Happy Burger Shop

Game Template

Compatible with **Unity v5.6+ & Unity 2017**
Supports Android, iOS, WebGL, Facebook, Windows and Mac

Dear Customer,

Thank you so much for purchasing this starter kit. 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.

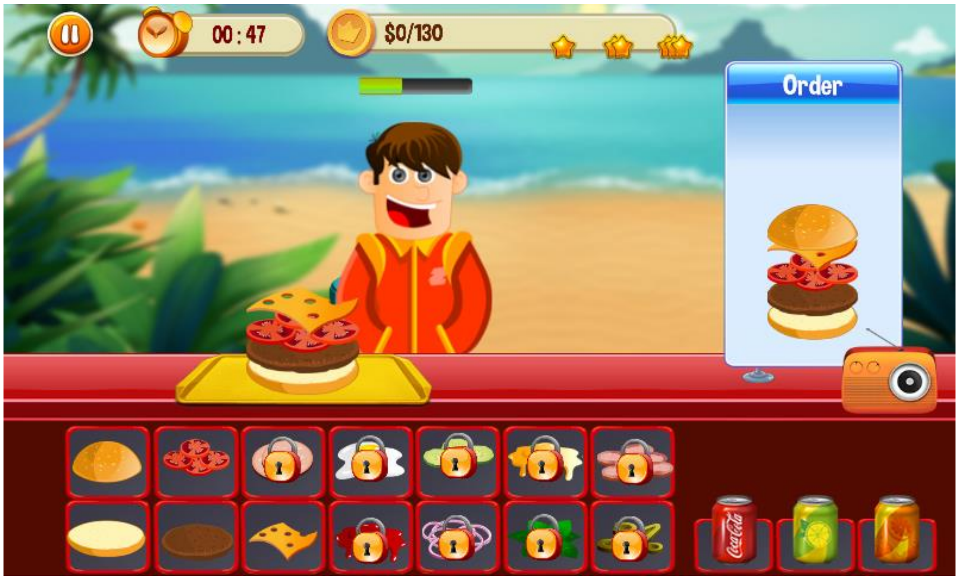
Overview

Happy Burger Shop, is a professionally made and easy to use time-management game template that can help you build a burger cooking game at no time. This game template can also be used as a flexible base for further development and experiments. It has all nuts and bolts to build a fully functional burger cooking game. It came with:

- **8** premade customers (with different textures and patience level)
- **14** predefined ingredients.
- **3** predefined drinks.
- Realtime order generation algorithm for customers!
- Full game flow with Menu, Shop, Share, Rate, Pause, Level Selection and Game scenes.

However, you are not limited by these numbers and can easily define and add more items to the game to deliver a rich game play experience to your players.

The game accepts both touch and mouse inputs, and can be tested on **Android**, **iOS**, **WebGL** and **StandAlone** platforms.



Classes

All these classes used inside the game template are well commented and you can easily extend them as you wish. Here you can take a quick look at the important classes:

- **MainGameController:** This class is the main controller of the game and manages customer creation, time management, money management, game state , win/lose state, showing information on UI, and showing/hiding/clearing orders in realtime.

- **CustomerController:** This class manages all thing related to a customer, including the entrance, order, patience, settle and animations. Basically this is how a customer object works:
  - 1. This instance has been created by MainGameController class.
  - 2. Customers make his order by creating a burger with ingredients available for this level (refer to CareerLevelSetup class)
  - 3. Customer enters the scene, and the order is shown on the board.
  - 4. Customer carefully inspect all your taps on ingredients untill you serve the right burger or ran out of time.
  - 5. If the order is served, customer will pay you with coin, and leaves with a happy face.
  - 6. If you ran out of time, customer gets angry and leaves the shop with no pay.

- **IngredientsController:** Main class for Handling all things related to ingredients. when we click on an ingredient, it should match the exact id of the product the current customer wants. So we first need to check if the player is tapping on the right ingredient. If tap is correct, we will add the ingredient to the delivery queue. if not, we delete the current delivery, and player needs to start from scratch

- **PlateController:** When we click on any ingredients, the game checks if the ID of selected ingredient matches the customer order. If this is the case, we need to add this ingredient to the server plate object to simulate the shape of customers order. The fact is that we are not doing anything special. We are just instantiating static ingredients based on the master queue array from the MainGameController.

- **CareerLevelSetup:** Use this class to set different missions for each level. when you click/tap on any level button, these values automatically gets saved inside playerPrefs and then gets read when the game starts.

- **CareerMapManager:** This will load the game scene with parameters set by you for the selected level. It will saves those values inside playerPrefs and they will be fetched and applied in the game scene.

- **customerMoneyController:** When customer receives the order, it leaves the money on the table. We use this controller on the money object to let it be collectable by player.

How to add new Ingredients, Drinks, Customers & Levels?

To add a new *customer* to the game:

1. Select a customer prefab from folder “Prefabs/Customers” and duplicate it with Ctrl+D;
2. Rename the new customer prefab to something appropriate (Like Customer-09).
3. In the inspector, set the desired **customerPatience** (in seconds).
4. Define new material and texture for this new customer and set them via prefab’s inspector. First set the main (happy) material (for Renderer component) and then set the 4 available materials in the CustomerMoods array.
5. Go to GameController object in the scene and update its **customers** Array by adding your newly created customer prefabs to the end of array.
6. Repeat the above steps to add as many customers as you like.
7. You are done!

To add a new *ingredient*:

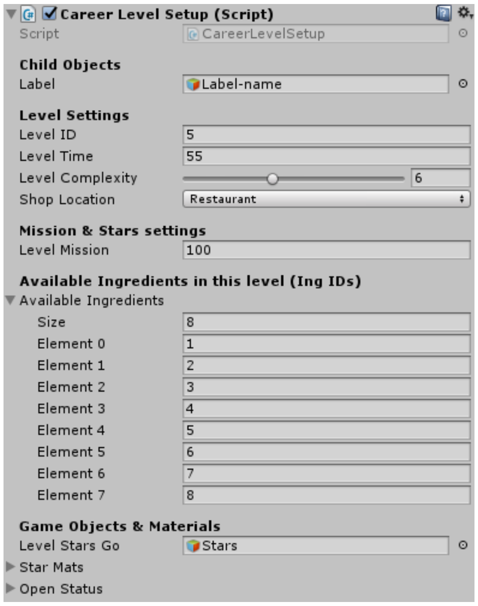
1. Pick an ingredient prefab from “Prefabs/Ingredients” and duplicate it with Ctrl+D;
2. Rename the new ingredient prefab to something appropriate.
3. Define new material and texture for this new ingredient and set them via prefab’s inspector.
4. Set a price for this ingredient.
5. Set an ID for this ingredients. Notice that we are using IDs from 1 to 14 for already available ingredients. So you need to start from 15 and above.
6. We are done with the new Prefab.
7. Open Game scene, go to Hierarchy and select **IngredientsHolder** game object. It contains 14 ingredients already.
8. Duplicate one of the available **Ingredient-xx** child objects and position it properly in the editor.
9. Set the newly created **ingredient-xx** child object’s Factory ID to an incremented size (15, 16, 17 , … and so on). This should match the ID you set for the prefab in step #5.
10. Select GameController and add the material of this new ingredient at the bottom of AvailableIngredients array.
11. Select serverPlate object and add the new ingredient prefab at the bottom of AvailableIngredients array.
12. Select all customer prefabs inside "Prefabs->Customers" and add the new ingredient prefab at the bottom of AvailableIngredients array.
13. You are done!

To add a new *Drink* item:

1. Define new material and texture for the new drink item and set them via prefab's inspector.
2. Go to Hierarchy and select **DrinksHolder** game object. It contains 3 drinks already.
3. Duplicate one of the available **Drink-xx** child objects and position it properly in the editor.
4. Set the newly created **Drink -xx** child object's DrinkID to an incremented size (104, 105, 106 , … and so on).
5. Select all customer prefabs in "prefabs/Customers" and update their **availableDrinks** int parameter with total number of available drinks.
6. Select **GameController** and add the material of the newly created drink to the bottom of **AvailableDrinks** array.
7. You are done!

To add a new Level to Level-Selection scene (extending the game):

1. Open LevelSelection scene, then select and clone one "Level-xx" button from the available buttons in hierarchy.
2. Rename this cloned button to something appropriate like "Level-11" as we already have 10 buttons inside the game.
3. Select the button and configure it's class properties (Career-Level-Setup) via inspector:
    a. **LevelID**: index of the level we want to load. Must be unique.
    b. **LevelTime**: time given to player to beat the level mission. (in seconds)
    c. **LevelComplexity**: indicates the complexity of the orders by customers in this level. this indicated the maximum number of ingredients an order can hold.
    d. **ShopLocation**: You can choose a location (background) for this level.
    e. **LevelMission**: beating default levelMission grants 1 star. for 2 stars, player needs to reach 25% more of the initial levelMission. for 3 stars, player needs to reach 50% more of the initial levelMission. You just need to set mission 1. Missions 2 & 3 will be calcualted automatically.
    f. **AvailableIngredients**: Ingredient IDs 1 & 2 are used for the bottom & top bread, and should always be available if you want your final burgers to look good. The ingredient IDs that are not added to this list will be locked in the game.



☑ Career Level Setup (Script)

Script: CareerLevelSetup

Child Objects

Label: Label-name

Level Settings

Level ID: 5

Level Time: 55

Level Complexity: 6

Shop Location: Restaurant

Mission & Stars settings

Level Mission: 100

Available Ingredients in this level (Ing IDs)

Available Ingredients

Size	8
Element 0	1
Element 1	2
Element 2	3
Element 3	4
Element 4	5
Element 5	6
Element 6	7
Element 7	8

Game Objects & Materials

Level Stars Go: Stars

Star Mats

Open Status

Setup Unity Ads (Video/Rewarded)

Unity ads is already integrated into the game template. All you need to do to make them show inside the game is to download the newest Unity Ads sdk from this link:

https://assetstore.unity.com/packages/add-ons/services/unity-ads-66123

Import the downloaded asset into the project and Unity video ads will be available inside "Menu" and "Game" scenes.

Final Word

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

Fruit Juice Maker Game Kit
Unity Assetstore

Master Archer Game Kit
Unity Assetstore

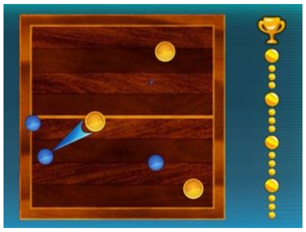
Space Frontier Game Kit
Unity Assetstore

Rock, Paper, Scissor Game
Unity Assetstore

Frenzy Farming Game Kit
Unity Assetstore

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