Unity Programmer Task

We appreciate your interest in the position and your enthusiasm for our upcoming projects! As part of our interview process, we assign a task to assess the participant's skills, creativity, and compatibility with our dynamic team. Team building is fundamental to our values, and we are seeking developers who bring exceptional ideas and seamlessly integrate with our team.

We look forward to seeing how your talents can enhance our team and shape the future of our game projects!

Introduction

We appreciate your interest in participating in this programming interview task. This assignment should be completed within **48 hours** of receiving these instructions, allowing for a fair evaluation of all candidates.

If you require an extension beyond the initial 48 hours, please notify us in advance. Keep in mind that the assessment considers development time, so only request additional time if you are confident it will significantly improve the result.

The primary objectives of this task are to evaluate the your abilities in the following areas:

- Unity and C# (Project must be developed using Unity version 6000.0.35f1)
- Code organization and file structuring
- Adherence to schedule
- Ability to complete tasks with open-ended instructions

For this specific interview, it is crucial to consider both design and aesthetics in addition to the playability of the prototype you create. Although this is primarily a programming test, we value programmers with a basic understanding of game design and art aesthetics. Ensure that your prototype reflects attention to all aspects, not just the code.

Requirements

Before you begin, please choose whether your prototype will be in 2D or 3D. Select the format that best showcases your abilities.

- Utilize either pre-made art assets from the Unity store or create your own for this task
- Utilize GitHub for the interview, making incremental commits during development instead of one large commit at the end.

- If using pre-written code from other projects, clearly specify the sections implemented during the interview
- Aim to deliver the highest quality game possible, as quality is a priority for our team. Thoroughly playtest the game before submission.

Instructions

---

1. Gameplay

- Implement character movement logic.
- Implement character animations.
- Implement character interaction with the world (e.g., picking up items, talking to NPCs).

2. Inventory Structure

- Implement a **UI slot-based inventory**.
- Implement interactions for:
    - Adding items to the inventory.
    - Removing items from the inventory.
    - Moving items within the inventory.
    - Dragging and swapping items between inventory slots.
    - Using or equipping items (e.g., consuming health potions, equipping weapons).

3. UI Design

- Ensure the UI dynamically updates based on inventory contents.
- Design a **clear, intuitive, and visually appealing** inventory UI.
- Show item details when **selected or hovered over** (via tooltip or dedicated panel).

4. Save and Load System

- Implement a **save system** for inventory state.
- Implement **loading** of inventory data when the game starts.
- Ensure **slot-based persistence** for each item.

Please note that any code generated during this interview will not be utilized in the development process at the company.

Suggested Additional Features

(Extra features, not mandatory)

1. World Building

- Elevate your world by showcasing your world-building and level design skills. Consider adding elements like NPCs, sounds, VFX, and any other extras to make it truly captivating.

Ensure these core functionalities are robust and polished. However, don't hesitate to go beyond the essentials by incorporating additional features that can showcase your creativity and deeper understanding of Unity's capabilities. These extra touches will not only demonstrate your technical proficiency but also your ability to deliver a comprehensive and engaging user experience.

Submission Guidelines

- The game should be executable and uploaded to GitHub for easy playtesting. Ensure that the GitHub link is set to public, not private. Also, please send a zipped copy of the .exe file to the email address at the bottom. Name the file as '**YOUR NAME_TASK**'.
- Provide a 300-word document explaining the system, your thought process during the interview, and a personal assessment of your performance. Attach this document as a PDF on GitHub.

Once completed, send the link to the **interviewer email** and mail it to the company's email: **hiring@ngp.careers**

Within the next few days, we will provide feedback along with the results. We wish you the best of luck and encourage you to enjoy the process!