Ben Townshend

CGP2011M

# Libraries

* SDL2/2.0.5@shearer12345/testing
* SDL2\_image/2.0.1@lasote/stable
* SDL2\_mixer/2.0.1@a\_teammate/testing
* RapidJSON/1.0.2@SamuelMarks/stable

# Main Features

* One player mode
* Two player mode
* Options menu where the audio level can be adjusted.
* Loading Screen.
* Pause menu.
  + Game simulation and audio halts when the game is paused.
* Rebindable keys.
* Render information logged to screen.
* Enemy pathfinding, with random and seek mode. The seek mode is enabled when the enemy is in the same row or column as the player, and has line of site.

# Additional Features

* Game simulation considers real time.
* Pickups
  + Biscuit
  + Power pill
* Player and Enemy Animations.
  + Animations stop when player is not moving.
* Players and Enemies collide and react appropriately to pickups, walls and each other.
* Once a player collects a pill, they gain a speed boost, can eat enemies which resets them to the spawn section and if the game is in two player mode, the opposing player is frozen from moving.
* Audio
  + Music
  + Sound Effects triggered by in game events.
* Players have several lives.
  + Lives illustrated on the display
* Temporary invulnerability and speed boost on lose of a life.
  + Indicated by the player becoming semi-transparent.
* Controller Joystick and DPad support.
* Menu supports Joystick, DPad, mouse and arrow key navigation.
* Advanced Window Functionality
  + Title is set.
  + Starts at half the width and height of the desktop and is centred.
  + Aspect ratio is maintained.
  + Can be toggled to and from full screen mode.
  + Can be moved and resized
* Player score is displayed on the screen.

# Reflection

The development process went well overall. Once a base build system was established, the program was built upon and refined. The use of classes and inheritance kept the code clear and reusable, allowing all the game objects to be created with ease. The project was stored in a private git repository making developing across multiple devices easier, as well as allowing incorrect code to be rolled back if I ever encountered issues.

I decided to develop the project outside of Visual Studio, using Atom instead. This is because it is my preferred development environment, but also because the visual studio solutions are generated by cmake, whereas working within Atom allows me to work independently from the contents of the build folder.

Due to issues with the conan support of SDL\_ttf, I decided to use a sprite sheet to represent my fonts. This solution worked well, but required more logic to be written behind the scenes. I created a class that manages a text object to make creating them easier.

# References

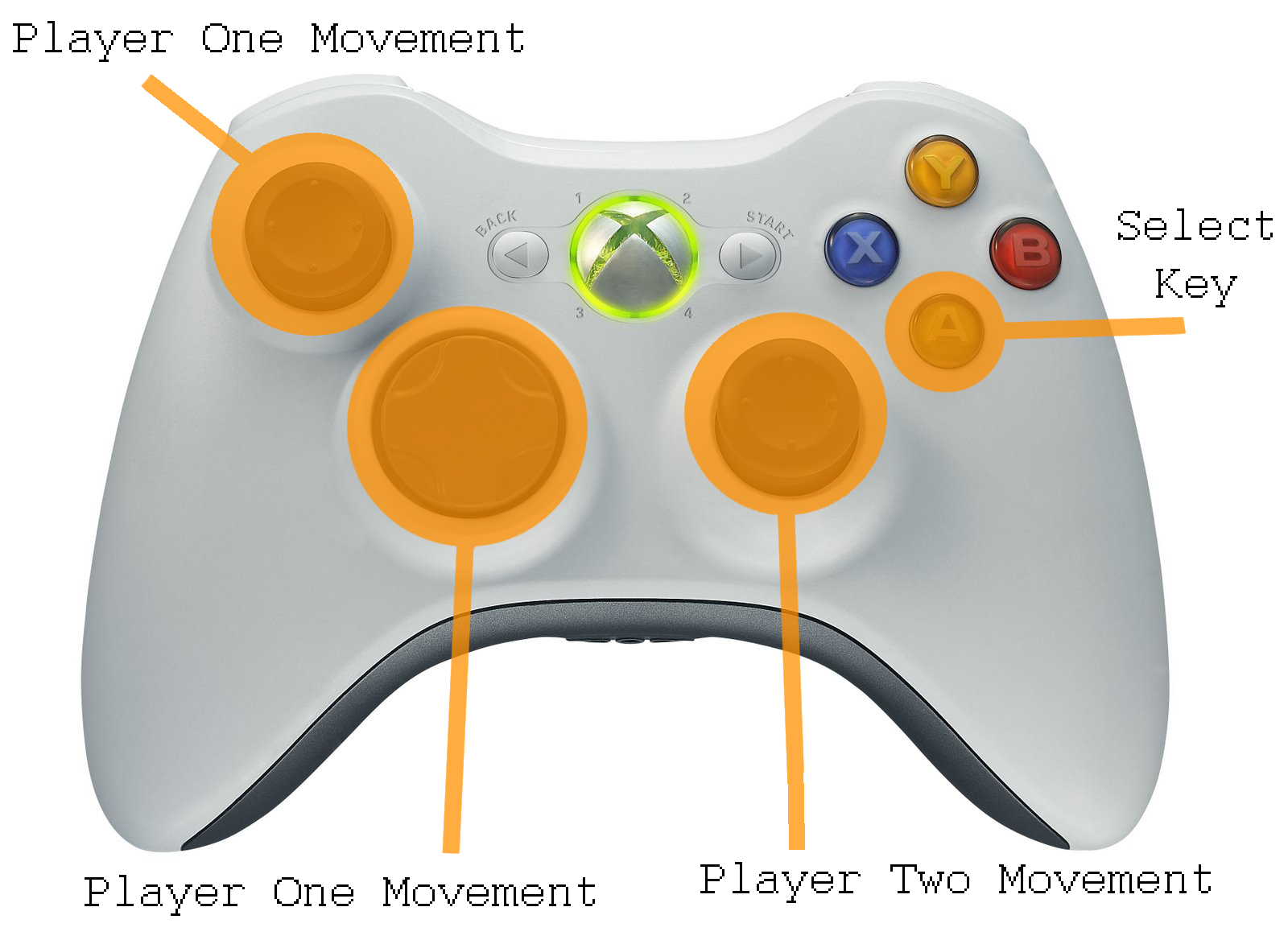
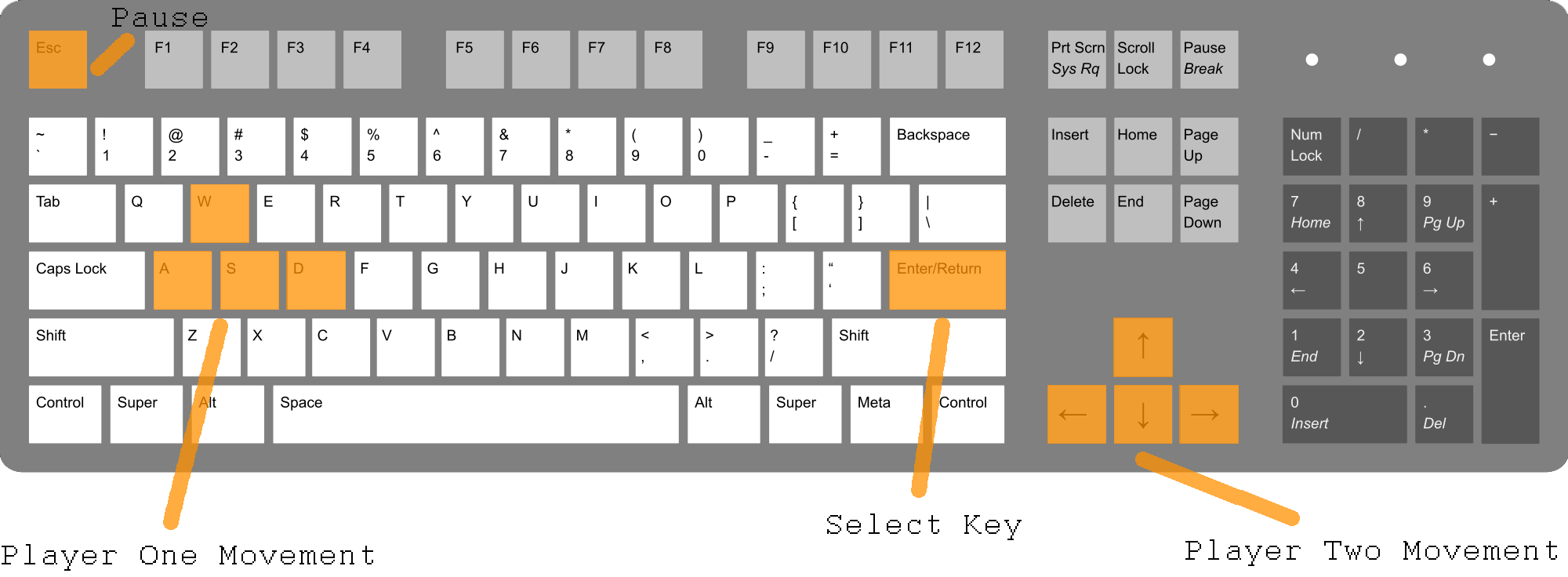
Main Menu and In-Game Music – Andrew *‘Glowtide’* Glover #26

Sound Effects Generated using <http://www.bfxr.net/>

I personally created all the Sprites.

# Evidence

|  |  |  |
| --- | --- | --- |
| https://i.gyazo.com/63e7c67d3c206a782444644105b1b2d3.png | One player mode | |
| https://i.gyazo.com/a1e5653f51070ee56f50c7242488993c.png | Two player mode | |
| https://i.gyazo.com/3499a17296d47fb49f903fa13604b246.png | Options menu | |
| https://i.gyazo.com/c27a6f19cd63c438d74ddf38769806c9.png | Loading screen | |
| https://i.gyazo.com/6bf9dc425d0d5b99223a6ad45cf7d7b1.png | Pause menu | |
| https://i.gyazo.com/08ee70d0afea3e90b3edb6e091d36b06.png | Rebindable keys | |
| https://i.gyazo.com/298a21996c7adc1a2e63d8af608c17ef.png | | Render info |

Control Scheme­­­­­­