BAHRIA UNIVERSITY KARACHI CAMPUS

Department of Software Engineering

COURSE: SEL 401
CLOUD COMPUTING
PROJECT PROPOSAL

CLASS: BSE – 6A (SPRING - 2024)

Smart Contract Auditor

Group Members

S. No.	Name	Enrollment #
01	Rizwan Akram (team lead)	02-131212-026
02	Abdul Hannan	02-131212-085

Submitted to:

Course Instructor: Engr. Muhammad Faisal
Lab Instructor: Engr. Noor us Sabah

Date: 2/05/2024

1. INTRODUCTION & BACKGROUND

Smart contracts are fundamental to **decentralized finance** and **blockchain technologies**, serving as the backbone of numerous applications. However, vulnerabilities in these contracts pose significant risks, leading to financial loss, security breaches, and loss of trust in blockchain ecosystems. This project proposes a comprehensive solution by building a "**Smart Contract Auditor**," a React application that audits **Solidity-based** smart contracts for vulnerabilities. The application will leverage an **AI model fine-tuned** on a dataset of vulnerable smart contracts, offering precise vulnerability detection and comprehensive reports.

2. PROBLEM STATEMENT

The inherent complexity of smart contracts and the need for manual auditing make them prone to errors and security flaws. Current auditing solutions often lack automated and comprehensive tools, resulting in missed vulnerabilities and potential losses. Thus, a robust, automated solution is necessary to detect vulnerabilities efficiently and generate comprehensive reports to mitigate potential risks.

3. PROPOSED SOLUTION

3.1. FEATURES OF THE PROJECT

1. **AI-Powered Vulnerability Detection:** The project utilizes a fine-tuned AI model, possibly **LLaMA 3** or another open-source model, trained on a dataset of vulnerable smart contracts. This model identifies security flaws with high accuracy, reducing manual labor and human error.
2. **Comprehensive Reporting:** Upon detecting vulnerabilities, the application generates detailed reports, offering insights into the type and severity of each vulnerability, along with mitigation strategies.
3. **React-Based Interface:** The application provides a user-friendly interface built on React, allowing for easy navigation and intuitive interaction with the tool.

3.2. METHODOLOGY

1. **Data Mining:** The project leverages a dataset of smart contracts, categorized by their vulnerabilities, to train and fine-tune the AI model. Data mining techniques extract relevant patterns and insights from the dataset to improve model performance.
2. **AI Training:** The AI model undergoes fine-tuning on the dataset, learning to detect vulnerabilities and provide precise outputs. The model's performance is evaluated and iteratively refined to ensure optimal results.
3. **Cloud Deployment:** The application and AI model are deployed on **Microsoft Azure**, ensuring reliable, scalable, and secure access for users worldwide.

3.3. TECHNOLOGIES TO BE USED

1. **React:** For building the application's frontend, providing an intuitive user experience.
2. **Python:** For backend development, including model training, data processing, and integration.
3. **LLaMA 3 or Similar AI Model:** For fine-tuning on the vulnerability dataset, detecting security flaws.
4. **Microsoft Azure:** For deploying the model and application, ensuring scalability and reliability.

4. PROJECT SCOPE

The **Smart Contract Auditor** is intended for organizations and individuals engaged in blockchain development, particularly those developing or maintaining Solidity-based smart contracts. By providing automated auditing, the project aims to minimize vulnerabilities, reduce financial risks, and promote trust in blockchain technology.

5. PROJECT ABSTRACT

This project proposes a **Smart Contract Auditor**, an AI-powered React application that audits Solidity-based smart contracts for vulnerabilities. By leveraging an AI model fine-tuned on a dataset of vulnerable smart contracts, the project offers comprehensive, automated auditing, generating detailed vulnerability reports. The tool is deployed on Microsoft Azure, providing secure and scalable access. It integrates data mining techniques, AI training, and cloud deployment to offer a robust solution, catering to blockchain developers and organizations.

6. MODULE DISTRIBUTION

Frontend Development (Abdul Hannan)

1. **Frontend Development:** Build a React-based interface that allows users to interact intuitively with the application. This includes designing user-friendly navigation, input forms for uploading smart contracts, and output displays for vulnerability reports.
2. **Backend Integration:** Work with Person 1 to integrate the backend with the frontend application, ensuring seamless communication for processing data and generating reports.
3. **Cloud Deployment:** Deploy the frontend application on Microsoft Azure, providing secure, scalable, and reliable access to users.

AI Model Training (Rizwan Akram)

1. **Data Preparation:** Collect a dataset of vulnerable smart contracts, process it, and categorize it according to their vulnerabilities. This person will apply data mining techniques to extract relevant patterns and insights from the dataset, ensuring it is ready for training.
2. **AI Model Training:** Fine-tune the chosen AI model (such as LLaMA 3 or a similar model) on the dataset, iteratively evaluating its performance. This includes modifying hyperparameters, training strategies, and refining the model to accurately detect vulnerabilities.
3. **Backend Integration:** Collaborate with Person 2 to integrate the trained AI model with the frontend application, ensuring seamless communication between the backend and frontend for smooth data processing and report generation.
4. **Cloud Deployment:** Deploy the AI model on Microsoft Azure, making it accessible for the application to use.

5. REFERENCES

- Research papers on data mining techniques for smart contract vulnerabilities.
- AI models fine-tuned for vulnerability detection, including LLaMA 3.
- Guides on deploying applications and models on Microsoft Azure.

Teacher's Signatures: _____

Remarks: _____

_____

_____