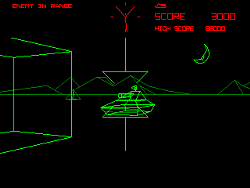
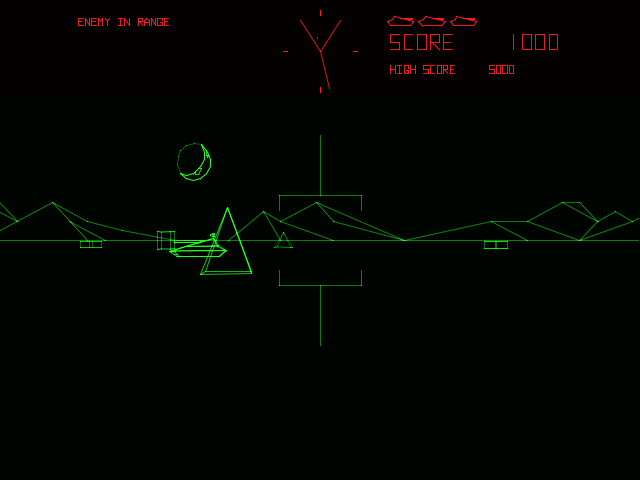
+

Do It Tomorrow

Summary

For this project we are going to be creating a Tank game. The game will be 3D based and will be programmed in XNA. It will be based on the 1980 arcade game “[BattleZone](http://en.wikipedia.org/wiki/Battlezone_(1980_video_game))”. The game will make use of 3D vector graphics. We will be using blender for 3D renders as it is open source.

Below are screenshots of the original BattleZone game:

The team has experience with the XNA framework which is a framework used to make games. This makes it the ideal tool for us. We will be working to get a playable prototype up and running in the coming weeks and we plan on bringing the game to this year’s Games Fleadh. The theme to Games Fleadh this year is Robot Tank by Activision. That game was basically a clone of the original 1980 BattleZone game.

We have decided to make our game as similar as possible to the original BattleZone game. We want to make us of wireframe graphics and use a similar control system. As well as this we want to add a few features that weren’t in the 1980 version. We would ideally like to get radar set up so that the player can easily locate the enemy once they go out of sight.

Version History

The design document has been a work in progress throughout the project. Because of this, there have been several versions of it. This version is our final design that we are submitting.

# Version 0.1

This is the first version of the design document. It includes all of our UML diagrams. This version was completed on 21st February 2013.

Game Overview

The game that we are creating is going to be a clone of the 1980 game produced by Atari called “BattleZone”. We will be creating our game using newer technologies than those that were available when the game was originally created so we hope to be able to add more modern features such as radar to the game.

BattleZone was significant in the gaming industry as it effectively was the first first-person shooter game and created a genre that would go on to dominate gaming in the years to come. It used 3D graphics and vector animation to create a more immersive gaming experience.

It’s little wonder so that when the game was released in November 1980 it became a huge hit. Gamers were treated to a 3D gaming experience like they had never seen before. The game represented the birth of 3D gaming.



-The game was originally an arcade game. It has since been ported to many different platforms.

Assumptions & Dependencies

# Operating System

We will be doing all of the programming for the game on Windows using Visual Studio. The game will be Windows based.

# End-User Characteristics

* The user will be able to control the game using the keyboard.
* They will be able to move the tanks camera to aim in different directions using the crosshair.
* The game will also be controllable with a gamepad.
* It will not be difficult to learn the controls of the game, but it will be difficult to master them.

Development Methods

When we had a meeting to decide on what method we would take to develop our game we soon came to the realisation that we could not use any of the usual development methodologies that we had used for different modules in our course so far.

High Level System Architecture

Our game will be developed while focusing on different areas. Each of these requires their own focus.

# Movement

The movement of the player, enemies and bullets will all be handled by an update function. The player’s movement will be defined by what the player presses on the keyboard or the gamepad depending on which they are using. The enemy will have its own AI logic.

# Collision Detection

We will be using bounding sphere collision detection in the game. The basic idea is that for each model you are using, or for each mesh in a model, you will construct a sphere from it.

# Testing

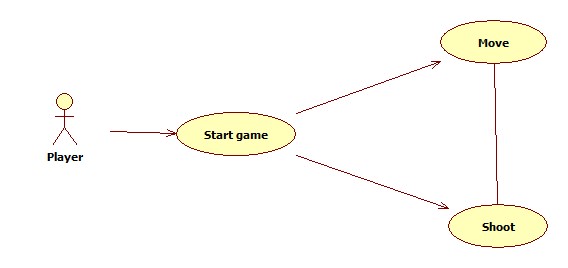
Between each of these steps we will be testing for possible bugs in the code and other problems.

# Game World

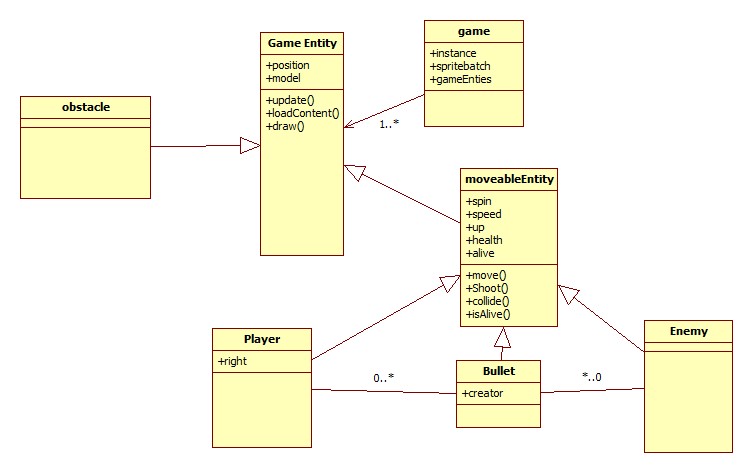
As with most elements of the game, we wanted to create a game world that stayed true to the original 1980 game ‘BattleZone’.

# Scoring

Use Case Diagram

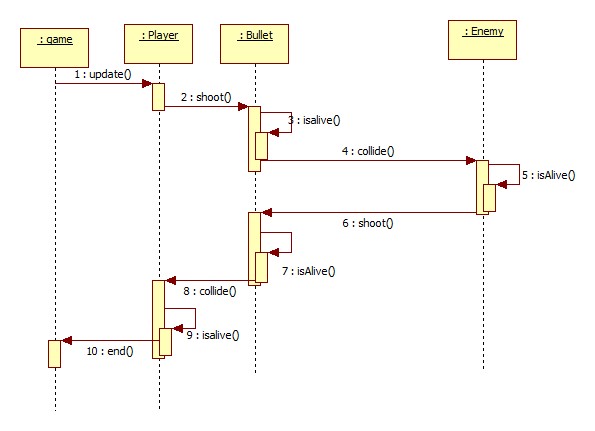


Class Diagram



The structure of the game will involve three main objects. The player, the enemy and the bullets. Each of these includes their own model and textures. Bullets will be created and destroyed when they are used by the player and the enemy.

Sequence Diagram



The sequence diagram above maps out what the player does when they are in the game. The player can shoot a bullet. The bullet travels until it either hits an enemy or hits some other obstacle. At this point the bullet exits the isAlive() state. If the bullet hits an enemy their health is checked and depending on whether their health hits 0 or not, they either continue on or get removed from the game.

GUI Prototype



The above picture is a mock-up of when we would like our “start game” screen to look like.