**Cross-Platform Development**

**Planning Document**

**Galaxian Clone**

Nicole Dance (Brown)

Contents

[Revision History 3](#_Toc23851135)

[Screen and GUI mock-ups specifying input of display adjustments made across platforms 4](#_Toc23851136)

[Main Menu 4](#_Toc23851137)

[Game Screen 5](#_Toc23851138)

[High Score Screen 6](#_Toc23851139)

[Identification and discussion of player control issues across platforms 7](#_Toc23851140)

[Screen size and aspect ratio differences 7](#_Toc23851141)

[API or software version requirements 7](#_Toc23851142)

[Deployment methods 7](#_Toc23851143)

[Platform-specific features or constraints 7](#_Toc23851144)

[Industry stand tools, APIs, or methods for handling cross-platform development issues 7](#_Toc23851145)

[Identification of any environmental considerations involved with using the technologies used in the project creation, and/or with the final product. 8](#_Toc23851146)

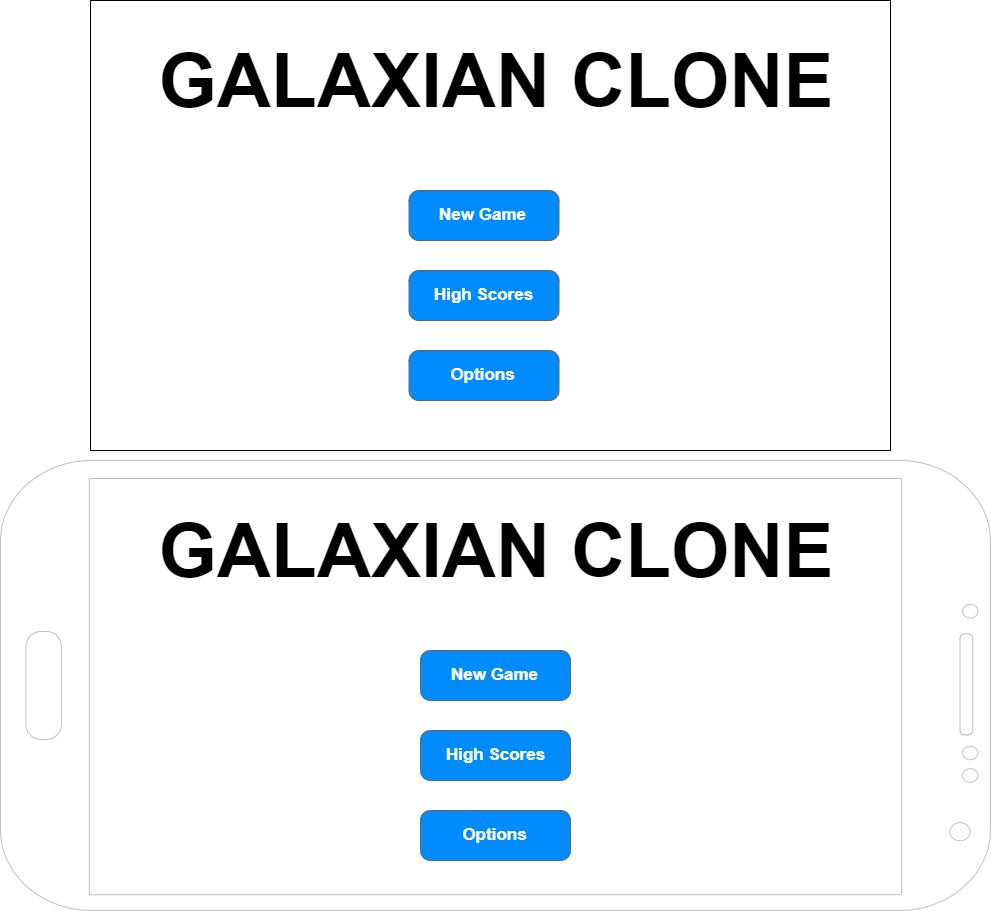
[Bibliography 9](#_Toc23851147)

# Revision History

|  |  |  |
| --- | --- | --- |
| **Date** | **Version** | **Description** |
| 5 November 2019 | 1.0 | Created Documents and added initial details |
|  |  |  |
|  |  |  |
|  |  |  |

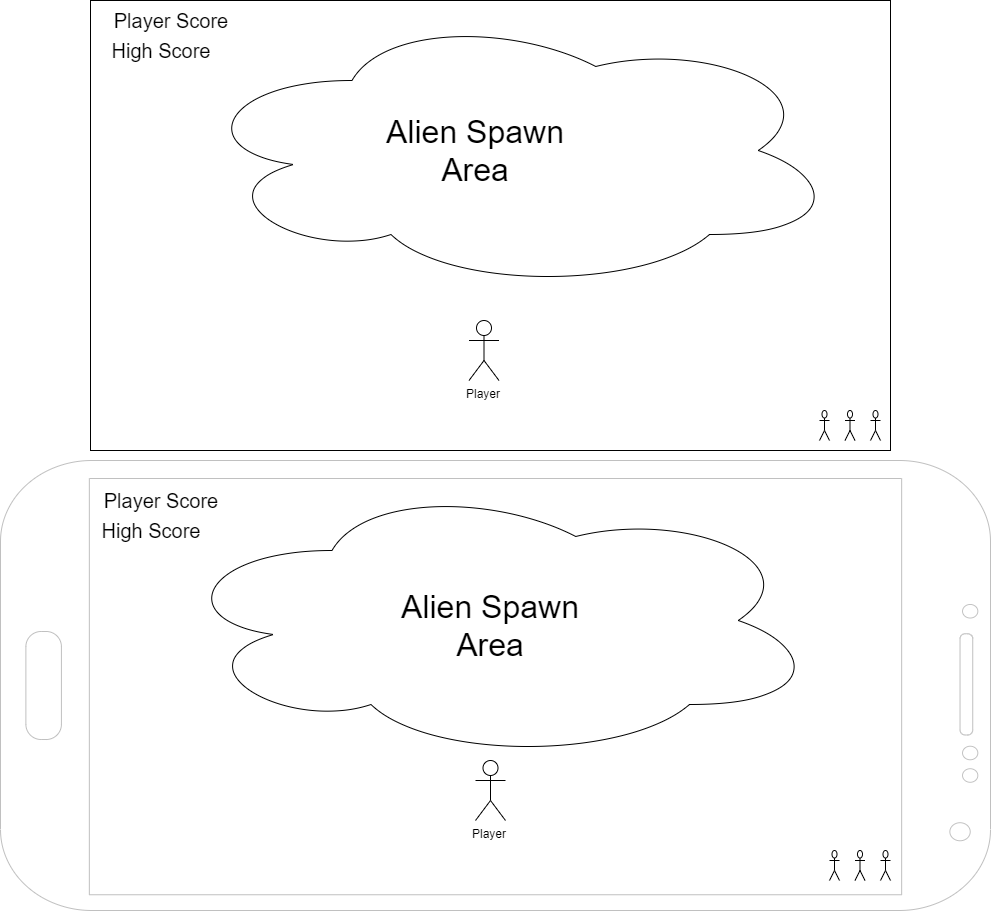
# Screen and GUI mock-ups specifying input of display adjustments made across platforms

## Main Menu



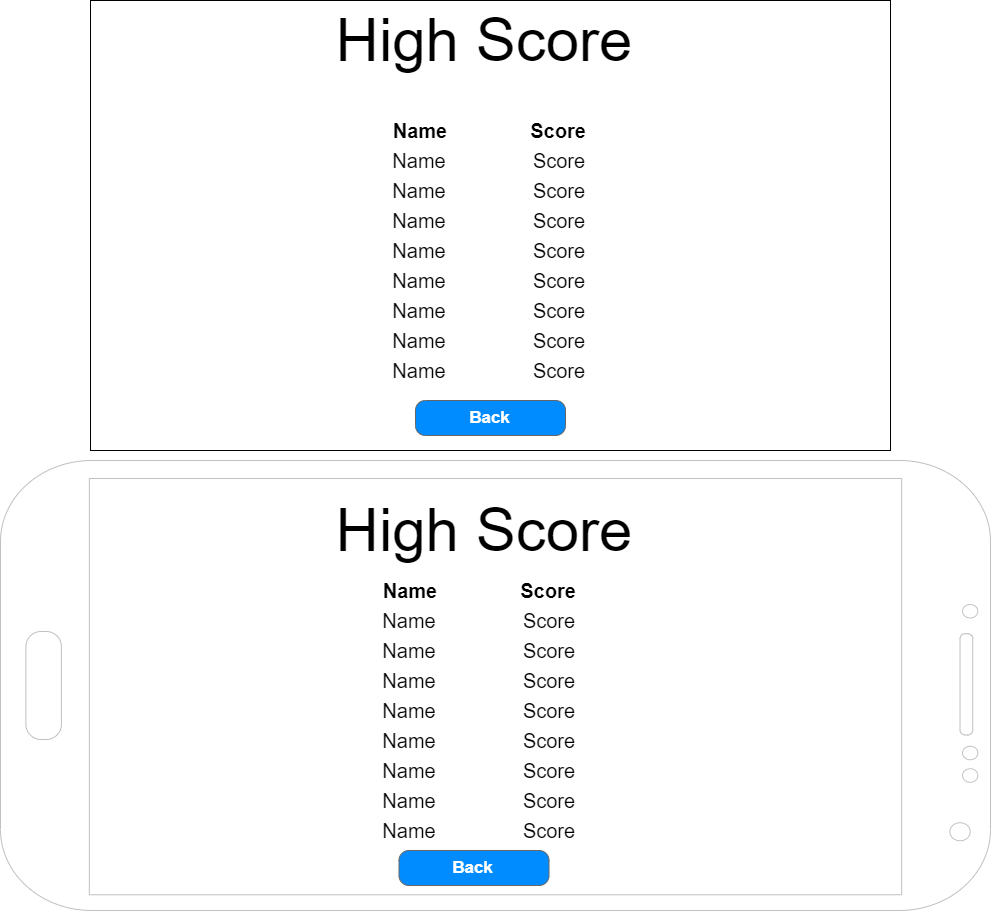
The Main Menu screen will appear to be almost identical between devices. Input wise it will be slightly different. On the Pc version, the user can either click the buttons with there mouse or move between them using the up and down arrow and press enter to select one. On the mobile version the user will tap the buttons.

## Game Screen



The Game Screen like the Main Menu will look similar between devices. The Pc controls will be using left and right arrows or a and d to move the player left and right and space to shoot. For the mobile, I will use the accelerometer movement where the player will tilt the phone left and right to move and then tap the screen to shoot.

## High Score Screen



This screen is pretty self-explanatory. The user will click or tap the back button to go back to the main menu.

# Identification and discussion of player control issues across platforms

As mentioned the devices I will be building for are PC and Android. They will both need very different ways of controlling them. The PC will use a mouse and keyboard controls. Android will need to be different as they have no keyboard or mouse attached. For the Android device I will make use of its touch screen for accepting taps on menu buttons and to shoot. I will also use the built-in accelerometer to detect the tilt of the phone to move the player.

I use Preprocessors like #if UNITY\_ANDROID to execute the relevant sections of scripts for the required functionality.

# Screen size and aspect ratio differences

With a large number of different devices comes a number of different screen resolutions and aspect ratios. My game will be developed for Android and PC so I started by looking into the most common resolutions used across these devices. The most common resolutions across these devices according to Reinisfischer is 640x360 (16:9), 1366x768 (16:9), and 1920x1080 (16:9). With that in mind, I will be building my app to work with the 16:9 aspect ratio. (Browser Wars / Screen Resolutions 2019, 2019)

Whilst I will be focusing on the 16:9 resolution. I will be checking the display across 2 other aspects ratios as well, 4:3 and 16:10. According to Rusen along with 16:9, these 3 make up the most common aspect ratios used today. (Rusen, 2019)

# API or software version requirements

To develop Galaxian I will be using Unity version 2019.2.5f1.

# Deployment methods

Unity will allow for the program to be developed once and deployed on both platforms. The pc version will be built and an installer created using the Innosetup tool.

An APK file will be built for the Android version. This can then be installed on any android device.

# Platform-specific features or constraints

The game I am creating Galaxian is a clone of a 1970’s arcade game. I will be modernizing it a bit and making a 3d version. The main Feature/constraint will be around the input controls. As mentioned above the PC will utilize keyboard and mouse controls while the Mobile will use touch and the accelerometer.

# Industry stand tools, APIs, or methods for handling cross-platform development issues

The tool used to create this cross-platform game will be the Unity engine. Unity allows a game to be built once and with some minor modifications including setting up the UI, controls, and API for different platforms. It can be programmed once and build on any platform you would like.

This speeds up the development lifecycle as you no longer have to reprogram the game in the various programs' native languages.

# Identification of any environmental considerations involved with using the technologies used in the project creation, and/or with the final product.

My game will not be virtual reality so there won’t be any real environment considerations. The will need to make sure they have space to move there phone but that shouldn't be an issue.

# Bibliography

*Browser Wars / Screen Resolutions 2019*. (2019, January 30). Retrieved from Reinisfischer: https://www.reinisfischer.com/browser-wars-screen-resolutions-2019?fbclid=IwAR3V1Ti-k0Ds3JYDcSFybS0qmwA4pNUD-SdDH4ZasdIqsG86n3C-8oOM61Q

Rusen, C. A. (2019, March 7). *What are the aspect ratio & orientation?* Retrieved from Digital Citizen: https://www.digitalcitizen.life/what-screen-resolution-or-aspect-ratio-what-do-720p-1080i-1080p-mean