Non Fungible Fun

**Presented by**: Mike Rae
Grace Ho
Muhammad Dosani
Mikhara Ramsing

What we'll cover

★ NFT Concept - NonFungible Fun

★ The FinTooney Journey

★ Intro to the team

★ Utilising AI to alter images and create NFT Avatars

★ Toonify API Integration/ Python

★ Creation of token (TOON) and crowdsale

★ Auction and bid contract - Summary

★ Key learnings

★ Questions

Non Fungible Fun - A digital avatar of the Fintech group
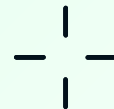
**A digital collection of Non Fungible Collectibles of the Fintech 2021 group**

➜ Digital Avatar of classmate created as an NFT
➜ Creation of a token Fintooneys (TOONS) to facilitate the sale on the Blockchain
➜ Smart contracts deployed for Auction and bidding of the NFTs
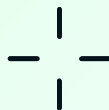
WORLD's FIRST PERSONALISED NFT MARKETPLACE

Create, personalise and sell your own NFTs

On the world's first personalised NFT marketplace
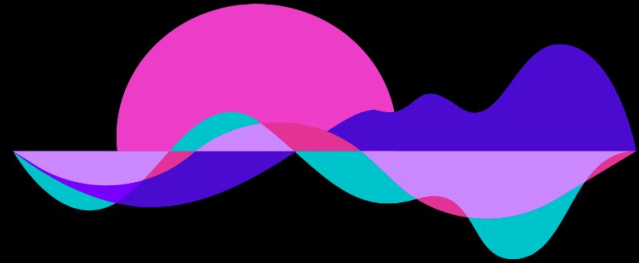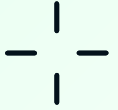
Fintooney

Where NFTs become FUN

NonFungible Fun

Team Intro

OUR TEAM
Week 1 FinTech Bootcamp 😁

**Mike
Rae**
Bright eyed
sales

**Mikhara
Ramsing**
Hopeful
Marketer

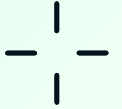**Grace
Ho**
Optimistic
CFO

**Muhammad
Dosani**
Visionary
Evangelist

OUR TEAM
...An intense 24 weeks later

**Mike
Rae**
Pro
Developer
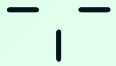
**Mikhara
Ramsing**
Pro
Developer

**Grace
Ho**
Pro
Developer

**Muhammad
Dosani**
Pro
Developer

The FinTooney Journey

User Journey

| Load image onto portal | → | Personalise image (AI / ML or artist) | → | NFT Created | → | Front End Auction Site | → | Purchase TOON* with ETH | → | Purchase NFT with TOON* | → | You own your personalised NFT! |

Architecture

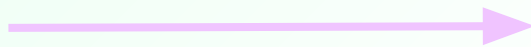| Web or Python portal | API call to Toonify (RapidAPI / Python code integration) | IPFS filestore (via Pinata) | ERC721 Token | ERC20 Token | Bid & auction process | Ownership transferred |

Solidity - Smart Contracts & auction →

TOONIFY API Integration/ Python

➔ API built out from RapidAPI using python (via Jupyter Notebook)

➔ Initial API call returned image in bytes which required additional code to convert back to JPEG format

```python
# Request to send an image for toonification - note returns

url = "https://toonify.p.rapidapi.com/v0/zombify" # set the
query = {
         "face_index": 1,
         "return_aligned":"true",
         }
headers = {
         "x-rapidapi-host": "toonify.p.rapidapi.com",
         "x-rapidapi-key": rapid_api_key,
         "accept": "image/jpeg"
         }
files = {"image": open("face2.jpg", "rb")} # select image t

response = requests.request("POST", url, files=files, header
```
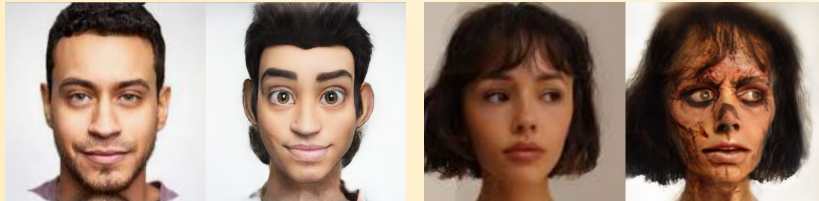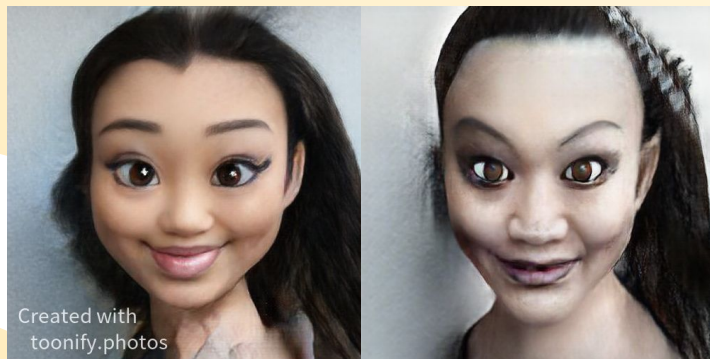

Test image #1

Test image #2

Utilising AI to alter images and create NFT Avatars

Toonify AI

Conversion of real images to selected toonify avatars

Created with toonify.photos

IPFS Storage

Decentralised storage via pinata to allow for URI connection in contracts and code

Image URI	CID
https://gateway.pinata.cloud/ipfs/QmSH5h9dE7yfWJcoMQPsTBcYvDe8PUavfKobG8G4U2oyPJ	QmSH5h9dE7yfWJcoMQPsTBcYvDe8PUavfKobG8G4U2oyPJ
https://gateway.pinata.cloud/ipfs/QmcnufRrCCiJs2PdfV49mzFRekkYqKQizSEb7wmuyennAb	QmcnufRrCCiJs2PdfV49mzFRekkYqKQizSEb7wmuyennAb
https://gateway.pinata.cloud/ipfs/QmTuqz87n5LuLDmDxuBWAgtetkyZT5XyufyupXGt6FR8sN	QmTuqz87n5LuLDmDxuBWAgtetkyZT5XyufyupXGt6FR8sN
https://gateway.pinata.cloud/ipfs/QmV8qz87bvABMVLsqKfprF1LUrNUU5KTNQ1E5DTyyKSiuc	QmV8qz87bvABMVLsqKfprF1LUrNUU5KTNQ1E5DTyyKSiuc
https://gateway.pinata.cloud/ipfs/QmdJtMaoPGpdBZcKxEnZLRKCGYNwBVj5q7GTk7VXKQZ77H	QmdJtMaoPGpdBZcKxEnZLRKCGYNwBVj5q7GTk7VXKQZ77H
https://gateway.pinata.cloud/ipfs/QmPTJfKSmQGfap5kr4y52zS7rFrUu29MKgsajtocohbAKw	QmPTJfKSmQGfap5kr4y52zS7rFrUu29MKgsajtocohbAKw
https://gateway.pinata.cloud/ipfs/QmXj8sdbZxMwn2vmFgjExatJhFTRPFSbt22jJHf1RbhKRS	QmXj8sdbZxMwn2vmFgjExatJhFTRPFSbt22jJHf1RbhKRS
https://gateway.pinata.cloud/ipfs/QmQ2hmHzpfH1HRSi1RTXYiGPSeWxsseMayMcg8LtuCRMry	QmQ2hmHzpfH1HRSi1RTXYiGPSeWxsseMayMcg8LtuCRMry
https://gateway.pinata.cloud/ipfs/Qmcrrefc9kBHrkh9Z212tKHSczo3wFj8f3WJFMKw2vdavd	Qmcrrefc9kBHrkh9Z212tKHSczo3wFj8f3WJFMKw2vdavd
https://gateway.pinata.cloud/ipfs/QmVEHdxwzcsH3bqKcYDLJUbvBvnkjFtC5WjELZM1uo9X5Z	QmVEHdxwzcsH3bqKcYDLJUbvBvnkjFtC5WjELZM1uo9X5Z
https://gateway.pinata.cloud/ipfs/Qmb7zi4vrDn6QDT6YsD8DJ9vZGAMcF6iazV7h2ZVuxwkj2	Qmb7zi4vrDn6QDT6YsD8DJ9vZGAMcF6iazV7h2ZVuxwkj2
https://gateway.pinata.cloud/ipfs/Qmf984q8w9HVyhyTgi9LtDbmiuWSeeanP7UUeJex8uvfhP	Qmf984q8w9HVyhyTgi9LtDbmiuWSeeanP7UUeJex8uvfhP
https://gateway.pinata.cloud/ipfs/QmdrwWJNfU7JBUAdkwAe423GhNffgCk7xLa8B49VP42YFd	QmdrwWJNfU7JBUAdkwAe423GhNffgCk7xLa8B49VP42YFd
https://gateway.pinata.cloud/ipfs/QmXFXHiCvtGmcnwz87SRK6z7cFLcpnVJVEbqGK3nXKccfh	QmXFXHiCvtGmcnwz87SRK6z7cFLcpnVJVEbqGK3nXKccfh
https://gateway.pinata.cloud/ipfs/QmdtujAWu6avraFMBrhwb7QzwstSFQUzTy6yNPeBs3G3od	QmdtujAWu6avraFMBrhwb7QzwstSFQUzTy6yNPeBs3G3od
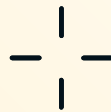
Utilising AI to alter images and create NFT Avatars

Halloweenified CLAUDIA DONG

Current price
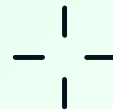
◆ **1.1** ($3,906.94)

[Buy now] [Make offer]

📈 **Price History**

All Time ⌄

Description

Creation of token and crowdsale

Token contract

- ➔ *Type*: ERC20 token
- ➔ *Name*: FintoonyCoin
- ➔ *Symbol*: TOON

Crowd sale contract and deployment

- ➔ Testing via;
  - ○ Javascript VM (local)
  - ○ Injected Web3
    - ● Ganache
    - ● Rinkeby Testnet

Solidity contract code example
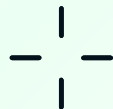
Auction and bid contract

```
contract ToonFinMarket is ERC721Full, Ownable {

    constructor() ERC721Full("FintooneyMarket", "TOON") public {}

    using Counters for Counters.Counter;

    Counters.Counter token_ids;

    address payable foundation_address = msg.sender;

    mapping(uint => FintooneyAuction) public auctions;

    modifier toonRegistered(uint token_id) {
        require(_exists(token_id), "NFT not registered!");
        _;
    }
}

contract FintooneyAuction is FintooneyCoin, SpenderRole, ERC20Spendable {
    address deployer;
    address payable public beneficiary;

    // Current state of the auction.
    address public highestBidder;
    uint public highestBid;

    // Allowed withdrawals of previous bids
    mapping(address => uint) pendingReturns;

    FintooneyCoin public acceptedToken;
```
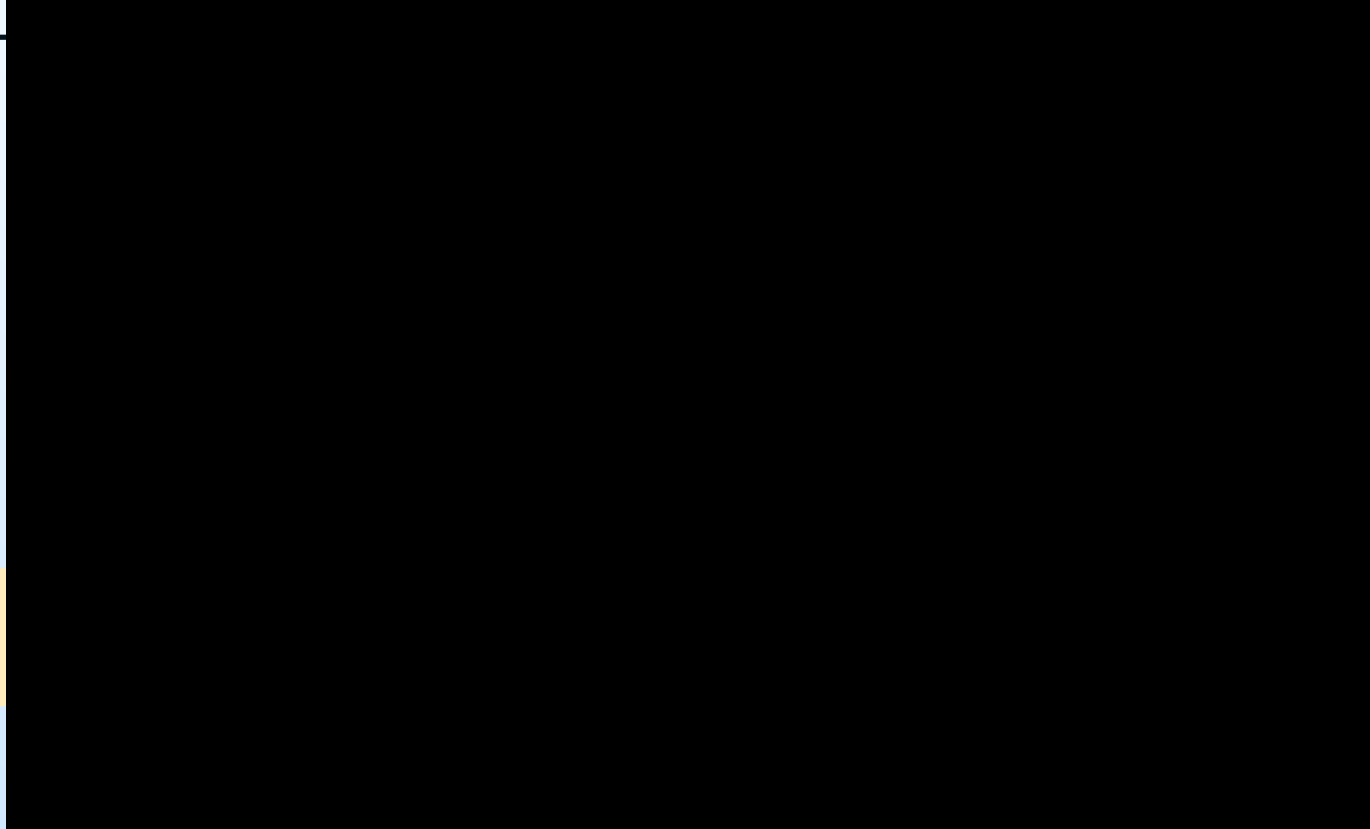
Fintooney Auction Contract

➔ Intended to set TOON as currency for auction but more complex than envisioned
➔ Set up rules for Auction: bid, bid time, highest bidder, withdraw bid, auction end

Fintooney Market Contract

➔ Intention: use ERC20 to buy ERC721
➔ Intention: Called for msg balance of TOON
➔ Auction held for each NFT
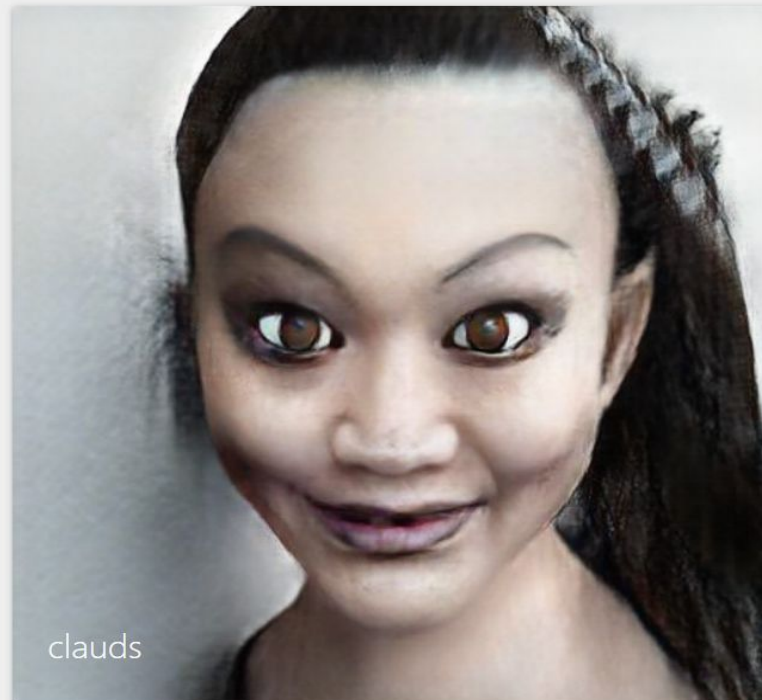
Proof of concept

Fintooney Markets

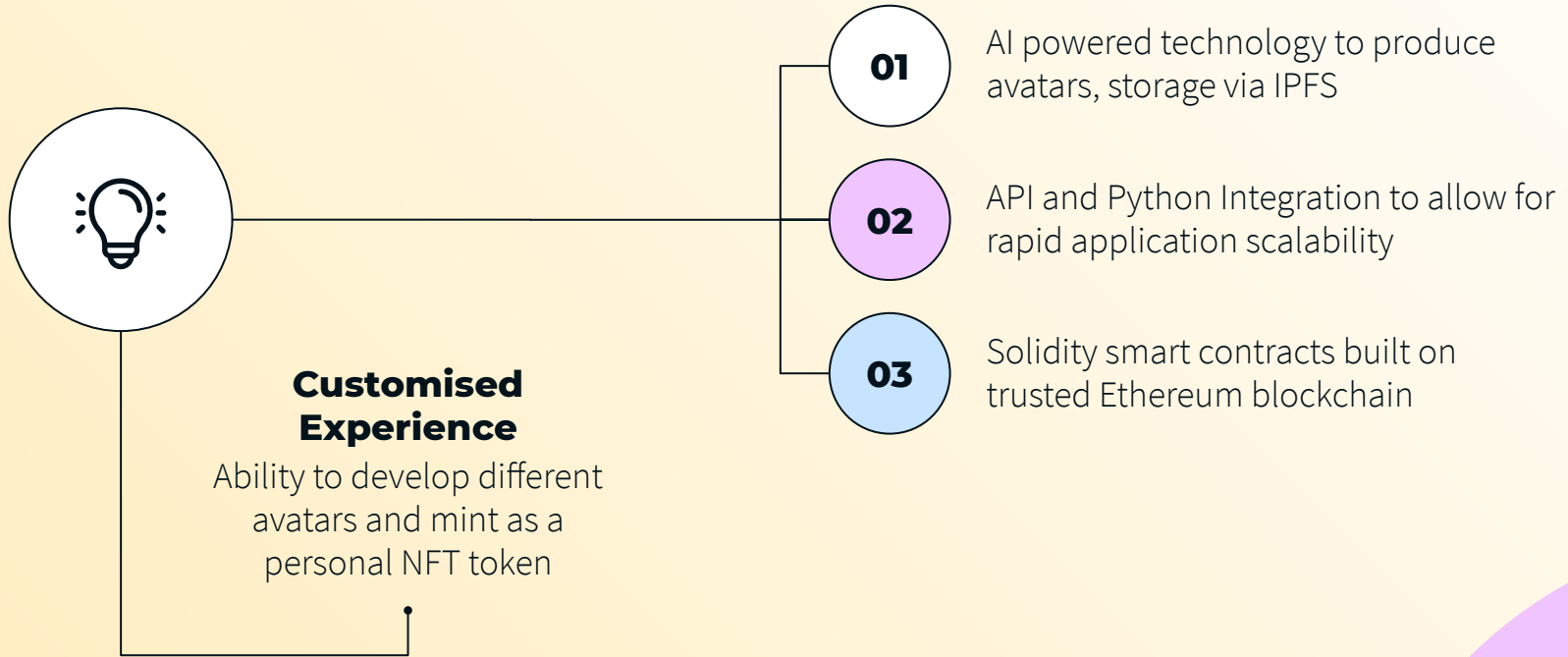Welcome to the Fintooney NFT Market. This is a platform where you can purchase your avatar NFTs.

2

Owner: 0x9Ec8b108930e69Dd2B3b71ef833BD221bfda7495
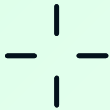

clauds

1

BID   END AUCTION

Summary

01 AI powered technology to produce avatars, storage via IPFS

02 API and Python Integration to allow for rapid application scalability

03 Solidity smart contracts built on trusted Ethereum blockchain

Customised Experience
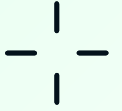Ability to develop different avatars and mint as a personal NFT token

Key learnings

→ Connecting multiple contracts in solidity and understanding the code as well as avoiding duplication
→ API Integration, conversion of image in bytes to jpeg
→ Front end development issues
→ NFT's generally sold using ETH by default, additional complexity to require purchase with token (FintooneyCoin)
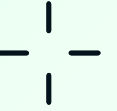
NEXT STEPS

→ Build Fintooney - a love child between Etsy and Open Sea - A customisable NFT platform

CRYPTO
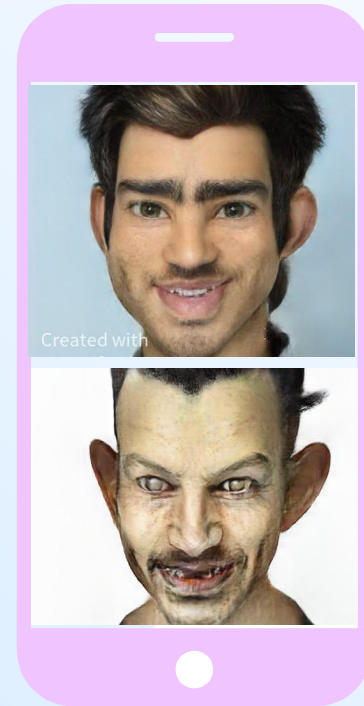
Do you recognise these faces?

How about these guys?

Questions?