Family Repair App Documentation

*Family Repair Bachelor of Innovation Team from Fall 2022*

This document was created on December 1, 2022

The application was developed through the Unity Game Engine developed by Unity Technologies, version 2020.3.16f1  
This engine was chosen due to the teams’ familiarity with it, as well as its ease of deployment for Android and iOS devices.

The application was developed using a GitHub repository. GitHub is an Internet hosting service for software development and version control. This allowed the team to easily work on the project concurrently all from their own computers. This keeps a history of changes made to the project as well as allows other developers to have easy access to the files for further development.  
  
Within the Unity Project, the main files for the application can be found within the file structure highlighted in the image below.

A picture containing text

Description automatically generated

Saved changes to the prefabs in the prefabs folder will apply to all instances of that prefab within all scenes of the project. This allows developers to make 1 change to a layout for example, and have that change be consistent along the entire project.

As of the writing of this document, there are only two custom C# scripts. The *SceneLoadingButton* script is a simple script that allows developers to type in the name of a scene they want to load using Unity buttons through their *‘On Click ()’* events.  
Graphical user interface

Description automatically generated

The *VerticalScaleToScreenHeight* script is a simple script that will scale *Rect Transform* components of an object to a percentage of the screen height. This is a dynamic script that allows the application to scale correctly to any screen size. There are many fields that are visible in the inspector for developers to use if the developer would rather like to use another *Rect Transform* instead of screen height, or adjust for padding, etc.

A screenshot of a computer

Description automatically generated with medium confidence

All the other scaling is done through the *Rect Transform* scaling settings which can be set here.

Graphical user interface

Description automatically generated with medium confidence

The scaling of objects in the editing scene does not matter as all the scaling gets set up when the application runs. If all the scaling scripts have the *‘Constant Update’*  Boolean ticked you can test the scaling is working by resizing the screen in free aspect while the scene is running.

We suggest that you run the scene and make changes in the inspector to see how the app will look, record your changes, and then make your changes outside of play mode and save as this will allow developers to see exactly how the app will look while running. This also lets the developer scale the screen around to see how their changes look in all screen sizes.