

BAB II

LANDASAN TEORI

2.1. Tinjauan Jurnal

Menurut (Asdiana,2014:354) Menyimpulkan bahwa: Makanan dan minuman adalah faktor penentu dalam memilih tujuan wisata kuliner. Oleh karena itu untuk mendukung daya tarik wisata perlu ada jasa penyediaan makanan dan minuman (*food and beverage service*) yang baik.

Menurut (Safaat 2011 : 5) Menyimpulkan bahwa: Android SDK *Tools* API yang diperlukan untuk mengembangkan aplikasi pada platform Android yang ditulis menggunakan bahasa pemrograman Java. Software Development Kit (SDK) Android mencakup perangkat pengembangan, emulator, dan libraries yang diperlukan untuk membangun aplikasi Android.

Perkembangan teknologi sistem operasi android berimplikasi pada perkembangan aplikasi dan cara pertukaran data dengan teknologi yang sudah ada. Perkembangan tools dan aplikasi baru menuntut untuk saling terintegrasinya antara teknologi baru dan teknologi lainnya. Dengan adanya teknologi baru seperti sistem android ini dipastikan akan adanya aplikasi baru yang harus diselaraskan dengan teknologi mapan yang sudah ada. Pada penulisan skripsi ini penulis merujuk pada beberapa jurnal tentang sistem operasi android dan kasus yang sesuai.

2.2. Konsep Dasar Program

Menurut Moh. Sjukani (Juli 2011:9) Menyimpulkan bahwa: Program adalah kumpulan instruksi-intruksi yang diberikan kepada komputer untuk melaksanakan suatu tugas untuk pertama kali, program di ketik ke dalam komputer melalui *keyboard* yang hasilnya dapat dilihat di screen atau layar monitor.

Bahasa pemrograman atau sering diistilahkan dengan bahasa komputer, adalah teknik komando atau industri standar untuk memerintah komputer. Bahasa pemrograman ini merupakan suatu himpunan dari *sintaks* dan *semantik* yang dipakai untuk mendefinisikan pada program komputer. Bahasa ini memungkinkan seorang programmer dapat menentukan secara persis data mana yang akan diolah oleh komputer, bagaimana data ini dapat disimpan atau diteruskan, dan jenis langkah apa secara persis yang akan diambil dalam berbagai situasi.

Menurut tingkat kedekatannya dengan mesin computer, bahasa pemrograman terdiri dari :

1. Bahasa mesin, yaitu dengan memberikan perintah kepada computer dengan memakai kode bahasa biner, contohnya 01100101100110
2. Bahasa tingkat rendah, atau dikenal dengan istilah bahasa rakitan (*Assembly*), yaitu memberikan computer dengan memakai kode-kode singkat (kode Mnemonic), contohnya Mov, SUB,CMP,JMP,JGE,JL dan LOOP.

3. Bahasa tingkat menengah, yaitu bahasa computer yang memakai campuran instruksi dalam kata-kata bahasa manusia dan instruksi yang bersifat simbolik, contohnya {,},?,<<,>>,&&.
4. Bahasa tingkat tinggi, yaitu bahasa computer yang memakai instruksi berasal dari unsur kata-kata manusia, contohnya *begin, end, if, for, while, and, or*.

Sebagian besar bahasa pemrograman digolongkan sebagai bahasa tingkat tinggi, hanya bahasa C yang digolongkan sebagai bahasa tingkat menengah dan Assembly yang merupakan bahasa tingkat rendah.

Pada program yang penulis rancang terdapat beberapa komponen yang diperlukan seperti sistem operasi Android. *SQLite Database, MySQL database*, bahasa pemrograman *PHP, apache web server* sebagai *web service*, bahasa pemrograman Java, *Markup Language XML* metode pengiriman data HTTP POST dan format pertukaran data berupa JSON.

2.2.1. Bahasa Pemrograman Java

Menurut Nazrudin safaat H. (2012:14) Menyimpulakan bahwa: Java merupakan bahasa pemrograman berorientasi objek yang dikembangkan oleh Sun Microsystems, suatu perusahaan yang terkenal dengan Workstation UNIX high end. Sejak dirilis pada tahun 1995, bahasa pemrograman java dengan cepat memperoleh popularitas dikalangan para pemrogram.

Program yang dihasilkan dengan bahasa java dapat berupa applet (aplikasi kecil yang berjalan diatas web browser) maupun berupa aplikasi mandiri yang

dijalankan dengan program java interpreter. Contohnya program yang tulis dengan bahasa java adalah Hot java yang berupa sebuah web browser.

1. Karakteristik Java

Java adalah sebuah bahasa pemrograman berorientasi obyek murni. Jadi program-program java berada dalam sebuah struktur kelas-kelas dan obyek-obyek. Pada dasarnya sintaks pada bahasa java mirip dengan sintaks pada bahasa C atau C++ java bertipe kuat (*Strongly-typed*). Ini berarti semua tipe data terikat secara traits atau dengan kata lain setiap nama variable disosialisasikan dengan sebuah tipe data tunggal yang dikenal pada saat kompilasi.

2. Keuntungan dan kelemahan dari Java :

- a. Memiliki tingkat fleksibilitas yang tinggi dan program java bersifat modular serta kode dapat digunakan kembali
- b. Java relative lebih muda untuk dipelajari
- c. Java lebih sederhana, karena telah menghilangkan beberapa fitur dari C++ yang membimbingkan dan mudah untuk dipelajari.
- d. Java memiliki kinerja yang tinggi dan handal
- e. Java dapat sebagai penterjemah sehingga dapat digunakan pada lingkungan komputer jenis apapun tanpa merubah kode program kembali.
- f. Java memiliki tingkat keamanan yang sangat baik
- g. Multithreading, sehingga program java dapat melakukan beberapa perintah (program) sekaligus.

3. Android

Menurut Nazarudin Safaat H, MT (Oktober 2015:18-19) Menyimpulkan bahwa: Android adalah sebuah sistem operasi untuk perangkat mobile berbasis Linux yang mencakup sistem operasi, middleware, dan aplikasi. untuk perangkat *protable* seperti *smartphone* dan computer tablet. Android menyediakan platform terbuka (*open source*). Bagi programmer untuk mengembangkan aplikasi sendiri pada berbagai perangkat dengan sistem android.

Pada awalnya Sistem operasi Android ini dikembangkan oleh android Inc, kemudian diakuisisi oleh google Inc. yang setelah itu terus menerus melakukan perkembangan ke teknologi yang lebih baik. Sistem operasi android ini punya berbagai macam seri, tergantung dari urutan pembuatannya. Android yang pertama kali muncul adalah 1.0 atau seri Apple Pie. Secara berurutan, berikut adalah versi-versi sistem operasi android dari awal hingga saat ini :

1. Android versi 1.0 (Android Apple Pie), dirilis 23 September 2008
2. Android versi 1.1 (Android Banna Bread), dirilis 9 Maret 2009
3. Android 1.5 (Android Cupake)
4. Android versi 1.6 (android Donut)
5. Android versi 2.0/2.1 (Android Éclair)
6. Android versi 2.2 (froyo:Frozen Youghurt)
7. Android versi 2.3 (Android Gingerbread)
8. Android versi 3.0 / 3.1 / 3.2 (Honeycomb)
9. Android versi 4.0 (IceCreamSandwich)
10. Android versi 4.1 (Android Jelly Bean)

11. Android versi 4.4 (Android Kitkat)
12. Android versi 5.0 (Android Lollipop)
13. Android versi 6.0 (Android Marshmallow)
14. Android versi 7.0 (Android Nougat)

2.2.2. *Exetensible Markup Language (XML) Untuk Android User Interface.*

User Interface (UI) Di Android dapat dibangun dengan dua cara, yakni dengan mendefinisikan XML atau dengan menulis kode Java. Mendefinisikan Struktur GUI dalam XML sangat disukai, karena sebagai salah satu dari prinsip *Model-Viewer-Control*, dimana UI haruslah selalu terpisah dari logika program.

Tata letak berbasis XML adalah hubungan *Widget* ‘satu sama lain dan *container* dalam Format XML. Android menjadikan XML sebagai basis *layout* untuk menjadikan resource, dan disimpan dalam *res/layout* direktori dalam project android.

1. *Resources*

Resources adalah file eksternal (non-kode file) yang digunakan oleh kode dan kompilasi pada saat membangun aplikasi. Android mendukung beberapa jenis file yang berbeda sebagai resources, yakni XML, PNG, dan file JPEG. Berikut Tipe-tipe resources and tempatnya.

A. Tipe-tipe Resources :

1. Layout-files --> “/res/layout/”
2. Images --> “res/drawable/”
3. Animasion --> “/res/anim/”

4. Styles, string and arrays --> “/res/values/”

- a. ‘array.xml’ untuk mendefinisikan arrays
- b. ‘colors.xml’ untuk mendefinisikan colors
- c. ‘dimensi.xml’ untuk mendefinisikan dimensions
- d. ‘strings.xml’ untuk mendefinisikan strings
- e. ‘styles.xml’ untuk mendefinisikan styles object

2. SQLite

SQLite Menurut tulisan dalam website sqlite.org SQLite merupakan mesin *database* yang digunakan dalam sebuah proses untuk mengimplementasikan data secara mandiri. Tidak seperti kebanyakan *database* SQL lainnya, SQLite tidak memiliki proses server yang terpisah. SQLite membaca dan menulis secara langsung ke *disk*.

2.3. Algoritma String Matching

Menurut Moh. Sjukani (Juli 2011:1) menyimpulkan bahwa: algoritma adalah alur pikiran dalam menyelesaikan suatu pekerjaan. Algoritma yang dipakai dalam pencarian string matching atau sering disebut juga dengan algoritma pencocokan string yaitu algoritma untuk melakukan pencarian semua kemunculan string pendek dan panjang, untuk string pendek yang disebut pattern dan string yang panjang disebut teks.

Contoh :

String pendek = pattern [0..n – 1]

String panjang = teks [0..m – 1]

2.4. Pengujian Software

Pengujian software merupakan elemen kritis dari jaminan kualitas perangkat lunak dan merepresentasikan kajian pokok dari spesifikasi, desain dan pengkodean. Pada dasarnya, pengujian merupakan satu langkah dalam proses rekayasa perangkat lunak yang dapat dianggap (paling tidak secara psikologis)

Pada tahap ini penulis mengimplementasikan pengujian dengan menggunakan dua macam pendekatan test yaitu :

a. *Black Box Testing*

Test case ini bertujuan untuk menunjukan fungsi perangkat lunak tentang cara beroperasinya, apakah pemasukan data telah berjalan sebagaimana yang diharapkan dan apakah informasi yang disimpan secara eksternal selalu dijaga kemutakhirannya.

b. *White Box Testing*

Pengujian sistem yang dilakukan adalah untuk memeriksa kekompakan antar komponen sistem (antar form) dengan tujuan utamanya adalah untuk memastikan elemen-elemen sistem berfungsi sesuai dengan diharapkan. Pengujian sistem termasuk juga pengujian program secara menyeluruh. Kesimpulan program yang diintegrasikan perlu diuji coba atau dites untuk melihat apakah sebuah program dapat menerima dengan baik, memproses dan memberikan keluaran program yang baik pula.

2.5. Peralatan Pendukung (*Tools System*)

Untuk mewujudkan sistem yang telah di rancang memerlukan sarana pendukung yakni berupa komponen-komponen yang sangat berperan dalam menunjang penerapan sisten terhadap pengolahan data.

1. Hardware

Dalam pembuatan program ini penulis menggunakan sebuah computer dengan spesifikasi sebagai berikut :

- a. Harddisk 500 GB
- b. Memory 2.00 GB
- c. Prosesor Intel (R) Core (TM) core prosesor i5-2450M CPU 2.50GHz

2. *Unified Modeling Languange (UML)*

UML mendefinisikan berbagai diagram, diantaranya : *use case diagram*, *class diagram*, *activity diagram*, *sequence diagram* dan *deployment diagram*.

A. Definisi UML

Menurut Sukamto dan Shalahuddin (2015:137) Menyimpulkan bahwa: UML (Unified Modeling Language), merupakan bahasa visual untuk pemodelan dan komunikasi mengenai sebuah sistem dengan menggunakan diagram dan tesk-teks pendukung. bahasa pemodelan untuk sistem atau perangkat lunak yang berparadigma ‘berorientasi objek’. Pemodelan (modeling) sesungguhnya digunakan untuk penyederhanaan permasalahan-permasalahan yang kompleks sedemikian rupa sehingga lebih mudah dipelajari dan dipahami.

B. Karakteristik UML

Unified Modeling Language (UML) adalah bahasa yang telah menjadi standar untuk visualisasi, menetapkan, membangun dan mendokumentasikan arti suatu sistem perangkat lunak

Unified Modeling Language (UML) dapat didefinisikan sebagai sebuah bahasa yang telah menjadi standar dalam industry untuk visualisasi , merancang dan mendokumentasikan sistem perangkat lunak.

Unified Modeling Language (UML) merupakan standard modeling language yang terdiri dari kumpulan-kumpulan diagram, dikembangkan untuk membantu para pengembang sitem dan software agar bisa menyelesaikan tugas-tugas seperti berikut :

C. Spesifikasi :

- a. Visualisasi
- b. Desain Arsitektur
- c. Kontruksi
- d. Simulasi dan Testing
- e. Dokumentasi

Diagram-diagram yang digunakan didalam UML :

1. Activity Diagram

- a. Menggambarkan proses bisnis dan urutan aktifitas dalam sebuah proses.
- b. Dipakai untuk busniess modeling untuk memperlihatkan urutan aktifitas proses bisnis.
- c. Struktur diagram ini mirip flowchart atau data flow diagram pada perancangan terstruktur.

- d. Sangat bermanfaat apabila kita membuat diagram ini terlebih dahulu dalam memodelkan sebuah proses untuk membantu secara keseluruhan.
- e. Activity diagram dibuat berdasarkan sebuah atau beberapa use case pada use case diagram.

2. *Use Case Diagram*

Menggambarkan fungsionalitas yang diharapkan dari sebuah sistem. Yang ditekankan adalah “apa” yang diperbuat sistem, dan bukan “bagaimana”

- a. Menggambarkan kebutuhan system dari sudut pandang user.
- b. Mengfokuskan pada proses komputerisasi (automated processes).
- c. Menggambarkan hubungan antara use case dan actor
- d. Use case menggambarkan proses system (kebutuhan system dari sudut pandang user)
- e. Secara umum use case adalah :
 - 1. Pola perilaku sistem
 - 2. Urutan transaksi yang berhubungan yang dilakukan oleh satu actor
- f. Use case diagram terdiri dari :
 - 1. Use case
 - 2. Actors
 - 3. Relationsip
 - 4. System boundary boxes (optional)
 - 5. Packages (optional)

3. *Sequence Diagram*

- a. *Sequence* diagram menggambarkan interaksi antara objek di dalam dan di sekitar sistem (termasuk pengguna, display, dan sebagainya) berupa

message yang digambarkan terhadap waktu. Sequence terdiri atas dimensi vertical (waktu) dan dimensi horizontal (objek-objek yang terkait)

- b. *Sequence diagram* bisa di gunakan untuk menggambarkan scenario atau rangkaian langkah-langkah yang dilakukan sebagai respons dari sebuah even untuk menghasilkan output tertentu. Mulai dari proses dan perubahan apa saja yang terjadi secara internal dan output apa yang dihasilkan
- c. Diagram ini secara khusus berasosiasi dengan use case diagram
- d. *Memperlihatkan* tahap-tahap demi tahap apa yang seharusnya terjadi untuk menghasilkan sesuatu didalam use case.

4. *Class Diagram*

- a. *Class* adalah sebuah spesifikasi yang jika diinstansiasi akan menghasilkan sebuah objek dan merupakan inti dari pengembangan dan desain berorientasi objek.
- b. *Class* menggambarkan keadaan (atribut/properti) suatu sistem sekaligus menawarkan layanan untuk memanipulasi keadaan tersebut (metode/fungsi).
- c. *Class diagram* menggambarkan struktur dan deskripsi *class*, *package* dan objek beserta hubungan satu sama lain.
- d. *Class* memiliki tiga area pokok :
 - 1. Nama (dan stereotype)
 - 2. Atribut
 - 3. Metoda

5. *Deployment Diagram*

- a. *Deployment* diagram memperlihatkan pemetaan software kepada hardware. Setiap model hanya memiliki satu deployment diagram.
- b. *Deployment* diagram menggambarkan arsitektur fisik dari perangkat keras dan perangkat lunak sistem. Menunjukkan hubungan computer dengan perangkat (nodes) satu sama lain dan jenis hubungannya. Didalam nodes, executeable component dan objek yang dialokasikan untuk memperlihatkan unit perangkat lunak yang dieksekusi oleh node tertentu dan ketergantungan komponen.