Assessment Project

Full Development Report

SEC-205: Distributed Ledger and Blockchain

Student Name: [Put your first name and last name here] ([Put your nickname here])

Student Email: [Put your CMKL email address here]

Chapter 1: Design Specification

Application Name: [Put the name of your selected application here, e.g., a voting application]

Application Description [10 Points]:

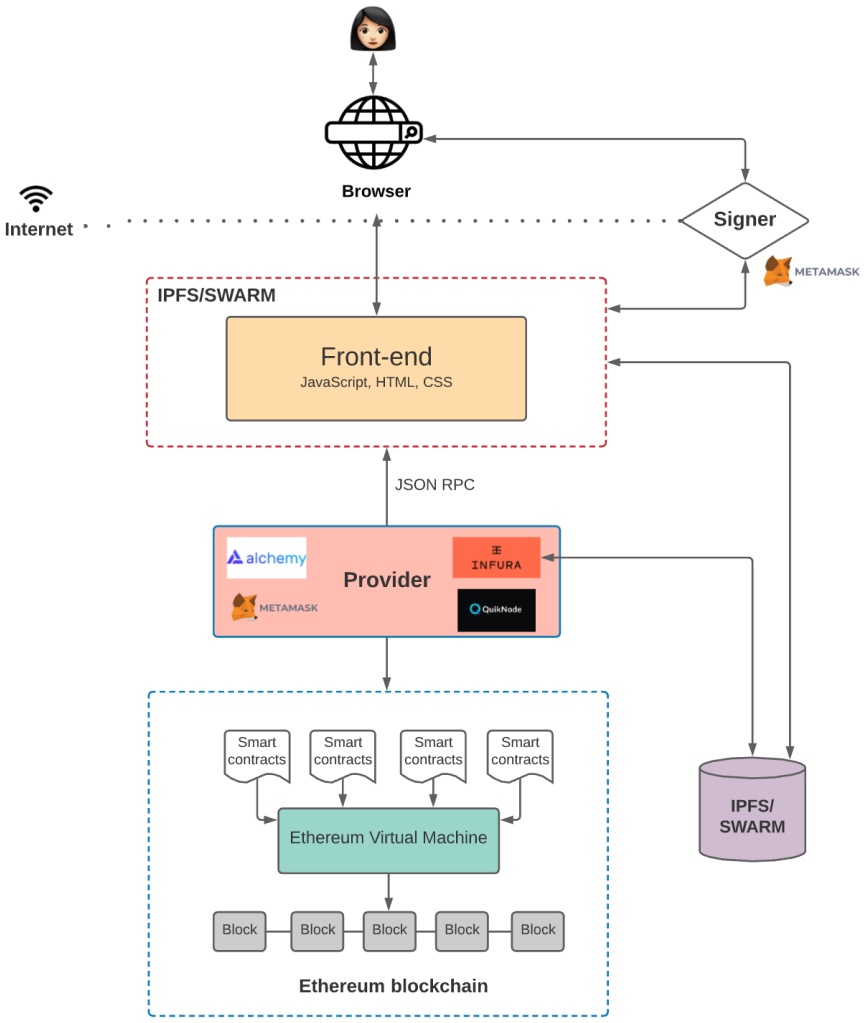
Write a section of paragraphs to comprehensively describe the Web 3.0 application that you selected for the assessment. This section should serve as an executive summary, providing a high-level overview of the application functionality, and objectives. The following questions must at least be answered:

1. What are the purposes and objectives of the selected Web 3.0 application in real-world situations?
2. Who will be users of this application when it launches to the real market?
3. Does this Web 3.0 application fit with the blockchain technology? If so, why?
4. Why do you select this application for your assessment?

Application Architecture [15 Points]:

Create a contextual diagram or deployment diagram to describe the architecture of your selected application. The diagram must clearly illustrate how your Web 3.0 application is structured in terms of modules and smart contracts. It should represent the overall system architecture and demonstrate how the infrastructure is prepared to support the application's operational environment. An example of a deployment diagram is shown in the figure below.

In addition, write a section of paragraphs that comprehensively explains the diagram(s) in detail. The description must cover all system components and clearly explain the interactions and data flow among them. It should be self-contained and written in a way that readers without prior knowledge of Web 3.0 applications can easily understand the overall architecture and workflow.



System Requirements [15 Points]:

Write a list of functional and non-functional requirements (e.g., security, privacy, scalability, adaptability, etc.) that reflect the commitments your developed Web 3.0 application must satisfy. Each requirement must be clearly written, unambiguous, and measurable.

The template for the requirements list is shown below:

Functional Requirements:

1. [Insert your first functional requirement here]
2. [Insert your second functional requirement here]
3. ...

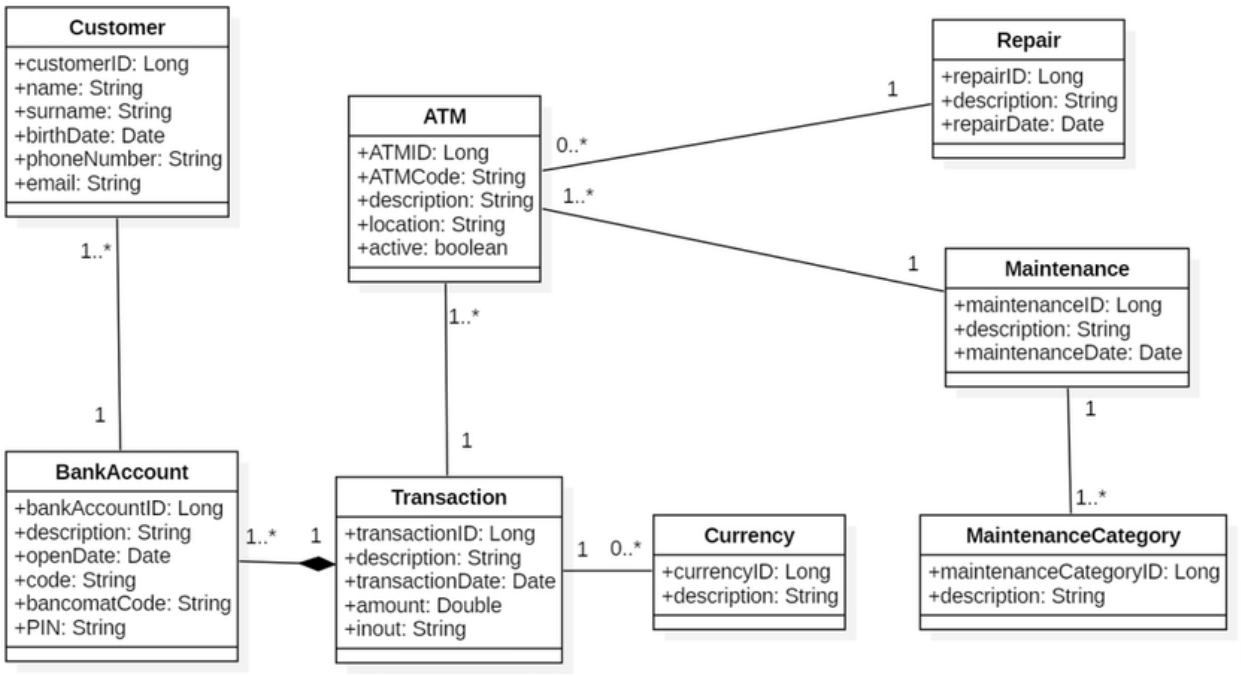
Non-Functional Requirements:

1. ([Specify the type of non-functional requirement in parentheses]) [Insert your first non-functional requirement here]
2. **Example:** (Security) [Insert your second non-functional requirement related to security here]
3. ...

Data Structure for Off-Chain and On-Chain Storage [15 Points]:

Create a class diagram or an Entity-Relationship (ER) diagram to illustrate the data structure of your developing Web 3.0 application. The diagram must clearly distinguish between off-chain databases and on-chain storage. You are required to carefully consider the different objectives and constraints of each storage type (e.g., cost, transparency, immutability, and performance). An example of a class diagram is provided below.

In addition, write a section of paragraphs that comprehensively explains the diagram. The description must cover all classes or entities presented in the diagram. The relationships among classes or entities must also be clearly described, including multiplicity and relationship labels. The explanation should be self-contained and written in a way that readers without prior knowledge of Web 3.0 applications can easily understand the overall data structure and design rationale.



Use Case Scenarios [15 Points]:

Create a use case diagram to illustrate all actors and use cases of the developing Web 3.0 application. The diagram must include both internal and external actors, and the associations between use cases should clearly represent their interactions and dependencies.

Then, write a set of use case descriptions to explain the details and significance of each use case. There must be **one use case description for each use case** presented in the diagram. The descriptions should be self-contained and written in a way that readers without prior knowledge of Web 3.0 applications can easily understand the system's functionality and interactions. The following table may be used as a template for defining each use case description.

Element	Detail
Name	The name of the use case. Typically written in the format: <action> + <object>
ID	A unique identifier for each use case (e.g., UC-001).
Description	A brief description explaining what the user intends to accomplish and the benefit he/she will obtain.
Actor	The type of user who interacts with the Web 3.0 application to accomplish a task. Actors should be identified by their role names.
Frequency of Use	How often this use case is expected to be executed.
Triggers	Specific actions performed by the user/actor within the Web 3.0 application that initiate the use case.
Preconditions	The states or conditions that must be satisfied before the use case can begin. If multiple postconditions exist, list them in bullet form: 1. [Insert first precondition] 2. [Insert second precondition] 3. ...
Postconditions	The states or conditions that must be satisfied after the successful completion of the use case. These conditions are fulfilled if the Main Course (Normal Flow) is executed successfully. Exceptions may prevent the postconditions from being achieved. If multiple postconditions exist, list them in bullet form: 1. [Insert first postcondition] 2. [Insert second postcondition] 3. ...
Invariants	The states or conditions that must be satisfied after the successful completion of the use case. These conditions are fulfilled if the Main Course (Normal Flow) is executed successfully. Exceptions may prevent the postconditions from being achieved. If multiple postconditions exist, list them in bullet form: 1. [Put your first invariant here] 2. [Put your second invariant here] 3. ...
Main Course/Normal Flow	The standard sequence of interactions between the user and the application: 1. Step 1 2. Step 2
Alternate Course/Flow	An alternative execution path. AC1: <condition for the alternate to be called> 1. Step 1 2. Step 2 AC2: <condition for the alternate to be called> 1. Step 1 2. Step 2
Exception Course/Flow	Exception handling scenarios managed by the Web 3.0 application. EX1: <condition for the exception to be called> 1. Step 1 2. Step 2 EX2: <condition for the exception to be called> 1. Step 1 2. Step 2

Smart Contract Structure [15 Points]:

Write a list of smart contracts that explains the overall structure of the smart contract environment for the developing Web 3.0 application. Each smart contract must be clearly described, including its purpose, objectives, and internal functions. For each function, specify its input parameters and expected output values (if any).

The following template may be used to structure your response:

- **[First Smart Contract Name]:** [Provide a detailed description of this smart contract, including its purpose, responsibilities, and role within the overall application.]
    - **[First Function Name]:** [Describe the function's purpose, input parameters, return values (if applicable), and any important logic or conditions.]
    - **[Second Function Name]:** [Describe the function's purpose, input parameters, return values (if applicable), and any important logic or conditions.]
    - ...
- **[Second Smart Contract Name]:** [Provide a detailed description of this smart contract, including its purpose, responsibilities, and role within the overall application.]
    - **[First Function Name]:** [Description of the function, including inputs and outputs.]
    - ...

Test Cases [15 Points]:

Create a comprehensive set of test cases to thoroughly evaluate the developing Web 3.0 application, with particular emphasis on the smart contract modules. The test cases must provide sufficient coverage and include all critical use cases, as well as relevant security and privacy scenarios.

The following table may be used as a template for documenting each test case:

Test Title	Priority	Test Case ID	Test Version Number	Test Case Date
The title of the test case.	H/M/L	A unique identifier of the test case.	The version indicator of the test case	The date that this test case has been created.
Test Description		**Test Designed By**	**Test Executed By**	**Execution Date**
A brief description explaining what the purposes of this test case or what aspect are to be evaluated with this test case.		The name of the designer of this test case.	The name of the tester who takes this test case to the execution.	The date that this test case has been recently executed.
Test Actors	**Test Dependencies**	**Test Conditions**	**Test Control**	
The list of actors who will relate to this test case	The list of dependencies that facilitate the execution of this test case.	The list of conditions to be met in order to execute this test case.	The list of control that has to be in place in order to execute this test case.	

Step ID	Step Description	Test Date	Expected Results	Actual Results	Pass/Fail	Additional Notes
1.	A brief description of the first test step.	The date this step is executed.	A list of expected results.	A list of actual results observed from the Web 3.0 application.	Pass/Fail	Remarks
2.	...	...	...	...	...	...
3.	...	...	...	...	...	...

Chapter 2 – Implementation of the Web 3.0 Application

Link to the Video File [80 Points]:

[Link to the cloud storage or the streaming platform where the presentation video is accessible.]

Link to the Git Repository for the Project Source Code [20 Points]:

[Link to GitHub or other Git repository that the project source code is accessible.]

Chapter 3 – Security and Privacy Analysis

Security Analysis [50 Points]:

Write a section of paragraphs to thoroughly analyze the current security status of the developing Web 3.0 application. Your analysis must align with the blockchain layered model discussed during the lecture (e.g., network layer, consensus layer, smart contract layer, and application layer). In addition, the application must be evaluated against common blockchain-specific attacks.

At a minimum, this section must address the following questions:

- Do you consider the application secure in terms of data protection? Why or why not?
- Are any personal data elements published on the blockchain? If so, what are they and what are the associated risks?
- Are security mechanisms implemented in the application?
    - If yes, what mechanisms are in place and how do they provide protection? Do they address security at all relevant blockchain layers?
    - If no, what potential security risks could threaten the application? What improvements would you recommend to enhance data security and overall system protection?

The analysis should be comprehensive, technically sound, and clearly justified.

Privacy Analysis [50 Points]:

Write a section of paragraphs to thoroughly analyze the privacy preservation status of the developing Web 3.0 application. The analysis must align with the topics discussed during the lectures (e.g., user anonymity, pseudonymity, unlinkability, and confidentiality). In addition, the application must be evaluated against potential privacy breaches or attacks to ensure a comprehensive assessment.

At a minimum, this section must address the following questions:

- Does the application use or store any personal data or sensitive information? If so, what types of data are involved?
- How does the application ensure anonymity, conditional privacy, and selective disclosure? If it does not fully support these properties, how would you recommend improving it?
- Various privacy-preserving techniques exist (e.g., encryption, hashing, zero-knowledge proofs, off-chain storage, mixers, etc.). Which techniques are already implemented in the application, and which are not? Provide justifications for your choices. Use the following table to support your analysis.

Techniques	Included?	Justifications
Zero Knowledge Proof	No	This technique is not included in our application b/c ...
...	...	...

The discussion should be analytical, well-justified, and technically grounded.