

BAB VI PENGUJIAN

Pada bab ini akan dijelaskan tentang pengujian terhadap aplikasi yang sebelumnya sudah diimplementasikan pada bab sebelumnya. Pengujian terdiri dari pengujian fungsional, yaitu menguji seluruh *use case* menggunakan teknik pengujian *Black-Box (Black-Box Testing)*, menguji beberapa *use case* yang berhubungan dengan perubahan data menggunakan teknik pengujian *White-Box (White-Box testing)* apakah sudah sesuai dengan skenario pengujian, dan pengujian non fungsional, yaitu memperhatikan tingkat *usability* menggunakan evaluasi heuristik.

Beberapa tahapan yang dilalui dengan mempertimbangkan metode yang telah dijelaskan pada bab metodologi penelitian, yaitu adanya uji coba yang akan melibatkan pakar yang berhubungan dengan disiplin ilmu yang diterapkan dalam aplikasi ini, kemudian dilanjutkan dengan uji coba lapangan yang akan melibatkan pihak SMAN 2 Malang sebagai obyek utama dalam aplikasi ini.

6.1 Pengujian Fungsional

6.1.1 Pengujian Validasi

Pada tahap ini, penulis melakukan pengujian validasi menggunakan teknik pengujian *Black-Box (Black-Box Testing)* untuk menguji seluruh *use case* apakah sudah sesuai dengan skenario pengujian yang sudah dibuat sebelumnya. Dalam hal ini akan ditampilkan tabel pengujian fungsional berisikan 11 *use case* yang telah dijabarkan sebelumnya dalam bab analisis dan perancangan.

6.1.1.1 Kasus Uji Validasi

a. Kasus Uji *Login*

Tabel 6.1 Kasus Uji untuk Pengujian Validasi *Login*

Nama Kasus Uji	Kasus Uji <i>Login</i> Sah
----------------	----------------------------

Objek Uji	Kebutuhan Fungsional (SRS_001)
Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa aplikasi dapat memenuhi kebutuhan fungsional untuk melakukan <i>login</i> bagi admin.
Prosedur Uji	1. Admin memasukkan <i>password</i> pada <i>field login</i> . 2. Admin menekan tombol OK.
Hasil yang Diharapkan	Sistem dapat melakukan penyeleksian kondisi login dan jika penyeleksian kondisi ini benar, maka halaman <i>Home</i> admin akan berhasil terbuka.

b. Kasus Uji Lihat Profil SMAN 2 Malang

Tabel 6.2 Kasus Uji untuk Pengujian Validasi Lihat Profil SMAN 2 Malang

Nama Kasus Uji	Kasus Uji Lihat Profil SMAN 2 Malang Sah
Objek Uji	Kebutuhan Fungsional (SRS_002)
Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa aplikasi dapat memenuhi kebutuhan fungsional untuk melihat Profil SMAN 2 Malang dilengkapi tombol tambah, edit, dan hapus bagi admin.
Prosedur Uji	1. Memilih menu yang ada di halaman <i>Home</i> (Profil Sekolah/Guru/ Kegiatan/Album/Denah)
Hasil yang Diharapkan	Sistem dapat menampilkan halaman yang sesuai dengan yang dipilih oleh Admin.

c. Kasus Uji Input Data

Tabel 6.3 Kasus Uji untuk Pengujian Validasi Input Data

Nama Kasus Uji	Kasus Uji <i>Input</i> Data Sah
Objek Uji	Kebutuhan Fungsional (SRS_003)
Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa aplikasi dapat memenuhi kebutuhan fungsional

	untuk memasukkan data baru bagi admin.
Prosedur Uji	<ol style="list-style-type: none"> 1. Memilih data menu yang akan ditambah datanya. 2. Menambah data dengan menekan tombol 'tambah'. 3. Memasukkan data baru sesuai form yang disediakan dan diakhiri dengan menekan tombol 'save'.
Hasil yang Diharapkan	Sistem telah berhasil menampilkan data beserta data yang sudah ditambahkan oleh admin.

d. Kasus Uji Ubah Data

Tabel 6.4 Kasus Uji untuk Pengujian Validasi Ubah Data

Nama Kasus Uji	Kasus Uji Ubah Data Sah
Objek Uji	Kebutuhan Fungsional (SRS_004)
Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa aplikasi dapat memenuhi kebutuhan fungsional untuk merubah data bagi admin.
Prosedur Uji	<ol style="list-style-type: none"> 1. Memilih menu ubah data dengan menekan tombol ubah. 2. Merubah data baik gambar atau teks sesuai form yang ditampilkan oleh sistem diakhiri dengan menekan tombol 'simpan'. 3. Menekan tombol 'lihat perubahan'.
Hasil yang Diharapkan	Sistem telah dapat menampilkan data profil dengan data yang sudah diubah oleh Admin.

e. **Kasus Uji Hapus Data**

Tabel 6.5 Kasus Uji untuk Pengujian Validasi Hapus Data

Nama Kasus Uji	Kasus Uji Hapus Data Sah
Objek Uji	Kebutuhan Fungsional (SRS_005)
Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa aplikasi dapat memenuhi kebutuhan fungsional untuk menghapus data yang sudah ada bagi admin.
Prosedur Uji	1. Memilih data yang akan dihapus. 2. Menghapus data dengan menekan tombol 'hapus'.
Hasil yang Diharapkan	Sistem dapat menampilkan data yang akan dihapus dan setelah dilakukan penghapusan, sistem berhasil menampilkan data profil setelah penghapusan data oleh admin.

f. **Kasus Uji Logout**

Tabel 6.6 Kasus Uji untuk Pengujian Validasi Logout

Nama Kasus Uji	Kasus Uji Logout Sah
Objek Uji	Kebutuhan Fungsional (SRS_006)
Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa aplikasi dapat memenuhi kebutuhan fungsional untuk keluar dari halaman <i>home</i> admin bagi admin.
Prosedur Uji	1. Menekan tombol logout pada halaman <i>home</i> . 2. Menekan tombol 'ya'
Hasil yang Diharapkan	Sistem dapat menampilkan halaman <i>home user</i> .

g. **Kasus Uji Lihat Profil Sekolah**

Tabel 6.7 Kasus Uji untuk Pengujian Validasi Lihat Profil Sekolah

Nama Kasus Uji	Kasus Uji Lihat Profil Sekolah Sah
Objek Uji	Kebutuhan Fungsional (SRS_007)

Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa aplikasi dapat memenuhi kebutuhan fungsional untuk melihat halaman-halaman dalam menu Profil Sekolah bagi pengguna.
Prosedur Uji	<ol style="list-style-type: none"> 1. Menekan tombol 'Profil Sekolah'. 2. Mengecek halaman-halaman dalam menu Profil Sekolah.
Hasil yang Diharapkan	Sistem dapat menampilkan halaman yang sesuai dengan yang dipilih oleh pengguna dalam menu Profil Sekolah.

h. Kasus Uji Lihat Data Guru

Tabel 6.8 Kasus Uji untuk Pengujian Validasi Lihat Data Guru

Nama Kasus Uji	Kasus Uji Lihat Data Guru Sah
Objek Uji	Kebutuhan Fungsional (SRS_008)
Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa aplikasi dapat memenuhi kebutuhan fungsional untuk melihat halaman-halaman dalam menu Guru bagi pengguna.
Prosedur Uji	<ol style="list-style-type: none"> 1. Menekan tombol 'Guru'. 2. Mengecek halaman-halaman dalam menu Guru.
Hasil yang Diharapkan	Sistem dapat menampilkan halaman yang sesuai dengan yang dipilih oleh pengguna dalam menu Guru.

i. Kasus Uji Lihat Data Kegiatan

Tabel 6.9 Kasus Uji untuk Pengujian Validasi Lihat Data Kegiatan

Nama Kasus Uji	Kasus Uji Lihat Data Kegiatan Sah
Objek Uji	Kebutuhan Fungsional (SRS_009)

Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa aplikasi dapat memenuhi kebutuhan fungsional untuk melihat halaman-halaman dalam menu Kegiatan bagi pengguna.
Prosedur Uji	1. Menekan tombol 'Kegiatan'. 2. Mengecek halaman-halaman dalam menu Kegiatan.
Hasil yang Diharapkan	Sistem dapat menampilkan halaman yang sesuai dengan yang dipilih oleh pengguna dalam menu Kegiatan.

j. Kasus Uji Lihat Data Album

Tabel 6.10 Kasus Uji untuk Pengujian Validasi Lihat Data Album

Nama Kasus Uji	Kasus Uji Lihat Data Album Sah
Objek Uji	Kebutuhan Fungsional (SRS_010)
Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa aplikasi dapat memenuhi kebutuhan fungsional untuk melihat halaman-halaman dalam menu Album bagi pengguna.
Prosedur Uji	1. Menekan tombol 'Album'. 2. Mengecek halaman-halaman dalam menu Album.
Hasil yang Diharapkan	Sistem dapat menampilkan halaman yang sesuai dengan yang dipilih oleh pengguna dalam menu Album.

k. Kasus Uji Lihat Data Denah

Tabel 6.11 Kasus Uji untuk Pengujian Validasi Lihat Denah

Nama Kasus Uji	Kasus Uji Lihat Denah Sah
Objek Uji	Kebutuhan Fungsional (SRS_011)

Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa aplikasi dapat memenuhi kebutuhan fungsional untuk melihat halaman-halaman dalam menu Denah bagi pengguna.
Prosedur Uji	1. Menekan tombol 'Denah'. 2. Mengecek halaman-halaman dalam menu Denah.
Hasil yang Diharapkan	Sistem dapat menampilkan halaman yang sesuai dengan yang dipilih oleh pengguna dalam menu Denah.

6.1.1.2 Hasil Pengujian Validasi

Tabel 6.12 Hasil Pengujian Validasi

No.SRS	Use Case	Hasil yang Diharapkan	Hasil yang Didapatkan	Status Validasi
Admin				
SRS_001	Login	Sistem berjalan sesuai dengan skenario, <i>password</i> yang dimasukkan sesuai dan kemudian sistem menampilkan halaman utama.	Sistem berjalan sesuai dengan skenario, <i>password</i> yang dimasukkan sesuai dan kemudian sistem menampilkan halaman utama.	(Valid)
SRS_002	Lihat Profil SMAN 2 Malang	Sistem dapat menampilkan halaman terpilih dilengkapi tombol pembaruan data pada menu profil sekolah, guru, kegiatan, album, dan denah sekolah.	Sistem dapat menampilkan halaman terpilih dilengkapi tombol pembaruan data pada menu profil sekolah, guru, kegiatan, album, dan denah sekolah.	(Valid)
SRS_003	Input Data	Sistem dapat menampilkan data beserta data yang sudah ditambahkan oleh admin.	Sistem dapat menampilkan data beserta data yang sudah ditambahkan oleh admin.	(Valid)
SRS_004	Ubah Data	Sistem dapat menampilkan data dengan data yang sudah diubah.	Sistem dapat menampilkan data dengan data yang sudah diubah.	(Valid)
SRS_005	Hapus Data	Sistem dapat menampilkan data yang akan dihapus. Dan setelah dilakukan penghapusan, sistem dapat	Sistem dapat menampilkan data yang akan dihapus. Dan setelah dilakukan	

		menampilkan data profil setelah penghapusan data.	penghapusan, sistem dapat menampilkan data profil setelah penghapusan data.	(Valid)
SRS_006	Logout	Sistem menampilkan peringatan konfirmasi untuk keluar dari halaman admin. Sistem dapat mengembalikan posisi pengguna ke halaman <i>home user</i> .	Sistem menampilkan peringatan konfirmasi untuk keluar dari halaman admin. Sistem dapat mengembalikan posisi pengguna ke halaman <i>home user</i> .	(Valid)
Pengguna				
SRS_007	Lihat Profil Sekolah	Sistem berhasil menampilkan halaman Profil Sekolah.	Sistem berhasil menampilkan halaman Profil Sekolah.	(Valid)
SRS_008	Lihat Data Guru	Sistem berhasil menampilkan halaman Menu Guru sesuai mata pelajaran.	Sistem berhasil menampilkan halaman Menu Guru sesuai mata pelajaran.	(Valid)
SRS_009	Lihat Data Kegiatan	Sistem dapat menampilkan halaman Menu Kegiatan .	Sistem dapat menampilkan halaman Menu Kegiatan .	(Valid)
SRS_010	Lihat Data Album	Sistem dapat menampilkan halaman Menu Album.	Sistem dapat menampilkan halaman Menu Album.	(Valid)
SRS_011	Lihat Denah	Sistem dapat menampilkan halaman Denah 2Dimensi setiap ruang dan denah 3Dimensi.	Sistem dapat menampilkan halaman Denah 2Dimensi setiap ruang dan denah 3Dimensi.	(Valid)

6.1.2 Pengujian Unit

Pada pengujian unit aplikasi Profil SMAN 2 Malang digunakan teknik *White-Box Testing* dengan teknik *Basis Path Testing*. Pada teknik *Basis Path Testing*, proses pengujian dilakukan dengan memodelkan algoritma pada suatu *flow graph*, menentukan jumlah kompleksitas siklomatis (*cyclomatic complexity*), menentukan sebuah basis set dari jalur independen dan memberikan kasus uji (*test case*) pada setiap basis set yang telah ditentukan.

6.1.2.1 Pengujian Unit untuk Algoritma Login

Proses login berfungsi untuk melakukan seleksi kondisi pada waktu *user* melakukan proses *login* dengan menekan tombol *Login* yang ada pada Halaman *Home*. Implementasi algoritma untuk melakukan seleksi kondisi *login* dijelaskan pada Gambar 6.1.

Nama Algoritma : login

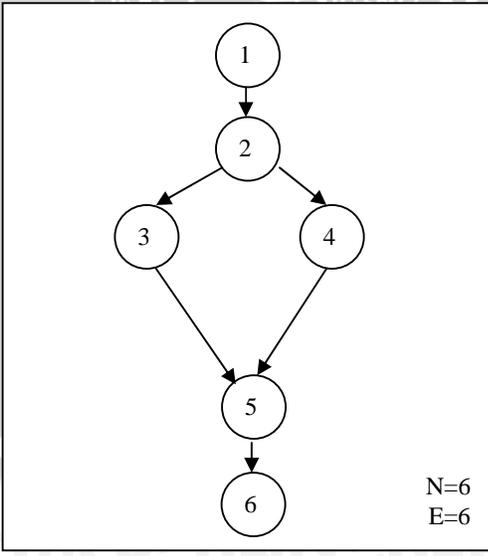
Deklarasi

- String → password

Deskripsi

- Masukan : password
- Proses
 1. Menekan tombol login pada halaman Home } 1
 2. Memasukkan password pada field 'pass' } 2
 3. Jika salah, maka tampilkan pesan "Password Salah!", set field = 0 } 3
 4. Jika benar, set field = 1 go adminhome of movie Home } 4
 5. Menutup halaman home user } 5
- Keluaran : halaman home admin } 6

Gambar 6.1 Implementasi Algoritma *Login*



Gambar 6.2 Flow Graph Algoritma *Login*

Pemodelan ke dalam *flow graph* yang telah dilakukan terhadap algoritma *login* menghasilkan jumlah kompleksitas siklomatis (*cyclomatic complexity*) melalui persamaan $V(G) = E - N + 2$, dimana $V(G)$ merupakan jumlah kompleksitas siklomatis, E merupakan sisi atau *edge* (garis penghubung antar *node*) dan N merupakan jumlah simpul (*node*).

$$\begin{aligned} V(G) &= E - N + 2 \\ &= 6 - 6 + 2 \\ &= 2 \end{aligned}$$

Berdasarkan dari nilai *cyclomatic complexity* yang telah didapatkan dari perhitungan maka ditentukan 2 buah basis set dari jalur *independent*, yaitu :

Jalur 1 : 1 – 2 – 3 – 5 – 6

Jalur 2 : 1 – 2 – 4 – 5 – 6

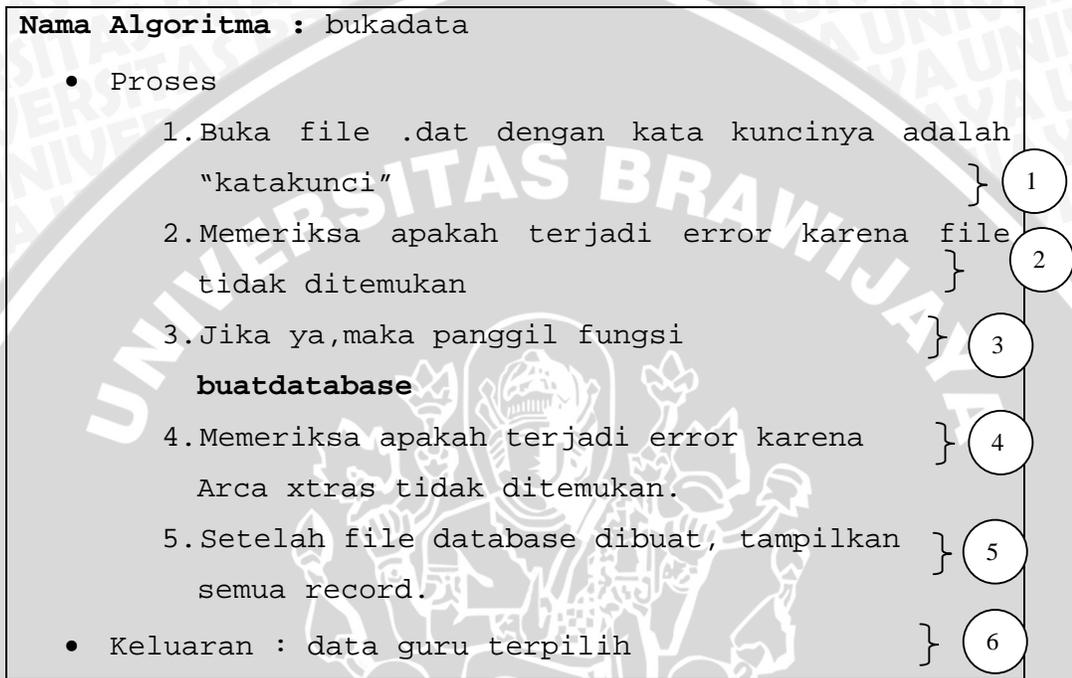
Penentuan kasus uji untuk jalur *independent* tersebut dan hasil eksekusinya dijelaskan pada Tabel 6.13.

Tabel 6.13 Kasus Uji Untuk Pengujian Unit Algoritma *Login*

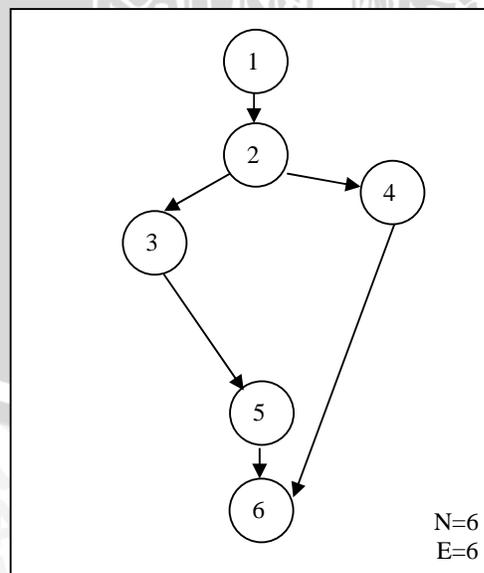
Jalur	Kasus Uji	Hasil yang diharapkan	Hasil yang didapatkan
1	Memberi data masukkan dengan parameter <i>password</i> dengan null (kosong) atau yang tidak ada pada <i>Cast Member</i> .	Proses akan terhenti dan akan ditampilkan pesan salah	Proses akan terhenti dan akan ditampilkan pesan salah
2	Memberi data masukkan dengan parameter <i>password</i> yang sudah diset pada <i>Cast Member</i> .	Ditampilkan halaman <i>Home Admin</i> pada layar	Ditampilkan halaman <i>Home Admin</i> pada layar

6.1.2.2 Pengujian Unit untuk Algoritma BukaData

Proses BukaData berfungsi untuk membuka data yang sudah atau akan dimasukkan ke dalam *database* yang akan dibuat. Implementasi algoritma untuk melakukan BukaData dijelaskan pada Gambar 6.3.



Gambar 6.3 Implementasi Algoritma BukaData



Gambar 6.4 Flow Graph Algoritma BukaData

Pemodelan ke dalam *flow graph* yang telah dilakukan terhadap algoritma *login* menghasilkan jumlah kompleksitas siklomatis (*cyclomatic complexity*) melalui persamaan $V(G) = E - N + 2$, dimana $V(G)$ merupakan jumlah kompleksitas siklomatis, E merupakan sisi atau *edge* (garis penghubung antar *node*) dan N merupakan jumlah simpul (*node*).

$$\begin{aligned} V(G) &= E - N + 2 \\ &= 6 - 6 + 2 \\ &= 2 \end{aligned}$$

Berdasarkan dari nilai *cyclomatic complexity* yang telah didapatkan dari perhitungan maka ditentukan 2 buah basis set dari jalur *independent*, yaitu :

Jalur 1 : 1 – 2 – 4 – 6

Jalur 2 : 1 – 2 – 3 – 5 – 6

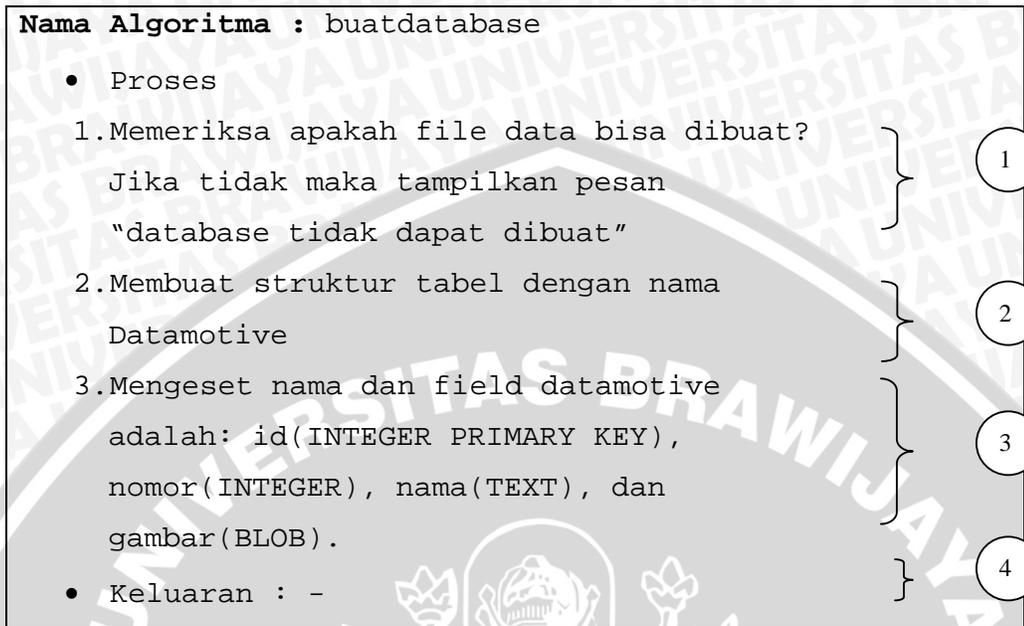
Penentuan kasus uji untuk jalur *independent* tersebut dan hasil eksekusinya dijelaskan pada Tabel 6.14.

Tabel 6.14 Kasus Uji Untuk Pengujian Unit Algoritma BukaData

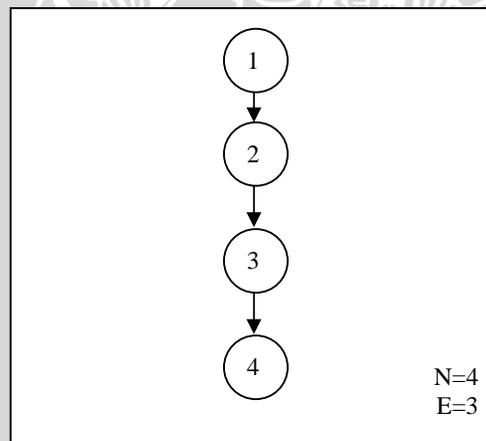
Jalur	Kasus Uji	Hasil yang diharapkan	Hasil yang didapatkan
1	Membuka file *dir.	File yang sudah dimasukkan secara otomatis muncul di layar	File yang sudah dimasukkan secara otomatis muncul di layar
2	Menghapus file .dat yang telah ada untuk kemudian diuji dalam pembuatan <i>database</i>	File null dan file .dat baru telah dibuat	File null dan file .dat baru telah dibuat

6.1.2.3 Pengujian Unit untuk Algoritma BuatDatabase

Proses BuatDatabase berfungsi untuk membuat sebuah *file* berekstensi .dat yang berfungsi untuk menyimpan data masukan dari *stage* dan *xtras*. Implementasi algoritma untuk melakukan seleksi kondisi BuatDatabase dijelaskan pada Gambar 6.5.



Gambar 6.5 Implementasi Algoritma BuatDatabase



Gambar 6.6 Flow Graph Algoritma BuatDatabase

Pemodelan ke dalam *flow graph* yang telah dilakukan terhadap algoritma *login* menghasilkan jumlah kompleksitas siklomatis (*cyclomatic complexity*) melalui persamaan $V(G) = E - N + 2$, dimana $V(G)$ merupakan jumlah kompleksitas siklomatis, E merupakan sisi atau *edge* (garis penghubung antar *node*) dan N merupakan jumlah simpul (*node*).

$$V(G) = E - N + 2$$

$$= 3 - 4 + 2$$

$$= 1$$

Berdasarkan dari nilai *cyclomatic complexity* yang telah didapatkan dari perhitungan maka ditentukan 1 buah basis set dari jalur *independent*, yaitu :

Jalur 1 : 1 - 2 - 3 - 4

Penentuan kasus uji untuk jalur *independent* tersebut dan hasil eksekusinya dijelaskan pada Tabel 6.15.

Tabel 6.15 Kasus Uji Untuk Pengujian Unit Algoritma BuatDatabase

Jalur	Kasus Uji	Hasil yang diharapkan	Hasil yang didapatkan
1	Membuka file *.dir	File .dat tercipta ketika file .dir dibuka.	File .dat tercipta ketika file .dir dibuka.

6.1.2.4 Pengujian Unit untuk Algoritma TambahRecordBaru

Proses TambahRecordBaru berfungsi untuk melakukan penambahan data baru ke dalam *file* .dat yang telah dibuat sebelumnya. Implementasi algoritma untuk melakukan TambahRecordBaru dijelaskan pada Gambar 6.7.

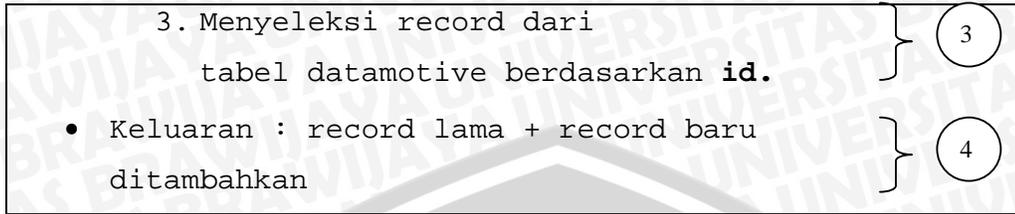
Nama Algoritma : tambahrecordbaru

Deklarasi

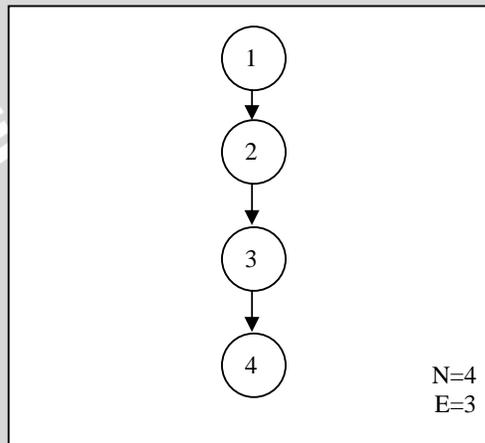
- String → id,nomor,nama,gambar

Deskripsi

- Masukan : id,nomor,nama,gambar
- Proses
 1. Menambahkan ruang record baru ke tabel **datamotive** dengan nilai masing-masing field adalah kosong (NULL). } (1)
 2. Memberi nilai pada variabel **id** dengan nomor posisi record terakhir } (2)



Gambar 6.7 Implementasi Algoritma TambahRecordBaru



Gambar 6.8 Flow Graph Algoritma TambahRecordBaru

Pemodelan ke dalam *flow graph* yang telah dilakukan terhadap algoritma *login* menghasilkan jumlah kompleksitas siklomatis (*cyclomatic complexity*) melalui persamaan $V(G) = E - N + 2$, dimana $V(G)$ merupakan jumlah kompleksitas siklomatis, E merupakan sisi atau *edge* (garis penghubung antar *node*) dan N merupakan jumlah simpul (*node*).

$$\begin{aligned}
 V(G) &= E - N + 2 \\
 &= 4 - 3 + 2 \\
 &= 1
 \end{aligned}$$

Berdasarkan dari nilai *cyclomatic complexity* yang telah didapatkan dari perhitungan maka ditentukan 1 buah basis set dari jalur *independent*, yaitu :

$$\text{Jalur 1 : } 1 - 2 - 3 - 4$$

Penentuan kasus uji untuk jalur *independent* tersebut dan hasil eksekusinya dijelaskan pada Tabel 6.16.

Tabel 6.16 Kasus Uji Untuk Pengujian Unit Algoritma TambahRecordBaru

Jalur	Kasus Uji	Hasil yang diharapkan	Hasil yang didapatkan
1	Memasukkan data baru melalui file *.dir dengan menekan tombol tambah dan mengisi <i>field</i> yang disediakan	<i>Record</i> baru sudah ditambahkan dalam record yang sudah ada sebelumnya	<i>Record</i> baru sudah ditambahkan dalam record yang sudah ada sebelumnya

6.1.2.5 Pengujian Unit untuk Algoritma SimpanPerubahanRecord

Proses SimpanPerubahanBaru berfungsi untuk melakukan penyimpanan data baru ke dalam *file* .dat yang telah dimasukkan dalam proses TambahRecordBaru. Implementasi algoritma untuk melakukan SimpanPerubahanRecord dijelaskan pada Gambar 6.9.

Nama Algoritma : simpanperubahanrecord

Deklarasi

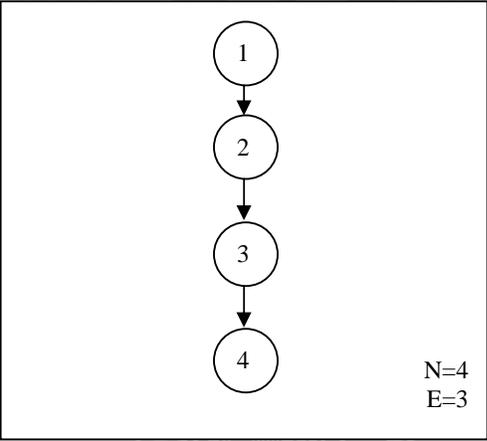
- String → id,nomor,nama,gambar

Deskripsi

- Masukan : id,nomor,nama,gambar
- Proses
 - Memberi nilai pada variabel **id** dengan posisi record yang sedang diseleksi. } (1)
 - Mengisi label datamotive pada } (2)
 - masing-masing field dengan data yang terdapat pada obyek Filled dan Image. } (3)
 - Menyeleksi semua record } (4)
- Keluaran : record lama + record baru } (4)

ditambahkan

Gambar 6.9 Implementasi Algoritma SimpanPerubahanRecord



Gambar 6.10 Flow Graph Algoritma SimpanPerubahanRecord

Pemodelan ke dalam *flow graph* yang telah dilakukan terhadap algoritma *login* menghasilkan jumlah kompleksitas siklomatis (*cyclomatic complexity*) melalui persamaan $V(G) = E - N + 2$, dimana $V(G)$ merupakan jumlah kompleksitas siklomatis, E merupakan sisi atau *edge* (garis penghubung antar *node*) dan N merupakan jumlah simpul (*node*).

$$\begin{aligned} V(G) &= E - N + 2 \\ &= 4 - 3 + 2 \\ &= 1 \end{aligned}$$

Berdasarkan dari nilai *cyclomatic complexity* yang telah didapatkan dari perhitungan maka ditentukan 1 buah basis set dari jalur *independent*, yaitu :

Jalur 1 : 1 – 2 – 3 – 4

Penentuan kasus uji untuk jalur *independent* tersebut dan hasil eksekusinya dijelaskan pada Tabel 6.17.

Tabel 6.17 Kasus Uji Untuk Pengujian Unit Algoritma SimpanPerubahanRecord

Jalur	Kasus Uji	Hasil yang diharapkan	Hasil yang didapatkan
1	Menekan tombol simpan setelah field tambah, edit data melalui file *.dir dirubah	Record baru, atau record lama yang dirubah sudah ditambahkan/ dirubah dalam record yang ada	Record baru, atau record lama yang dirubah sudah ditambahkan/ dirubah dalam record yang ada

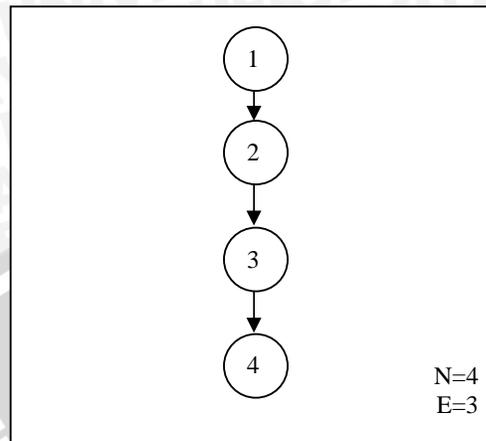
6.1.2.6 Pengujian Unit untuk Algoritma HapusRecord

Proses HapusRecord berfungsi untuk melakukan penghapusan data baru di dalam *file* .dat yang telah disimpan sebelumnya. Implementasi algoritma untuk melakukan HapusRecord dijelaskan pada Gambar 6.11.

Nama Algoritma : hapusrecord

- Proses
 1. Memberi nilai pada variabel **id** dengan posisi record yang sedang diseleksi. } 1
 2. Menghapus record dari tabel datamotive pada posisi record yang sesuai dengan nilai **id**. } 2
 3. Tampilkan semua record. } 3
- Keluaran : record setelah dilakukan penghapusan } 4

Gambar 6.11 Implementasi Algoritma HapusRecord



Gambar 6.12 Flow Graph Algoritma HapusRecord

Pemodelan ke dalam *flow graph* yang telah dilakukan terhadap algoritma *login* menghasilkan jumlah kompleksitas siklomatis (*cyclomatic complexity*) melalui persamaan $V(G) = E - N + 2$, dimana $V(G)$ merupakan jumlah kompleksitas siklomatis, E merupakan sisi atau *edge* (garis penghubung antar *node*) dan N merupakan jumlah simpul (*node*).

$$\begin{aligned} V(G) &= E - N + 2 \\ &= 4 - 3 + 2 \\ &= 1 \end{aligned}$$

Berdasarkan dari nilai *cyclomatic complexity* yang telah didapatkan dari perhitungan maka ditentukan 1 buah basis set dari jalur *independent*, yaitu :

$$\text{Jalur 1 : } 1 - 2 - 3 - 4$$

Penentuan kasus uji untuk jalur *independent* tersebut dan hasil eksekusinya dijelaskan pada Tabel 6.18.

Tabel 6.18 Kasus Uji Untuk Pengujian Unit Algoritma HapusRecord

Jalur	Kasus Uji	Hasil yang diharapkan	Hasil yang didapatkan
1	Menekan tombol hapus pada salah satu atau banyak data melalui file *.dir	Ditampilkan data setelah dilakukan penghapusan	Ditampilkan data setelah dilakukan penghapusan

6.2 Pengujian Non Fungsional

Pada tahap ini, penulis menguji apakah aplikasi sudah sesuai dengan prinsip-prinsip *usability* dengan menggunakan evaluasi heuristik. Evaluasi heuristik terdiri dari 10 tahap yang semuanya ditampilkan dalam kuisioner berupa langkah-langkah penerapan secara langsung dalam aplikasi.

6.2.1 Instrumen dan Pengumpulan Data

Aplikasi Profil SMAN 2 Malang akan diterapkan kepada subyek penelitian sebagai sumber data selama tahap uji coba pakar, dan uji coba lapangan. Pengumpulan data dilakukan dengan kuisioner dengan memakai evaluasi heuristik dalam pemberian tugas pada pengguna.

6.2.2 Uji Coba Pakar

Pada tahap ini, uji coba dilakukan oleh dosen dari Teknik Informatika Universitas Brawijaya, Manajemen Informatika Politeknik Negeri Malang, dan Desain Komunikasi Visual Universitas Negeri Malang berjumlah 30 orang yang berpengalaman dalam bidang desain aplikasi.

Uji coba ini akan dilakukan dua kali, pengujian pertama dilaksanakan setelah tahap pembuatan produk awal dan pakar akan memberikan penilaian sekaligus saran bagi perbaikan aplikasi. Sedangkan pengujian kedua dilaksanakan setelah tahap perbaikan produk, pengujian kedua ini bertujuan untuk memverifikasi apakah perbaikan yang dilakukan oleh penulis terhadap aplikasi

telah sesuai dengan masukan-masukan yang telah diberikan pada pengujian pertama.

Tabel 6.19 Hasil Kuisioner Pakar

No	Kategori	Skala0 (sangat baik)	Skala1 (baik)	Skala2 (cukup)	Skala3 (kurang)	Skala4 (sangat kurang)
1	Sistem harus menginformasikan dengan jelas kepada pengguna tentang apa yang sedang terjadi pada saat menggunakan sistem. Langkah: 1. Login (Halaman utama – klik admin – masukkan password ‘sma2’ – login), kemudian lihat apakah halaman yang dibuka setelah login dapat dikenali sebagai halaman admin? Hasil: Halaman admin, ditandai dengan kalimat ‘selamat datang,admin!’, dan ada tombol ‘ubah data’. 2. Logout (Halaman utama – klik ‘logout’ – klik ‘ya’), sekarang anda berada di halaman user. apakah dapat dibedakan halaman user dan halaman admin? Hasil: Halaman user tanpa ada kalimat ‘selamat datang,admin!’ dan tombol ‘ubah data’ tidak ada.	Pengujian 1				
		16,7% (5 Orang)	73,3% (22 Orang)	10% (3 Orang)	0%	0%
		Pengujian 2				
		66,7% (20 Orang)	33,3% (10 Orang)	0%	0%	0%
2	Bahasa ataupun simbol yang digunakan dalam sistem tersebut sebaiknya sinkron atau tidak jauh berbeda dengan kehidupan nyata. Langkah: 1. Di halaman utama, perhatikan pemilihan gambar tombol, apakah sudah sesuai dengan visualisasi di kehidupan nyata? 2. Perhatikan bahasa yang digunakan, apakah sudah sesuai dengan bahasa yang digunakan di kehidupan nyata?	Pengujian 1				
		30% (9 Orang)	63,3% (19 Orang)	6,7% (2 Orang)	0%	0%
		Pengujian 2				
		86,7% (26 Orang)	13,3% (4 Orang)	0%	0%	0%
3	Pengguna sering kali memilih fungsi yang salah, dan memerlukan sebuah “pintu darurat” untuk mengeluarkan pengguna dari situasi tersebut. Langkah: 1. Perhatikan setiap halaman, apakah sudah ada tombol ‘Kembali’ untuk mengembalikan pengguna ke halaman sebelumnya? 2. Lakukan klik pada sembarang tombol, dan perhatikan apakah ada tombol ‘Home’ untuk mengembalikan pengguna ke halaman utama?	Pengujian 1				
		70% (21 Orang)	20% (6 Orang)	10% (3 Orang)	0%	0%
		Pengujian 2				
		86,7% (26 Orang)	13,3% (4 Orang)	0%	0%	0%
4	Sistem dirancang dengan konsisten sehingga tidak membingungkan user.	Pengujian 1				
		46,7% (14 Orang)	43,3% (13 Orang)	10% (3 Orang)	0%	0%

	Langkah: 1. Perhatikan peletakan tombol 'Back' dan 'Home', apakah letaknya konsisten di setiap halaman? 2. Perhatikan Judul Halaman, apakah letaknya sudah konsisten di setiap halaman?									
		Pengujian 2								
		70% (21 Orang)	30% (9 Orang)	0%	0%	0%				
5	Sebuah konfirmasi yang mencegah kesalahan pengguna. Langkah: 1. Lakukan login, kemudian klik tombol logout, apakah ada konfirmasi sebelum sistem melakukan logout dari halaman admin? 2. Lakukan Login – klik menu 'guru' – klik 'MIPA' – klik 'hapus'. Apakah ada konfirmasi sebelum data benar-benar dihapus?	Pengujian 1								
		30% (9 Orang)	66,7% (20 Orang)	3,3% (1 Orang)	0%	0%				
		Pengujian 2								
		53,3% (16 Orang)	46,7% (14 Orang)	0%	0%	0%				
6	Meminimalkan ingatan pengguna dengan membuat objek, aksi, dan fungsi-fungsi mempunyai visualisasi sendiri-sendiri. Langkah: 1. Perhatikan setiap tombol, apakah memiliki fungsi dan visualisasi sendiri-sendiri? 2. Perhatikan tombol 'Back', 'Home', dan 'Map'. Apakah visualisasinya mudah dikenali?	Pengujian 1								
		50% (15 Orang)	40% (12 Orang)	10% (3 Orang)	0%	0%				
		Pengujian 2								
		80% (24 Orang)	20% (6 Orang)	0%	0%	0%				
7	Kemudahan pengguna dalam mencari informasi yang diinginkan. Langkah: 1. Perhatikan pemberian nama dari menu/tombol. Apakah memudahkan pengguna mencari informasi yang diinginkan? 2. Tekan tombol 'struktur' pada halaman utama, apakah fasilitas tersebut memudahkan pengguna mencari informasi yang diinginkan?	Pengujian 1								
		40% (12 Orang)	53,3% (16 Orang)	6,7% (2 Orang)	0%	0%				
		Pengujian 2								
		73,3% (22 Orang)	26,7% (8 Orang)	0%	0%	0%				
8	Tampilan informasi dan elemen desain tidak boleh berlebihan, karena keduanya akan bersaing secara visual dengan informasi yang lebih relevan di layar. Langkah: 1. Perhatikan beberapa halaman dalam menu Profil Sekolah, apakah konten dapat terbaca dengan jelas? 2. Perhatikan halaman guru (pilih salah satu tombol dalam menu guru), apakah konten terbaca dengan jelas?	Pengujian 1								
		20% (6 Orang)	60% (18 Orang)	20% (6 Orang)	0%	0%				
		Pengujian 2								
		56,7% (17 Orang)	43,3% (13 Orang)	0%	0%	0%				
9	Ketika pengguna melakukan kesalahan, beri pesan yang jelas dan mudah dimengerti menggunakan bahasa yang sederhana. Langkah:	Pengujian 1								
		36,7% (11 Orang)	40% (12 Orang)	23,3% (7 Orang)	0%	0%				

	1. Klik menu ‘denah sekolah’ – klik tombol ‘2D’ – klik gambar ‘KOPSIS’ – klik ‘Play Video’, karena video belum tersedia, maka ada peringatan dari sistem. Apakah mudah dimengerti? 2. Ketika membuka aplikasi untuk pertama kali, ada peringatan untuk registrasi xtras, apakah pengguna dapat keluar dari kondisi tersebut dengan mudah?	Pengujian 2				
		50% (15 Orang)	46,7% (14 Orang)	0%	3,3% (1 Orang)	0%
10	Adanya bantuan dan dokumentasi pada sistem akan sangat membantu pengguna. Langkah: 1. Klik tombol ‘petunjuk’ pada halaman utama, apakah ulasan yang diberikan cukup membantu dalam penggunaan sistem? 2. Apakah cetakan dokumentasi membantu pengguna dalam mempelajari sistem?	Pengujian 1				
		46,7% (14 Orang)	43,3% (13 Orang)	10% (3 Orang)	0%	0%
		Pengujian 2				
		76,7% (23 Orang)	23,3% (7 Orang)	0%	0%	0%
	Rata-rata (Pengujian 1)	38,68%	50,32%	11%	0%	0%
	Rata-rata (Pengujian 2)	70,01%	29,66%	0%	0,33%	0%

Data diatas menyebutkan dari 10 langkah dalam evaluasi heuristik yang ditujukan untuk 30 orang pakar rata-rata persentase yang didapatkan adalah 38,68% untuk Skala 0 (sangat baik), 50,32% untuk skala 1 (baik), 11% untuk skala 2 (cukup), dan tidak ada yang memberikan penilaian atau persentase 0% untuk skala 3 dan 4 (kurang, dan kurang sekali).

Kemudian setelah dilakukan perbaikan, maka dilakukan pengujian kedua kepada 30 pakar yang menghasilkan data berikut: 70,01% untuk Skala 0 (sangat baik), 29,66% untuk skala 1 (baik), 0,33% untuk skala 3 (kurang), dan tidak ada yang memberikan penilaian atau persentase 0% untuk skala 2 dan 4 (cukup dan kurang sekali). Data yang dihasilkan mengalami peningkatan signifikan sehingga aplikasi lebih baik dari sebelumnya menurut pakar.

6.2.3 Uji Coba Lapangan

Langkah selanjutnya adalah uji coba lapangan untuk mendapatkan evaluasi atas produk dari pihak sekolah. Pengujian dilakukan di SMAN 2 Malang dengan cara uji coba langsung di laboratorium komputer yang dilakukan oleh: Kepala Sekolah, 3 orang guru, 15 siswa kelas X, 15 siswa kelas XI, 15 siswa kelas XII, dan 1 admin.

Uji coba ini akan dilakukan dua kali, pengujian pertama dilaksanakan setelah tahap pembuatan produk awal dan pihak sekolah akan memberikan penilaian sekaligus saran bagi perbaikan aplikasi. Sedangkan pengujian kedua dilaksanakan setelah tahap perbaikan produk, pengujian kedua ini bertujuan untuk memverifikasi apakah perbaikan yang dilakukan oleh penulis terhadap aplikasi telah sesuai dengan masukan-masukan yang telah diberikan pada pengujian pertama. Dan hasil kuisioner akan dijelaskan dalam Tabel berikut:

Tabel 6.20 Hasil Kuisioner Pihak Sekolah

No	Kategori	Skala0 (sangat baik)	Skala1 (baik)	Skala2 (cukup)	Skala3 (kurang)	Skala4 (sangat kurang)
1	Sistem harus menginformasikan dengan jelas kepada pengguna tentang apa yang sedang terjadi pada saat menggunakan sistem. Langkah: 1. Login (Halaman utama – klik admin – masukkan password ‘sma2’ – login), kemudian lihat apakah halaman yang dibuka setelah login dapat dikenali sebagai halaman admin? Hasil: Halaman admin, ditandai dengan kalimat ‘selamat datang,admin!’, dan ada tombol ‘ubah data’. 2. Logout (Halaman utama – klik ‘logout’ – klik ‘ya’), sekarang anda berada di halaman user. apakah dapat dibedakan halaman user dan halaman admin? Hasil: Halaman user tanpa ada kalimat ‘selamat datang,admin!’ dan tombol ‘ubah data’ tidak ada.	Pengujian 1				
		42% (21 Orang)	46% (23 Orang)	12% (6 Orang)	0%	0%
		Pengujian 2				
		52% (26 Orang)	48% (24 Orang)	0%	0%	0%
2	Bahasa ataupun simbol yang digunakan dalam sistem tersebut sebaiknya sinkron atau tidak jauh berbeda dengan kehidupan nyata. Langkah: 1. Di halaman utama, perhatikan pemilihan gambar tombol, apakah sudah sesuai dengan visualisasi di	Pengujian 1				
		52% (26 Orang)	48% (24 Orang)	0%	0%	0%

	kehidupan nyata? 2. Perhatikan bahasa yang digunakan, apakah sudah sesuai dengan bahasa yang digunakan di kehidupan nyata?	Pengujian 2				
		56% (28 Orang)	44% (22 Orang)	0%	0%	0%
3	Pengguna sering kali memilih fungsi yang salah, dan memerlukan sebuah “pintu darurat” untuk mengeluarkan pengguna dari situasi tersebut. Langkah: 1. Perhatikan setiap halaman, apakah sudah ada tombol ‘Kembali’ untuk mengembalikan pengguna ke halaman sebelumnya? 2. Lakukan klik pada sembarang tombol, dan perhatikan apakah ada tombol ‘Home’ untuk mengembalikan pengguna ke halaman utama’?	Pengujian 1				
		44% (22 Orang)	50% (25 Orang)	6% (3 Orang)	0%	0%
		Pengujian 2				
		62% (31 Orang)	38% (19 Orang)	0%	0%	0%
4	Sistem dirancang dengan konsisten sehingga tidak membingungkan user. Langkah: 1. Perhatikan peletakan tombol ‘Back’ dan ‘Home’, apakah letaknya konsisten di setiap halaman? 2. Perhatikan Judul Halaman, apakah letaknya sudah konsisten di setiap halaman?	Pengujian 1				
		40% (20 Orang)	50% (25 Orang)	10% (5 Orang)	0%	0%
		Pengujian 2				
		64% (32 Orang)	36% (18 Orang)	0%	0%	0%
5	Sebuah konfirmasi yang mencegah kesalahan pengguna. Langkah: 1. Lakukan login, kemudian klik tombol logout, apakah ada konfirmasi sebelum sistem melakukan logout dari halaman admin? 2. Lakukan Login – klik menu ‘guru’ – klik ‘MIPA’ – klik ‘hapus’. Apakah ada konfirmasi sebelum data benar-benar dihapus?	Pengujian 1				
		48% (24 Orang)	46% (23 Orang)	6% (3 Orang)	0%	0%
		Pengujian 2				
		56% (28 Orang)	44% (22 Orang)	0%	0%	0%
6	Meminimalkan ingatan pengguna dengan membuat objek, aksi, dan fungsi-fungsi mempunyai visualisasi sendiri-sendiri. Langkah: 1. Perhatikan setiap tombol, apakah memiliki fungsi dan visualisasi sendiri-sendiri? 2. Perhatikan tombol ‘Back’, ‘Home’, dan ‘Map’. Apakah visualisasinya mudah dikenali?	Pengujian 1				
		14% (7 Orang)	66% (33 Orang)	20% (10 Orang)	0%	0%
		Pengujian 2				
		44% (22 Orang)	56% (28 Orang)	0%	0%	0%
7	Kemudahan pengguna dalam mencari informasi yang diinginkan. Langkah: 1. Perhatikan pemberian nama dari menu/tombol. Apakah memudahkan pengguna mencari informasi yang diinginkan? 2. Tekan tombol ‘struktur’ pada halaman utama, apakah fasilitas tersebut memudahkan pengguna	Pengujian 1				
		30% (15 Orang)	58% (29 Orang)	12% (6 Orang)	0%	0%
		Pengujian 2				

	mencari informasi yang diinginkan?	46% (23 Orang)	54% (27 Orang)	0%	0%	0%
8	Tampilan informasi dan elemen desain tidak boleh berlebihan, karena keduanya akan bersaing secara visual dengan informasi yang lebih relevan di layar. Langkah: 1. Perhatikan beberapa halaman dalam menu Profil Sekolah, apakah konten dapat terbaca dengan jelas? 2. Perhatikan halaman guru (pilih salah satu tombol dalam menu guru), apakah konten terbaca dengan jelas?	Pengujian 1				
		40% (20 Orang)	50% (25 Orang)	10% (5 Orang)	0%	0%
		Pengujian 2				
		48% (24 Orang)	52% (26 Orang)	0%	0%	0%
9	Ketika pengguna melakukan kesalahan, beri pesan yang jelas dan mudah dimengerti menggunakan bahasa yang sederhana. Langkah: 1. Klik menu 'denah sekolah' – klik tombol '2D' – klik gambar 'KOPSIS' – klik 'Play Video', karena video belum tersedia, maka ada peringatan dari sistem. Apakah mudah dimengerti? 2. Ketika membuka aplikasi untuk pertama kali, ada peringatan untuk registrasi xtras, apakah pengguna dapat keluar dari kondisi tersebut dengan mudah?	Pengujian 1				
		46% (23 Orang)	40% (20 Orang)	12% (6 Orang)	2% (1 Orang)	0%
		Pengujian 2				
		58% (29 Orang)	42% (21 Orang)	0%	0%	0%
10	Adanya bantuan dan dokumentasi pada sistem akan sangat membantu pengguna. Langkah: 1. Klik tombol 'petunjuk' pada halaman utama, apakah ulasan yang diberikan cukup membantu dalam penggunaan sistem? 2. Apakah cetakan dokumentasi membantu pengguna dalam mempelajari sistem?	Pengujian 1				
		48% (24 Orang)	42% (21 Orang)	8% (4 Orang)	2% (1 Orang)	0%
		Pengujian 2				
		58% (29 Orang)	42% (21 Orang)	0%	0%	0%
	Rata-rata Pengujian 1	40,4%	49,6%	9,6%	0,4%	0%
	Rata-rata Pengujian 2	54,4%	45,6%	0%	0%	0%

Pada pengujian pertama, data diatas menyebutkan dari 10 langkah dalam evaluasi heuristik yang ditujukan untuk 50 orang, rata-rata persentase yang didapatkan adalah 40,4% untuk Skala 0 (sangat baik), 49,6% untuk skala 1 (baik), 9,6% untuk skala 2 (cukup), 0,4% untuk skala 3 (kurang), dan tidak ada yang memberikan penilaian atau 0% untuk skala 4 (kurang sekali).

Sedangkan pada pengujian kedua, data diatas menyebutkan dari 10 langkah dalam evaluasi heuristik yang ditujukan untuk 50 orang, rata-rata persentase yang didapatkan adalah 54,4% untuk Skala 0 (sangat baik), 45,6% untuk skala 1 (baik), dan tidak ada yang memberikan penilaian atau 0% untuk skala 2,3, dan 4 (cukup, kurang, dan kurang sekali).

6.2.4 Perbaikan Produk

Dalam tahap ini akan dibahas perbaikan yang diberikan oleh pakar dan pihak sekolah sesuai komentar dan saran yang telah mereka berikan pada kuisisioner. Adapun hasil kuisisioner pengujian kedua (setelah dilakukan perbaikan produk) telah dijelaskan pada masing-masing tahap pengujian sebelumnya.

6.2.4.1 Perbaikan dari Pakar

1. Gunakan tombol 'halaman utama' lebih mirip seperti gambar rumah

Tanggapan penulis: penulis melakukan perbaikan perubahan tombol dari panah menghadap atas menjadi *icon* yang lebih mirip dan mudah dikenali sebagai rumah (*Home*) seperti gambar dibawah ini:



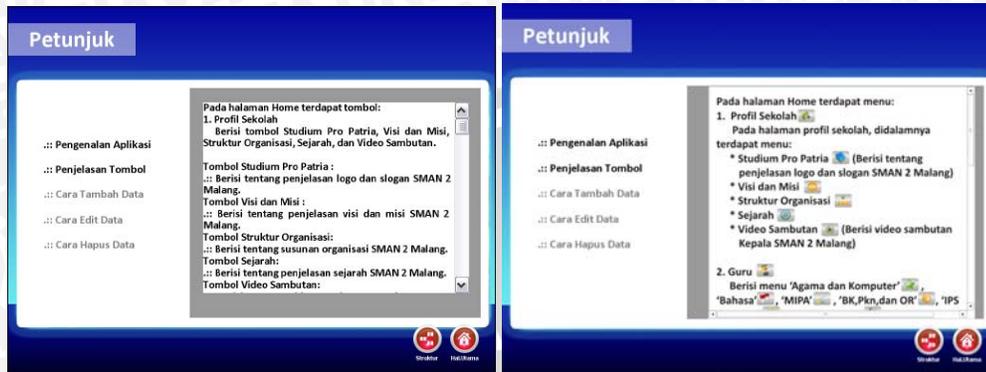
Gambar 6.13 Tombol 'Halaman Utama' sebelum dan sesudah direvisi

2. Perbaiki visualisasi petunjuk penggunaan aplikasi

Tanggapan penulis: Penulis telah memperbaiki visualisasi petunjuk sehingga lebih memudahkan pembaca memahami petunjuk yang diberikan.

Sebelum

Sesudah



Gambar 6.14 Penambahan gambar tombol pada halaman petunjuk

3. Pada halaman Denah, ada tombol 'back' yang tidak berfungsi

Tanggapan penulis: Penulis telah membuatnya berfungsi sesuai link yang dituju.

4. Perhatikan konsistensi peletakan menu

Tanggapan penulis: penulis telah merubah peletakan menu utama seperti menu 'Profil Sekolah', 'Guru', 'Kegiatan', 'Album', dan 'Denah Sekolah' menjadi sama dan lebih konsisten dalam hal peletakan menu, untuk lebih jelasnya perhatikan gambar dibawah ini:

Sebelum

Sesudah





Gambar 6.15 Hasil perbaikan konsistensi peletakan menu

5. **Penggunaan salah satu bahasa dalam aplikasi.**

Tanggapan penulis: penulis telah merubah beberapa keterangan tombol dari yang sebelumnya berbahasa inggris menjadi bahasa indonesia, contoh: *Home* – Ke Hal.Utama, *Back* – Ke Hal.Sebelumnya.

Sebelum

Sesudah



Gambar 6.16 Hasil perbaikan penggunaan bahasa Indonesia

6. **Tombol kurang intuitive, kurang memberikan feedback action**

Tanggapan penulis: sebelumnya tombol jika disorot oleh *cursor* tidak memberikan *feedback* seperti berganti warna atau mengeluarkan bunyi, dan

penulis telah mengganti semua tombol yang ada dalam aplikasi menjadi bisa memberikan *feedback* bagi pengguna seperti gambar dibawah ini:

Sebelum



Sesudah



Gambar 6.17 Hasil pemberian *feedback action* pada tombol

Sebagai contoh: sebelumnya, tombol 'intro' jika disorot tidak akan memberikan umpan balik apapun pada pengguna, dan setelah diperbaiki tombol 'awalan' (sebelum diperbaiki namanya intro) akan berubah warna menjadi hijau dan cursor berganti menjadi bentuk jari. Hal ini dapat memberikan *feedback action* bagi pengguna daripada sebelumnya.

7. Tambahkan buku manual supaya memudahkan pengguna

Tanggapan penulis: Buku manual telah disediakan sesuai masukan dari pakar. Hal ini penting mengingat jika panduan hanya diletakkan dalam aplikasi dan tidak ada dokumentasi, maka akan sulit bagi pengguna mengaplikasikan petunjuk yang sudah disediakan.

6.2.4.2 Perbaikan dari Pihak Sekolah

Pihak sekolah memberi masukan untuk perbaikan aplikasi, yaitu:

1. Lagu dalam aplikasi dibuat berbeda antara *user* dan admin.

Tanggapan penulis: penulis telah merubahnya sesuai keinginan pihak sekolah.

- Setiap denah ruangan diberi keterangan singkat disamping video yang menjelaskan tentang ruang tersebut.

Tanggapan penulis: penulis telah merubahnya sesuai keinginan pihak sekolah. Berikut tampilannya:

Sebelum

Sesudah



Gambar 6.18 Perbandingan halaman denah sebelum dan sesudah diperbaiki

- Sebaiknya setiap foto pada album kelas dilengkapi tombol untuk memperbarui data fotonya, karena setiap tahun siswanya berganti.

Tanggapan penulis: penulis telah merubahnya sesuai keinginan pihak sekolah.

Sebelum

Sesudah



Gambar 6.19 Halaman album kelas beserta tombol edit

Dengan hasil yang didapatkan penulis, dirasa sudah cukup aplikasi tersebut diaplikasikan di SMAN 2 Malang, masukan dari pakar dan pihak sekolah sangat bermanfaat untuk pengembangan aplikasi ini.