Andrew Martin
Practical Assignment 2

Ethereum Smart Contract Project

This project showcases a smart contract developed in Solidity for managing token balances on the Ethereum blockchain. It includes a middleware layer written in Python (utilizing a virtual environment) for interacting with the blockchain, and a gateway application to provide web service APIs for accessing smart contract functions. The project uses Truffle for smart contract compilation and migration, and Ganache CLI for simulating an Ethereum blockchain.

Prerequisites

Before starting, ensure you have the following installed:

- Node.js - JavaScript runtime required for Truffle and Ganache CLI.
- Truffle - A development framework for Ethereum.
- Ganache CLI - A personal Ethereum blockchain for testing.
- Python - Required for the middleware layer; ensure `venv` is installed.
- Git - Version control system for cloning the project repository.

Initialize Python Virtual Environment and Install requirements.txt

Create and activate a Python virtual environment in the project root:

```
python3 -m venv venv
source venv/bin/activate
pip install -r requirements.txt
```

Initialize Node Project

Initialize a Node.js project and install the necessary packages:

```
npm init -y
npm install web3 ganache-cli truffle
```

1. Run the Setup Script

Execute the `start_project.sh` script to automatically set up the project. This script initializes a local Ethereum blockchain using Ganache CLI, compiles and deploys the smart contract using Truffle, and starts both the middleware and gateway components.

```
./start_project.sh
```

Please ensure that the script has executable permissions. If not, grant them using `chmod +x start_project.sh`.

2. Test the Setup

After the setup script completes successfully, you can test the smart contract and API endpoints using the provided `client.py` script.

```
python3 client.py
```

This script will interact with the smart contract through the gateway and middleware, performing operations such as depositing tokens, withdrawing tokens, and querying account balances.

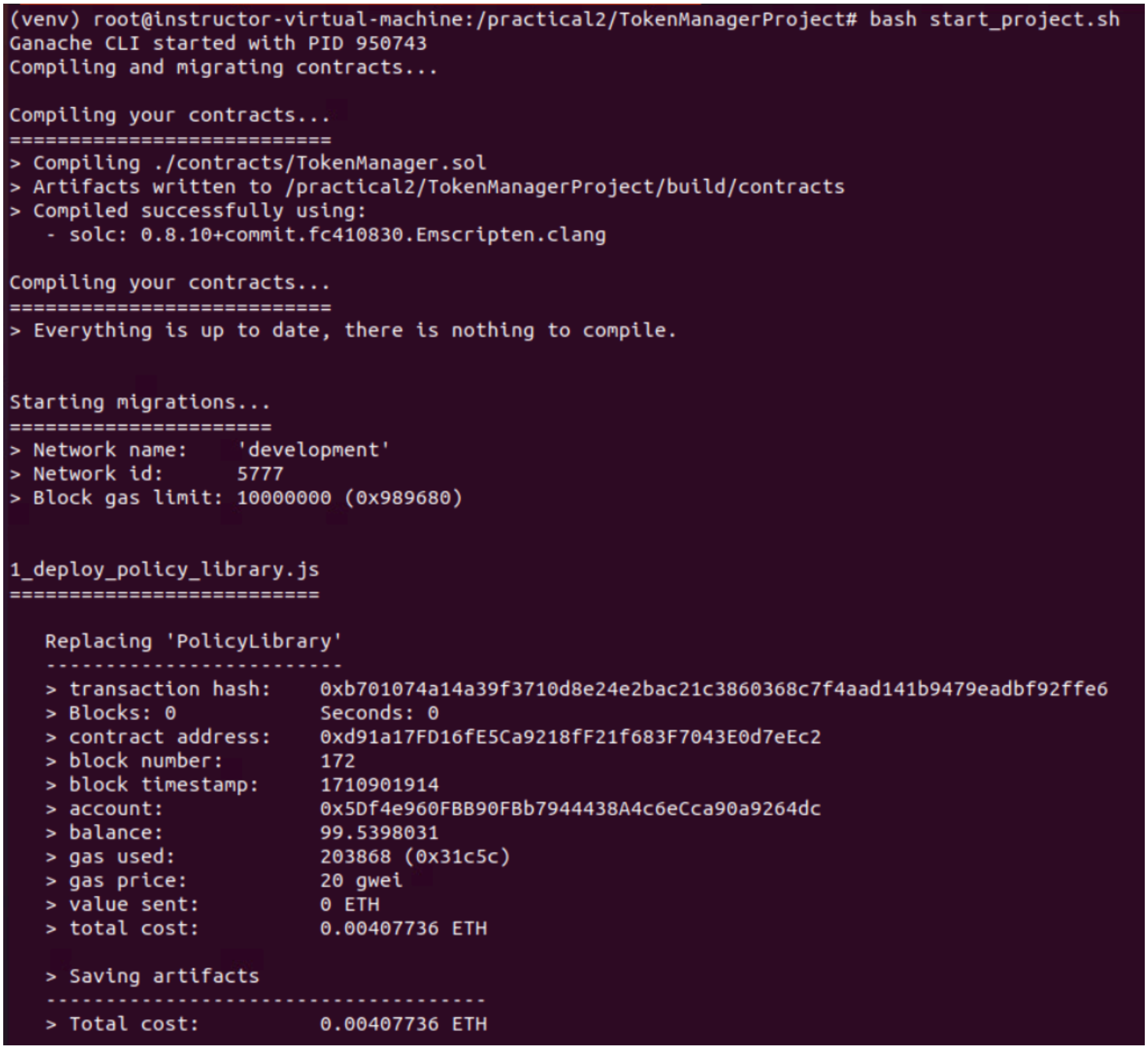
Additional Information

- The project includes a `package-lock.json` file, ensuring that you install the exact versions of Node.js dependencies that the project relies on.
- The project also includes a `requirements.txt` file, ensuring that you install the exact versions of Python dependencies that the project relies on.
- The Python middleware utilizes a virtual environment to manage dependencies, which is automatically handled by the `start_project.sh` script.
- The smart contract functions include deposits, withdraw, set policy, and query balance, providing a comprehensive demonstration of contract capabilities.

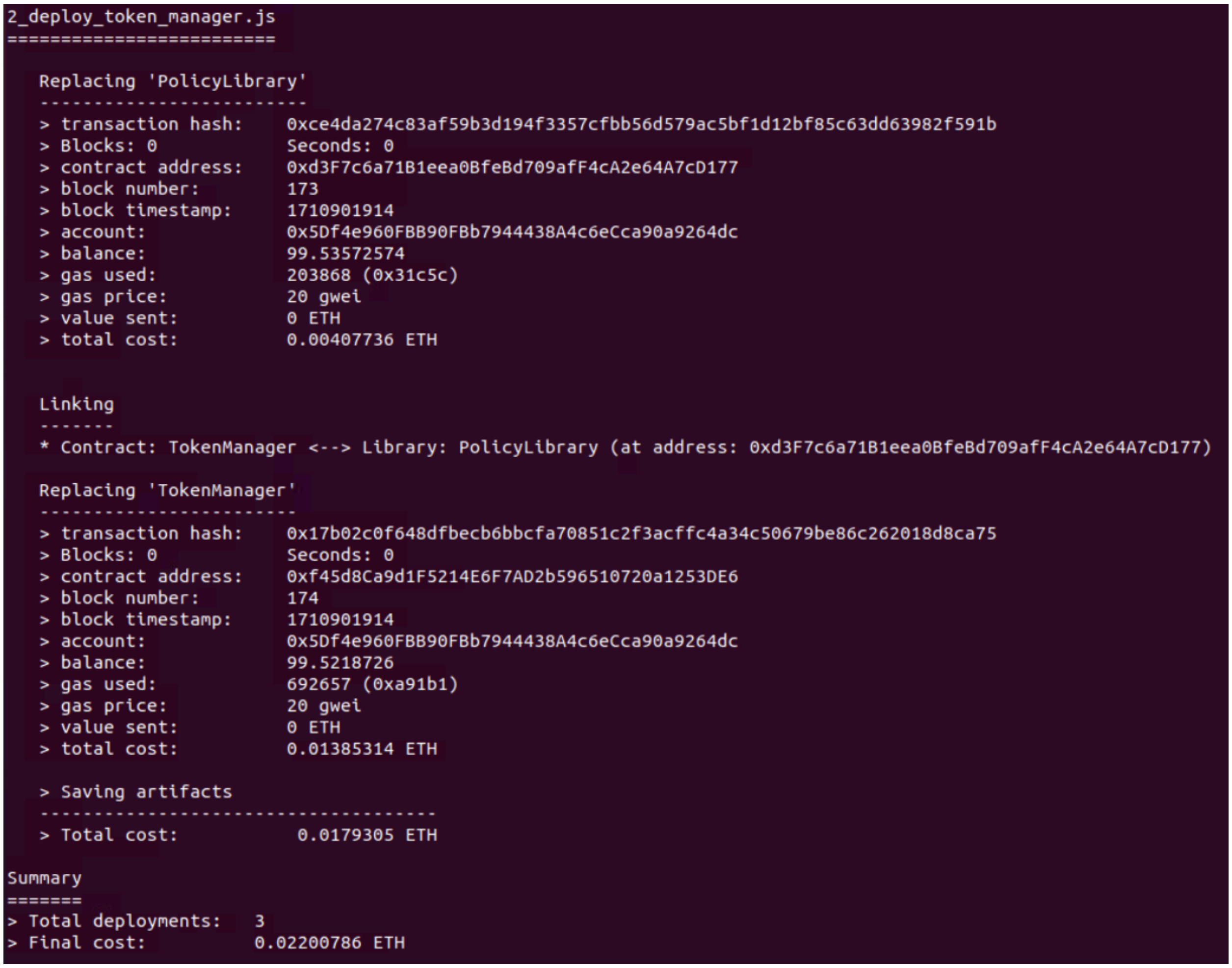
For more detailed information on the project components or configurations, please refer to the respective documentation or source code comments within the project repository.

Outputs

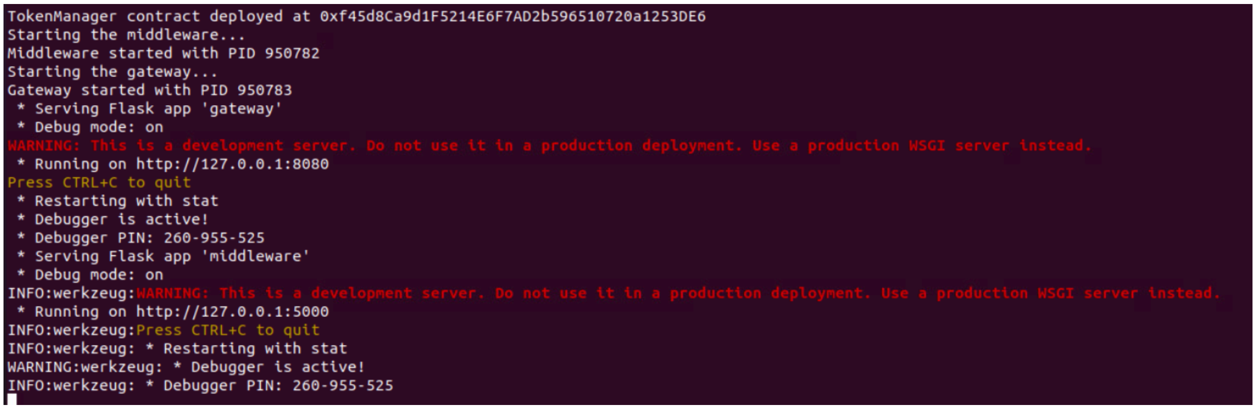
`start_project.sh`



```
(venv) root@instructor-virtual-machine:/practical2/TokenManagerProject# bash start_project.sh
Ganache CLI started with PID 950743
Compiling and migrating contracts...

Compiling your contracts...
===========================
> Compiling ./contracts/TokenManager.sol
> Artifacts written to /practical2/TokenManagerProject/build/contracts
> Compiled successfully using:
   - solc: 0.8.10+commit.fc410830.Emscripten.clang

Compiling your contracts...
===========================
> Everything is up to date, there is nothing to compile.


Starting migrations...
======================
> Network name:    'development'
> Network id:      5777
> Block gas limit: 10000000 (0x989680)


1_deploy_policy_library.js
==========================

   Replacing 'PolicyLibrary'
   -------------------------
   > transaction hash:    0xb701074a14a39f3710d8e24e2bac21c3860368c7f4aad141b9479eadbf92ffe6
   > Blocks: 0            Seconds: 0
   > contract address:    0xd91a17FD16fE5Ca9218fF21f683F7043E0d7eEc2
   > block number:        172
   > block timestamp:     1710901914
   > account:             0x5Df4e960FBB90FBb7944438A4c6eCca90a9264dc
   > balance:             99.5398031
   > gas used:            203868 (0x31c5c)
   > gas price:           20 gwei
   > value sent:          0 ETH
   > total cost:          0.00407736 ETH

   > Saving artifacts
   -------------------------------------
   > Total cost:          0.00407736 ETH
```

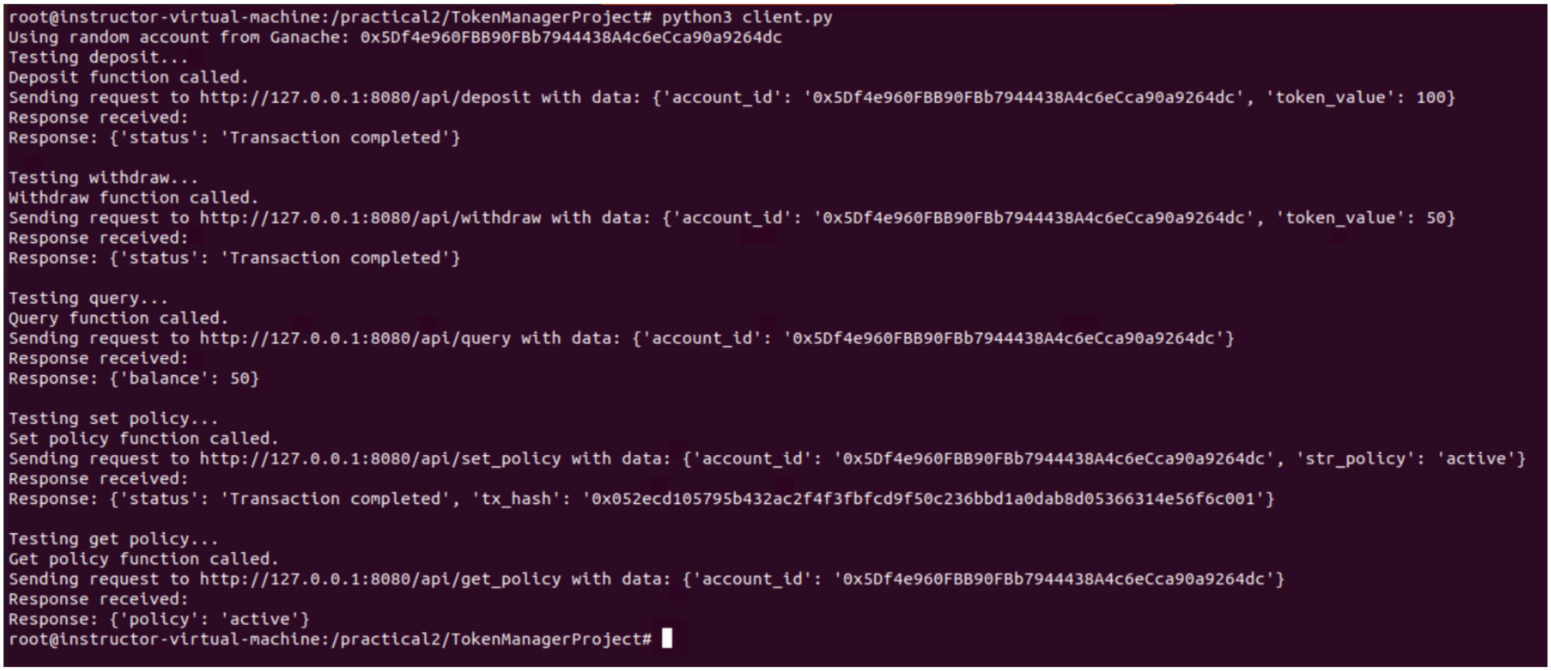


```
2_deploy_token_manager.js
=========================

   Replacing 'PolicyLibrary'
   -------------------------
   > transaction hash:    0xce4da274c83af59b3d194f3357cfbb56d579ac5bf1d12bf85c63dd63982f591b
   > Blocks: 0            Seconds: 0
   > contract address:    0xd3F7c6a71B1eea0BfeBd709afF4cA2e64A7cD177
   > block number:        173
   > block timestamp:     1710901914
   > account:             0x5Df4e960FBB90FBb7944438A4c6eCca90a9264dc
   > balance:             99.53572574
   > gas used:            203868 (0x31c5c)
   > gas price:           20 gwei
   > value sent:          0 ETH
   > total cost:          0.00407736 ETH


   Linking
   -------
   * Contract: TokenManager <--> Library: PolicyLibrary (at address: 0xd3F7c6a71B1eea0BfeBd709afF4cA2e64A7cD177)

   Replacing 'TokenManager'
   ------------------------
   > transaction hash:    0x17b02c0f648dfbecb6bbcfa70851c2f3acffc4a34c50679be86c262018d8ca75
   > Blocks: 0            Seconds: 0
   > contract address:    0xf45d8Ca9d1F5214E6F7AD2b596510720a1253DE6
   > block number:        174
   > block timestamp:     1710901914
   > account:             0x5Df4e960FBB90FBb7944438A4c6eCca90a9264dc
   > balance:             99.5218726
   > gas used:            692657 (0xa91b1)
   > gas price:           20 gwei
   > value sent:          0 ETH
   > total cost:          0.01385314 ETH

   > Saving artifacts
   -------------------------------------
   > Total cost:           0.0179305 ETH

Summary
=======
> Total deployments:   3
> Final cost:          0.02200786 ETH
```



```
TokenManager contract deployed at 0xf45d8Ca9d1F5214E6F7AD2b596510720a1253DE6
Starting the middleware...
Middleware started with PID 950782
Starting the gateway...
Gateway started with PID 950783
 * Serving Flask app 'gateway'
 * Debug mode: on
WARNING: This is a development server. Do not use it in a production deployment. Use a production WSGI server instead.
 * Running on http://127.0.0.1:8080
Press CTRL+C to quit
 * Restarting with stat
 * Debugger is active!
 * Debugger PIN: 260-955-525
 * Serving Flask app 'middleware'
 * Debug mode: on
INFO:werkzeug:WARNING: This is a development server. Do not use it in a production deployment. Use a production WSGI server instead.
 * Running on http://127.0.0.1:5000
INFO:werkzeug:Press CTRL+C to quit
INFO:werkzeug: * Restarting with stat
WARNING:werkzeug: * Debugger is active!
INFO:werkzeug: * Debugger PIN: 260-955-525
```

client.py



```
root@instructor-virtual-machine:/practical2/TokenManagerProject# python3 client.py
Using random account from Ganache: 0x5Df4e960FBB90FBb7944438A4c6eCca90a9264dc
Testing deposit...
Deposit function called.
Sending request to http://127.0.0.1:8080/api/deposit with data: {'account_id': '0x5Df4e960FBB90FBb7944438A4c6eCca90a9264dc', 'token_value': 100}
Response received:
Response: {'status': 'Transaction completed'}

Testing withdraw...
Withdraw function called.
Sending request to http://127.0.0.1:8080/api/withdraw with data: {'account_id': '0x5Df4e960FBB90FBb7944438A4c6eCca90a9264dc', 'token_value': 50}
Response received:
Response: {'status': 'Transaction completed'}

Testing query...
Query function called.
Sending request to http://127.0.0.1:8080/api/query with data: {'account_id': '0x5Df4e960FBB90FBb7944438A4c6eCca90a9264dc'}
Response received:
Response: {'balance': 50}

Testing set policy...
Set policy function called.
Sending request to http://127.0.0.1:8080/api/set_policy with data: {'account_id': '0x5Df4e960FBB90FBb7944438A4c6eCca90a9264dc', 'str_policy': 'active'}
Response received:
Response: {'status': 'Transaction completed', 'tx_hash': '0x052ecd105795b432ac2f4f3fbfcd9f50c236bbd1a0dab8d05366314e56f6c001'}

Testing get policy...
Get policy function called.
Sending request to http://127.0.0.1:8080/api/get_policy with data: {'account_id': '0x5Df4e960FBB90FBb7944438A4c6eCca90a9264dc'}
Response received:
Response: {'policy': 'active'}
root@instructor-virtual-machine:/practical2/TokenManagerProject# 
```

Included Files:

```
TokenManagerProject\
│
├───client.py
├───gateway.py
├───middleware.py
├───start_project.sh
├───requirements.txt
│
├───contracts\
│   ├───TokenManager.sol
│   └───PolicyLibrary.sol
│
├───migrations\
│   ├───1_deploy_policy_library.js
│   └───2_deploy_token_manager.js
│
├───build\
│   ├───contracts\
│   └─────PolicyLibrary.json
│           └──TokenManager.json
│
├───truffle-config.js
├───package.json
└───package-lock.json
```