

Secure Meeting Platform

Submitted by

KHAYATI SHARMA [RA2111003010710]

Under the Guidance of

Dr. Bibin Christopher V

Assistant Professor, Computing Technologies Department

In partial satisfaction of the requirements for the degree of

**BACHELOR OF TECHNOLOGY
in
COMPUTER SCIENCE ENGINEERING**

with specialization in CSE CORE



SCHOOL OF COMPUTING

**COLLEGE OF ENGINEERING AND TECHNOLOGY
SRM INSTITUTE OF SCIENCE AND TECHNOLOGY**

KATTANKULATHUR - 603203

MAY 2023



COLLEGE OF ENGINEERING & TECHNOLOGY
SRM INSTITUTE OF SCIENCE AND TECHNOLOGY
S.R.M. NAGAR, KATTANKULATHUR - 603 203
Chengalpattu District

BONAFIDE CERTIFICATE

Register No. RA2111003010710 Certified to be the bonafide work done by
KHAYATI SHARMA of II Year/IV Semester B.Tech Degree Course in the
Practical Software Engineering and Project Management 18CSC206J in **SRM**
INSTITUTE OF SCIENCE AND TECHNOLOGY, Kattankulathur during the
academic year 2022 – 2023.

SIGNATURE

Faculty In – Charge
Dr. Bibin Christopher V
Assistant Professor
Department of Computing Technologies
SRM Institute of Science & Technology

SIGNATURE

HEAD OF THE DEPARTMENT
Dr. M. Pushpalatha
Professor & Head
Department of Computing Technologies
SRM Institute of Science & Technology

ABSTRACT

During the ongoing global pandemic, faculty, staff and administrators at colleges and universities experienced an increase in meetings using web-based platforms. Challenges were identified related to the changes from face-to-face to web-based meetings, including internet connectivity, inadequate technology. Few meeting Platforms like Gmeet, Zoom, etc, were very popular. Current meeting platform lacks security. We have multiple options to join instances or to schedule the meeting or through the code. Our project aims at developing a Secure Meeting Platform that will track each meeting attendee's IP address and have the ability to block suspicious users before they join the meeting if a VPN or proxy is detected. There will be an automatic Proxy and VPN detector so that the intruder is not able to forward his/her traffic through the VPN server and join the meet after it. Our system will block this type of traffic and will make the meet secure. We need to record each user's IP address for tracking purposes, which may limit the platform's audience and automatically block the person who tries to do malpractices and send the alert to the host of the meeting. We also have the AI Chatbot which blocks unwanted messages or spamming in the chat box. In our project we discussed the prioritization of Stakeholders, Work Breakdown Structure, SWOT Analysis, Risk Management, System Architecture, Gantt chart and many Diagrams like- Use Case Diagrams, Class Diagrams, ER Diagrams, Sequence Diagram, Collaboration Diagram and we also made the framework for our website for better understanding of our project.

TABLE OF CONTENTS

CHAPTER NO	TITLE	PAGE NO
	ABSTRACT	
	LIST OF FIGURES	
	LIST OF TABLES	
	LIST OF ABBREVIATIONS	
1	PROBLEM STATEMENT	1
2	STAKEHOLDERS & PROCESS MODELS	4
3	IDENTIFYING REQUIREMENTS	6
4	PROJECT PLAN & EFFORT	7
5	WORK BREAKDOWN STRUCTURE & RISK ANALYSIS	12
6	SYSTEM ARCHITECTURE, USE CASE & CLASS DIAGRAM	15
7	ENTITY RELATIONSHIP DIAGRAM	18
8	DATA FLOW DIAGRAM	19
9	SEQUENCE & COLLABORATION DIAGRAM	21
10	DEVELOPMENT OF TESTING FRAMEWORK/USER INTERFACE	23
11	TEST CASES	25
12	REPORTING	27
13	ARCHITECTURE/DESIGN/FRAMEWORK/IMPLE -MENTATION	28
	CONCLUSION	
	REFERENCES	
	APPENDIX	

LIST OF FIGURES

FIGURE NO	TITLE	PAGE NO
2.1	Selection of Methodology	4
5.1	Work Breakdown Structure	12
5.2	Gantt Chart	13
6.1	System Architecture	15
6.2	Use Case Diagram	16
6.3	Class Diagram	17
7.1	ER Diagram	18
8.1	DFD Level 0	19
8.2	DFD Level 1	20
9.1	Collaboration Diagram	21
9.2	Sequence Diagram	22
13.1	Framework of the Website	28
13.2	System Architecture of the Website	29

LIST OF TABLES

TABLE NO	TITLE	PAGE NO
1.1	Stakeholder information	5
4.1	Project management plan	8
4.2	Effort and cost estimation	9
4.3	Infrastructure / resource cost	9
4.4	Maintenance and support cost	10
4.5	Identification of team members	10
4.6	Responsibility assignment matrix	11
5.1	Risk analysis (SWOT)	13
5.2	Risk analysis (RMMM)	14
10.1	Types of testing, methodology, tools	24
11.1	Functional test cases	25
11.2	Non-functional test cases	26
12.1	Manual test cases	27

LIST OF ABBREVIATIONS

CAPEX	Capital Expenditure
OPEX	Operating Expenditure
E1R1A1T1	Effort Requirement Activity Task
UX	User Experience
UAT	User Acceptance Testing
IR	Infrastructure Requirement
API	Application Development Interface
WBS	Work Breakdown Structure
SWOT	STrength,Weakness,Opportunity,Threats
RMMM	Remote Monitoring and Management
GANTT	Generalized Activity Normalization Time Table
DFD	Data Flow Diagram
ER	Entity Relationship
ISA	Specialization-Generalization
DB	Data Base

EX. 1

PROBLEM STATEMENT/ BUSINESS CASE

Project Title: Vcon-Online Secured meeting platform

Problem Statement:

Developing a Secured Meeting Platform-'Vcon' that will track each meeting attendee's IP address and have the ability to block suspicious users before they join the meeting if a VPN or proxy is detected. Machine learning algorithms will be used to analyze the pattern of suspicious activities in order to learn and generalize blocking rules.

The main goal is to overcome the identified problems listed below and propose innovative solutions.

- Solutions need to be devised so that the intruders/miscreants are identified.
- Miscreants would not be able to use IDs and Names of the identified participants/Students.
- It should get easy for the Host to block such intruders which do not usually happen.
- Messages(which are usually disturbing and offensive) need to be blocked in such a way that they are not displayed during the entire session.

A business case template is a document that outlines the justification for a proposed business project or initiative. It is used to present a compelling argument for why the project is necessary and how it will benefit the organization. It is a crucial tool for organizations to evaluate the feasibility and potential value of a proposed project before committing resources and investments.

ONE PAGE BUSINESS CASE TEMPLATE

	DATE	20.1.23
	SUBMITTED BY	Khayati Sharma
	TITLE / ROLE	Vcon-Secure Meeting Platform



THE PROJECT

In bullet points, describe the problem this project aims to solve or the opportunity it aims to develop.

1. The pandemic has caused a considerable increase in online meeting platforms for education and other official purposes.
2. As a result there have been cases of offenders joining the meeting and disturbing the meeting environment in various ways such as by sending offensive messages.
3. These misdeeds might disturb the purpose of the whole meeting and the decorum of the academic activities/sessions get spoiled.

THE HISTORY

In bullet points, describe the current situation.

1. Current Meeting platforms lack security, people join from different ids and commit fraud.
2. Unnecessary chatbox spam is common.

LIMITATIONS

List what could prevent the success of the project, such as the need for expensive equipment, bad weather, lack of special training, etc.

1. VPNs and proxy servers that are well-known will be easily detected and blocked, but those that aren't will be difficult to detect and block.
2. We need to record each user's IP address for tracking purposes, which may limit the platform's audience.
3. We'll need a team of highly skilled individuals to carry out the plan.

APPROACH

List what is needed to complete the project.

1. Develop a Secure Meeting Platform that will trace the IP address of each and every meeting attendee and will have the option to block the suspicious users before joining the meeting in case any VPN or Proxy is identified.
2. The pattern of suspicious activities will be analyzed with the machine learning algorithms to learn and generalize the blocking,

BENEFITS

In bullet points, list the benefits that this project will bring to the organization.

1. There will be an automatic Proxy and VPN detector so that the intruder is not able to forward his/her traffic through the VPN server and join the meet after it. Our system will block this type of traffic and will make the meet secure.
2. For Educational Institutes there will be additional security functions to restrict the students from doing any malpractice.

EX.2

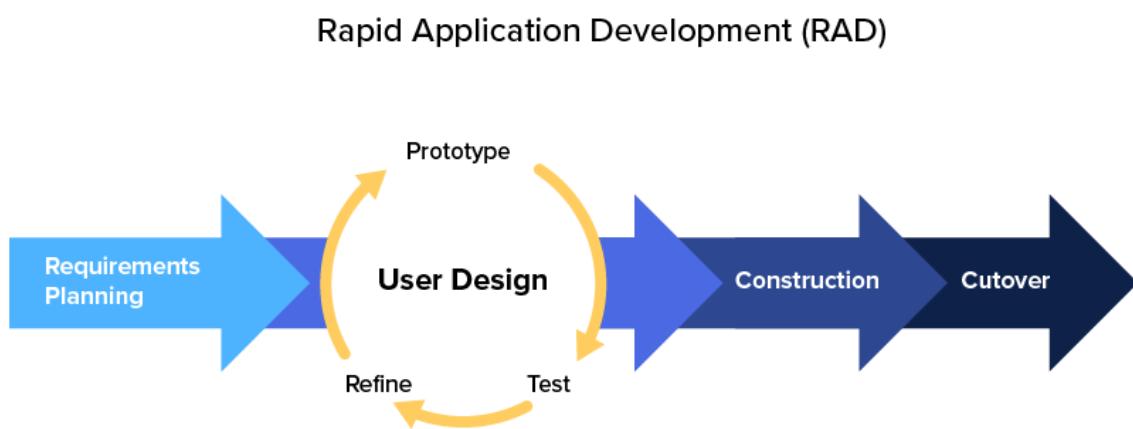
PROCESS METHODOLOGY AND STAKEHOLDER DESCRIPTION

A process model is a structured representation of a business process, software system or any other complex system that describes how the system functions, the steps it takes to accomplish its goals, and the interactions between the various components of the system.

The Rapid Application Development (RAD) model is a software development approach that emphasizes quick prototyping and iterative development. It was developed in response to the limitations of traditional Waterfall models that required extensive planning and documentation before any code could be written.

Selection of Methodology:- Rapid Application Development Model

- It takes less than three months.
- Each major function can be addressed by a separate team and then integrated to form a whole.
- More productivity with fewer people
- Development time is drastically reduced
- Requirements can be changed at anytime
- Encourages and prioritizes customer feedback
- Reviews are quick



Incorporate information to below table regarding stakeholders of the project.

Stakeholder Name	Activity/ Area /Phase	Interest	Influence	Priority (High/ Medium/ Low)
Owner	Achieve targets,Increase sales margin.	High	High	1
Sponsor	Provides a new market to expand ventures.Negotiate funding for projects.	Medium	Medium	3
Technical Team	Work on the project and do all the needful tech work	High	High	2
Project Manager	Lead the team in every aspect.Accountable for entire project success and failure.	Medium	Medium	2
Investors	Provides necessary financial resources according to project needs and budget.	Medium	Medium	5
Resource Manager	Ensuring adequate resources according to project needs and budget.	Medium	Medium	4
End Users	Provides Feedback.	Medium	Medium	5

EX. 3

SYSTEM/FUNCTIONAL AND NON-FUNCTIONAL REQUIREMENTS

System requirements refer to the minimum hardware and software specifications needed to run a particular software application or operating system. These requirements are typically listed by the software or operating system vendor, and they specify the minimum amount of RAM, processor speed, disk space, and other hardware specifications needed for the software to function properly.

Functional requirements refer to the features and functionality that a software system or application must have in order to meet the needs of its users. These requirements typically specify what the software must do, such as allowing users to perform certain actions or providing certain types of information.

Non-functional requirements, on the other hand, refer to characteristics of the software system or application that do not directly relate to its functionality, but are still important for ensuring the quality and usability of the software. These requirements typically specify how the software should perform, such as its reliability, performance, usability, and security.

System Requirements: 2 GB RAM and a dual-core 1Ghz processor(Recommended)

4 GB RAM and a dual-core 1Ghz processor(Optimal)

Functional Requirements: Monitoring the IP address of the people joining the meet and allowing only those IP's which are invited for the meeting and keeping the track of all the chat box messages through an AI trained model.

Non-Functional Requirements: Automatically block the person who tries to do malpractices and send the alert to the host of the meeting.

EX.4

PROJECT PLAN AND EFFORT

A project management plan (PMP) is a document that outlines the approach, methodology, and processes that will be used to plan, execute, monitor, and control a project. The PMP is typically developed by the project manager in collaboration with the project team and stakeholders, and it provides a roadmap for the successful completion of the project.

Here are the steps to calculate project effort:

1. Define the project scope: Before you can calculate project effort, you need to have a clear understanding of what the project entails. This includes identifying the project objectives, deliverables, and requirements.
2. Break down the project into tasks: Once you have defined the project scope, you need to break it down into smaller, more manageable tasks. This will help you estimate the effort required for each task.
3. Estimate the time required for each task: For each task, estimate the amount of time it will take to complete. This should include time for planning, execution, and any necessary review or approval processes.
4. Determine the resources required: In addition to time, you need to determine what resources will be required to complete each task. This might include people, equipment, or materials.
5. Multiply the time and resources: Once you have estimated the time and resources required for each task, multiply them together to calculate the effort required for that task.
6. Sum up the effort for all tasks: Add up the effort required for each task to calculate the total project effort.

Defining job roles and responsibilities is an important part of project planning and management. Clearly defining the roles and responsibilities of each team member helps to ensure that everyone understands their role in the project and what is expected of them. By clearly defining job roles and responsibilities, you can help ensure that your project team is well-organized, efficient, and focused on achieving the project objectives.

1. Project Management Plan

Focus Area	Details
Integration Management	Governance Framework Project Team Structure Roles & Responsibilities of Team Change Management (Change Control, Issue Management) Project Closure
Schedule Management	Define Milestones Schedule Control
Cost Management	Estimate Effort Assign Team Budget Control
Quality Management	Quality Assurance: Quality assurance will be managed including governance, roles and responsibilities, tools and techniques and reporting Quality Control: Specify the mechanisms to be used to measure and control the quality of the work products
Resource Management	Estimate and Manage the need People: People & Skills Required Finance: Budget Required Physical: Facilities, IT Infrastructure
Communication Management	Determine communication requirements, roles and responsibilities, tools and techniques. [Type of Communication, Schedule, Mechanism Recipient]

2. Estimation

2.1. Effort and Cost Estimation

Activity Description	Sub-Task	Sub-Task Description	Effort (in hours)	Cost in INR
Design the user screen	E1R1A1T1 (Effort-Requirement-Activity-Task)	Designing the meeting platform login portal	6	10000
	E1R1A1T2	Designing the security system	5	8000
	E1R1A1T3	Designing and Training the AI Chatbot	8	14000
	E1R1A1T4	Designing the admin or the meet host page to keep a track on everyone.	10	20000
	E1R1A1T5	Making and Integrating all the different systems in a single platform.	100	24000
Identify Data Source for displaying units of Energy Consumption		Go through Interface contract (Application Data Exchange) documents	5	10000
		Software Testing	8	16000
		Document	4	6000

Effort (hr)	Cost (INR)
1	2000

2.2 Infrastructure/Resource Cost [CapEx]

< OneTime Infra requirements >

Infrastructure Requirement	Qty	Cost per qty	Cost per item
IR1	PC's	4	80000
IR2	Hosting Server	1	5000
IR3	Wi-Fi	1	4000
IR4	AI Tool	1	70000

2.3 Maintenance and Support Cost [OpEx]

Category	Details	Qty	Cost per qty per annum	Cost per item
People	Network, System, Middleware and DB admin Developer , Support Consultant	3	2,000,000	6,000,000
License	Operating System Database Middleware IDE	10	10000	100,000
Infrastructures	Server, Storage and Network	20	20000	400,000

3. Project Team Formation

3.1. Identification Team members

Name	Role	Responsibilities
Harshika	Key Business User (Product Owner)	Provide clear business and user requirements
Harshika	Project Manager	Manage the project
Harshika	Business Analyst	Discuss and Document Requirements
Khayati	Technical Lead	Design the end-to-end architecture
Urnava	UX Designer	Design the user experience
Khayati,Urnava	Frontend Developer	Develop user interface
Khayati,Harshika	Backend Developer	Design, Develop and Unit Test Services/API/DB
Khayati,Urnava	Cloud Architect	Design the cost effective, highly available and scalable architecture
Harshika	Cloud Operations	Provision required Services
Khayati	Tester	Define Test Cases and Perform Testing

3.2. Responsibility Assignment Matrix

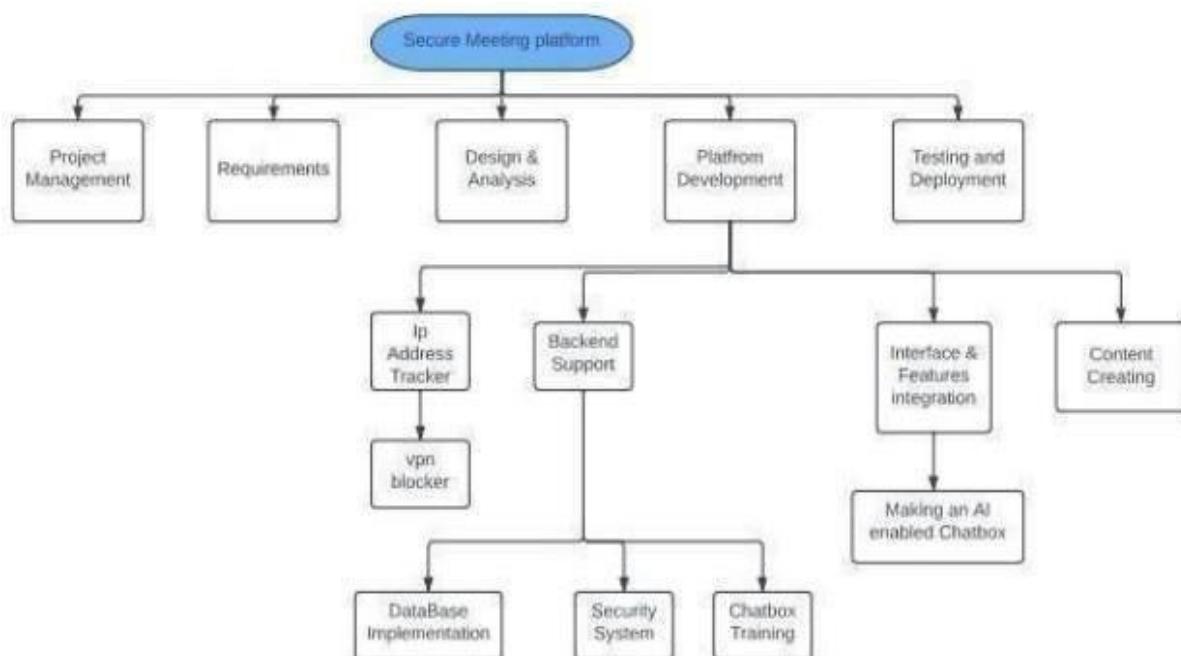
RACI Matrix		Team Members		
Activity		Urnava (Designer)	Khayati (Developer)	Harshika(Project Manager & BA)
User Requirement Documentation	-		C/I	A
Advertisement	C		C	R
Development	C		R	I
Website Design	R		A	I
Testing/Deployment	-		A	C
Bug Fixes	A		R	I
Update & Upgrade	-		C	A

A	Accountable
R	Responsible
C	Consult
I	Inform

EX.5

WORK BREAKDOWN STRUCTURE AND RISK ANALYSIS

Work breakdown structure (WBS) is a method for completing a complex, multi-step project. It's a way to divide and conquer large projects to get things done faster and more efficiently. Each WBS level represents a new and increasingly detailed definition of work needed to complete the project. A WBS structure must be constructed in a way that each new level in the hierarchy includes all the work needed to complete its parent task. This means that every parent task element must have more than one child task within it to consider the parent task element complete.



GANTT CHART



RISK ANALYSIS – SWOT & RMMM

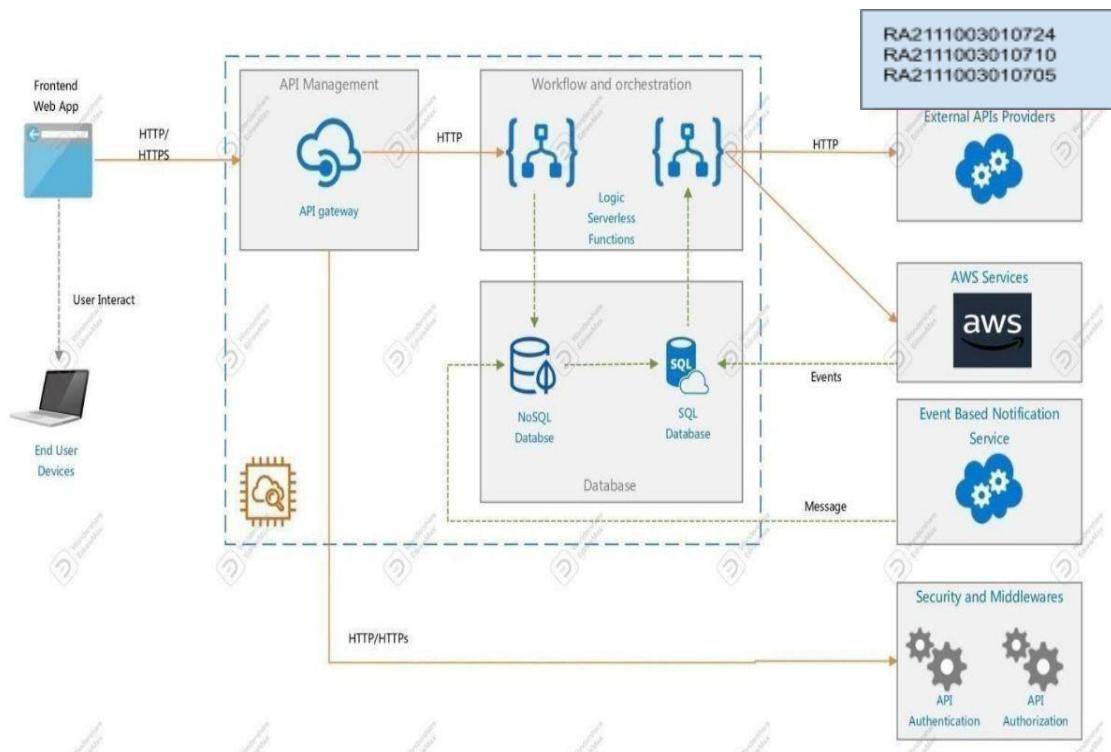
Strength	Weakness
Secure meeting platform Ip Tracking System AI-Enabled Chabot.	Tracking new or self-made VPN More research orientated Requires more execution time
Opportunities	Threats
Improve the learning experience maintaining the decorum of the meeting proper secure management of class	Ransomware Phishing

RESPONSE	STRATEGY	EXAMPLES
AVOID	Risk avoidance is a strategy where the project team takes action to remove the risk or protect from the impact	<ul style="list-style-type: none"> ● extending the schedule ● reducing/removing the scope ● change the execution strategy
TRANSFER	Risk transference involves shifting or transferring the risk threat and impact to the third party, rather transfers the responsibility and ownership	<ul style="list-style-type: none"> ● purchasing insurance ● performing bonds ● warranties ● contact issuance
MITIGATE	Risk migration is a strategy where the project team takes action to reduce the probability of the risk occurring. This does not risk or potential impact, but rather reduces the likelihood of it becoming real	<ul style="list-style-type: none"> ● increasing testing ● changing suppliers to a more stable one ● reducing process complexity
ACCEPT	Risk acceptance means the team acknowledges the risk and its potential impact but decides not to take any preemptive action to prevent it. It is dealt with only if it occurs	<ul style="list-style-type: none"> ● contingency reserve budgets ● management schedule float ● event contingency

EX.6

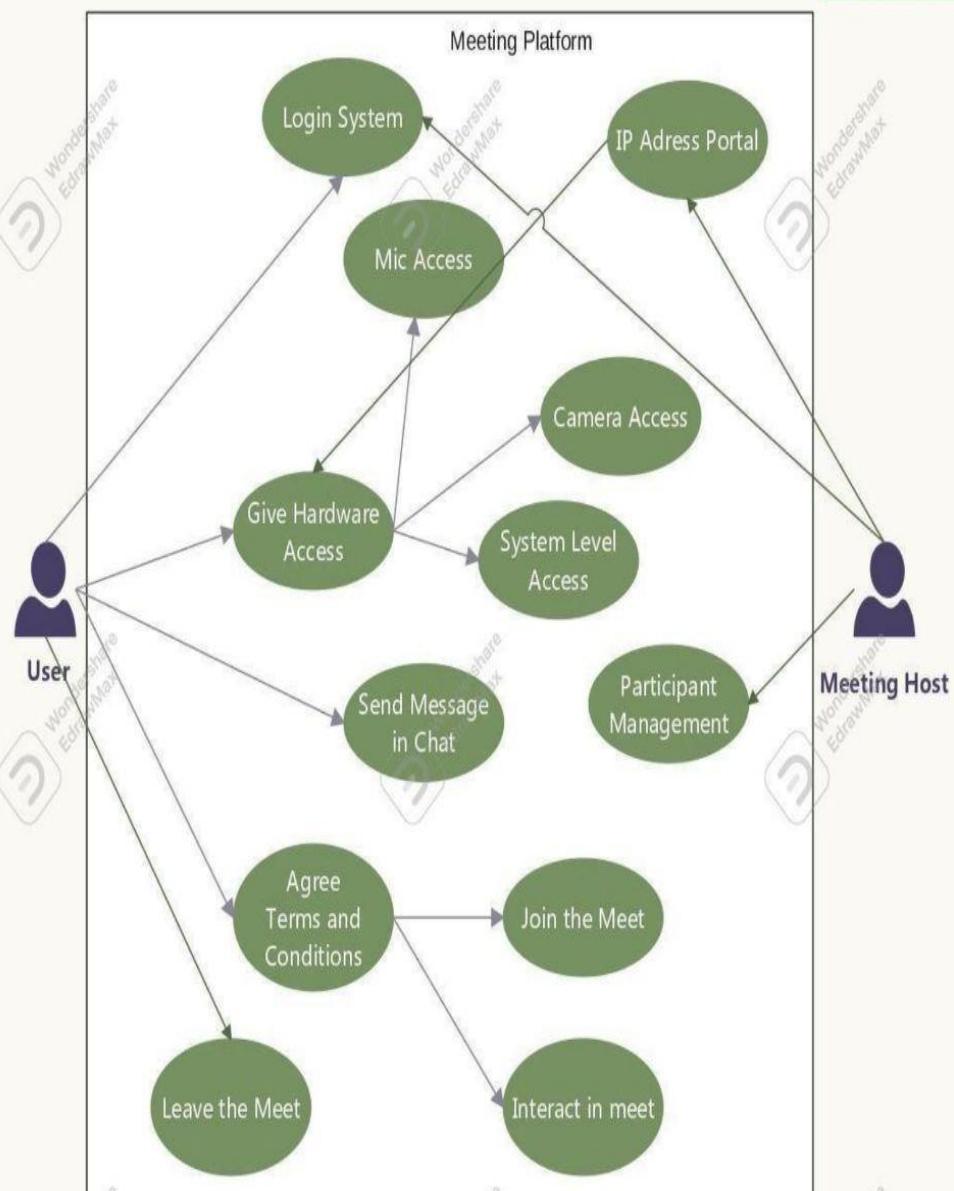
SYSTEM ARCHITECTURE, USE CASE & CLASS DIAGRAM

SYSTEM ARCHITECTURE:



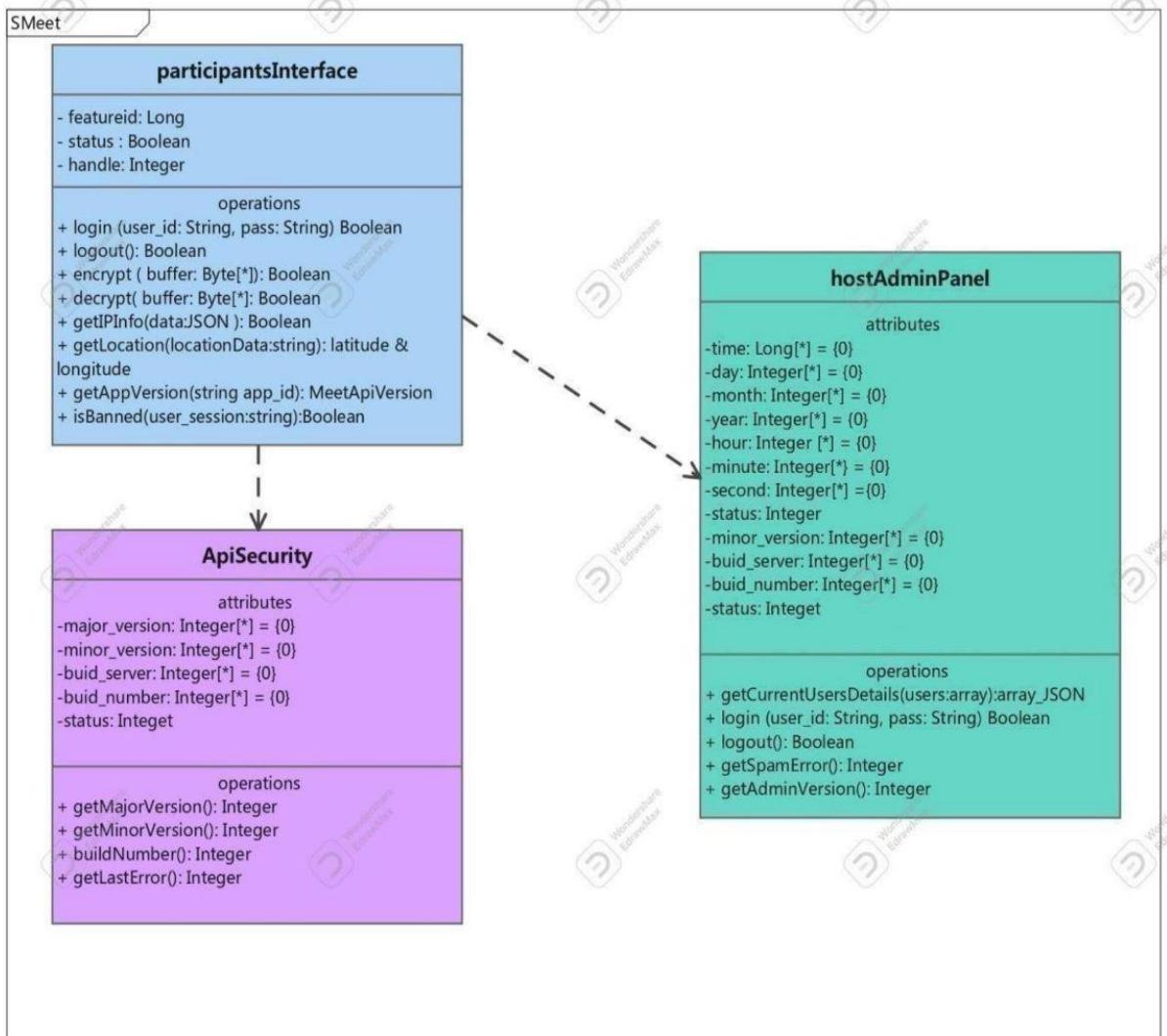
Secure Meet Platform Use Case

RA2111003010724
RA2111003010710
RA2111003010705



RA2111003010724
RA2111003010710
RA2111003010705

Secure Meet Platform Class Diagram

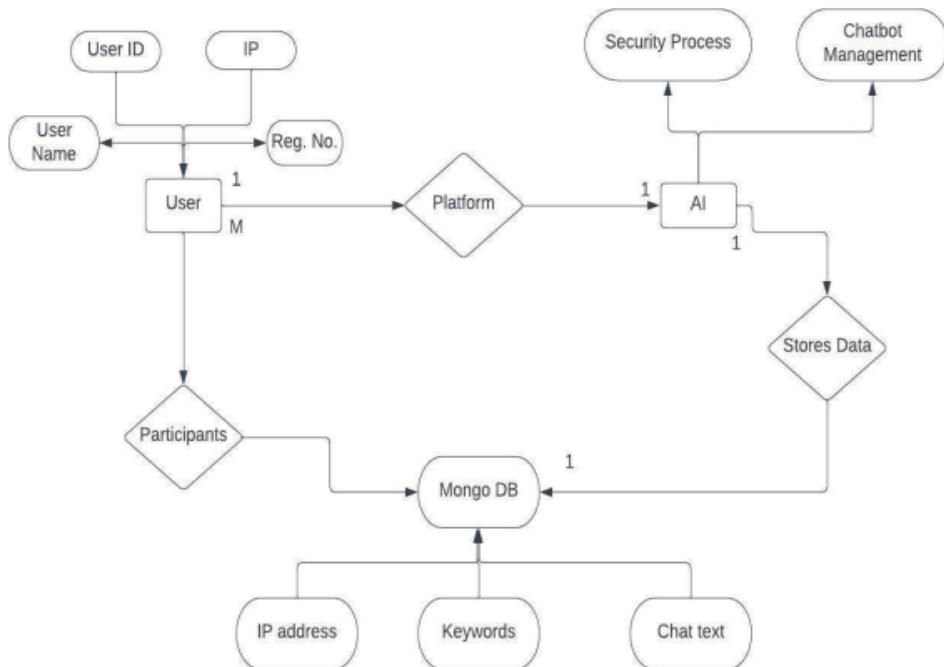


EX.7

ENTITY RELATIONSHIP DIAGRAM

An Entity Relationship (ER) Diagram is a type of flowchart that illustrates how “entities” such as people, objects or concepts relate to each other within a system. ER Diagrams are most often used to design or debug relational databases in the fields of software engineering, business information systems, education and research. Also known as ERDs or ER Models, they use a defined set of symbols such as rectangles, diamonds, ovals and connecting lines to depict the interconnectedness of entities, relationships and their attributes. They mirror grammatical structure, with entities as nouns and relationships as verbs.

ER Diagram of Secure Meeting Platform



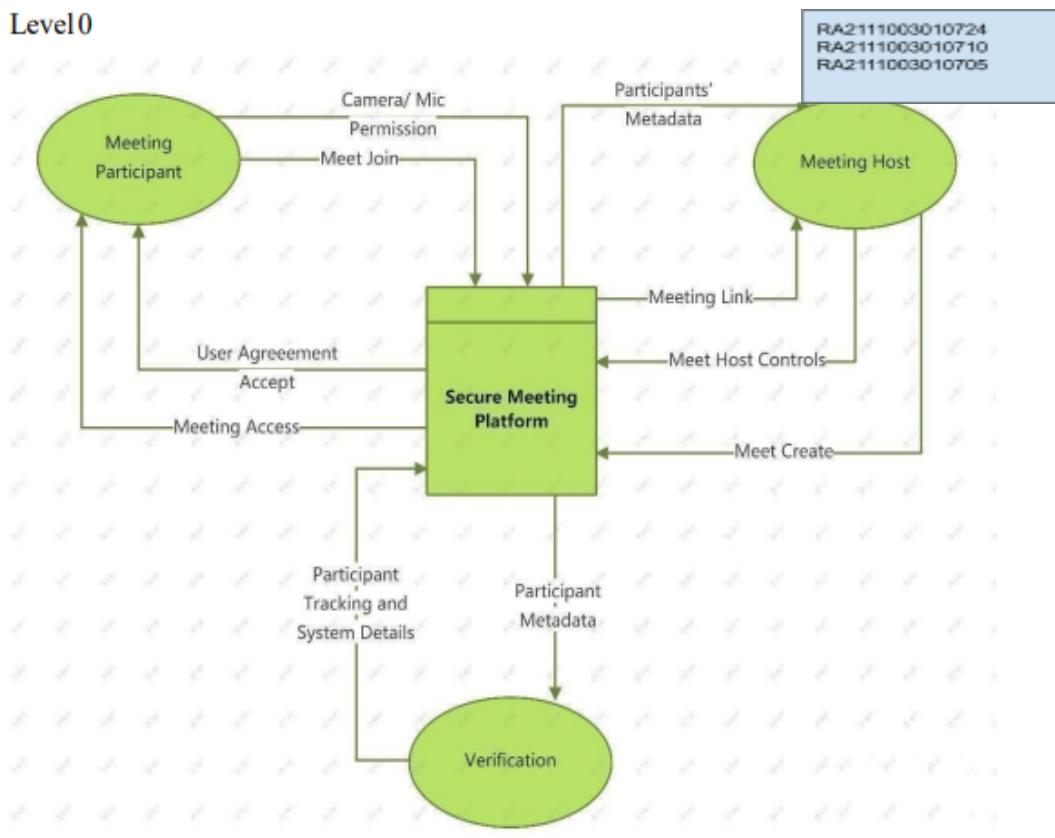
EX.8

DATA FLOW DIAGRAM

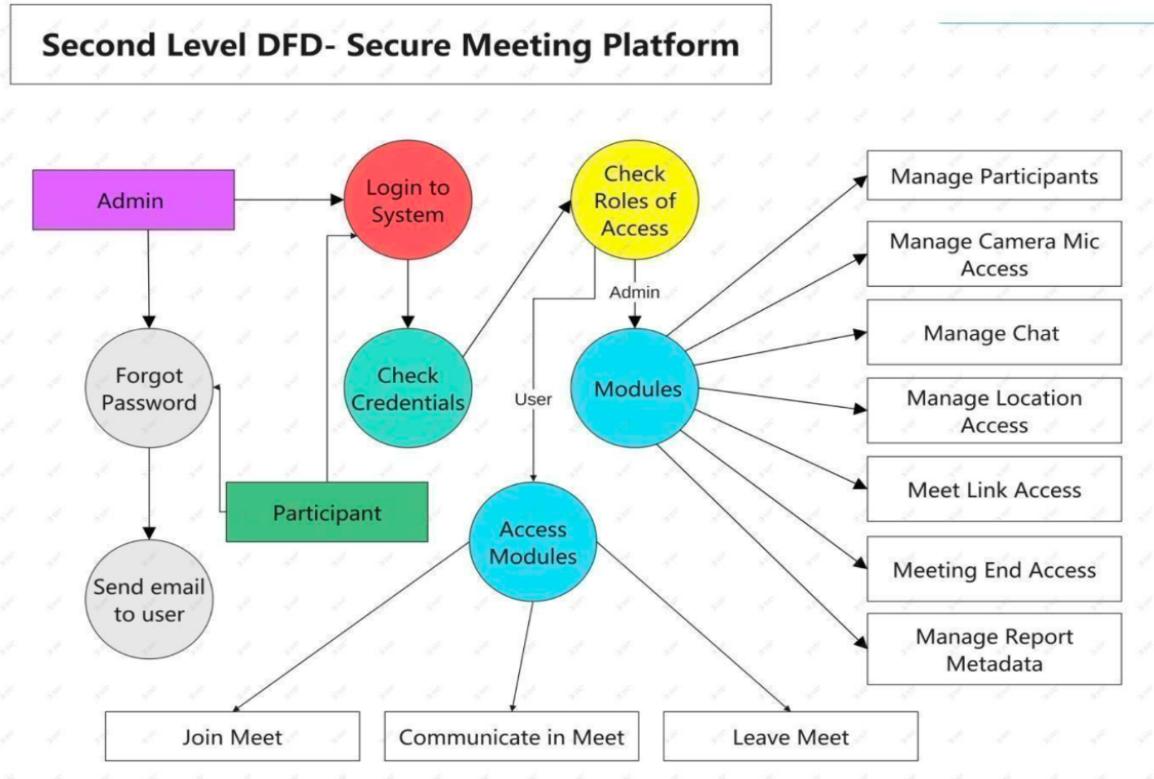
A data flow diagram (DFD) maps out the flow of information for any process or system. It uses defined symbols like rectangles, circles and arrows, plus short text labels, to show data inputs, outputs, storage points and the routes between each destination. Data flowcharts can range from simple, even hand-drawn process overviews, to in-depth, multi-level DFDs that dig progressively deeper into how the data is handled. They can be used to analyze an existing system or model a new one. Like all the best diagrams and charts, a DFD can often visually “say” things that would be hard to explain in words, and they work for both technical and nontechnical audiences, from developer to CEO. That’s why DFDs remain so popular after all these years.

DFD

Level0



Level 1

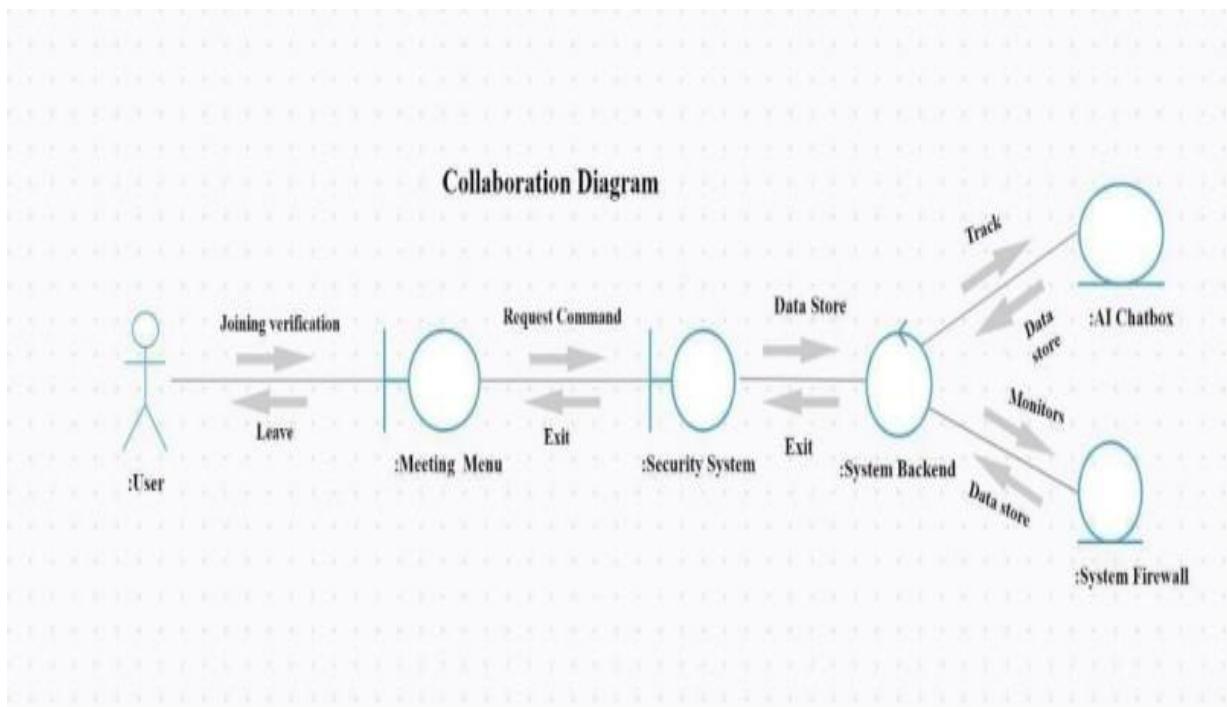


EX.9

SEQUENCE & COLLABORATION DIAGRAM

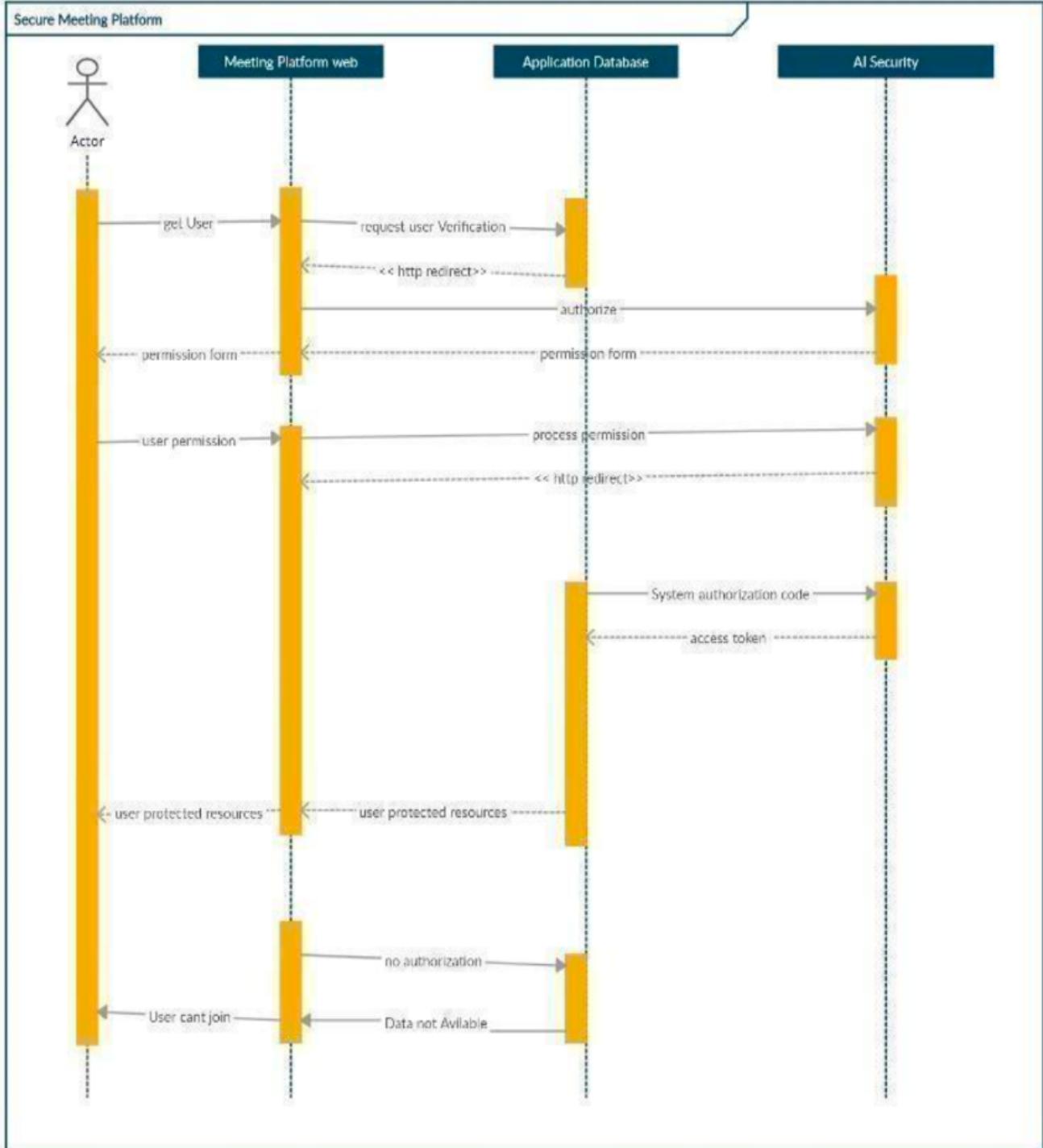
Collaboration Diagram:

Collaboration diagrams (also known as Communication Diagrams) are used to show how objects interact to perform the behavior of a particular use case, or a part of a use case. Along with sequence diagrams, collaboration diagrams are used by designers to define and clarify the roles of the objects that perform a particular flow of events of a use case. They are the primary source of information used to determine class responsibilities and interfaces.



Sequence Diagram:

A sequence diagram is a type of interaction diagram because it describes how—and in what order—a group of objects works together. These diagrams are used by software developers and business professionals to understand requirements for a new system or to document an existing process. Sequence diagrams are sometimes known as event diagrams or event scenarios.



EX.10

DEVELOPMENT OF TESTING FRAMEWORK / USER INTERFACE

Executive Summary

As the project is to be used to be for secure meeting platform:-

- i) The design should be simple,elegant, and formal.
- ii) The goal is to thoroughly test the user interface and all of its components.

In order to construct a cross-platform website that works on both desktop computers and mobile devices like Android and iOS, we want to upload party logos and utilize CSS frameworks like bootstrap.

Design Plan

The plan is to keep the design simple with large icons and buttons so that someone with weaker vision would face no problem during the meeting.

Consistent design,simplicity,lighter background colours,design that works across platforms and an emphasis on symbols should all be considered while creating this design so that even those with less literacy can easily recognize the members present in the meeting. Consistency transforms a good design into a great design. Consistency improves UX, general usability and the efficiency.

Test Plan

Scope of Testing

Speed: Ensure that our platform will deliver a quick and instant response to all types of users.

User Security: we promise to make our Security system intelligent enough to block all possible malpractices.

Interoperability: System should have the ability to gather relevant information, and its setting should be configurable when users select any channels. Overall, we will perform interoperability testing to help your users get the right thing from the platform at the right time.

Functionality: All of your Platform's features will work perfectly when our functional testing is complete. With our aid, you can ensure that the user's meeting experience is enhanced.

Comprehension Abilities: With testing, we can ensure an error-free and good texting experience from our system.

Types of Testing, Methodology, Tools

Category	Methodology	Tools Required
Functional Requirements	Manual	Word Template
Security Testing	RAD Security Testing	Zap Attack Proxy(ZAP)
User Acceptance Testing(UAT)	Operational Acceptance Testing	Rally Software
A/B Testing	Split-run testing(or) Bucket testing	Google Optimize
Adhoc Testing	RAD Development	Selenium

EX.11

TEST CASES

The test case is defined as a group of conditions under which a tester determines whether a software application is working as per the customer's requirements or not. Test case designing includes preconditions, case name, input conditions, and expected result. A test case is a first level action and derived from test scenarios.

Functional Test Cases

Test ID (#)	Test Scenario	Test Case	Execution Steps	Expected Outcome	Actual Outcome	Status	Remarks
1.	Verify User Registration	Verify Mobile Number and Email through OTP	<ol style="list-style-type: none">User clicks on User Registration linkEnter the mobile Number and Email Id on the text boxClick Register button	User should be taken to the next page for verification	User registered with his own email and password	Pass	Success
2.	Verify User behind a VPN or Proxy tunnel	Verify their IP address and match with well-known VPN Proxy server	1. While accessing user's IP address is used to verify automatically in backend	User should be accessed to join meet if passed	User is not behind a VPN or Proxy	Pass	Success
3.	User behavior check	Check chat box and voice for abusive words or behaviour	Chat box and voice are gone through AI/ML models to check for behaviour	User should be friendly in meet and should maintain behaviour	User if disobey the rule, IP address will be stored and will be	Pass	Success

					kicked out from meet		
4.	User Login Check	Check User ID password with registered users	Stored data will be checked with input data and authentication is checked	User will be authenticated with the original ID and Password	User most of the time uses weak password to pass	Pass	Success

Non-Functional Test Cases

Test ID (#)	Test Scenario	Test Case	Execution Steps	Expected Outcome	Actual Outcome	Status	Remarks
1.	Ensure security	One user should be allowed only one login	Will be checking login semaphore variable to check status	User can't login in multiple device or multiple session	User needs to login one device at a time	Pass	Success
2.	Handling traffic on website	Large number of requests should not overload the website	setting update limiter and load balancing fixes the issue	User gets faster access to the website	User's are divided into different servers for balancing sessions	Pass	Success

EX.12

MANUAL TEST CASE REPORTING

A test report is an organized summary of testing objectives, activities, and results. It is created and used to help stakeholders (product manager, analysts, testing team, and developers) understand product quality and decide whether a product, feature, or a defect resolution is on track for release.

Manual Test Cases:

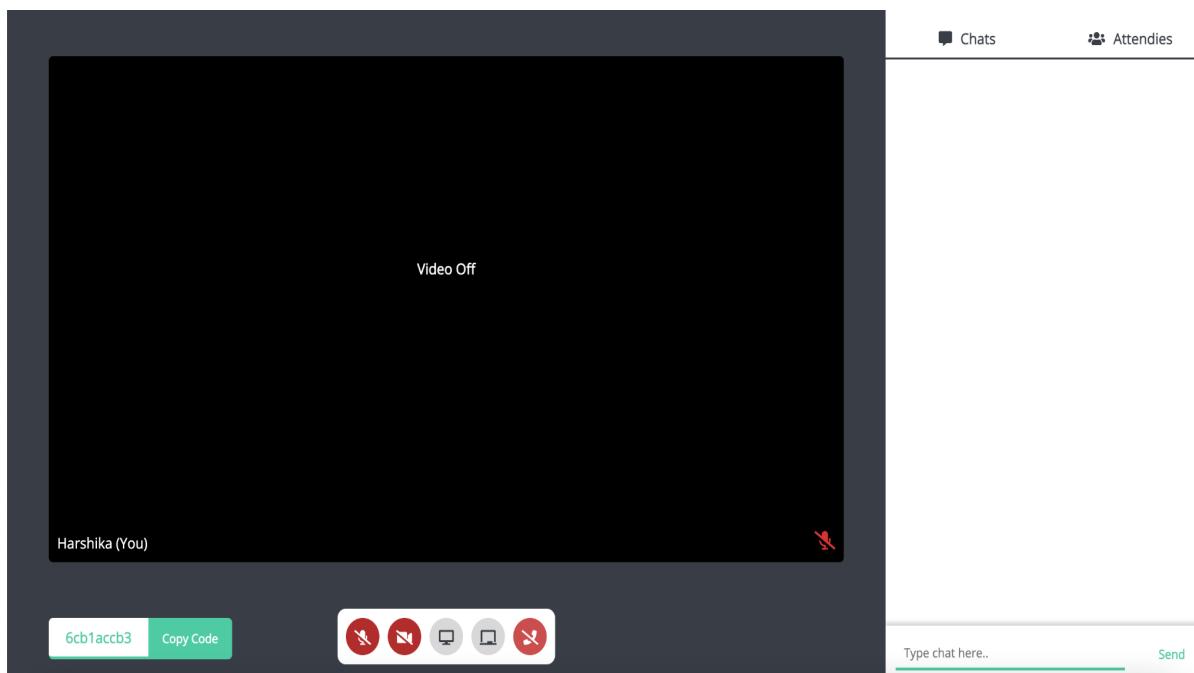
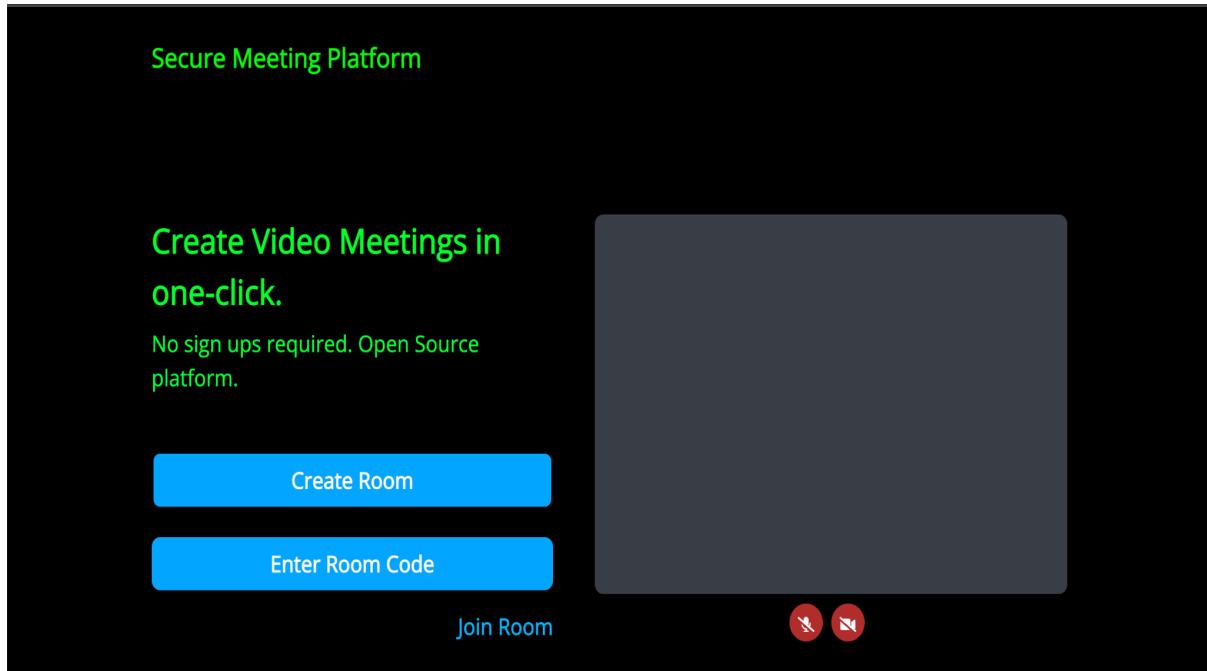
Test Area	Input
Login Verification	IP address and user details
Chatbox AI System	Message and Spanning
Blocking System	Admin Input and different security configuration

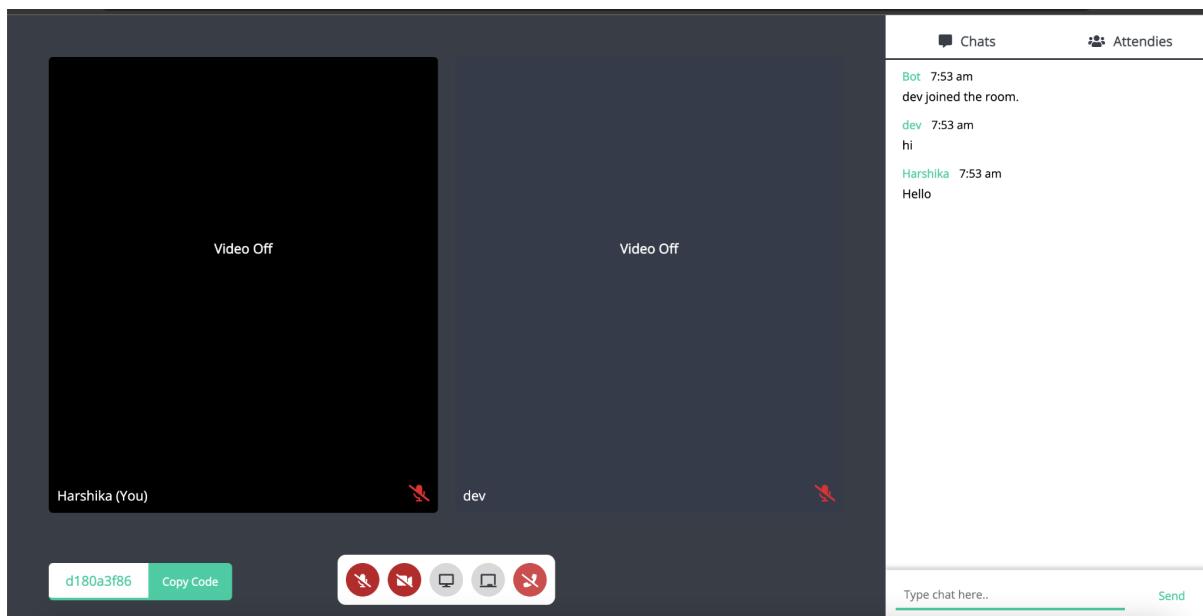
Category	Progress Against Plan	Status
Functional Testing	Amber	In-Progress
Non-Functional Testing	Amber	In-Progress

Functional	Test Case Coverage (%)	Status
Login Verification	40%	In-Progress
Chatbox AI System	40%	In-Progress
Blocking System	20%	In-Progress

EX.13

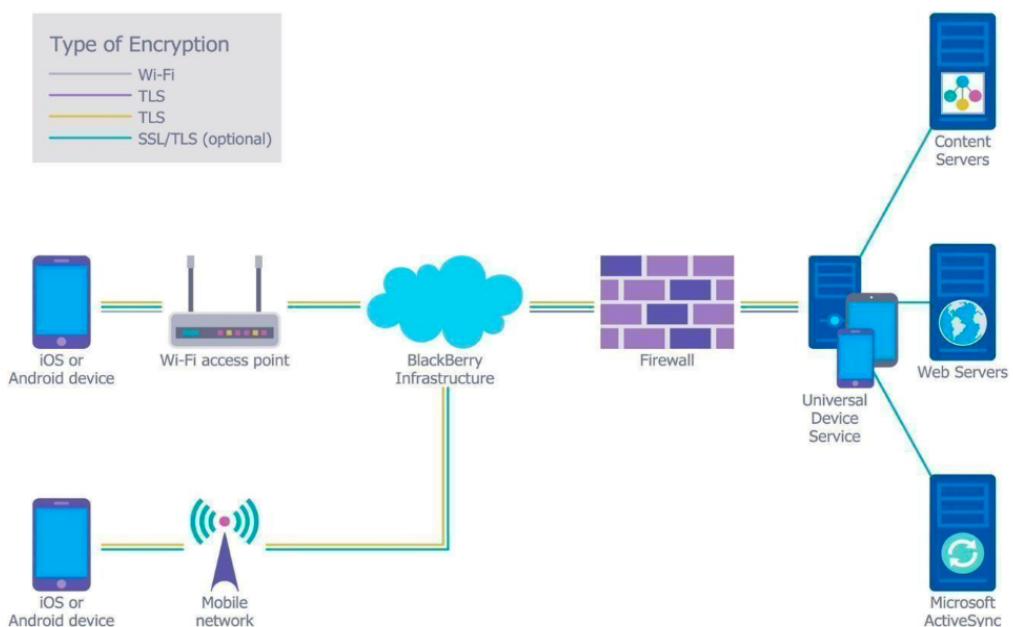
ARCHITECTURE DESIGN/FRAMEWORK/IMPLEMENTATION





A secure meeting platform that replaces all sorts of meeting systems with high-security features and is utilized to give a safe and protected atmosphere. It is built with ReactJS, Node.js, ExpressJS, MongoDB and SocketIO.

System Architecture



CONCLUSION

We have made the platform in which there will be an automatic Proxy and VPN detector so that the intruder is not able to forward his/her traffic through the VPN server and join the meet after it. Our system will block this type of traffic and will make the meet secure. We need to record each user's IP address for tracking purposes, which may limit the platform's audience and automatically block the person who tries to do malpractices and send the alert to the host of the meeting. We also have the AI chatbox which blocks unwanted messages or spamming in the chatbox. We have focused on the security of the user using the platform so that he is safe from frauds. Our website will first verify User's registration through OTP in email and mobile number and check for their IP address and match with VPN Proxy Servers. In our project we discussed the prioritization of Stakeholders, Work Breakdown Structure, SWOT Analysis, Risk Management, System Architecture, Gantt chart and many Diagrams like- Use Case Diagrams, Class Diagrams, ER Diagrams, Sequence Diagram, Collaboration Diagram and we also made the framework for our website for better understanding of our project.

REFERENCES

- 1) Gate Smashers, “*Software Development Life Cycle*”, [YouTube](#)
- 2) Jitsi “*Jitsi-meet*”, [Github](#)
- 3) Uy Nguyen, “*Documenting a Software Architecture*”, [Github](#)
- 4) Node.js, “*Node.js v16.15.1 Documentation*”, [NodeJS](#)
- 5) Express.js, “*Express.js v 5.x APP*”, [ExpressJS](#)
- 6) Typescript, “*TypeScript Documentation*”, [TypeScript](#)

APPENDIX

Front End part:

Index file detailing the structure and interconnection of all data

```
<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="utf-8" />

    <meta name="viewport" content="width=device-width, initial-scale=1" />

    <meta name="description" content="Secure Meeting platform." />

    <meta name="keywords" />

    <meta name="author" content="SMeet" />

    <link rel="icon" href="%PUBLIC_URL%/icon.png" />

    <meta name="theme-color" content="#000000" />

    <link rel="manifest" href="%PUBLIC_URL%/manifest.json" />

    <title>Video Meeting</title>

    <script async src="https://www.googletagmanager.com/gtag/js?id=UA-163893871-2"></script>

    <script>

        window.dataLayer = window.dataLayer || [];
        function gtag() {dataLayer.push(arguments);}
        gtag('js', new Date());
        gtag('config', 'UA-163893871-2');

    </script>
```

```
</head>

<body>

<noscript>You need to enable JavaScript to run this app.</noscript>

<div id="root"></div>

</body>

</html>
```

JS file for the functionality:

```
const express = require('express') const http = require('http')

var cors = require('cors')

const app = express()

const bodyParser = require('body-parser') const path = require("path")

var xss = require("xss")

var server = http.createServer(app)var io = require('socket.io')(server)

app.use(cors())

app.use(bodyParser.json())

if(process.env.NODE_ENV==='production'){

app.use(express.static( dirname+"/build"))

app.get("*", (req, res) => {

res.sendFile(path.join( dirname+"/build/index.html"))
```

```
})
}

app.set('port', (process.env.PORT || 4001))

sanitizeString = (str) => { return XSS(str)

}

connections = {}

messages = {}

timeOnline = {}

io.on('connection', (socket) => {

  socket.on('join-call', (path) => { if(connections[path] === undefined){ connections[path]= [] }

  }

  connections[path].push(socket.id)

  timeOnline[socket.id] = new Date()

  for(let a = 0; a < connections[path].length; ++a){ io.to(connections[path][a]).emit("user-joined", socket.id, connections[path]) }

  if(messages[path] !== undefined){

    for(let a = 0; a < messages[path].length; ++a){ io.to(socket.id).emit("chat-message", messages[path][a]['data'], messages[path][a]['sender'], messages[path][a]['socket-id-sender']) }

  }

  console.log(path, connections[path]) })

})
```

```
socket.on('signal', (toId, message) => {
  io.to(toId).emit('signal', socket.id, message)
})

socket.on('chat-message', (data, sender) => {data = sanitizeString(data)
  sender = sanitizeString(sender)

var key
var ok = false

for (const [k, v] of Object.entries(connections)) { for(let a = 0; a < v.length; ++a){
  if(v[a] === socket.id){

    key = k
    ok = true
  }
}

if(ok === true){

  if(messages[key] === undefined){

    messages[key] = []
  }

  messages[key].push({"sender":    sender,    "data":    data,    "socket-id-sender":    socket.id})
  console.log("message", key, ":", sender, data)
}
```

```
for(let a = 0; a < connections[key].length; ++a){ io.to(connections[key][a]).emit("chat-message",
data, sender, socket.id) }

}

socket.on('disconnect', () => {

var diffTime = Math.abs(timeOnline[socket.id] - new Date())var

key

for (const [k, v] of JSON.parse(JSON.stringify(Object.entries(connections)))) {for(let a = 0; a <
v.length; ++a){

if(v[a] === socket.id){

key = k

for(let a = 0; a < connections[key].length; ++a){ io.to(connections[key][a]).emit("user-left",
socket.id) }

var index = connections[key].indexOf(socket.id)

connections[key].splice(index, 1)

console.log(key, socket.id, Math.ceil(diffTime / 1000))

if(connections[key].length === 0){

delete connections[key]

}

}

}

}

})
```

```
})  
  
server.listen(app.get('port'), () => {  
  
  console.log("listening on", app.get('port'))  
  
})
```