



Lake Runner

Request for Proposal

Version 1.0

Document History

Version	When	Who	What
0.1	Feb. 4th, '23	Logan Finley	Initial Drafting.
0.2	Feb. 6th, '23	Logan Finley	Worked on Section 2, 3, and 9.
0.3	Feb. 6th, '23	Jenna-Luz Pura	Worked on formatting and Section 10.
0.4	Feb. 6th. '23	Ian King	Sections 1, 6, 9.
0.5	Feb. 6th. '23	Nathan Nguyen	Continued Section 2.
0.6	Feb. 6th. '23	Akhil Karri	Continued Section 2.
0.7	Feb. 7th, '23	Jenna-Luz Pura	Worked on Section 3.
0.8	Feb. 7th, '23	Logan Finley	Worked on <i>items</i> part of Section 2 and Section 5.
0.9	Feb. 8th '23	Logan Finley	Finished Section 4.
1.0	Feb. 8th, '23	Zaiden Espe	Finished Section 3.

Table of Contents

- 1.0 Problem Description
- 2.0 Project Objectives
- 3.0 Current System(s) – if any or similar systems
- 4.0 Intended Users and their Basic Interaction with the System
- 5.0 Known Interactions with Other Systems Within or Outside the Client Organization
- 6.0 Known Constraints to Development
- 7.0 Project Schedule
- 8.0 How to Submit Proposals
- 9.0 Dates
- 10.0 Glossary of Terms

1.0 Problem Description

Clear Lake Studios is looking for a group of ambitious developers to create a novel, creative, and exciting 2D runner game. Here at Clear Lake Studios, we believe that individuals around the world can connect and bond over their passion for video games. In the spirit of this vision, we wish to come out of the gate with a brand new concept for a runner-adventure game with a non-player controlled (NPC) lead character.

This game should draw on inspiration from drawing and platformer games to create a new experience in which the player does not control the lead character, but rather the world around them. This control may be done through platform creation, powerup use, or any other means the developing group may come up with! The most important thing is that this game embodies a spirit of excitement and adventure for players around the globe.

Interest in this product should be garnered primarily through creative mechanics and design choices, creating a cohesive product that draws in players to a new and engaging game world. Game mechanics should be smooth, accessible, and have a gradual difficulty curve. Design choices should be bright, fast-paced, and unique to enthrall new and continuing players alike.

2.0 Project Objectives

Clear Lake Studios is looking to publish a single player 2-D video game with unique and innovative platforming mechanics. To match our vision for this game, we would like any interested developers to submit game proposals that include the following features:

- 2-Dimensional
 - The game should be a 2-dimensional sidescroller. This means that the camera should show a side view of the character; the jumping corresponds to a character moving upward on screen, and then falling down. As the character travels left, right, up, or down the camera will move to follow and reveal more of the level. These are important ideas because they enable the core mechanics of platform drawing (explained below).
- Game Levels
 - Should contain pre-generated level terrain that the main character runs on. There should also be puzzles that the player has to solve in order to protect the character. Items can be pre-generated around the map or spawn randomly based on the player's current inventory.
- Items
 - Sunscreen
 - In this game, the character should be taking constant damage from the sun. To protect the character from taking sun damage, the character will need to be able to pick up a sunscreen item. The sunscreen item will fill up a sunscreen bar UI element. While the sunscreen bar UI element is not empty, the character will not take damage, but the UI element will deplete slowly. This item will be stored in the user's inventory after being picked up. A maximum of three sunscreen items can be in the player's possession at any given time.
 - Aloe Vera
 - The character will have a health bar, and there will be multiple ways for their health bar to be depleted. Some examples are taking damage from enemy entities and running out of sunscreen. The *aloe vera* will act as a health potion and replenish a portion of the character's health bar. The *aloe vera* model can take a similar shape to the sunscreen item, but it doesn't have to. This item will

be stored in the user's inventory after being picked up, and can be used at any time. A maximum of three *aloe vera* items can be in the player's possession at any given time.

- Brain Blast Bar
 - Since the player has no direct control over the movement of the character, it should ideally be difficult for the player to pick up every item in a level. To help the player overcome this difficulty, the *brain blast bar* will reverse the direction of the player's movement for a period of time, allowing them to pick up any items they may have recently missed. This item will be stored in the user's inventory. A maximum of two *brain blast bar* times can be in the player's possession at any given time.
- Sunglasses
 - This item will zoom the camera out and also give the player unlimited building materials for a specific period of time. This will give the player the ability to look ahead in the level and possibly place platforms in complex portions of the map. This item will be stored in the user's inventory system. A maximum of one *sunglasses* item can be in the player's possession at any given time.
- Platforms
 - Platforms will be drawn by the player via a click-and-drag mechanic. The platforms will have a rectangular shape, but the dimensions will be determined by the size of the box the player draws. There will be a limit on the size the player can make the platform. The amount and total area of the platforms the player can draw will be limited by a "platform resource" that the player does not collect, but slowly increases as the game progresses, but is depleted as the player draws platforms. The platforms will float in the game where they are drawn, but additional features such as gravity-affected platforms, moving platforms, and breakable platforms are possible.
- NPCs
 - Crocodiles
 - Crocodiles or similar themed enemies should damage the lead character if they are allowed to get close. These will act as hazards that the player must take steps to avoid or risk damage or game over.
 - Crab

- Crabs or similar themed enemies should steal an item from the player. If the player does not take steps to avoid these, then player options provided by items would quickly become limited, though this will not cause a game over.
 - Falling Coconuts
 - A hostile entity that falls at a constant rate in the direction of the character. The purpose of this entity is to prompt a response from the player to create a platform that protects the character.
 - Lead Character
 - The camera should follow the character at the same speed, but certain items may be able to manipulate the framing of the camera (for example to zoom out, which may offer an advantage to the player).
- UI
 - Building material - The player should be displayed the Building material he collected throughout his play on the HUD.
 - Health - The player should be displayed the amount of health he is having on the HUD
 - Sunscreen - The player should be displayed the amount of sunscreen he collected on the HUD.
 - Interactive Menu Display.

3.0 Current System(s) – if any / similar systems

Platform Drawing System

There are multiple examples of games in which the player may draw items or platforms to be inserted into the game world. The classic sandbox game “Line Rider” is a prime example of this mechanic, with players drawing platforms for a sledder to move along. In our version of this system, Clear Lake Studios hopes to see a feature in which the player may draw platforms that either float in the air or fall down to land. The NPC lead character may then run across these platforms to reach their eventual goal.

Platformer System

The platformer system is a common video game design. The platformer system is the base for how the rest of the systems, features, and aesthetics are designed. The platform system will determine the pacing, combat mechanics, and adventurous tone of the game. The camera should follow the player as they move through the map quickly in one direction, forcing the user to react and interact to the oncoming environment just as fast. An inspiration to the platformer system Clear Lake Studios wishes to create is Sonic The Hedgehog 2 (1992). The platform system is vibrantly colored, the screen is filled with visual inputs for the player to parse, the player moves quickly through the environment, the camera follows the player, and there are non-stop threats and obstacles for the player to deal with. A large difference between the platform system Clear Lake Studios wishes to create and the Sonic the Hedgehog 2 game is that in our system, the main character will not be controlled by the user, but will move automatically through the environment. The player will interact with the platform system by being forced to create dynamic platforms to allow the main character to travel through the environment quickly.

Inventory System

A core mechanic to this game is the interaction between the player and different items. In order to make the game more fulfilling, an inventory system is required to allow the player to store items for later. An example of this system is found in the game *Minecraft*. In this game, the player has a hot bar that is populated with different quantities of different items. Clear Lake Studios would like to see a similar hotbar inventory system implemented so that the player can easily select items to be used in the level. Please note

that we are not requesting a separate menu that the player can open using a keyboard shortcut; all items should be represented in a hotbar-like inventory system.

Overworld System

An overworld system connects the levels within the game. Games that follow a storyline are likely to have levels that build upon another or follow a certain sequence of events. Having an overworld system is similar to having a map, as it shows the sequencing in which levels have to be completed. Each Super Mario Bros game contains many different worlds, each with a certain amount of levels. These worlds and levels have to be played in a particular order and likely ends with Mario facing this nemesis Bowser. Clear Lake Studio wishes to incorporate this system into our new game. The NPC lead character will complete levels to run between lakes across the United States, ultimately arriving at the long-desired Malibu Surfrider beach in California.

4.0 Intended Users and their Basic Interaction with the System

Users

- People who enjoy novel and unique gameplay mechanics.
- Those interested in supporting independent video game development.
- Anyone who wants a casual and rewarding gameplay experience.
- Past, present, and future CS383 students

Uses

- Players can only interact with the environment around the character.
- Input will be received through a keyboard and mouse, gamepad controller, or mobile device touch screen.
- The main goal of interacting with the environment is to protect the character and help them reach the end of the level.

5.0 Known Interactions with Other Systems Within or Outside of the Client Organization

Unity Game Engine

We require that the game be developed using the Unity game engine. No other game engine is necessary, and so any submission the requests the use of another engine will not be accepted.

GitHub

The game files should be hosted privately on a GitHub repository to allow for easy collaboration and review. Once selected, we will request that the winning team invites Mr. Morton Thumbo to the repository using the email inbox@ClearLakeStudios.com.

External Game Stores

It should be trivial to port and host the game on the following game store platforms:

- Steam
- Epic Games
- Origin
- itch.io
- Google Play Store

6.0 Known Constraints to Development

Funding

Clear Lake Studios will supply up to \$20,000 for the development of this project, to be allocated as is appropriate for personnel payment and basic upkeep. Based on this funding level, the project is expected to take no more than three months. Additionally, it is expected that all game assets and resources shall be free or custom-made, as Clear Lake Studios will not approve funding for any unnecessary expenditures.

Game Engine

The proposal winners will be expected to build and run this entire game project in the Unity game engine as Clear Lake Studios has already established and maintained company standards utilizing this engine.

Performance

This game should be constructed to allow users on basic and/or weak hardware to execute it. Developers should be careful to minimize code and objects in the Unity project.

7.0 Project Schedule

Date	Description	Notes
Jan 24th, 2023	Initial group meeting	Git repository created and team member responsibilities dished out
Feb 14th, 2023	Systems analysis presentation and demo	PERT and GANTT charts prepared
Feb 16th, 2023	Game development progress report 1	Basic structure laid out
Feb 23rd, 2023	Game development progress report 2	Fundamental movement/collision mechanics implemented
March 7th, 2023	Game development progress report 3	Fundamental environment mechanics implemented
March 23rd, 2023	Game development progress report 4	Base game compiles and runs
March 30th, 2023	Game development progress report 5	Demo of project compiles and runs
April 6th, 2023	Game development progress report 6	Full game compiles and runs, may need polish
May 2nd, 2023	Post Mortem Presentation	Incorporating fixes from feedback session
May 4th, 2023	Final presentation and demonstration	

8.0 How to Submit Proposals

Please email all proposals to Morton Thrumbo at inbox@ClearLakeStudios.com, including the phrase “[Lakerunner Proposal]” in the subject line. Please send any questions to the same email and include the phrase “[Lakerunner Question]” in the heading. Note that all questions will be answered and sent out to all proposal respondents.

9.0 Dates

All proposals must be emailed to Morton Thrumbo by Thursday, February 23, 2023 at 5:00pm PST. Any proposals submitted after the deadline will be ignored. All applicants will be notified by February 27th, 2023 by 5:00pm.

10.0 Glossary of terms

Term	Definition
Demo	A demonstration of a product.
PERT	“Program Evaluation Review Technique” is a management tool to structure an upcoming project.
GANTT	A project management tool that demonstrates the division of time spent on a project.
NPC	Non-player character.
UI	User interface.
HUD	Heads up display. Essentially just that the status bars the player sees.
2-D	2 Dimensional

Respondents Proposals Criteria

Category		Weight
1	Platformer Mechanics	14%
2	Platform Drawing Mechanics	16%
3	Dynamic NPCs	12%
4	Unique and interesting items	12%
5	Overworld Map	8%
6	Inventory System	8%
7	Interesting Story	8%
8	Responsive UI	12%
9	Local Business Bonus	10%
Total		100%