

## **BAB 2**

### **TINJAUAN PUSTAKA**

#### **2.1 Landasan Teori**

Landasan teori adalah sebuah konsep yang disusun secara sistematis serta memiliki dasar yang kuat dalam sebuah penelitian. Tujuan dari landasan teori yaitu memberikan gambaran dari teori-teori yang terkait dalam penelitian. Landasan teori yang digunakan untuk membangun aplikasi rekomendasi resep makanan bayi ini meliputi penjelasan tentang aplikasi, Aplikasi Android, Android Studio, JavaScript, sistem rekomendasi, makanan bayi, resep, api dan UML.

#### **2.2 Aplikasi**

Aplikasi adalah suatu program komputer yang dibuat untuk mengerjakan dan melaksanakan tugas khusus dari pengguna. Program ini terdiri dari rangkaian instruksi yang ditulis dalam bahasa pemrograman tertentu untuk melakukan fungsi-fungsi yang diinginkan oleh pengguna. Dalam konteks penelitian, aplikasi dapat merujuk pada penggunaan atau penerapan suatu teori, metode, atau konsep dalam penelitian untuk mencapai tujuan tertentu. Tugas ini beragam, seperti pengolahan data, pengelolaan informasi, komunikasi, hiburan, atau tugas-tugas lainnya yang berhubungan dengan kebutuhan pengguna[10]. Menurut Hengki W.Pramana, Aplikasi adalah suatu unit perangkat lunak yang dibuat untuk melayani kebutuhan akan beberapa aktivitas seperti Sistem perniagaan, game pelayanan masyarakat, periklanan, atau semua proses yang hampir dilakukan manusia. Sedangkan menurut Ali Zaki dan Smitdev community, Aplikasi adalah komponen yang berguna melakukan pengolahan data maupun kegiatan-kegiatan seperti pembuatan dokumen atau pengolahan data.

#### **2.3 Android**

Android merupakan sistem operasi seluler yang banyak digunakan pada perangkat seluler seperti ponsel pintar dan tablet. Android dirancang terutama untuk perangkat layar sentuh, memungkinkan pengguna memanipulasi perangkat mereka dengan gerakan jari intuitif seperti mencubit, menggesek, dan mengetuk. Android adalah platform sumber terbuka, yang berarti pengembang dapat memodifikasi dan

menyesuaikan sistem agar sesuai dengan kebutuhan mereka. Menurut Yosef Murya Android adalah “sistem operasi berbasis linux yang digunakan untuk telepon seluler (Mobile) seperti telepon pintar (smartphone) dan komputer tablet (PDA)”[11].

#### **2.4 Android Studio**

Android studio ini adalah lingkungan pengembangan baru dan terintegrasi dengan penuh, yang telah dirilis oleh Google untuk sistem operasi Android dan dirancang untuk menjadi peralatan baru dalam pengembangan aplikasi dan memberi alternatif selain Eclips yang saat ini menjadi IDE yang banyak dipakai. Menurut Nadia Firly Android Studio “merupakan Integrated Development Environment (IDE) atau dalam artian lain adalah sebuah lingkungan pengembangan terintegrasi resmi yang memang dirancang khusus untuk pengembangan sistem operasi google Android”[12].

#### **2.5 Java**

Java adalah suatu teknologi didunia *software* komputer. Selain merupakan suatu bahasa pemrograman, Java juga merupakan suatu platform. Java merupakan teknologi dimana teknologi tersebut mencakup java sebagai bahasa pemrograman yang memiliki sintak dan aturan pemrograman sendiri. Juga mencakup java sebagai platform dimana teknologi ini memiliki virtual machine dan library yang diperlukan untuk menulis dan menjalankan program yang ditulis dengan bahasa pemrograman Java. Java merupakan suatu teknologi yang unik dan revolusioner dan merupakan teknologi pertama didunia *software* yang memiliki semboyan "write once, run anywhere" Semboyan tersebut telah terbukti karena banyak program java dapat dijalankan di berbagai platform sistem operasi seperti Linux, Windows maupun Unix[13].

Adapun karakteristik Java menurut Rickyanto adalah:

1. Sederhana: Java tidak memiliki sintaks yang aneh tetapi banyak menggunakan sintaks c++ yang sudah banyak dikenal, sehingga java tidak menyulitkan bagi para programmer. Bahkan java memberikan banyak keunggulan dan kemudahan dibanding C++

2. Berorientasi objek: Java merupakan pemrograman berorientasi objek yang murni. Dalam pemrograman Java semua adalah objek, terkecuali tipe primitif
3. Dapat didistribusikan Dengan mudah: Sifat distribusi dari Java sangat tampak sebagai applet dan library yang mampu bekerja dalam jaringan dan bekerja dengan objek terdistribusi (RMI) dengan sangat baik.
4. Aman: Program Java memiliki library security serta policy yang membatasi akses applet dikomputer client
5. Dinterpretasi oleh interpreter: Java memerlukan virtual machine yang bertindak sebagai interpreter yang menterjemahkan bytecode (file class) menjadi bahasa mesin yang dimengerti oleh komputer host.
6. Portabel: Java dapat dijalankan diberbagai platform tanpa perubahan kode: Multi threading. Java memiliki banyak kemampuan untuk menangani dan menjalankan banyak thread sekaligus.
7. Dinamix: Java merupakan teknologi yang terus berkembang dan hal ini tampak nyata sekali dengan library yang terus ditingkatkan kemampuan dan kelengkapannya.
8. Netral terhadap arsitektur hardware: Java dapat dijalankan dengan baik pada komputer yang memiliki arsitektur berbeda-beda.
9. Robust: Java merupakan teknologi yang mampu menolong programmer untuk menghasilkan program secara cepat dan handal karena Java mencegah adanya memori leaking, meniadakan pointer serta mencegah berbagai eror yang mungkin terjadi dengan adanya proses pengecekan awal pada kompilasi.

## **2.6 Sistem Rekomendasi**

Sistem rekomendasi menurut Hadi Susanto merupakan sistem yang bertujuan untuk memperkirakan informasi yang menarik bagi penggunanya dan juga membantu calon konsumen dalam memutuskan barang apa saja yang akan dibelinya. Sedangkan menurut David Eko Wibowo, Sistem rekomendasi merupakan sistem yang mengaplikasikan berbagai macam teknik pencarian pengetahuan untuk mengatasi masalah pembuatan rekomendasi secara personal. Berdasarkan pengertian para ahli maka dapat disimpulkan bahwa sistem rekomendasi merupakan sistem yang menerapkan berbagai teknik pencarian yang

bertujuan untuk memberikan informasi yang menarik yang diminati pengguna dan juga membantu pengguna dalam menentukan pilihan[14].

## **2.7 Makanan**

Makanan merupakan kebutuhan pokok manusia yang dibutuhkan oleh tubuh setiap saat dan memerlukan pengolahan yang baik dan benar agar bermanfaat bagi tubuh. Makanan adalah semua bahan dalam bentuk olahan yang dimakan manusia kecuali air dan obat-obatan[15].

### **2.7.1 Makanan Bayi**

Makanan Pendamping ASI (MP-ASI) adalah jenis makanan tambahan yang diperkenalkan kepada bayi setelah usia 6 bulan hingga 24 bulan. Penting untuk dicatat bahwa MP-ASI tidak dimaksudkan untuk menggantikan ASI; ASI harus tetap diberikan kepada bayi, minimal hingga usia 24 bulan. MP-ASI sebenarnya bertujuan untuk melengkapi ASI yang diberikan kepada bayi[16]. Pemberian MP-ASI bertujuan untuk meningkatkan asupan energi dan nutrisi yang diperlukan oleh bayi, karena ASI pada saat tertentu tidak lagi bisa memenuhi kebutuhan nutrisi bayi. Selain itu, pemberian MP-ASI juga membantu bayi dalam belajar makan dan memberikan kesempatan untuk memperkenalkan berbagai jenis makanan dan rasa pada bayi, sehingga bayi dapat mengembangkan kebiasaan makan yang baik sejak dini. Makanan Pendamping Air Susu Ibu (MP-ASI) adalah makanan atau minuman yang mengandung zat gizi yang diberikan pada bayi atau anak usia 6-24 bulan guna memenuhi kebutuhan gizi selain ASI[17].

### **2.7.2 Resep**

Resep merupakan sebuah petunjuk atau instruksi tertulis yang mencakup langkah-langkah menyiapkan dan menyiapkan makanan atau minuman dengan bahan dan takaran tertentu. Resep mencakup informasi tentang bahan yang dibutuhkan, jumlah yang dibutuhkan, petunjuk langkah demi langkah untuk memasak atau menyiapkan bahan. Dalam buku masak, resep sering kali disajikan dalam bentuk teks dan dapat dilengkapi dengan gambar untuk memudahkan pemahaman dan penyajian. Buku masak adalah sumber yang kaya akan berbagai resep dari berbagai budaya dan jenis masakan. Resep makanan merupakan

panduan bagi seseorang dalam mengolah bahan makanan menjadi hidangan yang dapat dikonsumsi[18].

## 2.8 API

*Application programming interface (API)* merupakan suatu dokumentasi yang terdiri dari interface, fungsi, kelas, struktur dan sebagainya untuk membangun sebuah perangkat lunak. Dengan adanya API ini, maka memudahkan programmer untuk “membongkar” suatu *software*, kemudian dapat dikembangkan atau diintegrasikan dengan perangkat lunak yang lain. API dapat dikatakan sebagai penghubung suatu aplikasi dengan aplikasi lainnya yang memungkinkan programmer menggunakan sistem function. Proses ini dikelola melalui sistem operasi. Keunggulan dari API ini adalah memungkinkan suatu aplikasi dengan aplikasi lainnya dapat saling berhubungan dan berinteraksi[19].

## 2.9 REST API

*REST API Representational State Transfer Application Programming Interface*) adalah kerangka kerja arsitektur yang memungkinkan pengiriman data melalui antarmuka standar seperti *HTTP*. Konsep *REST* ini dikembangkan oleh Roy Fielding, salah satu pendiri proyek *Apache HTTP Server*. Dalam konteks arsitektur *REST*, server bertindak sebagai penyedia sumber daya atau data yang dapat diakses oleh klien *REST*. Sumber data ini direpresentasikan dalam berbagai format, termasuk teks, *JSON (JavaScript Object Notation)*, atau *XML (eXtensible Markup Language)*. *JSON*, sebagai salah satu format pesan yang digunakan dalam *REST API*, memiliki keunggulan dalam kekompakan, kemudahan pembacaan oleh manusia, dan kemampuan pemrosesan oleh mesin[20]. Ini adalah pendekatan yang banyak digunakan untuk merancang dan membangun API karena kesederhanaan, skalabilitas, dan kemudahan integrasi dengan sistem yang berbeda[21].

Tahapan dalam membangun sebuah *REST API (Application Programming Interface)* umumnya meliputi langkah-langkah berikut:

1. Identifikasi fitur dan fungsi yang diperlukan, seperti operasi CRUD (*Create, Read, Update, Delete*) untuk resep makanan bayi.

2. Mendesain API: Tentukan teknologi dan framework yang akan digunakan untuk membangun API, seperti Node.js, Flask, atau Django. Definisikan model data untuk resep makanan bayi, termasuk atribut-atribut seperti nama resep, bahan-bahan, langkah-langkah, dan atribut lainnya yang relevan.  
Buat spesifikasi API secara detail, termasuk rute endpoint, metode HTTP yang didukung, dan format data permintaan dan *Respons*.
3. Kembangkan API: Buat rute endpoint untuk operasi CRUD pada resep makanan bayi, seperti membuat resep baru, mengambil daftar resep, mengupdate resep, dan menghapus resep.
4. Buat atau pilih database yang sesuai untuk menyimpan data resep makanan bayi, seperti MySQL, MongoDB, atau PostgreSQL.
5. API: API harus didokumentasikan untuk memudahkan pengembang menggunakannya. Dokumentasi harus menyertakan informasi tentang sumber daya, URI, metode HTTP, dan representasi.
6. Uji API: Langkah terakhir adalah menguji API untuk memastikannya berfungsi seperti yang diharapkan. Ini dapat dilakukan dengan menggunakan alat seperti Postman. pengujian unit untuk memastikan setiap komponen API berfungsi dengan baik. Uji rute endpoint menggunakan alat pengujian API seperti Postman atau curl untuk memverifikasi *Respons* dan perilaku API.

Langkah-langkah yang terlibat dalam pengembangan *REST API* termasuk mengidentifikasi sumber daya, merancang API, mengembangkan API, mendokumentasikan API, dan menguji API. Langkah-langkah ini dapat bervariasi tergantung pada persyaratan proyek, dan penting untuk mengikuti praktik terbaik guna memastikan bahwa API aman, andal, dan efisien.

## 2.10 UML

Unified Modeling Language atau UML adalah Bahasa pemodelan spesifikasi standar sistem yang digunakan untuk mendokumentasikan spesifikasi perangkat lunak yang akan dibangun. UML merupakan bahasapemodelan yang dipakai dalam pengembangan atau pembangunan sistem yang berorientasi

objek[22]. Didalam UML, terdapat diagram-diagram yang digunakan sebagai alat bantu pemodelan. Diagram-diagram tersebut adalah:

1. *Use Case Diagram*

*Use Case* diagram merupakan diagram yang menggambarkan kelakuan sistem yang dibuat. *Use Case* diagram menggambarkan sebuah interaksi antara satu atau lebih aktor dengan sistem yang dibuat. Secara sederhana, *Use Case* diagram digunakan untuk memahami fungsi apa saja yang ada didalam sebuah sistem dan siapa saja yang dapat menggunakan fungsi-fungsi tersebut.

2. *Activity Diagram*

Diagram aktivitas merupakan diagram yang menggambarkan memodelkan aliran kerja atau workflow dari urutan aktivitas dalam suatu proses yang mengacu pada *Use Case* diagram yang ada.

3. *Sequence Diagram*

*Sequence* diagram merupakan diagram yang menggambarkan pengaplikasian dari masing-masing *Scenario Use Case* yang ada pada sistem.

4. *Class Diagram*

Diagram kelas merupakan diagram yang menggambarkan relasi atau hubungan yang dimiliki oleh antar kelas yang didalamnya terdapat atribut dan fungsi dari suatu objek.

## 2.11 *Use Case diagram*

*Use Case* diagram adalah salah satu jenis diagram yang terdapat pada *UML* 2.0 yang memiliki tujuan untuk memodelkan interaksi antara sistem dan aktor atau yang biasa disebut *user*[23]. Diagram ini menyajikan informasi mengenai fitur dan fungsi sistem, serta menggambarkan hubungan antara sistem dengan *user*. Didalam *Use Case* diagram terdapat beberapa elemen yang dapat digunakan diantaranya:

1. *Actor*

Aktor merupakan entitas yang berinteraksi dengan sistem dan terlibat dalam satu atau beberapa *Use case*. Pada umumnya aktor digambarkan dengan stickman, gambar diatas merupakan contoh dari penggunaan aktor.

2. *Use case*

*Use Case* merupakan fungsi sistem yang dapat dipahami pengguna atau aktor. *Use Case* juga dapat disebut komponen gambaran fungsional dalam sebuah sistem. Sehingga pengguna dan sistem saling mengenali alur sistem yang dibuat

### 3. *System boundaries*

Sistem *boundaries* merupakan kesatuan dari sistem yang dibuat. Saat *Use Case* berada pada sistem *boundaries*, aktor akan berada diluar sistem. Sistem *boundaries* digunakan untuk mendefinisikan cangkupan sistem dan menjelaskan tanggung jawab dari masing-masing sistem, sub-sistem, dan komponen.

Terdapat beberapa *symbol* yang dapat digunakan dalam penggunaan *Use Case* diagram, berikut adalah gambar table berupa symbol beserta fungsi yang dapat digunakan dalam penggunaan *Use Case* diagram.

## 2.12 *Activity Diagram*

*Activity Diagram* digunakan untuk memodelkan proses bisnis atau alur kerja yang kompleks. Diagram ini umumnya digunakan untuk memperpresentasikan urutan aktivitas atau tindakan yang dilakukan oleh objek dalam suatu sistem atau proses bisnis[23]. *Activity Diagram* terdiri dari beberapa elemen diantaranya:

### 1. *Start dan End Point*

*Start dan end point* adalah elemen yang menunjukkan awal dan akhir dari aktivitas atau proses yang digambarkan dalam diagram.

### 2. *Decision Point*

*Decision point* digunakan untuk menggambarkan percabangan dalam alur kerja, dimana objek harus membuat keputusan antara beberapa pilihan.

### 3. *Fork dan Join*

*Fork dan join* digunakan untuk menunjukkan pemrosesan paralel atau bersamaan dari beberapa aktivitas atau proses.

### 4. *Action*

*Action* merupakan elemen yang menunjukkan tindakan atau aktivitas yang dilakukan oleh objek dalam proses atau alur kerja.

Dalam pembuatan *Activity Diagram*, terdapat beberapa langkah yang dapat diikuti, yaitu menentukan objek atau proses yang akan dimodelkan, menambahkan *start dan end point*, menambahkan elemen-elemen seperti *decision point, fork* dan *join*, serta menambahkan *action* atau aktivitas yang dilakukan objek.

### 2.13 *Sequence Diagram*

Sequence diagram dipergunakan untuk mempresentasikan interaksi antara objek atau komponen yang ada dalam suatu sistem Diagram ini digunakan untuk menggambarkan urutan pesan atau panggilan method yang dikirimkan antara objek dalam sistem[23]. *Sequence diagram* terdiri dari beberapa elemen, diantaranya:

#### 1. Objek

Objek merupakan elemen yang mempresentasikan komponen atau entitas dalam sistem, seperti kelas, interface.

#### 2. Pesan

Pesan digunakan untuk mempresentasikan panggilan method atau pertukaran informasi antara objek.

#### 3. *Lifeline*

*Lifeline* adalah garis vertical yang menunjukkan urutan waktu atau pengurutan panggilan method antara objek.

#### 4. *Activation*

*Activation* menunjukkan waktu pemrosesan yang dibutuhkan oleh objek dalam *meRespon* pesan atau panggilan method.

Pada umumnya, *sequence diagram* digunakan dalam proses analisis dan desain sistem yang kompleks, terutama dalam memodelkan interaksi antar objek dalam sistem. Sequence diagram dapat membantu dalam mengidentifikasi dan memperbaiki masalah atau kekurangan dalam interaksi antar objek, serta memudahkan dalam merancang dan mengimplementasikan sistem yang efektif dan efisien.