

**Team Zero**

Sad Pancake

Request for Proposal  
Version 1.5

Document History

|  |  |  |  |
| --- | --- | --- | --- |
| Version | When | Who | What |
| 1.0 | 1/29/2018 | Josue Espinosa, Jorge Olivas,  Alex McNurlin,  Zane Durkin,  Simon Barnes,  Hayden Lepla Tori Overholtzer | Initial Drafting |
| 1.1 | 1/31/2018 | Alex McNurlin | Add more content |
| 1.2 | 2/7/2018 | Zane Durkin | Adding Glossary terms |
| 1.3 | 2/8/2018 | Alex McNurlin | Add similar systems |
| 1.4 | 2/9/2018 | Josue Espinosa | Condensed RFP 1.0-2.0 |
| 1.5 | 2/9/2018 | Tori Overholtzer | Completed 7.0 Project Schedule |

Table of Contents

1. Problem description
2. Project Objectives
3. Current System(s) – if any  
   or similar systems
4. Intended users and their interaction with the system
5. Known interactions with other systems inside/outside the client organization
6. Known constraints to development
7. Project Schedule
8. Glossary of terms
9. Problem description / opportunity / expression of need

The University of Idaho is looking for a team that will design and develop an interactive 2D platformer made in Unity. The project will need to include elements from external sources. The game must fulfill all criteria listed below. Proposals must be submitted via email to mcnu5088@vandals.uidaho.edu by the date listed in the project timeline below.

1. Project Objectives
   * Level Design
     + A level design which offers replayability:
       - A collectible currency should exist in-game
       - There should be collectable tokens in each level that the player can collects– these should be used to unlock an in game object or item
       - There should be an in-level scoring system
     + There will be enemies in game that the character the player is playing as can interact with
     + The player should act in different ways depending on what material they are standing on (sand – slow down, sharp rocks – take damage, etc.)
     + The level should take the player in four directions – up, down, left, right
     + Moving platforms should be implemented
     + Levels will be unlockable: beating the level n-1 to unlock level n
   * Input
     + The game should accept user input
     + The user should be able to customize their controls
   * User Interface
     + Start Screen
     + Profile selection / creation
     + Pause Menu
   * Characters
     + A main character the player can interact with
     + Interactable non-player characters that affect gameplay
   * Game Camera
     + Intuitive camera that allows the user to view their position in the world
   * Audio
     + Sound effects for
       - Character movements
       - Selecting items
       - Picking up collectables
     + Menu / loading screen music
2. Current system(s) – if any / similar systems

Similar Systems to the desired game are 2D side-scrolling video-games where the player fights enemies and bosses to complete various levels. This includes:

* + Super Mario Bros
  + Kirby's Dreamland
  + Yoshi's Island

1. Intended users and their basic interaction with the system

The intended users for the video game are students from the University of Idaho, team zero members, the instructor for CS383, and a general public audience. All user interaction with the game should be via a desktop computer or laptop capable running the Unity executable.

1. Known interactions with other systems within or outside of the client organization.

The System must be able to run on a Windows 10 environment and must be able to utilize the keyboard for user input.   
The System must be able to run independently from any other programs on a given windows computer.

1. Known constraints to development

The System must be developed on Unity version 2017.03.0f3

The developers must use GitHub to keep a cloud-based repository of code

The System must be developed for Windows 10

Coding style must be follow the guide lines of 'Allman` coding style.

Function / variable naming must follow the C# Microsoft Standard Conventions

1. Project Schedule

All proposal made in response to this RFP are to be submitted by February 12nd, 2018 at 5pm PST. Proposals submitted after this time will not be accepted.

The submitted Proposals will be Evaluated February 13rd, 2018 until February 14th, 2018. If any additional information is needed from bidders during this period, the bidder(s) will be contacted.

The decision process for the winning bidder will be conducted immediately following the evaluation period and will be concluded no later than February 15th, 2018. All bidders will be notified of their bid’s acceptance or rejection by February 15th, 2018.

Contact negotiations with the winning bidder will begin immediately upon notification of their bid’s acceptance.

The Project Timeline will be as follows:

February 15th:  The code repository will be selected, and its directory structure defined. Client, Julie Beeston will be notified of the repository status. Class Diagrams, Sequence Diagrams and a Gantt timeline will also be created. By February 22nd there should be skeleton code for each proposed feature. The equivalence of a make file and public header files for C# shall also need to be established by this date. By March 1st, two functioning compile paths should be available in the repository, one for testing and one for production. A major test plan will also need to be established by this date. By March 15th a budget analysis needs to be provided. Gantt chats shall be finalized by this time. By March 29th a document shall be assembled containing the coding standards. By April 5th a document shall be assembled containing the user manual for the project. By April 12th a post mortem document will be completed and by May 3rd the final project will be completed.

1. Glossary of terms

*Run: To execute the program on the user's computer*

*System: The executable video game*

*Platformer: Game style where user's character jumps from platform to platform as they progress through the level*

*Project: The entirety of the Game and its components listed under section 2.0*

*Unity: A game design engine and integrated development environment*

*Replayability: Game style that offers incentives playing portions of the game which have already been beat*