

## **BAB II**

### **TINJAUAN TEORI**

#### **2.1 Analisis Kebutuhan Sistem**

Kebutuhan sistem merupakan faktor-faktor yang dibutuhkan oleh sistem, sehingga sistem yang dirancang sesuai dengan tujuan perancangannya [6]. Sedangkan, analisis sistem merupakan salah satu tahapan penting setelah tahap perencanaan pada pengembangan sebuah sistem. Proses analisis sistem ini dilakukan untuk menentukan kebutuhan sistem yang akan dibangun [7]. Selain itu, analisis sistem merupakan sebuah proses menemukan kelemahan-kelemahan dari sebuah sistem yang sudah ada atau sistem yang sudah berjalan, sehingga informasi tersebut dapat digunakan sebagai perbaikan, perbandingan, dan peluang untuk membangun sistem baru yang akan dirancang [8].

#### **2.2 Aplikasi *Mobile***

Aplikasi *mobile* merupakan sebuah aplikasi perangkat lunak yang dirancang agar dapat dijalankan pada perangkat *smartphone*, sehingga pengguna dapat menggunakan aplikasi tersebut kapan saja [9]. Aplikasi *mobile* dirancang untuk meningkatkan pengalaman pengguna melalui integrasi yang optimal dengan perangkat keras dan konektivitas [10].

#### **2.3 *Benchmarking***

*Benchmarking* merupakan proses membandingkan suatu kinerja yang dilakukan secara sistematis dan berkelanjutan guna meningkatkan mutu suatu organisasi. Konsep ini pertama kali digunakan pada dunia bisnis, di mana perusahaan membandingkan diri mereka dengan perusahaan-perusahaan unggulan guna mempelajari praktik terbaik yang dapat diterapkan [11]. *Benchmarking* ini bukan sekadar meniru, tetapi juga memodifikasi dan mengadaptasi praktik terbaik untuk mencapai peningkatan berkelanjutan, serta memperoleh informasi yang dapat membantu perbaikan dan inovasi suatu produk, layanan, dan proses bisnis [12].

## 2.4 *Unified Modelling Language (UML)*

*Unified modelling language* atau biasa disingkat menjadi UML merupakan suatu metode atau pemodelan secara visual yang digunakan oleh pengembang *software* dan *system analyst* untuk merancang, memvisualisasikan, dan mendokumentasikan sistem guna membantu dalam pengembangan sistem yang lebih terstruktur dan terorganisir [13]. Pemodelan ini menggunakan notasi grafis dan simbol untuk menghubungkan berbagai aspek dari sistem, seperti struktur, perilaku, dan interaksi antar komponen [14].

## 2.5 *Use Case Diagram*

*Use case diagram* merupakan salah satu jenis diagram UML (*Unified Modelling Language*) yang berfungsi menggambarkan hubungan interaksi antara pengguna atau *actor* dan sistem yang dikembangkan [15]. *Use case diagram* dibuat untuk memvisualisasikan fungsi-fungsi utama yang ditawarkan oleh sistem dari sudut pandang pengguna [16].

## 2.6 *Activity Diagram*

*Activity diagram* atau diagram aktivitas adalah bagian dari UML yang merupakan pengembangan dari *use case diagram*. *Activity diagram* berfungsi untuk menunjukkan aliran aktivitas dalam sistem yang akan dijalankan dan mengorganisasikan aliran tampilan sistem. Pada *activity diagram* terdapat beberapa elemen model yang umum digunakan yaitu *activities*, *actions*, *control nodes*, *object nodes*, dan *activity edges* [17].

## 2.7 *Class Diagram*

*Class diagram* merupakan diagram yang digunakan untuk menunjukan interaksi antar *class* yang ada di dalam sistem yang akan dikembangkan dan sebuah spesifikasi yang apabila diinstansiasi akan menghasilkan *object* [18]. *Class diagram* terdiri atas atribut dan metode, atribut merupakan variable yang dimiliki oleh *class* dan memvisualisasikan keadaan suatu sistem, sedangkan *method* atau metode adalah fungsi yang dimiliki oleh *class* [19].

## 2.8 *Sequence Diagram*

*Sequence diagram* merupakan diagram pada UML yang memvisualisasikan perilaku *object* pada *use case* dengan menjelaskan *lifetime* dan pesan yang dikirimkan dan diterima antar *object* [20]. *Sequence diagram* dibuat setelah mengetahui *object* yang terlibat pada *use case* dan metode yang dimiliki *class* yang diinstansiasi menjadi *object* tersebut. Jumlah *sequence diagram* yang dihasilkan minimal sejumlah pendefinisian yang ada pada *use case*, sehingga semakin banyak *use case* yang didefinisikan akan menghasilkan *sequence diagram* yang semakin banyak [21].

## 2.9 *Flow System*

*Flow system* merupakan sebuah diagram alur yang menggambarkan proses pada sebuah sistem [22]. *Flow system* memiliki elemen yang umum digunakan yaitu *start* dan *end* untuk menggambarkan proses awal dan akhir pada alur proses, *process* digambarkan dalam bentuk persegi panjang yang memiliki fungsi untuk menggambarkan tahapan yang dilakukan oleh sistem, *decision* untuk kondisi percabangan yang digambarkan dalam bentuk belah ketupat, proses *input* atau *output* yang digambarkan dalam bentuk jajar genjang, dan tanda panah untuk menunjukkan *flow* atau urutan alur antar tahapan yang ada pada sistem [23].