THE RUBRIC

Website Name: TBD

Group: 3

Sprint 3

Team Members: Hoan Ngo, Jean-Charles Adou, Kenny Nguyen, Mahdyah Hassanyar, Tanuj Patel

Date: April 6th 2023

TASK 1:

Planning Scheduling and Peer Evaluation

Assignee Name	Email	Task	Duration (hours)	Dependency	Due date	Evaluation
Hoan Ngo	hngo15@student.gsu.edu	Behavioral modeling(Judge)	4 Hours	None	5/6/2023	
Jean-Charles Adou	Jadou1@student.gsu.edu	TASK 6 : Communication and Collaboration Github	4 Hours	None	5/6/2023	
Kenny Nguyen	Knguyen164@student.gsu.edu	Class Diagrams Behavioral modeling(Moderator)	4 Hours	None	5/6/2023	
Mahdyah Hassanyar	msayedtahershah1@student.gsu.edu	Architecture modeling Implementation Coordination	4 Hours	None	5/6/2023	
Tanuj Patel	Tpatel50@student.gsu.edu	TASK5: Testing	4 Hours	None	5/6/2023	

TASK 2: Revise and Refine your System (Improved Problem Statement)

We have not received any feedback yet, the team is proceeding with the original plan.

5. Problem Statement:

- What is your product, on a high level?

A website for judges to record grades of competitors and grant awards at a poster presentation.

- Whom is it for?

Students and facilities have a great opportunity to win an award for their posters at certain presentations. Event organizers need a reliable and standardized judging form for this poster presentation. Individuals should be graded based on their submission (Ph.D., MS, BS, poster, demonstration). Based on the highest score, an award will be granted to the winner. Our clients would be organizers and judges of these events. However, our audience is everyone.

- What problem does it solve?

These titles and awards given at these events give a great opportunity for a lot of future career endeavors. However, the process should be treated fairly and securely. Many contestants are at different levels of academics, so the judging process should be separated from each level. Judges should be able to submit a scoring sheet based and only the user/organizer can see the grades. The award would be given from them based on these results.

- What alternatives are available?

If the event is small enough, some alternatives are using pen and paper. However, in larger events, this method would be obsolete as it will become time-consuming and accident-prone. Some apps allow judges to fill in rubrics and the app will calculate scores for the competitors, but they do not present any features that are relevant to individual criteria.

- Why is this project compelling and worth developing?

There are not a lot of competitors in the market. So, developing this project and website would allow more schools and universities the opportunity to host these events and give out scholarly awards. Individuals have the opportunity to do the event and can be graded for their efforts. The system would be secure, yet functional for incorporating this into an event.

- Describe the top-level objectives, differentiators, target customers, and scope of your product.

The top-level objective is to build a website for judges to record the scores of competitors at a poster presentation. The differentiators from other products are to have the judges join in by entering a random generated code so that judges can make their votes. The

target customers will be mostly schools and universities. The scope of our product is to keep it simple and maintain the functionality within the requirements.

- What are the competitors and what is novel in your approach?
There is an app called Rubric Scorer, where judges can fill in the rubric and the app calculates the total. We plan to create a website so that way anyone with any type of device would be able to use it. This app will allow the user to create a rubric and judges by using this app can vote and the admin can see the grades each candidate received in order of high to low.

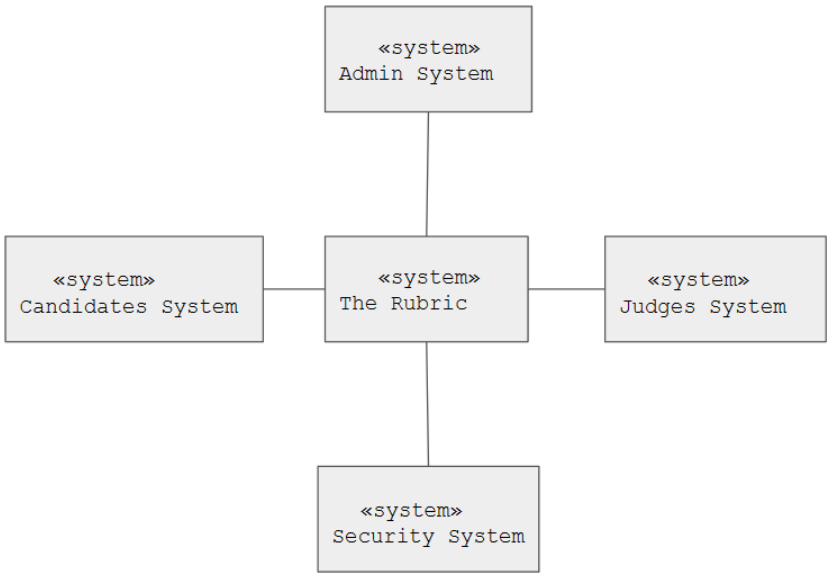
- Make it clear that the system can be built, making good use of the available resources and technology.
We have all the necessary tools and resources. If we need assistance, we will use any resources provided by the internet to hone our abilities. We are planning to use node.js to create the backend and we will be using React.js and Bootstrap in the front.

- What is interesting about this project from a technical point of view?
This project will be a little like Kahoot, users will be given an access number to vote. We do know that most colleges and universities do not have a system where they can perform their rubric process. It is still done using pen and paper, which is very time-consuming also because of many candidates and judges sometimes miscalculations happen. I talked to someone well experienced being a judge and he said they have been looking for this kind of software that could do such a task. Therefore, managing the database for the system will be rewarding for our future careers. It gives the experience to explore other back-end processes and develop other systems. The skills gained during this process will be an important asset for us.

- Make it clear that the system can be built, making good use of the available resources and technology.
We are planning to use node.js to create the backend and we will be using React.js and Bootstrap in the frontend. We have all the necessary tools and resources. If we need assistance, we will use any resources provided from the internet to hone our abilities.

- What is interesting about this project from a technical point of view?
This project will be a little like Kahoot, users are given an access number to vote. We do know that most colleges and universities do not have a system where they can perform their rubric process. It is still done using pen and paper, which is very time consuming also because of the large number of candidates and judges. Sometimes miscalculations happen. I talked to someone who is well experienced being a judge and he said they have been looking for a software that could do such a task. Therefore, managing the database

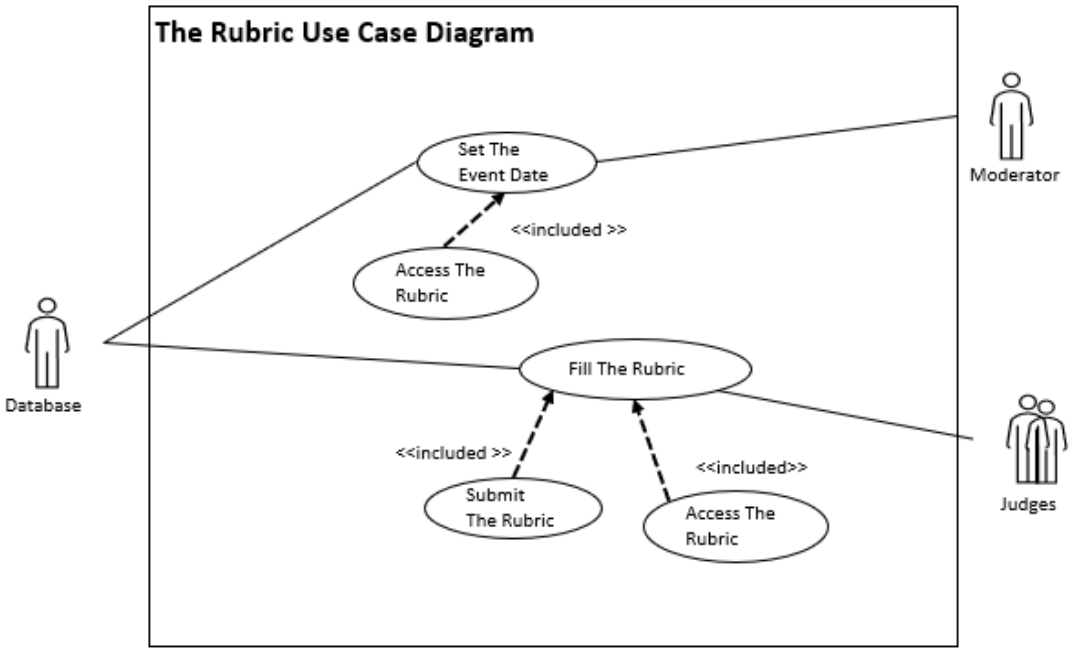
for the system will be rewarding for our future careers. It gives the experiences to explore other back-end processes and develop other systems. The skills gained during this process will be an important asset for us.



Context model:

In this context model The Rubric is the main system, and the admin system, candidates system, judges system are the systems in the main system. The admin system manages the composition of the desired rubric forms and access key to the Judges. The judges system manages the forms submitted by the judges. The candidates system calculates the final grades and displays them to the administrator. The security system will have the responsibility of securing the system, where only the administrator can login and only the judges who are provided with access key can vote or submit the rubric form. Also, only the administrator would be able to see the final result.

1. **Diagram**



2. Use Cases

Use Case no: 1

Use Case Name: Start New Event

Actors: Moderator

Description:

- After the user successfully logs in, the home page will appear.
- The user selects the “new event” button.
- The new page will prompt for a new name for the event.
- The user inputs the name and selects the “begin planning” button.
- The event has been created.

Exceptional Path: If the user does not give a valid name in step three the webpage will prompt the user for a new name.

Alternate Path: There is only one path in this case, unless an error occurs.

Pre-condition: The account has to be successfully logged in.

Post-condition:

Use Case no:2

Name of the use Case: Set the time of an event.

Description:

- The user selects the “Edit” option for the event.
- The user selects the “Set Event time” option for the event.
- The webpage prompts the user to edit/enter a new time for the event.
- The user selects “Save” option.
- The time has been set for event.

Exceptional path: If the user tries to place a time that isn’t reachable, the webpage will prompt the user it’s not reachable.

Alternate Path: In this case there is only one path.

Pre-Condition: The user has successfully login and is on the home page; The user also has created an event without the time set.

Use Case no: 3

Use Case Name: Change Time of Event

Actors: Moderator

Description:

- The user selects the “Edit” option for the event.
- The user selects the “Edit Event Time”option.
- The webpage prompts the user to edit/enter a new time for the event.
- Once new time is input, the user will select the “save new time”
- The new time has been set for the event.

Exceptional Path: If the user does not enter a valid time an error prompt will appear.

Alternate Path: There is only one path for this case.

Pre-condition: The user must have successfully login and is on the home page; The user also has create an event with the time set.

Use Case no: 4

Use Case Name: Manually close event

Actors: Moderator

Description:

- The user selects “Edit” option on the event.
- The use select “Close Event.” option.
- Event will be closed immediately.

Exceptional Path: If the user does not have event created, the webpage will prompt the user to create one.

Alternate Path: The event time has expired.

Pre-condition: The user must have successfully login and is on the home page; The user also has create an event with the time set.

Use Case no: 5

Use Case Name: Accessing results from the results page

Actors: Moderator

Description:

- The user select “Edit” option on the event..
- The user selects the “View Results” button.
- The webpage will display the results.

Exceptional Path: If the event timer hasn’t expire than the user must close the event.

Alternate Path: There is no alternate path in this case.

Pre-condition: The user must have successfully login and is on the home page; The user also has create an event. The event timer has expire/ close manually.

Use Case no: 6

Name of the use Case: Delete an event.

Description:

- The user selects the “Edit” option for the event.
- The user selects the “Delete this event” option.
- The Webpage will ask if the user is sure.
- The user selects the “Confirm” option.

The database will clear up space and remove any data from the event.

Exceptional path: If the user does have any event, they will be recommended to start one.

Alternate Path: There is only one path in this case unless the user changes their mind.

Pre-Condition: The user must have successfully login and is on the home page; The user also has create an event with the time set.

Use Case no: 7

Use Case Name: Acquiring Judge Access Code

Actors: Administration, Judge

Description:

- The user selects “Edit” option on the event.
- The user selects “Generate Access Code” option.
- The webpage will generate random code.

Exceptional Path: If the user has not created an event, the user will be prompt to make one.

Alternate Path: There is no alternate path in this case.

Pre-condition: The user must have successfully login and is on the home page; The user also has create an event with the time set.

Use Case no: 8

Name of the use Case: Using access code to form a page.

Description:

- After being given the access code, the judge selects the “Join Event as Judge” option.
- The website asks for the access code.
- The judge inputs the access code.
- If the access code matches the event, the judge will be directed to the form page.
- The judge selects the correct form and then selects the “Confirm” option.

Exceptional path: If the user does not enter anything but the correct code, the webpage will prompt the user to try again.

Alternate Path: In this case, there is only one path, unless an error occurs.

Pre-Condition: The access code has been generated and the judge is on the Welcome page.

Use Case no: 9

Name of the use Case: Finalizing/submitting their form.

Description:

- The user will fill in the appropriate rubric for candidates.
- Once the scoring is done, the user selects the “Submit” option.

Exceptional path: If the form does not have all the necessary grades, the webpage will give out the following message, “Some areas are left ungraded.”

Alternate Path: In this case, the user can also add additional comments, but it wouldn’t affect the final scoring.

Pre-Condition: The user has used the access code and is ready to submit the scoring sheet.

Use Case no: 10

Name of the use Case: Creating a new rubric

Description:

- The user selects the “Rubric” option.
- The user selects the “New Rubric” option.
- A new rubric has been created with the name “new_rubric”.
- Exceptional path: If the user created a rubric with the same name, then the website will notify the user.

Alternate Path: In this case there is only one path.

Pre-Condition: The user has successfully login and is on the home page.

Use Case no: 11

Name of the use Case: Editing the rubric.

Description:

- The user selects the “Rubric” option.
- The user selects the “Edit Rubric” option.
- The user will enter grades accordingly.
- It will save periodically to save progress. Once the user has finished, they will select the “Save Changes” option.
- The format of the rubric will be updated/added to the under that rubric name.

Exceptional path: If the user does not have a rubric to edit, the website will prompt to make new_rubric.

Alternate Path: In this case, there is no alternative path.

Pre-Condition: The user has successfully login and is on the home page; The user also has created a rubric.

Use Case no: 12

Use Case Name: Change/Add Question to rubric

Actors: Moderator

Description:

- The user selects the “Rubric” option.
- The user selects the “Edit Rubric” option.
- Th users will edit existing questions or input new questions.
- User will Select “Save Changes” button.
- New/Edited Questions will be updated/added to the under that rubric name.

Exceptional Path: If the user does not enter a valid question format an error prompt will appear.

Alternate Path: There is only one path for this case.

Pre-condition: The user must have successfully login and is on the home page; The user also has create a rubric.

TASK 3: System Requirements

Requirement number: 1

Use Case number: 1

Name: New Event (button)

Introduction: New event button allows moderators to set up a new event with basic information.

Inputs: Name of event, start time, end time,

Requirements Description:

1. End time must be after start time.

Outputs: All data should be saved in the database after all the correct information has been submitted. Show a pop-up saying a new event created successfully.

Requirement number: 2

Use Case number: 2

Name: Set event timer (button)

Introduction: Time editing button allows moderators to start the time of the event (start time, end time)

Inputs: new start time, new end time

Requirements Description:

1. Format mm/dd/yyyy, hh:mm:ss

Outputs: New time should be saved in the database after all the correct information has been submitted. Show a pop-up saying time changes successfully.

Requirement number: 3

Use Case number: 3

Name: Event Time Editing (button)

Introduction: Time editing button allows moderators to change the time of the event (start time, end time)

Inputs: new start time, new end time

Requirements Description:

1. Format mm/dd/yyyy, hh:mm:ss

Outputs: New time should be saved in the database after all the correct information has been submitted. Show a pop-up saying time changes successfully.

Requirement number: 4

Use Case number: 4

Name: close the event

Introduction: Moderators can manually delete an event in case they create an event by accident or don’t want it anymore.

Inputs: event id, delete command.

Requirements Description:

1. Events must exist in the database before deleting.

Outputs: event should be removed from database

Requirement number: 5

Use Case number: 5

Name: result page

Introduction: moderators can see the result of an event on result page

Inputs: event id

Requirements Description:

1. Event must be existing in database

Outputs: event's result

Requirement number: 6

Use Case number: 6

Name: Delete an event

Introduction: Delete event allows moderators to delete an existing event

Inputs: Information of the existing event

Requirements Description:

1. The event needs to exist in the database before deleting

Outputs: Event deleted from database. A pop-up shows an event deleted successfully.

Requirement number: 7

Use Case number: 7

Name: The user(admin) acquiring the judge access code

Introduction: Moderators create a unique access code for a specific rubric

Inputs: access code

Requirements Description:

1. An access code that is not existing in the database yet

Outputs: access code linking to a rubric

Requirement number: 8

Use Case number: 8

Name: The user(judge) using the access code to form page

Introduction: Judges can get to a form page by entering a unique access code linked to that form

Inputs: access code

Requirements Description:

1. Access code must be existing in database

Outputs: form page

Requirement number: 9

Use Case number: 9

Name: The user(judge) finalizing/submitting their form

Introduction: Judges submit their form after filling out the form

Inputs: Judges' answer

Requirements Description:

Outputs: Answers are saved in database

Requirement number: 10

Use Case number: 10

Name: Creating a new rubric

Introduction: Moderators can create new rubrics for their event

Inputs: Category, number of questions, questions' contents

Requirements Description:

1. Type of rubric

2. Grading scale

Outputs: All data should be saved in the database after all the correct information has been submitted. Show a pop-up saying new rubric created successfully.

Requirement number: 11

Use Case number: 11

Name: Formatting tools for question

Introduction: Moderators use different grading tool s and scales.

Inputs: Grade values

Requirements Description:

1. The user has created a rubric

Outputs: All data should be saved in the database after all the correct information has been submitted. Show a pop-up saying new rubric created successfully.

Requirement number: 12

Use Case number: 12

Name: Question Input

Introduction: Question editing tool to allow insert or remove question.

Inputs: New questions (add), edited questions (edit)

Requirements Description:

1. Moderator must enter new questions.

Outputs: New questions are saved in the database. Deleted questions are removed from the database.

TASK 5: Database Specification

Judge: Judge_id DataType INT(PK), Judge_firstName DataType VARCHAR, Judge_lastName DataType VARCHAR, Judge_email DataType VARCHAR

Contestant: contestant_id DataType INT(PK), contestant_firstName DataType VARCHAR, contestant_lastName DataType VARCHAR, contestant_email DataType VARCHAR

Moderator: moderatort_id DataType INT(PK), moderatort_email DataType VARCHAR ,moderatort_user_name DataType VARCHAR, moderatort_password DataType VARCHAR

Moderator: moderatort_id DataType INT (PK), moderatort_email DataType VARCHAR ,moderatort_user_name DataType VARCHAR, moderatort_password DataType VARCHAR

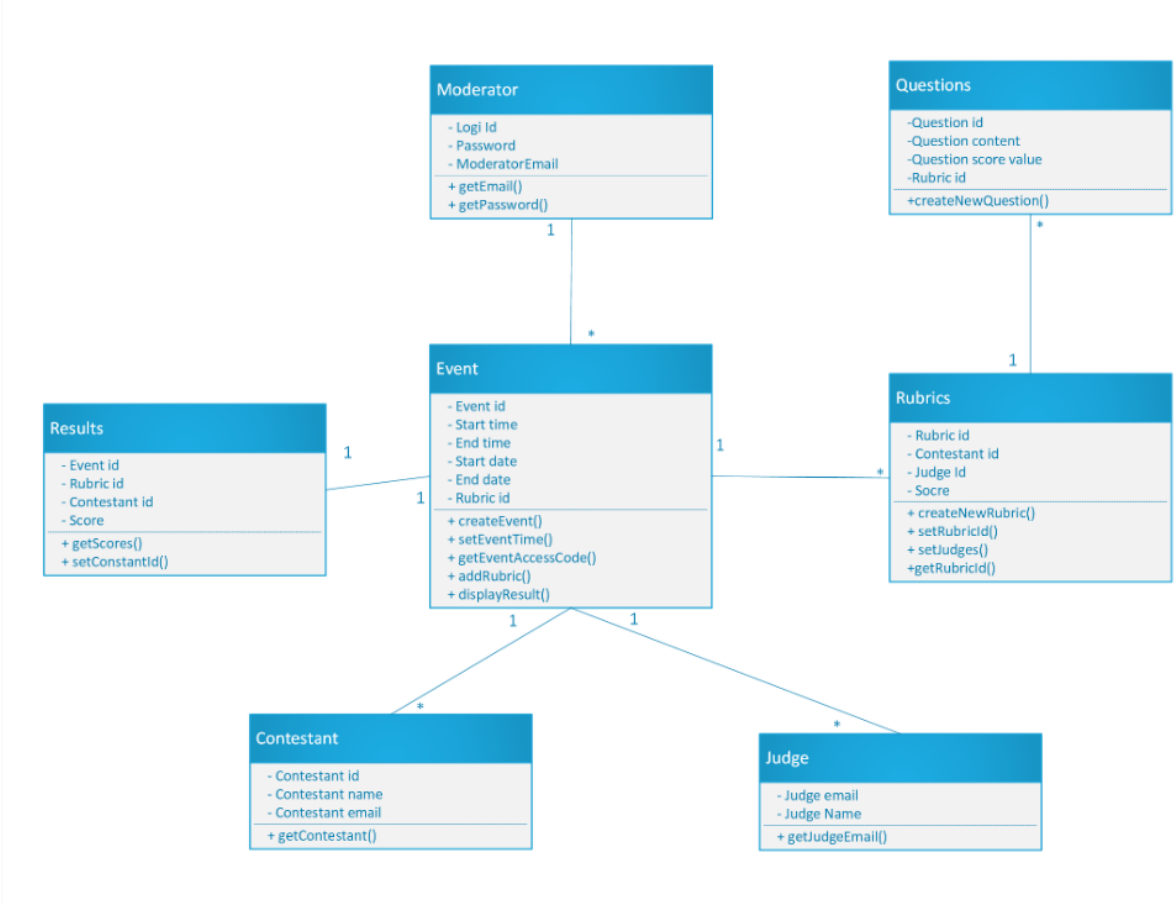
Event: Event_id DataType INT (PK),Start_Time DataType VARCHAR,End_Time DataType VARCHAR ,Start_Date DataType VARCHAR, End_Date DataType VARCHAR, Rubric_id (FK) DataType INT

Questions: Question_id DataType INT (PK), Question DataType VARCHAR,option_one DataType VARCHAR, ,option_one DataType VARCHAR,option_two DataType VARCHAR,option_three DataType VARCHAR,option_four DataType VARCHAR,option_five DataType VARCHAR, Rubric_id DataType INT (FK).

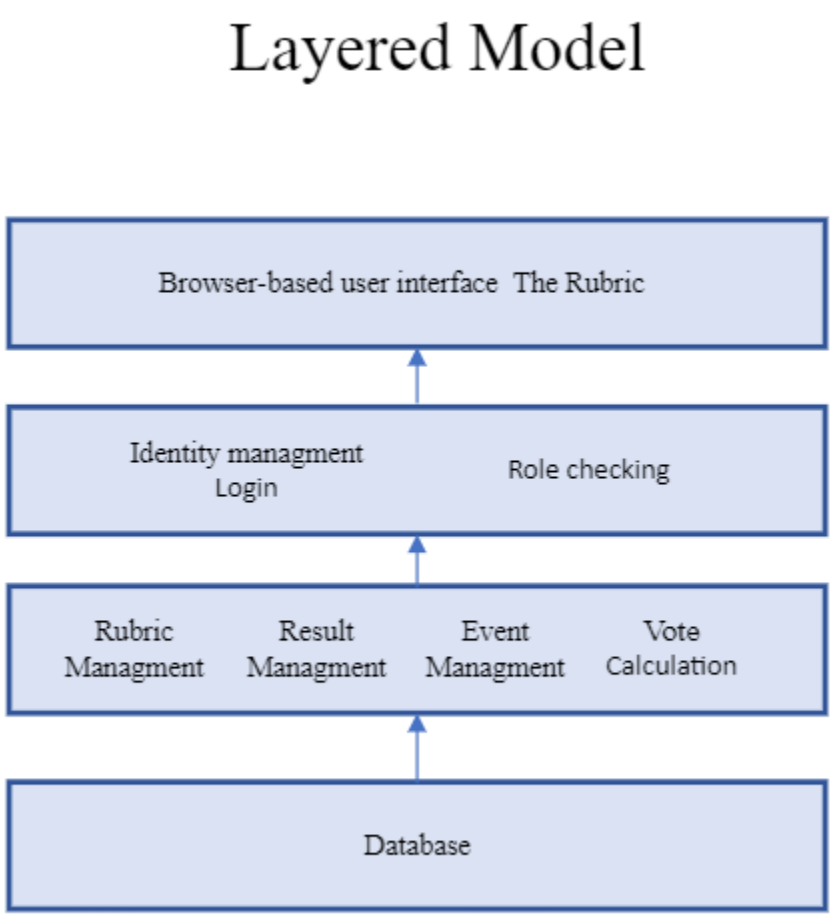
Rubric: Rubric_id DataType INT (PK), rubric_title DataType VARCHAR, question_number DataType VARCHAR, moderator_id DataType INT (FK), question_id DataType INT (FK).

Results:result_id DataType INT (PK), Event_id DataType INT (FK), Rubric_id DataType INT (FK), Contestant_id DataType INT (FK), juge_id DataType INT (FK), qestion1_score DataType INT, qestion2_score DataType INT, qestion3_score DataType INT,qestion4_score DataType INT, qestion5_score DataType INT.

1. Class Diagrams



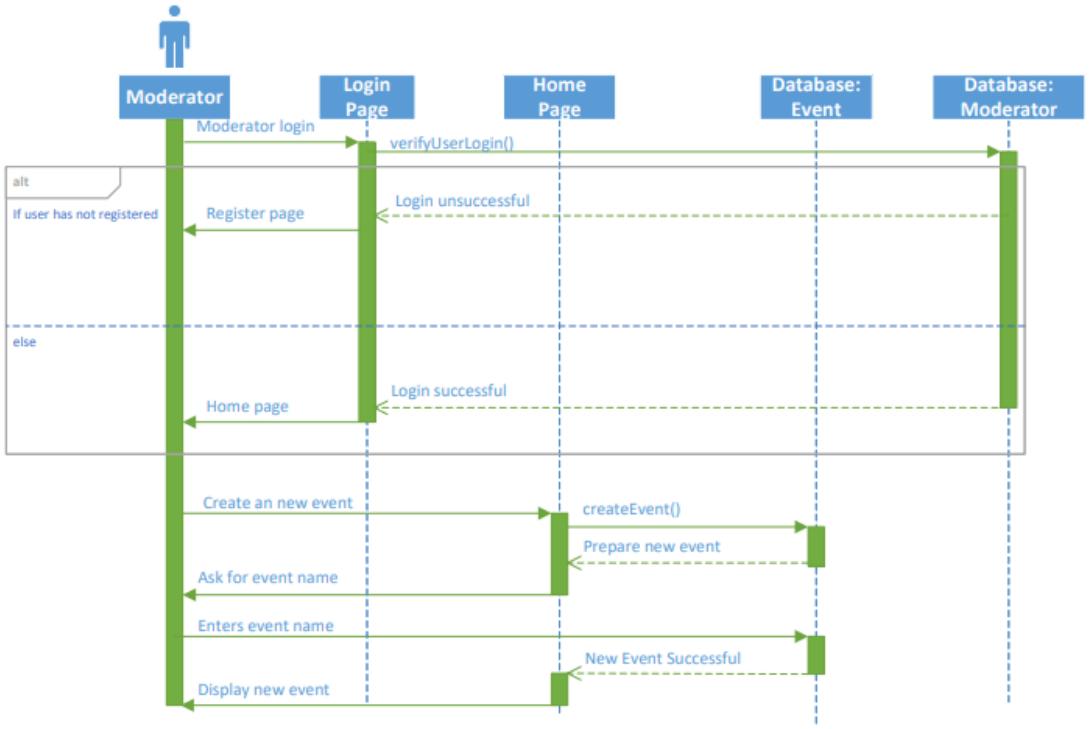
2. Architecture modeling:

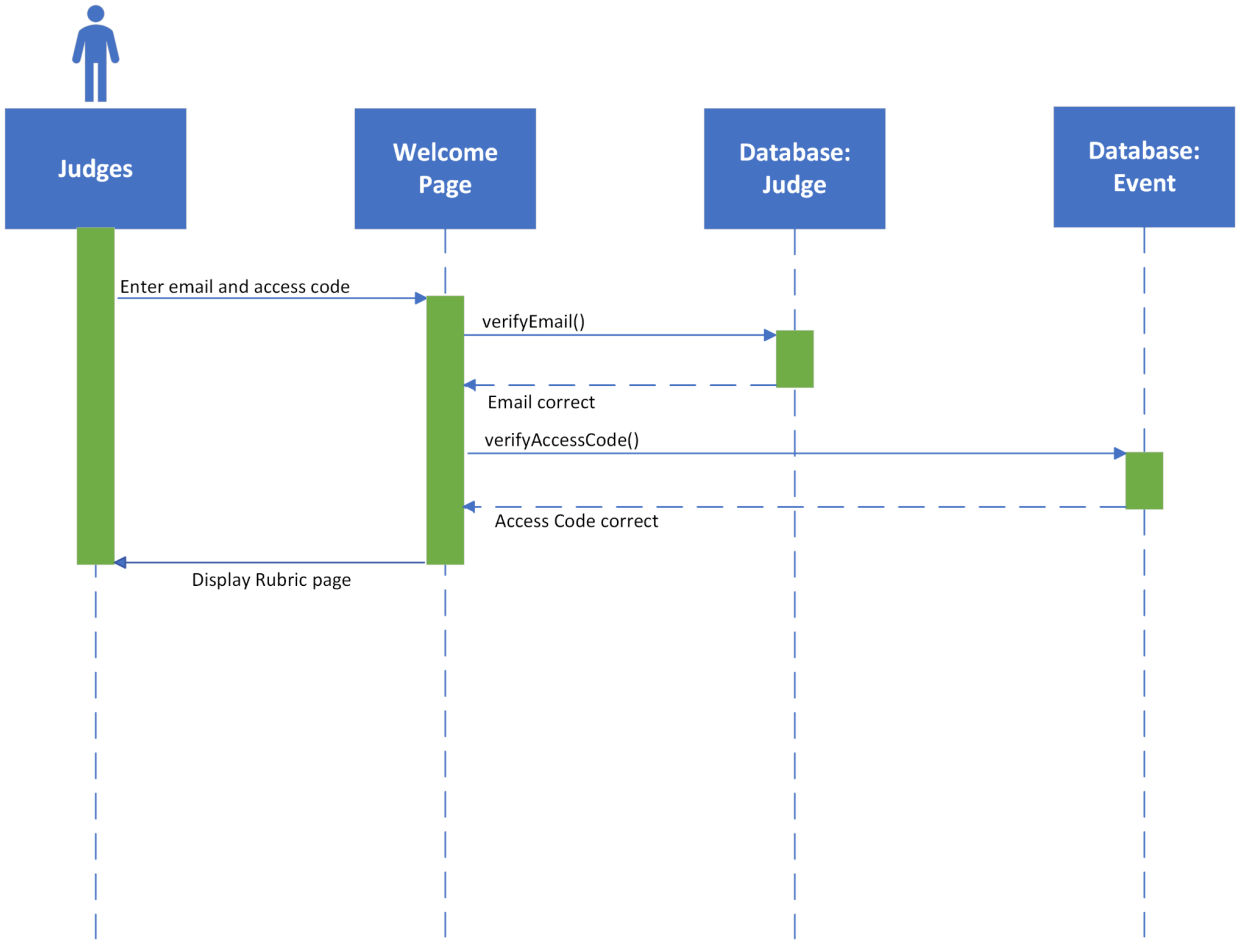


We have chosen the Layered model for our project as it allows replacement of the entire layer as long as the interface is maintained. Also, when changes occur only the adjacent layer is impacted. Most importantly, the redundant features in each layer can enhance security and

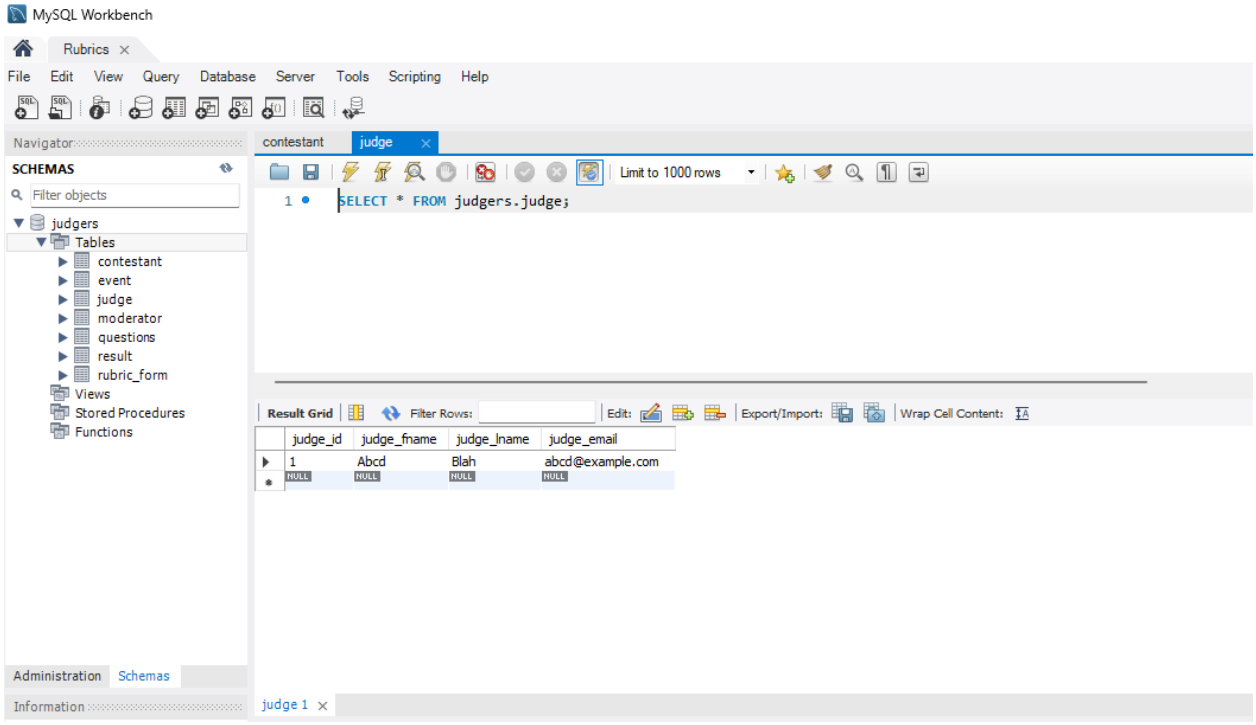
dependability. Also, this project is a team project and the development is spread across several teams with each team responsible for a layer of functionality.

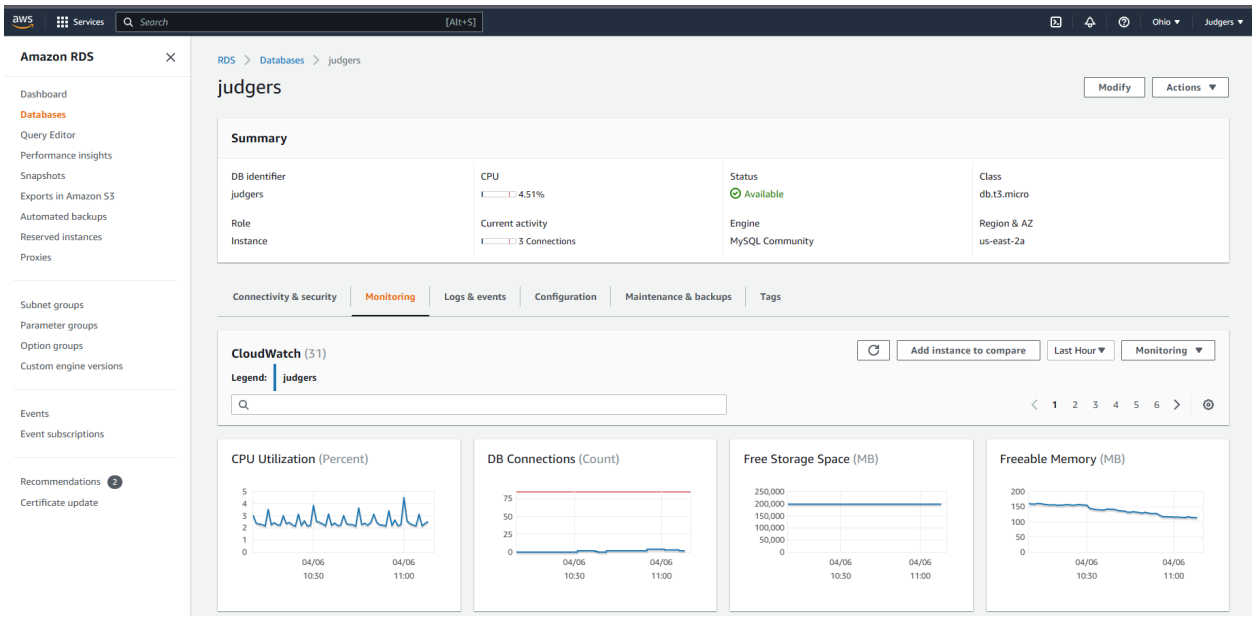
3. Behavioral modeling

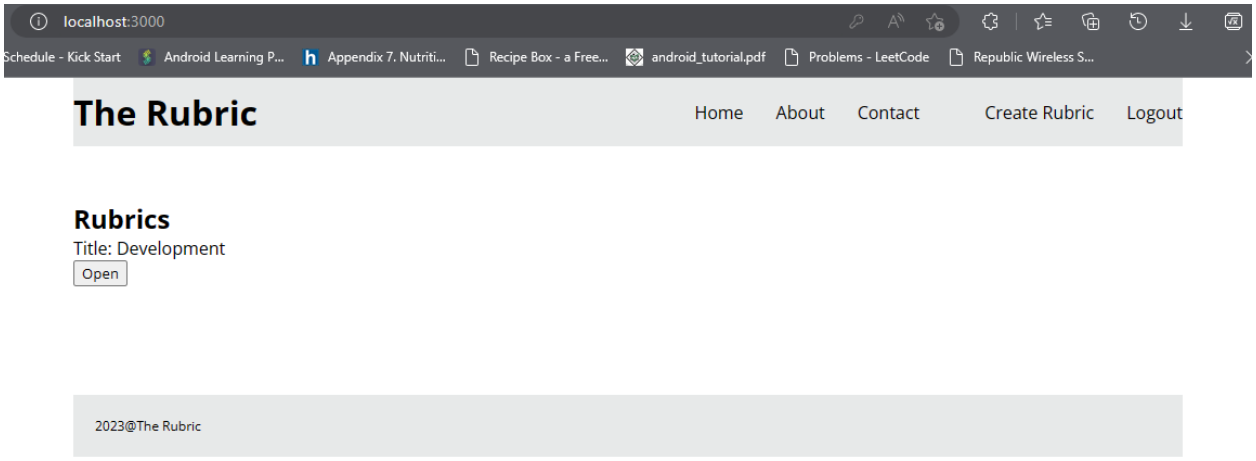




Implementations:


MySQL Workbench
Rubrics
File Edit View Query Database Server Tools Scripting Help
Navigator
SCHEMAS
Filter objects
judgers
Tables
contestant
event
judge
moderator
questions
result
rubric_form
Views
Stored Procedures
Functions
Administration Schemas
Information
contestant
judge
Limit to 1000 rows
1 SELECT * FROM judgers.judge;
Result Grid
Filter Rows:
Edit:
Export/Import:
Wrap Cell Content:
judge_id judge_fname judge_lname judge_email
1 Abcd Blah abcd@example.com
NULL NULL NULL NULL
judge 1


aws
Services
Search
[Alt+S]
Ohio
Judgers
Amazon RDS
Dashboard
Databases
Query Editor
Performance insights
Snapshots
Exports in Amazon S3
Automated backups
Reserved instances
Proxies
Subnet groups
Parameter groups
Option groups
Custom engine versions
Events
Event subscriptions
Recommendations 2
Certificate update
RDS > Databases > judgers
judgers
Modify
Actions
Summary
DB identifier
judgers
CPU
4.51%
Status
Available
Class
db.t3.micro
Role
Instance
Current activity
3 Connections
Engine
MySQL Community
Region & AZ
us-east-2a
Connectivity & security
Monitoring
Logs & events
Configuration
Maintenance & backups
Tags
CloudWatch (31)
Legend: judgers
Add instance to compare
Last Hour
Monitoring
CPU Utilization (Percent)
DB Connections (Count)
Free Storage Space (MB)
Freeable Memory (MB)
04/06 10:30
04/06 11:00



Test Cases:

Test Case ID: 1

Description: Register Account

Test Inputs: Email and Password given that are not already in Database

Expected Results: Success message and redirect to login screen

Dependencies: None

Initialization: Database has checked existing Email/Password

Test Steps:

1. Enter a valid Email/Password

2. Receive Success message and proceed to login page

Test Case ID: 1.5

Description: Register Account With existing email

Test Inputs: Enter Email and Password that is in database

Expected Results: Error message and re-prompt to enter email/password

Dependencies: None

Initialization: Database has checked existing Email/Password

Test Steps:

1. Enter an already used Email/Password

2. Receive Error message and re-prompt to enter email/password

Test Case ID: 2

Description: User signs in

Test Inputs: User Email and password

Expected Results: Successful login into website

Dependencies: None

Initialization: Email and password are already in database

Test Steps:

1. Enter email/password

2. Submit

Test Case ID: 2.5

Description: User signs in with incorrect Email/Password combo

Test Inputs: Incorrect User Email and password

Expected Results: Error message and re-prompt to enter Email/Password

Dependencies: None

Initialization: Email and password are not in database

Test Steps:

1. Enter incorrect email/password

2. Submit

Test Case ID: 3

Description: Moderator Creates New Event

Test Inputs: Moderator clicks the create new event button

Expected Results: Takes moderator to the next page with the event name, start time, end time, start date and end date

Dependencies: None

Initialization: Moderator is signed into website

Test Steps:

1. Click Create event

2. Enter required information and submit

Test Case ID: 4

Description: Judge Accesses rubric

Test Inputs: Judge inputs access code into website to get rubric

Expected Results: Judge gets taken to the next page with the rubric questions

Dependencies: None

Initialization: Judge must be signed in to website

Test Steps:

1. Click the access rubric button

2. Enter correct access code

Test Case ID: 4.5

Description: Judge Enters wrong access code

Test Inputs: Judge inputs an incorrect access code into website

Expected Results: Judge gets error message and can retry a new code

Dependencies: None

Initialization: Judge must be signed in to website

Test Steps:

1. Click the access rubric button

2. Enter incorrect access code

Test Case ID: 5

Description: Judge submits rubric

Test Inputs: Judge enters numerical scores for each question and then submits the rubric

Expected Results: Success screen and rubric data gets put into database

Dependencies: None

Initialization: Judge has accessed the rubric with access code

Test Steps:

1. Enter numerical data for each question

2. Submit

Test Case ID: 5.5

Description: Judge submits rubric with incorrect data type/ too high number

Test Inputs: Judge enters words and/or numbers that are outside the bounds

Expected Results: Error message and sent back to rubric screen

Dependencies: None

Initialization: Judge has accessed the rubric with access code

Test Steps:

1. Enter wrong data for questions

2. Submit

Test Case ID: 6

Description: Moderator Edits Event

Test Inputs: Moderator clicks the edit existing event button

Expected Results: Takes moderator to an editing page to change the event name, start time, end time, start date and end date

Dependencies: None

Initialization: Moderator must be logged in to website

Test Steps:

1. Select event

2.Click edit event button

3. Enter new event name, start time, end time, start date and end date

Test Case ID: 6.5

Description: Moderator Edits Event With wrong data types

Test Inputs: Moderator clicks the edit existing event button and enters incorrect data

Expected Results: Error Message and sent back to edit event screen

Dependencies: None

Initialization: Moderator must be logged in to website

Test Steps:

1. Select event

2.Click edit event button3. Enter incorrect event name, start time, end time, start date and end date

Communication and Collaboration Github: