Fast Food Rush – Hyper Casual Restaurant Management

Table of Contents

1. Introduction

Thank you for purchasing Fast Food Rush! We appreciate your support and are excited to provide you with a complete hyper-casual restaurant management game template. This documentation will guide you through the setup, usage, and customization of the template to help you get the most out of this package.

2. Getting Started

2.1 Package Contents

The package contains three restaurant scenes in “Scenes” folder, each showcasing different setups based on price configurations, restaurant layouts and the types of food they sell:

- Restaurant01: Burger
- Restaurant02: Coffee
- Restaurant03: Donut

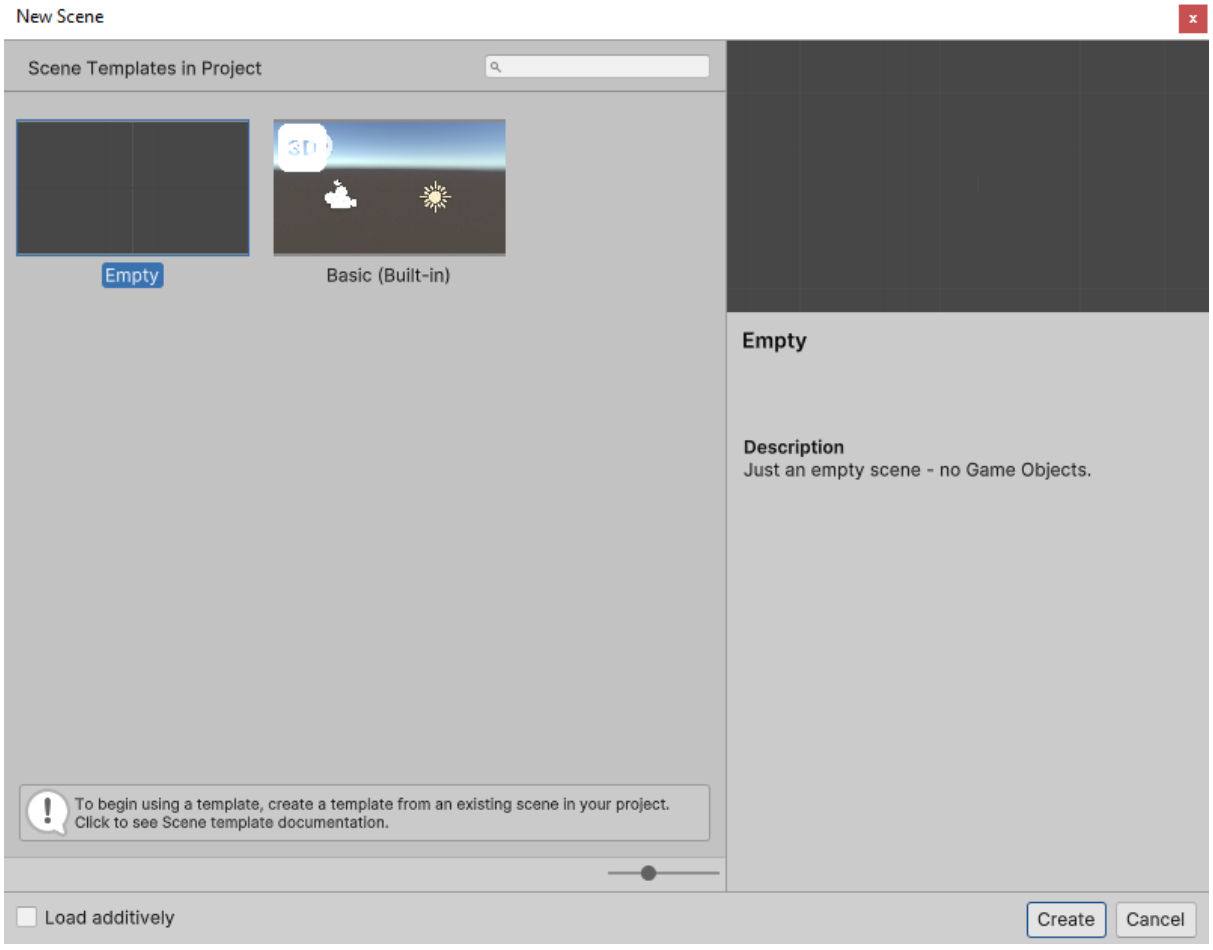
You can open these scenes in Unity to explore how the game template works. Inspect the layouts, game objects, and scripts to understand the structure and functionality. This will give you a good starting point for creating your own restaurant scenes.

2.2 Creating New Restaurant

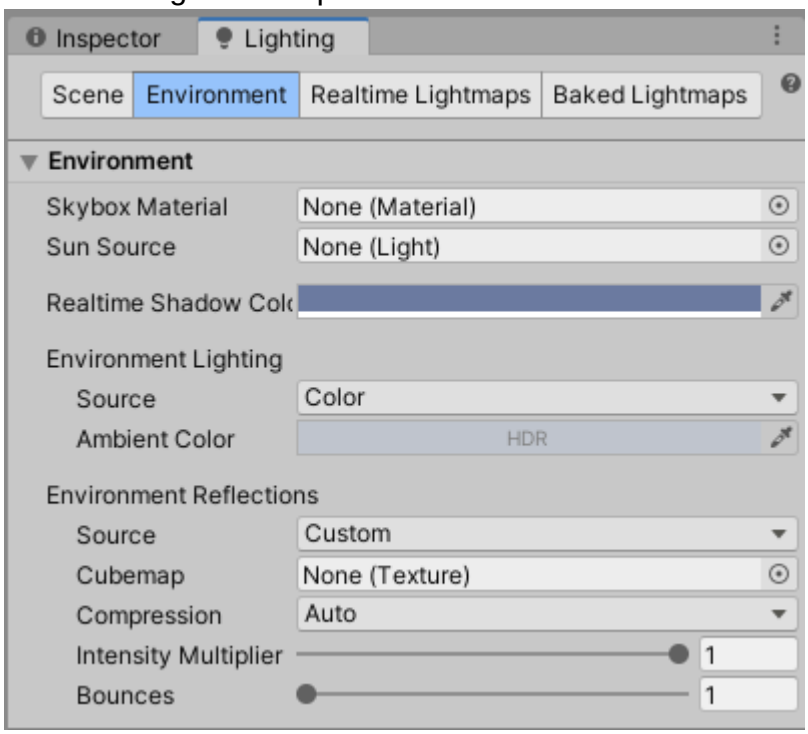
Follow these steps to create a new restaurant scene:

2.2.a Create a New Scene

- In Unity, go to File > New Scene > Empty > Create to create a new blank scene.

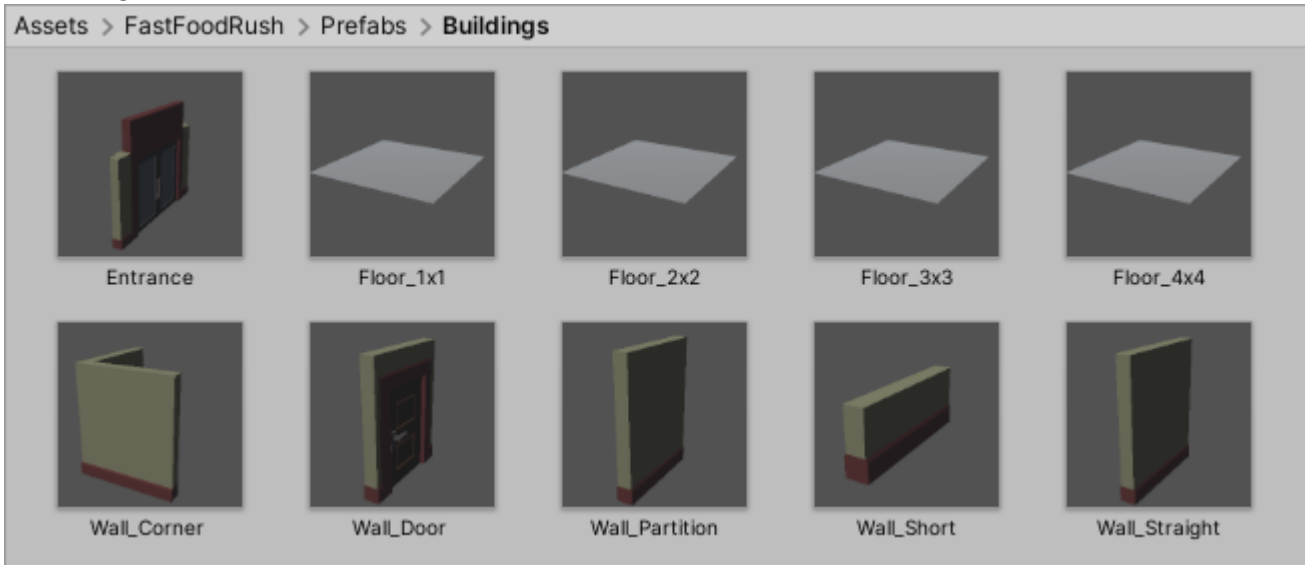


- **IMPORTANT:** Save the new scene with a name following the format **Restaurant0x** (e.g., Restaurant01, Restaurant02). This naming convention will be used later to facilitate switching between scenes (restaurants) and handling the saving and loading of restaurant data.
- Go to Window > Rendering > Lighting to set up the lighting for your newly created scene. Here is my suggested setup: Set Skybox Material to None, Environment Lighting Source to Color, set Ambient Color to your liking, and finally set Environment Reflections to Custom while leaving Cubemap to None.

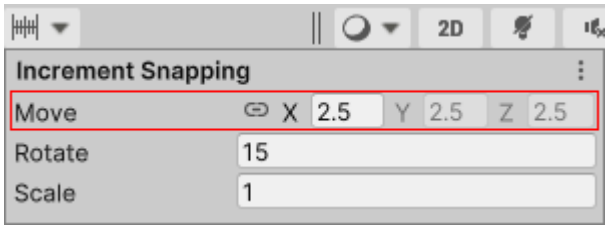


2.2.b Set Up the Restaurant Layout

- You can utilize the existing prefabs provided in the package located in: Assets/FastFoodRush/Prefabs/Buildings, to construct your own restaurant. They are designed in a modular way. Please refer to the sample restaurant scene for an idea of how to arrange them.



- Set the increment step for movement to 2.5 units, because these building prefabs were designed with this spacing in mind to ensure proper alignment and functionality.



- The materials for these prefabs are located in Assets/FastFoodRush/Models/Materials/Buildings.
- For better organization and easier management, I suggest parenting the building prefabs to an empty game object called something like “Building.”

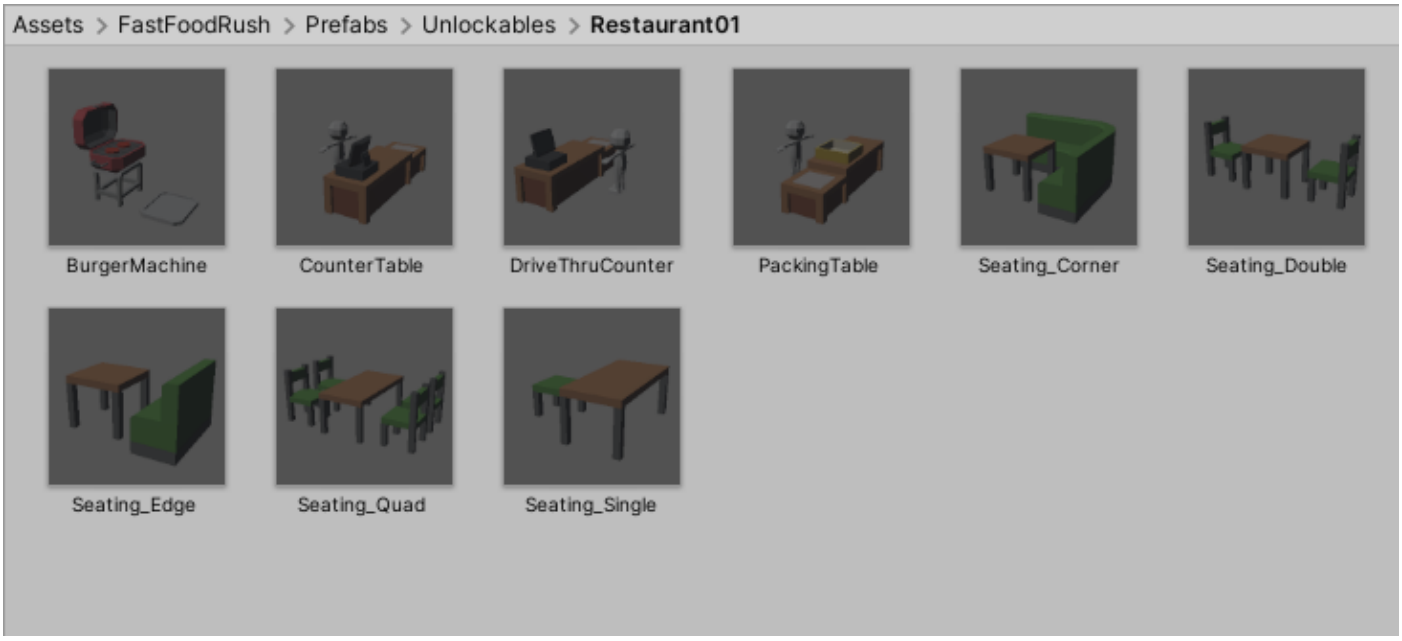
2.2.c Adding Core Game Objects

Drag and drop all these 6 prefabs located in Assets/FastFoodRush/Prefabs/Core to the scene:

- **AudioManager:** This will handle all background musics and sound effects for the game.
- **DirectionalLight:** The main and only light source in the game.
- **MainCamera:** This camera will follow the player smoothly and is configured to provide a perfect isometric view.
- **Player:** This character is under your control throughout the game.
- **PoolManager:** Objects such as money, food, trash, and packages, which are frequently used throughout the game, are pooled to minimize the impact of garbage collection on performance.
- **RestaurantManager:** This is the main component of your restaurant. It handles prices, UI, effects, unlockables and many other gameplay mechanics.

At this point, you can already test the scene and walk around your new restaurant. But of course, your restaurant is still empty, it's essential to add “objects” to your restaurant.

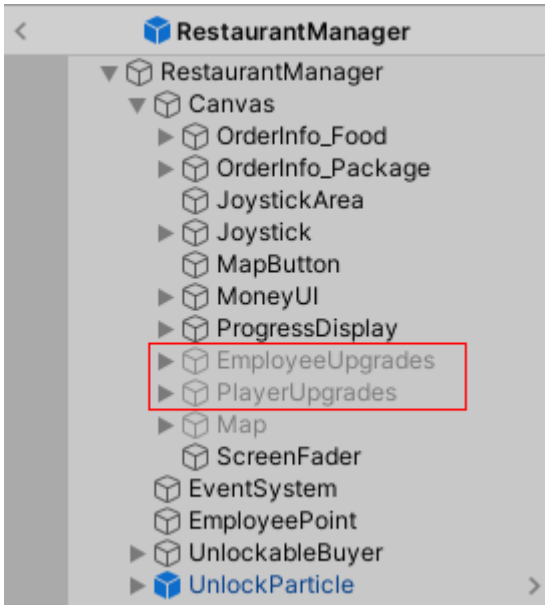
For now, please go to Assets/FastFoodRush/Prefabs/Unlockables/Restaurant01 and place objects in there to the scene:



- **Seating:** There are various types of seating available, including single, double, quad, edge, and corner. They all serve the same purpose: providing customers with a place to sit and enjoy their food.
- **Counter Table:** This station serves as a convenient location for customers to place their orders and for employees or player to prepare food items before serving.
- **Food Machine:** This essential equipment automates the process of cooking and preparing various food items.
- **Drive-Thru Counter:** The drive-thru counter allows customers (cars) to place and receive their orders without leaving their vehicles.
- **Packing Table:** Here, orders are packaged, and prepared for drive-thru customers.

In addition to those objects, we also require the following objects added to the scene:

- **Trash Bin:** After enjoying their meal, customers leave waste on the table. Employees or players need to dispose of the trash in the trash bin before the seating can be used again. There is made prefab located in Assets/FastFoodRush/Prefabs.
- **HR & GM Office:** Various upgrades may be purchased there. There are made prefabs located in Assets/FastFoodRush/Prefabs/Unlockables. **IMPORTANT**: These office objects have a child object with an "Activator" component. When a player enters the office, this component should activate an upgrade user interface. The HR (Human Resources) activator activates “Employee Upgrades,” while the GM (General Manager) activator activates “Player Upgrades.” These upgrade user interfaces are children of RestaurantManager > Canvas. You should link these interfaces to their corresponding activators.



Again, for better organization and easy management, I suggest parenting these objects to an empty game object called “Objects” or something like that.

2.2.d Finishing Restaurant Setup

- **Baking NavMesh:** After laying out the restaurant building and placing the necessary objects, we need to bake a NavMesh for customers and employees to walk on. To do this, navigate to Window > AI > Navigation. Next, switch to the Bake tab and click on the Bake button. Notice that all building prefabs in the package are already marked as Navigation Static, this is required for NavMesh baking to work.

- **Customer Spawn / Despawn:** You may want to position the spawn and despawn points (part of Counter Table) of customers (NavMesh Agent) outside the restaurant to prevent them from appearing and disappearing magically. To accomplish this, create a plane outside the restaurant and rebake the NavMesh. Once baked, you can disable the mesh renderer.
- **Assigning Unlockables:** Assign all "Unlockable" objects to the Restaurant Manager so that players can purchase or upgrade them. You can assign these unlockables multiple times; the first time a player purchases them, they will unlock the object, and subsequent purchases will upgrade them. Currently, unlockable objects inside "**Restaurant0x**" folder have three stages of unlocks, each with its own unique visual and statistical changes. The order in which you assign these unlockables also matters in determining the price automatically. Please refer to the existing restaurant scenes to learn how to assign these unlockables in order to maximize the feeling of game flow mechanics and progression.
- **Add Restaurant to the Map:** Open the Restaurant Manager prefab, and locate the "Map" object within the Canvas. Each restaurant is represented as a Unity Button and is a child of a "Pin" object (Unity Image). You can easily duplicate one and change the icon. Don't forget to assign this new restaurant to the "Map Buttons" field in the Map component.
- **Add Restaurant to the Build Settings:** Lastly, open File > Build Settings > Add Open Scenes, to include your new restaurant to the build.

2.2.e Customize Gameplay Elements

- **Modify Prices:** In Restaurant Manager, you can set the prices for restaurant upgrades and unlocks. These prices are determined by a base value and a factor, so adjust these settings to achieve your desired prices. Each upgrade (employee and player) and unlock price will rise exponentially based on your input values.
- **Modify the Restaurant's Food:** To change the type of food served in a restaurant, you must change the object assigned in the Pool Manager in the restaurant scene.
  IMPORTANT: The pool objects' names must not be changed, as these names are used in the pooling system (Money, Food, Trash, Package). For example, if you want to change a burger to a coffee, you can create a prefab of coffee from an imported model and rename it to Food.
- **Modify Stack Settings:** After changing the visual of the food, you may also need to adjust the "Stack Offset" parameters within Restaurant Manager. For example a cup of coffee is higher than a burger, so you need to increase the "Food Offset" parameter slightly.
- **Modify Seating Visual:** Each seating consists of table and chair Mesh Renderers. These renderers have an "Upgradeable Mesh" component, which handles their corresponding mesh upon upgrade as listed in the "Upgrade Mesh" field. The first element is the mesh data representing the first upgrade after unlock (level 2), and so on.

2.2.f Test Your Scene

- Playtest your new restaurant scene to ensure everything works as expected.
- You can add 10k money by pressing money icon on the screen (please remove on build).
- Make adjustments as needed to improve gameplay and user experience.
- By following these steps, you can create and customize your own restaurant scenes, adding new and unique gameplay experiences to your Fast Food Rush game.

3. Using Third-Party Assets

This package includes the following third-party assets:

3.1 DoTween

- Used for animations and transitions.
- Refer to the Third-Party Notices.txt file for license details.

3.2 SimpleInput

- Used for handling user inputs.
- Refer to the Third-Party Notices.txt file for license details.

4. Contact Information

For further assistance, please contact:

- Email: cryingsnowstudios@gmail.com
- Website: crying-snow.github.io