

## **BAB II**

### **LANDASAN TEORI**

#### **2.1. Studi Literatur**

Dalam penelitian “Pemanfaatan Asynchronous Learning Sebagai Inovasi Pembelajaran Mandiri Antarmuka Pada Pemrograman Game” terdapat penelitian terdahulu yang relevan dengan permasalahan yang akan diteliti, diantaranya sebagai berikut :

1. Dalam penelitian yang dilakukan oleh Samuel Gandang Gunanto yang berjudul “Game-Based Learning: Media Konstruktif Pembelajaran Mandiri bagi Siswa” menyatakan bahwa, Game dapat menjadi salah satu media alternatif yang mampu menerapkan komponen belajar yang sistematis. Hal ini juga membantu pendekatan materi belajar ke siswa dengan cara yang lebih alami. Sehingga proses pembelajaran pun menjadi lebih menyenangkan dan terjadi secara sukarela.(Gunanto, 2021)
2. Dalam penelitian yang dilakukan oleh Sonsoles López-Pernas, dkk yang berjudul “Putting It All Together: Combining Learning Analytics Methods and Data Sources to Understand Students’ Approaches to Learning Programming” menghasilkan pernyataan bahwa, dalam proses pembelajaran pemrograman siswa lebih memilih menggunakan banyak sumber daya pada berbagai tahap proses pembelajaran mereka. Sarana berbasis teks lebih disukai siswa karena dapat melakukan copy paste dari sumber tersebut. Sedangkan sarana berbasis video tidak sering digunakan oleh siswa. Jika siswa menemui kesulitan saat mengerjakan tugas, mereka lebih tertarik melemparkan pertanyaan ke forum untuk mencari bantuan. Mengetahui hal tersebut peneliti menemui beberapa cara untuk membantu siswa dalam proses pembelajaran, seperti memotivasi dan memfasilitasi siswa untuk menonton video dengan bantuan subtitle atau menambahkan fitur pencarian teks agar siswa merasa terbantu dengan adanya sarana berbasis video. Kemudian, mengajarkan siswa untuk berkolaborasi guna menyelesaikan tugas yang ada. Dan yang terakhir yaitu, sarana pendukung untuk pembelajaran siswa seperti LMS. (López-Pernas et al., 2021)

3. Dalam penelitian yang dilakukan oleh Georgiana Haldeman dan temantemannya yang berjudul “Providing Meaningful Feedback for Autograding of Programming Assignments” menyatakan bahwa, Autograding semakin banyak digunakan saat ini untuk memenuhi tantangan mengajar pemrograman. Pada penelitiannya menyajikan metodologi yang akan membantu menghilangkan kesenjangan antara autograding dan penilaian pengetahuan tugas pemrograman. Dengan cara memberikan umpan balik kepada siswa, Sehingga siswa dapat mengetahui letak kesalahan yang ada. (Haldeman et al., 2018)

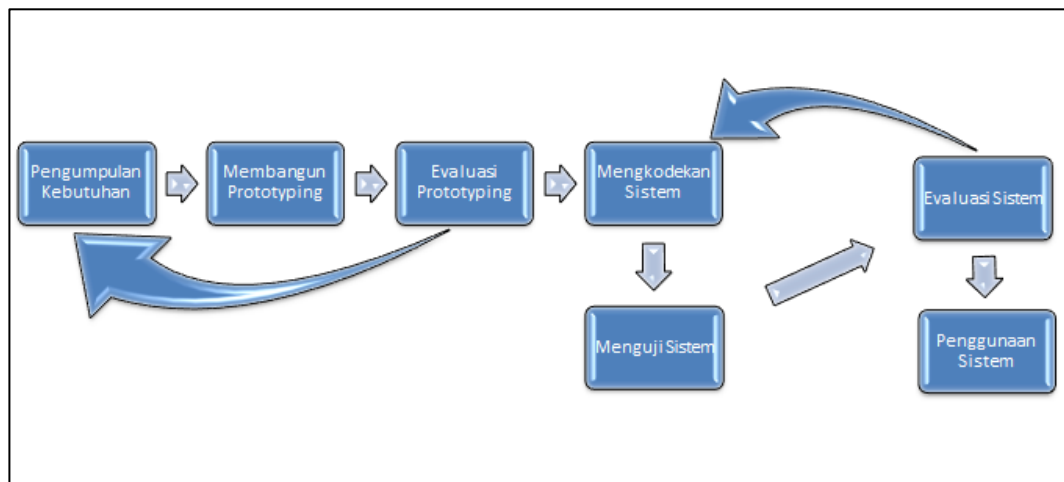
Dari beberapa penelitian terdahulu diatas, penulis menemukan ide untuk membuat penelitian yang berjudul “Pemanfaatan Asynchronous Learning Sebagai Inovasi Pembelajaran Mandiri Antarmuka Pada Pemrograman Game”. Pada penelitian ini memiliki tujuan untuk membantu siswa SMK untuk dapat membuat antarmuka game secara mandiri dengan materi yang dapat diakses kapanpun dan dimanapun. Tak hanya membantu para siswa untuk melakukan pembelajaran secara mandiri, Hal ini juga dapat membantu para guru dalam melakukan penyebaran materi, melihat perkembangan siswa, serta melakukan penilaian otomatis. Karena materi akan disebarakan melalui website dan nilai dari siswa pun akan muncul disana, sehingga guru dapat melihat perkembangan siswa dan terbantu dalam melakukan penilaian.

## **2.2. Dasar Teori**

### **2.2.1. Metode Prototype**

Dari beberapa pilihan metode pengembangan perangkat lunak, Penelitian ini akan menggunakan prototype model dalam pengerjaannya. Di model prototype, interaksi antara pengembang dan pelanggan sangat dibutuhkan selama proses penciptaan perangkat lunak. Sering kali, pelanggan hanya mendefinisikan perangkat lunak yang di inginkan secara umum, tidak mengidentifikasi proses input, proses alur kerja sistem, dan proses output yang rinci. (Pressman, 2000) Metode ini memberikan suatu kerangka kerja yang terstruktur untuk memecahkan masalah dengan langkah-langkah yang jelas dan terurut secara sistematis, sehingga memudahkan dalam mengidentifikasi dan mengatasi permasalahan yang dihadapi. Oleh karena itu, dengan adanya metode ini, akan membantu dalam menyelesaikan

permasalahan yang ada dengan mengikuti tahapan-tahapan yang tercantum pada gambar 2.1 di bawah ini.



Gambar 2. 1 Metode Prototype

Pada gambar diatas maka tahapan prototype bisa dijelaskan sebagai berikut :

1. Pengumpulan Kebutuhan. Pada tahapan awal ini dilakukannya identifikasi seluruh kebutuhan yang diperlukan dari sistem yang akan dibangun.
2. Membangun Prototyping. Tahapan selanjutnya yaitu membuat prototype atau gambaran sistem input dan output yang akan dibangun guna membantu pemahaman pelanggan.
3. Evaluasi Prototyping. Setelah itu pelanggan akan melakukan penilaian terhadap prototype yang telah dibuat dengan tolak ukur kesesuaian akan kebutuhan yang diperlukan.
4. Mengkodekan Sistem. Jika tahapan 3 diatas belum tepat dengan kebutuhan pelanggan, maka diulang. Tetapi jika sudah tepat, maka akan dilakukannya tahapan ini yaitu membuat prototype yang disepakati menjadi bahasa pemrograman yang sesuai.
5. Menguji Sistem. Setelah prototype sudah menjadi perangkat lunak, maka dilakukannya pemeriksaan sebelum digunakan. Biasanya pemeriksaan fitur dilakukan dengan pengujian black box, System Usability Scale (SUS), ataupun pengujian arsitektur lainnya.
6. Evaluasi Sistem. Selepas sistem diuji akan menghasilkan kesimpulan bahwa sistem sudah sesuai dengan kebutuhan atau belum. Jika belum sesuai, maka

tahapan 4 dan 5 diulangi. Jika sudah sesuai, maka tahapan ke 7 boleh dilaksanakan.

7. Penggunaan / Implementasi Sistem. Semua tahapan penciptaan sudah dilakukan dan sudah sesuai dengan kebutuhan pelanggan, Maka sistem sudah bisa digunakan.

### **2.2.2. Asynchronous Learning**

Terdapat beberapa cara pembelajaran salah satunya yaitu Asynchronous Learning. Asynchronous Learning atau pembelajaran asinkron merupakan salah satu proses pembelajaran yang proses pembelajarannya dilakukan secara tunda. Tunda yang dimaksud yaitu, pembelajar dan pendidik tidak harus berada di tempat yang sama dan tidak harus pula online pada waktu yang sama. Sedangkan materi pembelajaran dapat di unggah dan di akses melalui media pembelajaran. Dapat disimpulkan bahwa pembelajar dan pendidik tidak harus berada pada waktu dan tempat yang sama karena materi pembelajarannya dapat diakses melalui media yang digunakan, seperti e-learning dan grup pada sosial media.(Sulistio, 2021)

### **2.2.3. Pemrograman Game**

Menurut buku Belajar Membuat Game 2D dan 3D Menggunakan Unity, setuju dengan pernyataan dari IGDA (Internasional Game Developers Association) Education SID, bahwa game merupakan suatu kegiatan yang mengandung beberapa rules. Game juga merupakan bentuk permainan yang seringkali melibatkan konflik dengan pemain lain, bisa juga dengan sistem itu sendiri, ataupun dengan acak (keberuntungan). Biasanya game memiliki tujuan dengan adanya titik mulai dan titik akhir, tetapi tidak semua. (Siti & Putri, n.d.) Adapula beberapa tahapan yang harus disiapkan dalam membuat game seperti, membuat skenario game, menentukan tema atau tujuan game, menentukan rules atau aturan yang diterapkan pada game, dan memikirkan masalah atau konflik yang akan dibuat dalam game.

Game engine adalah media yang dirancang guna penciptaan dan pengembangan game. Media ini sudah dilengkapi dengan fitur-fitur yang bisa mempermudah pembuatan atau pengembangan game. Tak hanya itu, Dengan adanya game engine bisa juga membantu pengurangan biaya, kompleksitas, dan waktu dalam pembuatan atau pengembangan game. Unity merupakan salah satu game engine yang saat ini banyak digemari para pengembang game. Karena unity

menyediakan banyak keunggulan dibandingkan game engine lainnya, untuk saat ini. Unity mendukung game dalam 2D dan 3D. Para developer game yang menggunakan unity seringnya menggunakan C# dalam pembuatan atau pengembangan game. C# merupakan salah satu bahasa pemrograman yang populer untuk melakukan pembuatan atau pengembangan game. Secara historis script editor bawaan unity adalah MonoDevelop. Tetapi mulai unity 2018.1, Script editor bawaan unity adalah Visual Studio. (Halpern, 2019)

User Interface (UI) adalah desain antarmuka antara pengguna dan mesin atau perangkat lunak seperti komputer, handphone dan alat elektronik lainnya yang fungsi utamanya adalah memaksimalkan pengalaman pengguna. (Wibowo & Tan, 2021) Pemanfaatan user interface pada game sangat penting, dengan penerapan Graphic User Interface (GUI). GUI merupakan antarmuka yang dapat menjembatani sistem dengan pengguna, agar pengguna mudah dalam pengaplikasian sistem. (Assiroj et al., 2018) Desain user interface game berbeda dengan desain antarmuka media yang lain, karena game mengandung lebih banyak fiksi yang menyangkut player atau pengguna game. Desain antarmuka pada game pastinya memengaruhi tingkat kenyamanan dan seberapa banyak pemain menikmati game tersebut. (Wibowo & Tan, 2021)

#### **2.2.4. Computer Assistant Learning**

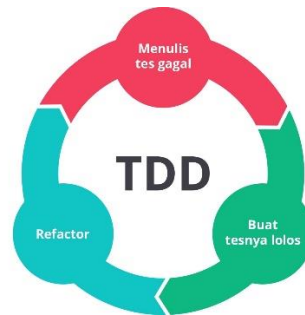
Pengujian bagian terkecil dari sebuah code bisa disebut dengan unit testing. Hal ini dilakukan untuk menemukan pembuktian kode yang ditulis sudah sesuai dengan kehendak programmer. Setelah programmer selesai menuliskan suatu kode/fungsi/method di dalam suatu class, maka unit testing ini diluncurkan. Bisa juga setelah programmer menambahkan suatu fungsionalitas atau refactoring unit testing dilakukan. (Salma, Ayu Aldila, n.d.)

Unit testing bisa dilakukan menggunakan tools tertentu. Ada beberapa unit testing tools yang sering digunakan, seperti JUnit, NUnit, JMockit, EMMA, dan PHPUnit. Materi pembelajaran yang akan dibangun pada penelitian ini akan menggunakan bahasa pemrograman C#, Maka unit testing tools yang digunakan yaitu NUnit. Menurut penjelasan NUnit dari website resminya, dijelaskan bahwa NUnit adalah framework yang digunakan untuk melakukan pengujian bahasa pemrograman .Net. Saat ini NUnit sudah memiliki beberapa versi yang tentunya

semakin membantu programmer, karena memiliki banyak fitur untuk berbagai platform .Net. (NUnit, n.d.)

### 2.2.5. Test Driven Development (TDD)

Perangkat lunak dikatakan berhasil apabila dapat memenuhi kebutuhan yang pengguna inginkan. Kemudian, bisa juga memudahkan dalam memberikan perubahan dan menemukan kesalahan. Perkara itu bisa diatasi oleh metode *Test Driven Development* (TDD). Biasanya testing dilakukan setelah fitur selesai dibuat, Tetapi berbeda dengan metode TDD yang membuat *test case* terlebih dahulu kemudian *producing code* dan *refactoring*. (Thohari & Amalia, 2018) Hal tersebut bisa digambarkan seperti pada gambar di bawah ini.



Gambar 2. 2 Metode Test Driven Development

Pada gambar di atas maka dapat dijelaskan seperti :

1. Warna merah menggambarkan bahwa kode tidak bekerja
2. Warna hijau menggambarkan bahwa kode sudah bekerja tetapi belum optimal
3. Warna biru atau refactor menggambarkan bahwa kode yang sudah dapat bekerja tadi dilakukan *refactoring*. Contohnya seperti menghilangkan redundansi.

Hal tersebut bisa membantu mempercepat waktu penciptaan atau pengembangan sistem dan mengurangi adanya redundansi pada kode yang ada. Implementasi TDD dapat membantu siswa dalam pembelajaran dan membantu guru dalam memperbaiki kinerjanya. Guru akan memfasilitasi siswa dalam pemahaman pembelajaran, sehingga hasil belajar siswa dapat meningkat. Selain itu dengan pola mengajar guru yang tepat juga dapat memperbaiki perilaku siswa agar semakin lebih baik dan meningkatkan mutu dan kualitas pembelajaran di kelas. Mengingat dari tujuannya yaitu ingin membuat

perubahan terhadap siswa dan sistem yang ada di sekolah ke arah yang lebih baik.(Prihantoro & Hidayat, 2019)

#### **1.2.6. BlackBox Testing**

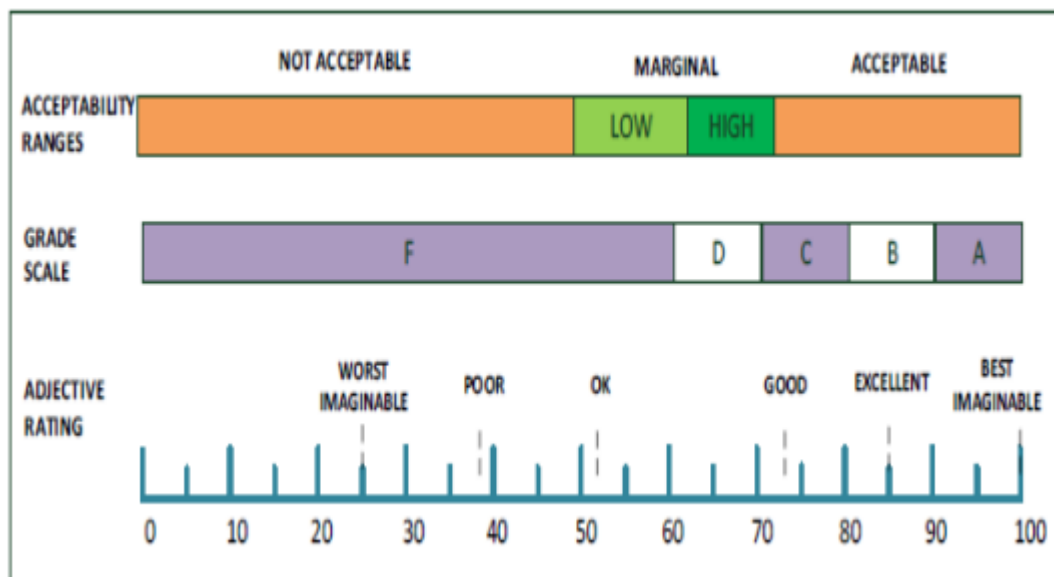
*Blackbox Testing* adalah metode pengujian yang difokuskan pada fungsionalitas perangkat lunak. Dalam metode ini, penguji dapat menentukan serangkaian kondisi input yang akan digunakan untuk menguji spesifikasi fungsionalitas program. Penguji tidak perlu mengetahui atau memperhatikan detail implementasi internal dari perangkat lunak, tetapi lebih berfokus pada perilaku dan output yang dihasilkan sesuai dengan spesifikasi yang telah ditentukan(Mustaqbal et al., n.d.). *Blackbox testing* merupakan metode pengujian yang dapat menggunakan berbagai teknik, seperti pengujian kasus uji, pengujian batasan, pengujian kesalahan, dan pengujian fungsional. Metode ini memiliki kelebihan dan kekurangan tertentu. Kelebihannya antara lain, tidak diperlukan pengetahuan tentang bahasa pemrograman tertentu, pengujian dilakukan dari perspektif pengguna yang membantu mengungkapkan ambiguitas atau inkonsistensi dalam spesifikasi persyaratan, serta adanya ketergantungan yang saling menguntungkan antara programmer dan tester. Namun, terdapat juga beberapa kekurangan pada blackbox testing, seperti kesulitan merancang kasus uji tanpa spesifikasi yang jelas, kemungkinan pengulangan uji yang sudah dilakukan oleh programmer, dan beberapa bagian back-end yang tidak diuji sama sekali.(Jaya, 2018)

#### **1.2.7. SUS**

Metode *System Usability Scale (SUS)* digunakan untuk mengukur tingkat kepuasan pengguna terhadap suatu sistem atau produk perangkat lunak. Metode ini membantu dalam memahami masalah yang dihadapi oleh pengguna saat menggunakan sistem. Metode yang dikembangkan oleh John Brooke pada tahun 1996 ini terdiri dari sepuluh pertanyaan sederhana yang mengenai kegunaan sistem. SUS membantu dalam memperoleh gambaran umum mengenai kegunaan produk dan pada akhir pengujian akan dilakukan perhitungan skor untuk mengevaluasi hasil(Kaya et al., 2019). Pengguna diminta untuk memberikan tanggapan menggunakan skala likert yang terdiri dari 5 poin, yang digunakan untuk

mengevaluasi sejauh mana mereka setuju atau tidak setuju dengan pernyataan yang diberikan.

Hasil dari pengujian SUS dihitung berdasarkan perolehan skor dari setiap pernyataan yang dijawab oleh pengguna. Perhitungan dilakukan dengan ketentuan pertanyaan dengan nomor ganjil, skor akan dikurangi 1 ( $X-1$ ), sedangkan untuk pertanyaan dengan nomor genap, skor akan dikurangi dari 5 ( $5-X$ ). Setelah itu, nilai skor dari setiap pertanyaan akan dijumlahkan, kemudian hasil penjumlahan tersebut akan dikalikan dengan faktor pengali 2,5. Skor tersebut kemudian diubah menjadi skor SUS yang berada dalam rentang 0 hingga 100 dengan cara mencari rata-rata dari semua hasil pengujian responden. Skor SUS tinggi menunjukkan tingkat kepuasan pengguna terhadap sistem atau produk. Terdapat *acceptability ranges* pada penilaian hasil SUS diantaranya yaitu *not acceptable*, *marginal* dan *acceptable*. Kemudian terdapat *grade scale* yang berisikan *grade* A, B, C, D, dan F. *Grade A* memiliki rentang skor lebih besar atau sama dengan 80,3. *Grade B* memiliki rentang skor lebih besar sama dengan 74 dan lebih kecil 80,3. *Grade C* memiliki rentang skor lebih bear 68 dan lebih kecil 74. *Grade D* memiliki rentang skor lebih besar sama dengan 51 dan lebih kecil 68. Yang terakhir yaitu *grade F* yang memiliki rentang skor lebih kecil dari 51. Selain itu, terdapat juga adjektif rating yang mencakup *worst imaginable*, *poor*, *ok*, *good*, *excellent*, dan *best imaginable* yang dapat dilihat pada gambar 2.2.



Gambar 2. 3 Penentuan Hasil Penilaian SUS



Pengukuran usability dilakukan untuk menilai interaksi antara pengguna dengan aplikasi. Pengukuran dilakukan mengikuti konsep user testing, dengan penekanan pada pengukuran dan bukan pengujian. Pengukuran dilakukan dalam rangka mengidentifikasi permasalahan usability yang dapat mempengaruhi interaksi sistem (perangkat lunak) dengan pengguna pada hasil perancangan aplikasi. (Huda, 2019). Hasil dari kuesioner yang ada digunakan untuk mendapatkan gambaran kepuasan responden akan pengoperasian aplikasi.

