Deliverable 2 – Requirements Specifications

CSCE 5430 (Fall 2021)

Project Title - Sociogram (Web Application)

Team Name – Code6

Group Members -

1. Sathwik Gaddi

2. Tanvi Thirunathan

3. Satish Thammaneni

4. Sai Karthik koncherlakota

5.. Raja Srinivas Nelluri

6. Sri Snigdha kotharu

7. Venkata Rajashekar Reddy Chinthala

8. Putta Ramya Priya

Version: 1.0

Document History

Version	Date	Author	Comments
1.0	30th Sep, 2021	Code6	Deliverable 2

I. TABLE OF CONTENTS

II. List of Figures and Tables

1. Introduction

This section gives a scope description and overview of the project functionalities included in this SRS document. Also, the purpose for this document is described and definitions are provided.

1.1 Purpose

The purpose of The SRS document defines the software development life cycle of Sociogram (web Application). It describes about the software requirement specifications of the first version of our project i.e., Sociogram. This SRS covers the requirement specification for all the subsystems involved in the project.

1.2 Document Conventions

The standards that are used throughout the documentation are detailed below

a. Document is formatted with TIMES NEW ROMAN font.

b. All the Section names are outlined BOLD and UNDERLINED with font size 14 and subsections are outlined BOLD with font size 12.

c. All the diagrams are labelled with figure number and relevant captions in Italics.

d. Each of the feature and their purpose are described in detail and has their own significance.

Acronym	**Abbreviation**
OS	Operating System
DB	Database
HTTP	Hyper Text Transfer Protocol
UI	User Interface
API	Application Programming Interface

Table 1.2.1 Acronyms and Abbreviations

1.3 Intended Audience and Reading Suggestions

This document is meant to be referred by the developers who are involved in programming the application, testers, instructor and graders to stay in line through all the phases/versions of the project development. Members who can access this report, have to go through the table of contents to understand the structure of the document.

The Section 2 “Overall Description “defines the high-level view of the product features, the operational environment, the design constraints and the assumptions and dependencies that can be encountered

The Section 3 “System Features” gives an overview of Product features and The In-detail description and the functional requirements of the features.

The Section 4 “External Interface Requirements” defines all the types of interfaces available for the application.

The Other Non-functional requirements, Development phases, Appendixes are described in sections 5,6,7 respectively.

1.4 Project Scope

It is an application that acts as a medium for people to connect across the globe with just a single tap. User can manage their individual accounts by posting media, reacting to other posts, finding friends on basis on similar interests, can get access to discussion forums, can book tickets for ongoing movies/events. The user-friendly nature for the application will be achieved by creating simple UI and responsive UI.

1.5 References

“IEEE Recommended Practice for Software Requirements Specifications” –

IEEE Standards Board

The Institute of Electrical and Electronics Engineers, Inc. 345 East 47th Street, New York, NY 10017-2394, USA

Copyright © 1994 by the Institute of Electrical and Electronics
Engineers, Inc. All rights reserved. Published 1994. Printed in
the United States of America

2. Overall Description

This section will give an overview of the whole system. The system will be explained in its context to show how application will allow users to interact and introduce the basic functionalities of it. It will also describe the type of users that will use the application and what functionalities are available for each type.

2.1 Project Perspective

Sociogram is a social media platform which is inspired from many prominent applications like Facebook and Instagram. The proposed system is the replacement for the existing systems which additionally include the following features:

a. Find friends who have the similar interests and hobbies.
b. Book tickets for the movies and events.
c. Discussion Forums to get the public answers/opinion.

Sociogram is a user-friendly web application which helps to explore the social world with just a single sign up. The user-friendly nature for the application will be achieved by creating simple UI and responsive UI. Users create/edit/manage their own individual accounts and can find people with similar interest/hobbies. They can share the ideas/articles/ideologies via personal accounts. Discussion forums will be accessible to all the users who signed up. Users can book the tickets for ongoing and upcoming events or movies.

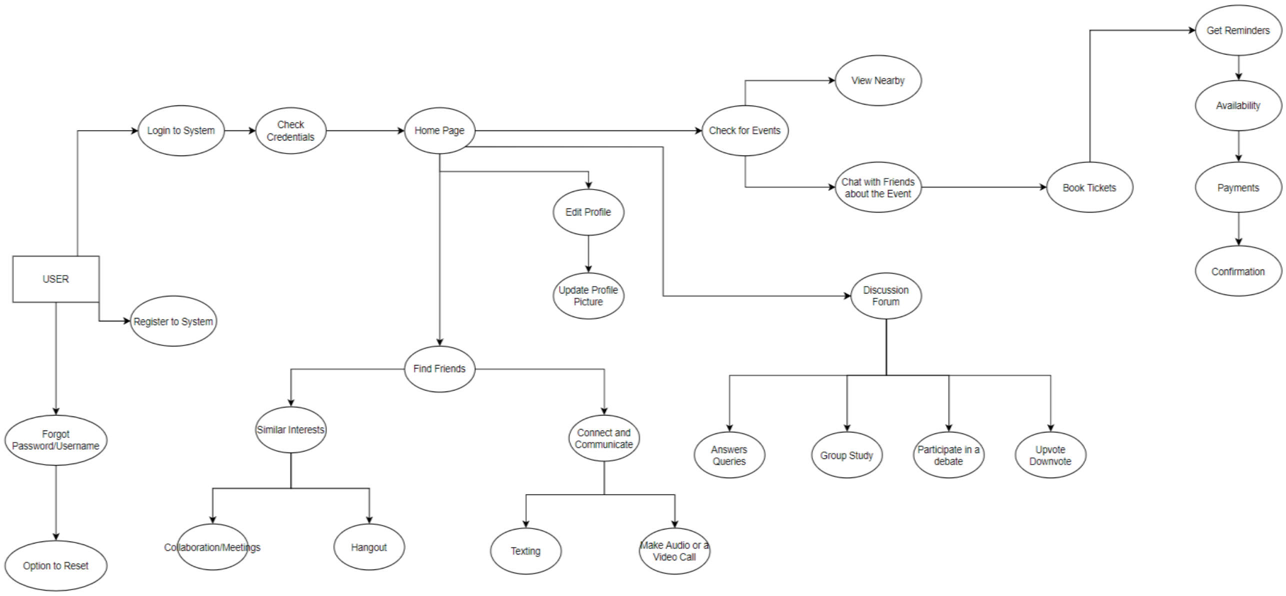

Fig 2.1.1 Overall Structure Diagram

2.2 Product Features

a. Find people with similar interests/hobbies.

b. Share media/articles/ideologies.

c. Connect/collaborate via messaging and audio calling

d. Discussion Forums

e. Explore/book tickets for ongoing/upcoming events.

2.3 User Classes and Characteristics

All the Users who have knowledge about browsers and internet can access the application. There are no specific types of users for the application. All the users come under a single category all the consumers will have access to every feature in the system.

2.4 Operating Environment

Our application is a web-based application which will be accessed via browsers like Google Chrome and Mozilla Firefox. Our application also supports Operating Systems like Windows, ubuntu and Mac Os.

The UI design for the application will be developed in HTML, CSS and JavaScript whereas the Backend Api's will be developed in Node Js.

All the user's data will be stored in the Mongo DB Atlas.

The Development will be done in Visual Studio Code. We use Windows 10 Operating System for development and Use Postman or locust for the load testing.

2.5 Design and implementation constraints

The First constraint is making sure the UI design of the application is simple and user friendly as the customer experience has to be much considered at.

The second constraint could be the availability of resources for the development of the application. All the developers and testers should have to make sure all the software and hardware requirements are readily accessible.

The Third constraint could be the performance or load testing that has to be done parallelly to ensure the best conduct and scalability of the application.

Lastly, developers involved in programming should maintain a set of standards that are to be followed by all of the coders to ensure there are no technical conflicts in the program.

2.6 User Documentation

{To be determined} – We currently do not have a user manual to guide the users on the Software.

2.7 Assumptions and Dependencies

It is assumed that user and other data will be used once the project is successfully built. Until then, data for testing will be made and used for providing the demo for the presentations.

As this is a web application and runs in the browser and need to be connected to the internet, it is assumed that the users will have decent internet connectivity.

It is assumed that user will be familiar with the usage of browser.

3. System Features

This section contains all of the functional and quality requirements of the system. It gives a detailed description of the system and all its features.

3.1 Find people with similar interests/hobbies

User can find friends based on similar interests, which means that the user gets to update/edit their interests/hobbies in their profile information such that they can connect to other users with similar diversion.

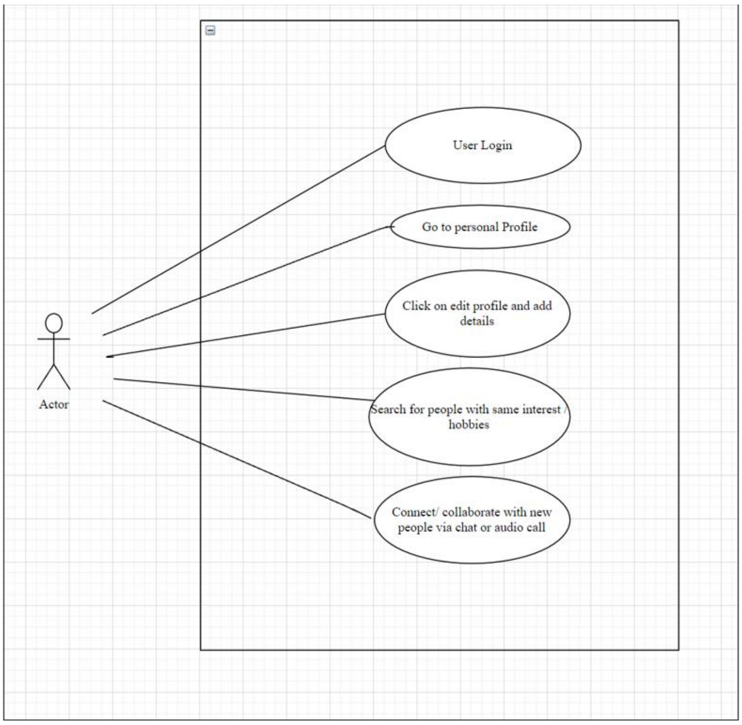
User Login
Go to personal Profile
Click on edit profile and add details
Actor
Search for people with same interest / hobbies
Connect/ collaborate with new people via chat or audio call

Fig 3.1.1 Use Case Diagram for Finding People

Functional Requirements –

REQ1 – User should provide his/her hobbies/interests while registration.

REQ2 – People who are a match should be retrieved from the updated Database.

REQ3 – Provide interface to enable connections between similar friends.

3.2 Upload media/articles/ideologies.

This website acts as a platform to share their ideas/articles/ideologies via their personal accounts. People who follow or are friends with that particular user will be able to view and react to the posts.

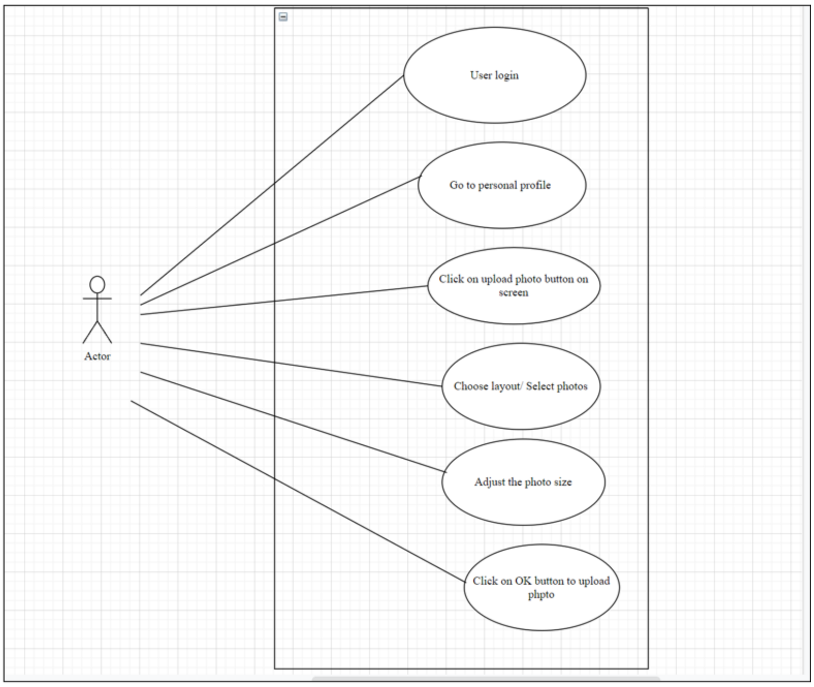
User login
Go to personal profile
Click on upload photo button on screen
Actor
Choose layout/ Select photos
Adjust the photo size
Click on OK button to upload phpto

Fig 3.1.2 Use Case Diagram for Uploading media

Functional Requirements –

REQ1 – Users should be able to upload the media to through the interface.

REQ2 – Retrieve and display the uploaded posts by friends.

REQ3 – Provide interface to react for the visible posts.

3.3 Messaging and Audio calling

Users can either message their friends or can make an audio call with a strong internet connectivity.

.

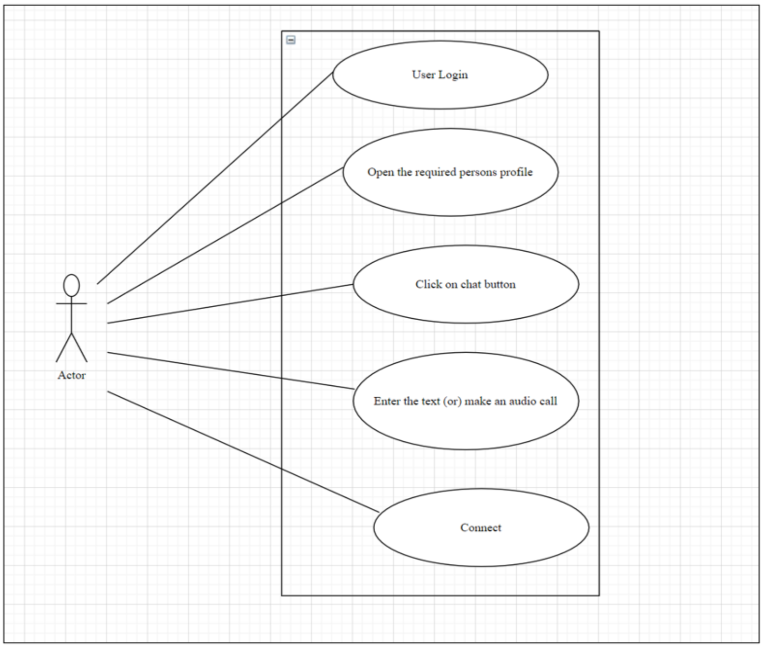
User Login
Open the required persons profile
Click on chat button
Enter the text (or) make an audio call
Connect
Actor

Fig 3.1.3 Use Case Diagram for Messaging

Functional Requirements –

REQ1 – Users should be able to connect to friends to message them.

REQ2 – Optimize the program to receive messages as fast as possible.

3.4 Discussion Forums

Discussion Forums is an open page which is accessible to all the users who are signed up to this site. When a public opinion/help is required, the users can post in their queries or can answer to the already existing/new posts.

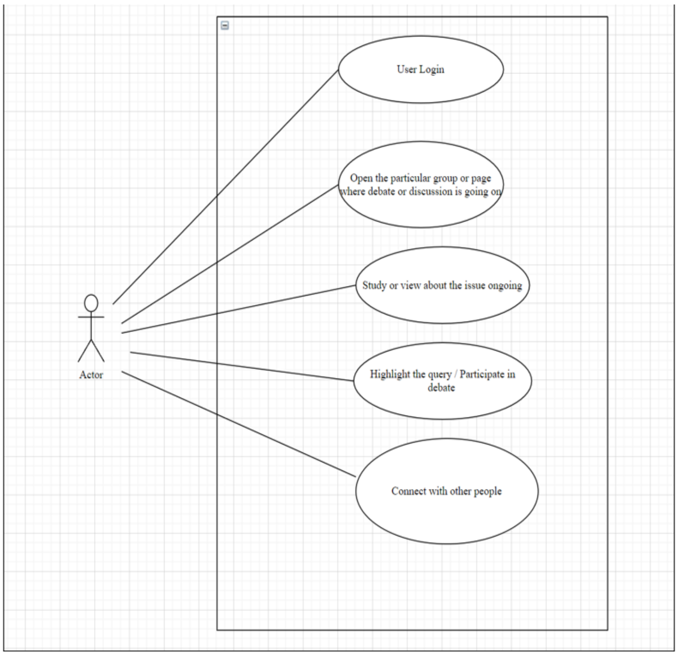
User Login
Open the particular group or page where debate or discussion is going on
Study or view about the issue ongoing
Highlight the query / Participate in debate
Connect with other people
Actor

Fig 3.1.3 Use Case Diagram for Discussion Forums

Functional Requirements –

REQ1 – Users should be able to post either question/opinions.

REQ2 – Provide interface to either upvote/downvote answers.

REQ3 – Allow multiple users to answer for the asked question/opinion.

3.5 Book Tickets

Users can look for and can book the tickets for ongoing or upcoming events/movies nearby as individual or group booking. Payments to the booked tickets will be made via online payment methods and the user who have booked the ticket will receive the updates and event reminders via registered email id.

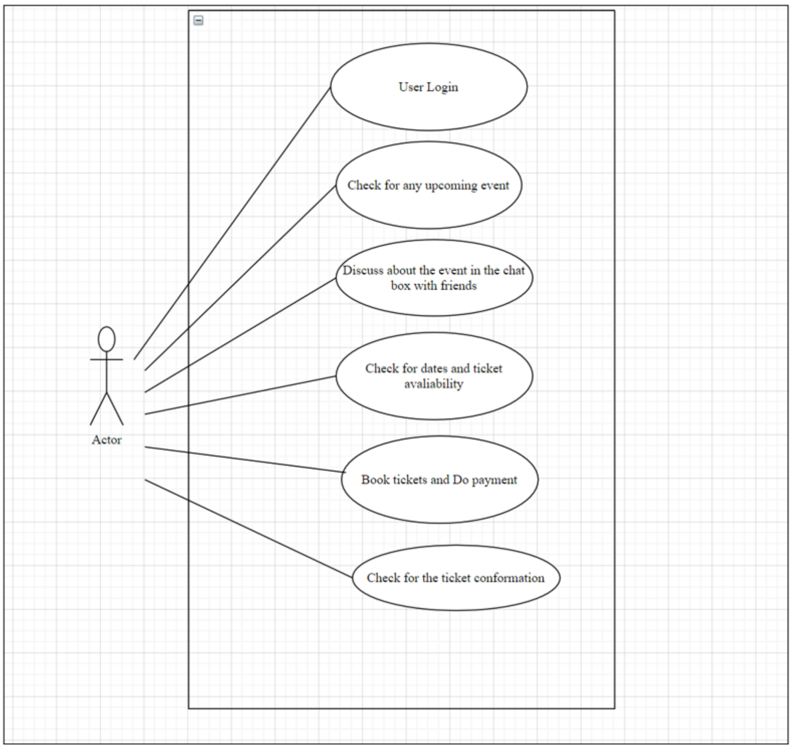
User Login
Check for any upcoming event
Discuss about the event in the chat box with friends
Check for dates and ticket avaliability
Actor
Book tickets and Do payment
Check for the ticket conformation

Fig 3.1.3 Use Case Diagram for Booking Tickets

Functional Requirements –

REQ1 – Allow users to view movies/events nearby.

REQ2 – Provide interface to book tickets.

REQ3 – Retrieve updated list of movies/events from the database.

4. External Interface Requirements

4.1 User Interfaces

The interface between user and software is designed by following the GUI standards. Below are the interfaces

a. Registration/signup
b. Login
c. Upload Media
d. Discussion Forum
e. Ticket Booking
f. Payment page for booked tickets

{To be determined}- The Sample Screens of UI/Backend Design

4.2 Hardware Interfaces

Currently there is no requirement for hardware interfaces.

4.3 Software Interfaces

Frontend UI design will be connected using tags to bootstrap to enhance the stylings of the user interface.

Backend server will be connected to MongoDB atlas using Mongo DB native drivers to secure enable database transactions.

4.4 Communication Interfaces

Backend Api's will be communicated with the front end by HTTP. While responding for a request the json web token will be shared to enhance the security for the data in the response.

Socket io will be used to enable communication within the friends and update the messages synchronously.

5. Other Non-Functional Requirements

5.1 Performance Requirements

1. Decrease the response time from above 400ms to below 300ms.
2. Implement database indexing to decrease the response time.

5.2 Safety Requirements

1. No chat data should be stored to ensure the privacy of the user.

2. Data backup should be implemented to prevent data loss during crash.

5.3 Security Requirements

1. Passwords are hashed using bcrypt hashing to ensure the security for the user.

2. The response data should be sent along with the json web token to authorize the user.

3. Database authorization should be achieved in Mongo DB Atlas.

5.4 Software Quality Attributes

Usability - The application should be able to satisfy a million number of consumer needs.

Functionality – It is the crucial requirement as it described how accurately and aptly the software features are functioning with the user requests.

Availability – The application has to perform the requested tasks by the users within 400ms of time.

6. Development Phases

We are going to develop our application in three different phases by adding of some extra features in each phase.

6.1 Phase 1

In this phase we are going to create a website for sociogram, and we are going to add three crucial features for sociogram.

1. User Profile:
   Create a login and sign-up page to get personalized user profile. The passwords are hashed using bcrypt algorithm to ensure the privacy of the user.
2. Upload data:
   Providing access to the clients to upload posts, media, and articles. Include a feature to add privacy for each of their post.
3. Find friends:
   Find friends with similar habits and hobbies with the help of suggestions. User will provide their interests when they register to find their identical.

Requirements

REQ1: User login page should have a form with username and password fields, a login button and a signup up button which on successful validation redirects to home page of the application and error message on Invalid credential confirmation. A Signup button should be added for new users to register for the application.
REQ2: Upon successful validation for login, a json web token should be sent from Node Js server to frontend to authorize user throughout the session.
REQ3: Signup page should have a form with First name, Last name, Username, password, confirm password, and a dropdown for interests and hobbies. On successful form validation by JavaScript the page should be redirected to Login page.
REQ4: Home Page of the application should have buttons to upload a new post, discussion forum, find friends and logout.
REQ5: Upload post screen should contain a form with fields for tagline, upload post and privacy.
REQ6: Find friends page should contain a list view to display people with similar interests.

6.2 Phase 2

In this phase we are adding two features they are:

1. Chatting features:

By using this feature users can communicate with their friends.

2. Discussion forums feature:

Open community page where users can raise queries, answer other queries, discuss with a whole lot of people.

Requirements

REQ1: Upon successful json web token authorization, user should be able to message their friends.
REQ2: Messaging page should consist a list view of friends.
REQ3: A close button should be available to end a chat with the user.
REQ4: Discussion forum page should consist a list view of all the discussions available in the application.
REQ5: Upvote and Downvote buttons should be available for each discussion.

6.3 Phase 3

Assigning booking ticket feature to the application. In this feature user can book tickets such as movies and events etc. Along with that we will fix bugs raised in our application and concentrate to improve performance will consider in this phase.

Requirements

REQ1: Booking page should contain a list view of all the events/movies.
REQ2: Locust should be used to perform load testing.

7. Other Requirements

Below are the appendixes such as Glossary, Analysis Models, to be determined List that are pertaining to the project.

7.1 Appendix A: Glossary

a. User – An individual who creates an account in Sociogram

b. Constraint – It defines a particular behavior or operation of the feature.

c. Assumptions – It defines the services of the software that are expected to be true.

d. Dependencies – It defines the relationship between the semantical and structural relationship of the project.

e. Interface – It acts as a bridge that helps software to interact with hardware devices.

f. Actor – I all the use case diagrams user is represented as actor.

7.2 Appendix B: Analysis Models

Below is the Application Architecture

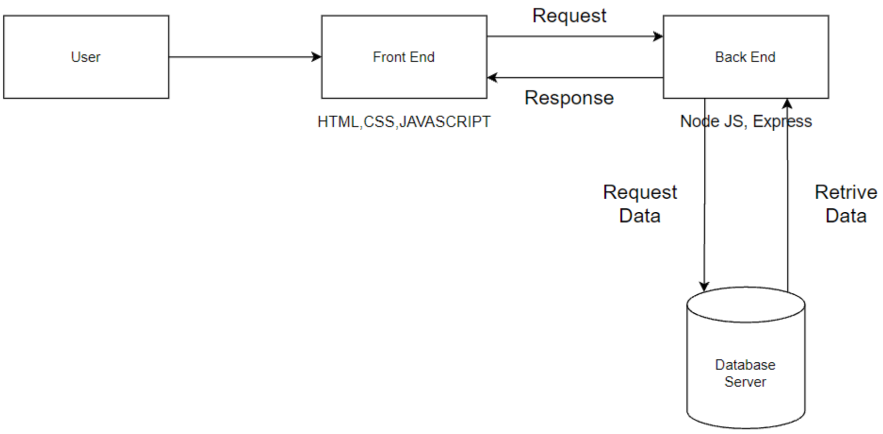

Fig 7.2.1 Architecture diagram for web application

7.3 To be Determined List

1. User Documentation
2. User Interfaces – Sample Screen Images

8 Member Contribution Table

Member name	Contribution Description	Overall contribution (%)	Note (if Applicable)
Sathwik Gaddi	Requirements Specifications	12.5 %	
Tanvi Thirunathan	Requirements Specifications	12.5 %	
Satish Thammaneni	Phase development and requirements	12.5 %	
SaiKarthik koncherlakota	Overall structure diagram	12.5 %	
Raja Srinivas Nelluri	Phase development and requirements	12.5 %	
Sri Snigdha kotharu	Use case diagrams	12.5 %	
Venkata Rajashekar Reddy Chinthala	Phase development and requirements	12.5 %	
Putta Ramya Priya	Use case diagrams	12.5 %	