

Rules & Spec

Introduction to Programming (I) Final Project

Outline

Rules

Project introduction

Grading spec

Outline

Rules

Project introduction

Grading spec

Rules

- 1 Person per Group
- Worth 35% of your final grade!
 - 10% Checkpoint 1 **(2025/11/09 Sunday)**
 - 10% Checkpoint 2 **(2025/11/24 ~ 2025/11/25 During Lab hours)**
 - 15% Checkpoint 3 **(2025/12/21 Sunday)**
- We will ask some questions in Checkpoint 2 and 3
- No score will be given if you don't attend
 - Please tell us in advance if you can't come, so you can reschedule!

Rules

Hackathon

- Must be done before 2025/11/09 16:00

Checkpoint 2

- Take Home Assignment
- Demo during Lab hour 2025/11/24 ~ 2025/11/25
- Submit to eeclass before 2025/11/23 23:59

Checkpoint 3

- Take Home Assignment
- Demo at 2025/12/21, Sunday (Time Schedule TBA)
- Submit to eeclass before 2025/11/23 23:59

Rules

- Use our template as a starting point, don't make it from scratch!!
- Only Python language is allowed
 - PyGame and PyTMX libraries will be use, if you want to use other library, contact TAs through eeclash discussion
- Discussion with your classmate is allowed, but do not directly copy the code
- We will check your code similarities at the end for submission. If the similarities is high, we will directly consider it as plagiarism.
 - Try to use your own coding style
 - Using LLMs is not recommended, If your code matches with someone else due to LLMs, we still count it as plagiarism!

Outline

Rules

Project introduction

Grading spec

Project Introduction

Inspired by 2D Pokemon game in the past



Project Introduction

Our Knock-Off : **Monster Go**



Project Introduction

- **Required**

- **Python 3.12.8 :** <https://www.python.org/downloads/release/python-3128/> (We will run your program using this version!)

- **Recommended**

- **Git:** <https://git-scm.com>
 - You can do fork and clone it directly if you understand git
- **VS-Code:** <https://code.visualstudio.com>
 - A cool text editor, designed for coding

Project Introduction

Project Website: <https://github.com/Skivap/NTHU-I2P-I-Final-Project-2025>

The screenshot shows the GitHub interface for the repository 'NTHU-I2P-I-Final-Project-2025' by user 'Skivap'. The repository is private and has 0 watches, 0 forks, and 0 stars. The main content area displays a list of files and folders with their commit history. The files include 'assets', 'exercise', 'saves', 'server', 'specs', 'src', '.gitignore', 'README.md', 'main.py', 'requirements.txt', and 'server.py'. The 'About' section on the right provides a description of the project as the 'Final Project for National Tsing Hua University Introduction to Programming 2025 Fall'. It also includes links to the README, activity, stars, watching, and forks. The 'Releases' section indicates that no releases have been published, and the 'Packages' section indicates that no packages have been published. The 'Contributors' section shows two contributors: 'Skivap' and 'kennethzhang31'. The 'Languages' section is partially visible at the bottom.

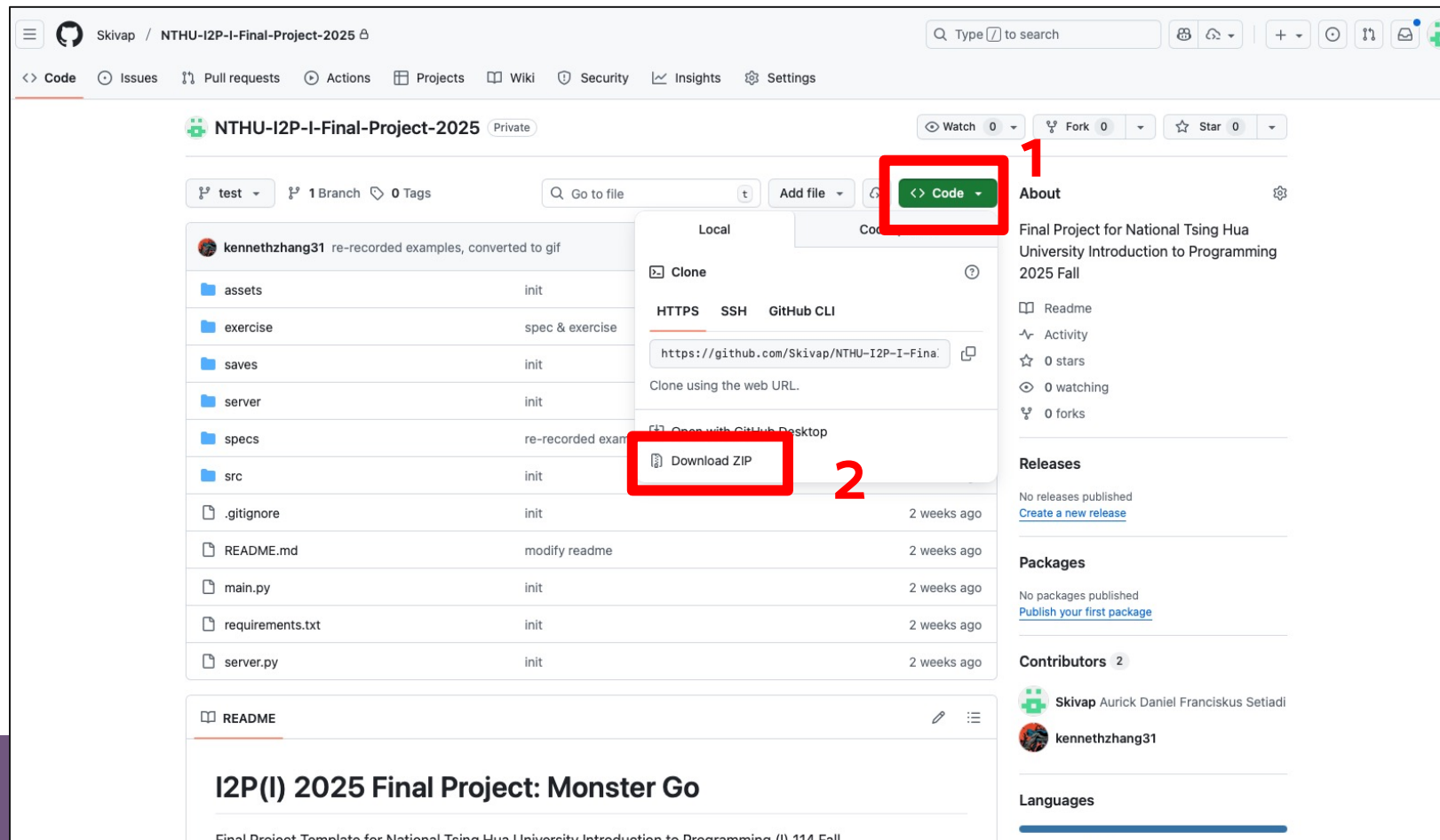
File/Folder	Commit Message	Commit Time
assets	init	2 weeks ago
exercise	spec & exercise	2 hours ago
saves	init	2 weeks ago
server	init	2 weeks ago
specs	re-recorded examples, converted to gif	40 minutes ago
src	init	2 weeks ago
.gitignore	init	2 weeks ago
README.md	modify readme	2 weeks ago
main.py	init	2 weeks ago
requirements.txt	init	2 weeks ago
server.py	init	2 weeks ago

README

I2P(I) 2025 Final Project: Monster Go

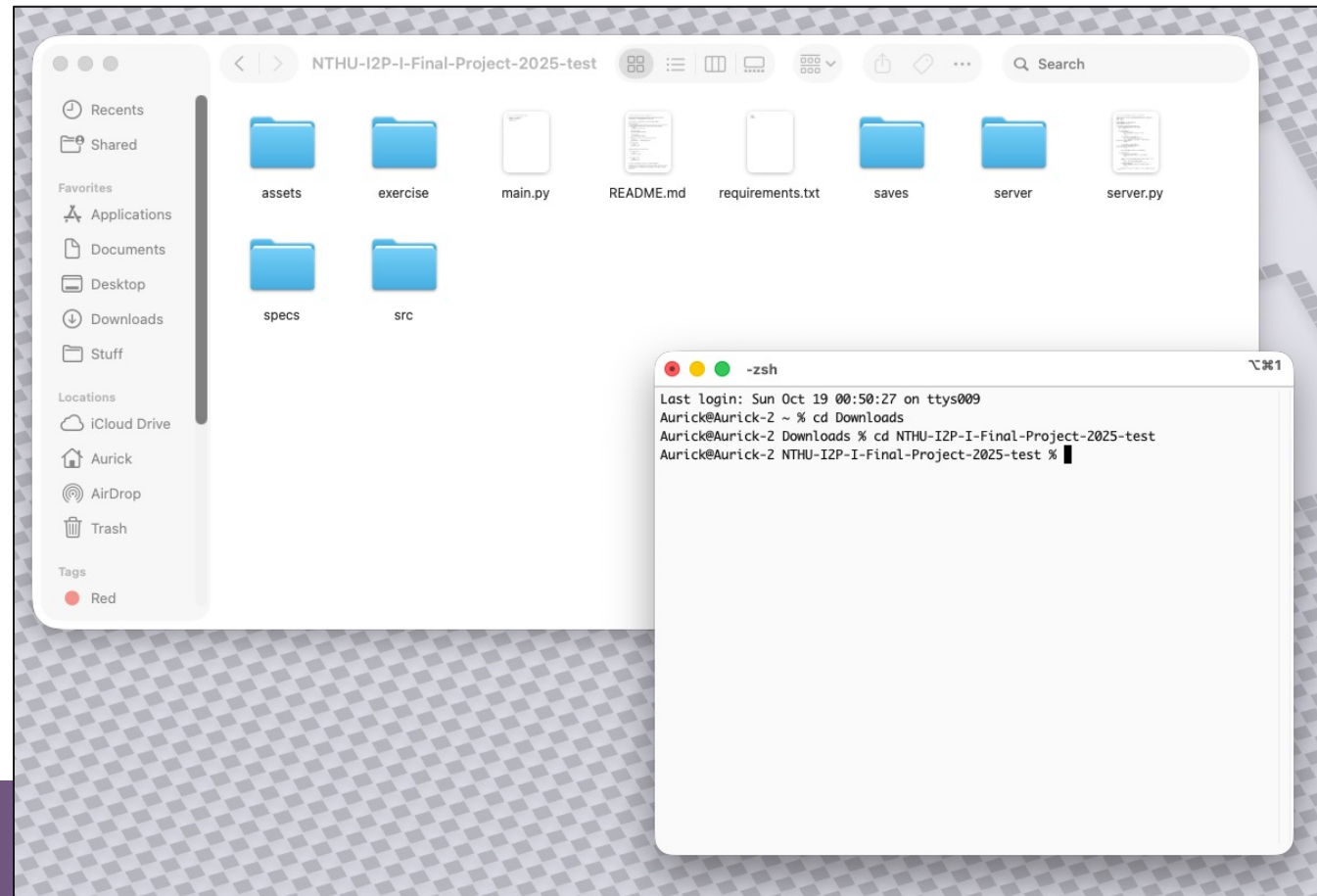
Project Introduction w/o Git

Download the source code



Project Introduction w/o Git

Go to project directory in terminal



Project Introduction w/o Git

- Create virtual environment for the project inside project directory

```
$ python3.12 -m venv .venv
```

bash/cmd/PowerShell

```
$ source .venv/bin/activate
```

bash

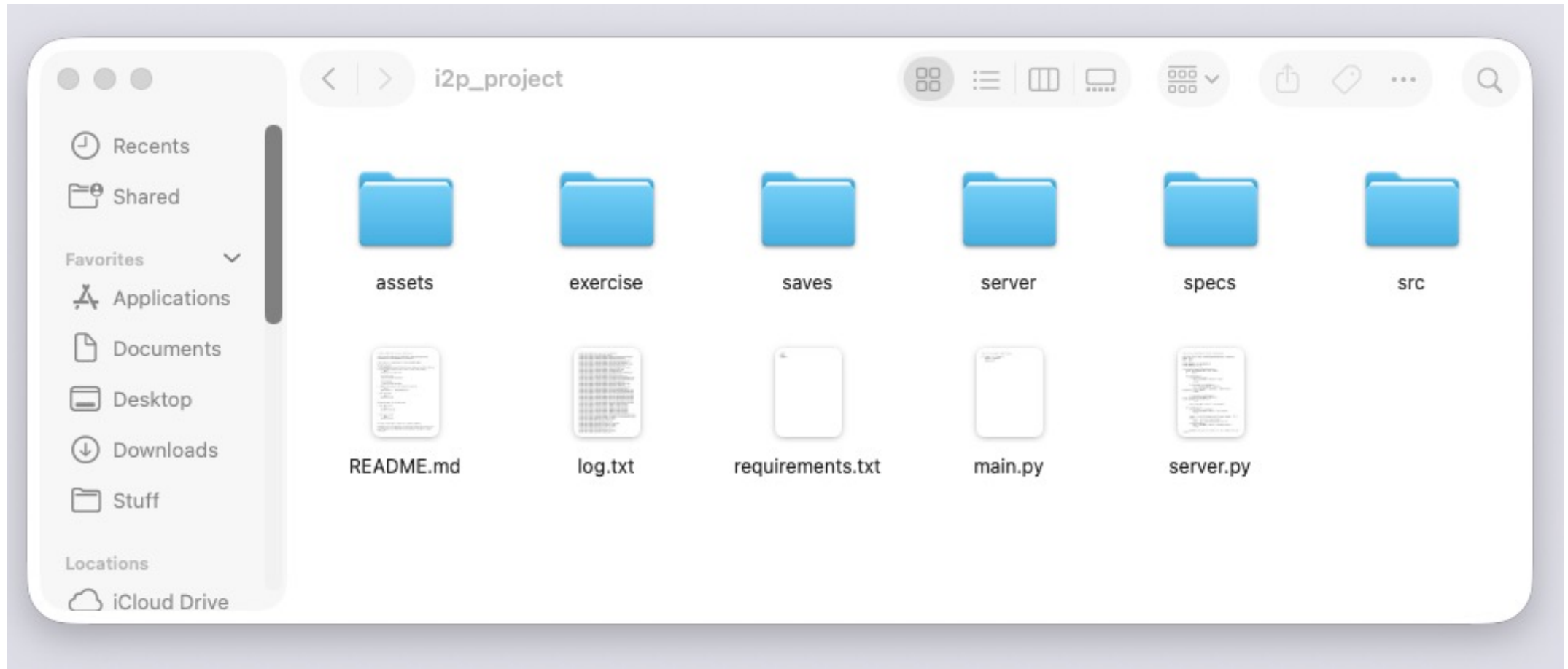
```
.venv/Scripts/activate
```

cmd/PowerShell

```
$ # Install libraries dependencies  
$ pip install -r requirements.txt  
  
$ # Run your program  
$ python main.py
```

bash/cmd.PowerShell

Project Introduction



Project Introduction: Online

- It's optional if you want to create it as online game
- We did the basic setup for you!

```
$ # Run the server  
$ python server.py
```

```
$ # Run your client (you can do it multiple times in different terminal to test)  
$ python main.py
```


Outline

Rules

Project introduction

Grading spec

Grading Spec

- Specified in **spec** folder
- Or you can open website below:
- <https://github.com/Skivap/NTHU-I2P-I-Final-Project-2025/blob/main/specs/main.md>
- If it's unclear you can ask TAs (only during Hackathon) or post it on eeclass discussion
- (NO EMAIL)

Grading Spec

Skivap / NTHU-I2P-I-Final-Project-2025

<> Code Issues Pull requests Actions Projects Wiki Security Insights Settings

test NTHU-I2P-I-Final-Project-2025 / specs / main.md

Skivap spec & exercise 4a428cd · 4 hours ago History

Preview Code Blame 63 lines (49 loc) · 3.1 KB

Project Specs Overview

Click a link in the Description column to open the detailed spec for each item. Each checkpoint totals to the score shown on its header row.

Main rule

- You can discuss with your friends, but make sure to write the code yourself
- We will check your code for plagiarism based on code similarity. Make sure to have different coding styles and approaches if you use some references to avoid plagiarism.
- If we detect any plagiarism, you will receive a 0 score for the entire project.
- If the game crashes when doing specific TODO, you will receive a 0 score for that TODO.
- Each checkpoint has their own deadline. No points will be given if it exceed the deadline.

Checkpoint 1 | Hackathon (Max: 10pts)

For each TODO, no partial points will be given if you do not complete it. Score is valid if you complete it within the hackathon day.

TODO	Description	Score
1. Button	Button Spec	2
2. Player Movement	Player Movement Spec	1

Grading Spec

test

NTHU-I2P-I-Final-Project-2025 / specs / main.md

63 lines (49 loc) · 3.1 KB

Raw

- You can discuss with your friends, but make sure to write the code yourself
- We will check your code for plagiarism based on code similarity. Make sure to have different coding styles and approaches if you use some references to avoid plagiarism.
- If we detect any plagiarism, you will receive a 0 score for the entire project.
- If the game crashes when doing specific TODO, you will receive a 0 score for that TODO.
- Each checkpoint has their own deadline. No points will be given if it exceed the deadline.

Checkpoint 1 | Hackathon (Max: 10pts)

For each TODO, no partial points will be given if you do not complete it. Score is valid if you complete it within the hackathon day.

TODO	Description	Score
1. Button	Button Spec	2
2. Player Movement	Player Movement Spec	2
3. Camera Control	Camera Control Spec	1
4. Collision	Collision Spec	2
5. Setting Scene	Setting Scene Spec	2
6. Teleport	Teleport Spec	1
7. Attendance	Attendance Spec	1

Checkpoint 2 (Max: 10pts)

Make sure you have completed all the TODOs in Checkpoint 1 before starting Checkpoint 2.

Some of the TODOs may have their own prerequisite. If there is a prerequisite and it's not full score, it will be marked directly as 0

Click for more detail

Grading Spec

Preview Code Blame 10 lines (6 loc) · 263 Bytes

Player Movement Spec

Score: 1 point

Keyword: [TODO HACKATHON 2]

☐ User can press WASD or arrow keys to move the player

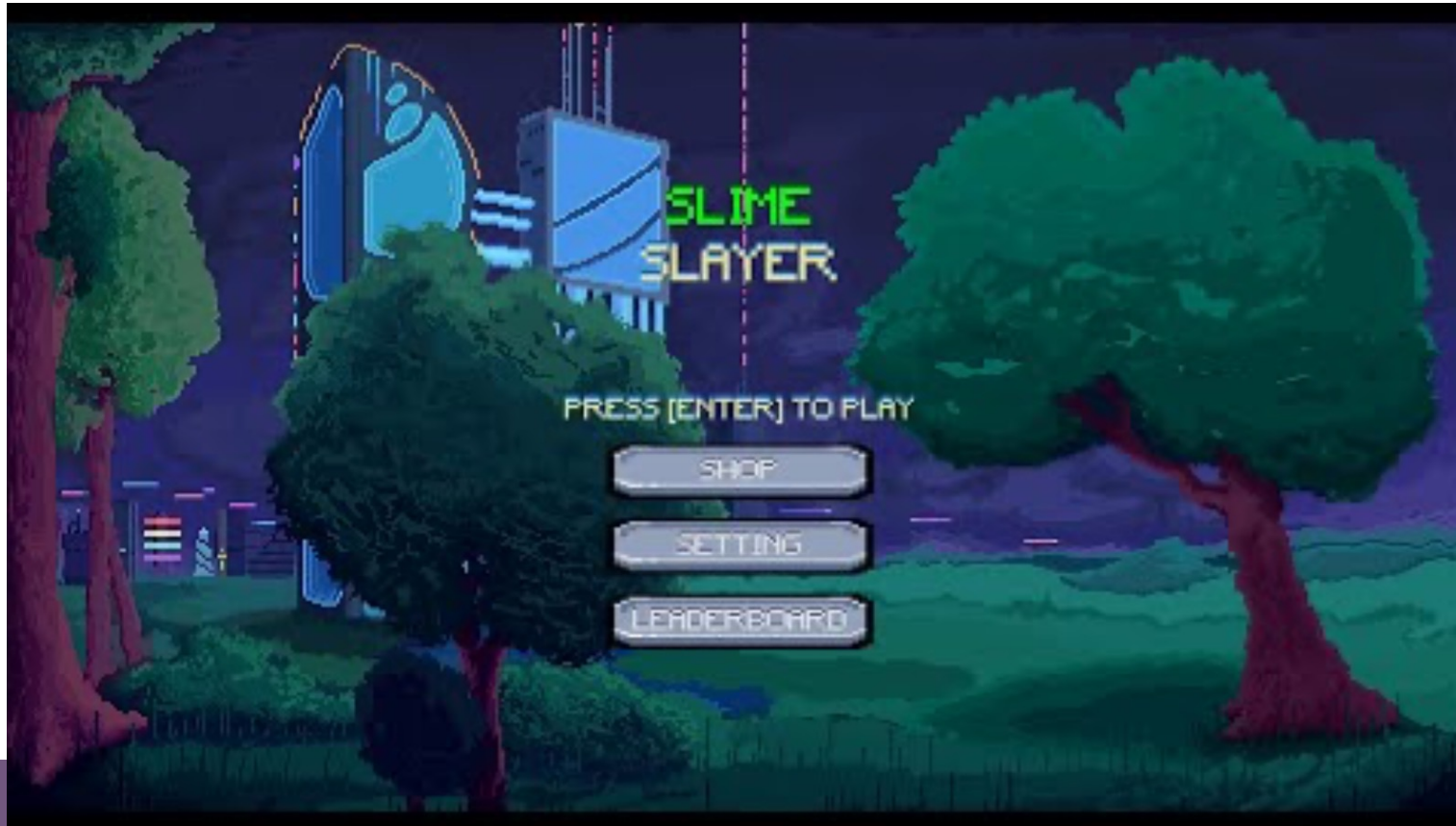
☐ Movement must be normalized (walk diagonal can't be faster than walking straight)

Grading Spec

- **Some TODOs have prerequisite TODOs.**
 - **If the prerequisite TODO is not full score, the dependent TODO will be mark as 0**
- **If you got full score for the Final Project and your Project is outstanding, a prize will be given**
- **So, make sure to become creative!**

Grading Spec

Last year theme: RPG



Grading Spec or Project Question

**Please ask on eeclass discussion!!!
(except if it's a personal question)**