UGANDA CHRISTIAN UNIVERSITY

FACULTY OF ENGINEERING, DESIGN, AND TECHNOLOGY

SECOND-YEAR, BACHELORS OF SCIENCE IN COMPUTER SCIENCE

EASTER SEMESTER EXAMINATION

IN

WEB PROGRAMMING (TAKE HOME - PRACTICAL)

Date: April 2023

Instructions:

- This is a **take-home** examination.
- **Attempt all questions.**
- A "take-home" exam is "a recognized alternative form of assessment where students take their assessment off-campus within a specified timeframe. It does not require students to necessarily take the test in their own home".
- The university has important rules for exams. Please carefully read the instructions below. Failure to comply with any of the instructions below may result in our being unable to accept or grade your exam or initiating disciplinary actions.

---

a) This exam is an "open book," which means you are permitted to use any materials handed out in class, your notes from the course, the textbook, and anything on the UCU learning management system.

b) The exam must be taken completely alone. Showing it or discussing it with anyone is forbidden.

c) Please don't consult with any other person regarding the exam. Don't check your exam answers with any person.

QUESTION

BACKGROUND

Google Developer Student Clubs (DSCs) are community groups for university and college students interested in learning about Google technologies and developer tools. These clubs are open to any student, regardless of their academic background or major.

DSCs are organized and supported by Google Developers, who provide resources and guidance to help students learn, grow, and build projects using Google's technologies. DSCs offer a variety of activities, including workshops, study jams, speaker sessions, hackathons, and hands-on projects.

The primary goal of DSCs is to help students become better developers by providing them with opportunities to learn from experienced developers, collaborate with other students, and build real-world projects using Google's technologies. Through these activities, students can gain valuable skills, build their portfolios, and prepare for careers in the tech industry.

DSCs are present in universities and colleges all around the world and are a great way for students to connect with like-minded individuals, learn new skills, and make an impact in their communities through technology.

*Using the above context, imagine that you are the **technology lead** for the club chapter of Uganda Christian University. The core team has decided to create a simple Web Application called **DSC_UCU_Community** for the **management*** of its members. The application should have features in the specification below.

FEATURES SPECIFICATIONS FOR DSC UCU COMMUNITY WE APPLICATION.

a) **Collect Member Information:** The application should collect basic information from the user, such as their name, email address, phone number, and any other relevant details that the club wants to gather.
b) **Validate Information:** Using **regular expressions** or any other method, the application should validate the information provided by the user to ensure that it is accurate and complete. For example, the application should verify that the user has provided a valid email address or phone number.
c) **Store Information:** The application should store the member's information in a database or other data storage system like **local storage** so that the club can access it later.
d) **Display Confirmation:** The application should display a confirmation message to the user after they have completed the registration process. This message should include a summary of the information that they have provided and any other relevant details about the club.
e) **Authenticate Users:** The application should allow users to authenticate themselves by logging in with their email and password or through a social media account. In the addition to this; when a user logs in, he/she should be presented with a simple welcome screen that contains a proper representation of his/her profile from the details they provided during registration.
f) **Provide Access to Member-only Content:** The application should be able to grant registered members access to any exclusive content, features, or benefits that the club provides.
g) **Manage Membership:** The application should be able to track and manage the membership of the club. For example, it should be able to show the number of registered members, and their status and allow for member updates, deletions, or other modifications.
h) **Send Email Notifications:** It should be able to send an email confirmation to the user after they have completed the registration process, and any other email notifications related to club activities or other updates.

USER ROLES

Below are some of the common user roles in the Google Student Developers Club:

a) **Club Member:** This is an ordinary member of the club.
b) **Club Lead:** The Club Lead is responsible for leading and managing the overall club, ensuring that it runs smoothly and that all members are engaged and productive.
c) **Event Coordinator:** The Event Coordinator is responsible for organizing and managing events and activities for the club. This includes scheduling, logistics, and outreach to members**.**
d) **Marketing Lead:** The Marketing Lead is responsible for promoting the club and its events to the wider community. This can include social media, flyers, and other promotional materials.
e) **Technical Lead:** The Technical Lead is responsible for overseeing the technical aspects of the club, including website development, app development, and other technical projects.
f) **Mentorship Lead:** The Mentorship Lead is responsible for providing guidance and support to club members who need help with their technical skills or professional development.
g) **Design Lead:** The Design Lead is responsible for creating visual designs for the club's website, social media accounts, and other promotional materials.
h) **Outreach Lead:** The Outreach Lead is responsible for reaching out to potential partners, sponsors, and collaborators to expand the club's network and resources.

Required

1. Choose an imaginary **development team** of a maximum of 10 members and specify their roles. (Random names can be used).
2. Briefly explain to your teammates the kind of **web architecture** you are going to use and why.
3. Draw the following diagrams as part of your **Analysis and Design** phase of the Web Application.
    i. Architecture Diagram
    ii. Flow Chart
    iii. ER Diagram
    iv. Use Case Diagram
4. Using **ReactJS develop** a Web Application based on requirements provided in the above context.

NOTE:

a) Extra marks will be awarded to candidates who show an effort of creativity and innovation.
b) Poor coding patterns and lack of creativity will lead to a loss of marks.
c) The code should be well commented on for easy understanding.
d) Neat and easy-to-understand diagrams should be drawn.

THE END