System Design

1. Diagram

Added attached file

1. Unity

* **Scene**

There are several scenes.

* + **Loading Scene (initiator scene)**

Before starting game, loading scene is working.

The following functions are performed.

* + - Logo Animation
    - Flash Effect
    - “Don’t Destroy Script”

The object of this script and child object don’t destroy.

* + - * PhotonView Script
      * “Don’t Destroy Information”
        + Player name

Public Get

Private Set

If player don’t set name, making randomized name.

* + - * + PlayerNetwork Instance
        + Number of players
        + Player controller
        + Current Player
        + Background music
        + PhotonView Load Scene
        + NFTs list
        + Selected NFTs
        + Wallet address
      * Set skybox
  + **Connect Wallet Scene**

Use chainsafe sdk or moralis.

* + - Add modules
    - Developing a Frontend UI
    - Setting up a Moralis Server
    - Set Environment
    - Creating a Moralis Server and aquiring a blockchain node.
    - Configuring the web3 wallet connection
    - Developing the app manager script
    - Fine – Tuning the UI

Functions

* + - Connect wallet in unity.
    - Get wallet address and save in script
    - After getting wallet address, get wallet game coin amount.
  + **Main Scene**

Basic menus and some functions are included.

Menus

* + - Different Rooms with different fees.
    - Withdraw
    - Choose NFTs
    - Rules or Guide

Functions

* + - Set different fees in different room. (0.1BNB, 0.5BNB, 1BNB)

**NOTE:** no decided game coin and room number.

* + - Get game coin amount in game contract.
    - PhotonView Load Scene.
  + **Choose NFTs Scene**
    - Get NFTs
      * Find NFTs by wallet address in ERC721
      * Get NFT address and index and their weight
    - Choose NFTs
      * Get images using NFT address, wallet address and index
      * Show images list
      * Double check function
      * Save selected NFTs in don’t destroy script.
      * Exception handling
  + **Role or Guide Scene**
    - Options

Select an option to see a description.

* + - Panel

Description

* + - Functions
      * Unity Editor Load Scene
      * Back button event
  + **Lobby Scene**
    - User Info
      * 2 NFTs cards of them choice
      * Username
      * Wallet Address
      * User Avatar
      * Their balance
      * Their marks
    - Functions
      * Add Table
        + Photon Create room
        + Create new room name and nick room name
        + OnIsMaster -> PhonNetwork.is MasterClient
        + Set player custom properties
      * Join Table
        + Photon Network Join Room
        + Find Room ID
        + Load Scene
      * Player Joined Room
        + Find photon player in a room
        + All Player left room
        + Check isMaster or no
        + Add player
      * Room List
        + Create Room Object with script
        + Instantiate Object when create room
        + Show number of players in a room
        + Show room fee
        + Show room name
        + Show player ping
        + Apply photon player
  + **Game Scene**
    - Players
      * Get and show players using photon RPC.
      * Show only their own NFTs at first
      * Show all NFTs at the end.
      * Get own NFT weight
      * Player avatar, player name, player’s NFTs, no show their balance.
      * Player postion
      * Player Management
    - Game Start
      * Disable room in room List
      * Player can’t leave room.
      * Coefficient of time
    - Flop Cards
      * 3 Randomized cards.
    - Win Result
      * Get All player’s weight.
      * Get Flop card’s weight.
      * The winner is decided by logic.
        + Weight

Five of a kind

Royal flush

Straight flush

Four of a kind

Full house

Flush

Straight

Three of a kind

Two pair

Pair

High Card

* + - * Calculator winner’s rewards.
      * Change all player’s balance.
        + Set 0 loser wallet address balance in contract.
        + add amount winner wallet address balance in contact.
        + Consider gas.
      * Withdraw
      * All player’s leave in a room.
        + Destroy room
        + Reset player’s info.
  + Result Scene
    - Show all players in a table.
    - Show all nfts in a table
    - Show player’s result.
    - Show winner’ rewards.
    - Back
* **Photon**
  + Player Network
    - MasterLoadedGame
    - NonMasterLoadedGame
    - RPC\_LoadGameOthers
    - RPC\_LoadGameScene
    - OnConnectedToMaster
    - OnPhotonPlayerDisconnected
    - OnDisconnectedFromPhoton
    - OnSceneFinishedLoading
  + Player Controller
    - UpdatePlayerMovement
      * Card position
      * Card reveal
      * Flop Card
    - Update
      * Count time
    - OnPhotonSerializeView
      * Transform.postion
      * Transfom.rotation
      * Animation status
    - SetName
    - PlayerInfo
  + RoomListing
    - Instance canvas
    - Destroy button
    - Set Room name
    - Set player’s number in a room.
    - Set Fees
    - Add Button
  + LobbyNetwork
    - OnConnectedToMaster
    - OnJoinedLobby
    - Update lobbystatusText
  + PlayerLayoutGroup
    - Get PhotonPlayers in a room
    - OnJoinedRoom
      * OnPhotonPlayerConnected
      * OnPhotonPlayerDisconnected
      * PlayerJoinedRoom
      * PlayerLeftRoom
    - OnRoomState
    - OnLeaveRoom

1. Blockchain

* Get Game token contract
* Create Game contract
  + Construct
    - Game token contract address
    - card NFT address
  + import
    - ownable
    - safeMath
    - IERC721
  + Variable
    - PokerNFT
    - Game token
    - payableAddress
    - PlayerList
  + Functions
    - setPayableAddress
      * parameter : addresss
      * onlyOwner
    - transferGameToken
      * parameter: amount
      * transferfrom
        + from player to owner
    - sendToWinner
      * check player’s balance
      * tansferfrom
        + from owner to player
      * player’s balance is 0 in playerlist.
    - SetPlayerAmount
      * Parameter: playerAddress, amount
      * OnlyOwner
      * Set player’s balance and address in playerlist.
* Control approve.

Owner have to control approved amount to send money to the winner.

1. Backend

* Server
  + Node.js
  + Provide API

1. Database

* Server
  + Node or Mongo
* Structure
  + User info
    - Username
    - User address
    - User avatar
    - User winner history -> link
    - User lose history -> link
    - Timestamp
  + Game History
    - User address
    - User win
    - User loses
    - Timestamp
  + Room
    - Room name
    - Room nickname
    - PlayerList in a room (array or json)
    - Status
  + Owner approved status
    - Id
    - Time
    - Approved amount
    - Address
  + Admin
    - Admin email
    - Admin password
    - Admin name
    - Admin allow
    - TimeStamp
  + Admin balance
    - ID
    - Earned amount
    - timestamp
    - Room Name
    - Approved gas
    - Total amount
    - Player number
  + Withdraw
    - Address
    - Balance
    - TimeStamp
    - Player name
  + Owner Withdraw
    - Balance
    - TimeStamp
  + Connect Us
    - Address
    - Title
    - Email
    - Content
    - TimeStamp
    - Status
    - Reply content