Team Project: A Flask-Based Web App with Visual Analytics

This project provides an excellent opportunity to challenge yourself with hands-on experiences. You will go through the different phases of the Software Development Life Cycle (SDLC) for developing a web application. ***Creativity*** *and* ***novelty*** are strongly encouraged in this project. By the time you finish this project, you will be able to showcase your technical skills and experiences to recruiters, your family, and your friends!

(1) Objectives and Requirements

The objective of this project is to analyze, design, and implement a database-driven web application. **Every team member is required to implement the following**:

- Adopts an MVC (Model-View-Controller) architecture;
- Uses all of the following technologies:
  - Bootstrap (HTML, CSS, Javascript, jQuery)
  - Flask
  - MySql
  - chart.js
- Designs at least 2 web pages (it is not allowed to share web pages with other team members);
- Implements all of the following database operations:
  - adding new records;
  - modifying existing records;
  - deleting existing records;
  - searching for records; AND
  - conducting visual analyses using database records and chart.js.

(2) Project Topic and Implementation Requirement

Your team is hired to develop a web application for managing a tennis ladder in a local tennis club. The ladder organizes year-round weekly tennis matches for the club members. The application is used to keep track of match scores and player rankings. You can develop the application for a real tennis club or an imaginary one. You can make other assumptions in the given context. Please be creative.

Four modules along with their use cases have been identified through a requirement analysis process. Each team member should pick one module and is responsible for the design, implementation, and testing of the chosen module (5 use cases) as your final project deliverable. The functional modules are assigned to the students based on the first letter of the last name.

1. Player Profile Management
   UC1: Create a new club member account (email, password, name, phone number, age, gender, UTR rating)
   UC2: Log in as a club member (store to the database a login activity record after each login)
   UC3: Modify a club member's profile
   UC4: Delete a club member's profile
   UC5: Display a bar chart to visualize the number of club members by gender, age group, or UTR rating level

2. Challenge Management:

UC1: Log in as a club member
UC2: Initiate a new challenge (any player can challenge another player by creating a new challenge request)
UC3: Accept a challenge request (a player can accept a challenge posted by other players)
UC4: Cancel/Delete a challenge request
UC5: Display a bar chart to visualize the number of wins/losses in all the challenges this member has played.

3. Match Management
   UC1: Create a new match record for a challenge match that has concluded (The match format is the best 2 out of 3 sets. In the first two sets, a player needs to win 6 games, by two, to win the set. If the players are tied at 6:6, a 7-point tiebreaker game will be played to decide the set winner. If the players are tied at 1:1 in the first two sets, they will play a 10-point tiebreak set in the third set. Only the set scores need to be recorded, e.g., 7:6, 1:6, 7:10)
   UC2: Modify an existing match record
   UC3: Delete an existing match record
   UC4: Search for match records by player
   UC5: Display a chart to visualize the number of wins/losses for a player

4. Membership Management (Only for a 4-person team)
   UC1: Create a membership invoice record (An annual membership fee of $100 is billed when a new member joins the club)
   UC2: Modify a membership invoice record (e.g., extend the due date)
   UC3: Delete a membership invoice record
   UC4: Search for a membership invoice record by member
   UC5: Display a chart to visualize the number of members who has/has not paid the membership dues in a given year

The system database contains the following tables (important attributes are provided, but additional attributes can be added. Underlined attributes are the primary keys):

1. **Member**(MEID, FirstName, LastName, Email, MPassword, Phone, Age, Gender, UTR, DateOfCreation)
2. **Match**(MAID, CID, DateOfMatch, MEID1Set1Score, MEID2Set1Score, MEID1Set2Score, MEID2Set2Score, MEID1Set3Score, MEID2Set3Score, WinnerMEID, LoserMEID)
3. **Challenge**(CID, ChallengerMEID, ChallengedMEID, DateOfChallenge, Notes)
4. **Membership**(MSID, MEID, StartDate, EndDate, InvoiceDate, DueDate, Amount, PaidDate)

You should implement the database in your MySql database as defined above. You can add additional attributes to the database tables. But you are not expected to remove any attribute for simplification. Do not use special characters (e.g., space, underscore) and MySql reserved words (e.g., user, password) when naming your database tables and attributes. Manually create at least 3 records in each database table.

(3) Team Presentation
Due: Friday, June 16, 2023

Each team has 10 minutes to demonstrate your web application. Students will vote for

the best project teams at the conclusion of the presentation session.

Attendance is required. You MUST stay during the entire presentation session to receive your project presentation grade. If you are absent from the slam session without documented excuses or the instructor's permission, you will receive a zero grade in your slam grade.

(4) Final Deliverable
Due: Monday, June 19, 2023

1) An implemented web application compressed in a .zip file;
2) A completed checksheet (which can be found on our Canvas site); and
3) A video demonstration of your system functions for grading purposes. You must demonstrate every single use case and show me evidence that the function actually works. If it involves database operations, please show me the database content before and after the database operation.

Late submissions will NOT be accepted.