

## BAB II LANDASAN TEORI

### 2.1 Tinjauan Pustaka

Dalam penelitian ini akan digunakan lima tinjauan pustaka yang nantinya dapat mendukung penelitian, berikut ini merupakan tinjauan pustaka yang diambil yaitu:

**Tabel 2.1** Tinjauan Pustaka

No.	Penulis	Judul	Masalah	Metode	Hasil
1	(Mawarni and Audina, 2018)	Sistem Informasi Pemesanan Pementasan Seni Kesenian Berbasis Website Studi Kasus Sanggar Cangget Budaya Kotabumi Lampung Utara	Saat ini Sanggar Cangget Budaya belum memiliki Sistem Informasi Berbasis Website untuk menyampaikan informasi sehingga masih ada keterbatasan dalam menyampaikan informasi, diantaranya informasi data Kesenian yang tersedia di Sanggar Cangget Budaya serta biaya untuk pemesanan perKesenianannya.	<i>Prototype</i>	Sistem informasi pemesanan pementasan seni Kesenian ini dapat digunakan untuk membantu pemesanan Kesenianan bagi pelanggan yang cukup efektif yang dapat memberikan peluang lebih besar bagi pengembangan organisas
2	(Nas, 2018)	Aplikasi Penyewaan Jasa Kesenian Pada Ukm Seni Dan Olahraga Politeknik Negeri Ujung	Apabila mahasiswa UKM Senior sudah tampil di acara-acara pernikahan, biasanya ada orang yang menanyakan	<i>Waterfall</i>	Hasil penelitian menunjukkan bahwa user bisa langsung melakukan pemesanan penyewaan jasa Kesenian

Tabel 2.1 Tinjauan Pustaka

No.	Penulis	Judul	Masalah	Metode	Hasil
		Pandang Berbasis Web	secara langsung bagaimana cara pemesanan, biaya penyewaan jasa, dan mekanisme penyewaannya. Pihak kampus antar jurusan atau program studi pun juga mengalami keterbatasan informasi mengenai cara pemesanan dan daftar harga paket Kesenianannya. Tentunya hal ini kurang efektif karena informasi mengenai UKM Seni dan Olahraga terkhusus seni Kesenian belum terpublikasikan secara optimal		dan pembelian produk yang tersedia pada web UKM Senior
3	(Alam and Rusdiyanto, 2020)	Aplikasi Jasa Pemesanan Jasa Kesenian Berbasis Web Mobile di Sanggar Seni Studio Lingga	yang menyediakan jasa kesenian yakni seni Kesenian dengan anggaran yang terjangkau. Dalam pengolahan data pemesanan jasa kesenian yang dilakukan oleh sanggar seni studio lingga masih konvensional, dimana calon	<i>Waterfall</i>	Perancangan sistem informasi pemesanan jasa kesenian ini dibuat agar dapat menghasilkan informasi tentang Paket Jasa Kesenian, Data Customer, Data Pemesanan Jasa

Tabel 2.1 Tinjauan Pustaka

No.	Penulis	Judul	Masalah	Metode	Hasil
			customer datang langsung ke sanggar seni studi lingga untuk memesan dan melihat informasi paket jasa kesenian.		Kesenian, dan laporan-laporan yang dihasilkan lebih mudah dan efektif
4	(Utami, 2020)	Sistem Informasi Pemesanan Kelompok Seni Di Kota Semarang Berbasis Website	pemesanan kelompok seni daerah dilakukan dengan cara datang langsung ke alamat kelompok Seni. Ketua Kelompok akan menanyakan beberapa informasi yang dibutuhkan ke pelanggan. Metode pemesanan seperti ini dirasa kurang efisien. Pelanggan harus datang langsung ke alamat Kelompok Seni untuk memesan Kelompok seni.	<i>Waterfall</i>	Dengan adanya sistem ini dapat memudahkan Masyarakat khususnya panitia acara dalam menggelar suatu acara yang akan memesan kelompok pertunjukan seni di Kota Semarang.
5	(Krismet al., 2021)	Rancang Bangun Perangkat Lunak Pemesanan Jasa Seni Dengan Metode Extreme Programming	Sebelumnya terdapat permasalahan dalam pemesanan jasa seni yaitu dengan promosi dan pemasaran yang masih manual seperti menggunakan brosur, mulut kemulut dan sosmed	<i>Extreme Programming</i>	Menghasilkan pemesanan jasa seni, untuk menjawab permasalahan mengenai transaksi pemesanan jasa seni yang ada di Bengkulu

## 2.2 Sistem Informasi

Sistem terdiri atas komponen-komponen yang saling berhubungan satu sama lain dan bekerja sama untuk mencapai suatu tujuan, maka sistem juga dapat diartikan sebagai bentuk gabungan suatu bagian untuk mencapai tujuan pengguna. Sistem yang lebih spesifik mengarah pada sistem informasi yang dimaksud yaitu sistem yang dibuat oleh manusia yang terdiri atas komponen dalam organisasi untuk mencapai tujuan yaitu menyajikan informasi (Widiawati and Siradjuddin, 2020).

Sistem didefinisikan sebagai kumpulan elemen-elemen berupa data, jaringan kerja yang saling terhubung, sumber daya manusia, teknologi hardware dan software yang saling terintegrasi untuk tujuan tertentu. Sehingga kesimpulan dari pembahasan tersebut yaitu sistem dapat digunakan sebagai acuan dalam mengelola suatu aktivitas yang saling terintegrasi antar bagian.

### 2.2.1 Karakteristik Sistem

Adapun karakteristik yang dimaksud adalah sebagai berikut (Rosa dan Shalahuddin, 2019):

1. Komponen Sistem (*Components*)

Suatu sistem terdiri dari sejumlah komponen yang saling berinteraksi, yang artinya saling bekerja sama membentuk satu kesatuan. Komponen-komponen sistem dapat berupa suatu subsistem atau bagian dari sistem.

2. Batas Sistem (*Boundary*)

Merupakan daerah yang membatasi antara suatu sistem yang lainnya atau dengan lingkungan luarnya.

3. Lingkungan Luar Sistem (*Environment*)

Lingkuangan luar dari suatu sistem adalah apapun diluar batas (boundary) dari sistem yang mempengaruhi operasi sistem. Lingkungan luar sistem dapat bersifat menguntungkan namun juga dapat merugikan sistem tersebut.

4. Penghubung Sistem (*Interface*)

Merupakan media penghubung antara satu subsistem dengan subsistem yang lain. Penghubung ini memungkinkan sumber-sumber daya mengalir dari satu subsistem ke subsistem yang lain. Keluaran suatu subsitem akan menjadi masukan untuk subsistem yang lain dengan melewati penghubung.

5. Masukan Sistem (*Input*)

Merupakan energi yang dimasukkan ke dalam sistem. Masukan ini dapat berupa masukan perawatan (maintenance input) dan masukan sinyal (signal input).

6. Keluaran Sistem (*Output*)

Merupakan hasil dari energi yang diolah dan diklasifikasikan menjadi keluaran yang berguna. Artinya keluaran dapat digunakan sebagai masukan untuk subsistem yang lain.

7. Pengolah Sistem (*Processes*)

Suatu proses yang akan mengubah masukan menjadi keluaran. Contohnya sistem akuntansi. Sistem ini akan mengelolah data transaksi menjadi laporanlaporan yang dibutuhkan oleh pihak manajemen.

#### 8. Sasaran Sistem (*Objective*)

Suatu sistem pasti memiliki sasaran. Operasi sistem tidak akan ada gunanya, jika sistem tidak mempunyai sasaran. Sasaran dari sistem akan menentukan masukan yang dibutuhkan sistem serta keluaran yang akan dihasilkan sistem.

### 2.2.2 Klasifikasi Sistem

Sistem merupakan suatu bentuk integrasi antara satu komponen dan komponen lain karena sistem memiliki sasaran yang berbeda untuk setiap kasus yang terjadi di dalam sistem tersebut. Oleh karena itu sistem dapat diklasifikasikan dari beberapa sudut pandang (Rosa dan Shalahuddin, 2019).

Adapun klasifikasi sistem diuraikan sebagai berikut:

#### 1. Sistem Abstrak dan Sistem Fisik

Sistem abstrak merupakan sistem yang berupa pemikiran atau ide-ide yang tidak tampak secara fisik, misalnya sistem telogi. Sedangkan sistem fisik diartikan sebagai sistem yang nampak secara fisik sehingga setiap makhluk dapat melihatnya, misalnya sistem komputer.

#### 2. Sistem Alamiah dan Sistem Buatan Manusia

Sistem alamiah merupakan sistem yang terjadi melalui proses alam, tidak dibuat oleh manusia, misalnya sistem tata surya, sistem galaksi, sistem reproduksi dan lain-lain. Sedangkan sistem buatan manusia merupakan sistem yang dirancang oleh manusia. Sistem buatan yang melibatkan interaksi manusia, misalnya sistem akuntansi, sistem informasi, dan lain-lain.

### 3. Sistem Deterministik dan Sistem Probabilistik

Sistem deterministik merupakan sistem yang beroperasi dengan tingkah laku yang sudah dapat diprediksi. Interaksi bagian-bagiannya dapat dideteksi dengan pasti sehingga keluaran dari sistem dapat diramalkan, misalnya sistem komputer, adalah contoh sistem yang tingkah lakunya dapat dipastikan berdasarkan program-program komputer yang dijalankan. Sedangkan sistem probabilistik merupakan sistem yang kondisi masa depannya tidak dapat diprediksi karena mengandung unsur probabilitas, misalnya sistem manusia.

### 4. Sistem Terbuka dan Sistem Tertutup

Sistem terbuka merupakan sistem yang berhubungan dan terpengaruh dengan lingkungan luarnya. Lebih spesifik dikenal juga yang disebut dengan sistem terotomasi, yang merupakan bagian dari sistem buatan manusia dan berinteraksi dengan kontrol oleh satu atau lebih komputer sebagai bagian dari sistem yang digunakan dalam masyarakat modern. Sistem ini menerima masukan dan menghasilkan keluaran untuk subsistem lainnya, misalnya sistem kebudayaan manusia. Sedangkan sistem tertutup merupakan sistem yang tidak berhubungan dan tidak terpengaruh dengan lingkungan luarnya. Sistem ini bekerja secara otomatis tanpa campur tangan dari pihak luar.

## 2.3 Kesenian

Kesenian adalah salah satu bagian dari kebudayaan yang dikagumi karena keunikan dan keindahannya. Kesenian merupakan hasil karya seni manusia yang

mengungkapkan keindahan serta merupakan ekspresi jiwa dan budaya penciptanya. Kesenian merupakan bagian dari budaya dan sarana yang digunakan untuk mengekspresikan rasa keindahan dari dalam jiwa manusia, keindahannya juga mempunyai fungsi lain. Ragam kesenian yang ada tersebut diantaranya adalah seni musik, seni rupa, seni teater, seni sastra dan seni Kesenian. Perwujudan seni yang ada di masyarakat merupakan cerminan dari diri kepribadian hidup masyarakat. Kesenian selalu melekat pada kehidupan manusia, dimana ada manusia di dalamnya pasti ada kesenian (Saril, *et al.*, 2019).

#### **2.4 Metode *First Come First Served* (FCFS)**

Metode *First Come First Served* (FCFS) adalah sistem antrian penjadwalan yang mengutamakan proses yang pertama kali diinput akan dilayani terlebih dahulu sampai selesai. Proses tersebut diberi jarak waktu yang diurutkan berdasarkan waktu kedatangan. Metode ini juga mengasumsikan bahwa pertama datang maka pertama yang akan dilayani. Proses yang pertama kali meminta jatah waktu untuk menggunakan aplikasi akan dilayani terlebih dahulu. Pada skema ini, proses yang meminta server pertama kali akan dialokasikan ke server. Penjadwalan FCFS ini merupakan penjadwalan nonpreemptive, Penjadwalan tidak berprioritas (Purwanto *et al.*, 2022).

Ketentuan penjadwalan FCFS adalah penjadwalan paling sederhana, yaitu proses-proses diberi jatah waktu pemrosesan berdasarkan waktu kedatangan, begitu proses mendapatkan jatah waktu pemroses, proses dijalankan sampai selesai. Penjadwalan ini dikatakan adil dalam arti resmi (dalam semantik/arti antrian, yaitu proses yang datang terlebih dahulu, dilayana terlebih dahulu), tapi dinyatakan tak adil karena pekerjaan yang memerlukan waktu yang lama



membuat pekerjaan yang pendek menunggu, dan pekerjaan yang tidak penting dapat membuat pekerjaan yang penting harus menunggu (Riadi and Ulum, 2021).

FCFS jarang digunakan secara mandiri tapi dikombinasikan dengan skema lain, misalnya keputusan berdasarkan prioritas proses. Untuk proses-proses berprioritas sama diputuskan berdasarkan FCFS (Purwanto *et al.*, 2022).

Berdasarkan kriteria penilaian penjadwalan :

1. Fairnes, Penjadwalan FCFS adil bila dipandang dari semantik antrian.
2. Efisiensi, Penjadwalan FCFS sangat efisien.
3. Waktu tanggap, Pejadwalan FCFS sangat tidak baik, tidak cocok untuk sistem interaktif apalagi waktu nyata.

Setiap proses yang berada pada status ready dimasukkan ke dalam FCFS queue (struktur data antrian) sesuai dengan waktu kedatangannya (Purwanto *et al.*, 2022). Dirumuskan :

$$TA = \textit{Waiting Time} + \textit{Length Of Execution}$$

Keterangan :

*Turn Around Time (TA)* = Waktu penyelesaian

*Waiting time* = Waktu tunggu

*Length Of Execution* = Lama Eksekusi

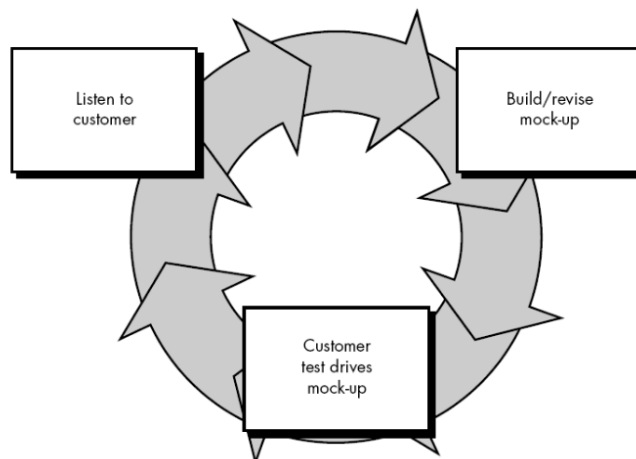
## 2.5 Metode Pengembang Sistem

Metode pengembang sistem merupakan metode yang digunakan sebagai alur proses dalam pengembangan, sehingga penelitian dapat di kembangkan sesuai tahapan dari metode pengembang sistem (Rosa dan Shalahuddin, 2019).

### 2.5.1 *Prototype*

Prototype merupakan metode yang digunakan untuk mendapatkan kebutuhan pengguna lebih spesifik dalam segi teknis. Model *prototype* dapat digunakan untuk menyambungkan ketidakpahaman pengguna mengenai teknis dan memperjelas spesifikasi kebutuhan yang diinginkan pengguna kepada pengembang perangkat lunak (Rosa dan Shalahuddin, 2019).

Prototype dapat dimulai dengan mengumpulkan kebutuhan pengguna terhadap perangkat lunak yang akan dibuat, kemudian dirancang menggunakan model *prototype* dalam bentuk mockup untuk mempermudah pengguna mengevaluasi *prototype* tersebut. Sehingga hasil dari sistem yang dikembangkan telah sesuai dengan permintaan dan kebutuhan pengguna.



**Gambar 2.1** Metode *Prototype*  
Sumber : (Rosa dan Shalahuddin, 2019)

## 1. Kelebihan *Prototype*

Berikut merupakan kelebihan metode *prototype* :

- a. Menghemat waktu dan Biaya pengembangan
- b. Adanya keterlibatan pemilik sistem sehingga kesalahan sistem bisa diminimalisir dari awal proses
- c. Membantu anggota tim untuk berkomunikasi secara efektif
- d. Klien memiliki kepuasan tersendiri karena sudah memiliki gambaran dari sistem yang akan dibuat.
- e. Implementasi atau penggunaan sistem lebih mudah karena klien sudah tahu gambaran sistem sebelumnya
- f. Kemudahan dalam memperkirakan pengembangan sistem selanjutnya
- g. Memungkinkan klien untuk mempersiapkan perangkat lunak yang cocok dengan sistem yang akan dibuat.

## 2. Kelemahan *Prototype*

Berikut merupakan kelemahan metode *prototype* :

- a. *Prototype* adalah metode yang menghabiskan banyak waktu jika klien kurang puas ditahapan awal.
- b. Klien terus menerus menambah *requirement* dari sistem, pegen dibuatkan yang seperti inilah seperti itulah, sehingga menambah kompleksitas pembuatan sistem.
- c. Sistem akan terhambat jika komunikasi kedua belah pihak tidak berjalan secara efektif.


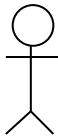
## 2.6 Unified Modelling Language (UML)


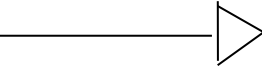
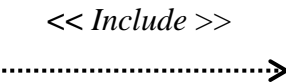

Menurut Rosa and Shalahuddin (2019) UML (*unified Modelling Language*) adalah bahasa visual untuk pemodelan dan komunikasi mengenai sebuah sistem dengan menggunakan diagram dan teks-teks pendukung. Berikut ini merupakan penjelasan tentang masing-masing diagram yang ada pada UML (*Unified Modelling Language*).

### 2.6.1 Use Case Diagram

*Use case* mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem informasi yang akan dibuat. *Use case* digunakan untuk mengetahui fungsi apa saja yang ada di dalam sebuah sistem informasi dan siapa saja yang berhak menggunakan fungsi-fungsi tersebut (Rosa and Shalahuddin, 2019). Berikut simbol-simbol yang akan digunakan dalam menggambarkan *Use Case Diagram* dapat dilihat pada Tabel 2.2.

**Tabel 2.2** Simbol *Use Case Diagram*

No	Simbol	Deskripsi
1.		<i>Use case</i> Fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit atau aktor, biasanya dinyatakan dengan menggunakan kata kerja di awal <i>frase</i> nama <i>use case</i> .
2.		Aktor Aktor seseorang/sesuatu yang berinteraksi dengan yang akan dibuat. diluar sistem informasi. Biasanya dinyatakan menggunakan kata benda


No	Simbol	Deskripsi
3.		Asosiasi/association merupakan komunikasi antara aktor dan use case yang berpartisipasi pada use case atau use case memiliki interaksi dengan aktor.
4.		Generalisasi ( <i>generalization</i> ) merupakan hubungan (umum – khusus) antara dua buah use case dimana fungsi yang satu adalah fungsi yang lebih umum
5.		Include berarti use case yang ditambahkan akan dipanggil saat use case tambahan dijalankan.
6.		Ekstensi ( <i>extend</i> ) merupakan use case tambahan ke sebuah use case yang ditambahkan dapat berdiri sendiri walau tanpa use case tambahan itu.


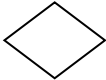

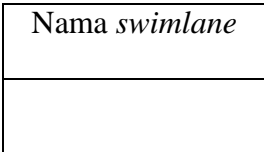

Sumber : (Rosa and Shalahuddin, 2019)

### 2.6.2 Activity Diagram

*Activity* diagram menggambarkan aliran kerja atau aktivitas dari sebuah sistem atau proses bisnis atau menggambarkan aktivitas sistem bukan apa yang dilakukan aktor, jadi aktivitas yang dapat dilakukan oleh sistem (Rosa and Shalahuddin, 2019). Berikut simbol-simbol yang akan digunakan dalam menggambarkan *activity diagram* dapat dilihat pada Tabel 2.3:

**Tabel 2.3** Simbol *Activity* Diagram

No.	Simbol	Keterangan
1.		Status awal aktivitas sistem, sebuah diagram aktivitas memiliki sebuah status awal.

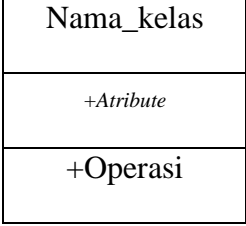


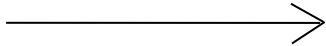
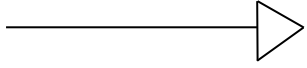
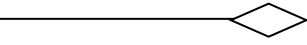
No.	Simbol	Keterangan
2.		Aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja.
3.		Percabangan ( <i>Decision</i> ) merupakan asosiasi percabangan dimana jika ada pilihan aktivitas lebih dari satu.
4.		Penggabungan ( <i>Join</i> ) merupakan asosiasi penggabungan dimana lebih dari satu aktivitas digabungkan menjadi satu.
5.		Swimlane Memisahkan organisasi bisnis yang bertanggung jawab terhadap aktivitas.
6.		Status akhir yang dilakukan sistem, sebuah diagram aktivitas memiliki sebuah status akhir.

**Sumber :** (Rosa and Shalahuddin, 2019)

### 2.6.3 Class Diagram

*Class diagram* mengembangkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem (Rosa and Shalahuddin, 2019). Berikut simbol-simbol yang akan digunakan dalam menggambarkan *Class Diagram* dapat dilihat pada Tabel 2.4.

Tabel 2.4 Simbol *Class* Diagram

No.	Simbol	Deskripsi
1.		Kelas pada struktur sistem.
2.	<p>Antar Muka/<i>Interface</i></p>  <p>Nama_<i>Interface</i></p>	Sama dengan konsep interface dalam pemrograman berorientasi objek.
3.	<p>Asosiasi / <i>Asociation</i></p> 	Relasi antar kelas dengan makna umum, asosiasi biasanya juga disertai dengan symbol
4.	<p>Asosiasi Berarah / <i>Directed Association</i></p> 	Relasi antar kelas dengan makna kelas yang satu digunakan oleh kelas yang lain, asosiasi biasanya juga disertai dengan symbol.
5.	<p>Generalisasi</p> 	Relasi antar kelas dengan makna generalisasi-spesialisasi (umum khusus)
6.	<p>Agregasi / <i>aggregation</i></p> 	Relasi antar kelas dengan maksna semua bagian ( <i>whole-part</i> )

**Sumber:** (Rosa and Shalahuddin, 2019)

## 2.7 Mobile

Aplikasi *mobile* yaitu program siap pakai yang direkap untuk melaksanakan suatu fungsi bagi pengguna atau aplikasi yang lain dan dapat digunakan oleh sasaran yang dituju sedangkan *mobile* dapat di artikan sebagai perpindahan dari

suatu tempat ketempat yang lain. Maka aplikasi *mobile* dapat di artikan sebuah program aplikasi yang dapat dijalankan atau digunakan walaupun pengguna berpindah-pindah dari satu tempat ketempat yang lain serta mempunyai ukuran yang kecil (Gunawan *et al.*, 2017).

## **2.8 PHP (*Hypertext Preprocessor*)**

PHP (*Hypertext Preprocessor*) merupakan bahasa pemrograman untuk membuat website atau situs dinamis dan mengenai rangkaian bahasa pemrograman antara *client side scripting* dan *server side scripting* (Heru 2018).

Bahasa pemrograman yang digunakan secara luas untuk penanganan, pembuatan dan pengembangan sebuah situs web dan biasanya bersamaan dengan HTML. Sehingga PHP merupakan bahasa pemrograman yang digunakan oleh pengembang untuk membuat sistem *website* dengan kumpulan bahasa HTML dan *script* lainnya (Oetomo and Maharginono, 2020)

## **2.9 Database**

*Database* yaitu himpunan kelompok data (arsip) yang saling berhubungan yang diorganisasikan sedemikian rupa agar kelak dapat dimanfaatkan kembali dengan cepat dan mudah (Siahaan, 2020). *Database* juga merupakan kumpulan data yang saling berhubungan yang disimpan secara bersama sedemikian rupa dan tanpa pengulangan (redundansi) yang tidak perlu untuk memenuhi berbagai kebutuhan

### **1. Fungsi Database**

Menurut Siahaan (2020) Fungsi *database* adalah untuk menghindari data ganda yang tersimpan. Suatu *database management system* (DBMS) dapat



diatur supaya bisa mengenali duplikasi data ketika diinput, selain untuk menghindari data ganda, *database* memiliki fungsi lainnya, seperti:

- a. Mengelompokan data dan informasi. Memudahkan dalam identifikasi data.
- b. Memudahkan proses akses, menyimpan, pembaharuan, dan penghapusan data.
- c. Menjadi alternatif terkait masalah penyimpanan ruang dalam suatu aplikasi.
- d. Menjaga kualitas data yang diakses sesuai input. Menunjang kinerja aplikasi yang memerlukan penyimpanan data.

## 2. Jenis *Database*

Menurut Putra and Kuswini (2019) Berikut merupakan jenis-jenis *database* :

### a. *Operational Database*

*Operational database* juga dikenal dengan nama *On Line Transaction Processing*. *Database* jenis ini berfungsi sebagai wadah mengelola data dinamis secara *real-time* atau langsung, *operational database* memberikan manfaat kepada penggunanya untuk melihat hingga memodifikasi data.

### b. *Database Warehouse*

*Database warehouse* digunakan untuk melakukan analisis dan pelaporan data. *Database warehouse* dianggap sebagai komponen inti *business intelligence*. *Database warehouse* yaitu *repository* sentral terpadu dan berasal dari satu atau lebih sumber yang berbeda.

### c. *Distributed Database*

*Distributed database* berbeda dengan sistem paralel yang terhubung erat dan memiliki sistem pada data tunggal. Basis data tidak terpasang pada

perangkat komputer atau sejenisnya yang serupa. Sistem ini terdistribusi melalui suatu situs yang tergabung dan tidak memiliki komponen fisik.

d. Relational Database

*Relational database* mengorganisir data berdasarkan model hubungan data. Basis data relasional ini digunakan oleh banyak perangkat lunak untuk mengatur dan memelihara informasi melalui hubungan setiap data. Beberapa produk *relational database* yang sering digunakan adalah SQL, Oracle, MySQL, SQLite, dan sebagainya.

e. End-User Database

Basis data yang satu ini dikembangkan oleh *end-user* melalui *workstation* mereka. Berbagai jenis berkas data dibuat sendiri dengan suatu prosedur tertentu. Adapun contohnya seperti *spreadsheet*, *word processing*, dan *download file*.

3. Database yang digunakan

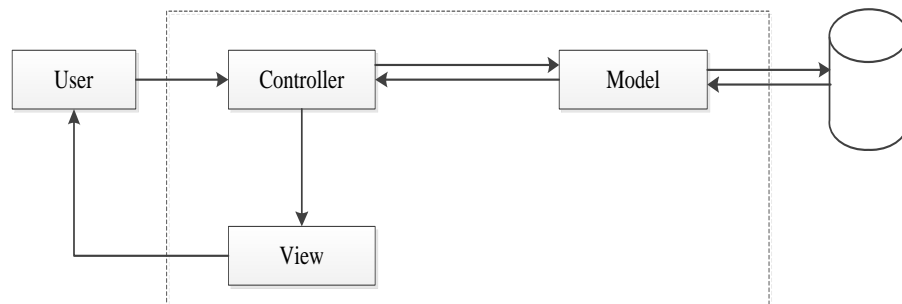
*MySQL* adalah singkatan dari *Structure Query Language* yang digunakan untuk mendefinisikan *structure* data, memodifikasi data pada basis data, menspesifikasi batasan keamanan (*security*), hingga pemeliharaan data. *Mysql* sering digunakan karena dapat digunakan secara gratis dan dapat dikembangkan lebih luas sesuai kebutuhan (Siahaan, 2020).

## 2.10 CodeIgniter

*CodeIgniter* adalah *Framework* untuk bahasa pemrograman PHP, yang dibuat Rick Ellis pada tahun 2006. *CodeIgniter* memiliki banyak fitur yang membantu para pengembang PHP untuk dapat membuat aplikasi secara mudah

dan cepat serta memiliki sifat yang fleksibel dapat mengembangkan dalam perangkat *web*, *desktop* maupun *mobile* (Raharjo, 2018). *Codeigniter* adalah sebuah aplikasi gratis yang berupa kerangka kerja untuk membangun website menggunakan bahasa pemrograman PHP.

*CodeIgniter* memiliki konsep atau pola *Model-View-Controller* (MVC) sehingga kode-kode dapat di sederhanakan.



**Gambar 2.2** Arsitektur MVC

Sumber: (Raharjo, 2018)

Berdasarkan arsitektur tersebut diketahui konsep *Model-View-Controller* yaitu:

1. *Model*

*Model* merupakan suatu fungsi yang digunakan mengelola database pada sistem untuk menangani struktur data dari *controller*.

2. *View*

*View* merupakan bagian untuk mengelola tampilan dari *website* atau dapat disebut sebagai *user interface* yang diatur bagian *controller*.

3. *Controller*

*Controller* merupakan kunci dalam konsep MVC dengan fungsi untuk menghubungkan *model* dengan *view*.

*CodeIgniter* merupakan sebuah *toolkit* yang ditujukan untuk orang yang ingin membangun aplikasi web dalam bahasa pemrograman PHP. Beberapa keunggulan yang ditawarkan sebagai berikut:

1. *CodeIgniter* adalah *framework* yang bersifat *free* dan *open source*
2. Memiliki ukuran file yang relatif lebih kecil dibanding *framework* lain
3. Aplikasi yang dihasilkan dapat berjalan cepat.
4. Menggunakan pola desain MVC sehingga satu file tidak banyak berisi kode, hal tersebut menjadikan kode mudah dibaca dan dipahami.
5. Dapat diperluas sesuai kebutuhan
6. Dokumentasi penerapan *CodeIgniter* dapat dipahami dengan mudah.

### **2.11 Pengujian *Black Box***

Pengujian sistem adalah proses untuk mengecek apakah suatu perangkat lunak yang dihasilkan sudah dapat dijalankan sesuai standar atau belum. Pengujian sistem dapat menggunakan metode *black box testing* yaitu merupakan pendekatan komplementer dari teknik *white box testing*, karena pengujian *black box testing* mampu mengungkap kesalahan yang lebih luas. *Black box testing* berfokus pada pengujian persyaratan fungsional perangkat lunak, karena untuk mendapatkan serangkaian kondisi input yang sesuai dengan persyaratan fungsional suatu program (Rosa and Shalahuddin, 2019).

Jadi, pengujian sistem dapat dilakukan dengan pengecekan *input*, pengecekan *output* dan pengecekan proses sebagai berikut:

- a. Pengecekan *input*, meliputi kelengkapan item-item input, kemudahan pengoperasian, kemudahan manipulasi data, dan pengendalian kesalahan.
- b. Pengecekan proses, dilakukan dengan pengecekan output program.
- c. Pengecekan *output*, meliputi pengecekan terhadap format dan bentuk-bentuk laporan

### **2.11.1 Kelebihan *Black Box Testing***

Kelebihan dari pengujian tersebut yaitu:

- a. Tidak perlu melihat *source code* secara detail.
- b. Mendeteksi kesalahan pengetikan (*typo*).
- c. Mendeteksi kesalahan *Design* atau *User Interface* dari sebuah *software* atau *website*.
- d. Menampilkan asumsi yang tidak sesuai dengan kenyataan, untuk di analisa dan diperbaiki.
- e. Seorang *Tester* tidak harus *Programmer*.

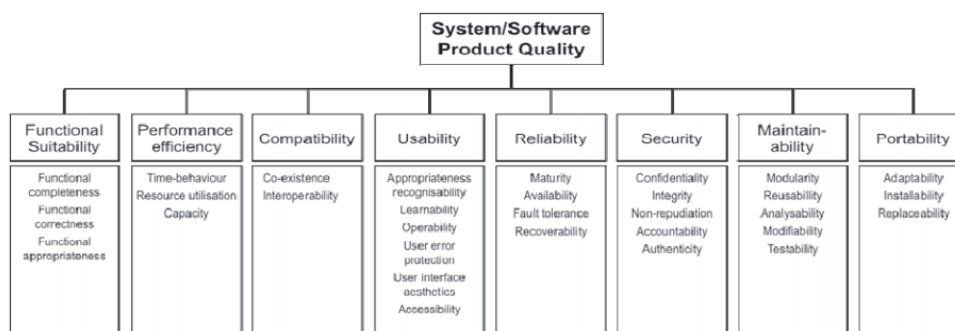
### **2.11.2 Kekurangan *Black Box Testing***

Kekurangan dari pengujian tersebut yaitu:

- a. Ketergantungan dengan dokumen dan design *software* tersebut.
- b. Tidak sampai level *code*, sehingga tester tidak mengetahui *level security* dari *software* tersebut.

## 2.12 Pengujian ISO 25010

ISO/IEC 25010 merupakan model kualitas sistem dan perangkat lunak yang menggantikan ISO/IEC 9126 tentang *software engineering* (*International Organisation for Standardisation*, 2011), dapat dilihat pada gambar berikut:



**Gambar 2.3** ISO 25010  
Sumber : (Mulyawan *et al.*, 2021)

Model kualitas produk terdiri dari delapan karakteristik yang berhubungan dengan sifat statis perangkat lunak dan sifat dinamis dari sistem komputer. Berdasarkan tahapan-tahapan ISO 25010 tersebut maka peneliti menggunakan pengujian terhadap kualitas perangkat lunak berupa aplikasi *web* dapat dinilai dari empat aspek yaitu:

### 1. *Functional Suitability*

Sejauh mana suatu produk atau sistem menyediakan fungsi yang memenuhi kebutuhan yang dinyatakan dan tersirat ketika digunakan di bawah kondisi tertentu. Sub pengujian tersebut sebagai berikut:

#### a. *Functional Completeness* (Kelengkapan Fungsional)

Sejauh mana rangkaian fungsi mencakup semua tugas dan tujuan pengguna yang ditentukan

#### b. *Functional Correctness* (Kebenaran Fungsional)

Sejauh mana suatu produk atau sistem memberikan hasil yang benar dengan tingkat presisi yang dibutuhkan.

Alat pengujian yang digunakan yaitu kuisisioner yang memiliki 2 jawaban pertanyaan seperti sukses atau gagal.

## 2. *Usability*

Sejauh mana suatu produk atau sistem dapat digunakan oleh pengguna tertentu untuk mencapai tujuan tertentu dengan efektivitas, efisiensi, dan kepuasan dalam konteks penggunaan tertentu. Sub pengujian tersebut sebagai berikut :

### a. *Appropriateness Recognizability*

Sejauh mana pengguna dapat mengenali apakah suatu produk atau sistem sesuai dengan kebutuhan mereka.

### b. *Learnability*

Sejauh mana suatu produk atau sistem dapat digunakan oleh pengguna tertentu untuk mencapai tujuan pembelajaran yang ditentukan untuk menggunakan produk atau sistem dengan efektifitas, efisiensi, bebas dari resiko dan kepuasan dengan tujuan tertentu konteks penggunaan.

Alat pengujian yang digunakan yaitu kuisisioner yang memiliki 5 jawaban pertanyaan seperti Sangat Setuju, Setuju, Ragu-ragu, Tidak Setuju dan Sangat Tidak Setuju.