Asteroids

Technical Design Document

*Version 1.0*

*Omar Saeed*

*Website: secondwarrior.com*

*Mobile: +201008569433*

# Table of Contents

[Table of Contents](#_f3hbr6p0ak37)

[Design History](#_e5akww2oi216)

[Game Overview](#_j2kld1ana13b)

[Game Summary](#_i566fs1eqqtn)

[Target Platform](#_w0h9hbmhvcr)

[Development Overview](#_h7psffrwssbg)

[Development Team](#_xxk0bak9cyj9)

[Development Software](#_drxstdfky6pr)

[External Code](#_3wx81gnjfbw1)

[Game Mechanics](#_hm65yr3rs85w)

[Main Technical Requirements](#_c2kb94gs6u8i)

[Architecture](#_yxdrr8vse4bb)

[Game Flow](#_93tph0q2eust)

[Graphics](#_dal8viau5j83)

[Audio](#_oc7d279r9ymu)

[Physics](#_w9h7t0xxygzf)

[Game Objects and Logic](#_jm2i9mtxtpro)

[User Interface](#_rvrktgcok4ku)

[Main Menu](#_qc8tfbuzct1k)

[In Game HUD](#_9sbf9gvn1b4w)

# 

# 

# Design History

Version 1.0: First produced version of the game.

# 

# Game Overview

--------------------------------------------------------------------------------------------------------------------

# Game Summary

As a space ship trying to destroy other enemy space ships, you are also trying to avoid being hit by space asteroids, while facing a strong enemy waves coming in sequence, Developed upon the mechanics of the legacy Asteroid mobile game.

## Target Platform

As an arcade game, it mainly target PCs, with the support to android and ios platforms.

# Development Overview

--------------------------------------------------------------------------------------------------------------------

# Development Team

[Omar Saeed](http://github.com/Omar47i/)

Game Design and implementation, UI Design, Animations, Sounds, AI, pretty everything.

# Development Software

Project is developed using Unity engine, Photoshop is used for UI image editing, Version control will be handled by [Git](http://git-scm.com/) with the repository hosted by [GitHub](http://github.com/).

# External Code

[iTween](https://www.assetstore.unity3d.com/en/#!/content/84) free unity plugin has been used to implement simple animations for some game elements e.g. fruit spawning, bomb spawning, and snake’s bite.

The game has been made from scratch, though i used 2D assets from my colleague.

# Game Mechanics

--------------------------------------------------------------------------------------------------------------------

# Main Technical Requirements

The most technical challenge is the enemy waves spawning algorithm, Most other aspects of the game are not so intensive: Firing on a spaceship/asteroid, Get hit by an enemy spaceship/asteroid, Player ship lives, Cannon upgrade and pickup, Scoring text, Teleporters that teleport player ship when reaching screen border, Player score and best score, Saving and loading player settings (Sound and Music) and best score, Sound management, Main menu and In game UI animations, Mobile controls and UI, Transition between scenes, etc..

# Architecture

There are two main modules in the game. These are: Home module and gameplay.

The home module hosts the Title panel animations, settings (sounds and music toggle), and play button.

The gameplay module hosts everything related to the gameplay.

## 

# Game Flow

The game has only one flow which is the **Main Game**, but with two scenes.

When game starts the title panel animation slides in, displaying the game title, space ship icon animating, play button and settings buttons, with the ability to click/tap on music or sounds toggle to switch on/off them or play button to start playing.

When you start the game the title panel animation slides out and a new scene is loaded which is the Gameplay scene.

New scene is opened with two HUD components which are score text and player lives (heart icons).

You start with a spaceship in the middle of the screen and after one second the enemy waves starts to enter the game area in a randomized position and direction.

Now you can shoot at the enemy ships and asteroids and get shot by them, Sometimes an upgrade pick up will appear to give the player extra cannons that fires three projectiles at once.

Getting hit by asteroids or enemy ships will result in losing a life, when you lose all your three lives the game over pop-up appears prompting the player to even restart or return the main menu.

When you click on restart or return a slide out animation starts.

# Graphics

Although the game is best played using a keyboard, It also target mobile device, All game sprites are compressed to reduce final build size and memory usage especially on mobile devices, reference resolution for the game is 16/9 though the game can be played on 4/3 devices.

# Audio

Game has main menu and gameplay music, music and sound effects can be toggled using the main menu toggle buttons.

# Physics

Game uses Dynamic rigid bodies to simulate physics collisions like player space ship, enemy space ship, and asteroid objects, While Kinematic rigid bodies are used for the double cannon pickup, all objects have physics layers that interact only with certain other layers, to reduce the overhead of calculating the physics for every other layer.

# Game Objects and Logic

**Player Space Ship**

The main game actor that is controlled by the player, As a space ship you can move forward and backward and turn in 360 degrees while shooting at the asteroids and enemy ships and consuming pickups.

**Asteroid**

The main hazard in the game, Asteroids have different level of HP that allows them to split on hit or explodes on one hit, they move in a certain direction and rotates around their centers while moving, can be hit by player and enemy space ships.

**Enemy Space Ship**

The main NPC in the game, enemy space ships come in different speeds and directions and they can shoot at the player and the asteroids.

**Double Cannon Pickup**

Upgrades the player ship shooting rate to be three projectiles at a time.

**Teleporter**

Invisible colliders that teleport the player when he reach end of the screen.

**Hit VFX**

Different versions of explosion particle effect that is spawned on hit.

**Scoring Text**

As a rewarding effect the scoring text is created after killing an enemy ship or an asteroid, a special scoring text is created when beating the last best score.

# User Interface

--------------------------------------------------------------------------------------------------------------------

# Main Menu

The main menu contains the title of the game associated with a spaceship icon and three buttons: Play, Music Toggle, and Sounds Toggle.

# In Game HUD

The game HUD contains:

* Current score text.
* Player Lives (heart icons)
* Game Over popup