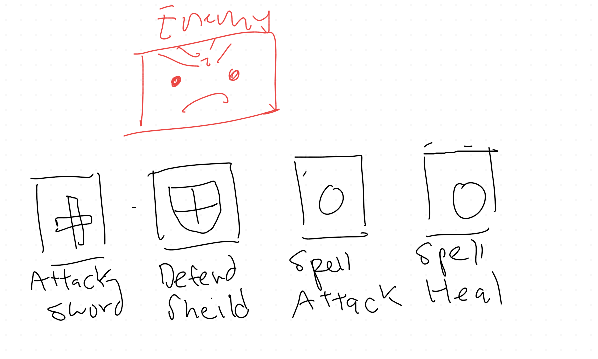
VR Battle game

Genre : Real-time stationary VR action game

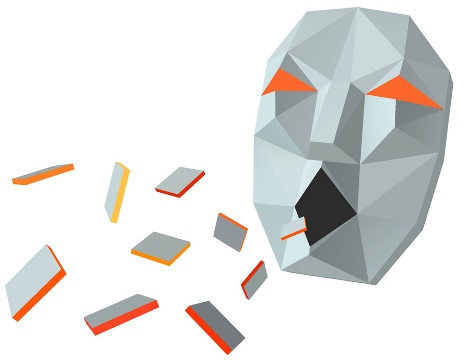
Game Description : Defeat enemies using magical cards that spawn different weapons and spells for you to use.



Player : VR Single player vs AI / Two player versus ( VR Single Player vs Mobile Player )

Gameplay

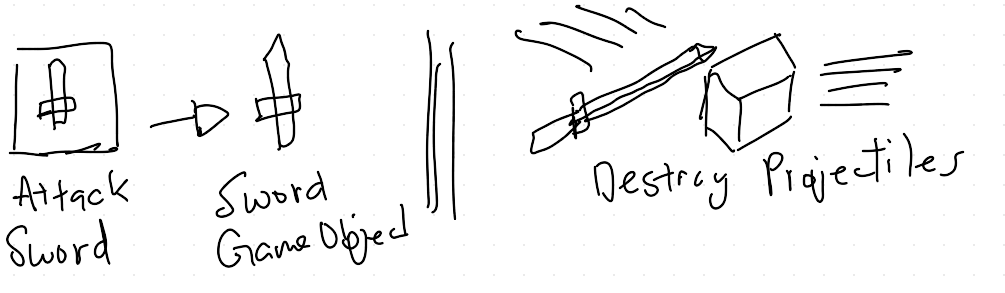
Enemy will constantly throw/summon/shoot objects to damage the player. Players need to dodge projectiles or receive damage.



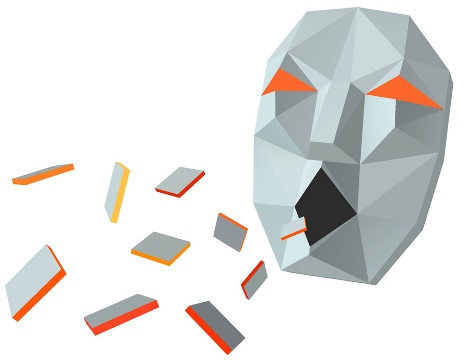
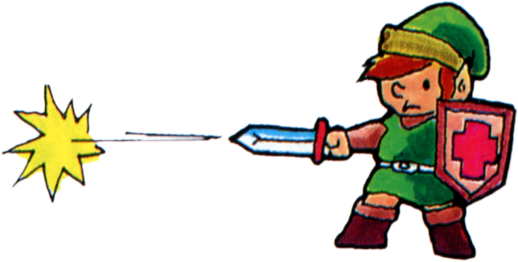
To win a game you need to deplete the enemy health. You use either Attack Sword or Spells to deplete the enemy health.

Use a card by picking it up. If you have enough mana to use the card after 3 seconds of holding the card, it will automatically triggers its card effect and transforms into a usable object for the player.

**Basic Attack - Sword**



The sword requires the player to charge it up by slashing projectiles sent by the enemy. After the sword is charged. Similar to beat saber.



Once charged the player aims the sword at the enemy to fire a laser type projectile to damage the enemy.

**Basic Defend – Shield**

Choosing the shield card gives you a shield. Limited number of blocks or damage that can be blocked. Each successful block rewards the player with more Mana.

**Basic Spell – Attack**

If you have enough mana you can use a spell orb. Simply throw it and it automatically homes in on the enemy.

**Basic Spell – Heal**

Use it to heal.

**Winning ! -**

Rewarded with the option to add a new card to the player deck to better defeat the next enemy.

Example : Stronger basic sword attack card with 3 damage instead of 1.

**Losing**

Start again at the beginning

Game System

**Game Manager –** Tracks game state. Gamehas four states. Game begins in the start state. Upon entering a new game state a c# event is called. Implemented a finite state machine to represent game state.

* Start State
  + Main menu screen
* Battle state
  + Battling enemy
* Win state
  + Just beat the enemy.
* Loss state
  + Loss to the enemy

**UI Manager –**

Responsible for handling the UI elements in the game and updating them. Needs to manage what UI shows up during the different game states.

* Start state UI
  + Show start Menu
* Battle state UI
  + Show Battle UI
  + Update UI during battle events
* Win state UI
  + Show Win state UI
    - Player rewards
    - Start next level
    - Return to main menu
    - Quit game
* Loss state UI
  + How Loss state UI
    - Retry
    - Return to menu
    - Quit game

**Enemy Manager**

Contains list of enemy characters prefabs and tracks **current level** in order to set the correct enemy for the level. Passes information to the **Enemy Generator.** Currently also manages the takeDamage and spendMana method for the enemy characters

**Enemy Generator**

Takes an enemyPrefab and spawns the enemy through instatiation.

**Deck Manager ( Player only )**

* Creates player card deck and puts it into a list.
* Shuffles the deck as well.
* Keeps track of cards in deck, hand, graveyard.
* Deck is created based on data from a scriptable object called deckData which contains a list of cards.
* Deck creates the Card Gameobjects based on what is in the player’s hand.

**Card Controller**

* Sets up card material based on card data
* Sets up card UI

Card Grab

* Manages VR interactions. Runs the methods to change the card material based on card state.

Card Material Controller

* Manages card material

**Player/Enemy Character**

Player and enemy both inherit from Character base. Contains character health/mana information. Player takeDamage and spendMana methods are on the player character.

**Enemy AI / Behaviour**

Thinking of using command pattern design. Create a list of attack commands and auto generate a list of commands for the Enemy to do.