Blockening

Game Concept:

The concept of the game blockening is to move left or right while ducking and weaving through the falling blocks without getting hit by the descending cubes. While doing this your score will go up the longer you live.

Features:

The features of blockening will be left to right movement, the ability to jump and climb the blocks that have fallen, blocks falling from the sky, lava will rise from the ground so you must climb the blocks and if there is time wall jumping of the side of any block you can climb.

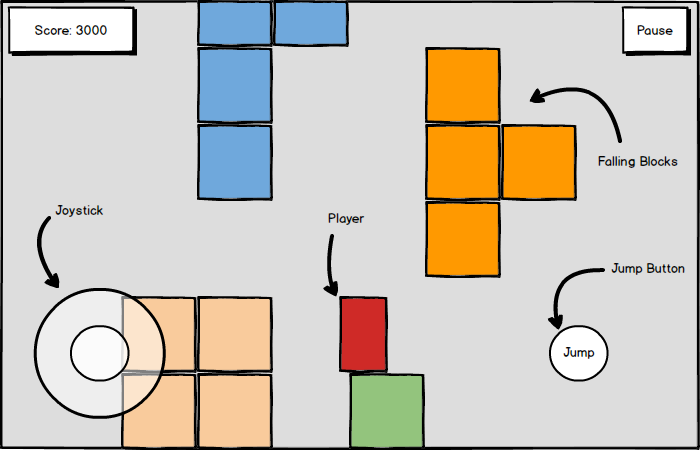
Technical Risks:

The features that are most likely to cause problems are the Lava rising form the bottom of the screen this I believe will be harder to pull off because it is something that we haven’t had a lot of practice at. Another feature that is likely to cause a problem is the falling block again this is for the same reason as the lava that being there has been minimal practice at this type of mechanic. And the last feature that may cause a problem is the side jumping this mechanic seems more complicated than the others and will require research to-do.

To reduce the risk of all three of the problems with these features Alp who is working on the majority of the programming will do research on these mechanics and features and how to produce such things to the best possible standard.

Game Flow:

The game will start with a menu to start or quit it will then jump into the continues scrolling game as the camera followers the player as they ascend the level. When the player dies either by falling block or lava it will show the game over screen it will also show the score.



GameObjects, Scripts and Systems:

Rising Lava

The Lava object will move up the screen at a fixed rate giving the illusion that it is following the player when the lava hits the player it will continue to rise until they game over screen is displayed. If the lava hits the player, then the game over screen displays.

Falling blocks

The falling blocks object will fall randomly from the top of the screen and the player will use these blocks to scale up the screen for as long as possible if the player is crushed by this game object then the game over screen plays.

Input Method:

The input method for the pc version of the game will be relatively simple. Using the keyboard as the main input device you will be able to move the player using the below keyCodes: - A or Left Arrow, Move the Player left.

- D or Right Arrow, Move the Player right.

- Space or W or Up-arrow, Make the Player jump.

As for other platforms, there will be similar controls but translated for the appropriate platforms:

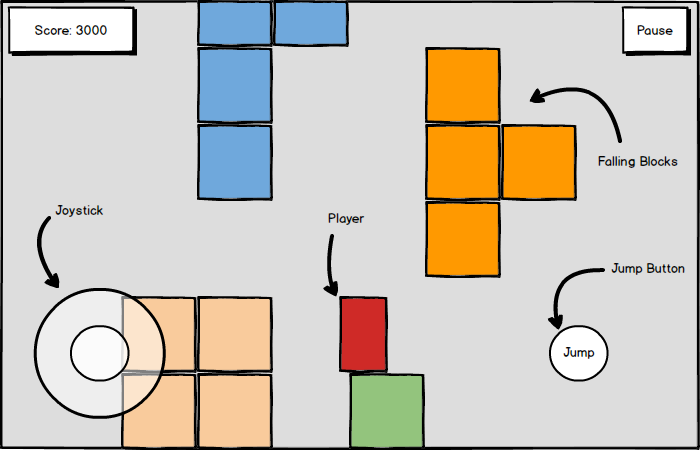
- PlayStation, PSVita and Shield: Joystick for Player Left and Right, X button for Player Jump.

- iOS and Android: Virtual on screen Joystick for Player Left and Right, On screen A button for Player Jumping.

User Interface:

Here is a mockup of what the game will more or less look like on all devices. This mockup is showing touch controls, which of course will not be present on PC, PS4 and PS Vita.

As for the Menus user interface, it will be basically the game logo with a basic background, a Play button, Scores button and Exit button.



Deliverables:

The deliverables for the game will vary for platform, each are listed below:

- Pc, will have an .exe file for launching the game and will not require the use of an installer file.

- Android and NVidia Shield, will have a.apk file installed for the app on the phone.

- iOS, will have a.ipa file installed for the app on the phone.

As for the PlayStation and PS vita builds they will just be pushed to the devices from Unity.System Requirements

System Requirements:

The devices that we’re targeting are ps4, ps vita, android, NVidia Shield and ios. The recommended hardware for PC you need to have a GPU with DX9 or DX11 with feature level 9.3 capabilities and at least Windows XP SP2+, For IOS it needs minimum IOS 7.0, android minimum requirements are OS 4.1 or later; ARMv7 (Cortex) CPU with NEON support or Atom CPU; OpenGL ES 2.0 or later. (look-up unity requirements). We’re going for portrait display on mobile.

Third Party Tools:

What third-party tools are you using? List Unity with version number and any other tools you might need. Include any asset packs you plan to use from the asset store. We will not be using any third party tools. Unity version we are using is 2017. 1.. We will be using an asset from the unity asset store, it is called Unity-Chan which is designed by unity, made by Flight Unit, Copyrighted and produced by Unity.

Coding Conventions:

We will be using Hungarian notation throughout the project.

Source Control:

We will be using GitHub for source control. For commit messages everyone should clearly point out what they have done to the project.

Team Members:

Liam – TDD Document

Alp – programing and demo of the game (did some of the TDD documents after the slideshow was over).

Thomas “Jeff” – PowerPoint presentation (did some of the TDD documents after the slideshow was over allowed use of the mockup he made for the slideshow in the TDD).