GitHub Username: perfalcon

Eat Monster

Description

A simple 2d game for kids which keeps the kids engaged whenever they get bored.
The user has to tap on the item (ice cream/food item) which is placed in front of person's face, as they tap on the score will be increased and accordingly the coins will be increased. As the score increases, the levels will be increased and the food items will be unlocked.

Intended User

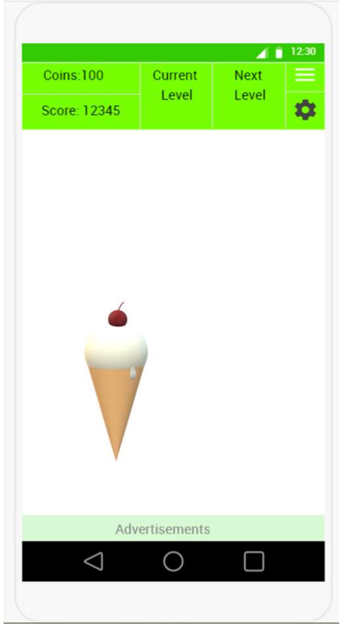
Kids of all ages

Features

- Encourages the kids to play
- Saves the score, coins
- Allows to change the skins (color schemes)
- Allows to buy extra coins through advertisements
- Displays advertisements at bottom on the main screen

User Interface Mocks

Screen 1 - Main Screen

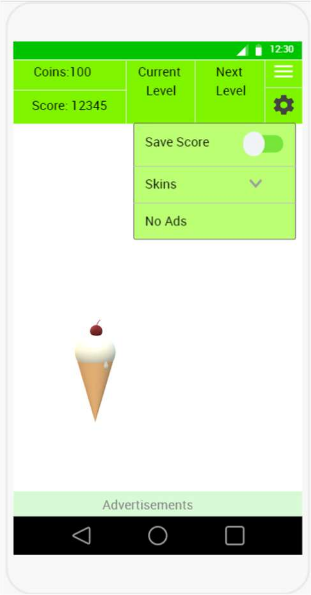


This is the first screen (main screen) where the game is active. This screen displays the coins and score in the first column, displays what current food item is being used in the second column, in the third column displays the next food item and in the fourth column displays the buttons for options and settings screens.

In the center of the screen is the actual game. It shows a food item and a funny(monster) face. Whenever a tap is done on the food, there will be a picture shown to indicate the part of the item is eaten.

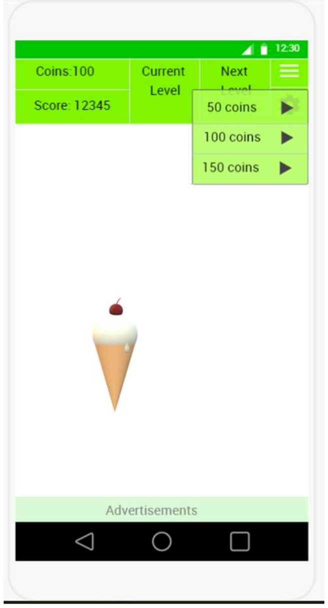
As the food item is eaten, the score will increase. Whenever the food item is eaten completely the coins count will increase. The coins can be collected on the screen by tapping them which will be shown occasionally and randomly spread across the center of the screen between the advertisements row and the top row. The frequency of the display of coins will increase if they tap on the advertisements and see them till end.

Screen 2 - Setting Screen



In this settings screen, the user is given options to save the score, change the skins, when tapped on the skins, it will show the available skins as a drop down, when tapped on any skin, accordingly the colors scheme for the monster face, the center of the screen, the colors of the top row except for the food items are updated. The “No Ads” will be greyed out for now, it will be enabled in the next releases, which enables them to pay a premium to disable the advertisements.

Screen 3 - Options Screen



In this options screen, the user is allowed to buy the coins by viewing the advertisements.

Key Considerations

How will your app handle data persistence?

Data persistence is done by Content Provider.

Describe any edge or corner cases in the UX.

From any screen, use will tap the ‘back’ button at the top left screen to go to the previous screen.

Describe any libraries you’ll be using and share your reasoning for including them.

This app will be developed completely in the Java language using the Android OS platform and will be using the following libraries as well:
Use the Glide for loading and caching of images
Use the Material Design libraries for the UI colors and design
Use the Google Play Game services for gaming features
Use the Google Mobile Ads for displaying the advertisements.
Use the Google Analytics to understand the behavior with and usage of the app.

Describe how you will implement Google Play Services or other external services.

As this app is a game, I will use the ‘Google Play Game services’ to provide the regular gaming needs like achievements, leader boards.

To generate the income, I will use the ‘Google Mobile Ads’ to show as a banner and rewarded video.

Stable versions for the libraries will used, following are the stable version as of this proposal writing.

Google Play Game Services : `com.google.android.gms:play-services-games:16.0.0`
Google Mobile Ads : `com.google.android.gms:play-services-ads:16.0.0`
Google Analytics: `com.google.android.gms:play-services-analytics:16.0.4`
Material Design : `com.android.support:recommendation:28.0.0`
Glide: `com.github.bumptech.glide:glide:4.7.1`
Gradle : `com.android.tools.build:gradle:3.0.1`
AndroidStudio :3.1.3

Next Steps: Required Tasks

Task 1: Project Setup

- Create the project in the AndroidStudio
- Create a repo in the GitHub.
- Configure the required libraries
- Gather the information on implementing Google Play Game Services and Google Mobile Ads.
- Collect the images (The face, food items, coins)
- Create the build variants

Task 2: Implement Screen 1

Build UI

- Create the Layout for the Screen-1

Develop the code for the Screen-1 (Main Activity)

- To display the face.
- To display the food item.
- To change the food item to show it is eaten as the user taps on the food item.
- To update the score.
- To update the coins.
- To display the current level and next level.
- To display the Ads using the Google Mobile Ads.

Debug

Enhance

Task 3: Implement Screen 2

Build UI:

- Create the Layout for the Screen-2
- Create the structure for the data storage to be implemented by Content Provider.

Develop the code for the Screen-2 (Settings)

- To save the score
- To display the skins
- To apply the color scheme as defined for the respective skin.
- To display the “No Ad”, but greyed out for now.
- Implement the Google Play Games Services

Debug

Enhance

Task 4: Implement Screen 3

Build UI:

- Create the Layout for the Screen-3

Develop the code for the Screen-3 (Options)

- To display the coin buying options with link for ads
- To play the ads when tapped on the play
- To update the coins when the ad is fully viewed.

Debug

Enhance

Describe the next task. For example, “Implement Google Play Services,” or “Handle Error Cases,” or “Create Build Variant.”

Describe the next task. List the subtasks. For example:

- Create layout
- Something else

Task 4: Test Cases

Develop test cases to test the Main Screen items.