Assignment 3

Due Date: December 4, 2023 at 12:30 PM

Overview

This assignment will be worth **10%** of your total grade. The goal is to deploy and connect to a smart contract to create a full-stack web3 application. If you do not delete your node_modules folder before submitting, you will receive an automatic 10% deduction.

Specifications

Part I - 25 points

Create a simple college career fair smart contract named "CareerFair" for Towson. This contract must be deployed to Sepolia's testnet.

For this career fair system, you will have seven total companies on initialization:

- Amazon
- Google
- Apple
- Microsoft
- Meta
- Gemini
- SecureEd

The career fair system will store information for all the students who have enrolled in the career fair. After creating the contract, we should add the owner of the career fair system. Below are the minimum function requirements for the career fair contract:

- *enroll* (): Save when a student enrolls for the career fair. If a student is currently enrolled, they cannot enroll again
- *add* (companyName): Only the owner can add a new company to the career fair contract. The owner cannot add a company that already exists (same companyName)

    - companyName: a string

- *getAttendees* (): Returns all of the student addresses enrolled in the career fair
- *unenroll* (): a student can unenroll from the career fair. A student cannot unenroll unless they are already enrolled

Feel free to add appropriate events/modifiers for the required functions or to add extra functions to enhance your contract.

Part II - 70 points

Build a simple React project named "careerfair-frontend" that communicates with your deployed "CareerFair" contract. The specifications are as follows:

- A button that shows "Connect Wallet". This button should:
    - If clicked, allow a user to connect their wallet to the app. If successful, remove the button from the UI. If unsuccessful, display an alert saying "Wallet could not be connected"
    - console.log if "No ETH wallet detected" if no ETH object is detected and should console.log "ETH detected" if ETH object is detected
- A button that shows "Add Company" **and** a textfield that:
    - Allows the owner of the contract to enter in a company name to be added. If unsuccessful, display an alert saying "Only the owner can add a company"
- A button that shows "Enroll" that allows the student to enroll
    - Display an alert if the student is already enrolled saying "You are already enrolled in the career fair"
- A button that ahows "Unenroll" that allows the student to unenroll
    - Display an alert if the student is not already enrolled saying "You are not enrolled in the career fair"
- A button that shows "See Attendees" that displays a list of all registered addresses
    - Display "no one is enrolled" if no students are registered for the career fair

Notes:

- I do not care about styling or layout. Five simple buttons, a textfield, and a label (or list of labels) is all that needs to be there

- Delete the node_modules folder before submitting since it is a very large folder. You will lose 10% of your project grade if you do not do this
- I should be able to *npm install* then *npm start* your create-react-app project. Ensure you are also submitting your ABI file within the project so it runs smoothly when downloaded. If you use another tool/leave out the necessary files, you will lose 10% of your project grade
- Do not submit your hardhat.config or anything other than the contract from your hardhat env

Part III - 5 points

Create an a3p3.pdf that has:

- The transaction hash of the deployed contract on Sepolia
- The hash of the actual contract on Sepolia

Submission

Please submit the following on Blackboard (you may need to .zip the contents into a single folder):

- careerfair.sol
- careerfair-frontend with the exact configurations to be downloaded and run (including the ABI file, but excluding node_modules)
- a3p3.pdf