

BAB II

LANDASAN TEORI

2.1. Penelitian Terdahulu

Pada penelitian ini peneliti membuat judul penelitian dengan nama “Sistem Informasi Pelayanan Membership Pada Zen Body Gym Berbasis Android”. Penelitian terdahulu yang peneliti cantumkan di bawah ini adalah sebagai acuan sehingga dapat memperkaya teori peneliti dalam mengkaji penelitian yang dilakukan. Berikut ini adalah penelitian yang peneliti kaji sebelumnya.

Penelitian yang dilakukan oleh Yuni Romauli Tobing dengan judul “Perancangan Sistem Informasi Membership Pada Vitka Fitness Berbasis Web”. Penelitian ini bertujuan untuk menghilangkan kesulitan dalam pengelolaan data member terutama pendaftaran member baru, perpanjang member, kunjungan dan pembuatan laporan [2].

Pada penelitian yang dilakukan oleh Yuni Romauli Tobing memiliki perbedaan dengan penelitian yang dilakukan oleh penulis yaitu tidak terdapat fasilitas penjadwalan pemilihan personal trainer, transaksi pembayaran hanya datang ke tempat.

Selanjutnya, penelitian yang dilakukan oleh Mohamad Rohmat dengan judul penelitian “Sistem Informasi Pelayanan Fitness Pada Rebel Gym Bandung Berbasis Desktop”. Penelitian ini bertujuan untuk membantu pegawai dalam melakukan proses pendaftaran member baru, penyewaan alat fitness, penyewaan instruktur,

kunjungan non member dan member, serta mempermudah pegawai dalam pembuatan laporan yang sebelumnya dilakukan secara manual [3].

Pada penelitian yang dilakukan oleh Mohamad Rohmat memiliki perbedaan dengan penelitian yang dilakukan oleh penulis yaitu penelitian ini memakai metode pendekatan terstruktur dan pada penelitian ini juga membahas penyewaan alat fitness, penjualan tiket untuk pemilihan kelas – kelas fitness, dan aplikasi ini bersifat *offline*.

2.2. Teori-teori Dasar

Berikut ini adalah istilah-istilah atau teori-teori dasar dalam penelitian ini yang tentunya harus peneliti ketahui dan mengerti guna untuk melakukan penelitian dengan baik.

2.2.1. Definisi Sistem

Sistem adalah suatu jaringan kerja dari prosedur-prosedur yang saling berhubungan, berkumpul bersamasama untuk melakukan suatu kegiatan atau untuk menyelesaikan suatu sasaran yang tertentu [4].

2.2.2. Sistem Informasi

Sistem informasi adalah suatu sistem dalam suatu organisasi yang mempertemukan kebutuhan pengolahan transaksi harian yang mendukung fungsi operasi organisasi yang bersifat manajerial dengan kegiatan strategi dari suatu organisasi untuk dapat menyediakan kepada pihak luar tertentu dengan informasi yang diperlukan untuk pengambilan keputusan [4].

2.2.3. Definisi *Fitness*

Fitness adalah aktivitas yang mampu membuat orang menjadi lebih bugar dengan olahraga latihan angkat beban (*weight lifting*), aerobik dan pemenuhan nutrisi [5].

Fitness olahraga yang paling efektif untuk menurunkan berat badan dan membentuk otot agar lebih berisi dan *sixpack*. Pengertian *fitnes* itu sendiri adalah olah raga untuk membakar lemak dengan difokuskan pada pembentukan otot tubuh dan juga bagian tubuh lainnya yang diinginkan. Pengertian ini merupakan pengertian secara umum. Olah raga kebugaran ini sangat bagus karena jika dilakukan rutin sangat baik untuk kesehatan tubuh. Tidak hanya dapat membentuk otot dan juga menurunkan berat saja, ternyata jika *fitnes* dilakukan rutin setiap hari tubuh akan menjadi sehat dan selalu bugar.

2.2.4. Sistem Informasi *Fitness*

Sistem Informasi *Fitness* adalah suatu sistem yang dirancang untuk keperluan pengelolaan data-data *Fitness* dengan penerapan teknologi komputer baik hardware maupun software. Hardware (perangkat keras) adalah peralatan-peralatan seperti komputer (PC maupun Laptop), Printer, CD ROM, Hard Disk, Handphone dan sebagainya. Sistem informasi *Fitness* merupakan suatu sistem berbasis komputer yang dirancang untuk keperluan pengelolaan data-data yang ada di pusat kebugaran, misalnya seperti pembayaran member dan non member, pengolahan dan penjadwalan sebagai bukti transaksi telah melakukan pembayaran [5].

2.2.5. Member *Fitness*

Member fitness adalah seseorang yang berlangganan menyewa sebuah tempat fitness sebagai sarana olahraga dalam jangka waktu tertentu. Sebelum seseorang terdaftar sebagai member fitness orang tersebut harus mendaftar pada tempat fitness tertentu untuk menjadi seorang member. Setiap member pada awal pendaftaran dikenakan biaya *join fee*, masa aktif *join fee* tersebut mengikuti masa aktif member tersebut di tambah satu bulan sebagai masa tenggang pendaftaran member [6].

2.2.6. Instruktur *Fitness*

Instruktur fitness adalah seseorang yang bertugas untuk memandu seseorang yang berlangganan menyewa sebuah tempat fitness sebagai sarana olahraga dalam jangka waktu tertentu. Sedangkan kegiatan yang dilakukan instruktur dalam memandu seseorang yang berlangganan menyewa sebuah tempat fitness sebagai sarana olahraga dalam jangka waktu tertentu disebut kegiatan instruktural fitness [4].

2.2.7. Android

Android adalah sebuah sistem operasi pada handphone yang bersifat terbuka dan berbasis Linux. Android menyediakan platform terbuka (*open source*) sehingga memudahkan bagi para pengembang untuk menciptakan aplikasi mereka sendiri android awalnya dikembangkan oleh *Android, Inc.*, dengan dukungan Google, yang kemudian android dibeli oleh Google pada tahun 2005 [7].

2.2.8. Web Server

Web server adalah sebuah bentuk server yang khusus digunakan untuk menyimpan halaman website atau homepage. Dengan kata lain web server merupakan bentuk server yang digunakan untuk menyimpan data-data dari website yang bisa dipanggil kapanpun oleh client. Web server menunggu permintaan dari client yang menggunakan browser seperti internet explorer, mozilla firefox, google chrome, dan program browser lainnya. Jika ada permintaan dari browser, maka webserver akan memproses permintaan kemudian memberikan hasil proses berupa data yang diinginkan kembali ke browser. [8].

2.2.9. Metode Pengembangan *Prototype*

Metode pengembangan perangkat lunak yang peneliti gunakan adalah *prototype*. *Prototype* ini metode pengembangan perangkat lunak yang berfokus untuk menghubungkan pemahaman kebutuhan pengguna dengan pengembang sehingga persepsi antara pengguna dan pengembang itu menjadi sama. Untuk dapat berinteraksi dalam proses kegiatan pengembangan maka metode *prototype* ini akan menghasilkan sebuah *prototype* sistem yang akan dibuat sebagai tahap awal dari sistem [9].

2.2.10. *Unified Modeling Language (UML)*

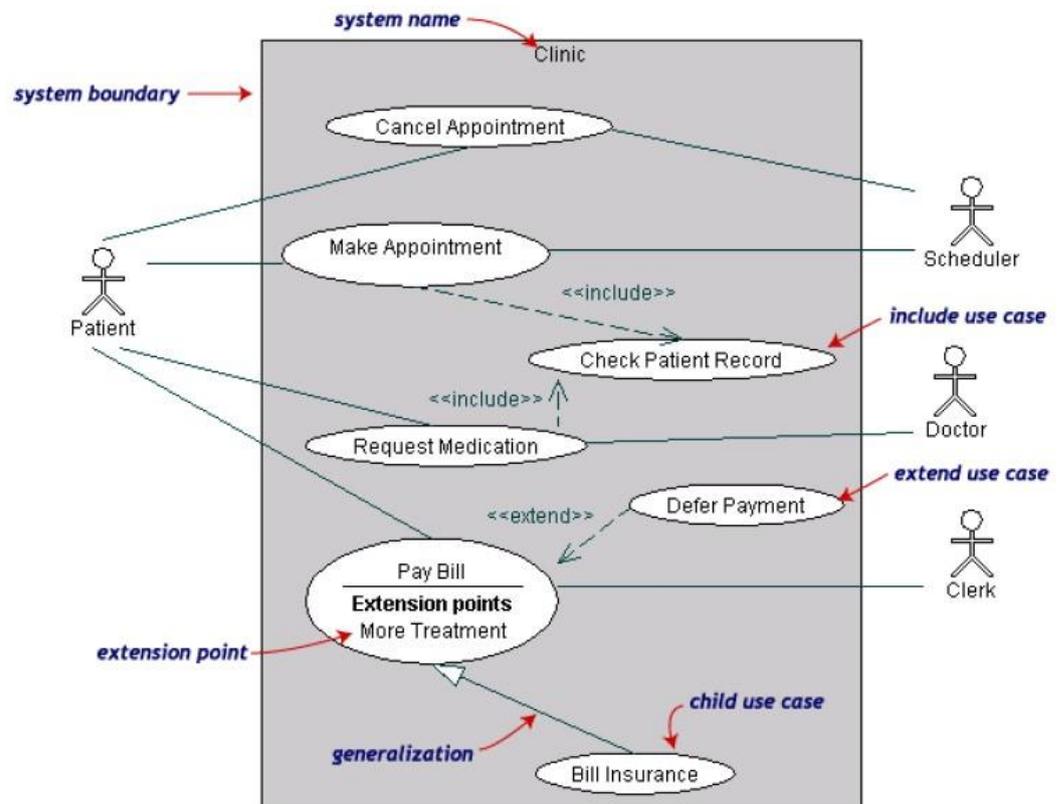
Unified Modelling Language (UML) adalah sebuah "bahasa" yg telah menjadi standar dalam industri untuk visualisasi, merancang dan mendokumentasikan sistem piranti lunak. UML menawarkan sebuah standar untuk merancang model sebuah sistem. Dengan menggunakan UML kita dapat membuat model untuk semua jenis aplikasi piranti lunak, dimana aplikasi tersebut dapat

berjalan pada piranti keras, sistem operasi dan jaringan apapun, serta ditulis dalam bahasa pemrograman apapun. Tetapi karena UML juga menggunakan *class* dan *operation* dalam konsep dasarnya, maka ia lebih cocok untuk penulisan piranti lunak dalam bahasabahasa berorientasi objek seperti C++, Java, C# atau VB.NET. Walaupun demikian, UML tetap dapat digunakan untuk modeling aplikasi prosedural dalam VB atau C [9]. UML mendefinisikan diagram-diagram sebagai berikut:

1. Use Case Diagram

Use case diagram menggambarkan fungsionalitas yang diharapkan dari sebuah sistem. Yang ditekankan adalah “apa” yang diperbuat sistem, dan bukan “bagaimana”. Sebuah *use case* merepresentasikan sebuah interaksi antara aktor dengan sistem. *Use case* merupakan sebuah pekerjaan tertentu, misalnya login ke sistem, meng-*create* sebuah daftar belanja, dan sebagainya. Seorang/sebuah aktor adalah sebuah entitas manusia atau mesin yang berinteraksi dengan system untuk melakukan pekerjaan-pekerjaan tertentu [10].

Contoh Use Case Diagram



Gambar 2. 1. Use Case Diagram

(Sumber : Pengantar Unified Modeling Language (UML). Ilmu Komputer [10, p.5])

2. Use Case Scenario

Use case scenario adalah suatu gambaran model yang berbentuk skenario sehingga nantinya ada beberapa kemungkinan-kemungkinan kejadian dalam menjalankan sebuah sistem tersebut. Kemungkinan-kemungkinan tersebut nantinya akan dipecah menjadi beberapa skenario. Di dalam proses sebuah sistem nantinya akan ada sekumpulan proses, proses inilah yang akan menjadi beberapa kemungkinan-kemungkinan yang kemudian bisa disebut dengan skenario use case [10].

3. *Activity Diagram*

Activity diagrams menggambarkan berbagai alir aktivitas dalam sistem yang sedang dirancang, bagaimana masing-masing alir berawal, *decision* yang mungkin terjadi, dan bagaimana mereka berakhir. *Activity diagram* juga dapat menggambarkan proses paralel yang mungkin terjadi pada beberapa eksekusi [10].

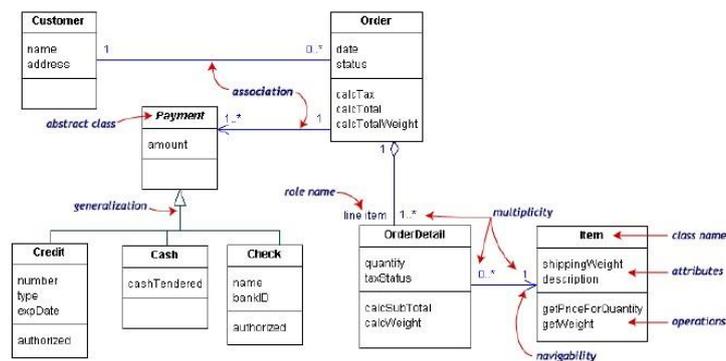
Activity diagram merupakan *state diagram* khusus, di mana sebagian besar *state* adalah *action* dan sebagian besar transisi di-*trigger* oleh selesainya *state* sebelumnya (*internal processing*). Oleh karena itu *activity diagram* tidak menggambarkan behaviour internal sebuah sistem (dan interaksi antar subsistem) secara eksak, tetapi lebih menggambarkan proses-proses dan jalur-jalur aktivitas dari level atas secara umum. Sebuah aktivitas dapat direalisasikan oleh satu *use case* atau lebih. Aktivitas menggambarkan proses yang berjalan, sementara *use case* menggambarkan bagaimana aktor menggunakan sistem untuk melakukan aktivitas [10].

4. *Class Diagram*

Class adalah sebuah spesifikasi yang jika diinstansiasi akan menghasilkan sebuah objek dan merupakan inti dari pengembangan dan desain berorientasi objek. *Class* menggambarkan keadaan (atribut/properti) suatu sistem, sekaligus menawarkan layanan untuk memanipulasi keadaan tersebut (metoda/fungsi). *Class diagram* menggambarkan struktur dan deskripsi *class*, *package* dan objek beserta hubungan satu sama lain seperti *containment*, pewarisan, asosiasi, dan

lain-lain. Hubungan yang digunakan untuk menggambarkan *class diagram* adalah sebagai berikut [10] :

1. Asosiasi, yaitu hubungan statis antar *class*. Umumnya menggambarkan *class* yang memiliki atribut berupa *class* lain, atau *class* yang harus mengetahui eksistensi *class* lain. Panah *navigability* menunjukkan arah *query* antar *class*.
2. Agregasi, yaitu hubungan yang menyatakan bagian (“terdiri atas..”).
3. Pewarisan, yaitu hubungan hirarkis antar *class*. *Class* dapat diturunkan dari *class* lain dan mewarisi semua atribut dan metoda *class* asalnya dan menambahkan fungsionalitas baru, sehingga ia disebut anak dari *class* yang diwarisinya. Kebalikan dari pewarisan adalah generalisasi.
4. Hubungan dinamis, yaitu rangkaian pesan (*message*) yang di-*passing* dari satu *class* kepada *class* lain. Hubungan dinamis dapat digambarkan dengan menggunakan *sequence diagram* yang akan dijelaskan kemudian.



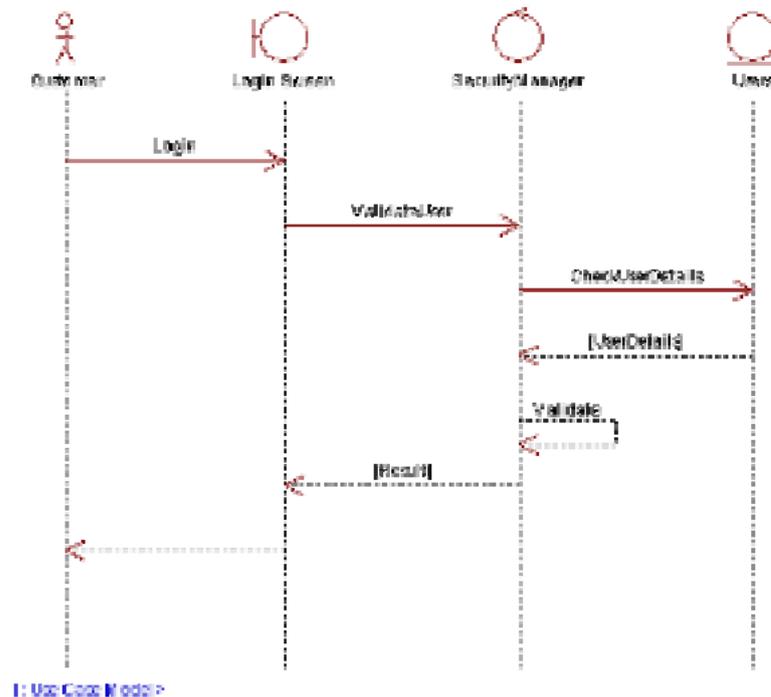
Gambar 2. 2. Class Diagram

(Sumber : Pengantar Unified Modeling Language (UML). *Ilmu Komputer* [10, p.6])

5. *Sequence Diagram*

Sequence diagram menggambarkan interaksi antar objek di dalam dan di sekitar sistem (termasuk pengguna, *display*, dan sebagainya) berupa *message* yang digambarkan terhadap waktu. *Sequence diagram* terdiri atas dimensi vertikal (waktu) dan dimensi horizontal (objek-objek yang terkait) [10].

Sequence diagram biasa digunakan untuk menggambarkan skenario atau rangkaian langkah-langkah yang dilakukan sebagai respons dari sebuah *event* untuk menghasilkan *output* tertentu. Diawali dari apa yang men-*trigger* aktivitas tersebut, proses dan perubahan apa saja yang terjadi secara internal dan *output* apa yang dihasilkan. Masing-masing objek, termasuk aktor, memiliki *lifeline* vertikal. *Message* digambarkan sebagai garis berpanah dari satu objek ke objek lainnya. Pada fase desain berikutnya, *message* akan dipetakan menjadi operasi/metoda dari *class*. *Activation bar* menunjukkan lamanya eksekusi sebuah proses, biasanya diawali dengan diterimanya sebuah *message*. Untuk objek-objek yang memiliki sifat khusus, standar UML mendefinisikan *icon* khusus untuk objek *boundary*, *controller* dan *persistent entity* [10].



Gambar 2. 3. Sequence Diagram

(Sumber : Pengantar Unified Modeling Language (UML). *Ilmu Komputer* [10, p.9])

2.3. Software yang Digunakan

Dalam penelitian ini peneliti menggunakan beberapa *software* untuk menunjang terselesaikannya penelitian ini dengan baik. Berikut ini adalah *software* yang peneliti gunakan untuk membangun dan mengembangkan aplikasi membership ini.

2.3.1. *Android Studio*

Android studio adalah IDE (*Integrated Development Environment*) resmi untuk pengembangan aplikasi Android dan bersifat *open source* atau gratis. Peluncuran *Android Studio* ini diumumkan oleh Google pada 16 mei 2013 pada *event Google I/O Conference* untuk tahun 2013. Sejak saat itu, *Android Studio*

mengantikan *Eclipse* sebagai *IDE* resmi untuk mengembangkan aplikasi Android [11].

2.3.2. *Firestore Realtime Database*

Firestore adalah BaaS (*Backend as a Service*) yang saat ini dimiliki oleh Google. *Firestore* sendiri merupakan solusi yang ditawarkan oleh Google untuk mempermudah pekerjaan *Mobile Apps Developer*. Dengan adanya *Firestore*, *apps developer* tentu saja dapat fokus mengembangkan aplikasi tanpa harus memberikan *effort* yang besar untuk urusan *backend* [11].

Firestore Realtime Database sendiri merupakan basis data berbasis *cloud NoSQL* yang menyinkronkan data di semua klien secara *realtime*, dan menyediakan fungsionalitas *offline*. Data yang diinputkan disimpan ke dalam *database Realtime* sebagai *JSON*. Semua klien yang terhubung dan saling berbagi dalam satu waktu, secara otomatis akan menerima pembaruan dengan data terbaru [11].



Gambar 2. 4. *Firestore Realtime Database*

(Sumber : Pengembangan Teknologi Mobile Untuk Sistem Kasir Rumah Makan di Kota Sampit Menggunakan *Firestore Realtime Database* [11])

2.3.3. Pemrograman Java

Java adalah nama sebuah bahasa pemrograman yang sangat terkenal. Sebagai bahasa pemrograman, java dapat digunakan untuk menulis program. Bahasa java dikembangkan di Sun Microsystem dan mulai diperkenalkan kepada public pada tahun 1995. Java merupakan bahasa yang berorientasi objek. Java mempunyai keunggulan yakni bersifat universal. Sebagai bahasa yang universal, java bias dijumpai di berbagai platform (Linux, UNIX, Windows, Mac, dan lain – lain). Hasil kompilasi java yang dinamakan *bytecode* dapat dijalankan di berbagai platform sepanjang di system target memiliki *Java Runtime Environment (JRE)* [12].



Gambar 2. 5. Java

(Sumber : Sistem Informasi Penjualan dan Pengendalian Persediaan Barang Menggunakan Metode Economic Order Quantity (EOQ) Menggunakan Bahasa Pemrograman Java dan Database MySQL Pada Toko Kansa Elpiji [12])

2.3.4. PHP

PHP merupakan bahasa pemrograman yang ditujukan untuk membuat program website dimana kode program yang telah dibuat di kompilasi dan dijalankan pada sisi server untuk menghasilkan halaman website yang dinamis. Pada awalnya PHP didirikan untuk Personal Home Page. Karena memiliki banyak manfaat dan dapat berkembang dengan baik maka PHP disebut dengan *hypertext Preprocessor* [13].

2.3.5. XAMPP

XAMPP merupakan paket PHP yang berbasis Open Source yang dikembangkan oleh sebuah komunitas Open Source. Dengan menggunakan XAMPP tidak perlu lagi bingung untuk melakukan penginstalan program-program yang lain, karena semua kebutuhan telah disediakan oleh XAMPP [13].



Gambar 2. 6. XAMPP

**(Sumber : SISTEM INFORMASI DATA BARANG INVENTARIS BERBASIS
WEB PADA KEJAKSAAN NEGERI TERNATE [13])**

2.3.6. Visual Studio Code

Visual studio code ini dibuat oleh Microsoft pada tahun 2015 dan sekarang sudah banyak yang menggunakannya. Editor ini keunggulannya bisa support dengan berbagai macam bahasa pemrograman.

Banyak yang menyukai editor ini karena plug-in yang bisa membantu dalam pengkodean program, salah satunya adanya support debugging, dan juga arsitektur editor yang sangat mudah dan enak digunakan sangat cocok. Berikut ini peneliti cantumkan keunggulan Visual Studio Code sebagai aplikasi editor yang terbaik menurut versi penulis yaitu :

1. Cross platform atau bisa digunakan di berbagai sistem operasi dari mulai windows, linux, macOS yang otomatis bisa memudah kita dalam melakukan pengkodean program dimanapun.
2. Lightweight adanya fitur ini membuat pengembang jadi lebih dimanja untuk bisa lebih mengontrol sepenuhnya tema yang digunakan, bahasa, commands dan juga debugger.
3. Editor yang sangat powerful karena memberikan fitur yang sangat begitu produktif, seperti adanya snippets, IntelliSense, autocorrect dan formatting.
4. Melakukan code debugging yang sangat mudah dengan cara mengawasi kode, variabel, expression dan juga call stack yang mana saja.
5. Control source code yang bisa terintegrasi seperti melakukan integrasi dengan Git. Tentu ini sangat memudahkan pengembang dalam melakukan controlling code bersama tim.

6. Dan yang terakhir adalah pengembang bisa membuat banyak terminal dalam satu editor, tentu ini menjadi keunggulan yang jarang ada di editor yang lain.

2.3.7. CSS (*Cascading Style Sheet*)

Cascading Style Sheet (CSS) merupakan salah satu bahasa pemrograman web untuk mengendalikan beberapa komponen dalam sebuah web sehingga akan lebih terstruktur dan seragam. Sama halnya styles dalam aplikasi pengolahan kata seperti Microsoft Word yang dapat mengatur beberapa style, misalnya heading, subbab, bodytext, footer, images, dan style lainnya untuk dapat digunakan bersama-sama dalam beberapa berkas (file). Pada umumnya CSS dipakai untuk memformat tampilan halaman web yang dibuat dengan bahasa HTML [14].

CSS dapat mengendalikan ukuran gambar, warna bagian tubuh pada *teks*, warna tabel, ukuran *border*, warna *border*, warna *hyperlink*, warna *mouse over*, spasi antar paragraf, spasi antar teks, margin kiri, kanan, atas, bawah, dan parameter lainnya. CSS adalah bahasa *style sheet* yang digunakan untuk mengatur tampilan dokumen. Dengan adanya CSS memungkinkan kita untuk menampilkan halaman yang sama dengan format yang berbeda [14].

2.3.8. *Template Bootstrap*

Penulis menggunakan *template bootstrap* sebagai tampilan halaman *website* yang dibuat agar lebih cepat dan lebih mudah karena *template*-nya dapat didownload secara gratis.

Bootstrap adalah *front-end framework* atau *framework* (kerangka kerja) yang digunakan untuk membuat tampilan *website* menjadi lebih mudah, lebih bagus, dan lebih menarik yang mengedepankan tampilan yang bisa dibuka dalam semua perangkat atau *device* baik itu *handphone*, *smartphone*, laptop atau komputer. Untuk mempermudah dan mempercepat pengembangan sebuah *website* maka *bootstrap* menyediakan HTML, CSS, dan *Javascript* yang siap pakai dan mudah digunakan untuk para pengembang *website*.