**Task 7: Design a Classic Game – Space Invaders**

Description of the Game

Gameplay in Space Invaders is relatively simple. The player controls a small ship that can only move laterally across the bottom of the screen and fires vertically. 5 rows of 11 aliens each advance slowly from one side of the screen to the other, dropping down one space and reversing direction when they reach either side. The player’s task is to acquire points by eliminating enemies and to destroy all the aliens before they reach the bottom of the screen and complete their “invasion.” As aliens are destroyed, the speed of the remaining enemies increase, as does the tempo of the music. Once all the enemies are destroyed, the wave resets and the difficulty increases.

The Invaders constantly shoot back at the player as they advance from side to side across the screen. To help avoid their attacks, the player can hide behind a few destructible barriers or “bunkers” near the bottom of the screen (four in the original version). Occasionally a “mystery ship” will appear near the top of the screen and move quickly from one side to the other while making a distinctive klaxon noise. Destroying it rewards the player with a sizeable point bonus.

Graphical user interface

Description automatically generated with low confidence

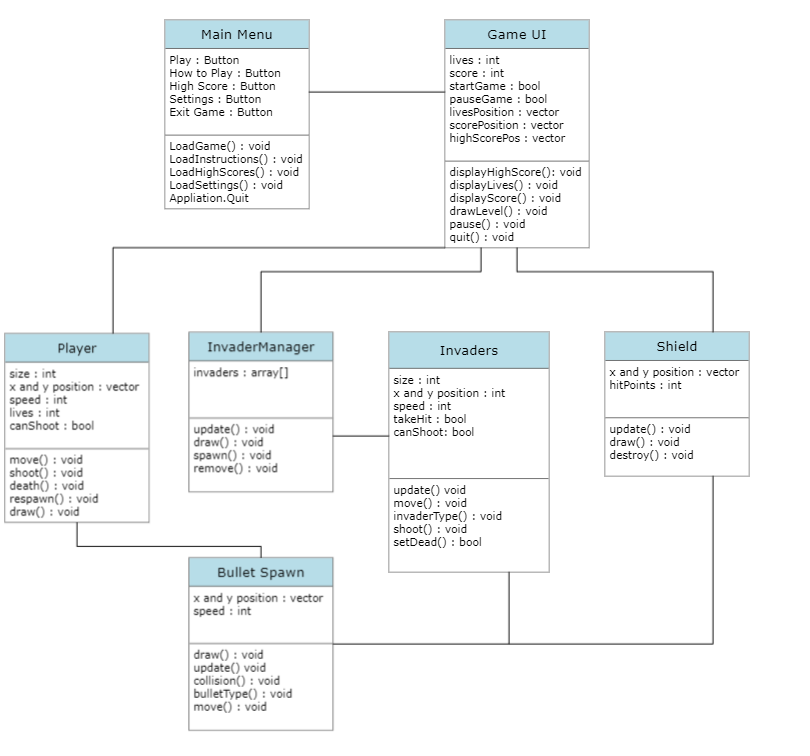
Programming Patterns

* Checking for collisions with bullets and player
* Checking for collisions with bullets and enemy
* Checking for collisions with bullets and shield
* Remove enemy once they are hit by player bullet
* Shift all enemies from side to side
* Move enemies forward towards player
* Player movement
* Fire bullet
* Add value to current score

Data Structures

Algorithms

Class Diagram



Testing Strategies

Functionality Testing – is done to confirm whether the end product works to the specifications. This form of testing mainly has the testers looking for generic problems within the game or its graphics and user interface e.g. game assets, stability issues, AV issues and game mechanics issues.

Performance Testing - is used to determine the application’s overall performance under real-time scenarios and load. Conducting this type of testing helps to ensure whether the present infrastructure allowing the smooth functioning of the game.

Ad-Hoc Testing - is a less structured way of testing and it is randomly done on any section of the gaming application. Specifically, there are two distinct types of ad hoc testing. This kind of testing works on the technique called “error guessing” and requires no documentation or process or planning to be followed. Since Ad hoc testing aims at detecting defects or errors through a random approach, with zero documentation, errors won’t be mapped to test cases.

Report and Repair – Throughout these testing procedures all bugs and errors need to be documented in a report. The development team can then sit down and come up with solutions to fix these bugs/errors. Multiple solutions might need to be implemented and tested to see which solution performs better.

UI and Menus

Menu Options – Play, How to Play, High Scores, Settings, Exit Game

Play – Starts the game

How to Play – Displays rules of game, how to move and shoot, the score value for the various types of aliens.

Settings – Adjust controls and basic graphical and audio settings

Exit Game – quits out of game

UI – Display player lives, display current score, display high score and spawn aliens, shields and player

Pause menu – Resume, settings or exit options

<https://cloud.smartdraw.com/editor.aspx?credID=-39210055&depoId=35485906&flags=128#depoId=35485906&credID=-39210055>

<https://github.com/LegatAbyssWalker/SpaceInvaders>

https://camo.githubusercontent.com/f6e8e17360e68e33efb61bd4b508c1f8185925bf7f5bd11bbb2bf31368d5c472/687474703a2f2f692e696d6775722e636f6d2f3834687863497a2e706e67