Wentworth
School of Computing & Data Science

Volcano Island

Game Design Document

Micah Schuster

Project Description

This game design document describes the details for the game *Volcano Island.* This is a real time obstacle avoidance game where the player controls a character and attempts to dodge moving objects.

1. Gameplay 1

1.1. Goals 1

1.2. User Skills 1

1.3. Game Mechanics 1

1.4. Items 2

1.5. Progression 2

1.6. Losing 2

2. Art Style 3

2.1. UI 3

3. Music and Sound 5

4. Technical Description 6

5. Other 7

Note:

There are several sections that we don’t need for this game:

- Characters
- Story
- Marketing
- Funding

If your future games (for this class or elsewhere) have need for these elements, be sure to include them.

1. Gameplay

The following section outlines the main gameplay elements of *Volcano Island*. At a fundamental level, the player character, a snowman, moves left and right to avoid obstacles, fireballs, that fall from above the play area.

1.1. Goals

The player will attempt to avoid obstacles in the game area for as long as possible.

1.2. User Skills

Volcano Island requires the following skills for success:

- Basic Keyboard Usage
- Predicting Future Trajectories

1.3. Game Mechanics

- Player Movement
  - The left and right arrow keys are used to move the player in the left and right direction, respectively.
  - There is no acceleration when beginning or ending movement. The player will always be stopped or moving at full speed.
  - The player cannot go below ground level, which is above the bottom of the play area, and cannot leave the left or right edge of the play area.
  - The player's collision zone will be follow the outline of the main body and head of the snowman.
- Obstacle Spawning
  - The obstacles will spawn above the play area and fall straight to the ground level.
  - The points where obstacles spawn from are fixed and spread out along the width above the play area.
  - As the game progresses, the spawn rate will increase to a maximum value.
  - The obstacles will be circular, have a constant size, approximately the same width as the player.
  - The collision zone for the obstacles will be circular and approximately the same size as the obstacle graphics.

1.4. Items

There are no items currently planned for *Volcano Island.*

1.5. Progression

Difficulty during the game will increase based on the amount of time played. *Volcano Island's* difficulty is determined by the spawn rate of the obstacles. This rate will slowly increase as the player successfully dodges obstacles. The maximum spawn rate represents the maximum difficulty that the player will encounter.

1.6. Losing

The player's health will be shown clearly in the UI and will be capped at three hearts. When the player collides with an obstacle, e.g. a fireball, one heart will be removed from the current health. When the player's health reaches zero, the game is over.

Note:

This is your main description of how the game will play.

Be Explicit!

Leave no room for confusion, especially if you are working in a team. This will help speed up development and keep everyone on the same page.

Be cognizant of your time restrictions for development and don't include more features than time allows. It's better to do one feature/mechanic really well than to do many mechanics poorly.

2. Art Style

The art style for Volcano Island is simple 2D art that is drawn in a vector application. All art is shaded as if the light is coming down from directly above.

Player Character:

- Snowman with two segments.
- Accessories include a top hat, buttons, nose and eyes.
- Snowman will have two animations, idle and moving.
- Snowman will "sweat" constantly.

Obstacles:

- Obstacles will appear to be fireballs.
- Fireballs will emit a fiery trail as they move toward the ground.
- Upon impact, fireballs will appear to explode.

Background:

- There will be a static background of mountains in the distance.
- There will be dark clouds overhead.

Foreground:

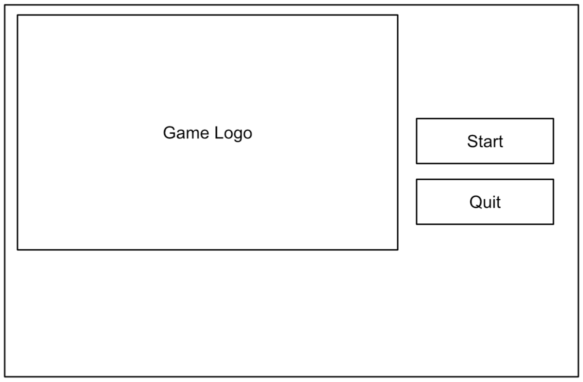
- The only foreground elements other that the obstacles and player is the ground that influences the fireball impact that restricts the player from moving up or down.

2.1. UI

There are three aspects to the user interface:

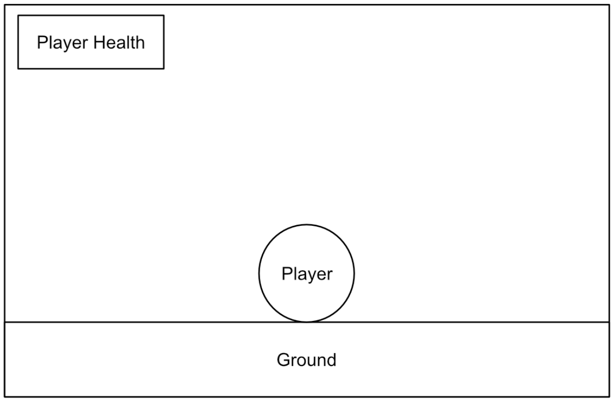
Main Menu:

- Contains the logo for the game and two buttons, Play and Exit.



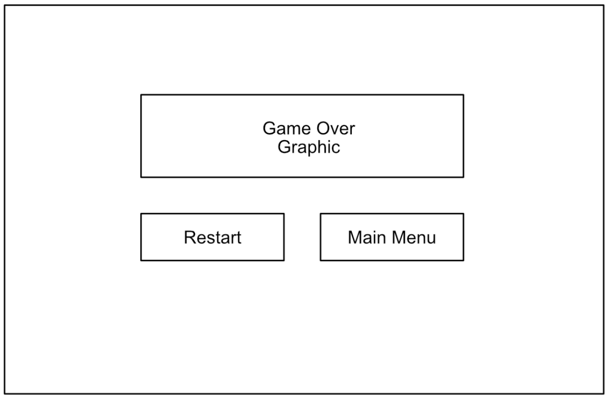
Gameplay:

- During regular gameplay, there will be minimal user interface.
- A heart icon and a number are shown in the upper left corner of the scree to represent the remaining player life.



Game Over:

- When the player health reaches zero, the game over screen will be displayed, freezing all action in the main game.



Note:

This section should outline the way that you want the game too look. This includes the style of the characters, animations, visual effects, etc. Also think about including information about how the UI will look.

Although the look of the game and some of the specifics may change throughout development, it's good for everyone to have a solid sense of the game's atmosphere.

Very often, the are and UI displayed in this section are based on concept art or basic visualization.

3. Music and Sound

Music:

The in-game music will be a background drum beat with a simple, repeating melody. In general, the music will be happy as the user plays the game. There will be no changes in the music as the game progresses.

Sound:

The sound effects will be minimal, only occurring when the player is hit by a fireball.

> Note:
>
> Use this section to describe the themes and genre of the in-game music and sounds. Be sure to include information about changes in music or sound based on situation (location, health, battle, etc.).
>
> *Volcano Island* is a very simple game, so this section is not extensive. However, for your own games, be sure to be as descriptive as possible.

4. Technical Description

Volcano Island will be built and tested for the following operating systems:

- Windows
- macOS

The game will be created using Unity game engine and the C# language.

The following software will be used to create the assets for the game:

Art:

- Affinity Designer
- Affinity Photo

Music:

- Bosca Ceoil

SoundFX:

- Audacity
- bfxr

Project management and organization will be handled by Trello and git will be used for version control.

Note:

Here you are just giving a summary of the platforms that you are targeting and tools that you will be using. Typically, the GDD is accompanied by a Technical Design Document that contains a more detailed description of *how* the game will be designed and implemented.

See: https://www.studytonight.com/3d-game-engineering-with-unity/tdd-and-gdd

5. Other

Dash Mechanic:

Upon pressing a key, probably space, the player will move in the facing direction very quickly. There will be a short cooldown and a subtle visual effect (particles) during the dash.

“Smart” Fireballs:

Rather than the fireballs going straight down after being spawned, they should target the current location of the player or slightly ahead of the facing direction of the player.

Extra Health:

The player should have the ability to “catch” additional heath containers that spawn in a similar way to fireballs.

Note:

These are other ideas that are either part of a wishlist (if you have development time) or ideas that don’t quite fit anywhere else.

Use this as a place to record ideas that may or may not end up in the game.

Be careful of feature creep!

These particular ideas are part of your homework assignment for this game.