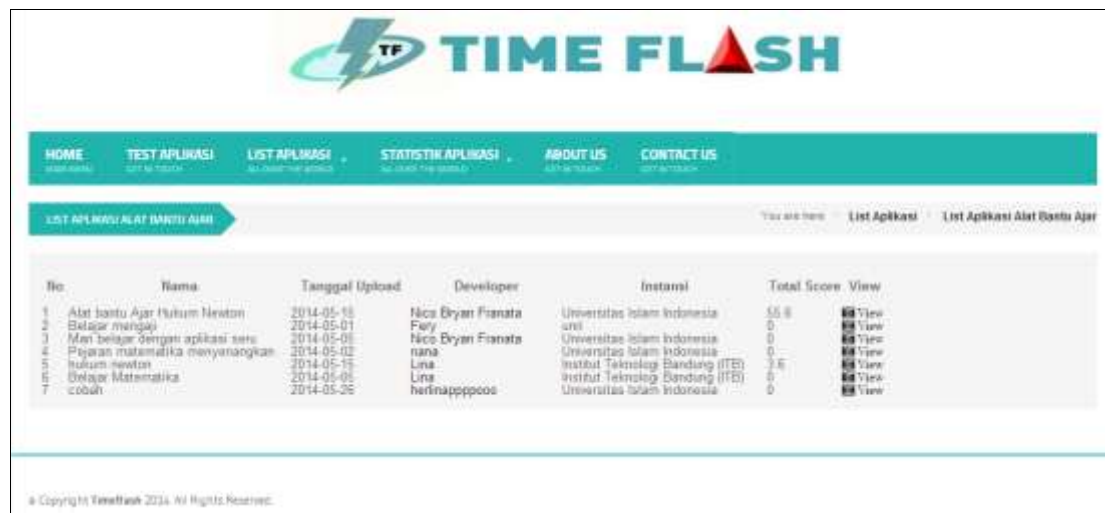
Halaman *list* aplikasi *game* berisi informasi tentang aplikasi *game* yang telah diunggah di TIMEFLASH. Informasi yang ditampilkan di halaman *list* aplikasi *game* berupa nama aplikasi *game*, tanggal *upload*, *developer* aplikasi, instansi *developer*, dan total score aplikasi. Gambar 4.27 merupakan tampilan untuk halaman *list* aplikasi *game*.



Gambar 4.27 Halaman *list* aplikasi *game*

4.1.21 Halaman *list* aplikasi alat bantu ajar

Halaman *list* aplikasi alat bantu ajar berisi informasi tentang aplikasi alat bantu ajar yang telah diunggah di TIMEFLASH. Informasi yang ditampilkan di halaman *list* aplikasi alat bantu ajar berupa nama aplikasi alat bantu ajar, tanggal *upload*, *developer* aplikasi, instansi *developer*, dan total *score* aplikasi. Gambar 4.28 merupakan tampilan untuk halaman *list* aplikasi alat bantu ajar.



Gambar 4.28 Halaman *list* aplikasi alat bantu ajar

4.1.22 Halaman *about us*

Halaman *about us* menampilkan informasi tentang TIMEFLASH. Tampilan untuk halaman *about us* dapat dilihat pada gambar 4.29.



Gambar 4.29 Halaman *about us*

4.1.23 Halaman *contact us*

Halaman *contact us* menampilkan *form* yang dapat digunakan *tester* untuk mengirim pesan kepada admin. *Tester* diminta untuk memasukkan nama, *email*, dan pesan yang akan dikirim. Tampilan untuk halaman *contact us* dapat dilihat pada gambar 4.30

TIME FLASH

HOME MAIN MENU TEST APLIKASI GET IN TOUCH LIST APLIKASI ALL OVER THE WORLD STATISTIK APLIKASI ALL OVER THE WORLD ABOUT US GET IN TOUCH CONTACT US GET IN TOUCH

CONTACT US You are here / Contact Us

Send us a message
 Silahkan mengisi form untuk meninggalkan pesan kepada administrator. Pesan akan dibalas pada alamat email yang diisi.

Name *

E-mail *

Comment *

Your message..

Post a Comment

© Copyright Timeflash 2014. All Rights Reserved.

Gambar 4.30 Halaman *contact us*

4.2 Pengujian

4.2.1 Pengujian masukan dan keluaran pada perangkat lunak

1. Masukan *sign up*

Form sign up akan diuji dengan memasukkan data seperti yang terlihat pada gambar 4.31. Hasil atau keluaran dari data yang dimasukan dapat dilihat pada gambar 4.32.

Registration form

Nama *

Tanggal lahir *

E-mail *

Instansi *

Password *

Confirm Password *

Poto Profil *
 IMG_1230.JPG

Gambar 4.31 Halaman masukan *sign up*

LOGIN TIMEFLASH

Selamat! pendaftaran berhasil! ✕
 silahkan login untuk masuk ke akun anda

Email

Password

Gambar 4.32 Halaman keluaran *sign up*

2. Masukan *login*

Data masukan *login* diuji dengan memasukkan data seperti pada gambar 4.33. Keluaran hasil *login* berupa halaman *home* yang terlihat pada gambar 4.34.



LOGIN TIMEFLASH

lana@gmail.com

password

SIGN IN

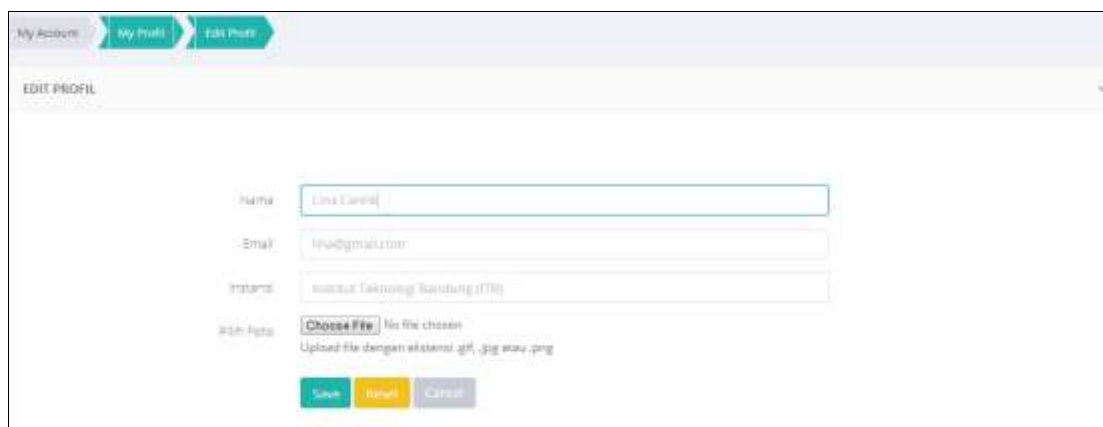
Gambar 4.33 Masukan *login*



Gambar 4.34 Keluaran *login*

3. Masukan *edit profil*

Contoh data *developer* yang akan diubah diperlihatkan pada gambar 4.35. Setelah data berhasil diubah, maka akan tampil notifikasi bahwa data berhasil diubah seperti diperlihatkan pada gambar 4.36.



My Account > My Profile > Edit Profile

EDIT PROFIL

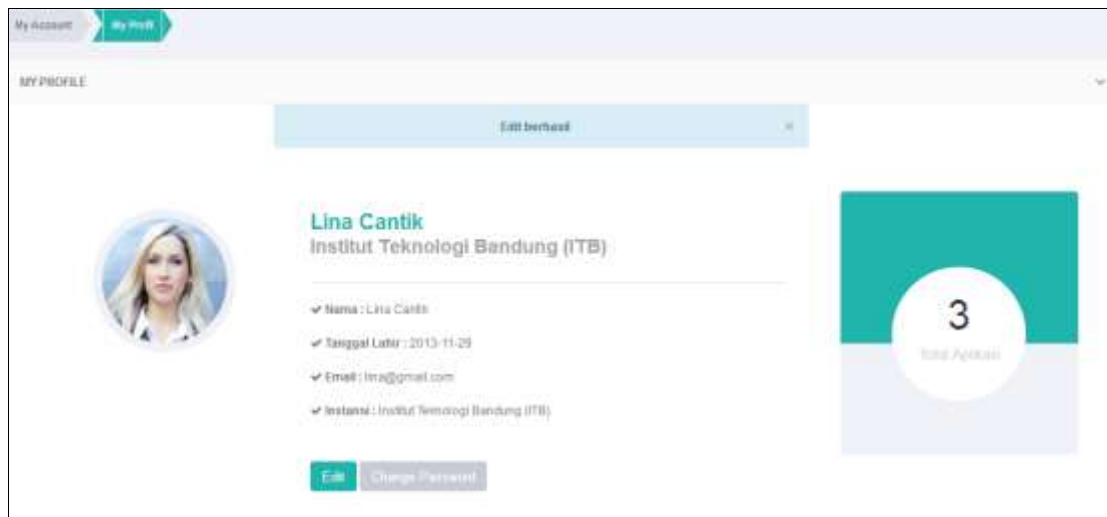
Nama: Lana Lani

Email: lana@gmail.com

Instansi: Pusat Teknologi Banding (PTB)

Upload Foto: No file chosen
Upload file dengan ekstensi gif, jpg atau png

Gambar 4.35 Masukan *edit profil*



My Account My Profile

MY PROFILE

Edit Profile

Lina Cantik
Institut Teknologi Bandung (ITB)

- ✓ Nama : Lina Cantik
- ✓ Tanggal Lahir : 2013-11-29
- ✓ Email : lina@gmail.com
- ✓ Instansi : Institut Teknologi Bandung (ITB)

Edit Change Password

3

Gambar 4.36 Keluaran *edit profil*

4. Masukan *edit password*

Pada *edit password*, *developer* diminta untuk mengisi *password* lama, *password* baru kemudian mengetik ulang *password* baru. Masukan *edit password* dapat dilihat pada gambar 4.37. Apabila *password* berhasil diubah maka sistem akan menampilkan pesan seperti terlihat pada gambar 4.38



Edit Password

FORM EDIT PASSWORD

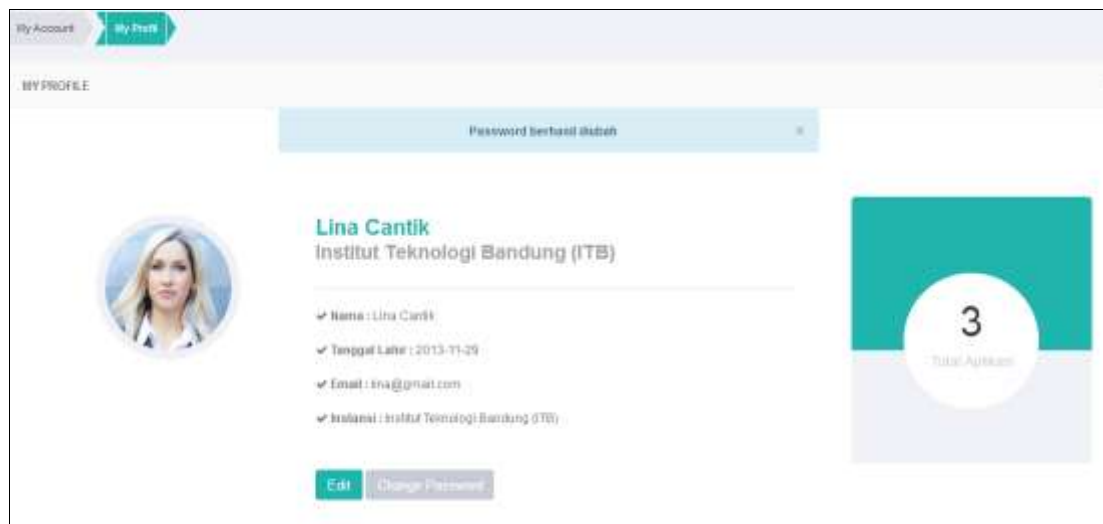
Password Lama

Password Baru

Confirm Password

Save Reset Cancel

Gambar 4.37 Masukan *edit password*



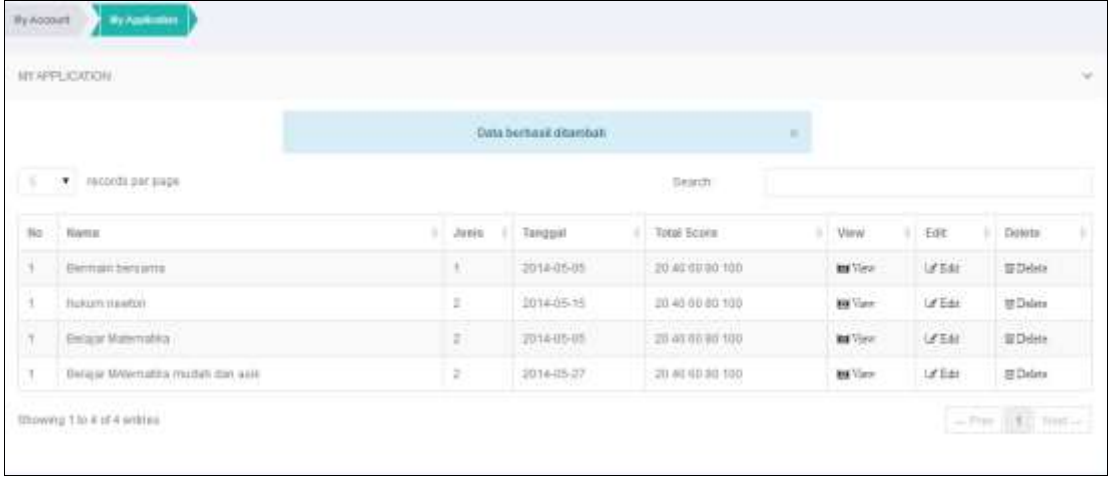
Gambar 4.38 Keluaran *edit password*

5. Masukan *upload* aplikasi

Data yang dimasukkan pada *upload* aplikasi dicontohkan pada gambar 4.39. Proses *upload* yang berhasil akan menampilkan halaman *my application* dengan menampilkan pesan bahwa aplikasi telah berhasil diunggah seperti terlihat pada gambar 4.40

 A screenshot of a web application's 'Upload Aplikasi' (Upload Application) form. The form is titled 'FORM UPLOAD APLIKASI' and is located under the 'Upload Aplikasi' tab. It contains several input fields: 'Judul' (Title) with the placeholder text 'Belajar Matematika mudah dan seru', 'Jenis' (Type) with a dropdown menu showing 'Aplikasi Sederhana', and 'Pilih File' (Select File) with a 'Choose File' button and the text 'Upload file dengan ekstensi file .swf'. Below the form fields are three buttons: 'Save' (green), 'Reset' (yellow), and 'Cancel' (grey).

Gambar 4.39 Masukan *upload*



My Account My Application

MY APPLICATION

Data berhasil ditambah

records per page Search

No	Nama	Jenis	Tanggal	Total Score	View	Edit	Delete
1	Bermian bersama	1	2014-05-05	20 40 00 00 100	View	Edit	Delete
1	Iskum newton	2	2014-05-15	20 40 00 00 100	View	Edit	Delete
1	Belajar Matematika	2	2014-05-05	20 40 00 00 100	View	Edit	Delete
1	Belajar Matematika mudah dan asik	2	2014-05-27	20 40 00 00 100	View	Edit	Delete

Showing 1 to 4 of 4 entries

Prev 1 Next

Gambar 4.40 Keluaran *upload*

6. Masukan *edit* aplikasi

Contoh masukan untuk *edit* aplikasi dapat dilihat pada gambar 4.41. Jika berhasil, sistem akan menampilkan pesan bahwa data berhasil diubah seperti terlihat pada gambar 4.42



My Account My Application

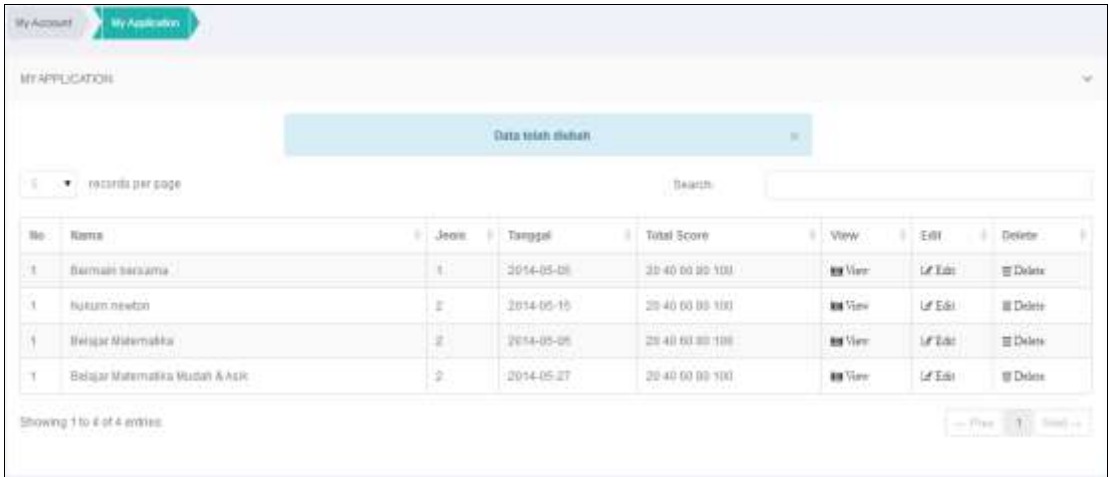
FORM EDIT APLIKASI

Nama: Belajar Matematika Mudah & Asik

Jenis Aplikasi: Ask Teacher App

Save Reset Cancel

Gambar 4.41 Masukan *edit* aplikasi



My Account My Application

MY APPLICATION

Data telah diubah

records per page Search

No	Nama	Jenis	Tanggal	Total Score	View	Edit	Delete
1	Bermian bersama	1	2014-05-05	20 40 00 00 100	View	Edit	Delete
1	Iskum newton	2	2014-05-15	20 40 00 00 100	View	Edit	Delete
1	Belajar Matematika	2	2014-05-05	20 40 00 00 100	View	Edit	Delete
1	Belajar Matematika Mudah & Asik	2	2014-05-27	20 40 00 00 100	View	Edit	Delete

Showing 1 to 4 of 4 entries

Prev 1 Next

Gambar 4.42 Keluaran *edit* aplikasi

7. Masukan uji aplikasi

Masukan pada uji aplikasi berupa data diri *tester* dan penilaian *tester* terhadap aplikasi. Pada tahap pertama *tester* diminta untuk mengisi data diri seperti pada gambar 5.43. Jika data telah dimasukkan maka akan tampil *form* pengujian seperti pada gambar 3.44. Gambar 3.45 merupakan tampilan sistem saat uji aplikasi berhasil dilakukan.

The screenshot displays the 'TIME FLASH' application interface. At the top, there is a navigation bar with links: HOME, TEST APLIKASI, LIST APLIKASI, STATISTIK APLIKASI, ABOUT US, and CONTACT US. Below this, a 'FORM TESTING' button is highlighted. The main content area is titled 'HUKUM NEWTON' and features a physics simulation. The simulation includes a pulley system with two masses, m_1 and m_2 , and a 'START' button. Input fields for 'Gravitational Acceleration' (set to 10 ms^{-2}), 'Mass object 1' (set to 1 kg), and 'Mass object 2' (set to 1 kg) are visible. Below the simulation, there are input fields for 'Nama*' and 'Email*', and 'Submit' and 'Reset' buttons.

Gambar 4.43 Masukan data diri *tester*



HOME
MAIN MENU

TEST APLIKASI
GET IN TOUCH

LIST APLIKASI
ALL OVER THE WORLD

STATISTIK APLIKASI
ALL OVER THE WORLD

ABOUT US
GET IN TOUCH

CONTACT US
GET IN TOUCH

FORM TESTING

You are here Form Testing

HUKUM NEWTON

Gravitational Acceleration = m/s²

START

Mass object 1 = kg

Input range (1 - 100) kg

Mass object 2 = kg

Input range (1 - 100) kg

Free Flash e-Learning  .com

Aspek Learnability
4

☐ Sangat Tidak Setuju ☐ Tidak Setuju ☐ Netral ☐ Setuju ☐ Sangat Setuju

Aspek Memorability
7

☐ Sangat Tidak Setuju ☐ Tidak Setuju ☐ Netral ☐ Setuju ☐ Sangat Setuju

Aspek Memorability
8

☐ Sangat Tidak Setuju ☐ Tidak Setuju ☐ Netral ☐ Setuju ☐ Sangat Setuju

Aspek Effolienoy
9

☐ Sangat Tidak Setuju ☐ Tidak Setuju ☐ Netral ☐ Setuju ☐ Sangat Setuju

Aspek Effolienoy
10

☐ Sangat Tidak Setuju ☐ Tidak Setuju ☐ Netral ☐ Setuju ☐ Sangat Setuju

Aspek Minimal Error Rate
14

☐ Sangat Tidak Setuju ☐ Tidak Setuju ☐ Netral ☐ Setuju ☐ Sangat Setuju

Aspek Minimal Error Rate
15

☐ Sangat Tidak Setuju ☐ Tidak Setuju ☐ Netral ☐ Setuju ☐ Sangat Setuju

Aspek High User Satisfaction
16

☐ Sangat Tidak Setuju ☐ Tidak Setuju ☐ Netral ☐ Setuju ☐ Sangat Setuju

Aspek Learnability
Apakah anda paham dengan alur kerja aplikasi ini?

☐ Sangat Tidak Setuju ☐ Tidak Setuju ☐ Netral ☐ Setuju ☐ Sangat Setuju

Gambar 4.44 Masukan data pengujian

No	Judul	Tanggal Upload	Developer	Instansi	Total Score	View	Uji Aplikasi
1	Alat bantu Ajar Hukum Newton	2014-05-15	Nico Bryan Pranata	Universitas Islam Indonesia	55.6	View	Uji
2	Permainan Seru Asik	2014-05-20	Nico Bryan Pranata	Universitas Islam Indonesia	1.8	View	Uji
3	Belajar munggi	2014-05-21	Fery	Universitas Islam Indonesia	0	View	Uji
4	Main belajar dengan aplikasi seru	2014-05-25	Nico Bryan Pranata	Universitas Islam Indonesia	0	View	Uji
5	Pecahan matematika menyenangkan	2014-05-22	nana	Universitas Islam Indonesia	0	View	Uji
6	Bermain Bersama sistem newton	2014-05-25	Lina Cantik	Institut Teknologi Bandung (ITB)	0	View	Uji
7	Belajar Matematika cendek	2014-05-26	Lina Cantik	Institut Teknologi Bandung (ITB)	3.6	View	Uji
8	Belajar Matematika Mudah & Asik	2014-05-27	herinappppoo	Universitas Islam Indonesia	0	View	Uji
9			Lina Cantik	Institut Teknologi Bandung (ITB)	0	View	Uji

Gambar 4.45 Keluaran hasil pengujian

8. Masukan kirim pesan *tester*

Contoh data yang dimasukan oleh *tester* untuk mengirim pesan dapat dilihat pada gambar 4.46. Sedangkan keluaran setelah memasukkan data dapat dilihat pada gambar 4.47.

Send us a message
Silahkan mengisi form untuk meninggalkan pesan kepada administrator. Pesan akan dibalas pada alamat email yang diisi.

Name =
Fenzal

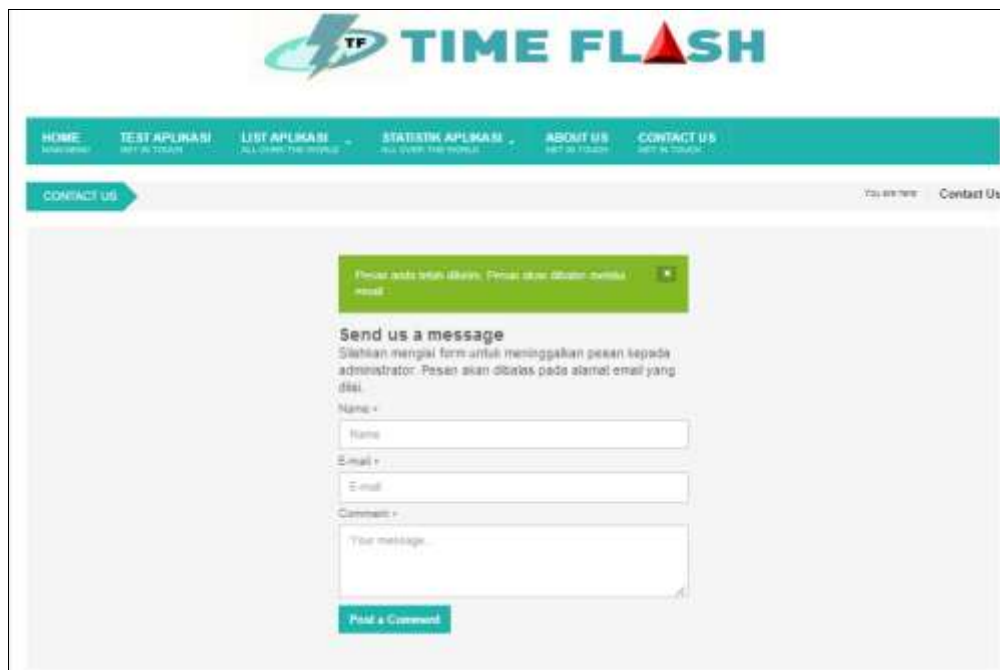
Email =
Fenzal@gmail.com

Comment =
Doain TA saya sukses ya min. D

Post a Comment

© Copyright TimeFlash 2014. All Rights Reserved.

Gambar 4.46 Masukan kirim pesan *tester*



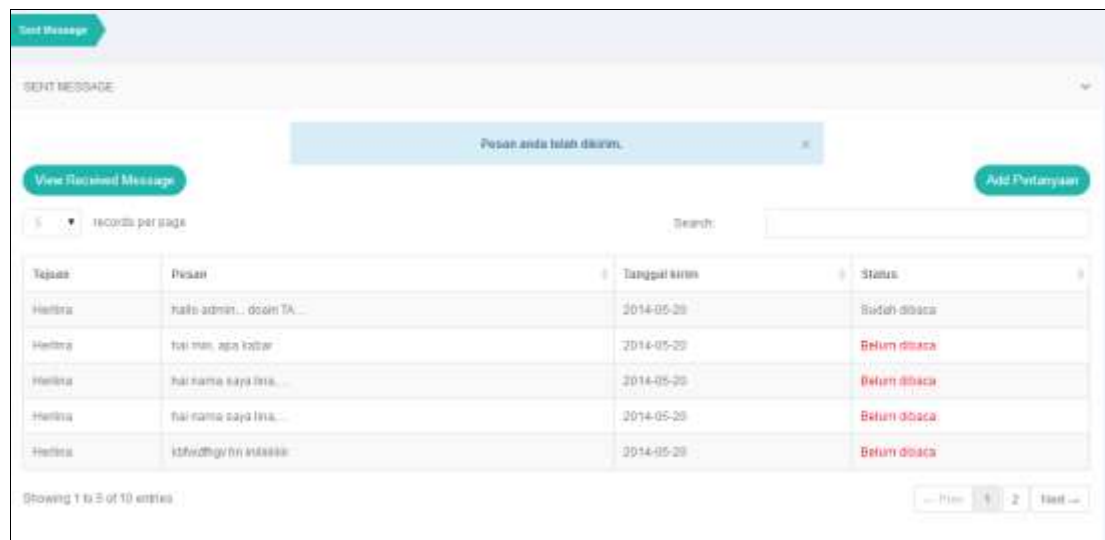
Gambar 4.47 Keluaran kirim pesan *tester*

9. Masukan kirim pesan *developer*

Data yang diisikan oleh *developer* untuk mengirim pesan kepada admin dapat dicontohkan seperti gambar 4.48 dan hasilnya digambarkan pada gambar 4.49.



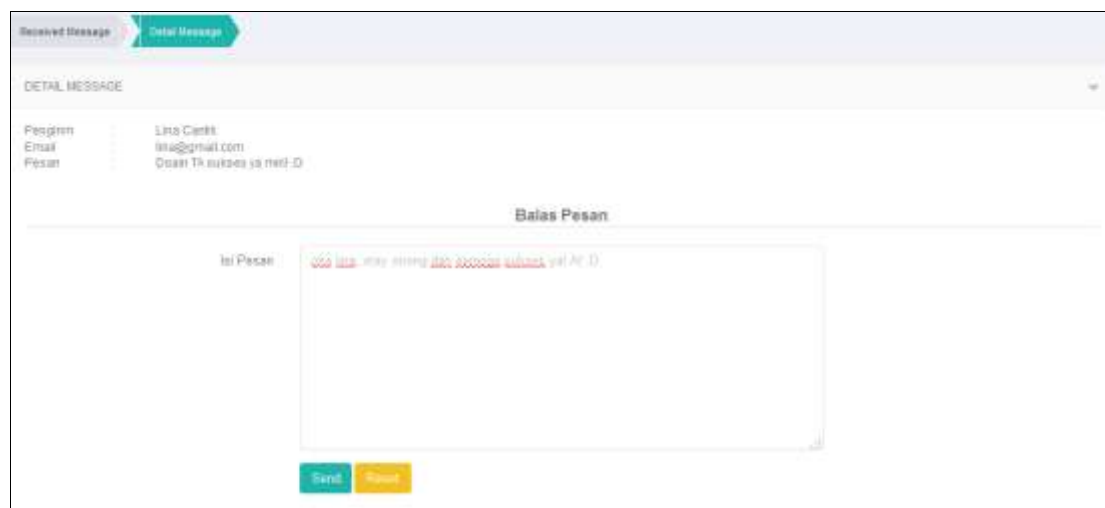
Gambar 4.48 Masukan *contact us*



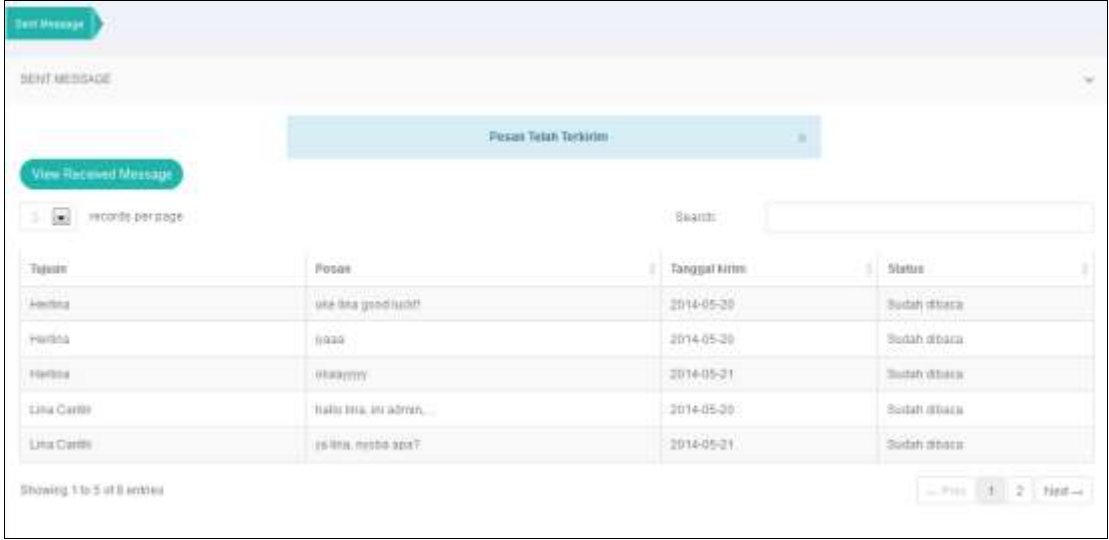
Gambar 4.49 Keluaran *contact us*

10. Masukan balas pesan

Contoh untuk data yang dimasukkan admin untuk membalas pesan kepada *developer* dapat dilihat pada gambar 4.50. keluaran setelah melakukan kirim pesan dapat dilihat pada gambar 4.51.



Gambar 4.50 Masukan balas pesan



The screenshot shows a web interface for 'Sent Message'. At the top, there's a 'Sent Message' header and a 'Pesan Telah Terkirim' (Message has been sent) notification. Below this is a 'View Received Message' button. A search bar is present with the text 'records per page' and a 'Search:' label. The main content is a table with the following data:

To	From	Date	Status
Admin	Ura Ma product	2014-05-20	Sudah dibaca
Admin	Ura Ma	2014-05-20	Sudah dibaca
Admin	Ura Ma	2014-05-21	Sudah dibaca
Lina Carli	Ura Ma, in admin	2014-05-20	Sudah dibaca
Lina Carli	Ura Ma, in admin	2014-05-21	Sudah dibaca

At the bottom, it says 'Showing 1 to 5 of 8 entries' and has pagination controls for 'Previous', '1', '2', and 'Next'.

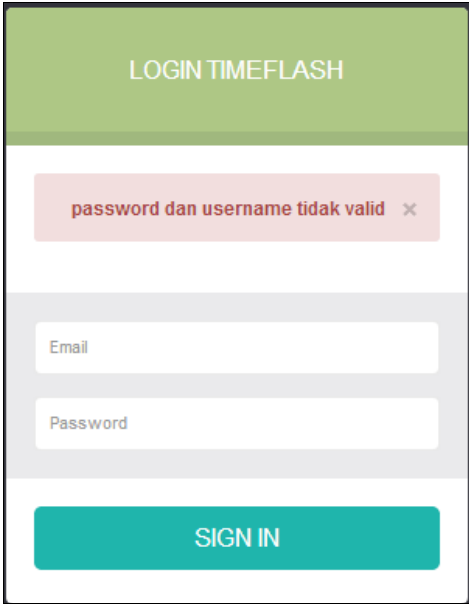
Gambar 4.51 Keluaran balas pesan

4.2.2 Penanganan kesalahan

Terkadang pengguna melakukan kesalahan dalam memasukkan data pada sistem. Kesalahan pengguna dapat diberi pemberitahuan agar pengguna mengetahui kesalahan yang dilakukan dan agar data yang dimasukkan ke *database* menjadi valid. Terdapat beberapa penanganan kesalahan yang dapat dilakukan oleh sistem, yaitu:

1. Penanganan kesalahan *login*

Kesalahan *login* terjadi ketika pengguna salah memasukkan *email* atau *password*. Sistem akan menampilkan pemberitahuan seperti gambar 4.52.



The screenshot shows a login form titled 'LOGIN TIMEFLASH'. It has a green header. Below the header is a red error message box that says 'password dan username tidak valid' with a close button. Below the error message are two input fields: 'Email' and 'Password'. At the bottom is a large green button labeled 'SIGN IN'.

Gambar 4.52 Penanganan kesalahan *login*

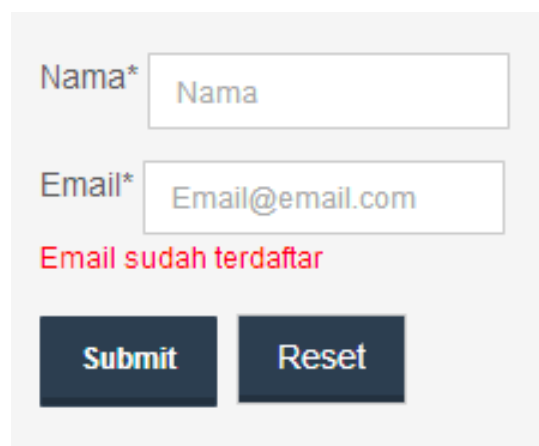
2. Penanganan kesalahan *email*

Kesalahan terjadi saat pendaftaran ketika pengguna memasukkan *email* yang sudah terdaftar sebagai *developer* di sistem TIMEFLASH. Saat terjadi kesalahan maka sistem akan menampilkan pesan kesalahan seperti gambar 4.53.

Kesalahan juga terjadi saat pengguna memasukkan *email* ketika mendaftarkan diri untuk menjadi *developer* di sistem TIMEFLASH dan saat pengguna mengubah data diri. Kesalahan untuk validasi *email* juga terjadi ketika *tester* mengisi *email* saat akan mengirim pesan dan saat akan memasukkan data diri pada tahap pengujian. Pengguna diharuskan memasukkan alamat *email* yang valid yaitu alamat *email* yang terdiri atas karakter untuk alamat *email*, simbol @ dan domain *email* seperti gmail.com. Pengguna yang memasukkan *email* dengan format yang tidak valid maka akan diberi pemberitahuan seperti pada gambar 4.54.

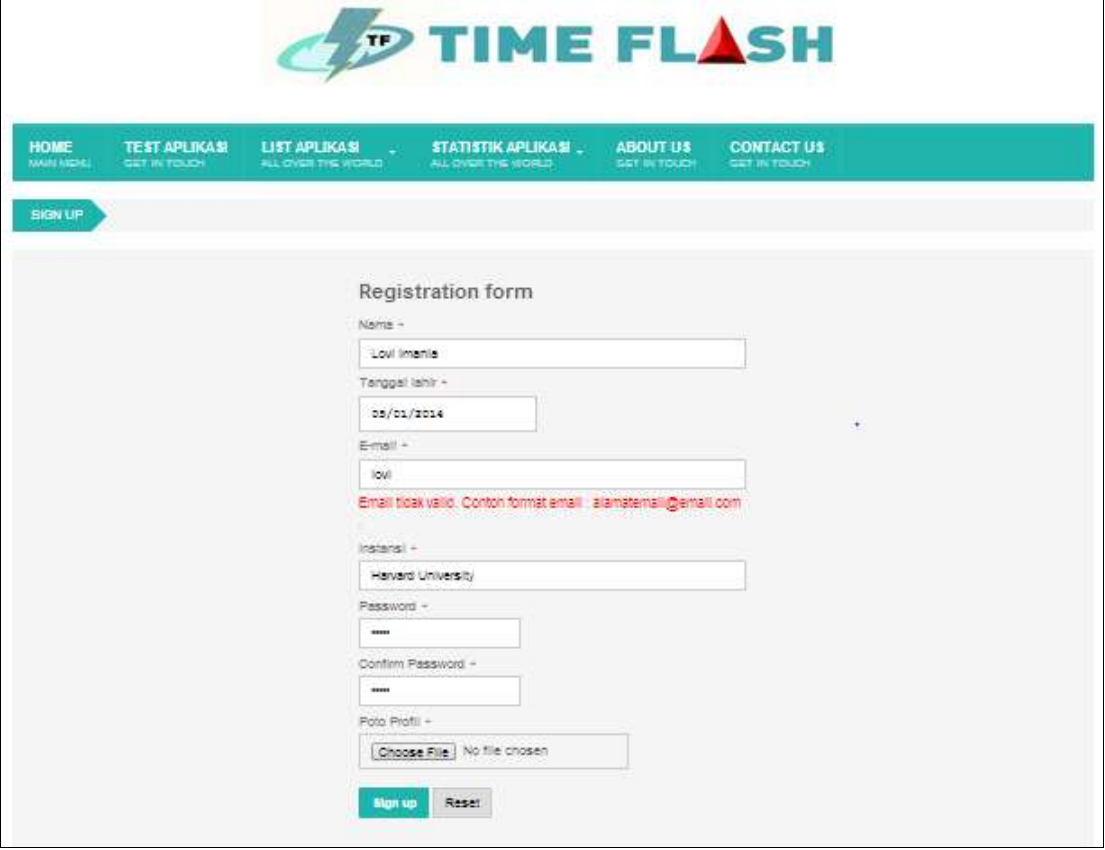
3. Penanganan kesalahan *password*

Terdapat 3 jenis penanganan kesalahan pada *password*. Saat pengguna salah memasukkan *password* yang kurang dari 5 karakter, maka akan tampil peringatan seperti pada gambar 4.55. Saat pengguna akan mengubah *password*, pengguna diminta untuk memasukkan *password* lama. Apabila *password* lama yang dimasukkan tidak sesuai, maka akan tampil peringatan kesalahan seperti pada gambar 4.56. Dan terakhir saat pengguna mengubah *password*, pengguna diminta untuk mengetik ulang *password* baru. Apabila *password* baru yang dimasukkan tidak sesuai dengan *password* konfirmasi, maka akan tampil peringatan seperti pada gambar 4.57.



The image shows a login form with two input fields: 'Nama*' containing 'Nama' and 'Email*' containing 'Email@email.com'. Below the email field, a red error message reads 'Email sudah terdaftar'. At the bottom, there are two dark blue buttons labeled 'Submit' and 'Reset'.

Gambar 4.53 Penanganan kesalahan *email* terdaftar



The screenshot shows the 'Registration form' on the TIME FLASH website. The form includes fields for Name, Date of Birth, Email, Institution, Password, Confirm Password, and Profile Photo. The Email field contains 'lovi' and a red error message is displayed below it: 'Email tidak valid. Contoh format email: alamatemail@email.com'. The Name field contains 'Lovi Imanita', the Date of Birth field contains '05/01/2014', the Institution field contains 'Harvard University', the Password and Confirm Password fields contain '****', and the Profile Photo field shows a 'Choose File' button and 'No file chosen'.

Gambar 4.54 Penanganan kesalahan *email* tidak valid



The screenshot shows the 'FORM EDIT PASSWORD' form. At the top, there is a red error message: 'Password minimal mencapai 5 karakter'. Below this, there are three input fields: 'Password Lama', 'Password Baru', and 'Confirm Password'. At the bottom, there are three buttons: 'Save', 'Reset', and 'Cancel'.

Gambar 4.55 Penanganan kesalahan jumlah karakter *password*

The screenshot shows a web form titled 'Edit Password'. At the top, there is a green button labeled 'Edit Password'. Below it, the form is titled 'FORM EDIT PASSWORD'. A red error message box at the top of the form area says 'Password lama salah'. Below this, there are three input fields: 'Password Lama', 'Password Baru', and 'Confirm Password'. At the bottom of the form, there are three buttons: 'Save' (green), 'Reset' (yellow), and 'Cancel' (grey).

Gambar 4.56 Penanganan kesalahan validasi *password*

The screenshot shows the same 'Edit Password' form. The red error message box now says 'Password tidak sesuai'. The input fields and buttons remain the same.

Gambar 4.57 Penanganan kesalahan *password* konfirmasi

4. Penanganan kesalahan *upload*

Saat pengguna mengunggah *file* aplikasi dengan ekstensi yang tidak diizinkan, maka sistem akan menampilkan pesan kesalahan seperti pada gambar 4.58.

The screenshot shows a web form titled 'Upload Aplikasi'. At the top, there is a green button labeled 'Upload Aplikasi'. Below it, the form is titled 'FORM UPLOAD APLIKASI'. The form has three main sections: 'Judul' (Title) with a text input field, 'Jenis' (Type) with a dropdown menu, and 'Pilih File' (Select File) with a 'Choose File' button. Below the 'Pilih File' section, there is a red error message: 'Ekstensi file tidak diizinkan, silahkan upload file dengan ekstensi .jar'. At the bottom of the form, there are three buttons: 'Save' (green), 'Reset' (yellow), and 'Cancel' (grey).

Gambar 4.58 Penanganan kesalahan *upload*

5. Penanganan kesalahan data kosong

Ketika pengguna tidak memasukkan data yang harus diisi pada *form* maka sistem akan menampilkan pesan kesalahan seperti terlihat pada gambar 4.59.

The screenshot shows a web form titled "FORM UPLOAD APLIKASI". It contains three input fields: "Judul", "Jenis", and "Pilih File". The "Judul" field has a red error message "Nama Aplikasi harus diisi". The "Jenis" field has a red error message "Jenis harus diisi". The "Pilih File" field has a red error message "Pilih File harus diisi". Below the fields are three buttons: "Save", "Reset", and "Cancel".

Gambar 4.59 Penanganan kesalahan data kosong

4.2.3 Pengujian URL (Uniform Resource Locator) Injection

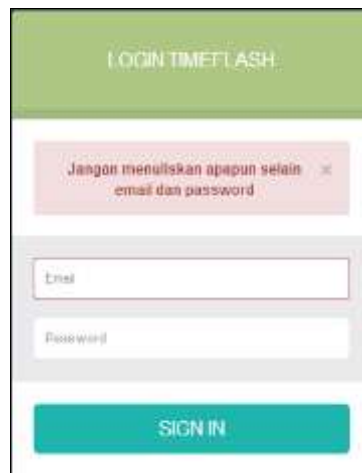
Pengujian URL *injection* merupakan pengujian terhadap URL. Pengujian ini dilakukan untuk memastikan bahwa pengguna tidak dapat mengakses halaman admin dan *developer* tanpa melakukan proses *login*. Walau telah mengisi URL untuk mengakses halaman admin atau *developer*, pengguna tidak dapat masuk ke halaman tersebut tanpa proses *login* dan kemudian akan ditampilkan peringatan seperti pada gambar 4.60.

The screenshot shows a login form titled "LOGIN TIMEFLASH". It has a green header bar with the title. Below the header is a red warning message box that says "Silahkan login terlebih dahulu". Below the warning message are two input fields: "Email" and "Password". At the bottom of the form is a green button labeled "SIGN IN".

Gambar 4.60 Pengujian URL *injection*

4.2.4 Pengujian SQL (Structured Query language) Injection

SQL *injection* merupakan kegiatan untuk mengakses *database* secara ilegal. Contoh ketika pengguna ingin masuk ke sistem tanpa menggunakan *email* dan *password* yang tidak valid. Misalnya Pengguna memasukkan *email@gmail.com* or 'x' sebagai *email* dan *password* or 'x'='x' sebagai *password* untuk dapat mengakses *database*. Untuk mencegah hal tersebut, sistem memberi keamanan terhadap *database* yang jika terjadi akan menampilkan peringatan seperti gambar 4.61.

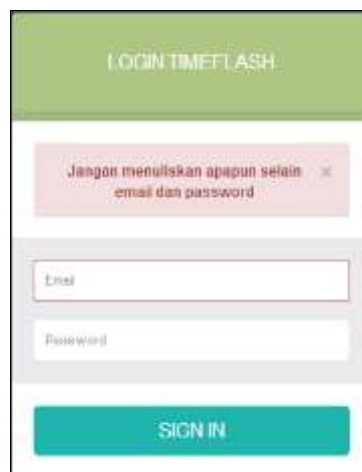


The image shows a login interface with a green header bar containing the text 'LOGIN TIMEFLASH'. Below the header is a light pink rectangular box with a warning message: 'Jangan menuliskan apapun selain x email dan password'. Underneath this box are two input fields: the first is labeled 'Email' and the second is labeled 'Password'. At the bottom of the form is a teal button with the text 'SIGN IN'.

Gambar 4.61 Pengujian SQL *injection*

4.2.5 Pengujian XSS (Cross Side Scripting)

Pengujian XSS juga dilakukan pada sistem ini. Pengujian XSS merupakan pengujian yang dilakukan untuk memastikan bahwa sistem tidak dapat diserang dengan cara memasukkan suatu karakter seperti tag `<script>`. Jika pengguna mencoba melakukan XSS terhadap sistem maka akan ditampilkan peringatan seperti pada gambar 4.62.



This image is identical to the one in Gambar 4.61. It shows a login interface with a green header bar containing the text 'LOGIN TIMEFLASH'. Below the header is a light pink rectangular box with a warning message: 'Jangan menuliskan apapun selain x email dan password'. Underneath this box are two input fields: the first is labeled 'Email' and the second is labeled 'Password'. At the bottom of the form is a teal button with the text 'SIGN IN'.

Gambar 4.62 Pengujian XSS injection

4.2.6 Pengujian kepuasan oleh pengguna

Pada penelitian ini juga dilakukan pengujian kepuasan pengguna terhadap sistem pengujian aplikasi *flash*. Pengujian dilakukan oleh 5 orang *developer game*. Pengujian dilakukan dengan mengisi kuesioner yang telah disediakan.

Dari hasil pengujian, diperoleh penilaian untuk pertanyaan yang bernilai 5 atau sangat setuju sebanyak 18.0%. Penilaian untuk pertanyaan yang bernilai 4 atau setuju sebanyak 60.0%. Penilaian untuk pertanyaan yang bernilai 3 atau netral sebanyak 22.0%. Penilaian untuk pertanyaan yang bernilai 2 sebanyak 0.0%. Penilaian untuk pertanyaan yang bernilai 1 atau sangat tidak setuju berjumlah 0.0%. Berdasarkan hasil pengujian, dapat disimpulkan bahwa sistem telah dapat memuaskan pengguna untuk digunakan sebagai sistem pengujian aplikasi *flash*.

4.2.7 Hasil Analisis

Berdasarkan hasil pengujian yang dilakukan maka dapat diambil hasil dari kinerja TIMEFLASH, sistem pengujian aplikasi *flash*. Berikut merupakan hasil dari analisis yang didapat:

1. Sistem telah dapat digunakan untuk menguji aplikasi *flash* untuk jenis *game* dan alat bantu ajar.
2. Sistem telah dapat menampilkan hasil pengujian aplikasi baik dalam bentuk teks maupun diagram.
3. Sistem telah dapat menampilkan informasi mengenai aplikasi yang diunggah.
4. Sistem telah dapat menampilkan informasi mengenai *developer* aplikasi *flash* yang terdaftar di TIMEFLASH.
5. Sistem telah mampu menangani kesalahan yang dilakukan oleh pengguna, baik kesalahan yang tidak disengaja atau kesalahan yang disengaja.

4.2.8 Kelebihan dan Kekurangan

Dalam membangun suatu sistem, tentu ditemukan kelebihan dan kekurangan. Adapun kelebihan yang terdapat pada TIMEFLASH, sistem pengujian aplikasi *flash* untuk *game* dan alat bantu ajar adalah sebagai berikut:

1. Memudahkan *developer* untuk melakukan pengujian aplikasi pada aspek *usability*. Data pengujian yang dibutuhkan dengan lebih cepat didapat.
2. Memudahkan *tester* dalam pengujian, karena sistem dapat menampilkan aplikasi dan pertanyaan pengujian dalam satu halaman.

3. Memudahkan proses pengujian, karena sistem berbasis *website* sehingga *tester* tidak harus menguji ditempat tertentu.
4. Memudahkan pengguna yang membutuhkan *developer* aplikasi *flash*, karena sistem menampilkan informasi mengenai *developer*.
5. Memudahkan pengguna sistem untuk mengetahui kesalahan yang dilakukan kesalahan karena sistem menampilkan pesan setiap kali terjadi kesalahan.

Selain kelebihan, sistem juga mempunyai kekurangan yaitu:

1. Dibutuhkan koneksi yang baik agar aplikasi *flash* dapat berjalan dengan sempurna.

BAB V

KESIMPULAN DAN SARAN

5.1 Kesimpulan

Berdasarkan hasil analisis, perancangan, pembuatan sistem hingga tahap implementasi maka dapat ditarik kesimpulan seperti berikut:

1. Pengujian aplikasi dengan metode *exploratory* telah berhasil dikembangkan untuk menguji aplikasi *game* dan alat bantu ajar
2. Sistem dapat dengan baik melakukan pengujian pada aspek *usability* dari tahap *upload* hingga menampilkan hasil pengujian.
3. Sistem dapat mempermudah *tester* dalam melakukan pengujian.
4. Berdasarkan hasil pengujian, sistem telah berjalan dengan baik dan telah berhasil lolos uji keamanan data.

5.2 Saran

Terdapat beberapa kekurangan pada sistem pengujian ini, sehingga diharapkan dapat dikembangkan menjadi lebih baik. Adapun saran untuk penelitian yang akan datang adalah:

1. Mengembangkan lebih lanjut agar sistem dapat digunakan oleh semua jenis aplikasi *flash*.
2. Sistem dapat menguji aplikasi yang terdiri dari beberapa *file*.

DAFTAR PUSTAKA

- [1] Kristanto, Andri. (2004). *Rekayasa Perangkat Lunak (Konsep Dasar)*. Yogyakarta: Gava Media.
- [2] Rizky, Soetam. (2007). *Interaksi Manusia Dan Komputer*. Yogyakarta: Graha Ilmu.
- [3] Santoso, Insap. (2010). *Interaksi Manusia Dan Komputer (Edisi 2)*. Yogyakarta: Andi.
- [4] Simarmata, Janner. (2004). *Rekayasa Perangkat Lunak*. Yogyakarta: Andi.
- [5] Sudawarman dan Dony Ariyus. (2007). *Interaksi Manusia & Komputer*. Yogyakarta: Andi.
- [6] Wicaksono, Tata. S. (2009). *White-box black-box testing*. Diakses pada tanggal 17 Maret 2014, dari website wordpress : <http://tataswicaksono.wordpress.com/2009/11/04/white-box-black-box-testing-2/>.

LAMPIRAN

1. Kuesioner kepuasan oleh pengguna
2. Paper penelitian

**Kuesioner Pengujian Sistem pengujian aplikasi *flash*
dengan metode *exploratory testing***

No	Keterangan	STS	TS	N	S	SS
1	Sistem menyediakan mekanisme yang jelas dan mudah dimengerti untuk mengunggah aplikasi					
2	Sistem menyediakan mekanisme pengujian aplikasi secara jelas dengan kriteria pengujian sesuai dengan jenis aplikasi					
3	Sistem menyediakan skor hasil pengujian yang mampu memberikan gambaran perbandingan kualitas antar aplikasi					
4	Informasi detail yang disediakan oleh sistem telah cukup untuk memberikan gambaran mengenai masing-masing aplikasi					
5	Sistem memberikan informasi yang jelas mengenai hasil skor yang diperoleh					
6	Informasi dan pesan kesalahan (<i>error message</i>) yang ditampilkan sistem dapat dengan mudah dipahami pengguna					
7	Alur penggunaan sistem mudah dipelajari					
8	Sistem memiliki tampilan yang menarik dengan pemilihan ukuran dan warna elemen antarmuka yang nyaman bagi pengguna					
9	Sistem memiliki tata letak elemen antarmuka yang memudahkan bagi pengguna					
10	Sistem memiliki potensi dan manfaat yang besar sebagai <i>tools</i> pengujian aplikasi <i>flash</i>					

SISTEM PENGUJIAN APLIKASI *FLASH* DENGAN METODE *EXPLORATORY TESTING*

Herlina

Jurusan Teknik Informatika
Universitas Islam Indonesia
Yogyakarta, Indonesia
herlina.riau@gmail.com

Beni Suranto

Jurusan Teknik Informatika
Universitas Islam Indonesia
Yogyakarta, Indonesia
beni.suranto@gmail.com

Pengujian merupakan suatu tahapan penting dalam membangun perangkat lunak. Melalui tahap pengujian, dapat diketahui kekurangan dari perangkat lunak. Salah satu jenis pengujian adalah pengujian usability. Pada pengujian ini dilakukan pengujian terhadap aspek-aspek usability pada perangkat lunak. Penelitian ini melakukan pengujian usability terhadap aplikasi flash. Aplikasi flash merupakan aplikasi yang banyak digunakan oleh mahasiswa teknik informatika UII untuk dijadikan sebagai tugas akhir khususnya aplikasi flash untuk game dan alat bantu ajar.

Pada penelitian ini dikembangkan sistem pengujian aplikasi flash menggunakan framework CodeIgniter dengan bahasa pemrograman PHP dan basisdata MYSQL. Aspek usability yang diterapkan untuk pengujian terdiri dari learnability, memorability, efficiency, minimal error rate, dan high user satisfaction.

Simpulan yang didapat berupa sistem yang dapat menguji aplikasi flash dengan menampilkan aplikasi dan form pertanyaan pengujian yang akan diisi oleh tester. Aplikasi yang telah diuji dapat diketahui score hasil pengujian sehingga developer dapat mengetahui aspek mana yang kurang diterapkan dalam membangun aplikasi. Melalui pengujian ini developer dapat memperbaiki kekurangan yang ada. Manfaat bagi developer lainnya, dapat mengetahui contoh aplikasi yang dapat dijadikan rujukan dalam membangun aplikasi flash.

Kata kunci:

Sistem pengujian aplikasi flash, pengujian usability, flash, game, alat bantu ajar

I. PENDAHULUAN

Dalam ilmu rekayasa perangkat lunak, terdapat beberapa langkah yang dilakukan untuk membangun suatu aplikasi. Beberapa di antaranya yaitu perancangan, implementasi dan pengujian perangkat lunak. Perangkat lunak dibangun sesuai rancangan, kemudian dilakukan pengujian. Pengujian merupakan tahapan penting dalam mengembangkan perangkat lunak. Pengembang perangkat lunak bisa jadi merasa sudah membangun sistem atau aplikasi sesuai yang diinginkan pengguna, tetapi ketika digunakan masih ditemukan beberapa kekurangan, untuk itu diperlukan tahap pengujian. Pengujian berguna untuk mengetahui apakah sistem yang sudah dibuat sudah sesuai dengan apa yang diharapkan atau belum.

Untuk mengukur kepuasan pengguna pada aplikasi flash yang digunakan, perlu dilakukan pengujian usability oleh para pengguna terhadap aplikasi tersebut. Pengujian ini berguna untuk mendapat respon pengguna terhadap performa aplikasi yang digunakan. Beberapa hal yang diuji seperti kemudahan penggunaan, penyajian informasi, alur penggunaan aplikasi dan lainnya.

Salah satu metode pengujian usability adalah metode *exploratory testing*. Pada pengujian ini pengguna diminta untuk mengeksplorasi atau menggunakan aplikasi yang diuji. Selanjutnya pengguna diharapkan dapat memberikan respon berupa saran dan kritik sebagai evaluasi aplikasi.

Pada penelitian ini pengujian dilakukan terhadap aplikasi flash. Aplikasi flash merupakan salah satu aplikasi yang semakin banyak digunakan pada setiap tingkatan umur. Misalnya saja, pada aplikasi game. Aplikasi game merupakan salah satu aplikasi yang banyak digemari, baik dari kalangan anak-anak hingga kalangan dewasa. Contoh lainnya terdapat pada alat bantu ajar. Aplikasi flash digunakan untuk membantu para pelajar agar lebih mudah dalam memahami apa yang dipelajari. Penggunaan flash

dalam beberapa aplikasi dikarenakan penampilannya yang menarik dan interaktif sehingga lebih mudah digunakan oleh para pengguna.

Jenis aplikasi *flash* yang diuji dalam penelitian ini adalah aplikasi *game* dan alat bantu ajar. Kedua jenis aplikasi tersebut merupakan jenis yang banyak digunakan oleh mahasiswa teknik informatika untuk menyelesaikan tugas akhir di Universitas Islam Indonesia.

II. TEORI PENDUKUNG

Berikut ini adalah teori yang mendukung penelitian :

A. Usability

Usability berasal dari kata *usable* yang berarti dapat digunakan dengan baik (Rizky, 2007 : 61). Definisi *usability* sendiri, menurut ISO (*International Standard Organization*) adalah *Usability is the extends to which a product can be used by specified users to achieve specified goals with effectiveness, efficiency and satisfaction in a specified context of use.* *Usability* adalah seberapa jauh sebuah produk atau sistem bisa digunakan oleh *user* yang spesifik untuk mencapai tujuan yang spesifik pula secara efektif, efisien dan memuaskan pada konteks penggunaan tertentu. Usabilitas merupakan suatu aspek yang menggambarkan tingkat kenyamanan dan penerimaan pengguna terhadap sistem yang digunakan. Terdapat beberapa faktor yang digunakan untuk mengukur tingkat usabilitas suatu sistem, yaitu:

1. *Learnability* (mudah dipelajari), Menunjukkan kualitas sistem agar mudah dipelajari, terutama bagi *user* baru (Santoso, 2010 : 75).
2. *Memorability* (mudah diingat), Sistem diberi petunjuk atau diberi tampilan tertentu agar *user* mudah mengingat fitur yang disediakan sistem.
3. *Efficiency* (efisien), Merupakan faktor yang dilakukan sistem untuk mendukung pengguna dalam menyelesaikan pekerjaannya (Santoso, 2010 : 76).
4. *Minimal error rate* (minim kesalahan), Pengembang sistem hendaknya memperhatikan hal ini. sistem dirancang sedemikian rupa sehingga *user* terhindar dari kesalahan.
5. *High user satisfaction* (kepuasan), Menunjukkan tingkat kepuasan yang dirasakan oleh *user* setelah menggunakan sistem.

B. Metode Exploratory Testing

Metode *exploratory testing* merupakan salah satu metode pengujian pada pengujian *usability*. Uji eksploratori yang termasuk dalam kategori *black-box testing*, bertujuan untuk

menguji sebuah sistem dan mencari titik-titik di mana pengguna mengalami kebingungan, kesalahan, atau unjuk kerjanya melambat. Pengujian ini dilakukan tanpa melihat dimana persoalan-persoalan terjadi atau bentuknya. Tujuan akhir dari uji eksploratori adalah daftar persoalan yang perlu ditinjau lebih lanjut. Uji eksploratori dapat digunakan pada setiap tahap pengembangan sistem, tetapi paling efektif apabila diimplementasikan sejak awal dan sering dilakukan (Santoso, 2010 : 79).

Pada pengujian eksploratori pengguna dapat melakukan pengujian tanpa persiapan dan penjadwalan khusus sebelumnya. Pengujian ini tidak memerlukan dokumen khusus yang biasa digunakan untuk menentukan langkah-langkah tertentu yang harus dilakukan pada pengujian. Pada pengujian eksploratori dapat mempelajari aplikasi dengan cara menjalankan aplikasi yang tersedia. Selanjutnya pengguna dapat melakukan pengujian sesuai panduan sistem.

III. ANALISIS SISTEM

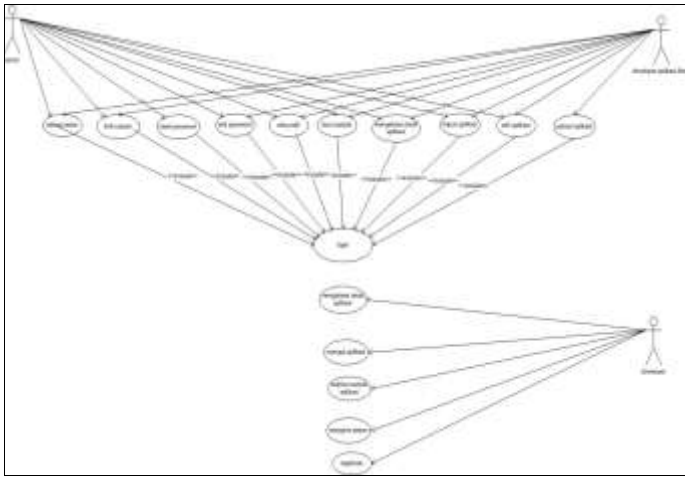
Berdasarkan analisis yang dilakukan diperlukan beberapa proses utama yang dibutuhkan pada sistem ini. Adapun tahapan-tahapan tersebut adalah:

1. *Tester* mendaftar sebagai *developer*.
2. *Developer* mengunggah aplikasi yang akan diuji.
3. *Tester* melakukan pengujian terhadap aplikasi.
4. Setelah data pengujian dimasukkan, *developer* dapat melihat *score* aplikasi yang diunggah.
5. Selain itu, semua pengguna sistem dapat membandingkan *score* semua aplikasi yang ditampilkan sistem dalam bentuk tabel dan diagram.

IV. PERANCANGAN

Sistem pengujian aplikasi *flash* ini dibuat dengan menggunakan *framework* Codeigniter dengan bahasa pemrograman PHP dan basisdata MYSQL. Program ini dibuat untuk memudahkan pengembang aplikasi dalam menguji aplikasi yang dibuat.

Tahap perancangan pada sistem ini menggunakan *use case diagram*. Terdapat 3 aktor yang berperan, yaitu *admin*, *developer* dan *tester*. Untuk diagram *use case* dapat dilihat lebih detail pada gambar 1.



Gambar 1. Rancangan *use case*

Aspek yang diuji pada aplikasi adalah aspek *usability* yang terdiri dari aspek *learnability*, *memorability*, *efficiency*, *minimal error rate* dan *high user satisfaction*.

Pada tahap pengujian *tester* diminta untuk mengisi pertanyaan mengenai aplikasi. Pertanyaan yang diujikan untuk aplikasi *game* adalah sebagai berikut

6. Aspek *learnability*
 - a. Alur permainan *game* ini mudah dipelajari
 - b. *Game* ini memberikan petunjuk penggunaan yang jelas
 - c. Mekanisme penghitungan skor atau poin pada *game* ini mudah dipahami
 - d. Anda mengerti cara menghitung *score* yang anda peroleh.
7. Aspek *memorability*
 - a. Anda dengan mudah menghafal alur *game*.
 - b. Anda dengan mudah menghafal lokasi dan fungsi elemen antarmuka *game*
8. Aspek *efficiency*
 - a. Anda dengan mudah dan cepat mampu mengakses fungsionalitas yang disediakan *game*
 - b. *Game* dapat dimainkan secara baik tanpa membutuhkan spesifikasi *hardware* yang tinggi
 - c. Informasi yang disediakan *game* ditampilkan dengan format yang ringkas dan informatif
9. Aspek *minimal error rate*
 - a. Anda tidak menemukan kesalahan teknis pada *game* ini.
 - b. Penanganan kesalahan di *game* ini sudah baik
 - c. Petunjuk dan informasi yang disediakan *game* mampu membantu Anda untuk menghindari kesalahan
10. Aspek *high user satisfaction*

- a. *Game* ini memberikan hiburan yang sesuai dengan ekspektasi Anda
- b. *Game* ini sangat menarik untuk sering Anda mainkan
- c. Tampilan *game* sangat menarik dengan desain antarmuka yang sesuai kebutuhan Anda
- d. Perbedaan tingkat kesulitan di tiap level pada *game* sesuai dengan ekspektasi Anda.

Sedangkan pertanyaan untuk alat bantu ajar adalah sebagai berikut:

6. Aspek *learnability*
 - a. Anda dapat dengan mudah memahami tahap pembelajaran pada alat bantu ajar ini.
 - b. Anda dapat memahami kegunaan dari fungsionalitas yang disediakan pada alat bantu ajar ini
 - c. Alat bantu ajar ini memberikan petunjuk penggunaan yang jelas.
7. Aspek *memorability*
 - a. Anda dengan mudah menghafal tahapan pembelajaran pada alat bantu ajar ini.
 - b. Anda dengan mudah menghafal lokasi dan fungsi elemen antarmuka alat bantu ajar.
8. Aspek *efficiency*
 - a. Anda dengan mudah dan cepat mampu mengakses fungsionalitas yang disediakan oleh alat bantu ajar ini.
 - b. Dengan menggunakan alat bantu ajar ini, anda membutuhkan waktu yang lebih singkat dalam memahami materi.
 - c. Alat bantu ajar ini membantu anda untuk memahami materi dengan tahapan pembelajaran yang ringkas dan mudah dipahami.
9. Aspek *minimal error rate*
 - a. Anda tidak menemukan kesalahan teknis pada alat bantu ajar ini.
 - b. Penanganan kesalahan pada alat bantu ajar sudah baik.
 - c. Petunjuk dan informasi pada alat bantu ajar mampu membantu anda untuk menghindari kesalahan.
10. Aspek *high user satisfaction*
 - a. Alat bantu ajar ini sangat membantu anda dalam memahami materi yang diajarkan.
 - b. Anda lebih memahami materi yang disampaikan melalui alat bantu ajar ini dibandingkan dengan materi yang disampaikan secara konvensional.
 - c. Tampilan alat bantu ajar sangat menarik dan dapat menambah minat anda untuk menggunakannya.

Alat bantu ajar ini menyajikan materi dengan cara yang interaktif dan mudah dipahami.

Setelah pengujian, sistem melakukan perhitungan *score* aplikasi. Adapun rumus yang digunakan untuk perhitungan *score* adalah:

$$\frac{(\text{nilai yang didapat saat pengujian} + \text{jumlah tester})}{(5 \times \text{jumlah pertanyaan})} \times \text{bobot aspek}$$

V. IMPLEMENTASI SISTEM

Sistem ini mengimplementasikan beberapa menu utama, yaitu :

1. Registrasi *developer*

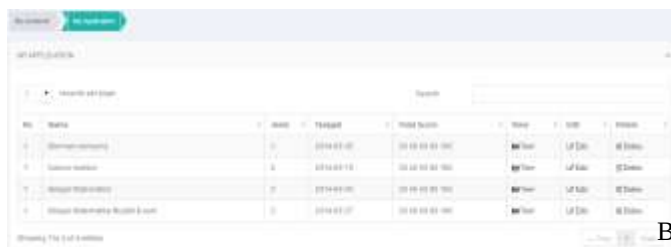
Digunakan oleh *tester* untuk mendaftar sebagai *developer* di timeflash. Pengguna diminta untuk memasukkan beberapa data yang setelah proses *login* akan ditampilkan pada menu *my profil*. Tampilan untuk informasi profil dapat dilihat pada gambar 2.



Gambar 2. Halaman profil *developer*

2. Upload aplikasi

Digunakan oleh *developer* untuk mengunggah aplikasi yang akan diuji. Informasi mengenai aplikasi yang telah diunggah ditampilkan pada menu *my application*. Tampilan untuk halaman *my application* dapat dilihat pada gambar 4.



Gambar 3. Halaman *my application*.

3. Uji aplikasi.

Digunakan oleh *tester* untuk menguji aplikasi. Aplikasi yang telah diuji, dapat dilihat detail *score* nya pada menu *view* pada halaman daftar aplikasi. Tampilan untuk detail aplikasi dapat dilihat pada gambar 4. Pada tahap ini kader dapat melakukan cetak laporan hasil kegiatan kesehatan yang ada di posyandu lansia tirtaadi, dengan cara memilih menu kelola laporan kesehatan, selanjutnya akan muncul halaman hasil kegiatan kesehatan.

4. View statistik

Menu ini menampilkan *score* semua aplikasi dalam bentuk diagram. Tampilan untuk *view statistik* dapat dilihat pada gambar 5.



Gambar 4. Halaman detail aplikasi



Gambar 5. Halaman *view statistik*

VI. KESIMPULAN

Berdasarkan hasil analisis, perancangan, pembuatan sistem hingga tahap implementasi maka dapat ditarik kesimpulan seperti berikut:

1. Sistem dapat dengan baik melakukan pengujian pada aspek *usability* terhadap aplikasi *flash* untuk *game* dan

alat bantu ajar dari tahap *upload* hingga menampilkan hasil pengujian.

2. Sistem dapat menambah antusias *developer* aplikasi *flash* untuk melakukan pengujian terhadap sistem yang dibuat.
3. Sistem dapat mempermudah *tester* dalam melakukan pengujian.
4. Berdasarkan hasil pengujian, sistem telah berjalan dengan baik dan aman.

VII. REFERENSI

- [1] Kristanto, Andri. (2004). *Rekayasa Perangkat Lunak (Konsep Dasar)*. Yogyakarta: Gava Media.
- [2] Rizky, Soetam. (2007). *Interaksi Manusia Dan Komputer*. Yogyakarta: Graha Ilmu.
- [3] Santoso, Insap. (2010). *Interaksi Manusia Dan Komputer (Edisi 2)*. Yogyakarta: Andi.
- [4] Simarmata, Janner. (2004). *Rekayasa Perangkat Lunak*. Yogyakarta: Andi.
- [5] Sudawarman dan Dony Ariyus. (2007). *Interaksi Manusia & Komputer*. Yogyakarta: Andi.

