



Algo Crypt

A Massively-Multiplayer Online Trading Card Game on the Algorand Blockchain

Overview

The blockchain hosts a bounty list of AI-generated enemies. Players build a deck of NFT cards to challenge them in a card game to the death. Players browse the list and select enemies to hunt, with the first player to defeat an enemy rewarded with an NFT representation of the defeated enemy as well as a random selection of cards from that enemy's deck. Players can trade enemy and card NFTs on an associated marketplace, collecting cards to use in PvE or PvP battles. Cards and enemies will have associated rarities and varying values.

The AI component includes use of GPT-3 to generate stat blocks for enemies (strength, intelligence, dexterity) from a thematic prompt. Then, the stat block and theme are passed to DALL-E to generate an image

to represent that stat block.

The blockchain component runs on Algorand. All cards and enemies are represented as NFTs using Algorand standard assets. The blockchain also validates games and winnings as a ledger of games and actions. The card and enemy marketplace will run on this blockchain as well using the CRPT asset. The use of blockchain also allows for validation of physical games with the introduction of a smart playmat and physical cards in a future project.

Instructions

Installation

Dependencies

The blockchain component of this project demo uses algokit, which requires Docker. Please install Docker Engine before attempting to run this demo.

The API and associated frontend require the Node Package Manager and React.js

The game and algorand components require Python 3.10+

FOR WINDOWS USERS: the bootstrap scripts are written as linux shell scripts. For maximum ease of use, use WSL with this project.

Install and Setup

Start by cloning this repository.

```
git clone git@github.com:AI-and-Blockchain/S23_Algo_Crypt.git
```

In the root directory, run

```
./bootstrap.sh  
./run_frontend.sh
```

This script will install all python dependencies, start an algokit localnet sandbox, and deploy the Membership contract that controls card distribution and game membership. You will then be redirected to the localhost port 3000 index. Add "auth" to the end of the URL to reach the signup page. After this script completes, in a new terminal, run the API.

```
./run_api.sh
```

To run the game, in the original terminal, run

```
./run_game.sh
```

This will open the signup page on port 3000 of localhost.

Create Account

To create an account, simply enter the details on the signup page and submit.

How to Play

- On startup you will be shown your player image, the enemy's image, both health bars, relevant stats, and a draw and discard pile.
- To start the game you can press the space bar to draw cards into your hand.
- You will be shown five cards and can toggle which cards you want to play by pressing the 1,2,3,4,5 keys (Each card in your hand corresponds to a different number)
- If you press a key to play a card it will show up in the play area where three cards can be played and you will see your attack and defense updated towards your left
- If you did not want to play that card you could press the number you originally pressed to play it to put it back in your hand.
- Once you have 3 cards played you cant press enter/return to actually play your cards.
- You will see what cards the enemy played and their stats updated as well and see how much damage both you and the enemy took from the cards played.
- If both of you still have enough health points to go on you can press the space bar again to draw cards and start the next round of play.
- If you have 0 health points and the enemy has a number greater than 0 you lose, and you will lose a certain amount of CRPT.
- If you the enemy has 0 health points (even if you reach 0 health points at the same time), you win the game and receive an NFT of the enemy and a subset of their deck.

Browse Marketplace

Technology Stack

Technologies

- GPT3
- Dalle
- Algorand

Frameworks

- FastAPI
- Postgres DB

Library dependencies

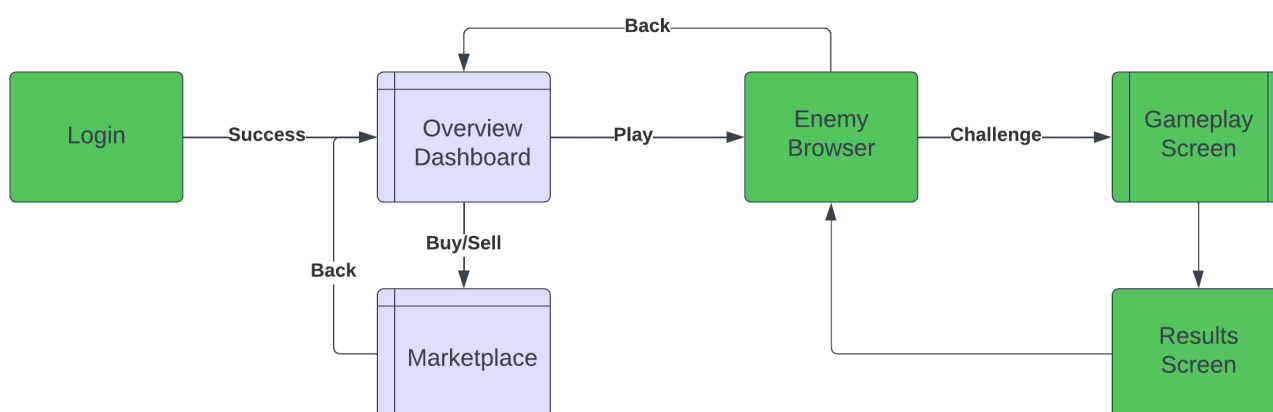
- PyGame
- PyTeal

Project Structure

```
- root
  - algo_crypt
    - game
    - contracts
    - api
    - marketplace
    - asset_generation
```

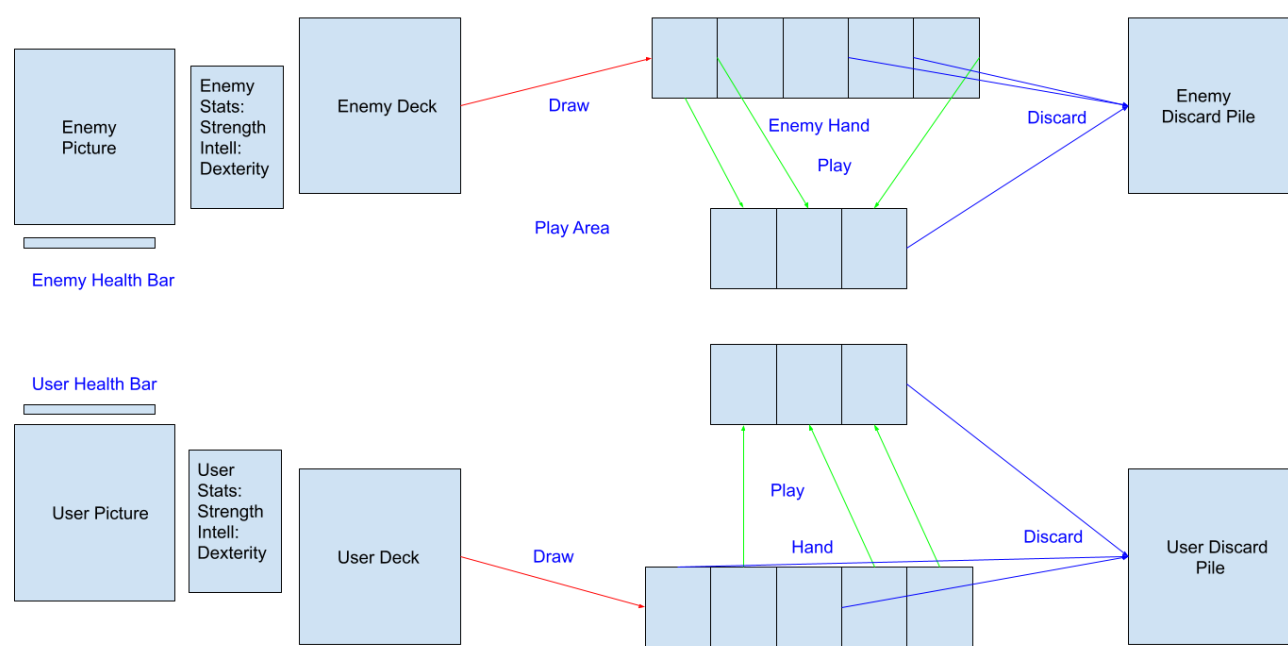
System Design

Overview



The project consists of the following subsections:

Gameplay



DEFEAT

YOU LOSE ____ CRPT!

VICTORY

YOU WIN:

Enemy NFT

Enemy Cards

New users will have 30 basic cards to start the game. After entering a battle with an enemy, you will have 5 cards in hand (5 is the upper limit for cards in hand). You and the enemy will take turns to play cards. On each turn, you can only play up to three cards, the rest will go to discard pile (after you run out of cards in hand and deck, your discard pile will be shuffled so that you can keep using these cards.) The player and the enemy will each have 30 health points (HP). The aim of the game is to make you enemy's HP become 0

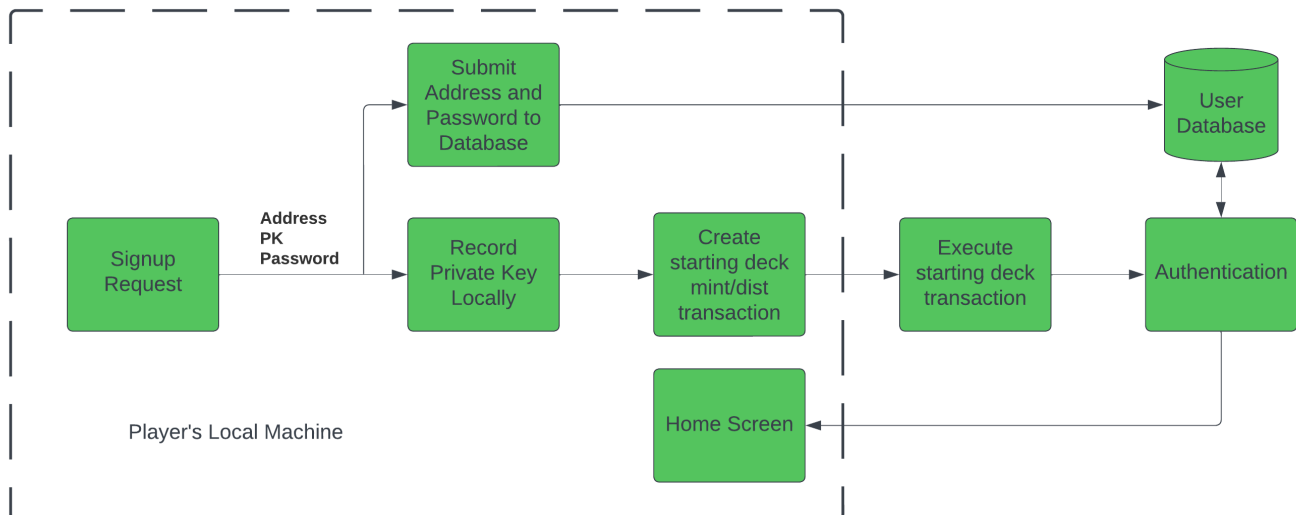
before he defeats you. After one player played one to three cards, it will become the other player's turn, and he should react to previous cards, and play his cards. When a player is out of cards in hand, deck, and discard pile, he is going to lose HP on each turn incrementally (-1 hp, -2 hp, ...). Challengers will always play first.

Each card consists of an ability(attack, block, dodge). Attack and block cards have physical and magical properties. One's attack and block is scaled by their strength and intelligence properties respectively. An enemy with more strength will have greater physical attacks and blocks while an enemy with more intelligence will have greater magical attacks and blocks. The attack value will become direct damage toward a player; in this case, the player can play block cards to decrease the damage or use dodge cards to avoid the attack.

NFT Minting and Distribution

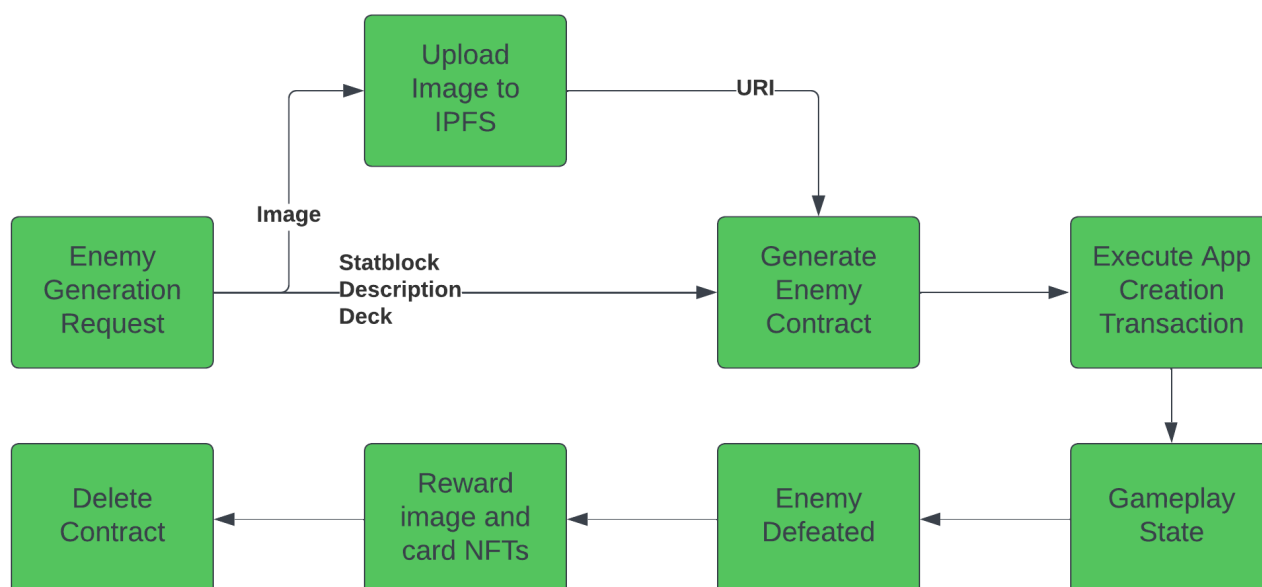
Card or enemy NFTs will be minted and/or distributed at the following stages.

Account Creation



Whenever a player creates a new account, the account receives a basic deck to their collection in the form of a set number of basic cards. Each of these cards must be generated as an asset with the unit name "CRPTCARD" and a descriptive asset name corresponding to the name of the card. The minting and distribution of these cards will be combined in an atomic transaction ensuring invalid NFTs do not leak into the supply.

Enemy Creation











When enemies are created, they will gain a pre-determined deck of cards similar to player account creation. However, the deck of cards will be supplied to the enemy creation script and designed by a developer. Each of the generated cards will have corresponding NFTs of the same "CRPTCARD" unit name which are then placed in escrow in the smart contract that corresponds to that enemy. When the enemy is defeated, a subset of those cards are randomly selected to reward to the victorious player, and the rest are deleted.

In addition to the cards, the enemy's image will be generated, uploaded to IPFS, and minted as an NFT before being placed in escrow in the smart contract. This NFT and a subset of that enemy's deck will then be awarded to the victorious player when the enemy is defeated. The enemy's contract will include a separate game state for every active challenge, and games will be played on-chain with transactions submitted to the enemy contracts.

Enemy Generation

A text based stat block and character background is generated from GPT3 based on a parmaterized prompt. The stat block is passed into DALE in order to generate a visual representation. The stat block and images are then combined to make the enemy card. Initially, these cards will be text-only. However, in the future, cards will be fully-featured and include images similar to the enemy cards.









Algo Crypt Enemy Browser



Name		Rarity	Deck
Gandalf			<div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div></div>
Gimli			<div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div></div>
Legolas			<div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div></div>
Shrek			<div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div></div>

NFT Marketplace

NFT marketplace for cards and defeated enemies (including rarities, value, etc).

Marketplace

Name		Rarity	Price
Gandalf			10 CRPT
Gmili			3 CRPT
Legolas			7 CRPT
Shrek			100 CRPT

Name	Rarity	Price
Slash		0.5 CRPT
Dodge		0.7 CRPT

The marketplace is a standard NFT marketplace where users exchange assets using atomic swap contracts. The assets to be traded are the playing cards and enemy cards received from defeating enemies on the chain.

Types of Cards

- Basic
- Common
- Uncommon
- Rare
- Mystic
- Legendary

Future Projects

The virtual game can be incorporated onto a physical playmat through RFID chips in the playmat and cards. This would allow for the physical game to be played for real stakes.

Reinforcement learning for enemy moves to create more adaptive and challenging enemy playstyles

The storage of enemies and NFTs in escrow contracts means that larger enemies could be introduced to be fought by multiple players at once. The enemy NFTs rewarded could then be shared using Algorand's fractional NFTs.

Allow a player vs. player arena where players can wager and challenge other players.

Developers

Joseph Xiao, RPI, xiaoj6@rpi.edu

Shawn George, RPI, georgs2@rpi.edu

Luke Williams, RPI, willil14@rpi.edu

Yichen Zhao, RPI, zhaoy17@rpi.edu