

Project Title- SolutionBox
Technology –

- **Frontend** : React.js , HTML , CSS, JavaScript
- **Backend** : Node.js
- **Database** : MongoDB
- **API Tools** : Postman

Product Functions/Modules

1. User Authentication

- Secure user registration and login system
- JWT-based authentication for secure access

2. Question & Answer System

- Users can post technical or general queries
- Community-driven answers with upvote/downvote mechanism
- Markdown support for better formatting (code snippets, lists, etc.)
- Tags and categories for better content organization

3. Search & Filtering

- Advanced search functionality for quick access to relevant discussions
- Filters based on tags, popularity, and latest posts
- Related questions suggestions for better engagement

4. Discussion Threads

- Nested comment system for follow-up discussions

5. Reporting & Analytics

- Graphical representation of top tags and discussions

6. Security & Validation

- Session timeout and secure password hashing

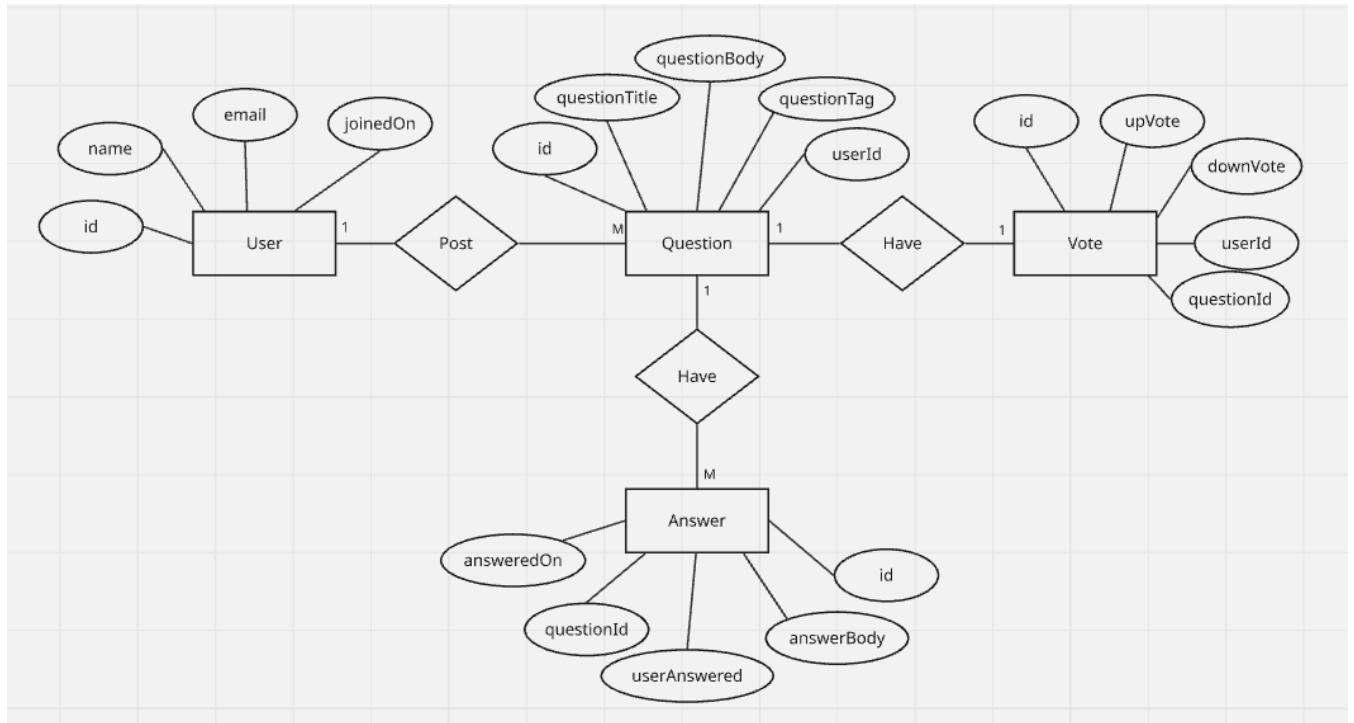
Users of system

General User

1. Posts questions and answers on website
2. Engages with content through upvotes, downvotes and comments.

3. Access of all question available on the platform.
4. Explore content using tags filters.
5. Update profile whenever user want.

E R Diagram :



Use Case Diagram :

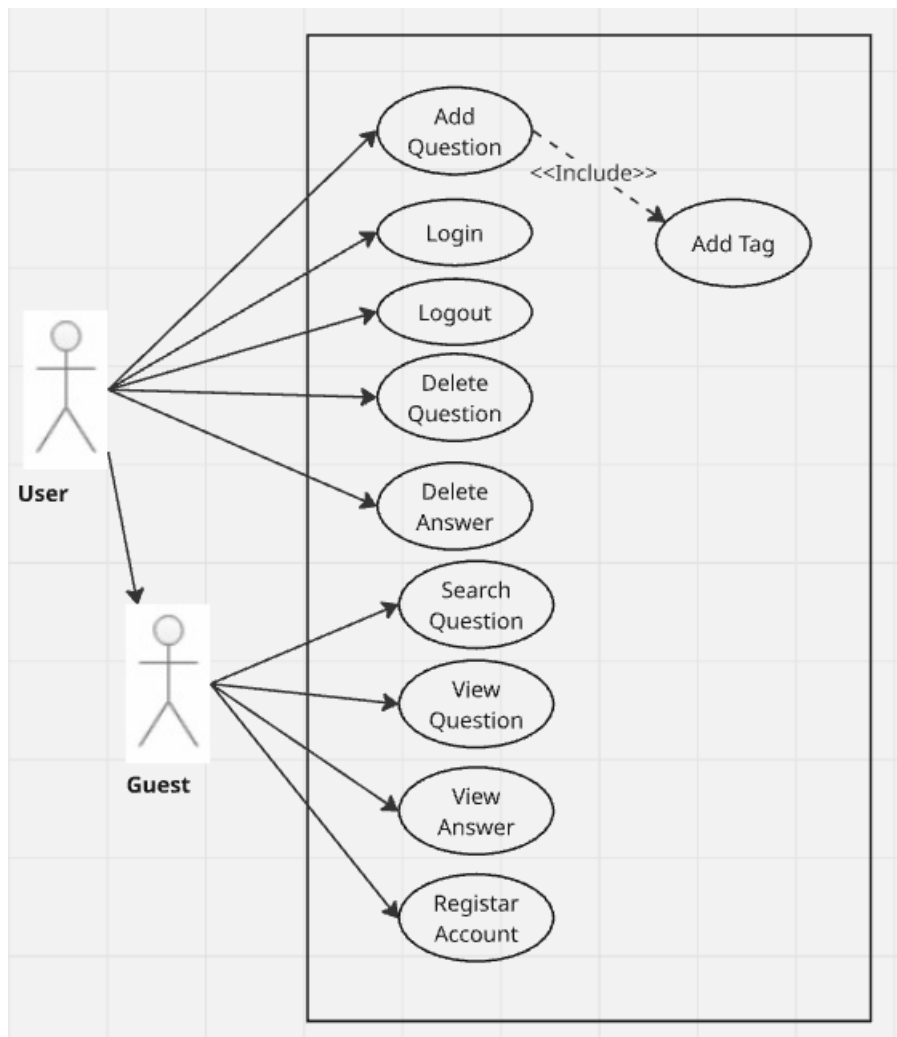
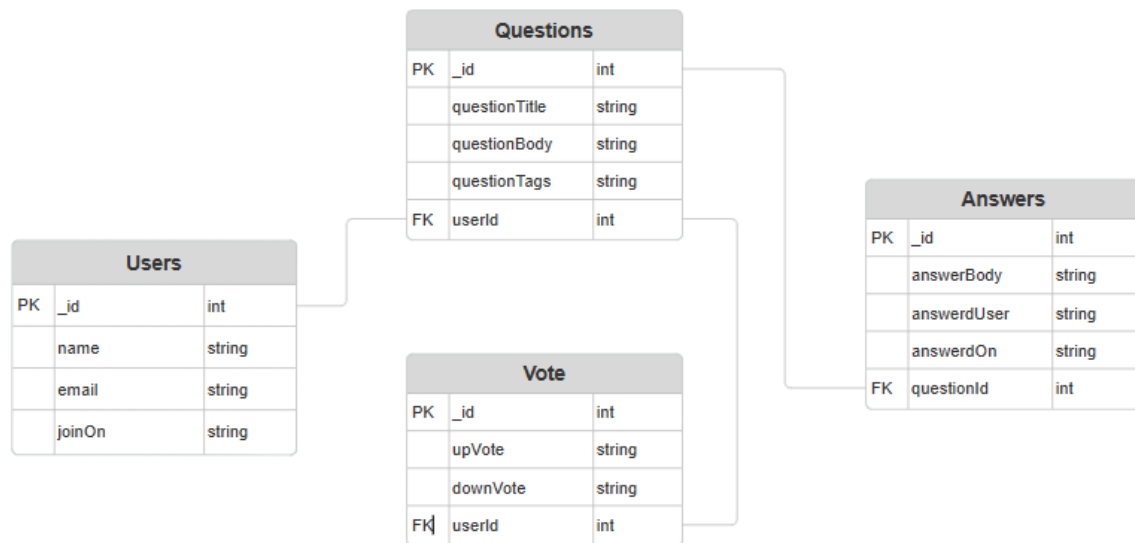


Table Structure :



Data Dictionary:

User Table:

Field name	Data Type	Description	Key
_id	int	Unique ID for each user	Primary
Name	string	Name of the user	
Email	string	Email address of the user	
joinOn	string	Date the user joined	

Question Table:

Field name	Data Type	Description	Key
_id	int	Unique ID for each question	Primary
questionTitle	string	Title of the question	
questionBody	string	Full content of the question	
questionTag	string	Tags related to the question	
userId	int	ID of the user who asked the question	Foreign (Users._id)

Vote Table:

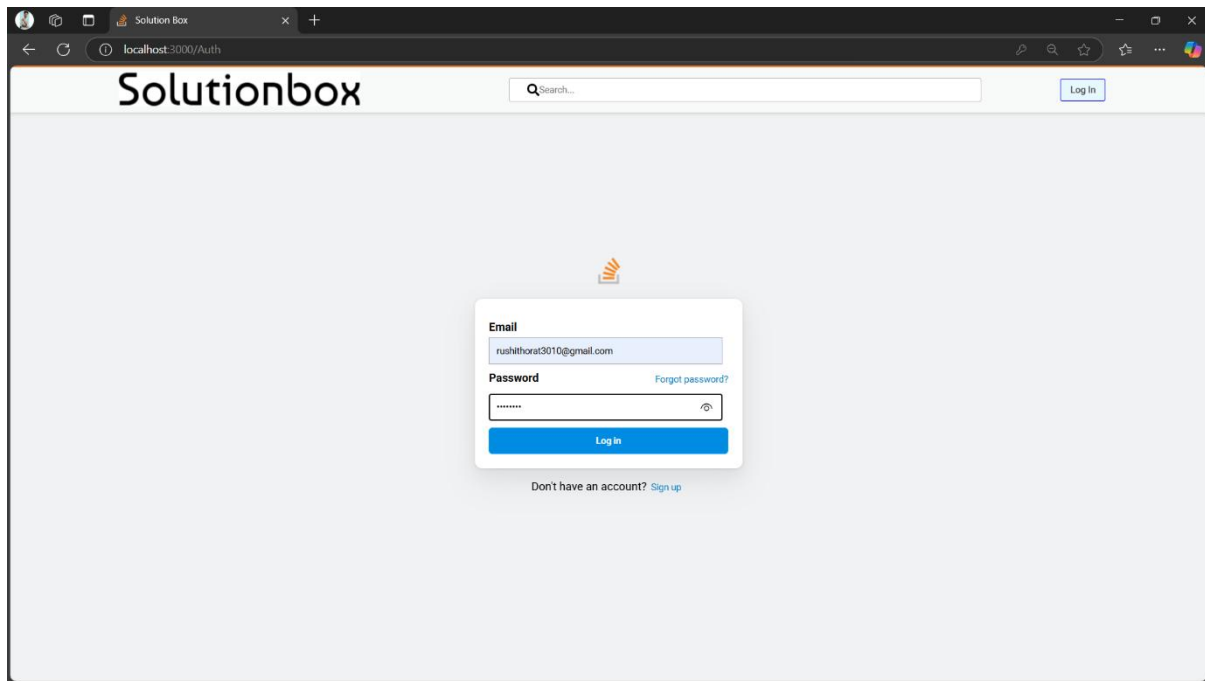
Field name	Data Type	Description	Key
_id	int	Unique ID for each vote entry	Primary
upVote	string	List or count of upvotes	
downVote	string	List or count of downvotes	
userId	int	ID of the user who cast the vote	Foreign (Users._id)

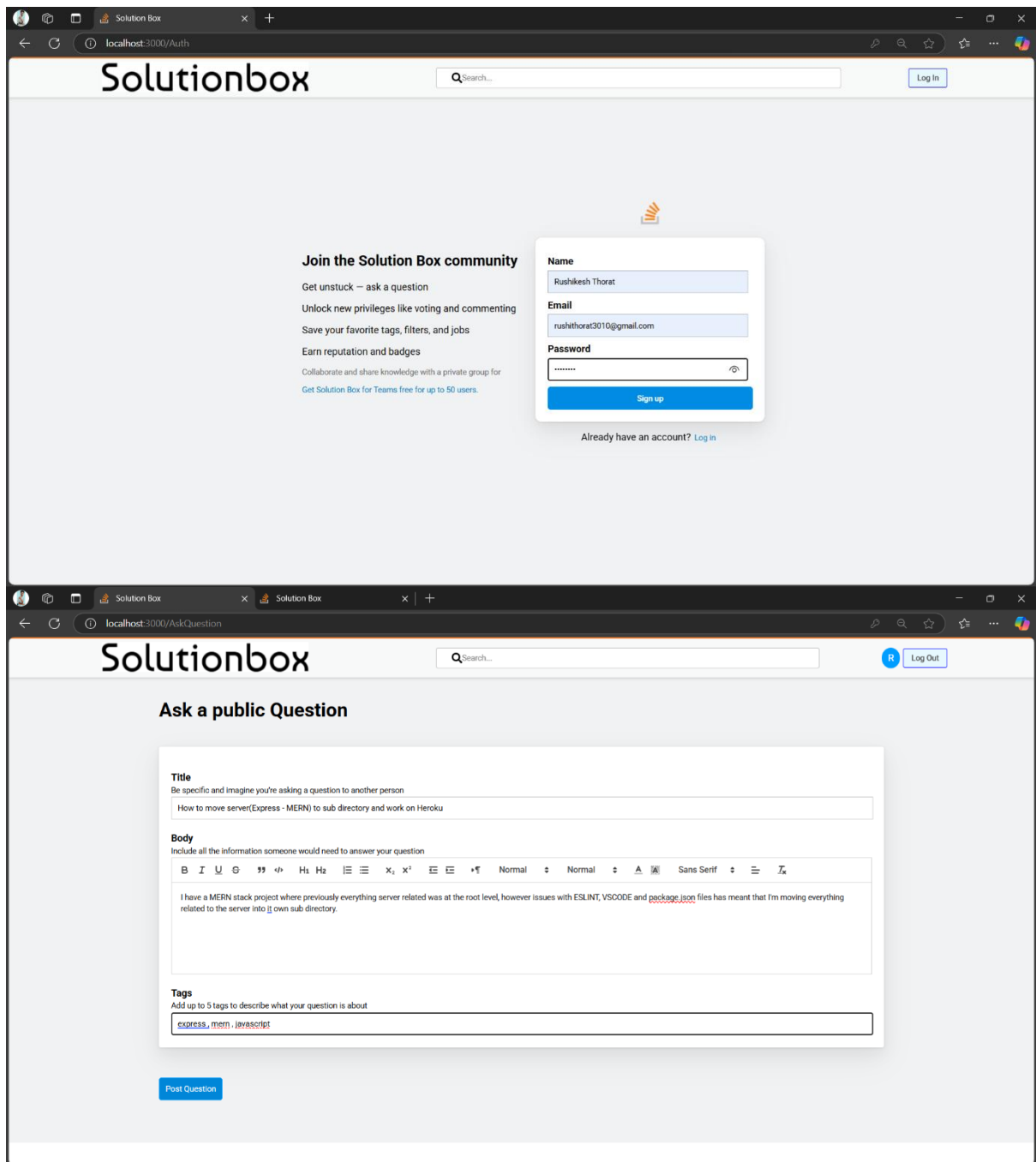
Answer Table:

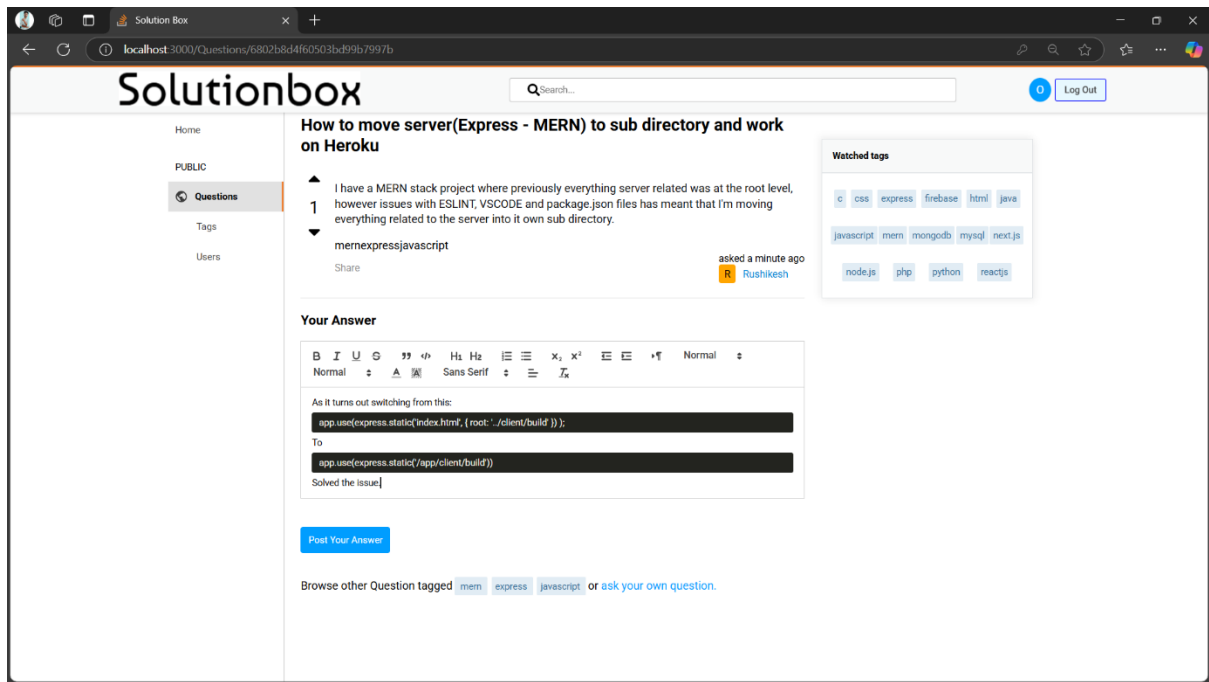
Field name	Data Type	Description	Key
_id	int	Unique ID for each answer	Primary
answerBody	string	Full content of the answer	
answerUser	string	Name or ID of the user who answered	

answerOn	string	Date the answer was posted	
questionId	int	ID of the related question	Foreign(Questions._id)

Sample input Output Screens:







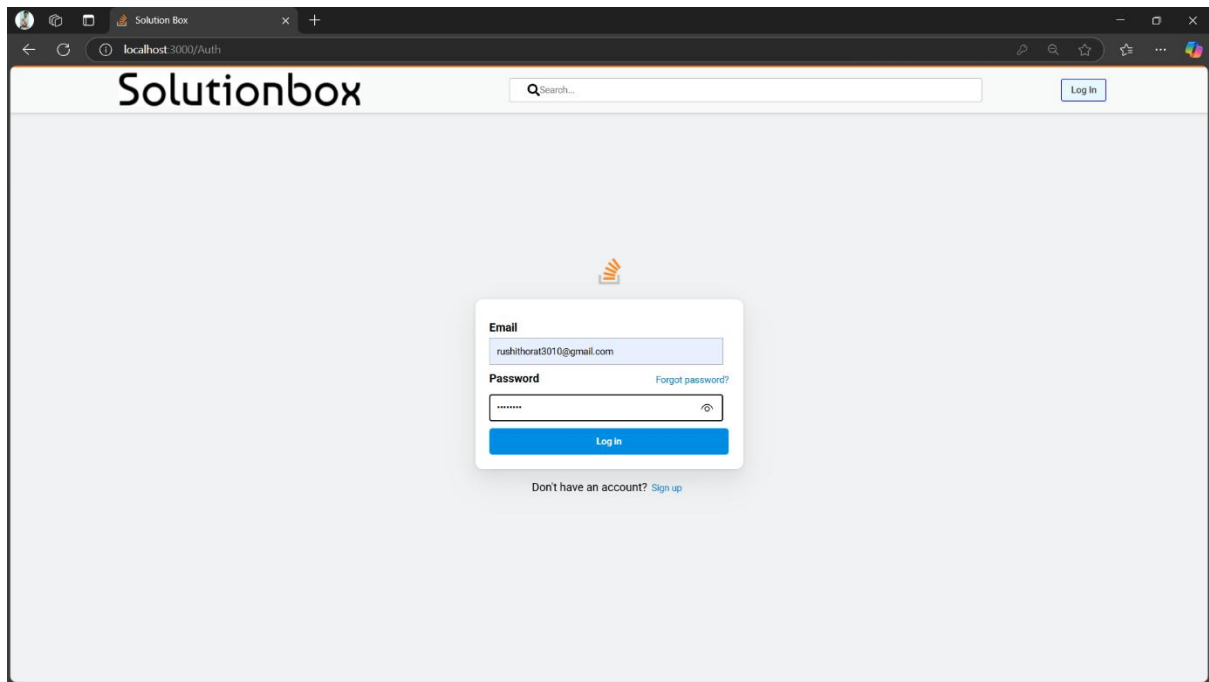
User Manual:

1 Login page:

The login page provides users with secure access by entering their email and password, with options to sign up or reset forgotten passwords.

- The Login Page of the SolutionBox platform is designed for simple, secure, and user-friendly authentication.
- Clean and centered login form with fields for email and password, including a show/hide password feature.

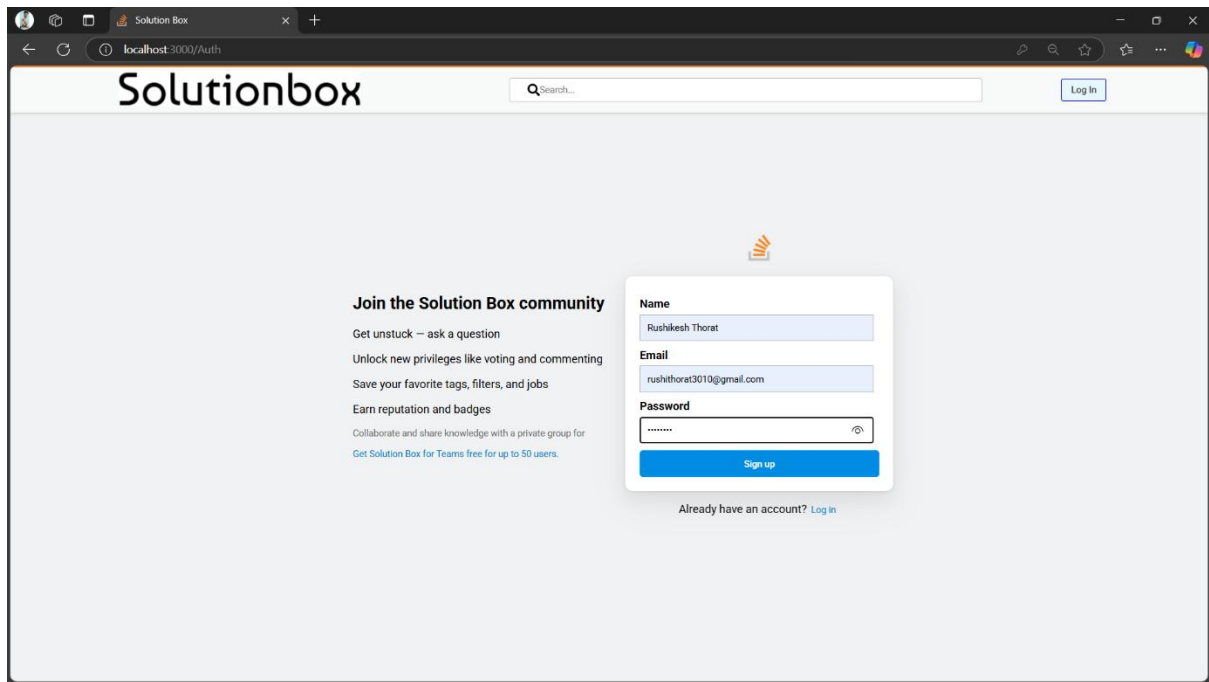
- Includes a " sign up " for new users to sign up.



2 Signup page:

The sign-up page enables new users to join the platform by filling out their name, email, and password, with features like password masking and easy access to login options.

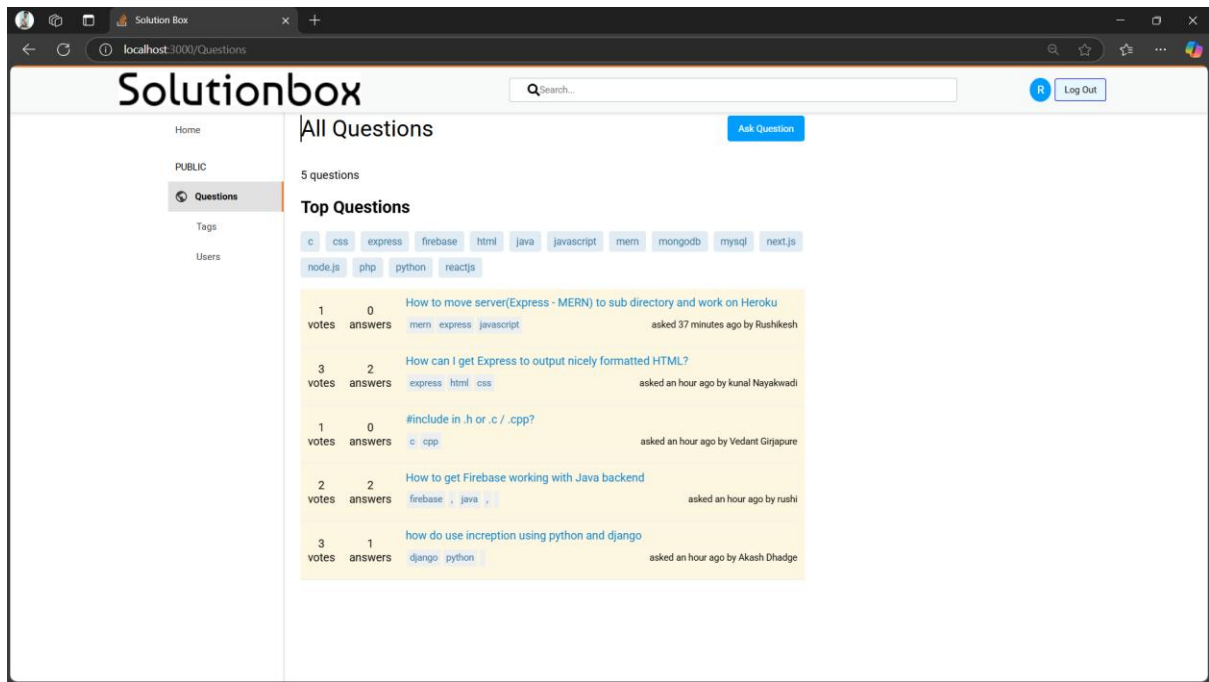
- The page features a **sign-up form** asking for the user's **Name**, **Email**, and **Password**, with a password visibility toggle.
- On the left side, there's a welcome message encouraging users to **join the community**, highlighting benefits like asking questions, saving tags, and earning reputation.
- Includes a **"Sign up"** button and a link to switch to the **login page** if the user already has an account.



3 Home page and question page:

The combined home and question page displays trending content and questions in one place, showcasing details like votes, answers, tags, and timestamps. It ensures easy navigation to other sections like tags and users for a seamless experience.

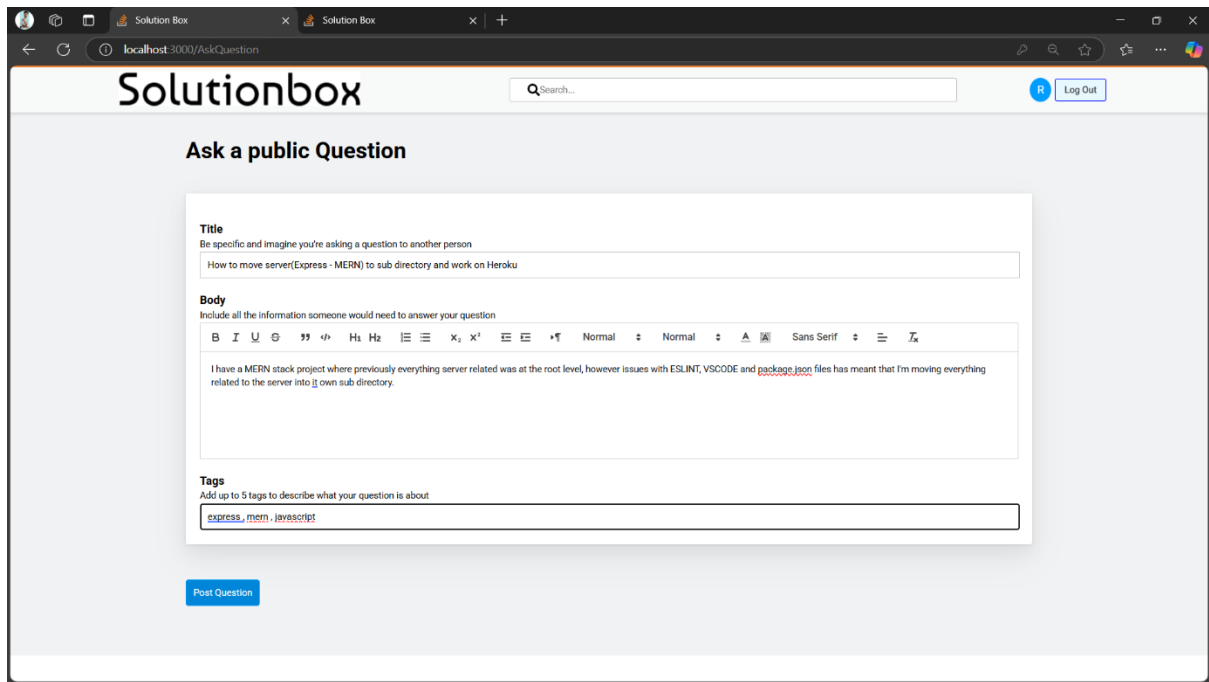
- Displays a list of top questions with vote and answer counts, tags, timestamps, and user names.
- Features a left sidebar with navigation options: Home, Questions, Tags, and Users.
- A search bar at the top allows users to find specific content, and there's a "Log Out" button for signed-in users.
- Users can filter content using tag chips, and view their watched tags on the right panel.
- Includes an "Ask Question" button to allow users to post new questions.



4 Ask Question:

The "Ask Question" page allows users to submit their questions by entering a title, body, and relevant tags. It ensures a clear and structured interface for posting questions publicly, along with a "Post Question" button to finalize submission.

- Users can enter a Title that clearly states their technical query.
- A rich text editor allows users to describe the issue in detail within the Body section.
- Tags input field helps categorize the question with relevant technologies (up to 5 tags).
- A "Post Question" button submits the query for public viewing and responses.



5 Answer Question:

The "Answer Question" page allows users to provide answers to questions by entering a detailed response in a text field. It also displays the question details, tags, and existing answers for reference. Users can submit their answers through a "Post Answer" button.

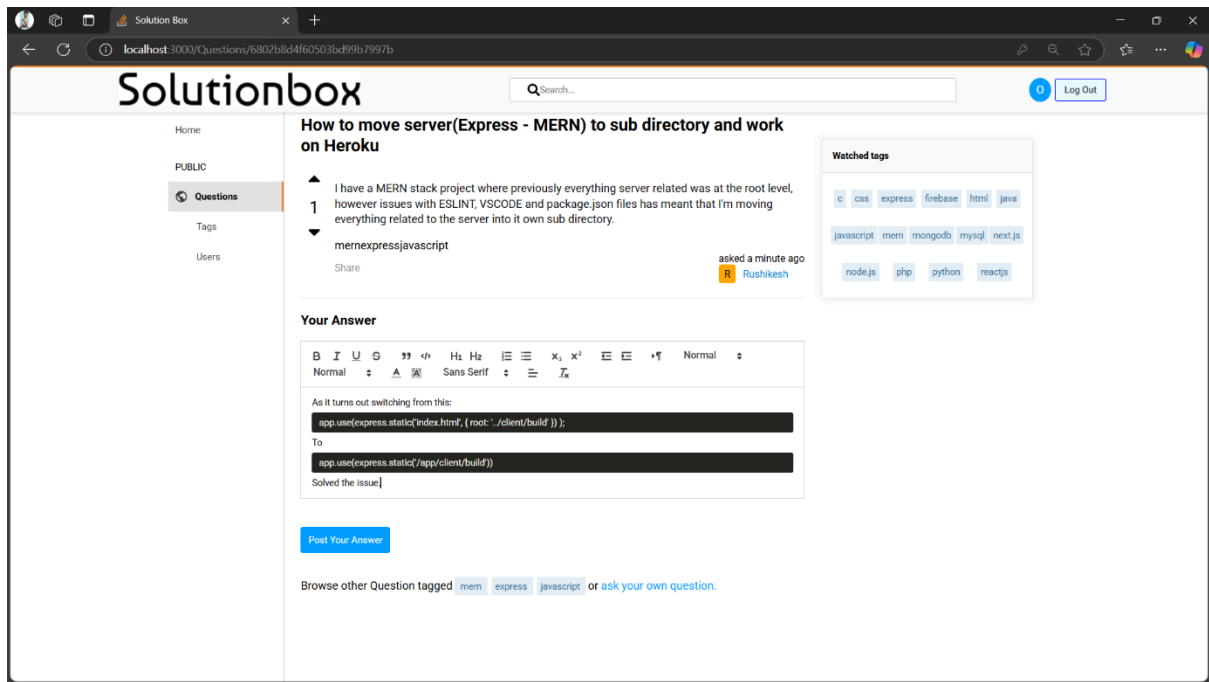
- The question is shown with:

- Title
- Description/body
- Tags
- User info (name & time posted)

- A vote system (upvote/downvote) is available next to each question.

- Below the question, users see a rich text editor to type their answers. It supports code formatting, headings, lists, and text styling.

- A "Post Your Answer" button submits the answer.

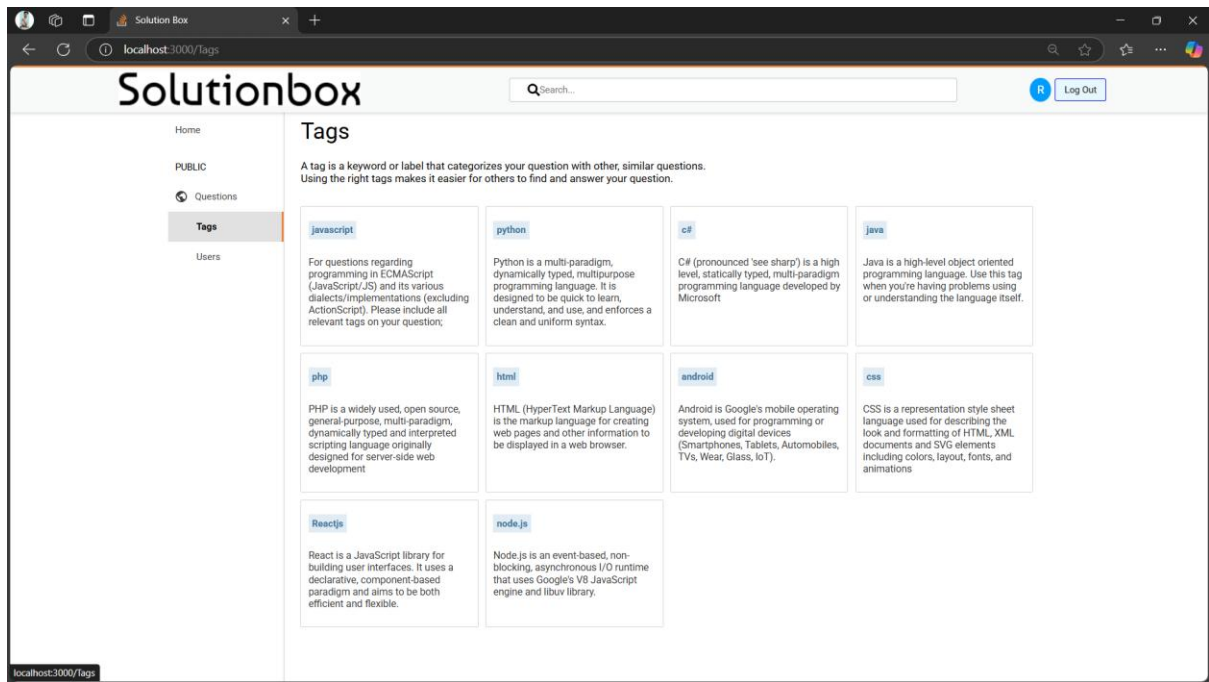


6 Tags page:

The tags page lists various programming languages and technologies, each accompanied by a brief description. It includes a search bar for easy navigation and allows users to explore topics such as JavaScript, Python, HTML, CSS, and more.

- Tags are shown as cards with:

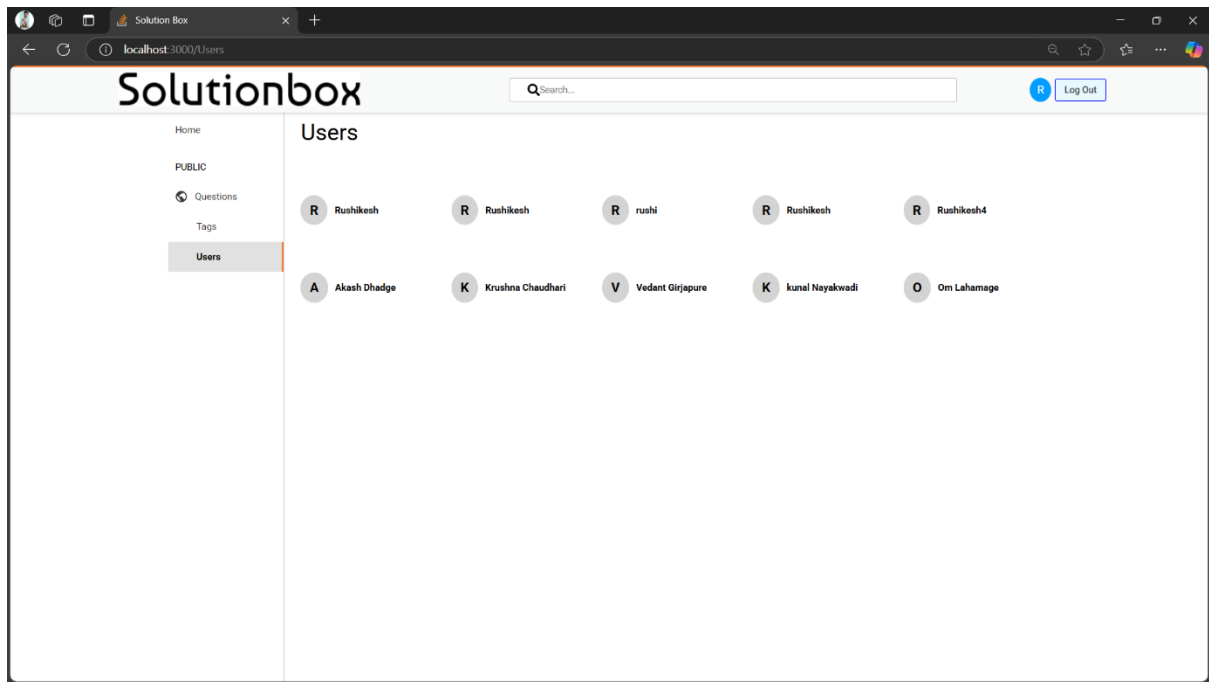
- Tag name (e.g., javascript)
- Short description



7 User page:

The user page showcases a list of all registered users along with their names and initials

- Displays the total number of users registered.
- Provides a visual representation of user icons.



8 Profile page:

The profile page allows users to view and manage their personal information, such as their name, avatar, and an "About" section for a short bio. It includes a "Tags Watched" section to display interests and options to edit the profile or log out. Navigation is streamlined with links to other sections like Home, Questions, and Users.

- Shows the user's name and avatar.
- Displays tags or topics the user follows (e.g., Java, MERN, Python).
- Features an "About" section for a personal bio
- Includes options to edit the profile and log out.

