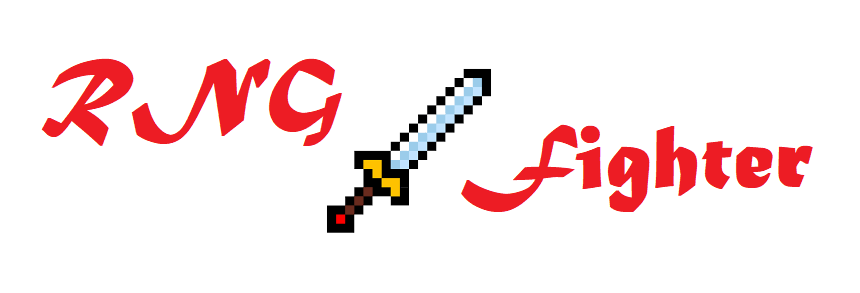
**Mobile Applications Project Design Document**

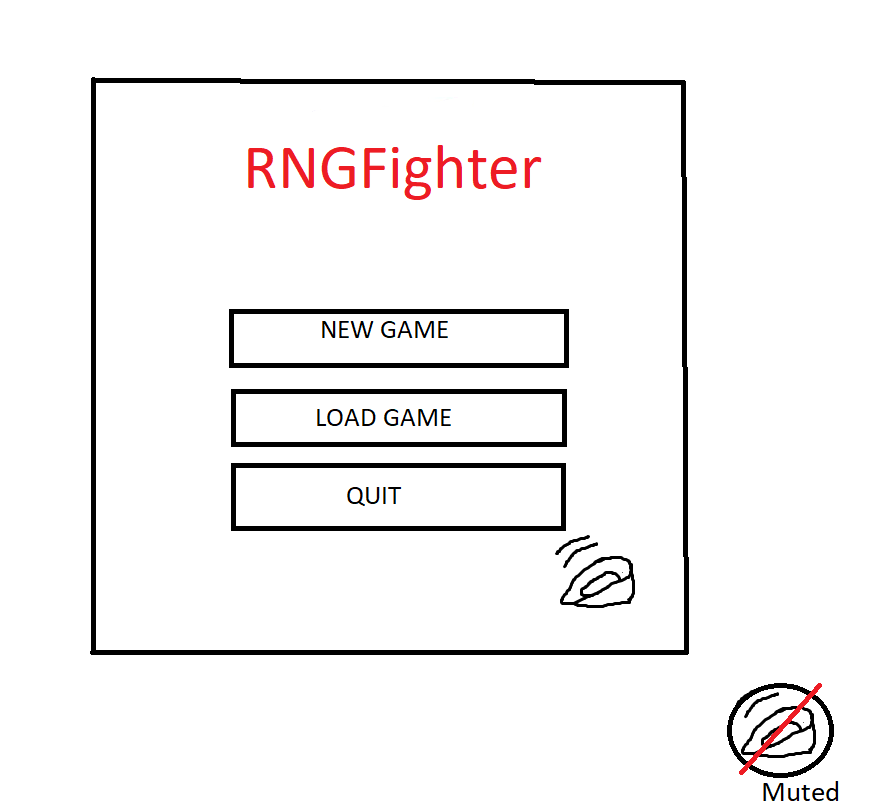
**G00349242 – TOMAS BRAZAS**

(Design may change as coding proceeds)

A **Windows Cross-Platform application** that will test your luck as you battle through the beasts to improve your score. RNG Fighter is a game that I’ve chosen to develop and design for my mobile applications module. Game consist of tapping buttons that will have either a positive or negative outcome towards your battle. The word RNG usually refers to a person being either lucky or unlucky. It is coded using **Xamarin** and **Visual Studio**.

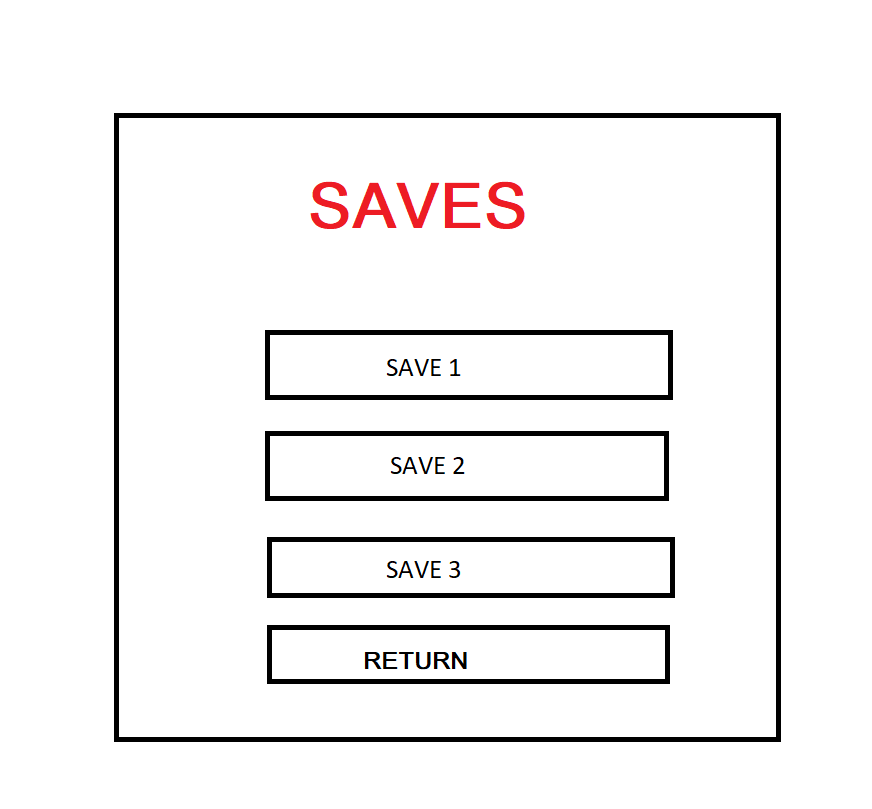
**Consists of:**

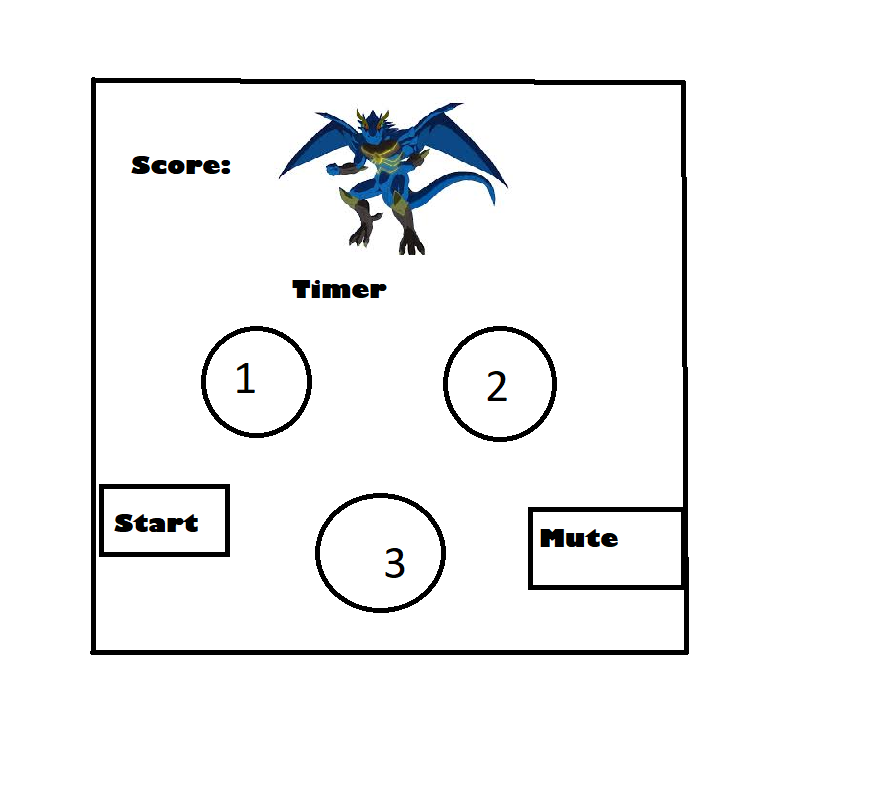
* Well Designed UI.
* Data Storage.
* Use of Sensors/Hardware.

**Well Designed UI**

I’ve chosen a simple and easy to read menu user interface. It is designed to fit the requirements and control them. A sound button is present that can be either unmuted or muted, this may mute the non-copyrighted background music and sound effects. The menu is designed and structured using stackpanels, grids and buttons. The UI will also be scalable.

**Data Storage**

Once the user interacts with one of the buttons the data storage component of the application will be presented. Tapping “Load Game” will bring up a Data Storage user interface such as:

This will be structured using stackpanels, grids and buttons. The “Save #” buttons will interact with a text file that will be created as the user taps one of the buttons. If previous file exists on that targeted Save, the user will receive an alert to either proceed with over-riding the existing file or cancel his action. Data storage in this project is using a JSON format text files to manage this data, variables such as Time, saveID, CurrentScore will be saved/registered. This is the same for “Load Game” button.

**Gameplay**

Once the user has initiated a save, the gameplay   
will commence. A “monster” is visible at the top  
of the screen with a timer.  
A timer will also be present, if the user does not  
succeed killing the monster in the given time an  
alert of “Game Over” will pop up. Buttons 1,2,3  
will have the RNG factor of this game. The outcomes  
will be user loses certain amount of time from his   
timer, monster loses health or user/monster gains   
health/time.

**Sensors/Hardware**

I will mainly focus on the user of the sound hardware to implement background music that will can be toggled and will be continuous throughout the user use of the application.

**Further Development**

These are some of the ideas that may be implemented to further develop the application.

* Reward System

The user may be rewarded “coins” for achieving score that may be spent in a “shop” that would be present in the main menu. These could be spent in unlocking new bosses/soundtracks for the game.

* HighScore System

All current high scores may be recorded in a separate text file. Once a user finishes a game the CurrentScore would be compared to this text file and replace any that are lower than CurrentScore. The high scores may be accessed on the main menu with a “High Scores” button.