Game Design Document

**Space-Shooters**

The space-shooters game is built on C++ programming language and SFML (Simple and Fast Multimedia Library) API. SFML is a cross-platform software development library designed to provide a simple application programming interface to various multimedia components in computers.

Author: Arth Patel

Date: 27th January

Copyright Information:

Version 0.1(draft) October 10, 2005

By Mark Baldwin

Baldwin Consulting

http://baldwinconsulting.org

The Game Design Document (GDD) it the blueprint from which a computer or video game is to be built. As such, every single detail necessary to build the game must be addressed in the document (or support documents). If it’s not in the document, then it probably won’t be in the game.

Below you will find an outline for a generic Game Design Document. The problem is that no generic GDD will be able to address all the various genres for which a game may be created. For example, consider the games PacMan, SimCity and Doom. All three games required detailed design documents, but if you think about it, those documents would be entirely different! As such, when using the outline below you will find sections that will be totally meaningless to your game. But also, there will be sections that your GDD requires to describe the game. Just because it’s not in my outline, it doesn’t mean that it doesn’t belong.

The GDD is a reference document. Members of the development team will constantly be using the document to find specific information for their specific needs. Consider the size such a document may grow to in order to document every piece of the game. We don’t want the GDD to cause information overload and then become a prop under somebody’s wobbly desk. As such it is important that you organize and format the document to make it easy to use. Also note that some of these sections might not appear in the GDD itself but instead would appear in supplemental documents such as an Art Bible or Test Plan. This helps make the overall document more manageable and readable.

One last comment, a game design document is meant to be a living document. Just as when the artist changes the design of his painting every time he takes his brush to the canvas, a computer or video game evolves as code and art are created. The GDD then is the communication tool from which all the members of the team can follow that evolution.

**Document Revision history:**

|  |  |  |  |
| --- | --- | --- | --- |
| Version | Date | Author(s) | Changes |
| 0.2 | January 16, 2013 | Emile Cormier |  |
| 0.3 | January 05, 2016 | David Burchill |  |

*(Use word’s “update table of contents” to generate / regenerate a table of contents. )*

Table of Contents

[1. Section I - Game Overview 6](#_Toc96250236)

[1.1. Game Concept 6](#_Toc96250237)

[1.2. Design History 6](#_Toc96250238)

[1.3. Feature Set 6](#_Toc96250239)

[1.4. Genre 6](#_Toc96250240)

[1.5. Target Audience 6](#_Toc96250241)

[1.6. Game Flow Summary 6](#_Toc96250242)

[1.7. Look and Feel 7](#_Toc96250243)

[1.8. Project Scope 7](#_Toc96250244)

[1.8.1. Number of locations 7](#_Toc96250245)

[1.8.2. Number of levels 7](#_Toc96250246)

[1.8.3. Number of NPC’s 7](#_Toc96250247)

[1.8.4. Number of weapons 7](#_Toc96250248)

[1.8.5. Etc. 7](#_Toc96250249)

[2. Section II - Gameplay and Mechanics 8](#_Toc96250250)

[2.1. Gameplay 8](#_Toc96250251)

[2.1.1. Game Progression 8](#_Toc96250252)

[2.1.2. Mission/challenge Structure 8](#_Toc96250253)

[2.1.3. Puzzle Structure 8](#_Toc96250254)

[2.1.4. Objectives – What are the objectives of the game? 8](#_Toc96250255)

[2.1.5. Play Flow – How does the game flow for the game player 8](#_Toc96250256)

[2.2. Mechanics 8](#_Toc96250257)

[2.2.1. Physics 8](#_Toc96250258)

[2.2.2. Movement 9](#_Toc96250259)

[2.2.3. Objects 9](#_Toc96250260)

[2.2.4. Actions 10](#_Toc96250261)

[2.2.5. Combat 10](#_Toc96250262)

[2.3. Screen Flow 10](#_Toc96250263)

[2.3.1. Screen Flow Chart 10](#_Toc96250264)

[2.3.2. Screen Descriptions 11](#_Toc96250265)

[3. Section III – Story, Setting and Character 11](#_Toc96250266)

[3.1. Story and Narrative 11](#_Toc96250267)

[3.1.1. Back story 11](#_Toc96250268)

[3.2. Game World 11](#_Toc96250269)

[3.2.1. General look and feel of world 11](#_Toc96250270)

[3.3. Characters 13](#_Toc96250271)

[3.3.1. Character #1 Player Aircraft 13](#_Toc96250272)

[3.3.2. Character #2 Enemy Aircraft - Small Ships 14](#_Toc96250273)

[3.3.3. Character #3 Enemy Aircraft – Alien’s Ships 15](#_Toc96250274)

[3.3.4. Character #4 Enemy Aircraft – Final Boss 15](#_Toc96250275)

[4. Section IV – Levels 17](#_Toc96250276)

[4.1. Level #1 17](#_Toc96250277)

[4.1.1. Objectives 17](#_Toc96250278)

[5. Section V - Interface 17](#_Toc96250279)

[5.1. Visual System 17](#_Toc96250280)

[5.1.1. HUD - What controls 17](#_Toc96250281)

[5.1.2. Menus 18](#_Toc96250282)

[5.1.3. Camera 18](#_Toc96250283)

[5.2. Control System 18](#_Toc96250284)

[5.3. Audio 18](#_Toc96250285)

[5.4. Sound Effects 18](#_Toc96250286)

[6. Section VI – Game Engine 19](#_Toc96250287)

[6.1. General Description 19](#_Toc96250288)

[6.2. Game Logic 19](#_Toc96250289)

[6.3. Support AI 19](#_Toc96250290)

[6.3.1. Player and Collision Detection 19](#_Toc96250291)

[6.3.2. Pathfinding 20](#_Toc96250292)

[7. Section VIII – Game Art 20](#_Toc96250293)

[7.1. Concept Art 20](#_Toc96250294)

[7.2. Characters 20](#_Toc96250295)

[7.3. Environments 20](#_Toc96250296)

[7.4. Equipment 20](#_Toc96250297)

# Section I - Game Overview

## Game Concept

* The big picture of the game is to defend the spaceship from the different enemies and defend the final boss to win the game. The spaceship introduced with different types of pickups during the mission. The pickups will be not generated randomly, spaceship must kill some enemies to generate some pickups. The pickups generally boost speed, upgrades the bullet type, and increases health of the player spaceship. The Player must keep their health safe (stay alive) to survive till the end.
* The main quest of the game is to defend enemies and kill the final boss to win the game and exceed to the next level. The failure of doing that will bring you to the first level of the game. If a player ended having 0% of health, or killed by enemy, the result will be the same.
* The main players of the space shooter will be Spaceship (player), little space shuttles (enemy), final boss.

## Design History

This is a living document and will change as you develop your game. This section maintains a history of any significant changes to your game.

## Feature Set

## Genre

## Target Audience

Anyone can play this game if they are above 10+ years. This game is user friendly and easy to play. Space-shooters can be played on worlds most used operating system – windows. The game will surely fulfil user’s expectations.

## Game Flow Summary

The space shooter is a 2D shooting game with little animations and graphics. The main character will be a spaceship which is controllable by player. The map looks similar as a dark space and player will experience like real space war. There will be enemies, final boss, pickups, health, score, menus, unique bullets, missiles. The game has different levels, and the difficulty of the game is based on the level up of game. The player spaceship has different animations that will make game more fun and interactive. The player spaceship has different levels of bullets, the bullets will be upgraded when level up or by pickups.

The game has different menu state. The player can pause/resume the game whenever wants or also quit the game. The game has music theme which make game more interesting and fun to play. Moreover, the bullets also have some animation that made game look cool. As mentioned above, the bullets will be upgraded as levels upgrades and it will bring more fun to the gameplay. For the main character, player can adjust 3 or 4 types of themes of the spaceship animation. There will be health for the player, enemies and final boss. Player has to be beat enemies to meet final boss and have to defend obstacles for killing the final boss. The player will win after killing final boss and will upgraded to next round.

The space-shooter game behavior is side-scroller. The screen will follow player spaceship as it moves right side. So, the action is from left to right.

## Look and Feel

The map looks similar as a dark space and player will experience like real space war. This game is user friendly and easy to play. Space-shooters can be played on worlds most used operating system – windows. The game will surely fulfil user’s expectations.

## Project Scope

A summary of the scope of the game.

### Number of locations

2 to 3 locations will be there.

### Number of levels

10 levels will be designed.

### Number of NPC’s

5 to 6 NPC’s are there.

### Number of weapons

2 to 3 weapons will be there.

### Etc.

The space-shooters game is likely to be resulted as more fun in general. Although, there is little graphics and animations in the game, it is effectively differentiating from other recent space-based game.

# Section II - Gameplay and Mechanics

## Gameplay

### Game Progression

The game is broken into different levels. The player spaceship must kill enemies and boss to proceed next level of the game. If failed, it will bring you to first level of game.

### Mission/challenge Structure

The game has different challenge, the player must avoid the enemies which are programmed to follow player aircraft. Failing to that will resulted death of the player. The player also must defeat the main boss to reach next level.

### Puzzle Structure

### Objectives – What are the objectives of the game?

• The small ships are programmed to follow the main character spaceship and damage it. It will happen if the main character spaceship enters these small ship’s allocated boundary to follow main character spaceship.

• The aliens ship just can shoot bullets towards the main character spaceship. If alien’s spaceship and main character ship get crashed with each-other, it will impact on main character spaceship health dramatically.

• The main character can avoid the small ships and aliens’ ships by moving further of their programmed boundary.

• The main opponent is the final boss of the level; it can damage more and hard to kill it. Enemy boss also can shoot missile towards the main character ship.

### Play Flow – How does the game flow for the game player

## Mechanics

What are the rules to the game, both implicit and explicit. This is the model of the universe that the game works under. Think of it as a simulation of a world, how do all the pieces interact? This actually can be a very large section.

### Physics

The space-shooters game is a side-scroller game. The screen will follow player spaceship as it moves right side. So, the action is from left to right.

For example,

The player sprite is movement is (x, y)

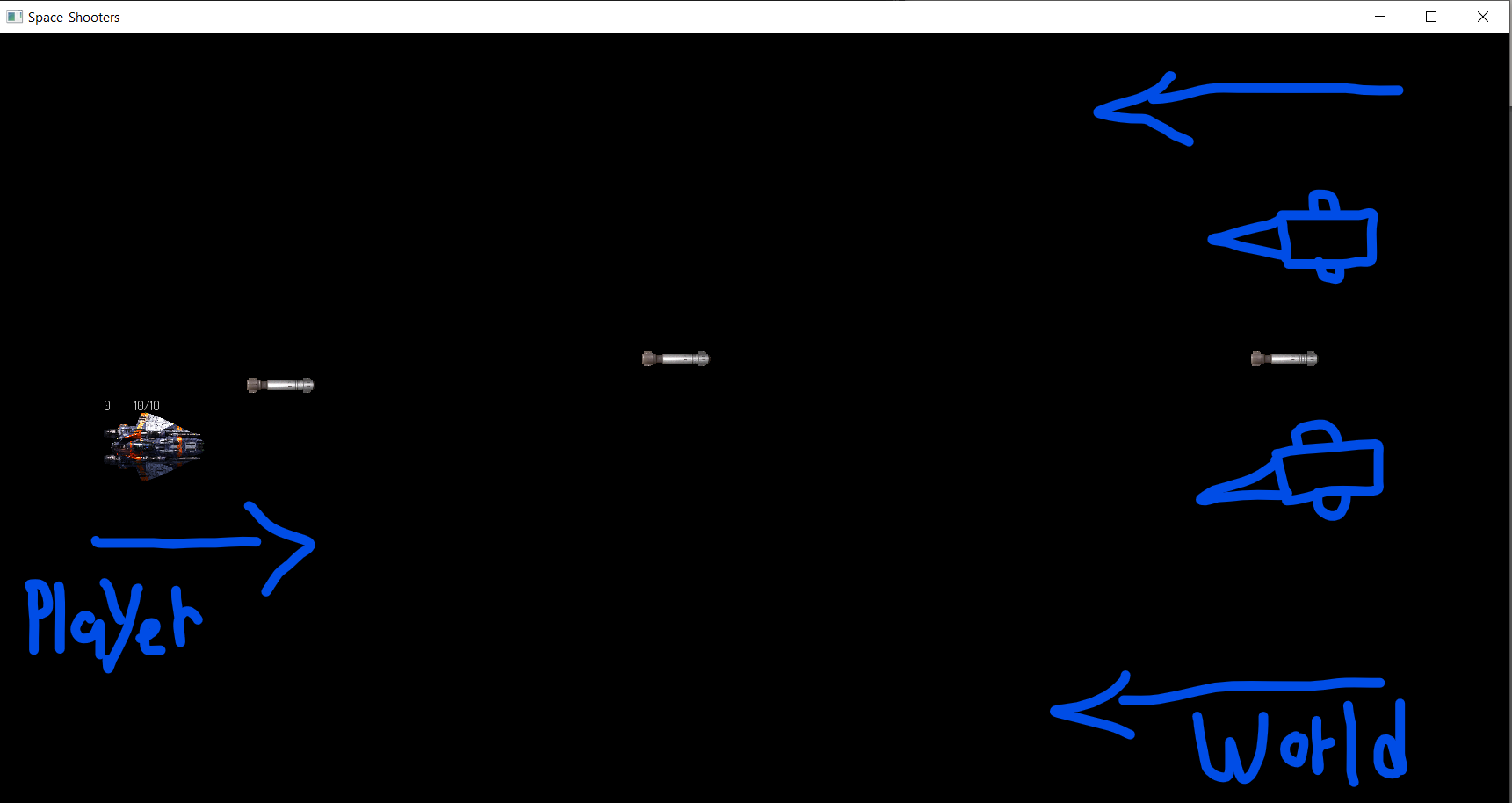
The bullet movement when the acceleration is 0, current velocity is acceleration \* direction of the bullet.

### Movement

#### General Movement

The aircraft movement is from left to right generally. (To move ahead).

The map movement behavior is right to left.



#### Other Movement

The player aircraft and the enemies will be on the same layer of the world. If they collide, it will destroy who have less health then other it collides with.

### Objects

#### Picking Up Objects

There will be different objects.

1. Health for player aircraft

2. Missiles for player aircraft

3. Upgraded bullets for player aircraft.

#### Moving Objects

There will be small asteroids that will move and some of them will be directed to player aircraft to make some damage.

### Actions

#### Switches and Buttons

There are many certain key controls for controlling the player spaceship and game behavior.

**W | UP** – upside of the map

**S | DOWN** – downside of the map

**D | Right** – right side of the map (in side-scroll game case, it refers to forward.)

**A | LEFT** – left side of the map (in side-scroll game case, it refers to backward)

**SPACE** – Shoot bullets.

**M** – Shoot missile

### Combat

The player aircraft, enemy aircraft will be on the same layer of the world.

There will be some collision between player aircraft and enemy aircraft.

The bullets will also on the same layer of the world. Bullets will be erased when it reaches to the windows max position. (Same direction of bullet)

## Screen Flow

### Screen Flow Chart

A graphical description of how each screen is related to every other



### Screen Descriptions

The game screen has different option to choose/select. Player can choose new game or quit the game.  
the player also can select the pause option during the game to pause the game and continue the game.

#### Main Menu Screen

The main menu is for selecting the options such as playing new game and quit the game.

# Section III – Story, Setting and Character

## Story and Narrative

Specific details like scripts and cut scenes may not be in this document but be in the Story Bible.

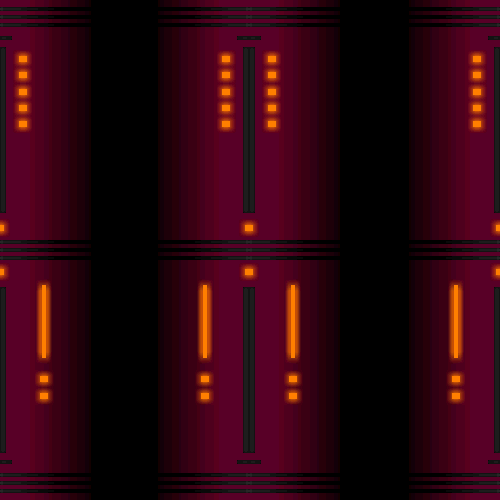
### Back story

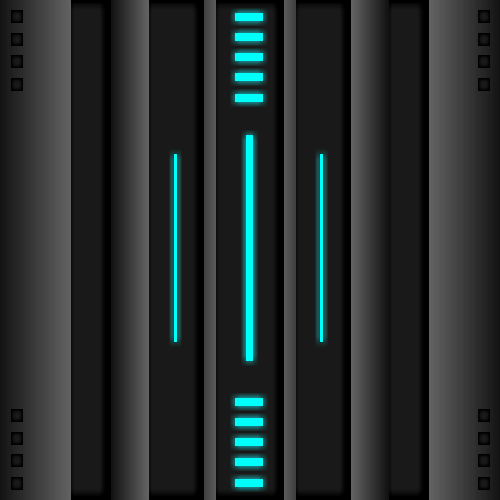
The spaceship is lost in space by mistake, and it must be survived and go to his destiny by defending some enemies and boss. The spaceship has different kinds of abilities that could be useful to stay alive and fight with boss.

## Game World

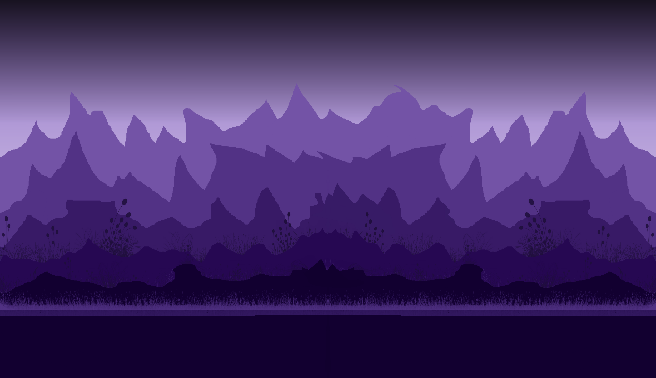
### General look and feel of world

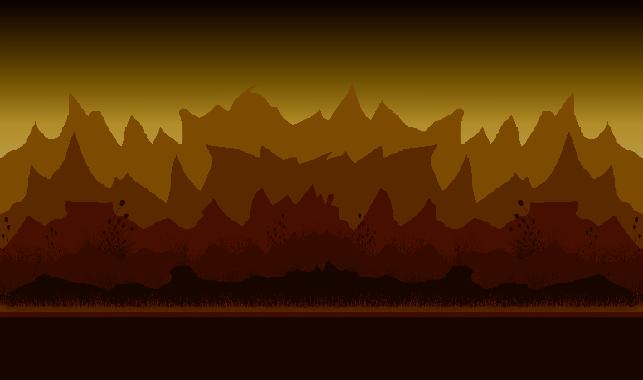
The space-shooter game is based on space map. The map is dark, colored, and somehow scary. There will be some mountains which have different colors. The map will be 2-dimentioned with some textures. The gameplay experience will be same as a space war. Overall, the space-shooter game is an outer space environment game.



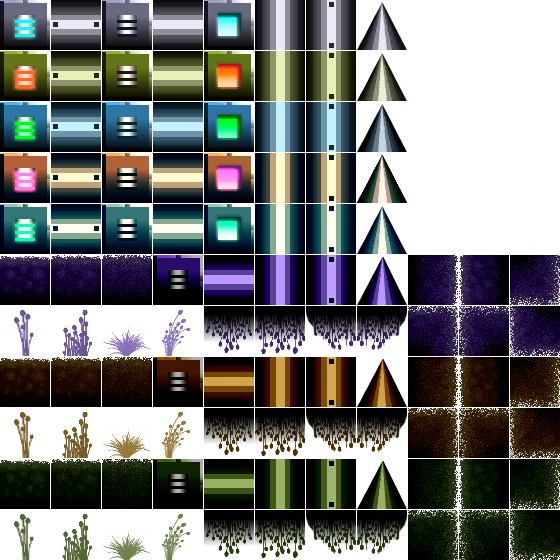


* Some background





* The map layout.



* Missile, upgraded missile, health pickups.

A picture containing device, gauge, close

Description automatically generatedA picture containing gauge

Description automatically generatedA picture containing device

Description automatically generated

## Characters

### Character #1 Player Aircraft

#### Back story

The main character is a spaceship having different launch paddles of bullets. The main character has bullets as well as missiles. The main character also has shield to protect itself from enemy bullets and missiles.

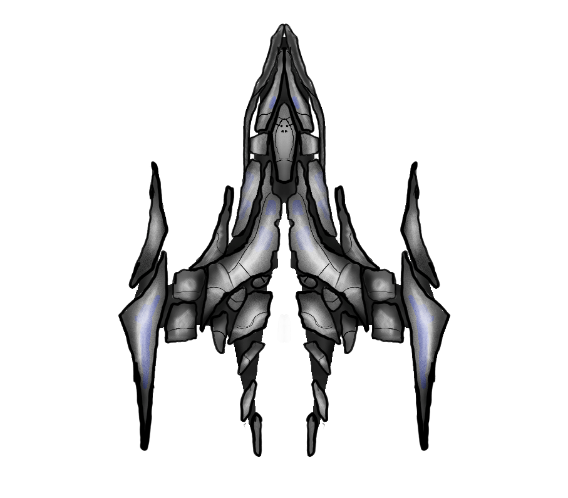
#### Personality

The spaceship Avtar will change as it reaches to different levels gradually.

#### Look

##### Physical characteristics

The main character Avtar is very dark, scary and looks like violent spaceship.



##### Animations

There will some cool animations according to player preference. It will make player aircraft more like a space war machine.

* + - * + Player gun animation

(Double click to play)

#### Special Abilities

The main character has bullets as well as missiles. The main character also has shield to protect itself from enemy bullets and missiles. The speed of spaceship is normal, there is no such power that boost up the speed of the spaceship.

#### Aura for player

The player spaceship has some aura which makes the ship good looking and aggressive.

Icon

Description automatically generated A close-up of a fire

Description automatically generated with medium confidence

The Aura will be rotate at 5\* to look player ship more Interesting

### Character #2 Enemy Aircraft - Small Ships

#### Back story

There will be many opponents in the game, first will be small ships that can damage the main character spaceship in certain amount.

#### Personality

The small ships are programmed to follow the main character spaceship and damage it. It will happen if the main character spaceship enters these small ship’s allocated boundary to follow main character spaceship.

#### Look

##### Physical characteristics



##### Animations

The small ships will glow at a certain amount of time.

#### Special Abilities

The small ships can cause small amount of damage to player aircraft. They will follow the player aircraft.

### Character #3 Enemy Aircraft – Alien’s Ships

#### Back story

Second opponent will be aliens’ ships that also could damage average amount of damage on main character spaceship.

#### Personality

The aliens ship just can shoot bullets towards the main character spaceship. If alien’s spaceship and main character ship get crashed with each-other, it will impact on main character spaceship health dramatically.

#### Look

##### Physical characteristics



##### Animations

The Alien’s ship has some glowing animation.

#### Special Abilities

The alien’s ships can cause more damage than small ships. It will also be programmed to follow the player aircraft. Alien’s spaceship can launch missile to kill player aircraft.

### Character #4 Enemy Aircraft – Final Boss

#### Back story

The final boss has advanced powers than player aircraft. The big aircraft who owns the specific area where the player aircraft loss is the final boss of the space-shooters game.

#### Personality

The main opponent of the game is the final boss of the level, it can damage more and hard to kill it. Enemy boss also can shoot missile towards the main character ship.

#### Look

##### Physical characteristics



##### Animations

The final boss has some glowing and particle animations.

#### Special Abilities

The final boss has superior abilities. It can shoot bullets, missiles, and big bombs towards the player aircraft. It also has more amount of health than player aircraft. It can cause ample amount of damage. It is hard to destroy the final boss of the game.

# Section IV – Levels

## Level #1

### Objectives

The game has different levels, and the difficulty of the game is based on the level up of game. The player spaceship must kill enemies and boss to proceed next level of the game. If failed, it will bring you to first level of game.

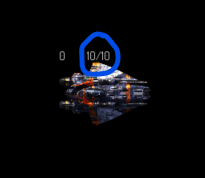
# Section V - Interface

## Visual System

### HUD - What controls

There are many HUD recognized in the game.

1. Health of player Aircraft



1. Missile ammo



1. Score
2. Some game controls
3. Game level number

### Menus

The game has menus for navigate through the game. The main menu displayed when the game is started. There will be 2 options available. First is to play the game and second is to quit the game. During the game, player can pause the screen and that is the second menu for the game. Player can simply press P back to return to the game or press ESC for returning to the main menu.

### Camera

The space-shooters game is a 3rd person shooting game structured around shooting, and in which the player can see the avatar on-screen in a third-person view.

## Control System

The space-shooters game is controlled by player keyboard. WASD is the commands for up, left, down, right respected. For second player, IJKL up, left, down, right are the controls.

## Audio

There will be game introduction audio for Space-Shooters game and also there will be gameplay audio. (double-click on the icons to play music)

1. Introduction audio

 [Reference](https://www.bensound.com/index.php?route=product/search&search=evolution&sort=relevance&order=DESC)

1. Gameplay audio

 [Reference](https://www.purple-planet.com/tracks/deep-roots)

## Sound Effects

Space-shooters game have different music for airplane, spaceship destroy music, enemy aircraft destroy music, bullets music, game-over music and many more. This section will be updated as the music will be finalized.

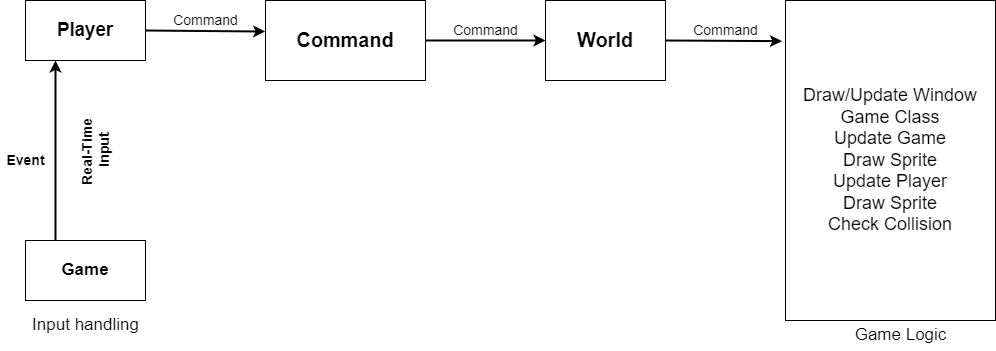
# Section VI – Game Engine

## General Description

The game engine is built on C++ programming language and SFML (Simple and Fast Multimedia Library) API. SFML is a cross-platform software development library designed to provide a simple application programming interface to various multimedia components in computers. It is a 2D rendering game engine.

## Game Logic

This is a rough game logic diagram. It will constantly updates as the game will developed.



## Support AI

### Player and Collision Detection

The player and enemy will be on the same layer. The collision must check between them. Also, pickups, bullets, enemy bullets, enemy missile, map will be on the different layers. There must be collision check between them also. The collision will be checked in the game class. If the collision event will become true, certain number of actions will be performed such as crash, destroy, increase health, decrease health etc.

1. Update players, bullets and combat first.
2. Update players (Player Aircraft)
3. Update bullets (Player bullets)
4. Enemy collision checks with bullets
5. Enemy take damage check
6. Enemy dead or not check
7. Window bound check (erase bullets)
8. Player-enemy collision check (intersects)
9. Enemy window bound check (erase enemy)

### Pathfinding

# Section VIII – Game Art

Only show concept art here (rough sketches)

## Concept Art

## Characters

## Environments

## Equipment