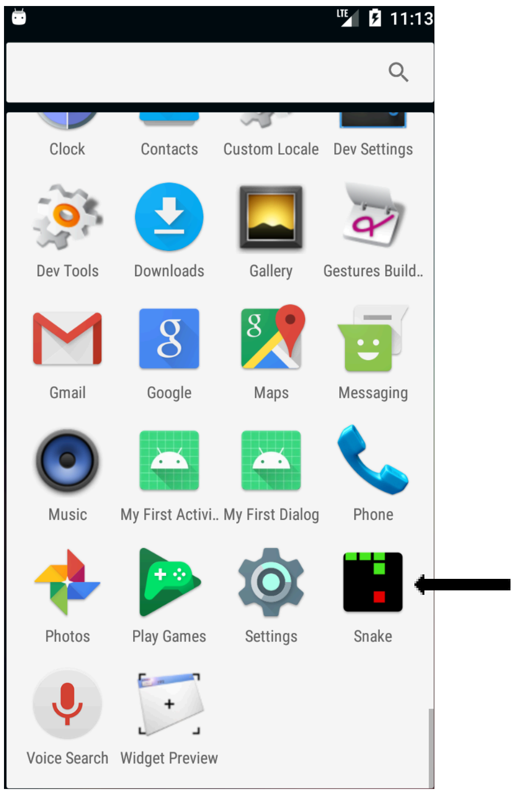
Snake User Guide

By Gabriel Shores for A290 Android Development

General description:
This app is a recreation of the classic game of Snake but for Android phones. Additionally, it comes with themes and navigational elements. For each section of the app this guide includes two parts: the **Guide** telling you how to operate it, and the **How it works** giving an overview of how it was created. More details about how it all works can be seen in the code and README.txt file. Each Java and XML file contains a header describing the file along with function headers and inline comments to make it easy to understand

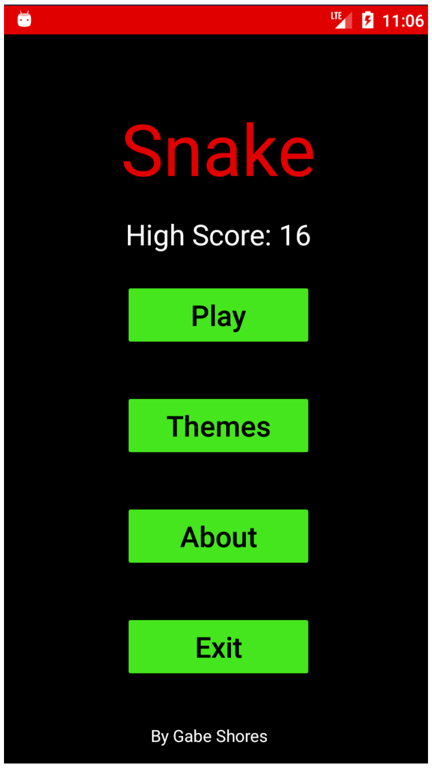
Launching the app



Guide
You have booted up your Android phone and are now ready to play a casual game of Snake. From your app list, navigate to an app entitled Snake. Simply click on it, and you are ready to go!
How it works

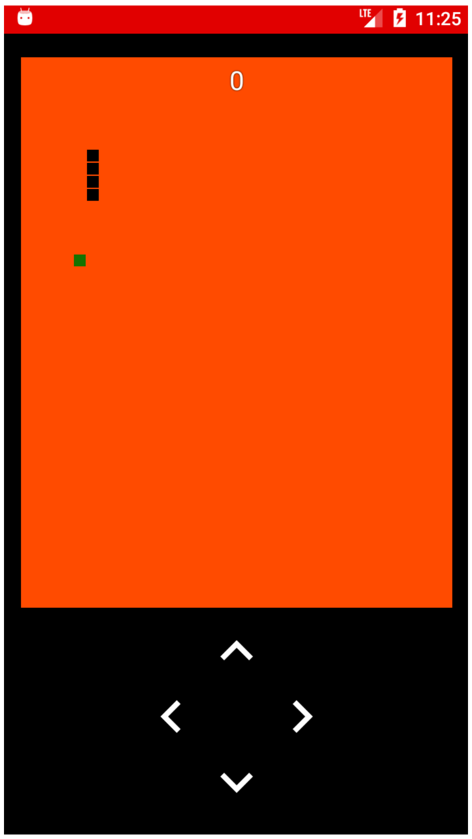
The app icon is just a png converted into and Android icon through Android Studio's tool. Launching the app is handled by Android Studio itself.

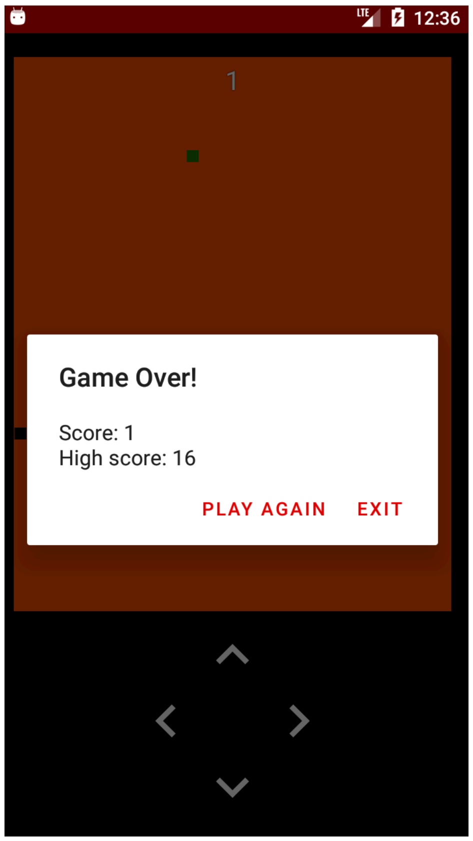
Home Screen/Main Activity



Guide:
After opening the Snake app you will find yourself here, the homescreen. At the very top is the title of the app, Snake, followed by your high score, which is updated and saved in local storage. Next, you have a series of buttons and an attribution to the developer. The first three buttons, following the squarish style of the game, bring you to their respective sections of the app, which are described in more detail further in this guide. Hitting the "Exit" button will exit the application and return to your Android phone's OS.
How it works:
This screen has a constraint layout, with the different views being constrained to one another. In the MainActivity.java class, onClickListeners are added to each button, and then an intent is started to go to their respective activity. The high score accesses a SharedPreferences object to see if a high score is stored, and if not, it subs in 0. Then, the text view containing the high score is updated.

Game Section/ Game Activity





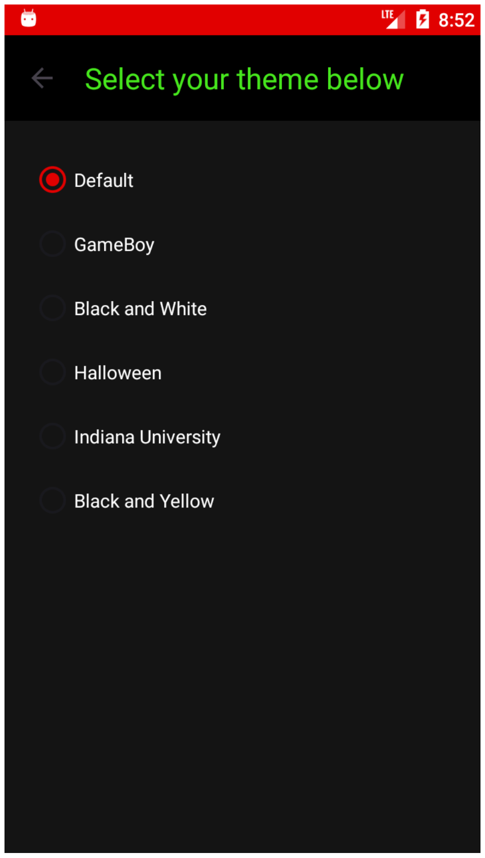
Game Over!

Score: 1
High score: 16

Guide:
You've now reached the game! Once you enter, don't be afraid if your game looks different; it renders based on your selected theme. Use the buttons on the bottom to change the direction your "Snake" (The sequence of connected squares) moves. Try to collect the "Apple" that randomly appears to expand your Snake, increasing your score and playing a beep. Once you die, a dialog box will pop up showing your current score and high score if you didn't beat your high score or saying that you got a new high score. At the bottom of the dialog is the option to exit or play again. Clicking exit will bring you back to the main menu, and play again will reset the game along with your score.
How it works:
This screen has a constraint layout, with the different views being constrained to one another. In the MainActivity.java class, onClickListeners are added to each button, and then an intent is started to go to their respective activity. Through a MediaPlayer object, the beep.mp3 sound is played whenever an apple is eaten. The high score accesses a SharedPreferences object to see if a high score is stored, and if not, it subs in 0. Then, the text view containing the high score is updated. The whole game is drawn on a Canvas tied to the surface view, utilizing the theme class to get its colors

and using the Timer class's scheduleAtFixedRate() class to update at a fixed interval, making it framerate independent. The back button is disabled to prevent exiting in the middle of the game, which can cause some issues. The dialog box is created upon death, and it is also locked, so the user is forced to use it to exit the main menu, ensuring every object is properly closed. The primary sequence of the app is the surface is created -> game starts -> game runs -> game over -> game starts or exits to menu.

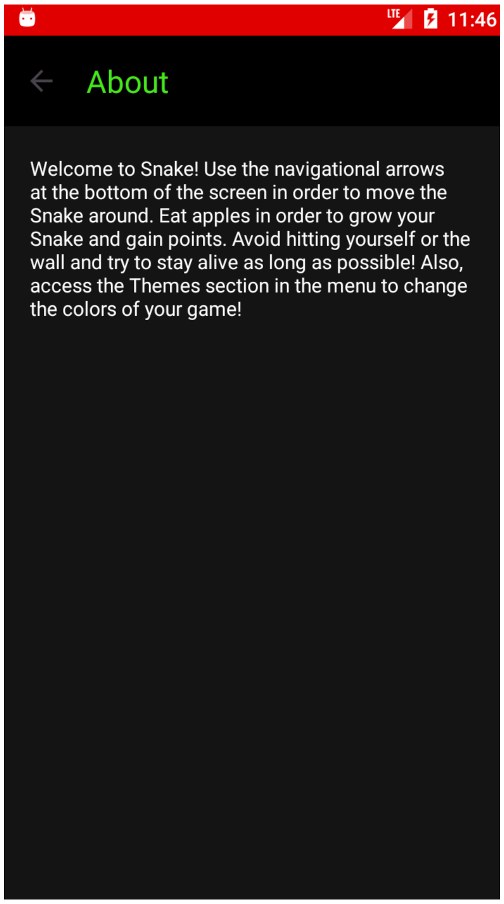
Themes/ Themes Activity



Guide:
Upon clicking the Themes button in the main menu, you should now be in the Themes Activity. Use the radio buttons to select your theme and save it to disk. These themes will then be used to determine the colors with which the actual game is drawn. If you want to exit the activity, you can then click the back button at the top or on your phone to go back to the main menu with all your settings saved.
How it works:
Once the theme activity is created it will fill in the radio button with the theme selected default if there is not one. Once one is clicked, the SharedPreferences object

is accessed and the theme updated, along with all other radio buttons being unchecked due to being in a RadioGroup. The back button is programmatically added to the toolbar and calls onBackPressed().

About Section/About Activity



Welcome to Snake! Use the navigational arrows at the bottom of the screen in order to move the Snake around. Eat apples in order to grow your Snake and gain points. Avoid hitting yourself or the wall and try to stay alive as long as possible! Also, access the Themes section in the menu to change the colors of your game!

Guide:
Upon clicking the About button in the main menu, you should now be in the About Activity. Reading the text will remind you of how to play, and you can click the back button at the top if you want to exit.
How it works:
The About section is a simple activity just utilizing a linear layout, toolbar, and a textview to display the text. A back button is added programmatically, but that is all that needs to be handled.