-Instruction-

Aqua Bubble Shooter Full Game Asset

What Included

Graphic

Editable PSD and PNG

- Premade Game Screen in PSD, 2048 X 2732.
- 4 Game Backgrounds in PNG, 2732 X 2048.
- 30 Colorful icons.
- 9 PSD Pop Ups like Win, Loose, Mission, Coin, Life and Booster Shop, Settings and Message, 2048 X 2732
- 60 vector icons for the buttons and design in PSD and PNG.
- More then 20 GUI.
- Premade Map Screen with in PNG and PSD.
- 4 Seamless Vertical and 4 Seamless Horizontal Backgrounds for Maps in PNG, 2048 X 2732.
- Premade Preloader Screen in PSD, 2732 X 2048.
- All elements are also in PNG, ready for code.

Font

Not Included

Arista Regular Link

Code

Ready game

Aqua Bubble Shooter Full Game Asset is a new powerful Pop game with complete solution of bubble shooter possibilities in casual, marine, underwater, shiny, colorful style for your mobile/web/video game. It included premade Game Scene, Map, Pop Ups, etc. Working on IOS, Android, Web. The package is provided with complete source code, graphics files, properly commented code and with a detailed documentation for easy getting started.

In the asset realized next functionality:

- *Full Pop game functionality;*
- *Customizable Balls functionality;*
- *20 tested levels with different targets (you can add any numbers of levels);*
- *Easy to use Visual Level Constructor;*
- *Included Levels Map functionality for 80 Levels;*
- *5 Shops (life, coins, 3 boosters);*
- *Connect player to Facebook;*
- *Ready to in App Purchasing;*
- *Ready for sounds;*

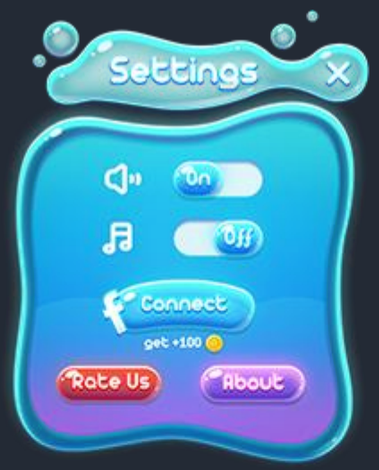
- Asset realize system of the players data saving (like coins, player level and game settings) on the device.
- Two premade Scene Pop Game and Level Map with ready buttons and Pop Ups functionality.
- GUI controller with Pop Ups

Note:

To work with layer styles in PSD files you need minimum Photoshop CC 2015 with Multiple Layer Styles.

All Pop Ups


You win
Your Score
14710
Save fish
Next


Settings
On
Off
Connect
get +100
Rate Us
About

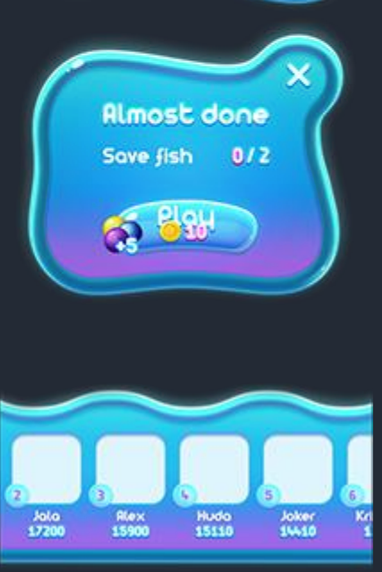

Life Shop
12h
5000
$9,99
+1
Watch
+5
$1,99


Mission
Level
112
Save fish
x 1
Play

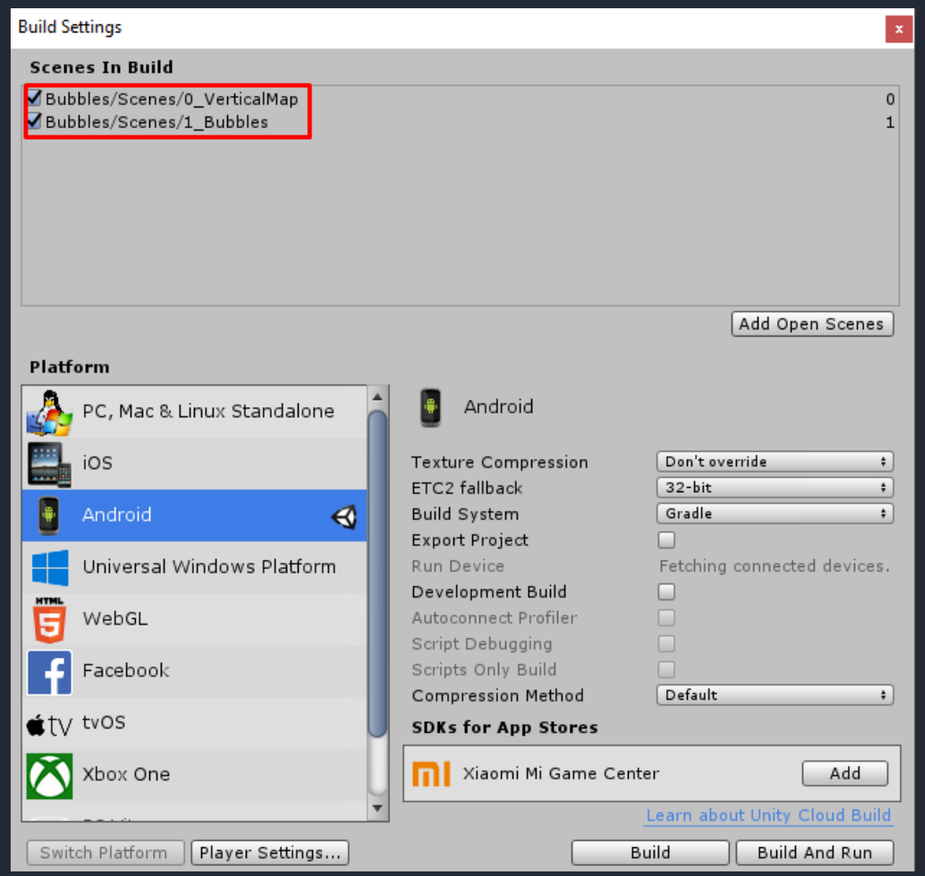

Failed
Save fish
Retry


Booster
Multicolor Ball
100
500
1000

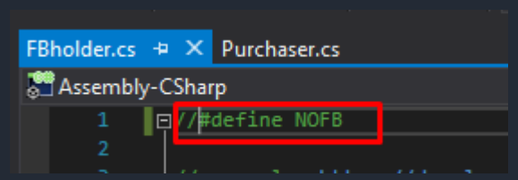

Coin Shop
12h
5000
$9,99
100
$1,99
1000
$4,99
5000
$9,99

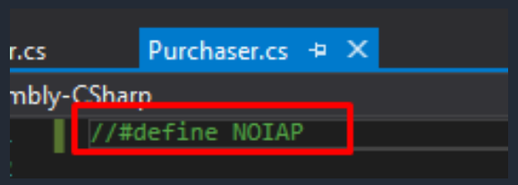

Almost done
Save fish
0 / 2
Play
Jola
17200
Alex
15900
Huda
15110
Joker
14410

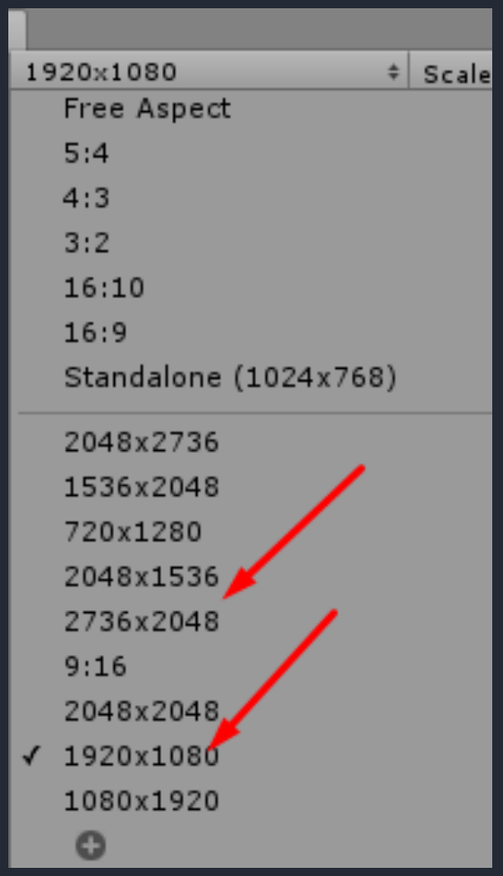
Creating project



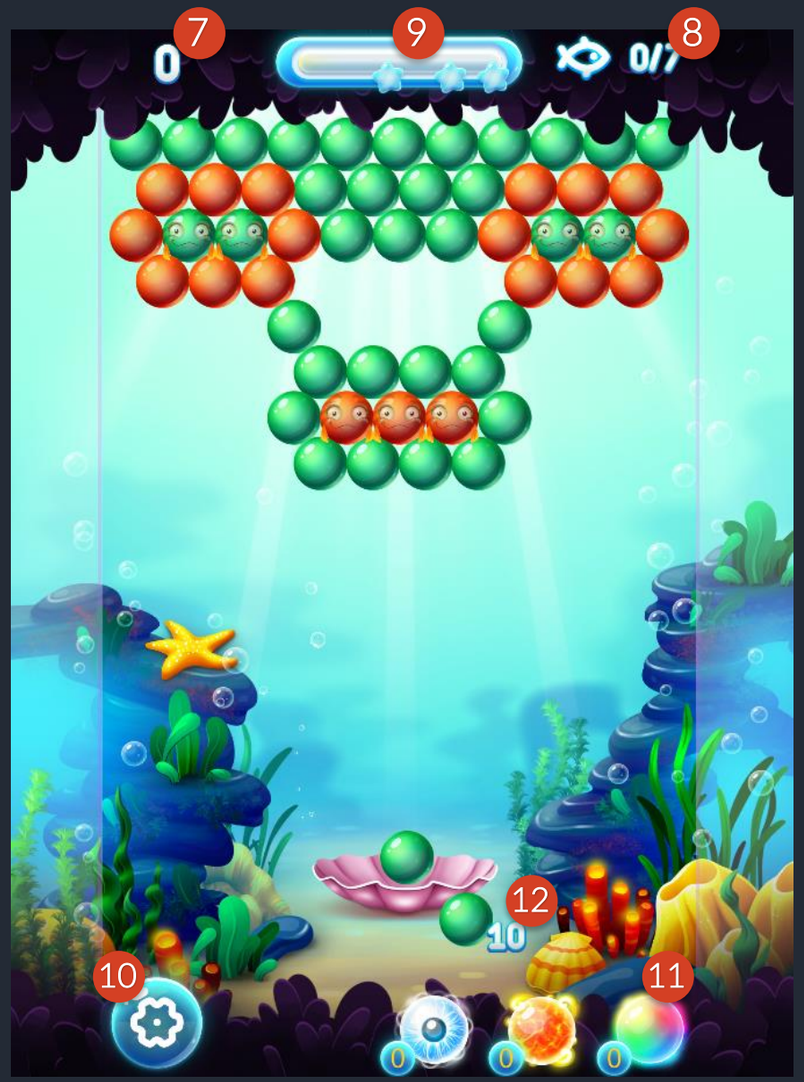
1. Create new project
2. Add if necessary Facebook SDK, comment #define NOFB (1 line FBHolder.cs)
3. Add if necessary in-app purchasing, comment #define NOIAP (1 line Purchaser.cs)
4. Import bubble asset
5. Open BuildSettings and add existing scenes
6. Close BuildSettings
7. Open scene 0_VerticalMap, press play
8. Set resolution for best fit (1080 x1920, 2048 x 2736, or 1536 x 2048)







Scenes Description



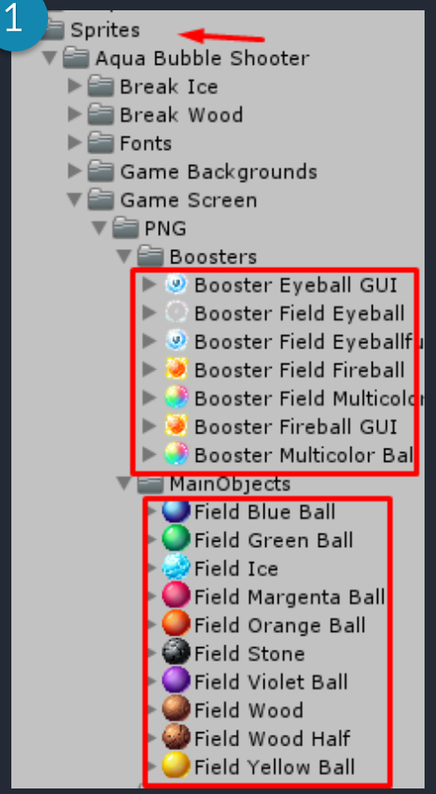
1. Scroll map with level buttons
2. Settings button
3. Coins shop button
4. Life shop button
5. Infinite life timer (or life count)

7. Score counter
8. Target counter
9. Score strip
10. Settings button
11. Boosters
12. Moves counter

Level Constructor

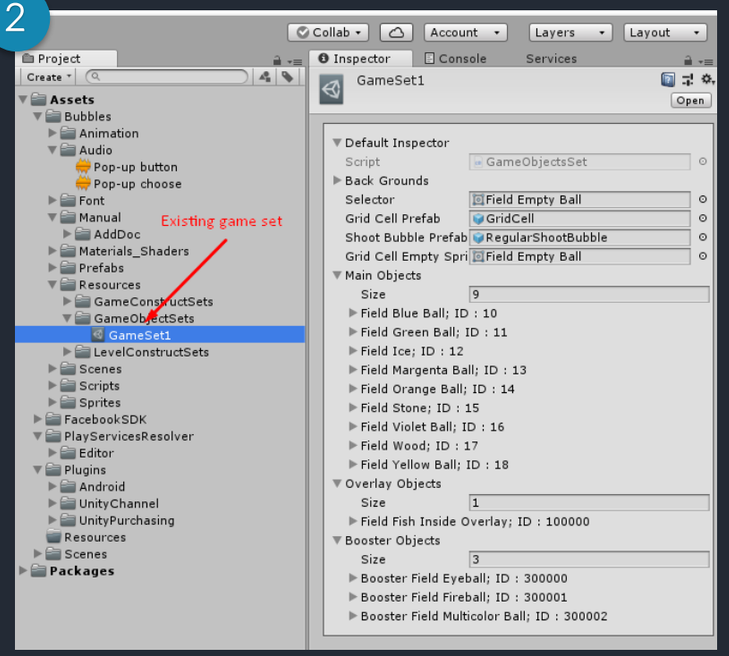
Create new objects set for your levels

1

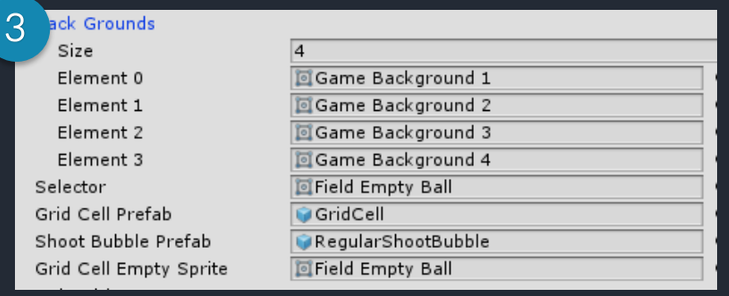


1. Copy a new images set in its own folder. Like here.
2. Duplicate the object GameSet1 (in the Folder Resources / GameObjectSets) CTRL + D. Or simple change images and prefabs in existing object set (GameSet1). We use 3 types of objects: main object, overlay and booster. The main object(bubble) is located on the game grid. Overlay used only with main object (over) like fish. Booster is used only for shooting.
3. In this section:
   - Backgrounds – level backrounds.
   - Selector – used to highlight the target.
   - Gridcell prefab – for bubble grid construct.
   - Shoot bubble prefab – gameobject with collider and ShootBubble script.
   - Gridcell empty sprite – sprite to highlight empty cells in edit mode.
4. Paste the objects images in the appropriate fields. If the object has additional states, paste its in the array <Protection State Images> for destroyable objects like wood.

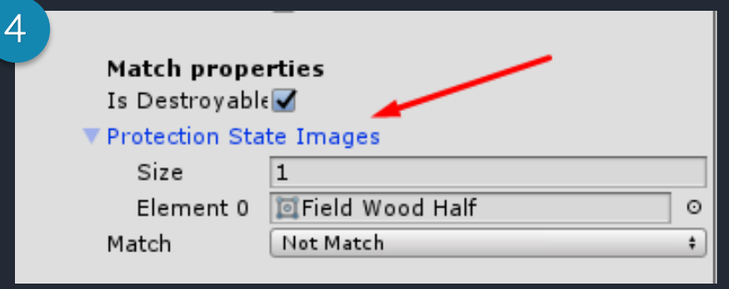
2



3

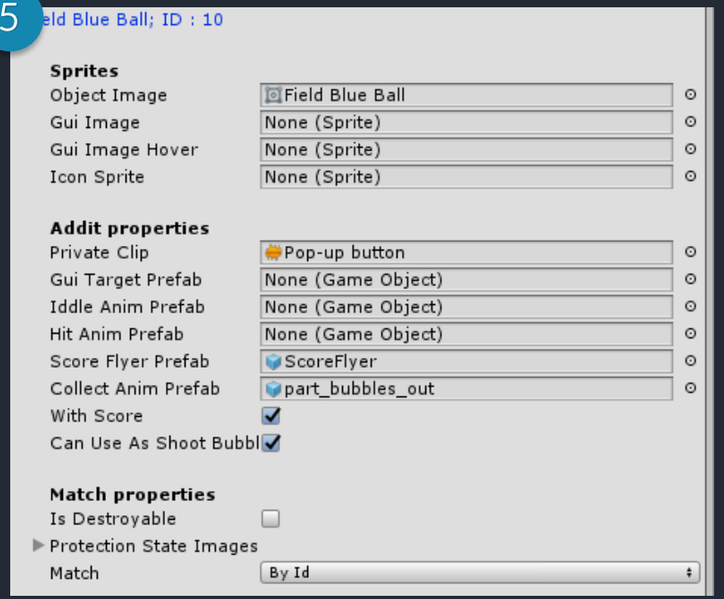


4

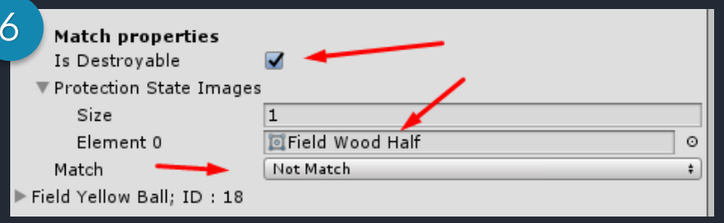


Level Constructor

5



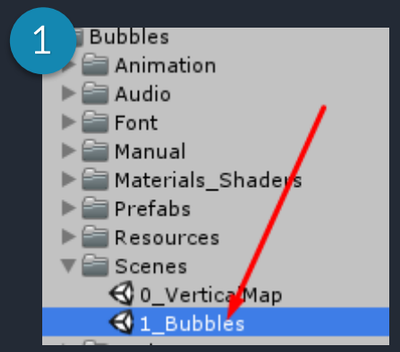
6

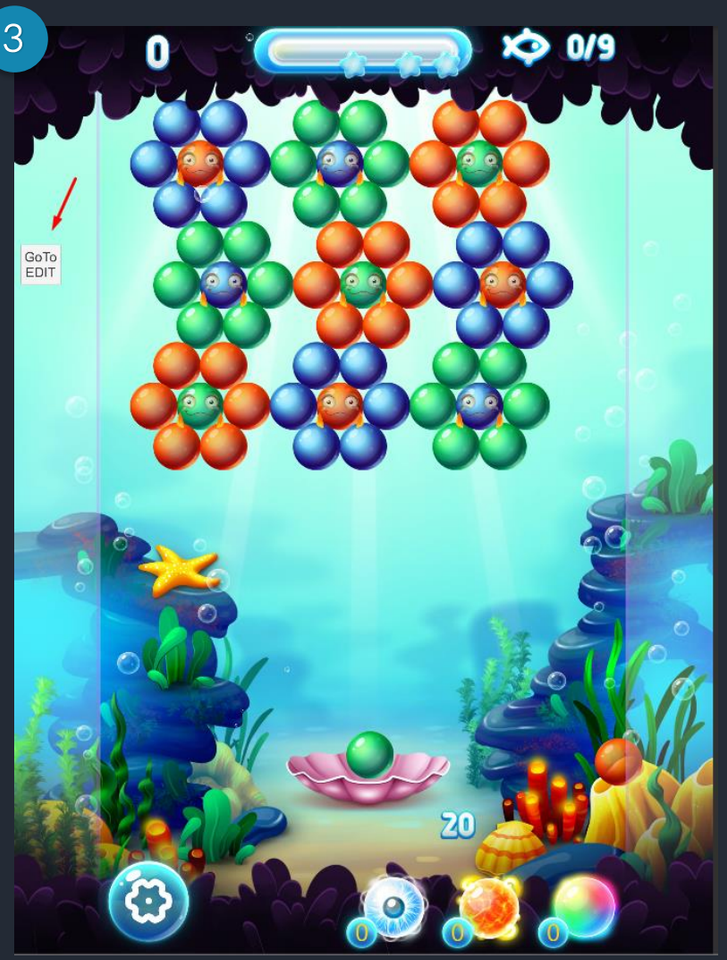


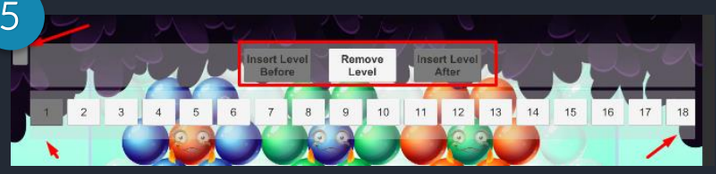
5. Object can has a different image for the GUI. Then you need paste a GUI images too.
   - If the object can used as shoot bubble, then check it.
   - If the object has score, then check <With Score> .
   - Regular objects will be grouped after shoot by color (ID) and collected.
   - Set Match property for such objects to By Id.
6. You can have also destroyable objects like wood with protection states. Usually these objects are not used as shoot bubbles and not grouped with another objects after shoot.

Level Constructor

Create and edit levels



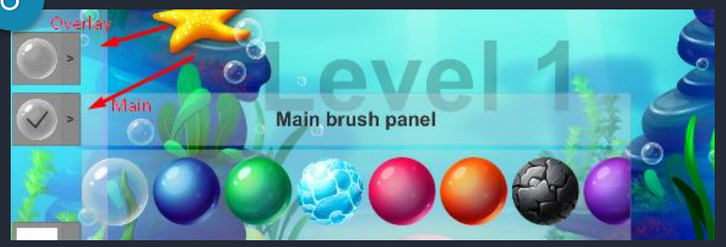




1. Load Construct Scene and press Play Button.
2. Set resolution for game window 2048x2732.
3. Press on the button <GoTo EDIT>.
4. Create or Edit Levels in RunTime mode.
5. In upper constructor panel you can select, create and delete levels. The selected level number and its grid are immediately display.

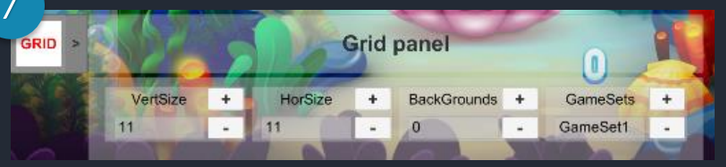
Level Constructor

6

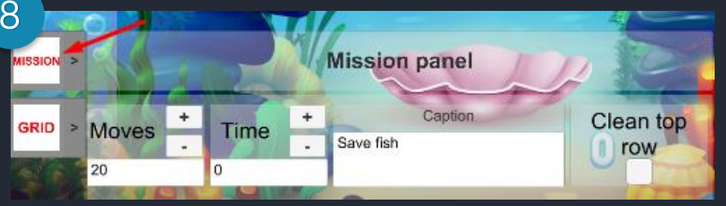


5

7



8



9

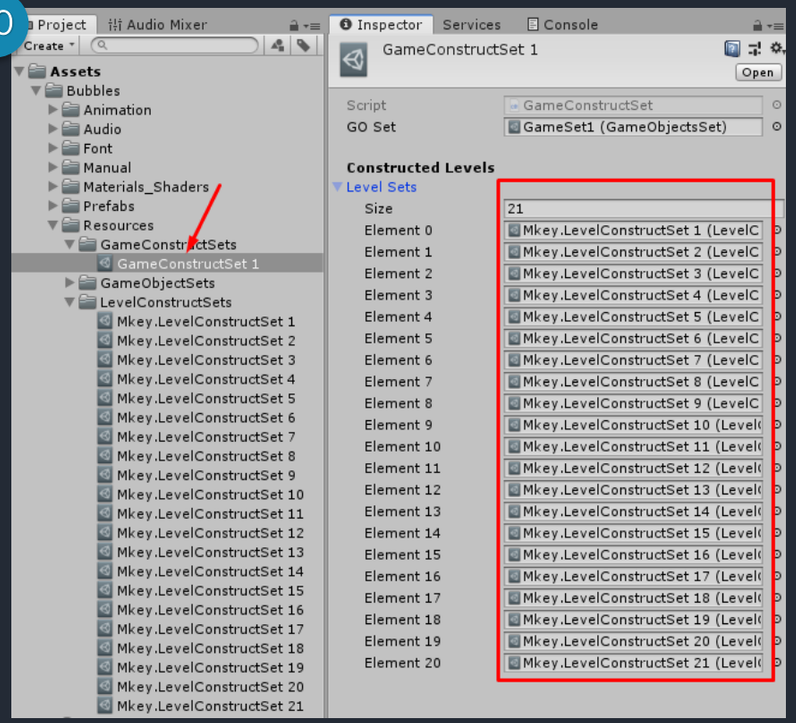
```
public enum LevelType { LoopTopRowLevel, TimeLevel, AnchorLevel, FishLevel }
```

```
public class MissionConstruct
{
    [SerializeField]
    private string description = "Mission";
    public string Description { get { return description; } }

    // time level
    [Space(8)]
    [SerializeField]
    private int timeConstrain = 0;
    public int TimeConstrain { get { return timeConstrain; } }  // priority 0 - remove all bubbles from board

    // loop top row
    [Space(8)]
    [SerializeField]
    private bool loopTopRow = true;
    public bool LoopTopRow { get { return loopTopRow; } }       // priority 1 - default (if not selected)
    //  [SerializeField]
    private int bubblesCount = 6; // default 6
    public int  BubblesCount { get { return bubblesCount; } }   // bubbles collect from top row (all default)

    // raise anchor
    [Space(8)]
    [SerializeField]
    private bool raiseAnchor = false;
    public bool RaiseAnchor { get { return raiseAnchor; } }     // priority 2

    // collect targets, like save fish
    [SerializeField]
    private ObjectSetCollection targets;
    public ObjectSetCollection Targets { get { return targets; } } // priority 3

    [SerializeField]
    private int movesConstrain = 0;
    public int MovesConstrain { get { return movesConstrain; } }
```

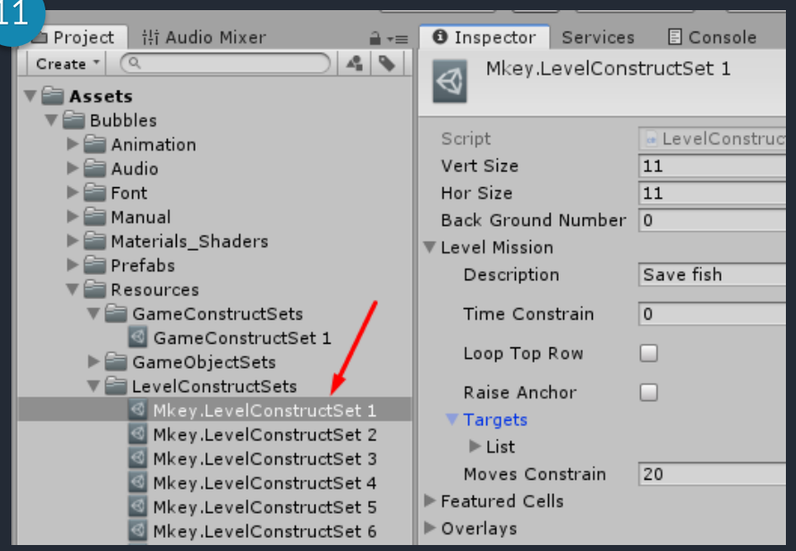
5. Grid move buttons.
6. Main and overlay brushes.
7. Grid settings panel. If you change horizontal size, set manually Left and Right colliders.
8. Mission panel. You can set only one from 4 possible missions:
   - Check clean top row, set moves count, write description
   - Set time, write description
   - Select overlay target, set moves count, write description
   - Check Raise anchor, set moves count, write description
9. MissionConstruct.cs with mission priorities.

Level Constructor

10



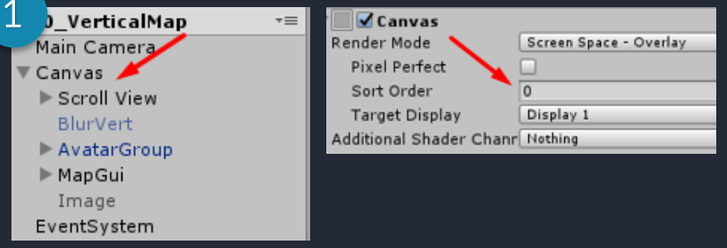
11



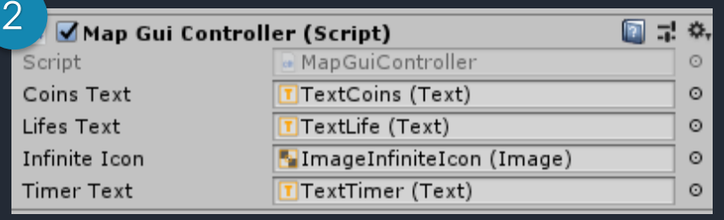
10. You can delete, add and adjust levels manually. The Object storing the game settings is GameConstuctSet 1. GameConstuctSet 1 consists some set of levels. It can be edited as ordinary objects in Unity in the Inspector Window.
11. Each level has its own LevelConstructSet object. It can be edited manually also.

Map scene scripts and settings

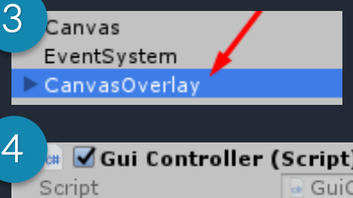
1



2



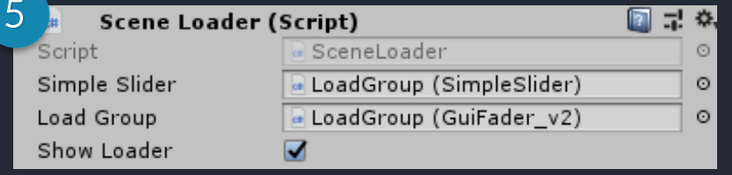
3



4

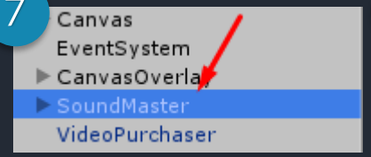
1. All object in map scene created on canvas. Canvas sort order = 0 (bottom canvas).
2. MapGui has attached script – MapGuiController.cs.
3. Overlay canvas (sort order 1) are used for instantiating pop up windows, and also contains LoadGroup object – simple pop up with scene loading progress. CanvasOverlay has 3 attached scripts:
   - GuiController.cs; - SceneLoader.cs; - DontDestroyObj.cs.
4. GuiController.cs – used for all pop up window instantiating.
5. SceneLoader.cs – used for loading scenes and show loading progress image.
6. DontDestroyObj.cs – prevents CanvasOverlay destroy by scene changing.
7. Scene object SoundMaster – object that controls game sounds.
8. Scene object FacebookHolder. FBHolder.cs contains methods for login, logout, get player info.
9. Scene objects VideoPurchaser and StorePurchaser. Contains data for the shop pop up.

5

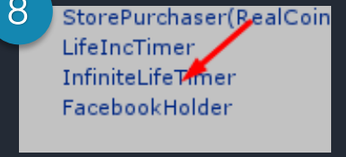


6

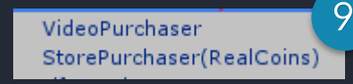
7



8

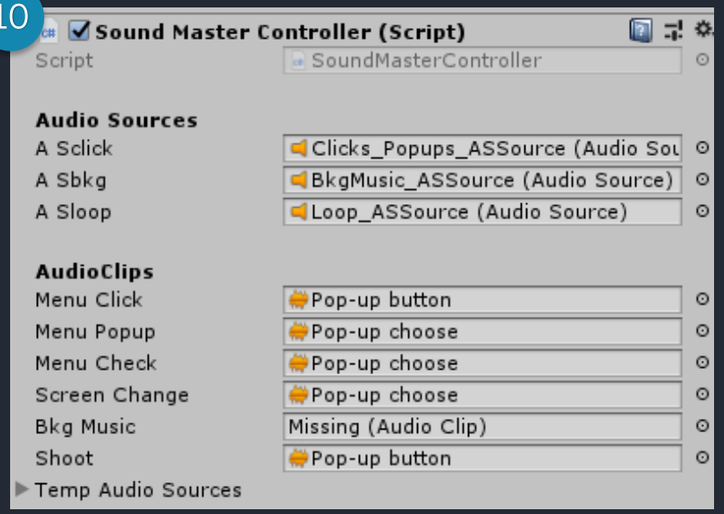


9

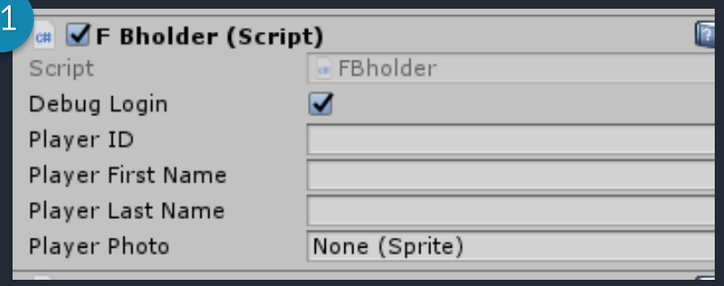


Map scene scripts and settings

10

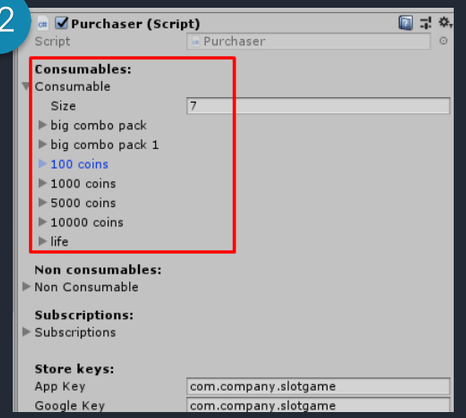


11

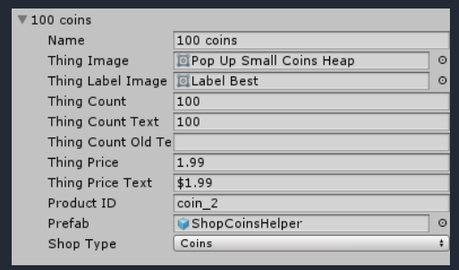


10. SoundMaster settings. Contains all sound clips. You can add your own audio clips here. Included audio is only for demo purpose. For playing any clip call SoundMasterController.Instance.Play…();
11. After login on Facebook, script is automatically fill next fields – playerID, player first name, player last name, player photo. You can access Facebook data using FBHolder.Instance.playerID FBHolder.Instance.playerFirstName, etc.
12. Store purchasing (for real money) settings. You can add your own consumables or non consumables goods here. Two shops (Coins and Life). Set unique ID for each product.
13. All settings for the goods in the shop. Look to the images to understand all fields.
14. LifeIncTimer and InfiniteLifeTimer – life control timers.

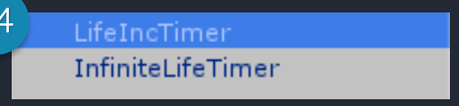
12



13

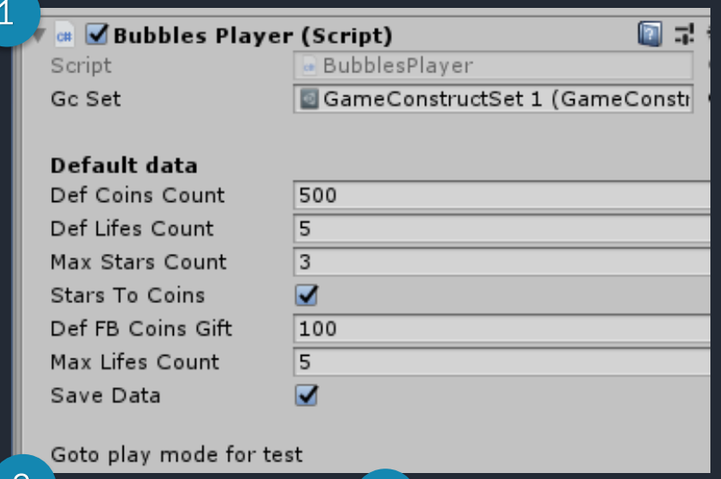


14



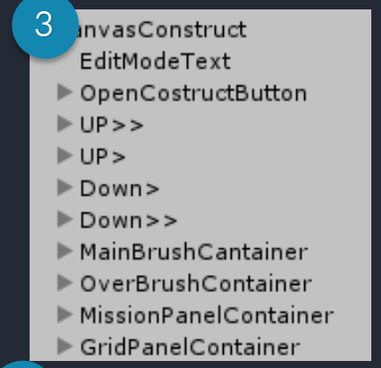
Game scene scripts and settings

1

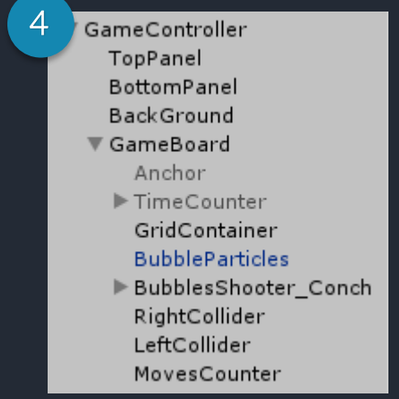


2

3

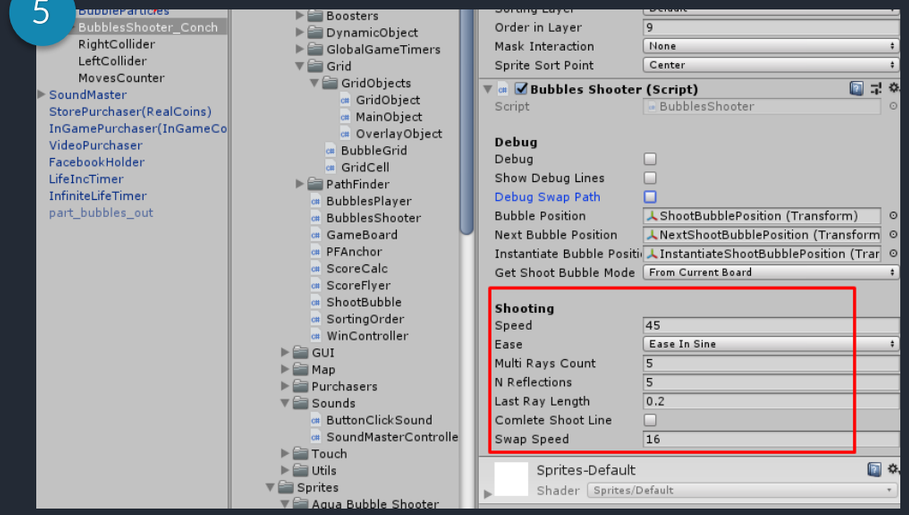


4



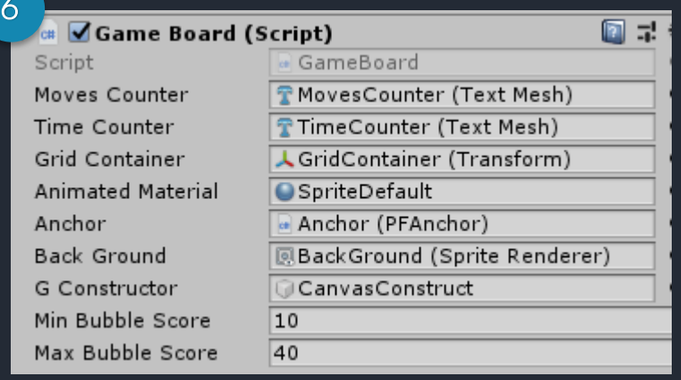
1. Bubbles player default settings, see BubblesPlayer.cs
2. Scene object CanvasMain contains SceneGUI object with HeaderMenu and FooterMenu. SimpleTouchPad – touch input helper for touch devices. CanvasOverlay used for pop ups.
3. Scene object CanvasConstruct contains all controls for level editing and creating.
4. GameController contains all game field objects.
5. Shooting settings, BubblesShooter.cs. Use any debug only temporary. This greatly reduces FPS. Bubbles Position – transforms for shootbubble, nextbubble and instantiating. GetShootBubbleMode – two variants{ FromBoardAtStart, FromCurrentBoard} . FromCurrentBoard – used only shoot bubbles from board at current time.

5



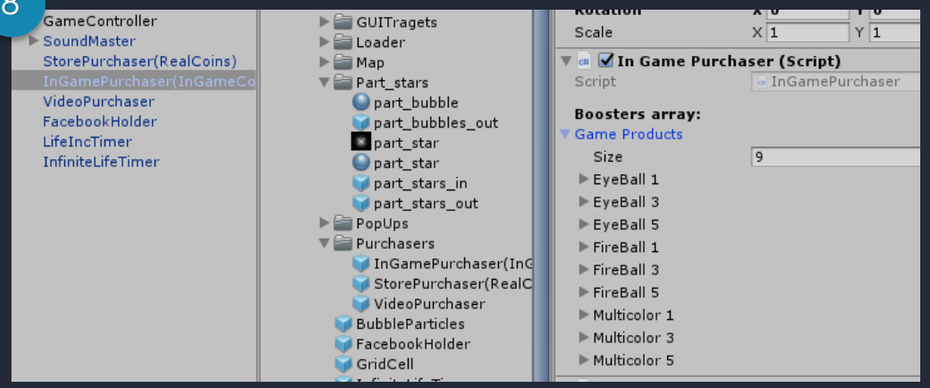
Game scene scripts and settings

6

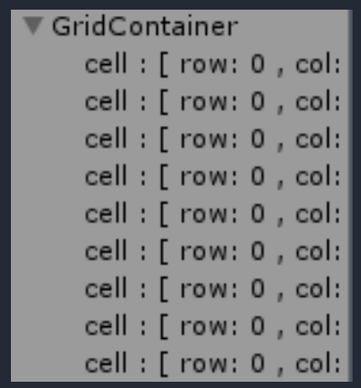


6. GameBoard contain references to scene objects. Buttons used only for debug purposes. Min Bubble score, Max Bubble score used for score calculating.
7. GridContainer contain at runtime all gridcells, and gridobjects.
8. InGamePurchaser (for game coins) - Booster shops.
9. Score Calc you can find in BubbleShooter.cs

8

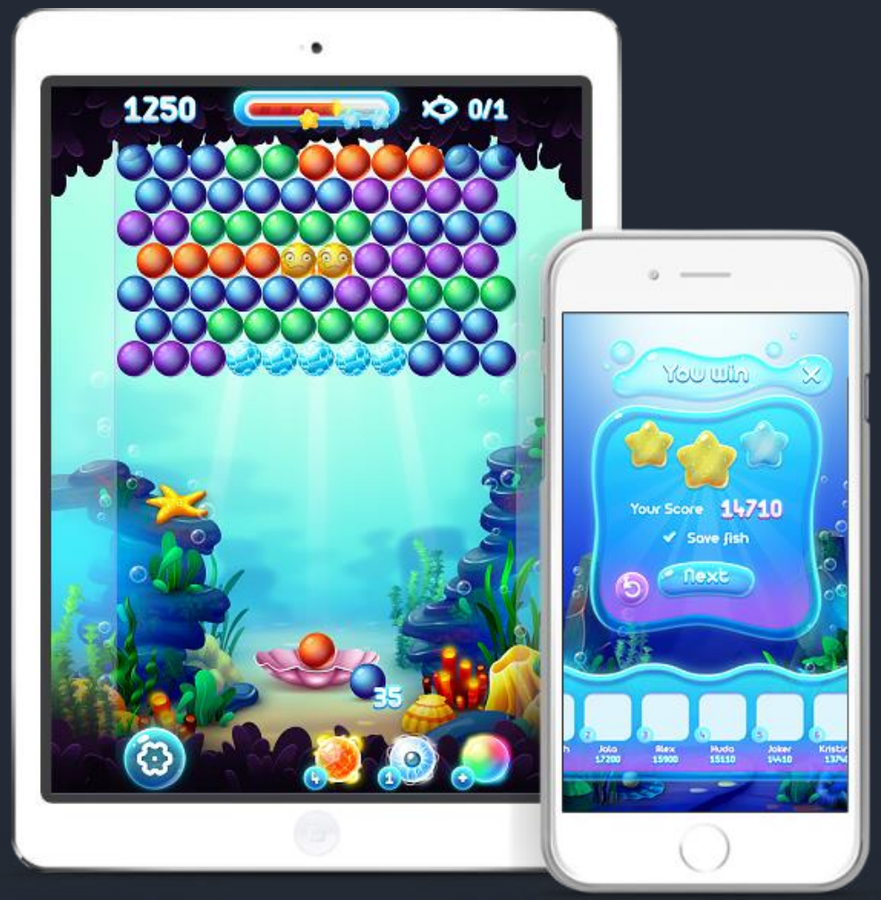


7



9

```
ivate int minBubbleScore;
private int maxBubbleScore;
public int GoodShoots { get; private set; }
public int BubbleScore { get { return Mathf.Min(GoodShoots, 8) * 5 + minBubbleScore; }}
private int AdditScore { get { return Mathf.Min(GoodShoots, 8) * 5 + 15; } }
public int ShootScore { get { return ShootAreaLength*(BubbleScore + AdditScore); } }
```



Thanks!

Master Key Company

Create you project for the moment!

Please rate our item, if you like it!

★★★★★

We are available for freelance work. If you are interested, please contact us individually.

Homepage : www.mkeystudio.com
Marketing Email: melanitta83@gmail.com
Tech Support: putchkov1975@gmail.com