



**PERANCANGAN APLIKASI PEMESANAN TIKET WISATA *ONLINE*
BERBASIS ANDROID MENGGUNAKAN *FIREBASE***

Disusun dan Diajukan Untuk Memenuhi Persyaratan Ujian Akhir
Memperoleh Gelar Sarjana Komputer Pada Fakultas Sains dan Teknologi
Universitas Pembangunan Panca Budi
Medan

SKRIPSI

OLEH:

NAMA : ANDI HAKIM NASUTION
NPM : 1514370779
PROGRAM STUDI : SISTEM KOMPUTER

**FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS PEMBANGUNAN PANCA BUDI
MEDAN
2019**

LEMBAR PENGESAHAN

PERANCANGAN APLIKASI PEMESANAN TIKET WISATA ONLINE BERBASIS ANDROID MENGGUNAKAN *FIREBASE*

Disusun Oleh:

Nama : Andi Hakim Nasution
NPM : 1514370779
Program Studi : Sistem Komputer

Skripsi Telah Disetujui oleh Dosen Pembimbing Skripsi

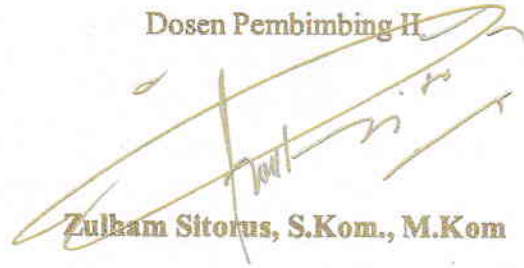
Pada Tanggal: 07 November 2019

Dosen Pembimbing I



Herdianto, S.Kom., M.T

Dosen Pembimbing II



Zulham Sitorus, S.Kom., M.Kom

Mengetahui,

Dekan Fakultas Sains Dan Teknologi



Sri Shindi Indira, S.T., M.Sc

Ketua Program Studi



Eko Hariyanto, S.Kom., M.Kom

SURAT PERNYATAAN

Saya yang bertanda tangan di bawah ini :

Nama : Andi Hakim Nasution
NPM : 1514370779
Prodi : Sistem Komputer
Konsentrasi : Keamanan Jaringan Komputer (KJK)
Judul Skripsi : Perancangan Aplikasi Pemesanan Tiket
Wisata *Online*
Berbasis Android Menggunakan *Firestore*

Dengan ini menyatakan bahwa :

1. Tugas Akhir/Skripsi saya bukan hasil Plagiat
2. Saya tidak akan menuntut perbaikan nilai Indeks Prestasi Kumulatif (IPK) setelah ujian Sidang Meja Hijau
3. Skripsi saya dapat dipublikasikan oleh pihak lembaga, dan saya
4. tidak akan menuntut akibat publikasi tersebut

Demikian pernyataan ini saya perbuat dengan sebenar-benarnya,
terima kasih

Medan, 03 Desember 2019

Yang membuat pernyataan



Andi Hakim Nasution

PERNYATAAN ORISINALITAS

Dengan ini saya menyatakan bahwa skripsi ini tidak terdapat karya yang pernah diajukan untuk memperoleh gelar kesarjanaan di perguruan tinggi, dan sepanjang pengetahuan saya juga terdapat karya atau pendapat yang pernah ditulis atau diterbitkan orang lain, kecuali yang secara tertulis diacu dalam skripsi ini dan disebutkan dalam daftar pustaka.

Medan, 03 Desember 2019



Andi Hakim Nasution

1514370779

Plagiarism Detector v. 1092 - Originality Report:

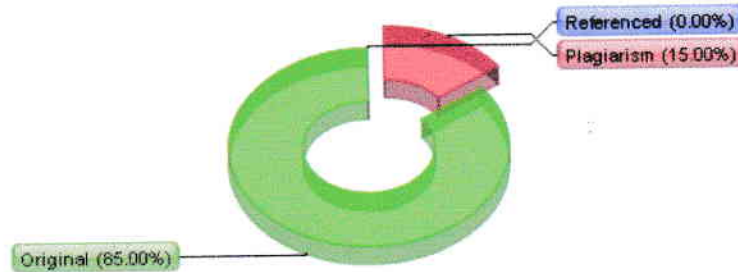
Analyzed document: 25/10/2019 09:58:37

"ANDI HAKIM NASUTION_1514370779_SISTEM KOMPUNTER.docx"

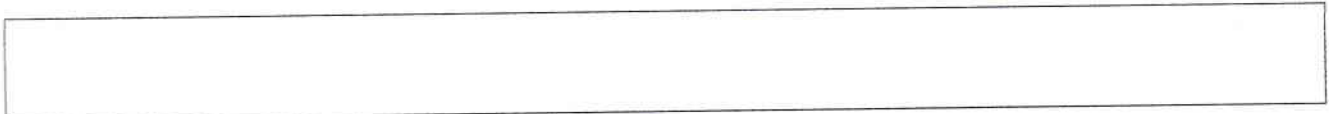
Licensed to: Universitas Pembangunan Panca Budi_License4



Relation chart:



Distribution graph:



Comparison Preset: Rewrite. Detected language: Indonesian

Top sources of plagiarism:

% 12	wrds: 271	https://www.slideshare.net/LidyaSeptiani/analisis-sistem-informasi-pemesanan-tiket-berbasi...
% 6	wrds: 130	http://eprints.umk.ac.id/1392/1/Hal.Judul.pdf
% 4	wrds: 92	http://elearning.upnjatim.ac.id/courses/MOBILETECHNOLOGYGANJIL20132014/work/52405e6d24ca12...

[Show other Sources:]

Processed resources details:

119 - Ok / 15 - Failed

[Show other Sources:]

Important notes:

Wikipedia:	Google Books:	Ghostwriting services:	Anti-cheating:
[not detected]	[not detected]	[not detected]	[not detected]



YAYASAN PROF. DR. H. KADIRUN YAHYA
UNIVERSITAS PEMBANGUNAN PANCA BUDI
LABORATORIUM KOMPUTER
Jl. Jend. Gatot Subroto Km 4,5 Sei Sikambing Telp. 061-8455571
Medan - 20122

KARTU BEBAS PRAKTIKUM

Yang bertanda tangan dibawah ini Ka. Laboratorium Komputer dengan ini menerangkan bahwa :

Nama : ANDI HAKIM NASUTION
N.P.M. : 1514370779
Tingkat/Semester : Akhir
Fakultas : SAINS & TEKNOLOGI
Jurusan/Prodi : Sistem Komputer

Benar dan telah menyelesaikan urusan administrasi di Laboratorium Komputer Universitas Pembangunan Panca Budi Medan.

Medan, 02 November 2019
Ka. Laboratorium



Telah Diperiksa oleh LPMU
dengan Plagiarisme... 15... %

04 NOV 2019

FM-BPAA-2012-041

Hal : Permohonan Meja Hijau



Medan, 02 November 2019
Kepada Yth : Bapak/Ibu Dekan
Fakultas SAINS & TEKNOLOGI
UNPAB Medan
Di -
Tempat

Telah di terima
berkas persyaratan
dapat di proses
Medan, 04/11/2019

Ka. BPAA

an.

TEGUH WAHYUNO, SE, MM.

Dengan hormat, saya yang bertanda tangan di bawah ini :

Nama : ANDI HAKIM NASUTION
Tempat/Tgl. Lahir : Medan / 6 Januari 1995
Nama Orang Tua : Husni Tamrin Nasution
N. P. M : 1514370779
Fakultas : SAINS & TEKNOLOGI
Program Studi : Sistem Komputer
No. HP : 085370725811
Alamat : Jl. Pemasarakatan Gg Hikmah

Datang bermohon kepada Bapak/Ibu untuk dapat diterima mengikuti Ujian Meja Hijau dengan judul **Perancangan Aplikasi Pemesanan Tiket Wisata Online Berbasis Android menggunakan Firebase**, Selanjutnya saya menyatakan :

1. Melampirkan KKM yang telah disahkan oleh Ka. Prodi dan Dekan
2. Tidak akan menuntut ujian perbaikan nilai mata kuliah untuk perbaikan indek prestasi (IP), dan mohon diterbitkan ijazahnya setelah lulus ujian meja hijau.
3. Telah tercap keterangan bebas pustaka
4. Terlampir surat keterangan bebas laboratorium
5. Terlampir pas photo untuk ijazah ukuran 4x6 = 5 lembar dan 3x4 = 5 lembar Hitam Putih
6. Terlampir foto copy STTB SLTA dilegalisir 1 (satu) lembar dan bagi mahasiswa yang lanjutan D3 ke S1 lampirkan ijazah dan transkripnya sebanyak 1 lembar.
7. Terlampir pelunasan kwintasi pembayaran uang kuliah berjalan dan wisuda sebanyak 1 lembar
8. Skripsi sudah dijilid lux 2 exemplar (1 untuk perpustakaan, 1 untuk mahasiswa) dan jilid kertas jeruk 5 examplar untuk penguji (bentuk dan warna penjiilidan diserahkan berdasarkan ketentuan fakultas yang berlaku) dan lembar persetujuan dosen pembimbing, prodi dan dekan
9. Soft Copy Skripsi disimpan di CD sebanyak 2 disc (Sesuai dengan Judul Skripsinya)
10. Terlampir surat keterangan BKKOL (pada saat pengambilan ijazah)
11. Setelah menyelesaikan persyaratan point-point diatas berkas di masukan kedalam MAP
12. Bersedia melunaskan biaya-biaya uang dibebankan untuk memproses pelaksanaan ujian dimaksud, dengan perincian sbb :

1. [102] Ujian Meja Hijau	: Rp.	100.000
2. [170] Administrasi Wisuda	: Rp.	1.500.000
3. [202] Bebas Pustaka	: Rp.	100.000
4. [221] Bebas LAB	: Rp.	5.000
Total Biaya	: Rp.	1.605.000
5. Uang kuliah	Rp.	1.705.000
		2.875.000
		4.580.000

7/4/19
Dir

Ukuran Toga : M



Hormat saya

ANDI HAKIM NASUTION
1514370779

Catatan :

- 1. Surat permohonan ini sah dan berlaku bila :
 - a. Telah dicap Bukti Pelunasan dari UPT Perpustakaan UNPAB Medan.
 - b. Melampirkan Bukti Pembayaran Uang Kuliah aktif semester berjalan
- 2. Dibuat Rangkap 3 (tiga), untuk - Fakultas - untuk BPAA (asli) - Mhs.ybs.





UNIVERSITAS PEMBANGUNAN PANCA BUDI
FAKULTAS SAINS & TEKNOLOGI
 Jl. Jend. Gatot Subroto Km. 4,5 Telp (061) 8455571
 website : www.pancabudi.ac.id email: unpub@pancabudi.ac.id
 Medan - Indonesia

Universitas : Universitas Pembangunan Panca Budi
 Fakultas : SAINS & TEKNOLOGI
 Dosen Pembimbing I : HEBDIANTO, S.KOM, M.T
 Dosen Pembimbing II : ZULHAM CITORUS, S.Kom, M.Kom
 Nama Mahasiswa : ANDI HAKIM NASUTION
 Jurusan/Program Studi : Sistem Komputer
 Nomor Pokok Mahasiswa : 1514370779
 Bidang Pendidikan : STRATA 1
 Judul Tugas Akhir/Skripsi : PERANCANGAN APLIKASI PEMESANAN TIKET WISATA;
 ONLINE BERBASIS ANDROID MENGGUNAKAN FIREBASE

TANGGAL	PEMBAHASAN MATERI	PARAF	KETERANGAN
15-2019	Don'yankon bab 1. Ilmu bahasa mabalah. Sebrilian di pabrak pmlisan dan pabrak	<i>[Signature]</i>	are sbrngn <i>[Signature]</i>
20/8-2019	Donyankon bab 2. Tamululu pabrak Mugeneri andri & studio, pabrak apn-apn sbrngn as br di pabrak am ukurpabrak pabrak an pabrak Mabala mabrak.	<i>[Signature]</i>	
19-2019	Donyankon bab 3. Pabrak megan Pabrak	<i>[Signature]</i>	
19/9-2019	Donyankon bab 4. pabrak an pabrak	<i>[Signature]</i>	
1/10-2019	Pabrak pabrak	<i>[Signature]</i>	
1/10-2019	Mel sbrngn	<i>[Signature]</i>	
1/11-2019	Mel sbrngn	<i>[Signature]</i>	

Medan, 10 April 2019
 Diketahui/Disetujui oleh :
 Dekan,



Sri Shindi Indra, S.T., M.Sc.



UNIVERSITAS PEMBANGUNAN PANCA BUDI

FAKULTAS SAINS & TEKNOLOGI

Jl. Jend. Gatot Subroto Km. 4,5 Telp (061) 8455571
website : www.pancabudi.ac.id email: unpab@pancabudi.ac.id
Medan - Indonesia

versitas : Universitas Pembangunan Panca Budi
 ultas : SAINS & TEKNOLOGI
 sen Pembimbing I : HERDIANTO, S.kom, MT
 sen Pembimbing II : ZULHAM SITOKUS, S.kom, M.kom
 na Mahasiswa : ANDI HAKIM NASUTION
 usan/Program Studi : Sistem Komputer
 mor Pokok Mahasiswa : 1514370779
 rjang Pendidikan : STRATA 1
 ul Tugas Akhir/Skripsi : PERANCANGAN APLIKASI PEMESANAN TIKET WISATA
 ONLINE BERBASIS ANDROID MENGGUNAKAN FIREBASE

TANGGAL	PEMBAHASAN MATERI	PARAF	KETERANGAN
5/ April 2019	Des. Rupa fungsional		
8/ 19	Revisi tulisan / isi		
9/ 19	konsep di proses Analisis		
7/9 19	Analisis di proses aplikasi di laptop		
2/10 19	kegiatan kelas/ program		
3/10 19	Des. Skema hasil		
4/11 - 19	* Aca sedang meja belajar		
12 19	Des. awal - awal		

Medan, 10 April 2019

Diketahui/Disetujui oleh :

Dekan



Sri Shindi Indira, S.T.,M.Sc.



UNIVERSITAS PEMBANGUNAN PANCA BUDI FAKULTAS SAINS & TEKNOLOGI

Jl. Jend. Gatot Subroto Km 4,5 Medan Fax. 061-8458077 PO.BOX : 1099 MEDAN

PROGRAM STUDI TEKNIK ELEKTRO	(TERAKREDITASI)
PROGRAM STUDI ARSITEKTUR	(TERAKREDITASI)
PROGRAM STUDI SISTEM KOMPUTER	(TERAKREDITASI)
PROGRAM STUDI TEKNIK KOMPUTER	(TERAKREDITASI)
PROGRAM STUDI AGROTEKNOLOGI	(TERAKREDITASI)
PROGRAM STUDI PETERNAKAN	(TERAKREDITASI)

PERMOHONAN JUDUL TESIS / SKRIPSI / TUGAS AKHIR*

Saya yang bertanda tangan di bawah ini :

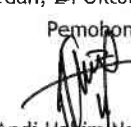
Nama Lengkap : Andi Hakim Nasution
 Tempat/Tgl. Lahir : Medan / 06 Januari 1995
 Nomor Pokok Mahasiswa : 1514370779
 Program Studi : Sistem Komputer
 Konsentrasi : Keamanan Jaringan Komputer
 Jumlah Kredit yang telah dicapai : 141 SKS, IPK 3.61
 Nomor Hp : 085370725811
 Dengan ini mengajukan judul sesuai bidang ilmu sebagai berikut :

No.	Judul
1.	Perancangan Aplikasi Pemesanan Tiket Wisata Online Berbasis Android Menggunakan Firebase

Catatan : Diisi Oleh Dosen Jika Ada Perubahan Judul

*Coret Yang Tidak Perlu


 (Ir. Bhakti Alamsvyah, M.T., Ph.D.)


Medan, ~~21~~ 24 Oktober 2019
 Pemohon,

 (Andi Hakim Nasution)


Tanggal :
 Disahkan oleh
 Dekan

 (Sri Sindhya Indira, S.T., M.Sc.)

Tanggal : 20/10/2019
 Disetujui oleh :
 Dosen Pembimbing I :

 (Herdianto, S.Kom., MT)

Tanggal : 25 OKTOBER 2019
 Disetujui oleh:
 Ka. Prodi Sistem Komputer

 (Eko Hariyanto, S.Kom., M.Kom.)

Tanggal : 25 OKTOBER 2019
 Disetujui oleh:
 a.a- Dosen Pembimbing II:

 (Zulham Sitorus, S.Kom., M.Kom.)

No. Dokumen: FM-UPBM-18-02	Revisi: 0	Tgl. Eff: 22 Oktober 2018
----------------------------	-----------	---------------------------

ABSTRAK

ANDI HAKIM NASUTION

**Perancangan Aplikasi Pemesanan Tiket Wisata *Online* Berbasis Android
Menggunakan *Firebase*
2019**

Memasuki musim liburan maupun akhir pekan biasanya setiap orang akan mengisinya dengan pergi berlibur ke tempat-tempat yang menyenangkan, bisa pergi ke tempat wisata baik yang berada di dalam kota, ataupun diluar kota. Sebelum pergi berlibur, terlebih dahulu harus merencanakan tempat yang akan dikunjungi nantinya. Setiap orang akan mencari-cari informasi mengenai tempat wisata yang cocok untuk dirinya atau keluarga dari berbagai sumber. Salah satu sumber yang sangat mudah digunakan saat ini ialah internet, masyarakat dapat dengan mudah mencari tujuan wisata dengan bantuan mesin pencari. Setelah mendapatkan tujuan wisata selanjutnya masyarakat akan pergi sendiri menggunakan kendaraan pribadi, ataupun mencari jasa penyedia layanan pergi ke tempat tersebut. Dengan pemanfaatan teknologi informasi, terdapat berbagai kemudahan di dalam kehidupan masyarakat, salah satunya ialah dalam pemesanan tiket wisata tidak diperlukannya lagi antri berlama-lama di loket pemesanan tiket, atau pergi jauh ke tempat penyedia layanan wisata. Dengan memanfaatkan perangkat android yang tersedia saat ini, pengguna bisa langsung melihat informasi wisata yang akan dituju, dan langsung membeli tiket wisata, sehingga terjadinya efisiensi waktu dan tenaga sebelum pergi berlibur ke tempat yang diinginkan. Aplikasi pemesanan tiket wisata yang terintegrasi dengan *database*, memudahkan pengguna dalam melihat informasi yang *realtime*, serta fasilitas apa saja yang akan didapatkan ketika membeli tiket tersebut.

Kata Kunci: Android, Tiket Wisata *Online*, *database*

DAFTAR ISI

	Halaman
KATA PENGANTAR	i
DAFTAR ISI	ii
DAFTAR GAMBAR	v
DAFTAR TABEL	vii
DAFTAR LAMPIRAN	viii
BAB I PENDAHULUAN	1
1.1 Latar Belakang Masalah	1
1.2 Rumusan Masalah	2
1.3 Batasan Masalah	3
1.4 Tujuan Penelitian	3
1.5 Manfaat Penelitian	4
BAB II LANDASAN TEORI	5
2.1 Sistem Operasi	5
2.2 Android	5
2.3 Android SDK (<i>Software Development Kit</i>)	6
2.4 Android Studio	7
2.4.1 <i>Activity</i>	9
2.4.2 <i>Intens</i>	12
2.4.3 <i>Service</i>	12
2.4.4 <i>Resource</i>	12
2.4.5 <i>Gradle</i>	13
2.4.6 <i>Debugger</i>	14
2.4.7 <i>Libraries</i>	15
2.4.8 <i>Recycler View</i>	16

2.5	<i>Java</i>	17
2.5.1	<i>JDK (Java Development Kit)</i>	18
2.6	<i>IDE (Integreted Development Environment)</i>	19
2.7	<i>XML (Extensible Markup Language)</i>	19
2.8	<i>JSON (Java Script Object Nonation)</i>	19
2.9	<i>AVD (Android Virtual Device)</i>	20
2.10	<i>Database</i>	21
2.11	<i>Firebase</i>	22
2.11.1	<i>Firebase Realtime Database</i>	22
2.11.2	<i>Firebase Storage</i>	23
2.11.3	<i>Firebase Authentication</i>	24
BAB III METODE PENELITIAN		26
3.1	Tahapan Penelitian	26
3.2	Metode Pengumpulan Data	29
3.3	Analisis Sistem Berjalan	30
3.3.1	Kebutuhan Sistem	32
3.4	Rancangan Penelitian	33
3.4.1	Analisis Sistem Usulan	33
3.4.2	Rancangan Secara Global	34
3.4.3	Struktur Data	40
3.4.4	Rancangan Antarmuka	46
BAB IV HASIL DAN PEMBAHASAN		53
4.1	Kebutuhan Spesifikasi <i>Minimum Hardware</i> dan <i>Software</i>	53
4.2	Pengujian Aplikasi dan Pembahasan	55
4.2.1	Hasil Tampilan Sistem	55

4.2.2 Pengujian Aplikasi	69
BAB V PENUTUP	78
5.1 Kesimpulan	78
5.2 Saran	79
DAFTAR PUSTAKA	
BIOGRAFI PENULIS	
LAMPIRAN-LAMPIRAN	

BAB I

PENDAHULUAN

1.1 Latar Belakang Masalah

Perkembangan teknologi informasi yang begitu memberikan banyak pengaruh dan perubahan dalam kehidupan manusia, mulai dari cara berfikir sampai perilaku manusia saat ini. Melihat kondisi yang ada pada saat ini dilingkungan masyarakat ketika ingin pergi liburan ke salah satu atau beberapa tempat wisata memiliki kesulitan dalam membeli tiket wisata, menghabiskan banyak waktu untuk pergi membeli tiket wisata ke suatu tempat, membuang tenaga yang tidak sedikit, dan biaya yang dikeluarkan untuk sekedar membeli tiket untuk berlibur.

Untuk mengatasi permasalahan tersebut telah dilakukan penelitian oleh (Hidayah N, et.al, 2010), pada jurnal “Sistem Informasi Pemesanan Tiket Pesawat Berbasis Web pada Bana Tour” yang menyatakan bahwa dalam proses pemesanan tiket pesawat masih secara manual melalui telepon yang dapat menimbulkan resiko terjadinya kesalahan terutama pada pencatatan nama dan nomor identitas pelanggan, dimana pada proses pemesanan tiket pesawat seorang *ticketing* akan mencatat nama pelanggan yang melakukan pemesanan tiket pesawat sesuai dengan data yang tertera pada identitas diri pelanggan seperti KTP, bagian *ticketing* ini menuliskan data pelanggan kepada sebuah buku yang membuat data-data jika sudah banyak akan menjadi menumpuk dan sulit dicari.

Dengan terus berkembangnya teknologi informasi, web menjadi alat bantu yang tidak hanya mampu menyediakan informasi, namun juga mampu untuk mengolah informasi. Proses pengolahan informasi dengan memanfaatkan teknologi web menyebabkan terjadinya media informasi yang dinamis. Sehingga untuk melihat jadwal tiket pesawat, tujuan, dan harga tiket dibuatlah sistem informasi pemesanan tiket pesawat berbasis web pada Bana Tour (PT. Wali Angkasamitra Utama), yang dapat mempermudah pelanggan dalam membeli tiket. (Hidayah N, et.al, 2010)

Melihat penyelesaian masalah pemesanan tiket diatas, maka dapat dilihat bahwa untuk melakukan pemesanan tiket harus menggunakan web browser terlebih dahulu dengan mengetikkan alamat situs pemesanan tiket, dan oleh karena itu peneliti mengembangkan lagi aplikasi untuk pemesanan tiket yang lebih efisien tanpa menggunakannya web browser, sehingga pemesanan tiket bisa menjadi lebih mudah. Dilihat dari latar belakang yang telah diuraikan diatas maka penulis mengangkat judul Skripsi “**Perancangan Aplikasi Pemesanan Tiket Wisata *Online* Berbasis Android Menggunakan *Firebase***”.

1.2 Perumusan Masalah

Dalam kesempatan ini penulis akan memberikan rumusan masalah dalam pembuatan tiket wisata sebagai berikut.

1. Bagaimana merancang aplikasi untuk pemesanan tiket wisata online pada *Platform* Android.

2. Bagaimana aplikasi pemesanan tiket wisata memudahkan penggunanya dengan mengedepankan *user interface* dan *user experience*, dan dapat melihat pembelian informasi tiket dengan mudah.

1.3 Batasan Masalah

1. Aplikasi pemesanan tiket wisata ini hanya dapat dijalankan mulai dari Android 5.0 (Lollipop) keatas.
2. Menggunakan Android Studio untuk membangun sistem dan memprogram aplikasi android, juga menggunakan Adobe XD untuk merancang *User Interfaces* aplikasi Android yang dibuat.
3. Informasi hanya dapat di perbaharui oleh *administrator* atau pengembang aplikasi di *database* sehingga konsumen bisa mendapatkan informasi tiket wisata terbaru.

1.4 Tujuan Penelitian

1. Untuk merancang aplikasi *mobile* Pemesanan Tiket Wisata menggunakan Android Studio.
2. Untuk membangun aplikasi yang dapat dilihat secara *update* ke konsumen.
3. Untuk membangun aplikasi yang dapat diakses 24 jam oleh konsumen.

1.5 Manfaat Penelitian

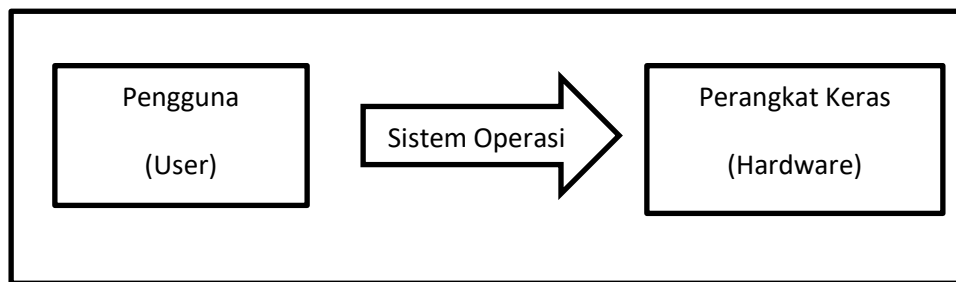
1. Manfaat bagi peneliti ialah menerapkan ilmu yang telah didapatkan selama masa perkuliahan dan sebagai pengembangan diri peneliti dalam pemrograman android.
2. Manfaat bagi Universitas, aplikasi ini dapat diterapkan dilingkungan kampus dan sebagai pembelajaran bagi mahasiswa yang akan datang, serta meningkatkan kualitas perguruan tinggi.
3. Manfaat bagi masyarakat, dalam pemesanan tiket untuk berwisata menjadi lebih mudah dan efisien, hanya dengan menggunakan perangkat android yang dimiliki

BAB II

LANDASAN TEORI

2.1 Sistem Operasi

Sistem operasi adalah salah satu bagian penting dalam sistem komputer. Seluruh perangkat keras komputer dikelola oleh sistem operasi, seperti processor, memori, media penyimpanan, perangkat I/O, dll. Sehingga sistem operasi menjadi penghubung perangkat keras dengan perangkat lunak. (Umar, 2015)



Gambar 2.1 Alur Kerja Sistem Operasi

2.2 Android

Android merupakan sistem operasi yang dirancang khusus untuk perangkat mobile, sistem operasi ini dibuat secara *open source* yang dapat dikembangkan oleh siapapun. Dan hingga sekarang secara resmi Google mengembangkan sistem operasi android ini, setelah resmi dibeli dari Android Inc, dan untuk pengembangannya dibentuklah *Open Handset Alliance*. (Sunaryo, et.al, 2013)

2.3 Android SDK (*Software Development Kit*)

Salah satu tools pendukung agar dapat menjalankan Android SDK (*Software Development Kit*) ialah API (*Application Programming Android*). Untuk dapat membuat aplikasi android diperlukannya Bahasa pemrograman Java, dan Kotlin. Kotlin sendiri Bahasa pemrograman yang resmi diumumkan oleh google pada tahun 2011 sebagai alternatif Bahasa pemrograman, dan kotlin dirancang untuk mempermudah programmer dalam menulis program. (Nicky, 2016)

Didalam Android SDK terdapat *libraries*, perangkat *virtual*, *debugger*. Android mencakup perangkat pengembang, kode sampel dengan sumber proyek android, *emulator* dan perpustakaan yang perlukan untuk membangun aplikasi Android. Android yang berjalan di bahasa pemrograman *java* dan berjalan di *Dalvik*, serta mesin *virtual* yang dirancang khusus untuk penggunaan embedded yang berjalan diatas *kernel* Linux. (Maiyana, 2018)

Dalam pengembangan Android, terdapat berbagai fitur yang harus dimiliki agar proses pembuatan aplikasi dapat berjalan dengan baik:

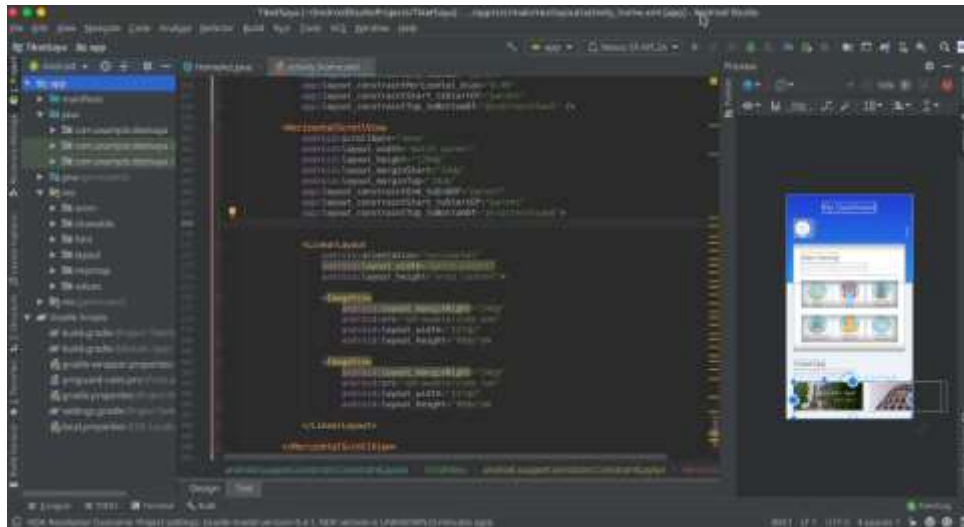
- a. Dukungan Grafis 2D dan 3D melalui *libraries* sehingga dapat berjalan optimal dan berdasarkan spesifikasi OpenGL ES 1.0
- b. Perangkat *Mobile* yang dioptimalkan oleh *Dalvik Virtual Machine* (DVM)
- c. Perubahan komponen dan reusable yang dapat mendukung proses pengembangannya
- d. Menggunakan database SQLite sebagai media penyimpanannya

- e. Didukung oleh Kamera, GPS, dan Kompas untuk mempermudah proses jalannya aplikasi
- f. Browser yang terintegrasi berdasarkan *Engine Open Source Webkit*.

Lingkungan pengembangan Android yang lengkap dan kaya termasuk perangkat *emulator*, *tools* untuk *debugging*, profil dan kinerja *memori* serta *plugins* untuk *IDE Android Studio*. (Nicky, 2016)

2.4 Android Studio

Untuk melakukan pengembangan terhadap aplikasi android, diperlukannya IDE (*Integrated Development Environment*), IDE yang sangat populer dalam pengembangan aplikasi android saat ini ialah Android Studio, yang langsung dikembangkan oleh Google sendiri menggunakan Bahasa pemrograman Java dan Kotlin. Pertama kali diluncurkan pada ajang Google IO pada 16 Mei 2013, dan secara resmi menggantikan eclipse dalam pengembangan aplikasi android.



Gambar 2.2 Halaman Utama Android Studio

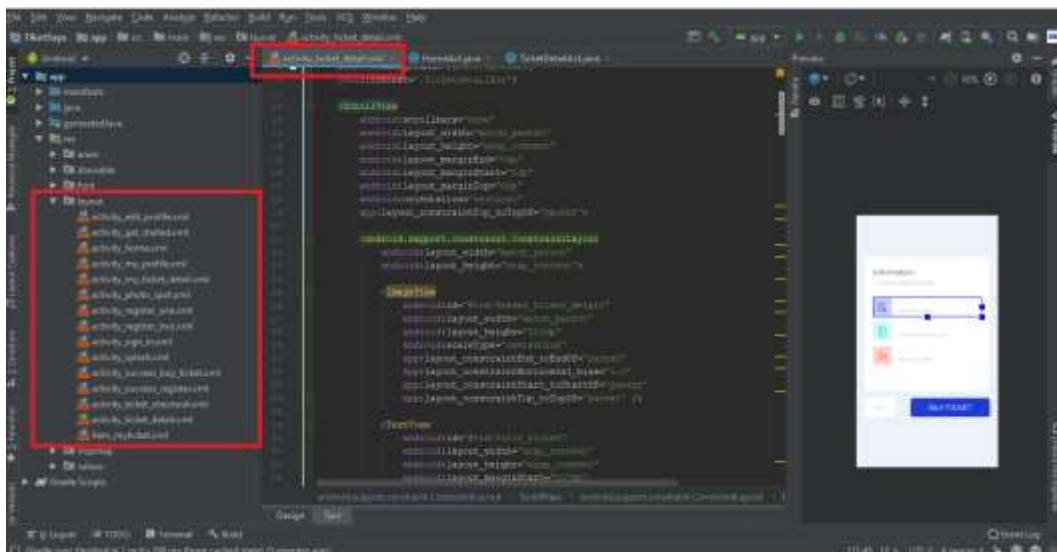
Android Studio dikembangkan sangat menyerupai dengan *clipse* berdasarkan *IntelliJ IDEA* lengkap dengan *ADT (Android Development Tools)* agar pengembang lebih mudah dalam mengoperasikannya:

- a. Menggunakan tampilan *GUI (Graphic User Interface)* agar dapat lebih mudah dijalankan.
- b. Menggunakan *Pembenahan* menanggapi *bug* dan *Refactory* agar lebih mudah dan cepat.
- c. Untuk keamanannya didukung oleh *App-signing* dan *proguard*.
- d. Menggunakan *gradle build* dalam proses pembuatan *project*
- e. Pada setiap aplikasi yang dikembangkan sistem *cloud* didukung oleh *Google Cloud Platform* di dalam *database firebase*.

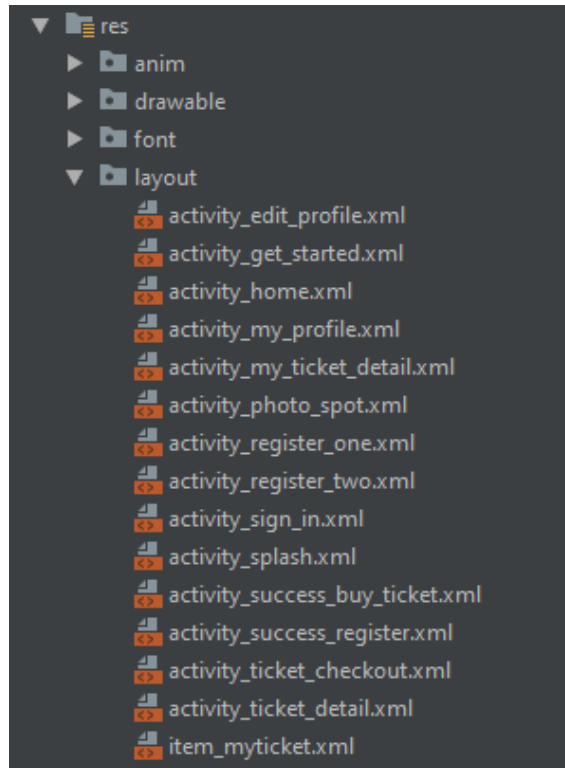
- f. Dalam mengukur kecepatan, kompetebelitas, dan memantau aplikasi digunakan *tools* yang disebut *Lint*.

2.4.1 Activity

Activity berguna untuk memberikan tampilan kepada pengguna, *activity* berisi informasi yang dapat dilihat oleh pengguna, serta pengguna dapat berinteraksi didalamnya. Terdapat beberapa fungsi di dalam sebuah *activity* seperti *button*, *label*, *layout* dan lainnya. Setiap interaksi yang dilakukan pengguna di dalam *activity*, maka prosesnya akan dilanjutkan dalam pemrograman java, yang nantinya akan diteruskan menuju *activity* lain atau *activity* tersebut berakhir. (Huda, 2011)



Gambar 2.3 Activity



Gambar 2.4 *Activity.xml*

a. Siklus *Activity*

Status activity dapat berjalan di dalam dua status, dan pengembang tidak dapat mengontrol semua status tersebut karena telah ditangani sistem seutuhnya. Kita hanya dapat menerima pesan melalui method `onXX()`. Dan setiap statusnya dapat dilihat berikut ini:

Tabel 2.1 Siklus *Activity*

<i>onPause()</i>	Sangat cocok untuk menaruh algoritma menyimpan. Dan aplikasi
------------------	--

	yang dijalankan akan dikembalikan kepada halaman sebelumnya dan biasanya dikarenakan ada <i>activity</i> baru yang dijalankan.
<i>onStop()</i>	Ketika aplikasi berjalan di <i>background</i> dengan waktu yang cukup lama
<i>onDestroy()</i>	Saat aplikasi benar-benar berhenti maka perintah ini dijalankan
<i>onCreate(Bundle)</i>	Menggunakan deklarasi variable atau dengan membuat <i>user interface</i> , dan biasa pada saat aplikasi pertama kali dijalankan
<i>onStart()</i>	Activity yang ditampilkan ke pengguna dengan mengindikasinya terlebih dahulu
<i>onRestart()</i>	Setelah status berhenti, <i>activity</i> kembali menampilkan <i>user interface</i> kepada pengguna.

<i>onResume()</i>	Sangat cocok meletakkan animasi disini karena aplikasi mulai berinteraksi dengan pengguna
<i>onRestoreInstanceState (Bundle)</i>	Saat activity kembali menginisialisasi dari status activity sebelumnya.yang kemudian disimpan.
<i>onSaveInstanceState(Bundle)</i>	Setiap status disimpan dengan mengijinkan method ini.

2.4.2 *Intens*

Intens merupakan sebuah *action* yang digambarkan secara terperinci seperti bagaimana mengambil sebuah gambar pada *activity* ataupun *database*.

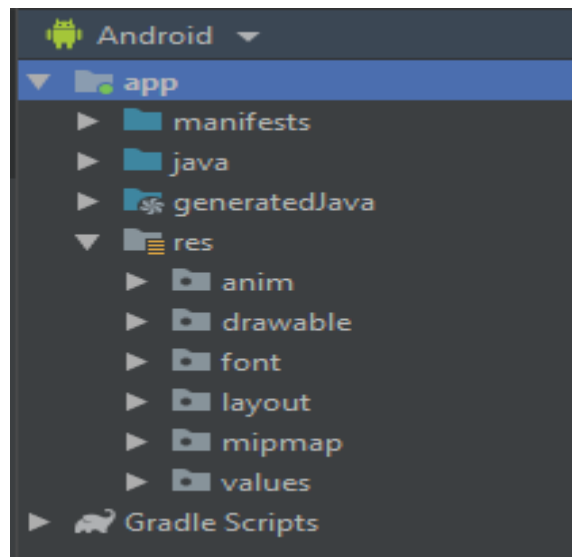
2.4.3 *Service*

Service tidak memiliki *user interface*, namun berjalan di belakang layar. Misalnya pemutar musik, sebuah *activity* digunakan untuk memilih lagu kemudian dijalankan. Agar pemutar musik bisa berjalan dibelakang aplikasi lain maka harus menggunakan *service*.

2.4.4 *Resource*

Untuk menyimpan file-file yang tidak berbentuk kode seperti gambar, icon, audio, video maka diperlukannya *resource*. Format gambar JPG dan PNG disimpan ke

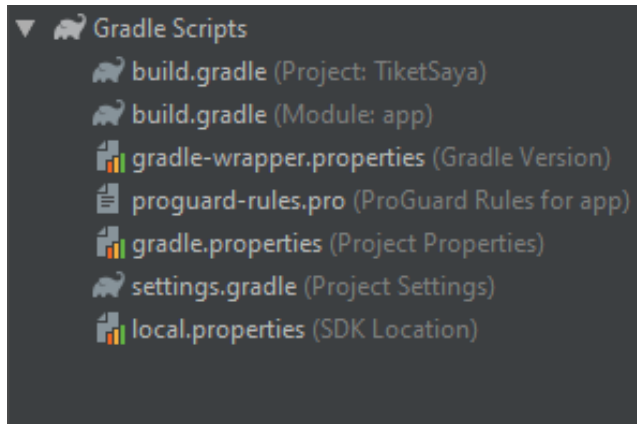
dalam *folder res/drawable*, dan icon aplikasi di simpan dalam *folder res/drawable-ldpi* dan video ataupun audio disimpan dalam *folder res/raw*. Dan untuk membuat sebuah *user interface* maka dibuatlah file xml yang berada di folder *res/layout*. (Huda, 2011)



Gambar 2.5 Resource

2.4.5 Gradle

Gradle merupakan salah satu fitur yang disediakan oleh android studio, dan gradle merupakan sebuah fitur *build automation*. Seorang *programmer* dapat terbantu dengan mudah dengan adanya fitur *gradle* ini, *Gradle* ini berisi *libraries* yang sangat mudah. Dimana dalam *file gradle* ini mempunyai pengaturan versi *compile* yang digunakan. (*Android Developer Fundamentals*, 2016)

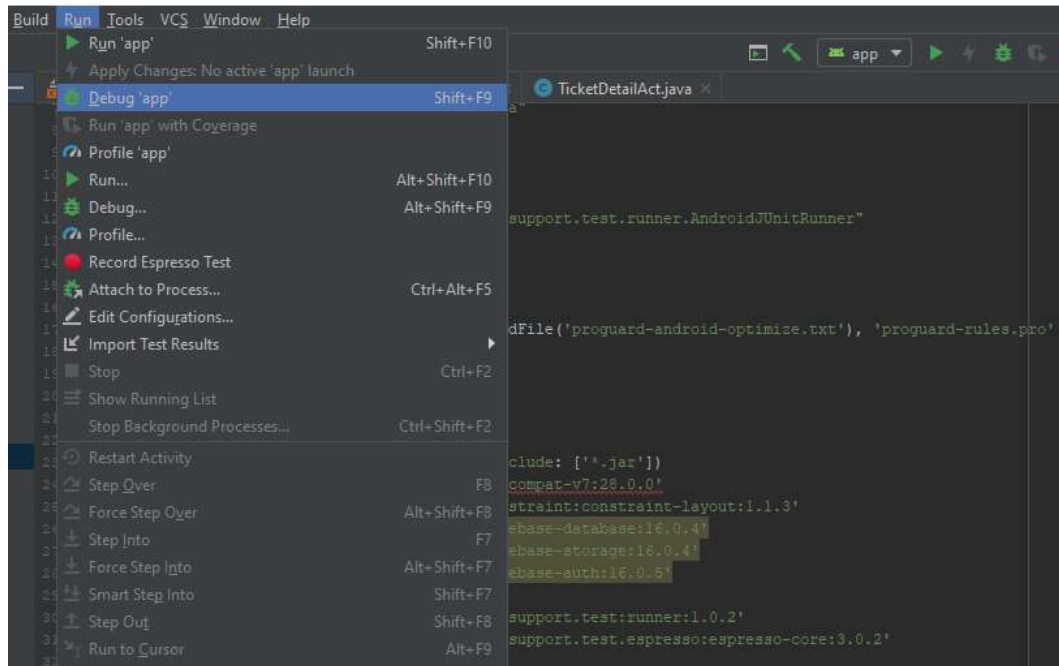


Gambar 2.6 *Gradle Scripts*

2.4.6 *Debugger*

Untuk memperbaiki atau menemukan suatu kesalahan pada kode program diperlukannya *debugger*, yang diperiksa oleh *debugger* ini ialah perilaku aplikasi yang salah, aplikasi yang menggunakan memori terlalu banyak sehingga menyebabkan terjadinya *crash* terhadap aplikasi. Dan berikut berbagai alasan terjadi sebuah *bug* dalam pembuatan aplikasi android.

- a. Proses desain dan penerapan aplikasi yang salah
- b. Kerangka kerja android yang memiliki batasan
- c. *Bug* pada perangkat
- d. Tidak menggunakan tata cara sebagaimana seharusnya aplikasi berjalan dan persyaratan yang dibuat agar aplikasi berjalan optimal.



Gambar 2.7 *Debugger*

2.4.7 *Libraries*

Libraries pada android studio berfungsi untuk sebagai alat bantu developer dalam pengembangan aplikasi, yang menyertakan sebuah fitur yang dapat di unduh melalui *gradle* dan dengan mudah menjalankan fungsinya. *Libraries* merupakan *tools* dari android SDK yang membuat fungsinya sangat dibutuhkan dalam pengembangan aplikasi android.

```

1  applicationId "com.example.ticketapp"
2  minSdkVersion 21
3  targetSdkVersion 28
4  versionCode 1
5  versionName "1.0"
6  testInstrumentationRunner "android.support.test.runner.AndroidJUnitRunner"
7
8  }
9
10 buildTypes {
11     release {
12         minifyEnabled false
13         proguardFiles getDefaultProguardFile('proguard-android-optimize.txt'), 'proguard-rules.pro'
14     }
15 }
16
17 dependencies {
18     implementation fileTree(dir: 'libs', include: ['*.jar'])
19     implementation 'com.android.support:appcompat-v7:28.0.0'
20     implementation 'com.android.support.constraint:constraint-layout:1.1.3'
21     implementation 'com.google.firebase:firebase-database:16.0.4'
22     implementation 'com.google.firebase:firebase-storage:16.0.4'
23     implementation 'com.google.firebase:firebase-auth:16.0.5'
24     testImplementation 'junit:junit:4.12'
25     androidTestImplementation 'com.android.support.test:runner:1.0.2'
26     androidTestImplementation 'com.android.support.test.espresso:espresso-core:3.0.2'
27
28     implementation 'com.github.flores17:shapeofview:1.2.2'
29     implementation 'com.squareup.picasso:picasso:2.5.2'
30     implementation 'com.android.support:recyclerview:28.0.0'
31 }

```

Gambar 2.8 Libraries

2.4.8 *Recycler View*

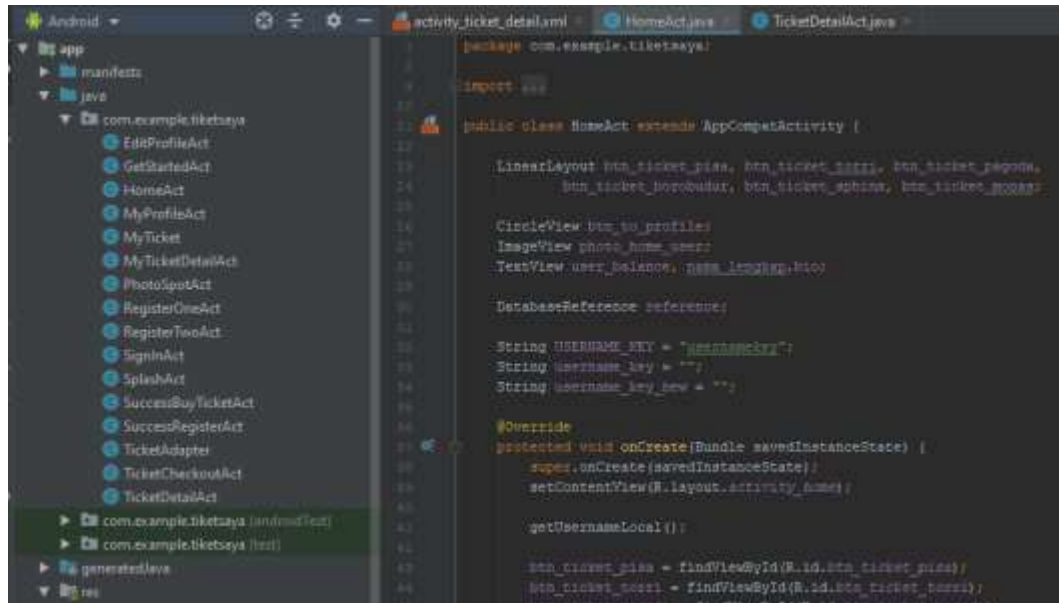
Recycler View berfungsi untuk membuat daftar *item* pada tingkat yang lebih lanjut, dan bisa memiliki kumpulan banyak data didalamnya. Kumpulan data yang digunakan dapat berubah-ubah sesuai dengan interaksi pengguna dalam menggunakan aplikasi. *Recycler View* ini dapat meminimalisir tampilan dan dapat menampilkan banyak data secara bergantian, sehingga sangat efektif dalam membuat *activity*.

Dalam menampilkan data pada *Recycler View* diperlukannya beberapa bagian sebagai berikut:

1. Pengelola *Layout*. Merancang tampilan untuk penyusunan komponen yang nantinya akan ditampilkan, layout yang digunakan beragam bisa *linear layout*, *constraint layout*, tergantung kebutuhan.
2. Data. Yang nantinya akan ditampilkan di *recycler view* sesuai dengan *layout* yang telah di desain, data nya bisa dari mana saja, bisa secara lokal ataupun diambil data *database*.
3. *View Holder*. Berisi tampilan menampilkan informasi untuk satu item dari *layout item*, dan berfungsi memperluas kelas *View Holder*.
4. *Adapter*. Berfungsi untuk menyiapkan data dan menampilkan data ke dalam *View Holder*, sehingga jika terjadi perubahan data maka adapter akan memperbaharui tampilan *item*, dan *adapter* bertugas untuk menghubungkan data dengan *Recycler View*. (*Android Developer Fundamentals*, 2016)

2.5 Java

Java merupakan Bahasa pemrograman yang dapat berjalan di berbagai *platform*, baik komputer ataupun perangkat *mobile*. Aplikasi yang dibuat menggunakan Bahasa pemrograman ini biasa di kompilasi ke dalam *bytecode* sehingga dapat dijalankan di berbagai JVM (*Java Virtual Machine*). Bahasa pemrograman java dibuat bersifat umum/non-spesifik, dan didesain untuk memanfaatkan seminimal mungkin ketergantungan penerapannya seminimal mungkin. (Poerwanta, 2013)



Gambar 2.9 Java

2.5.1 JDK (*Java Development Kit*)

Java adalah Bahasa pemrograman yang pertama kali diperkenalkan pada tahun 1990 oleh *Sun Microsystems*, dan menurut sun java ialah nama untuk sekumpulan teknologi untuk membuat dan menjalankan perangkat lunak pada lingkungan jaringan ataupun komputer yang berdiri sendiri. Kompiler dan interpreter dibutuhkan dalam pemrograman java yang berbentuk *Java Development Kit (JDK)*. Sebelum melakukan instalasi android SDK, diperlukannya penginstalan JDK terlebih dahulu, agar dapat mengompile aplikasi android yang dirancang. (Nicky, 2016).

2.6 IDE (*Integreted Development Environment*)

Integreted Development Environment adalah aplikasi pengembang perangkat lunak fungsi-fungsi terintegrasi yang dibutuhkan untuk mebangun sebuah perangkat lunak seperti *code editor*, *debugger*, *compiler*, dan sebagainya. (Maiyana, 2018)

2.7 XML (*Extensible Markup Language*)

XML merupakan sebuah teknologi lintas *platform*, dan *tools* untuk melakukan transmisi informasi. XML bukanlah pustakan ataupun program, tetapi merupakan teknologi dan sebuah standar yang memiliki berbagai aturan tertentu. Dan jika diartikan secara sederhana XML adalah sebuah file teks yang berisi berbagai tag yang di definisikan sendiri oleh pembuat dokumen XML tersebut. XML bukanlah hal yang baru, dan bukan pula sebagai pengganti HTML. XML ditujukan untuk focus pada data, dan HTML ditujukan untuk cara menampilkan data. (Deny S. W & Fitri N. H., 2018).

2.8 JSON (*Java Script Object Nonation*)

JSON (*Java Script Ohject Nonation*) adalah format pertukuran data yang mudah dibaca oleh manusia, dan ringan, serta mudah diterjemahkan dan dibuat (*generate*) oleh komputer. Format ini dibuat berdasarkan bagian dari Bahasa Pemrograman JavaScript, Standar ECMA-262 Edisi ke-3 - Desember 1999. JSON merupakan format teks yang tidak bergantung pada bahasa pemrograman apapun karena menggunakan gaya bahasa yang umum digunakan oleh programmer keluarga C termasuk C, C++, C#, Java, JavaScript, Perl, Python dll. Oleh karena sifat-sifat

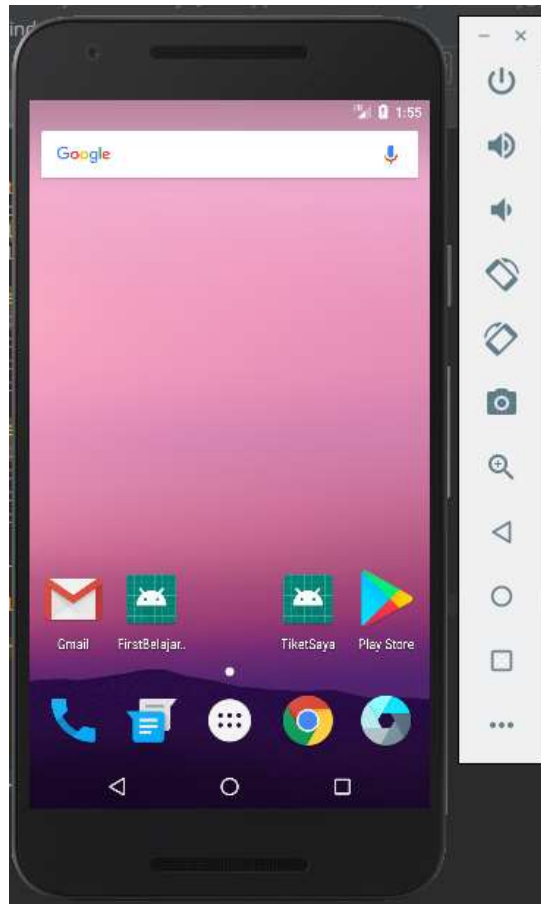
tersebut, menjadikan JSON ideal sebagai bahasa pertukaran data [JSON]. (Juansyah, 2015)

Struktur JSON terdiri dari dua struktur, yaitu:

- b. Kumpulan pasangan nama/nilai. Pada beberapa bahasa, hal ini dinyatakan sebagai objek (*object*), rekaman (*record*), struktur (*struct*), kamus (*dictionary*), tabel hash (*hash table*), daftar berkunci (*keyed list*), atau associative array.
- c. Daftar nilai terurutkan (*an ordered list of values*). Pada kebanyakan bahasa, hal ini dinyatakan sebagai larik (*array*), vektor (*vector*), daftar (*list*), atau urutan (*sequence*). (Juansyah, 2015).

2.9 *Android Virtual Device (AVD)*

Android Virtual Device merupakan emulator yang dijalankan ketika pengembang aplikasi android ingin menjalankan aplikasi yang dibuat. AVD ini selanjutnya digunakan sebagai tempat untuk melakukan uji coba dan menjalankan aplikasi Android tanpa harus menggunakan perangkat Android yang sebenarnya. Sebelum menjalankan AVD pengembang memilih perangkat apa yang cocok untuk diuji coba, misalnya emulator akan menjalankan perangkat nexus 7 dengan resolusi layar juga sesuai dengan perangkat aslinya, dan AVD bisa dibuat sebanyak yang dibutuhkan. (Maiyana, 2018)



Gambar 2.10 *Android Virtual Device (AVD)*

2.10 *Database*

Database atau biasa disebut Basis Data merupakan sekumpulan data yang saling berhubungan. Dan data biasa berbentuk dalam table-tabel yang saling berhubungan satu sama lainnya, dengan menggunakan *field*/kolom pada tiap table yang ada. (Prayitno & Safitri, 2015).

2.11 *Firestore*

Firestore adalah API yang disediakan oleh Google untuk penyimpanan dan penyelarasan data ke dalam aplikasi mobile seperti Android, iOS, atau web. Salah satu fasilitas yang disediakan oleh *Firestore* ialah *realtime database* yang menyimpan data ke *database* dan mengambil data darinya sangat cepat tetapi *Firestore* bukan hanya *realtime database*, lebih jauh dari itu. *Firestore* memiliki banyak fitur seperti autentikasi, *database*, *storage*, *hosting*, pemberitahuan dan lain-lainnya. (Payara R & Tanone R, 2018)



Gambar 2.11 *Firestore Database*

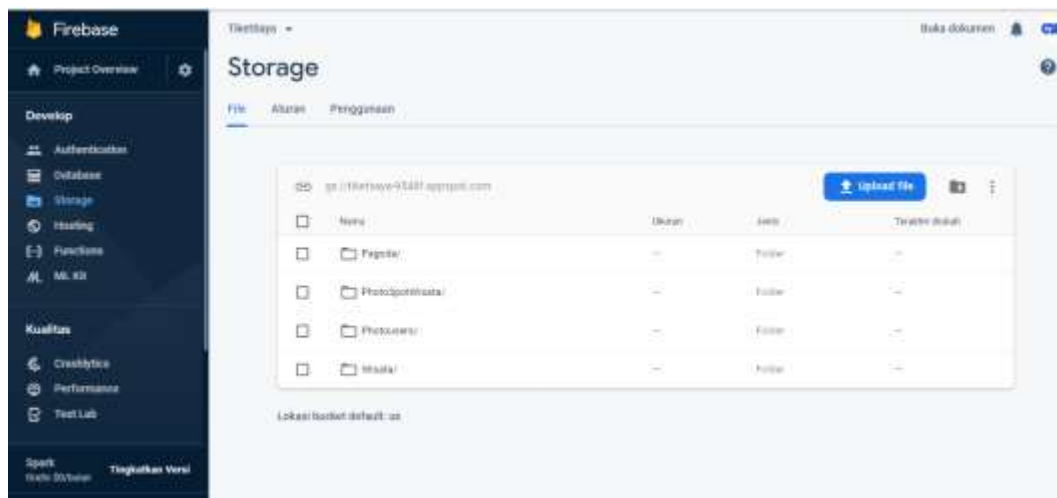
2.11.1 *Firestore Realtime Database*

Firestore Realtime Database merupakan *cloud database*. Data disimpan dalam format JSO (“Penerapan *Firestore Realtime Database* Pada Prototype Aplikasi Pemesanan Makanan Berbasis Android,” 2018) N dan di sinkronkan secara *realtime*

terhadap klien yang terhubung ke *database*. Ketika ingin membuat aplikasi *hybrid* lintas *platform*, seperti IOS dan Android maka semua klien akan berbagi satu *instance* *Realtime Database* dan Secara otomatis melakukan pembaruan data tertentu dari *database*. *Firestore Database* adalah basis data NoSQL berbeda dengan basis data relasional lainnya karena memiliki optimalisasi dan fungsional berbeda. Membuat *database firebase database* dapat *import* file JSON dan di *export* kemudian di *coding* melalui *text editor*, atau bisa langsung melalui halaman konsol *Realtime Database* secara manual. (Ilhami, 2017)

2.11.2 *Firestore Storage*

Firestore Storage digunakan untuk menyimpan *file* berupa gambar, audio, dan video. Dengan adanya *Firestore Storage* mempermudah *developer* android dalam menyimpan file gambar, audio, dan video ke *database* untuk aplikasi yang dibuat.



Gambar 2.12 *Firestore Storage*

Dan berikut beberapa kelebihan utama dari *Firestore* adalah sebagai berikut:

a. *Strong*

Firestore dapat melakukan unggahan dan unduhan pada semua kualitas jaringan internet. Ia dapat berhenti atau melanjutkan, sehingga menghemat *bandwidth* dan waktu pengguna.

b. *Secure*

Firestore terintegrasi dengan *Firestore Authentication* untuk menyediakan autentikasi yang mudah dan intuitif untuk pengembang. Model keamanan dapat diatur berdasarkan nama file, ukuran, *metadata* dan tipe kontennya.

c. *Scalable*

Firestore didukung oleh *Google Cloud Storage* hingga skala petabyte. Ini yang terpenting, karena ketika aplikasi yang dari *prototype* kemudian berkembang viral dengan jutaan pengguna. (Ilhami, 2017)

2.11.3 *Firestore Authentication*

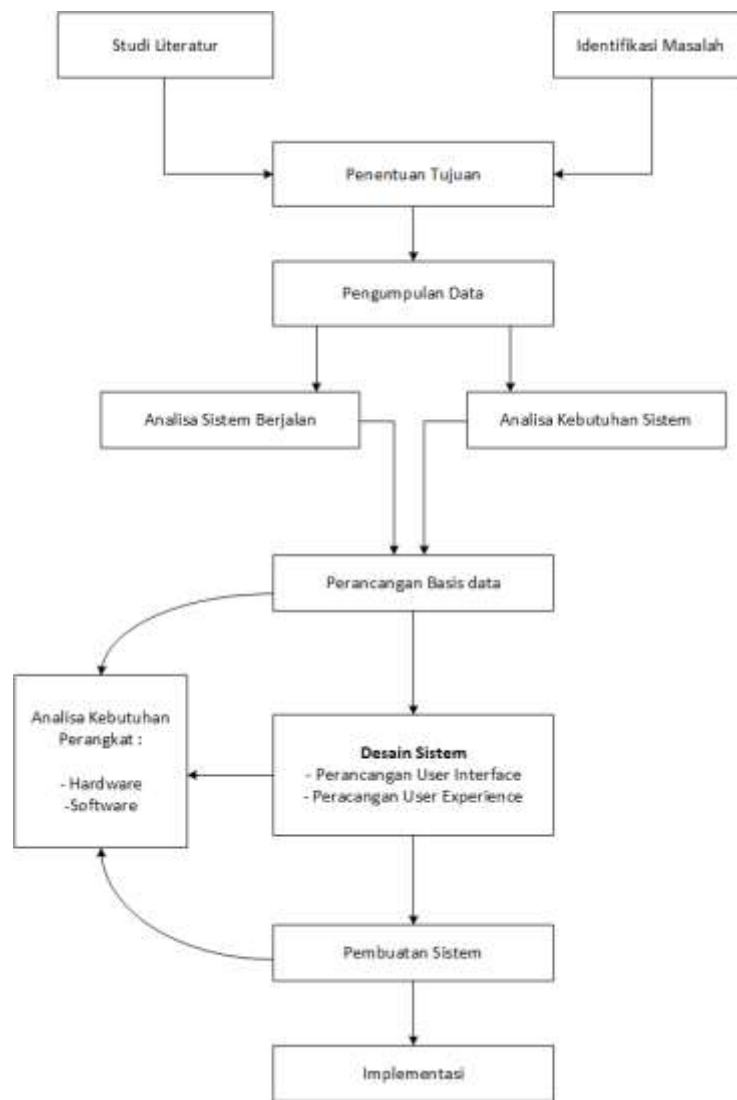
Dibutuhkannya autentikasi untuk mengetahui identitas pengguna yang dibutuhkan untuk sebagian aplikasi. Dengan mengetahui identitas pengguna memungkinkan aplikasi untuk secara aman menyimpan data di *cloud* dan memberikan pengalaman pribadi yang sama di semua perangkat pengguna. *Firestore* menyediakan layanan backend, SDK yang mudah digunakan dan pustaka siap pakai

untuk mengautentikasi pengguna ke aplikasi. Autentikasi yang didukung Firebase saat ini ialah menggunakan sandi, social media populer seperti Google, Twitter, Facebook, dan lainnya. (Ilhami, 2017)

BAB III

METODE PENELITIAN

3.1 Tahapan Penelitian



Gambar 3.1 Tahapan Penelitian

Dengan demikian sedikit penjelasan tentang alur tahapan penelitian *workflow* diatas proses pembuatan program:

a. Studi Literatur

Dimana studi literatur untuk mencari beberapa teori yang dapat digunakan maka dengan teori ini bisa dipahami dan dipraktekan untuk sebagai proses penerapannya.

b. Identifikasi Masalah

Melakukan identifikasi masalah untuk mencari solusi terhadap masalah tersebut, sehingga masalah dapat terselesaikan dengan baik.

c. Penentuan Tujuan

Setelah menemukan masalah yang sedang terjadi maka tujuan utamanya adalah membuat solusi dalam bidang teknologi informasi dengan sistem apa yang cocok untuk permasalahan yang ada, sehingga masalah dapat terselesaikan dengan baik.

d. Pengumpulan Data

Dimana pengumpulan data untuk mengambil beberapa teori-teori yang ada dan mengobservasi permasalahan yang ada yang kemudian untuk diangkat kedalam sistem dan kebutuhan perangkat untuk mencapai suatu proses tujuan yang diinginkan.

e. Analisa Sistem Berjalan

Dimana dengan menganalisa sistem berjalan berarti proses sistem seperti apa nantinya yang akan digunakan, mulai dari proses pembuatan *username* dan

password, kemudian login ke *database* sehingga dapat masuk ke halaman utama aplikasi. Tidak hanya itu user juga bisa melakukan transaksi pada aplikasi dan melakukan edit profile untuk kebutuhan data pengguna.

f. Analisa Kebutuhan Sistem

Dimana dengan menganalisa apa saja yang dibutuhkan untuk dijalankan pada sistem yang akan digunakan, seperti kebutuhan perangkat lunak, *database*, dan kebutuhan design tampilan yang digunakan nantinya, dan juga pengalaman yang baik bagi *user* saat penggunaan sistem.

g. Perancangan Basis Data

Dimana Basis data untuk penyimpanan data pengguna, gambar, *username* dan *password*, sehingga diperlukannya perancangan table-tabel data pengguna, dan kapasitas storage yang dibutuhkan pada sistem.

h. Analisa Kebutuhan Perangkat

Dimana Analisa kebutuhan perangkat ini merupakan kelengkapan dari proses pembuatan program, maka dari kebutuhan perangkat yang digunakan dalam membangun sistem tersebut, seperti perangkat keras dan perangkat lunaknya.

i. Desain Sistem

Dimana Sistem dibuat dan di desain semudah dan seramah mungkin terhadap pengguna, dengan mengutamakan pengalaman pengguna yang baik dan tata letak tombol dan gambar yang baik sehingga membuat pengguna dengan mudah menjalankan aplikasinya.

j. Pembuatan Sistem

Dimana desain sistem yaitu sebagai membuat gambaran dan alur konsep yang ingin diterapkan dari beberapa elemen terpisah dan disatukan maka hasil tersebut tercapai.

k. Implementasi

Dimana implemementasi tersebut diterapkan dari hasil program yang dicapai.

3.2 Metode Pengumpulan Data

Adapun proses yang dilalui penulis dalam pengumpulan data guna membantu dalam penulisan ini, yaitu:

a. Penelitian Kepustakaan

Penulis mencari buku teori penelitian di perpustakaan, tempat penjualan buku-buku gramedia, serta jurnal-jurnal resmi dari internet untuk mencari informasi tentang teori judul skripsi. Lebih tepatnya agar ketika mengangkat suatu teori tidak terkendala pada penelitian, maka peneliti telah melakukan observasi terhadap teori-teori yang digunakan dan penulis menemukan tujuan dari observasi, dengan ini penulis mengambil teori dari berbagai jurnal untuk membantu perancangan sistem yang digunakan.

b. Wawancara

Setelah melaksanakan observasi maka penulis akan mewawancarai beberapa mahasiswa ataupun masyarakat sekitar mengenai apa saja seharusnya yang akan dimuat dalam aplikasi yang nantinya akan dirancang, berupa tampilannya,

tempat wisata apa yang cocok untuk diinputkan ke dalam *database* dan aplikasi, menunya sebaiknya bagaimana, dengan melibatkan beberapa mahasiswa dan masyarakat berarti dapat meningkatkan pengalaman pengguna dalam aplikasi tersebut.

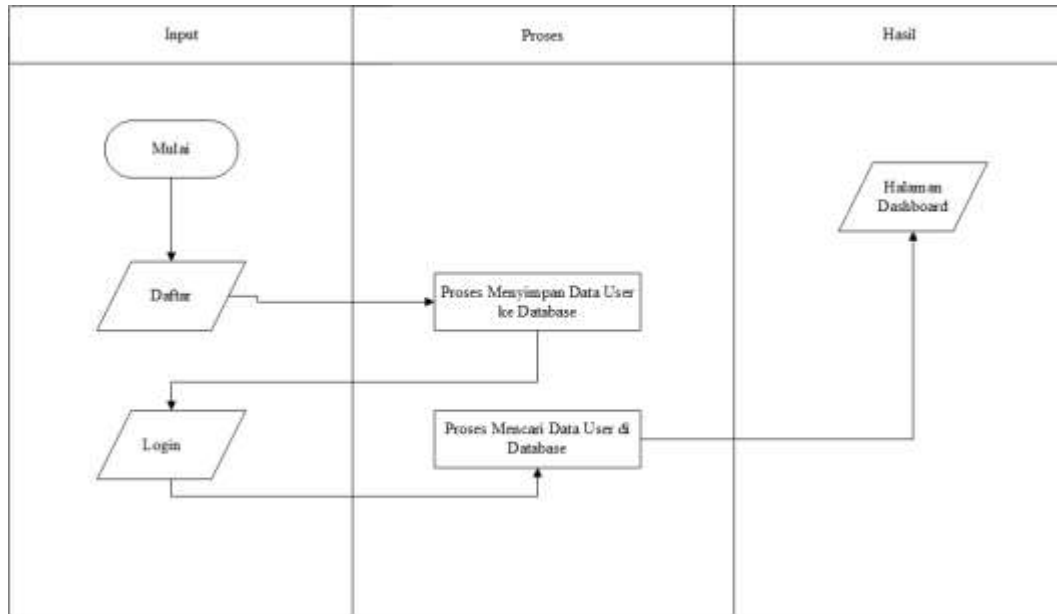
c. Kuisisioner

Penulis membuat kuisisioner untuk beberapa pengguna mengenai pengalaman pengguna dalam menggunakan aplikasi yang telah dibuat berupa *user interface* dan *user experience* aplikasi. Kuisisioner berupa beberapa pertanyaan mengenai aplikasi yang telah dibuat. Oleh karena itu penulis dapat memahami kekurangan pada aplikasi tersebut, sehingga penulis dapat menambahkan kekurangan dari programnya. Kuisisioner pun penting untuk sistem metode pengumpulan data karena kuisisioner ini dapat memberi pemahaman dari kelemahan dan kekurangan program yang di buat penulis tersebut.

3.3 Analisis Sistem Berjalan

Sistem yang dirancang saat ini adalah Aplikasi Pemesanan Tiket Wisata, dimana semua informasi yang berkaitan dengan wisata tersimpan didalam database. Sehingga setiap pengguna dapat mengakses informasi dan pembelian tiket dapat diakses dengan cepat tanpa harus bersusah payah antri di tempat pembelian tiket.

Adapun *Flow of Document* Sistem berjalan pada aplikasi ini adalah sebagai berikut:



Gambar 3.2 *Flow of Document* Sistem Berjalan

Dengan ini sedikit penjelasan pada gambar *Flow of Document* Sistem Berjalan diatas sebagai berikut:

1. Memulai aplikasi pemesanan tiket wisata.
2. Membuat akun user baru untuk dapat menggunakan aplikasi.
3. Maka proses menyimpan data pengguna ke *database*.
4. Login menggunakan akun yang telah dibuat sebelumnya.
5. Proses mencari *username* dan *password* pengguna yang telah didaftarkan di *database*.
6. Berada di Halaman *Dashboard*.

3.3.1 Kebutuhan Sistem

Dalam perancangan Aplikasi Android ini, penulis membutuhkan perangkat untuk membantu sistem berjalan diantaranya yaitu:

1. Sistem Operasi Linux

Dimana Sistem Operasi Linux dibutuhkan untuk sebagai awalan menjalankan *software programming*, dan *text editor*. Penulis memilih sistem operasi ini karena keterbatasan hardware laptop yang dimiliki, dengan menggunakan sistem operasi linux *software programming* menjadi lebih lancar ketika dijalankan.

2. *Software* Android Studio

Software ini dibutuhkan untuk membangun aplikasi android, dimana didalamnya telah terdapat *tools* pendukung yang sangat lengkap sehingga mempermudah menulis kode program bagi penulis.

3. *Software* Adobe XD

Untuk membuat tampilan pengguna, diperlukannya desain tampilan yang baik dan menarik ketika digunakan, sehingga diperlukannya *software* desain seperti Adobe XD.

4. Perangkat Android

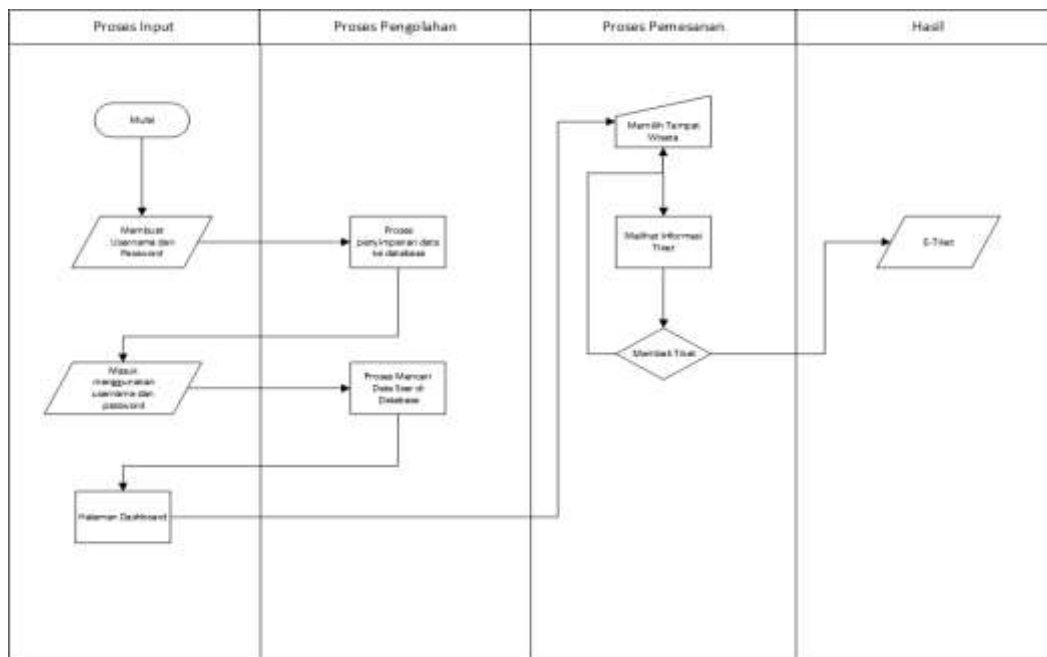
Untuk menguji coba aplikasi yang telah dibuat, diperlukannya perangkat android yang sesungguhnya, sehingga aplikasi dapat di uji kestabilannya dan efisiensinya.

3.4 Rancangan Penelitian

Rancangan penelitian diperlukan agar terciptanya sistem yang diperlukan oleh pasar dan berjalan lebih efisien serta menciptakan pengalaman yang baik terhadap pengguna, dan berikut ini rancangan yang diterapkan pada sistem.

3.4.1 Analisis Sistem Usulan

Dimana sistem ini digunakan untuk membeli tiket melalui perangkat mobile yang digunakan oleh pengguna, dan berikut gambar ini dapat menjelaskan proses pemesanan tiket sebagai berikut:



Gambar 3.3 *Flow of Document* Sistem Usulan

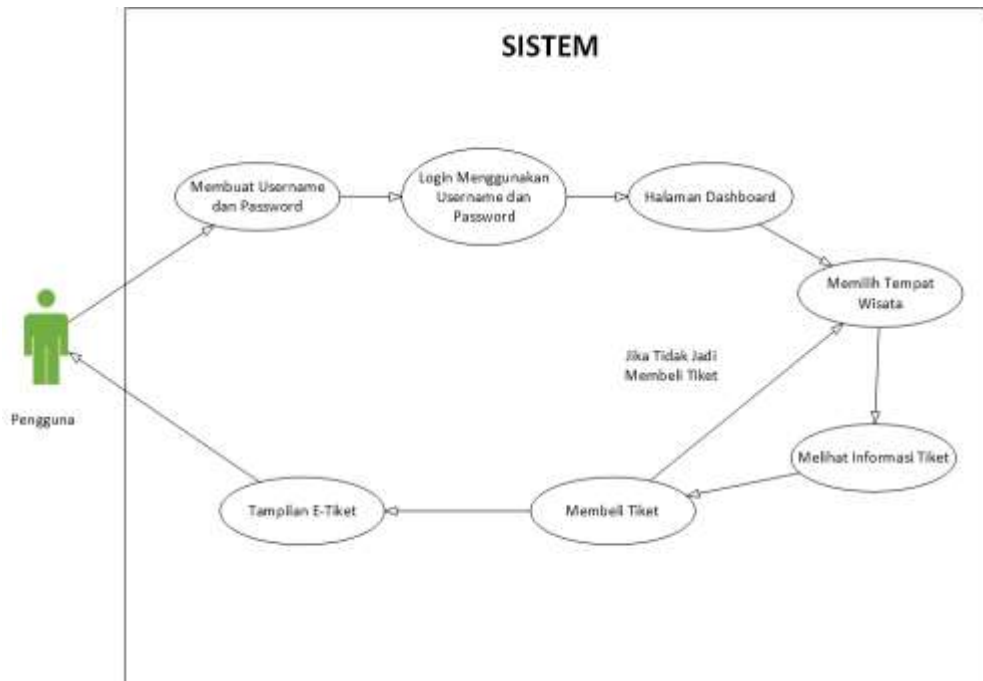
Berikut ini penjelasan analisis *Flow of Document* sistem berjalan dari gambar diatas sebagai berikut:

1. Membuka Aplikasi.
2. Membuat data pengguna baru berupa *username*, *password*, *email*, nama, dan bio.
3. Masuk menggunakan *username* dan *password* yang telah dibuat.
4. Berada dihalaman *dashboard*, yang berisi daftar tiket, foto profile, dan jumlah saldo yang dimiliki.
5. Memilih tempat wisata yang akan dituju.
6. Melihat informasi berupa jadwal keberangkatan, harga tiket, ketentuan apa saja yang berlaku.
7. Membayar tiket yang ingin dibeli, setelah dilakukan pembayaran maka saldo akan terpotong secara otomatis.
8. E-tiket berhasil di beli, dan e-tiket akan muncul dihalam profile pada menu *My Tickets*.

3.4.4 Rancangan Secara Global

a. Use Case Diagram

Use Case merupakan permodelan untuk kelakuan sistem informasi yang akan dibuat. *Use case* mendeskripsikan sebuah interaksi antara satu atau lebih aktor terhadap sistem yang akan dibuat. Secara garis besar, *use case* digunakan untuk mengetahui fungsi apa saja yang digunakan didalam sebuah sistem dan siapa saja yang berhak menggunakan fungsi-fungsi tersebut. (Heriyanto Y, 2018).



Gambar 3.4 Use Case Diagram Sistem Aplikasi Pemesanan Tiket Wisata

Gambar diatas merupakan *Use Case Diagram* Aplikasi Pemesanan Tiket Wisata, dimana tahap awal dari penggunaan aplikasi adalah membuat *username*, *password*, dan data-data pengguna seperti nama, *email*, biografi, serta foto pengguna. Setelah melakukan pendaftaran, pengguna dapat masuk menggunakan *username* dan *password* yang telah dibuat, setelah *login* berhasil maka akan menuju ke halaman *dashboard*.

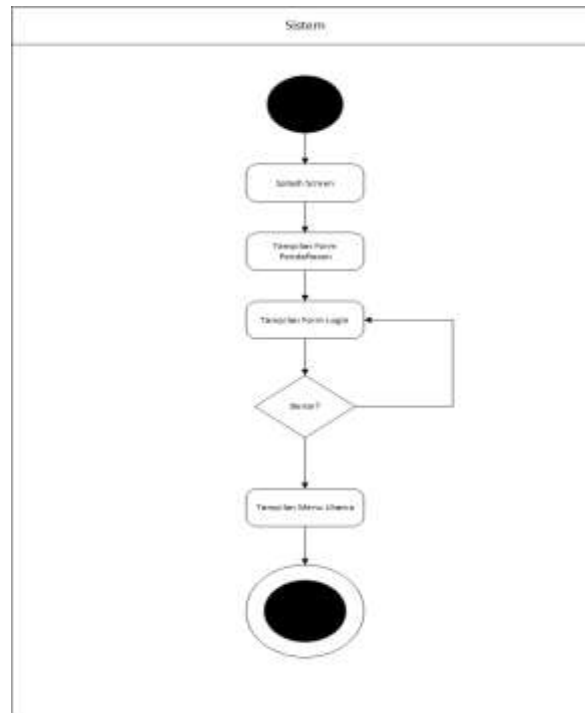
Pada halaman *dashboard* pengguna dapat memilih tujuan wisata, tersedia beberapa wisata yang ditawarkan pada aplikasi. Setelah memilih wisata, selanjutnya pengguna akan melihat informasi wisata tersebut, berupa nama wisata, waktu dan tanggal wisata, harga, fasilitas yang didapatkan serta ketentuan-ketentuan yang

berlaku. Selanjutnya pengguna dapat menekan tombol bayar untuk dapat membeli tiket. Setelah itu tiket elektronik akan ditampilkan, dan dapat dilihat di halaman profil pengguna terhadap tiket apa saja yang sudah dibeli. Pada halaman *dashboard* pengguna dapat mengubah profil nya, berupa nama, *email*, *password*, dan foto profil, untuk dapat terus menggunakan profil yang terkini.

b. Activity Diagram

Menurut Rosa dalam jurnal (Heriyanto Y, 2018) mengungkapkan “*Activity Diagram* menggambarkan *work flow* (aliran kerja) atau aktifitas dari sebuah sistem atau sebuah bisnis. Yang perlu diperhatikan disini ialah bahwa diagram aktifitas menggambarkan aktifitas sistem bukan yang dilakukan oleh aktor, jadi aktifitas dapat dilakukan oleh sistem”.

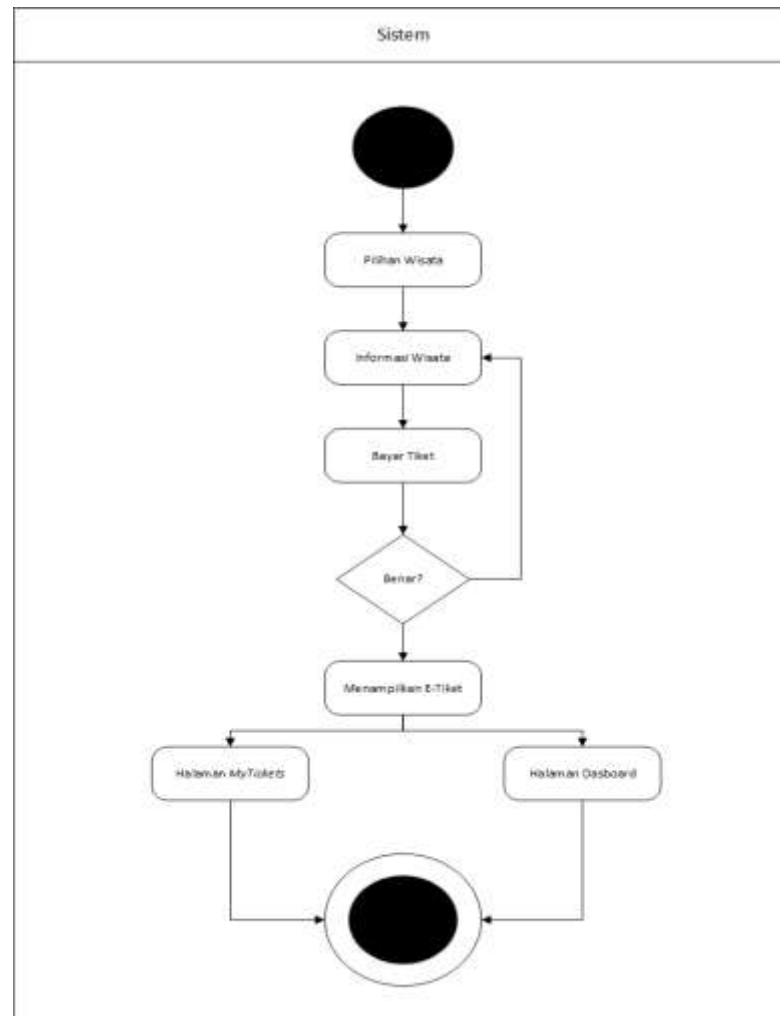
1. Proses Masuk ke Aplikasi



Gambar 3.5 *Activity Diagram* pada Aplikasi Pemesanan Tiket

Dengan adanya gambar diatas menjelaskan aktifitas *diagram* pada proses masuk ke dalam aplikasi, dimana ketika aplikasi dimulai maka akan menjalankan halaman *splash screen*, dan secara otomatis akan menuju ke halaman pendaftaran, ketika telah berhasil membuat data pengguna selanjutnya login menggunakan data yang telah dibuat, dan akan menuju ke halaman utama aplikasi.

2. Proses Pembelian Tiket

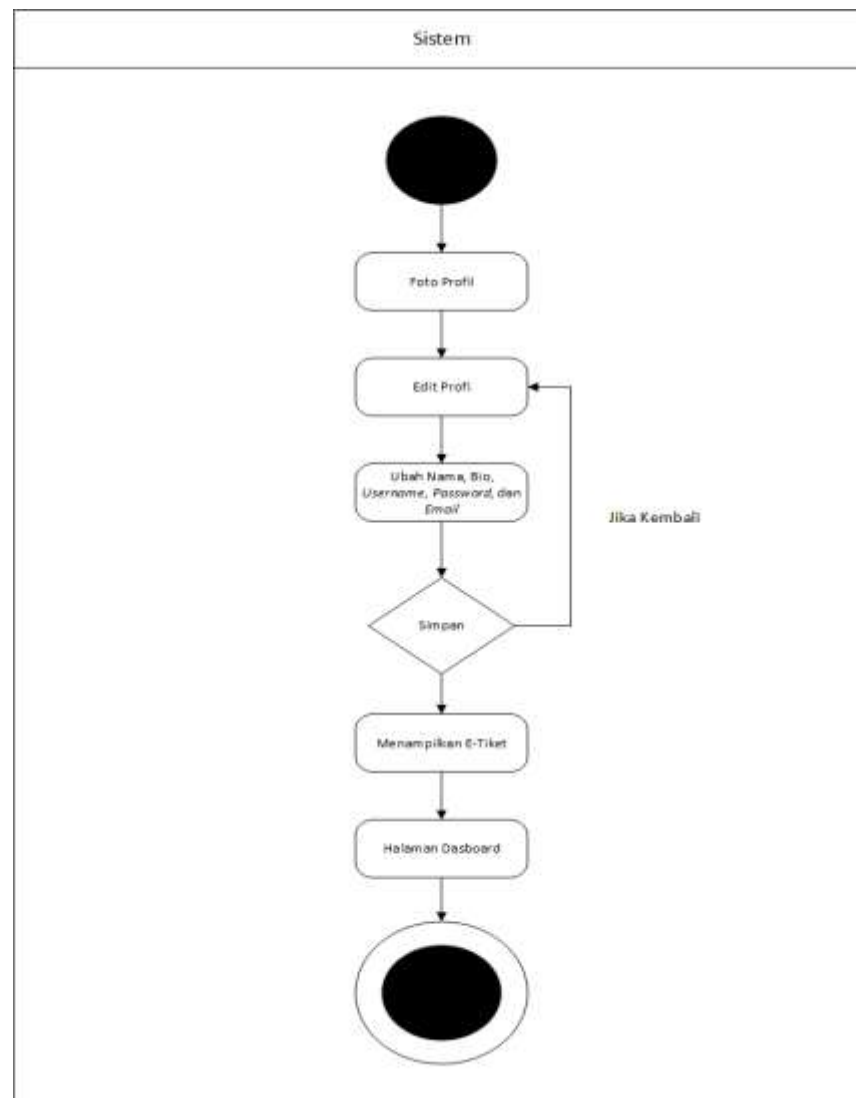


Gambar 3.6 Activity Diagram Pembelian Tiket

Dimana gambar diatas menjelaskan proses pembelian tiket, mulai dari pemilihan tiket wisata, kemudian menuju halaman informasi wisata dengan menampilkan informasi-informasi wisata yang akan dibeli seperti lokasi dan harga.

Setelah semua sudah sesuai maka tahap selanjutnya ialah membeli tiket, dan setelah membeli tiket maka akan menuju halaman *my tickets* atau halamana *dashboard*.

3. Proses *Edit Profile*



Gambar 3.7 Activity Diagram Edit Profile

Pada gambar diatas menjelaskan aktifitas *diagram* untuk melakukan *edit profile* pengguna, dengan memilih foto profil kemudian mengklik tombol *edit profile* maka akan menuju halaman edit profile, data berupa nama, bio, *username*, *password* dan *email address* dapat diubah sesuai keinginan, sistem akan menampilkan data pengguna sebelumnya agar mempermudah dalam perubahan data.

3.4.3 Struktur Database

Untuk menyimpan seluruh data pengguna dan kebutuhan data aplikasi diperlukannya sebuah basis data untuk dapat menampungnya, basis data yang digunakan pada aplikasi ini ialah *Firebase*, format yang digunakan pada basis data ini merupakan JSON, dan berikut struktur data pada *database* aplikasi pemesanan tiket wisata:

a. Database Tabel Data Tiket

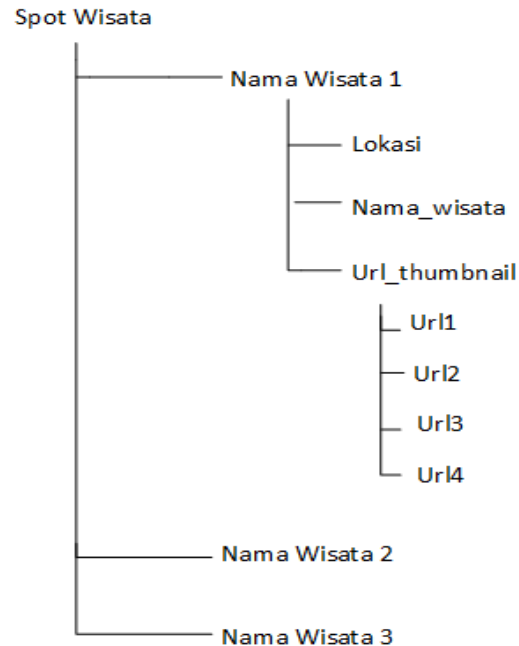


Gambar 3.8 Tabel Data Tiket *Database*

Dari tabel *database* diatas terlihat struktur tabel yang menyimpan beberapa data pengguna, dan berikut penjelasannya:

4. Data Tiket: Berisi nama pengguna
5. Date_wisata: Jadwal tanggal akan dilaksanakannya keberangkatan ke tempat wisata.
6. Id_tiket: ID tiket dibuat secara acak oleh sistem, ID berfungsi untuk mempermudah seorang *administrator* dalam mencari data pengguna jika dibutuhkan.
7. Ketentuan: Ketentuan apa saja yang harus ditaati ketika ingin berkunjung ketempat wisata.
8. Lokasi: Merupakan tempat wisata yang akan dikunjungi
9. Nama_wisata: Nama wisata yang akan dikunjungi
10. Time_wisata: Waktu berjalannya wisata, dari pagi hingga malam.

c. *Database Tabel Spot Wisata*



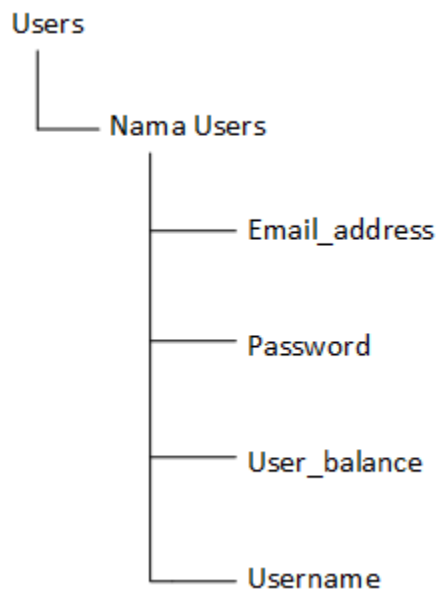
Gambar 3.9 Tabel *Spot Wisata Database*

Dari table *database* diatas terlihat struktur tabel yang menyimpan beberapa data pengguna, dan berikut penjelasannya:

1. *Spot Wisata*: Untuk menampung data wisata, yaitu tempat wisata apa yang akan tampil pada aplikasi.
2. Nama Wisata 1,2,3, dst: Tempat wisata yang tersedia, agar dapat dipilih melalui aplikasi.
3. Lokasi: Menampung data lokasi wisata.
4. Nama_wisata: Nama wisata yang dapat dikunjungi.

5. Url 1,2,3,4: Tempat menyimpan url gambar wisata, yang ditampilkan pada aplikasi.

d. Tabel *Users*



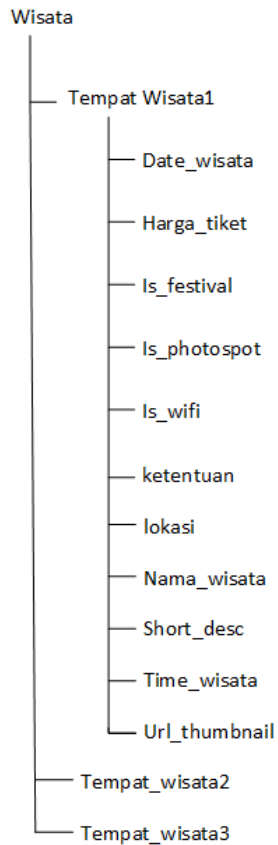
Gambar 3.10 Tabel *Users Database*

Dari tabel database diatas terlihat struktur table yang menyimpan beberapa data pengguna, dan berikut penjelasannya:

1. *Users*: Menampung data-data dengan nama pengguna.
2. *Nama Users*: Meyimpan seluruh nama *users* yang pernah mendaftar.
3. *Email_address*: Tempat menyimpan alamat email yang telah didaftar.
4. *Password*: Tempat menyimpan *password* pengguna.
5. *User_balance*: Jumlah saldo pengguna.

6. *Username*: Tempat menyimpan nama pengguna.

e. Tabel Wisata



Gambar 3.11 Tabel Wisata *Database*

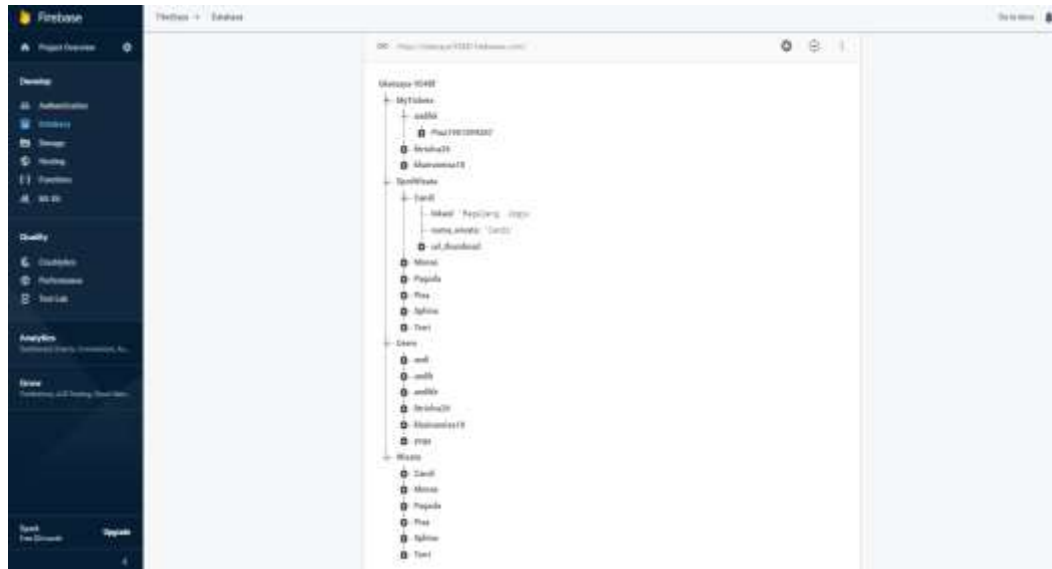
Dari tabel *database* diatas terlihat struktur tabel yang menyimpan beberapa data pengguna, dan berikut penjelasannya:

1. Wisata: Menyimpan daftar nama wisata yang akan ditampilkan di aplikasi
2. Tempat Wisata1, 2,3, dst: Daftar nama wisata yang tersedia saat ini.

3. Harga Tiket: Menyimpan harga tiket wisata
4. *Is-festival*: Menyimpan data *festival* apa yang sedang terjadi.
5. *Is_wifi*: Keterangan tersedia Wifi atau tidak di tempat wisata.
6. Ketentuan: Menyimpan ketentuan yang harus dipenuhi pengunjung ketika berwisata.
7. Lokasi; Menyimpan data lokasi wisata
8. Nama_wisata: Menyimpan nama wisata yang akan ditampilkan di aplikasi
9. *Short_desc*: Deskripsi singkat tentang wisata yang akan dikunjungi
10. *Time_wisata*: Menyimpan waktu keberangkatan ke tempat wisata.
11. *url_thumbnail*: Menyimpan gambar wisata yang ditampilkan di aplikasi.

f. *Database* Secara Keseluruhan

Setelah diuraikan beberapa tabel *database* diatas, berikut ini merupakan tampilan *database* secara keseluruhan.



Gambar 3.12 Database Firebase Aplikasi Pemesanan Tiket Wisata

Secara keseluruhan *database* terdiri dari table *mytickets*, *spot* wisata, data pengguna, dan data wisata. Setiap tabel memiliki data yang ditampungnya masing-masing, berupa data-data tiket, data-data pengguna ketika mendaftar seperti: data *username*, *password*, *email* dan lainnya. Tidak hanya data pengguna tersimpan di *database*, data wisata seperti nama wisata yang akan dituju, jadwal wisata, serta beragam ketentuannya.

3.4.4 Rancangan Antar Muka

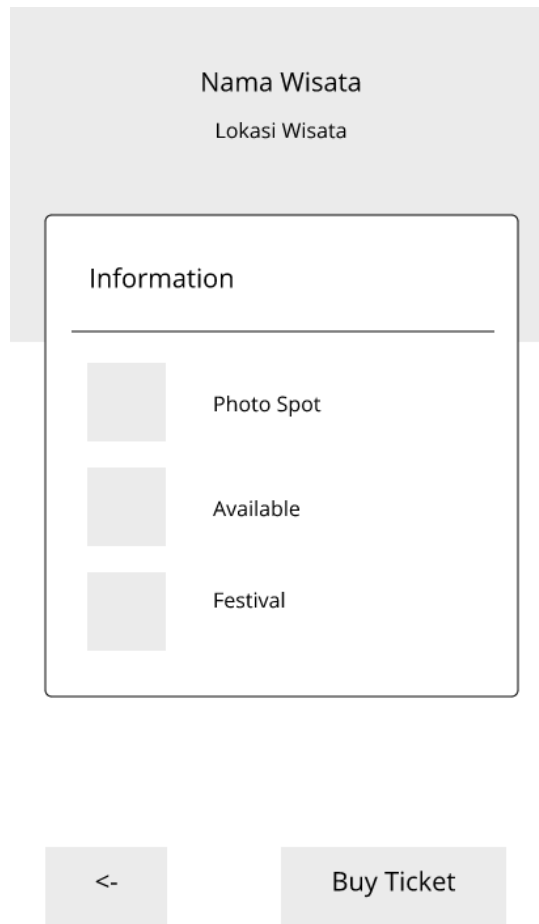
Perancangan antar muka merupakan gambaran tampilan awal ketika pengguna telah berhasil *login* ke aplikasi yang akan dibuat. Dan berikut merupakan rancangan sistem yang akan dibuat.

a. Rancangan *Dashboard*

Gambar 3.13 Halaman *Dashboard*

Halaman utama pada aplikasi berisi menu-menu penting di dalam aplikasi, seperti pemilihan wisata yang akan dikunjungi, foto profil pengguna, nama lengkap dan biografi. Tidak hanya itu jumlah saldo yang dimiliki pengguna juga ditampilkan pada halaman utama ini. Selain itu juga terdapat menu *spot* wisata yang direkomendasikan kepada pengguna serta histori pengguna jika pernah membeli tiket sebelumnya.

b. Rancangan Informasi Tiket



Gambar 3.14 Halaman Informasi Tiket

Ketika telah memilih wisata, maka akan ditampilkan informasi dari wisata yang akan dikunjungi, informasi yang ditampilkan berupa beberapa foto *spot* wisata, ketersediaan fasilitas yang didapatkan ketika membeli tiket wisata yang dituju. Tidak hanya itu terdapat juga informasi berupa *festival* yang sedang

berlangsung di tujuan wisata nantinya, setelah merasa yakin pengguna dapat menekan tombol *buy ticket* untuk menuju ke halaman *ticket detail*.

c. Rancangan Beli Tiket

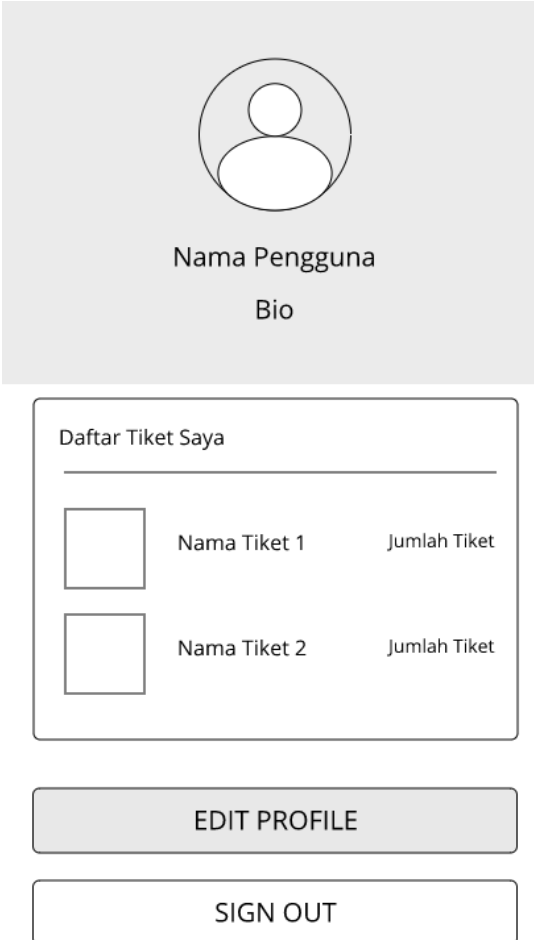
The wireframe shows a 'Buy Ticket' page layout. At the top is the title 'Buy Ticket'. Below it is a 'Ticket Details' section, which includes a placeholder for an image and labels for 'Nama Wisata' and 'Lokasi'. The 'Pricing' section displays 'Total Saldo Terakhir' with a 'Saldo' button, and 'Total Harga' with the value 'RP. xxx.xxx'. At the bottom are 'Back' and 'PAY NOW' buttons.

Gambar 3.15 Halaman *Buy Ticket*

Pada halaman *ticket details* berisi informasi nama wisata dan lokasi yang akan di beli, dan juga ditampilkan saldo terakhir pengguna beserta harga tiket yang akan dibeli. Setelah semua informasi dirasa sudah pas, pengguna dapat

menekan tombol *pay now*, setelah itu maka akan muncul *e-ticket* yang nanti dapat digunakan sebagai bukti pembelian.

d. Rancangan Profil



The image shows a user profile design. At the top is a grey rectangular area containing a circular placeholder for a profile picture, the text "Nama Pengguna", and "Bio". Below this is a white rounded rectangle titled "Daftar Tiket Saya" with a horizontal line underneath. It contains two rows of ticket information, each with a square placeholder, "Nama Tiket 1" or "Nama Tiket 2", and "Jumlah Tiket". At the bottom are two buttons: a grey "EDIT PROFILE" button and a white "SIGN OUT" button.

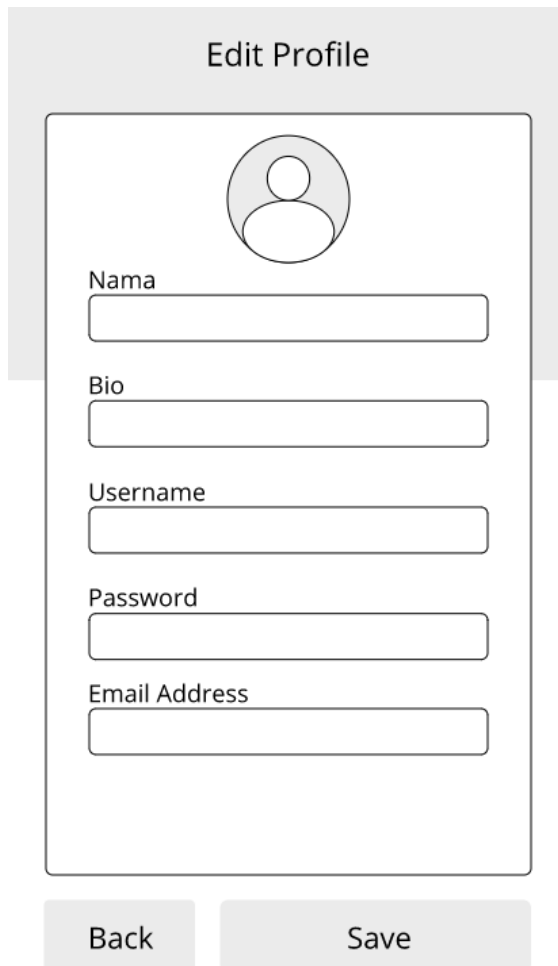
Gambar 3.16 Halaman Profil Pengguna

Profil pengguna dirancang untuk mempermudah pengguna dalam melihat data dirinya, data yang ditampilkan pada halaman ini berupa foto, nama pengguna, serta biografi. Selain itu pada *layout* dibawahnya terdapat daftar tiket

yang telah dibeli serta jumlahnya, jadi tidak perlu khawatir untuk melihat *e-ticket* yang telah dibelei sebelumnya.

Dibawah terdapat tombol menuju halaman edit profil jika ingin mengubah data diri, dan tombol sign out jika pengguna keluar dan mengganti akun dengan akun lainnya.

e. Rancangan *Edit Profile*



The image shows a mobile application interface for editing a user profile. The title 'Edit Profile' is centered at the top. Below the title is a circular placeholder for a profile picture. Underneath the picture are five text input fields, each with a label above it: 'Nama', 'Bio', 'Username', 'Password', and 'Email Address'. At the bottom of the form are two buttons: 'Back' on the left and 'Save' on the right.

Gambar 3.17 Halaman *Edit Profile*

Perancangan halaman *edit profile* tidak kalah pentingnya dengan halaman lainnya, halaman ini berfungsi untuk mengubah data pengguna yang pernah dibuat. Data yang dapat diubah seperti foto profil, nama, bio, *username*, *password*, dan *email address*. Halaman *edit profile* akan menampilkan data sesuai *field*-nya diatas dari *database*, sehingga dapat mempermudah pengguna sebelum mengubah seluruh informasi yang pernah dibuat.

BAB IV

HASIL DAN PEMBAHASAN

4.1 Kebutuhan Spesifikasi *Minimum Hardware dan Software*

Dalam perancangan dan pengujian aplikasi android, diperlukannya syarat minimal perangkat keras dan perangkat lunak agar bisa berjalan optimal, untuk itu berikut spesifikasi *hardware* dan *software* yang dibutuhkan.

a. *Hardware*

Kebutuhan hardware yang digunakan untuk menjalankan sistem adalah sebagai berikut:

1. Laptop dengan Processor Intel Core i3 M390, 2,67 Ghz

Dengan menggunakan processor Intel Core i3 generasi pertama dengan inti *dual core* masih cukup untuk menjalankan Android Studio dan *software* desain Adobe XD. Android Studio merekomendasi processor minimal dengan 2 inti agar mempermudah dalam *multitasking*, dan processor yang mendukung VR (*Virtual Reality*).

2. RAM 8 GB DDR3 10600, 1333 Mhz

Untuk dapat menjalankan Android Studio melalui halaman resminya, Android Studio menyarankan agar menggunakan RAM minimal 4 GB dan rekomendasi 8 GB agar lancar ketika menjalankan *Emulator*.

3. GPU AMD Radeon HD 6470 M

Disini saya menggunakan GPU AMD Radeon HD 6470 M, GPU dibutuhkan untuk dapat menjalankan software desain Adobe XD, driver VGA yang dibutuhkan ialah minimal tahun 2012. Selain itu GPU ini mempermudah dalam menjalankan *Emulator* Android pada Android Studio

4. SSD 250 GB + Harddisk 500 GB

Peyimpanan sangat dibuthkan untuk menyimpan file-file yang dibutuhkan dalam membangun aplikasi, untuk itu disini saya menggunakan 2 buah storage yaitu SSD (*Solid State Drive*) dan Harddisk. SSD berkapasitas 250 GB saya gunakan untuk menginstall sistem operasi windows + linux dan project Android. Saya memilih menggunakan SSD sebagai penyimpanan project karena SSD terkenal dengan kecepatannya yang bisa 10 kali lipat dibandingkan dengan Harddisk. Untuk Harddisk sendiri saya gunakan untuk menyimpan file-file gambar, software pendukung, dan lainnya yang tidak membutuh performa yang tinggi.

b. *Software*

1. Sistem Operasi Kali Linux 64bit dan Windows 10 64bit

Sebagai sistem utama dalam perancangan aplikasi android, dan agar dapat terinstallnya *software* perancang aplikasi yaitu Android Studio dan *software* untuk mendesain tampilan aplikasi yaitu Adobe XD.

2. Android Studio 64 bit

Untuk dapat melakukan pengembangan terhadap aplikasi android diperlukannya *software* pendukung yaitu Android Studio. Android Studio memiliki fitur lengkap yang mempermudah *developer* untuk mengembangkan aplikasi yang ingin dibuat, seperti *emulator* yang berfungsi untuk menguji coba aplikasi, fitur debugging untuk menguji apakah aplikasi masih mendapatkan bug agar bisa diperbaiki.

3. Adobe XD

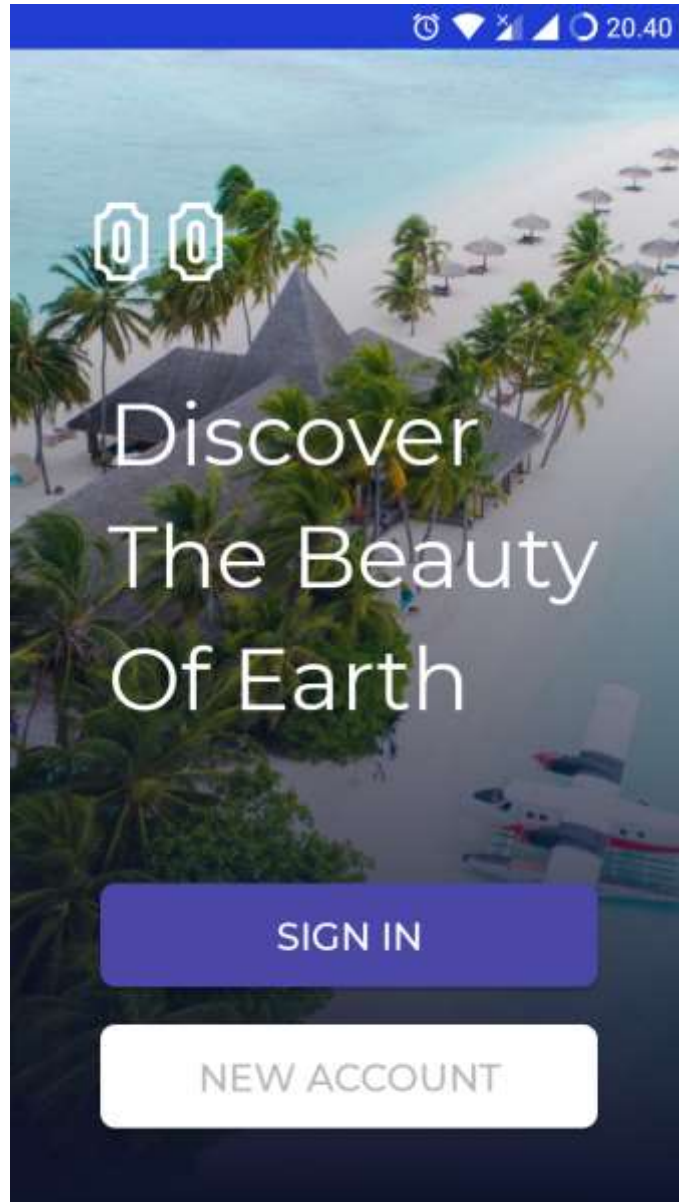
Untuk membuat tampilan aplikasi yang diinginkan, diperlukannya software desain seperti Adobe XD, membuat tampilan mulai dari halaman *login*, pendaftaran, hingga halaman *dashboard*. Pewarnaan, tata letak, serta animasi dapat dibuat dengan Adobe XD dan kemudian diterapkan pada Android Studio.

4.2 Pengujian Aplikasi dan Pembahasan

4.2.1 Hasil Tampilan Sistem

Berikut merupakan tampilan halaman aplikasi yang telah berhasil penulis buat.

- a. Tampilan Halaman Menuju *Sign in* dan *Sign Up*

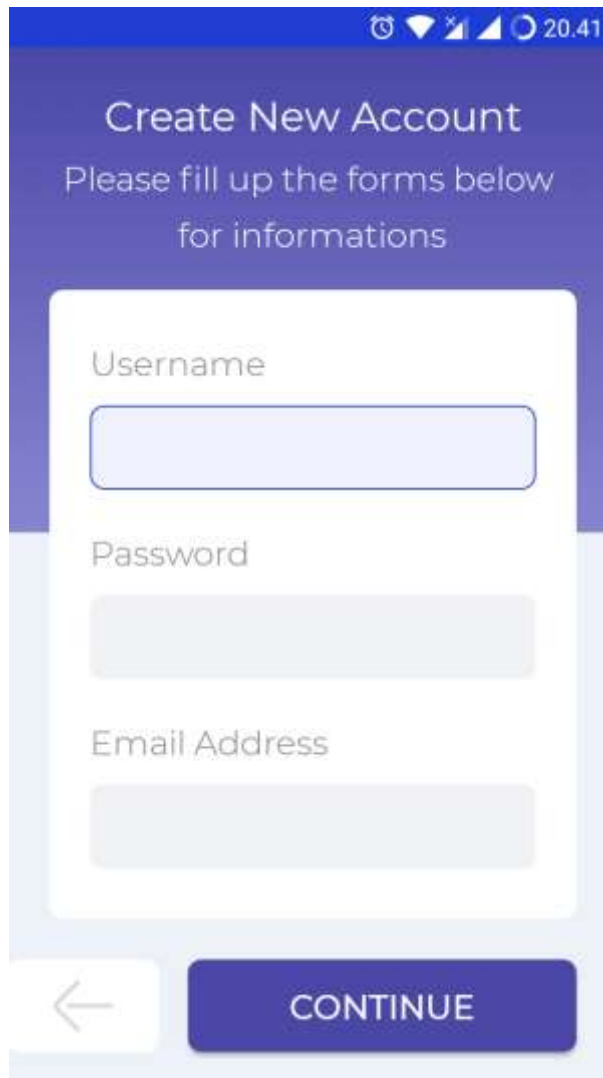


Gambar 4.1 Halaman Menuju *SIGN IN* dan *New Account*

Gambar diatas merupakan tampilan halaman ketika aplikasi pertama dijalankan, terdapat gambar *background* pantai yang menggambarkan

sebuah lokasi wisata. Dan terdapat teks besar yang menuliskan betapa indahny bumi. Dan terdapat juga 2 buah *button* yaitu *button Sign In* dan *button New Account*, yang ketika *button Sign In* di tekan maka akan menuju kehalaman *Sign In* untuk masuk ke halaman utama, dan ketika *button New Account* maka akan menuju ke halaman *New Account* untuk membuat data pengguna baru, dan pengguna dapat mengisi sesuai identitas yang dimiliki.

b. Tampilan Halaman *New Account*



20.41

Create New Account

Please fill up the forms below
for informations

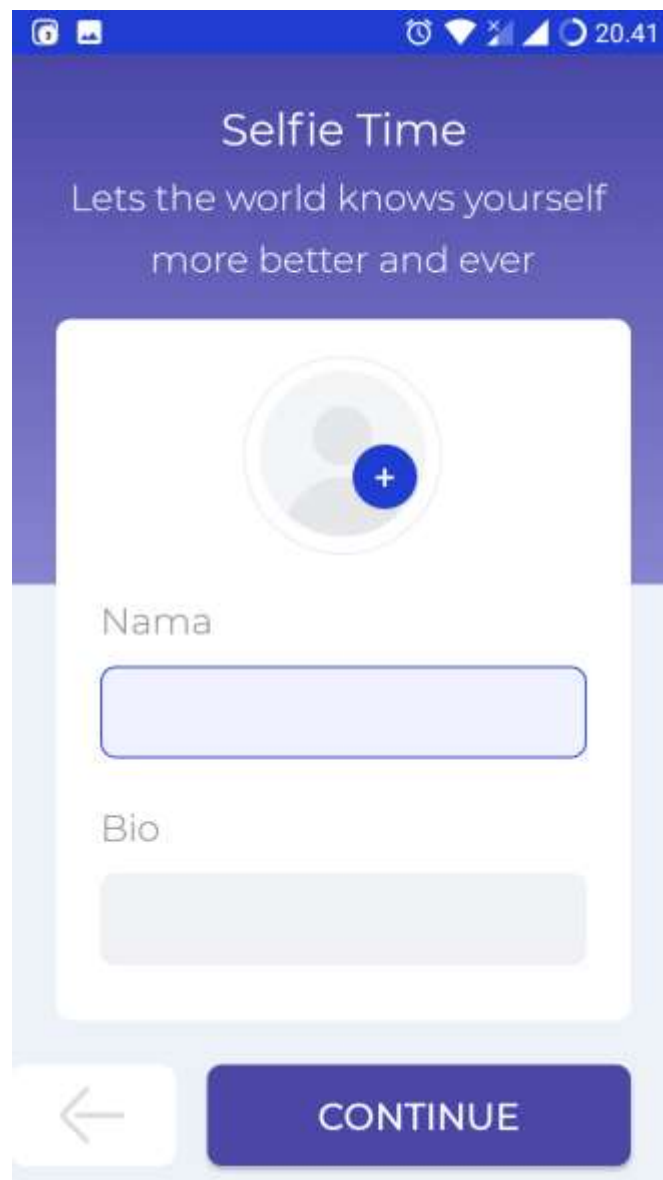
Username

Password

Email Address

← CONTINUE

Gambar 4.2 Halaman Membuat Akun Baru



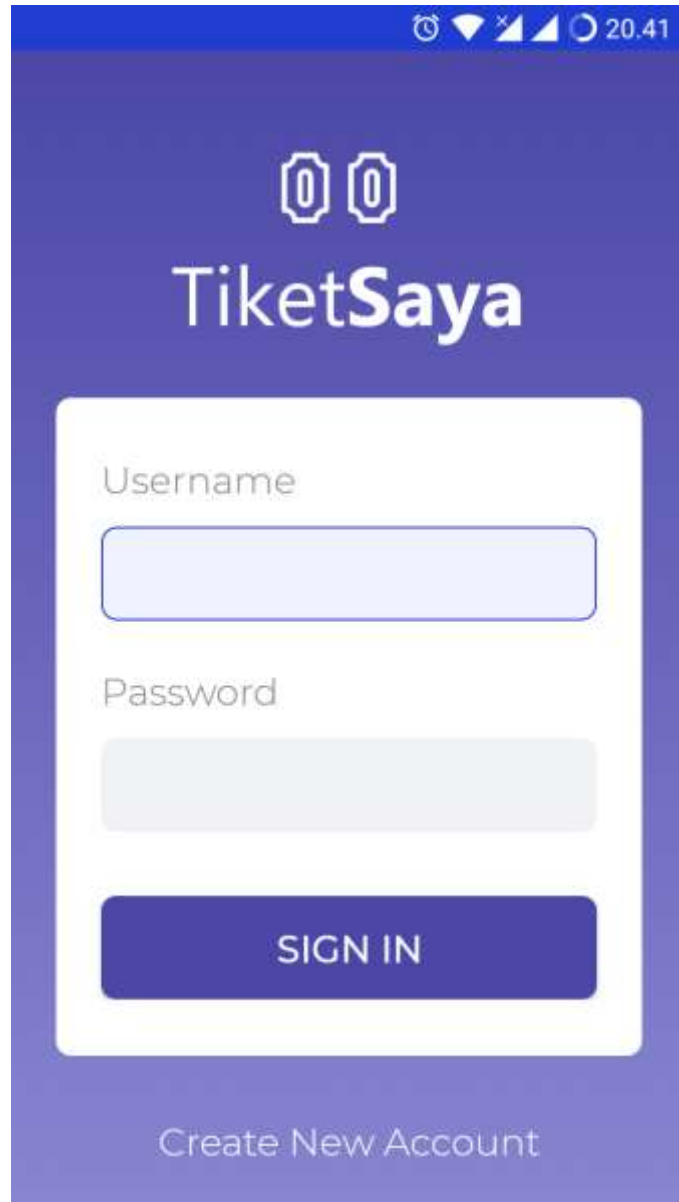
Gambar 4.3 Halaman Mengisi Nama, Bio, dan Unggah Foto

Pada halaman ini pengguna membuat akun dengan mengisi *username*, *password*, *email address*, nama, *bio*, dan juga mengupload foto. Setelah berhasil membuat akun baru, maka akan muncul notifikasi berhasil

membuat akun dan dapat menekan tombol *explore* untuk menuju ke halaman *dashboard*.



Gambar 4.4 Halaman Pendaftaran Berhasil

c. Tampilan Halaman *Sign In*

Gambar 4.5 Halaman *Sign In*

Halaman diatas merupakan halaman *Sign In* yang berfungsi untuk masuk ke aplikasi, pengguna dapat mengetikkan *username* dan *password*

yang telah dibuat sebelumnya, setelah berhasil masuk maka akan menuju halaman *dashboard*.

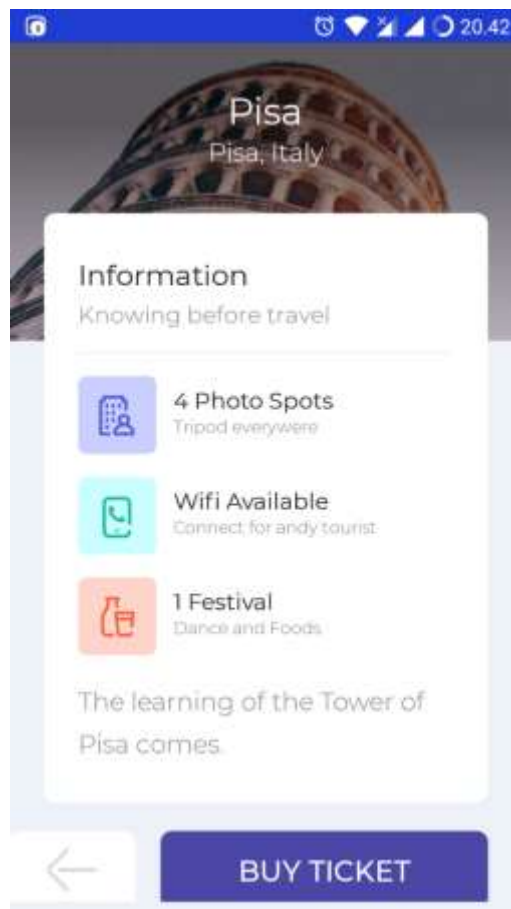
d. Tampilan Halaman *Dashboard*



Gambar 4.6 Halaman *Dashboard*

Gambar diatas merupakan halaman *dashboard* tiket, dimana pada halaman ini menampilkan beberapa informasi seperti profil pengguna, nama lengkap, *bio*, jumlah saldo, dan menu wisata yang tersedia. Untuk dapat membeli tiket wisata yang diinginkan, pengguna harus mengklik salah satu tiket yang tersedia di menu, dan jika ingin mengubah data pengguna, pengguna dapat mengklik foto yang ada di halaman *dashboard* tiket.

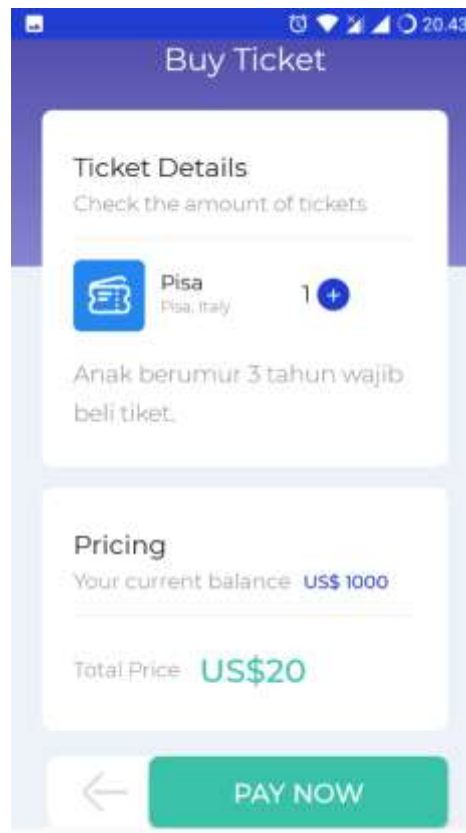
e. Halaman Informasi Tiket



Gambar 4.7 Halaman Informasi Tiket

Gambar diatas merupakan halaman informasi tiket yang akan dibeli, terdapat nama dan lokasi wisata yang akan dituju, dibagian *header* terdapat berupa gambar tempat wisata, dan dibagian *information* terdapat informasi berupa spot foto untuk melihat keindahan alam tempat yang akan dikunjungi, dan ketersediaan internet di tempat wisata juga *festival* yang akan berlangsung selama pengunjung berada disana. Setelah semuanya merasa cocok pengguna bisa menekan tombol klik dan akan menuju ke halaman pembelian tiket.

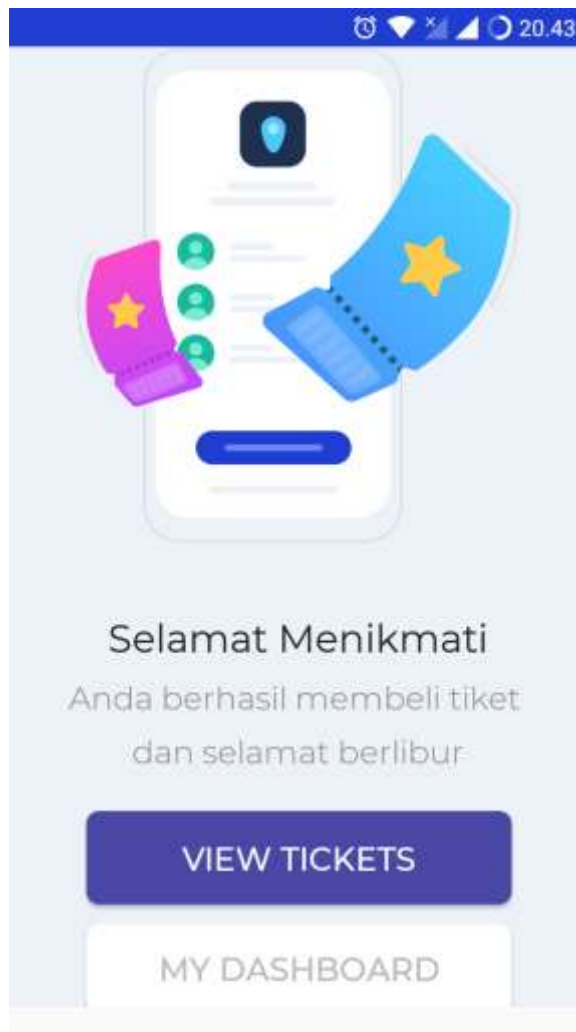
f. Halaman Membeli Tiket



Gambar 4.8 Halaman Membeli Tiket

Gambar diatas merupakan halaman membeli tiket, terdapat informasi tiket *detail* berupa nama tiket yang akan dibeli, jumlah tiket, harga tiket dan saldo pengguna, juga ketentuan dalam membeli tiket. Setelah semuanya dipilih maka selanjutnya adalah menekan tombol *Pay Now*, maka akan muncul halaman tiket berhasil dibeli.

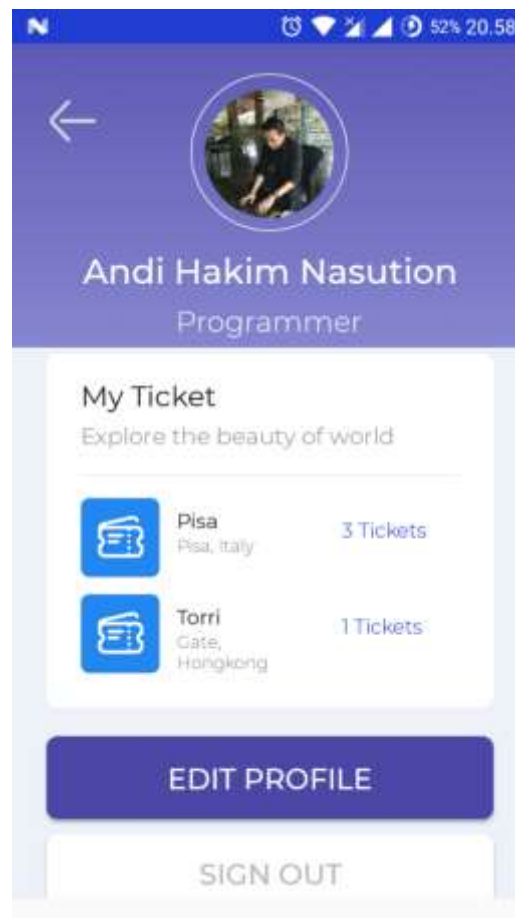
g. Halaman Berhasil Beli Tiket



Gambar 4.9 Halaman Berhasil Membeli Tiket

Gambar diatas ada halaman tiket berhasil dibeli, halaman ini akan muncul setelah pengguna melakukan pembayaran pada halaman membeli tiket. Pada halaman ini terdapat notifikasi selamat tiket berhasil dibeli dan selamat beribur, dan terdapat 2 button yaitu button *View Tickets* dan *My Dashboard*. *Button View Ticket* jika di klik maka akan menuju ke halaman tiket yang sudah dibeli, dan *button My Dashboard* jika di klik maka akan menuju ke halaman Dashboard aplikasi,

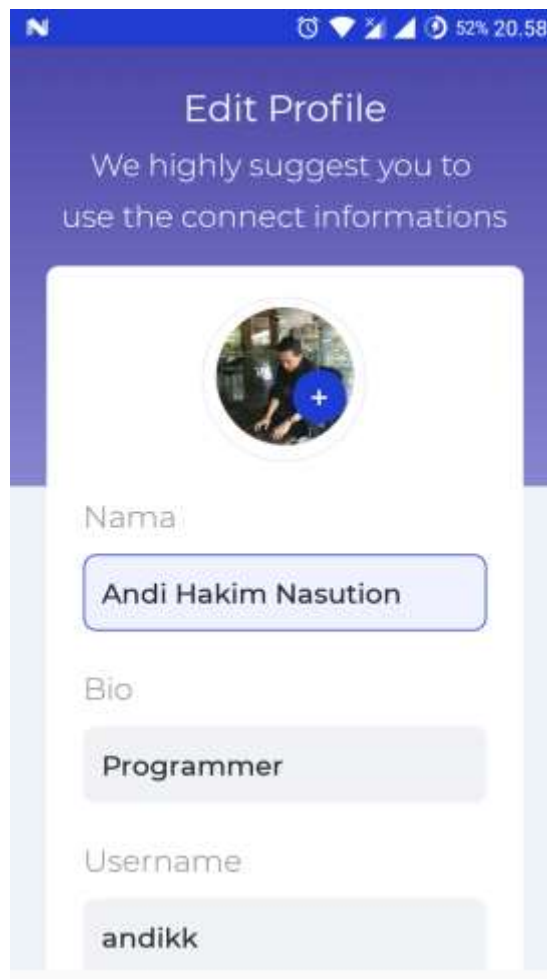
h. Halaman Profil Pengguna



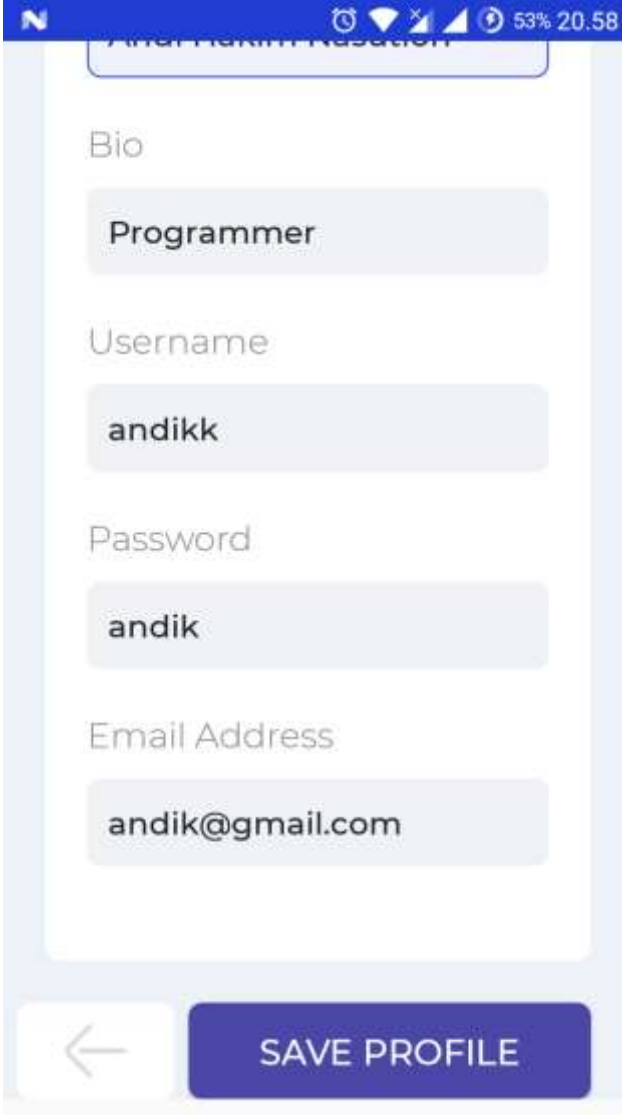
Gambar 4.10 Halaman Profil Pengguna

Gambar diatas merupakan halaman Profil Pengguna, untuk dapat membuka halaman ini pengguna dapat mengklik foto profil yang ada dihalaman *dashboard*. Pada halaman profil terdapat informasi berupa foto pengguna, nama, bio, dan informasi tiket yang sudah dibeli dengan jumlah tiketnya. Dan terdapat juga tombol *Edit Profile* untuk mengedit profil pengguna, juga tombol *Sign Out* jika ingin keluar dari aplikasi.

i. Halaman *Edit Profile*



Gambar 4.11 Halaman *Edit Profile*



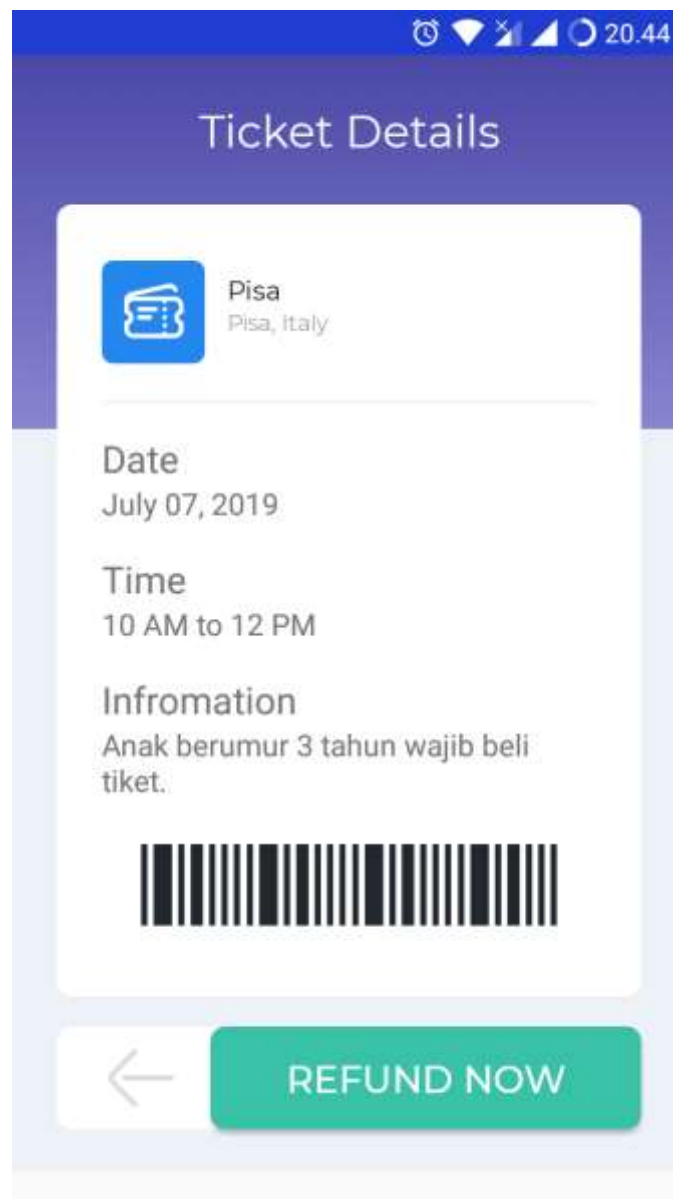
The image shows a mobile application interface for editing a profile. The screen is titled "Andi Hakim Resdation" at the top. Below the title, there are four input fields for profile information: "Bio" with the value "Programmer", "Username" with the value "andikk", "Password" with the value "andik", and "Email Address" with the value "andik@gmail.com". At the bottom of the form, there is a back arrow button on the left and a "SAVE PROFILE" button on the right. The status bar at the top of the phone shows a signal strength icon, a Wi-Fi icon, a battery icon at 53%, and the time 20:58.

Gambar 4.12 Halaman 2 *Edit Profile*

Gambar diatas merupakan halaman *Edit Profile*, pengguna dapat mengubah data yang dibuat pertama kali saat mendaftar. Data yang dapat diubah berupa nama, *bio*, *username*, *password*, *email address* dan foto

profil pengguna. Jika data telah diubah, pengguna dapat menekan tombol *Save Profile*, maka data akan tersimpan ke *database*.

j. Halaman *Ticket Details*



Gambar 4.13 Halaman *Ticket Details*

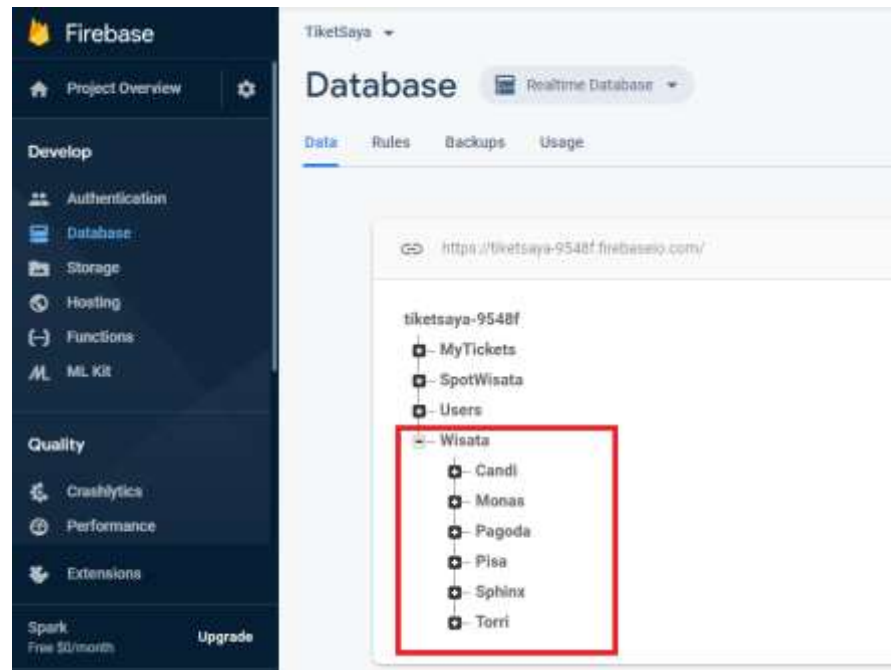
Gambar diatas merupakan halaman *Ticket Details*, dimana halaman ini menampilkan informasi nama tiket yang telah dibeli, jadwal keberangkatan, dan informasi ketentuan dalam pembelian tiket ini. Dan tersedia *barcode* untuk proses *claim* tiket ditempat keberangkatan. Terdapat juga tombol *Refund Now* untuk pembatalan keberangkatan.

4.2.2 Pengujian Aplikasi

1. Proses Pemasukan Data

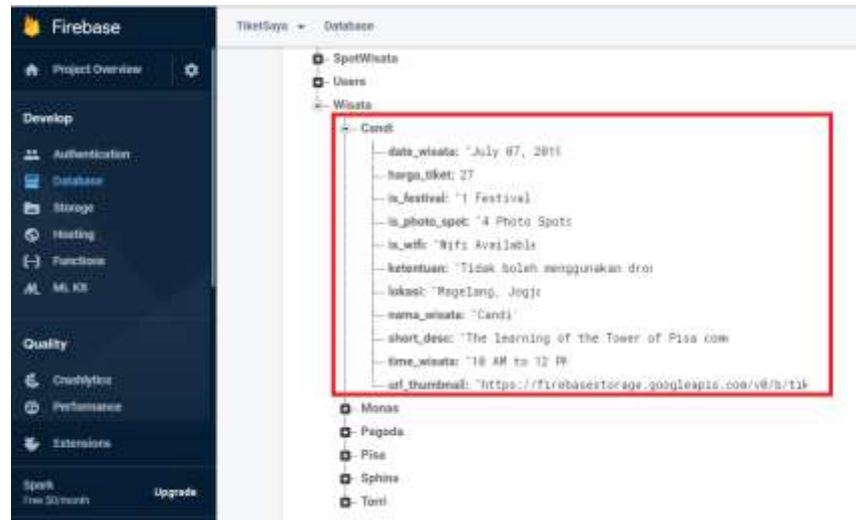
a. Pemasukan Data Oleh *Developer*

Pada database seorang *developer* memasukkan data wisata yang tersedia, sehingga nantinya data akan tampil pada aplikasi. Data wisata berupa tempat wisata yang tersedia saat ini seperti Candi, Monas, Pagoda, Pisa, Sphinx, dan Torri.



Gambar 4.14 Pengimputan Data ke *Database* oleh *Developer*

Didalam tabel *Wisata* terdapat data tiket, data tiket berupa *date wisata*, *harga tiket*, *festival*, *photospot*, *wifi*, *ketentuan*, *lokasi*, *nama wisata*, *short description*, *waktu wisata*, dan *url thumbnail*, semua data ini yang akan tampil pada aplikasi.



Gambar 4.15 Data Wisata di *Database*

b. Pemasukkan Data Oleh Pengguna

Pada saat aplikasi dijalankan pengguna akan mendaftar untuk membuat akun baru, pada proses ini pengguna menginputkan data pada form pendaftaran di halaman *New Account* pada aplikasi, dan ketika pengguna menekan tombol *Continue* maka seluruh data akan dikirim ke *database firebase* dan disimpan.

Create New Account
Please fill up the forms below for informations

Username: andihakimnst

Password: andik

Email Address: andihakimnst@gmail.com

Bio: Programmer

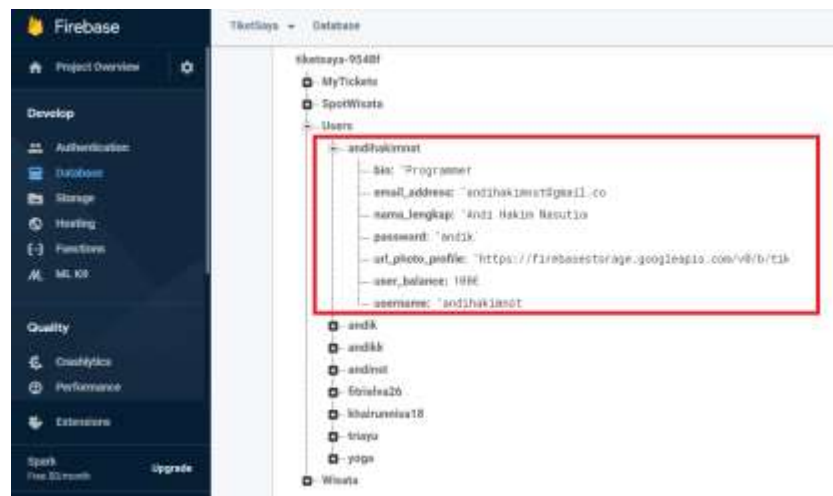
Username: andikk

Password: andik

Email Address: andik@gmail.com

CONTINUE SAVE PROFILE

Gambar 4.16 Penginputan Data oleh Pengguna



Gambar 4.17 Data Berhasil dibuat di *Database*

2. Proses Transaksi Data

Ketika pengguna melakukan *login* ke aplikasi maka terjadilah proses transaksi data dari *form login* ke *firebase*, aplikasi akan memeriksa apakah data yang diinputkan pengguna ada di *database*, jika ada maka selanjutnya akan menuju ke halaman *dashboard*.



Gambar 4.18 Pengguna Memasukkan *Username* dan *Password*


```

@Override
public void onDataChange(@NonNull DataSnapshot dataSnapshot) {
    dataSnapshot.getValue(Boolean.class);
    //Validasi data password baru
    String passwordFromDatabase = dataSnapshot.getString("password").trim();
    //Validasi password sesuai format
    if(password.equals(passwordFromDatabase)) {
        //Validasi password login ke sistem
        SharedPreferences sharedPreferences = getSharedPreferences("SIKADAH", MODE_PRIVATE);
        SharedPreferences.Editor editor = sharedPreferences.edit();
        editor.putString("username_login", username.getText().toString());
        editor.apply();

        //Validasi session
        Intent getIntent = new Intent(getApplicationContext(), LoginActivity.class);
        getIntent.putExtra("username_login", username.getText().toString());
        startActivity(getIntent);

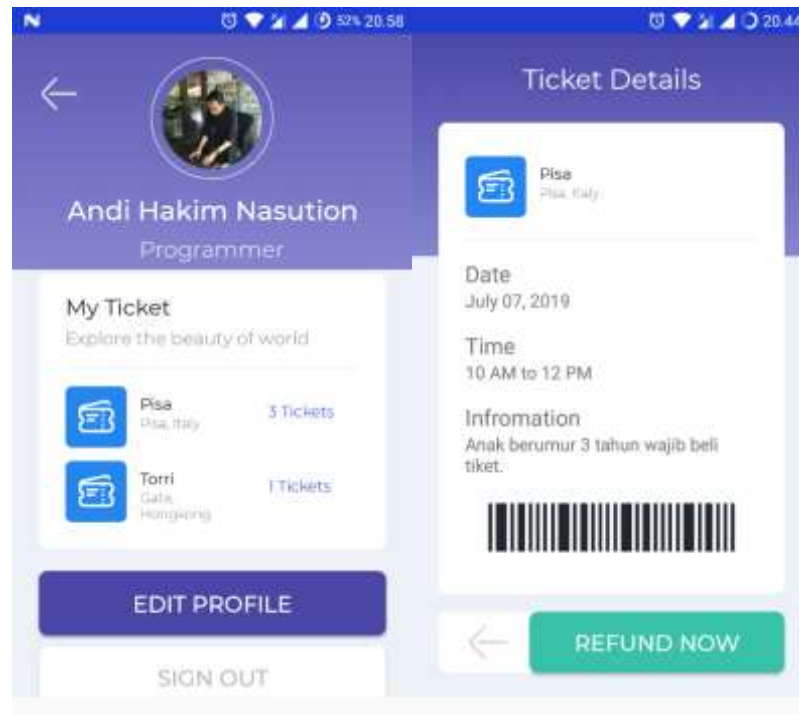
        //Validasi session login
        startActivity(getIntent);
        startActivity(getIntent);
        Toast.makeText(getApplicationContext(), "Pendaftaran Berhasil", Toast.LENGTH_SHORT, Toast.LENGTH_LONG);
    }
}

```

Gambar 4.19 Code untuk Melakukan Proses Login

3. Proses Pelaporan

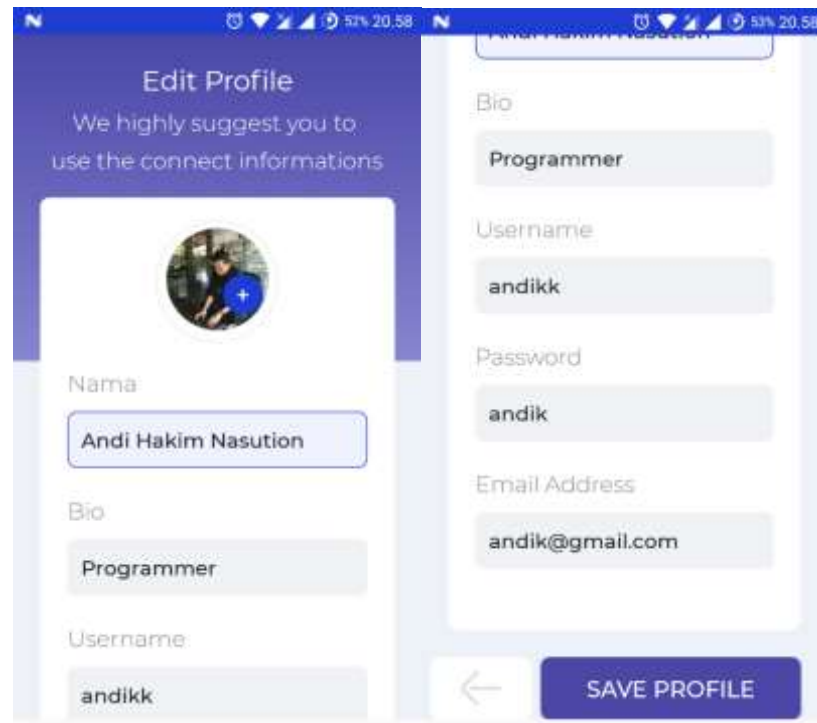
Setiap tiket yang telah berhasil dibeli maka akan muncul halaman notifikasi bahwa tiket berhasil dibeli. Data akan tersimpan di *database* dan akan ditampilkan kembali jika pengguna membuka halaman profil pengguna, proses pelaporan jumlah tiket ini lah yang bisa dilihat pengguna suatu saat jika dibutuhkan.



Gambar 4.20 Halaman Tiket di Profil Pengguna

4. Proses Keamanan Data

Proses keamanan data pengguna sangat penting dalam pembuatan aplikasi ini, *database* yang digunakan ialah database milik google yaitu *Firestore Database* yang memiliki keamanan yang kuat. Data akun pengguna pun dibuat sangat aman dengan fitur mengubah data pada halaman *edit profile* di aplikasi yang dapat digunakan pengguna mengubahnya secara berkala.



Gambar 4.21 Halaman Edit Profil Pengguna



Gambar 4.22 Database Setelah Edit Data Pengguna

BAB V

PENUTUP

5.1 Kesimpulan

Berdasarkan pembahasan dan penerapan Aplikasi Pemesanan Tiket Wisata Berbasis Android Menggunakan Firebase, maka dapat diambil kesimpulan sebagai berikut:

- a. Perancangan aplikasi pemesanan tiket wisata pada *platform* android bermaksud untuk mempermudah masyarakat dalam membeli tiket wisata secara *online* yang dapat menghemat waktu dan biaya sebelum pergi berliburan.
- b. Menggunakan sistem yang dibangun secara *realtime*, sehingga membuat pengguna dapat terus mendapatkan informasi tiket terbaru yang tersedia saat ini.
- c. Dengan menggunakan sistem yang dapat diakses 24 jam dalam 7 hari, sehingga mempermudah pengguna dalam membeli tiket wisata. Hal ini dimaksud untuk dapat membuat efisiensi waktu pengguna, kapan saja dan dimana saja dapat membeli tiket wisata melalui aplikasi ini.

5.2 Saran

Berikut merupakan saran yang penulis berikan berdasarkan pembahasan dalam suatu Perancangan Aplikasi Pemesanan Tiket Wisata *Online* Menggunakan *Firestore* yang akan diterapkan.

- a. Diharapkan kedepannya sistem ini tidak hanya dapat digunakan pada sistem operasi android tetapi dapat digunakan secara *multi platform* pada sistem operasi IOS, dan *Windows Phone* untuk mencakup lebih banyak pengguna.
- b. Diharapkan aplikasi ini memiliki lebih banyak fitur didalamnya, tidak sebatas untuk pemesanan tiket wisata saja, tetapi mendapat penambahan fitur seperti pembelian pulsa, token listrik, dan lain-lainnya. Dan oleh karena itu diperlukan pengembangan lebih lanjut terhadap aplikasi ini.

DAFTAR PUSTAKA

- Auto Parts Toyota Berbasis Aplikasi Java. *Jurnal TEKNOIF*, 1, 1–5.
- Azmi, Fadhillah, And Winda Erika. "Analisis Keamanan Data Pada Block Cipher Algoritma Kriptografi Rsa." *Cess (Journal Of Computer Engineering, System And Science)* 2.1: 27-29.
- Berbasis Android. (2018). *Jurnal Teknik Informatika Dan Sistem Informasi* 4(3), 397–406. Digital Berbasis Website Untuk Para Penulis. *IJSE – Indonesian Journal on*
- Erika, Winda, Heni Rachmawati, And Ibnu Surya. "Enkripsi Teks Surat Elektronik (E-Mail) Berbasis Algoritma Rivest Shamir Adleman (Rsa)." *Jurnal Aksara Komputer Terapan* 1.2 (2012).
- Hafni, Layla, And Rismawati Rismawati. "Analisis Faktor-Faktor Internal Yang Mempengaruhi Nilai Perusahaan Pada Perusahaan Manufaktur Yang Terdaftar Di Bei 2011-2015." *Bilancia: Jurnal Ilmiah Akuntansi* 1.3 (2017): 371-382.
- Hamdi, Muhammad Nurul, Evi Nurjanah, And Latifah Safitri Handayani. "Community Development Based On Ibnu Khaldun Thought, Sebuah Interpretasi Program Pemberdayaan Umkm Di Bank Zakat El-Zawa." *El Muhasaba: Jurnal Akuntansi (E-Journal)* 5.2 (2014): 158-180.
- Hartanto, S. (2017). Implementasi Fuzzy Rule Based System Untuk Klasifikasi Buah Mangga. *Techsi-Jurnal Teknik Informatika*, 9(2), 103-122.
- Harumy, T. H. F., & Sulistianingsih, I. (2016). Sistem Penunjang Keputusan Penentuan Jabatan Manager Menggunakan Metode Mfep Pada Cv. Sapo Durin. In *Seminar Nasional Teknologi Informasi Dan Multimedia* (Pp. 6-7).
- Herdianto, H. (2018). Perancangan Smart Home Dengan Konsep Internet Of Things (Iot) Berbasis Smartphone. *Jurnal Ilmiah Core It: Community Research Information Technology*, 6(2). <https://doi.org/10.4028/www.scientific.net/AMR.756-759.138>
- Huda, A. A. (2011). *24JAM!! Pintar Pemrograman Android #1* (Vol. 1).
- Ilhami, M. (2017). Pengenalan Google Firebase Untuk Hybrid Mobile Apps Berbasis Cordova. *Jurnal IT CIDA*, 3(1), 16–29.

- Juansyah, A. (2015). Pembangunan Aplikasi Child Tracker Berbasis Assisted – Global Positioning System (A-GPS) Dengan Platform Android. *Jurnal Ilmiah Komputer Dan Informatika (KOMPUTA)*, 1(1), 1–8. Retrieved from elib.unikom.ac.id/download.php?id=300375
- Khairul, K., Haryati, S., & Yusman, Y. (2018). Aplikasi Kamus Bahasa Jawa Indonesia Dengan Algoritma Raita Berbasis Android. *Jurnal Teknologi Informasi Dan Pendidikan*, 11(1), 1-6.
- Maiyana, E. (2018). *Pemanfaatan Android Dalam Perancangan Aplikasi Kumpulan Doa. Jurnal Sains Dan Informatika*, 4(1), 54–65.
<https://doi.org/10.22216/jsi.v4i1.3409>
- Marlina, L., Muslim, M., Siahaan, A. U., & Utama, P. (2016). Data Mining Classification Comparison (Naïve Bayes And C4. 5 Algorithms). *Int. J. Eng. Trends Technol*, 38(7), 380-383.
- Muttaqin, Muhammad. "Analisa Pemanfaatan Sistem Informasi E-Office Pada Universitas Pembangunan Panca Budi Medan Dengan Menggunakan Metode Utaut." *Jurnal Teknik Dan Informatika* 5.1 (2018): 40-43.
- Muttaqin, Muhammad. "Portal Academic Portal Innovation Based On Website In The Era Of Digital 4.0 Technology Now."
- Nicky, K. H. & S. P. S. (2016). *Culture is a way of life that developed and shared by a group of people , and inherited from one technology as a competitive sector that can added value to the business processes that run . The development of information and communication technology make.* 9(1), 9–18.
- Perwitasari, I. D. (2018). Teknik Marker Based Tracking Augmented Reality Untuk Visualisasi Anatomi Organ Tubuh Manusia Berbasis Android. *Intecom: Journal Of Information Technology And Computer Science*, 1(1), 8-18.
- Poerwanta, R. (2013). Perancangan Sistem Inventory Spare Parts Mobil Pada Cv . Prayitno, A., & Safitri, Y. (2015). Pemanfaatan Sistem Infomasi Perpustakaan Pt, P., & Rent, A. P. M. (2018). *Jurnal Usecase*. 2(2), 64–77.
- Putri, R. E., & Siahaan, A. (2017). Examination Of Document Similarity Using Rabin-Karp Algorithm. *International Journal Of Recent Trends In Engineering & Research*, 3(8), 196-201.

- Ramadhani, S., Suherman, S., Melvasari, M., & Herdianto, H. (2018). Perancangan Teks Berjalan Online Sebagai Media Informasi Nelayan. *Jurnal Ilmiah Core It: Community Research Information Technology*, 6(2).
- Rizal, Chairul. "Pengaruh Varietas Dan Pupuk Petroganik Terhadap Pertumbuhan, Produksi Dan Viabilitas Benih Jagung (*Zea Mays L.*)" Etd Unsyiah (2013). *Software Engineering*, 1(1), 1–10.
- Sunaryo, S., Handojo, A., & Andjarwirawan, J. (2013). Pembuatan Aplikasi Wisata Sejarah Pertempuran Surabaya 1945 Berbasis Android. *International*, 1, 1. [https://doi.org/10.1016/0025-5408\(81\)90013-1](https://doi.org/10.1016/0025-5408(81)90013-1)
- Tangerang, D. I. K., Ramadhan, T., Utomo, V., Polanda, D., Watrianthos, R., Purnama, I., ... Fisika, P. P. (2018). Aplikasi Komputer dan Smartphone Berbasis Android untuk Menangani Reservasi Hotel pada Citi Smart Hotel - BSD. *Jurnal Pendidikan Islam*, 5(2), 2. <https://doi.org/10.31294/jtk.v4i2.3597>
- Umar. (2015). *Pengembangan Perangkat Lunak Distribusi Gnu / Linux*. 6(1), 11–16.