**Unit 4 – Blockchain and Decentralized Applications**

**Lab 2 Manual**

**Introduction to Solidity Programming with VS Code and Truffle**

**Objective:**

In this lab, students will set up a local blockchain environment using Truffle and Ganache, deploy a simple smart contract, and interact with it through the Truffle console.

By the end of this lab, students will:

* Understand the basics of Solidity using VS Code.
* Write a simple smart contract and interact with it.
* Deploy and interact with the contract using VS Code and Ganache blockchain.

**Step 1:** Setting Up VS Code Environment

* Click on View menu, select Terminal from the drop down menu.
* Type *mkdir Lab2* by creating a folder
* Type *cd Lab2* to get into the directory
* Then, type *npm -v* to find out if you have npm installed if yes then type *npx create-react-app Lab2* to create a react application

A screen shot of a computer

Description automatically generated

* If the name of the project is in capital letters, I will throw an error, so always use the name in lowercase letter.

A screen shot of a computer

Description automatically generated

A screen shot of a computer program

Description automatically generated

* It asks for confirmation, press ‘y’.

A screenshot of a computer program

Description automatically generated

* After the installation is complete, close the terminal.
* Go to File menu and click on Open Folder. You can go to the directory. You will see the complete files created by the command in explorer area.
* A screenshot of a computer program

  Description automatically generated

**Step 2:** Install Ganache from [Ganache](https://archive.trufflesuite.com/ganache/)

A screenshot of a software

Description automatically generated

After installing, click on quick start and you will see the interface somewhat like this:

A screenshot of a computer

Description automatically generated

please do not close this application. We want it running at the backend.

**Step 3:** Come back to your VS Code terminal

Type *npm install truffle*

A screenshot of a computer screen

Description automatically generated

After the installation is complete, please close the terminal and open a new terminal.

**Step 4:**

Type *npm install -g truffle* to install truffle in your environment. Sometimes it gives issues so, please permit your running scripts enabled on your system.

Set the execution policy:

set-ExecutionPolicy -Scope CurrentUser -ExecutionPolicy RemoteSigned

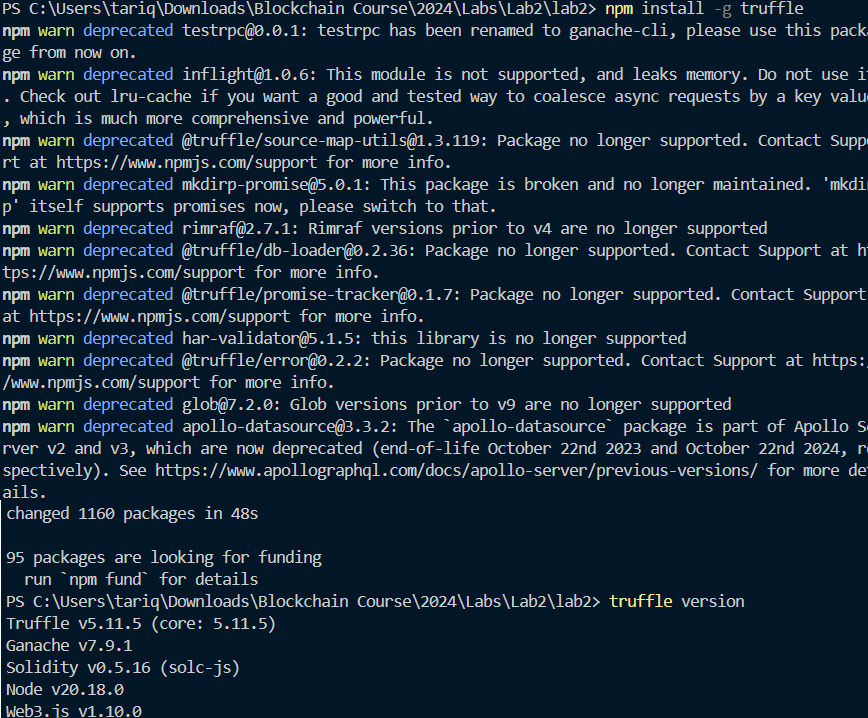
A screenshot of a computer screen

Description automatically generated

Revert the execution policy:

*set-ExecutionPolicy -Scope CurrentUser -ExecutionPolicy Restricted*

The truffle will be installed successfully



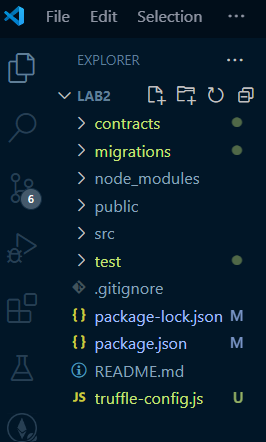
Now,

Initialize truffle by typing *truffle init*

A computer screen shot of white text

Description automatically generated

After initializing, you will see the truffle-config.js file in the explorer tab.



**Step 5:** Open Ganache, go to settings and add your project to it by adding the truffle-config.js file.

A screenshot of a computer

Description automatically generated

Then, click on save and restart.

Go back to VS Code terminal and type *truffle console*

You will see that the truffle console is open with ganache

A blue background with white text

Description automatically generated

Add a file to your contract and save it:

A screenshot of a computer screen

Description automatically generated

Or if want, you can add the contract which we have compiled and executed in our first lab.

In the truffle console now you can type *compile*

A screenshot of a computer

Description automatically generated

Now, if you see the Ganache, it holds your compiled contract

A screenshot of a computer

Description automatically generated

Since our contract is compiled successfully, now type *.exit* to exit from the truffle console environment.

Soon as you exit, then type *truffle migrate -reset*

A computer screen shot of a computer code

Description automatically generated

**Step 6:**

To verify get again into the *truffle console* by typing it in the terminal

In the console, type *const acc=await TestContract.deployed()* OR

*const instance = await TestContract.deployed();*

Now your contract is deployed successfully.

**Task:**

"🚀 Challenge Lab: Decentralized Dev in Action! 🚀

This week, you’re invited to dive into Truffle and Ganache on VS Code—getting hands-on with core blockchain tools! Here’s the URL you’ll need to get started: <https://ferdyhape.medium.com/remix-ide-and-ganache-a-beginners-guide-to-smart-contract-deployment-b0df68c48ae6>

While this isn't a graded task, those who complete it will be rewarded with chocolate 🍫 as a small bonus. Plus, who knows? If the professor is impressed, it *might* just turn into a credited assignment!

So, take this opportunity to explore freely, experiment with settings, and start building like a true blockchain developer. We’ll meet up later to chat about your findings and any new insights. I can't wait to see where you take it—have fun, and happy coding!"