Project Plan

[Summary of the Project 2](#_Toc184399669)

[Project Description 2](#_Toc184399670)

[Objectives of the Project 2](#_Toc184399671)

[Summary of Methodology 3](#_Toc184399672)

[Software Development Methodology: Agile 3](#_Toc184399673)

[Project Team Organization 4](#_Toc184399674)

[Tools 4](#_Toc184399675)

[Project Schedule 5](#_Toc184399676)

# Summary of the Project

## Project Description

Tank War is an engaging combat game that immerses players in strategic tank battles. Set in a visually stunning environment, players will navigate their tanks, engage in fierce battles, and utilize tactical maneuvers to outsmart opponents. The game combines action-packed gameplay with strategic elements, appealing to both casual and competitive gamers.

## Objectives of the Project

The primary goal is to create a polished, enjoyable, and engaging game that stands out in the tank combat genre. We aim to deliver high-quality gaming experience by incorporating user feedback throughout the development process and ensuring robust gameplay mechanics. To achieve this, we plan to implement following key features:

* Tank movement and controls​: Easy-to-learn controls ensure that players of all skill levels can jump right into the action.
* Combat mechanics (shooting, damage, etc.)​: Players can have experience to diverse combat strategies.
* AI behavior for enemy tanks​: Fight against intelligent enemy tanks, providing a challenging experience for solo players.
* Varied Environments(walls, health packs, etc.)​: Engage in varying environments that is randomly generated, promising a non-repeatable experience for player every game.

# Summary of Methodology

## Software Development Methodology: Agile

图示

描述已自动生成

The Tank War game development will be managed using Agile methodologies, specifically employing Scrum to facilitate iterative progress, enhance collaboration, and incorporate user feedback. The project aims to deliver a high-quality multiplayer tank combat game by November 30, 2024.

The development loop:

1. Find and state the Requirements​
2. Design the feasible architecture​
3. Write code​
4. Run Tests for the code​
5. Refactoring the existing code​

We used flat management. Each person runs the cycle independently and paralleling.

For project scheduling, sprints are used to review and plan the stage’s objectives.

Sprints:

* Duration: Each sprint will last for 2 weeks.
* Sprint Planning: At the beginning of each sprint, the team selects user stories from the product backlog to work on.
* Daily Stand-ups: Brief meetings to discuss progress, plans for the day, and any challenges faced.

Sprint Review and Retrospective:

Sprint Review: At the end of each sprint, the team demonstrates completed work to stakeholders and gathers feedback.

Retrospective: A meeting to reflect on what went well, what didn’t, and how the team can improve in the next sprint.

Advantages of using Agile development are various. Agile allows for quick adaptation to changing requirements and feedback, ensuring the game evolves based on needs. Regularly delivering functional increments enables early testing and feedback, providing stakeholders with a clear view of progress and allowing for adjustments before the final release. Agile fosters a collaborative environment where team members communicate openly, share ideas, and resolve issues collectively, leading to better teamwork and innovation. Continuous testing and integration throughout the development process help identify and fix issues early, resulting in a more polished final product.

## Project Team Organization

|  |  |
| --- | --- |
| **Member** | **Role** |
| **Hong Yifan** | Project Manager |
| **Zhao Kevin** | Software Engineer |
| **ZHONG Chenghao** | Software Engineer |
| **XU Haotian** | Software Engineer |
| **LIU Fengda** | Software Engineer |
| **ZHU Fu** | Software Engineer |

## Tools

1. Development Tools
   1. Eclipse Mars (4.5.0)
   2. External Libraries
      1. org.apache.maven.plugins
      2. org.mockito
2. Development Platform
   1. Windows 10
   2. Java 17
   3. Maven 3.3
3. Test Cases
   1. Junit 5
4. Coverage Checking
   1. EclEmma 2.0
5. Testing Platform
   1. Windows 10 + Internet Explorer / Chrome

# Project Schedule

|  |  |  |  |
| --- | --- | --- | --- |
| **Date Range** | **Sprint** | **Key Activities** | **Milestones** |
| **Oct 1 - Oct 14** | Sprint 1 | - Define Product Vision | - Product Vision Document Complete |
|  |  | - Create initial Product Backlog (user stories) | - Product Backlog Created |
|  |  | - Design game concept and mechanics |  |
| **Oct 15 - Oct 28** | Sprint 2 | - Develop core gameplay mechanics | - Basic Gameplay Prototype Ready |
|  |  | - Implement tank movement and controls |  |
|  |  | - Conduct initial testing on mechanics |  |
| **Oct 29 - Nov 11** | Sprint 3 | - Implement combat mechanics (shooting, damage) | - Combat Mechanics Functional |
|  |  | - Develop AI behavior for enemy tanks |  |
|  |  | - Begin user interface (UI) design |  |
| **Nov 12 - Nov 25** | Sprint 4 | - Finalize UI design | - UI Design Complete |
|  |  | - Conduct extensive testing and bug fixing | - Alpha Version Ready for Review |
| **Nov 26 - Nov 30** | Sprint 5 | - Make Improvement to current version |  |
|  |  | - Make final adjustments | - Final Version Ready for Launch |