Highschool Simulator

Project Planning

Version 3.0

Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| **Date** | **Version** | **Description** | **Author** |
| 19/02/2025 | 1.0 (Draft) | Project Planning pertama | Shevchenko Saviour Matthew Poluan |
| 24/02/2025 | 2.0(Revisi) | Revisi 1 | Shevchenko Saviour Matthew Poluan |
| 18/03/2025 | 3.0 | Revisi 2 | Shevchenko Saviour Matthew Poluan |

Contents

[1. Introduction 4](#_Toc197308188)

[1.1 Overview 4](#_Toc197308189)

[1.2 Goal 4](#_Toc197308190)

[1.3 Objectives 4](#_Toc197308191)

[1.4 Deliverables 5](#_Toc197308192)

[1.5 Assumption and Constraints 6](#_Toc197308193)

[2. Management Structure 6](#_Toc197308194)

[2.1 Project Lifecycle 7](#_Toc197308195)

[2.2 Project Organization 7](#_Toc197308196)

[2.2.1 External Interface 7](#_Toc197308197)

[2.2.2 Internal Structure 7](#_Toc197308198)

[2.2.3 Roles and Responsibility 8](#_Toc197308199)

[2.2.4 Staffing 9](#_Toc197308200)

[2.3 Communication 10](#_Toc197308201)

[2.4 Risk and Asset Management 10](#_Toc197308202)

[2.5 Start up 12](#_Toc197308203)

[2.6 Customer Identification 13](#_Toc197308204)

[2.7 Measures of Success 13](#_Toc197308205)

[2.8 Validation and Acceptance 13](#_Toc197308206)

[2.9 Support 14](#_Toc197308207)

[2.10 Close Out 14](#_Toc197308208)

[3. Planning and Control 14](#_Toc197308209)

[3.1 Estimate 14](#_Toc197308210)

[3.2 Resource Identification 15](#_Toc197308211)

[3.2.1 Staffing 15](#_Toc197308212)

[3.2.2 Time 16](#_Toc197308213)

[3.2.3 Materials 16](#_Toc197308214)

[3.2.4 Work Breakdown Structure 16](#_Toc197308215)

[3.2.5 Schedule 17](#_Toc197308216)

[3.2.6 Budget 18](#_Toc197308217)

[3.3 Tracking and Control 19](#_Toc197308218)

[4. Technical Process 20](#_Toc197308219)

[4.1 Engineering 20](#_Toc197308220)

[4.1.1 Environment 20](#_Toc197308221)

[4.1.2 Methods, Tools and Techniques 20](#_Toc197308222)

[4.2 Technology 21](#_Toc197308223)

[4.2.1 Environment 21](#_Toc197308224)

[4.2.2 Methods Tools and Techniques 21](#_Toc197308225)

[4.3 Infrastructure 22](#_Toc197308226)

[4.4 Code Growth Estimation 23](#_Toc197308227)

[4.5 Project Artifacts 23](#_Toc197308228)

[5. Supporting Plan 23](#_Toc197308229)

[5.1 Administration 23](#_Toc197308230)

[5.2 Configuration Management 24](#_Toc197308231)

[5.3 Quality Assurance 24](#_Toc197308232)

[5.4 Testing 24](#_Toc197308233)

[5.5 Deployment 25](#_Toc197308234)

[5.6 Procurement 25](#_Toc197308235)

[5.7 Training 25](#_Toc197308236)

[5.8 Installation Support 26](#_Toc197308237)

[5.9 Maintenance 26](#_Toc197308238)

[5.10 Staff Development 26](#_Toc197308239)

[5.11 Product Acceptance 27](#_Toc197308240)

[5.12 Documentation 27](#_Toc197308241)

[6. Lesson Learned 27](#_Toc197308242)

# Introduction

## Overview

Dalam era digital saat ini, game bukan hanya menjadi hiburan semata, tetapi juga memiliki potensi besar sebagai media edukasi. Salah satu tantangan dalam dunia pendidikan adalah bagaimana membuat pembelajaran lebih menarik dan interaktif, terutama bagi generasi muda yang sudah akrab dengan teknologi. Berdasarkan Sustainable Development Goals (SDG) 4, yaitu Pendidikan Berkualitas, diperlukan inovasi dalam metode pembelajaran untuk meningkatkan pemahaman dan keterlibatan siswa.

Sebagai bagian dari mata kuliah Game Design, Authoring, and Scripting, kami mengembangkan sebuah proyek game yang berfokus pada simulasi sekolah. Game ini bertujuan untuk memberikan pengalaman interaktif mengenai kehidupan sekolah, termasuk aspek akademik dan sosial yang dihadapi oleh siswa. Game kami dikembangkan agar dapat menjadi sarana edukatif yang menyenangkan serta membantu pemain memahami pentingnya manajemen waktu, belajar, dan berinteraksi dalam lingkungan sekolah.

Kami menerapkan metode *Agile Software Engineering* untuk memastikan fleksibilitas dan efektivitas dalam pengembangan. Pendekatan ini, membantu tim dapat beradaptasi dengan perubahan kebutuhan dan meningkatkan kualitas game berdasarkan umpan balik yang diterima. Game ini dikembangkan sebagai software *stand-alone*, yang dapat dijalankan langsung di perangkat pengguna tanpa memerlukan koneksi internet atau platform tambahan.

## Goal

Tujuan dari pembuatan **Project Planning** ini adalah untuk menyusun rencana kerja yang jelas dan terperinci sebagai acuan dalam proses pengembangan **Highschool Simulator**. Dokumen ini membantu tim dalam mengatur jadwal, membagi tugas, serta memastikan bahwa setiap tahap pembangunan berjalan sesuai rencana agar proyek dapat diselesaikan tepat waktu dan sesuai dengan target yang telah ditetapkan.

## Objectives

 **Menentukan Jadwal Pengembangan**: Membuat timeline yang mencakup seluruh tahap pengembangan, mulai dari perencanaan, desain, implementasi, hingga pengujian.

 **Membagi Tugas Secara Efektif:** Menentukan peran dan tanggung jawab masing-masing anggota tim agar pekerjaan lebih terstruktur dan efisien.

 **Menyiapkan Sumber Daya yang Diperlukan:** Mengidentifikasi kebutuhan seperti perangkat lunak, alat bantu, serta dokumentasi yang diperlukan selama pengembangan.

 **Mengelola Risiko dan Kendala**: Mengantisipasi kemungkinan hambatan serta menyiapkan solusi untuk memastikan proyek tetap berjalan sesuai rencana.

 **Menetapkan Tolok Ukur Keberhasilan**: Mendefinisikan indikator pencapaian di setiap tahap agar progres proyek dapat dievaluasi dengan jelas.

 **Memastikan Kualitas dan Keselarasan dengan Tujuan**: Melakukan review berkala untuk memastikan game yang dikembangkan sesuai dengan konsep dan standar yang diharapkan.

## Deliverables

Pada bagian ini jelaskan mengenai bagaimana bentuk akhir produk kalian, dan bagaimana caranya produk tersebut dapat diserahkan ke tangan client. Buat juga jadwal *delivery* dengan table seperti contoh di bawah.

**Top level Deliverables**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Planning | Prototype | Alpha Version | Beta Version | Final Version |
| Q4  January | Q2 – Q4  February | Q1-Q2  March | Q3-Q4  March | Q3  April |

|  |  |
| --- | --- |
| Deliverables | Deskripsi |
| Planning | Menentukan ruang lingkup, mekanik, dan visi keseluruhan sebelum mulai pengembangan. |
| Prototype | Menguji kelayakan mekanik inti sebelum melanjutkan ke pengembangan penuh. |
| Alpha version | Memastikan mekanik berfungsi seperti yang diharapkan; mungkin masih ada bug dan konten yang belum lengkap |
| Beta Version | Melakukan pengujian akhir untuk menyempurnakan pengalaman pemain, memperbaiki bug besar, dan menyempurnakan keseimbangan gameplay. |
| Final Version | Merilis game dengan kualitas terbaik dan memastikan pengalaman pemain optimal. |

## Assumption and Constraints

* **Metode Pengembangan** – Menggunakan metode **Agile**, dengan iterasi dan evaluasi berkala untuk memastikan progres berjalan sesuai target.
* **Sumber Daya Terbatas** – Pengembangan dilakukan dengan tools yang tersedia dan tidak bergantung pada software berbayar yang mahal.
* **Batas Waktu** – Proyek harus diselesaikan dalam **16 minggu**, sehingga fitur yang dikembangkan harus realistis dan sesuai dengan timeline.
* **Ukuran dan Performa Game** – Game harus dioptimalkan agar berjalan lancar di spesifikasi PC kelas menengah.

# Management Structure

## Project Lifecycle

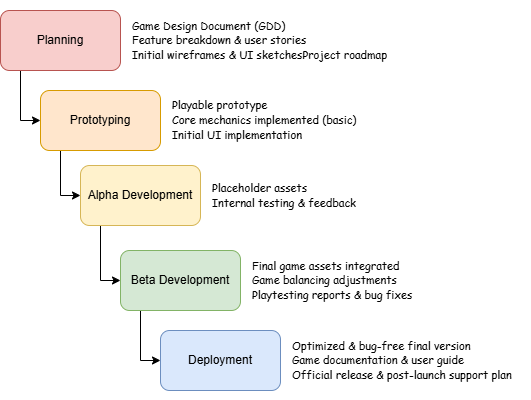


Figure 1 Project Lifecycle

## Project Organization

### External Interface

Komunikasi antar anggota tim dilakukan secara jarak jauh, mengingat setiap anggota bekerja dari rumah masing-masing dan hanya bertemu langsung saat kelas. Untuk memastikan koordinasi tetap efektif, tim menggunakan *WhatsApp* sebagai platform utama dalam berkomunikasi, baik untuk diskusi harian, pembagian tugas, maupun update progres.

Komunikasi dengan klien dilakukan melalui platform *Discord* untuk diskusi online, penyampaian progres, serta konsultasi terkait proyek. Selain itu, pertemuan langsung juga dilakukan sesuai dengan jadwal yang telah disepakati untuk mendapatkan umpan balik lebih mendetail dan memastikan proyek berjalan sesuai harapan.

### Internal Structure

Notes:

*Lead* (L) 🡪 Merupakan orang yang bertanggung jawab penuh pada bagiannya.

*Backup* (B) 🡪 Merupakan orang yang membantu atau menggantikan tugas *Lead*, apabila orang yang bertugas sebagai *Lead* sudah ada tugas lain atau berhalangan.

|  |  |  |
| --- | --- | --- |
| *Name* | *Technical* | *Management* |
| Daniel Prawira | (L) Software Developer | (L) Software Configuration management  (B) Software Quality Assurance |
| Dave Geraldo Jonathan Koro | (L) Sound Designer | (B) Requirement Management |
| Joel Daniel Mark Wenas | (L) Tester | (L) Software Quality Assurance |
| Kayla Fakhira | (L) Graphic Artist | (L) Plan tracker |
| Shevchenko Saviour Matthew Poluan | (L)  (B) Software Developer | (L) Project Manager  (B) Plan Tracker |
| Syedha Audry Timothy Piri | (B) Sound Designer | (L) Requirement Management  (B) Project manager |

### Roles and Responsibility

| Title | Role | Responsibility |
| --- | --- | --- |
| Software Developer | Writes and maintains the code, implementing features and ensuring functionality. | Implements core gameplay mechanics like studying, exams, NPC interactions, and time progression. Ensures smooth player controls and game logic. |
| Sound Designer | Creates and integrates sound effects, background music, and audio elements into the project. | Creates background music for different school settings (classroom, cafeteria, etc.), sound effects for interactions, and UI sounds. |
| Tester | Identifies bugs, tests features, and ensures the software runs as expected. | Conducts playtesting, checks for bugs, ensures mechanics like scheduling and exams work properly, and verifies UI/UX usability. |
| Graphic Artist | Designs visual elements such as character models, UI, and game environments | Designs character sprites, school environments, UI elements, and animations to match the game’s theme. |
| Software Configuration Management | Manages version control, tracks changes, and ensures consistency across development. | *Manages version control (e.g., using GitHub), keeps track of asset and code changes, and ensures the latest build is stable.* |
| Software Quality Assurance | Ensures the software meets quality standards through testing and process improvement. | Defines quality standards, ensures the game runs smoothly without major issues, and reviews if the game meets project goals. |
| Plan Tracker | Monitors the project timeline, ensuring tasks are completed on schedule. | Monitors development progress, checks if tasks are completed on time, and ensures the team follows the project timeline. |
| Project Manager | Oversees the entire project, coordinates the team, manages risks, and ensures goals are met. | Assigns tasks, resolves roadblocks, manages communication, and ensures the game is delivered on schedule. |
| Requirement Management | Defines and manages project requirements to align with the client’s needs. | Gathers and refines game requirements based on feedback from the team and the client, ensuring features align with expectations. |

### Staffing

| Role | Staff Member | Start Date | End Date |
| --- | --- | --- | --- |
| Software Developer | Daniel Prawira | 13/01/2025 | 28/04/2025 |
| Sound Designer | Dave Geraldo Jonathan Koro | 13/01/2025 | 28/04/2025 |
| Tester | Joel Daniel Mark Wenas | 13/01/2025 | 28/04/2025 |
| Graphic Artist | Kayla Fakhira | 13/01/2025 | 28/04/2025 |
| Software Configuration Management | Daniel Prawira | 13/01/2025 | 28/04/2025 |
| Software Quality Assurance | Joel Daniel Mark Wenas | 13/01/2025 | 28/04/2025 |
| Plan Tracker | Kayla Fakhira | 13/01/2025 | 28/04/2025 |
| Project Manager | Shevchenko Saviour Matthew Poluan | 13/01/2025 | 28/04/2025 |
| Requirement Management | Syedha Audry Timothy Piri | 13/01/2025 | 28/04/2025 |

## Communication

Tim berkomunikasi menggunakan *WhatsApp* sebagai media utama untuk berdiskusi, berbagi update, dan mengkoordinasikan tugas. Komunikasi dilakukan secara asinkron, memungkinkan setiap anggota untuk memberikan update sesuai dengan ketersediaan mereka. Jika diperlukan pembahasan yang lebih mendalam, tim akan mengadakan pertemuan langsung saat kelas atau mengatur diskusi virtual melalui *Discord*.

## Risk and Asset Management

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Risks** | **Description** | **Anticipation** | **Mitigation** | **Impact\*** |
| *Keterlambatan Pengembangan Fitur* | Tidak menyelesaikan fitur sesuai jadwal karena kendala teknis atau keterbatasan waktu. | Menggunakan Gantt Chart untuk tracking progress setiap anggota. | Membuat gantt chart darurat yang berbasis jadwal harian, untuk mengejar keterlambatan projek, sekaligus dengan melakukan pertemuan harian secara online | \*\*\* |
| Kualitas Aset Kurang | Grafik, suara, atau UI dalam game tampak tidak seragam atau kurang sesuai dengan konsep awal. | Memastikan ada komunikasi yang jelas antara **Graphic Artist** dan **Software Developer** agar aset sesuai kebutuhan. | Mengambil free asset yang tersedia dari sumber yang sama agar tidak terjadinya desain yang tidak konsisten | \*\* |
| Bugs & Glitches | Game tidak berjalan sesuai harapan, pengalaman pemain terganggu. | Segala perubahan kode akan diberitahukan kepada **Lead Software engineer** | Melakukan bug fixing darurat secepta mungkin ketika tersedia | \*\*\*\* |

Impact:

\* biasa saja, \*\* waspada, \*\*\* bahaya, \*\*\*\* sangat bahaya, \*\*\*\*\* bencana/kiamat

**Risk Management Analysis**

|  |  |
| --- | --- |
| Cause and Effect Analysis | |
| Masalah atau Peluang | Penyebab dan Dampak |
| * Game ini dapat memberikan pengalaman yang menarik dan edukatif tentang kehidupan siswa. * Dapat digunakan sebagai alat edukasi. | * Penyebab: Menambah fitur tanpa perencanaan yang matang. * Penyebab: Implementasi AI yang buruk. * Penyebab: Kurangnya pengujian. * Dampak: Keterlambatan, mekanik yang tidak lengkap, atau rilis dengan banyak bug. * Dampak: Perilaku siswa yang tidak realistis, membuat game kurang menarik. * Dampak: Banyaknya bug saat rilis, yang berujung pada ulasan negatif. |
| System Improvement Objectives | |
| Tujuan Sistem | Batasan Sistem |
| * Membuat simulasi sekolah menengah yang seru dan imersif. * Menyeimbangkan belajar, kehidupan sosial, dan tantangan bagi pemain. * Mengoptimalkan performa untuk gameplay yang lancar. * Merilis game tepat waktu (16 minggu) tanpa mengorbankan kualitas. | * Kendala Waktu: Harus selesai dalam 16 minggu. * Kendala Sumber Daya: Tim terbatas (6 anggota) dan alat yang tersedia. * Kendala Anggaran: Pendanaan kemungkinan terbatas, memengaruhi pembuatan aset atau pemasaran. |

## Start up

Tim terlebih dahulu melakukan **diskusi awal** guna menentukan konsep dasar game, termasuk mekanisme gameplay, fitur utama, dan target pengguna. Setelah itu, dilakukan **analisis kebutuhan** untuk mengidentifikasi sumber daya yang dibutuhkan, seperti software development tools, aset grafis, dan sound design.

Selanjutnya, tim membagi peran dan tugas sesuai dengan keahlian masing-masing anggota agar pengembangan lebih terstruktur. Setelah perencanaan awal selesai, proyek dimulai dengan pembuatan **project planner**, yang mencakup timeline kerja, milestone, serta metode yang akan digunakan, seperti Agile untuk memastikan progres tetap terkontrol.

Sebagai langkah awal pengembangan, tim membuat **prototipe dasar** untuk menguji konsep gameplay dan memastikan mekanisme inti dapat berjalan sebelum masuk ke tahap produksi penuh.

## Customer Identification

*Company Name* : Eterna Palace Games

*Contact Person* 1 : Louis Khrisna Putra

*Position* : Founder

*Contact Person* 2 : -

*Position* : -

*Address* : Universitas Pelita Harapan, Karawaci, Jalan M.H. Thamrin Boulevard

No.1100, Kelapa Dua, Tangerang Regency, Banten 15811

*Phone* : -

*E-mail* : eternapalace@gmail.com

## Measures of Success

* **Fitur Inti Berfungsi dengan Baik**

Gameplay utama seperti belajar, ujian, dan interaksi dengan NPC berjalan sesuai rencana tanpa bug besar.

* **Stabilitas dan Performa yang Baik**

Game dapat dijalankan tanpa crash atau lag signifikan di perangkat target, dan tidak ada bug kritis yang menghambat pengalaman bermain.

* **Konsistensi Visual dan Audio**

Aset grafis, UI, dan animasi memiliki gaya yang seragam dan sesuai dengan konsep sekolah.

Sound effects dan musik latar mendukung atmosfer game tanpa terasa mengganggu

## Validation and Acceptance

Sebelum versi penuh dirilis, setiap tahap pengembangan (Prototype, Alpha, Beta) akan ditunjukkan kepada klien untuk validasi guna memastikan game sesuai dengan ekspektasi. Klien akan memberikan feedback terkait mekanik gameplay, UI, dan pengalaman bermain, yang kemudian digunakan tim untuk perbaikan sebelum melanjutkan ke tahap berikutnya. Pada tahap Beta, fokus utama adalah penyempurnaan dan perbaikan bug, hingga akhirnya versi final diuji kembali oleh klien untuk mendapatkan persetujuan sebelum rilis resmi. Dengan pendekatan ini, game dapat dikembangkan secara iteratif dan memastikan kualitas sesuai standar sebelum sampai ke pengguna akhir.

## Support

**User Manual & Tutorial In-Game**

* Menyediakan panduanberupa video demo yang akan diupload ke platform YouTube yang menjelaskan cara bermain, termasuk mekanisme belajar, ujian, dan aktivitas lainnya.

**Umpan Balik dari Klien**

* Menerima feedback dari open testing dari relawan untuk perbaikan atau peningkatan game sebelum finalisasi.

## Close Out

Projek akan selesai jika sudah mencapai waktu yang ditentukan dalam kontrak.

# Planning and Control

## Estimate

|  |  |  |  |
| --- | --- | --- | --- |
| Task | Tanggal Mulai | Tanggal Selesai | PIC |
| Requirement specification | |  | | --- | | 01/02/25 |  |  | | --- | |  | | 04/02/25 | Shevchenko Saviour Matthew Poluan |
| Persetujuan proposal draft | 05/02/25 | 07/02/25 | Shevchenko Saviour Matthew Poluan |
| Persetujuan atas proposal Final | 08/02/25 | 10/02/25 | Shevchenko Saviour Matthew Poluan |
| Penandatanganan kontrak kerja | 11/02/25 | 12/02/25 | Shevchenko Saviour Matthew Poluan |
| Proses koding dan desain | 13/02/25 | 20/03/25 | Daniel Prawira |
| Program testing | 21/03/25 | 26/03/25 | Joel Daniel Mark Wenas |
| Perbaikan koding dan desain | 27/03/25 | 03/04/25 | Daniel Prawira |
| Iterative process dari requirement, koding dan desain. | 04/04/25 | 08/04/25 | Daniel Prawira |
| Acceptance testing dengan customer | 11/04/25 | 15/04/25 | Joel Daniel Mark Wenas |
| Penyelesaian kontrak kerja | 16/04/25 | 18/04/25 | Shevchenko Saviour Matthew Poluan |

## Resource Identification

### Staffing

|  |  |  |  |
| --- | --- | --- | --- |
| Nama | Usia (tahun) | Ability | Sifat dan Karakter |
| Daniel Prawira | 18 | |  | | --- | | Memiliki kemampuan coding dan logika pemrograman yang kuat. |  |  | | --- | |  | | |  | | --- | | Teliti, fokus, dan mampu menyelesaikan masalah teknis secara efisien. Cenderung tenang dan berpikir analitis. |  |  | | --- | |  | |
| Dave geraldo Jonathan Koro | 18 | |  | | --- | | Mampu membuat komposisi musik dan efek suara untuk game. |  |  | | --- | |  | | |  | | --- | | Kreatif dan ekspresif, memiliki kepekaan terhadap nuansa suara. Fleksibel dan terbuka terhadap masukan. |  |  | | --- | |  | |
| Joel Daniel Mark Wenas | 18 | |  | | --- | | Memiliki wawasan luas tentang gameplay karena sering bermain game. |  |  | | --- | |  | | |  | | --- | | Antusias, cepat memahami konsep permainan, dan mampu memberikan masukan berharga dari sudut pandang pemain. |  |  | | --- | |  | |
| Kayla Fakhira | 18 | |  | | --- | | Mampu membuat desain visual dan user interface game. |  |  | | --- | |  | | |  | | --- | | Rapi, terorganisir, dan memiliki kepekaan terhadap estetika visual. Senang bekerja sama dalam tim. |  |  | | --- | |  | |
| Shevchenko Saviour Matthew Poluan | 18 | Memiliki pengalaman dalam desain pixel art, yang membuatnya mampu menciptakan aset grafis bergaya pixel untuk proyek game. Selain itu, ia memiliki kemampuan dalam menyusun jadwal dan membagi peran dalam tim, memastikan setiap anggota memiliki tugas yang jelas dan proyek berjalan sesuai rencana. Mengetahui coding dasar-menengah menggunakan C++ | Introvert yang lebih nyaman bekerja secara mandiri atau dalam lingkungan yang tidak terlalu ramai. Meskipun begitu, ia tetap mampu berkolaborasi dalam tim dengan komunikasi yang jelas dan terstruktur. Dalam menghadapi tantangan, ia memiliki sifat santai, tidak mudah panik, dan mampu berpikir dengan tenang untuk menemukan solusi terbaik. |
| Syedha Audry Timothy Piri | 18 | |  | | --- | | Mampu membuat dan mengatur elemen musik serta sound effect dalam game. |  |  | | --- | |  | | Orangnya suka berteman dan energetik, bisa diajak bekerja sama dengan baik |

### Time

Jadwal rutin setiap harinya dalam pengerjaan Projek.

|  |  |
| --- | --- |
| Hari | Kegiatan |
| Senin | - |
| Selasa | Development & Design |
| Rabu | Testing & Bug Fixing |
| Kamis | - |
| Jumat | Playtesting & Feedback Gathering |
| Sabtu | Documentation |
| Minggu | - |

### Materials

|  |  |  |  |
| --- | --- | --- | --- |
| Tangible | # | Intangible | # |
| Laptop  Smartphone | 6  6 | RenPy  Visual Studio Code  Canva  FL Studio  -  - |  |

### Work Breakdown Structure

**Software Process WBS**

A diagram of a company

AI-generated content may be incorrect.

Figure 2 Work Breakdown Structure

**Development effort per program**

|  |  |
| --- | --- |
| Phase | %Effort |
| Requirements  Design  Coding  Playtesting | 15  25  40  20 |

### Schedule

**Brief Project Schedule**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| ID | Aktivitas | Week | | | | | | | | | | | | | | | |
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 |
| A | Pre-Production |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| AA | Finalisasi pembagian peran dan tugas |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| AB | Membuat Key Features |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  | Membuat desain Core Mechanics |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| B | Core Development |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| BA | Implementasi mekanika game |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| BB | Mendesain game assets |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| BC | Pembuatan audio |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| BD | Membuat prototype |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| BF | Revisi dan Re-desain Prototype |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  | UI/UX |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| C | Testing |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| CA | Debugging |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| CD | Playtesting |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| D | Deployment |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| DA | Finalisasi |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| DB | Presentasi |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| E | Documentation |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| EA | Proposal |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| EB | Project Planning |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |

**Detail Project Schedule**

**\*terlampir dalam file terpisah**

### Budget

Perencanaan biaya dalam projek :

|  |  |
| --- | --- |
| Kegiatan | Perkiraan Biaya (Rp) |
| Transportasi  Komunikasi (kuota)  Dokumentasi (soft copy)  Biaya tidak terduga | 0  50.000  0  50.000 |

## Tracking and Control

|  |  |  |
| --- | --- | --- |
| **Type** | **Action** | **PIC** |
| Schedule | *Plan tracker* berusaha mengingatkan tentang schedule kepada anggota tim lain. | Syedha Audry Timothy Piri |
| Quality | * Mendokumentasikan perubahan atas kode yang terjadi. * Melakukan testing di tiap modul | Daniel Prawira |
| Functionality | Verifikasi dan validasi software. Berkaitan dengan testing dan mengetes segala macam yang berhubungan dengan testing. | Daniel Prawira |
| Documentation | Membuat dokumentasi seperti Project Proposal, SRS, GDD, Project Plan, kontrak. | Shevchenko Saviour Matthew Poluan |

**Project Tracking dilihat dari segi resourcesnya**

|  |  |  |  |
| --- | --- | --- | --- |
| Checkpoint | Scheduled  (Week) | Cumulative Resources | Resources % |
| |  | | --- | | Requirement Specification Complete |  |  | | --- | |  | | 2 |  | 40 |
| High-Level Design Complete | 13 |  | 10 |
| Low-Level Design and Coding Complete | 10 |  | 10 |
| Unit Test Plan Complete | 10 |  | 10 |
| Approval and Acceptance Complete | 14 |  | - |
| Code Complete | 14 |  | 10 |
| Unit Testing Complete | 15 |  | 10 |
| Final Product Complete | 16 |  | 10 |

# Technical Process

## Engineering

### Environment

Lingkungan kerja untuk pengembangan game ini menggunakan **Windows** sebagai sistem operasi utama. Bahasa pemrograman yang digunakan adalah **Python**, dengan **Ren'Py** sebagai game engine utama. Ren'Py dipilih karena kemampuannya dalam membuat game berbasis visual novel dengan mekanisme interaksi yang fleksibel dan mudah dikembangkan. Seluruh pengembangan dilakukan dalam ekosistem Windows, dengan alat bantu seperti editor teks atau IDE yang mendukung Python, serta manajemen aset grafis dan audio yang sesuai dengan kebutuhan game.

### Methods, Tools and Techniques

Tim menggunakan metodeAgile, di mana pengembangan dilakukan secara bertahap melalui sprint pendek. Setiap anggota tim memiliki tugas spesifik yang dievaluasi secara berkala, memungkinkan iterasi dan perbaikan cepat berdasarkan feedback.

* + **Ren'Py**: Engine utama yang digunakan untuk membangun game berbasis visual novel dengan sistem interaksi dan storytelling.
  + **Visual Studio Code**: Digunakan sebagai editor utama untuk menulis kode, terutama dalam scripting dan logika game.
  + **Canva**: Digunakan untuk desain UI/UX, elemen visual sederhana, dan pembuatan konsep grafis yang mendukung tampilan game.
  + **FL Studio**: Software yang digunakan untuk membuat atau mengedit sound effects dan background music dalam game.

## Technology

### Environment

Dari sisi perangkat keras, pengembangan dilakukan di komputer dengan sistem operasi Windows, spesifikasi minimal mencakup prosesor Intel/AMD, RAM 8GB atau lebih, serta penyimpanan SSD untuk efisiensi kerja. Untuk manajemen data, aset game seperti gambar, audio, dan skrip disimpan dalam struktur direktori proyek Ren'Py, dengan penggunaan sistem kontrol versi seperti Git untuk kolaborasi tim. Akses internet digunakan untuk mengunduh pustaka tambahan, mencari referensi pengembangan, serta berbagi versi game dengan tim melalui platform cloud atau repository GitHub. Secara keseluruhan, lingkungan teknologi ini memastikan pengembangan yang lancar dan terorganisir sesuai dengan kebutuhan proyek.

### Methods Tools and Techniques

#### User Interface

**A blue and black background

AI-generated content may be incorrect.**

Figure 3Main Menu

A screenshot of a computer

AI-generated content may be incorrect.

Figure 4 Gameplay 1

#### Hardware Interface

Perangkat keras minimum yang harus digunakan agar game ini dapat berjalan adalah:

* Keyboard
* Mouse
* Monitor

#### Software Interface

Bagi User:

* Sistem Operasi *Windows 10* atau lebih baru

Bagi Developer:

Sama seperti *user,* ditambah *software* aplikasi pembangunan seperti : Ren’Py

Setelah mengunduh dan menginstal file game yang disediakan, pengguna dapat langsung menjalankan game tanpa membutuhkan koneksi internet. Proses instalasi mudah, dengan file game dan semua dependensi yang diperlukan diinstal secara otomatis ke perangkat. Data progres permainan dan pengaturan pribadi disimpan secara lokal di perangkat pengguna.

## Code Growth Estimation

Estimasi dan rencana pertumbuhan banyaknya kode pada setiap fase yang diselesaikan.

|  |  |
| --- | --- |
| Completed Project Phase | Code Growth Range (%) |
| Feasibility  Requirements  High-level design  Detail design  Implementation  Function Test  System Test | 0 - 10  100 – 200  75 -150  50 - 100  25 - 50  10 - 25  0 - 10 |

## Project Artifacts

**Artifact Produk yang dibangun meliputi:**

* + Game Design Document
  + Gantt Chart Project Planner
  + Game Prototype
  + Game Demo Video

# Supporting Plan

## Administration

Kontrak ini merupakan bentuk kesepakatan antara tim pengembang dan klien untuk membangun game sesuai dengan tujuan dan tenggat waktu (due date) yang telah disepakati bersama. Kontrak ini tetap memiliki nilai penting karena menjadi dasar komitmen kerja antara kedua pihak. Selain itu, keberhasilan penyelesaian proyek ini juga berkaitan langsung dengan penilaian dalam mata kuliah yang sedang dijalani. Oleh karena itu, kontrak ini berfungsi sebagai pedoman dan pengingat tanggung jawab masing-masing pihak selama proses pengembangan berlangsung.

## Configuration Management

* + Setiap perubahan konfigurasi, baik dalam kode, aset grafis, maupun audio, **harus dikomunikasikan terlebih dahulu dalam tim** sebelum diterapkan.
  + Perubahan **minor** seperti perbaikan bug kecil atau penyesuaian desain akan langsung dicatat dalam project tracker dan diterapkan setelah mendapatkan persetujuan tim.
  + Perubahan **major**, seperti **penggantian fitur inti, perubahan role anggota, atau pergantian tools utama**, akan dibahas dalam meeting sebelum diimplementasikan.
  + Keputusan yang dibuat harus berdasarkan kesepakatan tim, dengan mempertimbangkan dampak terhadap deadline dan kualitas proyek.
  + Semua anggota wajib mematuhi aturan yang telah disepakati bersama untuk memastikan proyek berjalan sesuai rencana.

## Quality Assurance

Untuk memastikan kualitas projectHighschool Student Simulator, tim menerapkan beberapa langkah seperti **uji coba internal, feedback dari klien,** serta **perbaikan berkelanjutan** berdasarkan hasil evaluasi. Setiap fitur diuji untuk memastikan tidak ada bug yang mengganggu gameplay, sementara aspek grafis, UI, dan audio diperiksa agar sesuai dengan standar yang ditetapkan. Selain itu, komunikasi rutin dengan klien dilakukan untuk memastikan game memenuhi ekspektasi dan kebutuhan yang telah disepakati. Dengan pendekatan ini, tim berupaya menghasilkan produk yang tidak hanya berfungsi dengan baik, tetapi juga memberikan pengalaman bermain yang menyenangkan bagi pengguna.

## Testing

Pengujian dilakukan dengan cara menyebarkan link prototype game agar dapat dicoba langsung oleh pengguna. Setelah bermain, pengguna diminta untuk mengisi form evaluasi yang berisi pertanyaan terkait pengalaman bermain, kemudahan penggunaan, serta saran perbaikan.

Testing dimulai setelah prototype pertama selesai dan dilakukan selama proses pengembangan berlangsung.

* **Blackbox Testing**  
  Fokus pada pengujian fungsi-fungsi utama dari game berdasarkan input dan output tanpa melihat struktur kode di dalamnya.
* **Acceptance Testing**  
  Pengujian dilakukan untuk memastikan bahwa fitur-fitur dalam game sudah sesuai dengan kebutuhan pengguna dan target yang telah disepakati.

Whitebox testing tidak dilakukan karena keterbatasan waktu dan sumber daya, serta kompleksitas pelaksanaannya yang tidak sesuai dengan skala proyek ini.

## Deployment

Implementasi dari proyek game ini dilakukan secara sederhana dan terbatas, mengingat tujuan utamanya adalah sebagai pemenuhan tugas mata kuliah. Game akan dijalankan dan didemonstrasikan langsung melalui perangkat milik tim saat sesi presentasi kepada dosen pengampu. Tidak ada proses deployment skala besar seperti publikasi di *marketplace* game pada umumnya.

## Procurement

Tim memiliki beberapa **kekuatan** yang mendukung kelancaran proyek, seperti **pembagian tugas yang jelas sesuai dengan keahlian masing-masing,** sehingga setiap anggota dapat fokus pada bagian yang dikuasai. Selain itu, meskipun masih berstatus mahasiswa tanpa gelar profesional, tim memiliki **kemauan untuk belajar dan berkembang,** serta berkomitmen menyelesaikan proyek dengan standar terbaik.

Namun, ada juga beberapa **kelemahan**, seperti **jadwal kuliah yang padat**, yang membuat pengaturan waktu kerja menjadi tantangan. Selain itu, **jarak tempat tinggal yang berjauhan** menghambat pertemuan langsung, sehingga tim harus mengandalkan komunikasi online.

## Training

Untuk membantu pengguna memahami cara bermain game, tim menyediakan video demo yang diunggah secara online. Video ini menampilkan penjelasan singkat mengenai tampilan game, cara bermain, fungsi tombol, serta alur dasar permainan.

Training tidak dilakukan secara langsung atau formal, karena game ini ditujukan untuk pengguna umum yang diharapkan dapat memahami mekanik permainan secara intuitif. Selain itu, dalam game juga disediakan elemen visual dan petunjuk dasar yang memudahkan pemain saat pertama kali bermain.

## Installation Support

Proses instalasi game ini sangat sederhana. Pengguna hanya perlu mengunduh file game yangdibutuhkan kemudian langsung menjalankannya di perangkat Windows tanpa perlu instalasi tambahan.

## Maintenance

Setelah game diserahkan kepada dosen selaku klien pada saat presentasi akhir dan sesuai dengan tenggat waktu yang telah ditentukan, tidak ada proses maintenance atau pengembangan lanjutan yang direncanakan. Proyek ini dibuat khusus untuk memenuhi tugas mata kuliah, tanggung jawab pemeliharaan sistem tidak berlanjut setelah penyerahan akhir.

## Staff Development

Tim kami terdiri dari 6 orang dengan latar belakang dan keahlian yang beragam. Ada anggota yang unggul dalam bidang pemrograman, desain grafis, musik, serta ada juga yang memiliki minat dan pengalaman dalam bermain game, yang membantu dalam merancang gameplay yang menyenangkan dan seimbang.

Pembagian tugas dilakukan berdasarkan kekuatan dan minat masing-masing anggota. Hal ini bertujuan agar setiap anggota dapat berkontribusi secara maksimal dan hasil kerja lebih optimal. Misalnya, anggota yang ahli dalam coding bertanggung jawab pada pengembangan sistem inti, yang mahir desain mengurus UI dan aset visual, yang berbakat musik mengurus sound dan efek audio, dan lainnya turut aktif dalam testing, dokumentasi, serta menjaga keseimbangan gameplay.

Proses pengembangan ini juga sekaligus menjadi sarana pelatihan kami. Sepanjang proyek, kami mengikuti mata kuliah *Software Engineering* yang memberikan bekal teori dan praktik langsung mengenai pengelolaan proyek, pembagian peran, penyusunan dokumentasi, serta komunikasi dalam tim dan dengan klien. Pengalaman ini sangat berharga sebagai latihan nyata sebelum terjun ke dunia kerja profesional.

## Product Acceptance

Product acceptance dilakukan pada tanggal **26 April 2025**, saat game diserahkan secara resmi kepada klien (dosen) sebagai bentuk penyelesaian proyek. Sebelum penyerahan, dilakukan **acceptance testing** melalui uji coba langsung dan evaluasi dari pengguna terbatas untuk memastikan fitur-fitur utama berjalan sesuai dengan yang direncanakan.

Dengan diterimanya produk oleh klien, maka kontrak kerja sama dianggap telah selesai dan seluruh tanggung jawab tim dalam pengembangan proyek ini telah dipenuhi.

## Documentation

Dokumentasi yang akan dibuat adalah berupa:

* Team Contract
* Contract with Client
* Proposal
* Project Planning
* Software Requirement Specification
* Gantt Chart
* Burndown Chart
* Weekly/Daily Meeting Report

# Lesson Learned

Melalui proses pembuatan project planning dan pengembangan game ini, tim kami memperoleh banyak pelajaran berharga, baik dari segi teknis maupun non-teknis. Kami belajar bahwa merancang dan mengembangkan sebuah game bukan hanya soal membuat fitur berjalan, tetapi juga membutuhkan perencanaan yang matang, pembagian tugas yang jelas, dan komunikasi tim yang efektif.

Kami juga menyadari pentingnya dokumentasi dan manajemen waktu, terutama karena proyek ini harus diselesaikan dalam waktu yang cukup terbatas. Tantangan yang muncul selama proses—seperti perbedaan jadwal, revisi desain, dan debugging—mendorong kami untuk belajar lebih fleksibel dan mencari solusi bersama.

Mata kuliah Software Engineering membantu kami memahami kerangka kerja proyek secara menyeluruh, mulai dari perencanaan, implementasi, hingga testing. Proyek ini bukan hanya latihan teknis, tetapi juga pengalaman nyata dalam bekerja sebagai tim dan menyelesaikan sebuah produk dari awal hingga akhir.