

# **GENERAL GUIDELINES FOR PROJECT REPORT SUBMISSION**



**BAJAJ INSTITUTE OF TECHNOLOGY**

**WARDHA - 442 001 (INDIA)**

**2022- 23**

# **DISCUSSION FORUM for BIT**

*Project report submitted to  
Dr. Babasaheb Ambedkar Technological University, Maharashtra  
in partial fulfilment of the requirements for the award of  
the degree*

## **Bachelor of Technology In Computer Engineering**

*By*

**Purva Dhopade (2046491245034)  
Ashwini Ukhalkar (2046491245008)  
Shreya Chinchmalatpure (2046491245048)**

under the guidance of

**Urvashi Pote  
Assistant Professor**



**Department of Computer Engineering  
Bajaj Institute of Technology  
Wardha-442 001 (India)  
2022-23**

**BAJAJ INSTITUTE OF TECHNOLOGY, WARDHA**

**DEPARTMENT OF COMPUTER ENGINEERING**

**CERTIFICATE**



*This is to certify that the Project report titled*

***DISCUSSION FORUM for BIT***

*has been successfully completed*

*by*

***Purva Dhopade (2046491245034)***

***Ashwini Ukhalkar (2046491245008)***

***Shreya Chinchmalatpure (2046491245048)***

*in partial fulfilment of the requirements for the award of  
the degree in Bachelor of Technology, Computer Engineering*

**Urvashi Pote  
Assistant Professor**

Department of Computer Engineering  
Bajaj Institute of Technology, Wardha

**Prof. Sheetal Kale  
Head of the Department**  
Department of Computer Engineering  
Bajaj Institute of Technology, Wardha

**Dr. N. M. Kanhe  
Principal**  
Bajaj Institute of Technology,  
Wardha

**Date: 12/06/2023**

**Place: Wardha**



## DECLARATION

I, hereby declare that the project report titled “**Discussion Forum for BIT**” submitted by me to the Bajaj Institute of Technology, Wardha, in partial fulfilment of the requirement for the award of Degree of B. Tech. in Computer Engineering discipline is a record of bonafide project work carried out by me under the guidance of **Urvashi Pote(Assistant Professor)**.

I, further declare that this submission by the undersigned represents my original work and I have quoted the references where others ideas/words have been included. I understand any violation of the above will levy a disciplinary action on me/us.

I, further declare that the work reported in this project report has not been submitted either in-part or in-full for the award of any other degree in any other Institute or University.

Date :- 12/06/2023

Place :- Wardha

S. No.	Name of Student	PRN Number	Signature
1.	Purva Dhopade	2046491245034	
2.	Ashwini Ukhalkar	2046491245008	
3.	Shreya Chinchmalatpure	2046491245048	

## ACKNOWLEDGEMENT

I express my heart felt gratitude to Urvashi Pote, Assistant Professor, Department of Computer Engineering, Bajaj institute of Technology, Wardha, for his guidance and encouragement throughout the project work.

I would like to express my sincere gratitude to Prof. Sheetal Kale, HOD of Computer Engineering, Bajaj institute of Technology, Wardha, for her guidance, help and useful suggestions about the project.

I am extremely thankful to Dr. Narendra Kanhe, Principal, Bajaj Institute of Technology Wardha, for providing me with infrastructural facilities to work in, without which this work might not have been possible.

Date: 12/06/2023

Place: Wardha

S. No.	Name of Student	PRN Number	Signature
1.	Purva Dhopade	2046491245034	
2.	Ashwini Ukhalkar	2046491245008	
3.	Shreya Chinchmalatpure	2046491245048	

## ABSTRACT

*In recent years due to the increasing use of Internet, it has made online discussion forum a common part of educational institutes.*

*This project is an ideal web-based discussion platform which provides a medium to share knowledge about what you've learned from your own studies and experiences. It allows users to share their thoughts and opinions, ask questions, and gain insights from other members in the forum. It is a great way for people to connect and learn from one another.*

*The objective behind developing this forum is to analyse the characteristics, benefits, challenges, and impact of online discussion forums in various contexts. This project aims at developing online platform for students. Any user can post the doubts, topics and at the same time give reply to other uses doubts.*

*The findings reveal that online discussion forums play a vital role in fostering meaningful discussions and promoting knowledge exchange among peers. Participants reported benefits such as access to a wealth of information, networking opportunities, and the ability to connect with their peers. However, challenges such as information overload, moderation issues, and potential for misinformation were also identified.*

*Furthermore, the online discussion forum contributes to the development of critical thinking, problem-solving, and digital literacy skills among students. It serves as a repository of collective knowledge and experiences, providing a valuable resource for academic and personal growth.*

*There are various stockholders involved for the effective running of the system. The admin who will monitor and moderate the discussions. The faculties for fostering academic discussions, sharing resources, and coordinating curriculum-related activities. Students can participate in academic discussions beyond the classroom, seek clarifications on assignments, and collaborate on group projects.*

*Overall, this report highlights the significant benefits of an online discussion forum for a college campus. It emphasizes the importance of fostering communication and collaboration among administrators, faculties, and students, ultimately enhancing the overall educational experience and fostering a vibrant and connected college community.*

## TABLE OF CONTENTS

Chapter No.	Title	Page No.
	Certificates	i
	Declaration	ii
	Acknowledgement	iii
	Abstract	iv
	Table of Contents	v
	Abbreviations	vii
<b>1.</b>	<b>Introduction</b>	<b>1</b>
	1.1 Introduction	
	1.2 Objectives	
	1.3 Organization of report	
<b>2.</b>	<b>Literature Survey</b>	<b>3</b>
	2.1 Effectiveness of Using Online Discussion Forum	
	2.2 A Design-Based Study for Development of a Model for a Successful Online Discussion Forum	
	2.3 Promoting college students' systems thinking in asynchronous discussions: Encouraging students initiating questions	
	2.4 Outcomes of Chat and Discussion Board Use in Online Learning: A Research Synthesis	
	2.5 A Methodological Study on Online Discussion Forum	
<b>3.</b>	<b>Methodology / System Requirements and Specifications</b>	<b>5</b>
	3.1 Stakeholders	
	3.2 Functional Requirements	
	3.3 Non-functional Requirements	



<b>4.</b>	<b>Implementation of Project</b>	<b>9</b>
<b>5.</b>	<b>Results and Discussions</b>	<b>27</b>
<b>6.</b>	<b>Conclusions and Future Scope</b>	<b>33</b>
	<b>References</b>	<b>34</b>
	<b>Annexure A</b>	<b>35</b>
	Appendix A: Glossary	
	Appendix B: Analysis Models	

---



## ABBREVIATIONS

- *Discussion Forum*: Online Discussion Forum i.e Question and Answer Platform for the students of BIT.
- *Admin*: In this system, the HOD of each department.
- *Blogs*: Learning Material for students.
- *Upcoming Events*: The events held in the Institute like (Coding Contests, Technical events etc.)
- *Trending Tags*: Trending Questions on Discussion Forum.

# **CHAPTER 1**

## **INTRODUCTION**

---

### **1.1 Introduction**

Technology has become an important part of education in 21st century. All modern educational settings are constantly embracing different technological tools and platforms to enhance effectiveness in the teaching and learning process. The use of technology to support teaching and learning have become essential skills for teachers and the learners in today's world. Technology has strong influence on students reading habits, access to information, communication and creative thinking. Reports state that information and communication technology have profound impact on students' access to information.

Discussion forums can be used for many purposes, such as helping students to review material prior to an assignment or exam, engaging students in discussion of course material before coming to class, and reflecting on material that they have read or worked with outside of class.

The main aim of online group discussion forum is to create an interactive platform where students can collaborate, share ideas and engage in discussions with their peers and teachers. The admin can also use the system to monitor and moderate the discussions, provide feedback, and make announcements.

### **1.2 Objectives**

Following are the main objectives for having a discussion forum for the college:

- **Knowledge Sharing:** Users can exchange information, insights, and experiences on specific topics, leading to problem-solving.

- **Collaboration and Networking:** Discussion forums is an excellent platform for Collaboration and Networking among peers.
- **Access to Diverse Perspectives:** This diversity of perspectives enriches discussions, promotes critical thinking, and encourages participants to consider different viewpoints.
- **Continuous Learning:** Discussion forums provide opportunities for continuous learning beyond formal education.
- **Enhanced Problem Solving:** Discussion forums offer a platform to present problems or challenges and seek solutions collectively.
- **Peer Support:** Online discussion forums provide a space for individuals to seek and offer support to others facing similar situations or challenges.

Discussion Forum for BIT enables you to create an open and collaborative environment where members can exchange ideas and learn from each other. Three different stakeholders Admin, Faculties and students each of whom would play different roles while interacting with the system. Discussion Forum provides a space for members to share resources, such as links to articles, videos, and other materials. All the stakeholders will have the facility to ask questions on the forum and get the desired response. The user can also find previously answered questions with the help of tags. Weekly Blogs, Activities or events will be posted by admin and moderator for guiding students.

### **1.3 Organization of Report**

The report is organized as follows:

Chapter 1 presents Introduction and Objectives of Discussion Forum.

Chapter 2 presents the literature survey on Discussion Forum.

Chapter 3 explain system requirements and specification.

Chapter 4 displays the implementation of the project.

Chapter 5 display the results and discussion of the project.

Finally, conclude in Chapter 6 with possible future plan.

## **CHAPTER 2**

### **LITERATURE SURVEY**

---

#### **2.1 Effectiveness of Using Online Discussion Forum**

- This article discusses benefits of using online discussion forums and peer to peer learning for enhancing student learning techniques.
- It identifying the use of technology tools available in higher education and deploying them effectively in the classroom in order to improve student learning strategies.
- It also mentions how higher institutions have started adopting hybrid or blended models of education that combines classroom and distributed learning environments.
- Online discussion forums are considered an extension of traditional learning methods for knowledge construction and self-assessment.

#### **2.2 A Design-Based Study for Development of a Model for a Successful Online Discussion Forum**

- This article reviews how the current COVID-19 pandemic has revealed the curtains of many important tools that can be used for online learning.
- The use of Online discussion forums has been identified as an important activity for engaging students besides on the educational units or assignments.
- Many educational institutions have shifted to online learning as it helps in fostering the student's engagement and learning

#### **2.3 Promoting college students systems thinking in asynchronous discussions: Encouraging students initiating questions**

- Discussion forum support students' development of critical thinking. The forum makes ideas accessible to all learners through displaying ideas in a communal space.
- ODF helps in clarifying queries, linking different ideas, and widening discussions by asking more questions.
- Students engaged with progressive discussions in the forum, such as responding to the teachers' prompting questions by elaborating and extending ideas, reflecting upon their learning process.

## **2.4 Outcomes of Chat and Discussion Board Use in Online Learning: A Research Synthesis**

- Online Discussion Forums can be particularly for students, as these communication connections are often students only means of connecting with each other and sharing ideas in an open forum.
- For example, a study by Chapman and Stone (2007) showed that if a student who posted to the board received positive responses from other students, that student felt obligated to respond to those positive posts. This ultimately allowed students to interact in various ways via the discussion board.

## **2.5 A Methodological Study on Online Discussion Forum**

- The Online Discussion Forum system can be used for any sort of discussion. It is a web-based application. The goal of this system is to allow users to connect with each other online and to provide new users with discussion services. It enables users to sign up for our website. A registered user can create new posts and respond to existing ones.
- The current study uses two primary concepts to support the system at two distinct ends termed the front end and the back end to completely automate the Online Discussion Forum, removing all of the constraints of the manually managed Online Discussion Forum.

## CHAPTER 3

# METHODOLOGY/ SYSTEM REQUIREMENTS AND SPECIFICATIONS

---

### 3.1 Stakeholders

It is a web portal with three different stakeholders Admin, Faculties and students each of whom would play different roles while interacting with the system whose main objective is:

**Admin:**

- Students profile and number of users and topics discussed by them will be visible.
- The trending tags / questions will be visible i.e the number of questions solved by students.
- Admin will be able to post the upcoming events (coding & cultural).
- Will be able to make announcements.
- Action can be taken against inappropriate content.

**Faculty:**

- Create a new discussion topic.
- Provide answers to questions.
- The trending tags / questions will be visible i.e the number of questions solved by students.
- Able to post comments on an existing discussion topic.
- Faculty will be able to post weekly Blogs for guiding students.
- Delete inappropriate posts made by user.

**Students:**

- Register an account using their email and password.
- Post questions and give answers and receive feedback.
- Find previously answered questions with the help of tags.
- Students dashboard / profile which will contain the number of questions asked, answered, comments given, number of tags used in questions.



- Trending questions will be visible.
- Upcoming events will be visible to students.
- Like and dislike posts made by other users.

### **3.2 Functional Requirement**

REQ-1: Authenticate and Login user to the discussion forum.

REQ-2: New users can register in the forum.

REQ-3: A registered user is able to view his profile. Additional information, questions asked by him and answers posted by him are also displayed.

REQ-4: Users can view questions and their corresponding answers.

REQ-5: A logged in user can ask questions.

REQ-6: A logged in user can post answers to any questions.

REQ-7: Provide an interface for the admin to approve posts so that posts are not visible without admin approval

REQ-8: To enable a logged in user to upvote and downvote answers.

REQ-9: Users can search questions with the help of tags.

REQ-10: Users get to know about upcoming events, trending tags and can also view weekly blogs.

REQ-11: Admin will have authority to manage and monitor every matter happening in the discussion forum.

### **3.3 Non-functional Requirements**

Few of the non-functional requirements are listed below:

#### **Performance Requirements**

- The system must be interactive and the delays involved must be less .

- Smooth working of discussion forum is must to ensure good user interactions.
- Server will be working 24X7.

### **Safety and Security Requirements**

- Only the users(students, faculties and staff) who are registered within the institute can login to the discussion forum
- Information transmission should be securely transmitted to server without any changes in information.

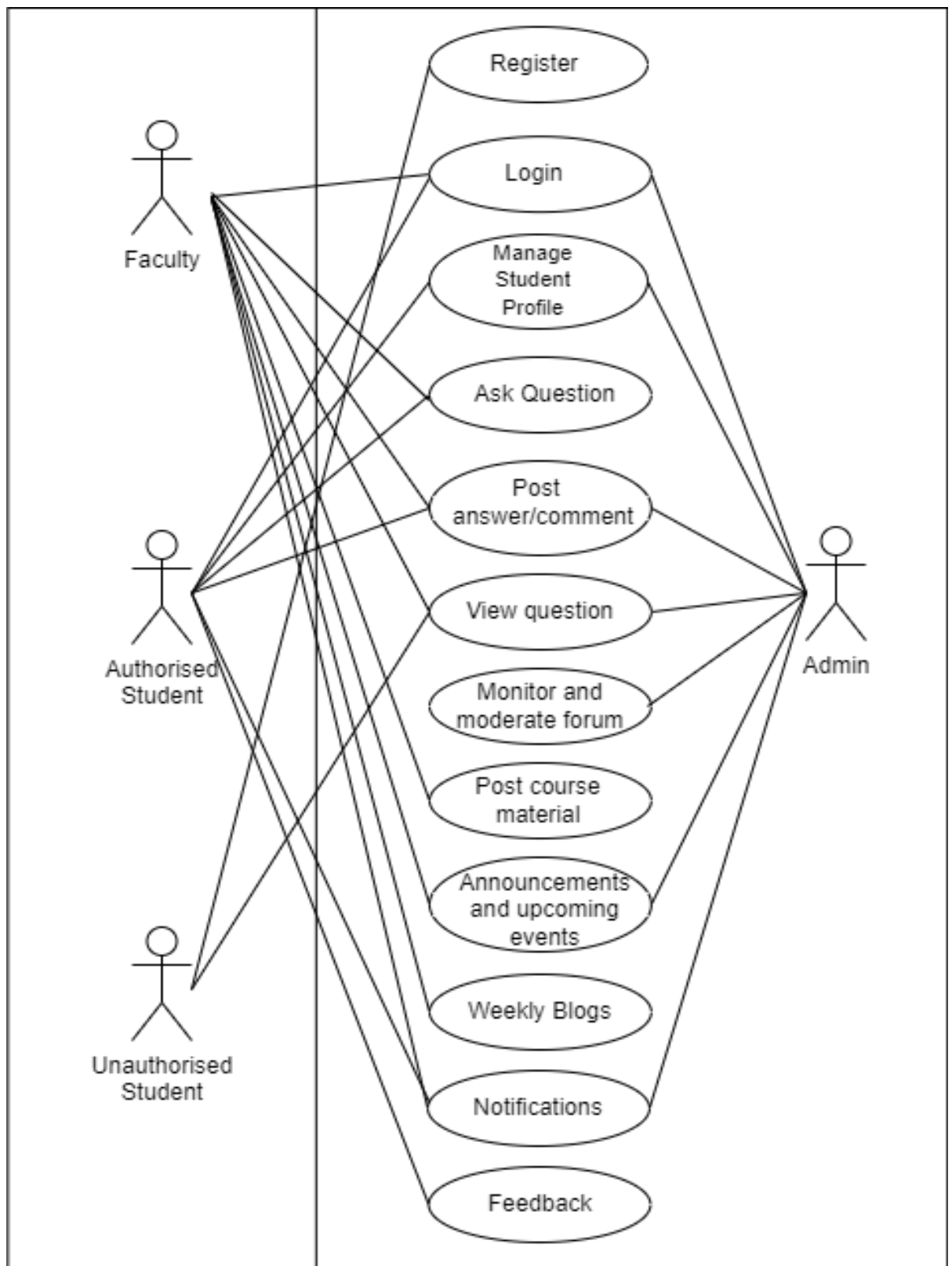
### **Security Requirements**

- Access to unauthorized users would be prevented.
- The system enables only a logged in user to ask questions and post answers.
- The content authorized by the admin can only be posted on the discussion forum.

### **Software Quality Attributes**

- As the system is easy to handle and navigates in expected way with no delays.
- The system functions accordingly and transverses quickly between its states.
- For security concern of users account, a proper login mechanism is used to avoid hacking.
- The System is user friendly.





**Fig. 3.1**  
**Use Case Diagram**

## CHAPTER 4

# IMPLEMENTATION OF PROJECT

---

### App.js

```
frontend > src > JS App.js > ...
1  import React from 'react';
2  import './App.css';
3  import Header from './components/Header/Header';
4  import { BrowserRouter as Router, Route, Routes } from 'react-router-dom'
5  import StackOverflow from './components/StackOverflow';
6  import Question from './components/Add-Question/Question';
7  import ViewQuestion from './components/ViewQuestion';
8  import Auth from './components/Auth';
9  import { useDispatch, useSelector } from 'react-redux';
10 import { login, logout, selectUser } from './features/userSlice';
11 import { auth } from './firebase';
12 import DBMS from './components/StackOverflow/DBMS';
13 import DS from './components/StackOverflow/DS';
14 import { Navigate } from 'react-router-dom';
15 import { useEffect } from 'react';
16 import AllUsers from './components/StackOverflow/AllUsers';
17 import MyProfile from './components/StackOverflow/MyProfile';
18 import UseEffectAPI from './components/StackOverflow/UseEffectAPI';
19 import NewUser from './components/StackOverflow/NewUser';
20
21 function App() {
22   const user = useSelector(selectUser);
23   const dispatch = useDispatch();
24   useEffect(() => {
25     auth.onAuthStateChanged((authUser) => {
26       if (authUser) {
27         dispatch(
28           login({
29             uid: authUser.uid,
30             photo: authUser.photoURL,
31             displayName: authUser.displayName,
32             email: authUser.email,
33           })
34         );
35       } else {
36         dispatch(logout());
37       }
38     });
39   }, [dispatch]);
```

```

39
40 const PrivateRoute = ({ component: Component, ...rest }) => (
41   <Route {...rest}
42     render={(props) =>
43       user ? (
44         <Component {...props} />
45       ) : (
46         <Navigate
47           to={{
48             pathname: "/",
49             state: {
50               from: props.location,},
51           }}
52         />)
53     />);
54
55 return (
56   <div className="App">
57     <Router>
58       <Header />
59       <Routes>
60         <Route path="/" element={<Auth />} />
61         <Route path="/stackoverflow" element={<StackOverflow />} />
62         <Route exact path="/add-question" element={<Question />} />
63         <Route exact path="/question" element={<ViewQuestion />} />
64         <Route exact path="/DBMS" element={<DBMS />} />
65         <Route exact path="/DS" element={<DS />} />
66         <Route exact path="/Board" element={<Board />} />
67         <Route exact path="/Newboard" element={<Newboard />} />
68         <Route exact path="/AllUsers" element={<AllUsers />} />
69         <Route exact path="/Myprofile" element={<MyProfile />} />
70         <Route exact path="/User" element={<UseEffectAPI />} />
71         <Route exact path="/NewUser" element={<NewUser />} />
72       </Routes>
73     </Router>
74   </div>
75   );}
76 export default App;
77

```



## Auth.js

```
frontend > src > components > Auth > index.js > index
71     return (
72       <div className="auth">
73         <div className="auth-container">
74           <div className="auth-login">
75             <div>
76
77             </div>
78
79           <div className="auth-login-container">|
80             <div className="sign-options">
81               <div onClick={handleSignInGoogle} className="single-option">
82                 
86                 <p>Login with google</p>
87               </div><br/>
88               <div className="single-option">
89                 
94                 <p>Login with Github</p>
95               </div>
96             </div>
97           </div>
98
99           {
100             register ? (<
101
102               
106               <div className="message">
107                 <p>A place to share knowledge and better understand concepts</p>
108               </div>
109
110
111               <div className="input-field">
112                 <p>Username</p>
113                 <input value={username} onChange={(e) => setUsername(e.target.value)} type="text" />
114               </div>
115
116               <div className="input-field">
117                 <p>Email</p>
118                 <input value={email} onChange={(e) => setEmail(e.target.value)} type="email" />
119               </div>
120               <div className="input-field">
121                 <p>Password</p>
122                 <input value={password} onChange={(e) => setPassword(e.target.value)} type="password" />
123               </div>
124               <button className="login-button" onClick={handleRegister}
125                 disabled={loading}
126                 style={{ marginTop: "10px", }}>
127                 {loading ? 'Registering...' : 'Register'}
128                 {/* Register */}
129               </button>
130             </>
131           ) : (
132             <>
133
134               
138               <div className="message">
139                 <p>A place to share knowledge and better understand concepts</p>
140
```





```

137 />
138 <div className='message'>
139   <p>A place to share knowledge and better understand concepts</p>
140
141 </div>
142 <div className="input-field">
143   <p>Email</p>
144   <input value={email} onChange={(e) => setEmail(e.target.value)} type="email" />
145 </div>
146 <div className="input-field">
147   <p>Password</p>
148   <input value={password} onChange={(e) => setPassword(e.target.value)} type="password" />
149 </div>
150 <button className='login-button' onClick={handleSignIn}
151   disabled={loading}
152   style={{ marginTop: "10px", }}>
153
154   {loading ? 'Signing In.....' : "Login"}
155   { /* login */ }
156 </button>
157
158 </>
159
160
161 <p
162   onClick={() => setRegister(!register)}
163   style={{ marginTop: "15px", marginRight: "188px", textAlign: "right", color: "#0005ff", textDecoration: "underline", cursor: "pointer", }}>
164   {register ? "Login" : "Register"} </p>
165
166 </div>
167 </div>
168 {error !== "" && (
169   <p
170     style={{
171       color: "red",
172       fontSize: "14px",
173     }}
174   >
175     {error}
176   </p>
177 )}
178 </div>
179 </div>
180
181 }
182
183 export default Index

```



## Index.js

```
frontend > src > components > StackOverflow > JS index.js > Index
1 import React, { useState } from 'react'
2 import './css/index.css'
3 import Main from './Main';
4 import Sidebar from './Sidebar'
5 import RightSidebar from './RightSidebar';
6 import axios from 'axios';
7
8 function Index() {
9
10   const [questions, setQuestions] = useState([]);
11
12   React.useEffect(() => {
13     async function getQuestion() {
14       await axios.get("/api/question").then((res) => {
15
16         console.log(res.data);
17         setQuestions(res.data.reverse());
18       }).catch((err) => {
19         console.log(err);
20       });
21     }
22     getQuestion();
23   }, []);
24
25   return (
26     <div className="stack-index">
27       <div className="stack-index-content">
28         <Sidebar/>
29         <Main questions={questions}/>
30         <RightSidebar/>
31       </div>
32     </div>
33   );
34 }
35
36 export default Index;
37
```



## AllQuestion.js

```
frontend > src > components > StackOverflow > AllQuestions.js > AllQuestions > AllTags
1 import { Avatar } from '@mui/material'
2 import React from 'react'
3 import { Link } from 'react-router-dom';
4 import './css/AllQuestion.css';
5 import convertHtmlToReact from '@hedgeoc/html-to-react';
6
7 function AllQuestions({ question }) {
8   let tags = JSON.parse(question?.tags[0])
9   function truncate(str, n) {
10     return str?.length > n ? str.substr(0, n - 1) + "...": str
11   }
12
13   return (
14     <div className="all-questions">
15       <div className="all-questions-container">
16         <div className="all-questions-left">
17           <div className="all-options">
18             <div className="all-option">
19               <p>0</p>
20               <span>votes</span>
21             </div>
22             <div className="all-option">
23               <p>{question?.answerDetails?.length}</p>
24               <span>answers</span>
25             </div>
26             <div className="all-option">
27               <p>2</p>
28               <span>Views</span>
29             </div>
30           </div>
31         </div>
32         <div className="question-answer">
33           <Link to={` /question?q=${question?._id}` }>{question?.title}</Link>
34           <div
35             style={{
36               maxWidth: "90%",
37               // display: "flex",
38             }}>
39             <div>{convertHtmlToReact(truncate(question?.body, 200))}</div>
40           </div>
41           <div
42             style={{
43               display: "flex",
44             }}>
45             {tags.map((_tag) => (
46               <span className="question-tags">{_tag}</span>
47             ))}
48           </div>
49           <div className="author">
50             <small>{new Date(question?.created_at).toLocaleString()}</small>
51             <div className="author-details">
52               <Avatar src={question?.user?.photo} />
53               <p>{question?.user?.displayName} ? question?.user?.displayName : String(question?.user?.email).split('@')[0]</p>
54             </div>
55           </div>
56         </div>
57       </div>
58     </div>
59   );
60 }
61
62
63 export default AllQuestions
```



## Question.js

```
frontend > src > components > Add-Question > # Question.js > Question
1 import React, { useState } from 'react'
2 import ReactQuill from "react-quill";
3 import "react-quill/dist/quill.snow.css"; //quill's css
4 import { TagsInput } from "react-tag-input-component";
5 import './Question.css';
6 import { useNavigate } from 'react-router-dom';
7 import { useSelector } from 'react-redux';
8 import { selectUser } from '../features/userSlice';
9 import axios from 'axios';
10
11 function Question() {
12   const user = useSelector(selectUser)
13   const [loading, setLoading] = useState(false)
14   const [title, setTitle] = React.useState("")
15   const [body, setBody] = React.useState("")
16   const [tags, setTags] = React.useState([])
17   const navigate = useNavigate();
18   const handleQuill = (value) => {
19     setBody(value)
20   }
21   const handleSubmit = async (e) => {
22     e.preventDefault();
23     if (title !== "" && body !== "") {
24       const bodyJSON = {
25         title: title,
26         body: body,
27         tag: JSON.stringify(tags),
28         user: user,
29       }
30       await axios.post("/api/question", bodyJSON).then((res) => {
31         alert("Question added successfully");
32         setLoading(false);
33         navigate("/stackoverflow");
34       })
35       .catch((err) => {
36         console.log(err);
37         setLoading(false);
38       });
39     }
40   };
41
42   return (
43     <div className="add-question">
44       <div className="add-question-container">
45         <div className="head-title">
46           <h1>Ask a public question</h1>
47         </div>
48         <div className="question-container">
49           <div className="question-options">
50             <div className="question-option">
51               <div className="title">
```





```

50     <div className="question-option">
51       <div className="title">
52         <h3>Title</h3>
53         <small>
54           Be specific and imagine you're asking a question to another
55           person
56         </small>
57         <input value={title} onChange={(e) => setTitle(e.target.value)}
58           type="text"
59           placeholder="e.g Is there an R function for finding teh index of an element in a vector?"
60       </div>
61     </div>
62   </div>
63   <div className="question-option">
64     <div className="title">
65       <h3>Body</h3>
66       <small>
67         Include all the information someone would need to answer your
68         question
69       </small>
70       <ReactQuill value={body} onChange={handleQuill}
71         className="react-quill"
72         theme="snow"
73       </div>
74     </div>
75   </div>
76   <div className="question-option">
77     <div className="tag">
78       <h3>Tags</h3>
79       <small>
80         Add up to 5 tags to describe what your question is about
81       </small>
82       <div className="tags-input-container">
83         <TagsInput
84           value={tags}
85           onChange={setTags}
86           name="tags"
87           placeholder="press enter to add new tag"/>
88       </div>
89     </div>
90   </div>
91 </div>
92 </div>
93 <button disabled={loading} type="submit" onClick={handleSubmit} className="button">
94   {loading ? "Adding question....." : "Add your question"}
95 </button>
96 </div>
97 </div>
98 )
99 }
100 export default Question

```



## User.js

```
frontend > src > components > StackOverflow > JS AllUsers.js > AllUsers
1 import React from 'react'
2 import Sidebar from '../StackOverflow/Sidebar.js';
3 import RightSidebar from '../StackOverflow/RightSidebar';
4 import './css/RightSidebar.css';
5 import './css/AllUsers.css'
6 import './css/MyProfile.css'
7
8 function AllUsers() {
9   return (
10     <div className='AllUser-main'>
11       <div className="Left"><Sidebar/></div>
12       <div className='Middle'>
13         <div className='top-left'>
14           <div className='image-profile'>
15             </img>
16           </div>
17           <div className='Profile-details'>
18             <p>Ashwini Ukhalkar</p>
19             <p>User created: 1 month ago</p>
20           </div>
21           <div className='count'>
22             <div className="details">
23               <p>1</p>
24               <span>Answers</span>
25             </div>
26             <div className="details">
27               <p>2</p>
28               <span>Questions</span>
29             </div>
30             <div className="details">
31               <p>0</p>
32               <span>Comments</span>
33             </div>
34             <div className="details">
35               <p>4</p>
36               <span>Tags</span>
37             </div>
38           </div>
39         </div>
40       </div>
41       <div className='Right'><RightSidebar/></div>
42     </div>
43   )
44 }
45
46 export default AllUsers
```



## Main.js

```
frontend > src > components > StackOverflow > JS Main.js > ...
1 import React from 'react'
2 import { Link } from "react-router-dom";
3 import { FilterList } from "@mui/icons-material";
4 import AllQuestions from './AllQuestions';
5 import "../css/Main.css";
6 import { useNavigate } from 'react-router-dom';
7 function Main({questions}) {
8   const navigate = useNavigate()
9   return (
10     <div className="main">
11       <div className="main-container">
12         <div className="main-top">
13           <h2>All Questions</h2>
14           <button onClick={() => navigate('/add-question')}>Ask Question</button>
15         </div>
16         <div className="main-desc">
17           <p>{questions && questions.length} Questions</p>
18           <div className="main-filter">
19             <div className="main-tabs">
20               <div className="main-tab">
21                 <Link>Newest</Link>
22               </div>
23               <div className="main-tab">
24                 <Link>Active</Link>
25               </div>
26               <div className="main-tab">
27                 <Link>More</Link>
28               </div>
29             </div>
30             <div className="main-filter-item">
31               <FilterList/>
32               <p>Filter</p>
33             </div>
34           </div>
35         </div>
36         <div className="questions">
37           {
38             questions?.map((_q, index) => (<><div key = {index} className="question">
39               <AllQuestions question = {_q}/>
40             </div></>))
41           }
42         </div>
43       </div>
44     </div>
45   )
46 }
47 export default Main
```



## AllUser.js

frontend > src > components > StackOverflow > UseEffectAPI.js > UseEffectAPI > getUsers

```
1 import React, {useEffect} from 'react'
2 import { useState } from 'react';
3 import './css/UseEffect.css'
4
5 const UseEffectAPI = () => {
6   const [users, setUsers] = useState([]);
7   const getUsers = async () => {
8     const response = await fetch('https://api.github.com/users');
9     console.log(response);
10    setUsers(await response.json());
11  }
12
13  useEffect(() => {
14    getUsers();
15  }, []);
16
17  return (
18    <>
19    <h2>List of Users</h2>
20    <div className='container-fluid mt-5'>
21      <div className='row text-center'>
22        {
23          users.map((curElem) => {
24            return (
25              <div className='col-10 col-md-4 mt-5' key = {curElem.id}>
26                <div className='card'>
27                  <div className="d-flex align-items-center">
28                    <div className="image"> <img src={curElem.avatar_url} className="rounded" width="110" /> </div>
29                    <div className="ml-3 w-100">
30                      <h4 className="mb-0 mt-0 textLeft">{curElem.login}</h4> <span className="textLeft">{curElem.title}</span>
31                      <div className="p-2 mt-2 bg-primary d-flex justify-content-between rounded text-white stats">
32                        <div className="d-flex flex-column"> <span className="articles">Questions</span> <span className="number1">38</span></div>
33                        <div className="d-flex flex-column"> <span className="followers">Answers</span> <span className="number2">980</span></div>
34                        <div className="d-flex flex-column"> <span className="rating">Comments</span> <span className="number3">8.9</span> </div>
35                      </div>
36                    </div>
37                  </div>
38                </div>
39              </div>
40            )
41          })
42        }
43      </div>
44    </div>
45    </>
46  )
47
48  export default UseEffectAPI
```





## Backend code :-

### Models > Answer.js

```
backend > models > JS Answer.js > [?] <unknown>
1  const mongoose = require('mongoose')
2  const answerSchema = new mongoose.Schema({
3    question_id: {
4      type: mongoose.Schema.Types.ObjectId,
5      ref: "Questions",
6    },
7    answer: String,
8    created_at: {
9      type: Date,
10     default: Date.now(),
11   },
12   user: Object,
13   comment_id: {
14     type: mongoose.Schema.Types.ObjectId,
15     ref: "Comments",
16   },
17 });
18
19 module.exports = mongoose.model("Answers", answerSchema);
```

### Models > Question.js

```
backend > models > JS Question.js > [?] questionSchema
1  const mongoose = require('mongoose')
2  const questionSchema = new mongoose.Schema({
3    title: String,
4    body: String,
5    tags: [],
6    created_at: {
7      type: Date,
8      default: Date.now()
9    },
10   user: Object,
11   comment_id: {
12     type: mongoose.Schema.Types.ObjectId,
13     ref: "Comments"
14   }
15 })
16
17 module.exports = mongoose.model("Questions", questionSchema);
```



## Models > Comment.js

backend > models > JS Comment.js > ...

```
1  const mongoose = require("mongoose");
2  const commentSchema= new mongoose.Schema({
3    question_id: {
4      type: mongoose.Schema.Types.ObjectId,
5      ref: "Questions",
6    },
7    comment: String,
8    created_at: {
9      type: Date,
10     default: Date.now(),
11   },
12   user: Object,
13 });
14 module.exports = mongoose.model("Comments", commentSchema);
```

## Routers > Answer.js

backend > routers > JS Answers.js > ...

```
1  const express = require("express");
2  const router = express.Router();
3  const AnswerDB = require("../models/Answer");
4
5  router.post('/', async (req, res) => {
6    const answerData = new AnswerDB ({
7      question_id: req.body.question_id,
8      answer: req.body.answer,
9      user: req.body.user,
10    });
11
12    await answerData.save().then((doc) => {
13      res.status (201).send({
14        status: true,
15        data: doc,
16      });
17    }).catch((err) => {
18      res.status(400).send({
19        status: false,
20        message: "Error while adding answer",
21      });
22    });
23  });
24
25  module.exports = router;
```



## Routers > Comment.js

```
backend > routers > JS Comment.js > ...
1  const express = require("express");
2  const router = express.Router();
3  const CommentDB = require("../models/Comment");
4
5  router.post('/:id', async (req, res) => {
6    try{
7      await CommentDB.create({
8        question_id: req.params.id,
9        comment: req.body.comment,
10       user: req.body.user,
11     })
12     .then((doc) => {
13       res.status(201).send({
14         status: true,
15         message: "Comment added successfully",
16       });
17     }).catch((err) => {
18       res.status(400).send({
19         status: false,
20         message: "Error while adding comment",
21       });
22     });
23   }catch (err) {
24     res.status(500).send({
25       status: false,
26       message: "Error while adding comment",
27     });
28   }
29 });
30
31 module.exports = router;
```



## Routers > index.js

backend > routers > JS index.js > ...

```
1  const express = require('express');
2  const router = express.Router();
3  const questionRouter = require('./Question');
4  const answerRouter = require('./Answers')
5  const commentRouter = require('./Comment')
6
7  router.get('/', (req, res) => {
8    res.send('Welcome to stackoverflow clone tutorial api');
9  });
10 router.use('/question', questionRouter);
11 router.use('/answer', answerRouter);
12 router.use('/comment', commentRouter);
13
14 module.exports = router;
```

## DB.js

backend > JS db.js > connect > connect

```
1  const mongoose = require('mongoose')
2  const express = require("express");
3  const app = express();
4
5  const url = "mongodb+srv://ashwiniukhalkar:AVNNV4B71iP12976@bit-forum.cygwj6g.mongodb.net/?retryWrites=true&w=majority";
6  module.exports.connect = () => {
7    mongoose.connect(url).then((res) => console.log("MongoDB is connected successfully")).
8      catch((err) => console.log('Error: ',err))
9  }
10 require("./UserDetails");
11 const User = mongoose.model("test");
12 app.get("/getAllUser", async(req, res)=>{
13   try {
14     const allUser = await User.find({});
15     res.send({ status: "ok", data: allUser});
16   } catch (error) {
17     console.log(error);
18   }
19 });
```





## Routers > Question.js

```
backend > routers > # Question.js > ...
1  const express = require("express");
2  const mongoose = require("mongoose");
3  const router = express.Router();
4  const QuestionDB = require("../models/Question");
5  router.post("/", async (req, res) => {
6    const questionData = new QuestionDB({
7      title: req.body.title,
8      body: req.body.body,
9      tags: req.body.tag,
10     user: req.body.user,
11   });
12   await questionData.save().then((doc) => {
13     res.status(201).send({
14       status: true,
15       data: doc,
16     });
17   });
18   .catch((err) => {
19     res.status(400).send({
20       status: false,
21       message: "Error adding question",
22     });
23   });
24 });
25 router.get("/", async (req, res) => {
26   QuestionDB.aggregate ([
27     {
28       $lookup: {
29         from: "comments",
30         let: { question_id: "$_id" },
31         pipeline: [
32           {
33             $match: {
34               $expr: {
35                 $eq: ["$question_id", "$$question_id"],
36               },
37             },
38           },
39           {
40             $project: {
41               _id: 1,
42               comment: 1,
43               created_at: 1,
44             },
45           },
46         ],
47         as: "comments",
48       },
49     },
50     {
51       $lookup: {
52         from: "answers",
53         let: { question_id: "$_id" },
54         pipeline: [
55           {
```



```

backend > routers > # Question.js > ...
55     {
56         $match: {
57             $expr: {
58                 $eq: ["$question_id", "$$question_id"],
59             },
60     },
61     {
62         $project: {
63             _id: 1,
64         },
65     },
66     ],
67     as: "answerDetails",
68 },
69 },
70 {
71     $project: {
72         __v: 0,
73     },
74 },
75 ])
76 .exec()
77 .then((questionDetails) => {
78     res.status(200).send(questionDetails);
79 })
80 .catch((e) => {
81     console.log("Error: ", e);
82     res.status(400).send(e);
83 });
84 });
85 router.get("/:id", async (req, res) => {
86     try{
87         QuestionDB.aggregate([
88             {
89                 $match: { _id: new mongoose.Types.ObjectId(req.params.id) },
90             },
91             {
92                 $lookup: {
93                     from: "answers",
94                     let: { question_id: "$_id" },
95                     pipeline: [
96                         {
97                             $match: {
98                                 $expr: {
99                                     $eq: ["$question_id", "$$question_id"],
100                                 },
101                             },
102                         },
103                         {
104                             $project: {
105                                 _id: 1,
106                                 user: 1,
107                                 answer: 1,
108                                 // created_at: 1,
109                                 question_id: 1,
110                                 created_at: 1,
111                             },
112

```



## server.js

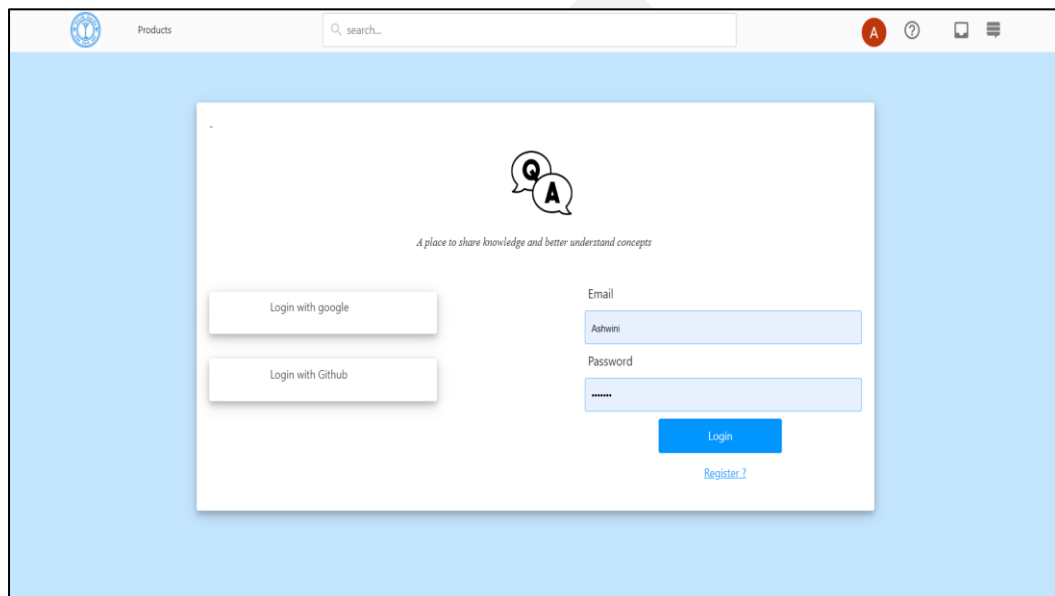
```
backend > JS server.js > ...
1  const express = require("express");
2  const cors = require("cors");
3  const path = require("path");
4  const app= express()
5  const bodyParser = require("body-parser");
6  const db = require('./db')
7  const PORT = process.env.PORT || 80;
8  const router = require("./routers")
9
10 //db connection
11 db.connect();
12 //middleware
13 app.use(bodyParser.json({ limit: "500mb" }));
14 app.use(bodyParser.urlencoded({ extended: true, limit: "500mb" }));
15 app.use(express.json())
16 //headers
17 app.use((req, res, next) => {
18   res.header("Access-Control-Allow-Origin", "*");
19   res.header("Access-Control-Allow-Headers", "*");
20   next();
21 });
22 // api
23 app.use('/api',router)
24 //static resources
25 app.use("/uploads", express.static(path.join(__dirname, "../uploads")));
26 app.use(express.static(path.join(__dirname, "../frontend/build")));
27 app.get("*", (req, res) => {
28   try {
29     res.sendFile(path.join(`${__dirname}/../frontend/build/index.html`));
30   } catch (e) {
31     res.send("Oops! error occurred");
32   }
33 })
34 //cors
35 app.use(cors())
36 //server listen
37 app.listen(PORT, () => {
38   console.log(`Stack Overflow Clone API is running on PORT No- ${PORT}`);
39 });
```



## CHAPTER 5

### RESULTS AND DISCUSSIONS

---

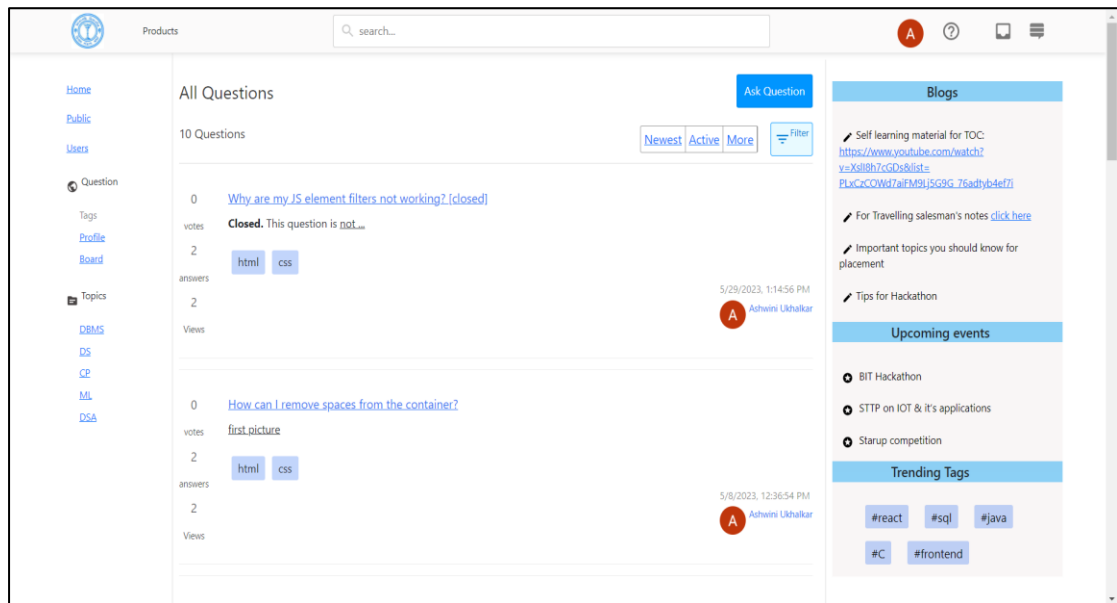


**Fig. 5.1**  
**Login of system**

This Page contains the login for the users. The user can also login using Google and GitHub. The user can also register.







**Fig. 5.2**  
**Frontpage of system**

This is the frontpage of the system. Here the user gets to see all the questions posted by different users. The sidebar contains the Blogs, upcoming events and trending tags

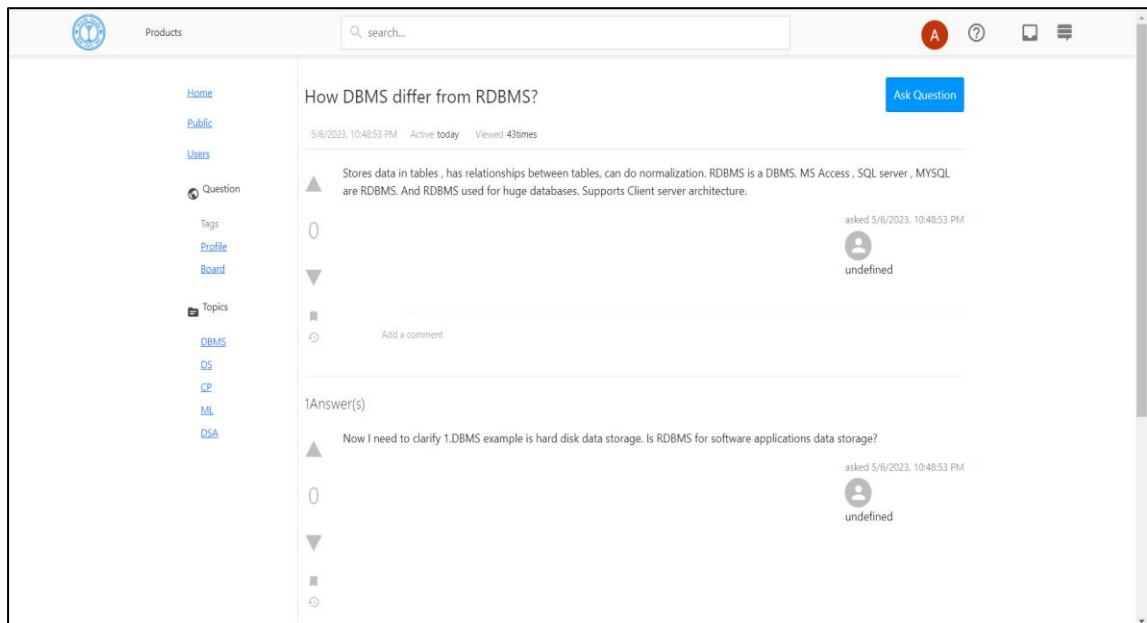


The screenshot shows a web application interface for asking a public question. At the top, there is a navigation bar with a logo on the left, the word "Products" in the center, and a search bar on the right. The main content area is titled "Ask a public question". Below this title is a form with three main sections: "Title", "Body", and "Tags". The "Title" section has a sub-header "Be specific and imagine you're asking a question to another person" and a text input field with a placeholder example: "e.g. Is there an R function for finding the index of an element in a vector?". The "Body" section has a sub-header "Include all the information someone would need to answer your question" and a rich text editor with a toolbar containing icons for bold, italic, underline, link, unlink, list, and code. The "Tags" section has a sub-header "Add up to 5 tags to describe what your question is about" and a text input field with a placeholder "press enter to add new tag". At the bottom of the form is a blue button labeled "Add your question".

**Fig. 5.3**  
**Ask Question**

Through this page the user can ask question. The user needs to give title, body and tags in order to post question.

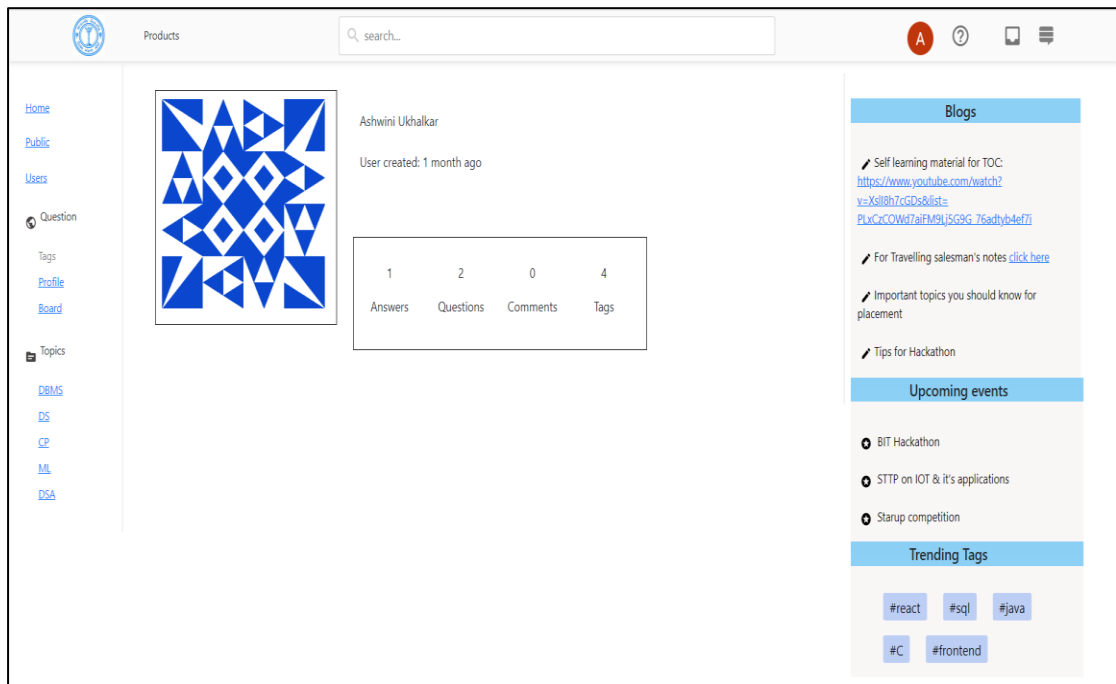




**Fig. 5.4**  
**Specific topic question page**

This page contains all the question related to a specific topic. In this page all the questions related to DBMS



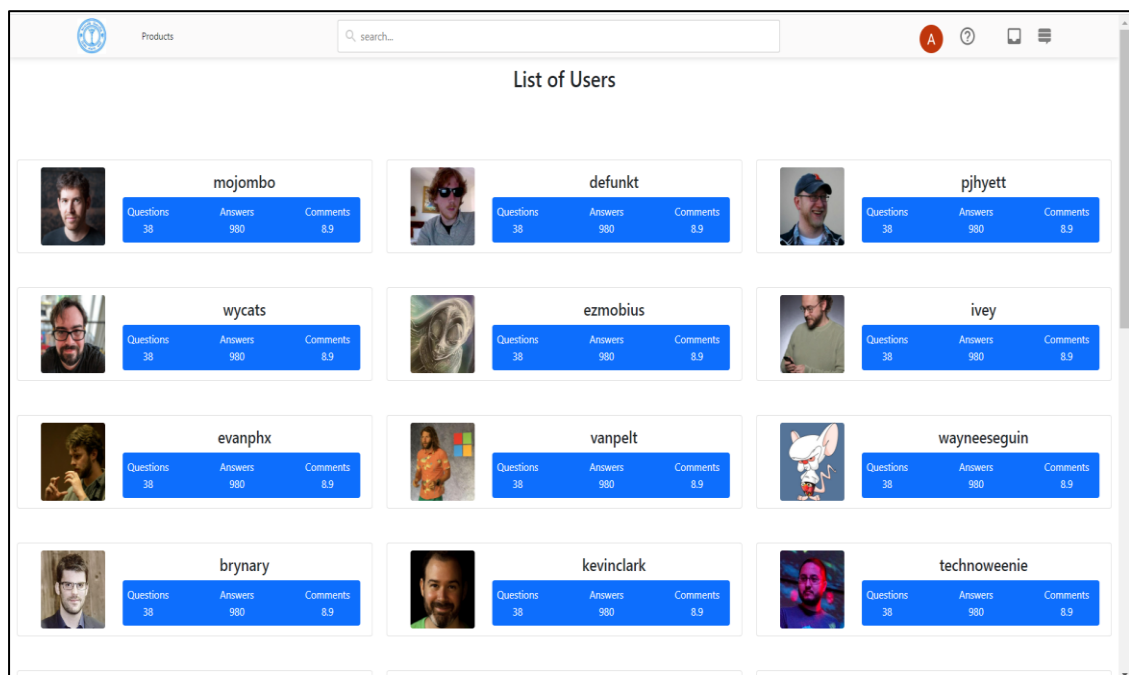


**Fig. 5.5**  
**Profile of user**

In this page the user can view his or her profile in which they can view the number of answers posted, questions posted, comments and tags used.







**Fig. 5.6**  
**All user page**

In this page we get to see all the users of the discussion forum. We also get to see the answers, questions and comments posted by them.

## **CHAPTER 6**

### **CONCLUSIONS AND FUTURE SCOPE**

---

In conclusion, the discussion forum has emerged as a powerful tool for fostering collaboration, knowledge sharing, and community engagement in today's digital age. Throughout this report, we have explored the various aspects and impacts of discussion forums, examining their benefits, challenges, and potential future developments.

First and foremost, discussion forums have proven to be an effective platform for connecting individuals with diverse backgrounds and perspectives. They facilitate open dialogue and allow participants to share their ideas, ask questions, and engage in meaningful discussions. This exchange of knowledge and experiences not only broadens individuals' horizons but also promotes critical thinking and intellectual growth.

Moreover, discussion forums play a crucial role in building online communities. They provide a space for like-minded individuals to connect, collaborate, and support one another. These communities often become hubs of expertise, enabling members to seek advice, mentorship, and opportunities for professional growth. The sense of belonging and shared purpose fostered within these communities creates a supportive and motivating environment for individuals to thrive.

Additionally, discussion forums have revolutionized education and learning. They provide a platform for students and educators to interact beyond the confines of the classroom. Students can seek clarification, discuss challenging concepts, and collaborate on projects. Educators, on the other hand, can facilitate discussions, share additional resources, and assess students' understanding in a more interactive and engaging manner. This shift towards online learning has been further accelerated by recent global events, highlighting the significance and potential of discussion forums in the education sector.

Hence, we can say discussion forums have transformed the way we communicate, learn, and collaborate. Their ability to connect individuals, foster communities, and promote knowledge exchange is unparalleled. While challenges exist, the benefits outweigh them, and with careful moderation and technological advancements, the potential for discussion forums to continue evolving and positively impacting various sectors remains vast. As we move forward in this digital era, the power of discussion forums will undoubtedly continue to shape and enrich our li



## REFERENCES

---

- [1] R. Seethamraju, "Effectiveness of using online discussion forum for case study analysis," *Education Research International*, vol. 2014, pp. 1–10, 2014. doi:10.1155/2014/589860
- [2] E. Abouzeid, A. Agarwal, and J. M. Frambach, "A design-based study for development of a model for a successful online Discussion Forum," *International Journal of Web-Based Learning and Teaching Technologies*, vol. 17, no. 1, pp. 1–18, 2022. doi:10.4018/ijwltt.294574
- [3] A. A. Gasmi, "Through the lens of students: How online discussion forums affect students' learning," *International Journal of Technology in Education*, vol. 5, no. 4, pp. 669–684, 2022. doi:10.46328/ijte.291
- [4] Y. Yu *et al.*, "Promoting college students' systems thinking in asynchronous discussions: Encouraging students initiating questions," *Frontiers in Psychology*, vol. 13, 2022. doi:10.3389/fpsyg.2022.1028655
- [5] K.-E. Tan, "Using online discussion forums to support learning of paraphrasing," *British Journal of Educational Technology*, vol. 48, no. 6, pp. 1239–1249, 2016. doi:10.1111/bjet.12491
- [6] S. Blackmon, "Outcomes of chat and discussion board use in Online learning: A research synthesis," *The Journal of Educators Online*, vol. 9, no. 2, 2012. doi:10.9743/jeo.2012.2.4



## ANNEXURE A

### Appendix A: Glossary

- Discussion Forum: Online Discussion Forum i.e Question and Answer Platform for the students of BIT.
- Admin: In this system, the HOD of each department.
- Blogs: Learning Material for students.
- Upcoming Events: The events held in the Institute like (Coding Contests, Technical events etc.)
- Trending Tags: Trending Questions on Discussion Forum.

### Appendix B: Analysis

