

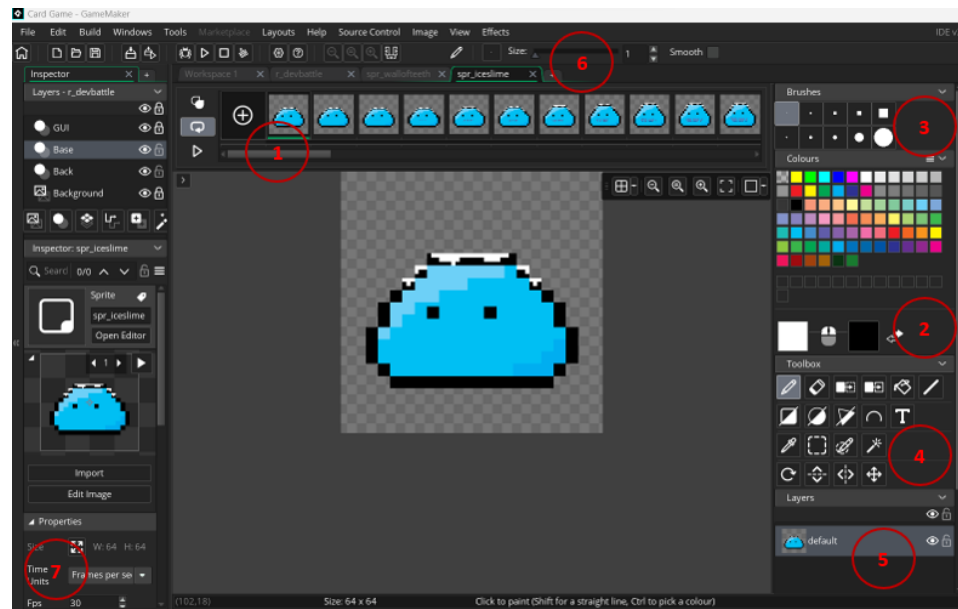
ART GENERATOR APP REPORT AND USER MANUAL

BY: CALEB WILLIAMS



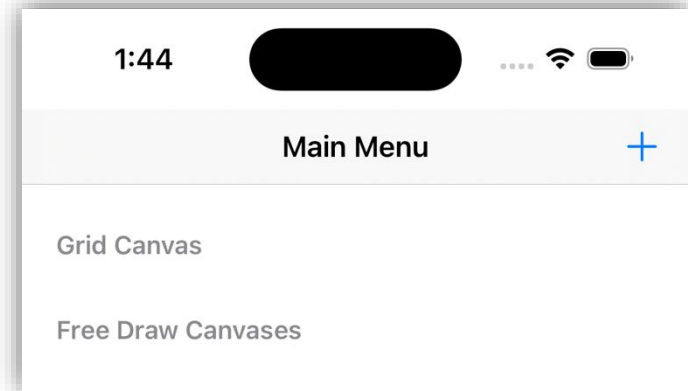
Overview

A Basic canvas app that mimics the basic structure of GameMaker. The image bellow represents the basic structure of GameMaker:



- 1) This is where you can insert multiple frames to create an animated sprite. There are alternate settings to change the frames per second and if the animation loops
- 2) A color palette and two currently active colors. The colors have an opacity slider as well. Color palettes can be imported from other sprites or presets
- 3) This changes the brush size and shape. You can create brushes out of preexisting sprites or chunks of sprites.
- 4) Many of the other tools that the editor uses. The only relevant ones here would likely be the pen, eraser, fill tool, and the text tool. Most of the other options are just quality of life improvements that would just clutter an iPhone display.
- 5) This introduces multiple layers to an image. When on a different layer, you can only edit any resources on that layer. This is incredibly useful when creating animations allowing the creator to create layers with appendages and other aspects of the sprites that move frequently.
- 6) This slider changes the size of the brush
- 7) Size of the canvas and frames per second editor

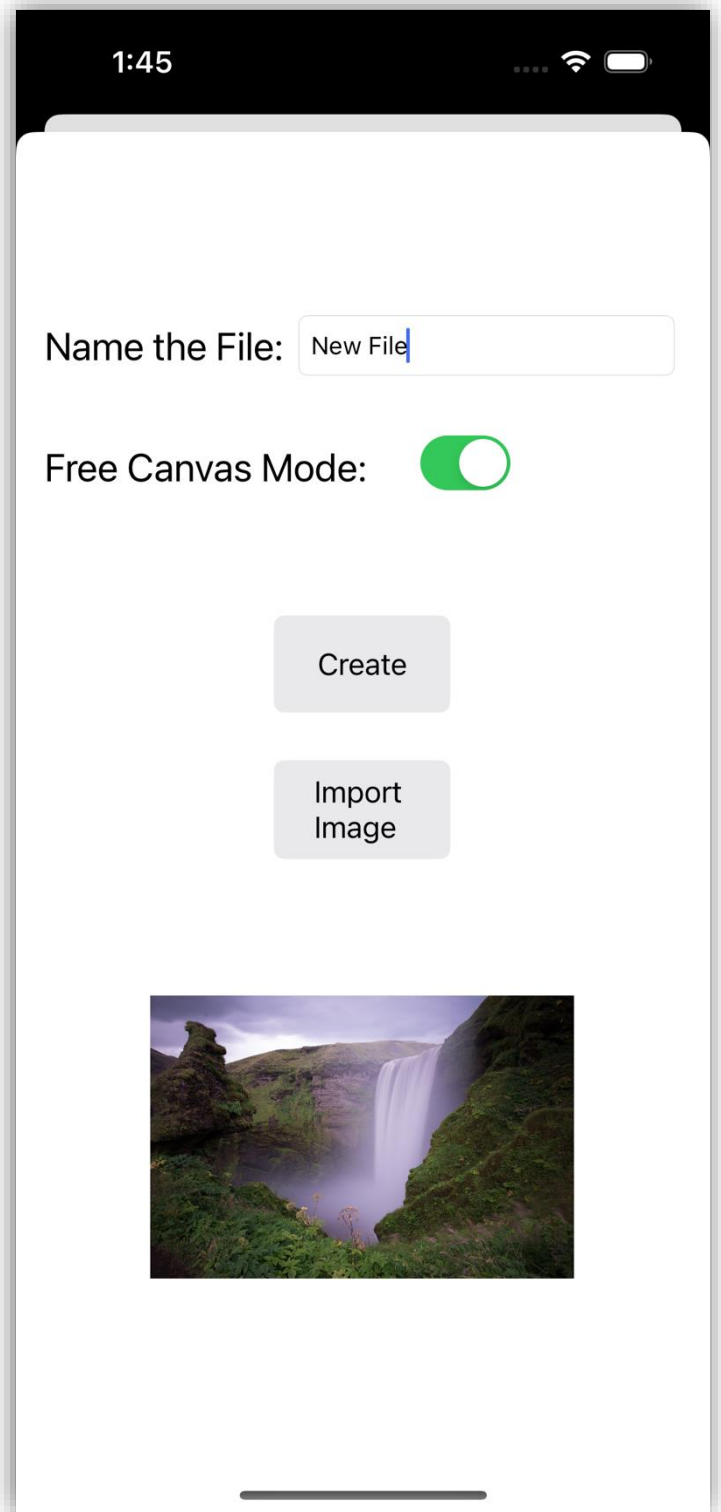
Walkthrough



I added a custom icon (that I made myself in a pixel art editor 😊). This is what the default home screen looks like and should contain two sections: grid canvas and free draw canvas. There should also be a button in the top right to add new files. When canvases are added, a preview of the canvas will display on the screen. Unfortunately, because of how the pictures are saved, the displayed picture is slightly cropped. This is because the pictures are saved by screeshoting the whole screen so when displaying the whole picture, you would see the UI bar.

Adding a New Canvas

When adding a new canvas, you have a couple of options. First, you have the choice of naming your file. Then you have an option of free canvas mode or grid canvas mode. Finally, you can import an image which will show up in a preview window.



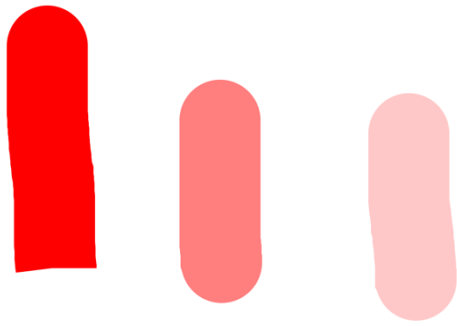
Free Draw Canvas View

You can draw freely which uses a Canvas view which I adapted from this online tutorial: <https://www.youtube.com/watch?v=kAiknPhkWmc>. There are multiple options such as an undo button which removes the last stroke. The reset button erases the saved image and all strokes on the canvas. The home button returns to the previous screen without saving and the save button does the same but saves the canvas. The color chooser allows the user to change the three chosen colors. The lower bars allow the opacity and the brush size to be altered.

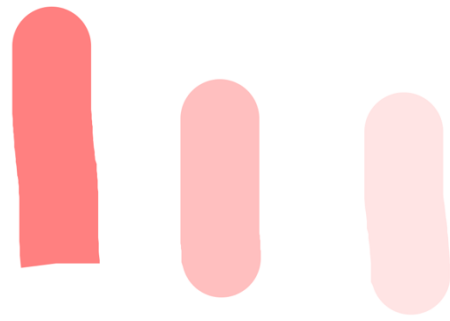
Issues and my Solutions

There was an issue with the saving of the images on this screen. This is because the canvas must have a background image and can not be set to clear. Because of this, it was impossible to save the background and represent this. My solution was to set the opacity of the canvas lower to still show the image if there is a saved image. THIS ONLY OCCURS IF THERE IS A SAVED IMAGE which will provide the desired functionality if there is no background image. There is no way around this problem without altering how the canvas is implemented and I would change my approach to this section if I could restart. Below I included a comparison between using the canvas with an blank canvas and one that has a saved image under it.





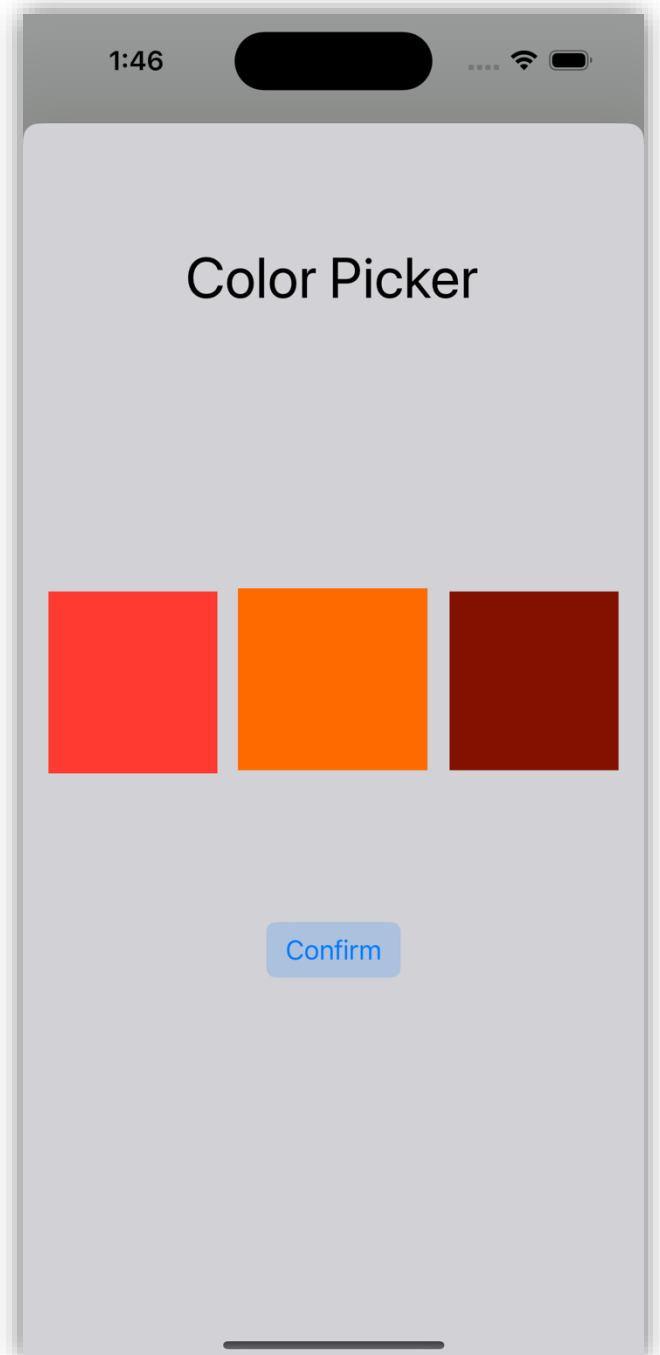
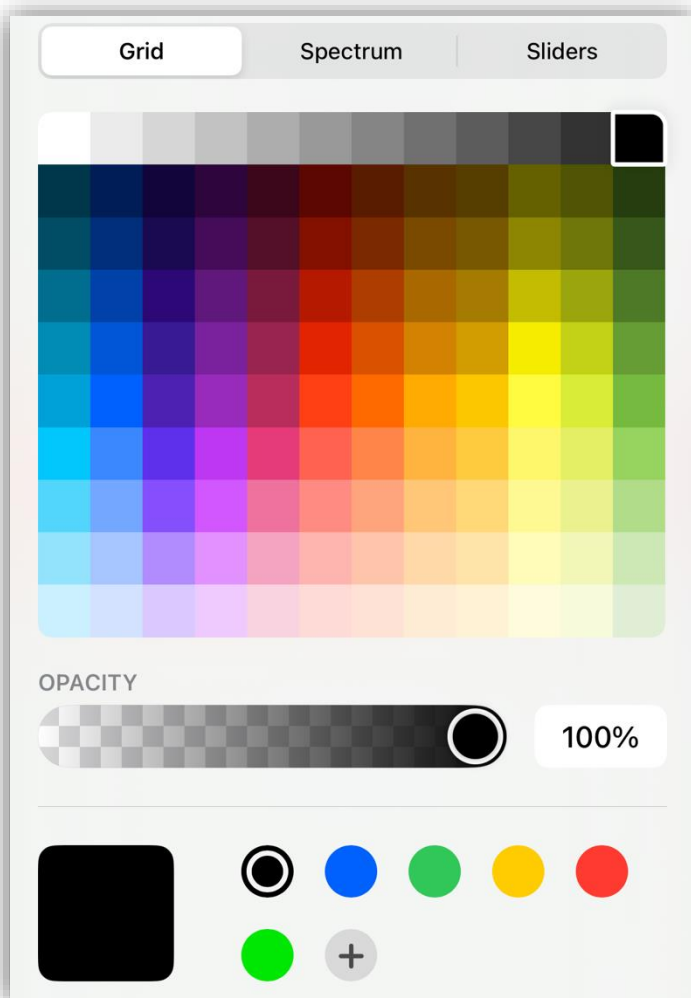
Normal Opacity (Default Blank Image)



Saved Image Opacity Issue

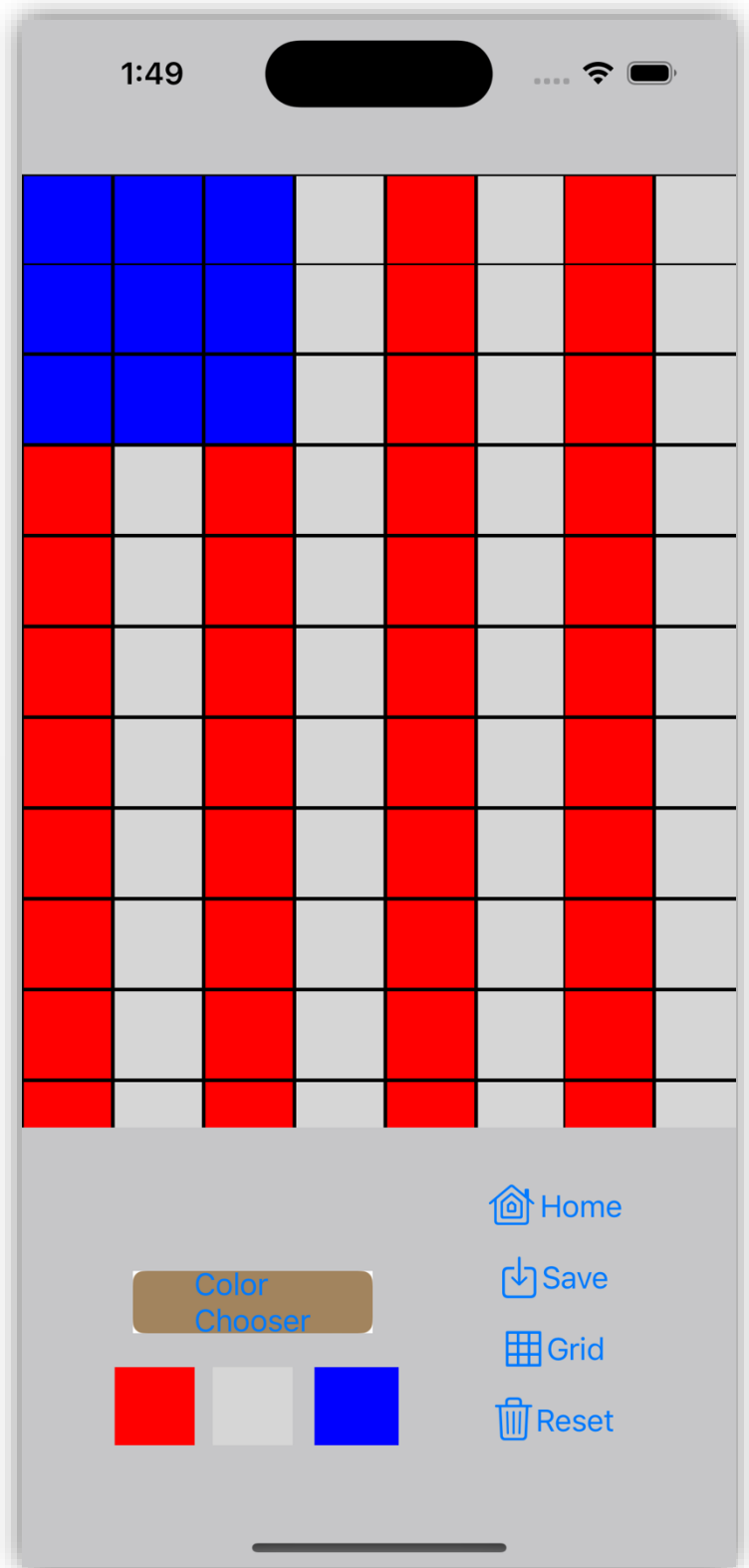
Color Picker Functionality

The view controller is passed an array that allows you to change the chosen three colors with the built in color changing functionality. This functionality is the same for the grid and free draw canvas.



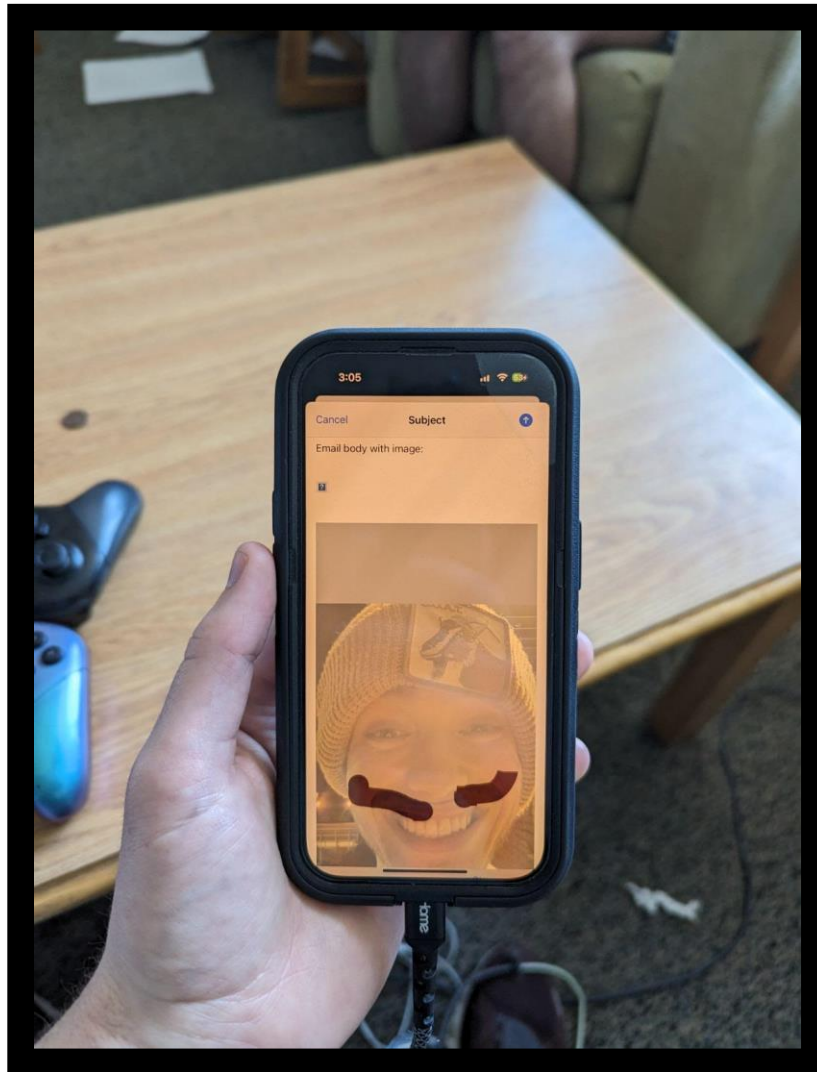
Grid Canvas View

The grid option allows for slightly different functionality than the free draw view. It offers a static sized grid that, when touched, will change the color based off of the selected color. This screen offers the same save, home, and reset functionality. The grid button is a replacement for the undo button (because of the ease of changing the tiles, the undo button was useless). The grid button alternates between three grid options: no grid, light grid, and heavy grid. They can be cycled between quickly for ease of access.



Export Functionality

When pressing the paper plane button in the main menu, it opens up an email with the image already attached. This was my solution to exporting the image. Be warned, THIS DOES NOT WORK IN THE SIMULATOR, the simulator does not have functionality for sending an email. Clicking the button will not crash the app in the simulator, it will do nothing.



Requirement Walkthrough

Before starting the app, I posted this list of requirements:

Must Have Features	Optional Features
Brush Size Slider	A Home Menu Allowing Multiple Sprites to be Saved or Edited
Opacity Slider	The Ability to Save Multiple Frames
Undo Button	Different Canvas Types (Pixel and Non-Pixel)
Reset Button	Zoom Functionality
Color Changing	Importing Color Pallets
Ability to Save or Export Sprites	Importing Pre-Existing Sprites
Different Brushes	
Advanced Color Selector	

Starting from the must have features, my program contains a brush size slider, opacity slider, undo button, reset button, color changing (with an advanced color chooser), and the ability to export and save sprites. Overall, I accomplished all of the conditions that I deemed necessary when setting out to start the assignment.

In regards to the optional features, I did implement a home screen but it does not contain the ability to edit the files. Also, there was not the ability to save multiple frames, import color pallets, or zoom. On the other hand, I did implement the functionality of two separate canvases and the ability to import pre-existing sprites. These two pieces of functionality where much more time consuming than expected and consumed most of my time.

User and System Requirements

User Requirements

- Creating a free draw canvas
- Creating a grid canvas
- Editing options
- Fluid color changer
- Ability to save art
- Ability to export art
- Organized menu

System Requirements

- Distinct canvases that both act as items in the menu list
- The ability to save/store canvases
- Functionality to import images
- Functionality to store a large color library
- A menu image display

Overall Thoughts

There are multiple areas where my app could have improved. If I would have had more time, I would have made my app look neater and more organized. The app looks pretty bland and colorless from the outside in. I also would have changed some of the fundamental foundational elements of the app that I implemented towards the start of the class. My poor early implementation when I barely understood Xcode resulted in many headaches and resulted in many different problems. I also would love to find a better fix to the opacity/saving problem featured in the free draw canvas even though I am content with the result I achieved. I'm happy with my app, but I do believe given another week to week and a half it would look and run more smoothly.