Ryan Herman

CMPSC 297 Project 1 Write Up

Description: Project 1 write up for developing a mock blockchain based off the functionality of bitcoin’s blockchain. The code developed follows along the helper code from the CMPSC 297 canvas page. I’ve added my own comments at the top of each of the files describing the purpose of that files andrelevant functions.

1. **Project Status**

Completed installation of NodeJS and NPM

Completed setup of package.json and .gitignore

Completed main index.js file

Completed ‘routes’ folder

Tested ‘brew’ server successfully at <http://localhost:8080/>

Completed src functions (block, blockchain, transaction)

Completed routing functions (chain, listtransactions, mine, newtransactions, validate)

1. **Future Modifications**

I have a few ideas of modifications I would like to add to this project if I have some extra time in the future. The first idea is to add in a mining reward system to the mine.js file. Similar to how Bitcoin pays out mining rewards for validating blocks, I believe I could implement a similar thing in this project. The idea is that for every block validated by a miner, they will receive a reward of currency mined.

Another future modification I would find interesting, however is likely too complicated for this project, is adding in the functionality of off-chain transaction similar to the Lighting Network used by Bitcoin. I would be interested to see how the actual code is built into a blockchain to handle these types of transactions.

1. **Project Functionality**

I used the code from the guide in the canvas notes, therefore much of the functionality is similar to the provided code. In this section I have summarized the functionality of the files included in Project 1:

* index.js: Initialize and set up imports, variables, and ports for blockchain
* brew.js: Quick attempt to setup a server client connection
* chain.js: Creates a view of entire current blockchain
* listtransaction.js: Lists current instance of transactions to be stored on a block
* mine.js: Mine a new block and append it to blockchain after validation
* newtransaction.js: Set up a new transaction
* validate.js: Check validity of blockchain
* block.js: Builds a block class and sets up mining functionality
* blockchain.js: Set up blockchain structure which is just an array of block types
* transaction.js: Builds a mock transaction using a fake IP address