

CS 447/547: Game Design Project 2

Design Document

Iron Legends

Travis Hall

travis.hall@email.wsu.edu

Bhadresh Patel

bhadresh.patel@email.wsu.edu

Michael Persons

michael.persons@email.wsu.edu

Abstract

Iron Legends is a top-down arcade-style tank combat game. The goal is simple: total domination! Players will strap themselves into the cockpit of a tank and blast, smash, and detonate anyone and everyone who dares to stand in their way. With two different modes of gameplay—single-player and networked multiplayer—combatants can pit their wits and their bravery against legions of computer-controlled tanks or test their mettle against friend and foe alike. Let the mayhem begin!

1 Game Overview

Iron Legends is an unrestrictive two-dimensional shooter where players are placed in control of a tank and let free to cause as much havoc as possible. That said, depending on the mode of gameplay—single-player or networked multiplayer—there are certain objectives that dictate the win scenario.

In the single-player game mode, various types of computer-controlled enemies will roam the battlefield trying to destroy the player. In order for the player to win, s/he must annihilate each of the enemies. To even the odds, each defeated enemy will have a chance to drop a power-up for the player's tank. For example, power-ups may include: increased speed, increased armor, or even weapons with different behaviors.

In multiplayer mode, the objective is different. Players are placed on teams and given an unlimited number of lives. Instead of being the last tank standing, the teams are required to destroy an enemy base while protecting their own. There is one catch; when destroyed, the player will suffer some negative effect, either by losing points from their score or by losing their accumulated power-ups.

In order to provide a fun game environment where players are not swallowed up in an incomprehensible chaos, while keeping the game hectic enough to be exciting, the game maps must be relatively large. In order to accomplish this, the game uses a scrolling screen to only view certain areas of the world at any given time, and care must be taken to prevent the combatants from getting lost.

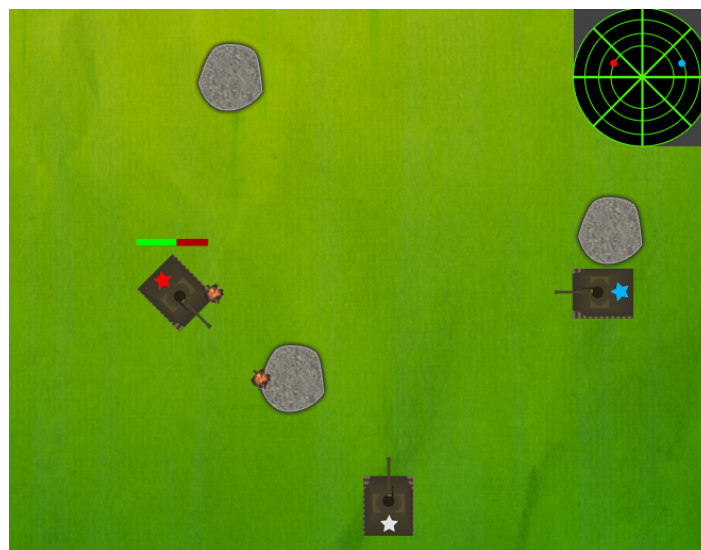


Figure 1: Initial Mockup—*Iron Legends*

1.1 Game Play

The main features of game-play are summarized below:

1.1.1 Common Features

- Scrolling screen that only reveals part of the battlefield at any given time.
- Items on the map that release power-ups when destroyed.
- Both single- and multiplayer modes, with distinct gameplay elements.

1.1.2 Single Player Features

- Objective: The player must annihilate all enemy tanks on the battlefield.
- Each time the player is shot, their total health is decremented.
- The player has three lives and each time the player's health is depleted, one life is lost.
- Enemy tanks are controlled by the computer artificial intelligence (AI).
- When enemies are destroyed they have a chance of dropping power-ups that the player may then collect.
- Each power-up changes the player's tank in some manner. *e.g.* speed boosts, armor boosts, repairs, new weapons, etc.

1.1.3 Multiplayer Features

- Objective: Destroy the enemy team's base while preventing them from destroying yours.
- A team loses if their base is destroyed.
- A team is declared the winner if their base is the last one remaining.

1.2 Possible Interactions

1.2.1 Player Interaction

- Select game mode (single or multiplayer)
 - If the game is multiplayer: create a lobby or join an existing game.
- Controlling the tank: movement and shooting.
- Weapon selection.

1.2.2 Entity Interaction

- Tank and obstacle collision.
- Tank and power-up collision.
- Weapon and Tank collision.
- Weapon and obstacle collision.
- Weapon and Base collision.

1.3 Entities

Player Base The main hub for a team, surrounded by simple obstacles and vulnerable to enemy attack.

Wall The most common obstacle. It can be destroyed.

Trees Object that stops tanks but not weapon-fire.

Rocks Cannot be destroyed and blocks all objects.

Power-ups Objects that have differing effects on player tanks.

Tanks The main entities for all players and enemies (see Tank section for more details).

1.3.1 Tanks

Player Tank The player begins with a Basic Tank and can collect power-ups through the duration of the game.

Basic Tank The most simple form of the tank. Moderately fast, moderately armored, and moderately armed.

Fast Tank A faster version of the Basic Tank. Because of its speed, it often poses a considerable threat to the bases and can be used for hit-and-run operations.

Armor Tank A slower, more heavily-armored version of the Basic Tank. It has greater health than the basic tank and can be used effectively as both a guard and attacker.

Jeep An extremely fast vehicle. As a trade-off for the speed, the Jeep loses its direct fire weaponry instead laying mines that explode in proximity to enemy tanks. Though the Jeep is the least directly-threatening "tank," skilled players will be able to use the Jeep to lay traps and use guerilla tactics to defeat their opponents.



Figure 2: High Res Mockup—*Basic Tank*

1.3.2 Power-ups

As enemy tanks are destroyed, they have a chance of dropping power-ups. The power-up will disappear after a time if not collected, but upon collection, the player's tank will gain certain benefits. The following list depicts some of the possible power-ups that will be included in the game:

Upgrade This upgrades the player's tank and its effects remain until the player dies. The upgrades can be in the form of additional weapons or different types of tanks (see the previous Tank section).

Repair Increases the player's health by a certain percentage.

Shield Provides a temporary shield to the player's tank, making it immune to all enemy attacks for the duration of its effects.

Extra Life Provides an extra life to the player.

Double Cannon Provides the ability to fire two shots at the same time with faster speed.

1.3.3 Minimap or Radar

A small portion of the screen will be devoted to some means of providing the player with a means of navigation. Because the screen contains only a certain area of the map, without this means of orienting and locating relative positions, players could easily get lost. This may either be implemented in the form of a radar or a minimap.

1.4 Why this is fun:

Players are placed into a competitive environment where they are forced to make rapid decisions that can result in either glory or despair. By including power-ups, the player is given a greater incentive to avoid damage, and there is a greater degree of intensity to the gameplay as a result.

To prevent players from getting bored, additional maps are provided and their varying layouts can result in vastly different gameplay experiences. In addition, the multiplayer mode allows friends to play together, bringing a certain social element to the game. Of course, being able to lord your victory over the others doesn't hurt.

2 Development Strategy

2.1 Code Reuse

Several snippets of existing code and concepts will be integrated into this project. The goal is to reuse as much of the code as possible without creating any integration issues. The following list depicts some of the possible areas for code reuse:

HedgeRunner Collision detection using the Separating Axis Theorem (SAT), screen-to-screen navigation, and text-input UI elements.

Tank Battle A rough base from which the gameplay has evolved and a good baseline from which the entities can be extended.

Timbre The concept of a scrolling screen game as well as composite graphics and the ability to generate graphics computationally.

2.2 Unknowns

This project will require several features for which the implementation is unknown to the authors at the time of this proposal. These unknowns can prove to be challenging and will thus be areas that will need to be identified early into the project's timeline. The list below describes some of those areas:

Networking The proper implementation to provide seamless multiplayer gameplay. This includes network optimizations.

Artificial Intelligence The algorithms for enemy Tanks that will provide fun gameplay. For example, a full and correct implementation of the traditional A* algorithm will create enemy tanks that are too strong. Certain refinements will need to be made such that players are not overwhelmed.

Command Separation Separating control-input from command execution (local, remote, AI, etc.)

Performance Networking and AI performance must be fast enough so that there is no adverse effects on the player's experience.

Architecture Separating the architecture so that the game can support single player and multi-player seamlessly and without requiring duplicated or separate code-bases.

Controls Effectively use keyboard and/or mouse controls for easy tank navigation.

2.3 Roles

Each of the team members will be involved in the various design decisions as well as major implementation areas. Though each member will have a main focus, it is not expected that they will devote themselves exclusively to those areas. The lists below show some of the group decisions as well as individual focuses:

- Networking utility/wrapper for client/server.
- The core of the AI.
- Render/Update logic vs. Control Logic.
- Configuration and navigation screens.
- General gameplay and UI layout.
- Integrating map and game information with the AI.
- Fleshing out various interactions and gameplay elements as they crop up.

Travis Hall Graphics/sprites, networking client/server wrapper for multiplayer mode

Bhadresh Patel AI Tanks, rendering/update logic for local and remote entities, player controls

Michael Persons Screen configuration/navigation, game setup/customization, collision detection and handling, networking message communication

2.4 Milestones

2.4.1 M1

- Scrolling screen gameplay
- Screen navigation (Help, some customization screens, etc)
- Simple map loading
- Controlling a single tank

2.4.2 M2

- Simple networked framework working
- Configuration screens for network setup
- Basic animations

2.4.3 Alpha

- Single player gameplay (computer opponents with limited AI)
- Separation between issuing commands and executing commands (to enable various types of input for AI, remote player, local player, etc.)
- Basic status information
- Tentative screen layouts

2.4.4 Release

- Networked and single player gameplay available
- Sound effects
- Finalized screen layouts, graphics, etc.

3 Technical Showpieces

3.1 Artificial Intelligence

Certain AI techniques will be harnessed for dictating the computer-controlled behavior. This includes searching (such as a modified version of the A* algorithm), and using those results to create different styles of opponents who attack using different strategies.

3.2 Multiplayer via Networking

Multiplayer mode is a challenge in that it not only requires that the game have a method of networked communication, but a method must be established that ensures a consistent experience across all players. With a poor implementation, players could potentially see enemies jumping around the screen due to the network latency. This is not desirable and so care has to be taken to avoid such situations.

Due to the myriad patterns available for doing such networking, there is always the risk that a chosen technique is inviable for unforeseen aspects.

4 High Bar

- Flying opponents like helicopter or bomber aircraft.
- Ability to collect and use rockets (for shooting helicopters).
- Advanced weapons: mines, laser or guided missile.
- Players can customize their tanks (perhaps color, varying tank abilities *e.g.* slow but strong, quick but weak, various types of ammunition, etc.).
- Movable turret (*i.e.* players could move the turret without moving the tank).

- Collectible ammunition (*i.e.* upon destroying enemies they will leave behind ammunition that a player can collect to score more points and increase the power).