# NFT Hunt

Our idea is to create a real world NFT hunt.

## **Game Perspective**

### NFT Creator

1. Place an NFT somewhere on the map (testnet).

### NFT Treasure Hunter

1. Find an NFT somewhere on the map (testnet).

## **Rules**

→ Users will travel to predetermined locations to collect NFT’s representing those areas.

→ The corresponding NFT will cost some token in order to acquire, and the only way to access the possibility of buying these tokens is by being in a very specific location.

→ If the user would like to purchase the NFT, they can initiate the smart contract and exchange some coins in their wallet for the corresponding NFT at that location.

## **Initial Steps**

After looking into the two leading geolocation projects, XYO and FOAM, it is obvious that the space has some serious development issues. Both of these projects are plagued by their inability to produce a dynamic location. This severely limits the usability of the network and renders it almost useless for our purposes.

Use External adapter to Google maps API for Geolocation data

As a temporary solution it is going to be best to use GPS data from a user’s phone. This would look like the following:

1. A user walks up to a predetermined location
2. The user sends its location data to an oracle
3. The oracle sends the GPS data to the smart contract with the corresponding location specifications.
4. Use VRF to generate NFT’s on different parts on the planet. One could be generated for every square mile. Can increase NFT generation based on density of population in those areas.(Backend)
5. If the parameters are met in the contract, the user will be given the ability to purchase the smart contracts bounty NFT with ether
6. The users wallet is then credited with the NFT
7. That contract expires
8. Takes a certain amount of time for that NFT to become purchasable in that location again.
9. Include a library (leaflet) to map, where the NFT’s will be located, show distance from current user location. (Front end)

Task

## **Next Steps**

Front End:

1. Get user’s localisation
2. Getting NTF around the user/on the map
   1. Deposit an NFT
   2. Display NFT on the map so the user can see it on their phone
   3. Identifying users

Back End:

1. Check out the google maps external adapter
2. Use the polygon testnet to write first smart contract
3. Once the user makes it to the NFT location, they are provided the option to buy the NFT
   1. If the user would like to buy the NFT, they will have to enter their public address so that the NFT can be sent there.
4. Add a backend server to keep track of the NFT location and name
5. Allow the app to connect to this backend server so that it can show the locations of these NFT’s

## **Maximize project rewards**

1. Use the Kraken Oracle as a way to get price data for link.
   1. Good for price at a certain time(Prediction market)
   2. Can use to allow people to show up and buy an NFT
2. Use Plasma network to deploy smart contract.
   1. This uses polkadot parachain(Allows design of networks with own logic(Defi Parachain, smart contract parachain, iot parachain))
   2. Polkadot relay chain does not support smart contract functionality
   3. Chainlink integration into plasm testnet parachain
   4. How to:
3. Polygon- Cheap VRF call
   1. Hosting server IPFS
   2. Providence Connections
      1. Protocol Labs
      2. Dierick
4. ENS - Ether Name Service- Montpelier NFT(Naming the contract for readability)

## **Ideas for further development**

1. How to maintain rarity of NFTs?
2. Randomly Spawning NFTs?
3. Trading of NFTs?
4. Synthetic
   1. Can trade any asset on the platform (Add an s )
5. A real world easter egg hunt that is cryptic and hard to decipher.
6. Artists can release their own artwork hidden around the world.
7. NFT auction at a certain location
   1. Creates a rarity factor for NFT’s being dropped in certain locations.
   2. Can hire artists to release these super rare NFT drops.
8. Levels (reward Hunter loyalty)
9. Changing radius in which you can buy/sell the NFT

## **Technical Read**

1. [Scavenger Hunt](https://ieeexplore.ieee.org/document/9253568): I suspect this might be very helpful to plan things, especially on the backend(?)

## **Competitors/Similar Projects**

1. [OVR](https://www.ovr.ai/): these guys seem to have a solid team and a product that’s been around for a little while. Definitely worth looking into.
2. [NFTEgg](https://nftegg.com/#store)
3. [Rare Pizza](https://www.rarepizzas.com/)

## 

## **NFT Marketplaces**

1. [Rarible](https://rarible.com/)
2. [Mintable](https://mintable.app/)
3. [OpenSea](https://opensea.io/)