

**RANCANG BANGUN APLIKASI EDUKASI WAYANG
SEBAGAI MEDIA PENGENALAN TOKOH PEWAYANGAN
BERBASIS ANDROID**

SKRIPSI



**Disusun Oleh :
ACHMAD BISRI ISMAIL
10.18.010**

**PROGRAM STUDI TEKNIK INFORMATIKA S-1
FAKULTAS TEKNOLOGI INDUSTRI
INSTITUT TEKNOLOGI NASIONAL MALANG
2014**

LEMBAR PERSETUJUAN DAN PENGESAHAN
RANCANG BANGUN APLIKASI EDUKASI WAYANG
SEBAGAI MEDIA PENGENALAN TOKOH PEWAYANGAN
BERBASIS ANDROID

SKRIPSI

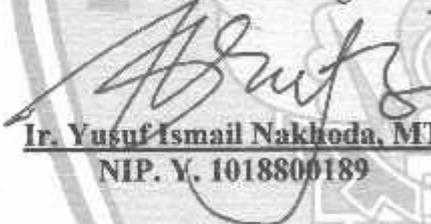
Disusun dan Diajukan untuk melengkapi dan memenuhi persyaratan guna mencapai Gelar Sarjana Komputer Strata Satu (S-1)

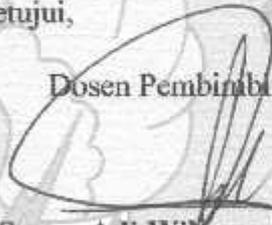
Disusun Oleh :
Achmad Bisri Ismail
10.18.010

Diperiksa dan Disetujui,

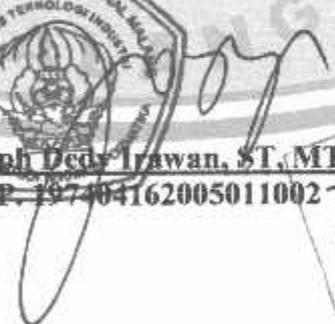
Dosen Pembimbing I

Dosen Pembimbing II


Ir. Yusuf Ismail Nakhoda, MT
NIP. Y. 1018800189


Suryo Adi Wibowo, ST, MT
NIP.P. 1031000438

Program Studi Teknik Informatika S-1
Fakultas Teknologi Industri


Joseph Dedy Irawan, ST, MT
NIP. 197404162005011002

MILIK
PERPUSTAKAAN
ITN MALANG

PROGRAM STUDI TEKNIK INFORMATIKA S-1
FAKULTAS TEKNOLOGI INDUSTRI
INSTITUT TEKNOLOGI NASIONAL MALANG
2014

**LEMBAR KEASLIAN
PERNYATAAN KEASLIAN SKRIPSI**

Saya yang bertanda tangan di bawah ini:

Nama : Achmad Bisri Ismail
NIM : 10.18.010
Program Studi : Teknik Informatika S-1
Fakultas : Teknologi Industri

Menyatakan dengan sesungguhnya bahwa Skripsi saya yang berjudul:
“RANCANG BANGUN APLIKASI EDUKASI WAYANG SEBAGAI MEDIA
PENGENALAN TOKOH PEWAYANGAN BERBASIS ANDROID”

Adalah Skripsi saya sendiri bukan duplikat serta mengutip atau menyadur
seluruhnya karya orang lain kecuali dari sumber aslinya.

Malang, 16 Februari 2014

Yang menyatakan


METERAI
TEMPEL
AFDESACF129365304
6000
DJP

Achmad Bisri Ismail

RANCANG BANGUN APLIKASI EDUKASI WAYANG SEBAGAI MEDIA PENGENALAN TOKOH PEWAYANGAN BERBASIS ANDROID

Achmad Bisri Ismail

Program Studi Teknik Informatika S-1
Fakultas Teknologi Industri
Institut Teknologi Nasional Malang
Jl. Raya Karanglo Km. 2 Tasikmadu-Malang
Email: sichabhie@gmail.com

**Dosen Pembimbing: 1. Ir. Yusuf Ismail Nakhoda, MT
2. Suryo Adi Wibowo, ST, MT**

Abstrak

Dewasa ini wayang sebagai produk budaya masih sering disalah pahami oleh orang asing, juga oleh orang kita sendiri. Posisi wayang sebagai produk kultural bangsa kita, sekarang ini posisinya makin terdesak. Hal ini disebabkan karena sulitnya memahami wayang bagi generasi mutakhir, dan masuknya nilai-nilai budaya dari luar yang kemudian lebih populer ketimbang wayang. Oleh karena itu, penulis berkeinginan untuk melestarikan budaya Indonesia ini dengan membuat suatu aplikasi pembelajaran berbasis Android untuk semua kalangan masyarakat. Mengingat semakin berkembang pesatnya smartphone smartphone yang ada dikalangan masyarakat dan salah satu kegunaan Android yang berfungsi sebagai media pembelajaran dan pengajaran.

Aplikasi ini dirancang sebagai edukasi wayang pada media pengenalan tokoh pewayangan yang didalamnya terdapat unsur-unsur sejarah tentang perkembangan wayang, cerita pewayangan, dan 150 macam tokoh pewayangan. Dari beberapa unsur tersebut kemudian dibuatlah 50 macam soal pertanyaan beserta jawaban yang berupa 3 pilihan ganda yang harus dijawab oleh pengguna.

Dalam proses implementasinya, aplikasi ini dibuat untuk jenis orientasi portrait. Dengan tingkat keakuratan kinerja dari fungsi sistem aplikasi mencapai 100 %. Hasil pengujian aplikasi ini meliputi pengujian kinerja aplikasi beserta layoutnya, pengujian operating system, dan pengujian user. Aplikasi wayang ini dapat digunakan kedalam sistem operasi Android antara versi Froyo (2.2.1) hingga versi Jelly Bean (4.1.2). Sedangkan ukuran untuk file .APK aplikasi wayang ini sebesar 27,5 MB.

Kata Kunci : *Android, Wayang, Edukasi*

KATA PENGANTAR

Puji syukur kehadiran Allah yang maha kuasa, karena telah memberikan rahmat dan hidayah-Nya sehingga penyusun dapat menyelesaikan skripsi dengan judul RANCANG BANGUN APLIKASI EDUKASI WAYANG SEBAGAI MEDIA PENGENALAN TOKOH PEWAYANGAN BERBASIS ANDROID sesuai dengan waktu yang ditentukan.

Penyusunan skripsi ini merupakan salah satu persyaratan untuk menyelesaikan program pendidikan Strata Satu (S-1) Teknik Informatika, Fakultas Teknologi Industri di Institut Teknologi Nasional Malang.

Pada penyusunan skripsi ini penyusun mengucapkan banyak terima kasih sebesar-besarnya kepada :

1. Bapak Suwarno Hadi dan Ibu Sarilah, yang merupakan kedua orang tua dan pendukung utama dari segi moril maupun materil.
2. Ir.Soeparno Djiwo, MT, selaku Rektor Institut Teknologi Nasional Malang.
3. Ir. Anang Subardi, MT, selaku Dekan Fakultas Teknologi Industri, Institut Teknologi Nasional Malang.
4. Joseph Dedy Irawan, ST, MT, selaku Ketua Program Studi Teknik Informatika, Institut Teknologi Nasional Malang.
5. Sonny Prasetio, ST, MT, selaku Sekertaris Program Studi Teknik Informatika, Institut Teknologi Nasional Malang.
6. Yusuf Ismail Nakhoda, MT, selaku Dosen Pembimbing I, yang selalu memberikan masukan.
7. Suryo Adi Wibowo, ST, MT, selaku Dosen Pembimbing II dan sekaligus Kepala Laboratorium Robotika
8. Semua dosen Program Studi Teknik Informatika yang telah membantu dalam penulisan dan masukan.
9. Semua teman seperjuangan yang telah membantu dalam terselesaikannya skripsi ini.

Penyusun menyadari bahwa skripsi masih jauh dari sempurna, oleh karena itu penyusun mengharapkan kritik dan saran dari pembaca, Semoga skripsi ini bisa bermanfaat bagi pembaca.

Malang, Februari 2014

Penyusun

DAFTAR ISI

HALAMAN JUDUL	i
LEMBAR PERSETUJUAN	ii
LEMBAR KEASLIAN	iii
ABSTRAK	iv
KATA PENGANTAR.....	v
DAFTAR ISI.....	vii
DAFTAR TABEL	ix
DAFTAR GAMBAR.....	x
BAB I PENDAHULUAN	
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. Metode Penelitian	3
1.6.1 Studi Literatur	3
1.6.2 Perancangan Aplikasi	3
1.6.3 Pembuatan Aplikasi	4
1.6.4 Uji Coba dan Evaluasi Aplikasi	4
1.6.5 Penyusunan Laporan dan Kesimpulan Akhir	4
1.7. Sistematika Penulisan	4
BAB II DASAR TEORI	
2.1. Wayang	6
2.2. Wayang Kulit Purwo.....	10
2.3. Aplikasi Edukasi	11
2.4. Metode <i>Auto-Complete</i>	11
2.5. Sistem Operasi <i>Android</i>	12
2.6. Konsep Umum <i>Android</i>	16
2.7. <i>Eclipse</i>	19
2.8. SDK (<i>Software Development Kit</i>)	21
2.9. JDK (<i>Java Development Kit</i>)	22
2.10. ADT (<i>Android Development Tools</i>)	22
2.11. AVD (<i>Android Virtual Device</i>)	22
2.12. Bahasa Pemrograman <i>Java</i>	23
2.13. <i>Database</i>	24
2.14. <i>SQLite</i>	26
2.15. <i>SQLite Manager</i>	27
BAB III ANALISA DAN PERANCANGAN	
3.1. Analisa Sistem	29
3.1.1. Analisis Prosedur yang Berjalan	29
3.1.2. Analisis Kebutuhan <i>Data</i>	31
3.1.3. Analisis Kebutuhan Sistem	31
3.1.4. Analisis Kebutuhan Perangkat Lunak	31

3.2. <i>Use Case Diagram</i>	32
3.3. Perancangan Sistem	33
3.3.1. Rancangan Struktur Navigasi.....	33
3.3.2. Rancangan Diagram Alir (<i>Flowchart</i>)	34
3.3.3. Rancangan Struktur Tabel	36
3.3.4. Rancangan <i>Interface</i>	37

BAB IV IMPLEMENTASI DAN PENGUJIAN

4.1. Implementasi.....	51
4.2. Implementasi <i>User Interface</i>	51
4.2.1. Tampilan <i>Splash Screen</i>	52
4.2.2. Tampilan <i>Home</i>	53
4.2.3. Tampilan Tentang	54
4.2.4. Tampilan Petunjuk	55
4.2.5. Tampilan Keluar	55
4.2.6. Tampilan <i>Menu Utama</i>	56
4.2.7. Tampilan Sejarah	57
4.2.8. Tampilan <i>Opsi Cerita</i>	58
4.2.9. Tampilan <i>Detail Cerita Ramayana</i>	59
4.2.10. Tampilan <i>Detail Cerita Mahabharata</i>	60
4.2.11. Tampilan <i>Opsi Tokoh</i>	61
4.2.12. Tampilan <i>List Tokoh Dewa dan Dewi</i>	62
4.2.13. Tampilan Fitur <i>Autocomplete</i>	63
4.2.14. Tampilan <i>Detail Tokoh Dewa dan Dewi</i>	64
4.2.15. Tampilan <i>List Tokoh Ramayana</i>	65
4.2.16. Tampilan <i>Detail Tokoh Ramayana</i>	66
4.2.17. Tampilan <i>List Tokoh Mahabharata</i>	67
4.2.18. Tampilan <i>Detail Tokoh Mahabharata</i>	68
4.2.19. Tampilan Soal	69
4.2.20. Tampilan hasil Nilai	70
4.2.21. Tampilan Simpan Data	71
4.2.22. Tampilan <i>Rating Data</i>	73
4.2.23. Implementasi Antar Muka	72
4.3. Pengujian	77
4.2.1. Pengujian Fungsional System	74
4.2.2. Pengujian Spesifikasi Hardware	77
4.2.3. Pengujian Operating System Android	78
4.2.4. Pengujian User	79

BAB V PENUTUP

5.1 Kesimpulan.....	81
5.2 Saran	81

DAFTAR PUSTAKA.....	84
---------------------	----

LAMPIRAN.....	85
---------------	----

DAFTAR TABEL

Tabel 3.1.	Tabel Soal	36
Tabel 3.2.	Tabel <i>User</i>	37
Tabel 4.1.	Tabel Pengujian Fungsional <i>System Android</i>	74
Tabel 4.2.	Tabel Pengujian Spesifikasi <i>Hardware</i>	77
Tabel 4.3.	Tabel Pengujian <i>Operating System Android</i>	78
Tabel 4.4.	Tabel Pengujian <i>User</i>	79

DAFTAR GAMBAR

Gambar 2.1 Peta Penyebaran Cerita Wayang	9
Gambar 2.2 Wayang Cina, Muangthai dan Kamboja	10
Gambar 2.3 Wayang Purwo Jawa Timur	11
Gambar 2.4 Cara Kerja <i>Autocomplete</i>	12
Gambar 2.5 Siklus <i>Activity Android</i>	16
Gambar 2.6 Kebutuhan <i>Software</i>	20
Gambar 2.7 Arsitektur Program <i>Java</i>	24
Gambar 2.8 Analogi <i>cursor</i> pada <i>SQLite Database</i>	27
Gambar 2.9 <i>SQLite Manager</i>	28
Gambar 3.1 <i>Flowchart Analisis yang Berjalan</i>	30
Gambar 3.2 <i>Use Case Diagram</i>	32
Gambar 3.3 Proses Interaksi Sistem Dengan <i>User</i>	33
Gambar 3.4 Struktur Navigasi <i>User</i>	33
Gambar 3.5 <i>Flowchart User Aplikasi Mobile Untuk Menu</i>	34
Gambar 3.6 <i>Flowchart User Aplikasi Mobile Untuk Proses Scoring</i>	35
Gambar 3.7 <i>Opening Layout</i>	37
Gambar 3.8 <i>Layout Home</i>	38
Gambar 3.9 <i>Layout Tentang</i>	39
Gambar 3.10 <i>Layout Petunjuk</i>	39
Gambar 3.11 <i>Layout Keluar</i>	40
Gambar 3.12 <i>Layout Menu Utama</i>	41
Gambar 3.13 <i>Layout Sejarah</i>	41
Gambar 3.14 <i>Layout Opsi Cerita</i>	42
Gambar 3.15 <i>Layout Cerita Ramayana</i>	43
Gambar 3.16 <i>Layout Cerita Mahabharata</i>	43
Gambar 3.17 <i>Layout Opsi Cerita</i>	44
Gambar 3.18 <i>Layout Tokoh Dewa Dan Dewi</i>	45
Gambar 3.19 <i>Layout Detail Tokoh Dewa Dan Dewi</i>	45
Gambar 3.20 <i>Layout Tokoh Ramayana</i>	46
Gambar 3.21 <i>Layout Detail Tokoh Ramayana</i>	47

Gambar 3.22 <i>Layout</i> Tokoh Mahabharata.....	47
Gambar 3.23 <i>Layout Detail</i> Tokoh Mahabharata.....	48
Gambar 3.24 <i>Layout</i> Soal.....	49
Gambar 3.25 <i>Layout Input</i> Nama User	49
Gambar 3.26 <i>Layout</i> Nilai.....	50
Gambar 4.1 Tampilan <i>Splash Screen</i>	52
Gambar 4.2 Tampilan <i>Home</i>	53
Gambar 4.3 Tampilan <i>Tentang</i>	54
Gambar 4.4 Tampilan <i>Petunjuk</i>	55
Gambar 4.5 Tampilan <i>Keluar</i>	56
Gambar 4.6 Tampilan <i>Menu Utama</i>	57
Gambar 4.7 Tampilan <i>Sejarah</i>	58
Gambar 4.8 Tampilan <i>Opsi Cerita</i>	59
Gambar 4.9 Tampilan <i>Detail Cerita</i> Ramayana	60
Gambar 4.10 Tampilan <i>Detail</i> cerita Mahabharata.....	61
Gambar 4.11 Tampilan <i>Opsi</i> Tokoh.....	62
Gambar 4.12 Tampilan <i>List</i> Tokoh Dewa Dan Dewi.....	63
Gambar 4.13 Tampilan Fitur Pencarian <i>Autocomplete</i>	64
Gambar 4.14 Tampilan <i>Detail</i> Tokoh Dewa Dan Dewi.....	65
Gambar 4.15 Tampilan <i>List</i> Tokoh Ramayana	66
Gambar 4.16 Tampilan <i>Detail</i> Tokoh Ramayana	67
Gambar 4.17 Tampilan <i>List</i> Tokoh Mahabharata	68
Gambar 4.18 Tampilan <i>Detail</i> Tokoh Mahabharata	69
Gambar 4.19 Tampilan Soal.....	70
Gambar 4.20 Tampilan Hasil Nilai	71
Gambar 4.21 Tampilan <i>Simpan Data</i>	72
Gambar 4.22 Tampilan <i>Rating Data</i>	73

BAB I PENDAHULUAN

1.1 Latar Belakang

Wayang adalah seni pertunjukkan asli Indonesia yang berkembang pesat di Pulau Jawa dan Bali. Selain itu beberapa daerah seperti Sumatera dan Semenanjung Malaya juga memiliki beberapa budaya wayang yang terpengaruh oleh kebudayaan Jawa dan Hindu.

UNESCO, lembaga yang membawahi kebudayaan dari PBB, pada 7 November 2003 menetapkan wayang sebagai pertunjukkan bayangan boneka tersohor dari Indonesia, sebuah warisan mahakarya dunia yang tak ternilai dalam seni bertutur (*Masterpiece of Oral and Intangible Heritage of Humanity*). Pertunjukan bayangan boneka (Wayang) di Indonesia memiliki gaya tutur dan keunikan tersendiri, yang merupakan mahakarya asli dari Indonesia. Untuk itulah UNESCO memasukannya ke dalam Daftar Representatif Budaya Takbenda Warisan Manusia pada tahun 2003.

Aplikasi edukasi merupakan suatu aplikasi yang dibuat hanya untuk dunia pembelajaran. Biasanya pada aplikasi ini mempunyai berbagai macam fitur - fitur dalam proses pembelajaran sehingga dapat memberikan kemudahan kepada pengguna. Dengan berkembangnya teknologi yang sangat pesat. Aplikasi edukasi ini sering digunakan untuk memperkenalkan tentang suatu hal yang sulit dilakukan oleh sistem pembelajaran manual.

Aplikasi ini dibuat dengan menggunakan sistem operasi Android. Dimana Android ini adalah suatu sistem operasi *open source* yang berbasis Linux untuk telepon seluler seperti telepon pintar dan komputer tablet. *Android* menyediakan *platform* terbuka bagi para pengembang untuk menciptakan aplikasi mereka sendiri untuk digunakan oleh bermacam peranti bergerak. Maka, dengan adanya sistem operasi *Android* ini penulis berkeinginan untuk menerapkannya kepada sistem pembelajaran tentang pengenalan tokoh – tokoh pewayangan. Karena dengan aplikasi wayang ini pengguna bisa belajar dan mengenali tokoh – tokoh dalam pewayangan kapanpun dan dimanapun dengan mudah tanpa harus mencari literatur – literatur di internet terlebih dahulu.

1.2 Rumusan Masalah

Berdasarkan latar belakang masalah diatas dapat dirumuskan permasalahan yang akan dibahas adalah :

1. Bagaimana merancang sebuah aplikasi yang mengedukasi kepada masyarakat agar mengenali dan mengerti dengan tokoh pewayangan yang ada di Indonesia.
2. Bagaimana menciptakan sebuah perangkat lunak dalam bentuk aplikasi yang dapat dijadikan sebagai sarana dalam pengenalan tokoh pewayangan yang berbasis *Android*.

1.3 Batasan Masalah

Adapun batasan masalah tugas akhir ini yaitu meliputi :

1. *Software* yang digunakan dalam pembuatan aplikasi ini adalah *Eclipse Juno*, sedangkan database yang digunakan adalah *SQLite*.
2. Jenis wayang yang digunakan pada aplikasi ini adalah wayang kulit purwo, yang didalamnya terdapat tokoh pewayangan yang disediakan sebanyak 150 macam tokoh, sedangkan jenis cerita yang disediakan yaitu cerita ramayana dan mahabharata serta soal yang disediakan sebanyak 50 macam soal yang diambil dari setiap tokoh dan cerita.
3. Jenis wayang yang digunakan pada aplikasi ini adalah wayang kulit purwo.
4. Media yang digunakan dalam aplikasi ini adalah *text*, dan gambar.
5. Fitur – fitur yang disediakan antara lain sejarah, tokoh, cerita, pencarian, tokoh dan cerita pewayangan, serta soal tes.

1.4 Tujuan Penelitian

Adapun tujuan penelitian ini adalah untuk merancang dan membuat suatu *software* aplikasi yang dapat mengedukasi kepada para penggunan dan dapat digunakan sebagai kamus data tokoh pewayangan yang dapat dibuka di *smartphone* khususnya *Android*.

1.5 Manfaat Penelitian

Adapun manfaat yang dapat diambil dari pembuatan aplikasi yang berbasis android ini adalah :

1. Bagi pengguna atau masyarakat, sebagai aplikasi *software* pembantu untuk mengenali tokoh – tokoh pewayangan. Sehingga, budaya Indonesia yang sangat berharga ini tidak luntur begitu saja oleh perkembangan teknologi serta pengaruh dunia barat.
2. Mempermudah semua lapisan masyarakat yang ingin mengetahui informasi tentang wayang.
3. Bagi pihak lain, sebagai bahan referensi dan tambahan informasi untuk pengkajian topik yang berkaitan dengan masalah yang sama dengan penelitian ini dan dapat digunakan sebagaimana mestinya.

1.6 Metode Penelitian

Untuk dapat mencapai keinginan dalam pembuatan aplikasi wayang sebagai media pengenalan tokoh pewayangan berbasis android ini, maka perlu dilakukan langkah-langkah sebagai berikut:

1.6.1 Studi Literatur

Penelitian ini dimulai dengan studi literatur yaitu pengumpulan data yang berhubungan dengan permasalahan yang dibahas sehingga dapat membantu penyelesaian masalah dalam perancangan aplikasi ini.

1.6.2 Perancangan Aplikasi

Pada tahap ini dilakukan perancangan terhadap system yang akan dibangun seperti :

1. Perancangan *opening layout* yang berisi gambar pembuka aplikasi ini.
 2. Perancangan *layout home* yang berisi 4 macam tombol yaitu tombol tentang, tombol petunjuk, tombol menu dan tombol keluar.
 3. Perancangan menu yang berisi 4 macam tombol yaitu tombol sejarah, tombol tokoh beserta proses pencarian, tombol cerita dan tombol soal.
-

Dalam tahap ini merupakan tahapan perancangan sistem secara keseluruhan, maka tahapan ini merupakan tahapan terpenting dalam pembuatan aplikasi *software* ini.

1.6.3 Pembuatan Aplikasi

Pada tahap ini sistem yang telah dirancang kemudian di implementasikan ke dalam sebuah bahasa pemrograman. Bahasa pemrograman yang digunakan adalah bahasa perograman *java*.

1.6.4 Uji coba dan Evaluasi Aplikasi

Uji coba ini bertujuan untuk memastikan bahwa masing-masing bagian dari sistem ini dapat bekerja sesuai yang diharapkan.

1.6.5 Penyusunan Laporan dan Kesimpulan Akhir

Pada tahap ini dilakukan penyusunan laporan tentang seluruh isi penelitian dan analisisnya ke dalam format penulisan tugas akhir dengan disertai kesimpulan akhir.

1.7 Sistematika Penulisan

Sistematika penyusunan proposal skripsi ini untuk memberikan gambaran secara umum dari proposal skripsi yang dibuat yang meliputi bab-bab sebagai berikut :

BAB I : PENDAHULUAN

Pada Bab I ini memuat tentang Latar Belakang, Rumusan Masalah, Batasan Masalah, Tujuan Penelitian, Manfaat Penelitian, Metode Penelitian serta Sistematika Penulisan.

BAB II : LANDASAN TEORI

Pada Bab II ini membahas tentang *software* yang akan digunakan, bahasa pemrograman yang digunakan, metode yang digunakan serta berbagai macam *software* pendukung yang akan digunakan dalam pembuatan aplikasi *software* ini.

BAB III : ANALISIS DAN PERANCANGAN SISTEM

Pada bab III ini akan membahas tentang gambaran umum dari tata letak menu dan tampilan yang akan dibuat. Dalam perancangan sistem akan ditentukan rancangan struktur menu, Proses Interaksi, *Use Case*, *Flowchart* dan *Layout Prototype*.

BAB IV : IMPLEMENTASI DAN PEMBAHASAN

Pada bab IV ini akan membahas tentang hasil implementasi dan analisis hasil uji coba program. Serta memaparkan hasil-hasil dari tahap penelitian, tahap analisis, desain, implementasi desain, hasil testing dan implementasinya, berupa penjelasan teoritik, baik secara kualitatif, kuantitatif, atau secara statistik.

BAB V : PENUTUP

Pada bab penutup ini akan membahas tentang kesimpulan dan saran. Kesimpulan diambil dari data-data hasil pengujian, apakah hasil yang didapat berhasil mengembangkan sistem yang sudah ada dan apakah hasil pengembangan sudah layak untuk diimplementasikan.

BAB II LANDASAN TEORI

2.1 Wayang

Wayang merupakan salah satu bentuk teater tradisional yang paling tua. Pada masa pemerintahan Raja Balitung, telah ada petunjuk adanya pertunjukan wayang, yaitu yang terdapat pada prasasti Balitung dengan tahun 907 Masehi, yang mewartakan bahwa pada saat itu telah dikenal adanya pertunjukan wayang.

Prasasti berupa lempengan tembaga dari Jawa Tengah, *Royal Tropical Institute*, Amsterdam, contoh prasasti ini dapat dilihat dalam lampiran buku *Claire Holt Art in Indonesia: Continuities and Changes*, 1967 terjemahan Prof. Dr. Soedarsono (MSPI-2000-hal 431). Tertulis sebagai berikut: Dikeluarkan atas nama Raja Balitung teks ini mengenai desa Sangsang, yang ditandai sebagai sebuah tanah perdikan, yang pelaksanaannya ditujukan kepada dewa dari serambi di Dalinan. Lagi setelah menghias diri dengan cat serta bunga-bunga para peserta duduk di dalam tenda perayaan menghadap Sang Hyang Kudur. "Untuk keselamatan bangunan suci serta rakyat" pertunjukan (ton-tonan) disakikan. Sang Tangkil Hyang sang (mamidu), si Nalu melagukan (macarita) Bhima Kumara, serta menari (mangigai) sebagai Kicaka, si Jaluk melagukan Ramayana, si Mungmuk beracting (mamirus) serta melawak (mebanol), si Galigi mempertunjukan Wayang (mawayang) bagi para Dewa, melagukan Bhimaya Kumara. Pentingnya teks ini terletak pada indikasi yang jelas bahwa pada awal abad ke-10, *episode-episode* dari Mahabharata dan Ramayana dilagukan dalam peristiwa-peristiwa ritual.

Bhimaya Kumara mungkin sebuah cerita yang berhubungan dengan Bima boleh jadi telah dipertunjukan sebagai sebuah teater bayangan (sekarang wayang purwa). Dari mana asal-usul wayang, sampai saat ini masih dipersoalkan, karena kurangnya bukti-bukti yang mendukungnya. Ada yang meyakini bahwa wayang asli kebudayaan Jawa dengan mengatakan karena istilah-istilah yang digunakan dalam pewayangan banyak istilah bahasa Jawa.

Dr. G. A. J. Hazeu, dalam detertasinya *Bijdrage tot de Kennis van het Javaansche Tooneel* (Th 1897 di Leiden, Negeri Belanda) berkeyakinan bahwa

pertunjukan wayang berasal dari kesenian asli Jawa. Hal ini dapat dilihat dari istilah-istilah yang digunakan banyak menggunakan bahasa Jawa misalnya, kelir, blencong, cempala, keyak, wayang. Pada susunan rumah tradisional di Jawa, kita biasanya akan menemukan bagian-bagian ruangan: emper, pendhapa, omah mburi, gandhok sen-thong dan ruangan untuk pertunjukan ringgit (pringgitan), dalam bahasa Jawa ringgit artinya wayang. Bagi orang Jawa dalam membangun rumahpun menyediakan tempat untuk pertunjukan wayang. Dalam buku *Over de Oorsprong van het Java-ansche Tooneel* - Dr.W Rassers mengatakan bahwa, pertunjukan wayang di Jawa bukanlah ciptaan asli orang Jawa. Pertunjukan wayang di Jawa, merupakan tiruan dari apa yang sudah ada di India. Di India pun sudah ada pertunjukan bayang-bayang mirip dengan pertunjukan wayang di Jawa.

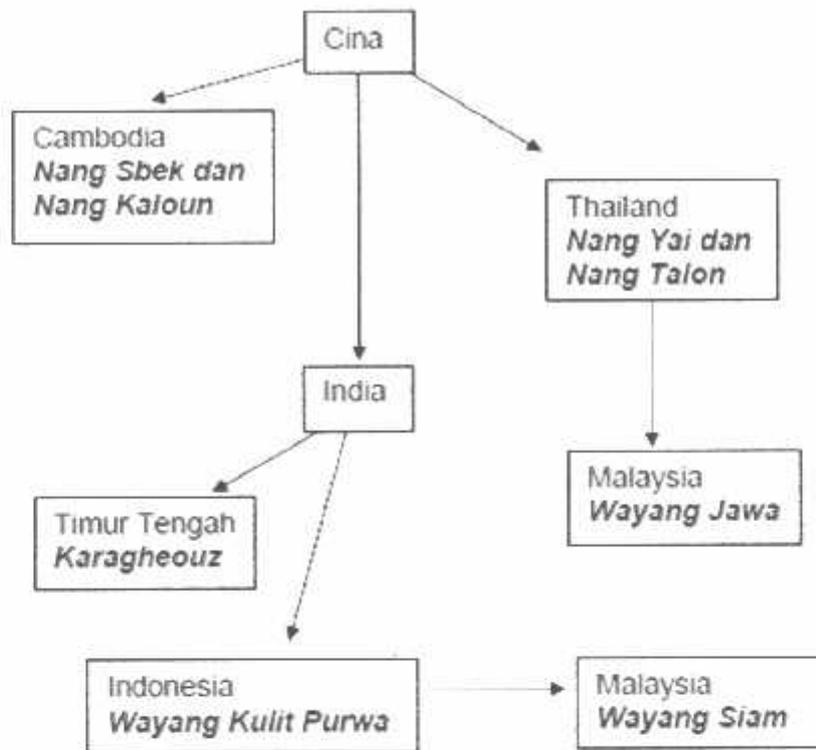
Dr.N.J. Krom sama pendapatnya dengan Dr. W. Rassers, yang mengatakan pertunjukan wayang di Jawa sama dengan apa yang ada di India Barat, oleh karena itu ia menduga bahwa wayang merupakan ciptaan Hindu dan Jawa. Ada pula peneliti dan penulis buku lainnya yang mengatakan bahwa wayang berasal dari India, bahkan ada pula yang mengatakan dari Cina. Dalam buku *Chineesche Brauche und Spiele in Europa* – Prof G. Schlegel menulis, bahwa dalam kebudayaan Cina kuno terdapat pertunjukan semacam wayang. Pada pemerintahan Kaizar Wu Ti, sekitar tahun 140 sebelum Masehi, ada pertunjukan bayang-bayang semacam wayang. Kemudian pertunjukan ini menyebar ke India, baru kemudian dari India dibawa ke Indonesia. Untuk memperkuat hal ini, dalam majalah *Koloniale Studien*, seorang penulis mengemukakan adanya persamaan kata antara bahasa Cina *Wa-yaah (Hokian)*, *Wo-yong (Kanton)*, *Woying (Mandarin)*, artinya pertunjukan bayang-bayang, yang sama dengan wayang dalam bahasa Jawa.

Meskipun di Indonesia orang sering mengatakan bahwa wayang asli berasal dari Jawa Indonesia, namun harus dijelaskan apa yang asli materi wayang atau wujud wayang dan bagaimana dengan cerita wayang. Pertanyaannya, mengapa pertunjukan wayang kulit, umumnya selalu mengambil cerita dari epos Ramayana dan Mahabharata. Dalam papernya *Attempt at a historical outline of the shadow theatre* Jacques Brunet, (Kuala Lumpur, 27-30 Agustus 19- 69), mengatakan, sulit untuk menyanggah atau menolak anggapan bahwa teater wayang yang terdapat di Asia Tenggara berasal dari India terutama tentang sumber cerita. Paper tersebut di

atas mencoba untuk menjelaskan bahwa wayang mempunyai banyak kesamaan terdapat di daerah Asia terutama Asia Tenggara dengan diikat oleh cerita-cerita yang sama yang bersumber dari Ramayana dan Mahabharata dari India. Sejarah penyebaran wayang dari India ke Barat sampai ke Timur Tengah dan ke timur umumnya sampai ke Asia Tenggara.

Di Timur Tengah, disebut *Karagheuz*, di Thailand disebut *Nang Yai dan Nang Talun*, di Cambodia disebut *Nang Shek dan Nang Koloun*. Dari Thailand ke Malaysia disebut *Wayang Siam*. Sedangkan yang langsung dari India ke Indonesia disebut *Wayang Kulit Purwa*. Dari Indonesia ke Malaysia disebut *Wayang Jawa*. Di Malaysia ada 2 jenis nama wayang, yaitu *Wayang Jawa* (berasal dari Jawa) dan *Wayang Siam* berasal dari Thailand.

Abad ke-4 orang-orang Hindu datang ke Indonesia, terutama para pedagangannya. Pada kesempatan tersebut orang-orang Hindu membawa ajarannya dengan Kitab Weda dan epos cerita maha besar India yaitu Mahabharata dan Ramayana dalam bahasa Sanskrit. Abad ke-9, bermunculan cerita dengan bahasa Jawa kuno dalam bentuk kakawin yang bersumber dari cerita Mahabharata atau Ramayana, yang telah diadaptasi kedalam cerita yang berbentuk kakawin tersebut, misalnya cerita-cerita seperti: *Arjuna wiwaha* karangan Empu Kanwa, *Bharatayuda* karangan Empu Sedah dan Empu Panuluh, *Kresnayana* karangan Empu Triguna, *Gatatkaca Sraya* karangan Empu Panuluh dan lain-lainnya. Pada jamannya, semua cerita tersebut bersumber dari cerita Mahabharata, yang kemudian diadaptasi sesuai dengan sejarah pada jamannya dan juga disesuaikan dengan dongeng serta legenda dan cerita rakyat setempat. Dalam mengenal wayang, kita dapat mendekatinya dari segi sastra, karena cerita yang dihidangkan dalam wayang terutama wayang kulit umumnya selalu diambil dari epos Mahabharata atau Ramayana. Kedua cerita tersebut, apabila kita telusuri sumber ceritanya berasal dari India. Mahabharata bersumber dari karangan *Viyasa*, sedangkan Epos Ramayana karangan *Valmiki*. Peta Penyebaran Cerita Wayang dari Cina. Adapun peta penyebaran cerita wayang ditunjukkan pada gambar 2.1.



Gambar 2.1 Peta Penyebaran Cerita Wayang

(Lihat: buku *Traditional Drama And Music of Southeast Asia – Edited by M.Taib Osman*, Terbitan Dewan Bahasa dan Pustaka, Kuala Lumpur. Th. 1974)

Hal ini diperkuat fakta bahwa cerita wayang yang terdapat di Asia terutama di Asia Tenggara yang umumnya menggunakan sumber cerita Ramayana dan Mahabharata dari India. Cerita-cerita yang biasa disajikan dalam wayang, sebenarnya merupakan adaptasi dari epos Ramayana dan Mahabharata yang disesuaikan dengan cerita rakyat atau dongeng setempat. Dalam sejarahnya pertunjukan wayang kulit selalu dikaitkan dengan suatu upacara, misalnya untuk keperluan upacara khitanan, bersih desa, menyingkirkan malapetaka dan bahaya. Hal tersebut sangat erat dengan kebiasaan dan adat-istiadat setempat.

Dalam menelusuri sejak kapan ada pertunjukan wayang di Jawa, dapat kita temukan berbagai prasasti pada jaman raja-raja Jawa, antara lain pada masa Raja Balitung. Namun tidak jelas apakah pertunjukan wayang tersebut seperti yang kita saksikan sekarang. Pada masa pemerintahan Raja Balitung, telah ada petunjuk adanya pertunjukan wayang. Hal ini juga ditemukan dalam sebuah kakawin

Arjuna wiwaha karya Empu Kanwa, pada jaman Raja Airlangga dalam abad ke-11. Oleh karenanya pertunjukan wayang dianggap kesenian tradisi yang cukup tua. Sedangkan bentuk wayang pada pertunjukan di jaman itu belum jelas tergambar bagaimana bentuknya.

Pertunjukan teater tradisional pada umumnya digunakan untuk pendukung sarana upacara baik keagamaan ataupun adat-istiadat, tetapi pertunjukan wayang kulit dapat langsung menjadi ajang keperluan upacara tersebut. Ketika kita menonton wayang, kita langsung dapat menerka pertunjukan wayang tersebut untuk keperluan apa. Hal ini dapat dilihat langsung pada cerita yang dimainkan, apakah untuk keperluan menyambut panen atau untuk *ngriwat* dan pertunjukan itu sendiri merupakan suatu upacara. Adapun wayang cina, muangthai dan kamboja ditunjukkan pada gambar 2.2. [7]



Gambar 2.2 Wayang Cina, Muangthai dan Kamboja

2.2 Wayang Kulit Purwo

Wayang Purwa adalah pertunjukan wayang yang pementasan ceritanya bersumber pada kitab Ramayana dan Mahabharata. Wayang tersebut dapat berupa wayang kulit, wayang golek atau wayang wong (orang). Pendapat para ahli, istilah purwa tersebut berasal dari kata parwa yang berarti bagian dari cerita Ramayana atau Mahabharata. Di kalangan masyarakat Jawa, terutama orang-orang tua kata purwa sering diartikan pula purba artinya zaman dulu. Sesuai dengan pengertian tersebut, maka wayang purwa diartikan pula sebagai wayang yang menyajikan cerita-cerita zaman dahulu. Pada wayang jenis ini banyak kita jumpai beberapa

ragam, sejarah dan asal mulanya. Adapun wayang purwo jawa timur ditunjukkan pada gambar 2.3. [7]



Gambar 2.3 Wayang Purwo Jawa Timur

2.3 Aplikasi Edukasi

Aplikasi Edukasi merupakan suatu *software* aplikasi yang dibuat untuk keperluan pendidikan. Dimana segala macam fitur – fitur yang dibuat dalam aplikasi ini bisa digunakan untuk membantu dan mempermudah proses pendidikan. Aplikasi-aplikasi dalam suatu paket biasanya memiliki antar muka pengguna yang memiliki kesamaan sehingga memudahkan pengguna untuk mempelajari dan menggunakan setiap aplikasi. Sering kali, mereka memiliki kemampuan untuk saling berinteraksi satu sama lain sehingga menguntungkan pengguna.

2.4 Metode Auto – Complete

Autocomplete merupakan metode pencarian yang dimiliki *google*. Metode ini juga ditanamkan pada *android* sebagai metode pencarian kata. Pada pembentukan fitur *autocomplete* atau *autocompletion*, struktur data yang diperlukan adalah *array* yang berisi *reserved words* dan nama fungsi dari berbagai bahasa pemrograman yang didukung oleh *text editor* atau *Integrated Development Environment (IDE)* serta daftar kata apa aja yang telah ditulis oleh pengguna. *Array* yang digunakan tersebut sudah terurut secara alfabetik.

Ketika pengguna mengetikkan karakter atau huruf pertama, fitur *autocomplete* atau *auto completion* akan bekerja dengan mencari kata yang berada dalam *array* yang karakter pertamanya sama dengan karakter yang diketikkan oleh pengguna. Setelah semua elemen *array* sudah diperiksa, elemen *array* yang bersesuaian dengan karakter pertama yang diketikkan oleh pengguna ditampilkan kedalam *menu list*. Proses pemrediksian berulang ketika pengguna mengetikkan karakter kedua, namun *array* yang diperiksa adalah *array* hasil dari pemeriksaan sebelumnya. Adapun salah - satu jenis pencarian yang menggunakan metode *autocomplete* ditunjukkan pada gambar 2.4.



Gambar 2.4 Cara Kerja *Autocomplete*

Cara kerja algoritma pemrediksiannya adalah sebagai berikut:

1. Pengguna mengetikkan karakter pertama.
2. Program akan mencari apakah karakter pertama tersebut ada di simpul pertama tingkat pertama pohon.
3. Jika ada maka keluarkan seluruh kata yang berada di seluruh anak simpul tersebut.
4. Jika tidak ada maka cari di simpul selanjutnya tingkat pertama hingga seluruh simpul tingkat pertama sudah dikunjungi.
5. Ulangi langkah b untuk karakter selanjutnya pada simpul tingkat kedua dan seterusnya. [4]

2.5 Sistem Operasi *Android*

Android adalah sistem operasi untuk telepon seluler yang berbasis *Linux*. *Android* menyediakan platform terbuka bagi para pengembang untuk menciptakan

aplikasi mereka sendiri sehingga dapat digunakan oleh bermacam peranti penggerak. Awalnya *Google Inc.* membeli *Android Inc.* pendatang baru yang membuat *software* (perangkat lunak) untuk telepon genggam. Kemudian untuk mengembangkan *Android* di bentuklah *Open Handset Alliance* yang merupakan gabungan dari 34 perusahaan peranti keras, peranti lunak dan telekomunikasi termasuk *Google*, *HTC*, *Intel*, *Motorola*, *Qualcomm*, *TMobile*, dan *NVidia*. Pada saat perilisan perdana *Android* pada tanggal 5 november 2007, *Android* bersama *Open Handset Alliance* menyatakan mendukung pengembangan standar terbuka pada perangkat seluler.

Di lain pihak, *Google* merilis kode-kode *Android* dibawah lisensi *Apache*, sebuah lisensi perangkat lunak dan standar terbuka perangkat seluler. Terdapat dua jenis distributor sistem operasi *Android*. Pertama yang dapat dukungan penuh dari *Google* atau *Google Mail Service (GMS)* dan kedua adalah yang benar-benar bebas distribusinya tanpa dukungan langsung dari *Google* atau dikenal sebagai *Open Handset Distribution (DHD)*.

Berikut ini merupakan perkembangan *Android* dari versi ke versi :

1. *Android* versi 1.1

Pada 9 Maret 2009, *Google* merilis *Android* versi 1.1. *Android* versi ini dilengkapi dengan pembaruan estetis pada aplikasi, jam *alarm*, *voice search* (pencarian suara), pengiriman pesan dengan *Gmail*, dan pemberitahuan *email*.

2. *Android* versi 1.5 (*Cupcake*)

Pada pertengahan Mei 2009, *Google* kembali merilis telepon seluler dengan menggunakan *Android* dan *SDK (Software Development Kit)* dengan versi 1.5 (*Cupcake*). Terdapat beberapa pembaruan termasuk juga penambahan beberapa fitur dalam seluler versi ini yakni kemampuan merekam dan menonton *video* dengan modus kamera, mengunggah *video* ke *Youtube* dan gambar ke *Picasa* langsung dari telepon, dukungan *Bluetooth A2DP*, kemampuan terhubung secara otomatis ke *headset bluetooth*, animasi layar, dan *keyboard* pada layar yang dapat disesuaikan dengan sistem.

3. *Android* versi 1.6 (*Donut*)

Donut (versi 1.6) dirilis pada September dengan menampilkan proses pencarian yang lebih baik dibanding sebelumnya, penggunaan baterai indikator dan kontrol applet VPN. Fitur lainnya adalah galeri yang memungkinkan pengguna untuk memilih foto yang akan dihapus, kamera, *camcorder* dan galeri yang diintegrasikan, CDMA / EVDO, 802.1x, VPN, *Gestures*, dan *Text-to-speech engine*, kemampuan dial kontak, teknologi *text to change speech* (tidak tersedia pada semua ponsel), pengadaan resolusi VWGA.

4. *Android* versi 2.0/2.1 (*Eclair*)

Pada 3 Desember 2009 kembali diluncurkan ponsel *Android* dengan versi 2.0/2.1 (*Eclair*), perubahan yang dilakukan adalah pengoptimalan *hardware*, peningkatan Google Maps 3.1.2, perubahan UI dengan *browser* baru dan dukungan HTML5, daftar kontak yang baru, dukungan *flash* untuk kamera 3,2 MP, *digital Zoom*, dan *Bluetooth* 2.1. Dengan semakin berkembangnya dan semakin bertambahnya jumlah handset *Android*, semakin banyak pihak ketiga yang berminat untuk menyalurkan aplikasi mereka kepada sistem operasi *Android*. Aplikasi terkenal yang diubah ke dalam sistem operasi *Android* adalah *Shazam*, *Backgrounds*, dan *WeatherBug*. Sistem operasi *Android* dalam situs Internet juga dianggap penting untuk menciptakan aplikasi *Android* asli, contohnya oleh *MySpace* dan *Facebook*.

5. *Android* versi 2.2 (*Froyo: Frozen Yoghurt*)

Pada 20 Mei 2010, *Android* versi 2.2 (*Froyo*) diluncurkan. Perubahan-perubahan umumnya terhadap versi-versi sebelumnya antara lain dukungan Adobe Flash 10.1, kecepatan kinerja dan aplikasi 2 sampai 5 kali lebih cepat, integrasi V8 JavaScript engine yang dipakai *Google Chrome* yang mempercepat kemampuan *rendering* pada *browser*, pemasangan aplikasi dalam SD Card, kemampuan *WiFi Hotspot portabel*, dan kemampuan *auto update* dalam aplikasi *Android Market*.

6. *Android* versi 2.3 (*Gingerbread*)

Pada 6 Desember 2010, *Android* versi 2.3 (*Gingerbread*) diluncurkan. Perubahan-perubahan umum yang didapat dari *Android* versi ini antara lain peningkatan kemampuan permainan (*gaming*), peningkatan fungsi copy paste, layar antar muka (*User Interface*) didesain ulang, dukungan format video VP8 dan WebM, efek audio baru (*reverb, equalization, headphone virtualization, dan bass boost*), dukungan kemampuan *Near Field Communication* (NFC), dan dukungan jumlah kamera yang lebih dari satu.

7. *Android* versi 3.0/3.1 (*Honeycomb*)

Android Honeycomb dirancang khusus untuk tablet. *Android* versi ini mendukung ukuran layar yang lebih besar. *User Interface* pada *Honeycomb* juga berbeda karena sudah didesain untuk tablet. *Honeycomb* juga mendukung multi prosesor dan juga akselerasi perangkat keras (*hardware*) untuk grafis. Tablet pertama yang dibuat dengan menjalankan *Honeycomb* adalah *Motorola Xoom*. Perangkat tablet dengan platform *Android* 3.0 akan segera hadir di Indonesia. Perangkat tersebut bernama *Eee Pad Transformer* produksi dari Asus. Rencana masuk pasar Indonesia pada Mei 2011.

8. *Android* versi 4.0 (ICS: *Ice Cream Sandwich*)

Diumumkan pada tanggal 19 Oktober 2011, membawa fitur *Honeycomb* untuk smartphone dan menambahkan fitur baru termasuk membuka kunci dengan pengenalan wajah, jaringan data pemantauan penggunaan dan kontrol, terpadu kontak jaringan sosial, perangkat tambahan fotografi, mencari *email* secara *offline*, dan berbagi informasi dengan menggunakan NFC. Ponsel pertama yang menggunakan sistem operasi ini adalah Samsung Galxy Nexus.

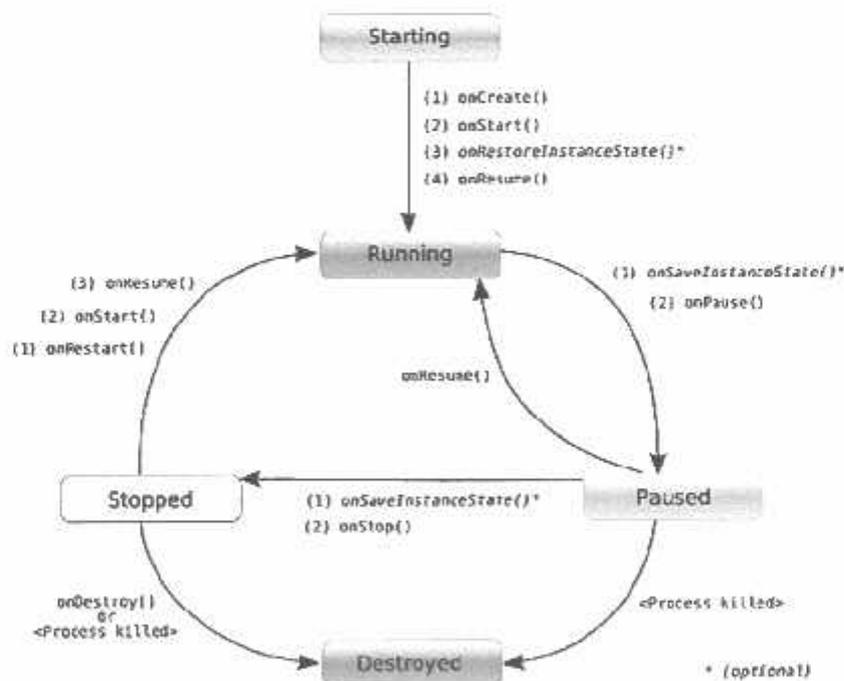
9. *Android* versi 4.1 (*Jelly Bean*)

Android Jelly Bean yang diluncurkan pada acara Google I/O lalu membawa sejumlah keunggulan dan fitur baru. Penambahan baru diantaranya meningkatkan input *keyboard*, desain baru fitur pencarian, UI yang baru dan pencarian melalui *Voice Search* yang lebih cepat. Tak

ketinggalan *Google Now* juga menjadi bagian yang diperbarui. *Google Now* memberikan informasi yang tepat pada waktu yang tepat pula. Salah satu kemampuannya adalah dapat mengetahui informasi cuaca, lalu-lintas, ataupun hasil pertandingan olahraga. Sistem operasi *Android Jelly Bean 4.1* muncul pertama kali dalam produk tablet Asus, yakni *Google Nexus 7*.^[2]

2.6 Konsep Umum *Android*

Perangkat berbasis *android* hanya mempunyai satu layar *foreground*. Normalnya saat menghidupkan *android*, yang pertama dilihat adalah *home*. Kemudian jika ingin menjalankan sebuah aplikasi catur, *User Interfacenya* (UI) akan menumpuk diatas layar sebelumnya (*home*). Kemudian bila melihat help-nya catur, maka UI help akan menimpa UI sebelumnya (catur), begitu seterusnya. Semua proses diatas direkam di *application stack* oleh sistem *Activity Manager*. Menekan tombol back hanya kembali ke halaman sebelumnya, analoginya mirip dengan browser dimana ketika ingin meng-klik tombol back browser akan kembali menampilkan halaman sebelumnya. Setiap *User Interface* diwakili oleh kelas *Activity* (*Activity class*). Adapun siklus *activity android* ditunjukkan pada gambar 2.5.



Gambar 2.5 Siklus *Activity Android*

Selama siklus ini berjalan, *activity* bisa mempunyai lebih dari 2 status. Kita tidak bisa mengontrol setiap status karena semuanya sudah ditangani oleh sistem. Namun kita akan mendapat pesan saat terjadi perubahan status melalui *Method onXX()*. Berikut penjelasan tentang status *activity* akan dijelaskan pada tabel 2.1.

Tabel 2.1 Status *Activity Android*

No.	Status	Keterangan
1.	<code>onCreate(Bundle)</code>	Dipanggil saat pertama kali aplikasi dijalankan. Kita dapat menggunakan ini untuk deklarasi variabel atau membuat <i>User Interface</i> .
2.	<code>onStart()</code>	Mengindikasikan <i>activity</i> yang ditampilkan ke pengguna (<i>User</i>).
3.	<code>onResume()</code>	Dipanggil saat aplikasi kita mulai berinteraksi dengan pengguna. Disini sangat cocok untuk meletakkan animasi ataupun musik.
4.	<code>onPause()</code>	Dipanggil saat aplikasi yang Kita jalankan kembali ke halaman sebelumnya atau biasanya karena ada <i>activity</i> baru yang dijalankan. Disini cocok untuk meletakkan algoritma penyimpanan (<i>save</i>).
5.	<code>onStop()</code>	Diapnggil saat aplikasi Kita berjalan di belakang layar dalam waktu cukup lama.
6.	<code>onRestart()</code>	<i>Activity</i> kembali menampilkan <i>User Interface</i> setelah status <i>stop</i> .
7.	<code>onDestroy()</code>	Dipanggil saat aplikasi benar-benar berhenti.
8.	<code>onSaveInstanceState(Bundle)</code>	<i>Method</i> ini mengijinkan <i>activity</i> untuk menyimpan setiap status intance.

		Misalnya dalam mengedit <i>teks</i> , kursor bergerak dari kiri ke kanan.
9.	<code>onRestoreInstanceState</code> (<i>Bundle</i>)	Diapanggil saat <i>activity</i> kembali menginisialisasi dari status sebelumnya yang disimpan oleh <code>onSaveInstanceState</code> (<i>Bundle</i>).

Adapun beberapa komponen yang sangat penting dalam pengembangan *android* ini adalah sebagai berikut:

1. *Activity*

Normalnya setiap *activity* menampilkan satu buah *User Interface* kepada pengguna. Misalnya sebuah *activity* menampilkan daftar menu minuman, kemudian pengguna dapat memilih satu jenis minuman. Contoh lainnya pada aplikasi sms, dimana satu *activity* digunakan untuk menulis pesan, *activity* berikutnya untuk menampilkan nomor kontak tujuan, atau *activity* lainnya digunakan untuk menampilkan pesan-pesan lama. Meskipun *activity-activity* diatas terdapat dalam satu aplikasi sms, namun masing-masing *activity* berdiri sendiri. Untuk pindah dari satu *activity* ke *activity* lainnya dapat melakukan suatu *event* misalnya tombol diklik atau melalui trigger tertentu.

2. *Service*

Service tidak memiliki *User Interface*, namun berjalan di belakang layar. Misalnya *music player*, sebuah *activity* digunakan untuk memilih lagu kemudian di-*play*. Agar *music player* bisa berjalan dibelakang aplikasi lain maka harus menggunakan *service*.

3. *Intens*

Intens adalah mekanisme untuk menggambarkan sebuah *action* secara detail seperti bagaimana cara mengambil sebuah foto.

4. *Content Providers*

Menyediakan cara untuk mengakses data yang dibutuhkan oleh suatu *activity*, misalnya kita menggunakan aplikasi berbasis peta (MAP). *Activity* membutuhkan cara untuk mengakses data kontak untuk prosedur navigasi. Disinilah peran *content providers*.

5. *Resources*

Resource digunakan untuk menyimpan *file-file non-coding* yang diperlukan pada sebuah aplikasi misalnya *file icon*, *file gambar*, *file audio*, *file video* atau yang lain. Gambar berformat JPG atau PNG sebuah aplikasi biasanya disimpan dalam folder *res/drawable*, *icon* aplikasi disimpan dalam *res/drawable-ldpi* dan file audio disimpan dalam folder *res/raw*. File XML untuk membentuk *User Interface* disimpan dalam folder *res/layout*.^[2]

2.7 Eclipse

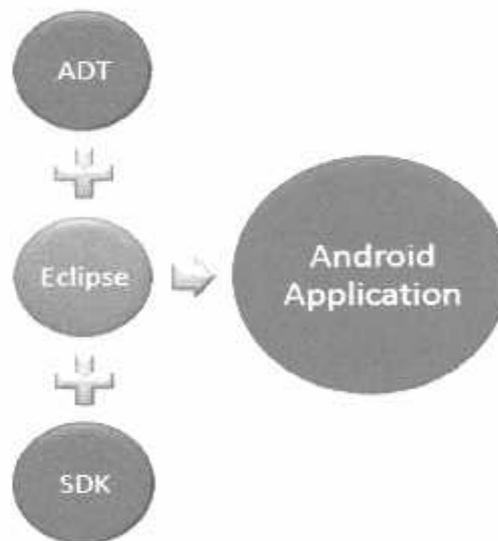
Eclipse adalah sebuah IDE (*Integrated Development Environment*) untuk mengembangkan perangkat lunak dan dapat dijalankan di semua Platform (*platform-independent*).

Berikut ini adalah sifat dari Eclipse:

1. *Multi-platform*: Target sistem operasi Eclipse adalah Microsoft Windows, Linux, Solaris, AIX, HP-UX dan Mac OS X.
 2. *Mult-language*: Eclipse dikembangkan dengan bahasa pemrograman *Java*, akan tetapi *Eclipse* mendukung pengembangan aplikasi berbasis bahasa pemrograman lainnya, seperti C/C++, Cobol, Python, Perl, PHP, dan lain sebagainya.
 3. *Multi-role*: Selain sebagai IDE untuk pengembangan aplikasi, *Eclipse* pun bisa digunakan untuk aktivitas dalam siklus pengembangan perangkat lunak, seperti dokumentasi, *test* perangkat lunak, pengembangan *web*, dan lain sebagainya.
-

Eclipse pada saat ini merupakan salah satu IDE favorit dikarenakan gratis dan *open source*, yang berarti setiap orang boleh melihat kode pemrograman perangkat lunak ini. Selain itu, kelebihan dari *Eclipse* yang membuatnya populer adalah kemampuannya untuk dapat dikembangkan oleh pengguna dengan komponen yang dinamakan *plug-in*.

Secara *default editor* ini belum bisa dipakai untuk mendevlop *android*. Agar bisa digunakan untuk membuat aplikasi *android* maka harus diinstall *plugin* dulu namanya ADT (*Android Development Tools*). Setelah terinstal, maka *eclipse* sudah siap digunakan, hanya saja belum dilengkapi library dan emulator. Untuk memenuhinya, maka perlu diinstal SDK *Android (Software Development Kit)*. Adapun kebutuhan *software* ditunjukkan pada gambar 2.6.^[2]



Gambar 2.6 Kebutuhan *Software*

2.8 SDK (*Software Development Kit*)

Android-SDK merupakan *tools* bagi para *programmer* yang ingin mengembangkan aplikasi berbasis google *android*. *Android SDK* mencakup seperangkat alat pengembangan yang komprehensif. *Android SDK* terdiri dari *debugger*, *libraries*, *handset emulator*, dokumentasi, contoh kode, dan tutorial. Saat ini *Android* sudah mendukung arsitektur x86 pada Linux (distribusi Linux apapun untuk desktop modern), Mac OS X 10.4.8 atau lebih, Windows XP atau Vista. Persyaratan mencakup JDK, Apache Ant dan Python 2.2 atau yang lebih

baru. IDE yang didukung secara resmi adalah Eclipse 3.2 atau lebih dengan menggunakan *plugin Android Development Tools (ADT)*, dengan ini pengembang dapat menggunakan *teks editor* untuk mengedit *file Java* dan *XML* serta menggunakan peralatan *command line* untuk menciptakan, membangun, melakukan *debug* aplikasi *Android* dan pengendalian perangkat *Android* (misalnya, *reboot*, menginstal paket perangkat lunak dengan jarak jauh).

Android SDK telah dirilis pada tanggal 12 November 2007. Dan pada tanggal 15 Juli 2008 tim *Android Developer Challenge* sengaja mengirimkan *email* ke semua pendatang di *Android Developer Challenge* untuk mengumumkan bahwa rilis SDK terbaru telah tersedia pada halaman *download* pribadi. *Email* tersebut juga ditujukan kepada pemenang *Android Developer Challenge* putaran pertama. Sebuah pernyataan bahwa *Google* telah menyediakan rilis SDK terbaru untuk beberapa pengembang dan bukan untuk orang lain.

Pada tanggal 18 Agustus 2008, *Android SDK 0.9 beta* dirilis. Rilis ini menyediakan API yang diperbarui dan diperluas, perbaikan pada alat-alat pengembangan dan desain terbaru untuk layar awal. Petunjuk untuk meng-*upgrade* SDK sudah tersedia pada rilis sebelumnya. Pada tanggal 23 September 2008, *Android 1.0 SDK* telah dirilis. Pada tanggal 9 Maret 2009, *Google* merilis versi 1.1 untuk telepon seluler *Android*. Rilis terbaru tersebut termasuk dukungan untuk pencarian dengan suara, harga aplikasi, perbaikan jam *alarm*, perbaikan pengiriman *gmail*, perbaikan surat pemberitahuan dan peta.

Pada pertengahan Mei 2009, *Google* merilis versi 1.5 (*Cupcake*) pada sistem operasi *Android* dan SDK. Pembaruan ini termasuk banyak fitur baru seperti perekaman *video*, dukungan untuk *bluetooth*, sistem *keyboard* pada layar dan pengenalan suara. Rilis ini juga membuka *AppWidget framework* kepada para pengembang yang memungkinkan orang untuk membuat *widget* sendiri pada halaman *home*. Pada September 2009 versi 1.6 (*Donut*) dirilis yang menampilkan hasil pencarian yang lebih baik dan penggunaan indikator baterai.

Aplikasi *Android* dipaketkan ke dalam *format .apk* dan disimpan pada *folder /data/app*. Pengguna dapat menjalankan perintah *adb root* untuk mengakses *folder* tersebut karena *root* memiliki izin untuk mengakses *folder* tersebut.^[2]

2.9 JDK (*Java Development Kit*)

Java Development Kit (JDK) adalah *Sun Microsystems* produk ditujukan untuk pengembang *Java*. Sejak diperkenalkannya *Java*, telah jauh SDK *Java* yang paling banyak digunakan. Pada tanggal 17 November 2006, Sun mengumumkan bahwa akan dirilis di bawah GNU *General Public License* (GPL), sehingga membuat perangkat lunak bebas. Hal ini terjadi sebagian besar pada tanggal 8 Mei 2007 Sun kontribusi kode sumber untuk *JDK Open*.^[2]

2.10 ADT (*Android Development Tools*)

Android Development Tools (ADT) adalah *plugin* untuk *Eclipse* yang didesain untuk pengembangan aplikasi *Android*. ADT memungkinkan *Eclipse* untuk digunakan dalam membuat aplikasi *Android* baru, membuat *User Interface*, menambahkan komponen berdasarkan *framework API Android*, *debug* aplikasi, dan pemaketan aplikasi *Android*.^[2]

2.11 AVD (*Android Virtual Device*)

Android Virtual Devices (AVD) adalah konfigurasi dari *emulator* sehingga kita dapat menjalankan perangkat *Android* sesuai model yang dipilih. Dimana setiap AVD terdiri dari :

1. Sebuah profil perangkat keras yang dapat mengatur *opsi* untuk menentukan fitur *hardware emulator*. Misal, kita dapat menentukan apakah menggunakan perangkat kamera, apakah menggunakan *keyboard QWERTY* fisik atau tidak, berapa banyak memori internal, dan lain-lain.
 2. Sebuah pemetaan versi *Android* dapat menentukan versi dari *platform Android* yang akan berjalan pada *emulator*.
 3. Pilihan lainnya. kita dapat menentukan *skin* yang ingin digunakan pada *emulator*, yang memungkinkan untuk menentukan dimensi layar, tampilan, dan sebagainya. Dapat menentukan *SD Card virtual* untuk digunakan pada *emulator*.^[2]
-

yang diciptakan. Pada saat yang sama, implementasi WWW dan Internet sedang mengalami perkembangan pesat. Di lain pihak, anggota dari proyek *Green* juga menyadari bahwa *Java* dapat digunakan pada pemrograman internet, sehingga penerapan selanjutnya mengarah menjadi teknologi yang berperan di *web*. Adapun arsitektur program *java* ditunjukkan pada gambar 2.7.^[5]

Bahasa/Alat pengembangan	Arsitektur Program			
	Modul Web Server	Scripting Web Server	Modul Web Browser	Scripting Web Browser
Java	servlet	JSP	Applet	Javascript
C++	CGI exe		ActiveX*	
Perl	CGI script			
Python	CGI script			
PHP		PHP script		
Visual Basic		ASP*	ActiveX*	VB Script*

*) Hanya di landasan Windows, tidak bisa di linux.

Gambar 2.7 Arsitektur Program *Java*

2.13 Database

Database adalah koleksi data item yang saling terkait terkelola sebagai satu unit. *Database* memiliki kapabilitas unik unik untuk menyajikan informasi ke banyak pengguna sesuai dengan permintaan masing-masing. Pengguna di sini bisa saja manusia atau aplikasi yang masuk secara sah ke *Database* untuk menyimpan atau mendapatkan data. Aplikasi tersebut adalah program yang dirancang untuk menyelesaikan tugas-tugas bisnis tertentu seperti sistem pemesanan barang, sistem penggajian, sistem akuntansi.

Database memungkinkan setiap pengguna (manusia atau aplikasi) mendapatkan informasi hasil pengolahan data yang berbeda satu sama lain. Yang satu minta laporan pendapatan bulan tertentu sementara yang lain mencetak transaksi hari ini, pengguna lain melakukan pencarian terhadap data tertentu. Masing-masing menggunakan data dari *Database* yang sama. Setiap perubahan data yang dilakukan oleh salah satu, akan juga ditemukan oleh yang lain.

Secara umum model *Database* dibagi menjadi dua yaitu :

1. *Flat Files*

Flat files adalah *file* sistem operasi yang *record* dalam filenya tidak berisi informasi tentang struktur *file* atau hubungan antar *record* yang dikomunikasikan ke aplikasi yang menggunakannya. *Flat file* bukan *Database* karena tidak masuk dalam kriteria di atas yang sudah dibahas.

2. *Hierarchical Model*

Model ini menyusun *record-record* dalam suatu hirarki seperti struktur organisasi. Setiap *file* dari *flat file* menjadi tipe *record* atau *node* dalam hirarki, dan untuk memudahkan disebut *record*. *Record-record* terhubung melalui pointer yang berisi alamat *record* yang terkait. *Pointer* memberi tahu sistem komputer dimana *record* tersebut tersimpan secara fisik, seperti alamat yang mengacu ke suatu gedung tertentu di dunia nyata. Setiap pointer membuat hubungan *parent-child*, yang juga disebut *one to-many relationship*. Satu *parent* bisa punya banyak *child*, tapi setiap *child* memiliki 1 *parent*. Misal 1 *Manager* memimpin banyak karyawan, tapi 1 karyawan memiliki 1 *Manager*. Model ini menyebabkan masalah dalam kasus Northwind. Satu *order* tidak hanya punya satu *customer*, tapi juga punya satu karyawan yang menerima *order* tersebut

3. *Relational Model*

Model ini berusaha mengatasi masalah pada model sebelumnya yang tidak fleksibel. Model relational *Database* mampu menghubungkan *record-record* yang dibutuhkan. Model ini dibangun sedemikian rupa sehingga permintaan atau *query* dapat bekerja dengan sekumpulan data, tidak dengan satu *record* seperti pada model hierarchical. Model relational menyajikan data dalam tabel dua dimensi, seperti *spreadsheet* hanya saja tidak dipresentasikan dalam tabel seperti *spreadsheet* tetapi menggunakan kombinasi atau *joining* relational tabel untuk menghasilkan *view* atau hasil tampilan.

4. *Object-Oriented Model*

Model ini berusaha menjawab kesulitan RDBMS (*Relational Database Management Systems*) untuk mengatasi tipe data yang kompleks seperti *file* citra, *file* gambar dan *file audio-video*. Ini semua karena adanya Internet dan

WWW yang memungkinkan pengiriman data yang kompleks itu. *Object* adalah pengelompokan logis dari data yang berelasi dan program logis yang mewakili hal nyata di dunia seperti *customer*, *employee*, *order*, *product*. Item data seperti : *customer ID*, *customer name* disebut variabel di model OO dan disimpan dalam tiap objek. Variabel-variabel hanya bisa diakses melalui *Method*. Sementara istilah *Method* adalah aplikasi program logis yang menggunakan *object* tertentu dan menyediakan fungsi tertentu seperti *check customer credit card limit*, *update alamat customer*.^[6]

2.14 SQLite

SQLite adalah *Database* yang bisa dibangun di *android*. *SQLite* memiliki fitur relasional *Database*, hampir sama dengan SQL pada desktop hanya saja *SQLite* membutuhkan memori yang sedikit. *SQLite* terdapat pada semua perangkat *android*, kita cukup mendefinisikan perintah SQL untuk meng-*create* atau meng-*update Database*, selanjutnya *system* pada *android* akan menangani hal-hal yang berhubungan dengan *Database*. *SQLite Database* otomatis akan tersimpan didalam path *data/data/nama_package/Database/nama_Database*.

Dalam pembuatan *Database SQLite* harus mengetahui 3 kelas penting antara lain :

1. *SQLiteDatabase*

SQLiteDatabase merupakan kelas yang mempunyai *Method* seperti :

1. *Insert()* untuk menambahkan baris ke *Database*
2. *Update()* untuk memperbarui baris pada *Database*
3. *Delete()* untuk menghapus baris pada *Database*
4. *execSQL()* untuk mengeksekusi sintak SQL.

2. *SQLiteOpenHelper*

SQLiteOpenHelper adalah subclass yang memiliki beberapa *Method* seperti :

1. *onCreate()* dijalankan jika sebelumnya belum ada *Database*
2. *onUpgrade()* dijalankan jika sebelumnya sudah ditemukan *Database* yang sama namun beda versi. *Method* ini bisa dimanfaatkan untuk mengubah skema *Database*.

3. `onOpen()` dijalankan jika *Database* sudah dalam keadaan open
4. `getWritableDatabase()` memanggil *Database* agar bisa dimasuki data
5. `getReadableDatabase()` memanggil *Database* agar bisa membaca datanya

3. *Cursor*

Setiap *query* yang dieksekusi pasti membawa nilai kembalian atau *feedback*. *Feedback* yang dihasilkan *query* ini disebut *cursor*. Jadi dengan kata lain, *cursor* merepresentasikan hasil *query* yang dieksekusi pada baris dan kolom tertentu. Adapun analogi *cursor* pada *SQLite database* ditunjukkan pada gambar 2.8.

[fitri,masakan]	→	<table border="1" style="border-collapse: collapse; text-align: center;"> <thead> <tr> <th style="padding: 2px 10px;">nama</th> <th style="padding: 2px 10px;">hobi</th> </tr> </thead> <tbody> <tr> <td style="padding: 2px 10px;">Ayib</td> <td style="padding: 2px 10px;">makan</td> </tr> <tr> <td style="padding: 2px 10px;">fitri</td> <td style="padding: 2px 10px;">masak</td> </tr> <tr> <td style="padding: 2px 10px;">ridhani</td> <td style="padding: 2px 10px;">baca</td> </tr> </tbody> </table>	nama	hobi	Ayib	makan	fitri	masak	ridhani	baca
nama	hobi									
Ayib	makan									
fitri	masak									
ridhani	baca									

Gambar 2.8 Analogi *cursor* pada *SQLite Database*

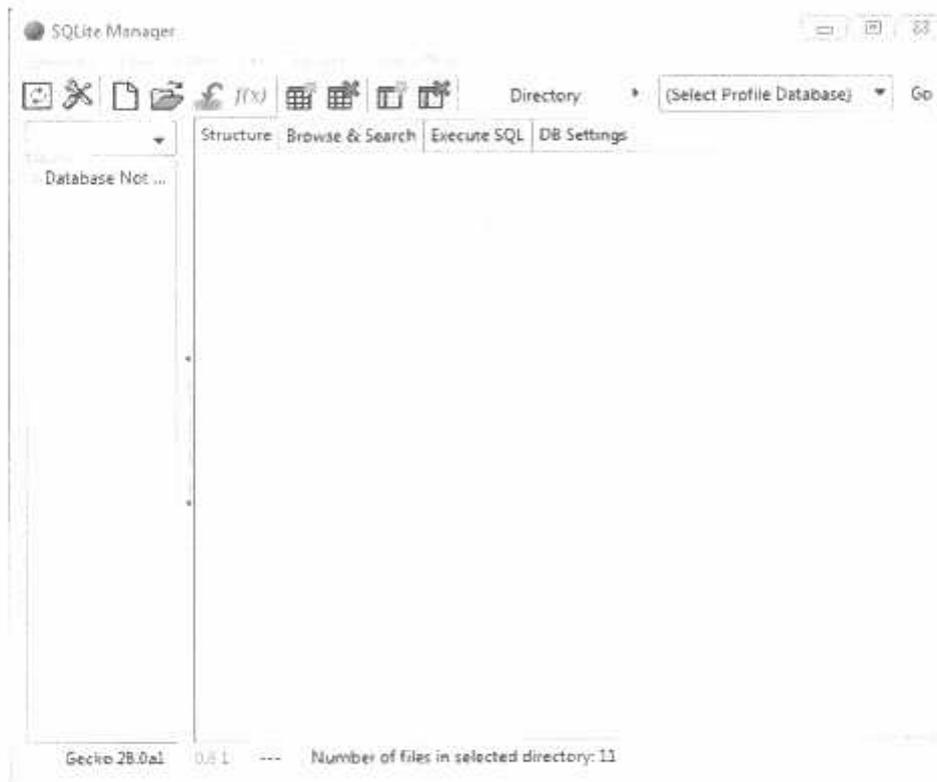
Cursor diwakili dengan `tKamu` panah berada pada posisi baris ke 2 dengan membawa data [fitri,masakan]. Berikut ini beberapa *Method* yang digunakan dalam *SQLite Database*.

1. `moveToFirst()` untuk pindah ke baris pertama.
2. `isAfterLast()` akan mengirim pesan balik jika posisi *cursor* sudah berada di baris terakhir.
3. `getLong()` untuk mengambil data pada kolom yang mempunyai tipe data long.
4. `getString()` untuk mengambil data pada kolom yang mempunyai tipe data String. ^[6]

2.15 *SQLite Manager*

SQLite Manager adalah sebuah program yang sangat membantu programmer dalam mengelola *Database SQLite* seperti membuat *table*, *index* dan menjalankan perintah-perintah SQL. Program ini sangat mudah digunakan dan

betul-betul bermanfaat untuk berbagai keperluan praktis. Satu hal lagi yang tidak kalah pentingnya: program ini gratis. Selain *SQLite Manager*, ada beberapa program lain yang tujuannya serupa namun saya menganjurkan program ini karena ia yang paling populer digunakan saat ini. *SQLite Manager* adalah sebuah program *add-on* pada program *Mozilla Firefox* sehingga pastikan pada komputer Anda telah terinstal program *browser* tersebut sebelumnya. Untuk menginstal *SQLite Manager*, silakan mengunduh *add-on*-nya pada alamat berikut ini dan ikuti instruksinya. Adapun *SQLite manager* ditunjukkan pada gambar 2.9.¹⁶¹



Gambar 2.9 *SQLite Manager*

BAB III ANALISIS DAN PERANCANGAN SISTEM

3.1 Analisis Sistem

Analisis sistem adalah teknik pemecahan masalah yang menguraikan bagian-bagian komponen dengan mempelajari seberapa bagus bagian-bagian komponen tersebut bekerja dan berinteraksi untuk mencapai tujuan mereka. Analisis sistem merupakan tahapan paling awal dari pengembangan sistem yang menjadi pondasi menentukan keberhasilan sistem informasi yang dihasilkan nantinya.

3.1.1 Analisis Prosedur yang Berjalan

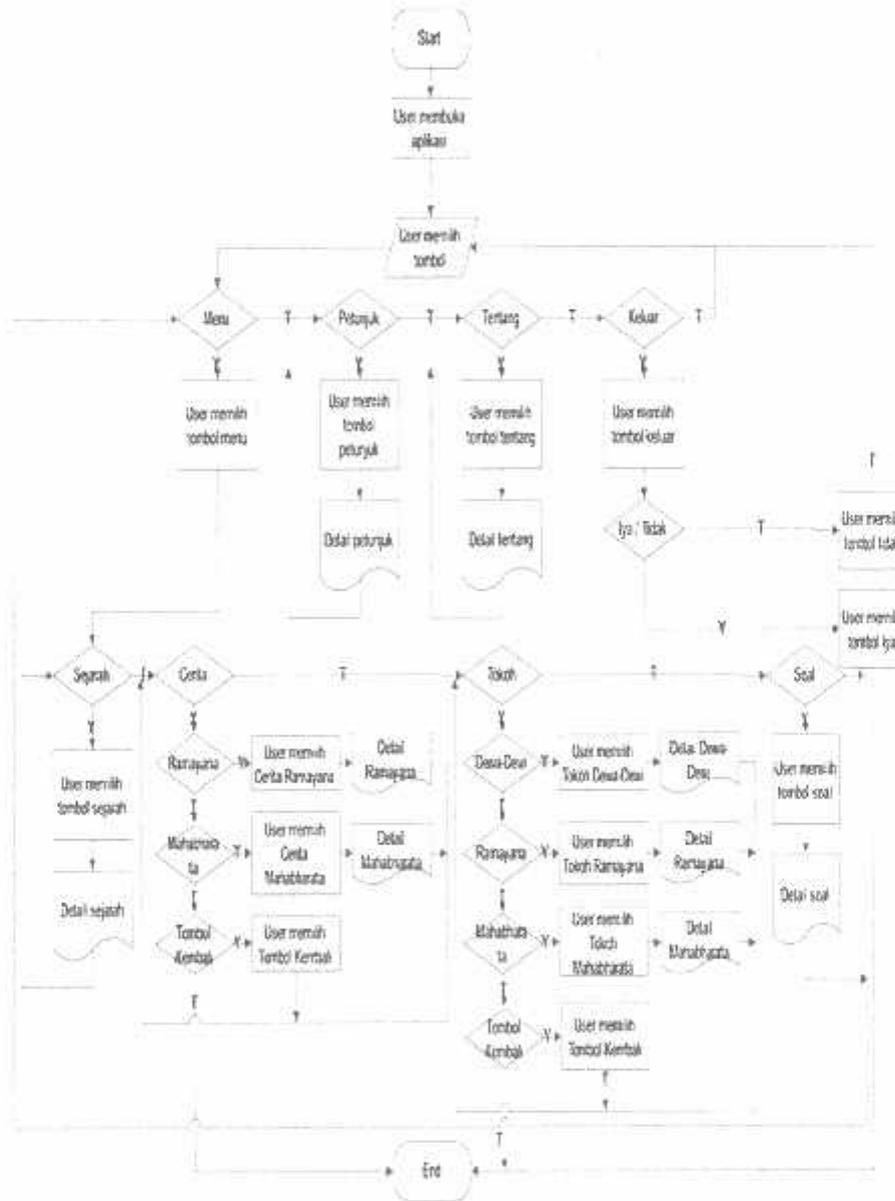
Analisis prosedur yang berjalan merupakan urutan kegiatan dari setiap tahapan yang menerangkan mengenai proses – proses yang dikerjakan, siapa saja yang mengerjakan, bagaimana proses tersebut dikerjakan dan *data* apa saja yang dibutuhkan oleh sistem. Berikut merupakan prosedur pemakaian aplikasi wayang ini :

1. *User* membuka aplikasi wayang yang ada di *smartphone android*.
2. *User* memilih tombol yang disediakan oleh aplikasi antara lain tombol *menu* yang berisi *menu* pada aplikasi wayang, tombol *tentang* yang berisi informasi tentang aplikasi, tombol *sound* yang berfungsi untuk mematikan dan menyalakan *backsound* , dan tombol keluar untuk keluar dari aplikasi ini.
3. Jika *user* memilih *menu*, maka pada aplikasi ini akan menuju ke *Interface menu* pilihan lain.
4. Pada *Interface* ini disediakan *gallery picture* beberapa wayang dan 4 macam tombol pilihan.
5. *User* disuruh untuk memilih 4 macam tombol yaitu tombol sejarah untuk mengetahui isi sejarah tentang perkembangan wayang, tombol cerita untuk mengetahui cerita ramayana dan cerita mahabharata, tombol tokoh untuk mengetahui berbagai macam tokoh dewa dan dewi, tokoh ramayana, dan

tokoh mahabharata, dan tombol soal yang memungkinkan *user* untuk belajar menjawab soal yang diberikan oleh sistem aplikasi ini.

6. Jika *user* memilih tombol tokoh maka, sistem akan menuju ke *Interface* tokoh yang didalamnya disediakan berbagai macam tokoh dan fitur pencarian tokoh.

Adapun bentuk *flowchart* untuk prosedur yang berjalan bisa dilihat pada gambar 3.1.



Gambar 3.1 *Flowchart* Analisis yang Berjalan

3.1.2 Analisis Kebutuhan Data

Sistem yang dibangun adalah sistem pembelajaran dengan menggunakan aplikasi *mobile android* tentang tokoh pewayangan. Pada aplikasi *mobile* ini digunakan untuk mengetahui informasi tentang sejarah wayang, tokoh dan cerita pewayangan yang yang diberi fitur tambahan berupa soal tes.

3.1.3 Analisis Kebutuhan Sistem

Dalam pembuatan aplikasi wayang berbasis *android* ini memerlukan spesifikasi perangkat keras yang digunakan. Adapun perangkat keras yang digunakan dalam pembuatan aplikasi *mobile android* ini adalah sebagai berikut :

1. RAM 4 GB DDR 3
2. Hardisk 500 GB
3. 14" WXGA LED, Max. Resolution 1366 x 768, Clear SuperView LED
4. Prosesor Intel(R) Core(TM) i5 CPU
5. VGA Intel HD Graphics

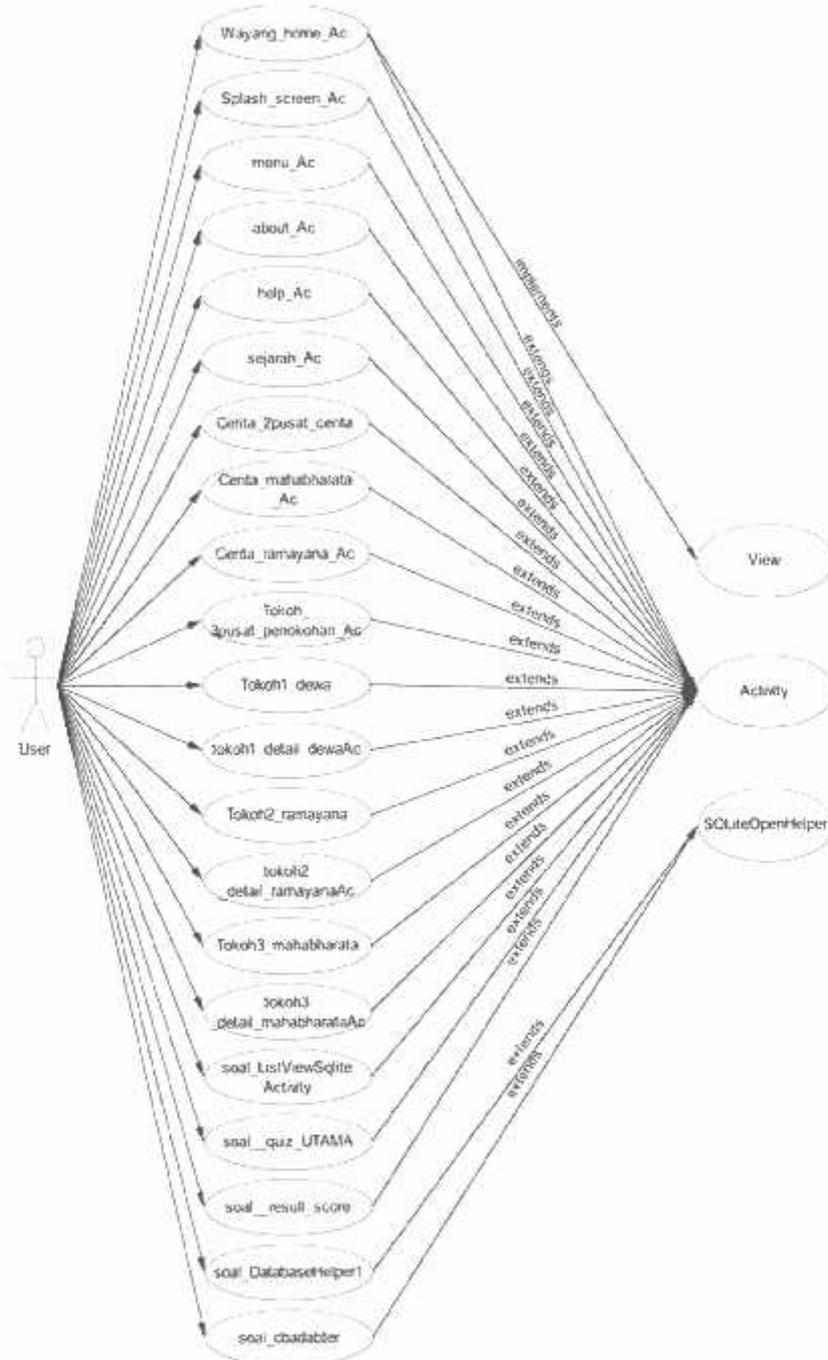
3.1.4 Analisis Kebutuhan Perangkat Lunak

Perangkat lunak yang digunakan untuk membangun *mobile android* tokoh pewayanga ini adalah sebagai berikut :

1. *Windows 7 Ultimate 64 Bit* sebagai sistem operasi.
 2. IDE *Eclipse Indigo*, perangkat lunak yang digunakan untuk membangun aplikasi *mobile*.
 3. *Adobe Photoshop CS5*, perangkat lunak yang digunakan untuk membuat logo wayang, gambar, *button*, dan lain - lain untuk desain perangkat lunak.
 4. *Java Development Kit (JDK)*, perangkat lunak yang digunakan untuk kompilasi kode - kode program.
 5. *Android Software Development Kit (SDK)*, perangkat lunak yang digunakan untuk pengembang aplikasi *android*.
 6. *Android Virtual Device (AVD)*, perangkat lunak yang digunakan sebagai emulator.
-

3.2 Use Case Diagram

Diagram *use case* merupakan suatu gambaran model fungsional dari sebuah sistem yang menggunakan fungsi – fungsi dari sebuah sistem. Adapun *use case* diagram ini ditunjukkan pada gambar 3.2.



Gambar 3.2 Use Case Diagram

3.3 Perancangan Sistem

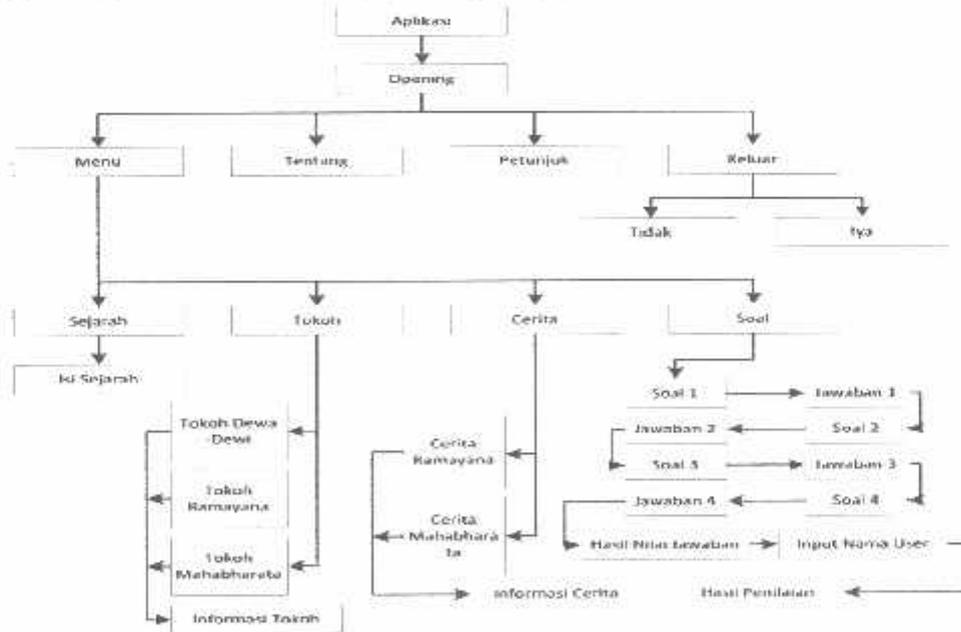
Rancangan sistem secara umum dilakukan dengan maksud untuk memberikan gambaran umum tentang sistem yang baru atau sistem yang akan diusulkan. Rancangan ini mengidentifikasi komponen-komponen sistem informasi yang dirancang secara rinci. Adapun proses interaksi sistem dengan *user* ditunjukkan pada gambar 3.3.



Gambar 3.3 Proses Interaksi Sistem Dengan *User*

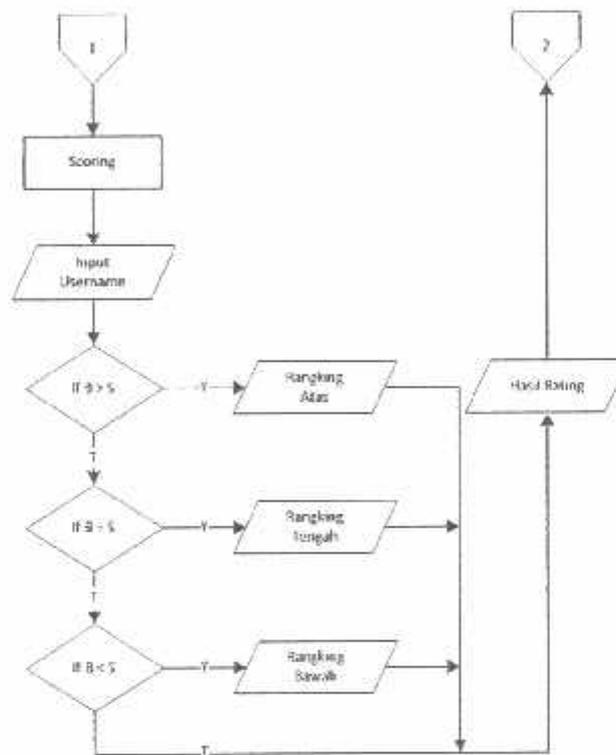
3.3.1 Rancangan Struktur Navigasi

Pembuatan struktur navigasi sangat diperlukan karena memberikan kemampuan bagi *user* untuk mengakses *Data* dengan cepat. Berikut adalah rancangan tampilan struktur navigasi *user* pada gambar 3.4.



Gambar 3.4 Struktur Navigasi *User*

Pada gambar 3.5 diatas menjelaskan tentang rancangan proses – proses yang dilakukan oleh sistem aplikasi *mobile* wayang. Ketika *user* pertama kali membuka aplikasi ini, maka *user* akan ditunjukan ke *opening* aplikasi berupa *splash screen* yang kemudian secara otomatis sistem melanjutkan eksekusinya ke *Interface menu home* yang berisikan empat macam *menu* proses yaitu *menu*, *tentang*, *sound*, dan keluar. Jika *user* memilih *menu*, maka sistem menuju ke *Interface menu* utama yang didalamnya berisikan *gallery* foto, sejarah, tokoh, cerita, dan soal tes. Adapun *flowchart* *user* aplikasi *mobile* untuk proses *scoring* ditunjukkan pada gambar 3.6.



Gambar 3.6 *Flowchart* *User* Aplikasi *Mobile* Untuk Proses *Scoring*

Pada gambar 3.6 diatas menjelaskan tentang ketika *user* memilih tombol soal dan selesai menjawab semua soal tersebut maka, sistem akan memilih mana jawaban yang paling banyak benar nya untuk di tampilkan ke *rating* paling atas dan mana jawaban yang paling sedikit benar nya untuk ditampilkan ke *rating* paling bawah.

3.3.3 Rancangan Struktur Tabel

Dalam pembuatan aplikasi *mobile* ini mempunyai beberapa tabel yang digunakan untuk keperluan dalam pengolahan *Database*. Tabel – tabel ini kemudian dikelompokkan sesuai dengan kriterianya masing masing. Agar dalam proses pengolahan *Datanya* lebih mudah.

1. Tabel Soal

Pada tabel soal ini berisi beberapa soal tes tentang pewayangan yang kemudian harus dijawab oleh *user*. Sehingga dengan adanya soal tes ini, *user* bisa menguji kemampuannya dalam mempelajari budaya Indonesia tentang pewayangan. Adapun tabel soal ditunjukkan pada tabel 3.1.

Tabel 3.1 Tabel Soal

No.	Field	Tipe Data	Keterangan
1	Id_	Varchar (100)	Id setiap soal
2	soal	Varchar (100)	Isi masing – masing soal
3	Jawaban_a	Varchar (100)	Isi jawaban a
4	Jawaban_b	Varchar (100)	Isi jawaban b
5	Jawaban_c	Varchar (100)	Isi jawaban c
6	Jawaban_benar	Varchar (100)	Isi jawaban yang benar

2. Tabel Skor_hasil

Pada tabel *user* ini terdapat informasi mengenai nama *user*. Kemudian dari nama *user* ini digunakan untuk menampilkan hasil *scoring* tadi. Sehingga ketika *user* selesai menjawab soal dan setelah itu mengisi nama ini, maka sistem akan menampilkan informasi di *layout* berupa *rating* terhadap jawaban *user* yang disertai nama *user* tersebut. Adapun tabel *user* ditunjukkan pada tabel 3.2.

Tabel 3.2 Tabel *User*

No.	Field	Tipe Data	Keterangan
1	Id	Varchar (100)	Id setiap <i>user</i>
2	name	Varchar (1000)	Nama masing – masing <i>user</i>
3	score	Varchar (100)	Hasil penilaian

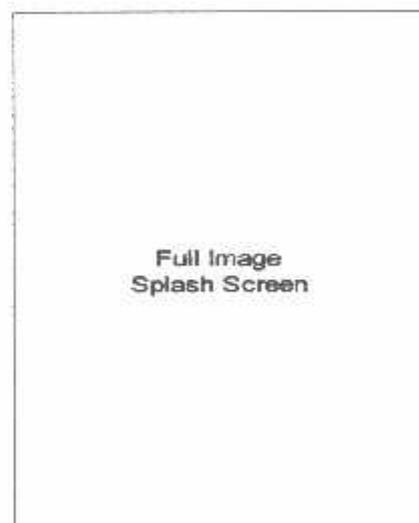
3.3.4 Rancangan *Interface*

Pada rancangan *Interface* ini menjelaskan tentang bentuk tampilan dari aplikasi *mobile* ini yang akan dibuat sesuai dengan tata letak *menu* pada *layout*. Tampilan ini dibuat sesuai dengan kenyamanan bagi pengguna atau *user-friendly*.

Dalam pembuatan aplikasi ini penulis memilih *RelativeLayout* untuk mengatur tampilan dari setiap *Interface*. *RelativeLayout* ini adalah sebuah *layout* dimana posisi dari sebuah komponen (*simbol, text, dsb*) letaknya bisa diatur terhadap komponen lainnya.

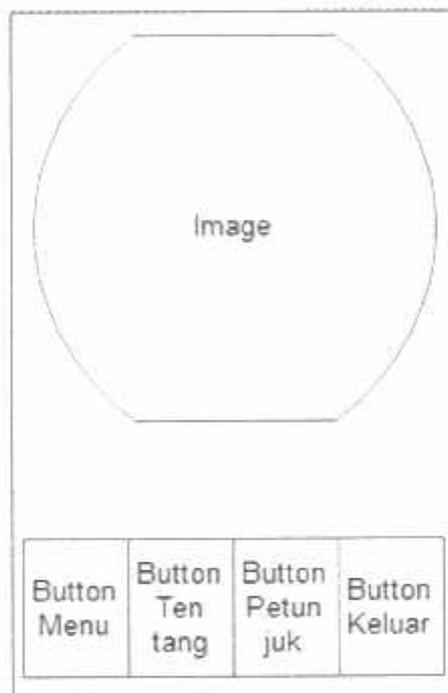
1. *Opening Layout*

Opening layout merupakan tampilan awal ketika aplikasi ini dijalankan. Pada *layout* ini hanya berisi gambar salah satu tokoh wayang yang berfungsi sebagai pembuka dari aplikasi ini. Ketika gambar ini ditekan maka sistem akan menuju ke *layout home*. Adapun *opening layout* ditunjukkan pada gambar 3.7.

Gambar 3.7 *Opening Layout*

2. *Layout Home*

Pada *layout home* memaparkan tentang desain *layout home*. Pada *layout* utama ini berisikan 4 *button* utama yaitu *button* petunjuk, *button* tentang, *button* Menu, dan *button* Keluar. Pada *button* tentang berisikan tentang informasi tentang aplikasi, *button* petunjuk berisikan tentang petunjuk penggunaan aplikasi, *button* Menu berisikan pilihan pada *menu* utama, sedangkan *button* Keluar berisikan pilihan keluar dari aplikasi ini atau tidak. Adapun *layout home* ditunjukkan pada gambar 3.8.



Gambar 3.8 *Layout Home*

3. *Layout Tentang*

Pada *layout* tentang ini memaparkan tentang informasi *software* yang dibuat seperti versi dan tahun pembuatan. Adapun *layout* tentang ditunjukkan pada gambar 3.9.



Gambar 3.9 *Layout Tentang*

4. *Layout Petunjuk*

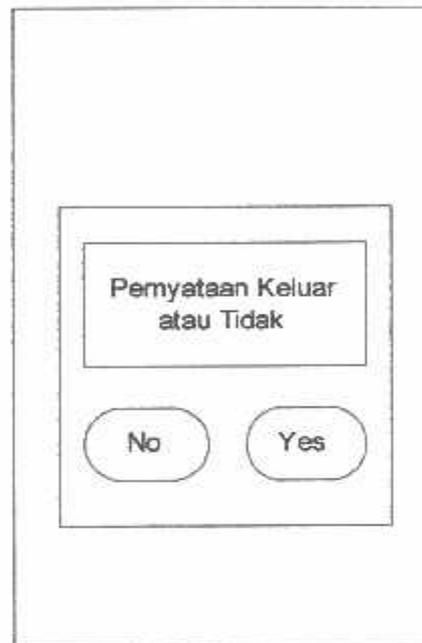
Pada *layout* petunjuk ini memaparkan tentang informasi serta petunjuk dalam penggunaan aplikasi wayang. Jadi dengan adanya petunjuk ini, *user* tidak perlu bertanya kepada siapa lagi disaat mengalami kesulitan dalam penggunaan aplikasi wayang ini. Tampilan petunjuk ini ditunjukkan pada gambar 3.10.



Gambar 3.10 *Layout Petunjuk*

5. *Layout Keluar*

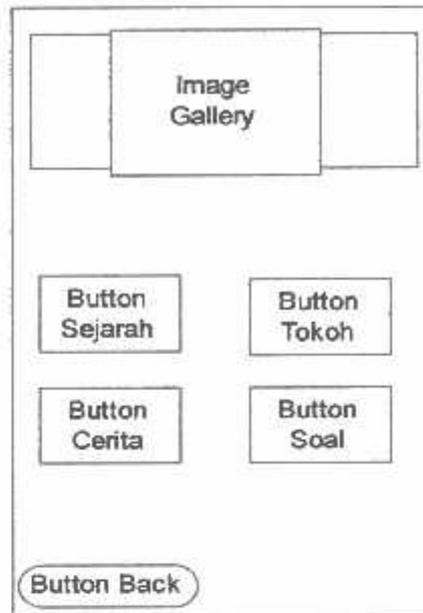
Pada *layout* keluar memaparkan tentang aplikasi ini ketika *user* ingin keluar dari aplikasi *mobile* tokoh pewayangan. Saat *user* menekan tombol keluar, maka *user* diberikan pertanyaan apakah *user* benar – benar ingin keluar dari aplikasi atau tidak. Pertanyaan ini muncul seperti *Interface* peringatan. Didalam *Interface* ini diberikan dua macam tombol pilihan yaitu tombol ya dan tombol tidak. Jika *user* memilih tombol iya maka aplikasi akan ditutup. Dan jika memilih tidak, maka aplikasi akan tetap berjalan. Adapun *layout* keluar ditunjukkan pada gambar 3.11.



Gambar 3.11 *Layout Keluar*

6. *Layout Menu Utama*

Pada *layout menu* utama ini memaparkan tentang tampilan dari *menu* utama. Pada *menu* utama ini berisikan 5 macam tombol yaitu tombol sejarah, tombol cerita, tombol tokoh, tombol soal dan satu tombol *back* untuk kembali ke *home*. Selain 5 tombol tersebut, pada *menu* utama ini juga disediakan tampilan *slide show gallery* tokoh wayang. Adapun *layout menu* utama ini ditunjukkan pada gambar 3.12.



Gambar 3.12 *Layout Menu Utama*

7. *Layout Sejarah*

Pada *layout* sejarah ini memaparkan tentang isi serta informasi dari sejarah wayang tersebut yang berujung pada pencarian asal mula budaya wayang yang ada di Indonesia. Pada *Interface* sejarah ini, dalam pembuatannya dengan menggunakan *scroll view* agar nantinya huruf yang dibaca oleh *user* bisa bergeser keatas dan kebawah. Adapun *layout* sejarah ini ditunjukkan pada gambar 3.13.



Gambar 3.13 *Layout Sejarah*

8. *Layout Opsi Cerita*

Pada *layout* cerita ini memaparkan tentang berbagai macam cerita tentang pewayangan yang diambil dari kisah perseteruan antara pandawa lima dengan kurawa yang menimbulkan perang bharatayudha. Data dari setiap cerita pewayangan tersebut disimpan kedalam *string.xml* yang kemudian ditampilkan kedalam *file* *cerita_ramayana.xml* dan *cerita_mahabharata.xml*. didalam tombol *opsi* cerita ini terdapat tampilan gambar yang berfungsi sebagai ilustrasi dari cerita ramayana dan cerita mahabharata. Adapun *layout opsi* cerita ditunjukkan pada gambar 3.14.



Gambar 3.14 *Layout Opsi Cerita*

9. *Layout Cerita Ramayana*

Pada *layout* cerita ramayana ini memaparkan tentang isi cerita dari kisah ramayana yaitu tentang penculikan dewi sinta oleh rahwana yang kemudian kerajaan rahwana diporak-porandakan oleh hanoman. Adapun *layout* cerita ramayana ini ditunjukkan pada gambar 4.15.



Gambar 3.15 *Layout* Cerita Ramayana

10. *Layout* Cerita Mahabharata

Pada *layout* cerita mahabharata ini memaparkan tentang isi cerita dari kisah mahabharata yaitu tentang perseteruan antara pandawa lima dengan kurawa yang menyebabkan adanya perang bharatayuda yang kemudian dimenangkan oleh pihak pandawa lima. Adapun *layout* cerita mahabharata ditunjukkan pada gambar 4.16.



Gambar 3.16 *Layout* Cerita Mahabharata

11. *Layout Opsi Tokoh*

Pada *layout Opsi Tokoh* ini memaparkan tentang berbagai macam tokoh tentang pewayangan yang kemudian diklasifikasikan menjadi 3 macam diantaranya tokoh dewa dan dewi, tokoh ramayana dan tokoh mahabharata. Data dari setiap tokoh pewayangan tersebut disimpan kedalam *string.xml* yang kemudian ditampilkan kedalam *file* tokoh_dewa_dewi.xml, tokoh_ramayana.xml dan cerita_mahabharata.xml. didalam tombol *opsi cerita* ini terdapat tampilan gambar yang berfungsi sebagai ilustrasi yang diambil dari tokoh dewa dan dewi, tokoh ramayana dan tokoh mahabharata. Adapun *layout opsi cerita* ditunjukkan pada gambar 3.17.



Gambar 3.17 *Layout Opsi Cerita*

12. *Layout List Tokoh Dewa Dan Dewi*

Pada *layout* tokoh dewa dan dewi ini memaparkan tentang nama – nama tokoh pewayangan dewa dan dewi. Jumlah tokoh dewa dan dewi ini sebanyak 30 macam. Selain itu pada *Interface* ini terdapat gambar tokoh wayang serta diberikannya fitur pencarian tokoh pewayangan. Dengan adanya pencarian ini, maka *user* bisa lebih mudah melakukan pencarian terhadap tokoh wayang yang ingin dituju. Adapun *layout list* tokoh dewa dan dewi ini bisa dilihat pada gambar 3.18.



Gambar 3.18 *Layout* Tokoh Dewa Dan Dewi

13. *Layout Detail* Tokoh Dewa Dan Dewi

Pada *layout detail* tokoh dewa dan dewi ini memaparkan tentang informasi dari setiap tokoh dewa dan dewi dalam pewayangan yang dipilih oleh *user*. Didalam *Interface* ini terdapat nama tokoh, gambar dari tokoh tersebut, serta informasi mengenai tokoh tersebut. Untuk mengetahui yang lebih jelasnya, pada *layout detail* tokoh dewa dan dewi ini bisa dilihat pada gambar 3.19.



Gambar 3.19 *Layout Detail* Tokoh Dewa Dan Dewi

14. *Layout List Tokoh Ramayana*

Pada *layout* tokoh ramayana ini memaparkan tentang nama – nama tokoh pewayangan ramayana. Jumlah tokoh ramayana ini sebanyak 30 macam. Selain itu pada *Interface* ini terdapat gambar tokoh wayang serta diberikannya fitur pencarian tokoh pewayangan. Dengan adanya pencarian ini, maka *user* bisa lebih mudah melakukan pencarian terhadap tokoh wayang yang ingin dituju. Adapun *layout list* tokoh ramayana ini bisa dilihat pada gambar 3.20.



The image shows a vertical list of search options. At the top is a button labeled "Cari Data". Below it are five text input fields, each containing the text "Nama Tokoh 1", "Nama Tokoh 2", "Nama Tokoh 3", "Nama Tokoh 4", and "Nama Tokoh 5" respectively. At the bottom of the list is a button labeled "Button Back".

Gambar 3.20 *Layout Tokoh Ramayana*

15. *Layout Detail Tokoh Ramayana*

Pada *layout detail* tokoh Ramayana ini memaparkan tentang informasi dari setiap tokoh ramayana dalam pewayangan yang dipilih oleh *user*. Didalam *Interface* ini terdapat nama tokoh, gambar dari tokoh tersebut, serta informasi mengenai tokoh tersebut. Untuk mengetahui yang lebih jelasnya, pada *layout detail* tokoh ramayana ini bisa dilihat pada gambar 3.21.



Gambar 3.21 *Layout Detail Tokoh Ramayana*

16. *Layout List Tokoh Mahabharata*

Pada *layout* tokoh mahabharata ini memaparkan tentang nama – nama tokoh pewayangan mahabharata. Jumlah tokoh mahabharata ini sebanyak 30 macam. Selain itu pada *Interface* ini terdapat gambar tokoh wayang serta diberikannya fitur pencarian tokoh pewayangan. Dengan adanya pencarian ini, maka *user* bisa lebih mudah melakukan pencarian terhadap tokoh wayang yang ingin dituju. Adapun *layout list* tokoh mahabharata ini bisa dilihat pada gambar 3.22.



Gambar 3.22 *Layout Tokoh Mahabharata*

17. *Layout Detail* Tokoh Mahabharata

Pada *layout detail* tokoh mahabharata ini memaparkan tentang informasi dari setiap tokoh mahabharata dalam pewayangan yang dipilih oleh *user*. Didalam *Interface* ini terdapat nama tokoh, gambar dari tokoh tersebut, serta informasi mengenai tokoh tersebut. Untuk mengetahui yang lebih jelasnya, pada *layout detail* tokoh mahabharata ini bisa dilihat pada gambar 3.23.



Gambar 3.23 *Layout Detail* Tokoh Mahabharata

18. *Layout Soal*

Pada *layout soal* ini memaparkan tentang soal – soal yang harus dijawab oleh *user*. Soal ini ditampilkan secara *random* dan *user* diberikan 10 macam pertanyaan yang berbeda yang diambil dari *database*. Jumlah soal yang ada dalam *database* ini sebanyak 50 soal dan setiap soal terdapat 3 macam jawaban. Adapun *layout soal* ini ditunjukkan pada gambar 3.24.



Gambar 3.24 *Layout Soal*

19. *Layout Input Nama User*

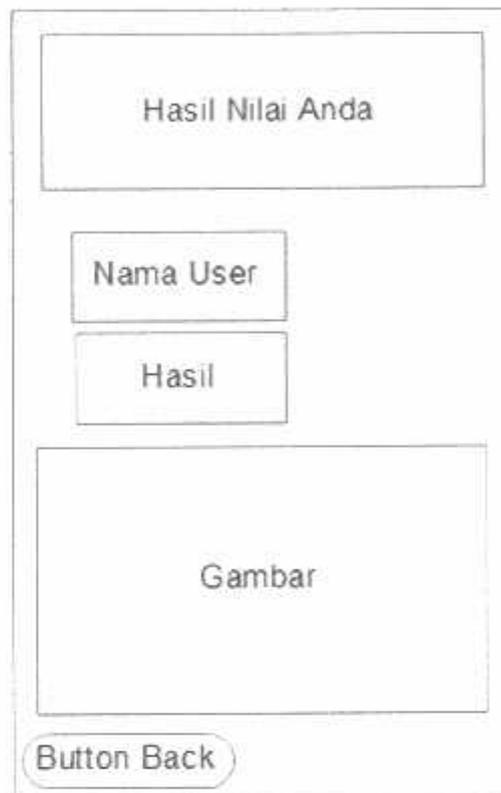
Pada *layout* input nama *user* ini digunakan ketika *user* sudah menjawab semua pertanyaan dari aplikasi *mobile* wayang dan sudah menekan tombol lihat hasil. *Input* nama *user* ini disimpan kedalam *Database* *SQLite* yang kemudian di tampilkan ke *layout* *scoring* yang disertai *score* serta banyaknya jawaban benar yang dijawab. *Layout* input nama *user* ini bisa dilihat pada gambar 3.25.



Gambar 3.25 *Layout Input Nama User*

20. *Layout Nilai*

Pada *layout* nilai ini memaparkan tentang hasil dari jawaban *user*. Didalam *Interface* ini, *user* bisa melihat namanya yang sudah diinputkan beserta hasil jawaban. Selain itu, *user* juga bisa mengetahui *rating* yang berhasil didapatkannya. *Layout scoring* ini bisa dilihat pada gambar 3.26.



Gambar 3.26 *Layout Nilai*

Seperti pada gambar 3.26 tersebut menjelaskan bahwa untuk penentuan *rating* ini dilakukan berdasarkan nilai yang berhasil didapatkan oleh *user*. Jika *user* mendapatkan nilai tertinggi maka *username* beserta nilainya akan ditampilkan ke posisi paling atas. Sedangkan, jika *user* mendapatkan nilai jawaban yang paling rendah maka *username* beserta nilainya akan ditampilkan di posisi yang paling bawah.

BAB IV IMPLEMENTASI DAN PENGUJIAN

Dalam bab ini akan dibahas tentang implementasi dan pengujian berdasarkan perencanaan dari sistem yang dibuat. Pengujian ini dilaksanakan untuk mengetahui kinerja dari suatu sistem yang dibuat apakah sudah sesuai dengan perencanaan yang diinginkan.

4.1 Implementasi

Pada tahapan ini akan membahas tentang bagaimana membuat penerapan dari perancangan sistem yang telah dibuat sebelumnya. Perancangan dan penerapan sistem yang dilakukan meliputi 2 aspek yaitu *user interface* atau tampilan antar muka sebagai penghubung dengan *user* dan implementasi program.

4.2 Implementasi *User Interface*

Implementasi *user interface* pada sistem aplikasi *mobile Android* tentang pewayangan ini mempunyai 22 (dua puluh dua) tampilan *interface* yaitu:

1. Tampilan *Splash Screen*
2. Tampilan *Home*
3. Tampilan Tentang
4. Tampilan Petunjuk
5. Tampilan Keluar
6. Tampilan *Menu* Utama
7. Tampilan Sejarah
8. Tampilan *Opsi* Cerita
9. Tampilan Cerita Ramayana
10. Tampilan Cerita Mahabharata
11. Tampilan *Opsi* Tokoh
12. Tampilan *List* Tokoh Dewa – Dewi
13. Fitur Pencarian *Autocomplete*
14. Tampilan *Detail* Tokoh Dewa – Dewi
15. Tampilan *List* Tokoh Ramayana

16. Tampilan *Detail* Tokoh Ramayana
17. Tampilan *List* Tokoh Mahabharata
18. Tampilan *Detail* Tokoh Mahabharata
19. Tampilan Soal
20. Tampilan Isi Soal
21. Tampilan *Input* Nama *User*
22. Tampilan *Rating*

4.2.1 Tampilan *Splash Screen*

Tampilan *splash screen* merupakan tampilan pertama ketika *user* menjalankan program ini. Pada tampilan ini, terdapat gambar wayang beserta nama aplikasi wayang ini yang dinamai jawi wayang. Adapun tampilan *splash screen* ditunjukkan pada gambar 4.1.

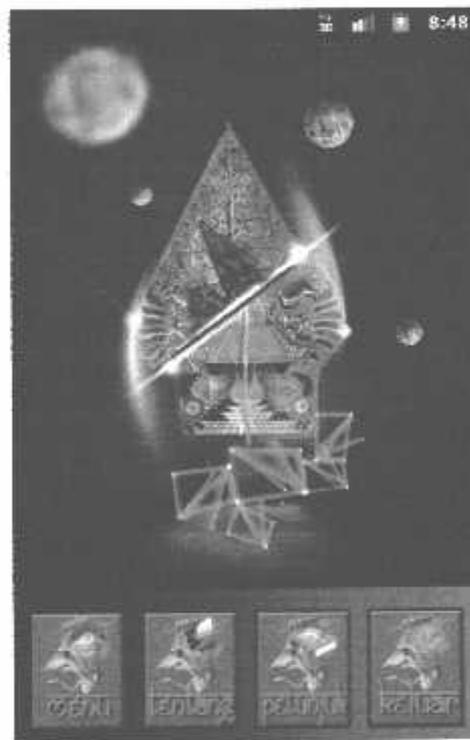


Gambar 4.1 Tampilan *Splash Screen*

Cara kerja dari tampilan *splash screen* tersebut yaitu dengan memanfaatkan *Fadein* dan *Fadeout* pada xml. Pada *Fadein* ini berfungsi untuk memunculkan gambar secara perlahan dan *Fadeout* untuk menghilangkan gambar secara perlahan.

4.2.2 Tampilan *Home*

Pada tampilan *home* ini merupakan tampilan utama dari aplikasi wayang ini yang muncul secara otomatis setelah tampilan *splash screen* berakhir. Adapun tampilan *home* ini bisa dilihat pada gambar 4.2.



Gambar 4.2 Tampilan *Home*

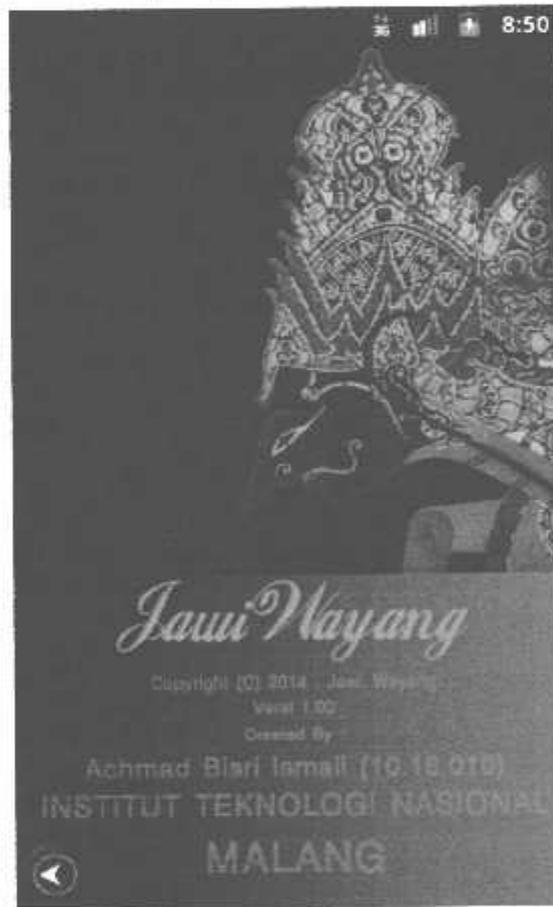
Pada tampilan *home* ini mempunyai 4 macam *Button* yang memiliki fungsi sebagai berikut:

1. *Button Menu* berfungsi untuk Menuju ke *interface Menu* utama.
2. *Button Petunjuk* berfungsi untuk memberikan bantuan dalam menggunakan aplikasi wayang ini.
3. *Button Tentang* berfungsi sebagai informasi tentang aplikasi wayang ini.

4. *Button* Keluar berfungsi ketika *user* ingin keluar dari aplikasi ini. Namun, ketika *user* memilih *Button* ini maka background *service* akan muncul dan memberikan dua macam pilihan yaitu antara iya atau tidak. Jika *user* memilih tidak, maka sistem masih tetap menjalankan aplikasi ini. Jika *user* memilih iya, maka sistem akan keluar dari aplikasi ini.

4.2.3 Tampilan Tentang

Tampilan tentang ini berisi tentang informasi dari aplikasi ini. Seperti tahun pembuatan aplikasi, nama aplikasi, gambar, serta versi yang dibuat. Adapun untuk tampilan tentang ini bisa dilihat pada gambar 4.3.



Gambar 4.3 Tampilan Tentang

Pada tampilan tentang hanya terdapat 1 *Button* yang berfungsi untuk kembali ke *Menu home*.

4.2.4 Tampilan Petunjuk

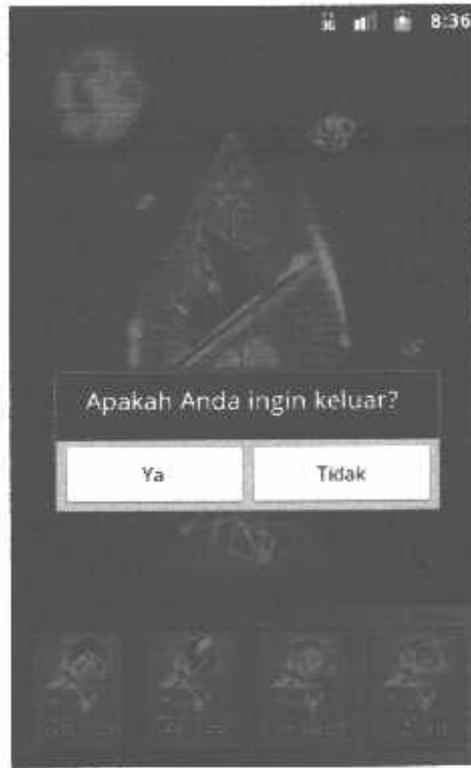
Tampilan petunjuk ini berisi tentang buku pedoman penggunaan aplikasi wayang ini. Jadi dengan adanya petunjuk ini, *user* tidak perlu bertanya kepada siapa lagi disaat mengalami kesulitan dalam penggunaan aplikasi wayang ini. Tampilan petunjuk ini ditunjukkan pada gambar 4.4.



Gambar 4.4 Tampilan Petunjuk

4.2.5 Tampilan Keluar

Pada tampilan keluar ini berfungsi untuk keluar dari aplikasi pewayangan ini. Ketika *user* menekan *Button* ini, maka *user* akan diberikan 2 macam pilihan diantaranya *Button* iya dan *Button* tidak. Jika *user* menekan *Button* tidak, maka sistem akan tetap menjalankan aplikasi pewayangan ini. Namun, jika *user* memilih *Button* iya maka, sistem akan keluar dari aplikasi ini. Adapun tampilan keluar ini ditunjukkan pada gambar 4.5.



Gambar 4.5 Tampilan Keluar

4.2.6 Tampilan Menu Utama

Pada tampilan *menu* utama ini terdapat *slide show Gallery* yang bisa digeser kekanan dan kekiri. *Slide show* ini berfungsi untuk mempercantik tampilan agar *user* tidak mengalami kejenuhan.

Didalam tampilan ini terdapat beberapa tombol yang diantaranya adalah :

1. Tombol sejarah : Tombol ini berfungsi untuk Menuju ke *layout* sejarah.
2. Tombol cerita : Tombol ini berfungsi untuk Menuju ke *layout* cerita yang berisi kumpulan dari berbagai macam cerita pewayangan.

3. Tombol tokoh : Tombol ini berfungsi untuk Menuju ke *layout* tokoh yang berisi kumpulan tokoh dalam pewayangan.
4. Tombol soal : Tombol ini berfungsi untuk Menuju ke *layout* soal.
5. Tombol *back* : Tombol ini berfungsi untuk kembali ke *home*.

Adapun tampilan *menu* utama ini ditunjukkan pada gambar 4.6.



Gambar 4.6 Tampilan *Menu* Utama

4.2.7 Tampilan Sejarah

Pada tampilan sejarah ini berisikan *detail* dari isi sejarah tentang pewayangan di Indonesia, terutama tentang wayang purwo. Didalam tampilan ini terdapat gambar tentang pewayangan, kemudian yang ada dibawahnya terdapat *text* keterangan tentang informasi dari isi sejarah ini yang bisa di *scroll* keatas dan

kebawah sesuai keinginan saat membaca. Dan juga terdapat tombol untuk kembali ke tampilan *Menu*. Tampilan sejarah ini bisa dilihat pada gambar 4.7.



Gambar 4.7 Tampilan Sejarah

4.2.8 Tampilan *opsi* Cerita

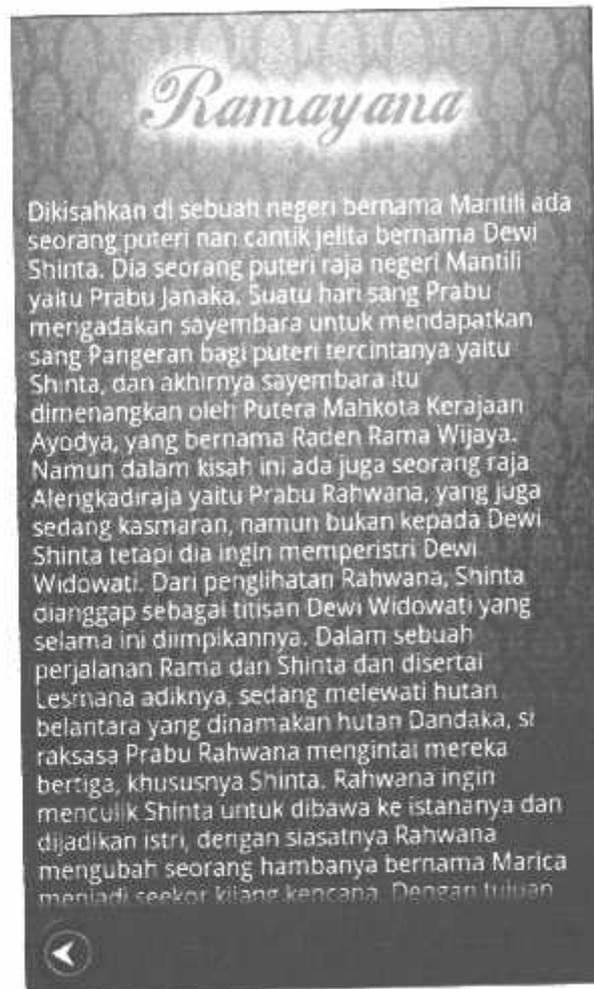
Pada tampilan ini berisikan 2 macam cerita yang diambil dari kisah pewayangan ramayanan dan mahabharata. Ke-2 macam cerita tersebut adalah cerita ramayana dan cerita mahabharata. Dalam tampilan ini terdapat 2 macam gambar yang digunakan sebagai ilustrasi dari cerita ramayana dan mahabharata ini yang juga berfungsi sebagai tombol untuk menuju ke cerita ramayana dan ke cerita mahabharata. Adapun tampilan *opsi* cerita ini ditunjukkan pada gambar 4.8.



Gambar 4.8 Tampilan *Opsi Cerita*

4.2.9 Tampilan *Detail Cerita Ramayana*

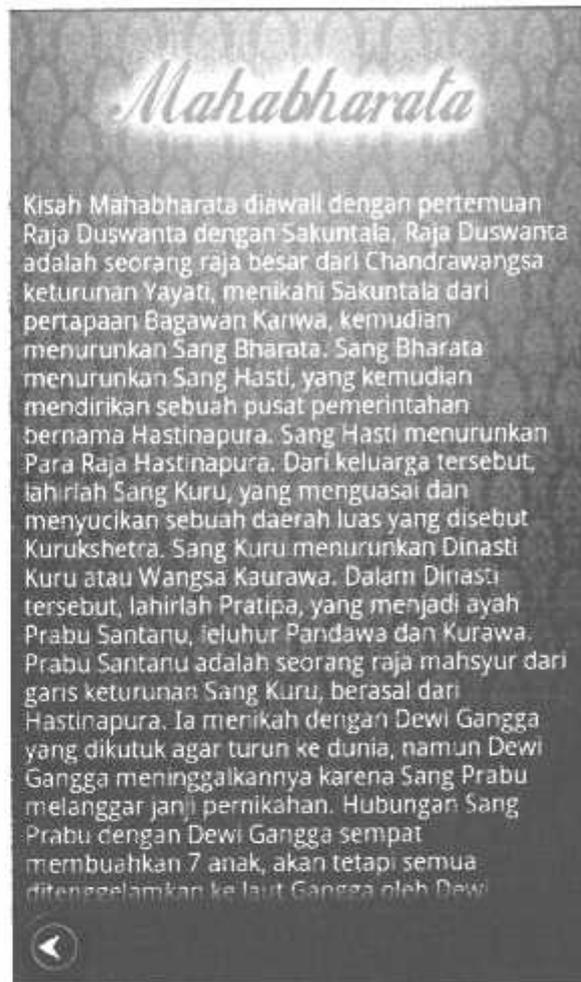
Pada tampilan *detail* cerita ramayana ini berisikan tentang informasi atau isi cerita yang diambil dari kisah ramayana yang kisahnya menceritakan kisah penculikan dewi sinta oleh rahwana. Cerita dari kisah ramayana ini termasuk jenis cerita singkat dari dewi sinta ditinggal rama untuk berburu hingga berhasil membawa dewi sinta pulang. Tampilan *detail* cerita ramayana ini dapat digeser keatas dan kebawah untuk mengetahui isi ceritanya. Selain itu, pada tampilan ini diberikan tombol kembali untuk kembali ke *Menu* utama aplikasi ini. Untuk tampilan *detail* cerita ramayana ini ditunjukkan pada gambar 4.9.



Gambar 4.9 Tampilan *Detail* Cerita Ramayana

4.2.10 Tampilan *Detail* Cerita Mahabharata

Pada tampilan *detail* cerita mahabharata ini berisikan tentang informasi atau isi cerita yang diambil dari kisah mahabharata yang kisahnya menceritakan tentang perseteruan antara pandawa dengan kurawa. Cerita dari kisah mahabharata ini termasuk jenis cerita singkat dari perang bharatayuda yaitu perang antara kebenaran melawan kejahatan. Tampilan *detail* cerita mahabharata ini dapat digeser keatas dan kebawah untuk mengetahui isi ceritanya. Selain itu, pada tampilan ini diberikan tombol *back* untuk kembali ke *Menu* utama aplikasi ini. Untuk tampilan *detail* cerita mahabharata ini ditunjukkan pada gambar 4.10.



Gambar 4.10 Tampilan *Detail* cerita Mahabharata

4.2.11 Tampilan *Opsi* Tokoh

Didalam tampilan *opsi* tokoh ini terdapat 3 macam tombol pilihan yang didasarkan pada pengklasifikasian setiap tokoh pewayangan. Ketiga macam tombol tersebut adalah tombol tokoh dewa dan dewi, tombol tokoh Ramayana dan tombol tokoh Mahabharata. Semua data tokoh yang ada pada aplikasi ini sebanyak 150 macam tokoh pewayangan. Adapun tampilan *opsi* tokoh tersebut ditunjukkan pada gambar 4.11.



Gambar 4.11 Tampilan *Opsi Tokoh*

4.2.12 Tampilan *List Tokoh Dewa Dan Dewi*

Pada tampilan *List* tokoh dewa dan dewi ini berisikan tentang berbagai macam jenis nama tokoh dalam pewayangan yang diambil dari tokoh dewa dan dewi. Tokoh yang diinputkan kedalam *List* ini sebanyak 30 macam tokoh. *User* bisa membaca dan memilih tiap tokoh yang ingin dibacanya dengan menggeser layar keatas dan kebawah untuk mengetahui tiap nama tokoh yang masih tersembunyi. Setelah *user* menemukan nama tokoh wayang yang dicarinya, *user* bisa menekan nama tokoh yang dipilihnya untuk mengetahui *detail* tokoh atau informasi tentang tokoh wayang tersebut. Tampilan *List* tokoh wayang ini ditunjukkan pada gambar 4.12.

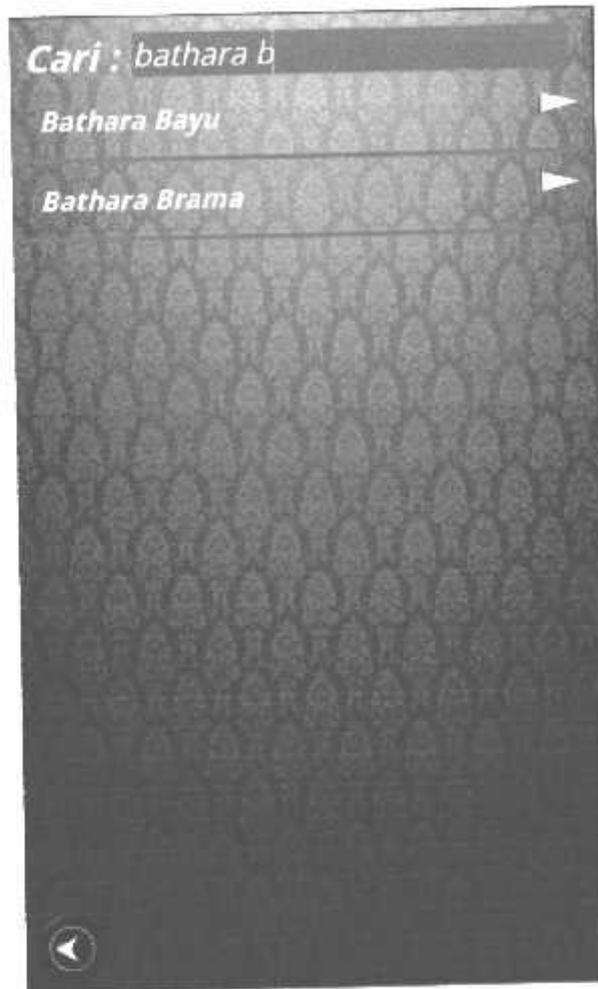


Gambar 4.12 Tampilan *List* Tokoh Dewa Dan Dewi

4.2.13 Tampilan Fitur *Autocomplete*

Pada aplikasi ini juga diberikan fitur berupa pencarian *Data*. Pencarian ini diberikan untuk mempermudah *user* dalam melakukan pencarian nama tokoh wayang yang diinginkannya. Fitur pencarian ini sangat berguna ketika *Data* yang ada sangat banyak. Fitur pencarian ini menggunakan metode *Autocomplete*. Dengan adanya metode *Autocomplete* ini, *user* tidak perlu menetik semua kata dalam nama tokoh wayang ini. *User* hanya cukup menetik kata awal hingga sampai pada kata yang diperlukan untuk memunculkan judul yang dicarinya. Setelah kata yang dicari oleh *user* ditampilkan oleh sistem, maka *user* tinggal

menekan kata tersebut untuk mengetahui *detail* informasi yang ingin dibacanya. Selain itu, kegunaan *Autocomplete* ini yaitu *user* tidak perlu mengetik maupun menghafal kata yang terlalu panjang atau sulit dihafalkan. Untuk tampilan pencarian *Data* tokoh pewayangan ini ditunjukkan pada gambar 4.13.

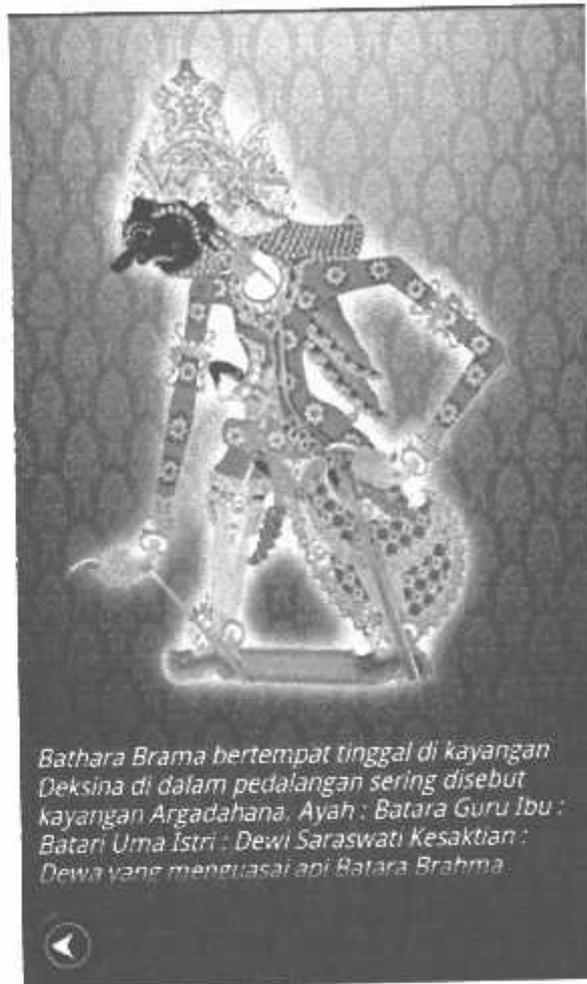


Gambar 4.13 Tampilan Fitur Pencarian *Autocomplete*

4.2.14 Tampilan *Detail* Tokoh Dewa Dan Dewi

Pada tampilan *detail* tokoh dewa dan dewi ini berisikan tentang informasi atau isi dari setiap jenis tokoh dewa dan dewi yang sudah dipilih. Tampilan *detail* tokoh dewa dan dewi ini pun juga bisa digeser keatas dan kebawah untuk mengetahui isi tokoh dewa dan dewinya. Selain itu, pada tampilan ini diberikan

tombol *back* untuk kembali ke *Menu* utama aplikasi ini. Untuk tampilan *detail* tokoh dewa dan dewi ini ditunjukkan pada gambar 4.14.



Gambar 4.14 Tampilan *Detail* Tokoh Dewa Dan Dewi

4.2.15 Tampilan *List* Tokoh Ramayana

Pada tampilan *List* tokoh ramayana ini berisikan tentang berbagai macam jenis nama tokoh dalam pewayangan yang diambil dari tokoh ramayana. Tokoh yang diinputkan kedalam *List* ini sebanyak 30 macam tokoh. *User* bisa membaca

dan memilih tiap tokoh yang ingin dibacanya dengan menggeser layar keatas dan kebawah untuk mengetahui tiap nama tokoh yang masih tersembunyi. Setelah *user* menemukan nama tokoh wayang yang dicarinya, *user* bisa menekan nama tokoh yang dipilihnya untuk mengetahui *detail* tokoh atau informasi tentang tokoh wayang tersebut. Tampilan *List* tokoh ramayana ini ditunjukkan pada gambar 4.15.



Gambar 4.15 Tampilan *List* Tokoh Ramayana

4.2.16 Tampilan *Detail* Tokoh Ramayana

Pada tampilan *detail* tokoh ramayana ini berisikan tentang informasi atau isi dari setiap jenis nama - nama tokoh ramayana yang sudah dipilih. Tampilan

detail tokoh ramayana ini pun juga bisa digeser keatas dan kebawah untuk mengetahui isi tokoh ramayananya. Selain itu, pada tampilan ini diberikan tombol *back* untuk kembali ke *Menu* utama aplikasi ini. Adapun tampilan *detail* tokoh ramayana ini ditunjukkan pada gambar 4.16.



Gambar 4.16 Tampilan *Detail* Tokoh Ramayana

4.2.17 Tampilan *List* Tokoh Mahabharata

Pada tampilan *List* tokoh mahabharata ini berisikan tentang berbagai macam jenis nama tokoh dalam pewayangan yang diambil dari tokoh mahabharata. Tokoh yang diinputkan kedalam *List* ini sebanyak 90 macam tokoh. *User* bisa membaca dan memilih tiap tokoh yang ingin dibacanya dengan menggeser layar

keatas dan kebawah untuk mengetahui tiap nama tokoh yang masih tersembunyi. Setelah *user* menemukan nama tokoh wayang yang dicarinya, *user* bisa menekan nama tokoh yang dipilihnya untuk mengetahui *detail* tokoh atau informasi tentang tokoh wayang tersebut. Tampilan *List* tokoh wayang ini ditunjukkan pada gambar 4.17.

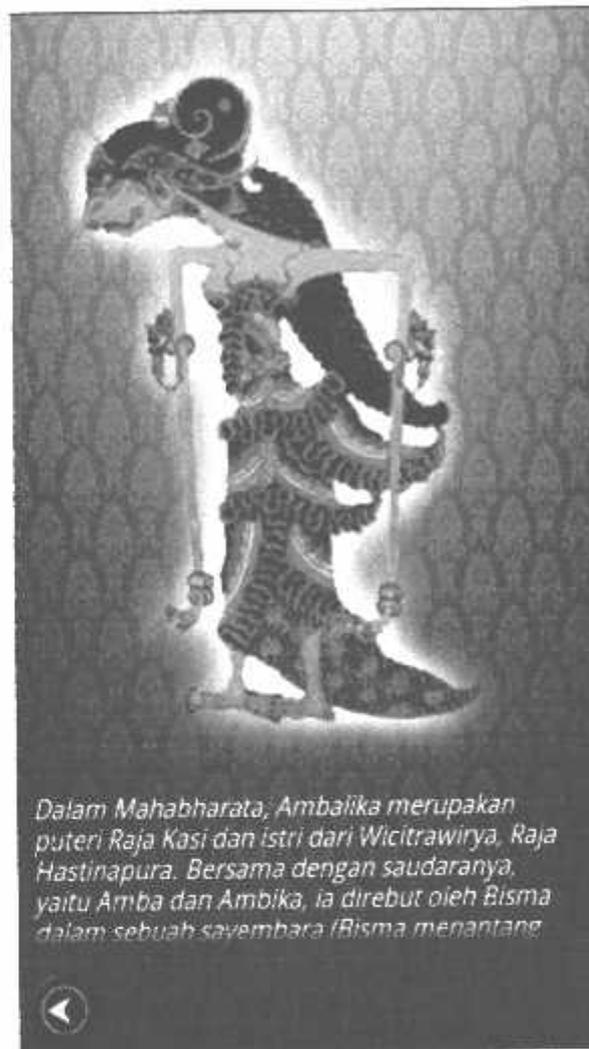


Gambar 4.17 Tampilan *List* Tokoh Mahabharata

4.2.18 Tampilan *Detail* Tokoh Mahabharata

Pada tampilan *detail* tokoh mahabharata ini berisikan tentang informasi atau isi dari setiap jenis nama - nama tokoh mahabharata yang sudah dipilih. Tampilan *detail* tokoh mahabharata ini pun juga bisa digeser keatas dan kebawah

untuk mengetahui isi tokoh mahabharatanya. Selain itu, pada tampilan ini diberikan tombol *back* untuk kembali ke *Menu* utama aplikasi ini. Adapun tampilan *detail* tokoh mahabharata ini ditunjukkan pada gambar 4.18.



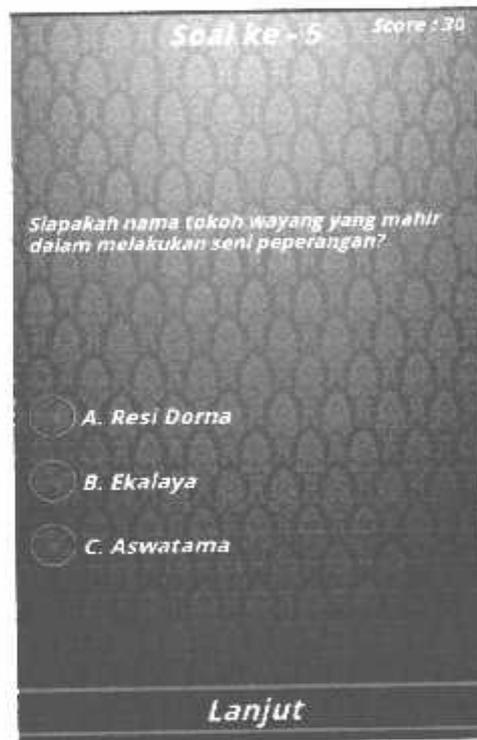
Gambar 4.18 Tampilan *Detail* Tokoh Mahabharata

4.2.19 Tampilan Soal

Pada tampilan soal berisikan tentang soal dan pertanyaan yang harus dijawab oleh *user*. Di dalam tampilan soal ini terdapat jenis soal, jenis pertanyaan, jawaban yang berupa *Radio Button*, nilai dan tombol *next* untuk Menuju ke soal

selanjutnya. *Database* pada soal ini menggunakan *Database SQLite Manager*. Soal yang ada dalam *Database* ini sebanyak 50 macam soal.

Cara kerja soal ini yaitu 50 macam soal yang ada di dalam *Database* ini diambil dan di *suffle* agar dalam proses pengambilannya secara acak. Selanjutnya soal ini dipilih hanya 10 macam soal saja yang kemudian harus dijawab oleh *user*. *User* bisa menjawab soal ini dengan menekan salah satu dari 3 tombol pilihan *Radio Button*. Setelah itu tekan tombol *next* untuk Menuju ke soal selanjutnya. Ketika *user* selesai menjawab soal pertama, jika jawaban benar maka hasil nilainya langsung di tampilkan ke soal yang kedua. Jadi *user* bisa langsung mengetahui apakah jawabannya benar atau tidak. Tampilan untuk soal ini ditunjukkan pada gambar 4.19.

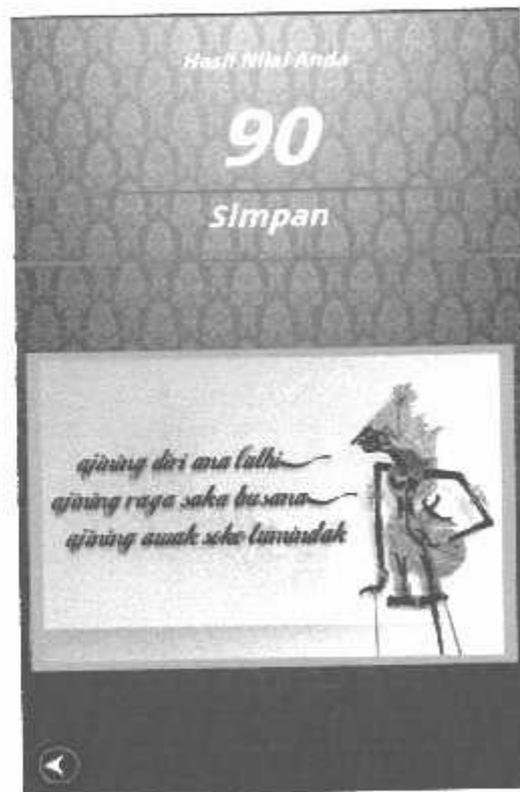


Gambar 4.19 Tampilan Soal

4.2.20 Tampilan Hasil Nilai

Tampilan hasil nilai merupakan tampilan ketika *user* selesai menjawab 10 macam pertanyaan yang ada dalam soal ini. Pada tampilan ini terdapat nilai akhir

user, tombol *save*, gambar *view*, dan tombol kembali ke *Menu* utama. Untuk nilai akhir *user* ini mempunyai nilai maksimal hingga 100. Sehingga, setiap *user* menjawab 1 pertanyaan dengan benar maka *user* akan mendapatkan nilai tambah 10 dan nilai ini akan terus bertambah setelah *user* berhasil menjawab pertanyaan dengan benar. Tetapi, ketika *user* menjawab pertanyaan dengan salah maka nilai ini tidak akan bertambah. Setelah itu, *user* bisa menyimpan hasil nilainya dengan menekan tombol *save* yang terletak dibawah nilai *user*. Jika *user* tidak ingin menyimpan nilainya, maka *user* memilih tombol kembali untuk Menuju ke *Menu* utama. Didalam tampilan ini juga terdapat gambar *view* berupa gambar tokoh wayang beserta salah satu kata mutiaranya. Tampilan untuk hasil nilai ini ditunjukkan pada gambar 4.20.



Gambar 4.20 Tampilan Hasil Nilai

4.2.21 Tampilan Simpan Data

Pada tampilan simpan *Data* ini merupakan tampilan berupa pesan *pop-up* setelah *user* menekan tombol simpan. Tombol *save* ini berfungsi untuk menyimpan *Username* beserta nilainya yang kemudian ditampilkan kedalam

Listview Rating nilai. *Data Username* dan nilai ini akan disimpan kedalam *Database*. Tampilan untuk simpan *Data* ini ditunjukkan pada gambar 4.21.

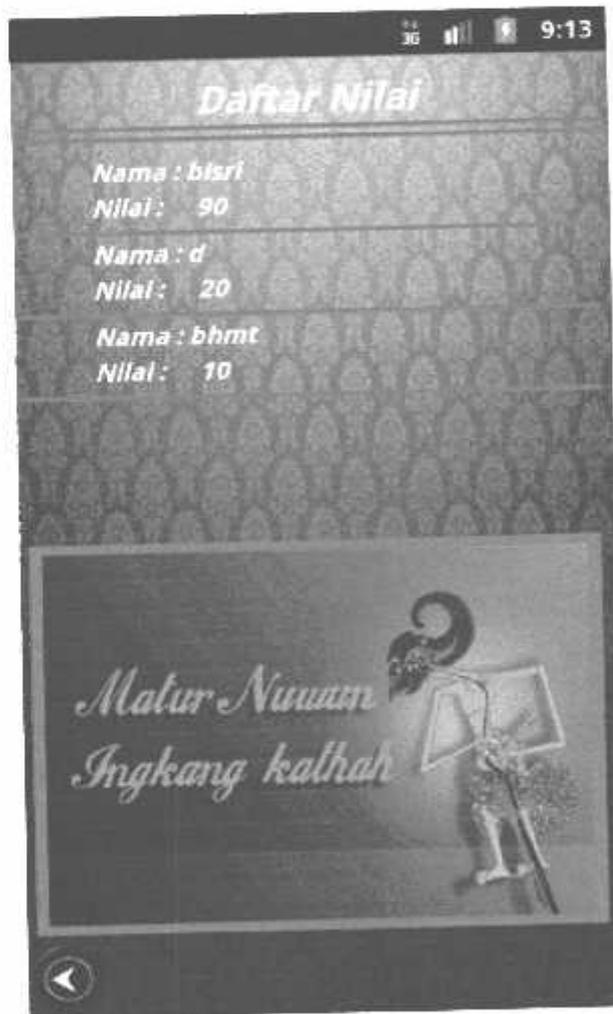


Gambar 4.21 Tampilan Simpan *Data*

Pada gambar 4.21 Menunjukkan bahwa *user* menginputkan nama. Nama ini kemudian disimpan kedalam *Database* beserta dengan nilai hasil yang diperoleh *user*. Setelah *user* selesai menginputkan namanya, maka *user* akan diberikan 2 macam pilihan tombol diantaranya adalah tombol “ok” yaitu untuk menyetujui kemudian data nilainya disimpan kedalam *database* dan tombol “cancel” untuk membatalkannya.

4.2.22 Tampilan *Rating Data*

Pada tampilan *Rating Data* ini berisi tentang semua *Data Username* dan nilai yang disimpan kedalam *Database*. *Username* dan nilai ini ditampilkan dalam bentuk *Listview* agar nantinya bisa digeser keatas dan kebawah untuk mengetahui daftar nilainya. Nilai ini diurutkan berdasarkan nilainya. Jika nilai jawabannya paling tinggi, maka *Username* dan nilai ini akan diletakkan ke urutan paling atas. Sebaliknya, jika nilai jawabannya paling rendah, maka *Username* beserta nilainya akan diletakkan ke urutan paling bawah. Didalam tampilan *Rating* ini juga terdapat tombol *back* untuk kembali ke *Menu* utama dan gambar *view* yang Menunjukkan gambar salah satu tokoh wayang yang disertai ucapan terima kasih dari si penulis. Tampilan untuk *Rating Data* ini ditunjukkan pada gambar 4.22.



Gambar 4.22 Tampilan *Rating Data*

Pengujian Kesesuaian Fungsi dalam Sistem <i>Android</i>				Keakuratan
No.	Menu	Proses	Hasil	Dalam (%)
9	Tombol <i>Opsi Ya</i> (Keluar)	Keluar dari Aplikasi <i>Jawi Wayang</i>	Y	100 %
10	Tombol <i>Opsi Tidak</i> (Keluar)	Kembali ke <i>Menu Home</i>	Y	100 %
11	Tombol <i>Menu Sejarah</i>	Menuju ke <i>Sejarah</i>	Y	100 %
12	Tombol <i>Kembali</i> (<i>Sejarah</i>)	Kembali ke <i>Menu Utama</i>	Y	100 %
13	Tombol <i>Menu Cerita</i>	Menuju ke <i>Opsi Cerita</i>	Y	100 %
14	Tombol <i>Kembali</i> (<i>Opsi Cerita</i>)	Kembali ke <i>Menu Utama</i>	Y	100 %
15	Tombol <i>Cerita Ramayana</i>	Menuju ke <i>Detail Cerita Ramayana</i>	Y	100 %
16	Tombol <i>Kembali</i> (<i>Ramayana</i>)	Kembali ke <i>Opsi Cerita Ramayana</i>	Y	100 %
17	Tombol <i>Cerita Mahabharata</i>	Menuju ke <i>Detail Cerita Mahabharata</i>	Y	100 %
18	Tombol <i>Kembali</i> (<i>Mahabharata</i>)	Kembali ke <i>Opsi Tokoh</i>	Y	100 %
19	Tombol <i>Tokoh</i>	Menuju ke <i>Opsi Tokoh</i>	Y	100 %
20	Tombol <i>Kembali</i> (<i>Opsi Tokoh</i>)	Kembali ke <i>Menu Utama</i>	Y	100 %
21	Tombol <i>Tokoh Dewa Dewi</i>	Menuju ke <i>Detail Tokoh Dewa dan Dewi</i>	Y	100 %
22	Tombol <i>Kembali</i> (<i>Dewa Dewi</i>)	Menuju ke <i>Opsi Tokoh</i>	Y	100 %
23	Tombol <i>Tokoh Ramayana</i>	Menuju ke <i>Detail Tokoh Ramayana</i>	Y	100 %

Pengujian Kesesuaian Fungsi dalam Sistem <i>Android</i>				Keakuratan
No.	Menu	Proses	Hasil	Dalam (%)
24	Tombol Kembali (Ramayana)	Menuju ke <i>Opsi</i> Tokoh	Y	100 %
25	Tombol Tokoh Mahabharata	Menuju ke <i>Detail</i> Tokoh Mahabharata	Y	100 %
26	Tombol Kembali (Mahabharata)	Menuju ke <i>Opsi</i> Tokoh	Y	100 %
27	<i>Image Gallery</i>	<i>Gallery Image Menu</i>	Y	100 %
28	Tombol Soal	Menuju ke Soal	Y	100 %
29	<i>Radio Button</i> Jawaban	Centang Pilihan Jawaban	Y	100 %
30	Tombol Lanjut	Menuju ke Soal Selanjutnya	Y	100 %
31	Nilai	Hasil Jawaban	Y	100 %
32	Tombol Simpan	Simpan <i>Username</i> dan Nilai	Y	100 %
33	Tombol Kembali (Nilai)	Kembali ke <i>Menu</i> Utama	Y	100 %
34	<i>Rating</i> Nilai	<i>Rating Data</i> Berdasarkan Nilai	Y	100 %
35	Tombol Kembali (<i>Rating</i>)	Kembali ke <i>Menu</i> Utama	Y	100 %
Rata - rata				100 %

Keterangan :

Y = Berhasil

X = Gagal

Perhitungan :

$$\text{Nilai Pengujian Tiap Menu} = \frac{100\%}{\text{Jumlah Pengujian Fungsional}} = \frac{100\%}{35} = 2,857\%$$

Berdasarkan hasil pengujian dari tabel 4.1 menjelaskan bahwa dari 35 jenis pengujian fungsional tersebut sudah berhasil dengan tingkat keakuratan mencapai 100 %. *Data* ini diambil dari rata-rata sistem aplikasi yang diuji.

4.3.2 Pengujian Spesifikasi *Hardware*

Pengujian aplikasi merupakan suatu pengujian dimana aplikasi *android* tentang wayang ini dapat dijalankan ke berbagai spesifikasi *smartphone*. Pengujian ini dilakukan untuk mengetahui *bug-bug* yang terjadi ketika menjalankan aplikasi *Android* jawi wayang ini.

Tabel 4.2 Tabel Pengujian Spesifikasi *Hardware*

Pengujian Spesifikasi <i>Hardware</i> pada <i>Smartphone</i>					Tingkat Akurasi Dalam (%)
No.	Type	Design Layout	Pixel	Portrait Orientation	
1	Sony Ericsson Xperia Ray	Y	480 x 854	Y	100 %
2	Samsung Galaxy Mini 2 S6500D	Y	320 x 480	Y	97,138 %
3	Idea Tab A1000-G	Y	1024 x 600	Y	100 %
4	Smartfren Andromax U2.V1.00	Y	540 x 960	Y	97,138 %
5	Sony Xperia L	Y	480 x 854	Y	100 %
Rata - rata					98,8552 %

Keterangan :

Y = Berhasil

X = Gagal

Perhitungan :

Tingkat Akurasi = Jumlah Keberhasilan Pengujian Fungsional x Nilai Pengujian Tiap *Menu*

Rata - rata = Jumlah Semua Tingkat Akurasi / Jumlah Pengujian = 494,276 / 5 = 98,8552 %

Dari *Data* uji tabel 4.2 menjelaskan bahwa aplikasi ini dibuat dengan orientasi layar portrait yang dapat berjalan 100 % pada *smartphone* yang mempunyai pixel sebesar 480 x 854, dan 1024 x 600. Hal ini terjadi karena dalam tampilan *layout* aplikasi ini, penulis hanya membuat *layout* yang berukuran 480 x 854, dan 1024 x 600 sehingga jika aplikasi ini dijalankan ke jenis *smartphone* yang berukuran *pixel* lain maka *layout*nya akan bergeser dengan tingkat akurasi sebesar 97.138 %. Rata – rata untuk keberhasilan Pengujian spesifikasi *hardware* ini terhadap 5 macam *device* yaitu mencapai 98,8552 %.

4.3.3 Pengujian *Operating System Android*

Pengujian *operating system android* merupakan pengujian aplikasi jawa wayang yang dilakukan pada jenis *operating system android* yang digunakan oleh *smartphone*. Pada pengujian ini menggunakan 5 macam *operating system android* dengan *type* yang berbeda.

Tabel 4.3 Tabel Pengujian *Operating System Android*

Pengujian <i>Operating System Android</i>				Tingkat Keakuratan dalam (%)
No.	<i>Android Version</i>	Type	Berjalannya Aplikasi	
1	<i>Frozen Yoghurt (Froyo)</i>	2.2.1	Y	100 %
2	<i>Gingerbread</i>	2.3.3	Y	100 %
3	<i>Honeycomb</i>	3.0.0	Y	100 %
4	<i>Ice Cream Sandwich</i>	4.0.0	Y	100 %
5	<i>Jelly Bean</i>	4.1.2	Y	100 %
Rata - rata				100 %

Keterangan :

Y = Berhasil

X = Gagal

Pada tabel 4.3 menjelaskan bahwa aplikasi wayang ini dapat berjalan di hampir semua *Operating system* antara versi *Froyo* (2.2.1) hingga ke versi *Jelly Bean* (4.1.2) dengan rata – rata dari tingkat keakuratan hingga mencapai 100 %.

4.3.4 Pengujian User

Pengujian *user* pada aplikasi *android* wayang ini dilakukan dengan memberikan beberapa pertanyaan kepada *user* yang didasarkan atas pengujian sistem aplikasi wayang. Pengujian *user* ini dilakukan kepada 20 orang responden untuk memberikan penilaian terhadap aplikasi *android* wayang ini. Adapun hasil dari pengujian *user* ini dapat dilihat pada tabel 4.4.

Tabel 4.4 Tabel Pengujian *User*

No.	Kriteria Penilaian	Keterangan			
		SB	B	C	K
1	Layout Antar Muka Aplikasi	17	3	-	-
2	Tentang Aplikasi	19	1	-	-
3	Kinerja Aplikasi	15	4	1	-
4	Tingkat Kemudahan Aplikasi	13	5	2	-
5	Fungsi Aplikasi Sebagai Media Edukasi Dalam Pengenalan Tokoh Pewayangan	18	2	-	-
Jumlah		82	15	3	0
Rata – rata dalam (%)		82%	15%	3%	0%

Keterangan :

SB : Sangat Baik

B : Baik

C : Cukup

K : Kurang

- : Tidak Ada Nilai

Berdasarkan pada tabel 4.4, hasil pendataan yang dapat diambil dari 20 orang responden adalah sebagai berikut:

1. *Layout* Antar Muka Aplikasi.

- a. SB : 17 responden $\frac{17}{20} \times 100\% = 85\%$
 b. B : 3 responden $\frac{3}{20} \times 100\% = 15\%$
 c. C : 0 responden 0%
 d. K : 0 responden 0%

2. Tentang Aplikasi

- a. SB : 19 responden $\frac{19}{20} \times 100 \% = 95 \%$
 b. B : 1 responden $\frac{1}{20} \times 100 \% = 5 \%$
 c. C : 0 responden 0 %
 d. K : 0 responden 0 %

3. Kinerja Aplikasi.

- a. SB : 15 responden $\frac{15}{20} \times 100 \% = 75 \%$
 b. B : 4 responden $\frac{4}{20} \times 100 \% = 20 \%$
 c. C : 1 responden $\frac{1}{20} \times 100 \% = 5 \%$
 d. K : 0 responden 0 %

4. Tingkat Kemudahan Aplikasi.

- a. SB : 13 responden $\frac{13}{20} \times 100 \% = 65 \%$
 b. B : 5 responden $\frac{5}{20} \times 100 \% = 25 \%$
 c. C : 2 responden $\frac{2}{20} \times 100 \% = 10 \%$
 d. K : 0 responden 0 %

5. Fungsi Aplikasi Sebagai Media Edukasi Pengenalan Tokoh Pewayangan.

- a. SB : 18 responden $\frac{18}{20} \times 100 \% = 90 \%$
 b. B : 2 responden $\frac{2}{20} \times 100 \% = 10 \%$
 c. C : 0 responden 0 %
 d. K : 0 responden 0 %

Dari *data* kuisisioner diatas dapat dianalisis bahwa:

1. Hasil analisis sebesar 85% dari jumlah responden berpendapat bahwa *layout* aplikasi ini Sangat Baik dan 15% berpendapat Baik.
 2. Hasil analisis sebesar 95% dari jumlah responden berpendapat bahwa aplikasi ini Sangat Baik dan 5% berpendapat Baik.
 3. Hasil analisis sebesar 75% dari jumlah responden berpendapat bahwa kinerja dari aplikasi ini Sangat Baik, 20% berpendapat Baik, dan 5% berpendapat Cukup.
-

4. Hasil analisis sebesar 65% dari jumlah responden berpendapat bahwa tingkat aplikasi ini Sangat Baik, 25% berpendapat Baik, dan 10% berpendapat Cukup.
 5. Hasil analisis sebesar 90% dari jumlah responden berpendapat bahwa fungsi aplikasi ini sebagai media pembelajaran menyatakan Sangat Baik dan 10% berpendapat Baik.
-

BAB V PENUTUP

5.1 Kesimpulan

Aplikasi wayang ini merupakan suatu aplikasi yang dibuat sebagai media pembelajaran bagi semua kalangan lapisan masyarakat khususnya kaum terpelajar. Agar nantinya masyarakat bisa tahu tentang sejarah, cerita, maupun tokoh wayang.

Setelah melalui beberapa tahapan perancangan, implementasi, dan pengujian aplikasi wayang ini dapat disimpulkan sebagai berikut :

1. Semua fungsi yang digunakan pada aplikasi ini sudah mencapai tingkat keakuratan 100 %.
2. Aplikasi wayang ini dapat diterima dengan sangat baik oleh semua kalangan masyarakat sebagai media edukasi dalam pengenalan tokoh pewayangan. Hal ini terbukti dari hasil kuisioner hingga mencapai 90 %.
3. Aplikasi wayang ini dapat dijalankan dengan sangat baik di *smartphone* yang mempunyai ukuran layar 480 x 854 *pixel*, dan 1024 x 600 *pixel* dengan tingkat keakuratan mencapai 100 %.
4. Aplikasi ini dapat dijalankan pada OS *android* antara versi *Froyo* (2.2.1) sampai versi *Jelly Bean* (4.1.2) dengan tingkat akurasi sebesar 100 %.

5.2 Saran

Dalam pembuatan aplikasi wayang sebagai media pembelajaran ini masih jauh dari sempurna. Karena masih banyaknya kekurangan - kekurangan yang perlu untuk ditambahkan antara lain :

1. Aplikasi wayang ini dapat dijalankan dengan baik pada *smartphone* yang mempunyai ukuran layar 480 x 854 *pixel*, dan 1024 x 600 *pixel*, sehingga untuk proses kedepannya aplikasi wayang ini dapat dijalankan pada *smartphone* yang mempunyai ukuran layar 240 x 432 *pixel*, 720 x 1280 *pixel*, dan 320 x 480 *pixel*.
2. Didalam aplikasi jawi wayang ini belum ada suara dan efek animasinya, sehingga untuk proses kedepannya dapat berikan suara dan efek animasi.

3. Didalam aplikasi jawi wayang ini belum ada informasi tentang alat musik pertunjukkan wayang dan jenis gending atau lagu pengiring pertunjukkan pewayangan, sehingga untuk proses kedepannya dapat diberikan informasi tentang alat musik pertunjukkan dan jenis gending atau lagu pengiring pertunjukkan wayang.
-

DAFTAR PUSTAKA

- [1] Arif Akbarul Huda. *Membuat Sendiri Aplikasi Android Untuk Pemula*. Yogyakarta.
- [2] Huda, Arif Akbarul. *24 Jam Pintar Pemrograman Android version 2.1*. Yogyakarta :PENERBIT ANDI, 2012.
- [3] Ivan Michel Siregar, S.T., M.T. 2011. *Membongkar Source Code Berbagai Aplikasi ANDROID*. Yogyakarta: GAVA MEDIA.
- [4] Kevin Levandi Temotius. 2011. *Penggunaan Ternary Search Tree dalam Melakukan Autocomplete*. Program Study Teknik Informatika. Bandung.
- [5] Muhardin Endy. 2008. *Materi Pelatihan Java Fundamental*. Jakarta.
- [6] Proboyekti Umi. 2004. *Dasar – Dasar Database*.
- [7] Solichin, Drs., Ki Waluyo. 2007. *Mengenal Tokoh Wayang Jilid I-IV*. Surakarta: CV. ASIH JAYA.
- [8] Supriyono, dkk. 17 Agustus 2008. *Pedalangan Untuk Sekolah Menengah Kejuruan*.
- [9] Yasasusastra, J. Syahban. 2011. *Mengenal Tokoh Pewayangan*. Yogyakarta: PUSTAKA MAHARDIKA.

LAMPIRAN



PERKUMPULAN PENGELOLA PENDIDIKAN UMUM DAN TEKNOLOGI NASIONAL MALANG
INSTITUT TEKNOLOGI NASIONAL MALANG

FAKULTAS TEKNOLOGI INDUSTRI
FAKULTAS TEKNIK SIPIL DAN PERENCANAAN
PROGRAM PASCASARJANA MAGISTER TEKNIK

PT. BNI (PERSERO) MALANG
BANK NIAGA MALANG

Kampus I : Jl. Bendungan Sigura-gura No. 2 Telp. (0341) 551431 (Hunting). Fax. (0341) 553015 Malang 65145
Kampus II : Jl. Raya Karanglo, Km 2 Telp. (0341) 417636 Fax. (0341) 417634 Malang

Malang, 21 Oktober 2013

Nomor : ITN-70/INF/TA/2013
Lampiran : —
Perihal : Bimbingan Skripsi

Kepada : Yth. Bpk/Ibu Ir. Yusuf Ismail Nakhoda, MT
Dosen Pembina Program Studi Teknik Informatika S-1
Institut Teknologi Nasional
Malang

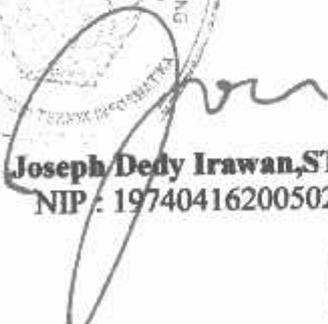
Dengan Hormat,
Sesuai dengan permohonan dan persetujuan dalam proposal skripsi untuk mahasiswa :

Nama : ACHMAD BISRI ISMAIL
Nim : 1018010
Prodi : Teknik Informatika S-1
Fakultas : Teknologi Industri

Maka dengan ini pembimbingan kami serahkan sepenuhnya kepada Saudara/i selama waktu 6 (enam) bulan, terhitung mulai tanggal :

21 OKTOBER 2013 S/D 21 MARET 2013

Sebagai satu syarat untuk menempuh Ujian Akhir Sarjana Teknik, Program Studi Teknik Informatika S-1.
Demikian agar maklum dan atas perhatian serta bantuannya kami sampaikan terima kasih.

Mengetahui
Program Studi Teknik Informatika S-1
Ketua,

Joseph Dedy Irawan, ST., MT.
NIP. 197404162005021002

Form S-4a



INSTITUT TEKNOLOGI NASIONAL MALANG

Fakultas Teknologi Nasional Malang

Program Studi Teknik Informatika S1

FORMULIR BIMBINGAN SKRIPSI

Nama : Achmad Bisri Ismail
NIM : 1018010
Masa Bimbingan : 21 Oktober 2013 – 21 Maret 2014
Judul Skripsi : Rancang Bangun Aplikasi Edukasi Wayang Sebagai Media Pengenalan Tokoh Pewayangan Berbasis Android

No.	TANGGAL	URAIAN	PARAF PEMBIMBING
1	10-01-2014	Demo Program	
2	13-01-2014	Revisi Program	
3	14-01-2014	Revisi Program	
3	16-01-2014	Revisi Program	
3	16-01-2014	Bimbingan Jurnal Seminar Hasil	
4	17-01-2014	Bimbingan Laporan Bab 1-5	
5	12-02-2014	Acc laporan skripsi	

Malang, 14 Februari 2014
Dosen Pembimbing

Ir. Yusuf Ismail Nakhoda, MT.

NIP.Y.1018800189



PERKUMPULAN PENGELOLA PENDIDIKAN UMUM DAN TEKNOLOGI NASIONAL MALANG
INSTITUT TEKNOLOGI NASIONAL MALANG

FAKULTAS TEKNOLOGI INDUSTRI
FAKULTAS TEKNIK SIPIL DAN PERENCANAAN
PROGRAM PASCASARJANA MAGISTER TEKNIK

PT. BNI (PERSERO) MALANG
BANK NIAGA MALANG

Kampus I : Jl. Bendungan Sigura-gura No. 2 Telp. (0341) 551431 (Hunting), Fax. (0341) 553015 Malang 65145
Kampus II : Jl. Raya Karanglo, Km 2 Telp. (0341) 417836 Fax. (0341) 417834 Malang

Malang, 21 Oktober 2013

Nomor : ITN-70/INF/TA/2013
Lampiran : —
Perihal : Bimbingan Skripsi

Kepada : Yth. Bpk/Ibu **Suryo Adi Wibowo,ST.MT**
Dosen Pembina Program Studi Teknik Informatika S-1
Institut Teknologi Nasional
Malang

Dengan Hormat,
Sesuai dengan permohonan dan persetujuan dalam proposal skripsi untuk mahasiswa :

Nama : ACHMAD BISRI ISMAIL
Nim : 1018010
Prodi : Teknik Informatika S-1
Fakultas : Teknologi Industri

Maka dengan ini pembimbingan kami serahkan sepenuhnya kepada Saudara/i selama waktu 6 (enam) bulan, terhitung mulai tanggal :

21 OKTOBER 2013 S/D 21 MARET 2013

Sebagai satu syarat untuk menempuh Ujian Akhir Sarjana Teknik, Program Studi Teknik Informatika S-1.
Demikian agar maklum dan atas perhatian serta bantuannya kami sampaikan terima kasih.

Mengetahui
Program Studi Teknik Informatika S-1
Ketua,

Joseph Dedy Irawan,ST., MT.
NIP : 197404162005021002

Form S-4a



INSTITUT TEKNOLOGI NASIONAL MALANG
Fakultas Teknologi Nasional Malang
Program Studi Teknik Informatika S1

FORMULIR BIMBINGAN SKRIPSI

Nama : Achmad Bisri Ismail
NIM : 1018010
Masa Bimbingan : 21 Oktober 2013 – 21 Maret 2014
Judul Skripsi : Rancang Bangun Aplikasi Edukasi Wayang Sebagai Media Pengenalan Tokoh Pewayangan Berbasis Android

No.	TANGGAL	URAIAN	PARAF PEMBIMBING
1	20-11-2013	Bimbingan Laporan Bab 1-3	
2	22-11-2013	Revisi Laporan Bab 1-3	
3	28-11-2013	Revisi Laporan Bab 1-3	
4	19-12-2013	Demo Program	
5	15-01-2014	Bimbingan Laporan Bab 4-5	
6	16-01-2014	Revisi Laporan Bab 4-5	
5	16-01-2014	Bimbingan Jurnal Seminar Hasil	
6	17-01-2014	Revisi Jurnal Seminar Hasil	
7	18-01-2014	Revisi Jurnal Seminar Hasil	
8	11-02-2014	Bimbingan Laporan Bab 1-5	

Malang, 14 Februari 2014
Dosen Pembimbing

Suryo Adi Wibowo, ST., MT.
NIP : 1031000438



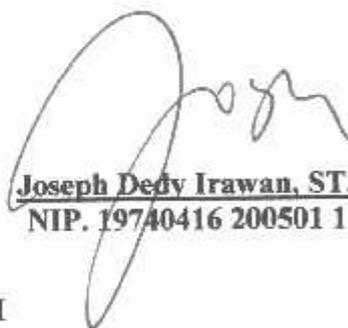
**BERITA ACARA UJIAN SKRIPSI
FAKULTAS TEKNOLOGI INDUSTRI**

Nama : Achmad Bisri Ismail
NIM : 10.18.010
Program Studi : Teknik Informatika S1
Judul : Rancang Bangun Aplikasi Edukasi Wayang Sebagai Media
Pengenalan Tokoh Pewayangan Berbasis Android

Dipertahankan dihadapan Majelis Penguji Skripsi Jenjang Strata Satu (S-1) pada:

Hari : Selasa
Tanggal : 18 Februari 2014
Nilai : 86.87 (A)

**Panitia Penguji Skripsi,
Ketua Majelis Penguji**


Joseph Dedy Irawan, ST, MT
NIP. 19740416 200501 1 002

Dosen Penguji I


Michael Ardita, ST, MT.
NIP. P. 103 100 0 434

Dosen Penguji II


Ali Mahmudi, B.Eng, PhD
NIP. P. 103 100 0 429



**INSTITUT TEKNOLOGI NASIONAL
FAKULTAS TEKNOLOGI INDUSTRI
PROGRAM STUDI TEKNIK INFORMATIKA S-1
Jl. Karanglo KM.2, Malang**

FORMULIR PERBAIKAN SKRIPSI

Dalam pelaksanaan ujian skripsi jenjang Strata Satu (S-1) Program Studi Teknik Informatika, maka perlu adanya perbaikan skripsi untuk mahasiswa:

Nama : Achmad Bisri Ismail
NIM : 10.18.010
Program Studi : Teknik Informatika S1
Judul : Rancang Bangun Aplikasi Edukasi Wayang Sebagai Media Pengenalan Tokoh Pewayangan Berbasis Android

No.	Penguji	Tanggal	Uraian	Paraf
1	Penguji I	18 Februari 2014	- Garis bawah (link) pada Bab II perlu dibuang.	
2	Penguji II	18 Februari 2014	- Tambahkan Bathara Guru, Bathara Ismaya, Syiwa, dan Saraswati. - Lampirkan 2 questionnaire yang sudah terisi. - Lampirkan 2 script program.	

Dosen Penguji I

Michael Ardita, ST, MT.
NIP. P. 103 100 0 434

Dosen Penguji II

Ali Mahmudi, B.Eng, PhD
NIP. P. 103 100 0 429

Mengetahui

Dosen Pembimbing I

Ir. Yusuf Ismail Nakhoda, MT.
NIP. Y. 1018800189

Dosen Pembimbing II

Suryo Adi Wibowo, ST, MT.
NIP.P. 1031000438

ANGKET SKRIPSI
ACHMAD BISRI ISMAIL (1018010)

Judul : Rancang Bangun Aplikasi Edukasi Wayang Sebagai Media Pengenalan
Tokoh Pewayangan Berbasis Android

Nama Responden : Liosih
Alamat : Desa Kulur
Pekerjaan : Bagong

1. Bagaimana menurut anda tentang layout antar muka aplikasi jawi wayang ini ?
a. Sangat baik
 b. Baik
c. Cukup
d. Kurang
2. Bagaimana menurut anda mengenai aplikasi ini ?
 a. Sangat baik
b. Baik
c. Cukup
d. Kurang
3. Bagaimana menurut anda mengenai kinerja aplikasi ini ?
a. Sangat baik
 b. Baik
c. Cukup
d. Kurang
4. Bagaimana menurut anda mengenai tingkat kemudahan aplikasi ini ?
a. Sangat baik
 b. Baik
c. Cukup
d. Kurang
5. Seberapa baikkah menurut anda tentang fungsi aplikasi edukasi wayang ini sebagai media pengenalan tokoh pewayangan ?
a. Sangat baik
 b. Baik
c. Cukup
d. Kurang

Tanda Tangan



SCRIPT PROGRAM

Nama Class	Wayang_home_Ac.java
Script	<pre>package com.example.abimanyu; import com.example.a.R; import android.app.Activity; import android.app.AlertDialog; import android.content.DialogInterface; import android.content.Intent; import android.os.Bundle; import android.view.Menu; import android.view.View; import android.view.Window; import android.widget.Toast; import android.widget.ImageButton; public class Wayang_home_Ac extends Activity implements View.OnClickListener { ImageButton bAlert, bToast; /** Called when the activity is first created. */ @Override public void onCreate(Bundle savedInstanceState) { super.onCreate(savedInstanceState); this.requestWindowFeature(Window.FEATURE_NO_TITLE); setContentView(R.layout.home); /** ngelink ke Menu Alhamdulillah */ ImageButton menu = (ImageButton) findViewById(R.id.MENU_Home); menu.setOnClickListener(new View.OnClickListener() { public void onClick(View Menu) { // TODO Auto-generated method stub Intent myIntent = new Intent(Menu.getContext(), menu_Ac.class); startActivityForResult(myIntent, 100); } }); /** ngelink ke About Alhamdulillah */ ImageButton pindah = (ImageButton) findViewById(R.id.ABOUT_Home2); pindah.setOnClickListener(new View.OnClickListener() { @Override public void onClick(View About) { // TODO Auto-generated method stub Intent myIntent = new Intent(About.getContext(),</pre>

```

        about_Ac.class);
        startActivityForResult(myIntent,
        100);
    }
});

/** ngelink ke help Alhamdulillah */
ImageButton help = (ImageButton)
findViewById(R.id.HELP_Home);
help.setOnClickListener(new
View.OnClickListener() {
    @Override
    public void onClick(View Help) {
        // TODO Auto-generated method stub
        Intent myIntent = new
        Intent(Help.getContext(),
        help_Ac.class);
        startActivityForResult(myIntent,
        100);
    }
});

/** untuk Keluar Alhamdulillah */
bAlert = (ImageButton)
findViewById(R.id.EXIT_Home);
bAlert.setOnClickListener(this);
bToast = (ImageButton)
findViewById(R.id.EXIT_Home);
bToast.setOnClickListener(this);
}
/** jarak */

/** keluaz_1 Alhamdulillah */
@Override
public void onClick(View view) {
    if (view == bAlert) {
        AlertDialog.Builder ad = new
        AlertDialog.Builder(this);
        ad.setMessage("Apakah Anda ingin
        keluar?");
        ad.setPositiveButton("Ya", new
        DialogInterface.OnClickListener() {
            public void onClick(DialogInterface
            dialog, int id) {
                // closeDialog.this.finish();
                Intent exit = new
                Intent(Intent.ACTION_MAIN);
                exit.addCategory(Intent.CATEGORY_HOME);
                exit.setFlags
                Intent.FLAG_ACTIVITY_NEW_TASK);
                startActivity(exit);
            }
        });
        super.onDestroy();
        ad.setNegativeButton("Tidak",

```

	<pre> new DialogInterface.OnClickListener() { public void onClick(DialogInterface dialog, int id) { dialog.cancel(); } }); ad.show(); } else { Toast.makeText(this, "Anda memilih tombol TCAST", Toast.LENGTH_SHORT).show(); } } @Override public boolean onCreateOptionsMenu(Menu menu) { // Inflate the menu; this adds items to the action bar if it is present. getMenuInflater().inflate(R.menu.activity_main, menu); return true; } } </pre>
Nama Layout	<p style="text-align: center;">home.xml</p>
Script	<pre> <RelativeLayout xmlns:android="http://schemas.android.com/apk/res/ android" xmlns:tools="http://schemas.android.com/tools" android:layout_width="match_parent" android:layout_height="match_parent" android:paddingBottom="@dimen/activity _vertical_margin" android:paddingLeft="@dimen/activity _horizontal_margin" android:paddingRight="@dimen/activity _horizontal_margin" android:paddingTop="@dimen/activity _vertical_margin" android:background="@drawable/home" tools:context=".Wayang_home_Ac" > <ImageButton android:id="@+id/MENU_Home" android:layout_width="wrap_content" android:layout_height="wrap_content" android:layout_alignParentBottom="true" android:layout_alignParentLeft="true" android:layout_marginBottom="9dp" android:layout_marginLeft="11dp" android:background="@null" android:contentDescription="@string/menu" </pre>

```
android:src="@drawable/icon_menu_kecil" />
```

```
<ImageButton  
    android:id="@+id/ABOUT_Home2"  
    android:layout_width="wrap_content"  
    android:layout_height="wrap_content"  
    android:layout_alignTop="@+id/MENU_Home"  
    android:layout_toRightOf="@+id/MENU_Home"  
    android:layout_marginLeft="11dp"  
    android:background="@null"  
    android:contentDescription="@string/about"  
    android:src="@drawable/icon_about_kecil" />
```

```
<ImageButton  
    android:id="@+id/HELP_Home"  
    android:layout_width="wrap_content"  
    android:layout_height="wrap_content"  
    android:layout_alignTop="@+id/ABOUT_Home2"  
    android:layout_toRightOf="@+id/ABOUT_Home2"  
    android:layout_marginLeft="11dp"  
    android:background="@null"  
    android:contentDescription="@string/sound"  
    android:src="@drawable/icon_help_kecil" />
```

```
<ImageButton  
    android:id="@+id/EXIT_Home"  
    android:layout_width="wrap_content"  
    android:layout_height="wrap_content"  
    android:layout_alignTop="@+id/HELP_Home"  
    android:layout_toRightOf="@+id/HELP_Home"  
    android:layout_marginLeft="11dp"  
    android:background="@null"  
    android:contentDescription="@string/exit"  
    android:src="@drawable/icon_exit_kecil" />
```

```
</RelativeLayout>
```

Tampilan
Program



Nama Class	menu_Ac.java
Script	<pre> package com.example.abimanyu; import com.example.a.R; import android.app.Activity; import android.content.Context; import android.content.Intent; import android.content.res.TypedArray; import android.os.Bundle; import android.view.View; import android.view.ViewGroup; import android.view.Window; import android.widget.BaseAdapter; import android.widget.Gallery; import android.widget.ImageButton; import android.widget.ImageView; public class menu_Ac extends Activity { Integer[] pics = { R.drawable.gallerya, R.drawable.galleryb, R.drawable.galleryc, R.drawable.galleryd, R.drawable.gallerye }; ImageView imageView; /** Called when the activity is first created. */ @Override public void onCreate(Bundle savedInstanceState) { super.onCreate(savedInstanceState); this.requestWindowFeature(Window.FEATURE_NO_TITLE); setContentView(R.layout.menu); /** ngelink Home Alhamdulillah */ ImageButton home = (ImageButton) findViewById(R.id.Button_Menu_toHome); home.setOnClickListener(new View.OnClickListener() { public void onClick(View Home) { // TODO Auto-generated method stub Intent myIntent = new Intent (Home.getContext(), Wayang_home_Ac.class); startActivityForResult (myIntent, 100); finish();//menghapus history class ini } }); </pre>

```

Gallery ga = (Gallery)
findViewById(R.id.gallery_menu);
ga.setAdapter(new ImageAdapter(this));

/** ngelink ke sejarah */
ImageButton sejarah = (ImageButton)

findViewById(R.id.SEJARAH_menu);
sejarah.setOnClickListener(new
View.OnClickListener() {
    public void onClick(View Sejarah) {
        // TODO Auto-generated method stub
        Intent myIntent = new

        Intent (Sejarah.getContext(),
        sejarah_Ac.class);
        startActivityForResult (myIntent,
        100);
        finish();//menghapus history class
        ini
    }
});

/**ngelink ke cerita */
ImageButton cerita = (ImageButton)

findViewById(R.id.CERITA_menu);
cerita.setOnClickListener(new
View.OnClickListener() {
    public void onClick(View Cerita) {
        // TODO Auto-generated method stub
        Intent myIntent = new

        Intent (Cerita.getContext(),
        Cerita_2pusat_cerita_Ac.class);
        startActivityForResult (myIntent,
        100);
        finish();//menghapus history class
        ini
    }
});

/** ngelink Tokoh Alhamdulillah */
ImageButton tokoh = (ImageButton)
        findViewById(R.id.TOKOH_menu);
tokoh.setOnClickListener(new
View.OnClickListener() {
    public void onClick(View Tokoh) {
        // TODC Auto-generated method stub
        Intent myIntent = new

        Intent (Tokoh.getContext(),
        Tokoh_3pusat_penokohan_Ac.class);

```

```

        startActivityForResult(myIntent,
        100);
        finish();//menghapus history class
        ini
    }
});

/** ngelink Soal Alhamdulillah */
ImageButton soal = (ImageButton)
        findViewById(R.id.SOAL_menu);
soal.setOnClickListener(new
View.OnClickListener() {
    public void onClick(View Tokoh) {
        // TODO Auto-generated method stub
        Intent myIntent = new

        Intent(Tokoh.getContext(),
        soal_quiz_UTAMA.class);
        startActivityForResult(myIntent,
        100);
        finish();//menghapus history class
        ini
    }
});

/**jarak*/
}
public class ImageAdapter extends BaseAdapter {
    private Context ctx;
    int imageBackground;
    public ImageAdapter(Context c) {
        ctx = c;
        TypedArray ta =

        obtainStyledAttributes
        R.styleable.Gallery1);
        imageBackground = ta.getResourceId(
R.styleable.Gallery1_android_galleryItemBackgroun
d,1);
        ta.recycle();
    }

    public int getCount() {
        return pics.length;
    }
    public Object getItem(int arg0) {
        return arg0;
    }
    public long getItemId(int arg0) {
        return arg0;
    }
    public View getView(int arg0, View arg1,
    ViewGroup arg2) {
        ImageView iv = new ImageView(ctx);

```

	<pre> iv.setImageResource (pics [arg0]); iv.setScaleType (imageView.ScaleType.FIT_XY); iv.setLayoutParams (new Gallery.LayoutParams (299, 221)); iv.setBackgroundResource (imageBackground); return iv; } } } </pre>
Nama Layout	menu.xml
Script	<pre> <RelativeLayout xmlns:android="http://schemas.android.com/apk/ res/android" xmlns:tools="http://schemas.android.com/tools" android:layout_width="match_parent" android:layout_height="match_parent" android:paddingBottom="@dimen/activity _vertical_margin" android:paddingLeft="@dimen/activity _horizontal_margin" android:paddingRight="@dimen/activity _horizontal_margin" android:paddingTop="@dimen/activity _vertical_margin" android:background="@drawable/menu" android:gravity="center_horizontal" android:orientation="vertical" > <ScrollView android:layout_width="fill_parent" android:layout_height="fill_parent" android:orientation="vertical" > <LinearLayout android:layout_width="fill_parent" android:layout_height="wrap_content" android:orientation="vertical" > <Gallery android:id="@+id/gallery_menu" android:layout_marginTop="3dp" android:layout_width="320dp" android:layout_height="225dp" android:contentDescription="@string/ hello_world" android:scaleType="fitXY" > </Gallery> </LinearLayout> </ScrollView> <ImageButton </pre>

```
android:id="@+id/SOAL_menu"  
android:layout_width="wrap_content"  
android:layout_height="wrap_content"  
android:layout_alignParentBottom="true"  
android:layout_alignParentRight="true"  
android:layout_marginBottom="35dp"  
android:background="@null"  
android:src="@drawable/tsoal" />
```

```
<ImageButton  
android:id="@+id/CERITA_menu"  
android:layout_width="wrap_content"  
android:layout_height="wrap_content"  
android:layout_alignParentBottom="true"  
android:layout_alignParentLeft="true"  
android:layout_marginBottom="35dp"  
android:background="@null"  
android:src="@drawable/tcerita" />
```

```
<ImageButton  
android:id="@+id/SEJARAH_menu"  
android:layout_width="wrap_content"  
android:layout_height="wrap_content"  
android:layout_centerVertical="true"  
android:layout_toLeftOf="@+id/SOAL_menu"  
android:background="@null"  
android:focusableInTouchMode="false"  
android:src="@drawable/tsejarah" />
```

```
<ImageButton  
android:id="@+id/Buton_Menu_toHome"  
android:layout_width="wrap_content"  
android:layout_height="wrap_content"  
android:layout_alignParentBottom="true"  
android:layout_alignParentLeft="true"  
android:background="@null"  
android:contentDescription="@null"  
android:src="@drawable/back" />
```

```
<ImageButton  
android:id="@+id/TOKOH_menu"  
android:layout_width="wrap_content"  
android:layout_height="wrap_content"  
android:layout_alignParentRight="true"  
android:layout_centerVertical="true"  
android:background="@null"  
android:focusableInTouchMode="false"  
android:src="@drawable/ttokoh" />
```

```
</RelativeLayout>
```

Tampilan
Program

