

BAB 2

TINJAUAN PUSTAKA DAN DASAR TEORI

2.1. Tinjauan Pustaka

Penelitian yang dilakukan oleh Irma Eka Ayu Novita dan Hernawan Sulistyanto (2015) dengan judul Pengembangan Aplikasi Untuk Mengetahui Kebutuhan Jumlah Kalori dikembangkan dengan menggunakan rumus *Harris-Benedict* serta rumus *BMI*(*Body Mass Index*) dengan hasil keluaran status gizi, kebutuhan kalori, berat badan ideal, serta saran menu makanan. Aplikasi tersebut adalah aplikasi web yang dibuat dengan menggunakan bahasa pemrograman *PHP* dan *database MySQL*.

Adapun penelitian yang dilakukan oleh Andika Dwi Saputro (2013) yang berjudul Aplikasi Penghitung Kebutuhan Kalori Harian Untuk Penderita Obesitas Berbasis Android dibuat dengan bahasa pemrograman *eclips* dan *database SQLite*. Aplikasi tersebut dikembangkan dengan menggunakan rumus *Formula Ireton-Jones* dan *BMI*.

Sedangkan aplikasi yang akan saya buat ini berjudul Sistem Informasi pada Club Happy Fams Berbasis Web Mobile. Sistem informasi tersebut dibuat berdasarkan aturan bisnis Club Happy Fams yaitu sebuah club kebugaran dan penurunan berat badan. Pada sistem informasi ini akan membahas tentang pendaftaran anggota club, pengelolaan laporan customer, penghitungan jumlah kalori per hari, penghitungan perkiraan jumlah kalori makanan dan riwayat perkembangan konsultasi yang bisa diakses oleh *customer*. Aplikasi ini dibuat dengan menggunakan rumus *Harris-Benedict* serta menggunakan bahasa pemrograman *PHP* dan *database MySQL*.

2.2. Dasar Teori

Dasar teori yang mendukung pembuatan sistem ini adalah sebagai berikut:

2.2.1. Penghitungan Jumlah Kebutuhan Kalori Per Hari

Perhitungan untuk menghitung jumlah kebutuhan kalori per hari pada sistem ini menggunakan rumus *Harris-Benedict*.

Rumus Harris-Benedict memperhitungkan usia, jenis kelamin, berat badan, tinggi badan, hingga level aktivitas fisik.

Skala level aktifitas yang digunakan adalah :

No.	Level Aktivitas	Skala
1.	Kerja ringan	0,2
2.	Kerja sedang	0,3
3.	Kerja berat	0,4
4.	Kerja sangat berat	0,5

Rumus untuk menghitung jumlah kebutuhan kalori per hari adalah:

Kebutuhan kalori = *Basal Metabolic Rate (BMR)* + Aktifitas + Efek termik makanan

- Rumus untuk menghitung *Basal Metabolic Rate* pria yaitu:

$$BMR = 66 + (13,8 \times (\text{berat badan dalam kilogram})) + (5 \times (\text{tinggi badan dalam cm})) - (6,8 \times \text{usia})$$
- Rumus untuk menghitung *Basal Metabolic Rate* wanita yaitu:

$$BMR = 65,5 + (13,8 \times (\text{berat badan dalam kilogram})) + (5 \times (\text{tinggi badan dalam cm})) - (6,8 \times \text{usia})$$

Aktifitas = level aktifitas x BMR

Efek termik makanan = 10%(BMR + Aktifitas)

Contoh :

- Amira adalah perempuan berumur 25 tahun dengan tinggi badan 160 cm dan berat badan 60 kg. Setiap harinya Amira melakukan aktifitasnya sebagai pelayan restoran. Berapa kebutuhan angka kecukupan gizi per hari Amira ?

Penghitungan :

$$\begin{aligned} \text{BMR} &= 65,5 + (13,8 \times 60) + (5 \times 160) - (6,8 \times 25) \\ &= 1.523,5 \end{aligned}$$

$$\begin{aligned} \text{Aktifitas} &= 0,3 \times 1.523,5 \\ &= 457,05 \end{aligned}$$

$$\begin{aligned} \text{Efek termik} &= 10 \% \times (1.523,5 + 457,05) \\ &= 198.055 \end{aligned}$$

$$\begin{aligned} \text{Akg} &= \text{BMR} + \text{aktifitas} + \text{efek termik} \\ &= 1.523,5 + 457,05 + 198,055 \\ &= 2.178,605 \text{ Kkal} \end{aligned}$$

Jadi jumlah kebutuhan kalori per hari Amira adalah 2.178,605 Kkal

2.2.1. Pengertian Sistem Informasi

Sistem informasi adalah suatu sistem di dalam suatu organisasi yang mempertemukan kebutuhan pengolahan transaksi harian, mendukung operasi, bersifat manajerial dan kegiatan strategis dari suatu organisasi dan menyediakan pihak luar tertentu dengan laporan-laporang yang diperlukan (Sutabri,2004 : 36).

2.2.2. Internet

Menyebutkan bahwa internet merupakan singkatan atau kependekan dari *international network*, yang didefinisikan sebagai suatu jaringan komputer yang sangat besar, dimana jaringan komputer tersebut terdiri dari beberapa jaringan – jaringan kecil yang saling terhubung satu sama lain. (Oetomo, 2002). Lebih lanjut dijelaskan pula, jaringan komputer yang sangat besar ini bisa mencakup jaringan seluruh dunia.

2.2.6. HTML

HTML merupakan salah satu *form* yang digunakan dalam pembuatan dokumen dan aplikasi yang berjalan di halaman *web*. (Suyanto, 2007c:23).

2.2.7. MySQL

MySQL merupakan perangkat lunak yang digunakan untuk membangun database yang sering digunakan di lingkungan linux. MySQL merupakan *software open source* yang berarti *free* untuk digunakan. Selain di lingkungan linux, MySQL juga tersedia di lingkungan windows. (Sulhan, 2007:118).

Perintah MySQL yang umum digunakan adalah perintah dasar yaitu *cread*, *read*, *update* dan *delete*. Perintah-perintah dasar tersebut dalah sebagai berikut:

a. Perintah *Create*

Perintah *create* digunakan untuk membuat *database* dan tabel.

Perintahnya adalah

Untuk membuat *database*

```
CREATE nama_database;
```

Untuk membuat tabel

```
CREATE nama_tabel (
Nama_field type_data(panjang_data),
Nama_field type_data(panjang_data),
.....
```

```
) ;
```

b. Perintah *Read*

Perintah tersebut digunakan untuk mengambil data dari *database*.

Perintahnya adalah:

```
SELECT nama_field FROM nama_tabel;
```

c. Perintah *Update*

Perintah update digunakan untuk mengedit data pada *database*.

Perintahnya adalah:

```
UPDATE    nama_tabel    SET    nama_field=data_baru
WHERE kondisi;
```

2.2.8. PHP

PHP adalah suatu bahasa *server side* yang didesain khusus untuk aplikasi web. PHP dapat disisipkan di antara bahasa HTML dan arena bahasa *server side*, maka bahasa PHP akan dieksekusi di *server*, sehingga yang dikirim ke *browser* adalah “hasil jadi” dalam bentuk HTML dan kode PHP tidak akan terlihat. (Kadir, 2002).

Berikut beberapa perintah *query* SQL di PHP yang digunakan :

- a. `mysqli_query($sql)` digunakan untuk menerjemahkan proses *query* ke MySQL dari PHP.
- b. `mysqli_fetch_array($query)` digunakan untuk mengambil hasil *query* SQL agar dapat diolah dan dibaca oleh PHP.

2.2.9. Framework

Framework adalah sekumpulan *library* yang diorganisasikan pada sebuah rancangan arsitektur untuk memberikan kecepatan, ketepatan, kemudahan dan konsistensi didalam pengembangan aplikasi dari definisi tersebut. (Siena, 2009).

2.2.10. Bootstrap

Bootstrap merupakan sebuah *framework* *css* yang memudahkan pengembang untuk membangun website yang menarik dan responsif. Tidak konsistensinya terhadap aplikasi individual membuat sulitnya untuk mengembangkan dan pemeliharannya. Bootstrap adalah *css* tetapi dibentuk dengan *LESS*, sebuah *pre-processor* yang memberi fleksibilitas dari *css* biasa. Bootstrap memberikan solusi rapi dan seragam terhadap solusi yang umum, tugas *interface* yang setiap pengembang hadapi. Bootstrap dapat dikembangkan dengan tambahan lainnya karena ini cukup fleksibel terhadap pekerjaan desain yang dibutuhkan. (Otto, 2011).

Pengembangan interface yang responsif pada bootstrap menggunakan sistem grid. Sistem grid yang dipakai pada bootstrap terdiri dari dua belas grid dalam satu layar *device*.

Ketika mengunduh plugin bootstrap *css* dan *javascript* dihimpun dalam sebuah folder dengan nama *css* untuk kode-kode *css* dan *js* untuk kode-kode *javascript*. Penggunaan *css* dan *javascript* pada bootstrap dipanggil dalam tag `<head></head>` pada kode HTML dan menggunakan *class* pada komponen yang digunakan. Contoh penggunaan bootstrap adalah sebagai berikut:

Pemanggilan plugin *css* dan *javascript*

```
<head>
  <link rel="stylesheet" href="css/bootstrap.min.css"
type="text/css" media="all">
  <link rel="stylesheet" href="css/style.css" type="text/css"
media="all">
  <link rel="stylesheet" href="css/swipebox.css">
  <!-- //Custom-Theme-Files -->
  <!-- Web-Fonts -->
  <link rel="stylesheet"
href="//fonts.googleapis.com/css?family=Open+Sans:400,300,600,700,
800" type="text/css">
```

```

<link rel="stylesheet"
href="//fonts.googleapis.com/css?family=Montserrat:400,700"
      type="text/css">
<!-- //Web-Fonts -->
<!-- Default-JavaScript-File -->
<script type="text/javascript" src="js/jquery-
2.1.4.min.js"></script>
<script type="text/javascript"
src="js/bootstrap.min.js"></script>
</head>

```

Penggunaan komponen bootstrap

```

<form class="navbar-form navbar-left" role="search">
  <div class="form-group">
    <input type="text" class="form-control" placeholder="Search">
  </div>
  <button type="submit" class="btn btn-default">Submit</button>
</form>

```

2.2.11. DAD Level 0

Diagram Alir Data (DAD) atau **Data Flow Diagram (DFD)** adalah suatu diagram yang menggunakan notasi-notasi untuk menggambarkan arus dari data sistem, yang penggunaannya sangat membantu untuk memahami sistem secara logika, tersruktur dan jelas. DFD merupakan alat bantu dalam menggambarkan atau menjelaskan DFD ini sering disebut juga dengan nama Bubble chart, Bubble diagram, model proses, diagram alur kerja, atau model fungsi.

Simbol-simbol dalam DAD :

1. Terminator



Gambar 2.1 Simbol Terminator

Gambar 2.1 menunjukkan simbol terminator. Terminator dapat berupa orang, sekelompok orang, organisasi, departemen di dalam

organisasi, atau perusahaan yang sama tetapi di luar kendali sistem yang sedang dibuat modelnya. Terminator dapat juga berupa departemen, divisi atau sistem di luar sistem yang berkomunikasi dengan sistem yang sedang dikembangkan.

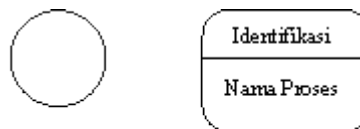
2. Arus



Gambar 2.2 Simbol Arus

Gambar 2.2 merupakan simbol arus. Arus data ini mengalir diantara proses (Process), simpanan data (data store) dan kesatuan luar (external entity).

3. Proses



Gambar 2.3 Simbol Proses

Gambar 2.3 merupakan simbol proses. Suatu proses adalah kegiatan atau kerja yang dilakukan oleh orang, mesin, atau komputer dan hasil suatu arus data yang masuk ke dalam proses untuk dilakukan arus data yang akan keluar dari proses.