

System Requirements Specification

Students Message Board(SMB)

Team no:28

Ketan Sabne(CS20BTECH11043)

Sreeya Tipirisetty (ES20BTECH11030)

Tejaswi Suravarapu (ES20BTECH11027)

Shashidhar Rao Thumula (ES20BTECH11036)

25 February 2023

Contents

1	Introduction	3
1.1	Purpose	3
1.2	Scope	3
1.2.1	In Scope:	3
1.2.2	Out of Scope:	3
1.3	Definitions, Acronyms and Abbreviations	3
1.3.1	Acronyms and Abbreviations:	3
1.3.2	Definitions:	4
1.4	References for above definitions	4
1.5	Overview	4
2	Overall Description	4
2.1	Product perspective	4
2.2	Product Functions	5
2.3	User characteristics	6
2.4	Principal actors	6
2.5	General constraints	6
2.6	Assumptions and Dependencies	6
3	specific requirements	6
3.1	Functional Requirements	6
3.1.1	Use cases related to authenticity	7
3.1.2	Use cases related to profile	8
3.1.3	Use cases related to posting messages and tags	10
3.1.4	Use cases related to groups	12
3.1.5	Use cases related to database and system	13
3.2	Performance requirements	14
3.3	Design Constraints	15
3.4	External interface requirements	15
4	Future extensions	15
5	appendix	16

1 Introduction

1.1 Purpose

Students Message Board is a web-based application where the community of IITH can discuss any campus-related queries that includes academics, non-academics and also projects. The application also allows users to upload images and codes. This document outlines the characteristics of the Students Message Board and helps as a blue print for the developers and a proof of software quality for clients.

1.2 Scope

Here we outline which features are included in the project's scope and which aspects are not.

1.2.1 In Scope:

- Allows user to login through IITH email.
- Can be accessed through internet.
- Allows user to post question using proper title, tag, description, images and codes.
- Allows user to endorse other's answers.
- Allows user to close their question thread when they accept an answer.

1.2.2 Out of Scope:

- Email ids other than community cannot login to the Student message board.
- Not functional in the offline mode.

1.3 Definitions, Acronyms and Abbreviations

1.3.1 Acronyms and Abbreviations:

- SMB: Students message Board
- SA: System Administrator
- IIT: Indian Institute of Technology
- Devs: Developers
- MVC: Model view controller
- DOS: Denial of service

- DDOS: Distributed Denial of service
- AES: Advanced Encryption Standard
- pass_len: Length of password

1.3.2 Definitions:

- System Administrator: It is a person responsible for managing and maintaining computer systems, networks, servers, and other IT infrastructure within an organization.
- MVC: It is a software design pattern used to separate the concerns of an application's data, user interface, and control logic into three distinct components: the Model, the View, and the Controller.
- DDOS: It is a type of cyber attack where a large number of devices (IoT devices) are used to flood a target system or network with a massive volume of traffic, requests, or other data in order to overwhelm its resources and cause it to become unavailable to legitimate users.

1.4 References for above definitions

- wikipedia

1.5 Overview

The below shows the overview of the SRS document:

- Section 2 gives an overall description of the software.
- Section 3 gives the exact requirements that the software is expected to meet.
- Section 4 gives some possibilities of future extensions of this project.
- Section 5 shows some user screens.

2 Overall Description

2.1 Product perspective

Student Message Board is meant to serve as a common platform where students can interact with the fellow students and faculty. User can add his interests under his profile so that users with similar interests can network easily. Student Message Board application will enable a student/faculty to post questions to the IIT Hyderabad community and accept answers from them. Users can also see the previous questions posted by other users by searching relevant tags. However this application not restricted only to post questions, clubs and others

can announce their upcoming events and webinars. A User can join different teams in the college and network with the students of similar interests. The product is designed to be a web application that runs in a browser context.

2.2 Product Functions

Class of use case	Use case	Description of use case
Use cases related to authenticity	User registration in the portal	To create account in the web app
	User login in to the portal	Delete account
	Login to the portal	Deleting user from the portal
	Logout	Logout from the portal
Use cases related to Profile	View Profile	User can view and edit his profile
	Update Profile	User can update his information
	Change Password	to change account password
	Forgot Password	User can create new password
Use cases related to messages and tags	Adding Questions to the board	User can post questions to the IITH community
	Adding Tags to the Question	User can add tags to the questions so that other uses can find his question easily
	Uploading images and adding codes in question	User can upload the code snippet or images of codes/ problems
	Deleting the Question thread	User can delete the question he posted
	Editing the Question	User can edit the question he posted by visiting his profile
	Replying Thread	User can respond to the questions posted by the other users
	Endorsement of answers	User can endorse the answers by other users by liking the answer
	Sharing the thread	User can share the question thread to others
	Searching for question	User can find past questions by entering relevant tags in search bar
Use cases related to teams	Explore	User can find about all the Teams present in IIT Hyderabad
	Joining a team	User can join the club he likes by clicking on Join the team
	Exit team	to User can exit from a team

Use cases related to database and system	Administrating system seeing recent activity Deleting activity	System administrator will monitor system and data. User can see his previous questions and answers.Same with admin. User can delete his previous activity like questions, answers and comments
--	--	--

2.3 User characteristics

Here is list of some user characteristics:

- Accessibility: Website is accessible to everyone who has access to internet.
- Technical proficiency: Although some students might not have moderate Tech, considering that our webpage is easy to navigate and handle.
- profession: User can either be student or faculty.

2.4 Principal actors

Principal actors of SMB are user,admin,SA (as of now Devs are SA) and developers.In this way,we have 4 principal actors.

2.5 General constraints

- The development of SMB is constrained by 0\$ of budget.
- Software is bounded to be developed in 3 months.
- For good functionality,it requires stable internet connection.
- Browser should be moderately updated.

2.6 Assumptions and Dependencies

- It is assumed that user will submit correct information while registration.
- Admin will verify new accounts.
- For good functionality,user will require stable internet connection and modern updated software.
- SMB will depend on server-side programming language and database.

3 specific requirements

3.1 Functional Requirements

The functional requirements and use cases are listed below.

3.1.1 Use cases related to authenticity

- **Use Case 1:** User registration in the portal

Primary actor: User

Pre Conditions: Valid Email address , Internet Connectivity

Main Scenario:

1. Visit the website .
2. User needs to click on sign-up.
3. Login Page will be displayed.
4. User will create account using valid mail address.

Alternate Scenario:

1. Email address already in use
2. Invalid Email address

- **Use Case 2:** User login in to the portal

Primary actor: user

Pre Conditions: Registered email address, Internet connectivity

Main Scenario:

1. Visit the website
2. Login page will be open where user has to fill username and password.
3. User enters the details
4. After login user can access the website (can ask/ answer / search for the questions).

Alternate Scenario:

1. username doesn't exist.
2. incorrect password.

- **Use Case 3:** Delete account

Primary actor: User,Admin

Pre Conditions: Registered Email address , Internet connectivity,access to admin

Main Scenario:

1. Visit the website.
2. Login page will be displayed.
3. User needs to enter their credentials.
4. The user can access their profile after successfully logging in and click the "Delete Account" option.
5. The user will be redirected to a new page where they need to enter the credentials for verification.

6. If the user's credentials are valid, the portal deletes the account.
7. If user's account is inactive for more than 1 year, then admin can delete account or save it backup servers for emergency.

Alternate Scenario:

1. The user will be prompted to enter their information again if the credentials are incorrect.

- **Use Case 4:** Log out

Primary actor: User

Pre Conditions: Registered Email address , Internet connectivity

Main Scenario:

1. Assume user has done all his work on web.
2. user has to come to original board page.
3. user will click on logout button.
4. portal Asks for confirmation.
5. user will be logged out by clicking yes.

Alternate Scenario:

1. User will not be able to log out without confirmation.

3.1.2 Use cases related to profile

- **Use Case 5:** View profile

Primary actor: User,Admin

Pre Conditions: Internet connectivity , Registered user has already logged in, Access of database to Admin

Main Scenario:

1. User needs to click on "view profile" button on home page.
2. User will be directed to profile page.
3. Admin can see user's profile from database.

- **Use Case 6:** Update profile

Primary actor: User

Pre Conditions: Internet connectivity , Registered user has already logged in

Main Scenario:

1. User needs to click on "view profile" button on home page.
2. User will be directed to profile page.
3. There user can modify their profile by clicking "update profile" button.

Alternate Scenario:

1. User failed to modify changes if he/she press incorrect response.

- **Use Case 7:** Change password

Primary actor: User

Pre Conditions: Internet connectivity , Registered user has already logged in

Main Scenario:

1. User needs to click on "view profile" button on home page.
2. User will be directed to profile page.
3. There user can change their password by clicking on "change password" button.
4. Then the user can change password but before that he will authenticated by asking him about his previous password.

Alternate Scenario:

1. User cannot keep previous password as new one.
2. User failed to give correct previous password.
3. User failed to change password as pass_len is less than 8 or no use of special characters in new password.

- **Use Case 8:** Forgot password

Primary actor: User

Pre Conditions: Internet connectivity, already has account

Main Scenario:

1. User opens portal.
2. user clicks on "Forgot password?".
3. User will be asked to enter e-mail address.
4. User will get link to change password on his e-mail.
5. User will click on link and resets his password.
6. User tries to login using new reseted password then login will be successful.

Alternate Scenario:

1. User didn't get link due to some issues.
2. User failed to change password as passwords filled in 'new password' to 'confirm password' don't match.

3.1.3 Use cases related to posting messages and tags

- **Use Case 9:** Adding Questions to the board

Primary actor: User

Pre Conditions: Internet connectivity , Registered user has already logged in

Main Scenario:

1. User needs to click on "Ask Question" button on home page.
2. User will be directed to Questioning page where user can describe their question.
3. User will type his/her question and send it.

Alternate Scenario:

1. User failed to send question as it's already posted by someone before.
2. User failed to send as it didn't click on send button.

- **Use Case 10:** Adding Tags to the Question

Primary actor: User,admin

Pre Conditions: Internet connectivity , Registered user has already logged in,access to admin

Main Scenario:

1. After clicking on "Ask Question" button on home page the user will be directed to Questioning page.
2. There will be an option to add the tags related to the Question.
3. User will add some tags to question.
4. Admin will verify tags and also he can give tags to questions.

Alternate Scenario:

1. User tries to give same tag again to same question.
2. Admin failed to give tags or verify tags due to some reason.

- **Use Case 11:** Uploading images and adding codes in question

Primary actor: User

Pre Conditions: Internet connectivity , Registered user has already logged in

Main Scenario:

1. User needs to click on "Ask Question" button on home page.
2. User will be directed to Questioning page where user can describe their question.
3. In the description user can also upload images clicking on "upload images" icon.

4. In the description user can also add code by clicking on "syntax highlight" icon.

- **Use Case 12:** Deleting the Question thread

Primary actor: User (who posted),admin

Pre Conditions: Internet connectivity , Registered user has already logged in

Main Scenario:

1. User needs to click on "view profile" button on home page.
2. User will be directed to profile page.
3. There user can view the Questions added by them by clicking on "Uploaded Questions" button.
4. Then user need to select the Question and then have to tap on delete icon .
5. The portal asks for confirmation then by clicking yes question thread will be deleted.
6. For admin, if he sees any inappropriate content he can delete thread and can flag user.

Alternate Scenario:

1. User clicking on "No" will not delete the question.
2. Admin not able to delete thread due to access issue (to database).

- **Use Case 13:** Editing the Question

Primary actor: User

Pre Conditions: Internet connectivity ,user logged in

Main Scenario:

1. User will go to profile page.
2. There user can view the Questions added by him by clicking on "Uploaded Questions" button.
3. Then user need to select the Question and then have to tap on edit icon to make changes.
4. User will edit some things. Then the changes will be saved by clicking on "save changes" button.

Alternate Scenario:

1. User clicking on "Cancel" will not edit the question.

- **Use Case 14:** Replying Thread

Primary actor: Any user

Pre Conditions: Internet connectivity , Registered user has already logged in

Main Scenario:

1. Any user reads the question according to the tags they are interested in.
 2. User will be able to answer the question in text or code format.
 3. User will just type or copy paste code, then will click on send.
 4. Reply will be sent for that question.
- **Use Case 15:** Endorsement of answers
Primary actor: Any User
Pre Conditions: Internet connectivity , Registered user has already logged in
Main Scenario:
 1. User needs to click on the question thread.
 2. User will be able to endorse the answer given by other user.
 - **Use Case 16:** Sharing the thread
Primary actor: Any User
Pre Conditions: Internet connectivity , Registered user has already logged in
Main Scenario:
 1. User needs to click on question thread.
 2. User will be able to share the thread by clicking share button.
 - **Use Case 17:** Searching for question
Primary actor: Any User
Pre Conditions: Internet connectivity , Registered user has already logged in
Main Scenario:
 1. User needs to clicks on the search button on the home page.
 2. User will be able to see all the previous questions along with answers related to their search.

3.1.4 Use cases related to groups

- **Use Case 18:** Explore
Primary actor: Any User
Pre Conditions: Internet connectivity , Registered user has already logged in
Main Scenario:
 1. User needs to clicks on the explore option available on the navigation tab.
 2. User will be able to see all the clubs and departments details.

3. User will be able to join particular club or department discussions which will be displayed in communities page(after joining).

- **Use Case 19:** Joining a team

Primary actor: User(Author)

Pre Conditions: Internet connectivity, Logged into the account

Main Scenario:

1. User needs to select Explore where he will be shown the list of communities available
2. User has to select the community he want to join
3. User will be shown the description of the community and members of the community
4. User has to select Join community to get added into the community

Alternate Scenario:

1. Exiting or losing internet connection in the middle of the process will terminate the process

- **Use Case 20:** Exit team

Primary actor: User(Author)

Pre Conditions: Internet connectivity, Logged into the account, Member of the community

Main Scenario:

1. user has to select the team profile
2. User will be directed to team profile
3. user has to select "Exit teamy" icon.
4. user will be asked to confirm his action.
5. Upon confirm to exit team, the user will be removed from theteam.

Alternate Scenario:

1. Exiting or losing internet connection in the middle of the process will terminate the process

3.1.5 Use cases related to database and system

- **Use Case 21:** Administrating system

Primary actor: System administrator(SA),Devs

Pre Conditions: Access of databases and system to SA and Devs

Main Scenario:

1. SA and Devs has access to the database.
2. Devs can update webpage.

3. SA can monitor the server performance and backup the data.
4. Devs modified web-page successfully.

Alternate Scenario:

1. Between SA and Devs atleast one of them has no access.
2. Modification of page failed by Devs.

- **Use Case 22:** seeing recent activity

Primary actor: user,admin

Pre Conditions: Access of databases and system to admin ,user has account

Main Scenario:

1. For user,user will go to profile page.
2. User will click on 'activity' option there.
3. User will be redirected to activity page and is able to see activity.
4. Admin can see activity as he has access.

- **Use Case 23:** Deleting activity

Primary actor: User

Pre Conditions: User has already account and logged in, stable internet connection

Main Scenario:

1. Like above user will go to 'activity' page.
2. user can delete activity by going to each post and delete using delete option there.
3. User will be asked to confirm, after clicking 'yes' activity will be deleted.

Alternate Scenario:

1. User failed to delete activity as he clicked on 'no' option.

3.2 Performance requirements

- **Response time:**SMB should respond to user actions (such as posting a message or searching for content) within 3 seconds.
- **Compatibility:** SMB should be compatible with a wide range of web browsers and operating systems, so that users can access it from any device they choose. we can use MVC pattern for this.
- **Searching algorithm** should perform it's work in less time.
- **Server availability:** The message board should be available 24×7 to use, with minimal downtime for maintenance or modifications.

- High traffic: SMB should be able to sustain to high traffic as of now let's target it as 1000 users at once.
- Security: SMB should be designed to be secure. It should be able to prevent unauthorized access to any account and protecting user data.

3.3 Design Constraints

- Usability: The SMB should be designed to be easy to use and navigate, even for users who may not be familiar with message board platforms. We can also add help icon on web-page for such users.
- Cost: As the project has assigned 0\$ budget. So,SMB should be able work smoothly without using any money.
- Security: In database, passwords of users should be encrypted using most advanced encrypting(AES) and hashing algorithms.
- Fault Tolerance: Data should not be corrupted in case of system crash or any cyber attack.
- Development:For developing SMB, backend and frontend should be able to communicate properly.Integration must be done in good manner.
- UI/UX: interface should be easy to navigate and understand.

3.4 External interface requirements

The login screen on the homepage allows users to sign in using their email address and password. If the user doesn't already have an account, the option to do so is offered, and doing so will bring them to a new registration page. For logged in users they will be directed to the home page where they can post a new question or can search for similar previous question/answer. They can navigate to questions page or related tags or communities. In the navigation bar we can find home, profile, information, help and logout buttons. The communities page contains different groups related to different departments or clubs etc. In the profile page we can find details of the user, asked questions, recent activity, update profile, delete account and change password options. The information page contains the details of respective clubs and departments.

4 Future extensions

We have also discussed about further modified features to be included in software. The list is below:

1. Usage of more optimized algorithm for searching(questions or tags).
2. passwords will be salted along with hashing so that it won't be cracked easily.

3. Chatbots will be developed, in case if user asks question within chatbot's knowledge, then chatbot will answer question.
4. More strong architecture for web-page by adding some dummy servers which will help again DOS and DDOS attacks.
5. Add analytics and reporting features to SMB, allowing faculties to track student participation in discussions.
6. Adding some fanciness to SMB. e.g, giving badges to users whose responses are helpful,etc.

5 appendix

Below we have added some expected look of our pages. Let's look at it.

Student Message Board

Help

Email address:

Password:

Login

[Forget password?](#)

Sign up

Figure 1: Login Screen

Home Profile Explore

Help Logout

Profile

Name :
Occupation :
Email address :
Photo :

Asked Questions

Recent Activity

Saved Questions

Update Profile

Change Password

Delete Account

Figure 2: Profile page

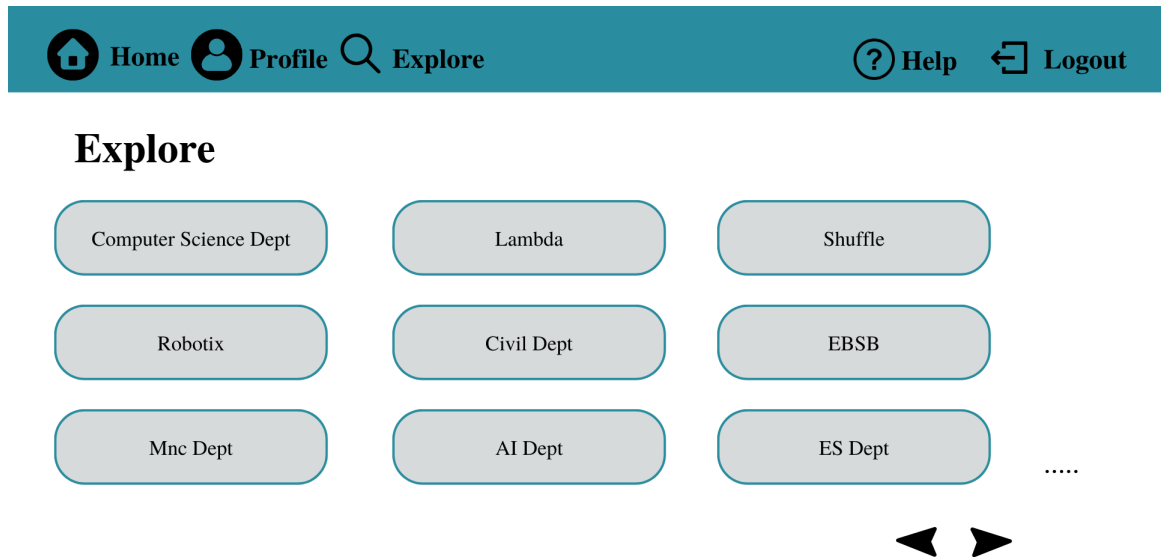


Figure 3: Explore page.

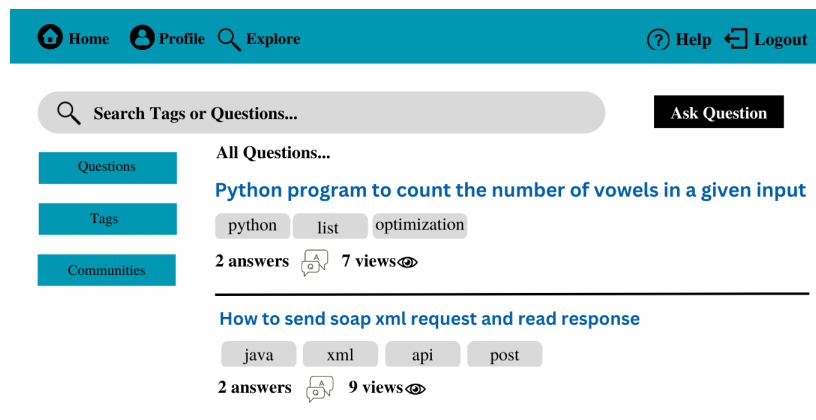


Figure 4: Home Screen

[Home](#) [Profile](#) [Explore](#)












[Help](#) [Logout](#)

Ask a public Question

Add Title

Next

Write the description of your problem

B I           

Next

Figure 5: Ask Question page

[Home](#) [Profile](#) [Explore](#)

[Help](#) [Logout](#)

Add Tags

Next

post your question

Figure 6: Ask Question Cont.

Home

Profile

Explore

Help

Logout

Search Tags or Questions...

Ask Question

Questions

Tags

Communities

Python program to count the number of vowels in a given input

asked 2 days ago

last modified a minute ago

share

save

I have written a basic code to count the number of vowels in a given string using a for loop. I am wondering if there is a more elegant way of writing the code?

The code which works:

```
words = input(str("Enter a word:")) #String input from user
words_list = list(words) #Convert user input to a list
vowels = ["a", "e", "i", "o", "u"] #Define vowels list
number = 0 #No. of vowels set to 0
for word in words_list:
    if word in vowels:
        number += 1
print(f"The number of vowels in {words} are {number}")
```

python

list

optimization

Figure 7: View Question page

Home

Profile

Explore

Help

Logout

Search Tags or Questions...

Ask Question

Questions

Tags

Communities

2 Answers

3 endorsements

fastest code for most strings

```
sum(map(lambda string: string.lower().count, 'aeiou'))
```

Using high performance numpy operations is faster for very large strings

```
import numpy as np
words = input(str("Enter a word:"))
npstringarray = np.frombuffer(words.lower(), dtype=np.uint8)
vowels = np.frombuffer('aeiou', dtype=np.uint8)
count = np.isin(npstringarray, vowels).sum()
```

using list a list comprehension (instead of a generator) with an if statement after instead of summing true and false is slightly faster, but still slower than using map count

```
vowels = set('aeiou')
count = sum(1 for c in words.lower() if c in vowels)
```

answered 3 hrs ago

last modified 1 hr ago

endorse

Figure 8: View Question Cont

20

[Home](#) [Profile](#) [Explore](#)

[Help](#) [Logout](#)












[Ask Question](#)

Questions

Tags

Communities

Your Answer

B I           

[post your answer](#)

Figure 9: Answering the Question.

[Home](#) [Profile](#) [Explore](#)

[Help](#) [Logout](#)

Need Help ?

[Search](#)

About ▼

FAQ's ▼

Contacts ▼

Figure 10: Help Page.

21