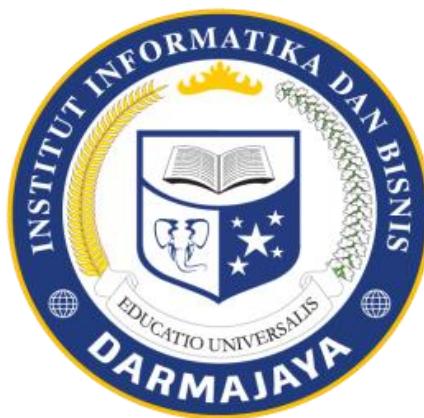


**PERANGKAT LUNAK E-LEARNING ELEMEN-ELEMEN SENI
TEATER BERBASIS ANDROID**

SKRIPSI

Diajukan Sebagai Salah Satu Syarat untuk Mencapai Gelar
SARJANA KOMPUTER
Pada Jurusan Teknik Informatika
Institut Informatika dan Bisnis Darmajaya



Disusun Oleh:
THIU AAN RIAN HERMAWAN
NPM. 1411010098

JURUSAN TEKNIK INFORMATIKA
INSTITUT INFORMATIKA DAN BISNIS DARMAJAYA
BANDAR LAMPUNG

2019

PERNYATAAN



Saya yang bertanda tangan dibawah ini, menyatakan bahwa skripsi yang saya buat ini adalah hasil karya saya sendiri, tidak terdapat karya yang pernah diajukan untuk memperoleh gelar kesarjanaan di suatu perguruan tinggi atau karya yang pernah ditulis atau diterbitkan orang lain kecuali yang secara tertulis diacu dalam naskah ini dan disebutkan dalam daftar pustaka. Karya ini adalah milik saya dan pertanggung jawaban sepenuhnya berada di pundak saya.

Bandar Lampung, 6 Maret 2019



Thiu Aan Rian Hermawan
NPM. 1411010098

HALAMAN PERSETUJUAN

Judul Skripsi

: PERANGKAT LUNAK E-LEARNING

**ELEMEN-ELEMEN SENI TEATER BERBASIS
ANDROID**

Nama Mahasiswa

: Thiu Aan Rian Hermawan

No. Pokok Mahasiswa

: 1411010098

Jurusan

: Teknik Informatika

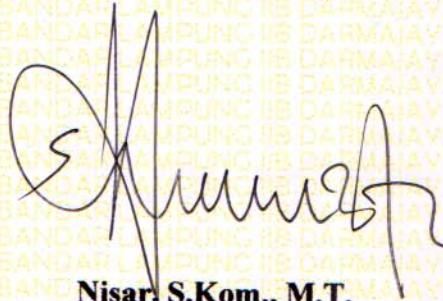
Dosen Pembimbing

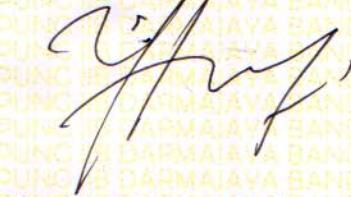
Nisar, S.Kom., M.T.
NIK. 00130100

Ketua Jurusan

Yuni Arkhiansyah, S.Kom., M.Kom.
NIK. 00480802







HALAMAN PENGESAHAN

Telah Diuji dan Dipertahankan Didepan Tim Pengaji Skripsi
JurusanTeknik Informatika Institut Informatika dan Bisnis Darmajaya
Bandar Lampung dan Dinyatakan Diterima untuk
Memenuhi Syarat Guna Memperoleh
Gelar Sarjana Komputer

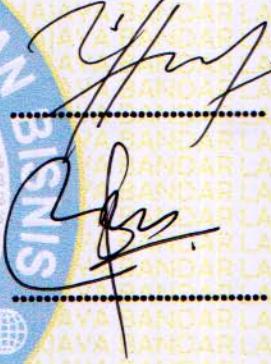
Mengesahkan

1. Tim Pengaji

Ketua : Yuni Arkhiansyah, S.Kom, M.T.I

Anggota : Yuni Puspitasari, S.Kom, M.T.I

Tanda Tangan


.....

.....

2. Dekan Fakultas Ilmu Komputer

Sriyanto, S.Kom.,M.M.,Ph.D

NIK. 00210800

Tanggal Lulus Ujian Skripsi : 6 Maret 2019

PERSEMBAHAN

Semoga hasil karya pikiran ini dapat menjadi persembahan terbaikku untuk :

- Allah SWT Alhamdulillah, atas segala Nikmat, Rahmat, dan Kekuatan yang senantiasa engkau berikan.
- Ibu Heliana dan Bapak Umar Hamdani sembah sujud dan hormat ku, terima kasih atas cinta dan kasih sayangnya, serta do'a dan semangat yang telah ibu dan bapak berikan kepada ananda.
- Adikku tercinta Ferry Rendy Saputra serta seluruh keluarga dan saudara yang telah memberi semangat dan do'a sehingga laporan skripsi ini dapat terselesaikan dengan baik.
- Untuk Dosen Pembimbing SKRIPSI (Bapak Nisar, S.kom., M.T., dan Ketua Jurusan Teknik Informatika (Bpk. Yuni Arkhiansyah, S.Kom.,M.Kom.) .
- Keluarga besar Unit Kemahasiswaan Bidang Seni Lampung, Seniman Kampus Lampung (SIKAM) dan Dewan Kesenian Lampung (DKL) terkhusus untuk keluarga besar ku UKMBS Komunitas Biroe Institut Informatika Dan Bisnis Darmajaya.
- Teman-teman satu genggaman yang erat dan terkhususnya angkatan 2014 Teknik Informatika selaku keluarga besar Angkatan sekaligus sebagai ruang belajar bersama angkatan 2014 yang di namai saveNet.

MOTTO

**“Adanya sebuah penyelesaian bukan berarti untuk
sebuah akhiran dalam sebuah berproses”**

**“Berbagi dunia dengan ciptaan lainya bukan ciptaan
yang di bagikan dalam dunia nya”**

RIWAYAT HIDUP

1. Identitas

- a. Nama : THIU AAN RIAN HERMAWAN
- b. NPM : 1411010098
- c. Tempat / Tanggal Lahir : TASIK RUGUK, 09 JUNI 1994
- d. Agama : ISLAM
- e. Alamat : JL. Lintas Pantai Timur, Dusun Tasik Desa
Ruguk RT/RW 021/010 Kec. Ketapang,
Kab. Lampung Selatan, Prov. Lampung
- f. Suku : Lampung
- g. Kewarganegaraan : Indonesia
- h. E-Mail : thuaanrianhermawan@gmail.com
- i. No. Telp/HP : 0822 – 7960 - 6541

2. Riwayat Pendidikan Telah Ditempuh

- a. Pendidikan Sekolah Dasar MI Al – Khoiriyah , tamat tahun2006.
- b. Sekolah Menengah Pertama MTs Darul Kamal tamat tahun 2009,
- c. Sekolah Menengah Atas SMA Islam Ma’arif tamat tahun 2012.
- d. Pada tahun 2014 Penulis diterima di IIB Darmajaya Jurusan S-1
Teknik Informatika.

Yang menyatakan
Bandar Lampung, 6 Maret 2019

Thiu Aan Rian Hermawan
NPM. 1411010098

ABSTRAK

PERANGKAT LUNAK E-LEARNING ELEMEN -ELEMEN SENI TEATER BERBASIS ANDROID

Oleh

**Thiu Aan Rian Hermawan
1411010098**

Teater adalah seni yang dapat dikatakan sebagai seni campuran. Hal ini dikarenakan dalam seni teater di mana unsur-unsur seni lain seperti sastra, seni rupa, arsitektur, musik, dan tari masuk di dalamnya dan menciptakan sebuah karya seni yang disebut teater serta di butuhkanya tambahan ilmu pengetahuan atau wawasan yang akan di terapkan dari sumber pematerian yang pasti dan terpercaya.

Belajar menggunakan cara *e-learning* dapat membantu untuk mendapatkan sebuah informasi pada tahap pembelajaran seni teater hal ini dikarenakan *e-learning* merupakan pembelajaran yang tidak harus di dalam suatu ruangan dan tetap dapat digunakan meskipun berbeda waktu dan tempat untuk mempelajari seni teater secara teori pada perangkat mobile bersistem opearasi Android dan aplikasi *e-learning* ini di bangun dengan bahasa pemrograman java menggunakan android studio sebagai pengembangan aplikasi android. Pengujian di lakukan dengan *Black Box Testing* yaitu berfokus pada persyaratan fungsional dari perangkat lunak dan untuk mengetahui kesalahan fungsi terhadap perangkat lunak yang di bangun. Hasil penelitian adalah aplikasi yang di bangun dapat menjadi alternatif untuk mempelajari seni teater secara materi meskipun tidak dalam satu ruangan antara guru dan siswa tetapi dapat berinteraksi dalam bentuk text secara *real time* dalam proses pembelajaran seni teater.

Kata Kunci : *Android, E-Learning, Seni Teater.*

ABSTRACT

ANDROID-BASED THEATER ARTS ELEMENTS E-LEARNING SOFTWARE

By:

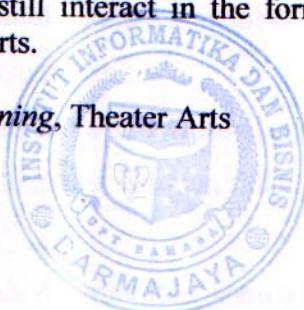
**Thiu Aan Rian Hermawan
1411010098**

Theater is an art that can also be said as a mixed art. This is because in the art of theater there are other elements of art such as literature, fine arts, architecture, music, and dance in it and create knowledge or insights that will be applied from definite and reliable soldering sources.

Learning to use *e-learning* can help to get information at the theater arts learning stage. this is because e-learning is learning that does not have to be in a room and can still be used even though different times and places to learn theater art in theory on a mobile operating system with android operating system and this *e-learning* application was built with the *java* programming language using android studio as an android application development. Testing is done with *Black Box Testing* which focuses on the functional requirements of the software and to find out the malfunction of the software being built.

The result of the study was an application that was built to be an alternative to learning theater arts materially even though it is not a relationship between teachers and students can still interact in the form of text in real time in the learning process of theater arts.

Keywords: *Android, E-Learning, Theater Arts*



PRAKATA

Assalamu'alaikum Wr.Wb

Segala puji syukur saya panjatkan kepada Allah SWT berkat-Nya sehingga penulis dapat menyelesaikan penyusunan skripsi ini.

Penulis menyadari bahwa dalam penyusunan skripsi ini tidak lepas dari bantuan semua pihak. Oleh karena itu dalam kesempatan ini penulis menyampaikan terima kasih yang sebesar-besarnya kepada :

1. Bapak DR. Andi Desfiandi, S.E., M.A. Selaku Ketua Yayasan Alfian Husin.
2. Bapak Ir. Firmansyah Y. Alfian, MBA., MSc. Selaku Rektor IIB Darmajaya.
3. Bapak Dr. RZ.Abdul Aziz, S.T., M.T., Ph.D. Selaku Wakil Rektor I Bidang Akademik dan Riset Institut Informatika dan Bisnis Darmajaya.
4. Bapak Yuni Arkhiansyah, S.Kom., M.Kom. Selaku Ketua Jurusan Teknik Informatika.
5. Bapak Nisar, S.Kom., M.T., Selaku Dosen Pembimbing yang telah membantu membimbing dan mengarahkan serta memberikan petunjuk sehingga skripsi ini dapat diselesaikan dengan baik.
6. Para dosen, staf dan karyawan IIB Darmajaya Bandar Lampung yang telah memberi bantuan baik langsung maupun tidak langsung selama saya menjadi mahasiswa.
7. Semua Pihak yang telah memberikan bantuan dan petunjuk sehingga saya dapat lebih mudah dalam menyusun skripsi ini.

Demikian banyaknya bantuan berbagai pihak kepada penulis, tentunya tidak menutup kemungkinan bahwa hasil dari laporan ini masih ada kekurangan dan masih jauh dari kesempurnaan. Oleh karena itu, kritik dan saran guna perbaikan di masa depan adalah mutlak sangat penulis perlukan. Semoga Laporan Skripsi ini bermanfaat bagi setiap pembacanya.

Wassalamu'alaikum Wr.Wb

DAFTAR ISI

	Halaman
HALAMAN JUDUL	i
PERNYATAAN.....	ii
HALAMAN PERSETUJUAN	iii
HALAMAN PENGESAHAN.....	iv
PERSEMBERAHAN.....	v
MOTTO	vi
RIWAYAT HIDUP	vii
ABSTRAK	viii
ABSTRACT	ix
PRAKATA	x
DAFTAR ISI.....	xi
DAFTAR GAMBAR.....	xvi
DAFTAR TABEL	xviii

BAB I PENDAHULUAN

1.1 Latar Belakang Masalah.....	1
1.2 Identifikasi Masalah.....	2

1.3 Rumusan Masalah	2
1.4 Batasan Masalah	3
1.5 Tujuan Penelitian	3
1.6 Manfaat Penelitian	3
1.7 Sistematika Penulisan	4
BAB I PENDAHULUAN.....	4
BAB II LANDASAN TEORI.....	4
BAB III METODOLOGI PENELITIAN	4
BAB IV HASIL PENELITIANDAN PEMBAHASAN.....	4
BAB V KESIMPULAN DAN SARAN	5
DAFTAR PUSTAKA	5
LAMPIRAN.....	5

BAB II LANDASAN TEORI

2.1 Teater	6
2.2 <i>E-Learning</i>	6
2.3 Multimedia	7
2.3.1 Pengertian Multimedia.....	7
2.3.2 Objek-Objek Multimedia	7
2.4 Android	8
2.4.1 Versi Android.....	9
2.4.2 Statistik Distribusi OS Android	10
2.5 Perangkat Lunak Pengembangan Sistem	11
2.5.1 Android Studio.....	11
2.5.2 Elemen Android	12

2.5.3 <i>Firebase</i>	13
2.5.4 <i>Adobe Photoshop</i>	15
2.6 Model Prototype.....	15
2.7 Unified Modelling Language (UML)	16
2.7.1 <i>Use Case Diagram</i>	17
2.7.2 <i>Activity Diagram</i>	18
2.7.3 <i>Sequence Diagram</i>	19
2.7.4 <i>Class Diagram</i>	19
2.8 Pengujian <i>Black Box Testing</i>	19
2.9 Penelitian Terdahulu	20

BAB III METODELOGI PENELITIAN

3.1 Metode Penelitian.....	26
3.1.1Teknik Pengumpulan data.....	26
3.2 Metode Pengembangan Perangkat Lunak.....	26
3.2.1Komunikasi	27
3.2.2Perencanaan Cepat	27
3.3 Pemodelan Desain.....	29
3.3.1Desain UML (<i>Unified Modelling Language</i>)	29
3.3.2Rancangan <i>Activity Diagram</i>	31
3.4 Basis Data	36
a. Sinkronisasi <i>real-time</i> untuk data JSON.....	36
b. Membangun aplikasi tanpa server	36
c. Dioptimalkan untuk pengguna offline	36
d. Keamanan berbasis pengguna yang kuat	36

e. Bagian dari <i>platform Firebase</i>	37
3.5 Desain Antar Muka Aplikasi	37
3.5.1 Rancangan Tampilan Halaman <i>Splash Screen</i>	37
3.5.2 Rancangan Tampilan Halaman Login Admin	37
3.5.3 Rancangan Tampilan Halaman Utama	38
3.5.4 Rancangan Tampilan Halaman Materi	39
3.5.5 Rancangan Tampilan Halaman Materi Detail.....	40
3.5.6 Rancangan Tampilan Halaman Hapus Materi	41
3.5.7 Rancangan Tampilan Halaman <i>Update Materi</i>	41
3.5.8 Rancangan Tampilan Halaman <i>Upload Materi</i>	42
3.5.9 Rancangan Tampilan Halaman <i>Chat Room</i>	42
3.5.10 Rancangan Tampilan Halaman Tentang Aplikasi	43
3.6 Pembentukan <i>Prototype</i>	43
3.6.1 Pembuatan aplikasi	43
3.6.2 Pengujian aplikasi	43
3.7 Penyerahan Sistem dan Umpam Balik	44

BAB IV HASIL PENELITIAN DAN PEMBAHASAN

4.1 Hasil Penelitian	45
4.2 Pembahasan.....	45
4.2.1 Instalasi Perangkat Lunak	45
4.2.2 Penggunaan Perangkat Lunak.....	45
4.3 Pengujian Sistem Perangkat Luna	55
4.3.1 Pengujian Instalasi	55
4.3.2 Pengujian Penggunaan	57

4.3.3 Pengujian Antarmuka.....	59
4.3.4 Pengujian Data Base	60
4.3.5 Pengujian Sistem Operasi Minimum	61
4.4 Kelebihan dan Kekurangan Perangkat Lunak.....	62
4.4.1Kelebihan dari Perangkat Lunak ini adalah sebagai berikut:.....	62
4.4.2Kelemahan dari Perangkat Lunak ini adalah	62

BAB V SIMPULAN DAN SARAN

5.1 Simpulan	63
5.2 Saran	63

DAFTAR PUSTAKA

LAMPIRAN

DAFTAR GAMBAR

Gambar	Halaman
Gambar 2.1 Evolusi Android.....	9
Gambar 2.2 Statistik distribusi OS Android.....	11
Gambar 2.3 Logo Android Studio	12
Gambar 2.4 logo firebase.....	13
Gambar 2.5 Diagram Prototype.....	16
Gambar 3.1 use case diagram admin	29
Gambar 3.2 use case diagram user	30
Gambar 3.3 Rancangan <i>Activity Diagram</i> Menu <i>Splash Screen Admin dan User</i>	
.....	31
Gambar 3.4 Rancangan <i>Activity Diagram</i> Menu <i>Login Admin</i>	32
Gambar 3.5 Rancangan <i>Diagram Menu</i> Artikel Admin dan User.....	32
Gambar 3.6 Rancangan <i>Diagram Menu</i> Pencarian Admin dan User	33
Gambar 3.7 Rancangan <i>Diagram Menu Chat</i> Admin dan User	33
Gambar 3.8 Rancangan <i>Diagram Menu</i> Tentang Admin dan User	34
Gambar 3.9 Rancangan <i>Diagram Menu Upload</i> Admin.....	34
Gambar 3.10 Rancangan <i>Diagram Menu Hapus</i> Admin	35
Gambar 3.11 Rancangan <i>Diagram Menu Update</i> Admin.....	35
Gambar 3.12 Tampilan ancangan halaman <i>SplashScreen admin</i>	37
Gambar 3.13 Tampilan rancangan halaman login admin.....	38
Gambar 3.14 Tampilan rancangan halaman menu utamaUser.....	38
Gambar 3.15 Tampilan rancangan halaman menu utama Admin	39
Gambar 3.16 Tampilan rancangan halaman materi Admin.....	39
Gambar 3.17 Tampilan rancangan halaman materi User	40
Gambar 3.18 Tampilan rancangan halaman materi detail Admin dan User	40
Gambar 3.19 Tampilan rancangan halaman Hapus materi Admin	41

Gambar 3.20 Tampilan rancangan halaman Update materi Admin	41
Gambar 3.21 Tampilan rancangan halaman Upload materi Admin.....	42
Gambar 3.22 Tampilan rancangan halaman Chat Room Admin dan User	42
Gambar 3.23 Tampilan rancangan halaman Tentang Aplikasi Admin danUser .	43
Gambar 4.1 gambar instalasi aplikasi.....	45
Gambar 4.2 Tampilan icon	46
Gambar 4.3 Tampilan halaman splash screen	46
Gambar 4.4 Tampilan halaman login	47
Gambar 4.5 Tampilan halaman Utama.....	47
Gambar 4.6 Tampilan halaman menu materi	48
Gambar 4.7 Tampilan halaman menu Detail.....	48
Gambar 4.8 Tampilan Halaman ChatRoom	49
Gambar 4.9 Tampilan Tentang	49
Gambar 4.10 Tampilan halam Hapus Materi	50
Gambar 4.11 Tampilan Halaman Update Materi	50
Gambar 4.12 Tampilan Halaman Upload Materi	51
Gambar 4.13 Tampilan Chat Room Admin.....	51
Gambar 4.14 Tampilan Icon	52
Gambar 2.15 Tampilan Halaman Splash Screen	52
Gambar 4.16 Tampilan Halaman Utama	53
Gambar 2.17 Tampilan Halaman Menu mater	53
Gambar 4.18 Tampilan halaman materi detail	54
Gambar 4.19 Tampilan Chat Room User	54
Gambar 4.20 Tampilan Tentang Aplikasi	55

DAFTAR TABEL

Tabel	Halaman
Tabel 2.1 Simbol Use Case Diagram.....	17
Tabel 2.2 Simbol Activity Diagram.....	18
Tabel 2.3 Penelitian Terdahulu.....	20
Tabel 4.1 pengujian instalasi	56
Tabel 4.2 Pengujian Pada Halaman Chat Room	57
Tabel 4.3 antarmuka perangkat lunak	59
Tabel 4.4 akses database perangkat lunak.....	60
Tabel 4.5 pengujian sistem operasi di bawah minimum	61

BAB I

PENDAHULUAN

1.1 Latar Belakang Masalah

Pada umumnya setiap individu manusia sudah mengenal seni yang di terapkan di dalam kehidupan sehari hari, seni itu sendiri sudah menjadi kebutuhan manusia dan sudah ada pada diri manusia sejak lahir. Kekayaan seni di setiap wilayah memiliki keanekaragaman dan keunikan tersendiri dari berbagai suku bangsa dan masing-masing suku bangsa memiliki keanekaragaman seni budaya nya sendiri yang setiap seni budaya nya sendiri memiliki nilai yang sangat tinggi baik seperti dari bidang musik, rupa, tari, ataupun teater. tapi tidak jarang pula kesenian yang ada pada suatu dearah sering terlupakan bahkan sudah mulai punah karena kurang nya di kenal dan di lestariakan.

Teater adalah seni yang dapat dikatakan sebagai seni campuran. Hal ini dikarenakan dalam seni teater di mana unsur-unsur seni lain seperti sastra, seni rupa, arsitektur, musik, dan tari masuk di dalamnya dan menciptakan sebuah karya seni yang disebut teater dan proses untuk mempelajari bidang seni teater membutuhkan waktu yang cukup lama dalam proses berlatih serta di butuhkanya tambahan ilmu pengetahuan atau wawasan yang akan di terapkan dari sumber pematerian yang pasti dan terpercaya, adapun kendala dalam proses pembelajaran secara materi untuk rancangan dan persiapan pementasan seni teater adalah minimnya sumber belajar seperti : buku, *workshop* dan artikel dari sebuah *website*.

Disisi lain seseorang yang ingin mempelajari seni teater mempunyai kesibukan dalam beraktivitas yang cukup padat, sehingga di perlukanya sebuah tempat sharing bersifat *real time* dan media perangkat lunak yang dapat mempelajari tentang seni di bidang teater dalam kesibukan sehari-hari yang cukup padat.

Belajar menggunakan cara *E-Learning* dapat membantu untuk mendapatkan informasi pada tahap pembelajaran seni teater hal ini dikarenakan *E-Learning* merupakan pembelajaran yang tidak harus di dalam suatu ruangan dan tetap dapat digunakan meskipun berbeda waktu dan tempat untuk mempelajari seni teater secara teori pada perangkat mobile bersistem opearasi Android.

Dengan menggunakan android yang sistem operasi untuk telepon seluler berbasis linux yang akan menjalankan perangkat lunak, sehingga perangkat lunak tersebut dapat di gunakan dengan semestinya seperti perangkat lunak untuk memudahkan mempelajari elemen-elemen seni teater meskipun memiliki aktivitas yang cukup padat.

Berdasarkan masalah yang telah di sampaikan di atas maka penulis mengangkat penelitian dengan “**PERANGKAT LUNAK E-LEARNING ELEMEN-ELEMEN SENI TEATER BERBASIS ANDROID**” sebagai alat bantu belajar dalam seni teater.

1.2 Identifikasi Masalah

Berdasarkan yang di dapat dari latar belakang maka di identifikasikan sebagai berikut :

- a. Minimnya sumber belajar seni teater dari dari sumber yang terpercaya.
- b. Sedikitnya sumber panduan dari orang khusus yang memang terbiasa bergelut dalam bidang tersebut.
- c. Kurangnya tempat sharing dalam forum untuk berdiskusi dalam mempelajari seni teater.

1.3 Rumusan Masalah

Berdasarkan identifikasi masalah yang ada maka rumusan masalah dari penelitian ini adalah :

- a. Bagaimana memberikan sebuah materi pembelajaran seni teater hasil dari sumber terpercaya.

- b. Bagaimana membangun sebuah tempat untuk berdiskusi (*sharing*) bersifat *real-time*.
- c. Bagaimana membangun sebuah perangkat lunak pembelajaran seni teater agar dapat membantu proses pembelajaran seni teater secara teori yang dapat digunakan dalam aktifitas sehari-hari.

1.4 Batasan Masalah

Berdasarkan rumusan masalah diatas diperoleh batasan masalah yaitu :

- a. Materi pembelajaran hanya di fokuskan pada materi pembelajaran seni teater.
- b. Materi pembelajaran bersifat *real-time*.
- c. Perangkat lunak yang di bangun berupa media pembelajaran seni teater menggunakan pemrograman *java script* pada android studio.
- d. Aplikasi yang di bangun dapat di ajalan dengan minimum versi *system operasi* android 6.0 dengan level API 23

1.5 Tujuan Penelitian

Berdasarkan rumusan masalah di atas, maka tujuan dari penelitian ini adalah :

- a. Untuk menambah pengetahuan dalam mempelajari seni teater secara teori yang di dapat dari banyak sumber terpercaya.
- b. Untuk memperkenalkan sekaligus mempelajari seni dalam bidang teater.
- c. Dengan membangun perangkat lunak pembelajaran seni teater yang bersifat *E-Learning*, maka dapat mempercepat proses pembelajaran.

1.6 Manfaat Penelitian

Manfaat dari penelitian ini adalah sebagai berikut :

Manfaat yang dapat diperoleh dengan adanya sistem belajar menggunakan metode *E-Learning* dapat meningkatkan pengetahuan dan proses belajar seni teater.

1.7 Sistematika Penulisan

Agar memudahkan dalam memberikan gambaran secara utuh penulisan skripsi ini, maka sistematika penulisan ini di bagi 5 (lima) bab sebagai berikut :

BAB I PENDAHULUAN

Bab ini menjelaskan tentang latar belakang dibuatnya perangkat lunak *E-Learning* elemen-elemen seni teater, identifikasi masalah yang di dapat, perumusan masalah yang di dapat, batasan masalah yang di buat, tujuan, manfaat dan sistematika penulisan yang di terapkan.

BAB II LANDASAN TEORI

Dalam bab ini memuat tentang teori-teori yang mendukung penelitian yang akan di lakukan oleh penulis / peneliti. Apabila penelitian memerlukan analisa statistika maka pada bab ini di cantumkan juga teori statistika yang di gunakan dan hipotesa (bila di perlukan).

BAB III METODOLOGI PENELITIAN

Bab ini berisi metode yang di gunakan dalam penyelesaian permasalahan yang di tanyakan dalam perumusan masalah dan analisa yang di lakukan dalam membangun perangkat lunak *E-Learning* elemen-elemen seni teater berbasis android. Selain itu, bab ini membahas prosedur sistem baru yang di ajukan, use case diagram, activity diagram, perancangan tatap muka.

BAB IV HASIL PENELITIAN DAN PEMBAHASAN

Bab ini menjelaskan tentang hasil dari pengkodean yang di lakukan, sehingga yang di bahas pada bab ini adalah bagaimana tampilan sistem saat di jalankan. selanjutnya di paparkan tentang instalasi perangkat lunak dan bagaimana sistem ini di uji.

BAB V KESIMPULAN DAN SARAN

Bab ini berisi tentang kesimpulan dan saran yang di harapkan dapat bermanfaat dalam perangkat lunak *E-Learning* elemen-elemen seni teater bebasis android.

DAFTAR PUSTAKA**LAMPIRAN**

BAB II

LANDASAN TEORI

2.1 Teater

Teater adalah seni yang dapat dikatakan sebagai seni campuran. Hal ini dikarenakan dalam seni teater di mana unsur-unsur seni lain seperti sastra, seni rupa, arsitektur, musik, dan tari masuk di dalamnya dan menciptakan sebuah karya seni yang disebut teater. Seni teater bisa disebut juga seni kerja sama, sehingga masalah kedudukan tidak terstruktur seperti satu tingkat komando dari atas ke bawah, dengan sutradara yang paling tinggi tingkatnya. Kerja sama dalam seni teater berarti kebersamaan, yaitu komitmen setiap pendukungnya melalui komunikasi yang bebas dan terbuka untuk menciptakan sebuah karya seni yang handal. (Aisyatur Radhiah, Taat Kurnita, 2017).

2.2 E-Learning

Melihat perkembangan *E-Learning* dari masa kemasanya yang terus berkembang mengikuti perkembangan teknologi, maka dapat disimpulkan bahwa *E-Learning* akan menjadi sistem pembelajaran masa depan. Alasan efektifitas dan fleksibilitas akan menjadi alasan utama. Pemanfaatan dari perkembangan *E-Learning* berdasarkan waktu, terbagi menjadi dua jenis yaitu *synchronous* dan *asynchronous*. *Synchronous* berarti pada waktu yang sama interaksi terjadi antara guru dan murid melalui website. Implementasi *synchronous* ini adalah virtual *classroom*, Sedangkan *asynchronous* memberikan keleluasaan kepada murid untuk belajar kapan pun tanpa harus secara langsung pada waktu yang sama berinteraksi dengan guru. Metode *asynchronous* dapat berupa *embeddedlearning*, *course* dan *discussion groups*. (Hasibuan & Meitro, 2014).

2.3 Multimedia

2.3.1 Pengertian Multimedia

Menjelaskan bahwa Multimedia adalah kombinasi data atau media untuk menyampaikan sebuah informasi sehingga informasi tersebut tersaji dengan lebih menarik.

Sedangkan definisi lainnya menyatakan bahwa Multimedia merupakan kombinasi dari teks, grafis, seni, suara, animasi dan video yang dikirimkan oleh komputer atau peralatan elektronik lain. (Arfida & Harahap, 2014).

2.3.2 Objek-Objek Multimedia

Objek-objek multimedia adalah sebagai berikut :

a. Teks

Teks adalah bentuk data multimedia yang paling mudah disimpan dan dikendalikan. Kebutuhan teks tergantung pada kegunaan aplikasi multimedia.

b. Grafik

Grafik menjadi nilai dan unsur tambah suatu penyajian data. Gambar digunakan dalam presentasi multimedia untuk menarik perhatian.

c. Gambar Vektor

Gambar vektor disimpan sebagai serangkaian instruksi yang digunakan untuk membuat suatu gambar yang dinamakan algoritma, yang menentukan bentuk kurva, garis dan berbagai bangun yang diwakilkan oleh gambar (*picture*). Untuk menyimpan gambar yang tidak terlalu banyak mengandung unsur perubahan warna, gambar vektor adalah pilihan yang lebih tepat.

d. Gambar Bitmap

Gambar bitmap adalah gambar yang tersimpan dalam rangkaian *pixel* (titik-titik). Komputer akan mengatur tiap titik di layar sesuai dengan detil warna *bitmap*.

e. Suara (*Audio*)

Penyampaian sebuah informasi yang sering disertai desain grafis dan teks yang menarik, akan terasa membosankan apabila tidak disertai dengan suara.

f. Video

Video menyediakan sumber daya yang kaya dan membuat aplikasi multimedia lebih hidup. Namun kendala yang dihadapi adalah ukuran file yang terlalu besar. Untuk itu diperlukan software lain untuk memperkecil ukuran file video.

g. Animasi (*Animation*)

Animasi dalam multimedia merupakan penggunaan komputer untuk menciptakan gerak pada layar.

2.4 Android

Android adalah sebuah sistem operasi untuk *smartphone* dan tablet. Sistem operasi dapat diilustrasikan sebagai 'jembatan' antara piranti (*device*) dan penggunanya, sehingga pengguna bisa berinteraksi dengan *device*-nya dan menjalankan aplikasi-aplikasi yang tersedia pada *device*. (Sari, 2016).

Android adalah sebuah sistem operasi untuk perangkat *mobile* berbasis *linux* yang mencakup sistem operasi, *middleware*, dan aplikasi. Android adalah sistem operasi untuk telepon seluler yang berbasis *linux*. Android menyediakan platform terbuka bagi para pengembang untuk membuat aplikasi mereka sendiri. Pada awalnya dikembangkan oleh Android Inc, sebuah perusahaan pendatang baru yang membuat perangkat lunak untuk ponsel yang kemudian dibeli oleh Google Inc. Untuk pengembangannya, dibentuklah Open Handset Alliance (OHA), konsorsium dari 34 perusahaan perangkat keras, perangkat lunak, dan telekomunikasi termasuk *Google*, *HTC*, *Intel*, *Motorola*, *Qualcomm*, *T-Mobile*, dan *Nvidia* (Nazrudin Safaat H, 2015, p.1).

2.4.1 Versi Android

Adapun versi-versi android yang pernah dirilis adalah sebagai berikut :

- a. Android Versi 1.0 Alpha - Dirilis pada 23 September 2008.
- b. Android Versi 1.1 Bender (Beta) - Dirilis pada 9 Februari 2009.
- c. Android Versi 1.5 CupCake - Dirilis pada 27 April 2009.
- d. Android Versi 1.6 Donut - Dirilis pada 15 September 2009.
- e. Android Versi 2.0 - 2.1 Eclair - Dirilis pada 26 Oktober 2009.
- f. Android Versi 2.2 Frozen Yoghurt - Dirilis pada 10 Mei 2010.
- g. Android Versi 2.3 GingerBread - Dirilis pada 6 Desember 2010.
- h. Android Versi 3.0 - 3.2 HoneyComb - Dirilis pada 22 Pebruari 2011.
- i. Android Versi 4.0 Ice Cream Sandwich - Dirilis pada 19 Oktober 2011.
- j. Android Versi 4.1 - 4.3 Jeally Bean - Dirilis pada 27 Juni 2012.
- k. Android Versi 4.4 KitKat - Dirilis pada 31 Oktober 2013.
- l. Android Versi 5.0 Lollipop - Dirilis pada 5 Juni 2014.
- m. Android Versi 6.0 Marshmallow - Dirilis pada 17 Agustus 2015.
- n. Android Versi 7.0 Nougat - Dirilis pada 18 Juli 2016.
- o. Android Versi 8.0 Oreo - Dirilis pada 21 Agustus 2017.



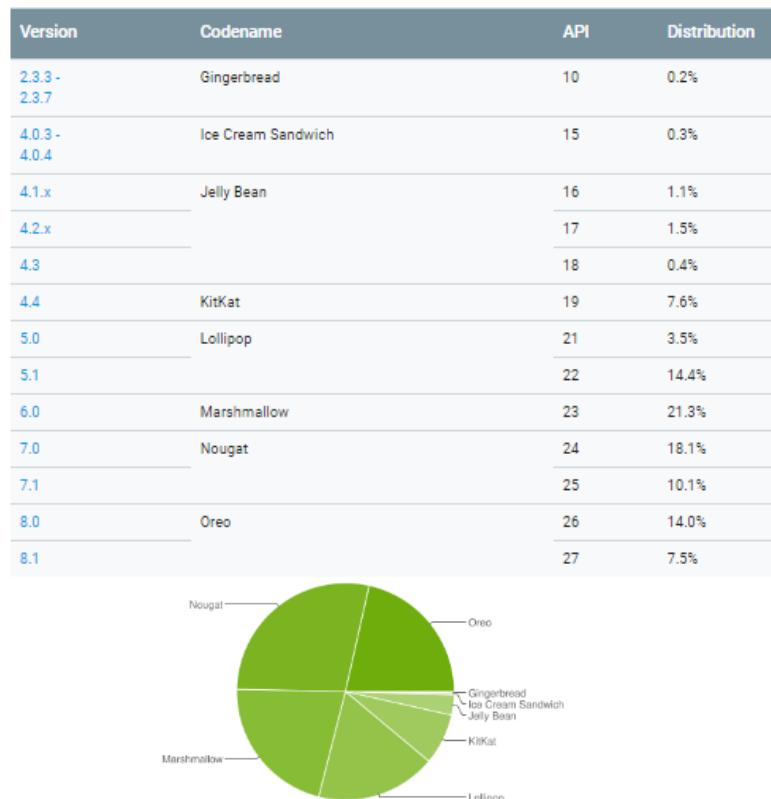
Gambar 2.1 Evolusi Android.

(sumber : <http://en.miui.com/thread-475925-1-1.html>)

2.4.2 Statistik Distribusi OS Android

Google baru-baru ini merilis angka statistik distribusi sistem operasi Android di berbagai perangkat mobile di seluruh dunia pada 26 Oktober 2018. Dalam statistik tersebut terlihat Android 6.0 *Marshmallow* menjadi yang tertinggi dengan persentase total 21.3%. Di urutan kedua berhasil dikuasai oleh Android 7.0 *Nougat* yang memiliki *persentase* mencapai 18.1%. Kedua sistem operasi ini memang layak menduduki angka tertinggi karena saat ini smartphone Android memang masih banyak yang menggunakan Android *Marshmallow* dan *Nougat* mengingat vendor-vendor di seluruh dunia masih mempercayakan kedua sistem operasi ini pada smartphone buatannya. Sementara pada urutan ketiga diduduki Android 5.1 *Lollipop* dengan *persentase* 14.4% yang dudusul oleh Android 8.0 *Oreo* dengan presentase 14.0 %. kedua sistem operasi ini masih banyak digunakan pada smartphone *entry-level* di berbagai negara. Selanjutnya, Android 7.1 *Nougat* dengan persentase total 10.1%. Selanjutnya, Andorid 4.4 *Kitkat* dengan presentase 7.6%. Selanjtnya Android 8.1 *Oreo* dengan *presentase* 7.5% dan android versi 5.0 *Lollipop* sudah mulai ditinggalkan pengguna Android karena hanya memiliki persentase 3.5% dam umtuk urutan di bawah nya yaitu oleh android dengan versi 4.1.x, 4.2.x dan 4.3 *Jelly Bean* dengan total 3.0% berikutnya android versi 4.0.3 dan 4.0.4 *Ice cream sandwich* dengan *presentase* 0.3 % di urutan terakhir yaitu android versi 2.3.3-2.3.7 *Ginger bread* dengan *presentase* 0.2% Hal ini tentu wajar karena Google sendiri sudah menghentikan dukungan untuk sistem operasi *Ginger bread* ini pada Q1 2017. Sementara versi *Ice Cream Sandwich* yang meski terus melempem tetapi masih akan didukung dalam waktu lebih lama.

Berikut data statistik pengguna sistem operasi android yang dikumpulkan selama periode 7 hari yang berakhir pada 26 Oktober 2018, Setiap versi dengan distribusi kurang dari 0,1% tidak ditampilkan.



Gambar 2.2 Statistik distribusi OS Android (26 Oktober 2018)

(sumber :<https://developer.android.com/about/dashboards/>)

2.5 Perangkat Lunak Pengembangan Sistem

Untuk membangun perangkat lunak *E-Learning* elemen-elemen seni teater, diperlukan beberapa jenis perangkat lunak, yaitu sebagai berikut :

2.5.1 Android Studio

Android Studio adalah sebuah IDE (*Integrated Development Environment*) untuk Android *development* yang diperkenalkan oleh Google. Android Studio merupakan pengembangan dari Eclipse IDE, dan dibuat berdasarkan IDE Java populer, yaitu *IntelliJ IDEA*.

Android studio mempunyai fitur-fitur yang dapat mempermudah penulis dalam penelitiannya.



Gambar 2.3 Logo Android Studio

(sumber : <https://developer.android.com/studio/index.html>)

2.5.2 Elemen Android

a. *Dalvik Virtual Machine (DVM)*

Salah satu element kunci dari Android adalah *Dalvik Virtual Machine* (DVM). Android berjalan di *Dalvik Virtual Machine* (DVM) bukan di Java Virtual Machine (JVM), sebenarnya banyak persamaannya dengan Java Virtual Machine (JVM) seperti *Java ME (Java Mobile Edition)*, tetapi Android menggunakan virtual machine sendiri yang dirancang untuk memastikan beberapa fitur-fitur berjalan lebih efisien pada perangkat mobile.

b. *Android SDK (Software Development Kit)*

Android SDK adalah *tools API (Application Programming Interface)* yang diperlukan untuk mulai mengembangkan aplikasi pada plat formAndroid menggunakan bahasa pemrograman *Java*. Android merupakan subset perangkat lunak untuk ponsel yang meliputi sistem operasi *middleware* dan aplikasi kunci yang di-release oleh *Google*. Saat ini disediakan Android SDK (*Software Development Kit*) sebagai

alat bantu dan API untuk mulai mengembangkan aplikasi pada *platform* Android menggunakan bahasa pemrograman *Java*. Sebagai *platform* aplikasi *-neutral*, Android memberi kesempatan untuk membuat aplikasi yang dibutuhkan.

c. ADT (Android *Development Tools*)

Android *development tools* adalah plugin yang di desain untuk IDE Android Studio yang memberikan kemudahan dalam mengembangkan aplikasi Android dengan menggunakan IDE Android Studio. Dengan menggunakan ADT untuk Android Studio akan memudahkan dalam membuat aplikasi *project* Android, membuat GUI aplikasi, dan menambahkan komponen-komponen yang lainnya.

2.5.3 *Firebase*

Firebase adalah layanan pihak ketiga, *Firebase* bisa dikatakan sebagai layanan DbaaS (*Database as a Service*) dengan konsep *realtime*. Tidak hanya berfungsi sebagai tempat penyimpanan data, tapi juga di sediakan API untuk implementasi web *socket*. *Firebase* menyediakan *library* untuk berbagai *client platform*. Browser menggunakan Java script API dan mobile menggunakan OBJ-C atau Android API.

Berikut penjelasan beberapa fitur yang tersedia pada *firebase* :



Gambar 2.4 logo firebase

(sumber : <https://firebase.google.com/>)

a. *Firebase Cloud Messaging (FCM)*

Firebase Cloud Messaging untuk Android (FCM) adalah layanan yang membantu pengembang mengirim data dari server untuk aplikasi mereka Android pada perangkat Android. Ini bisa menjadi pesan ringan memberitahu aplikasi Android bahwa ada data baru yang akan diambil dari server (misalnya, film diunggah oleh seorang teman), atau bisa juga pesan yang berisi sampai dengan 4KB data payload (sehingga aplikasi seperti instant messaging dapat mengkonsumsi pesan langsung). Layanan FCM menangani semua aspek antrian pesan dan pengiriman ke aplikasi target Android berjalan pada perangkat target.

b. *Firebase Authentication*

Firebase Authentication menyediakan layanan backend, SDK yang mudah digunakan, dan library UI yang siap pakai untuk mengautentikasi pengguna ke aplikasi Anda. *Firebase Authentication* mendukung autentikasi menggunakan sandi, nomor telepon, penyedia identitas gabungan yang populer, seperti *Google*, *Facebook*, dan *Twitter*, dan lain-lain.

c. *Firebase Storage*

Cloud Storage untuk *Firebase* adalah layanan penyimpanan objek yang andal, sederhana, dan hemat biaya yang dibuat untuk skala *Google*. *Firebase* SDK untuk *Cloud Storage* menambahkan keamanan *Google* pada upload dan download file untuk aplikasi *Firebase* Anda, bagaimana pun kualitas jaringannya. Anda dapat menggunakan SDK kami untuk menyimpan gambar, audio, video, atau konten buatan pengguna lainnya. Di server, Anda dapat menggunakan [Google Cloud Storage](#) untuk mengakses file yang sama.

d. *Firebase Realtime Database*

Firebase Realtime Database adalah *database* yang di-host di *cloud*. Data disimpan sebagai *JSON* dan disinkronkan secara realtime ke setiap *klien* yang terhubung. Ketika Anda membuat aplikasi lintas *platform* dengan *SDK* *Android*, *iOS*, dan *JavaScript*, semua *klien* akan berbagi *sebuah instance Realtime Database* dan menerima update data terbaru secara otomatis

2.5.4 *Adobe Photoshop*

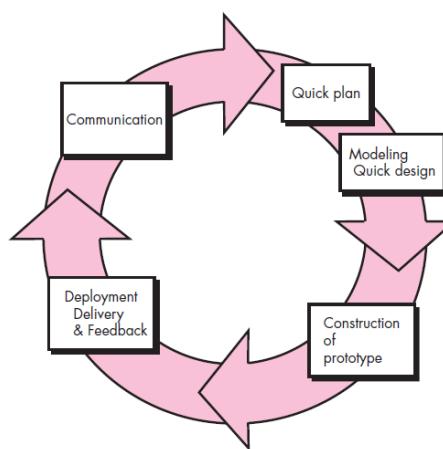
Adobe Photoshop adalah *Software* (Perangkat Lunak) buatan *Adobe Systems* yang digunakan untuk pengeditan foto / gambar, termasuk pembuatan efek grafis. *Adobe Photoshop* sering digunakan oleh *Foto grafer Digital* dan Perusahaan Iklan (*Advertising*). (Rini Agustina dan Ade Chandra, 2017).

2.6 Model Prototype

Metode *Prototype* adalah proses pembuatan model sederhana *software* yang mengijinkan pengguna memiliki gambaran dasar tentang program serta melakukan pengujian awal. *Prototype* memberikan fasilitas bagi pengembang dan pemakai untuk saling berinteraksi selama proses pembuatan, sehingga pengembang dapat dengan mudah memodelkan perangkat lunak yang akan di buat. Metode ini cocok digunakan untuk mengembangkan sebuah perangkat lunak yang dikembangkan kembali. Metode ini dimulai dengan pengumpulan kebutuhan pengguna. Kemudian membuat sebuah rancangan kilat yang selanjutnya akan dievaluasi kembali sebelum di produksi secara benar. *Prototype* bukanlah merupakan sesuatu yang lengkap, tetapi sesuatu yang harus dievaluasi dan dimodifikasi kembali. Segala perubahan dapat terjadi pada saat *prototype* dibuat untuk memenuhi kebutuhan pengguna dan saat yang sama memungkinkan pengembangan untuk lebih memahami kebutuhan pengguna secara baik (Pressman, 2012).

Berikut adalah tahapan dalam metode *prototype* :

1. Komunikasi (*Communication*) dan pengumpulan data awal, yaitu komunikasi dengan klien dan *user* untuk menentukan kebutuhan.
2. Perencanaan cepat (*Quick Plan*), yaitu pembuatan perencanaan analisis terhadap kebutuhan pengguna.
3. Pemodelan perancangan cepat (*Modeling Quick Design*), yaitu membuat rancangan desain program.
4. Pembentukan *prototype* (*Construction of prototype*), yaitu pembuatan aplikasi berdasarkan dari pemodelan desain yang telah dibuat.
5. Penyerahan sistem dan umpan balik(*Development Delevery and Feedback*), yaitu memproduksi perangkat secara benar sehingga dapat digunakan oleh pengguna.



Gambar 2.5 Diagram Prototype

(sumber : Roger S.Pressman, Ph.D. (2012). Rekayasa Perangkat Lunak)

2.7 Unified Modelling Language (UML)

Mendefinisikan bahwa, Unified Modelling Language (UML) adalah sebuah “bahasa” yang telah menjadi standar dalam industri untuk visualisasi, merancang dan mendokumentasikan sistem piranti lunak, UML menawarkan sebuah standar untuk merancang model sebuah sistem. Tujuan Penggunaan UML yaitu untuk memodelkan suatu sistem yang menggunakan konsep berorientasi objek dan menciptakan bahasa

pemodelan yang dapat digunakan baik oleh manusia maupun mesin. (Rosa dan Shalahuddin2016, p.133).

Menurut Rosa dan Shalahuddin (2016, p.140) tipe-tipe diagaram UML adalah sebagai berikut :

2.7.1 *Use Case Diagram*

Use case diagaram adalah gambar dari beberapa atau seluruh aktor dan *use case* dengan tujuan yang mengenali interaksi mereka dalam suatu sistem. *Use case* diagram menggambarkan fungsionalitas yang diharapkan dari sebuah sistem, yang ditekankan adalah “apa” yang diperbuat sistem, dan bukan “bagaimana”. Sebuah *use case* mempresentasikan sebuah interaksi antara *actor* dan sistem.

Dalam use case diagram terdapat istilah seperti aktor, *use case* dan *case relationship*. Penjelasan simbol *use case* diagram ditunjukan pada tabel 2.1.

Tabel 2.1 Simbol Use Case Diagram.

Simbol	Keterangan
	Aktor : Seseorang atau sesuatu yang berinteraksi dengan sistem yang sedang dikembangkan.
	<i>Use case</i> : perungkat tertinggi dari fungsionalitas yang dimiliki sistem.
	Association : adalah relasi antara <i>actor</i> dan <i>use case</i> .
	Generalisasi : untuk memperlihatkan struktur pewaris yang terjadi.

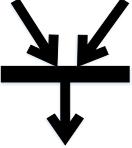
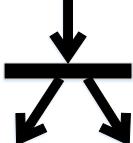
2.7.2 Activity Diagram

Activity Diagram menggambarkan rangkaian aliran dari aktifitas, digunakan untuk mendeskripsikan aktivitas yang dibentuk dalam suatu operasi sehingga dapat juga digunakan untuk aktifitas lainnya seperti *use case* atau interaksi Activity Diagram berupa *flow chart* yang digunakan untuk memperlihatkan aliran kerja dari sistem. Notasi yang digunakan dalam *activity diagram* ditunjukan pada tabel 2.2.

Tabel 2.2 Simbol Activity Diagram.

Simbol	Keterangan
	<i>Activity</i> : Memperlihatkan bagaimana masing-masing kelas antarmuka saling berinteraksi satu sama lain.
	<i>Initial Node</i> : Bagaimana objek dibentuk atau diawali
	<i>Activity Final Node</i> : Bagaimana objek dibentuk dan diakhiri.
	<i>Decision</i> : Asosiasi percabangan dimana jika ada pilihan aktifitas lebih dari satu.
	<i>Swim lane</i> : Memisalkan organisasi bisnis yang bertanggung jawab terhadap aktifitas yang terjadi.

Lanjutan Tabel 2.2

	Join : Digunakan untuk menunjukkan kegiatan yang digabungkan.
	Fork : Digunakan untuk menunjukkan kegiatan yang dilakukan secara paralel

2.7.3 Sequence Diagram

Sequence Diagram menggambarkan kolaborasi dinamis antara sejumlah dan untuk menunjukkan rangkaian pesan yang dikirim antar objek juga interaksi antar objek, sesuatu yang terjadi pada titik tertentu dalam eksekusi sistem. Sequence diagram menjelaskan interaksi objek yang disusun berdasarkan urutan waktu. Dalam sequence diagram terdapat 2 simbol yaitu : *Actor*, untuk menggambarkan pengguna sistem. *Lifeline*, untuk menggambarkan kelas dan objek.

2.7.4 Class Diagram

Class diagram menggambarkan struktur data dan deskripsi class, *package*, dan objek beserta hubungan satu sama lain. *Class* diagram berfungsi untuk menjelaskan tipe dari objek sistem dan hubungannya dengan objek yang lain. *Class* memiliki 3 area pokok yaitu nama, atribut dan metode.

2.8 Pengujian *Black Box Testing*

Black-Box Testing atau Pengujian Kotak Hitam atau juga disebut *Behavioral Testing*, berfokus pada persyaratan fungsional dari perangkat lunak. Artinya, teknik *Black-Box Testing* memungkinkan untuk

mendapatkan *set* kondisi masukan yang sepenuhnya akan melaksanakan semua persyaratan fungsional untuk suatu program (Pressman, 2012).

Black-Box Testing bukan merupakan *alternatif* dari pengujian *White-BoxTesting*. Sebaliknya, *Black-Box Testing* adalah pendekatan komplementer yang mungkin untuk mengungkap kelas yang berbeda dari kesalahan daripada metode *White-Box Testing*.

Black-BoxTesting mencoba untuk menemukan kesalahan dalam kategori berikut :

- a. Fungsi tidak benar atau hilang.
- b. Kesalahan *interface* atau antarmuka.
- c. Kesalahan dalam struktur data atau akses *database* eksternal.
- d. Kesalahan kinerja atau perilaku.
- e. Kesalahan inisialisasi dan terminasi.

2.9 Penelitian Terdahulu

Tabel 2.3 berikut ini adalah beberapa penelitian terdahulu yang berkaitan dengan *e-learning*.

Tabel 2.3 Penelitian Terdahulu

No	Nama	Judul	Terbit	Kesimpulan / Saran	Analisa
1	Aisyatur Radhiah, Taat Kurnita, Aida Fitri	Pembelajaran Seni Teater Pada Materi Rancangan Dan Persiapan Pementasan Teater DiKelas IX	Volume II, Nomor 2:108 – 116 Mei 2017	Pemanfaatan media, metode pembelajaran, dan sumber belajar yang tepat dapat menciptakan suasana belajar yang menyenangkan bagi siswa. Kepala sekolah diharapkan	Pada penelitian ini hanya penerapan pembelajaran seni teater di sekolah dengan menggunakan metode ceramah guru terhadap siswanya dalam satu ruangan. Dengan adanya saran yang

Lanjutan Tabel 2.3

		SMP NEGERI 5 Kluet Utara Kabupaten Aceh Selatan		dapat menyediakan media -media dan sumber belajar yang di butuhkan seperti buku baik cetak maupun online, proyektor dan lain sebagainya yang dapat menunjang proses pembelajaran.	di cantumkan, maka di perlukanya sebuah aplikasi yang dapat membantu proses pembelajaran seni teater meskipun dalam keadaan tidak dalam satu ruangan dengan dukungan materi dari sumber yang di percaya.
2	Muham mad Said Hasibuan	System E- Learnig Dengan Pendekata n Evaluasi Pembelaja ran	(SNATI 2009) Yogyaka rta, 20 Juni 2009.	Dibutuhkan sumber daya manusia yang handal untuk melakukan pengembangan dan peningkatan sistem agar lebih interaktif dan sesuai kebutuhan pengguna Dibutuhkan pelatihan - pelatihan khusus bagi para guru dan siswa agar mampu menggunakan sistem dengan baik. Penyajian konten yang baik sangat diperlukan	Pada penelitian ini penerapan e-learning pada media berbasis website sehingga pengguna di butuhkanya pelatihan-pelatihan khusus untuk menggunkan sistem tersebut dengan baik dan memiliki tahapan tahapan tertentu berbeda jika menggunakan media berbasis aplikasi maka pengguna akan lebih mudah dalam menggunakan

Lanjutan Tabel 2.3

				<p>untuk memperoleh keselarasan dengan gaya belajar siswa, sehingga mampu meningkatkan minat belajar siswa.</p> <p>Peningkatan daya interaksi antara siswa dan guru sangat dibutuhkan untuk memperoleh hasil yang maksimal.</p>	Perangkat lunak untuk proses pembelajaran.
3	Pujianto	Integrasi E-Learning Dengan Media Sosial Menggunakan Single Sign On System	Vol. 13, No.1, Juni 2013	<p>Sebuah web e-learning yang ingin di integrasikan dengan web media sosial janevala.com harus diaktifkan terlebih dahulu layanan jaringannya.</p> <p>Untuk integrasi web media social yang berbasis e-portofolio dengan e-learning harus memiliki public key atau sertifikat. Untuk pengujian single sign on sistem pada web</p>	Di dalam penelitian ini memiliki beberapa tahap proses untuk mendapatkan akses jaringan dan proses pembelajaran sedangkan dalam penelitian sekarang cukup memiliki aplikasi e-learning tersebut peserta dan terhubung dengan internet bisa langsung mendapatkan materi dan juga akses

Lanjutan Tabel 2.3

				e-learning dan web media sosial pengguna dapat login melalui e-learning kemudian aktifkan menu network server.	komunikasi dengan pemateri.
4	Silahuddin	Penerapan E-Learning Dalam Inovasi Pendidikan	Jurnal Ilmiah CIRCUIT Vol.1 No.1 Juli 2015	Pengembangan sistem E-learning juga diharapkan tidak hanya memperhitungkan masalah finansial dan profitabilitas, tetapi juga memperhatikan sisi psikologis pelajar dan mampu mengakomodasi berbagai kepribadian dan cara belajar masing-masing peserta	Aplikasi e-learning dapat di gunakan di luar ruangan sehingga peserta memiliki keluasan dalam berproses untuk belajar dengan ada nya aplikasi pembelajaran <i>e-learning</i> seni teater mampu berdiskusi dan mendapatkan sebuah arahan secara materi dari sumber para ahli nya untuk mempelajari seni teater.
5	Mohammand Yazdi	E-Learning Sebagai Media	Jurnal Ilmiah Foristek	Di samping itu prinsip sederhana, personal, dan cepat	Dengan adan nya kebutuhan pembelajaran <i>e-</i>

Lanjutan Tabel 2.3

	Pembelajaran Interaktif Berbasis Teknologi Informasi	Vol. 2, No. 1, Maret 2012	perlu di pertimbangkan. Untuk menambah daya tarik dapat pula menggunakan teori game. Oleh karena itu prinsip dan komunikasi pembelajaran perlu di desain seperti layaknya pembelajaran konvensional. Disini perlunya pengembangan model e-learning yang tepat sesuai dengan kebutuhan.	<i>learning</i> maka dalam penelitian ini di berikan akses interaksi untuk berdiskusi antar user dan <i>admin</i> yang bersifat <i>chat room</i> .
--	--	---------------------------	--	--

Berdasarkan penelitian diatas dapat disimpulkan bahwa proses penerapan belajar menggunakan metode *E-Learning* dilakukan dengan cara yang berbeda-beda dengan miliki tujuan yang sama yaitu menciptakan proses pembelajaran tidak dalam satu ruangan yang di pada bidangnya masing-masing.

Berdasarkan penelitian diatas maka penulis akan melakukan penelitian Tentang pembelajaran elemen-elemen seni teater yang menggunakan *E-Learning* dengan tipe *Asynchronous* interaksi yang dapat terjadi pada waktu yang tidak bersamaan, jadi peserta dapat mengambil pembelajaran pada waktu yang berbeda dengan pengajar yang memberikan materi pembelajaran tentang seni teater dan

peserta dapat berkomunikasi dengan peserta lainya dan juga pemateri sehingga peserta dapat berdiskusi (*sharing*) mengenai materi yang di dapat nya.

Penelitian ini merupakan pengembangan dari beberapa penelitian sebelumnya serta digunakan langkah-langkah baru dalam proses *E-Learning* agar menjadi pembeda dari penelitian sebelumnya.

BAB III

METODELOGI PENELITIAN

3.1 Metode Penelitian.

3.1.1 Teknik Pengumpulan data

Dalam tahapan ini yang dilakukan dalam pengumpulan data ini meliputi sebagai berikut:

3.1.1.1 Wawancara

Dalam hal ini penulis melibatkan beberapa pihak-pihak terkait untuk dilakukan wawancara demi mendapatkan data yang akurat. Diantara pihak-pihak yang terlibat dalam proses wawancara adalah Dewan Kesenian Lampung Komite bidang seni teater.

3.1.1.2 Studi Pustaka

Pengumpulan data dengan cara mengumpulkan literatur yang bersumber dari jurnal, buku, pakar ataupun dari hasil penelitian orang lain yang berkaitan dengan penelitian yang akan dilakukan.

3.1.1.3 Observasi

Dalam metode observasi ini penulis diberikan kesempatan untuk melakukan pengumpulan data dengan cara mendatangi langsung Kantor Dewan Kesenian Lampung.

3.2 Metode Pengembangan Perangkat Lunak

Pada tahapan pengumpulan perangkat lunak, penelitian ini dilakukan berdasarkan metode pengembangan sistem yang dipilih yaitu metode pengembangan sistem *Prototype*.

3.2.1 Komunikasi

Komunikasi harus dilakukan dengan yang tepat. Data objektif dan relevan dengan pokok pembahasan menjadi indikator keberhasilan suatu penelitian. Komunikasi dilakukan dengan cara mengadakan interaksi dengan admin dan user sehingga kebutuhan perangkat lunak dapat terpenuhi.

3.2.2 Perencanaan Cepat

Quick Plan (rencana cepat) merupakan tahapan dengan melakukan analisis dan perencanaan setelah kita mendapatkan data-data dari tahapan komunikasi. Analisa kebutuhan *non fungsional* adalah sebuah langkah dimana seseorang pembangun perangkat lunak menganalisis sumber daya yang akan menggunakan perangkat lunak yang dibangun. Analisis kebutuhan *non fungsional* tidak hanya menganalisis siapa saja yang akan menggunakan aplikasi tetapi juga menganalisis perangkat keras dan perangkat lunak agar aplikasi dapat berjalan dengan baik. Analisis *non fungsional* yang dilakukan dibagi dalam tiga tahapan, yaitu :

3.2.2.1 Analisis Kebutuhan Pengguna (*Admin Dan User*)

Aplikasi untuk menentukan konsentrasi skripsi dan rekomendasi bahasa pemrograman ini akan digunakan oleh *user* dan *admin* dengan ketentuan sebagai berikut:

- a. Menggunakan Sistem Operasi berbasis Android.
- b. Menggunakan *Sistem Operasi Android* minimum versi 6.0 *Marshmallow* level API 23.

3.2.2.2 Analisis Kebutuhan Perangkat Lunak (*Software*)

Analisis kebutuhan perangkat lunak yang digunakan untuk membangun aplikasi visualisasi 3D pada android adalah sebagai berikut :

- a. Perangkat lunak sistem operasi pada PC adalah *Microsoft Windows 7 64 bit.*
- b. Perangkat lunak sistem operasi pada android minimum adalah Android versi 6.0. (*Marshmallow*) dengan level API 23.
- c. Perangkat lunak untuk pembuatan program adalah *Android Studio.*
- d. Perangkat lunak yang di gunakan untuk pembuatan *data base* adalah *Firebase.*
- e. Perangkat lunak pembuatan komponen aplikasi adalah *Adobe Photoshop.*

3.2.2.3 Analisis Kebutuhan Perangkat Keras (*Hardware*)

Analisis kebutuhan perangkat keras yang digunakan untuk membangun sebuah sistem adalah sebagai berikut :

Spesifikasi minimum untuk PC :

- a. *Processor Intel(R) Core(TM) i3-2330M @2.20GHZ 2.20GHz*
- b. Ram 4 Gb.
- c. *Harddisk 500 Gb.*
- d. *Keyboard dan Mouse.*

Spesifikasi minimum untuk android :

- a. *Processor Marshmallow 6.0.*
- b. Ram 3 Gb.
- c. *Storage 16 Gb.*

Analisa tersebut bukanlah hal yang mutlak, namun merupakan pendapat peneliti tentang minimum penggunaan perangkat keras yang dipakai dalam pengembangan aplikasi.

3.3 Pemodelan Desain

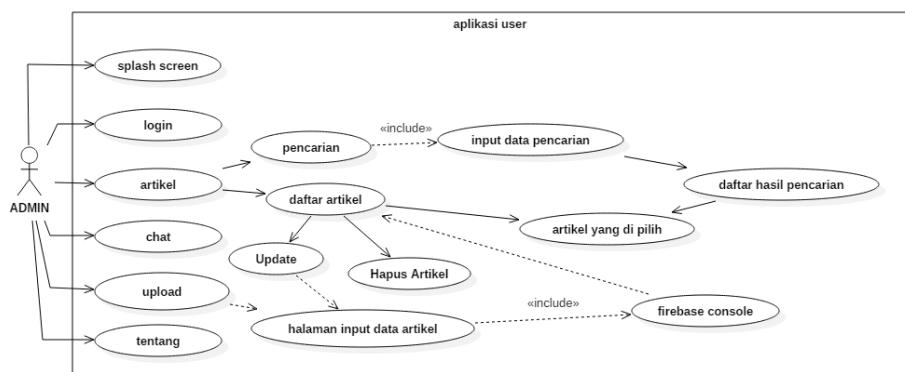
Analisis kebutuhan dari tahap sebelumnya akan dipelajari dalam fase ini dan desain *system* disiapkan. Desain Sistem membantu dalam mendefinisikan arsitektur *system* secara keseluruhan.

3.3.1 Desain UML (*Unified Modelling Language*)

UML (*Unified Modelling Language*) yaitu suatu metode permodelan secara visual untuk sarana perancangan sistem berorientasi objek, atau definisi UML yaitu sebagai suatu bahasa yang sudah menjadi standar pada visualisasi, perancangan dan juga pendokumentasian sistem software. Pada Penelitian kali ini UML yang dipakai adalah *Use Case*, *Activity Diagram*, dan *Class Diagram*.

3.3.1.1 Rancangan *Use Case Diagram*

- a. Rancangan *use case diagram* admin dapat dilihat pada gambar 3.1

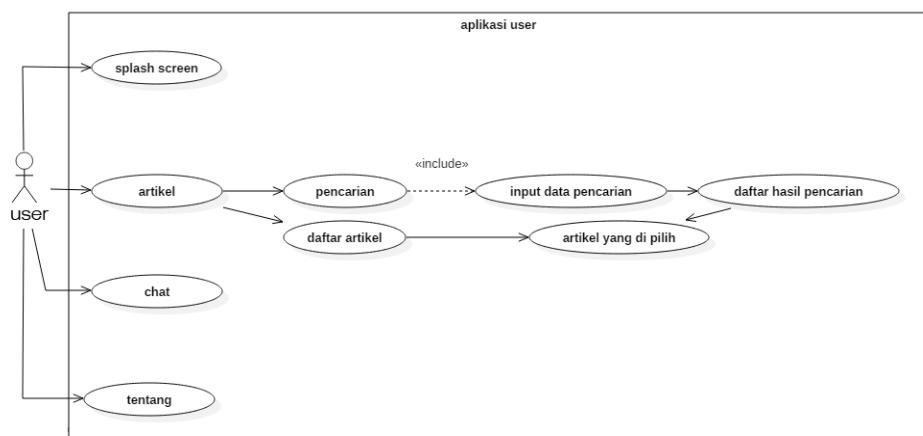


Gambar 3.1 *use case diagram* admin

Berdasarkan *use case diagram* pada gambar 3.1 dapat dijelaskan secara singkat fungsi dari *use case* *Splash Screen* salah satu kondisi yang digunakan untuk menggambarkan sebuah aplikasi atau program dalam proses loading (menyiapkan konten) sebelum ke tahap halaman berikutnya yaitu halaman menu utama terkhusus untuk aplikasi admin sebelum ke halaman menu utama harus terlebih dahulu manuju ke halaman login yang di dalamnya terdiri dari 2 fitur yang harus di isi yaitu email dan password yang sudah terdaftar di dalam database, di dalam halaman menu utama berisi

beberapa tombol di antaranya adalah : tombol *chat* berfungsi sebagai menuju pada halaman komunikasi antara admin dan user, tombol upload berfungsi sebagai memberikan atau memposting sebuah materi tentang seni teater fitur ini hanya dimiliki oleh aplikasi admin, tombol tentang berfungsi sebagai menuju pada halaman yang berisi profil pembuat aplikasi tersebut sedangkan tombol artikel berfungsi sebagai menuju pada halaman isi materi yang berbentuk teks yang dapat dipilih sehingga pengguna akan diarah menuju halaman materi secara detail yang berisi photo, isi materi berupa text, judul materi nama pemilik artikel dan di dalam halaman menu materi terdapat tools pencarian materi berfungsi sebagai mempermudah pengguna aplikasi dalam mencari materi tersebut sedangkan khusus untuk admin memiliki beberapa fitur di dalam menu utama di antara adalah : tools Hapus Materi, tools Update materi.

b. Rancangan *use case diagram* *user* dapat dilihat pada gambar 3.2



Gambar 3.2 *use case diagram* *user*

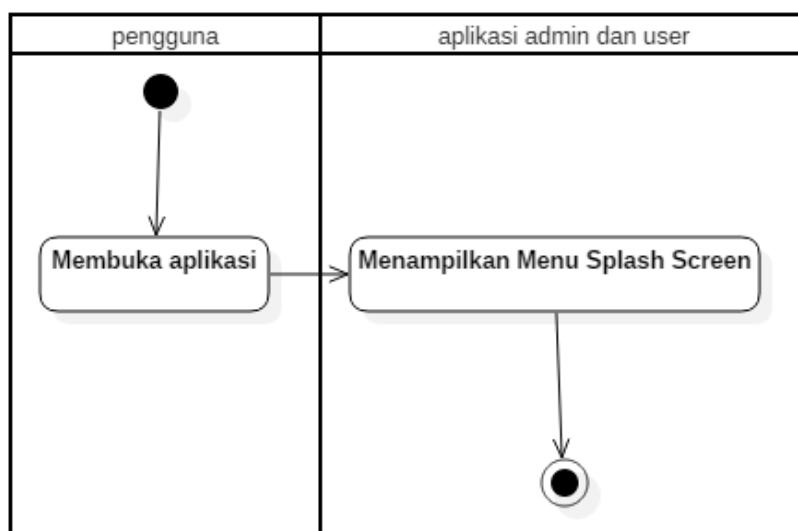
Berdasarkan *use case diagram* pada gambar 3.2 dapat dijelaskan secara singkat fungsi dari *use case Splash Screen* salah satu kondisi yang digunakan untuk menggambarkan sebuah aplikasi atau program dalam proses loading (menyiapkan konten) sebelum ke tahap halaman berikutnya yaitu halaman menu utama berisi beberapa tombol di antaranya adalah : tombol *chat* berfungsi

sebagai menuju pada halaman komunikasi antara admin dan user, tombol tentang berfungsi sebagai menuju pada halaman yang berisi profil pembuat aplikasi tersebut sedangkan tombol artikel berfungsi sebagai menuju pada halam isi materi yang berbentuk teks yang dapat di pilih sehingga pengguna akan di arah menuju halaman materi secara detail yang berisi photo, isi materi berupa text, judul materi nama pemilik artikel dan di dalam halaman menu materi terdapat tools pencarian materi berfungsi sebagai mempermudah pengguna aplikasi dalam mencari materi tersebut.

3.3.2 Rancangan *Activity Diagram*

- Rancangan *Activity Diagram* Menu *Splash Screen Admin* dan *User*

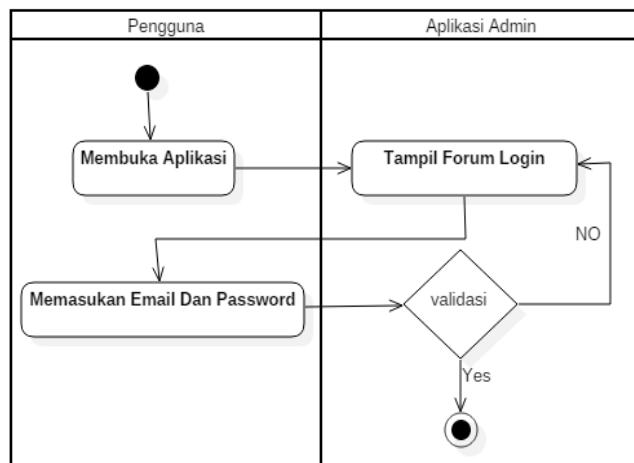
Rancangan *Activity Diagram* Menu *Splash Screen Admin* dan *User* dapat dilihat pada gambar 3.3.



Gambar 3.3 Rancangan *Activity Diagram* Menu *Splash Screen Admin Dan User*

b. Rancangan *Activity Diagram Menu Login Admin*

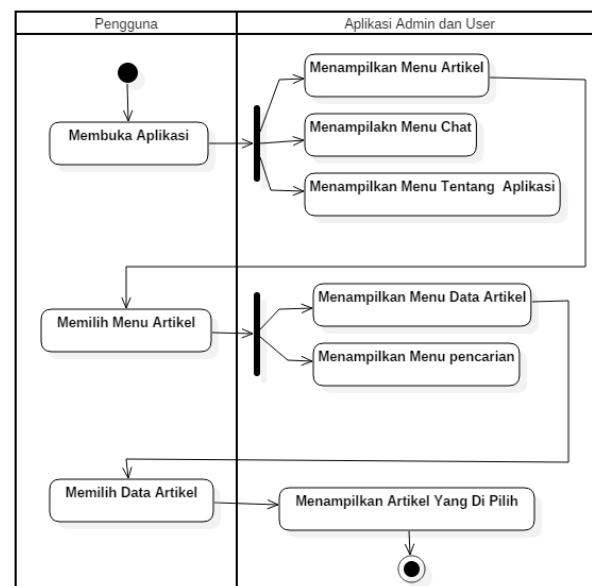
Rancangan *Activity Diagram Menu Login Admin* dapat dilihat pada gambar 3.4.



Gambar 3.4 Rancangan *Activity Diagram Menu Login Admin*

c. Rancangan *Activity Diagram Menu Artikel Admin dan User*

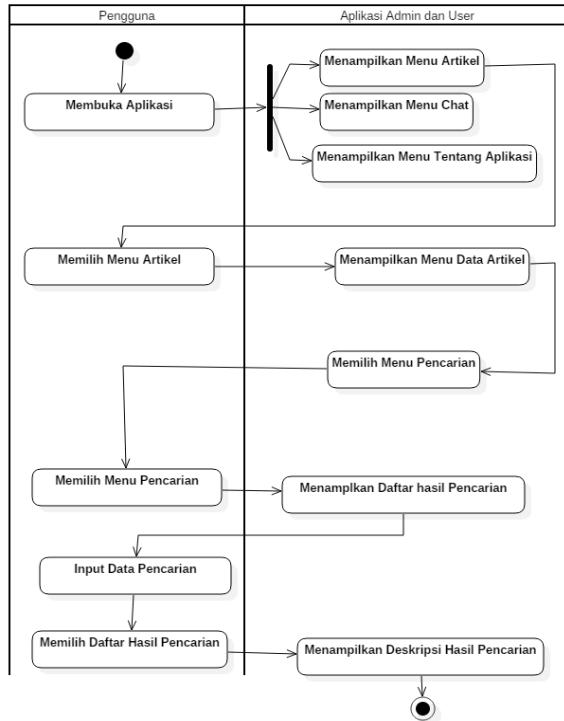
Rancangan *Activity Diagram Menu Artikel Admin dan User* dapat dilihat pada gambar 3.5.



Gambar 3.5 Rancangan *Activity Diagram Menu Artikel Admin dan User*

d. Rancangan *Activity Diagram* Menu Pencarian Admin dan User

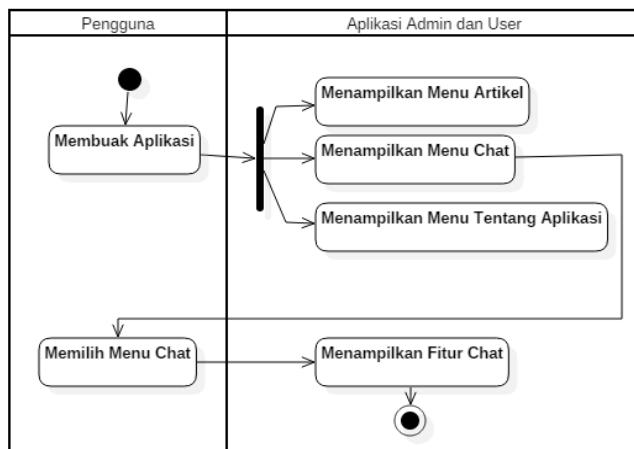
Rancangan *Activity Diagram* Menu Pencarian Admin dan User dapat dilihat pada gambar 3.6.



Gambar 3.6 Rancangan *Activity Diagram* Menu Pencarian Admin Dan User

e. Rancangan *Activity Diagram* Chat Admin dan User

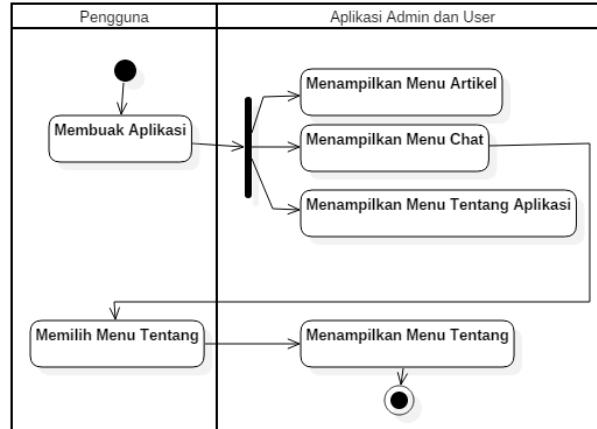
Rancangan *Activity Diagram* Menu Chat Admin dan User dapat dilihat pada gambar 3.7.



Gambar 3.7 Rancangan *Activity Diagram* Chat Admin Dan User

f. Rancangan *Activity Diagramm Menu Tentang Admin dan User*

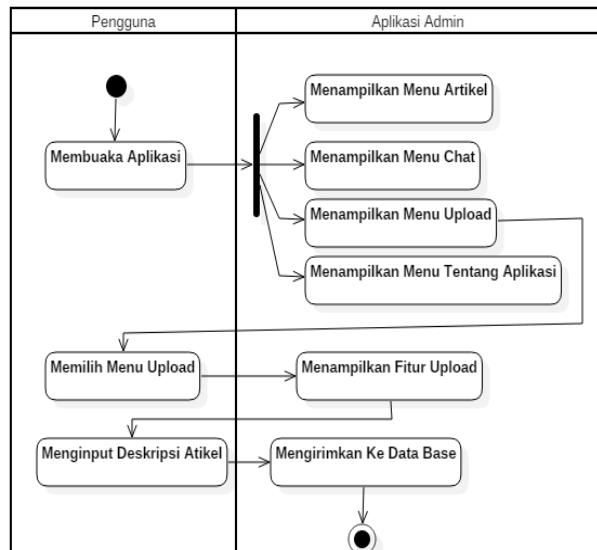
Rancangan *Activity Diagramm Menu Tentang Admin dan User* dapat dilihat pada gambar 3.8.



Gambar 3.8 Rancangan *Activity Diagramm Menu Tentang Admin Dan User*

g. Rancangan *Activity Diagram Menu Upload Admin*

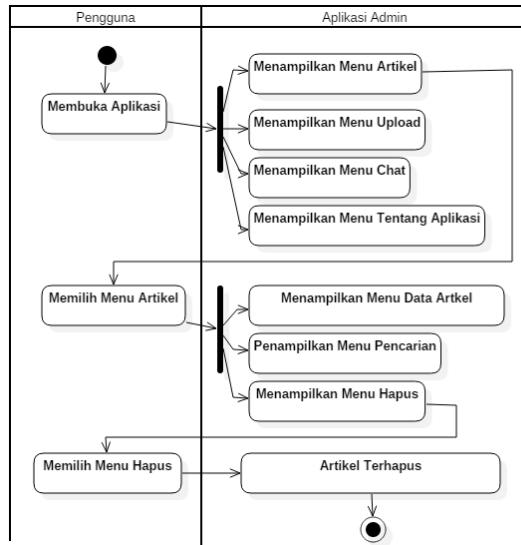
Rancangan *Activity Diagram Menu Upload Admin* dapat dilihat pada gambar 3.9.



Gambar 3.9 Rancangan *Activity Diagram Menu Upload Admin Dan User*

h. Rancangan *Activity Diagram* Menu Hapus Admin

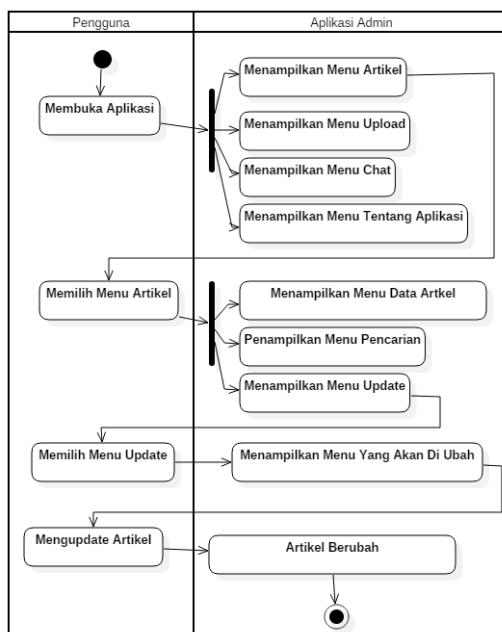
Rancangan *Activity Diagram* Menu Hapus Admin dapat dilihat pada gambar 3.10.



Gambar 3.10 Rancangan *Activity Diagram* Chat Menu Hapus Admin

a. Rancangan *Activity Diagram* Menu Update Admin

Rancangan *Activity Diagram* Menu Update Admin dapat dilihat pada gambar 3.11.



Gambar 3.11 Rancangan *Activity Diagram* Menu Update Admin

3.4 Basis Data

Basis data yang dipergunakan adalah layanan yang ada pada basis data *firebase*. Berdasarkan pernyataan yang dikeluarkan oleh *Google Developers* (<https://firebase.google.com/products/database/>) berikut adalah keunggulan dari basis data *firebase* :

a. Sinkronisasi *real-time* untuk data JSON

Firebase Realtime Database adalah *data base No SQL* yang diunggah dan disimpan di *cloud*, yang dapat digunakan untuk menyimpan dan menyinkronkan data antara pengguna secara *real-time*.

b. Membangun aplikasi tanpa server

Basisa data yang *real time* dikirimkan dengan SDK seluler dan web sehingga pengembang dapat membuat aplikasi tanpa memerlukan server. pengembang juga dapat menjalankan kode *back end* yang merespons peristiwa yang dipicu oleh database pengembang menggunakan [Cloud Function](#) untuk [Firebase](#).

c. Dioptimalkan untuk penggunaan *offline*

Ketika pengguna Anda *offline*, SDK *Realtime Database* menggunakan *cache* lokal pada perangkat untuk menayangkan dan menyimpan perubahan. Ketika perangkat online, data lokal otomatis disinkronkan.

d. Keamanan berbasis pengguna yang kuat

Realtime Database terintegrasi dengan *Firebase Authentication* untuk menyediakan proses autentikasi yang mudah dan intuitif bagi *developer*. pengembang dapat menggunakan model keamanan deklaratif kami (*Firebase*) untuk mengizinkan akses berdasarkan identitas pengguna atau pencocokan pola pada data.

e. Bagian dari *platform Firebase*

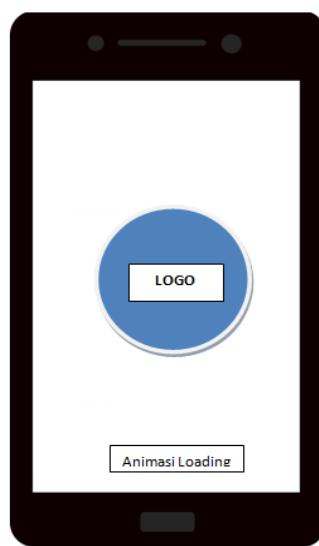
Firebase membantu pengembang membuat aplikasi berkualitas tinggi, mengembangkan basis pengguna, dan menghasilkan lebih banyak uang. Setiap fitur berfungsi secara independen dan bekerja sangat baik jika dipakai bersama.

3.5 Desain Antar Muka Aplikasi

Desain aplikasi merupakan rancangan bagaimana bentuk dan rupa tampilan aplikasi yang ingin dibuat.

3.5.1 Rancangan Tampilan Halaman *Splash Screen*

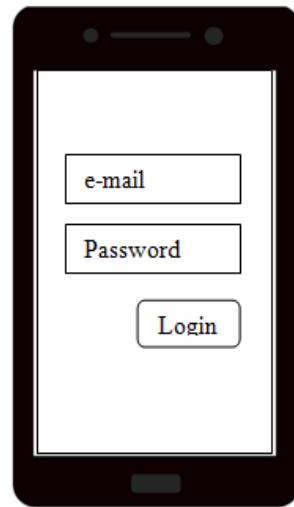
Pada rancangan tampilan *Splash Screen Admin* dan *User* terdapat logo, animasi *loading* dan nama pembuat aplikasi. Tampilan rancangan halaman *splash screen* dapat dilihat pada gambar 3.12.



Gambar 3.12 Tampilan rancangan halaman *splash Screen Admin* dan *User*

3.5.2 Rancang Tampilan Halaman Login Admin

Pada rancangan tampilan halaman login terdapat dua fitur input text diantaranya input text email, input text password dan tombol untuk login di dalam aplikasi tersebut. Tampilan rancangan halaman menu utama dapat dilihat pada gambar 3.13

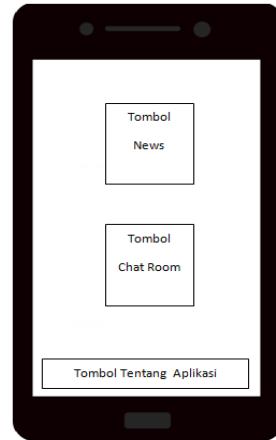


Gambar 3.13 Tampilan rancangan halaman login admin

3.5.3 Rancangan Tampilan Halaman Utama

a. Rancangan Tampilan Halaman Utama *User*

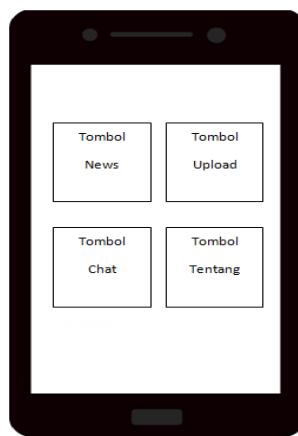
Pada rancangan tampilan menu utama terdapat tombol-tombol yang dapat dipilih di dalam aplikasi tersebut. Tombol-tombol tersebut diantaranya tombol Materi, *Chat* dan Tentang Aplikasi. Tampilan rancangan halaman menu utama dapat dilihat pada gambar 3.14.



Gambar 3.14 Tampilan rancangan halaman menu utama *User*

b. Rancangan Tampilan Halaman Utama Admin

Pada rancangan tampilan menu utama terdapat tombol-tombol yang dapat dipilih di dalam aplikasi. Tombol-tombol tersebut diantaranya tombol Materi, *Upload*, *Chat* dan Tentang Aplikasis yang berfungsi sebagai berpindah halaman sesuai dengan target halaman tersebut. Tampilan rancangan halaman menu utama dapat dilihat pada gambar 3.15.

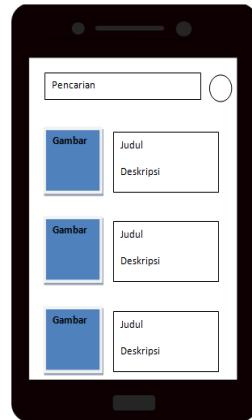


Gambar 3.15 Tampilan rancangan halaman menu utama *Admin*

3.5.4 Rancangan Tampilan Halaman Materi

a. Rancangan Tampilan Halaman Materi Admin

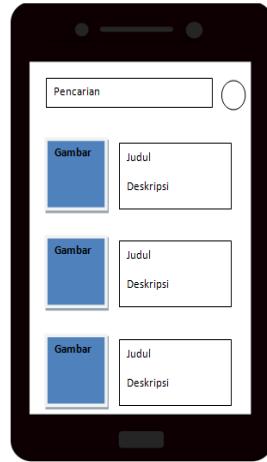
Pada rancangan tampilan Halaman Materi *Admin* terdapat menu-menu yang dapat dipilih di dalam aplikasi. Menu-menu tersebut diantaranya Pencarian materi, hapus materi, pilihan materi dan Update Materi. Tampilan rancangan halaman menu utama dapat dilihat pada gambar 3.16.



Gambar 3.16 Tampilan rancangan halaman materi *Admin*

b. Rancangan Tampilan Halaman Materi *User*

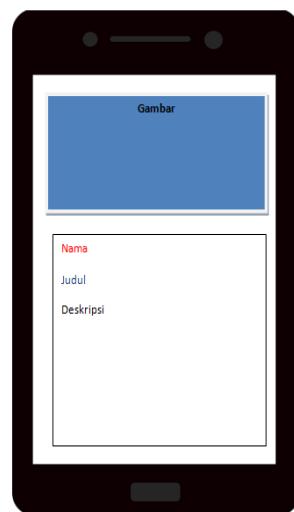
Pada rancangan tampilan Halaman Materi *User* terdapat menu-menu utama yang dapat dipilih di dalam aplikasi. Menu-menu tersebut diantaranya Pencarian materi, dan pilihan Materi. Tampilan rancangan halaman menu utama dapat dilihat pada gambar 3.17.



Gambar 3.17 Tampilan rancangan halaman materi *User*

3.5.5 Rancangan Tampilan Halaman Materi Detail

Pada rancangan tampilan halaman materi secara detail untuk *Admin* dan *User* terdapat menu-menu utama yang dapat dilihat di dalam aplikasi. Menu-menu tersebut diantaranya Nama, Judul, Deskripsi dan Gambar. Tampilan rancangan halaman materi secara detail dilihat pada gambar 3.18.



Gambar 3.18 Tampilan rancangan halaman materi detail *Admin* dan *User*

3.5.6 Rancangan Tampilan Halaman Hapus Materi

Pada rancangan tampilan halaman hapus materi untuk *Admin*. Tampilan rancangan halaman halaman materi secara detail dilihat pada gambar 3.19



Gambar 3.19 Tampilan rancangan halaman hapus materi *Admin*

3.5.7 Rancangan Tampilan Halaman *Update* Materi

Pada rancangan tampilan halaman Update materi untuk *Admin* terdapat menu-menu utama yang dapat menginput data didalam aplikasi. Menu-menu tersebut diantaranya Nama, Judul, Deskripsi dan Gambar. Tampilan rancangan halaman halaman materi secara detail dilihat pada gambar 3.20.



Gambar 3.20 Tampilan rancangan halaman *Update* materi *Admin*

3.5.8 Rancangan Tampilan Halaman *Upload Materi*

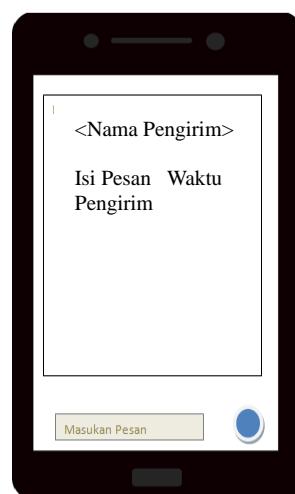
Pada rancangan tampilan halaman Update materi untuk *Admin* terdapat menu-menu utama yang dapat menginput data didalam aplikasi. Menu-menu tersebut diantaranya Nama, Judul, Deskripsi dan Gambar. Tampilan rancangan halaman materi secara detail dilihat pada gambar 3.21.



Gambar 3.21 Tampilan rancangan halaman *Update* materi *Admin*

3.5.9 Rancangan Tampilan Halaman *Chat Room*

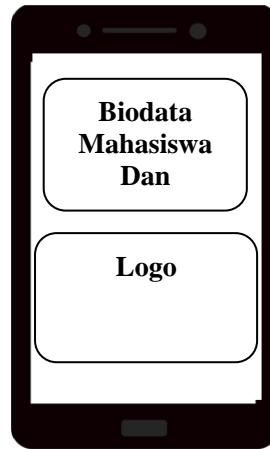
Pada rancangan tampilan halaman *Chat Room* untuk *Admin* dan *User* terdapat menu-menu utama yang dapat menginput data berupa text dan melihat text percakapan di dalam aplikasi tersebut. Menu-menu tersebut diantaranya Nama, isi pesan dan waktu pengiriman. Tampilan rancangan halaman halaman materi secara detail dilihat pada gambar 3.22.



Gambar 3.22 Tampilan rancangan halaman *Chat Room Admin* dan *User*

3.5.10 Rancangan Tampilan Halaman Tentang Aplikasi

Pada rancangan tampilan halaman Tentang Aplikasi untuk *Admin* dan *User* terdapat menu-menu utama yang dapat dilihat data berupa *text*, tombol dan gambar di dalam aplikasi. Tampilan rancangan halaman materi secara detail dilihat pada gambar 3.23.



Gambar 3.23 Tampilan rancangan halaman Tentang Aplikasi *Admin* dan *User*

3.6 Pembentukan *Prototype*

Pada tahap inilah aplikasi dibuat dengan menggunakan bahasa pemrograman tertentu dan diuji bagaimana program berjalan.

3.6.1 Pembuatan aplikasi

Dalam penelitian ini aplikasi dibuat pada dasarnya dengan menggunakan *Android Studio* dan juga perangkat lunak penunjang yang telah dijelaskan pada sub bab analisis kebutuhan perangkat lunak. Sedangkan untuk bahasa pemrograman yang digunakan adalah Java.

3.6.2 Pengujian aplikasi

Pengujian aplikasi dilakukan dengan menggunakan metode *black-box testing*. Pengujian yang akan dilakukan yaitu dengan menguji lama waktu *loading* atau *respon time* dari masing-masing halaman yang terdapat pada aplikasi. Pengujian dilakukan dengan menggunakan tiga buah perangkat yang berbeda yang memiliki spesifikasi dengan keriteria tinggi, sedang, dan rendah dilihat dari segi perangkat kerasnya.

Berikut adalah ketiga perangkat tersebut dan spesifikasinya :

1. Perangkat Kategori Di Bawah Minimun
 - a. Versi Sistem Operasi : Android Versi 4.2.2 (*JellyBean*)
 - b. Kecepatan *Processor* : 1,7 Ghz
 - c. Kapasitas RAM : 512 MB
2. Perangkat Kategori Rendah
 - a. Versi Sistem Operasi : Android Versi 6.0 (*Marshmallow*)
 - b. Kecepatan *Processor* : 2,02 Ghz
 - c. Kapasitas RAM : 3 Gb
3. Perangkat Kategori Sedang
 - a. Versi SistemO perasi : Android Versi 7.0 (*Noughat*)
 - b. Kecepatan *Processor* : 1,4 Ghz
 - c. Kapasitas RAM : 3 Gb
4. Perangkat Kategori Tinggi
 - a. Versi Sistem Operasi : Android versi 8.1.0(*Oreo*)
 - b. Kecepatan *Processor* : 1,99 Ghz
 - c. Kapasitas RAM : 4 Gb

3.7 Penyerahan Sistem dan Umpan Balik

Penyerahan system dan umpan balik merupakan tahapan ketika aplikasi telah selesai dibuat. Aplikasi yang telah dibuatkan distribusikan melalui *Google Play Store* sehingga semua orang bisa mengunduhnya.

BAB IV

HASIL PENELITIAN DAN PEMBAHASAN

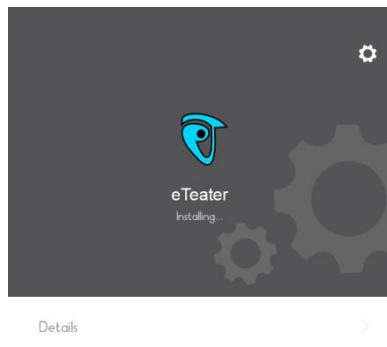
4.1 Hasil Penelitian

Setelah melalui analisa dan perancangan, maka tahapan berikutnya adalah pembuatan kode program. Tatap muka dari aplikasi ini adalah sebagai berikut.

4.2 Pembahasan

4.2.1 Instalasi Perangkat Lunak

Perangkat lunak elearning elemen-elemen seni teater akan diinstal pada sistem operasi Android.



Gambar 4.1 gambar instalasi aplikas

4.2.2 Penggunaan Perangkat Lunak

4.2.2.1 Perangkat Lunak Admin

a. Icon

Penggunaan adalah saat aplikasi sudah terinstal di sebuah sistem operasi Android dengan tampilan *icon* pada smartphone pengguna sebagai berikut:



Gambar 4.2 Tampilan *icon*

b. Halaman *Splash Screen*

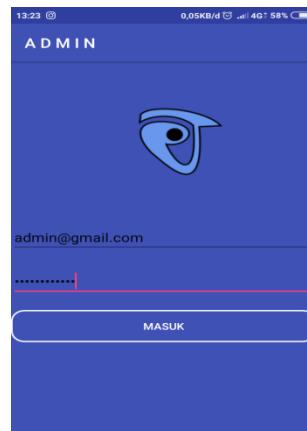
Pada saat aplikasi telah terinstal dan terpasang pada *smartphone* pengguna maka akan muncul halaman *splash screen* sebagai launcher tampilan pertama dari Aplikasi *E-Learning* elemen-elemen seni teater. Hasil dari tampilan halaman *splash screen* yang dapat di lihat pada gambar 4.3.



Gambar 4.3 Tampilan halaman *splash screen*

c. Halaman Login

Pada saat aplikasi telah terinstal dan terpasang pada *smartphone* pengguna maka akan muncul halaman *login* berfungsi sebagai *system security* dari Aplikasi *E-Learning* elemen-elemen seni teater. Hasil dari tampilan halaman *splash screen* yang dapat di lihat pada gambar 4.4.



Gambar 4.4 Tampilan halaman *splash screen*

d. Halaman Utama

Berikut merupakan tampilan halaman utama dari Aplikasi *E - Learning* elemen-elemen seni teater yang dapat di lihat pada gambar 4.5.



Gambar 4.5 Tampilan halaman Utama

e. Halaman Menu Materi

Berikut merupakan tampilan menu materi dari Aplikasi *E-Learning* elemen-elemen seni teater yang dapat di lihat pada gambar 4.6.



Gambar 4.6 Tampilan halaman menu materi

f. Halaman Materi Detail

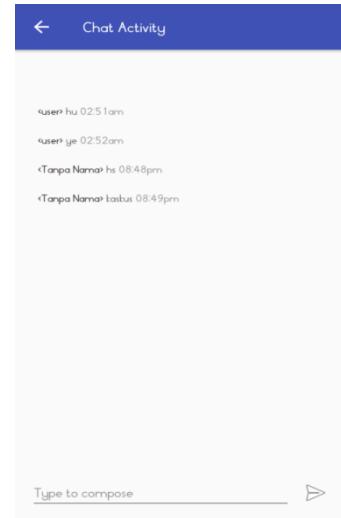
Berikut merupakan tampilan materi secara detail materi dari Aplikasi *E - Learning* elemen-elemen seni teater yang dapat di lihat pada gambar 4.7.



Gambar 4.7 Tampilan halaman menu Detail

g. Halaman *Chat Room*

Berikut merupakan tampilan *chatroom* tempat berdiskusi mengenai seni teater dari Aplikasi *E - Learning* elemen-elemen seni teater yang dapat di lihat pada gambar 4.8.



Gambar 4.8 Tampilan Halaman *Chat Room*

h. Halaman Tentang

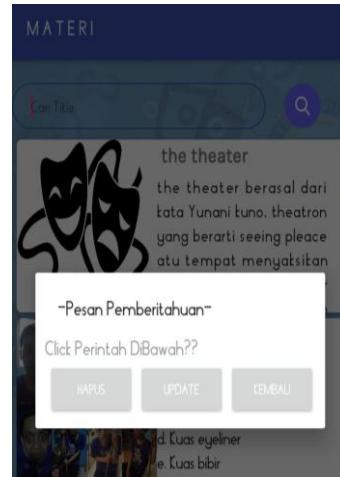
Berikut merupakan tampilan informasi aplikasi dari Aplikasi *E - Learning* elemen-elemen seni teater yang dapat di lihat pada gambar 4.9.



Gambar 4.9 Tampilan Tentang

i. Hapus Materi

Berikut merupakan tampilan hapus materi yang terdapat pada halaman Materi dari Aplikasi *E - Learning* elemen-elemen seni teater.



Gambar 4.10 Tampilan halam Hapus Materi

j. Update Materi

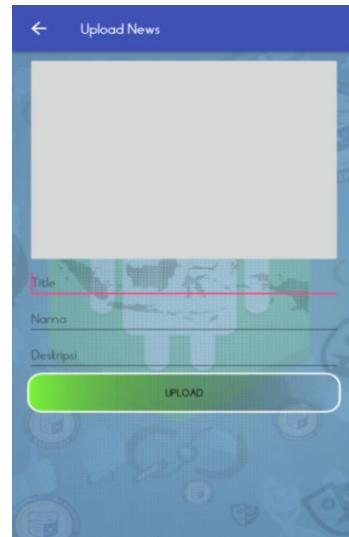
Berikut merupakan tampilan update materi dari Aplikasi *E - Learning* elemen-elemen seni teater yang dapat di lihat pada gambar 4.11.



Gambar 4.11 Tampilan Halaman Update Materi

k. Upload Materi

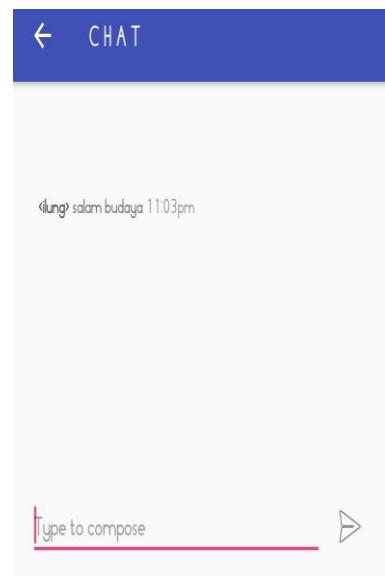
Berikut merupakan tampilan upload materi dari Aplikasi *E -learning* elemen-elemen seni teater yang dapat di lihat pada gambar 4.12.



Gambar 4.12 Tampilan Halaman Upload Materi

l. Halaman *Chat Room Admin*

Berikut merupakan tampilan *chatroom* dari Aplikasi *E-Learning* elemen-elemen seni teater yang dapat di lihat pada gambar 4.13.



Gambar 4.13 Tampilan *Chat Room Admin*

4.2.2.2 Perangkat Lunak User

a. Icon

Penggunaan adalah saat aplikasi sudah terinstal di sebuah sistem operasi Android dengan tampilan *icon* pada *smartphone* pengguna sebagai berikut:



Gambar 4.14 Tampilan Icon

b. Halaman *Splash Screen*

Pada saat aplikasi telah terinstal dan terpasang pada smartphone pengguna maka akan muncul halaman splash screen sebagai launcher tampilan pertama dari Aplikasi *E-Learning* elemen - elemen seni teater. Hasil dari tampilan halaman splash screen terlihat di gambar 4.15.



Gambar 4.15 Tampilan Halaman *Splash Screen*

c. Halaman Utama

Berikut merupakan tampilan halaman utama dari Aplikasi *E-Learning* elemen-elemen seni teater yang dapat dilihat pada gambar 4.16.



Gambar 4.16 Tampilan Halaman Utama

d. Menu Materi

Berikut merupakan tampilan menu materi dari Aplikasi *E-Learning* elemen-elemen seni teater yang dapat dilihat pada gambar 4.17.



Gambar 4.17 Tampilan Halaman Menu mater

e. Materi Detail

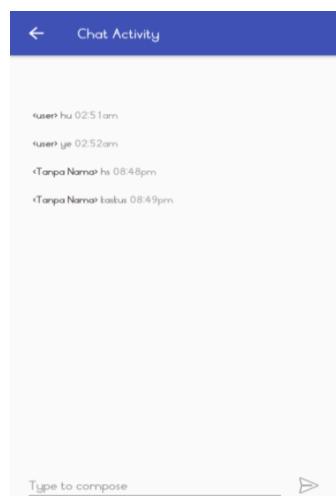
Berikut merupakan tampilan materi secara detail materi dari Aplikasi *E - Learning* elemen-elemen seni teater yang dapat dilihat pada gambar 4.18.



Gambar 4.18 Tampilan halaman materi detail

f. Halaman *Chat Room User*

Berikut merupakan tampilan *chat room* materi dari Aplikasi *E - Learning* elemen-elemen seni teater yang dapat dilihat pada gambar 4.19.



Gambar 4.19 Tampilan *Chat Room User*

g. Tentang

Berikut merupakan tampilan informasi aplikasi dari Aplikasi *E-Learning* elemen-elemen seni teater yang dapat di lihat pada gambar 4.20.



Gambar 4.20 Tampilan Tentang Aplikasi

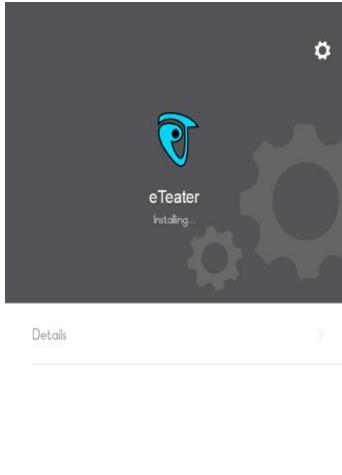
4.3 Pengujian Sistem Perangkat Lunak

Proses pengujian sistem perangkat lunak dilakukan dengan menggunakan metode Metode *Black-Box Testing*, yang merupakan salah satu cara pengujian perangkat lunak yang mengutamakan pengujian terhadap kebutuhan fungsi dari suatu program dengan menemukan kesalahan fungsi pada perangkat lunak tersebut. Dalam tahap pengujian aplikasi ini dilakukan pada berbagai perangkat *mobile* dengan spesifikasi yang berbeda-beda.

4.3.1 Pengujian Instalasi

Pengujian instalasi dilakukan apakah aplikasi yang telah dibuild dapat berjalan diatas sistem operasi Android. Hasil dari pengujian instalasi dapat dilihat pada tabel 4.1.

Tabel 4.1 pengujian instalasi

No.	Spesifikasi	Hasil Pengujian	Keterangan
2	RAM 3 GB Android Nougat 7.1.2 Octa-core Max 1.40GHz, Display 5 inch		BERHASIL
3	RAM 2 GB Android versi 8.1.0 (Oreo) 1.99 GHz Display 5.5 inch		BERHASIL

Lanjutan Tabel 4.1

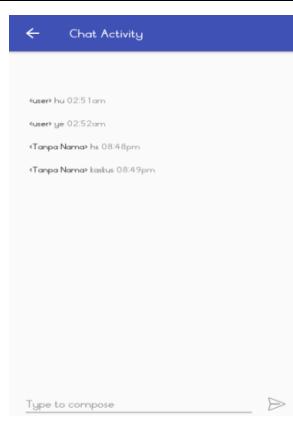
1	Ram 3 GB		BERHASIL
	Android		
	Marshmallow 6.0		
	Octa-core (4x2.3 GHz Mongoose & 4x1.6 GHz Cortex-A53)		
	Display 5.5 inch		

4.3.2 Pengujian Penggunaan

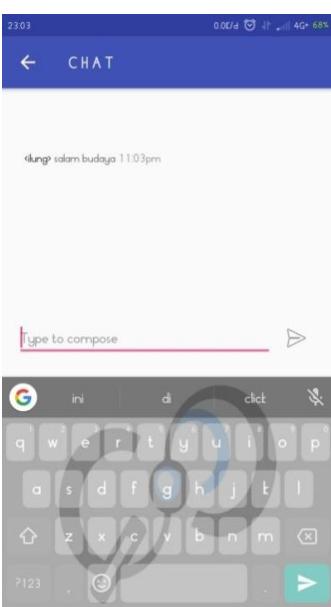
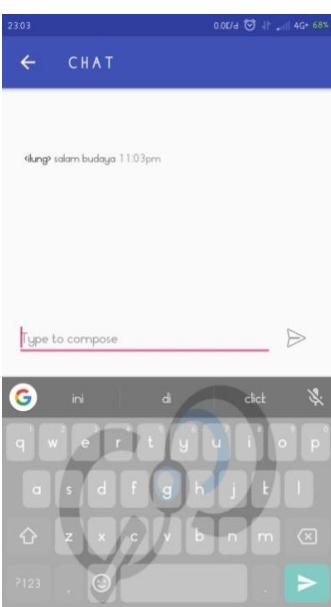
Pada tahapan pengujian penggunaan, akan dilakukan pengujian di beberapa smartphone dengan cara sebagai berikut:

Hasil dari pengujian dapat dilihat pada tabel 4.2.

Tabel 4.2 Pengujian Pada Halaman Chat Room

No.	Spesifikasi	Hasil Pengujian	Keterangan
1	RAM 4 GB		BERHASIL
	Android versi 8.1.0 (Oreo)		
	1.99 GHz,		
	Display 5.5 inch		

Lanjutan Tabel 4.2

	Ram 3 GB		
2	Android Marshmallow 6.0 Octa-core (4×2.3 GHz Mongoose & 4×1.6 GHz Cortex- A53) Display 5.5 inch		BERHASIL
3	Ram 4 GB Android Nougat7.1.2 Octa-core Max 1.40GHz, Display 5 inch		BERHASIL

4.3.3 Pengujian Antarmuka

Pengujian antarmuka perangkat lunak dapat dilihat pada tabel 4.3

Tabel 4.3 antar muka perangkat lunak

No.	Spesifikasi	Hasil Pengujian	Keterangan
1	Ram 3 GB Android Marshmallow 6.0 Octa-core (4×2.3 GHz Mongoose & 4×1.6 GHz Cortex-A53) Display 5.5 inch		BERHASIL
3	RAM 4 GB Android versi 8.1.0 (<i>Oreo</i>) 1.99 GHz, Display 5.5 inch		BERHASIL
2	Ram 4 GB Android Nougat7.1.2 Octa-core Max 1.40GHz, Display 5 inch		BERHASIL

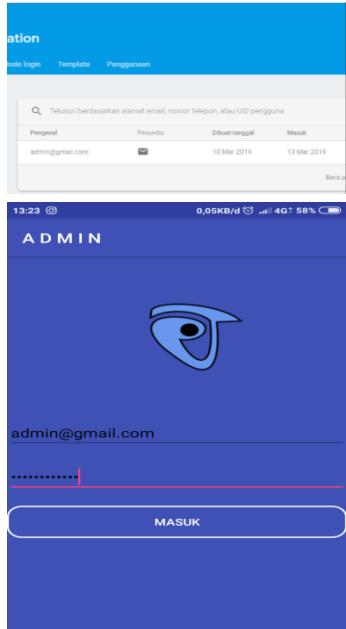
4.3.4 Pengujian Database

Pengujian akses *database* perangkat lunak dapat dilihat pada tabel 4.4

Tabel 4.4 akses database perangkat lunak

No.	Spesifikasi	Hasil Pengujian	Keterangan
1	Ram 3 GB Android Marshmallow 6.0 Octa-core (4×2.3 GHz Mongoose & 4×1.6 GHz Cortex-A53) Display 5.5 inch		BERHASIL
2	RAM 4 GB Android versi 8.1.0 (Oreo) 1.99 GHz, Display 5.5 inch		BERHASIL

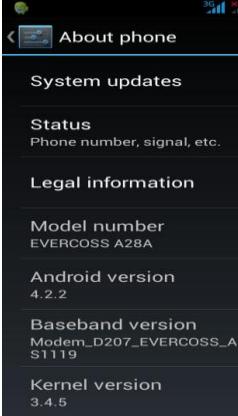
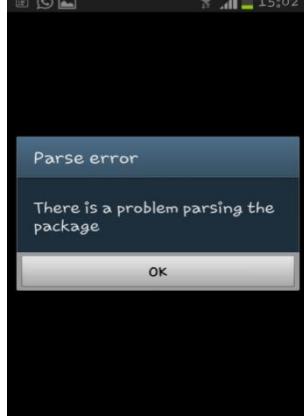
Tabel Lanjutan 4.4

3	<p>Ram 4 GB</p> <p>Android Nougat7.1.2</p> <p>Octa-core Max 1.40GHz,</p> <p>Display 5 inch</p>		BERHASIL
---	--	--	----------

4.3.5 Pengujian Sistem Operasi Minimum

Pengujian ini dilakukan untuk mengetahui dapat atau tidaknya jika aplikasi diinstall di perangkat *smartphone* yang memiliki sistem operasi Android *platform* dibawah minimum. Hasil pengujian dapat dilihat pada tabel 4.5.

Tabel 4.5 pengujian sistem operasi di bawah minimum

No	Android Version	Hasil	Keterangan
1			Dari hasil pengujian pada versi 4.2.2 tersebut didapatkan hasil bahwa <i>perangkat lunak</i> tidak dapat di instal pada versi Android dibawah minimum 6.0

4.4 Kelebihan dan Kekurangan Perangkat Lunak

4.4.1 Kelebihan dari Perangkat Lunak ini adalah sebagai berikut:

- a. Pada perangkat lunak yang berbentuk aplikasi ini dapat diinstal pada *smartphone* merek apapun yang sudah berbasis Android versi 6.0 keatas.
- b. Memiliki fitur untuk berdiskusi mengenai pematerian seni teater yang di berikan oleh admin.
- c. Aplikasi ini dapat menjadi alternatif untuk mempelajari seni teater secara materi.
- d. Dapat berinteraksi dalam bentuk text antar pengguna aplikasi secara *real time*.

4.4.2 Kelemahan dari Perangkat Lunak ini adalah sebagai berikut:

- a. Perangkat Lunak *E-Learning* elemen-elemen seni teater ini hanya dapat dijalankan pada smartphone yang berbasis sistem operasi Android minimum dengan versi 6.0 dan tidak dapat di operasikan pada *system* operasi android versi sebelumnya.
- b. Bagian *chat room* hanya dapat menerima pesan text nama dan waktu pengiriman, tidak dapat mengirim pesan berbentuk image, dokumen, suara dan video.
- c. *Chat* tidak dapat di hapus melalui perangkat lunak.
- d. Tidak memiliki fitur notifikasi dalam penerimaan data pesan maupun materi seni teater.

BAB V

SIMPULAN DAN SARAN

5.1 Simpulan

Berdasarkan pembahasan dari bab-bab sebelumnya, maka dapat diambil kesimpulan sebagai berikut :

- a. penulisan yang menggunakan metode prototype.
- b. aplikasi terdiri dari 2 aplikasi yang dapat di gunakan yaitu aplikasi untuk admin berfungsi sebagai pemberi materi seni teater dan aplikasi user untuk pengguna yang ingin mempelajari seni teater.
- c. Aplikasi *admin* memiliki halaman login mengantisipasi penyalah gunaan aplikasi admin tersebut.
- d. kedua aplikasi tersebut harus terhubung dengan internet agar aplikasi memiliki akses mengirim data pematerian seni teater ke sebuah *database* dan menerima data seni teater dari *database*.

5.2 Saran

Perancangan dan implementasi yang telah dilakukan ini masih jauh dari sempurna, untuk penelitian selanjutnya terdapat beberapa saran yang dapat dipakai untuk pengembangan yang lebih baik lagi.

1. Langkah-langkah strategis yang telah di lakukan pada bab sebelumnya agar dapat implementasikan secara nyata oleh pihak Dewan Kesenian Lampung dengan harapan memperkenalkan sekaligus mempelajari seni dalam bidang teater dari sumber terpercaya sekaligus memperkenalkan dan melestarikan seni teater.
2. Program aplikasi yang dibangun hanya menggunakan bahasa pemrograman *Java Script* pada program Android Studio dan penyimpanan data pada *platform firebase*. penulis mengharapkan dapat dikembangkan kembali pada

penelitian selanjutnya menggunakan perograman dan penyimpanan data yang lebih tepat.

3. Aplikasi ini dapat dikembangkan lebih lanjut sehingga mampu memberikan sebuah notifikasi pada aplikasi dalam penerimaan data baru yang di upload oleh *admin* dan *user*.
4. Penulis mengharapkan dapat dikembangkan sebuah fitur untuk mendapatkan akses melihat dan mengisi profil dari masing-masing pengguna aplikasi tersebut.

DAFTAR PUSTAKA

- Aisyatur Radhiah, Taat Kurnita, A. F. (2017). Pembelajaran seni teater pada materi rancangan dan persiapan pementasan teater di kelas ix smp negeri 5 kluet utara kabupaten aceh selatan, *II*, 108–116.
- Arfida, S., & Harahap, R. E. (2014). Implementasi Media Pembelajaran Teknik Pengkodean Barcode Berbasis Multimedia. *Proseding Seminar Bisnis & Teknologi IIB Darmajaya*, 409. Retrieved from <https://jurnal.darmajaya.ac.id/index.php/sembistik/article/view/226/112>.
- Erma Susanti, M. S. (2008). Rancang bangun aplikasi, *9330*, 53.
- Hasibuan, M. S., & Meitro. (2014). implementasi metode synchronous E-learning berbasis LMS MOODLE. *Sembistik 2014 Ibi Darmajaya*, (December 2014), 730.
- Neil Smyth. (2015). Android Studio Development Essentials. E-Book.
- Pujianto. (2013). Dengan media sosial menggunakan, *13*(1), 33.
- Roger S.Pressman, Ph.D. (2012). Rekayasa Perangkat Lunak (pendekatan praktisi edisi 7). Yogyakarta: penerbit C.V ANDI.
- Sari, Y. P. (2016). Yuni Puspita Sari (Studi Kasus : Kampus Kebidanan Adila Bandar Lampung) Informatics and Business Institute Darmajaya Smartphone mobile sebagai dewasa dan lebih ini produk lebih diminati mahasiswa kebidanan dituntut untuk mengetahui segala hal berkaitan, *16*(1), 66.
- Shalahuddin, M., & Rosa, A.S. 2016. Rekayasa Perangkat Lunak. Penerbit Informatika, Bandung.
- Supratman, E., & Purwaningtias, F. (2018). Model Pembelajaran E-Learning pada SMA Negeri SUMSEL Palembang Berbasis Schoology, 98–103.

Wahana. (2013). Android Programming. Semarang: penerbit C.V ANDI OFFSET.

LAMPIRAN



Dewan Kesenian Lampung (The Lampung Arts Council)

SURAT KETERANGAN Nomor : 007/DKL/Sekret/II/2019

Yang bertanda tangan di bawah ini atas nama Ketua Dewan Kesenian Lampung (DKL) menerangkan bahwa :

Nama : Thiu aan Rian Hermawan
NPM : 1411010098
Jurusan : S1 Teknik Informatika
Jenjang : Strata Satu (S1)

Adalah benar telah melakukan penelitian di Dewan Kesenian Lampung dari tanggal 15 April 2018 s.d. 31 Juli 2018.

Demikian surat keterangan ini dibuat untuk dapat dipergunakan sebagaimana mestinya.

Bandarlampung, 11 Februari 2019

Dewan Kesenian Lampung
An. Ketua Umum,

Bagus S. Pribadi
Sekretaris Umum



Institut Informatika & Bisnis

DARMAJAYA

Yayasan Alifian Husin

Jl. Zainal Abidin Pagar Alam No. 93 Bandar Lampung 35142 Telp 787214 Fax. 700261 <http://darmajaya.ac.id>

FORMULIR

BIRO ADMINISTRASI AKADEMIK KEMAHASISWAAN (BAAK)

FORM KONSULTASI/BIMBINGAN SKRIPSI/TUGAS AKHIR *)

NAMA : THIV AAM RIAN HERMAWAN
NPM : 1411010098
PEMBIMBING I : MISAF
PEMBIMBING II :
JUDUL LAPORAN : PERANGKAT LUMAK E-LEARNING ELEMEN-
ELEMEN CEMI TEATER BERBASIS ANDROID
TANGGAL SK : s.d (6+2 bulan)

No	HARI/TANGGAL	HASIL KONSULTASI	PARAF
1	12-19	dee Bab III, IV, V	
2	tasker la Sarang	
3	Guna duga .	
4			
5		dee. staf	
6		dee Sidang	
7			
8			
9			
10			

*) Coret yang tidak perlu

Bandar Lampung,
Ketua Jurusan



Institut Informatika & Bisnis

DARMAJAYA

Yayasan Alfan Husin

Jl. Zainal Abidin Pagar Alam No. 93 Bandar Lampung 35142 Telp 787214 Fax. 700261 http://darmajaya.ac.id

FORMULIR

BIRO ADMINISTRASI AKADEMIK KEMAHASISWAAN (BAAK)

FORM KONSULTASI/BIMBINGAN SKRIPSI/TUGAS AKHIR *)

NAMA : THIU AAN RIAN HERMAWAH
N PM : 1411010098
PEMBIMBING I : MISAR
PEMBIMBING II :
JUDUL LAPORAN : PERANGKAT LUMAK E-LEARNING ELEMEN - ELEMEN SENI TEATER BERBASIS ANDROID
TANGGAL SK : s.d (6+2 bulan)

No	HARI/TANGGAL	HASIL KONSULTASI	PARAF
1	11-3-18	ace lucu.	31.
2	23-11-18	Perbaiki Bab I & II	31.
3	7-12-18	perbaiki Caji	31.
4	11-12-18	Perbaiki. Dulu perbaiki teel risu	31.
5			31.
6	14-12-18	perbaiki Caji / perbaiki	31.
7	18-1-18	Bab I & II	31.
8		lengkap paper Bab IV + V	31.
9	25-1-18	Upload play store	31.
10	30-1-18	perbaiki Caji Bab III, IV, V	31.

*) Coret yang tidak perlu

Bandar Lampung,
Ketua Jurusan(Yuni Arifiani S.Pd., M.Kom
NIK. 0018001)



SURAT KEPUTUSAN
REKTOR IIB DARMAJAYA

NOMOR : SK.0027/DMJ/DFIK/BAAK/I-19

Tentang
Dosen Pembimbing Skripsi
Program Studi S1 Teknik Informatika

REKTOR IIB DARMAJAYA

Memperhatikan : 1. Bahwa dalam rangka usaha peningkatan mutu dan peranan IIB Darmajaya dalam melaksanakan Pendidikan Nasional perlu ditingkatkan kemampuan mahasiswa dalam Skripsi.

Menimbang : 2. Laporan dan usulan Ketua Program Studi S1 Teknik Informatika.

Mengingat : 1. Bahwa untuk mengefektifkan tenaga pengajar dalam Skripsi mahasiswa perlu ditetapkan Dosen Pembimbing Skripsi.

2. Bahwa untuk maksud tersebut dipandang perlu menerbitkan Surat Keputusan Rektor.

3. UU No.20 Tahun 2003 Tentang Sistem Pendidikan Nasional.

4. Peraturan Pemerintah No.60 Tahun 2010 tentang Pendidikan Sekolah Tinggi

5. Surat Keputusan Menteri Pendidikan Nasional Republik Indonesia No.165/D/0/2008 tertanggal 20 Agustus 2008 tentang Perubahan Status STMIK-STIE Darmajaya menjadi Informatics and Business Institute (IBI) Darmajaya

6. STATUTA IBI Darmajaya

7. Surat Ketua Yayasan Pendidikan Alfian Husin No. IM.003/YP-AH/X-08 tentang Persetujuan Perubahan Struktur Organisasi

8. Surat Keputusan Rektor 0383/DMJ/REK/X-08 tentang Struktur Organisasi.

Menetapkan

Pertama : Mengangkat nama-nama seperti tersebut dalam lampiran Surat Keputusan ini sebagai Dosen Pembimbing Skripsi mahasiswa Program Studi S1 Teknik Informatika.

Kedua : Pembimbing Skripsi berkewajiban melaksanakan tugasnya sesuai dengan jadwal yang telah ditetapkan.

Ketiga : Pembimbing Skripsi yang ditunjuk akan diberikan honorarium yang besarnya sesuai dengan ketentuan peraturan dan norma penggajian dan honorarium IBI Darmajaya.

Keempat : Surat Keputusan ini berlaku sejak tanggal ditetapkan dan apabila dikemudian hari terdapat kekeliruan dalam keputusan ini, maka keputusan ini akan ditinjau kembali.

Ditetapkan di : Bandar Lampung

Pada tanggal : 30 Januari 2019

a.n. Rektor IIB Darmajaya,

Dekan Fakultas Ilmu Komputer

Sriyanto, S.Kom., M.M., Ph.D.
NIK. 00210300

1. Ketua Jurusan S1 Teknik Informatika
2. Yang bersangkutan
3. Arsip

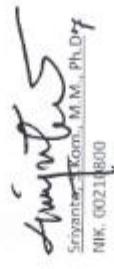
Lampiran : Surat Keputusan Rektor IIB Darmajaya
Nomor : SK.0027/DMI/DFIK/BAK/I/19
Tanggal : 30 Januari 2019
Penulis : Pembimbing Penulisan Skripsi
Program Studi Strata Satu (S1) Teknik Informatika

Judul Penulisan Skripsi dan Dosen Pembimbing
Program Studi Strata Satu (S1) Teknik Informatika

NO.	NAMA	NPM	JUDUL	PEMBIMBING
1	*Gokh Wyenda Muhammad	1411010018	Aplikasi Bahasa Pemrograman Java Disertai Compiler Berbasis Android (Studi Kasus Pemrograman Meningah IIB Darmajaya)	Apri Triansah, S.Kom, M.T.I
2	*Iskandar Saputro Bakri	1411010107	Rancang Bangun Aplikasi Kamus Senboyan PT Kereta Api Indonesia (PERSERO) Berbasis Android	Haryanto Wibowo, S.Kom, M.T.I
3	*Thiu Aan Rian Hermawan	1411010098	Perangkat Lunak E-Learning Elemen - Elemen Seni Teater Berbasis Android	Nisar, S.Kom, M.T

Keterangan * Surat Keputusan Perpanjangan

A.n. Rektor IIB Darmajaya
Dekan Fakultas Ilmu Komputer


Sriyanti, M.M., Ph.D.
NIK. 00211800

HALAMAN SPLASH SCREEN

```
package com.example.artilung.aan;

import android.content.Intent;

import android.support.v7.app.AppCompatActivity;

import android.os.Bundle;

import android.view.View;

import android.view.animation.Animation;

import android.view.animation.AnimationUtils;

import android.widget.ImageView;

import android.widget.ProgressBar;

import com.github.ybq.android.spinkit.style.ThreeBounce;

public class SplashActivity extends AppCompatActivity implements

ImageView.OnClickListener {

    Animation animasi;

    ProgressBar progress;

    @Override

    protected void onCreate(Bundle savedInstanceState) {

        super.onCreate(savedInstanceState);

        setContentView(R.layout.activity_splash);

        ProgressBar progressBar =

        (ProgressBar) findViewById(R.id.spin_kit);

        ThreeBounce ThreeBounce = new ThreeBounce();

        progressBar.setIndeterminateDrawable(ThreeBounce);
```

```
bg = (ImageView) findViewById(R.id.bg);

animasi = AnimationUt (getApplicationContext() ,
R.anim.animasi);

bg.setVisibility(View.VISIBLE);

bg.startAnimation(animasi);

Thread timer = new Thread() {

public void run() {

try {sleep(3000);} catch (InterruptedException e) {

e.printStackTrace();

} finally {Intent i = new Intent(SplashActivity.this,
HomeActivity.class);

startActivity(i);

SplashActivity.this.finish();

}

public void onAnimationEnd(Animation animation) {

if (animation == animasi) {
```

HALAMAN LOGIN

```
package com.example.bilung.thiu;

import android.app.AlertDialog;

import android.content.DialogInterface;

import android.content.Intent;

import android.support.annotation.NonNull;

import android.support.design.widget.Snackbar;

import android.support.v7.app.AppCompatActivity;

import android.os.Bundle;

import android.text.TextUtils;

import android.util.Log;

import android.view.View;

import android.widget.Button;

import android.widget.EditText;

import android.widget.ProgressBar;

import android.widget.RelativeLayout;

import android.widget.Toast;

import com.example.bilung.thiu.model.Admin;

import com.google.android.gms.auth.api.Auth;

import com.google.android.gms.auth.api.signin.GoogleSignInAccount;

import com.google.android.gms.auth.api.signin.GoogleSignInOptions;

import com.google.android.gms.auth.api.signin.GoogleSignInResult;

import com.google.android.gms.common.ConnectionResult;
```

```
import com.google.android.gms.common.SignInButton;

import com.google.android.gms.common.api.GoogleApiClient;

import com.google.android.gms.tasks.OnCompleteListener;

import com.google.android.gms.tasks.Task;

import com.google.firebase.auth.AuthCredential;

import com.google.firebase.auth.AuthResult;

import com.google.firebase.auth.FirebaseAuth;

import com.google.firebase.auth.FirebaseUser;

import com.google.firebase.auth.GoogleAuthProvider;

import com.google.firebaseio.database.DatabaseReference;

import com.google.firebaseio.database.FirebaseDatabase;

import butterknife.BindView;

import butterknife.ButterKnife;

import butterknife.OnClick;

public class LoginActivity extends AppCompatActivity implements

View.OnClickListener{

private static final String TAG = "LoginActivity";

private static final int RC_SIGN_IN = 9001;

private DatabaseReference mDatabase;

private FirebaseAuth mAuth;

private EditText edtEmail;

private EditText edtPass;

private Button btnMasuk;
```

```
super.onCreate(savedInstanceState);

setContentView(R.layout.activity_login);

mDatabase = FirebaseDatabase.getInstance().getReference();

mAuth = FirebaseAuth.getInstance();

edtEmail = (EditText) findViewById(R.id.tv_email);

edtPass = (EditText) findViewById(R.id.tv_pass);

btnMasuk = (Button) findViewById(R.id.btn_masuk);

private void signIn() {

    if (!validateForm()) {

        String email = edtEmail.getText().toString();

        String password = edtPass.getText().toString();

        mAuth.signInWithEmailAndPassword(email, password)

            .addOnCompleteListener(this, new OnCompleteListener<AuthResult>(){

        {

            public void onComplete(@NonNull Task<AuthResult> task) {

                Log.d(TAG, "signIn:onComplete:" + task.isSuccessful());

                if (task.isSuccessful()) {

                    onAuthSuccess(task.getResult().getUser());} else {

                    Toast.makeText(LoginActivity.this, "Login Error",

String email = edtEmail.getText().toString();

String password = edtPass.getText().toString();
```

```
mAuth.createUserWithEmailAndPassword(email, password)

    .addOnCompleteListener(this, new OnCompleteListener<AuthResult>()
    {

        public void onComplete(@NonNull Task<AuthResult> task) {
            Log.d(TAG, "createUser:onComplete:" + task.isSuccessful());

            if (task.isSuccessful()) {
                onAuthSuccess(task.getResult().getUser());
                Toast.makeText(LoginActivity.this, "Sign Up Failed",
                        Toast.LENGTH_SHORT).show();
            }
        }

        private void onAuthSuccess(FirebaseUser user) {
            String username = usernameFromEmail(user.getEmail());
            writeNewAdmin(user.getUid(), username, user.getEmail());
            startActivity(new Intent(LoginActivity.this, AwalActivity.class));
            finish();
        }

        private boolean validateForm() {boolean result = true;
            if (TextUtils.isEmpty(edtEmail.getText().toString())) {
                edtEmail.setError("Required");
                result = false;  if
                    (TextUtils.isEmpty(edtPass.getText().toString())) {
                edtPass.setError("Required");result = false;
            }
        }
    }
}
```

```
private void writeNewAdmin(String userId, String name, String email) {  
  
    Admin admin = new Admin(name, email);  
  
    mDatabase.child("admins").child(userId).setValue(admin);  
  
    public void onClick(View v) {int i = v.getId();if (i ==  
R.id.btn_masuk)  
{signIn();
```

HALAMAN MENU UTAMA

```
package com.example.artilung.aan;

import android.content.Intent;

import android.support.v7.app.AppCompatActivity;

import android.os.Bundle;

import android.view.View;

import android.widget.Button;

public class HomeActivity extends AppCompatActivity {

    private Button btnChat, btnTentang, btnArtikel;

    protected void onCreate(Bundle savedInstanceState) {

        super.onCreate(savedInstanceState);

        setContentView(R.layout.activity_home);

        btnChat = (Button) findViewById(R.id.btnChat);

        btnArtikel = (Button) findViewById(R.id.btnArtikel);

        btnTentang = (Button) findViewById(R.id.btnTentang);

        btnArtikel.setOnClickListener(new View.OnClickListener() {

            Intent intent = new Intent(HomeActivity.this, MainActivity.class);

            startActivity(intent);

        });

        public void onClick(View v) {

            Intent intent = new Intent(HomeActivity.this, ChatActivity.class);

            startActivity(intent);

        }

    }

}
```

HALAMAN UPLOAD

```
package com.example.bilung.thiu;
```

```
import android.app.AlertDialog;

import android.content.ContentResolver;

import android.content.Context;

import android.content.Intent;

import android.graphics.Bitmap;

import android.net.Uri;

import android.provider.MediaStore;

import android.support.annotation.NonNull;

import android.support.v7.app.AppCompatActivity;

import android.os.Bundle;

import android.support.v7.widget.RecyclerView;

import android.text.TextUtils;

import android.util.Log;

import android.view.Menu;

import android.view.MenuItem;

import android.view.View;

import android.webkit.MimeTypeMap;

import android.widget.Button;

import android.widget.EditText;

import android.widget.ImageButton;

import android.widget.Toast;

import com.example.bilung.thiu.m_Firebase.FirebaseHelper;

import com.example.bilung.thiu.model.Teater;
```

```
import com.google.android.gms.tasks.Continuation;
import com.google.android.gms.tasks.OnCompleteListener;
import com.google.android.gms.tasks.OnFailureListener;
import com.google.android.gms.tasks.OnSuccessListener;
import com.google.android.gms.tasks.Task;
import com.google.firebase.auth.FirebaseAuth;
import com.google.firebase.auth.FirebaseUser;
import com.google.firebaseio.database.DataSnapshot;
import com.google.firebaseio.database.DatabaseError;
import com.google.firebaseio.database.DatabaseReference;
import com.google.firebaseio.database.FirebaseDatabase;
import com.google.firebaseio.database.ValueEventListener;
import com.google.firebase.storage.FirebaseStorage;
import com.google.firebase.storage.OnPausedListener;
import com.google.firebase.storage.OnProgressListener;
import com.google.firebase.storage.StorageReference;
import com.google.firebase.storage.UploadTask;
import com.ittianyu.bottomnavigationviewex.BottomNavigationViewEx;
import java.io.FileNotFoundException;
import java.io.IOException;

public class UploadActivity extends AppCompatActivity {
    private static final String TAG = "Uploadactivity";
```

```
private static final int ACTIVITY_NUM = 2;

private Context mContext = UploadActivity.this;

private ImageButton btnImages;

private EditText textNama;

private EditText textTitle;

private EditText textDesc;

private Button btnUpload;

DatabaseReference db;

FirebaseHelper helper;

public static final String FB_STORAGE_PATH = "image/";

public static final String FB_DATABASE_PATH = "Teater";

private StorageReference storage;

private FirebaseDatabase database;

private FirebaseAuth mAuth;

private DatabaseReference mDatabaseUsers;

private FirebaseUser mCurrentUser;

private Uri imgUri;

private UploadTask uploadTask;

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);
```

```
setContentView(R.layout.activity_upload

db = FirebaseDatabase.getInstance().getReference();

helper = new FirebaseHelper(db);

btnImages = (ImageButton) findViewById(R.id.btnImage);

textNama = (EditText) findViewById(R.id.nametv);

textTitle = (EditText) findViewById(R.id.titletv);

textDesc = (EditText) findViewById(R.id.deskripsitv);

bnUpload = (Button) findViewById(R.id.btnUpload);

mCurrentUser = mAuth.getCurrentUser();

mDatabaseUsers =
FirebaseDatabase.getInstance().getReference("Teater");

btnImages.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View view) {

Intent galleryIntent = new Intent(Intent.ACTION_GET_CONTENT);

galleryIntent.setType("image/*");

startActivityForResult(galleryIntent, GALLERY_REQUEST_CODE);

protected void onActivityResult(int requestCode, int resultCode,
Intent data) {

super.onActivityResult(requestCode, resultCode, data);

if (requestCode == GALLERY_REQUEST_CODE && resultCode ==
RESULT_OK && data != null && data.getData() != null) {

System.out.println("gambar "+imgUri);try {public void clickupload
() {

btnUpload.setOnClickListener(new View.OnClickListener() {
```

```
final ProgressDialog dialog = new
ProgressDialog(UploadActivity.this);

dialog.setTitle("Uploading image");

dialog.show();

final StorageReference ref = storage.child(FB_STORAGE_PATH +
System.currentTimeMillis() + "." + getImageExt(imgUri));

uploadTask = ref.putFile(imgUri);

Task<Uri> urlTask = uploadTask.continueWithTask(new

throws Exception {if (!task.isSuccessful()) {throw
task.getException();}

@Override

public void onComplete(@NonNull Task<Uri> task) {

if (task.isSuccessful()) {Uri downloadUri = task.getResult();

dialog.dismiss();Teater imageUpload = new Teater();

imageUpload.setName(textNama.getText().toString());

imageUpload.setTitle(textTitle.getText().toString());

imageUpload.setDeskripsi(textDesc.getText().toString());

imageUpload.setUrl(downloadUri.toString());

String uploadId = mDatabaseUsers.push().getKey();

mDatabaseUsers.child(uploadId).setValue(imageUpload);

Toast.makeText(UploadActivity.this, "Upload Sukses",
Toast.LENGTH_SHORT).show();} else {dialog.dismiss();

Toast.makeText(UploadActivity.this, "Upload Gagal",
```

```
Toast.LENGTH_SHORT).show();  
  
public void onProgress(UploadTask.TaskSnapshot taskSnapshot) {  
  
    double progress = (100.0 * taskSnapshot.getBytesTransferred()) /  
  
    taskSnapshot.getTotalByteCount();
```

HALAMAN ARTIKEL

```
package com.example.artilung.aan;

import android.support.v7.app.AppCompatActivity;

import android.os.Bundle;

import android.widget.ImageView;

import android.widget.TextView;

import com.squareup.picasso.Picasso;

public class DetailActivity extends AppCompatActivity {

    TextView nameTxt, descTxt, titleTxt;

    ImageView gambar;

    @Override

    protected void onCreate(Bundle savedInstanceState) {

        super.onCreate(savedInstanceState);

        setContentView(R.layout.activity_detail);

        titleTxt = (TextView) findViewById(R.id.titletv);

        nameTxt = (TextView) findViewById(R.id.nametv);

        descTxt= (TextView) findViewById(R.id.destv);

        gambar = (ImageView) findViewById(R.id.gambar);

        titleTxt.setText(getIntent().getStringExtra("TITLE"));

        descTxt.setText(getIntent().getStringExtra("DESKRIPSI"));

    }

}
```

HALAMAN PROFIL

```
package com.example.artilung.aan;

import android.content.Context;

import android.support.v7.app.AppCompatActivity;

import android.os.Bundle;

import android.util.Log;

import android.widget.ImageView;

import android.widget.ViewFlipper;

public class ProfilActivity extends AppCompatActivity {

    private static final String TAG = "ProfilActivity";

    private static final int ACTIVITY_NUM = 4;

    private Context mContext = ProfilActivity.this;

    protected void onCreate(Bundle savedInstanceState) {

        super.onCreate(savedInstanceState);

        setContentView(R.layout.activity_profil);

        int images [] ={R.drawable.bgsliderdj, R.drawable.logodkl,
        R.drawable.

        bgsliderkombir, R.drawable.bgsliderthiu};

        flipper = findViewById(R.id.flipper);

        for (int image: images){flipperImages(image)

    public void flipperImages (int image){

        ImageViewimageView = new ImageView(this);

        imageView.setImageResource(image);

        flipper.addView(imageView);
```

```
flipper.setFlipInterval(4000); // 4 detik  
  
flipper.setAutoStart(true);  
  
flipper.setInAnimation(this, android.R.anim.slide_in_left);  
  
flipper.setOutAnimation(this, android.R.anim.slide_out_right);
```

SOURCE CODE PROGRAM

HALAMAN SPLASH SCREEN

```
package com.example.artilung.aan;

import android.content.Intent;

import android.support.v7.app.AppCompatActivity;

import android.os.Bundle;

import android.view.View;

import android.view.animation.Animation;

import android.view.animation.AnimationUtils;

import android.widget.ImageView;

import android.widget.ProgressBar;

import

com.github.ybq.android.spinkit.style.ThreeBounce;

public class SplashActivity extends AppCompatActivity

implements

ImageView bg;

Animation animasi;

ProgressBar progress;

@Override

protected void onCreate(Bundle savedInstanceState) {
```

```
super.onCreate(savedInstanceState);

setContentView(R.layout.activity_splash);

ProgressBar progressBar =
(ProgressBar) findViewById(R.id.spin_kit);

ThreeBounce ThreeBounce = new ThreeBounce();

progressBar.setIndeterminateDrawable(ThreeBounce);

bg = (ImageView) findViewById(R.id.bg);

animasi = AnimationUt (getApplicationContext(),
R.anim.animasi);

bg.setVisibility(View.VISIBLE);

bg.startAnimation(animasi);

Thread timer = new Thread() {

public void run() {

try {sleep(3000);} catch (InterruptedException e) {

e.printStackTrace();

} finally {Intent i = new Intent(SplashActivity.this,
HomeActivity.class);

startActivity(i);

SplashActivity.this.finish();
}
}
```

HALAMAN LOGIN

```
package com.example.bilung.thiu;

import android.app.AlertDialog;

import android.content.DialogInterface;

import android.content.Intent;

import android.support.annotation.NonNull;

import android.support.design.widget.Snackbar;

import android.support.v7.app.AppCompatActivity;

import android.os.Bundle;

import android.text.TextUtils;

import android.util.Log;

import android.view.View;

import android.widget.Button;

import android.widget.EditText;

import android.widget.ProgressBar;

import android.widget.RelativeLayout;

import android.widget.Toast;

import com.example.bilung.thiu.model.Admin;

import com.google.android.gms.auth.api.Auth;

import com.google.android.gms.auth.api.signin.GoogleSignInAccount;
```

```
import  
com.google.android.gms.auth.api.signin.GoogleSignInOptions;  
  
import  
com.google.android.gms.auth.api.signin.GoogleSignInResult;  
  
import com.google.android.gms.common.ConnectionResult;  
  
import com.google.android.gms.common.SignInButton;  
  
import  
com.google.android.gms.common.api.GoogleApiClient;  
  
import com.google.android.gms.tasks.OnCompleteListener;  
  
import com.google.android.gms.tasks.Task;  
  
import com.google.firebase.auth.AuthCredential;  
  
import com.google.firebase.auth.AuthResult;  
  
import com.google.firebase.auth.FirebaseAuth;  
  
import com.google.firebase.auth.FirebaseUser;  
  
import com.google.firebase.auth.GoogleAuthProvider;  
  
import com.google.firebaseio.database.DatabaseReference;  
  
import com.google.firebaseio.database.FirebaseDatabase;  
  
import butterknife.BindView;  
  
import butterknife.ButterKnife;  
  
import butterknife.OnClick;
```

```
public class LoginActivity extends AppCompatActivity
implements View.OnClickListener{

private static final String TAG = "LoginActivity";

private static final int RC_SIGN_IN = 9001;

private DatabaseReference mDatabase;

private FirebaseAuth mAuth;

private EditText edtEmail;

private EditText edtPass;

private Button btnMasuk;

super.onCreate(savedInstanceState);

setContentView(R.layout.activity_login);

mDatabase =
FirebaseDatabase.getInstance().getReference();

mAuth = FirebaseAuth.getInstance();

edtEmail = (EditText) findViewById(R.id.tv_email);

edtPass = (EditText) findViewById(R.id.tv_pass);

btnMasuk = (Button) findViewById(R.id.btn_masuk);

private void signIn() {

if (!validateForm()) {

String email = edtEmail.getText().toString();
```

```
private DatabaseReference mDatabase;

private FirebaseAuth mAuth;

private EditText edtEmail;

private EditText edtPass;

private Button btnMasuk;

super.onCreate(savedInstanceState);

String password = edtPass.getText().toString();

mAuth.signInWithEmailAndPassword(email, password)

.addOnCompleteListener(this, new

OnCompleteListener<AuthResult>() {

public void onComplete(@NonNull Task<AuthResult> task)

{

Log.d(TAG, "signIn:onComplete:" + task.isSuccessful());

if (task.isSuccessful()) {

onAuthSuccess(task.getResult().getUser());} else {

Toast.makeText(LoginActivity.this, "Login Error",

String email = edtEmail.getText().toString();

String password = edtPass.getText().toString();

mAuth.createUserWithEmailAndPassword(email, password)

.addOnCompleteListener(this, new

OnCompleteListener<AuthResult>() {
```

```
String password = edtPass.getText().toString();

mAuth.signInWithEmailAndPassword(email, password)

.addOnCompleteListener(this, new
OnCompleteListener<AuthResult>() {
    public void onComplete(@NonNull Task<AuthResult> task)
    {

        Log.d(TAG, "signIn:onComplete:" + task.isSuccessful());

        if (task.isSuccessful()) {

            onAuthSuccess(task.getResult().getUser()); } else {

            Toast.makeText(LoginActivity.this, "Login Error",
            }

        }

        String email = edtEmail.getText().toString();

        String password = edtPass.getText().toString();

        mAuth.createUserWithEmailAndPassword(email, password)

.addOnCompleteListener(this, new
OnCompleteListener<AuthResult>() {
    public void onComplete(@NonNull Task<AuthResult> task)
    {

        Log.d(TAG, "createUser:onComplete:" +
        task.isSuccessful());

        if (task.isSuccessful()) {

            onAuthSuccess(task.getResult().getUser());
            }

        }

    }
}
```

```
Toast.makeText(LoginActivity.this, "Sign Up Failed",
Toast.LENGTH_SHORT).show();

private void onAuthSuccess(FirebaseUser user) {
    String username = usernameFromEmail(user.getEmail());
    writeNewAdmin(user.getUid(), username,
    user.getEmail());
    startActivity(new Intent(LoginActivity.this,
    AwalActivity.class));
    finish();
}

private boolean validateForm() {boolean result = true;
if (TextUtils.isEmpty(edtEmail.getText().toString())) {
    edtEmail.setError("Required");
    result = false; if
    (TextUtils.isEmpty(edtPass.getText().toString())) {
        edtPass.setError("Required");result = false;
}

private void writeNewAdmin(String userId, String name,
String email) {
    Admin admin = new Admin(name, email);
    mDatabase.child("admins").child(userId).setValue(admin)
    ;
```

HALAMAN MENU UTAMA

```
package com.example.artilung.aan;

import android.content.Intent;

import android.support.v7.app.AppCompatActivity;

import android.os.Bundle;

import android.view.View;

import android.widget.Button;

public class HomeActivity extends AppCompatActivity {

    private Button btnChat, btnTentang, btnArtikel;

    protected void onCreate(Bundle savedInstanceState) {

        super.onCreate(savedInstanceState);

        setContentView(R.layout.activity_home);

        btnChat = (Button) findViewById(R.id.btnChat);

        btnArtikel = (Button) findViewById(R.id.btnArtikel);

        btnTentang = (Button) findViewById(R.id.btnTentang);

        btnArtikel.setOnClickListener(new

View.OnClickListener() {

            Intent intent = new Intent(HomeActivity.this,
MainActivity.class);

            startActivity(intent);

```

HALAMAN UPLOAD

```
package com.example.bilung.thiu;

import android.app.ProgressDialog;

import android.content.ContentResolver;

import android.content.Context;

import android.content.Intent;

import android.graphics.Bitmap;

import android.net.Uri;

import android.provider.MediaStore;

import android.support.annotation.NonNull;

import android.support.v7.app.AppCompatActivity;

import android.os.Bundle;

import android.support.v7.widget.RecyclerView;

import android.text.TextUtils;

import android.util.Log;

import android.view.Menu;

import android.view.MenuItem;

import android.view.View;

import android.webkit.MimeTypeMap;

import android.widget.Button;

import android.widget.EditText;
```

```
import android.widget.ImageButton;

import android.widget.Toast;

import

com.example.bilung.thiu.m_Firebase.FirebaseHelper;

import com.example.bilung.thiu.model.Teater;

import com.google.android.gms.tasks.Continuation;

import com.google.android.gms.tasks.OnCompleteListener;

import com.google.android.gms.tasks.OnFailureListener;

import com.google.android.gms.tasks.OnSuccessListener;

import com.google.android.gms.tasks.Task;

import com.google.firebase.auth.FirebaseAuth;

import com.google.firebaseio.auth.FirebaseUser;

import com.google.firebaseio.database.DataSnapshot;

import com.google.firebaseio.database.DatabaseError;

import com.google.firebaseio.database.DatabaseReference;

import com.google.firebaseio.database.FirebaseDatabase;

import com.google.firebaseio.database.ValueEventListener;

import com.google.firebaseio.storage.FirebaseStorage;

import com.google.firebaseio.storage.OnPausedListener;

import com.google.firebaseio.storage.OnProgressListener;

import com.google.firebaseio.storage.StorageReference;
```

```
import com.google.firebase.storage.UploadTask;

import
com.ittianyu.bottomnavigationviewex.BottomNavigationViewEx;

import java.io.FileNotFoundException;

import java.io.IOException;

public class UploadActivity extends AppCompatActivity {

    private static final String TAG = "Uploadactivity";
    private static final int ACTIVITY_NUM = 2;
    private Context mContext = UploadActivity.this;
    private ImageButton btnImages;
    private EditText textNama;
    private EditText textTitle;
    private EditText textDesc;
    private Button btnUpload;
    DatabaseReference db;
    FirebaseHelper helper;
    public static final String FB_STORAGE_PATH = "image/";
    public static final String FB_DATABASE_PATH = "Teater";
    private StorageReference storage;
```

```
import com.google.firebase.storage.UploadTask;

import
com.ittianyu.bottomnavigationviewex.BottomNavigationViewEx;

import java.io.FileNotFoundException;

import java.io.IOException;

public class UploadActivity extends AppCompatActivity {

    private static final String TAG = "Uploadactivity";
    private static final int ACTIVITY_NUM = 2;
    private Context mContext = UploadActivity.this;
    private ImageButton btnImages;
    private EditText textNama;
    import android.widget.ImageButton;
    import android.widget.Toast;
    import
    com.example.bilung.thiu.m_Firebase.FirebaseHelper;
    import com.example.bilung.thiu.model.Teater;
    import com.google.android.gms.tasks.Continuation;
    import com.google.android.gms.tasks.OnCompleteListener;
    import com.google.android.gms.tasks.OnFailureListener;
    import com.google.android.gms.tasks.OnSuccessListener;
```

```
import com.google.android.gms.tasks.Task;

import com.google.firebase.auth.FirebaseAuth;

import com.google.firebase.auth.FirebaseUser;

import com.google.firebaseio.database.DataSnapshot;

import com.google.firebaseio.database.DatabaseError;

import com.google.firebaseio.database.DatabaseReference;

import com.google.firebaseio.database.FirebaseDatabase;

import com.google.firebaseio.database.ValueEventListener;

import com.google.firebaseio.storage.FirebaseStorage;

import com.google.firebaseio.storage.OnPausedListener;

import com.google.firebaseio.storage.OnProgressListener;

import com.google.firebaseio.storage.StorageReference;

import com.google.firebaseio.storage.UploadTask;

import

com.ittianyu.bottomnavigationviewex.BottomNavigationViewEx;

import java.io.FileNotFoundException;

import java.io.IOException;

public class UploadActivity extends AppCompatActivity {

private static final String TAG = "Uploadactivity";

private static final int ACTIVITY_NUM = 2;
```

```
private FirebaseDatabase database;

private FirebaseAuth mAuth;

private DatabaseReference mDatabaseUsers;

private FirebaseUser mCurrentUser;

private Uri imgUri;

private UploadTask uploadTask;

@Override

protected void onCreate(Bundle savedInstanceState) {

    super.onCreate(savedInstanceState);

    setContentView(R.layout.activity_upload

    db = FirebaseDatabase.getInstance().getReference();

    helper = new FirebaseHelper(db);

    btnImages = (ImageButton) findViewById(R.id.btnImage);

    textNama = (EditText) findViewById(R.id.nametv);

    textTitle = (EditText) findViewById(R.id.titletv);

    textDesc = (EditText) findViewById(R.id.deskripsitv);

    bnUpload = (Button) findViewById(R.id.btnUpload);

    mCurrentUser = mAuth.getCurrentUser();

    mDatabaseUsers =
        FirebaseDatabase.getInstance().getReference("Teater");
```

```
btnImages.setOnClickListener(new View.OnClickListener() {
{
    @Override
    public void onClick(View view) {
        Intent galleryIntent = new Intent(Intent.ACTION_GET_CONTENT);
        galleryIntent.setType("image/*");
        startActivityForResult(galleryIntent,
        GALLERY_REQUEST_CODE);
    }
    protected void onActivityResult(int requestCode, int resultCode, Intent data) {
        super.onActivityResult(requestCode, resultCode, data);
        if (requestCode == GALLERY_REQUEST_CODE && resultCode ==
        RESULT_OK && data != null && data.getData() != null) {
            System.out.println("gambar "+imgUri);
            try {
                clickupload();
            }
            btnUpload.setOnClickListener(new View.OnClickListener() {
            {
                final ProgressDialog dialog = new
                ProgressDialog(UploadActivity.this);
                dialog.setTitle("Uploading image");
                dialog.show();
            }
        }
    }
}
```

```
final StorageReference ref =
storage.child(FB_STORAGE_PATH +
System.currentTimeMillis() + "." +
getImageExt(imgUri));

uploadTask = ref.putFile(imgUri);

Task<Uri> urlTask = uploadTask.continueWithTask(new
throws Exception {if (!task.isSuccessful()) {throw
task.getException();}

@Override
public void onComplete(@NonNull Task<Uri> task) {
if (task.isSuccessful()) {Uri downloadUri =
task.getResult();
dialog.dismiss();Teater imageUpload = new Teater();
imageUpload.setName(textNama.getText().toString());
imageUpload.setTitle(textTitle.getText().toString());
imageUpload.setDeskripsi(textDesc.getText().toString());
;
imageUpload.setUrl(downloadUri.toString());
String uploadId = mDatabaseUsers.push().getKey();
mDatabaseUsers.child(uploadId).setValue(imageUpload);
Toast.makeText(UploadActivity.this, "Upload Sukses",

```

HALAMAN ARTIKEL

```
package com.example.artilung.aan;

import android.support.v7.app.AppCompatActivity;

import android.os.Bundle;

import android.widget.ImageView;

import android.widget.TextView;

import com.squareup.picasso.Picasso;

public class DetailActivity extends AppCompatActivity {

    TextView nameTxt, descTxt, titleTxt;

    ImageView gambar;

    @Override

    protected void onCreate(Bundle savedInstanceState) {

        super.onCreate(savedInstanceState);

        setContentView(R.layout.activity_detail);

        titleTxt = (TextView) findViewById(R.id.titletv);

        nameTxt = (TextView) findViewById(R.id.nametv);

        descTxt= (TextView) findViewById(R.id.destv);

        gambar = (ImageView) findViewById(R.id.gambar);

        titleTxt.setText(getIntent().getStringExtra("TITLE"));

    }
}
```

HALAMAN PROFIL

```
package com.example.artilung.aan;

import android.content.Context;

import android.support.v7.app.AppCompatActivity;

import android.os.Bundle;

import android.util.Log;

import android.widget.ImageView;

import android.widget.ViewFlipper;

public class ProfilActivity extends AppCompatActivity {

    private static final String TAG = "ProfilActivity";
    private static final int ACTIVITY_NUM = 4;
    private Context mContext = ProfilActivity.this;

    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_profil);

        int images [] ={R.drawable.bgsliderdj, R.drawable.
logodkl, R.drawable.
bgsliderkombir, R.drawable.bgsliderthiu};

        flipper = findViewById(R.id.flipper);

        for (int image: images){flipperImages(image)}
    }

    public void flipperImages (int image){

        ImageViewimageView = new ImageView(this);
```

```
imageView.setImageResource(image);

flipper.addView(imageView);

flipper.setFlipInterval(4000); // 4 detik

flipper.setAutoStart(true);

flipper.setInAnimation(this,
    android.R.anim.slide_in_left);

flipper.setOutAnimation(this,
    android.R.anim.slide_out_right);

if (task.isSuccessful()) {Uri downloadUri =
task.getResult();

dialog.dismiss();Teater imageUpload = new Teater();

imageUpload.setName(textNama.getText().toString());

imageUpload.setTitle(textTitle.getText().toString());

imageUpload.setDeskripsi(textDesc.getText().toString())

public void onClick(View view) {

Intent galleryIntent = new
Intent(Intent.ACTION_GET_CONTENT);

galleryIntent.setType("image/*");

startActivityForResult(galleryIntent,
GALLERY_REQUEST_CODE);

protected void onActivityResult(int requestCode, int
resultCode, Intent data) {

super.onActivityResult(requestCode, resultCode, data);
```