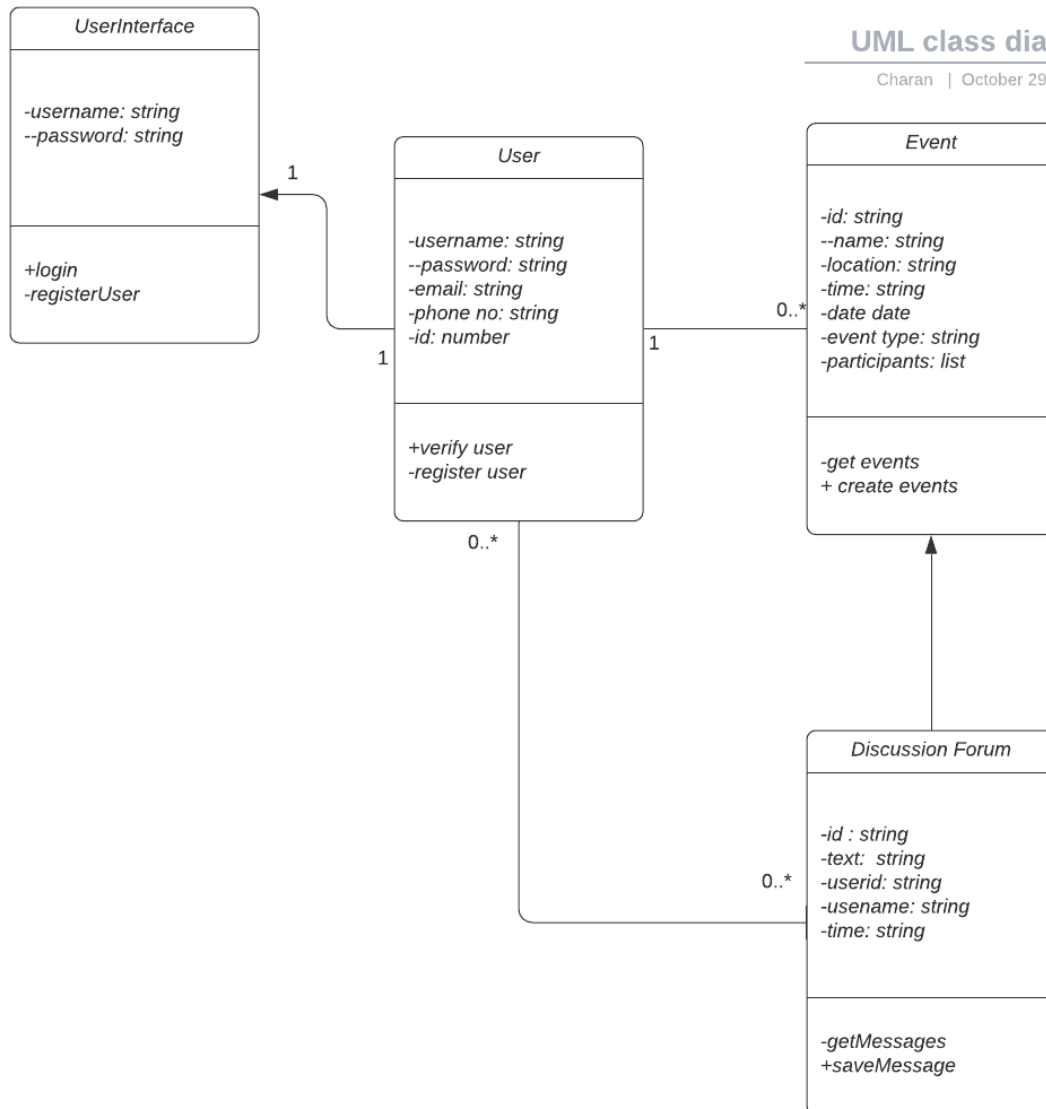


a) Requirements:

- Develop the architecture and working code of backend for sports management module.
- Make sure that the backend application manages objects efficiently , eg ('database connection' object is created once and managed throughout the application when it is running.)
- As mentioned in the 'Deliverable-2', create API endpoints to manage the following
 1. User registration
 2. User login verification
 3. Get all event data (UPCOMING,ONGOING)
 4. Post a message in discussion forum
 5. Retrieve all messages posted in the discussion forum
 6. **Login Page:** This will be the initial screen when we open the application. In this screen if the user already exists then they give their details and authenticate themselves or if they are a new customer then there will be a register link, on clicking it will navigate to the register screen.
 7. **User Registration Page:** In this page the new customer will fill in the details to register himself. The page contains all the necessary fields that are needed to register the new customer.
 8. **Dashboard Page:** This page will open after the user successfully authenticates themselves. This will be the main page containing all the information about the ongoing events, incoming events and discussion forum. So, it consists of 3 sub screens.
 9. **Ongoing Event Page:** This will be the subpage of the dashboard page. It will display all the information related to the ongoing events by rendering the data received from the backend API.
 10. **Incoming Event Page:** This will be the subpage of the dashboard page. It will display all the information related to the incoming events by rendering the data received from the backend API.
 11. **Discussion Forum Page:** This will be the subpage of the dashboard page. It will display all the information related to the discussion forum by rendering the data received from the backend API.
- Super User Page:** This is the admin user page. Admin will use admin credentials to log in where he will be redirected to the Create events page where he will be able to create/modify events.

b) UML diagram

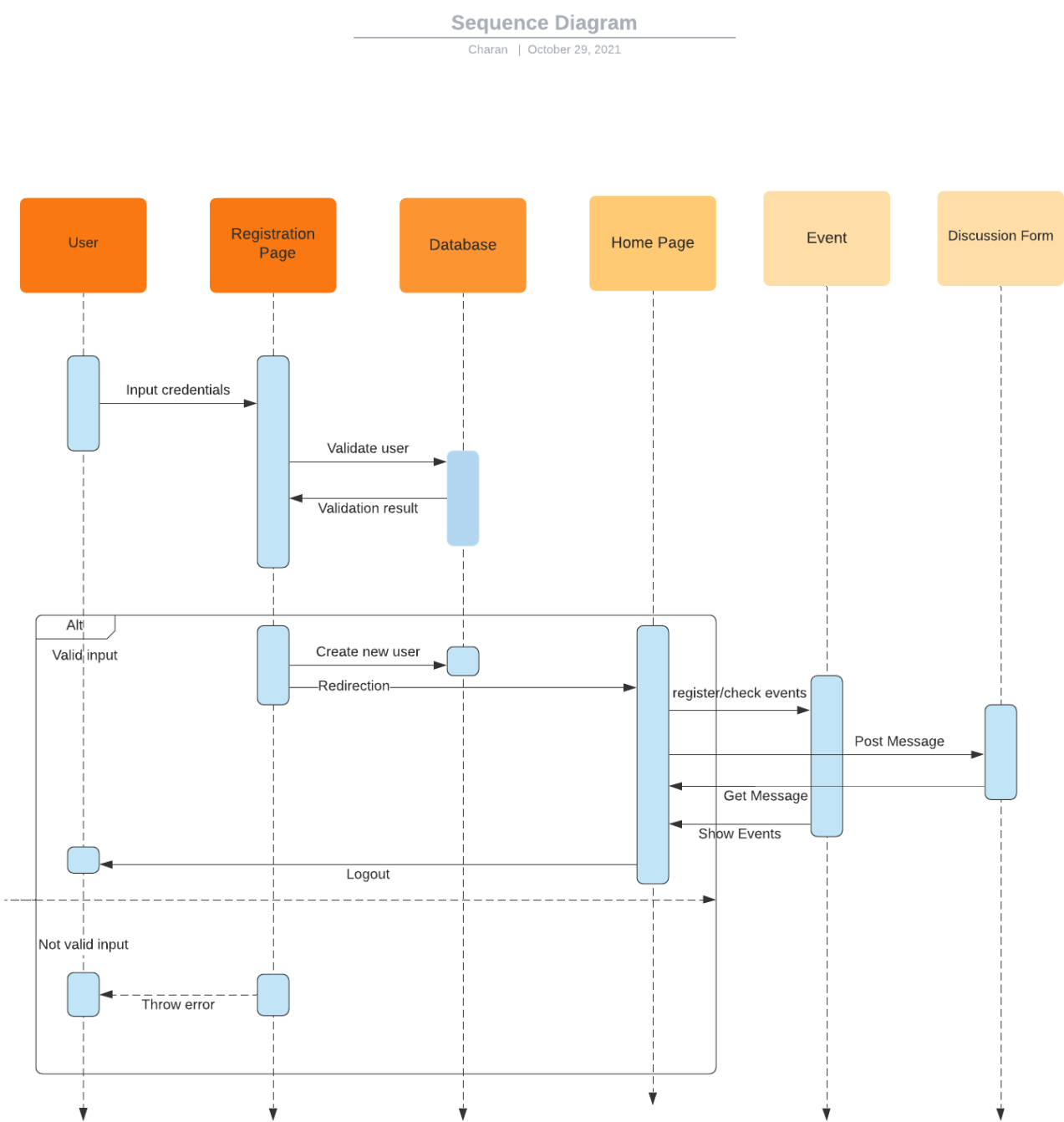
Class diagram



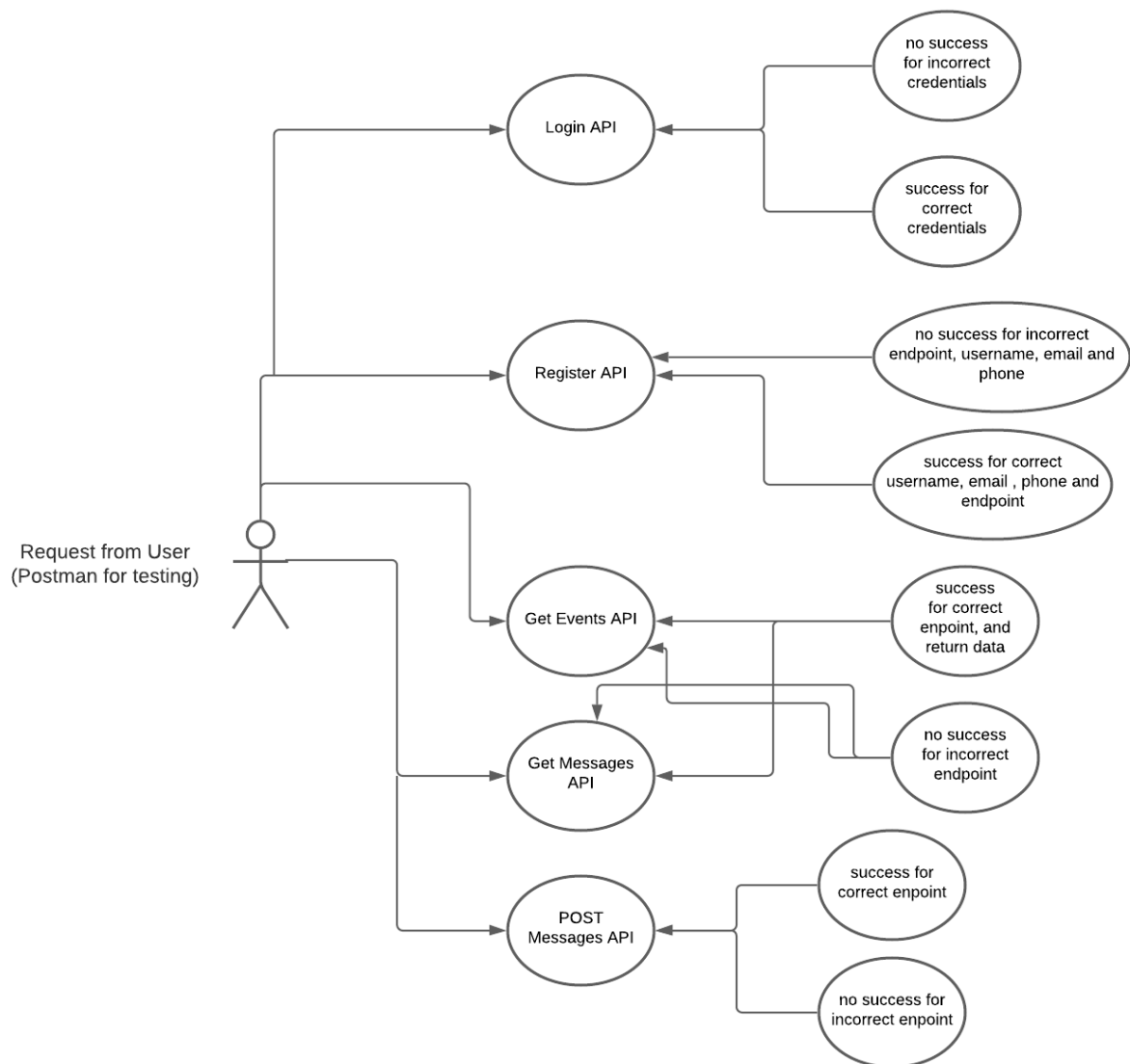
UML class diagram

Charan | October 29, 2021

sequence diagram



Use case diagram



c) Test cases

Test Cases

Test Case 1:

Verify whether the user is able to open the default app url and is able to see the login page.

Test Case 2: User enters the details with valid Credentials.

endpoint : /user/login , method : POST

requestBody : {"username" : "root", "password" : "root"}
response: true

Test Case 3: User enters the details with invalid Credentials.

endpoint : /users/login , method : POST
requestBody : {"username" : "root", "password" : "root"}
response: false

Test Case 4:

Click on the signup link in the login page and check whether the page correctly routes to the signup page .

Test Case 5: User registers with existing username.

endpoint : /user/register , method : POST
requestBody : {"username" : "root", "password" : "root", "email" : "sample@gmail.com", "phone" : "+19829982891"}
response: {"isRegistered" : false, "error" : "username already exists"}

Test Case 6: User registers with existing email.

endpoint : /user/register , method : POST
requestBody : {"username" : "user", "password" : "root", "email" : "sample@gmail.com", "phone" : "+19829982891"}
response: {"isRegistered" : false, "error" : "email already exists"}

Test Case 7: User registers with existing Phone number.

endpoint : /user/register , method : POST
requestBody : {"username" : "root", "password" : "root", "email" : "user@gmail.com", "phone" : "+19829982891"}
response: {"isRegistered" : false, "error" : "phone already exists"}

Test Case 8: User registration details are valid.

endpoint : /user/register , method : POST
requestBody : {"username" : "root", "password" : "root", "email" : "sample@gmail.com", "phone" : "+19814582891"}
response: {"isRegistered" : true}

Test Case 9 :

Click on the login link in the signup page and check whether the page correctly routes to the Login page .

Test Case 10: Click on ongoing events page.

endpoint : /sports/events , method : GET
response: [{"id" : 1, "name" : "Long Jump", "Location" : "USA", "time" : "9:00 – 10:00", "date" : "11-10-2021", "eventType" : "ONGOING", "participants" : [{"id" : 1, "name" : "John", "eventId" : 1}]}]

Test Case 11:

endpoint : /sports/sportevent , method : GET
response: false

Test Case 12: When user clicks on discussion forum tab.

endpoint : /forum/discussions , method : GET
response: {[{"id" : 1,"text" : "Hello","userId" : 1,"time" : "2021-10-11 20:35:43.110","username" : "John"}]}

Test Case13:

endpoint : /forum/messages , method : GET
response: false

Test Case14: when user post a message in discussion forum.

endpoint : /forum/discussions , method : POST
requestBody : {"text" : "I would like to know the timing of the High Jump event","userId" : 1}
response: true

Test Case15: When post message is not delivered.

endpoint : /forum/messages , method : POST
requestBody : {"text" : "I would like to know the timing of the High Jump event","userId" : 1}
response: false

Test Case 16 :

After successfully entering the hardcoded username and password in the Login page , the check page should correctly navigate to the Sports Management page .

Test Case 17 :

In the Sports Management page , check whether the Ongoing Event page should be selected by default..

Test Case 18:

In the Ongoing Event page , check whether all the hardcoded ongoing events are displayed properly without any UI collapse/CutOff on the screen .

Test Case 19:

In the Ongoing Event page, click the View Participants label in any one of the ongoing event cards, then it should show the hardcoded data as table only for that ongoing event card.

Test Case 20:

Click the Upcoming Event tab, then the screen should successfully display the Upcoming event cards without any error.

Test Case 21:

In the Upcoming Event page , check whether all the hardcoded Upcoming events are displayed properly without any UI collapse/CutOff on the screen .

Test Case 22:

In the Upcoming Event page, click the View Participants label in any one of the Upcoming event cards, then it should show the hardcoded data as table only for that Upcoming event card.

Test Case 23:

Click the Discussion Forum tab, then the screen should successfully display all the hardcoded details related to the forum without any error.

Test Case 24:

In the Discussion Forum page , check whether all the UI components are displayed properly without any UI collapse/CutOff on the screen .

Test Case 25:

In the Discussion Forum page , click Add New Post Button , then the screen should successfully open the editor without any error.

D) User Manual

Signup

Username *

john

Email *

john@gmail.com

Phone Number *

9848022338

Password *

....

SUBMIT

Existing User? [Login](#)

- End-User should create an account to register or participate an event.
- End-user must contain name, email-id, phone number and password to register.

Login

New User? [Sign up](#)

email or mobile number *

9848022338

Password *

....

LOGIN

- If user has an existing account, then he/she can directly login with “username” and “password”.
- Username must be the email-id or mobile number.

Ongoing Event:

Sports Management

Logout

ONGOING EVENTS

UPCOMING EVENTS

JOIN EVENT

DISCUSSION FORUM

ABOUT US

Event ID : 2

Event Name : High jump

Event Location:Chennai

Date:2021-10-03

Time:9:00 - 10:00

VIEW PARTICIPANTS

Event ID : 4

Event Name : Hockey

Event Location:Delhi

Date:2021-10-03

Time:10:00 - 11:00

VIEW PARTICIPANTS

Event ID : 7

Event Name : Javeline

Event Location:Mumbai

Date:2021-10-04

Time:9:00 - 10:00

VIEW PARTICIPANTS

- User can view the ongoing event, time, date and location of the event.

View participant:

BACK

Participant Name	Event ID	Event Name	Event Location
John	2	High jump	Chennai

- By clicking on view participant user can view the detailed information about the participant's name, event id, event name, and event location.

Upcoming event:

Sports Management

Logout

Event ID : 1
Event Name : Long jump
Event Location:Chennai
Date:2021-10-04
Time:9:00 - 10:00

VIEW PARTICIPANTS

Event ID : 10
Event Name : 100 M Dash
Event Location:Kolkatta
Date:2021-10-06
Time:11:00 - 12:00

VIEW PARTICIPANTS

Event ID : 3
Event Name : Badminton
Event Location:Mumbai
Date:2021-11-03
Time:14:00 - 15:00

VIEW PARTICIPANTS

Event ID : 5
Event Name : Squash
Event Location:Delhi
Date:2021-10-04
Time:10:00 - 11:00

VIEW PARTICIPANTS

Event ID : 6
Event Name : Weight Lifting
Event Location:Delhi
Date:2021-10-04
Time:9:00 - 10:00

VIEW PARTICIPANTS

Event ID : 8
Event Name : Tennis
Event Location:Kolkatta
Date:2021-10-05
Time:9:00 - 10:00

VIEW PARTICIPANTS

Event ID : 9
Event Name : Archery
Event Location:Kolkatta
Date:2021-10-05
Time:11:00 - 12:00

VIEW PARTICIPANTS

- This is about the upcoming event where end-user will check the upcoming events schedule.

Discussion Forum:

Add New Post

john2
hello

J

October 27, 2021
05:52 PM CDTjohn2
helloandi

J

October 27, 2021
05:54 PM CDT

Previous

1

2

Next

- Participant can resolve their doubts by using discussion forum.
- By clicking “Add New Post” button user can add new post so that he/she can resolve the queries.

About US:

Sports Management

Logout

“We are Code Geeks”

We as a team contributed hard enough to develop the Sports management system application. This application is an online platform where students can register and authenticate using the credentials. All the sports events that are going around in the university and the upcoming events that are going to be happened within a month are being displayed. The key theme of this application is to provide students about the updated information about the activities going around the university. Behind this successful application, we have collaborated a team and went through all phases of the software lifecycle and gained expertise on troubleshooting and maintain the application

Meet the Developers:

Anudeep: He has around 4+ years hands-on Java full stack and expertise on the IntelliJ and VS Code software's.

Charan: He holds an overall experience of 5+ years dealing with React and Angular frameworks and expertise in handling performance issues and applying optimization methods.

Abhiram: He has around 3+ years of experience in core Java and advanced Java applications and strong expertise in Spring boot and Hibernate concepts.

Aditya: He has more than 3 years of experience in React framework. Strong hands-on expertise in writing MySQL queries and designing the database.

Nitesh Reddy: He has around 3+ experience in handling Java spring boot and hibernate applications. Expertise in designing Oracle and MySQL databases.

Chaitanya: He holds an overall experience of 5+ years dealing with React and Angular frameworks. Strong backend expertise in designing MySQL database and writing automated test scripts.

Manvita: She has more than 3 years of experience in Angular framework. Strong hands-on expertise in writing Oracle queries and designing the database.

Saikirishna: He has around 3+ years of experience in core Java and advanced Java applications and strong expertise in Spring boot and Hibernate concepts.

- Here user can find the details of members of who developed the application.

E)Instructions:**PREREQUISITES**

List of Software required for running the application: Node js, Java, SQL

Ide Tools Required: STS/ECLIPSE/IntelliJ ,Vscode and SQL Workbench

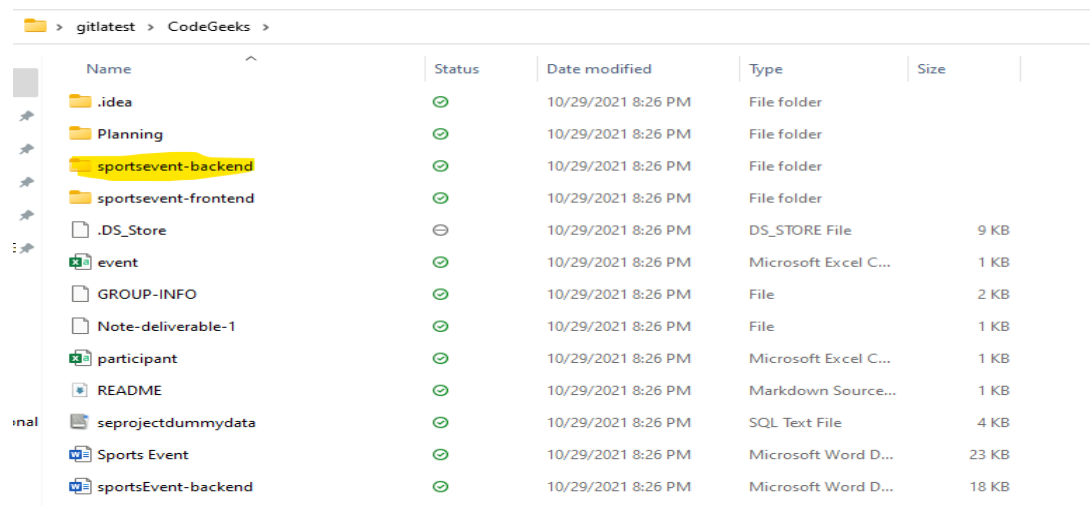
Clear instructions on how to compile/run both your program and your test cases

1)Clone/download the project directory from the git hub using the url

<https://github.com/CharanPottabathini/CodeGeeks>

2)Use the ide tools such as eclipse/IntelliJ/STS for using the backend application and VSCode for using the frontend.

3)Open the IntelliJ IDE and open the sportevent-backend directory present in Codegeeks directory



Name	Status	Date modified	Type	Size
.idea	✓	10/29/2021 8:26 PM	File folder	
Planning	✓	10/29/2021 8:26 PM	File folder	
sportevent-backend	✓	10/29/2021 8:26 PM	File folder	
sportevent-frontend	✓	10/29/2021 8:26 PM	File folder	
.DS_Store	⊖	10/29/2021 8:26 PM	DS_STORE File	9 KB
event	✓	10/29/2021 8:26 PM	Microsoft Excel C...	1 KB
GROUP-INFO	✓	10/29/2021 8:26 PM	File	2 KB
Note-deliverable-1	✓	10/29/2021 8:26 PM	File	1 KB
participant	✓	10/29/2021 8:26 PM	Microsoft Excel C...	1 KB
README	✓	10/29/2021 8:26 PM	Markdown Source...	1 KB
seprojectdummydata	✓	10/29/2021 8:26 PM	SQL Text File	4 KB
Sports Event	✓	10/29/2021 8:26 PM	Microsoft Word D...	23 KB
sportsEvent-backend	✓	10/29/2021 8:26 PM	Microsoft Word D...	18 KB

4)First run the program [CreateTablesAndSchema.java](#) so that the basic setup in database such as tables which are required for our application will be created in the database.

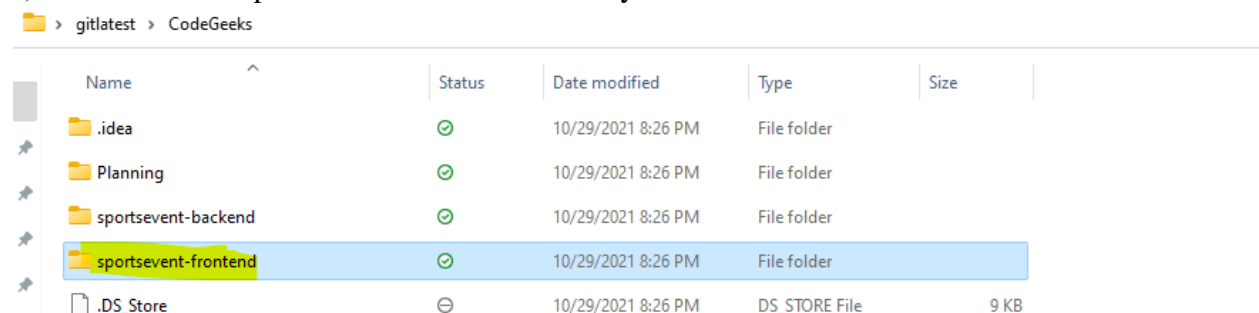
5)download and import the .sql files present in the code geeks directory to your database so that the dummy data will be created for our application

6)After creation of the tables in database run the [SportseventBackendApplication.java](#) program in order to start our backend server in the port 8080 .

7) Once the backend server is started ,the API's which are used to fetch/upload data will be available and ready to consume

8)After setting up the backend application we need to start the front end part of our application which is based on reactjs

9)Load the folder sports event-frontend directory



Name	Status	Date modified	Type	Size
.idea	✓	10/29/2021 8:26 PM	File folder	
Planning	✓	10/29/2021 8:26 PM	File folder	
sportevent-backend	✓	10/29/2021 8:26 PM	File folder	
sportevent-frontend	✓	10/29/2021 8:26 PM	File folder	
.DS_Store	⊖	10/29/2021 8:26 PM	DS_STORE File	9 KB

10)In order to run the frontend application we require node js as we need to install the node modules

11)Open the sports event-frontend application in vs code and enter the command **npm install** so that the node modules which are required for the application will be loaded into our project directory

12)After the installation of the node modules enter the command **npm start** front end part of application will get started at the port 3000

13)The Chrome browser will be opened with login page of the application is started and ready to use

14)Signup into the application by providing the details Username ,email, phone no, password and if any user is already registered with same mobile number or username or email the application will not accept the registration of the user by providing the error message that user already exists with that information

15)Provide unique details and get registered into the application after registering into the application sign in with provided details

16)The user can be logged in into the application by providing the details email id or mobile num and password

17)The home page consisting of the tabs **Ongoing events, Upcoming events, Join events, Discussion forum and About us** is displayed

18)Click on the **Ongoing events** tab, A page which list of ongoing events is displayed similarly by selecting the Upcoming events the list of upcoming events will be displayed

19)By clicking on the **Discussion forum** tab, A page which consist of messages that has been posted by different users will be displayed

20)By clicking on the **About Us** tab, A page with information of the developers will be displayed

F) Have implemented the review comments we have received during the code inspection.

G) What went well:

Everything went well according to the plan.

What could be improved.

The feedback we have received could have figured upfront.

H)**Member's contribution**

Member name	Contribution description	Overall Contribution (%)	Note (if applicable)
Charan Pottabathini	Ongoing and Upcoming Events User Interface	12.5%	
Nitesh Reddy Vedula	User Registration User Interface	12.5%	
Venkata Aditya Pavan Tripasuri	JUnit Test Cases and Sample Test Cases	12.5%	
Anudeep Raj Kyatham	Super User Page & Create Events User Interface	12.5%	
Saikrishna Gattu	Login Screen User Interface	12.5%	
Abhiram Nagol	Create Events Backend	12.5%	
Chaitanya Maniveer	Discussion Forum User Interface	12.5%	
Manvitha Kancharla	User Interface Design for Cards	12.5%	