

METEOR MEHEM

DESIGN DOCUMENT

By Kris for the 2024 Pirate Software Game Jam

Introduction

<i>Game Summary</i>	-----	2
<i>Inspiration</i>	-----	2
<i>Player Experience</i>	-----	2
<i>Platform</i>	-----	3
<i>Development Software</i>	-----	3
<i>Genre</i>	-----	3
<i>Target Audience</i>	-----	3

Concept

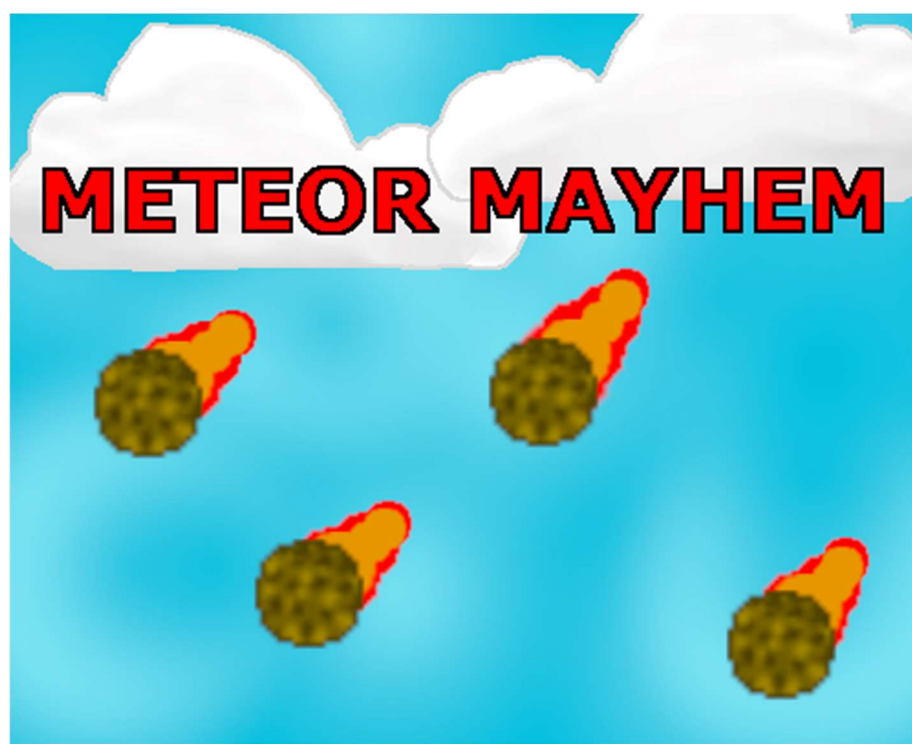
<i>Overview</i>	-----	3
<i>Theme Interpretation</i>	-----	4
<i>Primary Mechanics</i>	-----	4
<i>Secondary Mechanics</i>	-----	5

Art / Audio

<i>Design</i>	-----	5
<i>Sound Effects</i>	-----	5

Game Experience

<i>UI</i>	-----	6
<i>Controls</i>	-----	6



Introduction

Game Summary Pitch

Meteor Mayhem is an arcade style shooter about defending the planet from meteors, while keeping track of how many you have destroyed.

Inspiration

I pulled inspiration from most all arcade shooters that I have found provide a fun and casual experience, while also allowing for more focused and score-based gameplay when desired. Examples of games I pulled inspiration from include Space Invaders, Brick Breaker, and Ball Shooting.

Player Experience

In a single game window, the player will be shown the controls and thrown right into the action. While shooting the meteors out of the sky the player will notice they sometimes split and the scale of the danger spreads rapidly. To successfully manage amount of meteors that are falling, the player must focus fire to attempt and destroy meteors, collect power-ups, and plan where to position themselves to channel the meteors into a more manageable area.

Platform

The game is developed to be released on windows PC.

Development Software

Python 3.11.7 and PyGame for programming

GIMP for graphics and game art

Audacity and Free Sound Effects for Sound Effects

Genre

Arcade, 2D, Shooter, Casual

Target Audience

Given the nature of this game the audience is someone who is looking for a fun game in which to sink 10 – 15 minutes and have fun competing with themselves and seeing how far they can get.

Concept

Gameplay Overview

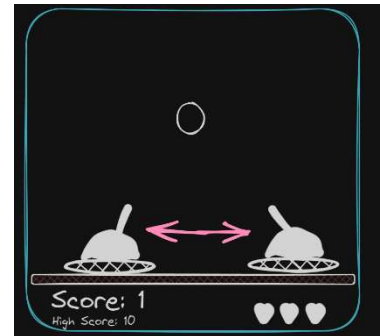
The player controls a tank in which they can move left and right on the map. While this is not required adjusting the position of the tank can allow for the player to push the meteors to one side of the screen and make it easier to destroy them. The player can also collect power-ups to aid in the defense and help them achieve a higher score.

Theme Interpretation (It's Spreading)

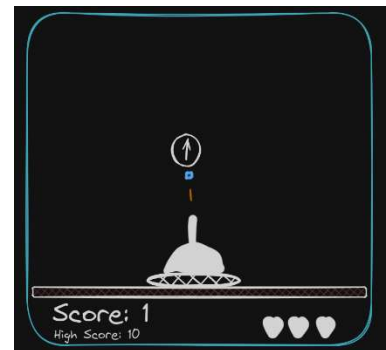
It's Spreading – As the game progresses the number of meteors in the sky will split and spread if not properly managed. Even if the player does everything right if RNG says there will be more, then there will be more.

Primary Mechanics

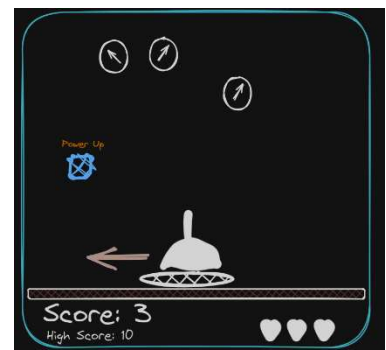
Movement – The player can move Left and Right to allow them to change the angle in which their rounds will impact the meteor.



Shooting – The player will be able to aim and fire using their mouse. This allows for both movement and aim affecting the possible trajectory of the meteor.



Power-Ups – The player will be able to collect Power-ups that will allow for them to have Rapid-fire, freeze the meteors in place, or Gain more health.

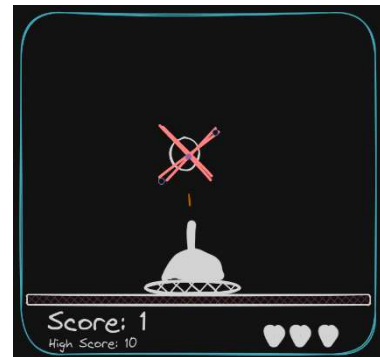


Secondary Mechanics

Meteor Splitting – The meteors will always split if less than four exist and then have a 1 in 6 chance of splitting after that.



Meteor Exploding – If a meteor is hit more than five times it will explode and be deleted from the screen.



Art / Sound

Design

The art for this game is pixel art to help give the feel of a casual arcade shooter. It is also colorful and not limited to further project the casual nature of the game.

Sound Effects

The game will not have music and limited sound effects to give off the energy of an old arcade shooter.

Game Experince

UI

The UI for this game will be simple and consist of the players score and health located in the ground below the players tank. The only other UI element in the game will be a line that appears at the top of the screen to denote where the meteors are at once they are above the limit of the screen.

Controls

Keyboard – [A] and [D]

Mouse – Right-Click and Mouse movement