

## **BAB II** **LANDASAN TEORI**

### **2.1 Tinjauan Pustaka**

1. Oleh I Putu Yoga Indrawan<sup>1</sup> , Putu Gede Surya Cipta Nugraha, 2021. Rancangan dan Implementasi Sistem E-Learning Berbasis Web. Permasalahan yang muncul pada SMP Negeri 1 Singaraja adalah tidak adanya sarana pendukung pembelajaran yang bisa diakses kapan saja dan dimana saja oleh para siswa. Hal ini mengakibatkan terhambatnya proses pembelajaran antara guru dan siswa. Metode yang digunakan yaitu Waterfall. Hasil penelitian menunjukkan bahwa rancangan sistem elearning sudah sesuai dengan kurikulum dan kebutuhan guru dan peserta didik. Sehingga dapat disimpulkan bahwa rancangan sistem e-learning layak diimplementasikan sebagai sarana pendukung pembelajaran di sekolah.
2. Oleh Iskandar, Setiawan Assegaff, 2017. Pengembangan E-Learning Dalam Pembelajaran Pada Iain Sts Jambi. Permasalahan Saat ini konsep e-learning sudah banyak diterima oleh masyarakat dunia, terbukti dengan maraknya implementasi e-learning khususnya di lembaga pendidikan untuk menunjang sistem pembelajaran yang didukung oleh teknologi informasi. Metode yang digunakan Pengembangan Sistem *Elearning*. Hasil analisis sistem eleraning yang sedang diimplementasikan IAIN STS Jambi menunjukkan dari fitur-fitur yang tersedia dalam menujung proses pembelajaran terdapat fitur quiz yang merupakan bagian penting untuk fitur pelaksanaan Evaluasi Pembelajaran belum digunakan secara efektif dikarenakan untuk

mengaplikasikan fitur ini sangat kompleks melalui settingan yang banyak sehingga dosen merasakan kesulitan dalam penerapannya.

Oleh Hotmian Sitohang, Rosmiati, Era Elisa Semberson Sinaga, 2021. Aplikasi E-Learning Berbasis Web untuk Pembelajaran Jarak Jauh. Permasalahan yang Saat ini seluruh dunia diguncangkan dengan kedatangan virus corona (covid 19) yang begitu menakutkan. Virus ini begitu cepat menular dan membahayakan sampai dapat menghilangkan nyawa. Dengan adanya virus ini semua kegiatan dilakukan dari rumah, sehingga membuat perusahaan dibidang teknologi bergerak begitu pesat untuk menciptakan software yang dibutuhkan saat ini, seperti meeting online, class online, e-learning, webinar, dan lainnya. Metode yang digunakan *Rapid Application Development (RAD)*. Hasil penelitian media ini dapat membantu guru sejarah dalam memberikan materi pelajaran dan soal-soal latihan. Kelebihan aplikasi ini juga terdapat kuis yang nantinya bila siswa mengerjakan akan tampil skor nilai dan juga dilengkapi dengan kunci jawaban.

4. Oleh Musdalifa Nur, 2021. Media Pembelajaran E-Learning Menggunakan Aplikasi Edmodo Dalam Masa Pandemi. Permasalahan Media pembelajaran *E-learning* sedang marak digunakan saat masa pandemi seperti saat ini, dimana ada banyak sekali aplikasi-aplikasi/Platform yang bisa kita jadikan sebagai media pembelajaran *E-learning* Edmodo adalah aplikasi yang bisa digunakan dengan Smartphone, mudah diaplikasikan bagi guru/tenaga pendidik juga bagi siswa, tidak hanya itu orang tua juga mampu memantau perkembangan proses pembelajaran anak-anaknya. Metode yang digunakan adalah Edmodo. Hasil penelitian Edmodo mampu memberikan fitur-fitur

yang bisa dimanfaatkan oleh guru dalam mengelola materi ajaran, membuat tugas serta hal-hal yang bisa mendukung jalannya proses pembelajaran, tidak hanya itu penggunaan aplikasi ini cenderung mudah dipahami serta mampu meningkatkan kreatifitas dalam proses pembelajaran, tanpa meninggalkan suatu yang penting yaitu peran orang tua dalam proses pembelajaran sang anak juga diikutsertakan dalam fitur di *Edmodo*, dibalik kelebihan pasti ada kelemahannya, tetapi pihak *Edmodo* selalu memberikan sebuah solusi serta meningkatkan kualitas aplikasinya.

5. Oleh Weni Andiani, Happy Fitria, 2021. Pembelajaran Daring Menggunakan Media *Online* Selama Pandemi Covid-19 Pada Siswa Sd Negeri 103 Palembang. Permasalahan yang ada pada Pandemi Covid-19 telah membuat Indonesia dan seluruh dunia beralih dari sistem pembelajaran tatap muka ke sistem pembelajaran *online* untuk mengetahui bagaimana pembelajaran daring (dalam jaringan) dengan media *online* selama pandemi Covid-19 pada siswa SD Negeri 103 Palembang. Metode yang digunakan *daring, media online* , *pembelajaran*. Hasil penelitian menemukan bahwa media *online* yang digunakan pada siswa tempat penelitian diantaranya *Whatsapp group, Youtube, dan Quizziz*. Penggunaan media *online* ini dinilai cukup efektif dalam pembelajaran pada siswa SD Negeri 103 Palembang. Kelemahan pembelajaran daring, guru lebih banyak memberikan penugasan kepada siswa. Penjelasan yang diberikan oleh guru sangat kurang. Sehingga siswa merasa berat dan sulit memahami materi dalam mengikuti proses pembelajaran karena didominasi oleh pengerjaan tugas.

## **2.2 Konsep Pembelajaran Berbasis TIK**

Menurut Munir (2017), Teknologi informasi khususnya komputer telah banyak merubah tatanan dan peran pendidikan. Sebagai contoh dahulunya guru merupakan sumber informasi yang utama bagi peserta didik dengan hadirnya komputer melalui jaringan internet telah membuat guru bukanlah satu-satunya sumber informasi tapi informasi dapat diakses dari komputer melalui jaringan internetnya, proses belajar mengajar yang disampaikan secara klasikal dengan metode ceramah yang membosankan tapi dengan hadirnya teknologi komputer menyebabkan pembelajaran dapat dilakukan secara individual dan menyenangkan.

Menurut Ucu *et al.*, (2018), Peranan teknologi komputer pada aktivitas manusia pada saat ini memang begitu besar. Komputer telah menjadi fasilitator utama bagi kegiatan-kegiatan disemua sektor kehidupan termasuk dalam sektor pendidikan. Komputer telah memberikan andil besar terhadap perubahan-perubahan yang mendasar pada struktur, operasi, dan manajemen sistem pendidikan dan pembelajaran. Berkat teknologi komputer ini berbagai kemudahan dapat dirasakan dalam proses pembelajaran seperti persentasi mengajar, akses informasi (e-learning) dan pembuatan pembelajaran berbasis komputer.

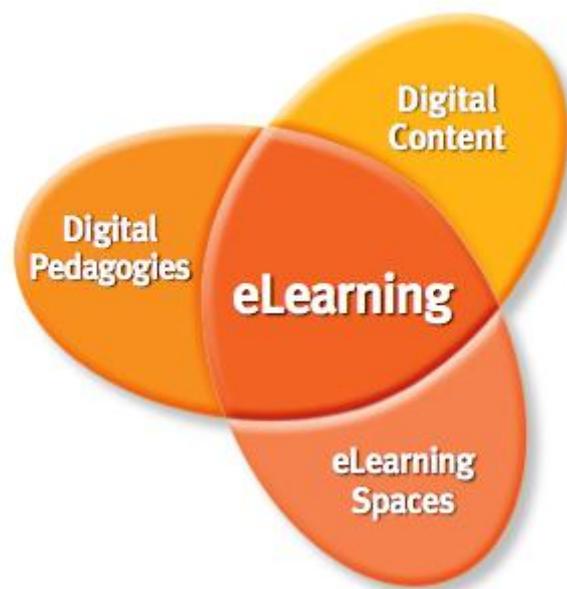
Masuknya komputer dalam proses belajar akan melahirkan suasana yang menyenangkan karena peserta didik dapat mengendalikan kecepatan belajar sesuai dengan kemampuannya. Lalu gambar dan suara yang muncul membuat peserta didik tidak cepat bosan, sebaliknya justru merangsang untuk mengetahui lebih jauh lagi. Dengan desain program pembelajaran yang menarik diharapkan peserta

didik menjadi tekun, sehingga diharapkan menjadi lebih unggul di bidangnya, lebih cerdas, lebih kreatif, dan lebih mampu melihat persoalan dari segi lain, kini dan masa datang. Suasana menyenangkan seperti ini jarang dinikmati peserta didik.

### 2.3 Konsep *E-learning*

Konsep *e-learning* menurut Huda and Faiza (2019) adalah Strategi pengajaran dengan menerapkan konsep belajar *blended learning* yang menampilkan keseimbangan antara pengajaran virtual (kelas maya) dan tatap muka (kelas fisik) yang berlangsung dalam waktu yang sama. *E-learning* tidak terbatas pada studi mandiri atau model pembelajaran jarak jauh, hal ini perlu menjadi pertimbangan penting untuk diterapkan di sekolah.

Berikut adalah strategi pengajaran *e-learning* terdiri dari komponen yang saling terkait dan saling tergantung yaitu:



**Gambar 2.1** Strategi *E-Learning*

Berdasarkan gambar tersebut maka konsep *e-learning* dapat di bagi menjadi tiga bagian yaitu :

### **2.3.1 Digital Pedagogies**

Menurut Rosyid (2016) pedagogi pada dasarnya adalah latihan berpikir kritis yang diarahkan untuk belajar dan mengajar seperti pemahaman aktivitas pengelolaan kelas, peran motivasi siswa, rencana pembelajaran, dan penilaian pembelajaran. Pedagogi juga mendeskripsikan pengetahuan dari metode mengajar yang berbeda-beda meliputi pengetahuan untuk mengetahui bagaimana mengorganisasikan aktivitas di kelas agar konstruksi pengetahuan siswa (pembelajaran) kondusif.

Sehingga Pedagogi adalah tentang menggunakan alat-alat digital secara serius seperti halnya menentukan kapan tidak menggunakan alat-alat digital, dan tentang memperhatikan dampak dari alat-alat digital pada pembelajaran.

### **2.3.2 Digital Content**

Menurut Munir (2017) perkembangan konten digital yaitu Perkembangan pembelajaran digital. Dikembangkannya konten digital dalam pembelajaran digital menjadikan proses pembelajaran menjadi menarik, fleksibel, atraktif, dan interaktif. Bagaimana suatu konten digital itu bekerja mengemas informasi, apa makna informasi yang dapat diinterpretasi dari program atau kemasan pesannya, sampai pada bagaimana orang mendapat materi pembelajaran tanpa dibatasi ruang dan waktu.

Sehingga Perangkat lunak yang disediakan untuk kebutuhan pembelajaran dengan konten digital seperti multimedia. Bahkan di sekolah yang telah menerapkan teknologi informasi, pembelajar diberikan laptop untuk memudahkan mereka belajar secara digital, sehingga pembelajaran paperless (mengurangi pemakaian kertas dalam aktivitas mengajar) bisa dicapai. Konten digital ini kemudian dibagikan kepada pembelajar untuk kebutuhan pembelajar mengulang pelajaran di rumah. Objek pembelajaran atau konten ini dapat mencakup *file* media yang dikembangkan dalam penilaian item, simulasi, teks, atau *video*.

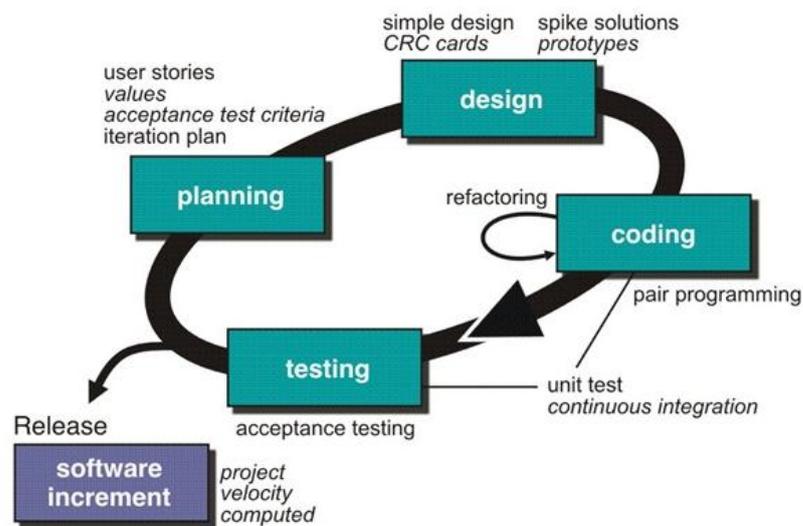
### **2.3.3 *E-learning Space***

*E-learning Space* atau ruang pengajaran cerdas menurut Badami *et al.*, (2018) yaitu mencakup ruang pengajaran fisik dan virtual dalam pengaturan institusi. Konsep *Smart Teaching Spaces* mendefinisikan furnitur penting dan spesifikasi teknis ruang kelas fisik yang ideal dan fitur paling penting dari platform pembelajaran virtual. Juga fasilitas dan fungsinya untuk ruang kuliah dan ruang pengajaran kolaboratif akan ditentukan dalam paket ini. Ruang Pengajaran Cerdas secara keseluruhan membawa fleksibilitas untuk belajar dan menjadikan pembelajaran itu ada di mana-mana.

*E-learning Space* merupakan ruang belajar elektronik dengan menyediakan wadah atau tempat belajar menggunakan sebuah aplikasi yang diterapkan dengan pemanfaatan teknologi. ruang belajar mengadopsi kelas pintar, juga pembelajaran virtual menggunakan sistem secara *online* dengan media internet.

## 2.4 Metode *Extreme Programming*

Menurut Suryantara (2017) *extreme programming* berdasarkan sejarah singkat bahwa pengembangan perangkat lunak banyak digunakan untuk pengembangan yang lebih cepat dengan meliputi tahapan *planning*, *design*, *coding* dan *testing*. Berikut merupakan konsep *Extreme programming*.



**Gambar 2.2** *Extreme Programming*

Sumber: (Suryantara, 2017)

### 2.4.1 Sejarah Singkat *Extreme Programming*

*Extreme Programming* merupakan salah satu metodologi rakayasa perangkat lunak yang banyak digunakan untuk pengembangan aplikasi oleh para developer, diperkenalkan oleh kent beck yang ditunjukan untuk menangani sebuah proyek penggajian yang dikenal sebagai C3 (Chrysler Comprehensive Compensatun). Proyek tersebut dimulai pada maret 1996 yan terancam gagal karena rumitnya sistem mengalami kegagalan pada proses testing, kemudian pihak Chrysler menyewa Kent Beck sebagai kunsultan dibidang *software engineerung* yang kemudian disebut sebagai pencetus XP.

### **2.4.2 Kerangka Kerja *Extreme Programming***

Pengembangan yang dilakukan menggunakan XP dengan proses yang lebih cepat dengan tahapan seperti *planning*, *design*, *coding* dan *testing*.

#### 1. *Planning*/Perencanaan

Tahap ini dimulain dengan pemahaman konteks bisnis dari aplikasi dengan mendefinisikan keluaran seperti fitur, fungsi, penentuan waktu dan biaya serta alur pengembangan.

#### 2. *Design*/Perancangan

Tahap perencanaan secara sederhana dengan alat mendesain kartu CRC (*Class Responsibility Collaborator*) yang digunakan untuk pemetaan kelas-kelas yang akan diguanakna pada diagram UML.

#### 3. *Coding*/Pengkodean

Hal utama dalam pengembangan menggunakan XP yaitu *pari programming* (Proses pembuatan program melibatkan 2 atau lebih programmer).

#### 4. *Testing*/Pengujian

Tahap ini fokus pda pengujian fitur pada aplikasi sehingga tidak ada kesalahan dan sesuai dengan proses bisnisnya.

### **2.4.3 Nilai Inti *Extreme Programming***

Nilai inti pada pengembangan sistem yaitu :

#### 1. *Communication* (Komunikasi)

Konmunikasi antar tim yang digunakan untuk saling berbagi pengetahuan dalam pengembangan.

#### 2. *Simplicity* (Kesederhanaan)

Melakukan semua proses dengan sederhana dan mencoba mencari solusi yang paling sederhana.

3. *Feedback* (Masukan)

Masukan untuk mengetahui kemajuan dari proses dan kualitas perangkat lunak yang dibuat.

4. *Courage* (Kesalahan)

Kesuksesan pengembangan aplikasi harus memiliki keberanian, keyakinan dan integritas dalam pekerjaan.

5. *Respect* (Menghormati)

Menerapkan siklus pendek dan integrasi *continue*.

#### **2.4.4 Aspek Dasar *Extreme Programming***

Aspek dasar pada penerapan metode *extreme programming* yaitu :

1. *The Planning Game*

Proses pendek dan cepat, mengutamakan aspek teknik, memisahkan unsur bisnis dengan unsur teknis dan pertemuan intensif antara klien dengan developer. Pada XP proses ini menggunakan terminologi “game” karena Beck menyarankan untuk menggunakan teknik *score card* dalam menentukan *requirements*.

2. *Small Releases*

Menyelesaikan bagian –bagian aplikasi dan melakukan persentasi kepada *client*, setelah mendapatkan persetujuan maka dilakukan penerapan keaplikasi.

3. *Metaphor*

Menggambarkan visi yang luas terhadap tujuan dari pengembangan perangkat lunak. Dengan Tujuan diharapkan komunikasi antara klien dengan developer akan berlangsung lebih baik.

4. *Simple Design*

Menghindari desain yang rumit dalam sebuah pengembangan perangkat lunak. Dengan desain yang simpel apabila terjadi perubahan dapat meminimalkan kesalahan.

5. *Refactoring*

Melakukan perubahan pada kode program dari perangkat lunak dengan tujuan meningkatkan kualitas dari struktur program tersebut tanpa mengubah cara program tersebut bekerja.

6. *Testing*

Membuat test terhadap aplikasi berdasarkan model test yang telah ditentukan.

7. *Pair Programming*

Dua orang programmer saling bekerjasama di komputer yang sama untuk menyelesaikan sebuah unit.

8. *Colletive Ownership*

Saling berbagi pengetahuan agar tidak saling ketergantungan pada programmer tertentu ataupun berbagai hambatan akibat perbedaan gaya menulis program dapat diperkecil.

#### 9. *Coding Standard*

Dengan adanya coding standards yang telah disepakati terlebih dahulu maka pemahaman terhadap program akan menjadi mudah untuk semua programmer dalam tim.

#### 10. *Continuous Integration*

Melakukan build sesering mungkin berbagai kesalahan pada program dapat dideteksi dan diperbaiki secepat mungkin.

#### 11. *40-hours Week*

Beck berpendapat bekerja 8 jam sehari dan 5 hari seminggu adalah maksimal untuk tiap programmer.

#### 12. *On-Site Customer*

XP menganjurkan bahwa ada anggota dari klien yang terlibat pada proses pengembangan perangkat lunak. Apabila ada kesalahan dalam pengembangan diharapkan klien dapat segera memberikan masukan untuk koreksinya.

### **2.4.5 Tujuan *Extreme Programming***

Tujuan metode *extreme programming* untuk menghasilkan perangkat lunak yang berkualitas tinggi dan lebih produktif dan mengurangi biaya selama ada perubahan dalam pengembangan perangkat lunak menggunakan siklus pengembangan perangkat lunak singkat.

### **2.4.6 Daur Hidup Metodologi *Extreme Programming***

Metode XP dapat diterapkan bila:

1. Adanya perubahan yang sangat cepat
2. Memiliki resiko yang tinggi pada pembuatan aplikasi

3. Dalam tim pengembangan aplikasi dengan sedikit programmer
4. Mampu mengotomatisasikan uji sistem
5. Keterlibatan peran serta klien secara langsung
6. Harus ada komunikasi yang baik

#### **2.4.7 Keuntungan dan Kerugian *Extreme Programming***

Keuntungan pada penerapan metode XP yaitu:

1. Dalam hal XP menjalin komunikasi yang baik dengan klien pada pengembangan aplikasi
2. Saling menghargai antar developer dan meningkatkan komunikasi
3. Dapat menjadi pembelajaran bagi orang lain
4. Klien mendapatkan umpan balik yang akurat mengenai aplikasi yang dibuat
5. Dengan XP dapat mengubah pemikiran pelanggan terhadap aplikasi yang dibuat
6. Developer tidak berkerja secara berlebihan
7. Dengan XP dapat membuat keputusan yang bersifat teknikal

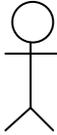
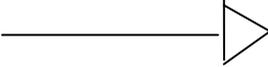
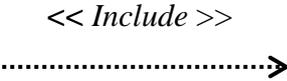
#### **2.5 *Unified Modelling Language (UML)***

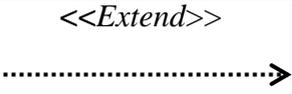
Menurut Rosa and Shalahuddin (2019) UML (*unified Modelling Language*) adalah bahasa visual untuk pemodelan dan komunikasi mengenai sebuah sistem dengan menggunakan diagram dan teks-teks pendukung. Berikut ini merupakan penjelasan tentang masing-masing diagram yang ada pada UML (*Unified Modelling Language*).

### 2.5.1 Use Case Diagram

Menurut Rosa and Shalahuddin (2019) *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. Berikut simbol-simbol yang akan digunakan dalam menggambarkan *Use Case Diagram* dapat dilihat pada tabel 2.1.

**Tabel 2.1** Simbol *Use Case Diagram*

No	Simbol	Deskripsi
1.		<i>Usecase</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
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.
		Ekstensi ( <i>extend</i> ) merupakan use case tambahan

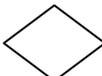
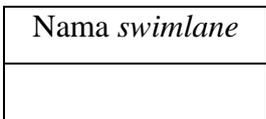
6.		ke sebuah use case yang ditambahkan dapat berdiri sendiri walau tanpa use case tambahan itu.
----	---	--

Sumber : (Rosa and Shalahuddin, 2019)

### 2.5.2 Activity Diagram

Menurut Rosa and Shalahuddin (2019) *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. Berikut simbol-simbol yang akan digunakan dalam menggambarkan *activity diagram* dapat dilihat pada tabel 2.2 berikut ini :

**Tabel 2.2** Simbol *Activity Diagram*

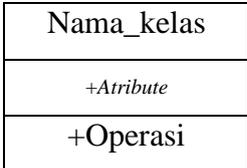
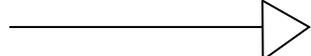
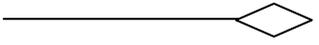
No.	Simbol	Keterangan
1.		Status awal aktivitas sistem, sebuah diagram aktivitas memiliki sebuah status awal.
2.		Aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja.
3.		Percabangan ( <i>Decision</i> ) merupakan asosiasi percabangan dimana jika ada pilihan aktivitas lebih dari 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)

### 2.5.3 Class Diagram

Menurut Rosa and Shalahuddin (2019) *Class diagram* mengembangkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem. Berikut simbol-simbol yang akan digunakan dalam menggambarkan *Class Diagram* dapat dilihat pada tabel 2.3.

**Tabel 2.3** 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)

## **2.6 Metode Pengujian Sistem**

Metode pengujian sistem merupakan metode yang digunakan untuk melakukan testing pada sistem yang dibangun sehingga di peroleh hasil berupa sistem yang sesuai fungsinya.

### **2.6.1 ISO 25010**

Menurut *International Organisation for Standardisation* (2011), ISO/IEC 25010 adalah Model kualitas sistem dan perangkat lunak yang menggantikan ISO/IEC 9126 tentang *software engineering* sebagai bentuk konsep pengujian terhadap kelayakan sistem yang telah dibangun.

*Product quality* ini juga digunakan untuk tiga model kualitas yang berbeda untuk produk perangkat lunak antara lain:

- a. Kualitas dalam model penggunaan,
- b. Model kualitas produk, dan
- c. Data model kualitas

Model kualitas produk terdiri dari delapan karakteristik yang berhubungan dengan sifat statis perangkat lunak dan sifat dinamis dari sistem komputer. Model ini berlaku untuk sistem komputer dan produk perangkat lunak. Karakteristik yang didefinisikan oleh kedua model tersebut relevan untuk semua produk perangkat lunak dan sistem komputer. Karakteristik dan subkarakteristik memberikan terminologi yang konsisten untuk menentukan, mengukur dan mengevaluasi kualitas sistem dan perangkat lunak. Mereka juga menyediakan seperangkat karakteristik kualitas yang sesuai dengan persyaratan kualitas yang dapat dibandingkan untuk kelengkapan.

## **2.6.2 Tahapan Pengujian Sistem**

Tahapan pengujian sistem digunakan untuk mengetahui proses pengujian yang akan dilakukan, berikut adalah tahapan pengujian ISO/IEC 25010:

### **1. *Functional Suitability***

Sejauh mana perangkat lunak mampu menyediakan fungsi yang memenuhi kebutuhan yang dapat digunakan dalam kondisi tertentu. Karakteristik ini dibagi menjadi beberapa karakteristik yaitu.

- a. *Functional completeness*, sejauh mana fungsi yang disediakan mencakup semua tugas dan tujuan pengguna secara spesifik.
- b. *Functional correctness*, sejauh mana produk atau sistem menyediakan hasil yang benar sesuai kebutuhan.
- c. *Functional appropriateness*, sejauh mana fungsi yang disediakan mampu memfasilitasi penyelesaian tugas dan tujuan tertentu.

### **2. *Compatibility***

Sejauh mana sebuah produk, sistem atau komponen dapat bertukar informasi dengan produk, sistem atau komponen dan/atau menjalankan fungsi lain yang diperlukan secara bersamaan ketika berbagi perangkat keras dan environment perangkat lunak yang sama. Karakteristik ini dibagi menjadi 2 karakteristik yaitu.

- a. *Co-existence*, sejauh mana produk atau sistem dapat menjalankan fungsi yang dibutuhkan secara efisien sementara berbagi sumber daya dengan produk atau sistem yang lain tanpa merugikan produk atau sistem tersebut.

- b. *Interoperability*, sejauh mana dua atau lebih produk, sistem atau komponendapat bertukar informasi dan menggunakan informasi tersebut.

### 3. *Usability*

Sejauh mana sebuah produk atau sistem dapat digunakan oleh user tertentu untuk mencapai tujuan dengan efektif, efisiensi, dan kepuasan tertentu dalam konteks penggunaan. Karakteristik ini terbagi menjadi beberapa karakteristik yaitu.

- a. *Appropriateness recognizability*, sejauh mana pengguna dapat mengetahui apakah sistem atau produk sesuai kebutuhan mereka.
- b. *Learnability*, sejauh mana produk atau sistem dapat digunakan oleh pengguna untuk mencapai tujuan tertentu yang belajar menggunakan sistem atau produk dengan efisien, efektif, kebebasan dari resiko dan kepuasan dalam konteks tertentu.
- c. *Operability*, sejauh mana produk atau sistem mudah dioperasikan dan dikontrol.
- d. *User error protection*, sejauh mana produk atau sistem melindungi pengguna terhadap membuat kesalahan.
- e. *User interface aesthetics*, sejauh mana antarmuka pengguna dari produk atau sistem memungkinkan interaksi yang menyenangkan dan memuaskan pengguna.

- f. *Accessibility*, sejauh mana produk atau sistem dapat digunakan oleh semua kalangan untuk mencapai tujuan tertentu sesuai konteks penggunaan.

#### **4. *Reliability***

Sejauh mana sebuah sistem, produk atau komponen dapat menjalankan fungsi tertentu dalam kondisi tertentu selama jangka waktu yang ditentukan.

Karakteristik ini terbagi menjadi beberapa subkarakteristik yaitu.

- a. *Maturity*, sejauh mana produk atau sistem mampu memenuhi kebutuhan secara handal di bawah keadaan normal.
- b. *Availability*, sejauh mana produk atau sistem siap beroperasi dan dapat diakses saat perlu digunakan.
- c. *Fault tolerance*, sejauh mana produk atau sistem tetap berjalan sebagaimana yang dimaksud meskipun terjadi kesalahan pada perangkat keras atau perangkat lunak.
- d. *Recoverability*, sejauh mana produk atau sistem mampu dapat memulihkan data yang terkena dampak secara langsung dan menata ulang kondisi system seperti yang diinginkan ketika terjadi gangguan.

#### **5. *Security***

Sejauh mana sebuah produk atau sistem melindungi informasi dan data sehingga seseorang atau sistem lain dapat mengakses data sesuai dengan jenis dan level otorisasi yang dimiliki. Karakteristik ini terbagi menjadi beberapa karakteristik yaitu.

- a. *Confidentiality*, sejauh mana produk atau perangkat lunak memastikan data hanya bisa diakses oleh mereka yang berwenang untuk memiliki akses.

- b. *Integrity*, sejauh mana produk atau perangkat lunak mampu mencegah akses yang tidak sah untuk memodifikasi data.
- c. *Non-repudiation*, sejauh mana peristiwa atau tindakan dapat dibuktikan telah terjadi, sehingga tidak ada penolakan terhadap peristiwa atau tindakan tersebut.
- d. *Accountability*, sejauh mana tindakan dari suatu entitas dapat ditelusuri secara unik untuk entitas.
- e. *Authenticity*, sejauh mana identitas subjek atau sumber daya dapat terbukti menjadi salah satu yang diklaim.

## **6. *Portability***

Sejauh mana keefektifan dan efisiensi sebuah sistem, produk atau komponen dapat dipindahkan dari satu perangkat keras, perangkat lunak atau digunakan pada lingkungan yang berbeda. Karakteristik ini dibagi menjadi beberapa karakteristik yaitu.

- a. *Adaptability*, sejauh mana produk atau sistem dapat secara efektif dan efisien disesuaikan pada perangkat lunak, perangkat keras dan lingkungan yang berbeda.
- b. *Installability*, sejauh mana produk atau sistem dapat berhasil dipasang atau dihapus dalam lingkungan tertentu.
- c. *Replaceability*, sejauh mana produk atau sistem dapat menggantikan produk atau sistem lain yang ditentukan untuk tujuan yang sama pada lingkungan yang sama.

## **7. *Performance Efficiency***

Kinerja relatif terhadap sumber daya yang digunakan dalam kondisi tertentu.

Karakteristik ini terbagi menjadi beberapa subkarakteristik yaitu.

- a. *Time behaviour*, sejauh mana respon dan pengolahan waktu produk atau sistem dapat memenuhi persyaratan ketika menjalankan fungsi.
- b. *Resource utilization*, sejauh mana jumlah dan jenis sumber daya yang digunakan oleh produk atau sistem dapat memenuhi persyaratan ketika menjalankan fungsi.
- c. *Capacity*, sejauh mana batas maksimum parameter produk atau sistem dapat memenuhi persyaratan.

## **8. *Maintainability***

Sejauh mana keefektifan dan efisiensi dari sebuah produk atau sistem dapat dirawat. Karakteristik ini terbagi menjadi beberapa subkarakteristik yaitu.

- a. *Modularity*, sejauh mana sistem terdiri dari komponen terpisah sehingga perubahan atau modifikasi pada salah satu komponen tersebut memiliki dampak yang kecil terhadap komponen yang lain.
- b. *Reusability*, sejauh mana aset dapat digunakan lebih oleh satu sistem atau digunakan untuk membangun aset lain.
- c. *Analyzability*, tingkat efektivitas dan efisiensi untuk mengkaji dampak perubahan pada satu atau lebih bagian-bagian produk atau sistem, untuk mendiagnosis kekurangan atau penyebab kegagalan produk, untuk mengidentifikasi bagian yang akan diubah.
- d. *Modifiability*, sejauh mana produk atau sistem dapat dimodifikasi secara efektif dan efisien tanpa menurunkan kualitas produk yang ada.

- e. *Testability*, tingkat efektivitas dan efisiensi untuk membentuk kriteria uji dari produk, sistem atau komponen dan uji dapat dilakukan untuk menentukan apakah kriteria tersebut telah terpenuhi.