Team Lead 6

Kurian Georly Kunnathushery, Josie Wicklund, Ryan Kirkpatrick

Implementing the the Game into three Platforms.

Windows, Android, and IOS

How to Implement a game into Windows or MAC using unity?

- Open unity in your system (Make Sure the developing game does not have any error)
- Open File
- Press Built Settings
- Choose PC or MAC
- Browse the location where you want to save the file and press build

How to Implement a game into Android using unity?

Steps:

- Open unity in your system (Make Sure the developing game does not have any error)
- Open File
- Press Built Settings
- Choose Android
- Press switch platform

While choosing Android

- Connect your android phone to pc
- From phone, Open settings
- Open about phone
- Software information
- Tap build number seven times
- Open developer options
- Enable usb debugging

After enabling usb debugging:

- Download unity remote 5 app from the play store and install it to your phone.
- Open the unity remote 5 app
- In pc, choose your phone model from built settings
- Test the game from the system to your phone and make sure it works.
- Then build and install .Apk file to your phone

Demo on showing How to test the game in Android

How to Implement a game into IOS using unity?

Steps:

- Open unity in your system (Make Sure the developing game does not have any error)
- Open File
- Press Built Settings
- Choose IOS
- Press switch platform

After Switching Platform

- Download and install Unity Remote 5 App from App Store
- Connect your Apple device to Mac or PC
- If using PC, download ITunes from Microsoft Store
- Test the game by connecting your IOS device to PC or MAC

Building the game onto an IOS device

- First you will need to download xCode, have an AppleID, and create a free apple developer account.
- Open .xcodeproj file created after you build from unity.
- This will open in XCode where you will build the game onto your IOS device.
- You have to change a lot of settings on xCode and IOS device

Procedures for testing the game in android, and ios platforms.

Along with the normal game need to develop two new scripts based on the game:

a. Haptic Joystick
b. Gyro sensor (device that sense angular velocity)

Haptic

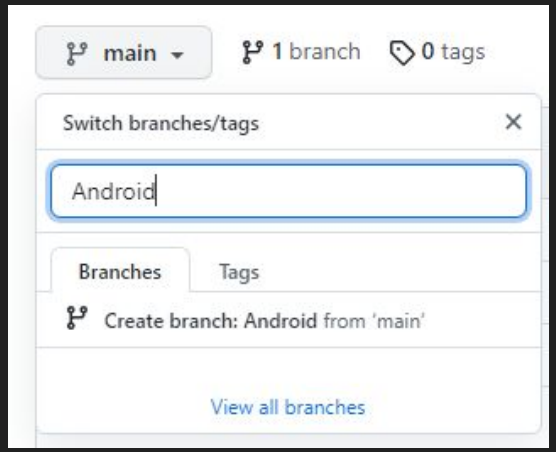
Acceleration vs Gyro

Gyro is way more confusing

```
public class HapticMovement : MonoBehaviour
{
        public Text gyro;
        public CharacterController2D controller;
        public float runSpeed = 40f;
        float horizontalMove = 0f;
        bool jump = false;
        bool crouch = false;
        // Use this for initialization
        void Start()
        {

        }
        // Update is called once per frame
        void Update()
        {


                if (Input.acceleration.x >= 0.3f)
                {
                        //gyro.text = Input.acceleration.x.ToString();
                        horizontalMove = runSpeed;
                }
                else if (Input.acceleration.x <= -0.3f)
                {
                        //gyro.text = Input.acceleration.x.ToString();
                        horizontalMove = -runSpeed;
                }
                else
                {
                        horizontalMove = 0f;
                }

                controller.Move(horizontalMove * Time.fixedDeltaTime, crouch, jump);
                jump = false;
        }

}
```

Branching

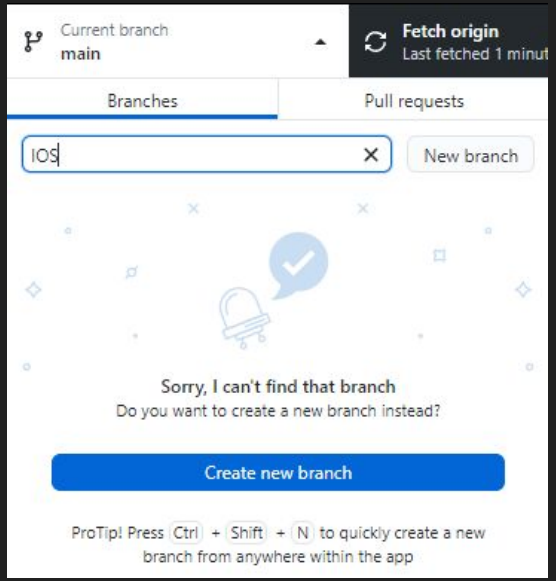
Website



Terminal

```
git checkout -b <new-branch>
```

Desktop Application



Potential issues with changing platforms

- Inputs (mouse, touch screen)
- Outputs (screen size, vibrations)
- Software compatibility
- Hardware (computation power)

QR code

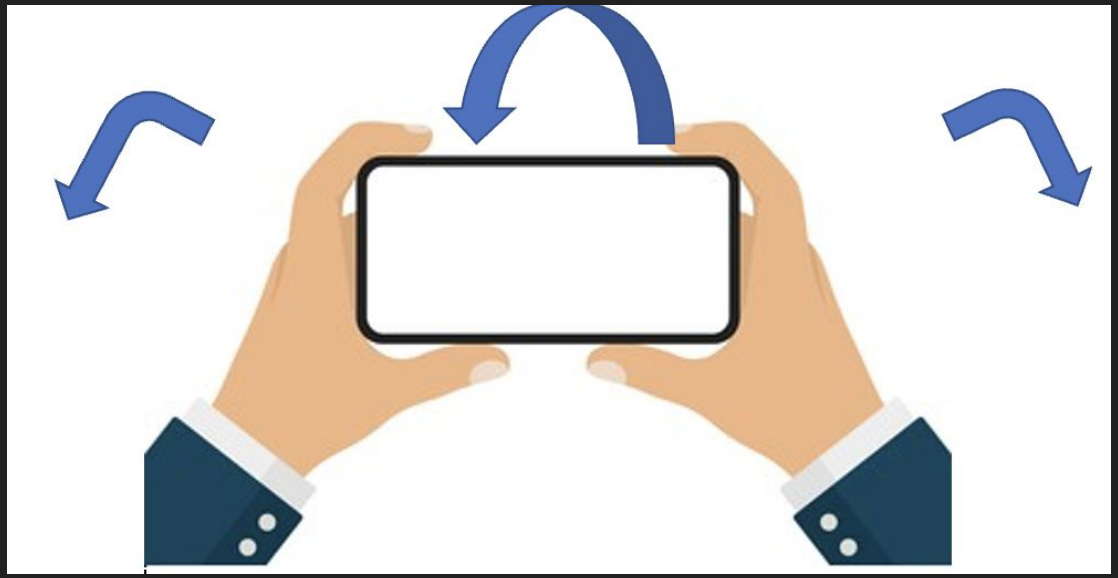
Android

User Manual

Tilt left to move left.

Tilt towards you to jump.

Tilt right to move right.



Remaining Points:

- Midterm & Ethics Quiz [April 14th] - 70
- Project Feedback - 10
- Oral Exam Prework [Due April 16th]
- Oral Exams [21st - 22nd] - 200
- Pair programming - 40
- Post Mortem [May 5th] - 50
- Final demo - 40

440 points left.

Goal:

- Over 900 -> A
- Over 800 -> B
- Over 700 -> C

Example Sign Up

What to study for Thursday

- From “Mapping Models to Code” the 4 different types of transformations and the names of the two “spaces”
- Singleton Pattern
- A * pathfinding in Unity
- ACM/IEEE-CS Joint Task Force on Software Engineering Ethics and Professional Practices
  Ethical principles. (Know the 8 titles and how to apply them.)
- What is/is not covered under copyright
- The four ways to argue fair use

Oral Exams

- Don’t forget to submit the prework! April 16th
- See the marking key for what you will be graded on.
- On the day of the exam come to JEB 324 about 10 minutes early and be set up and ready when your start time begins.
    - Ensure that your Gantt chart is up to date and all your code is your directory in the main (only) branch on GIT. (If it is not in your directory running as part of the group's game you can not get a mark for it.)
    - You can not bring any reference materials except what is in your code, but you can feel free to document your code as much as you want.

Final Demo

- On the last Thursday of class:
  - Team Lead 6’s ensure you game is available so that so that everyone can play it.
  - Have fun!
  - Focus on ensuring that the “public” enjoys playing each game.
  - Invite your friends!
- After class fill in the feedback from on Canvas.

Post Mortem

- This is a group presentation: Everyone must participate if they want to share in the mark.
- This is a timed presentation: I will have a stopwatch and you will lose marks if you either go over or under.
- The focus of the presentation is on how you would improve the process if you were to do your project over. Talk in terms of coupling vs cohesion in how you project was split up between your members.

See Canvas for the specific marking key and stopwatch (in the dropbox).

2 Minute Set up, 12 Minute Presentation

User Manual How to play the game

- Slide your phone left and right to move the character forward and backward in the game
- Tilt your phone upwards to make the character jump