Event Management System (EVM)

Problem Statement

Every student eagerly looks forward to any upcoming events conducted by colleges or schools such as tech fests, cultural, symposiums, hackathons, etc. They get to enjoy such wonderful events due to the work that is done by the hosting party andthe admins behind it. But sometimes, hosting and managing an event can be stupendous and a hassle. The admins need to manage each sub-event and the participants manually, which is very ineffective. The information about the event is also not very apparent. Further, human errors can affect the certification process. Hence, theflow of the event can be scabrous at times. This leads to an underwhelming experiencefor the students, but this can be tackled by a centralized event management system.

Software Requirement Specification

1.1Abstract

This project is aimed at developing an online event management system that is of importance to any institute to host an event. The Event Management System (EVM) is an Internet based application that helps the users to book for their preferred event in online mode. Meanwhile, the admins would be able to host and manage the events effectively and efficiently. This system can be used to automate the workflow of ticket purchase, ticket generation and event organization. At the end of each round in an event the participants who have been qualified can be kept track for the upcoming rounds by the hosts of the event. The generation of e-certificates for the winners at the end of each event is also automated. There are features like automatic email notifications, SMS notifications, track of events, certificate sharing, report generation etc in this system.

1.1 Introduction

This Software Requirements Specification provides a complete description of all the functions and specifications of the Online Event Management System. The expected audience of this document is the business development manager, Service Delivery and Support Manager, including members of the organization who will use this system.

Anyone who has hosted or managed an event will know the frustration of tracking each sub-events, and handling the data of each participant. This ingenious online event management software with its powerful features, cuts out the physical movements and enables admins to have a bird’s eye view of the events in a particular organization, and make required decisions based on the data.

There are many advantages of utilizing a centralized web-based event management system:

- ➢ Eliminates paper-based ticket purchase forms.
- ➢ Tickets can be purchased online.
- ➢ Eliminates the paperwork to keep track of the participants who show up to the event after registration.
- ➢ Both the participants as well as the hosts can view the progress of the events.
- ➢ Instantaneous generation of e-certificates for all the participants.
- ➢ Summarized report at the end of the event facilitates the host to interpret the results.

Participants can easily:

- ➢ Purchase tickets from home remotely.
- ➢ Keep track of the progress of events registered.
- ➢ Acquire e-certificates.
- ➢ Know about their qualification status at the end of each round.

Hosts can easily:

- ➢ Manage the data of the participants.
- ➢ Generate e-certificates automatically.
- ➢ Promote the qualified participants at end of each round.
- ➢ Notify the participants know about their status in the event they participate.

1.2 Scope

The Event Management System is designed to help the interested users purchase tickets for the event, they also have the facility to register and track sub-events that they wish to participate in. Consequently, the event hosts can manage the event, and approve the participants who have won the sub-events. The application also allows multiple such events to be hosted.

1.3 Intended Audience

The document is intended for requirements engineer, developer, testers, project manager and the users of the system.

1.4 Users and Characteristics

1.4.1 Participant

- ➢ Participants can check through/search for their interested sub-events in an event.
- ➢ Could easily view the details of the event and purchase tickets.

1.4.2 Host

- ➢ Hosts set up the event with all the details.
- ➢ Hosts create sub-events and assigns the admin who thereafter takes over the charge for the event.

1.4.3 Admin

- ➢ Admins would be able to handle and manage all the sub-events hosted by hosts in an event.
- ➢ Admins qualify the participants to the next round of the event.

2.0 Functional Requirements

Following is a list of functionalities of the system. There are hosts and users. Users of the application are school/college students who are willing to participate in school/college events. The hosts are the event in charges, who want to host events like hackathons, symposiums and tech/non-tech fests.

2.1 User Functionalities

2.1.1 Login/Sign up

The user who is willing to participate in any event must login/sign up before registering for the intended event.

- ➢ The user can sign up/login using their Gmail account.
- ➢ Once logged in the user can view events that are open for registration.

2.1.2 Register for events/sub events

The user can search for events that they want to register for, and register for the events. The user can additionally register for sub-events.

- ➢ The user selects teammates in case it is a team event or they can participate alone.
- ➢ Once the user selects team members, they might have to make payments depending on the requirements of the event.
- ➢ After the payment is made, the user has to wait for the host to approve the payment.
- ➢ Once approved, the team is provided with a unique QR code with a register id and a ticket.
- ➢ The team can now register for as many sub-events that they are allowed to register for.

2.1.3 Payment

The user has to make a payment for the event to acquire a ticket. The payment can be done using online transactions like UPI.

- ➢ The user has to pay a fee in order to get a ticket for the event.
- ➢ Once the payment is done using an online transaction, the user has to send the transaction id to the host for verification.

- ➢ After verification, the user is provided with a ticket indicating that they are a participant of the event.

2.1.4 Logout

The user can log out of the application whenever the user wishes to not use the application.

- ➢ The user can log out of the application, when the user is done using the application.
- ➢ Once the user is logged out of the application, they have to login back into the application to access their account.

2.2 Host Functionalities

2.2.1 Login

The host/event manager who wants to host an event in the platform has to login to the system before setting up the event.

- ➢ The host can log in to the system using their credentials.
- ➢ Once logged in, the host can see the events that are currently hosted by him if present.
- ➢ Host can start to set up a new event after successful login.

2.2.2 Create Event

The host can set up an event providing all the details required.

- ➢ The host can provide an event name, event poster.
- ➢ Host provides the venue of the event, cost of the ticket for the event registration.
- ➢ The host can provide the start and end date of the event.
- ➢ Host provides the start and end date/time for the registration.
- ➢ A small description of the sub-event can be provided by the host during this setup phase.

2.2.3 Create sub-event

The host can create sub-events for a created event.

- ➢ The host can provide a sub-event name, sub-event poster.
- ➢ Host provides the venue of the sub-event, cost of the ticket for the sub-event registration.
- ➢ The host can provide the start and end date of the sub-event.
- ➢ A small description of the sub-event can be provided by the host during this setup phase.
- ➢ Number of entries, prizes can be provided by the host.

2.2.4 Add admins

The host can assign the admins for each sub-event.

- ➢ The admin has to hold an account as an user in the system to be assigned as an admin by the host.
- ➢ The host has to add the username of the admin he wishes to assign.
- ➢ Admins are given the permissions as hosts to the sub-event assigned.

2.2.5 Approve payments

The host can view payments made by the users for the event. The host can approve or deny the request made by the users, if the host chooses to approve the payment then a ticket is generated along with a QR code and sent to the participant.

- ➢ The host can view the ticket request made by the user in their dashboard.
- ➢ The host/admin can approve/deny the requests.

- Once the request is approved, a ticket is generated, with a unique QR code and sent to the user. The user now becomes the participant of the event.
- Once the request is denied then the payment needs to be reversed.

2.2.6 QR generation

The host would allot a unique QR code to the participants.

- After the payments are cross checked by the host, a unique QR code would be generated.
- The QR generation is based on the user Id and the event the participant has registered for.

2.2.7 Set up e-certificate

The host should provide the template of the e-certificate to be awarded to the participants.

- The host while setting up the event has to provide a template for E-certificate.
- The template can be uploaded by the host.

2.2.8 Open/Close registration

Automatic opening and closing of deadlines for the event registration.

- Registration functionality is activated automatically based on the start date/time provided by the host.
- Registration functionality is deactivated automatically based on the end date/time provided by the host.

2.2.9 Update/Delete event

The host can update the event before the end of the event, and the host can delete the event once the event is completed.

- The host can choose to update any information regarding the event, before the completion of the event.
- The host can also change the sub-event's information.
- The host is also provided with additional functionality, which allows them to delete the event after the end of the event.
- They can also delete and sub-event before the start of the event.

2.2.10 Logout

The host can log out of the application whenever the user wishes to not use the application.

- The host can log out of the application, when the host is done using the application.
- Once the host is logged out of the application, they have to login back into the application to access their account.

3.0 Non-Functional Requirements

These are requirements that are not functional in nature. Specifically, these are the constraints the system must work within. The application must be compatible to work with all the browsers. This system will make use of Internet security such that the website is reliable and secure.

Sample Modules

Apply	1. Enable participants to register for their required sub-event. 2. Email will be sent to the participants to keep track of.
Edit	1. Participants would be permitted to edit the username,password,email and other contact details.

Dashboard	1. Participants would be able to view their registered events. 2. Event progress at the end of each round would be updated. 3. E-certificates would be uploaded for both the participants and the winners.
Search	1. Participants aren’t limited to events on the dashboard, they can search events on the basis of their interest.
Verify	1. QR code on registration given to the participants must be scanned to verify the participants attendance. 2. The Login process would have prior verification, keeping participants' data secure.
Payment History	1. Payment log history maintains the date of payment,amount of payment and sub-events registered in an event.

4.0 Hardware Requirements

➢ A computer with a CPU speed of 1GHz or higher processor.
➢ 8GB of RAM and 512GB of ROM.
➢ A camera to scan the QR code of the participants.
➢ Internet connection with a bandwidth of 10Mbps.

5.0 Technologies

- ➢ HTML, CSS, Bootstrap, JavaScript
- ➢ Java Servlets
- ➢ MySQL
- ➢ Apache Tomcat 9.0(local)

6.0 Operating Environment

- ➢ Windows/ Linux / Mac OS
- ➢ Web Browser: Google Chrome/ Mozilla Firefox/ Microsoft Edge

7.0 Design and Implementation Constraints

- ➢ The communication between the portal software and the database will be in SQL.
- ➢ The portal layout will be produced with HTML/CSS.
- ➢ A structured RDBMS allows for less flexibility, but it provides reliability and consistency.
- ➢ The source code must follow the coding conventions of technologies used.

8.0 Performance Requirements

- ➢ The load time for user interface screens shall take no longer than two seconds.
- ➢ The log in information shall be verified within five seconds.
- ➢ Once payment is approved ticket generation shall be done within 10 seconds.
- ➢ Updates on the events shall be reflected within 2 minutes.

9.0 Security Requirements

Users and Hosts will be able to log in to the Event Management System. Hosts will have access to the event setup and management subsystems. Users who are given admin rights will have access to the event management subsystem as well. Access to the respective subsystems of the application as per the user role in the event will be protected by a user login screen that requires a valid user Id and password.

10.0 Software Quality Attribute

10.1 Reliability

The reliability of the overall program depends on the reliability of the separate components.

10.2 Security

- ➢ Passwords will be saved encrypted in the database in order to ensure the user's privacy.
- ➢ The system will be protected against vulnerabilities such as SQL injection attacks.
- ➢ Gmail Authentication is required for logging into their respective accounts.

10.3 Availability

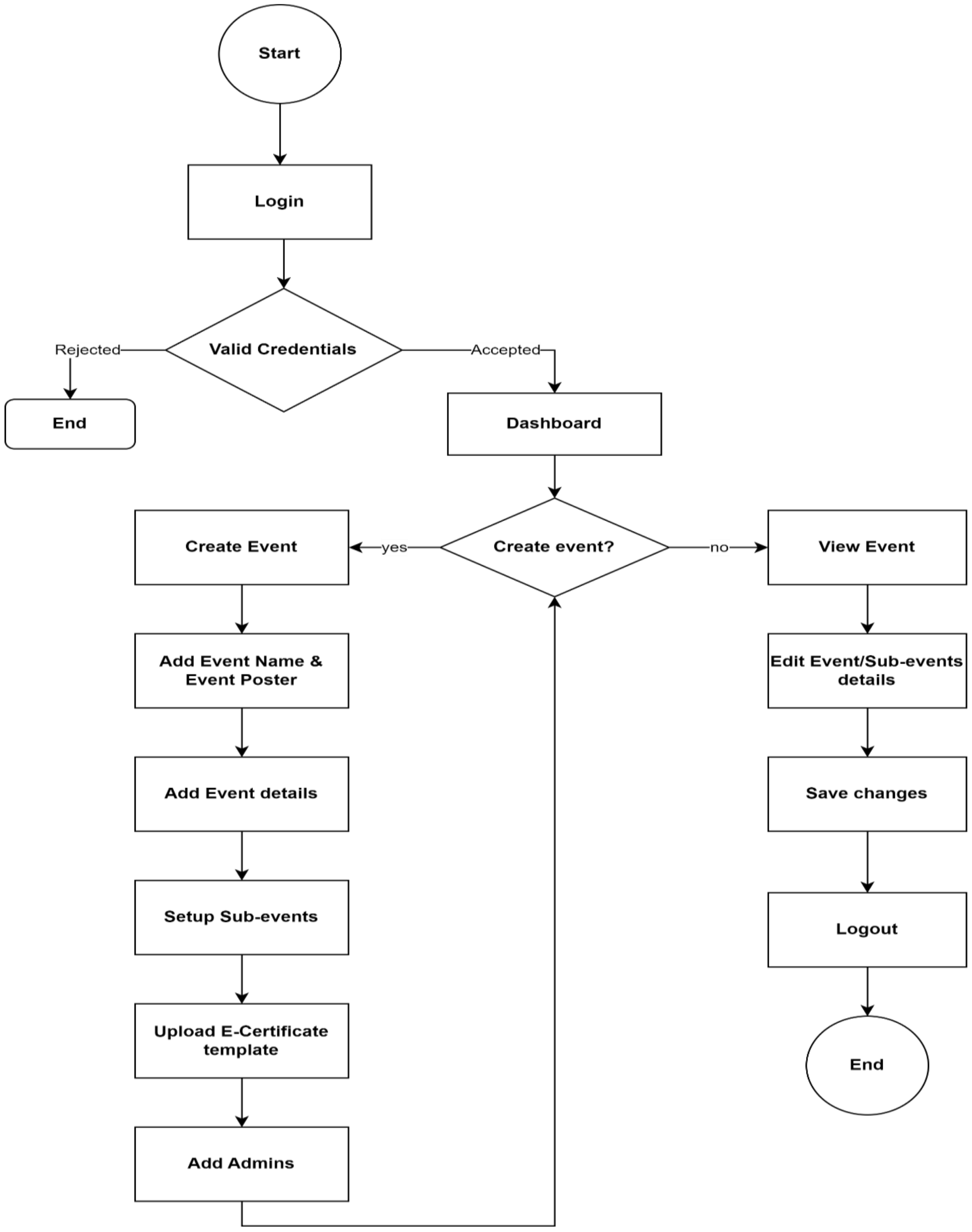
The system shall be available 24*7, except in case of any unexpected issues.

10.4 Usability

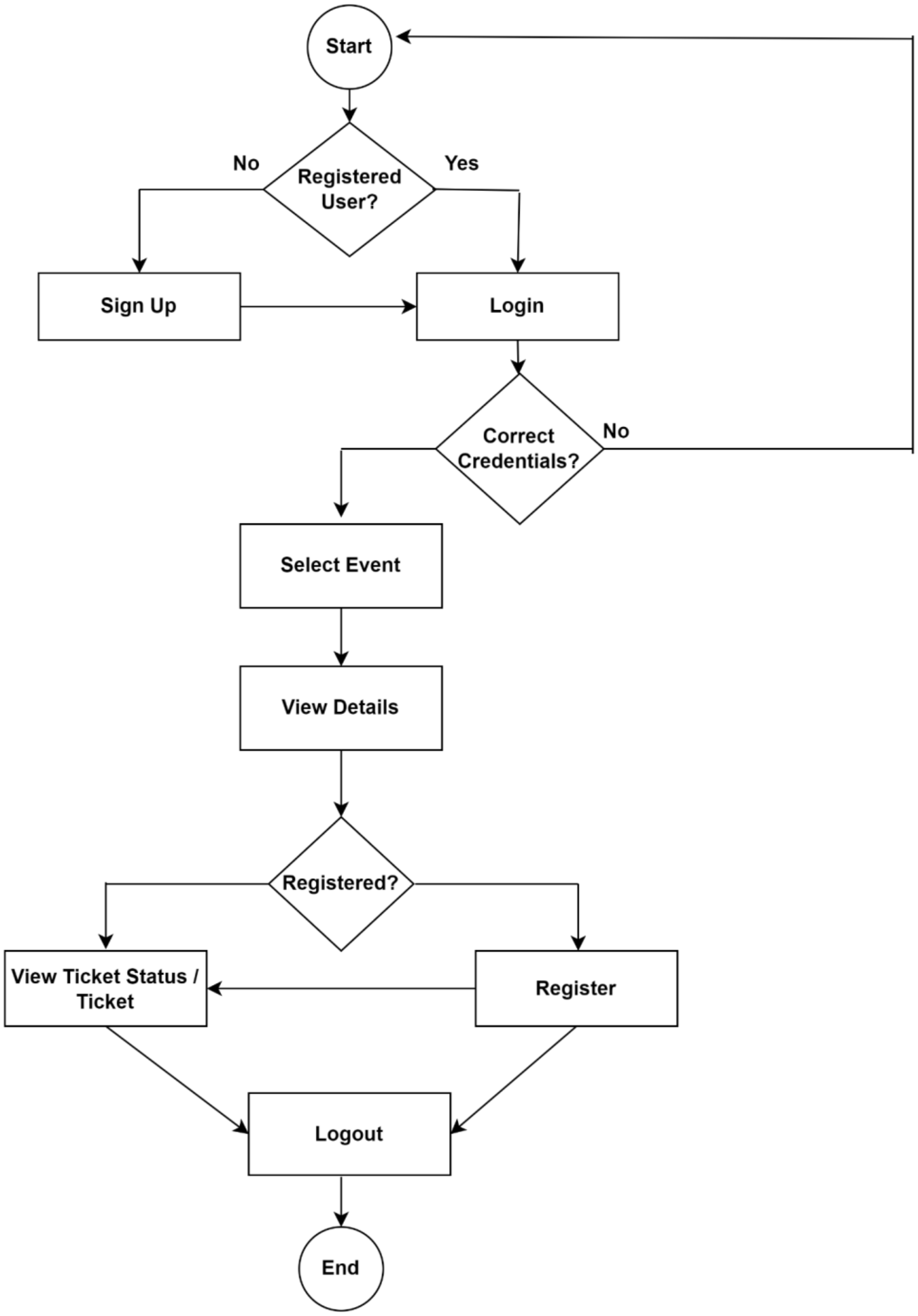
The interface should be easy to learn without a tutorial and allow users to accomplish their goals without errors.

11.0 System Flow Diagram

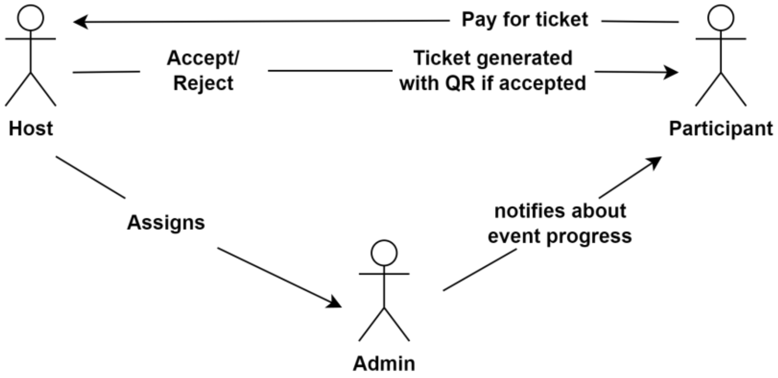
11.1 Host



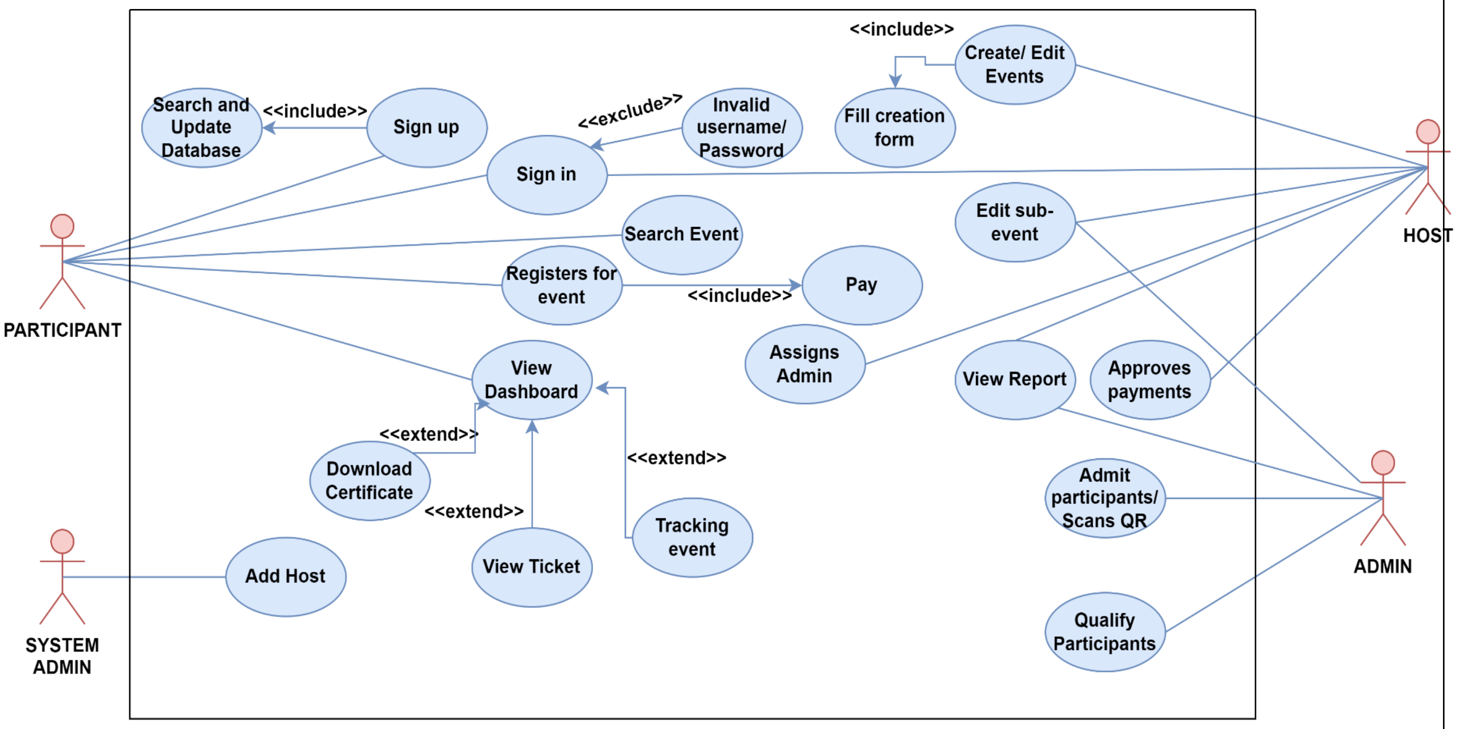
11.2 Participant



11.3 Event Workflow Diagram


Pay for ticket
Accept/ Reject
Ticket generated with QR if accepted
Host
Participant
Assigns
notifies about event progress
Admin

12.Use Case Diagram-



12.1. Use Cases-

- **Sign up:** Allows participants to sign up for the application
- **Search and Update Database:** On sign up, the database is updated automatically.
- **Sign in:** Allows both participants and managers with an account to log intothe application.

- ➢ **Invalid username/password:** On Authentication, the entered username and password is checked against the database to determine if it is a valid authentication or not.
- ➢ **Search event:** Allows the participants to search for the desired event.
- ➢ **Register for event:** Allows the participants to register for the desired event.
- ➢ **Pay:** For registering for an event, participants need to make a payment to the host in order to get a ticket for the event.
- ➢ **View dashboard:** Allows the participants to view the history of events attended by them.
- ➢ **Download certificate:** On dashboard, the participants can download the certificates of the event that they participated in.
- ➢ **View ticket:** On dashboard, the participants can view the ticket for registered events.
- ➢ **Tracking events:** On dashboard, the participants can track the events that they are participating in.
- ➢ **Add host:** The system admin can add the host and give them hosting authorization on host request.
- ➢ **Create/Edit events:** Allows the host to create the event and the sub-events in it.
- ➢ **Fill creation form:** On creation of an event, the host needs to fill a form to indicate the event details.
- ➢ **Edit sub-events:** Managers can edit the details of the sub-events, but only host can create new sub-events
- ➢ **Assign admins:** The host can authorize users to be admins, to give them admin authorization for the event and its sub-events.
- ➢ **Approve payments:** The host can approve requests with payment made by the participant to produce a QR code with a ticket for the participant on approval.
- ➢ **Admit participants/Scans QR:** The admins can admit the participants for the sub-event based on scanning the participants QR.
- ➢ **Qualify participants:** The admins can select the participants who have qualified for the next round and they can announce winners too.
- ➢ **View Report:** The host can view a summarized report at the end of the event depicting the analysis of the event.

12.2. Identification of Actors-

- **Participant:** User using the application to participate in the events.
- **System Admin:** Maintenance engineer who adds the host directly to the system.
- **Host:** A manager with the capability of creating new events and adding admins.
- **Admin:** A manager who was given authorization by the host only for the setting up of the sub-events.

12.3. Identification of Scenarios-

Scenario 1

Actors	Participant
Goals	Log into the system
Pre-conditions	-
Post-conditions	The Participant is logged into the application
success scenario	If the participant has an account already, the participant is taken to the login page directly. If not they are taken to the sign up page where they are requested to fill the details needed for setting up an account. After successful setup, the database is updated and they are directly logged into the application. Whereas in the login page they are requested to give the registered mail and password. If they are correct, they are successfully logged into the application.
Failure scenario	If the participant has an account already, the participant is taken to the login page directly. In the login page they are requested to give the registered mail and password. If they are incorrect, they are not logged into the application

Scenario 2

Actors	Participant
Goals	Log out of the system
Pre-conditions	The Participant must be logged in
Post-conditions	The Participant will be logged out of the application
success scenario	The participants go to their profile and log out from the event management system. The profile is only applicable to the users who have logged into the system.
Failure scenario	the participants are not logged into the system, so their profile is not setup, thus they can’t log out.

Scenario 3

Actors	Participant
Goals	Register for an event
Pre-conditions	The Participant must be logged in
Post-conditions	The Participant receives an acknowledgement mail and a ticket with it.
success scenario	the participant logs into the event management system and searches for the event that they are interested in, and they register for the event giving details requested by the host, and they make a successful payment. They submit the request with the invoice of the payment, and they have to wait for the host to approve the request. When the host accepts the request, the participant receive a mail with their unique ticket for the event

Failure scenario	the participant logs into the event management system and searches for the event that they are interested in, and they register for the event giving details requested by the host, and they make a successful payment. They submit the request with an invoice of the wrong payment, and they have to wait for the host to approve the request. The host sees the wrong invoice and the participant receives a mail with a message intimating that they have sent a wrong invoice.

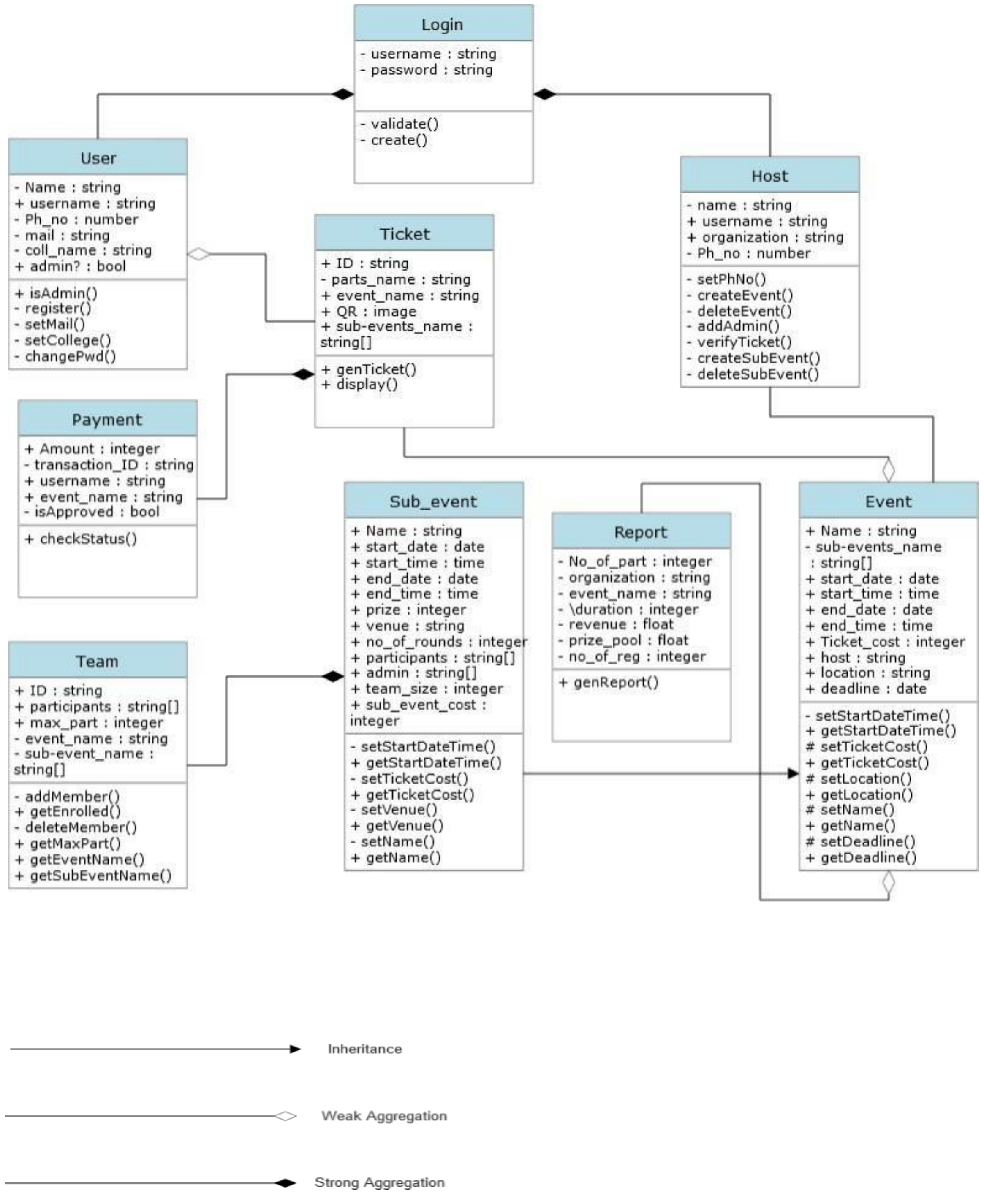
Scenario 4

Actors	Host
Goals	Add admins to sub-events
Pre-conditions	The host must have a event setup
Post-conditions	The host assigns admins to respective sub-events
success scenario	The host creates sub-events inside a pre-existing event. In each sub-event edit page, they have the option to add admins. When they click on the add-admin button, a pop-up prompts the host to enter the admins username, if the username is valid, the admins are given access to editing just that sub-event.
Failure scenario	The host creates sub-events inside a pre-existing event. In each sub-event edit page, they have the option to add admins. When they click on the add-admin button, a pop-up prompts the host to enter the admins username, if the username is invalid, an error message is shown to the host stating that a valid username to be give

Scenario 5

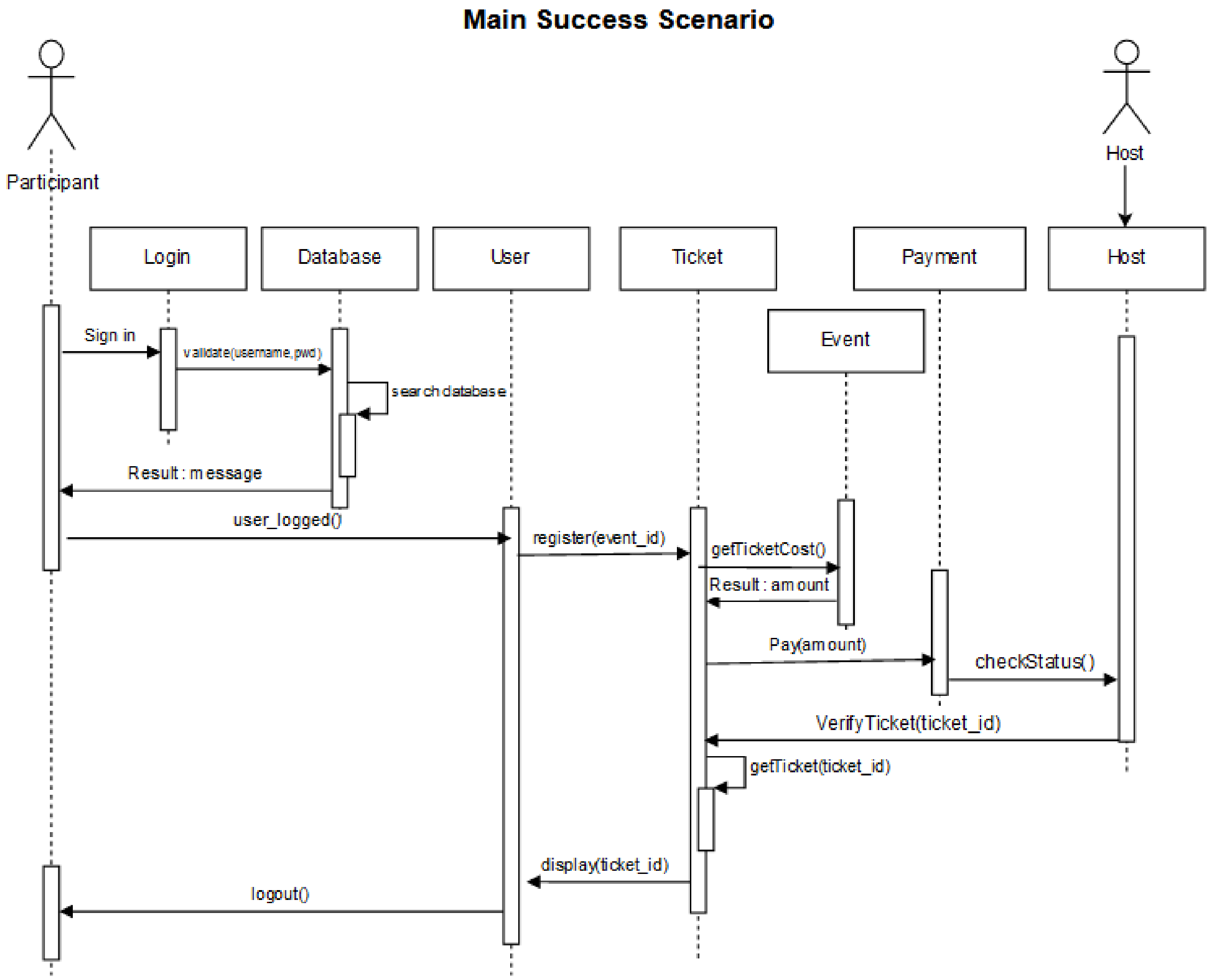
Actors	Host
Goals	Viewing report
Pre-conditions	The host must have a finished event
Post-conditions	The host can infer the analysis from the report
success scenario	Once an event is completed, the event’s status is updated in the dashboard of the host. The host goes to their dashboard and they can go into the respective event to view the report of it.
Failure scenario	The host goes to their dashboard and go to an ongoing event, the report is not yet generated thus they cannot view it.

13.CLASS DIAGRAM



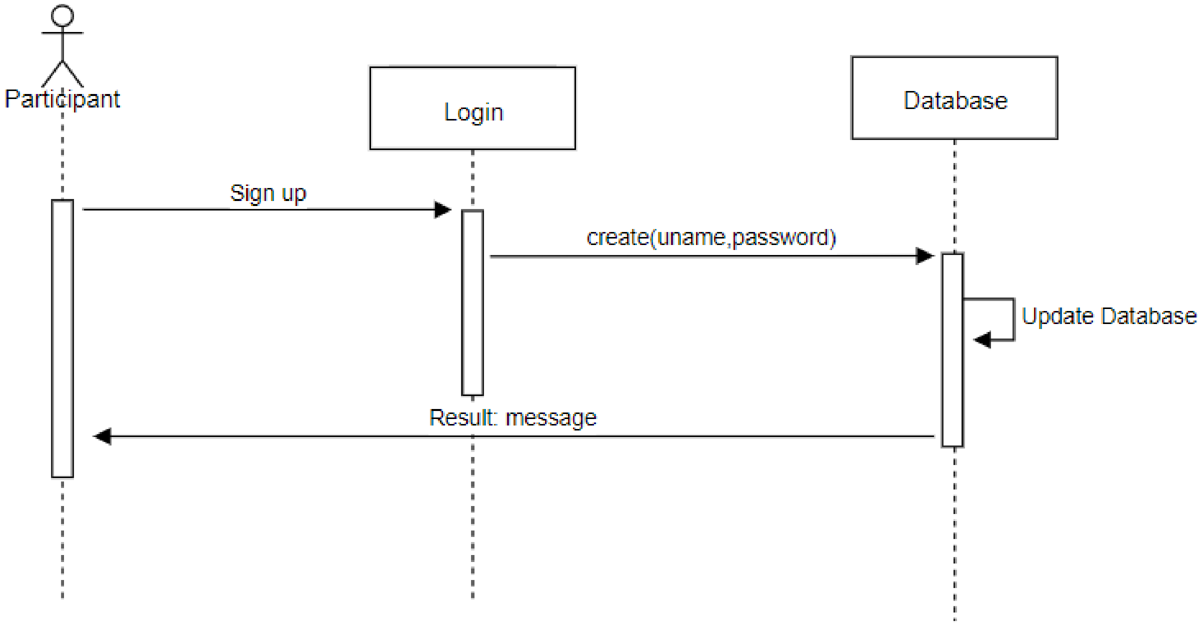
14.Sequence Diagram-

14.1.Sequence diagram for the main success scenario - Participant logs into the application successfully and registers for an event which involves ticketacquisition by making a payment to the host of the event. The payment should be approved by the host in order for the user to get a ticket for that event. Once approved the user can log out of the application if he choosesesso.

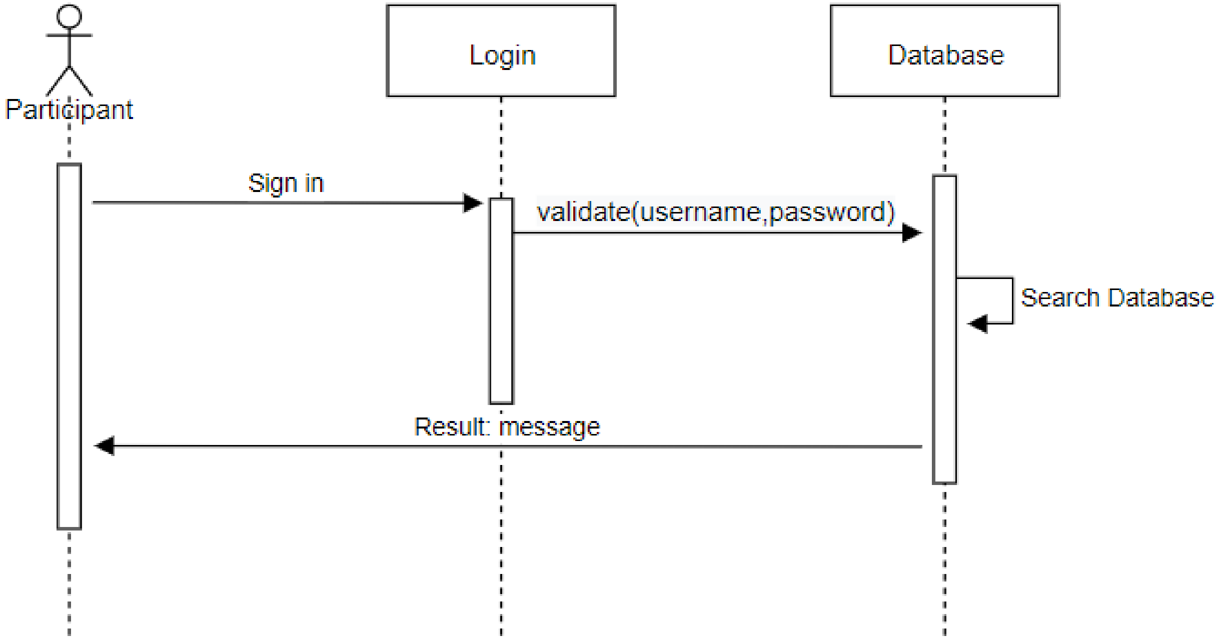


14.2. Sequence diagram for each use case -

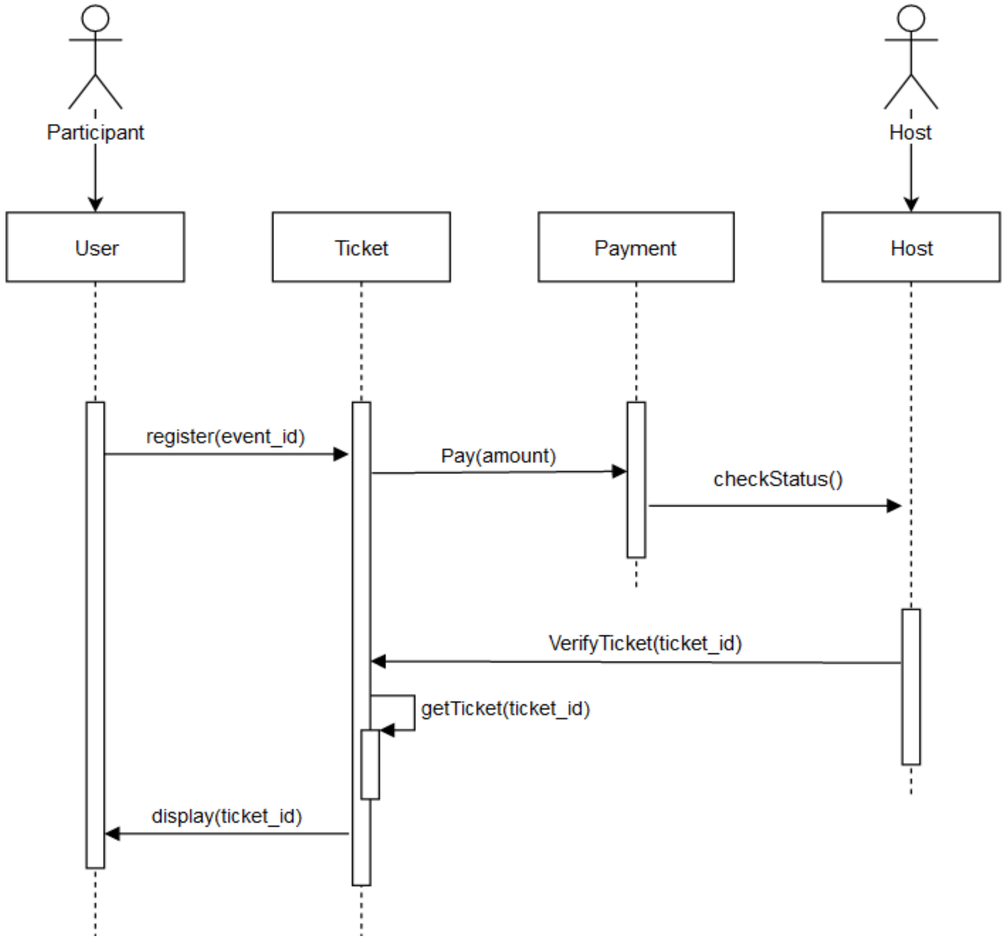
Signup



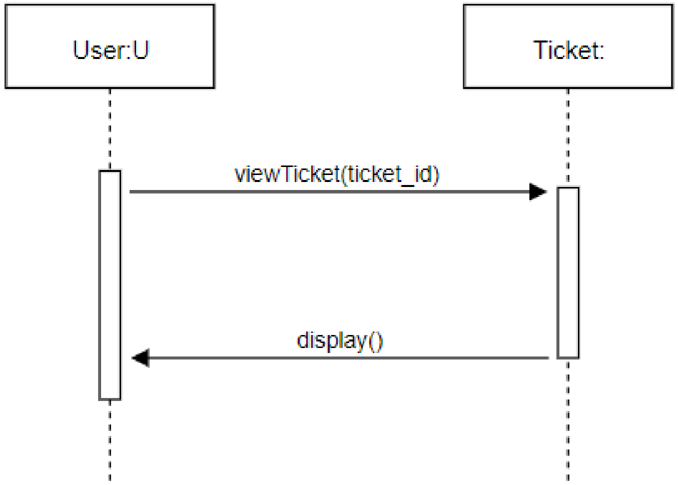
Login



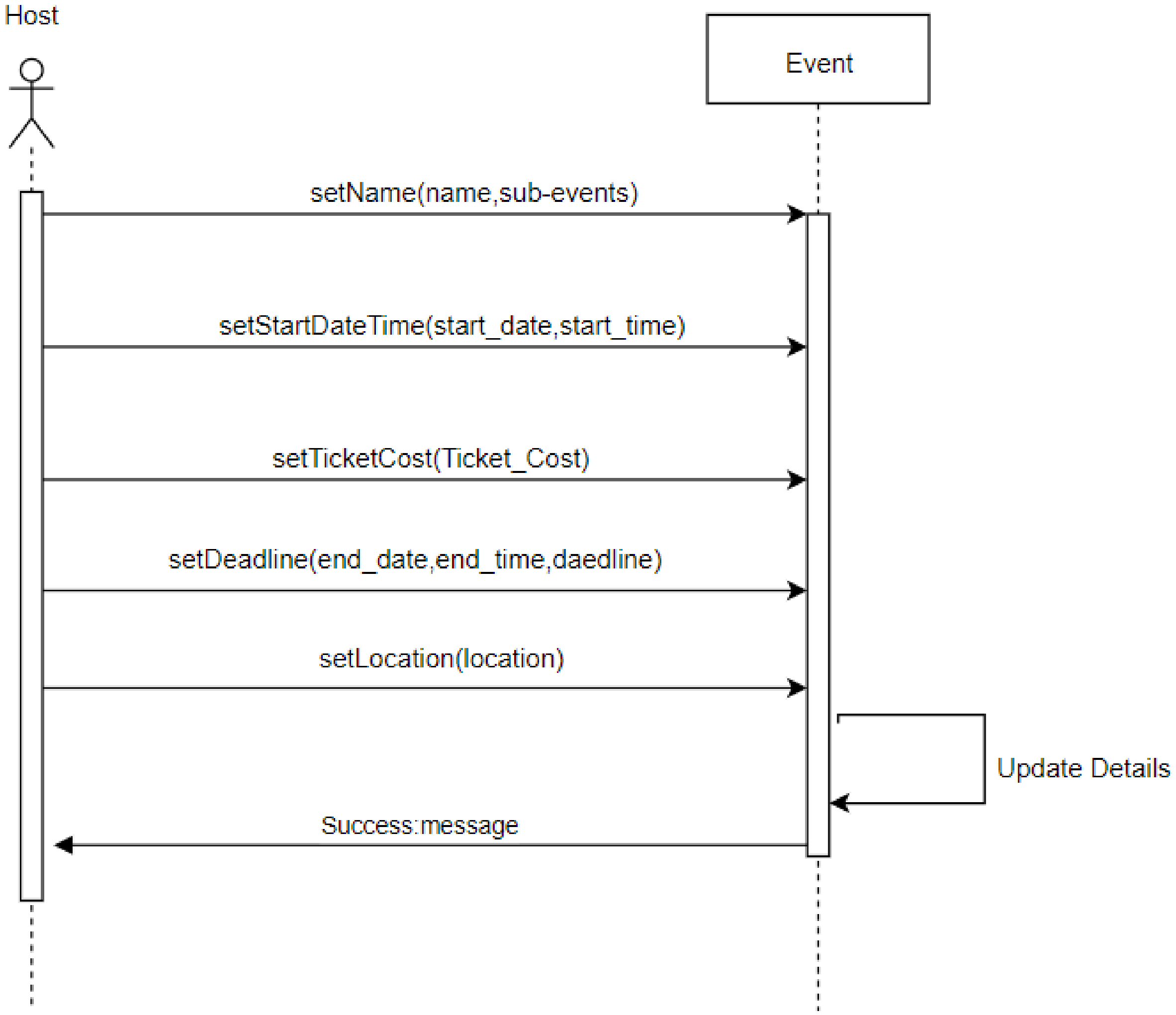
Registering for an event



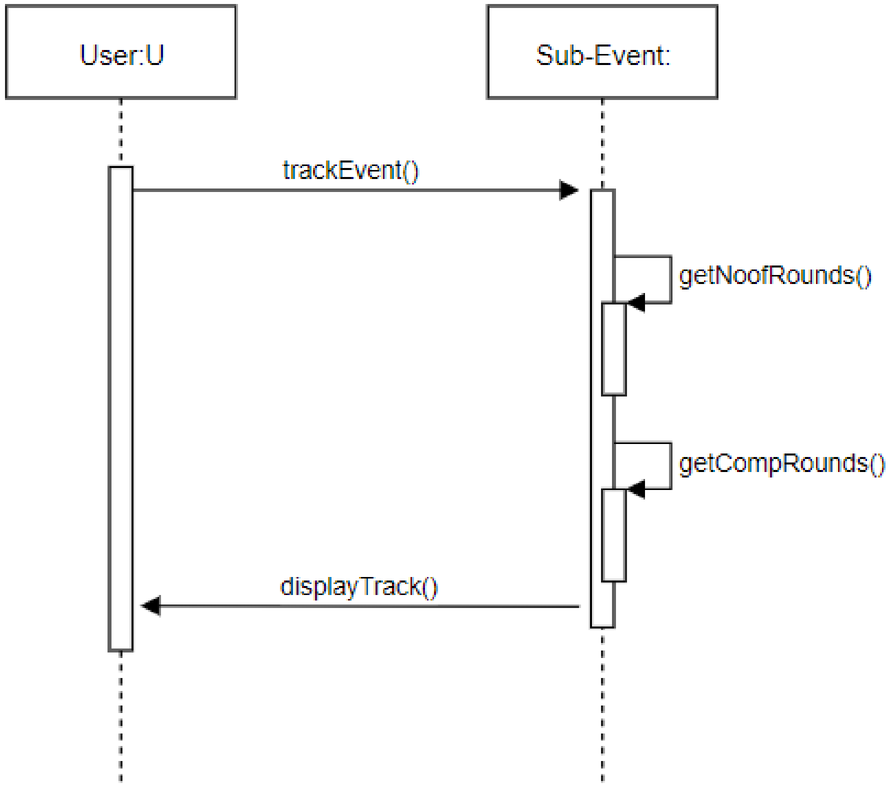
View Ticket



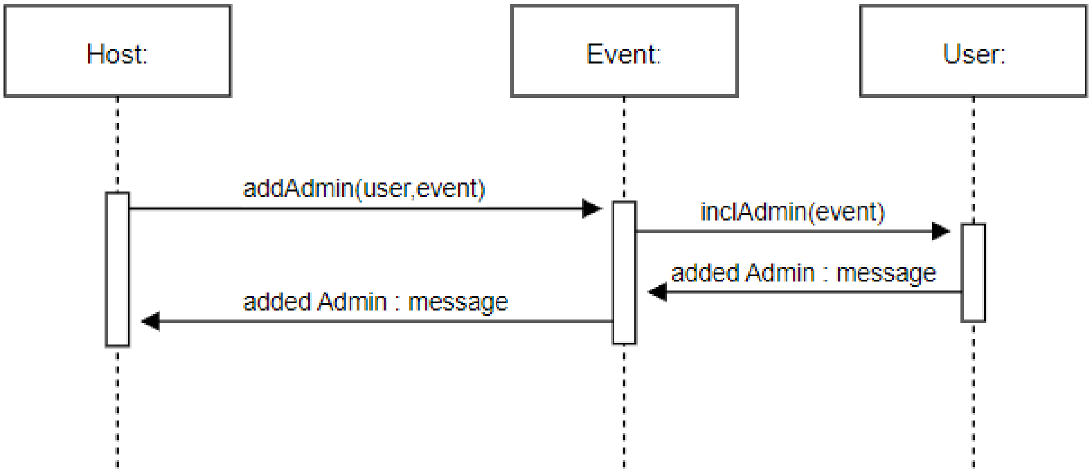
Create Event/ Sub-event

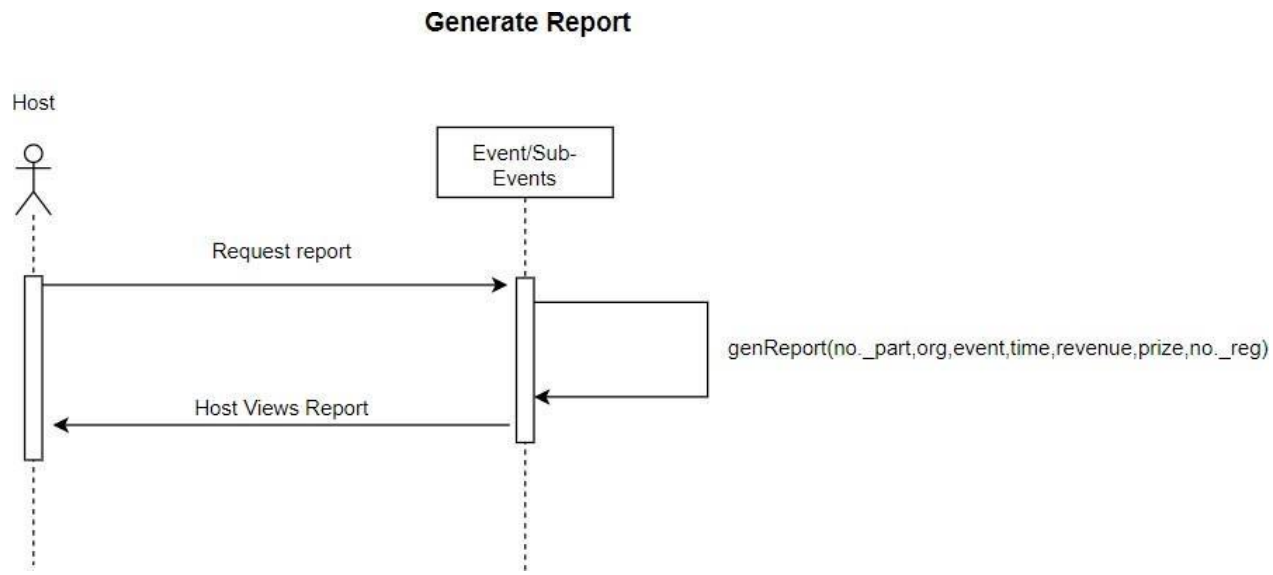

Host
Event
setName(name,sub-events)
setStartDateTime(start_date,start_time)
setTicketCost(Ticket_Cost)
setDeadline(end_date,end_time,daedline)
setLocation(location)
Update Details
Success:message

Track Event

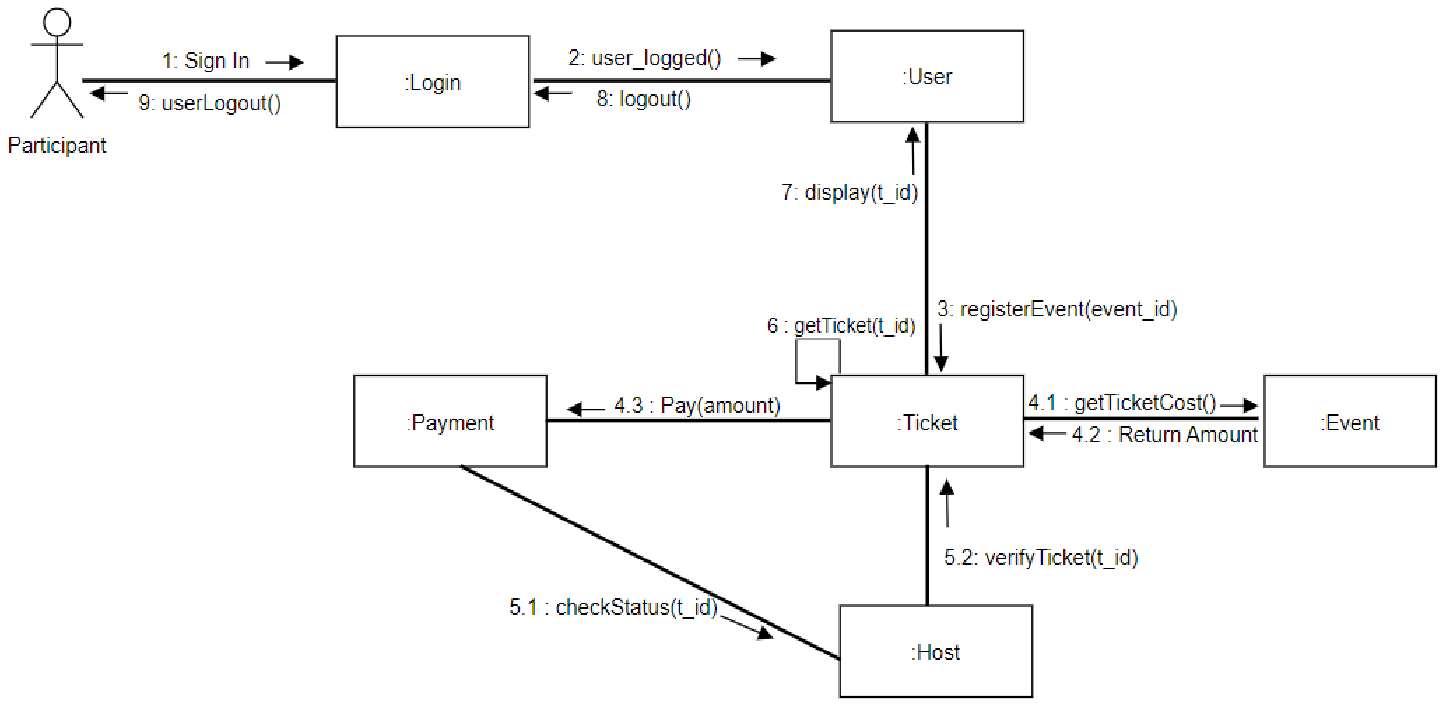


Assign Admins

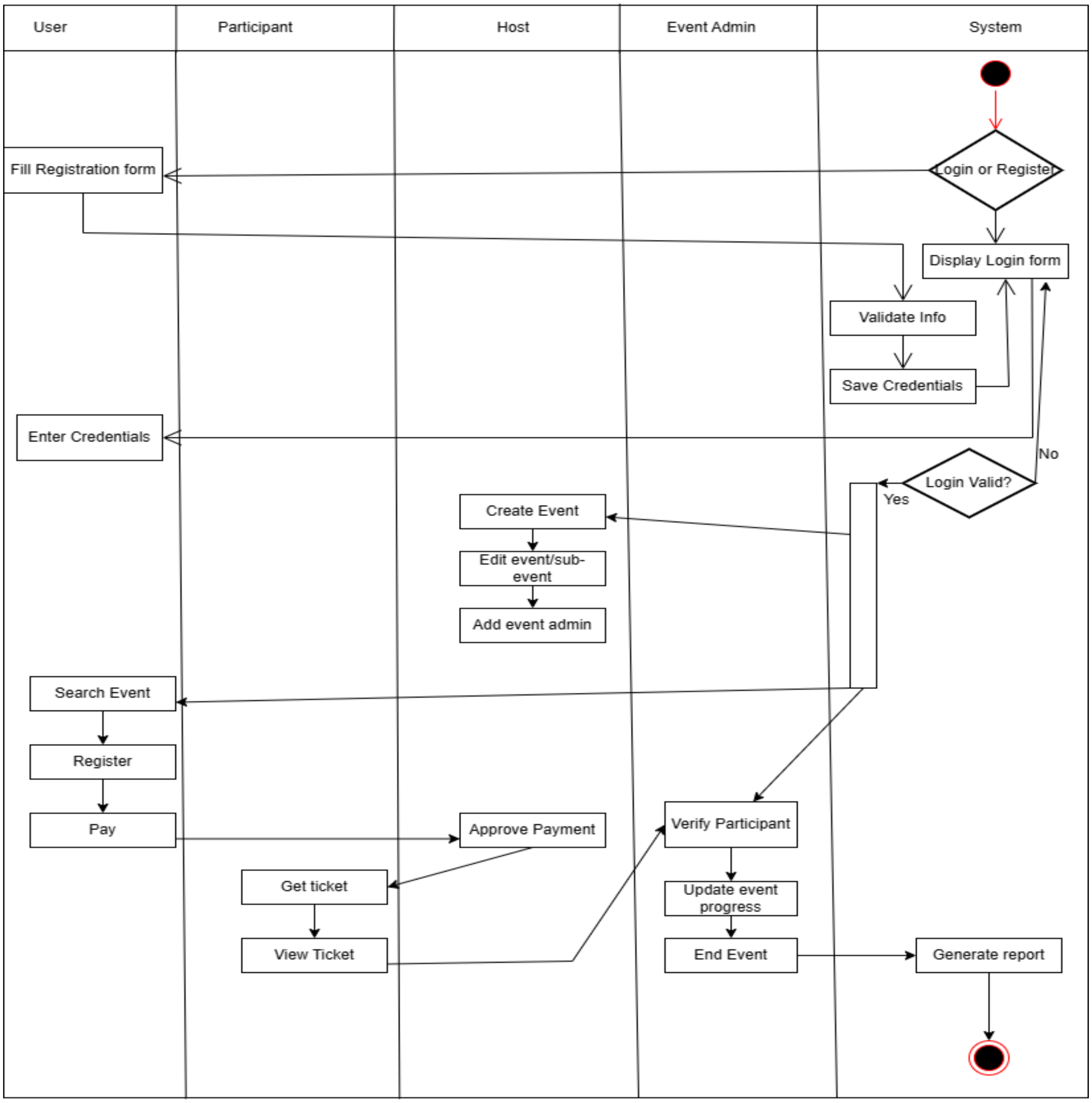




15.Collaboration diagram for the main success scenario -

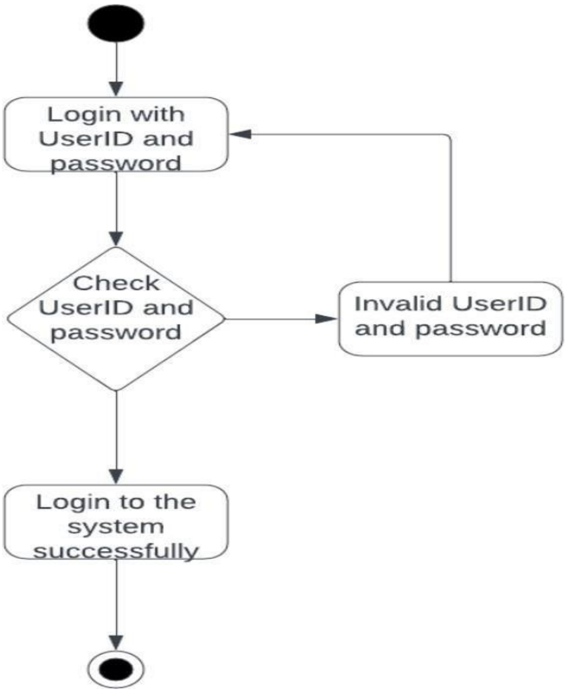


16.Activity Diagram

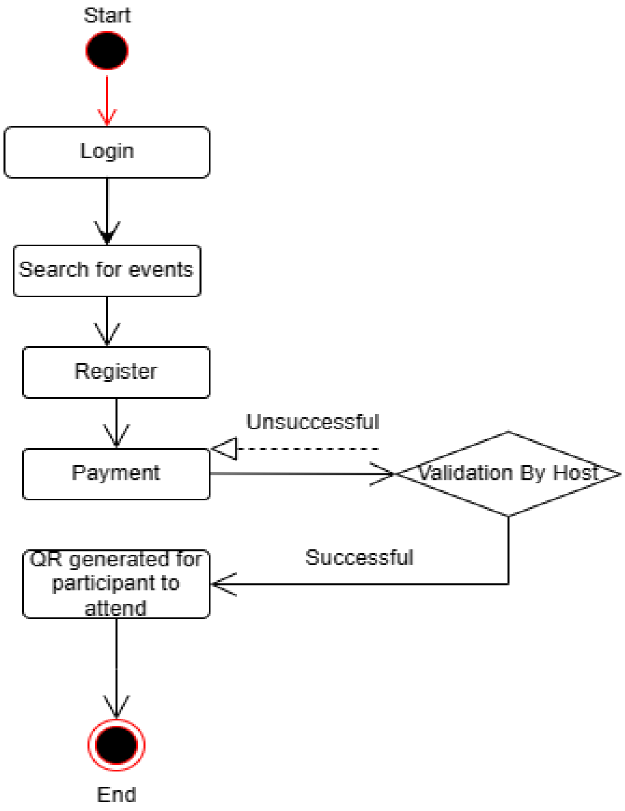


16.1. Sub-Functions

Login

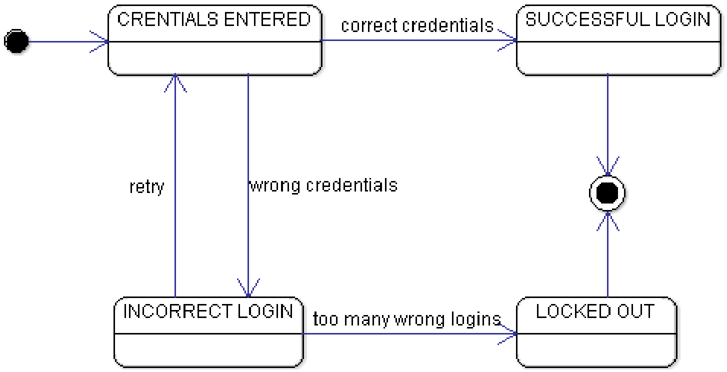

Login with UserID and password
Check UserID and password
Invalid UserID and password
Login to the system successfully

Payment

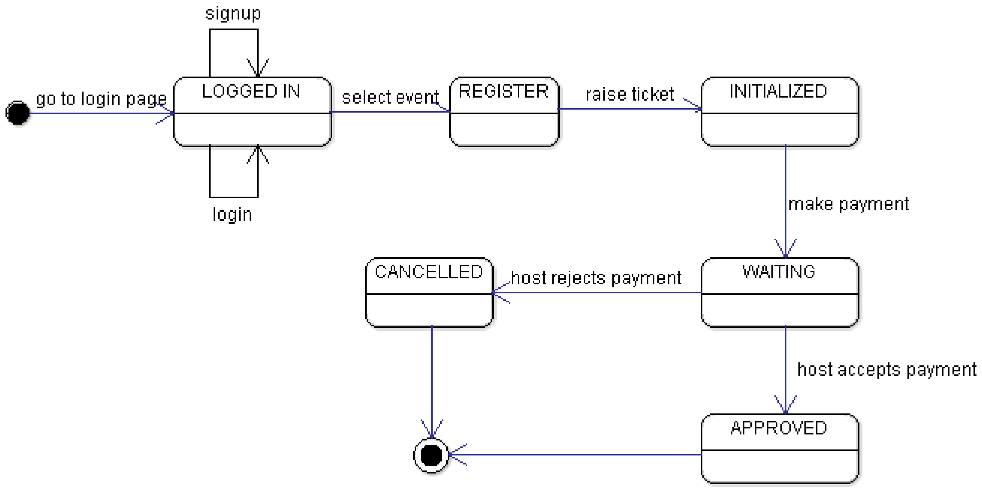


17.STATE MACHINE DIAGRAM

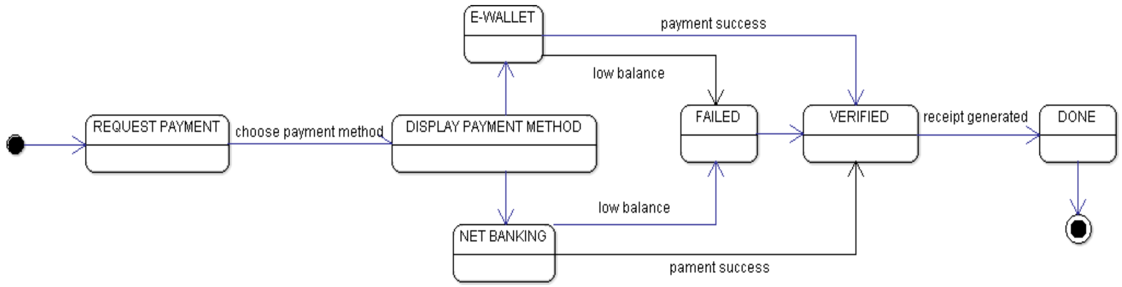
LOGIN



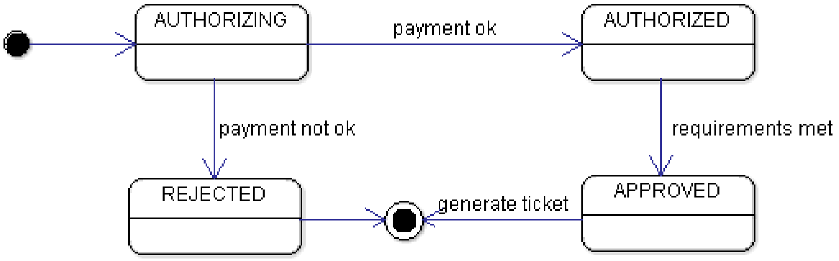
TICKET



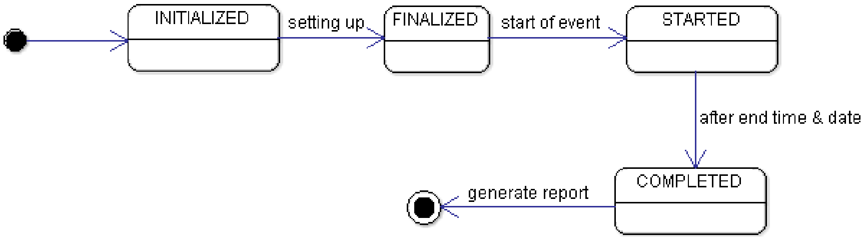
PAYMENT



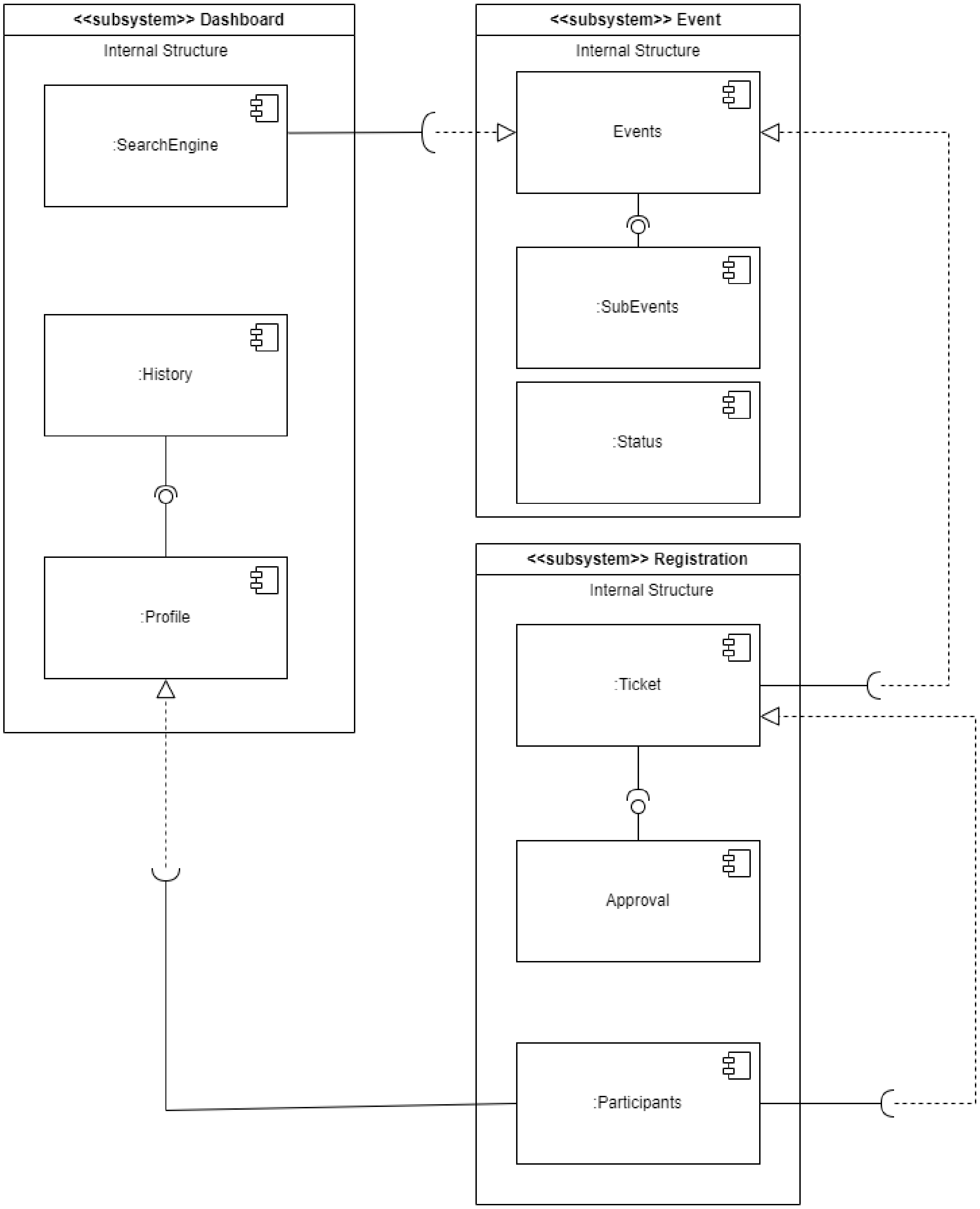
PAYMENT AUTHORIZATION



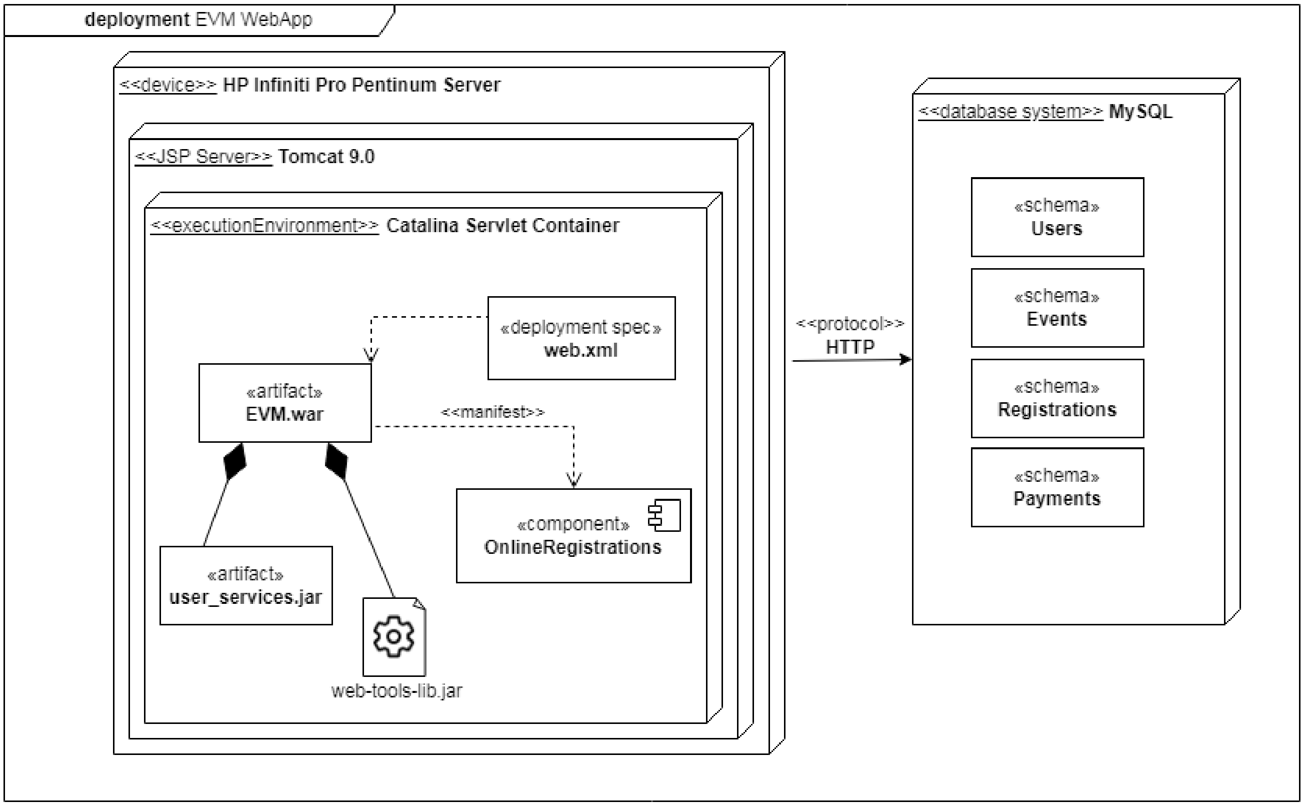
EVENTS/SUB-EVENTS



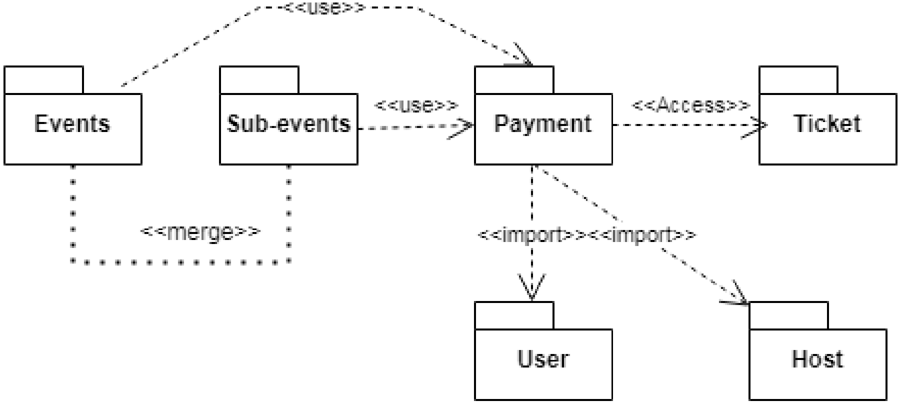
18.Component Diagram



19.Deployment Diagram

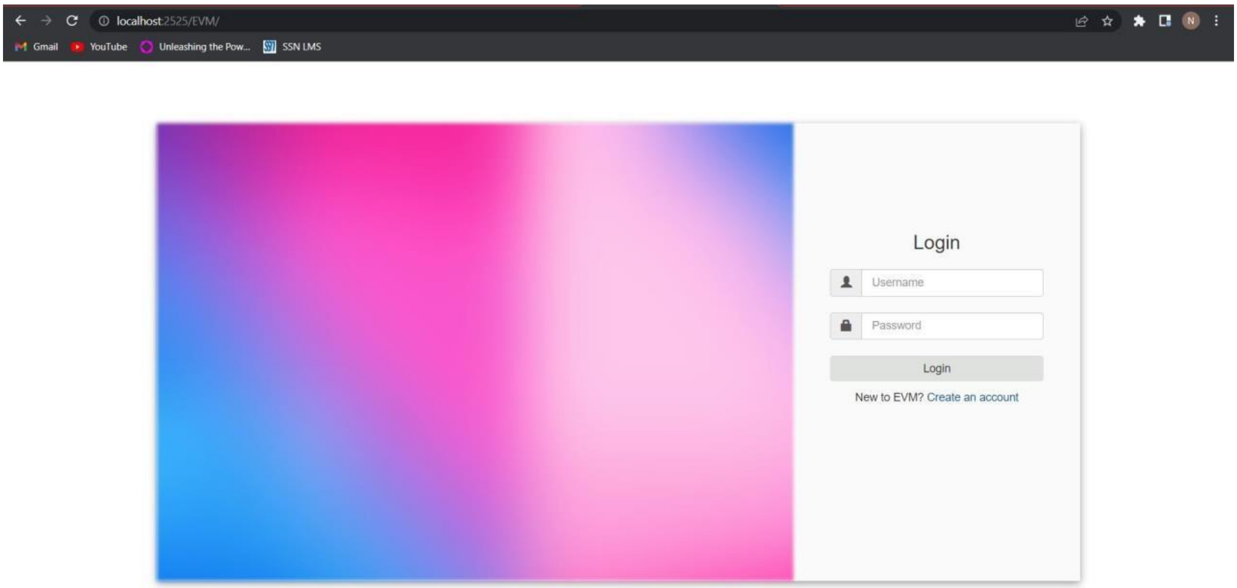


20.Package Diagram

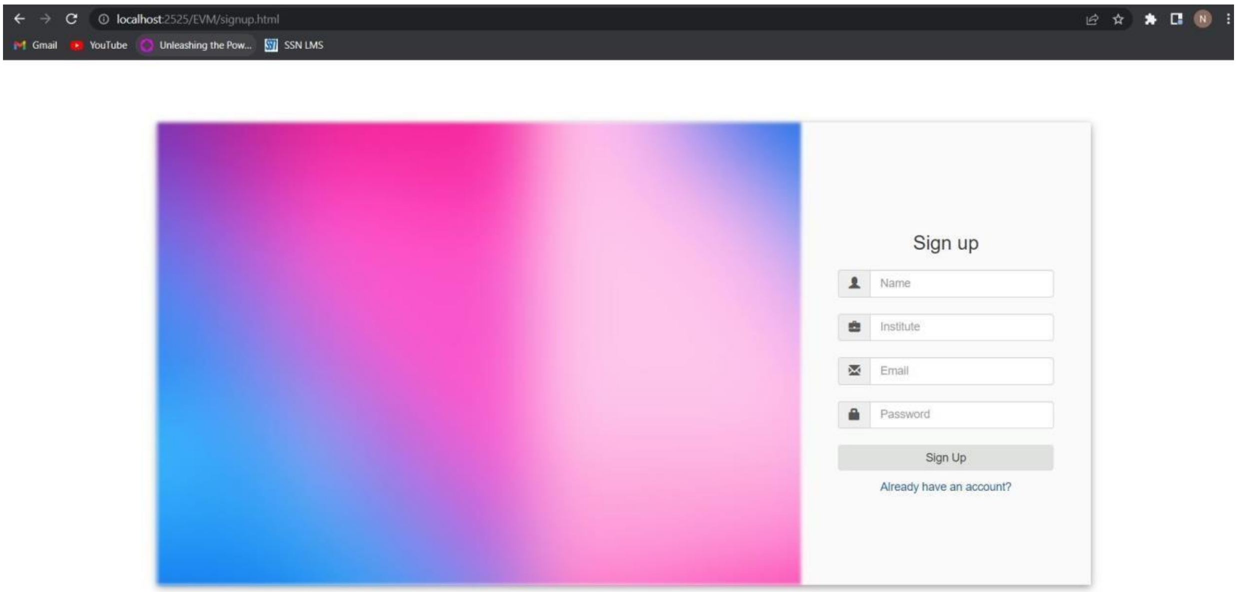


21.Front-end view of the Event Management System:

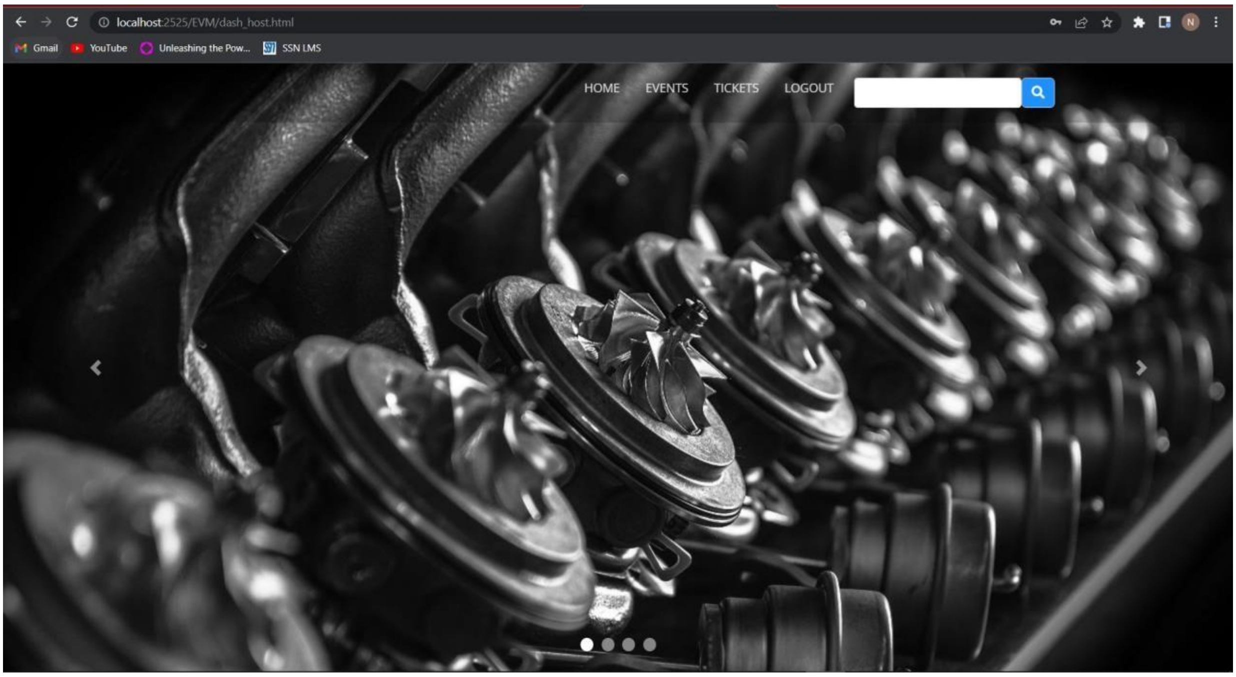
Login:



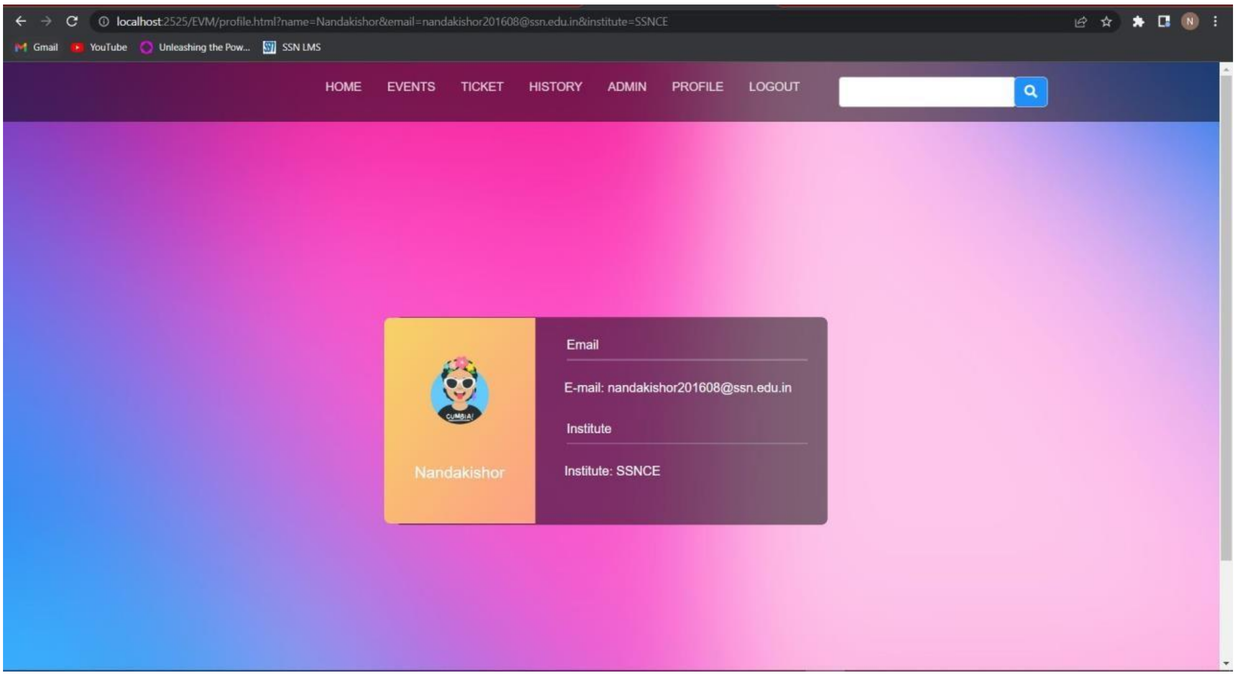
Sign-Up:



Host Dashboard:



Profile:


localhost:2525/EVM/profile.html?name=Nandakishor&email=nandakishor201608@ssn.edu.in&institute=SSNCE
Gmail
YouTube
Unleashing the Pow...
SSN LMS
HOME
EVENTS
TICKET
HISTORY
ADMIN
PROFILE
LOGOUT

Nandakishor

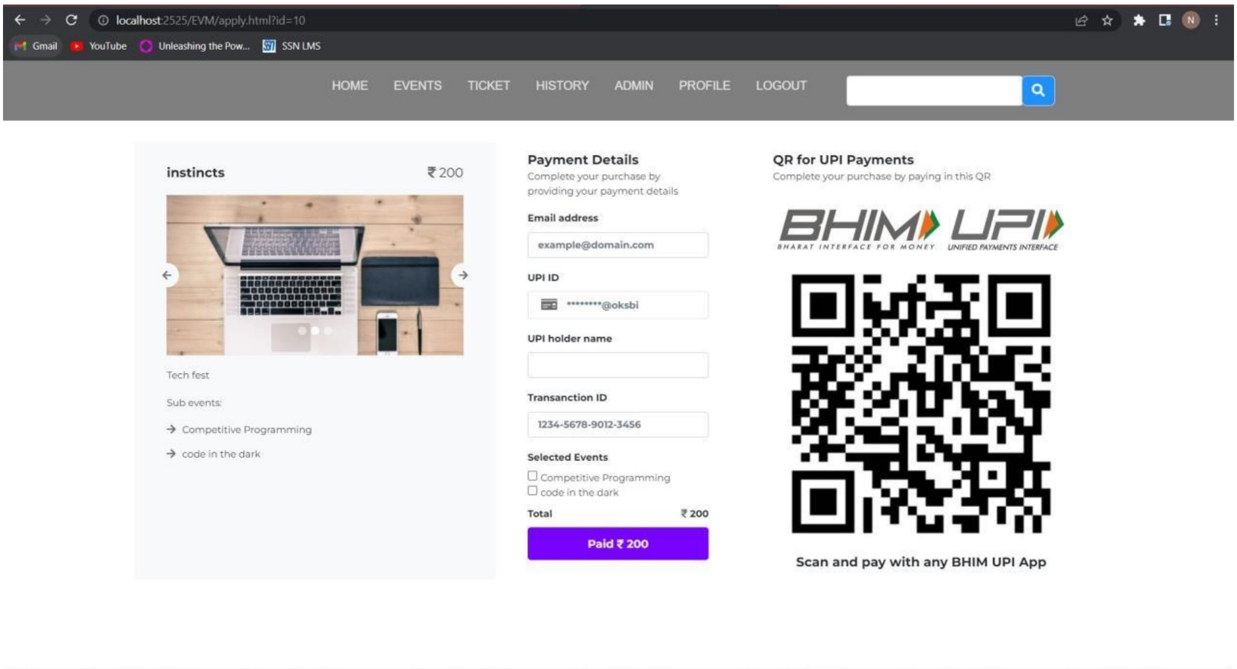
Email

E-mail: nandakishor201608@ssn.edu.in

Institute

Institute: SSNCE

Payment:



localhost:2525/EVM/apply.html?id=10

instincts ₹ 200

Tech fest

Sub events:

→ Competitive Programming

→ code in the dark

Payment Details

Complete your purchase by providing your payment details

Email address: example@domain.com

UPI ID: ********@oksbi

UPI holder name:

Transaction ID: 1234-5678-9012-3456

Selected Events

□ Competitive Programming

□ code in the dark

Total ₹ 200

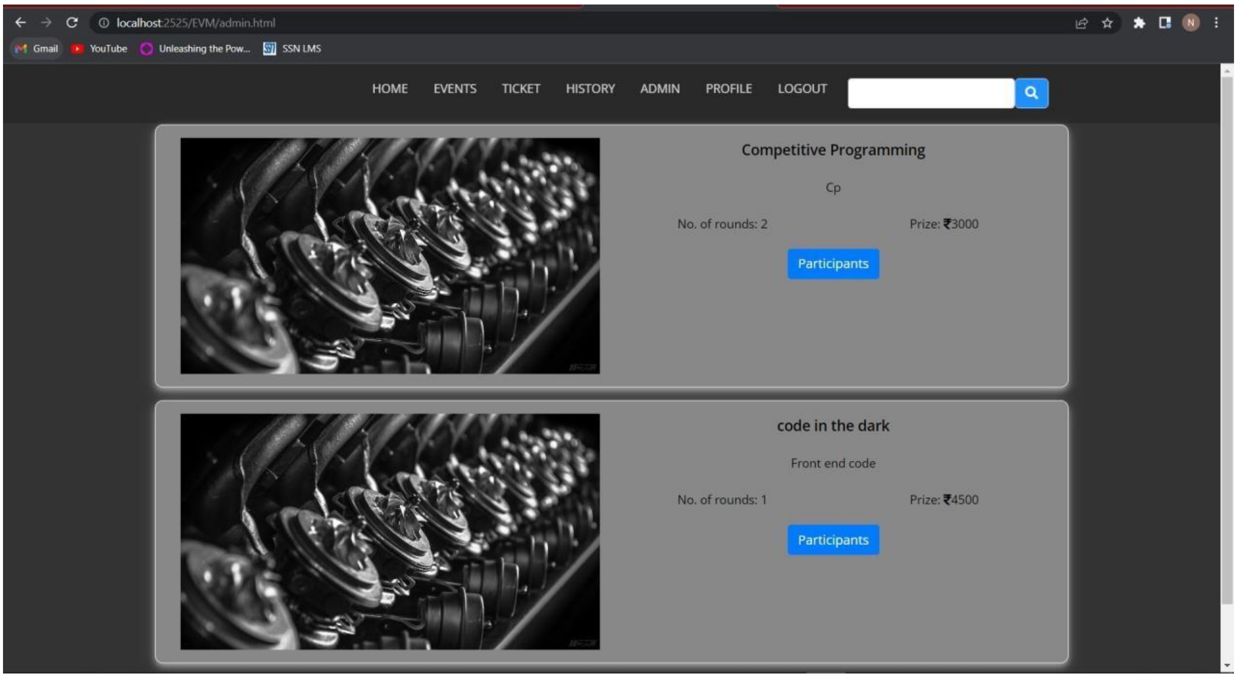
Paid ₹ 200

QR for UPI Payments

Complete your purchase by paying in this QR

Scan and pay with any BHIM UPI App

Admins:



localhost:2525/EVM/admin.html

Competitive Programming

Cp

No. of rounds: 2 Prize: ₹3000

Participants

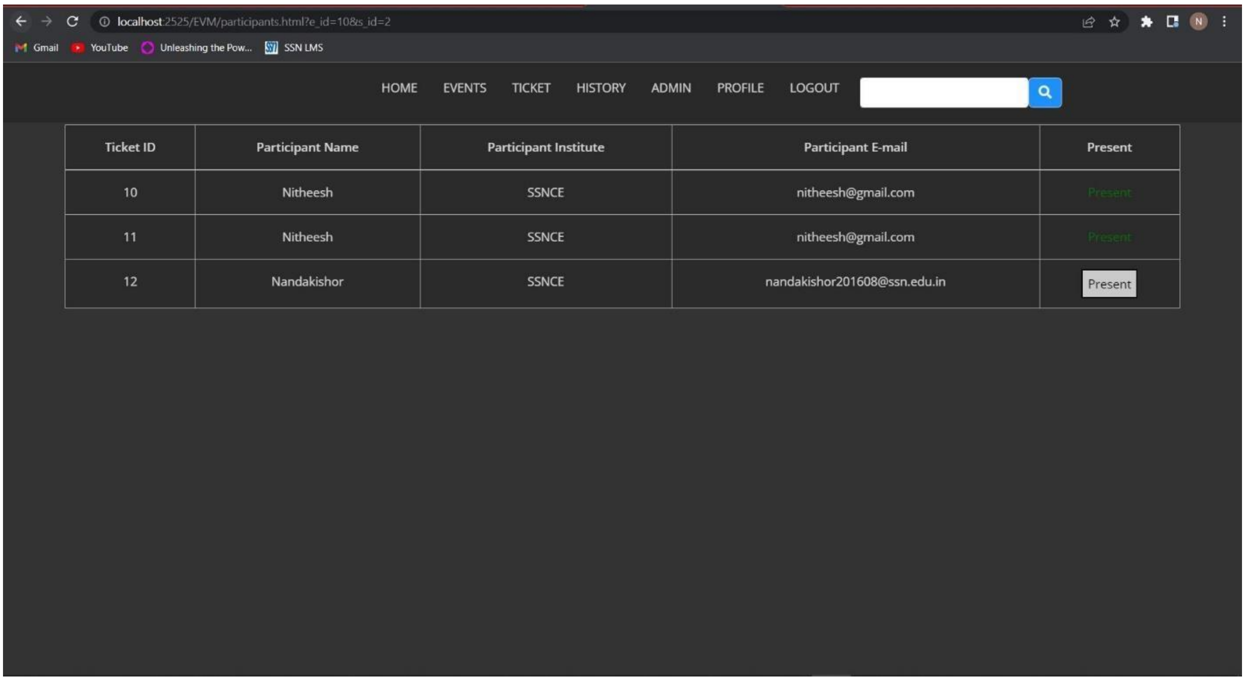
code in the dark

Front end code

No. of rounds: 1 Prize: ₹4500

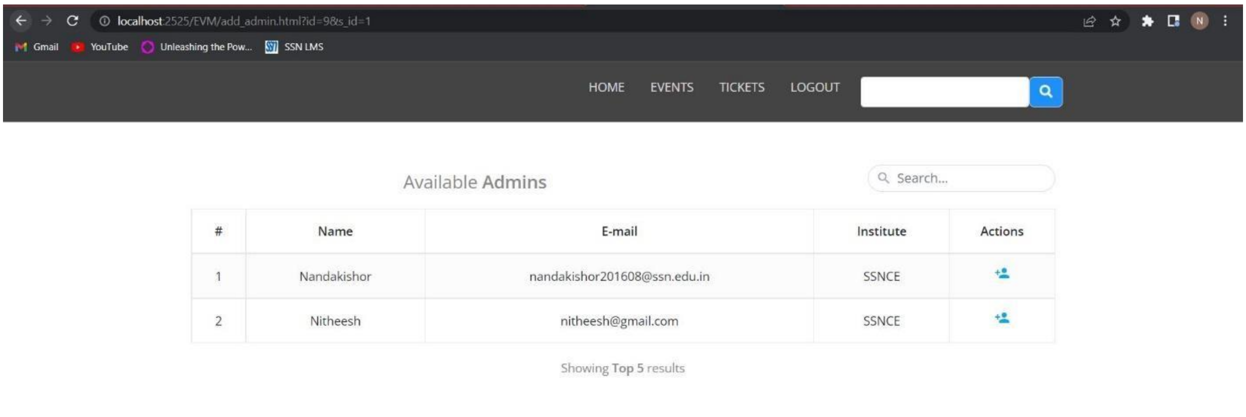
Participants

Participants:



Ticket ID	Participant Name	Participant Institute	Participant E-mail	Present
10	Nitheesh	SSNCE	nitheesh@gmail.com	Present
11	Nitheesh	SSNCE	nitheesh@gmail.com	Present
12	Nandakishor	SSNCE	nandakishor201608@ssn.edu.in	Present

Add Admin:

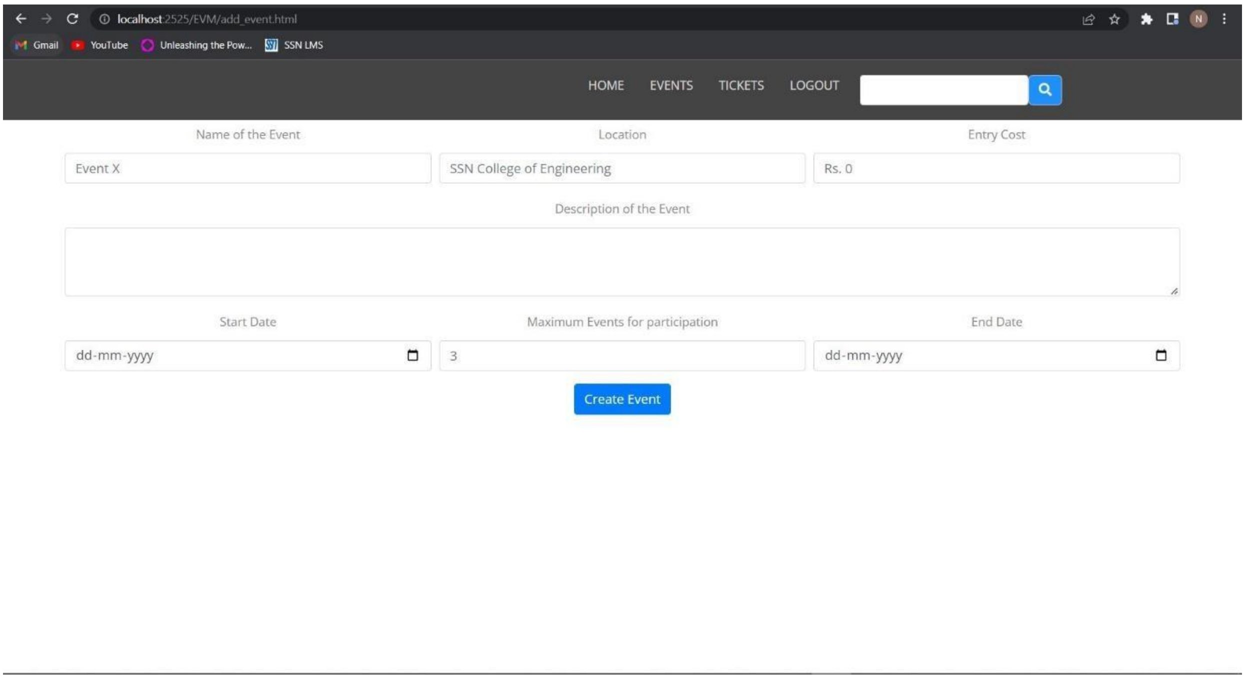


Available Admins

#	Name	E-mail	Institute	Actions
1	Nandakishor	nandakishor201608@ssn.edu.in	SSNCE	+👤
2	Nitheesh	nitheesh@gmail.com	SSNCE	+👤

Showing Top 5 results

Add Event:



Name of the Event: Event X

Location: SSN College of Engineering

Entry Cost: Rs. 0

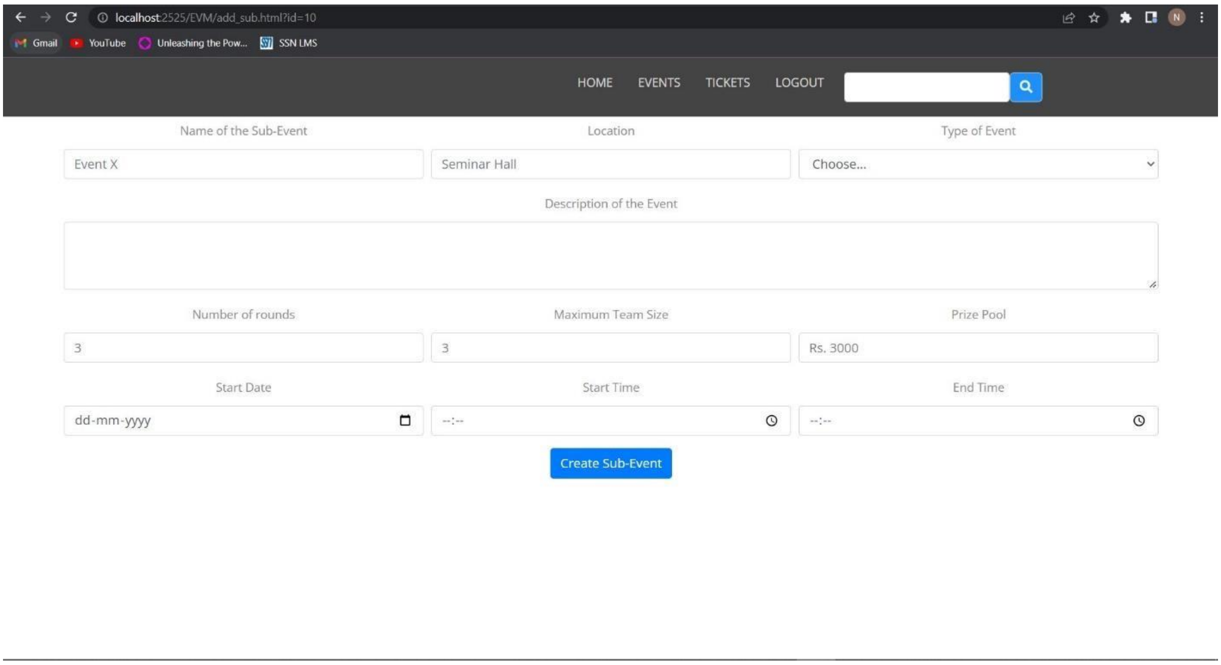
Description of the Event:

Start Date: dd-mm-yyyy

Maximum Events for participation: 3

End Date: dd-mm-yyyy

Add Sub-event:



Name of the Sub-Event: Event X

Location: Seminar Hall

Type of Event: Choose...

Description of the Event:

Number of rounds: 3

Maximum Team Size: 3

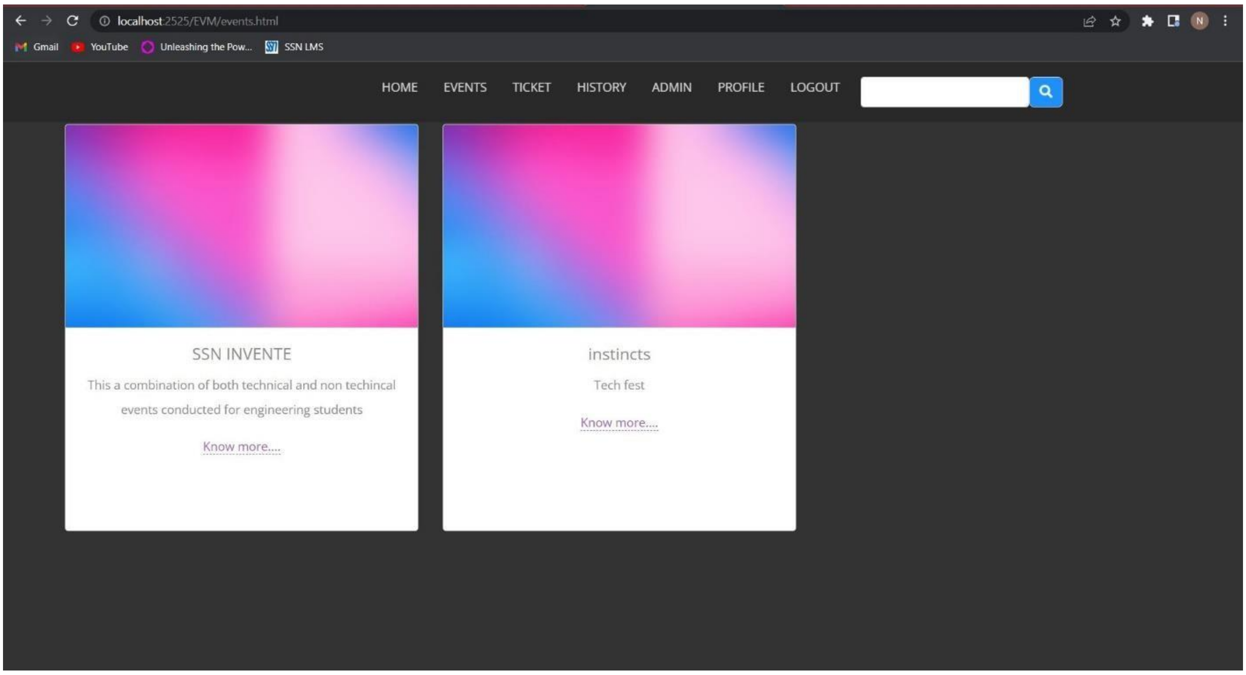
Prize Pool: Rs. 3000

Start Date: dd-mm-yyyy

Start Time: --:--

End Time: --:--

Events:



SSN INVENTE

This a combination of both technical and non techincal events conducted for engineering students

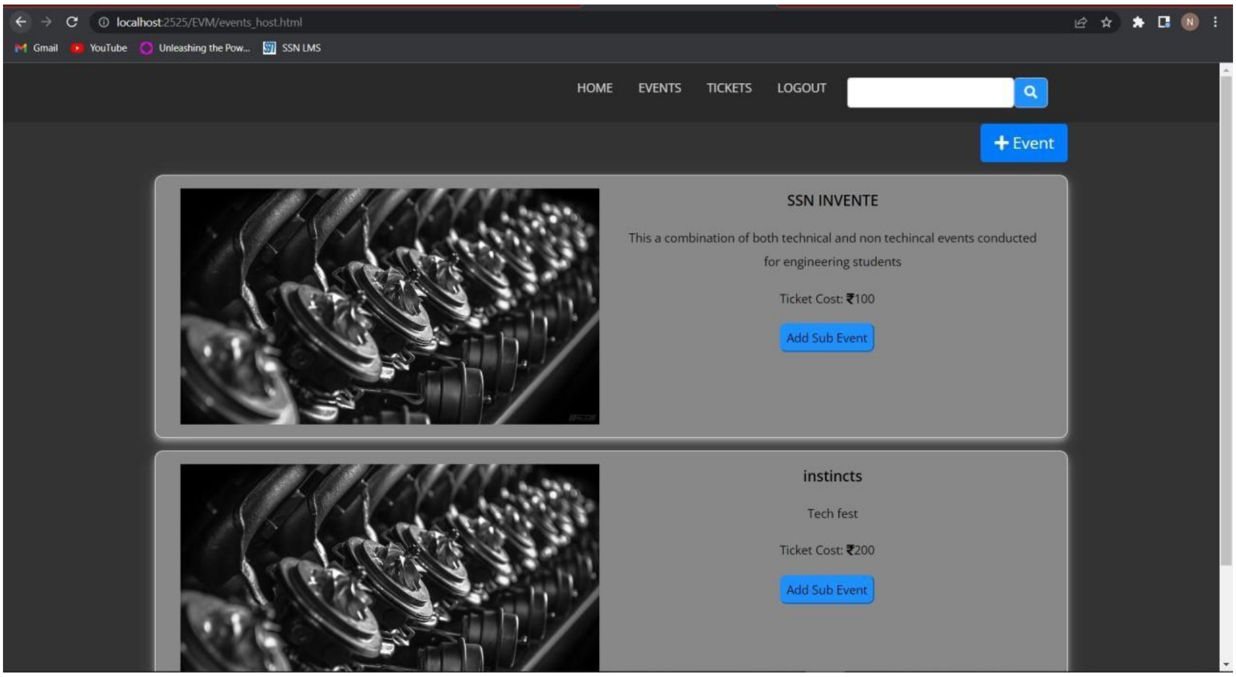
Know more....

instincts

Tech fest

Know more....

Event Host:



SSN INVENTE

This a combination of both technical and non techincal events conducted for engineering students

Ticket Cost: ₹100

Add Sub Event

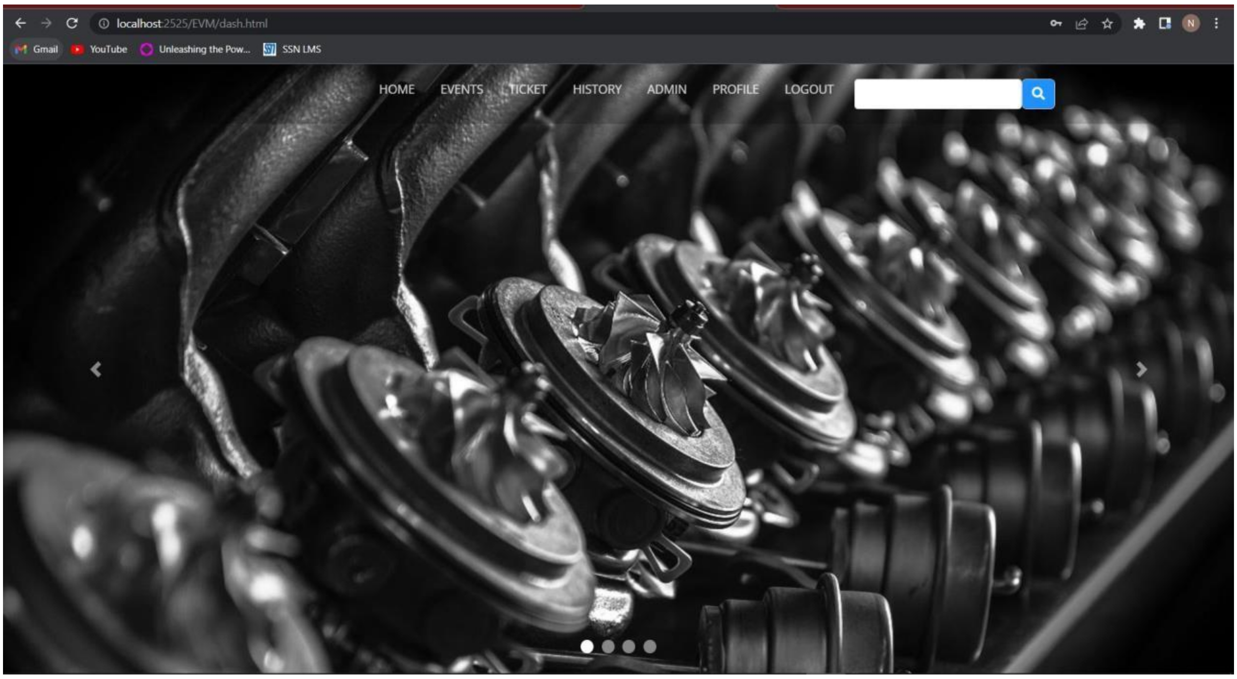
instincts

Tech fest

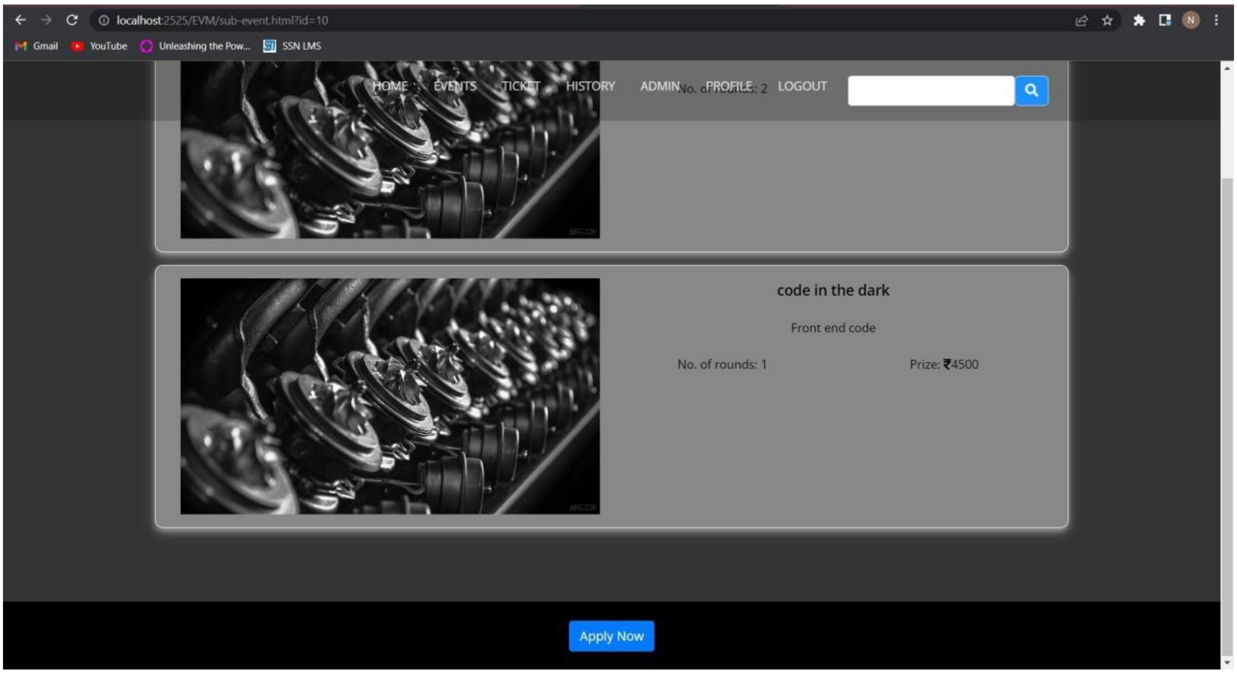
Ticket Cost: ₹200

Add Sub Event

User Dashboard:



Sub-event:



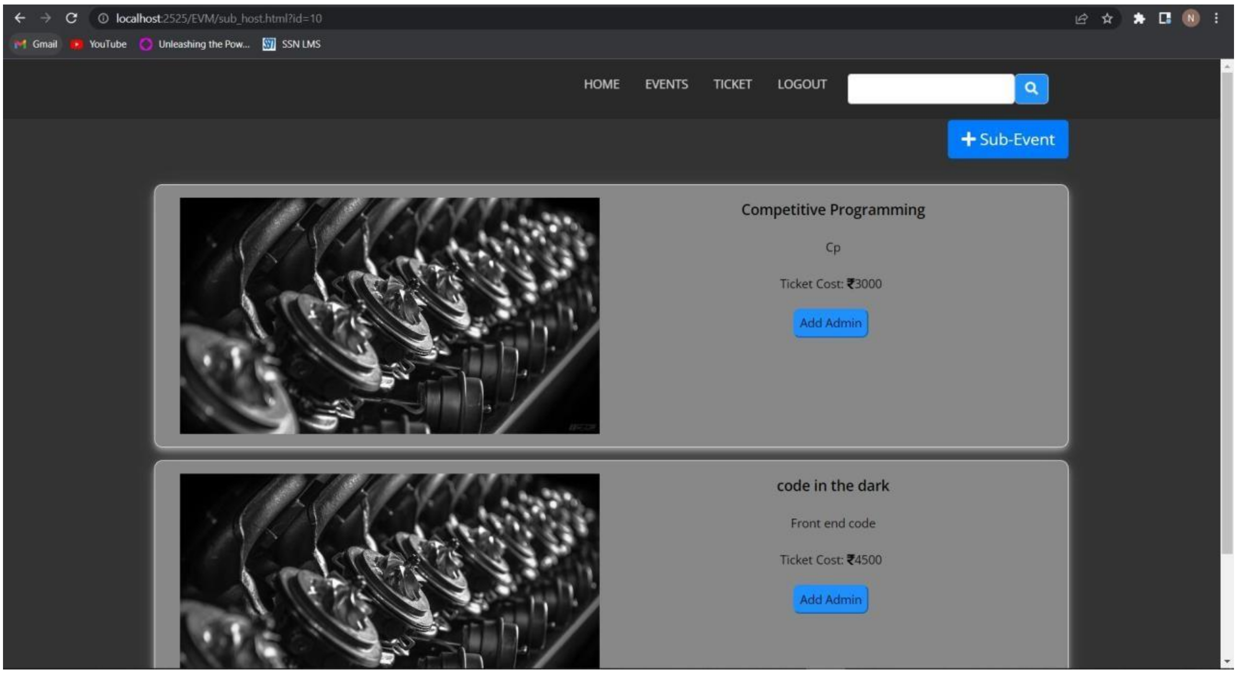
code in the dark

Front end code

No. of rounds: 1

Prize: ₹4500

Sub-event Host:



Competitive Programming

Cp

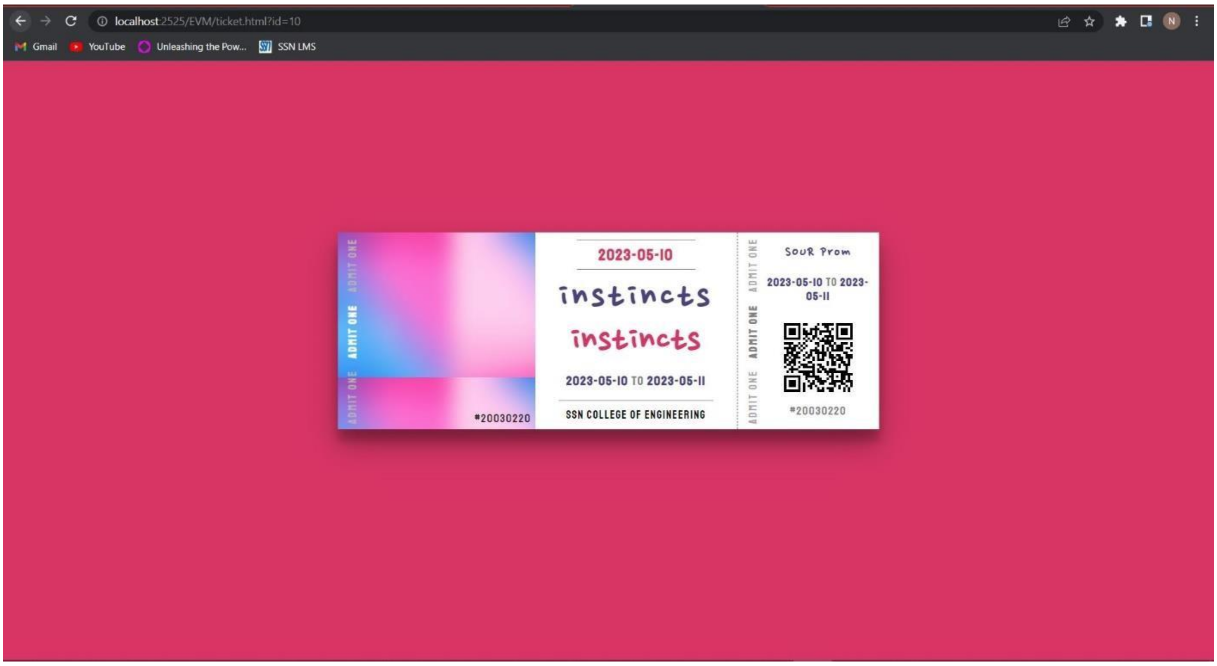
Ticket Cost: ₹3000

code in the dark

Front end code

Ticket Cost: ₹4500

Ticket:



2023-05-10

instincts

instincts

2023-05-10 TO 2023-05-11

SSN COLLEGE OF ENGINEERING

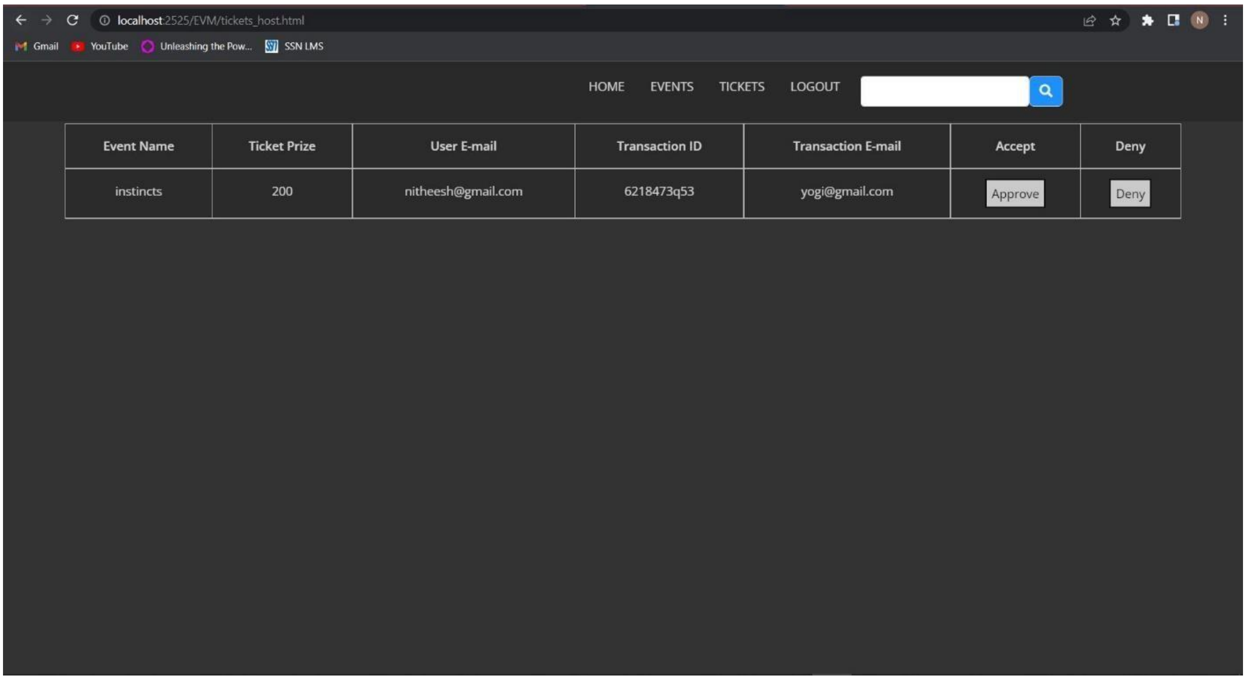
#20030220

SouR Prom

2023-05-10 TO 2023-05-11

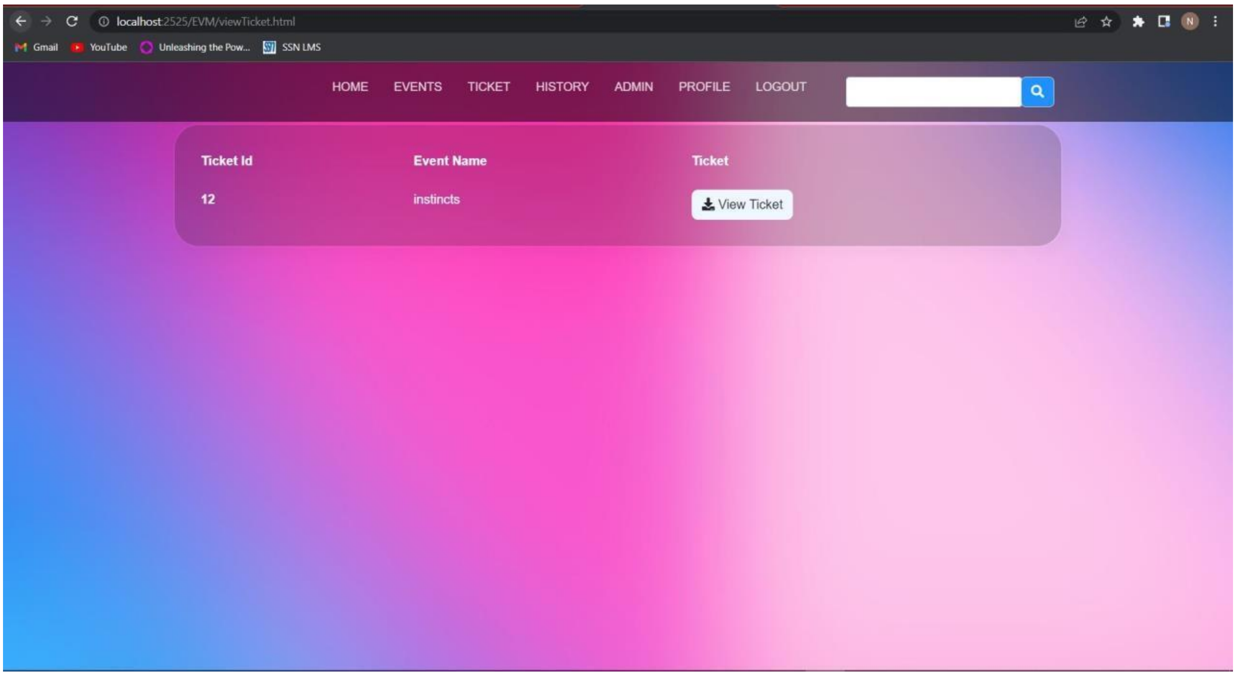
#20030220

Approve Ticket:



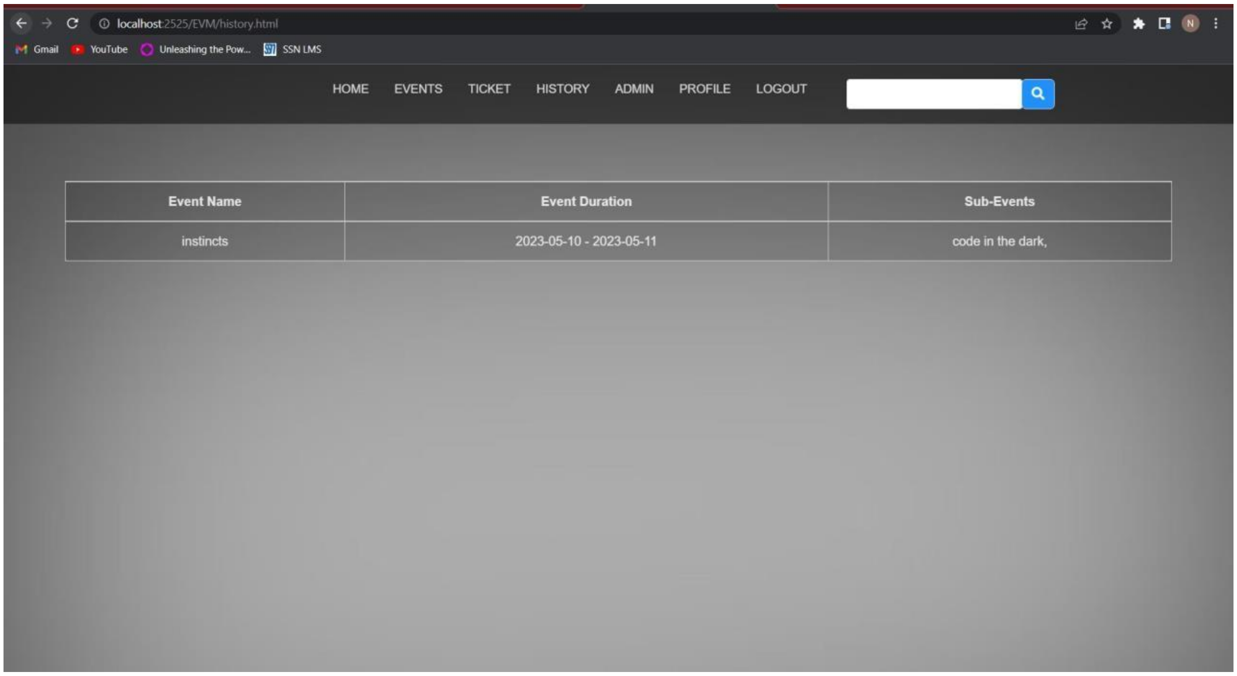
Event Name	Ticket Prize	User E-mail	Transaction ID	Transaction E-mail	Accept	Deny
instincts	200	nitheesh@gmail.com	6218473q53	yogi@gmail.com	Approve	Deny

View Ticket:



Ticket Id	Event Name	Ticket
12	instincts	View Ticket

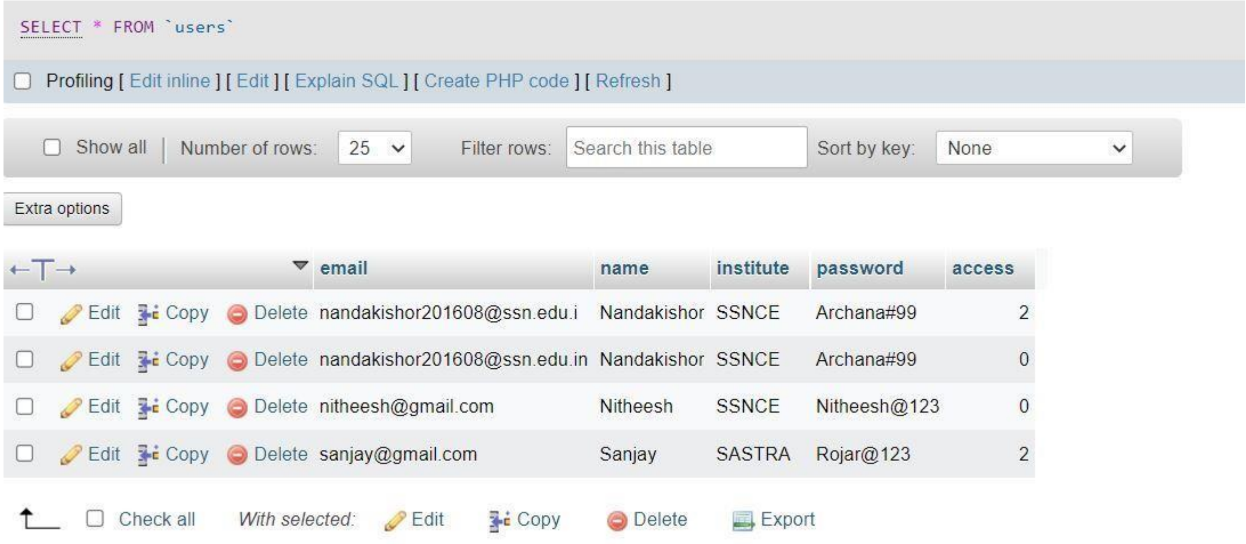
History:



Event Name	Event Duration	Sub-Events
instincts	2023-05-10 - 2023-05-11	code in the dark,

22.Maintained Databases

Users Table:



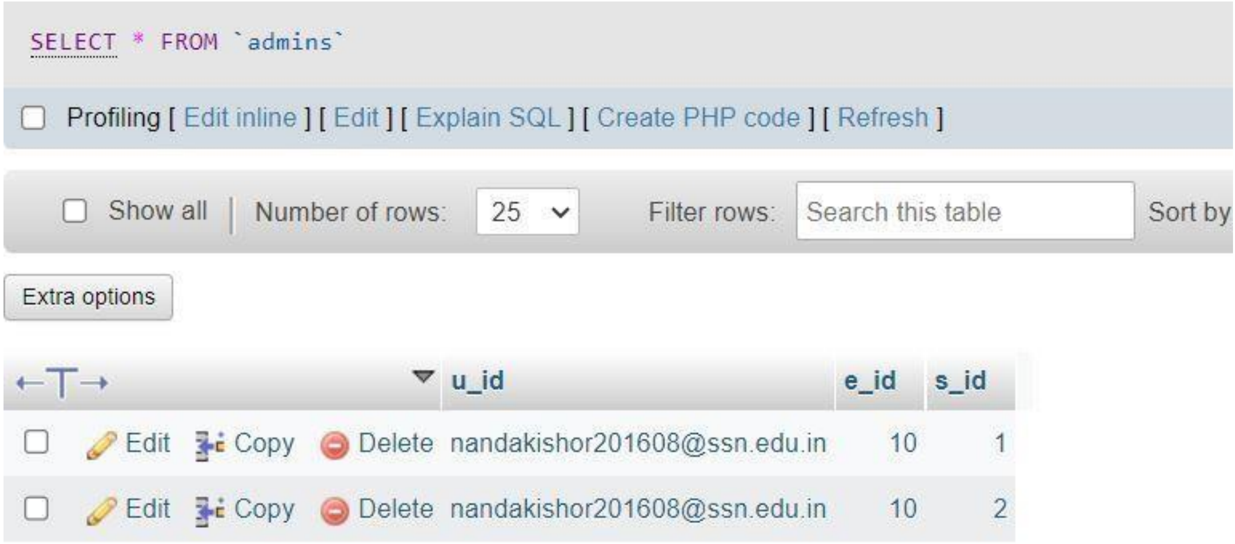
```
SELECT * FROM `users`
```

□ Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]
□ Show all
Number of rows: 25
Filter rows: Search this table
Sort by key: None
Extra options

			email	name	institute	password	access
□	Edit	Copy	Delete	nandakishor201608@ssn.edu.i	Nandakishor	SSNCE	Archana#99	2
□	Edit	Copy	Delete	nandakishor201608@ssn.edu.in	Nandakishor	SSNCE	Archana#99	0
□	Edit	Copy	Delete	nitheesh@gmail.com	Nitheesh	SSNCE	Nitheesh@123	0
□	Edit	Copy	Delete	sanjay@gmail.com	Sanjay	SASTRA	Rojar@123	2

□ Check all
With selected:
Edit
Copy
Delete
Export

Admins Table:

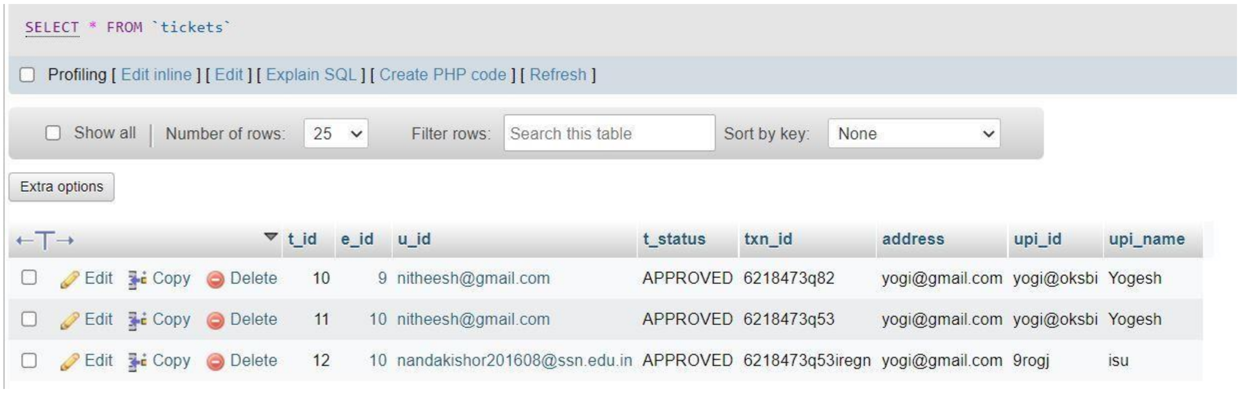


```
SELECT * FROM `admins`
```

□ Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]
□ Show all | Number of rows: 25 Filter rows: Search this table Sort by
Extra options

	←T→	u_id	e_id	s_id
□	Edit Copy Delete	nandakishor201608@ssn.edu.in	10	1
□	Edit Copy Delete	nandakishor201608@ssn.edu.in	10	2

Tickets Table:

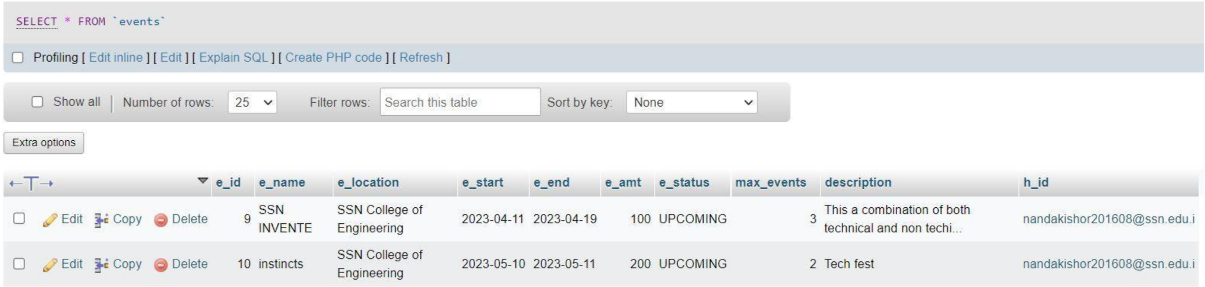


```
SELECT * FROM `tickets`
```

□ Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]
□ Show all | Number of rows: 25 Filter rows: Search this table Sort by key: None
Extra options

	←T→	t_id	e_id	u_id	t_status	txn_id	address	upi_id	upi_name
□	Edit Copy Delete	10	9	nitheesh@gmail.com	APPROVED	6218473q82	yogi@gmail.com	yogi@oksbi	Yogesh
□	Edit Copy Delete	11	10	nitheesh@gmail.com	APPROVED	6218473q53	yogi@gmail.com	yogi@oksbi	Yogesh
□	Edit Copy Delete	12	10	nandakishor201608@ssn.edu.in	APPROVED	6218473q53iregn	yogi@gmail.com	9rogj	isu

Events Table:

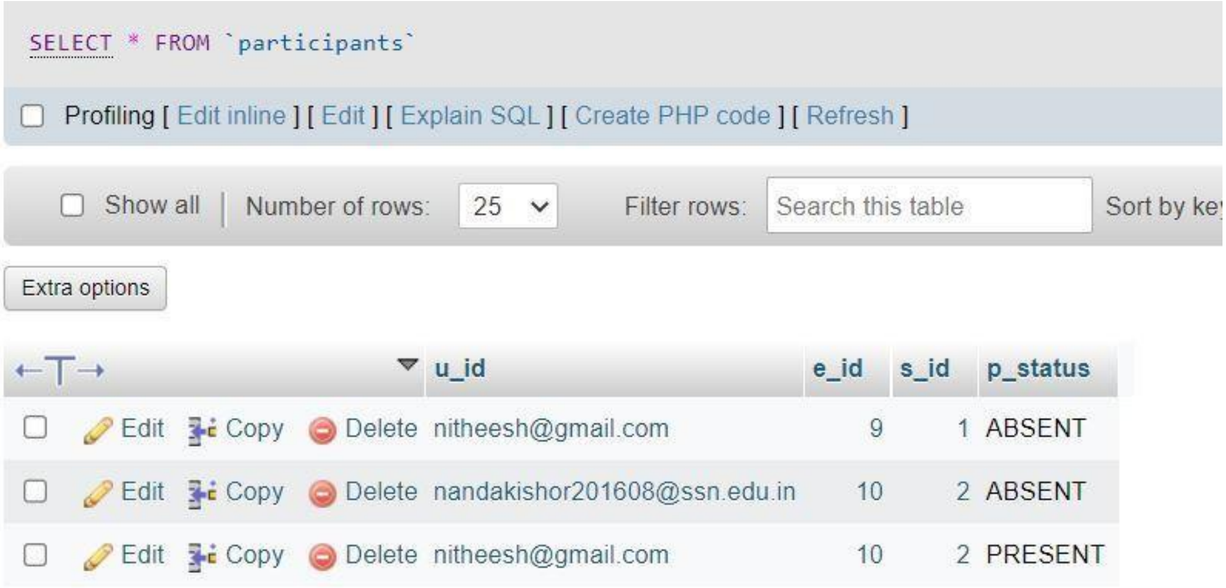


```
SELECT * FROM `events`
```

□ Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]
□ Show all | Number of rows: 25 Filter rows: Search this table Sort by key: None
Extra options

	←T→	e_id	e_name	e_location	e_start	e_end	e_amt	e_status	max_events	description	h_id
□	Edit Copy Delete	9	SSN INVENTE	SSN College of Engineering	2023-04-11	2023-04-19	100	UPCOMING	3	This a combination of both technical and non techi...	nandakishor201608@ssn.edu.i
□	Edit Copy Delete	10	instincts	SSN College of Engineering	2023-05-10	2023-05-11	200	UPCOMING	2	Tech fest	nandakishor201608@ssn.edu.i

Participants Table:



```
SELECT * FROM `participants`
```

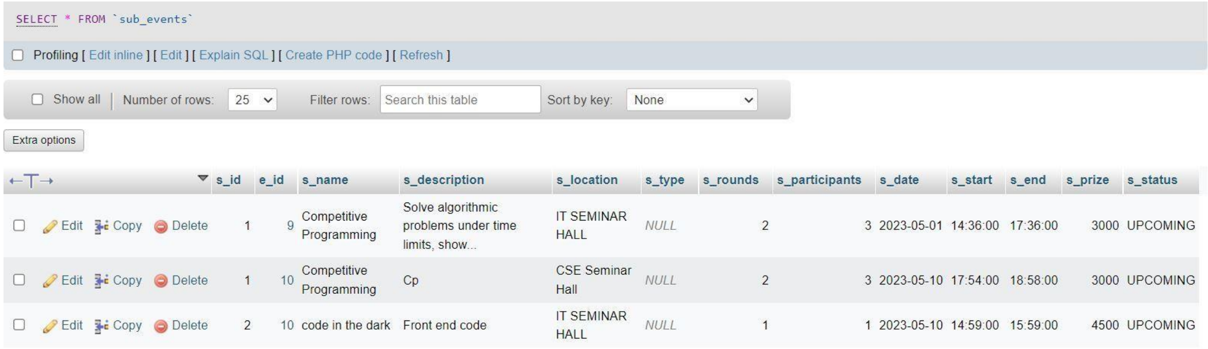
□ Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]

□ Show all | Number of rows: 25 Filter rows: Search this table Sort by ke

Extra options

←T→	u_id	e_id	s_id	p_status
□ Edit Copy Delete	nitheesh@gmail.com	9	1	ABSENT
□ Edit Copy Delete	nandakishor201608@ssn.edu.in	10	2	ABSENT
□ Edit Copy Delete	nitheesh@gmail.com	10	2	PRESENT

Sub-events Table:



```
SELECT * FROM `sub_events`
```

□ Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]

□ Show all | Number of rows: 25 Filter rows: Search this table Sort by key: None

Extra options

←T→	s_id	e_id	s_name	s_description	s_location	s_type	s_rounds	s_participants	s_date	s_start	s_end	s_prize	s_status
□ Edit Copy Delete	1	9	Competitive Programming	Solve algorithmic problems under time limits, show...	IT SEMINAR HALL	NULL	2	3	2023-05-01	14:36:00	17:36:00	3000	UPCOMING
□ Edit Copy Delete	1	10	Competitive Programming	Cp	CSE Seminar Hall	NULL	2	3	2023-05-10	17:54:00	18:58:00	3000	UPCOMING
□ Edit Copy Delete	2	10	code in the dark	Front end code	IT SEMINAR HALL	NULL	1	1	2023-05-10	14:59:00	15:59:00	4500	UPCOMING