You can create, save and edit database on Unit project.

But it’s not good for secure. So, we need backend system.

Also, this game is NFT game on blockchain network.

We can also use blockchain for database. It is possible. You can create new contract on network for database. But there is fee and speed problem.

Therefore, we need to have backend system.

You prefer NoSQL. This is good choice. So, I will use MongoDB for database.

Simply Diagram.

Client

Backend

Blockchain

Contract

Database

Middleware

Photon or Mirror

1. **Register and login system**

* Database (Auth)

Email, password, PlayDataID, Token, Created Time, Updated Time,

* Questions
  + Do you need to send a verification code to my email to sign up?
  + Need to store player location information for secure?

1. **Chatting system**

* Database

\_PlayerDataID, Description, Attach, CreateTime

* Questions
  + Do you need save Chatting log?
  + What information show on chatbox?

Avatar, PlayerName, PlayerLevel, chat content, attached file.

1. **PlayerData**

* Database

PlayerName, PlayerLevel, Avatar, Level

Gold, MaxGold, Elixir, BlackElixir, MaxElixir, MaxBlackElxir, Gem

OwnedBuildings []

FriendID, AuthID,

* Questions
  + What info should I add?
  + Is it ok to not have the OwnedBuildings structure?
  + Or Is it better to create ownedbuildlings document? (document is same table on mysql)

1. **Shopping**

* Database

Title, SubTitle, ItemName, ItemInfo, ItemPrice (Coin, Drop, Diamond etc), RequireLevee, ItemBalance (Total, Used),

AvailableFlag

* Questions
  + Do you need create shopping database?

I mean Do you need to save shopping info on database?

1. **Wallet & balance**

* Database

PlayerID, walletaddress, Network, Balance (coin, drop, diamond etc),

* Questions
  + Which network do you use ?
  + Did you deploy game contract on network?
  + Player just purchase game coin (coin, drop, diamond etc) using their cryptocurrency?
  + Is this game p2p game? Can player earn while play game?

Need to create more database information?