# BATTLE SHIPS

**Unity 3d Game Framework**

Requires **Unity V5.3.4+**

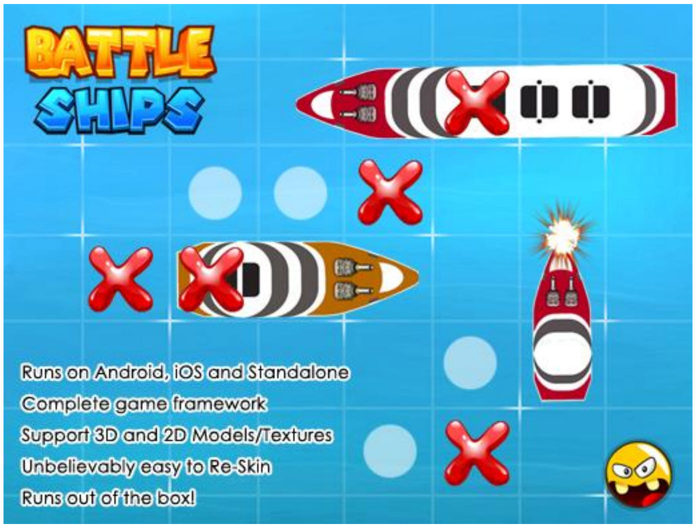
Supports Android, iOS, WebPlayer, Windows and Mac

Dear Customer,

Thank you so much for purchasing this game framework.  Here you can find the most important information on how to use this project efficiently. All script and code assets are fully commented, but if you ever needed a hand on a block of code or anything else, feel free to contact us at www.finalbossgame.com . We'll try our best to support you with your questions as soon as possible.

## Overview

Battleship is a classical naval strategy, in which the player tries to sink the hidden ships of the opponent. Each player is given 5 shots at the start of each round, and they are free to target any tile on the map to shoot a missile. If the tile is taken by a ship which belongs to the opponent, the tile is marked with a "Hit Flag". Players should continue this procedure and the one who sank all the ships is the winner.

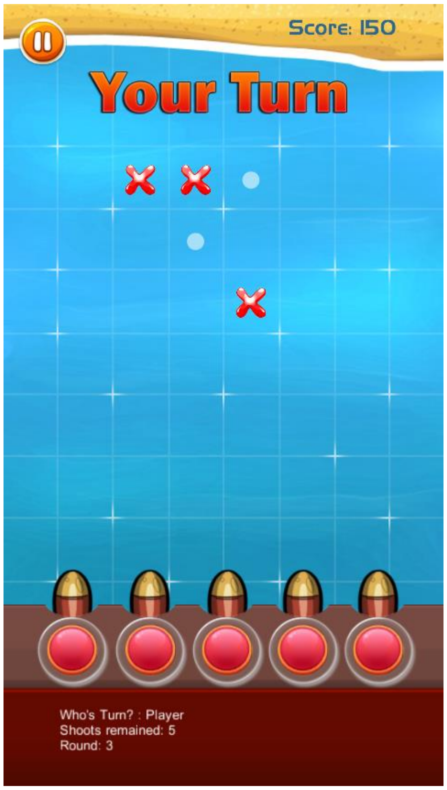

BATTLE SHIPS
Runs on Android, iOS and Standalone
Complete game framework
Support 3D and 2D Models/Textures
Unbelievably easy to Re-Skin
Runs out of the box!

The game accepts both touch and mouse inputs, and thus, can be tested on **Android**, **iOS**, **WebPlayer** and **Stand-Alone** platforms.

This game kit needs no 3rd party plug-ins to works. It runs and builds out of the box. All you need to do is to load the kit inside Unity, set the project on the desired platform and hit "Build" to receive your game in no time!

## Game Play

Battleship is a turn-based strategy game. Each player must wait for their turns, and have the opportunity to shoot at 5 desired tiles on the game map. The purpose of the game is to sink all opponent ships, so player must carefully inspect the map and guess the positions that an enemy ship might reside.



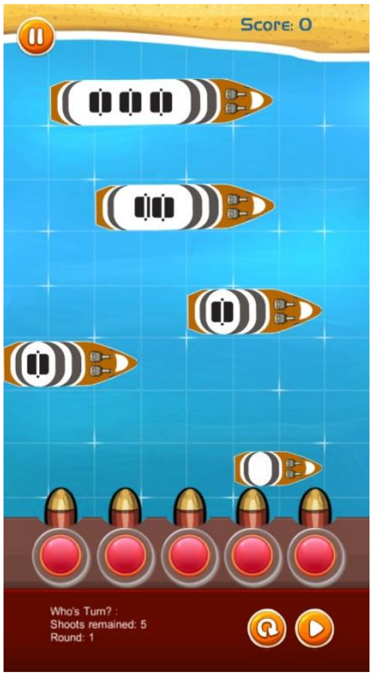
Who's Turn? : Player
Shoots remained: 5
Round: 3

When player or AI shoots at a tile, it will set a flag. If there was any opponent ship in that tile, the flag is set as correct-hit (red-X) and if the shoot is missed, flag is set as a simple white circle.

When all parts of a ship is hit, it will get revealed.

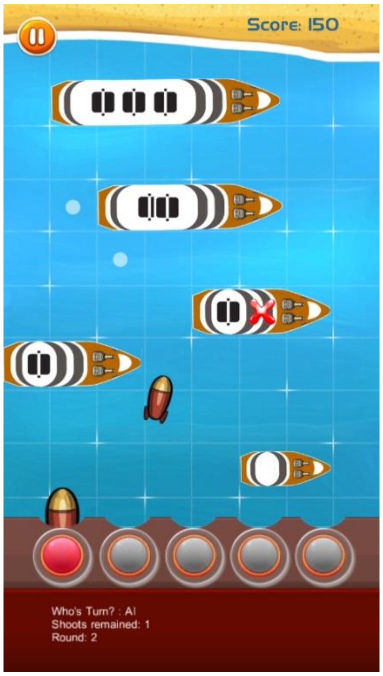
Before the game starts, player has the option to change his initial formation with one of currently available formation designed for the game. There is currently 4 initial formations. In the next updates, we will add the option to manually re-position the ships and rotating them to your liking, resulting in a more advanced tactics.

There is also a simple score manager which counts a combo bonus, when you repeatedly hit the correct tiles. Your high-score will gets saved inside the game.



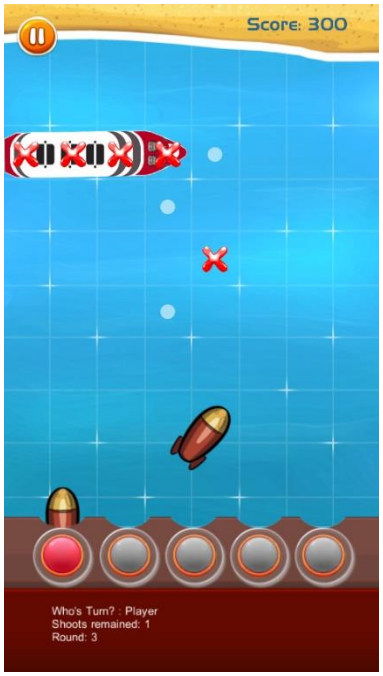
Who's Turn? :
Shoots remained: 5
Round: 1

Starting the game



Who's Turn? : AI
Shoots remained: 1
Round: 2

AI's turn



Who's Turn? : Player
Shoots remained: 1
Round: 3

Sinking an AI's ship

# Classes

This game framework uses a few separate classes to control the game's main routine. All these classes are fully commented and you can easily guess the dataflow. But we will try to introduce them here at a glance:

- **GameController:** This is the main game controller class. It handles tile instantiation, game initiation, turn management, calculating the score, monitoring and updating global arrays, and almost all the events that happen inside the game. If you want to start tweaking the framework, this is the place you need to start.

- **FormationPresets:** In the initial version of this battleship game framework, we are using a simple formation system for ship placement which fetches a random formation from the pool of available formations designed for the game, whenever the game starts or player taps on the shuffle button. You are free to add unlimited number of formations. Player can use the shuffle button before starting the game to select a good formation for his ships.

- **MapTileController:** This controller saves the state of each individual tile object on the scene and presents them to other controller for further usage. It also handle the after-shoot events and set the correct flags when a missile is hit/not-hit.

- **MissileMover:** This class rotates and moves the missile towards the selected target tile. Target tile will be set by other controllers.

- **PlayerInput:** This class handles all player input interactions on the game map, including tap on tiles, shooting a missile, checking if the missile hit a ship, setting the scores and switching the turns after all shoots has been performed.

- **ShipController:** Main ship controller. Controls ship size, health and occupied tiles. Ships are invisible at the start and will not be revealed until completely destroyed. When all tiles which a ship resides are hit, the ship gets completely destroyed. When its player's turn, all destroyed ships are always visible.

- **TurnManager:** This is a simple class which is used to show/hide a plane that shows the turn of each side when the turn switches. This object will slide in/out from the top of the screen.

## Got any questions?

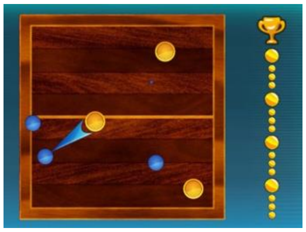
If you have any questions, feel free to ask us at http://www.finalbossgame.com and we will get back to you as soon as possible.

## Our Other Cool Game Kits

**Frenzy Farming Game Kit**
Unity Assetstore

SKY BURGER
Unique & Highly addictive game-play
Compatible with Android, iOS and Standalone
Unlimited number of levels & Missions
AdMob banner, Interstitial & Video Ads
Quick Rate + Share widgets included!
Mobile Friendly
Very easy to Re-Skin
Runs out of the box!

**Sky Burger Game Kit**
Unity Assetstore



**Turn Based Ball Fight Game Kit**
Unity Assetstore

**Swing Game template**
Unity Assetstore

**Stealth Action Game Kit**
Unity Assetstore

**Restaurant & Cooking Starter Kit**
Unity Assetstore

**Endless Space Pilot Game Kit**
Unity Assetstore

**Snakes & Ladders Framework**
Unity Assetstore

**Finger Soccer Game Kit**
Unity Assetstore

**Monster Blaster! Game Kit**
Unity Assetstore

**Real Estate Tycoon Game Kit**
Unity Assetstore

**Head Soccer game kit**
Unity Assetstore