Proposal for Orbital 23

Team Name:

Project Sage

Proposed Level of Achievement:

Artemis

Motivation

We are **curious about the game development scene** and want to utilise Orbital as an opportunity to **learn the relevant technologies and processes** involved in creating a production-ready game that is **extendable and battle-tested**. Through our game, we also hope to provide a whole suite of player-oriented features to value-add and create an overall positive gaming experience that is engaging and fulfilling.

Aim

For end-users:
- Create a game that will be **engaging** and **provide much needed respite** from work as well as serve as a platform for **creating a community** based on the shared experiences from having played the game.
- Taking inspiration from the classic Chinese novel Journey to the West, we aim to bring our players into the mysterious world of Chinese fiction, populated by the Chinese pantheon and all sorts of magical creatures and monsters, full of conflict and drama in between them. Through the game play, we hope the players can enjoy themselves in the Chinese mythical realm and emerge more knowledgeable of the Chinese **cultural elements** that are heavily depicted in the fiction and incorporated into our game.

For the developers (us):
- Pick up **software engineering patterns and principles** and apply them to create an extensible production-ready game
- Utilise **comprehensive testing frameworks** to produce code that is battle-tested and bug-free
- Learn more about **game design** and what is involved in making a good and enjoyable game
- Have a better knowledge of our ethnic cultural elements as reflected in the fiction novel Journey to the West from which we take inspiration for our game.

Game Loop

1. Players clear chapters which are adapted from Journey to the West
   a. The gameplay is in the form of a real-time combat roguelite where players control a set of two (swappable) characters to clear mobs and eventually the boss of the chapter
   b. From each chapter, players can:
      i. obtain new equipment
      ii. unlock new characters
2. The goal of the game is to get through all the chapters which signifies completing the "Journey to the West"
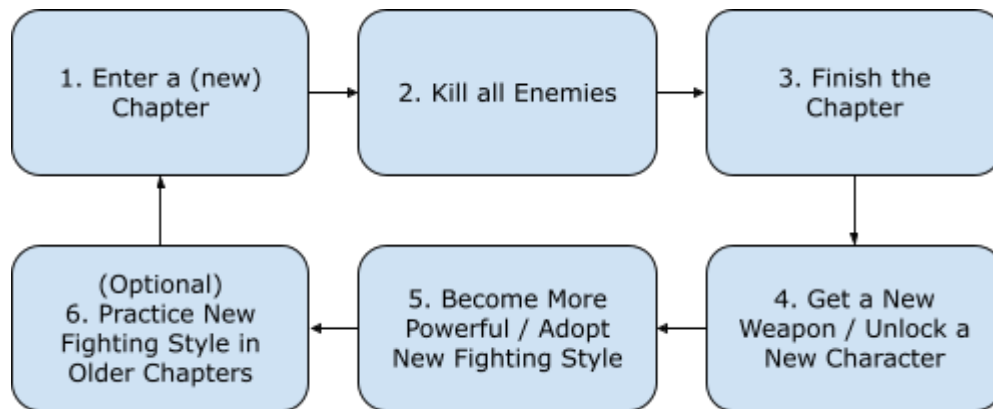
Game Loop Illustration

User Stories

- As a novice player, I can expect there to be an easy-to-follow tutorial and intuitive UI to allow me to easily be familiarised with the various aspects of the game.

- As an intermediate player, I can play through chapters with unique mechanics and difficulties to keep me engaged in the game and its concept

- As an advanced player, I would continue to be challenged and engaged by new chapters with unique enemy mechanisms and new bosses.

- As a general player, I can expect there to be rewards for clearing chapters to acknowledge the time and effort committed to learn and play through chapters.

Planned Implementation Details:

Game Title	Odyssey
Game Genre	Adventure, RPG, Roguelite
Game Theme/Inspiration	Journey to the West (a classic Chinese mythology novel, from which we will model our game after)
Perspective	Third-person
Movement and Combat Control Style	Top-down
Chapter Map Layout	Room-based (the player has to clear all enemies in a room before being able to move into another)
Art Style	Pixel art

Features

Features to include:

- Movement System
    - **8-directional movement** coupled with **4-directional sprite animation** inspired by the movement mechanics of **Moonlighter**

- Health System
    - The player controls **two characters** and each character has a given health entering a chapter, which can be healed by **in-game pickups**

- Weapon System
    - Each weapon carries **distinct damage values** but **do not change** the **mode of attack** which is specific to each character
    - No in-chapter weapon switching

- Combat System
    - **Melee combat** but the player can **switch** between two characters in-battle, each with their **unique attack range, damage** and **frequency** as well as **special attacks**
    - Combat in 4-directions

- Enemy System
    - Enemies with **themes** and **mechanisms** unique to each chapter adapted from *Journey to the West*

- Save System
    - Players have the option of **saving progress between chapters** (but not in chapter)

- Chapter System
    - Each **chapter** is **completed** when **all the enemies are killed**, awarding the player a new character or new weapon

- Map System
    - **Procedural Dungeon Generation:** Inspired by games like **Binding of Isaac**, each chapter will have a **fixed number of rooms** but have their **relative position determined randomly** each time the player enters the chapter and

the boss room will only show up after a **predetermined** number of rooms have been **cleared**. Thus providing **novelty** and **uniqueness** to each chapter run.

Systems listed below will only be added if those above have been properly developed and polished:

- Enemy Reward System
    - Each enemy defeated will drop certain buff items that either increase attack damage or heals the character. Alternatively, the item dropped can also be coins that can be used to purchase weapons or potions in the shop.

- Shop System
    - Weapons with different designs and animation will be sold here. However, they only vary from the default weapon of each character in terms of their damage values, they do not change the attack mode of the characters.
    - Potions effects including healing and temporary damage boost would also be sold here.

- Potion System
    - Before setting off for a chapter, the player has to decide the number of potions for them to bring and use in the chapter. The player would not be able to purchase additional potions inside the chapter itself and the potions unused at the end of a chapter would be wasted.

Graphic Illustration of Game Aspects

- Sample Art Style:
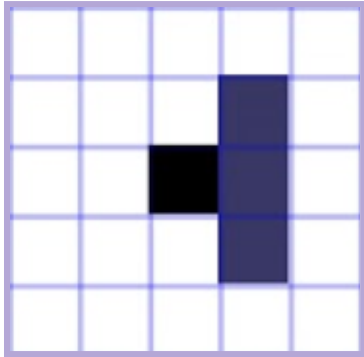

Moonlighter

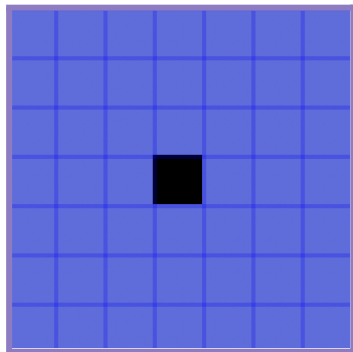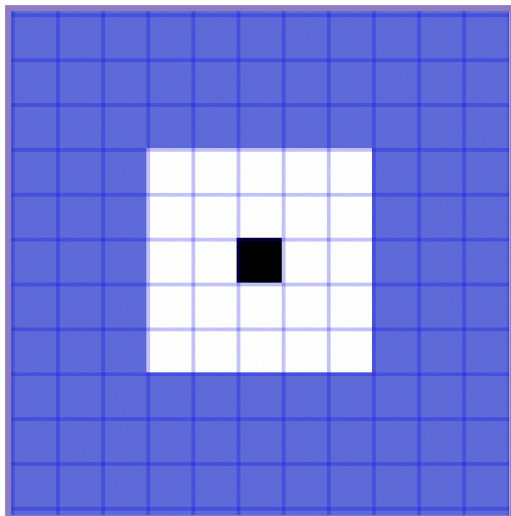- Sample Combat Animation:
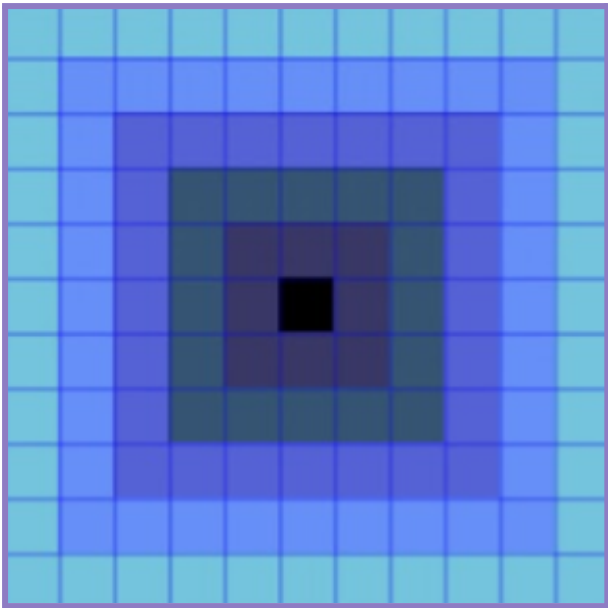
Moonlighter

Sample Attack Modes

Character Swapping: Inspired by mechanics in games like **Genshin Impact**, we intend to introduce the ability for players to **swap between two characters** within the chapter itself.

To incentivize players to swap between characters, we intend to assign a **unique set of attacks** to **each character** which will be suitable for **certain scenarios**, **rewarding players** for **effort** spent on learning the **mechanics of available characters**.

Attack Mode of Character 1 (Sword-like Style)	
■	Character Position (1 * 1 grid)
■ ■ ■ ■ ■	Attack Range (1 * 1 grid) (Blocks with Higher Colour Intensity Represents Higher Damage)
Range of (Right) Basic Attack	Range of Every Third Basic Attack
Range of Special Attack	

Attack Mode of Character 2 (Limited Range Projectile Style)	
■	Character Position (1 * 1 grid)

		Attack Range (1 * 1 grid)
Range of both Basic Attack and Every Third Basic Attack		

Basic Attack	One projectile released per attack, in the direction the character is currently facing, however the projectile can only reach enemies within the purple range above. The projectile delivers damage and slows the enemy down.
Every Third Basic Attack	A projectile bigger than the basic attack projectile will be released, with the same slowing effect but lower damage.
Range of Special Attack	

Special Attack	All enemies within the purple attack range will be frozen for 2 seconds.

- Sample Game State Bar

Game State Bars showing current character, health and energy

	Character Icon: shows the character that is currently used in combat. It would show the icon of another character once that character is summoned to replace the current one.
	Health Bar:each character has its own health bar, distinguishable by colour. When one character's health bar diminishes to zero, that character can no longer be used throughout the current chapter.
	Energy Bar: both characters share the same energy bar. The energy will increase gradually as time passes during combat. The energy is used for switching between characters and for the characters' special attack. This means when the energy is low, the player can neither switch between characters or use special attack.

Proposed Chapters

Chapter I	
Chapter Title	Land of Woman
Mechanic	**Seduction:** Mobs possessing this power can affect the player's control of the character
Enemies	**Mobs** Seductive Maidens are female monsters who possess the power of **seduction** **Boss** The Hundred-Flower Maiden is the boss of this chapter, with the ability to summon seductive maidens as well as enhanced seduction (same effect but amplified in strength and duration)
Chapter II	
Chapter Title	Land of Darkness
Mechanic	**Darkness:** Player's field of vision is reduced significantly for all stages/rooms in the chapter
Enemies	**Mobs** Shadow Demons are demons known for their ability to blend into the darkness and strike when players least expect it **Boss** The ruler of the Land of Darkness, the King of Darkness is the boss of this chapter, with control over shadows and darkness.

Timeline & Development Plan

Sprint	Date	Targets	Remarks
1	08 May - 14 May	1. Team Advisor Consultation 2. Liftoff Submission - Poster and Video 3. Storyboarding - Sample Themes from *Journey to the West* 4. Gameplay Ideation - Brainstorming Game System Designs 5. Tech Stack Familiarisation (Unity & Aseprite) 6. Asset Design	
	15 May - 21 May	1. Tech Stack Familiarisation 2. Asset Design 3. Player Movement System 4. Player Combat System (Basic Attack)	
2	22 May - 28 May	1. Player Combat System (Basic Attack + Avatar Swap) 2. Health System 3. Enemy System (Movement)	
	29 May - 04 June	1. Player Combat System (Avatar Swap) 2. Enemy System (Movement + Level 1 Attack) 3. Weapon System 4. GUI	
3	05 June - 11 June	1. Player Combat System (Special Attack + Finish-up) 2. Enemy System (Level 1 Boss) 3. Weapon System 4. Map Design	
	12 June - 18 June	1. Enemy System (Level 2 Movement, Attack and Special Attack) 2. Map Design 3. GUI 4. Save System	
4	19 June - 25 June	1. Enemy System (Level 2 Movement, Attack and Special Attack) 2. Map Design 3. GUI 4. Save System	
	26 June - 02 July	1. Save System 2. Quest System 3. NPC Town	
5	03 July - 09 July	1. Quest System 2. NPC Town	
	10 July - 16 July	1. Small Group Playtesting/feedback and Game Fixes 2. Polish Map Design 3. Polish Avatar Design	

		4. Polish Enemy Design	
6	17 July - 23 July	1. Small Group Playtesting/feedback and Game Fixes 2. Polish Map Design 3. Polish Avatar Design 4. Polish Enemy Design	
	24 July - 30 July	1. Large Group Playtesting/feedback and Game Fixes 2. Polish Map Design 3. Polish Avatar Design 4. Polish Enemy Design	
7	31 July - 06 Aug	1. Large Group Playtesting/feedback and Game Fixes 2. Polish Map Design 3. Polish Avatar Design 4. Polish Enemy Design	
	07 Aug - 13 Aug	1. Large Group Playtesting/feedback and Game Fixes 2. Finalise Game Elements	
8	14 Aug - 20 Aug	1. Large Group Playtesting/feedback and Game Fixes 2. Finalise Game Elements 3. Final Submission Preparation	
	21 Aug - 27 Aug	1. Large Group Playtesting/feedback and Game Fixes 2. Finalise Game Elements 3. Final Submission Preparation	

Tech Stack

[Please list down the technologies that you are planning to use.]

Technology / Game Engine
1. C#
2. Unity
3. Github or other Version Control Systems

Design / Animations
1. Aseprite
2. photoshop
3. Audacity (and freesound) for soundtracks
4. Blender
5. GIMP

Qualifications

While we have limited technical skills and expertise (especially in the field of game development), we hope to push ourselves and maximise our learning potential from the Orbital program by committing to the rigor and requirements of Artemis level:

Gavin

- Modules:
    - CS1101S, CS2030S & CS2040S
- Hackathons:
    - Solana Builder's League 2022 Frontend Tech Lead: https://surfprotocol.vercel.app/
        - Helmed frontend development for Surf, a software-as-a-service aimed at utilising AI to generate production ready smart contracts for Solana
    - NUS Fintech Month 2023 Frontend Tech Lead: https://pandabond-frontend.vercel.app/
        - Helmed frontend development for PandaBonds, a bonds-as-a-service concept aimed at leveraging on blockchain bonds as a means for SMEs and startups to have an alternative for raising capital
    - Bithack 2023 Fullstack Tech Lead: https://bithack-frontend.vercel.app/
        - Developed a full stack application including frontend in NextJS, backend in Flask, database with MongoDB and a working ML model for Hustle, a jobs-as-a-service solution targeting the retrenched in the tech industry by utilising ML to match applicants with jobs as well as to upskilling education opportunities
- Proficiencies:
    - Languages: Javascript, Java, Python
    - Frameworks: React, NextJS, ExpressJS, Flask
    - Other Technologies: MongoDB, REST API, GraphQL, Figma, TailwindCSS

Xinrong

- Modules:
    - CS1101S, CS2030S & CS2040S
- Projects:
    - CVWO 2023 Winter Assignment
        - Individual attempt at building an online forum with CRUD functions
- Proficiencies:
    - Languages: Javascript, Java
    - Framework: React