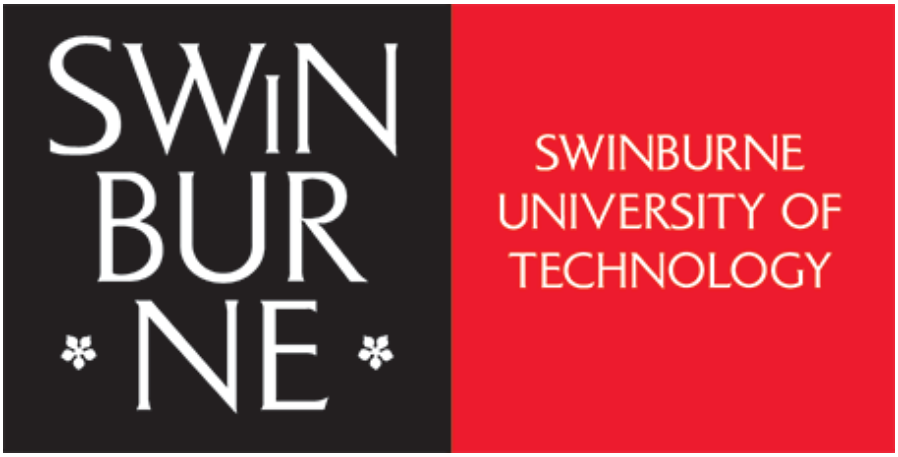


2023-COS30049-Computing Technology Innovation Project

Workshop Guide

Note: It's crucial to modify the distribution of time according to your specific requirements. You might also need to devise your own PowerPoint presentation in line with the guidelines of the workshop.

Workshop 01

Blockchain, Bitcoin, and Ethereum

Objective: By the end of this workshop, students should have an understanding of the basic concepts of blockchain, Bitcoin, and Ethereum, and be able to answer questions and engage in discussions on these topics.

Workshop Structure:

1. Ice-breaker (20 minutes): introduce yourself and the students' introduction.
2. Introduction to the unit (30 minutes) emphasizing the unit structure, assessments, and unit pass criteria. Lastly, let students select groups.

    - About group selection and assignment description

      Every student is required to register for one group from a predefined set, with each group comprising three students. During the first week of this semester, students will have the opportunity to autonomously select the group they wish to join. For each group set, the three consecutive projects are listed below:

      For group set 1, which prefers students that have software engineering background, we have

Assignment 1-1: Decentralised Trading Platform - Static Website (Front-end) (weight: 30%),

Assignment 1-2: Decentralised Trading Platform - Dynamic Website (Back-end) and Smart Contract Development (weight: 40%),
Assignment 1-3: Smart Contract Code Review and Illustration Report (weight: 30%).

“Transaction Tracing Platform”
A web-based service or application designed to provide users with the ability to track and trace blockchain transactions.

Here are several features and functionalities that such a platform might offer, but the list is not exhaustive:

1. Address Information: The platform might allow users to explore specific addresses to view their transaction history, token balances, and interactions with smart contracts.
2. Transaction Analytics: The platform might provide analytical tools and visualisations to help users investigate transaction details and patterns. (e.g., It can show the transaction interaction among different addresses)
3. User-Friendly Interface: The platform would likely have an intuitive and user-friendly interface to cater to both inexperienced and experienced users.
4. etc.

For group set 2, which prefers students that have cybersecurity background, we have

Assignment 2-1: Security Auditing Platform - Static Website (Front-end) (weight: 30%),

For this assignment, students are expected to develop a static website that serves as the user interface for the security auditing platform. The primary objective is to demonstrate a clear understanding of front-end web technologies such as HTML, CSS, and JavaScript (with libraries like jQuery or Bootstrap), and their application to create a user-friendly, responsive, and intuitive interface.

Assignment 2-2: Security Auditing Platform - Dynamic Website (Back-end integrated) and Application of Contract Audit Tools (weight: 40%),

Building on the front-end development in Assignment 2-1, this assignment requires the integration of the back-end, thereby transforming the platform into a fully functional dynamic website. The back-end should be developed using Node.js and should handle requests and responses related to the submission of contracts for audit and the retrieval of audit results. The platform should now be integrated with a contract audit tool (for example, Slither) that will perform the actual audit of the submitted.

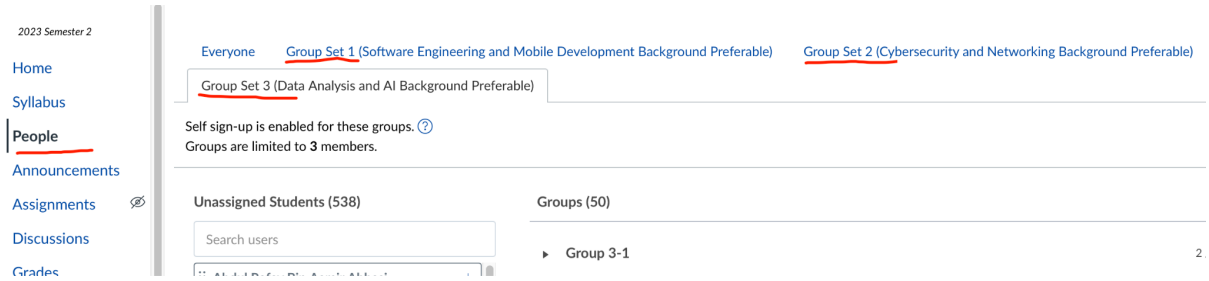
Assignment 2-3: Security Case Study and Auditing Report (weight: 30%).

For this assignment, you are expected to select a real-world case of a security breach or vulnerability exploitation in a blockchain-based application, ideally in a smart contract, and conduct a thorough case study and audit report on the incident. This assignment aims to assess your understanding of blockchain security concepts, your ability to identify and analyze vulnerabilities, and your ability to clearly and effectively communicate complex information.

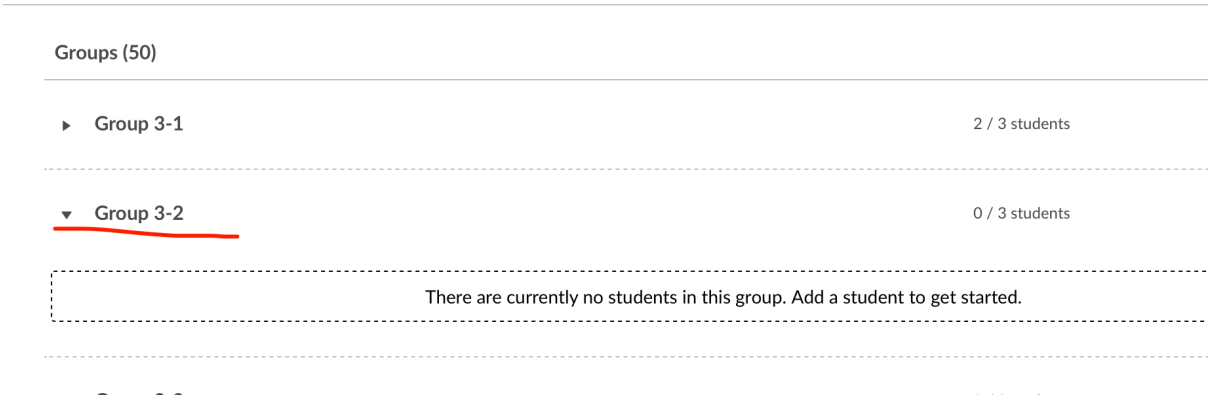
For group set 3, which prefers students that have data analysis and AI background, we have

Assignment 3-1: Transaction Tracing Platform - Static Website (Front-end) (weight: 30%),
Assignment 3-2: Transaction Tracing Platform - Dynamic Website (Back-end integrated) (weight: 40%),
Assignment 3-3: Transaction Analysis and Tracing Report (weight: 30%).

Step 1: go to People section, and select a group set.



Step 2: To join a group from the list, simply select the one that best aligns with your preferences and click to confirm your choice. Feel free to switch between different groups until you find the right fit. However, please note that once you have submitted the first assignment, group changes will no longer be allowed. Within each group, one student will be designated as the group leader, responsible for coordinating and leading the team's projects.



3. Introduction (10 minutes): Start by explaining the key concepts of blockchain, Bitcoin, and Ethereum.
    - Blockchain: A type of database that stores information in blocks that are chained together. This system is decentralized, meaning it doesn't rely on a single authority.
    - Bitcoin: A digital or virtual cryptocurrency created in 2009, based on blockchain technology. It's decentralized and uses peer-to-peer technology for instant payments.
    - Ethereum: A global, open-source platform for decentralized applications. It uses its native cryptocurrency, Ether (ETH), and enables Smart Contracts and Distributed Applications (ĐApps) to be built and run without any downtime, fraud, control, or interference from a third party.

4. Quiz - Multiple Choice Questions (MCQs) (15 minutes):
    - Which of the following is a key feature of a blockchain?
      a) Centralized System
      b) Decentralized System
      c) Both a and b
      d) None of the above

    - Which technology is Bitcoin based on?
      a) Internet of Things (IoT)
      b) Artificial Intelligence (AI)
      c) Blockchain
      d) Augmented Reality (AR)

    - What can Ethereum be used to build?
      a) Decentralized Applications
      b) Smart Contracts
      c) Cryptocurrencies
      d) All of the above

- Which of the following is the native cryptocurrency of the Ethereum platform?
  a) Litecoin
  b) Ripple
  c) Ether
  d) Bitcoin

- What is the primary purpose of mining in Bitcoin?
  a) To generate new Bitcoins
  b) To process and validate transactions
  c) To keep the network secure
  d) All of the above

- What is a "smart contract"?
  a) A legal document for technology companies
  b) A self-executing contract with the terms of the agreement directly written into code
  c) A contract signed between two cryptocurrency miners
  d) A software program that controls the transfer of digital currencies or assets between parties under certain conditions

- Which of the following is an advantage of blockchain technology?
  a) High scalability
  b) Immutability
  c) Centralization
  d) Low security

- What programming language is predominantly used to write Ethereum smart contracts?
  a) Solidity
  b) Python
  c) JavaScript
  d) C++

- Which of the following cryptographic algorithms is primarily used in the Bitcoin blockchain for the generation of public and private keys?

  a) RSA (Rivest-Shamir-Adleman)
  b) ECC (Elliptic Curve Cryptography)
  c) AES (Advanced Encryption Standard)
  d) DES (Data Encryption Standard)

- How does the Ethereum network handle "gas" in a transaction that runs out of gas before it's complete?

  a) It stops the transaction and returns the remaining gas to the sender's account
  b) It allows the transaction to complete but marks it as invalid
  c) It stops the transaction and consumes all the provided gas
  d) It pauses the transaction until the sender's account has enough gas

- What is the primary role of Merkle trees in blockchain technology?

  a) They are used to control the creation of new blocks.
  b) They serve to confirm transaction validity within a block.
  c) They function as a random number generator for each transaction.
  d) They are used to keep track of the number of miners on the network.

  Explanation: Merkle trees, named after Ralph Merkle, are used in blockchains to efficiently summarize all the transactions in a block, allowing for efficient and secure verification of the contents of large data structures. If a single detail in any of the transactions or the order of the transactions changes, so does the Merkle root. This makes it easy to verify if a particular transaction is included in a block (through a process called "Merkle proof") without having to hold the entire list of transactions.

5. Group Discussion (20 minutes): Divide students in to groups of 3-4 and assign them a topic to discuss. Some topic suggestions are:

   a. *How has blockchain technology impacted the financial industry?*

   b. *What are the potential advantages and challenges of cryptocurrencies like Bitcoin?*

   c. *How is Ethereum different from Bitcoin and what are its potential applications?*

6. Step-by-Step Guide to Using Remix for Ethereum Smart Contract Development (20 minutes)

   a. Accessing Remix:
   Open your web browser and navigate to the Remix Ethereum IDE website at https://remix.ethereum.org.

   b. Creating a New File:
   On the left side of the screen, click on the second icon from the top to open the "File explorers" tab. Click on the "+" icon to create a new file. Give it a name ending with .sol, such as MyContract.sol.

   c. Writing the Smart Contract:
   In the text editor area in the center of the screen, you can start writing your Solidity smart contract. For instance, here is a simple smart contract:

   ```
   ----------------------------------------------------------------
   ----------------
   solidity
   pragma solidity ^0.5.1;
   ```

```
contract MyContract {
    string public myString = "Hello, world!";

    function setMyString(string memory newString) public {
        myString = newString;
    }
}
```

---

d. Compiling the Smart Contract:
   Click on the fourth icon from the top on the left side to open the "Solidity compiler" tab. Under the compiler section, select the appropriate compiler version that matches your contract (in this case 0.5.1+commit.c8a2cb62 or higher) and then click on the "Compile" button.

e. Deploying the Smart Contract:
   Click on the third icon from the top on the left side to open the "Deploy & run transactions" tab. Choose the correct contract from the "Contract" dropdown (if you have multiple in your file), then click "Deploy".

f. Interacting with the Smart Contract:
   Once deployed, the contract will appear under the "Deployed Contracts" section. Here you can interact with your contract's functions. For the example contract, you could change the greeting by entering a new string in the "setMyString" field and clicking the "Transact" button, and then see the updated greeting by clicking the "myString" button.

7. Reflection and Closing (5 minutes): Give students the opportunity to share what they learned, found interesting, or had difficulty understanding. Offer additional resources for them to learn more about blockchain, Bitcoin, and Ethereum.