

Game Train: Blockchain Project

Overview

This project is intended to evaluate students' understanding of the Bootcamp. Students are encouraged to submit this project within stipulated time frame, this ensures their ability to understand Business Requirements and Production skills.

Project is based on the Jackpot game. Users can join the jackpot and wait for the winner's announcement. Once the winner is announced, Winner gets all the jackpot tokens while others get NFTs which can be rare according to their staked amount. Users can sell their NFTs to a Marketplace and buy from it as well.

Features

Authentication

- User should be able to authenticate using Metamask
- User should be able to create his/her profile
 - Firstname, Lastname, picture

Jackpot

- Admin should be able to launch a Jackpot
- Admin can postponed the Jackpot before Launch date
- User should be able to see Next Launch Date of Jackpot
- User should be able to participate in Jackpot
 - Must pay amount to participate in jackpot
- After 15 days of Jackpot launch, a CRON job triggers a function which disperses all Jackpot amounts to the winning participant. Winning participant is determined randomly.

- Winning amount is the sum of all the amount submitted by participants upon joining Jackpot
- Non-Winning participants get a Unique NFT according to their Amount. More amount means a more rare NFT.

NFTs

- Admin should be able to mint NFTs with Metadata
 - Picture
 - Opensea metadata standard
- User should be able to see their NFTs

Token

- User should be able to see their ERC20 tokens
- User should be able to transfer their ERC20 tokens to any other account

Marketplace

- User should be able to list their NFTs on marketplace
- User should be able to buy NFTs

Contracts

- ERC20
- ERC721
- Marketplace
- Jackpot

Tech Stack

- Nodejs as backend
- Mongodb as database
- Any EVM based Blockchain (ethereum, binance, polygon)
- Any Frontend technology

Deployment

- Smart contracts shall be deployed on Testnets

Deadline

29th May, 2022 23:59

Note

You must understand the Business requirements first. You are supposed to build Smart Contracts and Backend with utmost perfection. You must build APIs to interact with the system. Frontend part is Optional, however recommended and has an edge. Frontend is not supposed to be fancy but should be clean and concise.