# 1. Using Ethereum TestRPC and Web3js to interact with ethereum smart contract proj.sol at remix browser.

# 2. TestRPC and Web3js for providing UI interface and facilitate interaction with client.

# 3. Project dependencies are generated and stored at code\proj\ as json files named package.

# 4. Web3js is Etherium javascript API used to interact with a local node through RPC calls(TestRPC).

# 5. Smart contract at the Remix Browser executed with javascriptVM shown in pic.

# 6. To interact with user the smart contract run option is set to Web3provider and web3js.

# 7. And ethereum contact is run on port 8545.

# 8. Remix runs at https but interaction is to be done locally so we change the https of remix browser to http.

Developed webbased Dapp…

In module 3 folder video and web3js coded index.html file and along with its associated .css file

Also implementation without remix is also present in different folder along with difference in both implementations details…

# 

# 