CS 591, Mobile Application Development

Worksheet 4

Project Planning, Accelerometer, and Gestures

Date: _____

Team Members: _____

This worksheet is to be done in collaboration with your project team. Although we are working with Android, the concepts apply for any device.

Part I: Initial Project Ideas:

- **Meet with your team for 30 minutes**
- **Based on what you know how to do now, and what you would like to learn, come up with 3 ideas for projects. Be as descriptive as possible. Please make at least one of your selections a BU SPARK project.**
- **Describe third party libraries you would like to work with (e.g., - facebook, Instagram, ebay, paypal, etc.).**
- **Identify what the pinch points might be, and what you would need to learn to implement your project ideas.**
- **Be ready to briefly present your ideas to the class.**

Part II: Accelerometer Measurements.

1. **Write an app that measures movement along the x,y,and z planes. When the device moves "significantly" in any one direction report this in a log and via a Toast. You essentially don't want to report every small movement, just large ones. You can define what is considered *significant* via experimentation.**

2. **Add a SeekBar to the App to allow users to *tweak* what is considered a significant move.**

3. **Add a WebView Component to the screen, and have it load the following urls based upon the size of the biggest move. If the biggest move is in the:**
    a. **x direction, you will load http://google.com**
    b. **y direction, you will load http://yahoo.com**
    c. **z direction, you will load http://bing.com**

    Notes: (1) It might be best to test the WebView first, ensure you can load web pages with a simple button click. You may need to enable Javascript and/or provide permissions for internet access. Refer to the Android Documentation as needed. (2)Only perform these actions if the change in direction was "significant". (3) Feel free to lock the screen in portrait or landscape to avoid intermediate destruction of Views. Use

```
  setRequestedOrientation(ActivityInfo.SCREEN_ORIENTATION_LANDSCAPE);
or
  setRequestedOrientation(ActivityInfo.SCREEN_ORIENTATION_PORTRAIT);
```

4. **Make it so that if you shake your device <u>really, really fast</u>, it displays the following URL:** https://jumpingjaxfitness.files.wordpress.com/2010/07/dizziness.jpg

Part III: Pulling it all together.

- **Create an App that Has 5 Activities, Name them Home, North, East, South, West.**
- **Put some sort of text or image in each of the Activities. Feel free to use whatever you like, just make clear which activity the user is on.**
- **The Launcher Activity should be the Home. Use flings to open the other Activities.**
    - **Right Fling: Opens East Activity**
    - **Left Fling: Opens West Activity**
    - **Upwards Fling: Opens North Activity**
    - **Downwards Fling: Opens South Activity.**
    - **Shaking the device feverishly should cause the image to also shake for at least 2 seconds**

Part IV: Hangman Game in English.

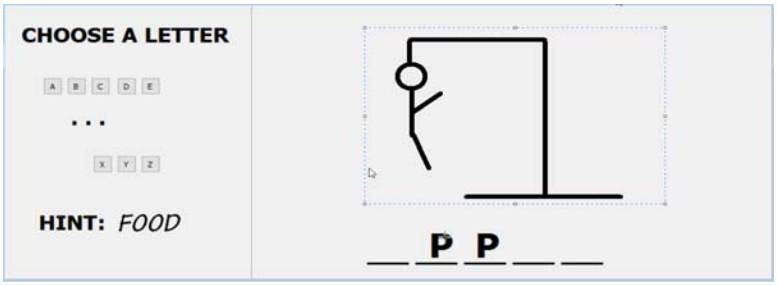
1. **Create a Hangman Game. This is in preparation for our Lesson on Fragments next week. Target both a Portrait and Landscape views as shown below. This is a design challenge as well, so please make design choices on your own, (eg.,- how to render the hangman stick figure, how to tell the user they won or lost, etc.) Your game should be able to run at least one complete game and handle both winning and losing, no scoring is necessary. *Include a "New Game" Button which will spawn a new and different game when clicked.***

Panel 1: Contains a "Choose the letter" set of buttons. Provide a menu of letters, just use buttons. Be sure to disable the button after it's been selected.

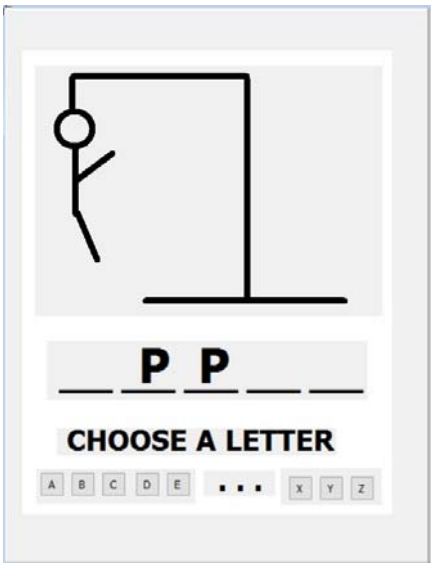
Panel 2: Contains a Hint Message

Panel 3: Contains the Main Game Play Screen.

For a Tablet in Landscape mode the display should appear as below.



For a phone or a Tablet in Portrait Mode, the interface should be simpler, laid out vertically, and not show a hint.



General Game Play:

After choosing a letter, the user will be notified whether the letter was correct or not. If correct the letter will be displayed in the proper area, if incorrect, another body part will hang. The game ends when the user successfully identifies the word or is completely hung.