

BAB II

TINJAUAN PUSTAKA DAN DASAR TEORI

2.1 Tinjauan Pustaka

Dalam penelitian ini mengambil beberapa penelitian yang pernah dilakukan sebelumnya sebagai acuan dan pedoman dalam membangun aplikasi. Berikut adalah beberapa sumber acuan tersebut.

Tinjauan pustaka yang pertama diperoleh dari Mufid Ajidarma (2019) melakukan penelitian dengan judul “Aplikasi Perhitungan Kebutuhan Kalori dan Perhitungan Kalori Dari Makanan Yang Dikonsumsi”. Aplikasi dibuat dengan menggunakan bahasa pemrograman HTML, CSS, PHP dan MySQL. Aplikasi ini dapat digunakan untuk mengetahui kebutuhan kalori dan sebagai pedoman dalam menentukan pola makan yang sehat dan seimbang.

Tinjauan pustaka yang kedua diperoleh dari Randy Pradana Kushatmaja (2017) melakukan penelitian dengan judul “Rancang Bangun Perangkat Lunak Penghitung Kalori dan Pengatur Pola Makan Pada Sistem Operasi Android”. Penelitian ini membangun sebuah aplikasi android yang digunakan untuk penghitung kalori dan pengatur pola makan sehingga pola hidup sehat dapat terjaga.

Tinjauan pustaka yang ketiga diperoleh dari Agung Prajuhana Putra, Sufiatul Maryana, Muhammad Iqbal Suriansyah (2020) melakukan penelitian dengan judul “Panduan Makanan Sehat untuk Diabetes Mellitus Berbasis Android”. Penelitian ini membangun sebuah aplikasi android yang digunakan untuk mendapatkan informasi resep makanan dan menjaga asupan kalori per harinya untuk penderit

diabetes melitus meliputi resep makanan pagi, siang dan malam. Kemudian diuraikan bahan apa saja yang digunakan dan berapa jumlah asupan kalori yang didapat beserta gambar dari makanan tersebut.

Tinjauan pustaka yang keempat diperoleh dari Tri Sudaryono (2017) melakukan penelitian dengan judul “Rancang Bangun Alat Pengukur Kalori Makanan Pintar Berbasis *Smartphone* Android”. Penelitian ini merancang dan membangun sebuah alat ukur timbangan sebagai alat ukur untuk menghitung jumlah kalori yang ada di setiap makanan. Proses pengukuran dikendalikan melalui *smartphone* android. Hasil pengukuran akan menentukan apakah kalori yang dibutuhkan sekali makan sudah sesuai atau kurang atau bahkan kelebihan.

Usulan yang akan diajukan dengan judul “Aplikasi Penghitung Kadar Gula Pada Makanan Berbasis Android”. Penelitian ini membangun sebuah aplikasi android yang digunakan menampilkan kadar gula pada makanan, menghitung total kadar gula dalam makanan dan mencatat kondisi maupun kegiatan harian pengguna aplikasi atau penderita diabetes mellitus. Aplikasi ini bisa memudahkan pengguna baik yang menderita penyakit diabetes melitus ataupun yang tidak, untuk mengontrol dan memantau kadar gula darah.

Perbandingan tinjauan pustaka bisa dilihat dalam tabel 2.1 di bawah ini.

Tabel 2.1 Perbandingan Tinjauan Pustaka

No	Nama Peneliti	Topik	Metode	Teknologi /Interface
1.	Mufid Ajidarma (2019)	Aplikasi Perhitungan Kebutuhan Kalori	<i>Waterfall</i> (Pengembangan)	HTML, CSS

		Dan Perhitungan Kalori Dari Makanan Yang Dikonsumsi	dan <i>Black-Box (Testing)</i>	MySQL, PHP/Web
2.	Randy Pradana Kushatmaja (2017)	Rancang Bangun Perangkat Lunak Penghitung Kalori dan Pengatur Pola Makan Pada Sistem Operasi Android	Metode Deskriptif Kualitatif	Java, Bootstrap/Android
3.	Agung Prajuhana Putra, Sufiatul Maryana & Muhammad Iqbal Suriansyah (2020)	Panduan Makanan Sehat untuk Diabetes Mellitus Berbasis Android	<i>Rule Based System</i> (Sistem Berbasis Aturan)	Java/Android
4.	Tri Sudaryono (2017)	Rancang Bangun Alat Pengukur Kalori Makanan Pintar Berbasis Smartphone Android	<i>Waterfall</i> (Pengembangan)	Java, Arduino/Android
5.	Penelitian Saat Ini.	Aplikasi Penghitung Kadar Gula Darah Berbasis Android	<i>Waterfall</i> (Pengembangan), <i>Black-Box</i> dan <i>White-Box (Testing)</i>	Dart, MySQL, Laravel 7/Android

2.2 Dasar Teori

2.2.1 Diabetes Melitus dan Kadar Gula Darah

Diabetes melitus atau kencing manis adalah penyakit kronis yang disebabkan oleh warisan dari orang tua dan atau kekurangan produksi insulin oleh pankreas, atau oleh tidak efektifnya insulin yang diproduksi. Akibatnya, terjadi peningkatan konsentrasi glukosa dalam darah, yang pada akhirnya akan merusak banyak sistem tubuh, khususnya pembuluh darah dan saraf. Diabetes melitus terdiri dari dua tipe yaitu tipe 1 dan tipe 2. Namun, diabetes melitus tipe terjadi lebih umum dan menyumbang 90% dari semua kasus diabetes di dunia. (Profil Kesehatan DIY, 2019).

Faktor- faktor yang dapat memicu terjadinya diabetes melitus pada diri seseorang adalah konsumsi makanan dan minuman yang tidak seimbang (tinggi kalori, rendah serat, atau *fast food*), jarang berolahraga, kegemukan, stres, dan istirahat yang tidak teratur. (Nur Rasdianah, 2016).

Diabetes tipe 1 dapat berkembang dengan cepat dalam beberapa minggu, bahkan beberapa hari saja. Sedangkan pada diabetes tipe 2, banyak penderitanya yang tidak menyadari bahwa mereka telah menderita diabetes selama bertahun-tahun, karena gejalanya cenderung tidak spesifik. Beberapa ciri-ciri diabetes tipe 1 dan tipe 2 meliputi sering merasa haus, sering buang air kecil, sering merasa sangat lapar, turunnya berat badan tanpa sebab yang jelas, lemas, pandangan kabur dan luka yang sulit sembuh. (alodokter.com).

Orang yang menderita diabetes perlu memperhatikan faktor-faktor penting dalam mengontrol kadar gula darahnya seperti makanan yang dikonsumsi,

kegiatan fisik atau olahraga, dan juga pengecekan kadar gula darah secara rutin. Penderita diabetes juga perlu mengetahui apakah gula darahnya berada pada kadar yang rendah, normal atau tinggi.

Salah satu faktor penting yang mempengaruhi kadar gula adalah makanan. Makanan mengandung glukosa yang akan berguna bagi tubuh sebagai sumber energi. Semua makanan dan minuman yang dikonsumsi umumnya akan mempengaruhi kadar gula darah dalam tubuh. Orang yang menderita diabetes perlu memperhatikan makanan yang masuk agar gula darah tetap stabil dan terhindar dari komplikasi diabetes. Selain itu, perlu juga mengetahui makanan yang ingin dikonsumsi. Umumnya, penderita diabetes dianjurkan untuk mengonsumsi makanan yang memiliki indeks glikemik rendah. Indeks glikemik (IG) adalah ukuran yang menunjukkan seberapa cepat makanan dapat meningkatkan gula darah dalam tubuh. Makanan dengan indeks glikemik rendah berarti membutuhkan waktu lebih lama untuk diproses menjadi glukosa di dalam tubuh. Dengan begitu, gula darah akan cenderung relatif stabil. (hellosehat.com)

Kadar gula darah darah sendiri merupakan istilah yang mengacu kepada tingkat gula darah di dalam darah. Konsentrasi gula darah, atau tingkat glukosa serum, diatur dengan ketat di dalam tubuh. Seseorang dikatakan menderita diabetes jika memiliki kadar gula darah puasa >126 mg/dL dan pada uji sewaktu >200 mg/dL.⁵ Kadar gula darah sepanjang hari bervariasi dimana akan meningkat setelah makan dan kembali normal dalam waktu 2 jam. (Adriansyah L. Putra, 2015). Untuk mengetahui kadar gula dalam darah terdapat beberapa tes yang bisa digunakan.

- Tes gula darah sewaktu

Tes gula darah sewaktu dilakukan untuk mengetahui kadar gula darah saat itu juga. Jenis tes gula darah ini bisa dilakukan kapan saja dengan pengambilan sampel darah. (sehatq.com)

Tabel 2.2 Tabel Kategori Kadar Gula Sewaktu

Kadar Gula	Kategori
<70 mg/dL	Rendah
70-200 mg/dL	Normal
>200 mg/dL	Tinggi/Risiko diabetes

- Tes gula darah puasa

Tes gula darah puasa harus dilakukan setelah seseorang berpuasa terlebih dulu. Umumnya pemeriksaan ini dilakukan pada pasien dengan kondisi prediabetes dan diabetes. Biasanya, seseorang dianjurkan untuk berpuasa sekitar 8-10 jam sebelum melakukan pemeriksaan gula darah puasa. (sehatq.com)

Tabel 2.3 Tabel Kategori Kadar Gula Puasa

Kadar Gula	Kategori
<70 mg/dL	Rendah
70-100 mg/dL	Normal

100–125 mg/dL	Prediabetes
≥ 126 mg/dL	Tinggi/Risiko diabetes

2.2.2 Android

Android adalah sistem operasi seluler yang dikembangkan oleh Google. Ini digunakan oleh beberapa *smartphone* dan *tablet*. Sistem operasi (OS) android didasarkan pada kernel Linux. Android adalah sistem operasi *open-source*, artinya android menyediakan *platform* terbuka bagi para pengembang dalam menciptakan aplikasi mereka sendiri untuk digunakan oleh berbagai macam perangkat *mobile*. Oleh karena itu, ponsel berbasis android seringkali memiliki antarmuka yang berbeda meskipun mereka menggunakan OS yang sama. Dalam pengembangan aplikasi android terdapat beberapa pilihan bahasa pemrograman yaitu Java, Kotlin, Python, C# dan Dart.

2.2.3 Flutter SDK

Flutter SDK merupakan perangkat *User Interface (UI) open-source* lintas *platform* yang dikembangkan oleh Google yang memungkinkan pengembang untuk mengembangkan aplikasi iOS, Android, Web (versi *beta*) dan Desktop (versi pengembangan) dengan basis kode yang sama. Sasaran pengembangannya adalah untuk memberikan aplikasi berkinerja tinggi di berbagai *platform* dan sangat cocok digunakan untuk mengembangkan aplikasi yang memerlukan tampilan yang indah. Fitur-fitur yang dimiliki oleh Flutter adalah :

1. *Hot reload* yang berjalan di Dart *Virtual Machine* untuk melihat perubahan tanpa melakukan *compile* ulang.
2. Setiap *project* flutter mempunyai sebuah *file* bernama *pubspec.yaml* atau sering juga disebut *pubspec*. *Pubspec file* menentukan *dependencies* yang dibutuhkan oleh sebuah *project* flutter seperti *package-package* tertentu (berserta versi-versinya), *font* dan juga gambar.
3. Kerangka kerja yang reaktif.
4. Kaya akan *widget* yang menerapkan *material design*.
5. Dart devtools untuk menguji dan men-*debug* aplikasi.

Flutter menggunakan bahasa pemrograman Dart yang bisa mudah dipahami jika sudah mengenal konsep pemrograman berorientasi objek (kelas, metode dan variabel) dan konsep pemrograman imperatif (perulangan dan kondisional). (docs.flutter.dev, 2020).

2.2.4 Framework Laravel

Laravel adalah *framework* aplikasi web kontemporer, *open-source* dan digunakan secara luas untuk perancangan aplikasi web yang cepat dan mudah. Laravel dibuat oleh Taylor Otwell dan berada dibawah lisensi MIT. *Framework* ini dibuat untuk pengembangan website mengikuti konsep MVC (Model, View, Controller). Konsep MVC merupakan suatu konsep yang memisahkan pengembangan aplikasi berdasarkan komponen- komponen utama yang membangun sebuah aplikasi seperti manipulasi data, *user interface* dan cara memproses datanya. Komponen pola MVC terdiri dari :

1. Model mewakili struktur data pada sistem aplikasi. Model berisi fungsi-

fungsi yang membantu dalam memanipulasi data di dalam basis data (*insert, update, delete, search*).

2. View adalah bagian yang mengatur tampilan halaman web atau *user interface*.
3. Controller merupakan bagian yang menghubungkan model dan *view*. Controller berisi perintah-perintah yang berfungsi untuk memproses suatu data dan mengirimkannya ke halaman web.

Laravel mempunyai sintaks yang ekspresif, jelas dan menghemat waktu. *Framework* ini dikembangkan dengan tujuan bahwa pengembangan web harus dapat dinikmati dan penuh kreatifitas. Pengembangan web dengan Laravel mempermudah proses pengembangan web dengan mempermudah tugas-tugas yang umum seperti *routing, authentication, sessions, dan caching*. (B. Hermanto, 2019).

2.2.5 MySQL

MySQL adalah sebuah perangkat lunak sistem manajemen basis data SQL (*Database Management System*) atau DBMS yang *multithread, multi-user*, dengan sekitar 6 juta instalasi di seluruh dunia. MySQL adalah perangkat lunak yang merupakan sebuah implementasi dari sistem manajemen basis data relasional (RDBMS - *Relational Database Management System*) yang didistribusikan secara gratis dibawah lisensi GPL (*General Public License*). Setiap pengguna dapat secara bebas menggunakan MySQL, namun dengan batasan perangkat lunak tersebut tidak boleh dijadikan turunan yang bersifat komersial.

Pada berbagai kepentingan, MySQL memiliki fungsi dan kegunaan yang sangat penting dalam menangani data. Sebagai contoh, MySQL dapat menangani persediaan barang dan mencatat data penjualan dalam sistem toko *online* (*e-commerce*). Dengan adanya sistem MySQL maka manajemen toko *online* dapat mengetahui persediaan barang yang ada didalamnya apakah masih tersedia atau kosong. Selain itu MySQL dapat memberikan laporan berupa informasi penjualan baik secara harian, mingguan, bulanan bahkan sampai tahunan. (YM Kusuma Ardhana., S.T., M.Kom., 2014).

2.2.6 Metode Waterfall

Metode penelitian yang diterapkan pada penelitian ini adalah dengan pengembangan metode *waterfall*. Metode *waterfall* merupakan model pengembangan sistem informasi yang sistematis dan sekuensial. Metode *waterfall* memiliki tahapan-tahapan sebagai berikut :

1. *Requirements analysis and definition*

Layanan sistem, kendala, dan tujuan ditetapkan oleh hasil konsultasi dengan pengguna yang kemudian didefinisikan secara rinci dan berfungsi sebagai spesifikasi sistem.

2. *System and software design*

Tahapan perancangan sistem mengalokasikan kebutuhan-kebutuhan sistem baik perangkat keras maupun perangkat lunak dengan membentuk arsitektur sistem secara keseluruhan. Perancangan perangkat lunak melibatkan identifikasi dan penggambaran abstraksi sistem dasar perangkat lunak dan hubungannya.

3. *Implementation and unit testing*

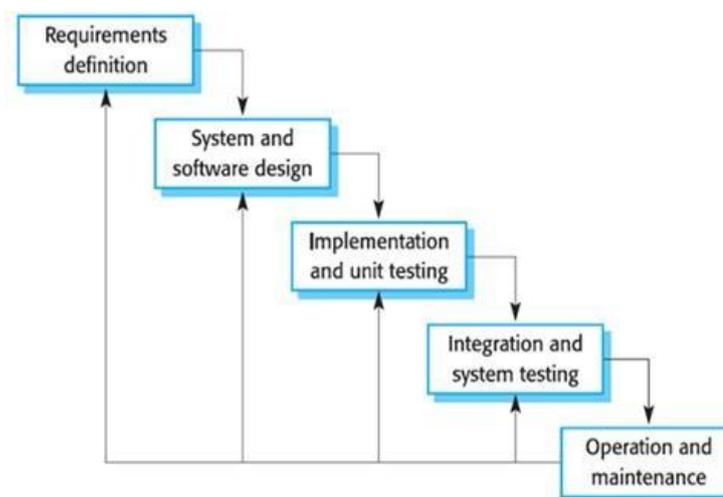
Pada tahap ini, perancangan perangkat lunak direalisasikan sebagai serangkaian program atau unit program. Pengujian melibatkan verifikasi bahwa setiap unit memenuhi spesifikasinya.

4. *Integration and system testing*

Unit-unit individu program atau program digabung dan diuji sebagai sebuah sistem lengkap untuk memastikan apakah sesuai dengan kebutuhan perangkat lunak atau tidak. Setelah pengujian, perangkat lunak dapat dikirimkan ke *customer*.

5. *Operation and maintenance*

Biasanya (walaupun tidak selalu), tahapan ini merupakan tahapan yang paling panjang. Sistem dipasang dan digunakan secara nyata. *Maintenance* melibatkan pembetulan kesalahan yang tidak ditemukan pada tahapan-tahapan sebelumnya, meningkatkan implementasi dari unit sistem, dan meningkatkan layanan sistem sebagai kebutuhan baru.



Gambar 2.1 Bagan Metode Waterfall

Gambar 2.1 adalah bagan metode *waterfall* yang merupakan metode pengembangan sistem yang digunakan pada penelitian ini. (Ginjar Wiro Sasmito, 2017).

2.2.7 Black-Box Testing

Pada *Black-box Testing* dilakukan pengujian yang didasarkan pada detail aplikasi seperti tampilan aplikasi, fungsi-fungsi yang ada pada aplikasi, dan kesesuaian alur fungsi dengan bisnis proses yang diinginkan oleh *customer*. *Black-box Testing* ini lebih menguji ke tampilan luar (*interface*) dari suatu aplikasi agar mudah digunakan oleh *customer*. Pengujian ini tidak melihat dan menguji *source code* program. *Black-box Testing* bekerja dengan mengabaikan struktur *control* sehingga perhatiannya hanya terfokus pada informasi domain. (B. Hermanto, 2019).

Keuntungan penggunaan metode *Black-box Testing* adalah :

1. Penguji tidak perlu memiliki pengetahuan tentang bahasa pemrograman tertentu.
2. Pengujian dilakukan dari sudut pandang pengguna, ini membantu untuk mengungkapkan ambiguitas atau inkonsistensi dalam spesifikasi persyaratan.
3. *Programmer* dan *tester* keduanya saling bergantung satu sama lain.

Kekurangan dari metode *Black-box Testing* adalah :

1. Kemungkinan memiliki pengulangan tes yang sudah dilakukan oleh *programmer*.

2. Beberapa bagian *back-end* tidak diuji sama sekali. (Tri Snadhika Jaya, 2018).

2.2.8 White-Box Testing

White-box Testing adalah salah satu cara untuk menguji suatu aplikasi atau *software* dengan melihat modul untuk memeriksa dan menganalisis kode *program* ada yang salah atau tidak. Jika modul ini dan telah diproduksi dalam *output* yang tidak memenuhi persyaratan, kode akan dikompilasi ulang dan diperiksa lagi sampai mencapai apa yang diharapkan, singkatnya *White-box Testing* ini menguji dengan cara melihat *source code* dari suatu aplikasi (*software*) yang diuji tanpa memperdulikan tampilan antar muka atau UI (*User Interface*) dari aplikasi tersebut. (B. Hermanto, 2019).

White-box Testing merupakan metode desain uji kasus yang menggunakan struktur kontrol dari desain prosedural untuk menghasilkan kasus-kasus uji. Ketika program diterjemahkan ke dalam kode sumber bahasa pemrograman, maka akan terjadi kesalahan pengetikan. Banyak yang terdeteksi dengan mekanisme pemeriksaan sintaks, tetapi banyak juga yang tidak terdeteksi sampai dengan dimulainya uji coba. Karena alasan tersebut, maka uji coba *White-box Testing* diperlukan selain *Black-box Testing*. (Muhammad Nuris, 2015)