

PROBLEM

In the past few years, cryptocurrency has become a popular tradable currency and is continuing to significantly grow. A common way cryptocurrency is used is through NFT's (Non-Fungible Token). NFTs are a unique and non-interchangeable unit of data stored on a digital ledger, otherwise known as a blockchain, and can be associated with reproducible items, such as photos, audio, videos and other types of digital files. Users can create, buy and sell NFTs on NFT marketplaces using cryptocurrency. A major issue users face on NFT marketplaces is its lack of clarity and complexity, as well as unaffordable prices. We made sure that our NFTies marketplace is user-friendly and will allow users to create NFTs with ease and is inexpensive to users who have created NFTs in the past and for those just getting started.

CURRENT SYSTEM

The current system for our application entails two different smart contracts necessary for minting an NFT to a test network based on Ethereum, and a basic structure for both front-end and back-end continuing from last semester. The contracts are ERC721 but can be updated to ERC1155. Our team was split into 3 smaller teams consisting of front-end, back-end, and crypto. As part of the front-end team I worked with a React.js/Bootstrap stack to create a user friendly web application. The back-end focused on a Django/Python stack to create the

REQUIREMENTS

For my part of the project I continued working on the front-end from last semester using a React.js/Bootstrap stack to create a navigable and responsive user interface for our marketplace.

System Design:

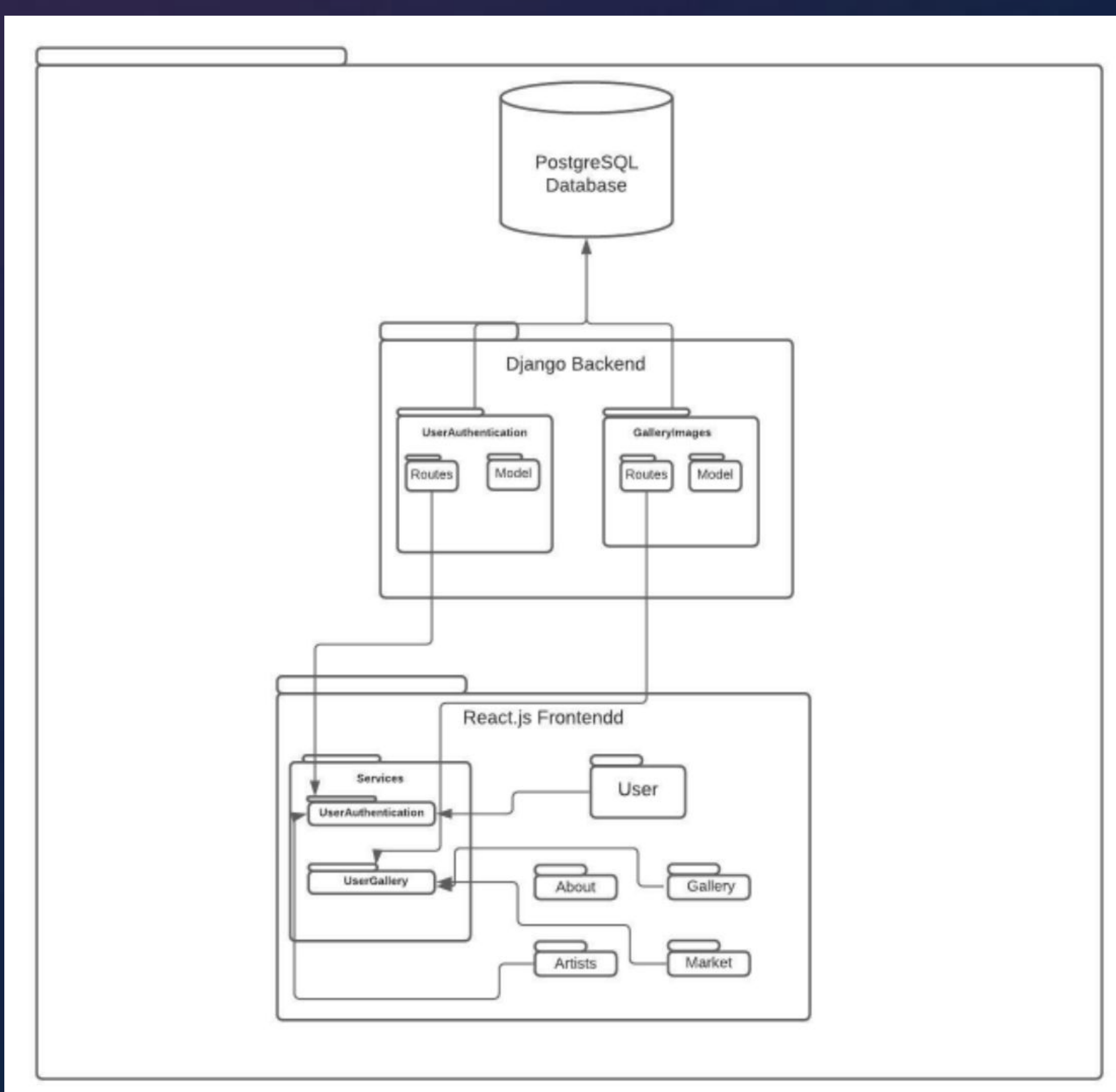
Front-End: React.js (JavaScript), CSS, HTML

Back-End: Python, Django

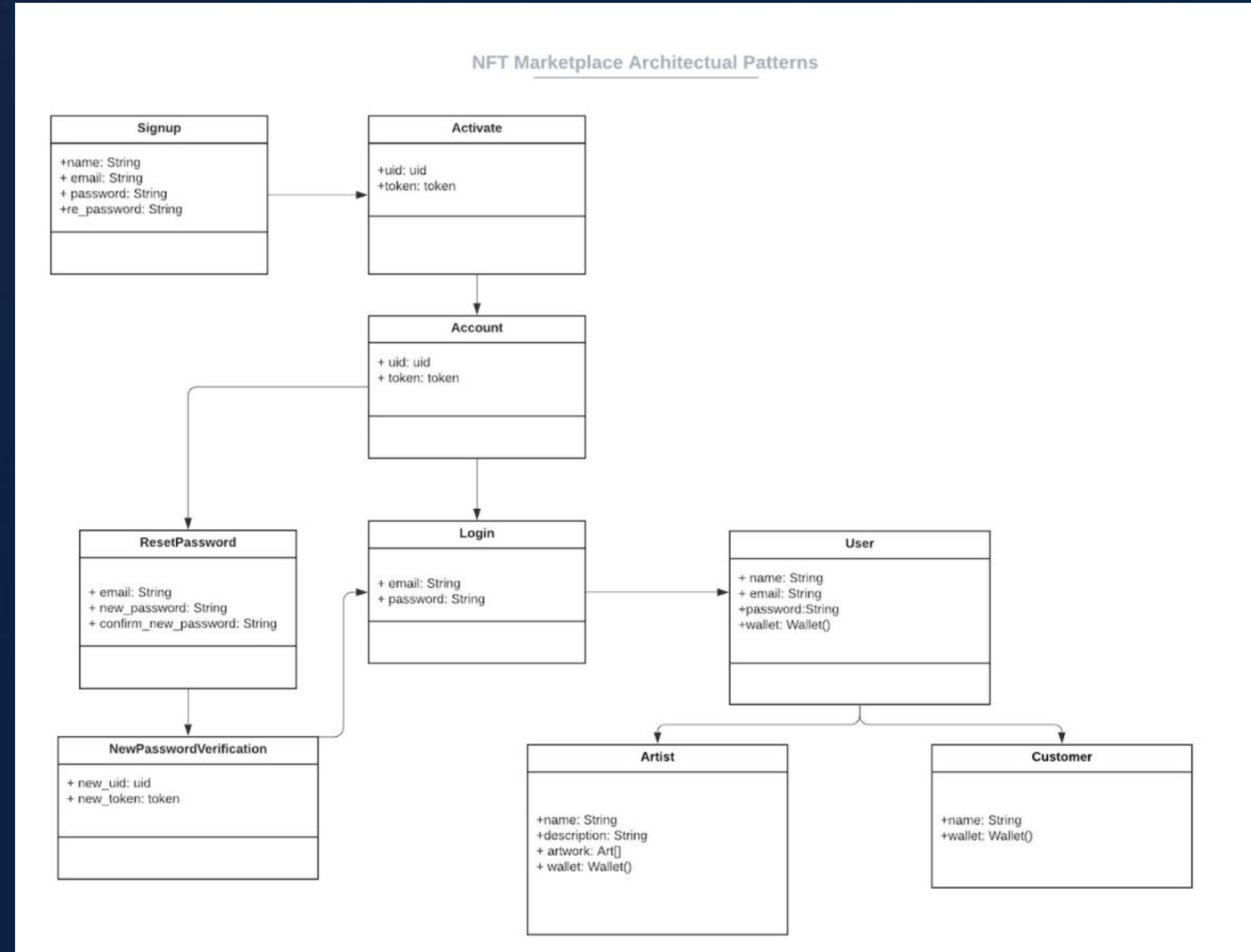
Databases: PostgreSQL/PostgreSQL Terminal

Crypto: Ethereum, solidity language-based smart contract

SYSTEM DESIGN



OBJECT DESIGN



IMPLEMENTATION

Front-End: React.js (JavaScript), CSS,

HTML

Back-End: Python, Django

Databases: PostgreSQL/PostgreSQL

Terminal

Crypto: Ethereum, solidity

language-based smart contract

VERIFICATION

For verification, we used a web console (Chrome DevTools) and Visual Studio Code for testing and debugging front-end and back-end. We used metamask to guarantee a user can log in. To verify minting NFT's was successful, we viewed input from forms on the user interface as shown in the screenshot below that were sent to the blockchain contract.

SUMMARY

Working on this project both in Capstone I & II, I can say that there has been significant progress and the user interface has improved from both a UI perspective and functionality thanks to the research done on crypto. There is still, however, lots of room for more features to be added.

SCREENSHOTS

