FRENZY FARMING

Flexible Time-Management Strategy Game Kit

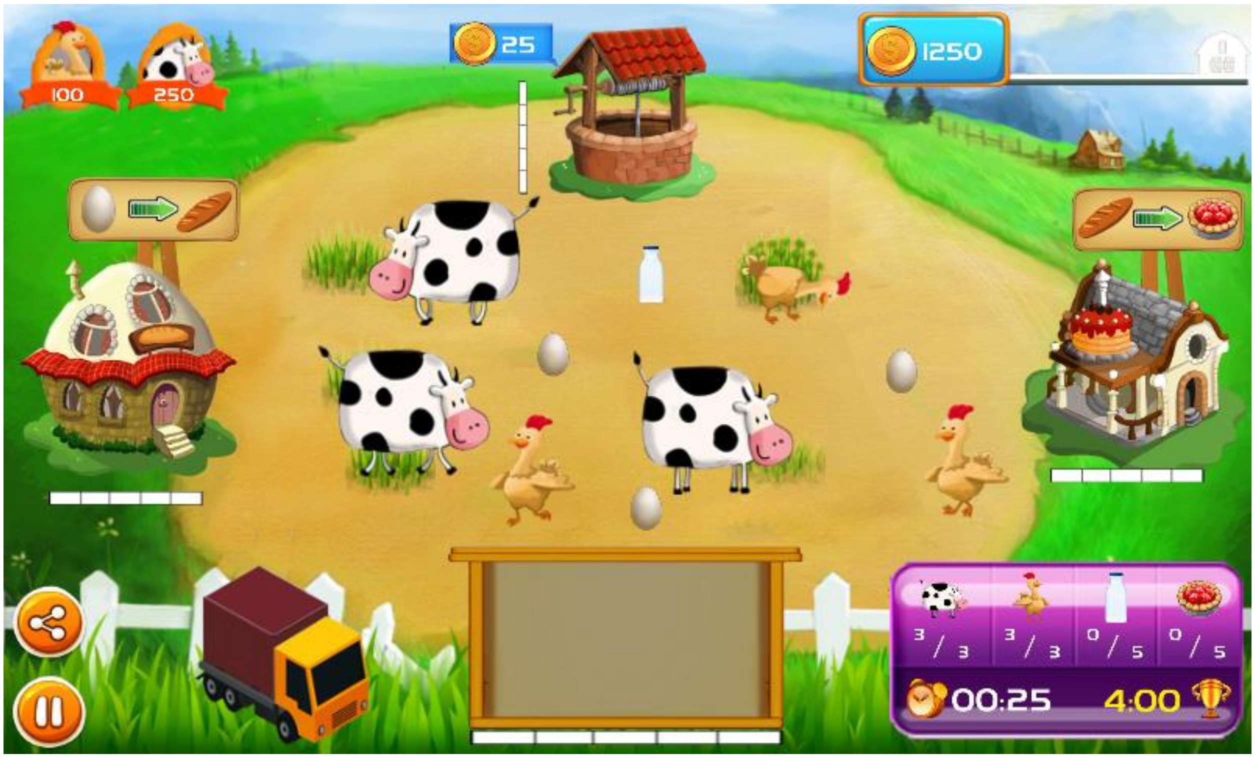
Supports Android, iOS, Windows and Mac | Supports both 2D and 3D structures

Dear Customer,

Thank you so much for purchasing this game kit. Here you can find detailed information on how to use this product with maximum proficiency. All scripts and code assets are fully commented, but if you ever needed a hand on a segment of the codes or anything else, feel free to contact us at http://www.finalbossgame.com . We'll try our best to support you with your questions as soon as possible and till you feel most satisfied.

Overview

Have you ever wondered what it would be like to run your own farm? Looking after chickens, cows and picking their products such as eggs, milk, etc… Did you ever wanted to manage your own factories and mix different items to make new ones? Well, **Frenzy farming** is the answer to all your needs. This is a unique time-management strategy game kit which allows you to design and play hundreds of different levels with different missions, themes, objectives and properties.



This project works flawlessly with both touch and mouse inputs, and thus, can be tested on **Android**, **iOS** and **Stand-Alone** platforms simultaneously. The kit also works on both Unity3d Free and Pro, and again to your benefit, needs no 3rd party plug-in or add-on to work.

It just works ***right out of the box***. You do not have to do anything to make it work. Just load the project inside Unity and hit play.

Hint:

- When you load the kit for the first time, always add all scenes to the "Scenes in build" list via ***file->BuildSettings***, to make sure you experience a smooth transition and game flow.
- The game requires these Tags to work:
  - Food
  - Product
  - Hunger
  - Well
  - levelSelectionItem
  - Pet
  - Chicken
  - Cow
  - AdManager
  - HungerArrow

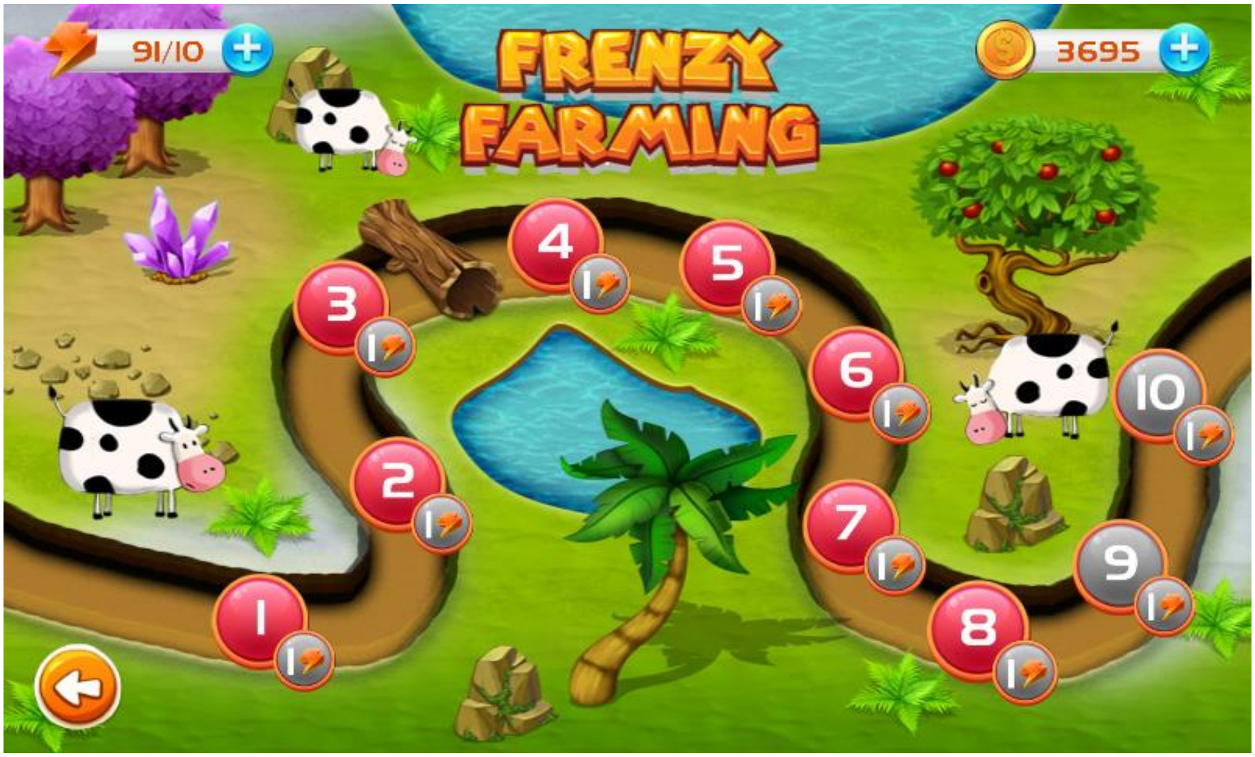
  Make sure to maintain them in your personalized project or replace them with proper equivalent.

Game Play

Frenzy farming game kit uses a famous game play system. In this game, you start each level with certain starting money and time limits. You must buy different pets and feed them with grass (as food) in order to receive their products which is egg and milk.

Then you have access to some factories that receive these raw products and transfer them into something else which can be sold at a very higher price.

You also have access to a hunger which stores all your collected items, and you can use a truck to take these items to the city and sell them for you. Truck will return to your farm with your money and you can use this money to buy more foods, pets, factories, etc…



In each level, you have to pass all the missions in a certain time. If you beat all the missions in the gold time limit, you also receive an additional prize which can be used later to unlock certain items in the game. Beating the mission in the normal time doesn't include an extra prize.

In the beginning of each level, you are instructed about the missions you need to pass. You always have access to level missions inside the game.

If you ran out of money, you cannot buy food. And without food, your pets will starve to death. You have to take good care of your pets, or you might not be able to pass the missions.

Game-Play Mechanics/Settings

In this game kit, you have the absolute freedom to design unlimited number of levels, each with their own settings and attributes. You can easily assign up to 4 missions for each level (from a dropdown list of all available mission types), you can set the time limits, starting money, available pets, available factories, and many other settings that helps you create a unique game play experience for your players.

Playing each level requires one unit of energy which can be obtained as daily/hourly prize and is also obtainable from within the game shop (IAP). This way you can boost your player's engagement as they will return to the game several times a day to receive their free hourly prize.

We have integrated AdMob sdk into this game kit, so you can show Banner Ads, Interstitial Ads and Rewarded Video Ads inside the game. We have also implemented a few simple routines to show banner ads when the game starts and we call a cached full screen ad when starting/pausing inside a level.

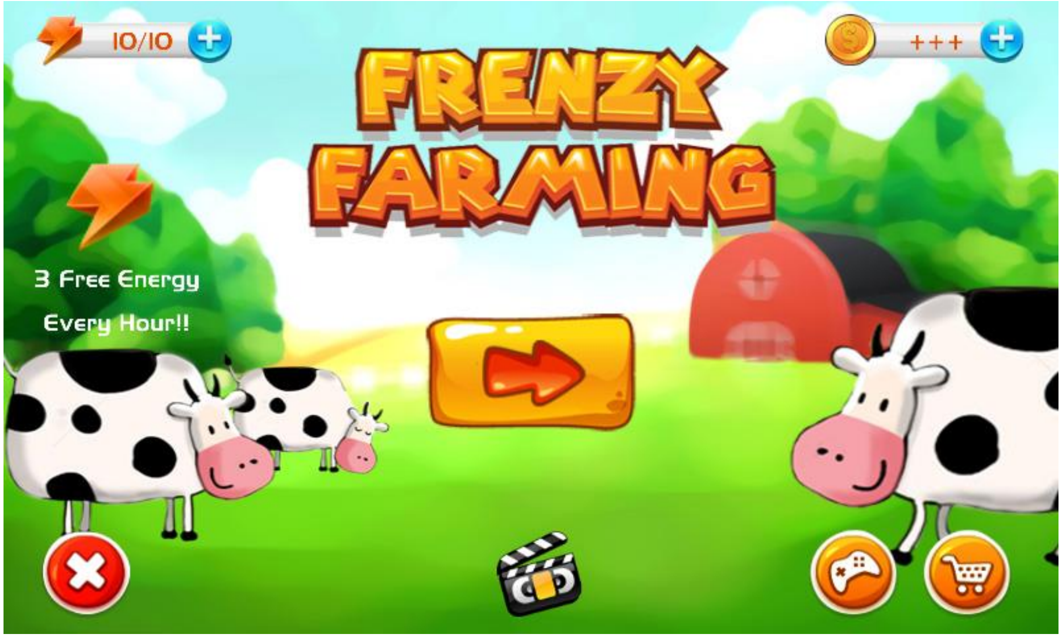
All you have to do is to open the Init scene, find the adManager object and set your own Admob IDs in the respective fields. You will have a fully working ad monetization strategy in no time.

You are free to use the AdManager functions anywhere you want to show more ads to the player. You just need to keep the balance between gameplay/advertisement to provide the best solution for your audience.

Social options

We have implemented a few social options like "***Sharing Screenshot***" on social media or a nice "***Rate-Us***" button on the menu scene to help you have a better control on customer retention.

You can place an instance of "Share screenshot" button anywhere in your game. This button takes a screenshot of the current scene and calls the sharing intent and lets you choose the target app to share the joy.



Introduction to Scripts, Classes and GameObjects

All scripts in this kit are fully commented and have full description about what they do. But here we take a closer look at some more important classes used in the kit.

GameController:
This is the most important class of the game. You can use this class to assign up to 4 missions to each level, configure the time, starting money, prizes, available pets, available factories, etc… to a level. This class also monitor game status every frame and handle different events live win or lose.

FactoryController:
A factory is a gameobject which takes a product as the input, performs some processing, and delivers another product as the output. We already have 2 factories that can be used inside the game. (We will add more in the next updates). Each factory needs some specific properties in order to work. You can set the factory type, number of required input items, the time it take to process the input item, the prefab of input product and also the prefab of the output product. Once they are all set, your factory is ready to work.

PauseManager:
This class manages pause and un-pause states. Please note that the pause scene is the best place to show your full screen ads. In this project, we are calling a full sized ad when user hits the pause button. We are also using pause function at the beginning of each level to stop the game and show the level missions to the player. When player hits the start button, we start the game by un-pausing the game.

PetManager:
This is the main pet controller. This class can control unlimited number of pets. You just need to set the ID, product and some minor attributes of each pet to setup a brand new pet. This class controls all behaviors of a pet inside the game, like:

- moving
- looking for food
- feeling hungry
- moving to food
- eating
- dying

We have 2 pets for now, but we will add more pets in the next updates. It's also very easy to add new pets to the game. Once you have the textures, all you have to do is to duplicate an already existing pet prefab, set its ID, set its output product and other minor variables. Then you need to assign correct material & sound clips to the new prefab and in the end, you must set a new tag for this pet prefab.

ShopManager:
This is the main shop manager class. It gives you the tools and options required for selling virtual items (for real money via IAB) inside your game. You need to integrate your own IAB system into the project, and then use the IAP functions in the places that is indicated inside this class.

HungerController:
This is the main hunger controller class. Hunger stores all products that is picked up by the player and provide useful reports for other classes when requested. Hunger has a limited space and each product requires a certain amount of free space to be stored. If hunger ran out of free space, it lets the player know and player has to use the truck to sell stored items to free up hunger's space.

TruckManager:
Truck object is used to sell items collected in the hunger. It will loads all items at once, get them to city market and gives you the sell money after returning to your farm.

WellController:
This is the main controller for water well object. This object is used to buy water. Water can get used inside the game to plant grass and grass will be used as the main food for pets. If you ran out of food, you have to buy additional water. Your pets will die easily if you do not feed them with grass.

How to add new levels to the kit?

There is two way of creating new levels. The easy way is to clone one of the available game levels, and build your own level on top of it. The other way is to start from scratch and create everything on your own. In both cases, you must have all the required controllers in the scene. Here we use a step by step tutorial on how to add a new level to the kit, configure the objects, add some missions and use a setting to make it all work together.

To add a new level, follow these steps:

1. Go to AssetFolder->Scenes, select the “GameLevel-10” scene and hit Ctrl+D to duplicate it. It will automatically get renamed to GameLevel-11.
2. Open the GameLevel-11 scene. Select the GameController object in the hierarchy and open missions array of the GameController component.
3. Set the number of required missions in the **Size** field. Let’s try this with 4.
4. Select the first to forth mission type and required amount as below:
    a. Egg Required / 12
    b. Money Required / 600
    c. Chicken Required / 8
    d. Bread Required / 4
5. Set the starting money to 350.
6. Set the prize and gold prize to 220 and 175.
7. Set the level gold and level fail time to 210 and 330.
8. Make sure that Pet buttons (in the UI) and both factory objects are active in the scene.
9. Make sure that truck object is active in the scene.
10. Drag an instance of chicken pet prefab into the hierarchy. We want to start the game with one chicken.
11. Drag 3 instance of grass prefab into the hierarchy and reposition them inside the dirt area of the scene.
12. Hit play and check if everything is working as intended.
13. Add this new level to the list of scenes via file->buildSettings->ScenesInBuild.
14. Go to Map scene and add a new LevelSelectionButton to the map. Set its ID to 11 to reflect the integer number of the map you created in step 1.
15. Your new level is ready!

Re-skinning process

It's very easy to re-skin the whole project. All the images you see in the game are located inside the "Textures" folder. You can re-design all the elements you see in this folder and replace them with the new ones you've created. You can see the results almost instantly inside the Unity editor.

Are you looking for a new prototype, fast res-skinning or custom modifications? Do not hesitate to contact us. We are available for custom jobs and can offer complete solutions that meet all sorts of budgets.

Contact us

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

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]()