

BAB II

KAJIAN PUSTAKA DAN LANDASAN TEORI

2.1 Kajian Pustaka

Penelitian yang berkaitan dengan pembuatan menu resep makanan sudah pernah dilakukan sebelumnya. Alpiandi (2014) dalam penelitiannya yang berjudul "Aplikasi Resep Masakan Khas Kalimantan Barat Berbasis Android" berasumsi bahwa selama ini sudah banyak buku tentang resep makanan sehingga ketika orang-orang ingin mencari sebuah resep makanan maka mereka akan sulit mencari kategori makanan yang sesuai. Solusi yang diberikan oleh peneliti yaitu menjadikan telepon seluler sebagai media lain dalam pencarian resep makanan dengan cara mengembangkan aplikasi yang bertujuan untuk memudahkan orang-orang dalam pencarian resep makanan khas Kalimantan Barat berbasis *Android*. Pengembangan aplikasi ini menggunakan bahasa pemrograman *java*, *MySQL* sebagai tempat penyimpanan data, dan *Eclipse* sebagai *software*. (Alpiandi, 2014)

Penelitian tentang aplikasi catatan pernah dilakukan oleh Muhammad Fakhur Rozi dengan judul "Aplikasi *Note* Aman Berbasis Android". Dalam penelitian ini peneliti menggambarkan bagaimana aplikasi *note* yang ada pada *smartphone* saat ini kurang aman dikarenakan metode yang digunakan yaitu *cloud computing* dimana dengan metode ini, penyadapan dan bocornya data masih bisa terjadi. Solusi yang diberikan oleh peneliti adalah mengembangkan sebuah aplikasi *note* yang mempunyai *password* yang terenkripsi dengan cara menggunakan algoritma *AES*. Pengembangan aplikasi ini menggunakan *Database SQLite* sebagai tempat penyimpanan data dan *Eclipse* sebagai *software* untuk mengembangkan aplikasi. (Rozi, 2013)

Penelitian tentang perawatan bayi sudah pernah dilakukan oleh Rizqi Akbar Gumilar dalam skripsi yang berjudul "Perancangan Aplikasi Panduan Merawat Bayi Usia 0 sampai 12 Bulan Berbasis Web". Dalam penelitian ini Rizqi beranggapan bahwa merawat dan mengasuh bayi dengan cara yang kurang atau

Tidak tepat bisa mengakibatkan penyesuaian diri yang kurang baik ketika bayi tersebut beranjak dewasa. Agar perawatan bayi dilakukan dengan tepat maka peneliti mengembangkan suatu aplikasi berbasis *web* sebagai media informasi untuk berbagi pengetahuan tentang perawatan bayi. Peneliti menggunakan beberapa *software* dalam pengembangan aplikasi *web* ini yaitu *Database MySQL* dan *PHP*. (Gumilar, 2014)

Penelitian tentang kalkulator berat badan ideal bayi pernah dilakukan oleh Fefi Silviadewi dengan judul “Rancang Bangun Mobile Application Perhitungan Berat Badan Ideal dan Informasi Status Gizi Balita Berbasis Android Dengan Metode Prototype” (2015). Dalam penelitian ini Fefi berasumsi bahwa masih banyak orangtua yang salah persepsi tentang berat badan bayi seperti orangtua yang beranggapan bayi terlihat lucu ketika memiliki badan yang gemuk, padahal bayi yang gemuk belum tentu sehat karena bisa saja mereka salah asupan gizi dan bisa terkena penyakit. Dengan permasalahan yang ada, Fefi mengembangkan sebuah aplikasi berbasis android yang dapat memantau pertumbuhan anak balita sehingga orangtua bisa mengontrol makanan yang diberikan kepada anak. Adapun *software* yang digunakan dalam pengembangan aplikasi ini adalah *Eclipse* dan *CorelDraw*. (Fefy Silviadewi, 2015)

Berdasarkan penelitian-penelitian di atas penulis berinisiatif untuk membuat sebuah aplikasi yang menggabungkan semua fitur utama yang ada pada penelitian di atas yaitu seperti fitur resep MPASI dimana fitur ini dapat menampilkan resep makanan untuk bayi berumur 6 sampai 7 bulan dan 8 sampai 12 bulan. Resep MPASI bisa membantu para ibu untuk membuat makanan-makanan tersebut. Selain fitur resep MPASI juga terdapat fitur catatan yang berguna bagi *user* dalam mencatat hal-hal penting tentang bayi. Terdapat pula fitur tips perawatan dimana fitur ini menampilkan informasi-informasi seputar cara merawat bayi. Lalu penulis juga memberikan fitur menu kalkulator berat badan ideal bayi dimana pada fitur ini *user* dapat menghitung dan melihat berat badan yang ideal untuk bayi. Bahasa pemrograman yang digunakan oleh para peneliti di atas dengan penulis juga hampir sama yaitu *PHP* dan *java*.

Adapun beberapa perbedaan yaitu penulis menggunakan android studio dan *Database MySQL* sebagai *software* untuk membantu pengembangan aplikasi *monitoring* pertumbuhan dan perkembangan bayi. Selain itu, penulis juga membuat beberapa pembaruan dari penelitian yang sudah dijelaskan di atas seperti catatan yang bisa diedit dan bisa memasukkan tanggal, lalu info perawatan bayi yang dibuat dalam bentuk aplikasi berbasis android sehingga *user* tidak harus membuka *browser* lagi. Fitur resep MPASI juga disertai dengan informasi kandungan gizi yang terdapat pada resep masakan tersebut dan penulis menambahkan fitur favorit agar *user* bisa menjadikan satu resep atau lebih menjadi favorit. Kemudian *user* juga bisa melihat grafik pertumbuhan bayi mereka ketika memasukkan umur dan berat badan bayi.

2.2 Landasan Teori

2.2.1 Tumbuh Kembang Bayi

Pertumbuhan adalah proses membesarnya sel pada tubuh. sedangkan perkembangan adalah suatu proses penyempurnaan setiap fungsi organ tubuh. (HH., 2013)

Tumbuh kembang adalah suatu proses yang bertahap dan kontinu pada bayi. jika ingin mengoptimalkan tumbuh kembang bayi maka diperlukan interaksi sosial secara teratur dan berkala pada setiap tahap perkembangan. (Adriana, 2013)

2.2.2 Makanan pendamping ASI(MPASI)

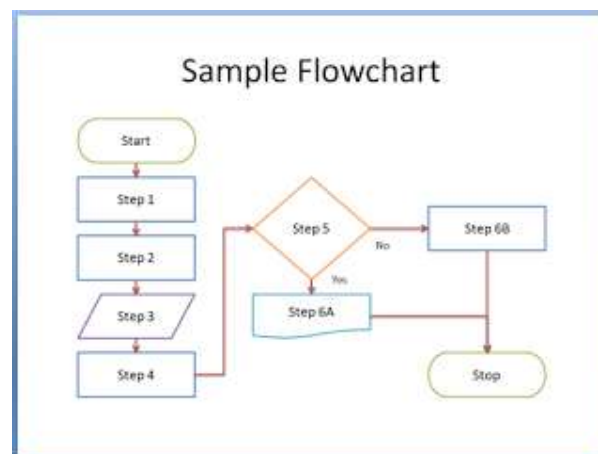
MPASI merupakan makanan tambahan untuk bayi berumur 6 bulan atau lebih yang berguna untuk penambahan gizi dan nutrisi yang tidak terdapat pada ASI (WHO, 2009, p. 19).

IDAI menyebutkan bahwa Makanan pendamping ASI adalah makanan tambahan selain ASI untuk menambah nutrisi dan diberikan kepada bayi secara bertahap mulai dari jenis makanan, jumlah makanan, dan frekuensi makanan.

2.2.3 Flowchart

Flowchart adalah sebuah bagan alir yang menggambarkan jalannya sebuah sistem (Jogiyanto, 2005).

Menurut Indrajani (2011, p22), *Flowchart* merupakan penggambaran secara grafik dari langkah-langkah dan urutan prosedur suatu program. Biasanya mempermudah penyelesaian masalah yang khususnya perlu dipelajari dan dievaluasi lebih lanjut.



Gambar 2.1 Contoh Flowchart

2.2.4 Android

Android adalah sebuah sistem operasi *open source* yang mempunyai banyak *API Library* untuk membuat aplikasi berbasis *mobile* dengan tampilan yang menarik dan berfungsi dengan bagus. ((Meier, 2009, p. 1). Menurut J.F. DiMarzio dan Jubilee, *android* merupakan suatu sistem operasi yang menggunakan *java* sebagai bahasa pemrograman dan berbasis *linux* serta dirancang khusus untuk telepon seluler layar sentuh seperti *smartphone* dan tablet. (DiMarzio, 2008, p.6; Jubilee Enterprise, 2015, p.1)

Tabel 2.1 Tingkatan Android

| Nama | Versi | Tanggal Rilis |
|-------------------|-----------|------------------------|
| Cupcake | 1.5 | 27 April 2009 |
| Donut | 1.6 | 15 September 2009 |
| Éclair | 2.0-2.1 | 26 Oktober 2009 |
| Froyo | 2.2-2.2.3 | 20 Mei 2010 |
| Gingerbread | 2.3-2.3.7 | 6 Desember 2010 |
| Honeycomb | 3.0-3.2.6 | 22 Februari 2011 |
| Icecream Sandwich | 4.0-4.0.4 | 18 Oktober 2011 |
| Jellybean | 4.1-4.3.1 | 9 Juli 2012 |
| Kitkat | 4.4-4.4.4 | 31 Oktober 2013 |
| Lollipop | 5.0-5.1.1 | 12 November 2014 |
| Marshmallow | 6.0-6.0.1 | 5 Oktober 2015 |
| Nougat | 7.0 | Agustus/September 2016 |

2.2.5 Android Studio

Android Studio adalah sebuah *software* resmi yang dikembangkan Google untuk membantu para *developer* dalam mengembangkan aplikasi (Kevin Grant, 2014, p. 16). Dengan menggunakan *Android Studio* pengembangan aplikasi menjadi lebih singkat dan jauh lebih mudah dibandingkan sebelumnya (Adam Gerber, 2014, p. 25). *Android Studio* menawarkan fitur lebih banyak untuk meningkatkan produktivitas anda saat membuat aplikasi *android*, misalnya:

1. Sistem versi berbasis Gradle yang fleksibel
2. Emulator yang cepat dan kaya fitur
3. Lingkungan yang menyatu untuk pengembangan bagi semua perangkat Android
4. *Instant Run* untuk mendorong perubahan ke aplikasi yang berjalan tanpa membuat APK baru
5. Template kode dan integrasi GitHub untuk membuat fitur aplikasi yang sama dan mengimpor kode contoh

6. Alat pengujian dan kerangka kerja yang ekstensif
7. Alat *Lint* untuk meningkatkan kinerja, kegunaan, kompatibilitas versi, dan masalah-masalah lain
8. Dukungan bawaan untuk *Google Cloud Platform*, mempermudah pengintegrasian *Google Cloud Messaging* dan *App Engine*

2.2.6 Bahasa Pemrograman *Java*

Java adalah sebuah bahasa pemrograman yang bisa dijalankan di berbagai *platform* (Kadir, 2015, p. 6). Ketika kita menggunakan bahasa *java* sebagai penunjang pengerjaan program, kita bisa membuat aplikasi windows dan mengeksplor *Database*, mengontrol *handheld device*, dan lainnya (Burd, 2014, p. 12). *Java* mempunyai *library* dan *tools* yang banyak sehingga memudahkan pekerjaan *developer*.

2.2.7 *PHP*

PHP adalah bahasa *Script* yang populer dimana *PHP* ditulis untuk berkomunikasi dengan *MySQL* (atau *Database* lainnya) pada *back-end* dan *browser* pada *front-end* (Delisle, 2012, p. 8). Dengan kata lain *PHP* dapat melakukan *create*, *read*, *update*, dan *delete* pada *Database* yang sudah dikoneksikan.

PHP atau singkatan dari *Personal Home Page* merupakan bahasa skrip yang tertanam dalam *HTML* untuk dieksekusi bersifat server side (Nugroho, 2006, p. 61).

2.2.8 *Android Virtual Devices*

AVD adalah sebuah *emulator* yang mengizinkan aplikasi android untuk diuji di komputer atau laptop sendiri tanpa harus menginstal aplikasi pada perangkat android (Smyth, 2015, p. 33). AVD terdiri dari:

1. Sebuah Profil perangkat keras. Kita bisa mengatur opsi untuk menentukan fitur *hardware emulator*. Misalnya menentukan apakah menggunakan perangkat kamera, apakah menggunakan *keyboard QWERTY* fisik atau tidak, berapa banyak memori internal, dan lainnya.
2. Sebuah pemetaan versi Android dimana kita dapat menentukan versi dari *platform* Android yang akan berjalan pada *emulator*.

3. Pilihan lainnya seperti kita dapat menentukan *skin* yang ingin digunakan pada *emulator* yang memungkinkan untuk menentukan dimensi layar, tampilan, dan sebagainya

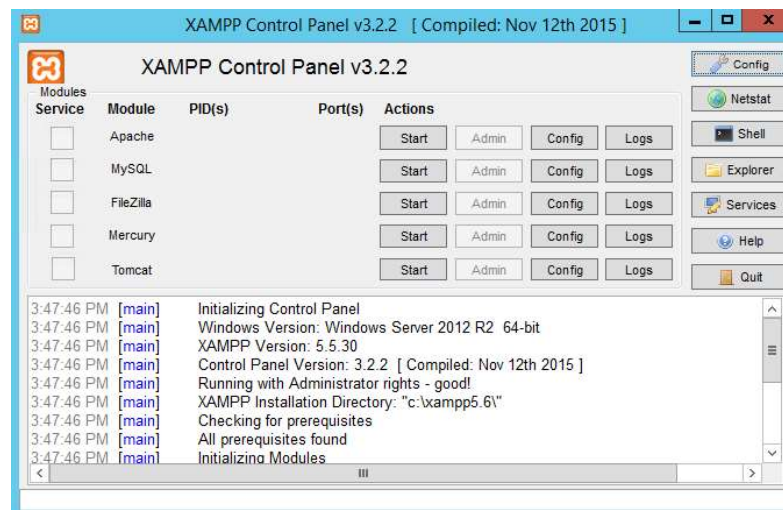
2.2.9 Netbeans

Menurut Tim Bouderau dan David Salter, *Netbeans* adalah lingkungan pengembangan terintegrasi (IDE) yang terkenal dengan sistem *open source* yang berbasis bahasa pemrograman java untuk digunakan dalam pengembangan *PHP*, *C/C++*, *HTML*, dan lainnya (Boudreau, 2003, p. 8; Salter, 2014, p. 7). Selain itu dengan menggunakan *Netbeans*, para *developer* dapat mengembangkan aplikasi jauh lebih cepat (Friesen, 2011, p. 8)

2.2.10 XAMPP

Menurut Yogi Wicaksono (2008:7) *XAMPP* adalah sebuah *software* yang berfungsi untuk menjalankan *website* berbasis *PHP* dan menggunakan pengolah data *MySQL* dikomputer *local* (Wicaksono, 2008, p. 7).

Menurut Janet Valade *XAMPP* adalah kit instalasi yang berisi *Apache*, *PHP*, dan *MySQL*. *XAMPP* mempunyai versi yang tersedia untuk *windows* dan *linux* (Janet Valade, 2013, p. 24). Dengan menggunakan *XAMPP developer* bisa membuat *website* tanpa harus terhubung ke internet.



Gambar 2.2 Tampilan XAMPP

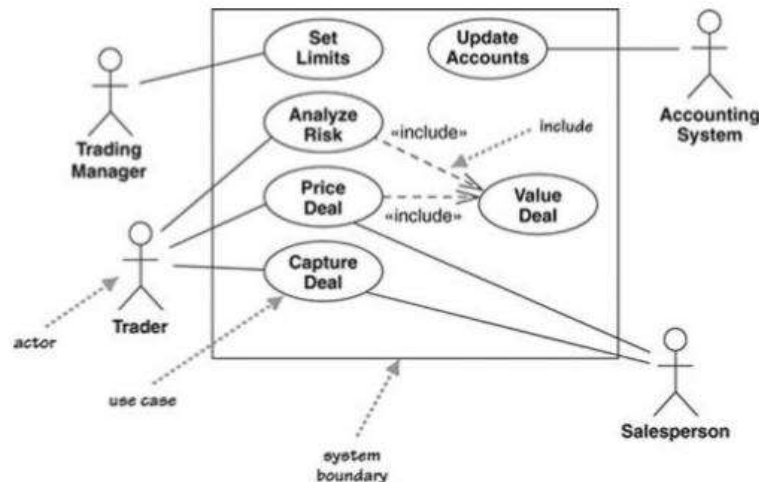
2.2.11 Unified Modeling Language (UML)

UML adalah bahasa standar *modeling* yang digunakan untuk menspesifikasi, menggambarkan, membangun perancangan perangkat lunak dan pengembangan sistem serta mendokumentasikan artifak (James Rumbaugh, 1999, p. 3).

Penggunaan *UML* dapat membantu para *developer* dalam pengumpulan komponen-komponen yang diperlukan dan bagaimana fungsi kerjanya. Dengan menggunakan *UML*, para *developer* juga bisa memperlihatkan desain aplikasi dengan para *client* dengan tampilan yang mudah dipahami. Model *UML* yang digunakan dalam aplikasi *monitoring* pertumbuhan dan perkembangan bayi adalah sebagai berikut:

1. Use Case Diagram

Use case adalah sebuah diagram yang menggambarkan keperluan fungsional dari sebuah sistem (Fowler, *UML Distilled*, 1997, p. 79). Cara kerja *use case* yaitu dengan menggambarkan interaksi antara aktor dengan sistem itu sendiri. Aktor di sini diperankan oleh *user*.



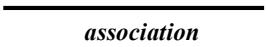
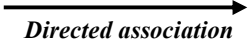
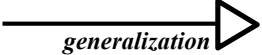

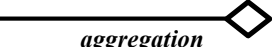
Gambar 2.3 Contoh Usecase

2. *Class diagram*

Class diagram menggambarkan tipe-tipe objek dan relasi apa saja yang terjadi di dalam sistem tersebut. *Class diagram* mempunyai tiga elemen yaitu *operation*, *attributes*, dan nama kelas *attributes* dan *operation* di dalam kelas mempunyai tiga sifat yaitu:

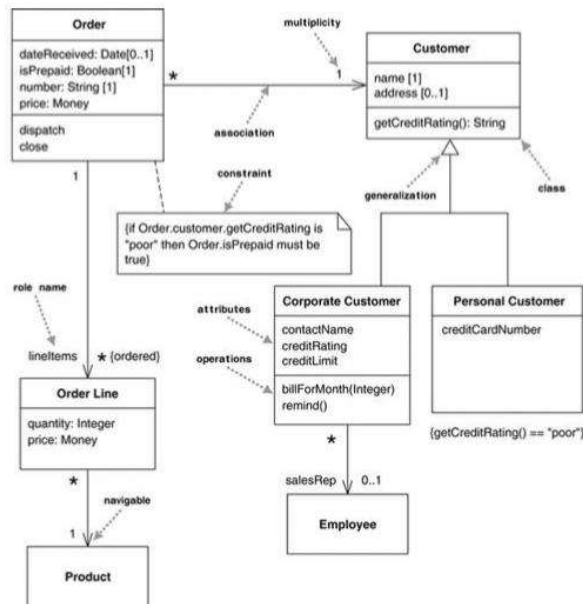
1. *Public*: dapat dipanggil di semua kelas
2. *Protected*: hanya dapat dipanggil oleh kelas yang bersangkutan
3. *Private*: hanya bisa dipanggil di dalam kelas itu sendiri

Adapun simbol-simbol relasi yang menghubungkan antara satu kelas dengan kelas lainnya yaitu:

| Nama dan Gambar | Fungsi |
|---|---|
|  | Menghubungkan antara satu kelas dengan kelas lainnya |
|  | Menghubungkan antara satu kelas yang menggunakan kelas lain |
|  | Menghubungkan antara kelas utama dengan kelas yang diwarisi |
|  | Menghubungkan kelas yang saling berkegantungan |
|  | Hubungan antara satu kelas yang menjadi atribut bagi kelas lain |

Tabel 2.2 Penjelasan relasi *class diagram*

Berikut adalah contoh *class diagram*



Gambar 2.4 Contoh *Class Diagram*

2.2.12 *Black Box Testing*

Menurut Rex Black, *black box testing* adalah suatu metode pengujian dimana *tester* hanya fokus pada apa yang seharusnya dilakukan oleh sistem (Black, 2009, p. 3). Sebuah tes dapat dikatakan berhasil ketika sebuah sistem dapat memproses data dan hasil yang ada sesuai dengan apa yang diharapkan. Ketika menggunakan metode *black box*, *tester* tidak perlu mengetahui bagaimana struktur dan desain data yang ada di dalam sistem. Mereka hanya melihat apakah sistem terjadi *bugs* atau tidak (Tim Riley, 2010, p. 270).