

BAB II

TINJAUAN PUSTAKA

2.1 Teori Dasar

Berdasarkan penelitian yang dilakukan oleh peneliti, diperoleh beberapa teori dasar yang berkaitan dengan penelitian yang dapat membantu peneliti mengembangkan dalam penulisan dan menyelesaikan penelitian yang dilakukan ini, Beberapa teori dasar tersebut ialah sebagai berikut:

2.1.1 Pengertian Android

Android merupakan salah satu perangkat lunak open source berbasis *Linux* yang dikembangkan untuk perangkat mobile *Smartphone* yang terdiri dari Sistem Operasi, *Middleware*, dan Aplikasi kunci. Android merupakan perangkat lunak mobile *open source* yang menyediakan platform terbuka ditujukan kepada para pengembang untuk dapat mengembangkan dan menciptakan aplikasi mereka sendiri. (Murtiwyati and Lauren 2013).

Sistem operasi Android sampai saat ini telah berjalan jauh sejak sistem operasi Android ini pertama kali diperkenalkan oleh Open Handset Alliance pada akhir tahun 2007. Android adalah sistem operasi terbuka atau *open source* yang telah di populerkan oleh perusahaan teknologi Google, Dari awal diluncurkan hingga saat ini, banyak sekali perangkat nirkabel yang telah menggunakan sistem operasi Android ini, contohnya adalah tablet, *net-book*, *set-top box*, *smartphone*, dan juga *Smart TV*. Pada Tahun 2000 Perusahaan Google telah mengakuisisi perusahaan yang bernama Android (Simanjuntak and Alfisyahri 2020).

2.1.2 Perkembangan Android

Dari awal perilisan perdana Sistem Operasi Android pada 5 November 2007 hingga saat ini, Sistem Operasi Android telah mengalami banyak sekali perkembangan setiap tahunnya, perkembangan ini bertujuan untuk memperbaiki kekurangan dan meningkatkan kinerja serta penambahan fitur – fitur terbaru yang bermanfaat, Berikut ini adalah ringkasan perkembangan Sistem Android yang disusun dalam bentuk Tabel (Kusniyati and Pangondian Sitanggang 2016).

Tabel 2.1 Perkembangan sistem operasi Android

Versi	Kode Nama	Tahun Rilis
Android 1.0	Astro Boy	2008
Android 1.1	Bender	2008
Android 1.5	Cupcake	2009
Android 1.6	Donut	2009
Android 2.0 & 2.1	Eclair	2009
Android 2.2	Frozen Yoghurt	2010
Android 2.3	Gingerbread	2010
Android 3.0 & 3.2	Honeycomb	2011
Android 4.0	Ice Cream Sandwich	2011
Android 4.1	Jelly Bean	2012
Android 4.4	KitKat	2013
Android 5.0 & 5.1	Lollipop	2014
Android 6.0	Marshmallow	2015

Android 7.0 & 7.1	Nougat	2016
Android 8.0 & 8.1	Oreo	2017
Android 9.0	Pie	2018
Android 10	Quince Tart	2019
Android 11	Red Velvet Cake	2020
Android 12	Snow Cone	2021

Sumber: (*Penelitian Kusniyati dan Pangondian Sitanggang, 2016*)

2.1.3 Struktur Aplikasi Android

Menurut (Ceryna Dewi et al. 2018) Sistem Operasi Android memiliki struktur atau fundamental aplikasi yang ditulis dengan bahasa pemrograman Java. Kode java ini dikompilasi bersamaan dengan *resource file* yang dibutuhkan oleh sebuah aplikasi. Proses ini di package oleh *tools* yang dinamakan *apttools* ke dalam sebuah paket Android. Kemudian dari proses ini nantinya akan menghasilkan sebuah *file* dengan *ekstensi* apk. File apk ini merupakan sebuah *file* yang nantinya akan di *instal* di Sistem Operasi Android yang kemudian akan berjalan sebagai sebuah aplikasi pada perangkat mobile. Dalam sebuah aplikasi Android, terdapat 4 komponen yaitu:

1. *Activities*

Merupakan komponen yang berfungsi untuk menyajikan tampilan pada layar dari sebuah aplikasi untuk pengguna (*user interface*)

2. *Service*

Merupakan komponen yang tidak memiliki tampilan pada layar (*user interface*), namun komponen ini berjalan ketika aplikasi sedang digunakan, komponen ini berjalan di *backgrounds*.

3. *Broadcast Receiver*

Merupakan komponen yang berfungsi untuk menerima dan bereaksi untuk mengirimkan sebuah notifikasi ke layar perangkat mobile dari sebuah aplikasi yang sedang dijalankan oleh pengguna.

4. *Content Provider*

Merupakan komponen yang memiliki fungsi untuk mengumpulkan aplikasi data dari aplikasi lainnya secara spesifik, sehingga dapat digunakan untuk aplikasi lainnya.

2.1.4 Software Yang Diperlukan Untuk Membangun Aplikasi Android

Menurut Ceryana Dewi (Ceryana Dewi et al. 2018) untuk membangun sebuah sistem operasi Android dapat dilakukan pada beberapa perangkat lain seperti Komputer berbasis sistem operasi Mac, Windows, dan Linux. Dan dalam proses pembangunannya dibutuhkan beberapa *tools* yang dibutuhkan sebagai syarat untuk membangunnya, *tools* ini gratis dan bisa di *download* melalui *website*. Berikut ini adalah beberapa tools yang dibutuhkan untuk membangun aplikasi Android.

1. JDK (*Java Development Kit*)
2. Android SDK
3. ADT (*Android Development Tools*)

2.2 Teori Khusus

Pada teori khusus ini peneliti akan menjelaskan mengenai teori yang berkaitan dengan penelitian, Seperti menjelaskan mengenai *software* pendukung yang digunakan. Mulai dari teori mengenai metode penelitian yang digunakan, UML, bahasa pemrograman yang akan digunakan, dan semua *software* yang digunakan.

2.2.1 Chatbot

Menurut (Suryani and Amalia 2017), *Chatbot* merupakan sebuah sistem pemrograman yang cerdas yang di hasilkan dari pemrosesan Bahasa Alami atau Natural Language Processing (NLP) dimana ini merupakan salah satu cabang dari Kecerdasan Buatan atau Artificial Intellegence (AI).

Chatbot merupakan salah satu teknologi *Artificial intelegence* yang berbasis audio dan teks yang di program untuk menyimulasikan percakapan antara program dengan manusia layaknya percakapan manusia dengan manusia. *Chatbot* akan menyimulasikan percakapan dan memberi respon kepada pengguna berdasarkan kata kunci dengan pola yang mirip dari basis tekstual yang telah di program pada aplikasi *Chatbot* (E-commerce and Oktavia 2019)

Komponen *Chatbot* terdiri dari dua yaitu program *Bot* dan *Brain File*. Program *Bot* memiliki fungsi untuk mengakses *input* yang di berikan oleh pengguna kemudian *input* dari pengguna tersebut akan di analisa oleh program kemudian akan di terukan ke *Brain File* yang akan menghasilkan respon yang sesuai kepada pengguna. Komponen program *Bot* terdiri dari dari beberapa bagian

yaitu *scanner* dan *parser*, sedangkan *brain file* terdiri dari kosakata, kepribadian, dan pengetahuan (*Knowledge*), semakin banyak pengetahuan yang disimpan di *brain file* maka akan semakin besar ukuran *brain file* tersebut.(E-commerce and Oktavia 2019)

Menurut (Wijaya and Pebriantara 2018), *Chatbot* merupakan sebuah program komputer yang dirancang untuk melakukan percakapan antara komputer dengan manusia dengan menggunakan metode tekstual. Menurut program dibangun agar dapat meyakinkan atau menyimulasikan bagaimana manusia sebagai lawan bicara dapat berperan sebagai mitra percakapan, Cara kerja sebenarnya adalah *bot* menerima *input* dari pengguna, kemudian *bot* akan memindai kata kunci yang sesuai dengan *input* dari pengguna sebelumnya, kemudian *bot* akan membalas *input* dari pengguna sebelumnya dengan kata kunci yang paling cocok atau kata kunci yang mirip dari basis data tekstual

Saat ini sistem *Chatbot* sudah banyak di implementasikan di banyak jejaring sosial besar untuk memudahkan penggunaannya dalam menanggapi pertanyaan dari pengguna, seperti pada jejaring sosial Twitter, Windows Live Messenger, dan juga pada portal online yang populer seperti eBay yang merupakan web lelang daring dan PayPal yang merupakan aplikasi pembayaran secara daring (Cahya and Triayudi 2021)

2.2.2 Software Pendukung

Pada penelitian ini peneliti menggunakan beberapa *software* pendukung yang digunakan untuk merancang dan membangun aplikasi *Chatbot* mobile

berbasis Android. Adapun *software* pendukung tersebut diantaranya adalah sebagai berikut:

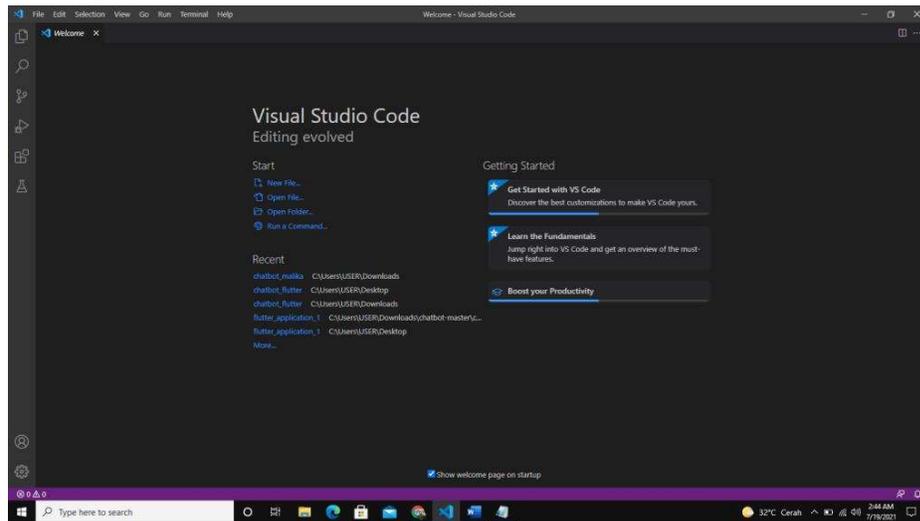
3.2.2.1 Visual Studio Code

Visual Studio Code merupakan *Integrated Development Environment* (IDE) atau kode editor yang di dibuat oleh sebuah perusahaan teknologi terbesar dibidang pengembangan komputer bernama *Microsoft Corporation*. *Visual Studio Code* ini sudah tidak asing lagi dan banyak digunakan oleh banyak pengembang untuk mengembangkan aplikasi dalam bentuk *native code* (bahasa mesin yang berjalan di atas windows), ataupun *managed code* (dalam bentuk *Microsoft Intermediate Language* di atas *.NET Framework*). Selain itu *Visual Studio Code* ini juga digunakan untuk mengembangkan aplikasi *Windows Mobile* dan *Silverlight* yang berjalan pada *.NET Compact Framework*.

Visual Studio Code sudah mendukung beberapa bahasa pemrograman yang berbeda. Adapun beberapa bahasa pemrograman tersebut adalah bahasa pemrograman *C++*, *Visual C#*, *Visual Basic*, *M*, *Phyton*, dan juga *Ruby*. Bahasa pemrograman ini terdapat dalam *pack extra*, dimana *pack extra* ini terpisah dari *Visual Studio Code* (Putri and Azpar 2016).

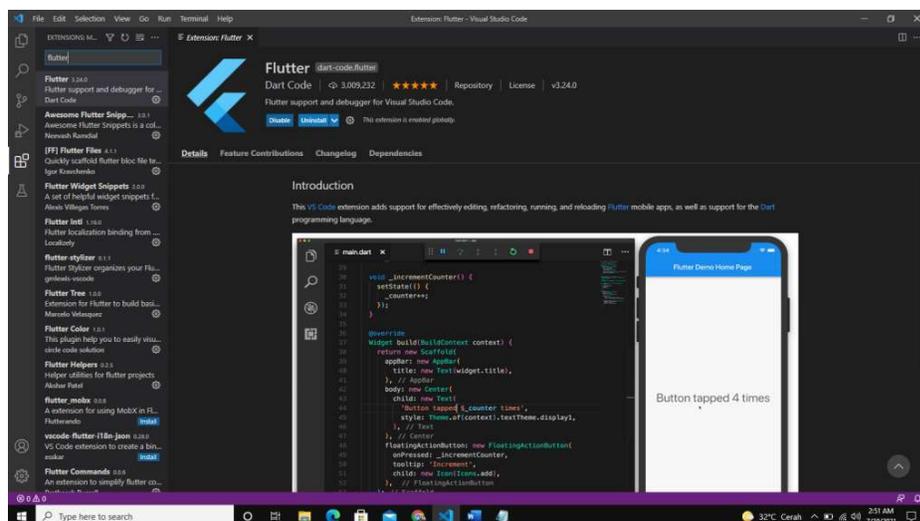
Berikut ini adalah langkah – langkah untuk membuat aplikasi android menggunakan visual studio code:

1. Menjalankan *Software Visual Studio Code* yang telah di instal pada komputer, berikut merupakan tampilan awal dari *Visual Studio Code*:



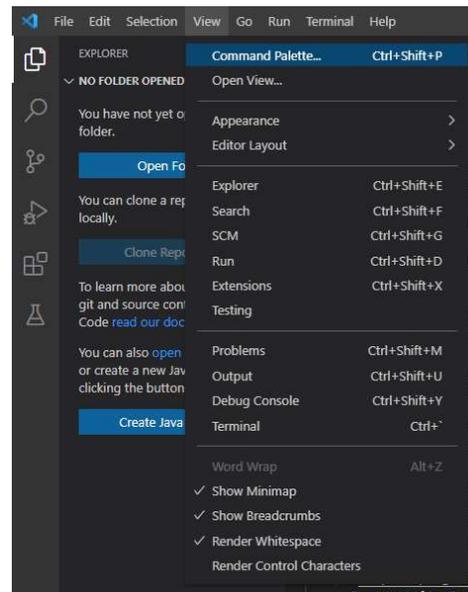
Gambar 2.1 Tampilan awal *Visual Studio Code*
Sumber: (Data peneliti, 2021)

2. Langkah kedua adalah menginstal ekstensi flutter (Peneliti menggunakan framework *flutter* untuk merancang aplikasi), pada *Visual Studio Code*, pilih menu *Extension* yang terletak pada bagian kiri, Pada kolom pencarian ketikkan *Flutter* dan *enter*, Pilih Flutter 3.24.0 kemudian pilih *Instal*



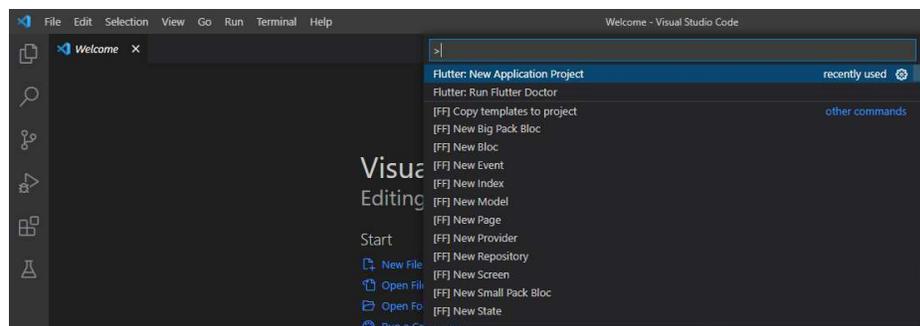
Gambar 2.2 Menginstal ekstensi flutter pada *Visual Studio Code*
Sumber: (Data Peneliti, 2021)

- Langkah ketiga ialah membuat project Flutter baru dengan *Visual Studio Code*, Pada menu *view* yang terletak pada bagian atas, Pilih *Command Palette*



Gambar 2.3 Menu *Command Palette* pada Visual Studio Code
Sumber: (Data peneliti, 2021)

- Pilih *Flutter: New Application Project*, Lalu akan muncul pilih lokasi penyimpanan untuk menyimpan *project*, lalu membuat nama *project*



Gambar 2.4 Menjalankan menu Flutter: New Application Project
Sumber: (Data Peneliti, 2021)

3.2.2.1 Android Studio

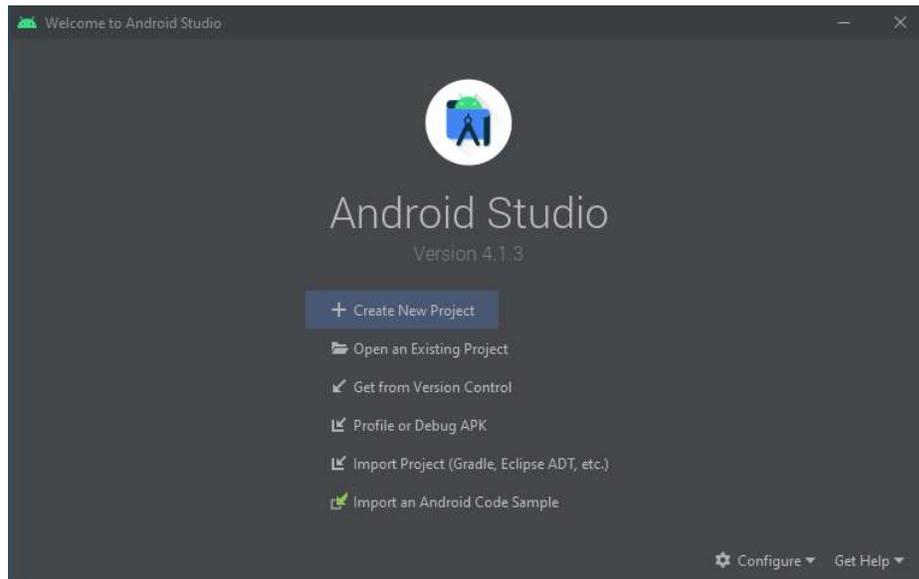
Menurut (Juansyah 2015), Android Studio merupakan *Integrated Development Environment* (IDE) yang resmi untuk mengembangkan aplikasi berbasis Android, Android Studio bersifat *open source* atau gratis dan mudah diunduh melalui *website*. Android studio diluncurkan pada 16 Mei 2013, dan diumumkan oleh Google pada *event Google I/O Conference*

Android Studio dikembangkan berdasarkan *IntelliJ IDEA* yang mirip dengan *Eclipse* yang disertai dengan *Plugin ADT (Android Development Tools)*. Adapun fitur – fitur yang dimiliki oleh Android Studio adalah sebagai berikut:

1. Project yang digunakan berbasis *Gradle Build*.
2. Proses *Refactory* dan pembenahan bug dilakukan dengan cepat
3. Terdapat *tools* baru yaitu "*Lint*" yang dapat memonitor kecepatan, kegunaan, dan kompetibilitas aplikasi dengan cepat.
4. Dapat menggunakan Proguard & App-Signing sebagai keamanan.
5. Memiliki GUI (*Graphical User interface*) yang lebih mudah.
6. Sudah mendukung *Google Cloud Platform*.

Android Studio dalam penelitian ini digunakan untuk menjalankan emulator perangkat android, Emulator ini digunakan untuk menguji *code* yang sedang di bangun, berikut adalah cara untuk menjalankan emulator *Android Studio*:

1. Membuka *Software Android Studio* yang telah diinstal pada perangkat komputer



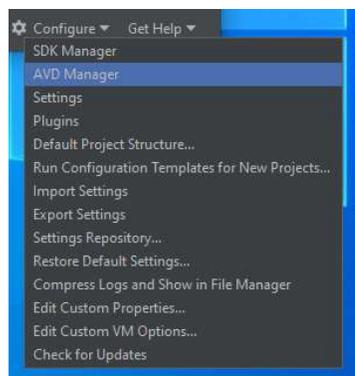
Gambar 2.5 Tampilan awal Android Studio
Sumber: (Data peneliti, 2021)

2. Pilih menu *Configure* yang terletak pada bagian bawah kanan



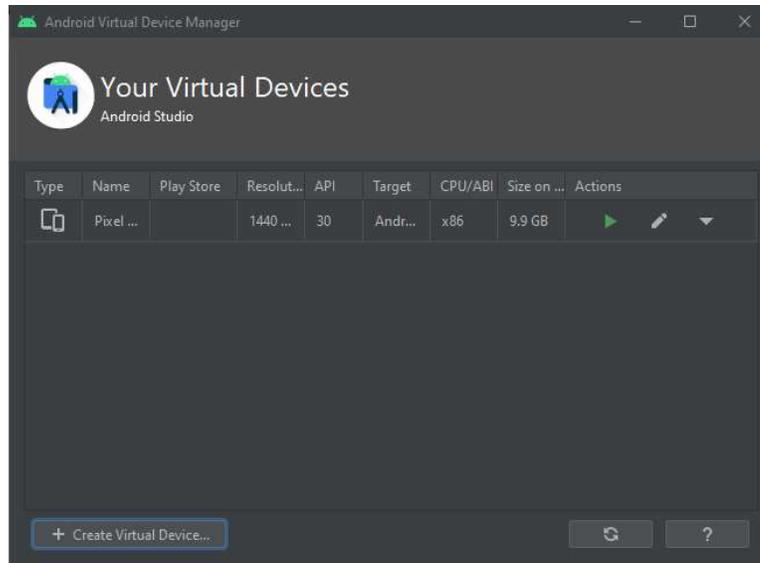
Gambar 2.6 Menu Configure
Sumber: (Data peneliti, 2021)

3. kemudian muncul beberapa pilihan, pilih *AVD Manager*



Gambar 2.7 Menu *AVD Manager*
Sumber: (Data peneliti, 2021)

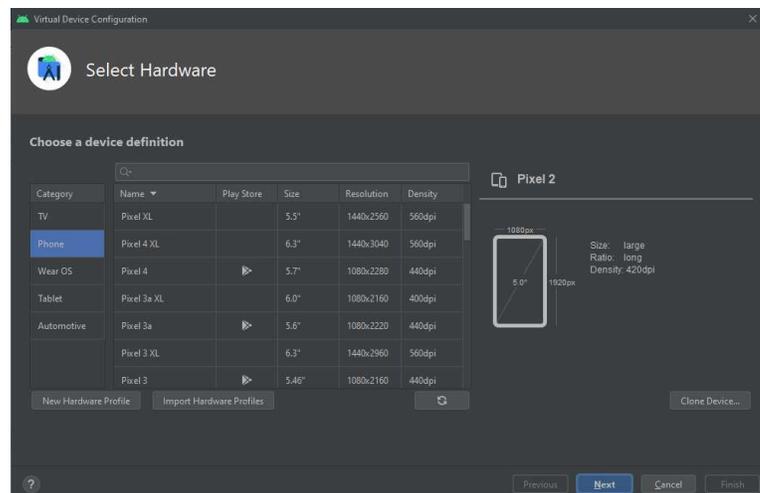
5. Pada tampilan *Your Virtual Devices* pilih menu *Create Virtual Device*



Gambar 2.8 Tampilan Your Virtual Devices

Sumber: (Data peneliti, 2021)

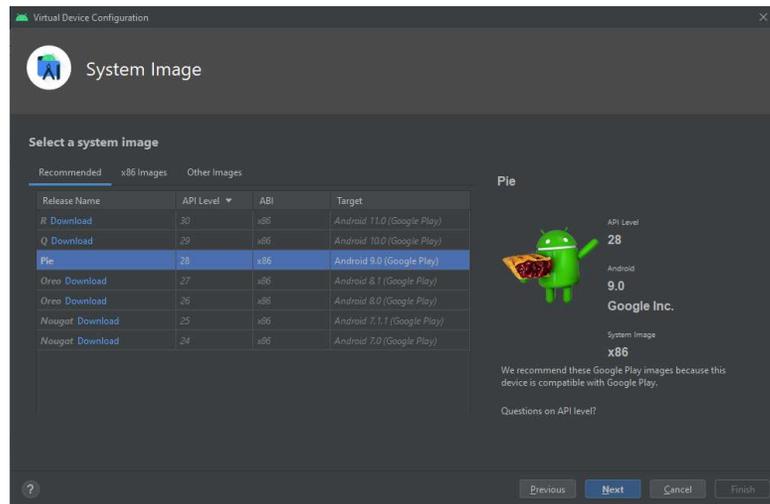
6. Kemudian pada tampilan *Select Hardware* pilih kategori *Phone*, lalu pilih Pixel 4 XL, kemudian pilih menu *Next*



Gambar 2.9 Memilih perangkat virtual Pixel 4 XL

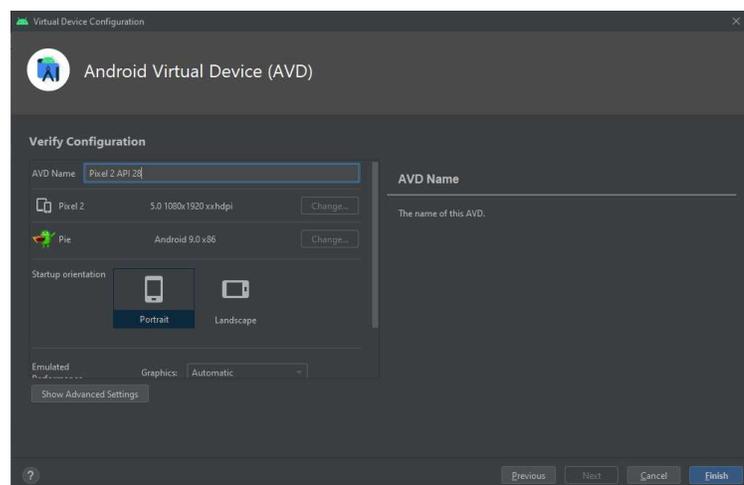
Sumber: (Data peneliti, 2021)

7. Kemudian pada tampilan *Select Image* Pilih pada kolom *Release Name Pie*, *API Level 28*, *ABI x86*, dan target *Android 9.0 (Google Play)*, Kemudian Pilih *Next*



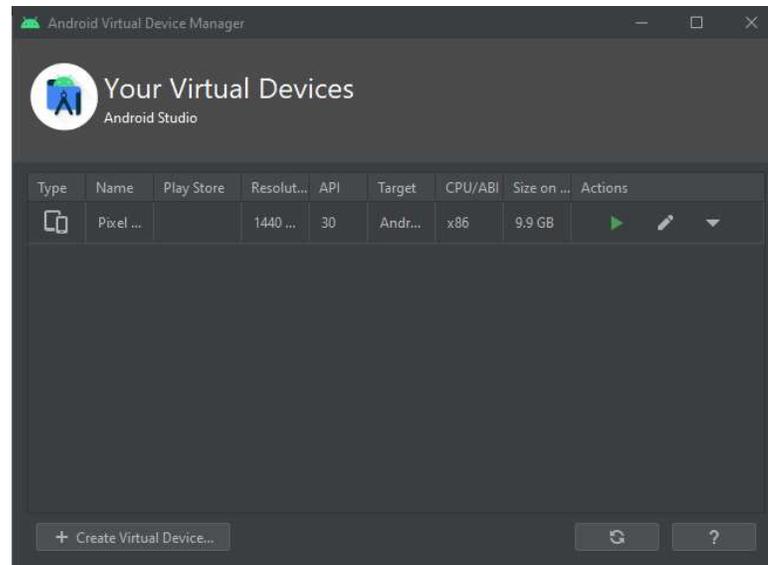
Gambar 2.10 Tampilan *System Image*, memilih sistem virtual
Sumber: (Data peneliti, 2021)

8. Kemudian pada tampilan *Android Virtual Device*, *Verify Configuration*, bisa langsung memilih tombol finish



Gambar 2.11 Tampilan *Android Virtual Device*, *Verify Configuration*
Sumber: (Data peneliti, 2021)

9. Kemudian pada tampilan *Your Virtual Device*, akan tampil perangkat virtual yang telah dibuat sebelumnya, untuk menjalankan perangkat virtual tersebut bisa dengan menekan tombol *play* pada kolom *action*



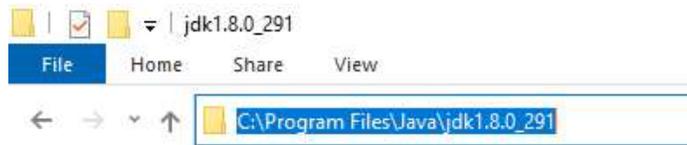
Gambar 2.12 Tampilan Your Virtual Devices, Pilih tomo *play*
Sumber: (Data peneliti, 2021)

3.2.2.1 Java Development Kit (JDK)

Menurut (Juansyah 2015), *Java Development Kit* (JDK) merupakan sekumpulan dari beberapa perangkat lunak yang berfungsi untuk dapat mengembangkan aplikasi atau perangkat lunak yang berbasis Java.

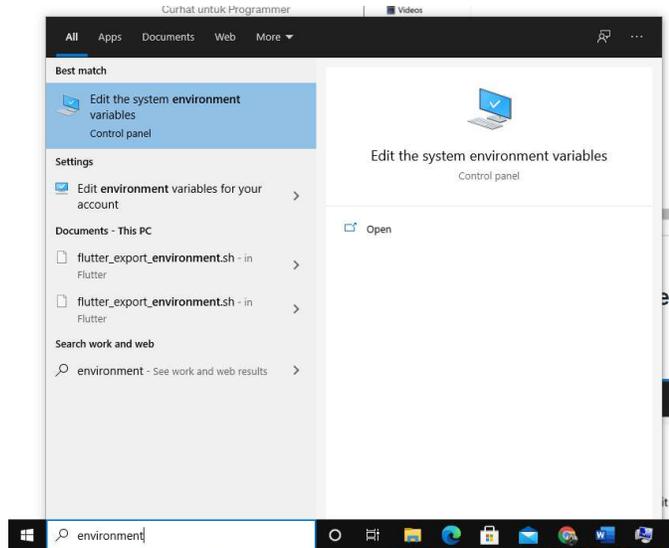
Java Development Kit (JDK) perlu diinstal di perangkat komputer dan mengatur *path* JDK pada sistem *Windows*, agar *project* perancangan aplikasi android yang berhubungan dengan java dapat berjalan dengan baik ketika melakukan *coding* dan menjalankan aplikasi. Berikut cara melakukan *path* JDK pada perangkat komputer:

1. Langkah pertama adalah menyalin *link folder* lokasi penginstalan JDK, yaitu C:\Program Files\Java\jdk1.8.0_291



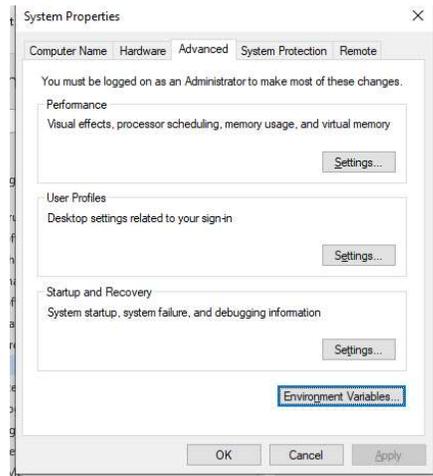
Gambar 2.13 Link folder lokasi penginstalan JDK
Sumber: (Data Peneliti, 2021)

2. Buka windows search bar kemudian ketikkan *environment*, kemudian pilih *Edit the system environment variables*



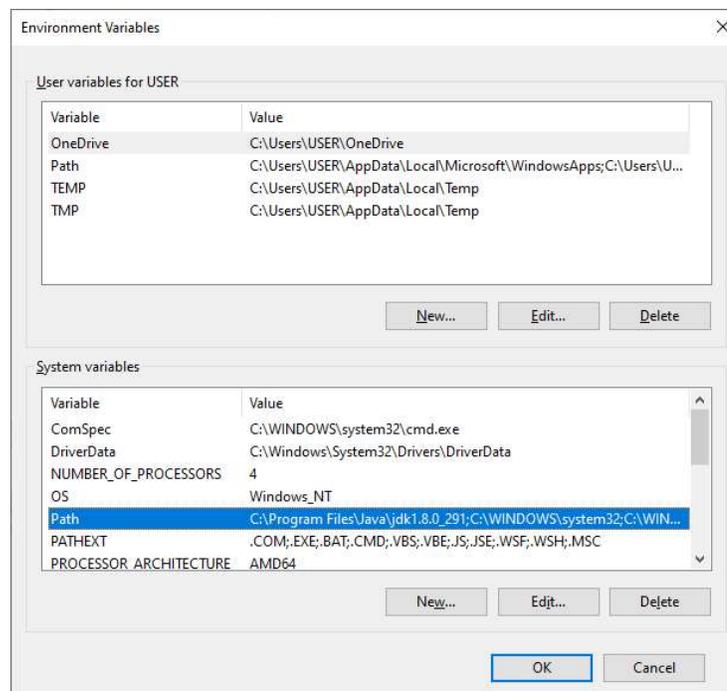
Gambar 2.14 Buka Edit system environment variables
Sumber: (Data peneliti, 2021)

3. Kemudian akan muncul *system properties*, setelah itu pilih menu *advanced* kemudian pilih menu *Environment Variables* untuk mengatur path



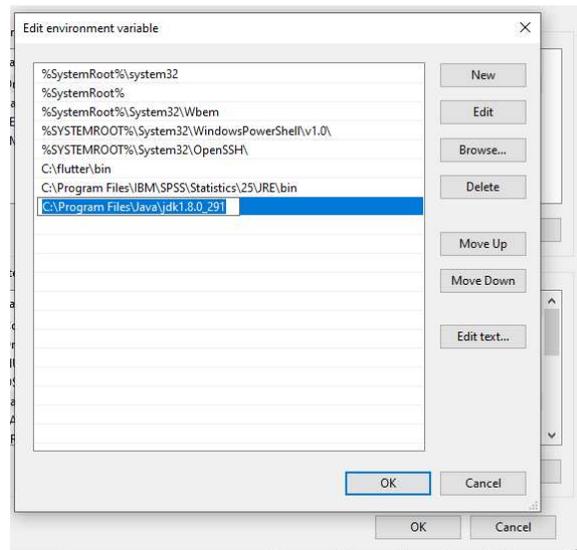
Gambar 2.15 System Properties
Sumber: (Data peneliti, 2021)

4. Langkah selanjutnya yaitu pada kolom *System variables*, pilih variabel *path* kemudian klik tombol *edit*



Gambar 2.16 Pilih variabel *path*
Sumber: (Data Peneliti, 2021)

5. Kemudian pilih tombol baru dan masukan salinan link folder lokasi penginstalan JDK yang sudah disalin sebelumnya, lalu pilih tombol ok



Gambar 2.17 Buat *Environment Variable* baru
Sumber: (Data peneliti, 2021)

3.2.2.1 Flutter

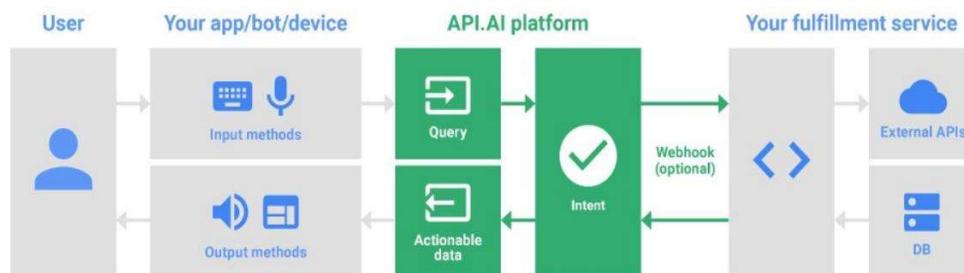
Menurut (Enggar Krisnada and Tanone 2020), Flutter merupakan *Software Development Kit (SDK)* atau sebuah *framework open source* yang dikembangkan oleh perusahaan Google yang digunakan untuk membuat dan mengembangkan aplikasi yang dapat berjalan di perangkat mobile berbasis Android dan iOS. Dalam proses pengkodean, flutter menggunakan bahasa pemrograman Dart.

Dalam pemrosesan pembuatan dan pengembangan pada framework ini semua kodenya di *compile* dalam kode native-nya (Android NDK, LLVM, AOT-Compiled) tanpa ada interpreter pada prosesnya, karena hal ini lah proses *compile* dalam *framework* inimenjadi lebih cepat.

3.2.2.1 *Dialogflow*

Menurut (E-commerce and Oktavia 2019), *Dialogflow* merupakan sebuah platform yang dikembangkan oleh *Google*, Platform ini menyediakan layanan *Natural Language Processing (NLP)* dan *Natural Language Understanding (NLU)*. Layanan *dialogflow* ini berfokus pada interaksi manusia dengan komputer dengan menggunakan bahasa alami dan analisa semantik.

Natural Language Processing (NLP) berfungsi untuk memproses pengetahuan secara alami dari sebuah kalimat, susunan, dan arti dari kata – kata dalam kalimat tersebut, sedangkan *Natural Language Understanding (NLU)* berfungsi untuk memproses pengetahuan berdasarkan analisa semantik dari suatu kalimat.



Gambar 2.18 Alur Proses *Dialogflow*
Sumber: (Diunduh Dari *Google* Gambar)

Alur proses *Dialogflow* diawali saat pengguna mengirim permintaan dalam bentuk teks ke dalam platform *Dialogflow*, kemudian permintaan tadi akan di proses dalam intents untuk memetakan permintaan dan tindakan yang akan diberikan sebagai respon dari sistem kepada pengguna tadi. Kemudian jika terdapat informasi tambahan yang akan diberikan sebagai sebuah respon,

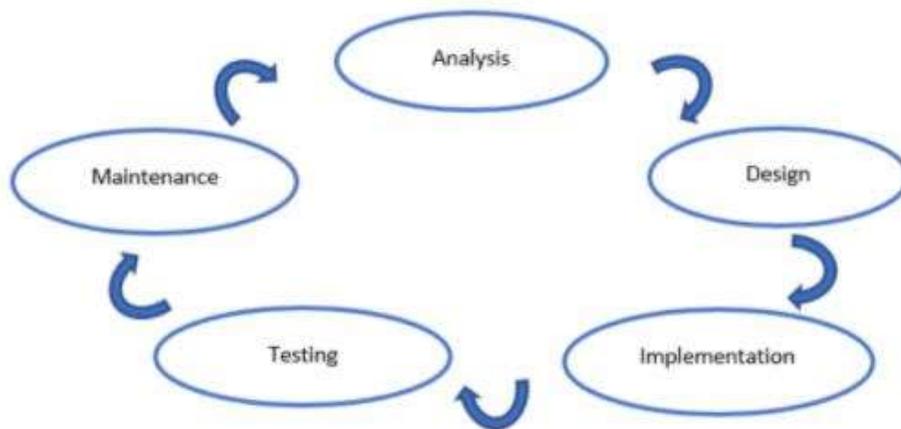
Dialogflow akan mengirim *fulfillment*, ini akan mendapatkan informasi dari sumber luar kemudian pengguna akan mendapatkan jawaban dari pertanyaan yang telah diajukan.

3.2.2.1 *Google Cloud*

Google Cloud merupakan pusat dari data yang dimiliki oleh klien atau pengguna, dalam artian data yang dimiliki oleh pengguna tersimpan di komputasi awan bukan di perangkat komputer miliknya. Tujuan dari *Google* adalah untuk memanfaatkan sumber daya komputer miliknya yang sangat banyak yang tersebar di seluruh dunia dan juga untuk mempercepat proses – proses kinerja pada web yang sedang beroperasi. Tidak semua aplikasi dapat berjalan dengan baik pada sistem yang membentuk jaringan komputasi awan milik google tersebut, Aplikasi yang dapat berjalan dengan baik adalah aplikasi yang telah mengikuti aturan-aturan yang ditetapkan oleh pihak Google. Adapun Aplikasi tersebut dinamakan dengan *Google App Engine (GAE)* (Hakim 2016).

2.2.3 Metode Penelitian

Dalam proses penelitian untuk membangun sebuah aplikasi, diperlukan adanya metode penelitian yang berfungsi sebagai struktur kegiatan penelitian menjadi lebih terarah, adapun peneliti menerapkan metode penelitian *Waterfall* (SDLC). Metode *Waterfall* (SDLC) ini merupakan metode penelitian perancangan aplikasi yang sudah memiliki kerangka yang berisi tahapan – tahapan yang akan dilakukan untuk mengembangkan perangkat lunak.



Gambar 2.19 Tahapan Metode SDLC

Sumber: (Penelitian Nilam Cahya & Agung Triayudi, 2021)

Berikut ini adalah tahapan – tahapan dalam melakukan metode Waterfall SDLC dalam sebuah penelitian:

3.2.2.1 Analisis

Pada tahap ini peneliti melakukan analisis terhadap struktur dan alur pada salah satu website layanan wisata Kota Bali, kemudian peneliti mencoba untuk mengembangkan menjadi sebuah aplikasi mobile dan wisata di Kota Batam. Peneliti melakukan Pengumpulan data untuk pengembangan aplikasi mobile *Chatbot* ini dengan beberapa cara sebagai berikut:

1. Melakukan studi pustaka dengan mengumpulkan studi kasus yang serupa.
2. Melakukan observasi langsung dengan salah satu aplikasi mobile *Chatbot* informasi wisata yang telah diunduh melalui playstore.
3. Melakukan wawancara dengan salah satu pemilik layanan wisata di kota Batam yaitu Galang Bahari.

Setelah peneliti berhasil mengumpulkan data – data terkait yang diperlukan, selanjutnya adalah mnegolah data, dan mengklasifikasi jenis data wisata dan jenis data kuliner, Setelah data sudah di klasifikasi mafa data akan dirapikan dan di formatkan kembali untuk di olah selanjutnya.

3.2.2.1 Desain

Desain adalah tahap dimana peneliti membuat proses dan tampilan fitur aplikasi yang perlu ditampilkan berdasarkan analisis dan pengumpulan data pada tahap sebelumnya. Pada tahap ini peneliti melakukan rancangan dan pemodel sistem *Chatbot* untuk pengguna, pada tahap ini akan menghasilkan desain yang nantinya akan di lakukan pengkodegan untuk proses pembuatan aplikasi mobile ini, proses pengkodegan akan merujuk pada desain yang telah di buat pada tahap ini oleh peneliti.

3.2.2.1 Implementasi

Pada tahap ini peneliti melakukan proses pengodingan sesuai dengan desain lyang telah di buat pada tahap sebelumnya, yaitu tahap desain. Pada tahap implementasi atau pengkodegan ini, penulis menggunakan *framework* Flutter sebagai kerangka dalam penulisan program, *Visual Studio Code* sebagai *text editor*, serta menggunakan *Android Studio* untuk menampilkan visual dari program yang sedang di tulis pada *text editor* dengan cara debug atau run program sehingga dapat mengetahui langsung hasil dari pengkodegan dan bagian - bagian yang masih kurang.

3.2.2.1 *Testing*

Pada tahap ini peneliti akan melanjutkan proses dari tahap pengkodean sebelumnya, hasil dari pengkodean sebelumnya akan di compile dan di *extract* menjadi sebuah file apk yang dapat di instal dan dijalankan pada perangkat mobile berbasis Android. Pada tahap ini peneliti akan mencoba hasil dari aplikasi yang telah di koding, apakah sudah sesuai dengan desain, apakah sudah sesuai dengan hasil yang di inginkan, dan apakah masih ada yang *error* atau hasil yang tidak sesuai dengan yang seharusnya. Proses testing ini perlu dilakukan sebelum mempublikasikan ke umum dan digunakan oleh banyak pengguna lainnya agar aplikasi benar – benar dapat digunakan dengan maksimal dan sesuai dengan fungsinya.

3.2.2.1 *Maintenance*

Pada tahap ini peneliti sudah memastikan bahwa aplikasi dapat berjalan dengan baik dan sudah sesuai dengan fungsinya serta peneliti sudah melakukan publikasi terhadap aplikasi yang dirancang, namun peneliti terus melakukan pemeliharaan terhadap aplikasi dan memerhatikan tanggapan dan masalah yang mungkin ditemukan oleh pengguna lainnya, sehingga peneliti dapat melakukan perbaikan terhadap kekurangan yang ditemukan dan melakukan pengembangan fitur yang mungkin dapat lebih membantu banyak pengguna lainnya.

2.2.4 UML

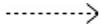
Menurut (M Teguh Prihandoyo 2018), *Unified Modeling Language* adalah sebuah metode yang digunakan peneliti untuk memodelkan secara visual mengenai rancangan atau pembuatan sebuah perangkat lunak yang

berorientasikan pada objek. *UML (Unified Modeling Language)* ini merupakan sebuah standar penulisan bagi peneliti dimana di dalam *UML* ini terdapat sebuah proses, penulisan kelas dalam bahasa yang spesifik terhadap rancangan perangkat lunak yang akan di bangun. Dalam *UML* ini terdapat beberapa diagram yang sering digunakan oleh peneliti untuk melakukan pembuatan atau pengembangan sebuah perangkat lunak atau aplikasi, Diantaranya adalah sebagai berikut:

2.2.5 Use Case Diagram

Menurut (Suendri 2018), *Use case diagram* merupakan gambaran mengenai tampilan luar dari sistem perangkat lunak atau aplikasi yang peneliti buat model nya. Model *use case* dijabarkan dalam bentuk diagram *use case*. diagram tidaklah identik dengan jenis model nya, karena model lebih luas dari diagram. *Use case* menggambarkan urutan aktor yang memberikan nilai terukur dari sistem perangkat lunak. Berikut ini merupakan tabel *use case* diagram:

Tabel 2.2 Simbol *Use Case Diagram*

No	Gambar	Nama	Keterangan
1		<i>Actor</i>	Himpunan peran pengguna ketika berinteraksi dengan use case
2		<i>Dependency</i>	Hubungan dimana perubahan pada elemen mandiri akan mempengaruhi elemen yang tidak bergantung pada elemen mandiri
3		<i>Generalization</i>	Hubungan dimana objek anak berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk
4		<i>Include</i>	Menspesifikasikan bahwa usecase sumber secara <i>eksplisif</i>

5		<i>Extend</i>	<i>Use case</i> target memperluas perilaku dari <i>use case</i> sumber pada suatu titik yang diberikan
6		<i>Association</i>	Menghubungkan objek satu dengan objek lainnya.
7		<i>System</i>	Paket yang menampilkan sistem secara terbatas
8		<i>Use Case</i>	Deskripsi dari urutan aksi – aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu aktor
9		<i>Collaboration</i>	Interaksi aturan dan elemen lain yang bekerja sama untuk menyediakan perilaku yang lebih besar dari jumlah dan elemennya.
10		<i>Note</i>	Elemen fisik yang eksis saat aplikasi dijalankan dan mencerminkan suatu sumber daya komputasi

Sumber: (Jurnal Srinurhaya, 2015)

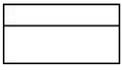
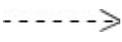
3.2.2.1 *Class Diagram*

Menurut (Suendri 2018), *Class Diagram* merupakan set objek yang mana memiliki atribut dan perilaku yang sama, Kelas ini kadang juga disebut dengan sebutan kelas objek, *Class Diagram* memiliki tiga area pokok yaitu sebagai berikut:

1. Nama, kelas harus memiliki sebuah nama
2. Atribut, merupakan kelengkapan dari sebuah kelas, Nilai yang terdapat pada suatu kelas hanya dapat di proses sebatas atribut yang dimiliki
3. Operasi, Merupakan sebuah proses yang dilakukan oleh sebuah kelas, baik proses untuk kelas itu sendiri ataupun untuk proses kelas lainnya.

Berikut ini merupakan simbol dari *Class Diagram*:

Tabel 2.3 Simbol *Class Diagram*

No	Gambar	Nama	Keterangan
1		<i>Generalization</i>	Hubungan dimana objek anak berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk
2		<i>Nary Association</i>	Upaya untuk menghindari asosiasi lebih dari dua objek
3		<i>Class</i>	Himpunan dari objek yang terdiri dari atribut dan operasi yang sama
4		<i>Collaboration</i>	Interaksi aturan dan elemen lain yang bekerja sama untuk menyediakan perilaku yang lebih besar dari jumlah dan elemennya.
5		<i>Realization</i>	Operasi yang dilakukan oleh suatu objek
6		<i>Dependency</i>	Hubungan dimana perubahan pada elemen mandiri akan mempengaruhi elemen yang tidak bergantung pada elemen mandiri
7		<i>Association</i>	Menghubungkan objek satu dengan objek lainnya.

Sumber: (*Jurnal Srinurhaya, 2015*)

3.2.1.1 *Activity Diagram*

Menurut (Suendri 2018), Yaitu diagram yang menunjukkan aktivitas sebuah sistem dalam bentuk aksi, Mulai dari aktivitas awal dimulainya masing – masing aksi dalam sebuah sistem, keputusan yang akan mungkin terjadi , hingga

proses berakhirnya aksi pada sistem. *Activity diagram* dapat menggambarkan suatu proses lebih dari satu aksi dalam waktu yang bersamaan.

Tabel 2.4 Simbol *Activity Diagram*

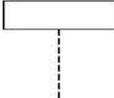
No	Gambar	Nama	Keterangan
1		<i>Activity</i>	Menampilkan interaksi satu sama lain dari setiap kelas antarmuka
2		<i>Action</i>	State dari sebuah sistem yang mengartikan eksekusi dari suatu aksi
3		<i>Initial Node</i>	Simbol yang mengartikan bagaimana sebuah objek dibentuk atau diawali
4		<i>Activity Final Node</i>	Simbol yang mengartikan bagaimana objek dibentuk dan dihancurkan
5		<i>Fork Node</i>	Suatu aliran yang pada tahap tertentu berubah jadi beberapa aliran

Sumber: (*Jurnal Srinurhaya, 2015*)

3.2.1.2 *Sequence Diagram*

Menurut (Suendri 2018), Merupakan gambaran tahap demi tahap dari sebuah sistem, yang diantaranya termasuk kronologi atau urutan nya, dan perubahan secara logis yang seharusnya yang dilakukan untuk menghasilkan sesuatu yang sesuai dengan *use case diagram*.

Tabel 2.5 Simbol *Sequence Diagram*

No	Gambar	Nama	Keterangan
1		<i>Life Line</i>	Objek <i>entity</i> , suatu antarmuka yang saling beraksi
2		<i>Message</i>	Suatu spesifikasi dari sebuah komunikasi antar objek yang memuat informasi sebuah aktifitas yang sedang terjadi
3		<i>Message</i>	Suatu spesifikasi dari sebuah komunikasi antar objek yang memuat informasi sebuah aktifitas yang sedang terjadi

Sumber: (*Jurnal Srinurhaya, 2015*)

2.2.6 Bahasa Pemrograman *Dart*

Bahasa pemrograman *Dart* merupakan suatu bahasa yang diciptakan oleh perusahaan teknologi Google, dimana Google menciptakan bahasa ini untuk menggantikan bahasa pemrograman Java, Bahasa pemrograman *dart* menggunakan *static typing* dimana sebelum menggunakan variabel, suatu variabel perlu di definisikan terlebih dahulu (Tjandra and Chandra 2020)

Dart dapat berjalan di banyak perangkat, seperti pada web, *Dart* menggunakan *dart2js* dimana *dart* akan diubah terlebih dahulu ke *Javascript* untuk dapat dibaca oleh *browser*. Kemudian pada perangkat *Desktop*, *Dart* menggunakan *dart2aot* yang merubah bahasa *dart* menjadi bahasa mesin sehingga dapat dibaca oleh *desktop*. Dan pada perangkat *mobile*, *Dart* menggunakan *Flutter* sebagai alat untuk merancang aplikasi agar dapat digunakan pada perangkat *mobile* (Tjandra and Chandra 2020)

Dart dapat melakukan pemrograman *asynchronous*, yang merupakan sebuah fungsi *non-blocking* yang dapat memungkinkan sebuah program dapat dijalankan oleh kode lain sambil menunggu fungsi *asynchronous* selesai. Contohnya ialah ketika sebuah aplikasi sedang mengambil informasi dari laman *web*, aplikasi juga dapat menampilkan halaman *loading*. *Dart* juga dapat melakukan pemrograman *reactive*, yaitu pemrograman *asynchronous* dengan *stream* data. (Tjandra and Chandra 2020)

2.2.7 Wisata

Kota Batam adalah salah satu kota yang terletak di antara perairan selat Malaka dan selat Singapura. Sektor pariwisata di kota Batam mulai dilirik pada tahun 2008. Sektor pariwisata di kota Batam di ambil alih oleh Dinas Pariwisata dan Kebudayaan Kota Batam. Pemerintah kota Batam mulai meningkatkan wisatawan dari mancanegara, upaya meningkatkan jumlah wisatawan dari mancanegara ini akan diikuti oleh pengeluaran dari wisatawan mancanegara selama berkunjung, hal ini akan meningkatkan penerimaan devisa non migas dan perkembangan sektor ekonomi terkait, sehingga hal ini dapat menumbuhkan dampak positif terhadap pertumbuhan ekonomi di kota Batam (Arisandi and Pradana 2018)

Tamu asing merupakan setiap pengunjung yang mengunjungi suatu negara selain negara nya untuk keperluan sesuatu yang didorong dari beberapa hal tanpa bermaksud untuk mendapatkan penghasilan dari negara yang dikunjungi. Pada tahun 2016 tercatat bahwa wisatawan mancanegara yang telah mengunjungi kota

Batam melalui bandar udara Hang Nadim telah mencapai 1.432.472 orang (Arisandi and Pradana 2018)

Terdapat banyak wisata di kota Batam. Karena kota Batam merupakan kota kepulauan maka kebanyakan wisata yang terdapat di kota ini banyak di dominasi oleh wisata bahari. Objek wisata bahari di kota Batam masih banyak di kelola mandiri oleh masyarakat. Berikut ini merupakan daftar objek wisata pantai yang terdata oleh Dinas Pariwisata Kota Batam hingga tahun 2015 (Arisandi and Pradana 2018)

Tabel 2.6 Objek Wisata Pantai

No	Objek Wisata Pantai	Lokasi
1	Tanjung Memban	Kel. Batu Besar, Kec. Nongsa
2	Kampung Melayu	Kel. Batu Besar, Kec. Nongsa
3	Nongsa	Kel. Sambu, Kec. Nongsa
4	Teluk Mata Ikan	Kel. Sambu, Kec. Nongsa
5	Batu Besar	Kel. Batu Besar, Kec. Nongsa
6	Melayu	Kel. Rempangcate, Kec. Galang
7	Melur	Kel. Sijantung, Kec. Galang
8	Pulau Abang	Kel. Pulau Abang, Kec. Galang
9	Pulau Hantu	Kel. Pulau Abang, Kec. Galang
10	Pulau Rano	Kel. Pulau Abang, Kec. Galang
11	Pulau Karas	Kel. Karas, Kec. Galang
12	Permata Subangmas	Kel. Subangmas, Kec. Galang

13	Mirota	Kec. Galang
14	Setokok	Kel. Setokok, Kec. Bulang
15	Melawa	Kel. Sekanakraya, Kec. Blk Padang
16	Tanjung Pinggir	Kel. Tanjungpinggir, Kec. Sekupang
17	Tanjung Datuk	Kel. Tanjungpinggir, Kec. Sekupang
18	Tanjung Buntung	Kel. Bengkonglaut, Kec. Bengkong
19	Pasir Putih	Kel. Kampung Bagan, Kec. Sei Beduk

Sumber: (D. Arisandi, M. Pradana, 2018)

Kebanyakan objek wisata yang terdapat pada label diatas masih dikelola oleh masyarakat melalui koordinasi Lembaga Pemberdayaan Masyarakat Desa (LPMD) sehingga mengakibatkan kurangnya promosi ke wisatawan mancanegara dan hanya banyak diketahui oleh wisatawan lokal saja, selain itu karena masih di kelola oleh masyarakat sehingga kurangnya sumber daya manusia untuk menangani dan melayani wisatawan asing mancanegara yang biasa menggunakan bahas asing atau bahasa Inggris (Arisandi and Pradana 2018)

2.2.8 Kuliner

Kota Batam juga memiliki banyak sekali kuliner dan tempat makan yang tersebar luas di kota, ada restaurant, *franchise*, dan tempat makan yang menyediakan kuliner khas makanan kota Batam, Makanan khas di kota ini adalah makanan yang berasal dari suku Melayu yang sudah lumayan terkenal di Indonesia (Erick Santana Perangin-Angin, Fendi Hidayat, ST. 2018)

Keberagaman jenis kuliner dan banyaknya tempat makan yang ada di kota Batam ini tentu saja dapat membuat pengunjung wisatawan lokal maupun wisatawan mancanegara akan mengalami sedikit kesulitan untuk menentukan jenis kuliner apa yang akan mereka cicipi dan dimana tempat makan yang akan mereka kunjungi. Maka dari itu peneliti akan melakukan perancangan aplikasi *Chatbot* mobile yang dapat diakses oleh banyak wisatawan (Erick Santana Perangin-Angin, Fendi Hidayat, ST. 2018)

Selain itu ada beberapa jenis kuliner yang sangat istimewa dan sudah banyak dikenal di kota Batam, yaitu jenis kuliner Seafood, untuk lokasinya sendiri sangat mudah ditemui, wisatawan hanya perlu pergi menuju ke pinggir pantai, karena disetiap pinggir pantai yang ada di kota Batam hampir semuanya tersedia restoran yang menyediakan menu makan Seafood, namun tidak sedikit juga menu seafood ini ditemukan di tengah kota, tepatnya di pusat kota Batam yaitu Batam Centre, Nagoya Hill, dan di di setiap mall – mall yang ada di kota Batam (Erick Santana Perangin-Angin, Fendi Hidayat, ST. 2018).

2.3 Penelitian Terdahulu

Tabel 2.2 Penelitian Terdahulu

No	Nama Author	Judul Penelitian	Masalah Yang Diteliti	Metode Yang Digunakan	Hasil Penelitian
1	Dheby s Suryan i & Eka	Aplikasi <i>Chatbot</i> Objek Wisata Jawa	Jawa Timur menjadi destinasi wisatawan, memiliki banyak tempat menarik,	Menggunakan AIML (Artificial Intellegence Markup	Aplikasi mobile yang dapat melakukan percakapan,

	Larasati Amalia	Timur Berbasis AIML (<i>Vol.3, No.2, Oktober 2017</i>)	namun tidak banyak diketahui oleh banyak orang karena kurangnya informasi.	Languange) yang di implementasikan pada aplikasi <i>Chatbot</i> untuk mendapatkan informasi tempat wisata.	memberikan jawaban dari pertanyaan pengguna mengenai lokasi wisata di Jawa Timur
2	I Nyoman Satria Paliwa het, I Made Sukarsa, I Ketut Gede Darma Putra	Pencarian Informasi Wisata Daerah Bali menggunakan an Teknologi <i>Chatbot</i> (<i>Vol.8, No.3, Desember 2017</i>)	Wisatawan harus mencari sendiri mengenai kebudayaan dan tempat wisata yang ada di bali biasanya melalui poster, surat kabar, majalah, dan lain-lain sehingga ini kurang efektif	Menggunakan AIML (Artificial Intellegence Markup Languange) untuk merancang aplikasi <i>Chatbot</i>	Aplikasi mobile <i>Chatbot</i> berbasis pesan text yang dapat memberi informasi kebudayaan dan wisata yang ada di Pulau Bali.
3	David Domarco & Ni Made Satvika Iswari	Rancang Bangun Aplikasi <i>Chatbot</i> Sebagai Media Pencarian Informasi	Banyak anime yang sudah beredar di internet, namun belum ada <i>Chatbot</i> yang di bangun untuk menangani	Penelitian ini menggunakan Brill Tagger, yaitu sebuah algoritma POS Tagging yang menggunakan metode induksi	aplikasi <i>Chatbot</i> untuk mendapatkan informasi seputar anime berbasis

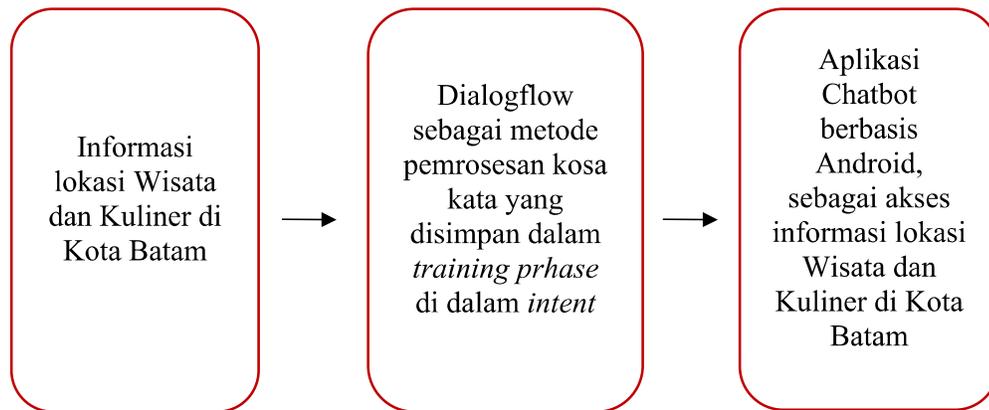
		Anime Menggunakan Regular Expression Pattern Matching <i>(Vol.9, No.1, Juni 2017)</i>	pencarian yang lebih spesifik dari seluruh anime yang ada	berbasis transformasi untuk memahami tata bahasa natural	teks menggunakan regular expression pattern matching
4	Elisabet Nila S. C. P & Irawan Afrianto	Rancang Bangun Aplikasi <i>Chatbot</i> Informasi Objek Wisata Kota Bandung dengan Pendekatan <i>Natural Language Processing</i> <i>(Vol.04, No.01, Maret 2015)</i>	Informasi objek wisata di Kota Bandung melalui website <i>www.bandungtourism.com</i> masih kurang jelas karena hanya melihat tampilan saja, dan data masih belum akurat dan masih terdapat banyak data yang kurang rapi	Dalam perancangan aplikasi, penulis menggunakan metode <i>Text Mining</i> sebagai media penalarannya	Dari hasil penelitian menghasilkan aplikasi <i>Chatbot</i> dengan tampilan yang bagus dan proses komunikasi yang lebih rinci dan terbaru.
5	Erick Wijaya	Rancangan Bangun	Siswa SMA IT Alia Tangerang	Dalam penelitiannya,	Aplikasi <i>Chatbot</i>

	& Peby Pebria ntara	Aplikasi Pembelajaran an dengan Memanfaat kan <i>Chatbot</i> API Dialogflow dan Moodle Berbasis Android Pada SMA IT ALIA Tangerang <i>(Vol.03,</i> <i>No.02,</i> <i>Desember</i> <i>2018)</i>	mengalami kesulitan untuk mendapatkan materi - materi dengan mudah dan cepat pada saat diluat jam sekolah, dan siswa mengalami kesulitan untuk memahami materi sekolah ketika mereka izin tidak masuk	penulis merancang aplikasi menggunakan metode <i>Waterfall</i>	yang berjalan di perangkatataa <i>Android,</i> yang berfungsi untuk mendapataka n materi dan memahami materi diluar jam sekolah
--	------------------------------	---	--	---	---

Sumber: *(Data peneliti, 2021)*

2.4 Kerangka Pemikiran

Kerangka pemikiran merupakan sebuah jalur pemikiran yang di rancang oleh peneliti berdasarkan kegiatan dari penelitian yang dilakukan. Menurut Mujiman (dalam skripsi Diah, 2011:30) menyatakan bahwa kerangka pemikiran merupakan konsep berisikan hubungan antara variabel bebas dan variabel terikat dalam rangka memberikan jawaban sementara (Ningrum 2017).



Gambar 2.20 Kerangka Pemikiran
Sumber: (Data Peneliti, 2021)