



DoConnect

Capstone
project

Submitted By:
C13 Group-1

DoConnect

Mentor : Shekhar Kumar Hela

Group Members :

1. Sureka A R K
2. Sugandha Malhotra
3. Arish Ahmad
4. Rama Krishna Dokina

Problem Statement

- DoConnect is a popular Q and A form in which technical questions are asked and answered.
- For above application there are two types of users :
 1. User
 2. Admin

Introduction

- DoConnect is a popular Q n A website for professional and enthusiast technologists in which technology related questions are asked and answered.
- There are many websites where people can ask and answer questions like StackOverflow, Quora, etc.
- DoConnect features questions and answers on a wide range of topics in computer programming.
- In this project we have created a website to provide all the services to the users.
- There are 2 users in this application:
 1. User
 2. Admin

Introduction

- DoConnect is process of posting technical questions and answers.
- A user can easily access all the facilities of the Internet to find relevant answers of their technical domain.
- Unlike traditional users have to wait till the next day to get their doubt cleared from their respective mentors but now they can easily get the solutions within few minutes
- To achieve this we have used Spring boot and Angular to develop our project called DoConnect.
- We have used angular for developing the frontend part and java spring boot for backend and to store the data we have used MySQL Database.

Objectives

- The main objective of the project DoConnect is to create a simple online platform where users can find answers for their questions with the experts approval.

Functional requirements of User

- User should be able to Register , Login and Logout of the application.
- User should be able to Post any question under any topic.
- User should be able to Search the question using the search box.
- User should be able to Answer any question.
- User should be able to Answer more than one question and more than one time.
- User should be able to upload images to refer.
- User should be able to chat with other users.

Functional Requirements of Admin

- Admin should be able to Register, Login and Logout of the application.
- Admin should be able to get Mail as soon as any new Question is Posted or any Answers are added.
- Admin should be able to Approve the Question and Answer. Any Question or Answer will be visible on the platform only if it is approved.
- Admin should be able to Delete inappropriate Questions or Answers.

Technology Stack

Technologies Used

- Technologies:
 - Angular
 - Spring Boot
- Languages:
 - TypeScript
 - Java
 - SQL

IDE

- Visual Studio Code
- Spring Tool Suite
- MySql Workbench

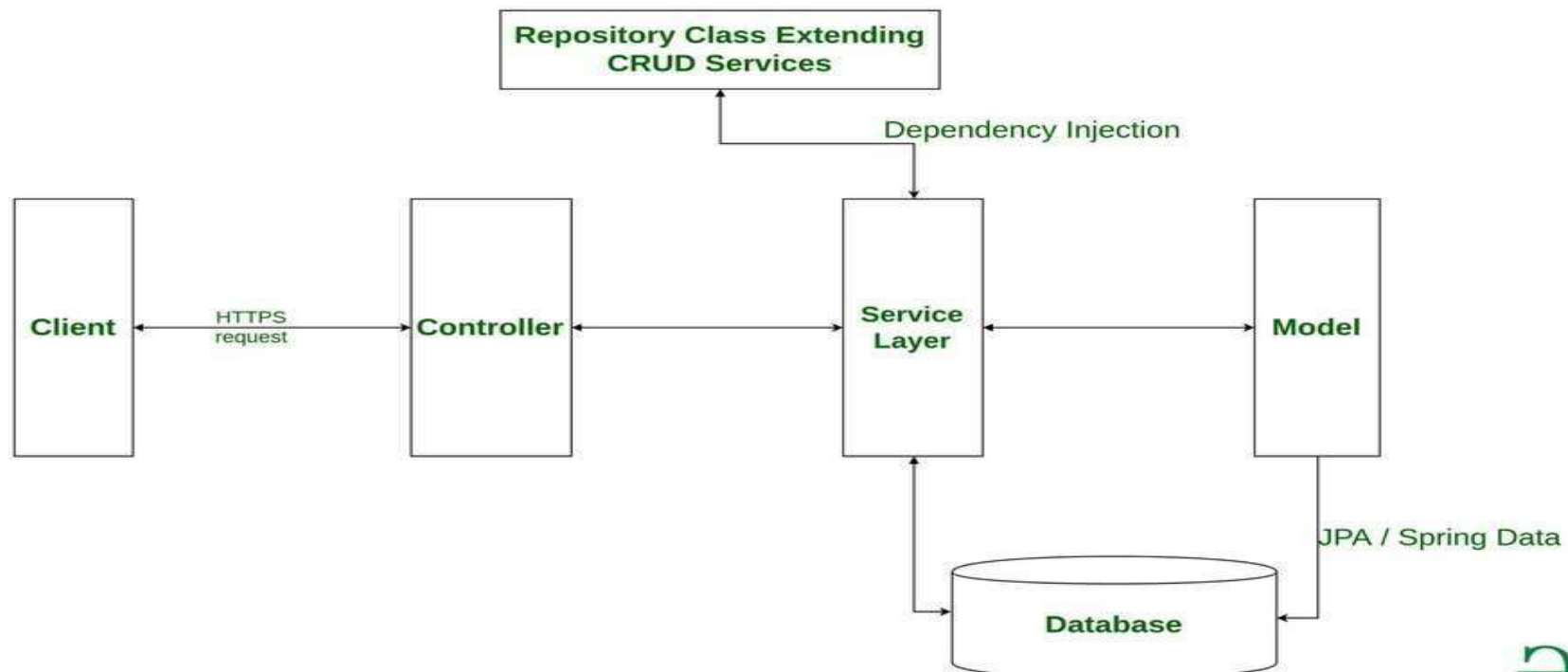
Angular Architecture

There are 8 main blocks of Angular:

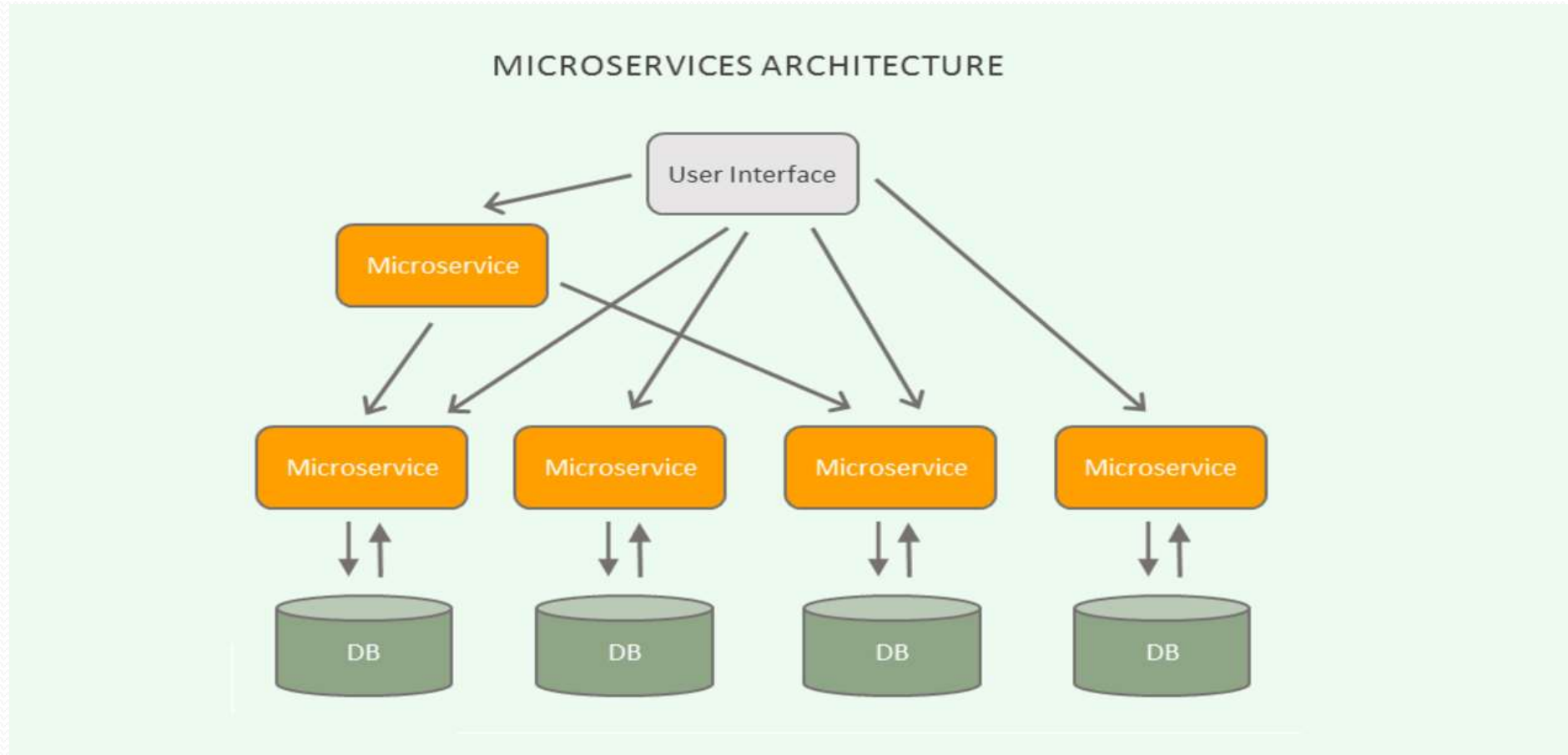
- Module
- Component
- Metadata
- Template
- Data Binding
- Service
- Directive
- Dependency Injection

SpringBoot Architecture

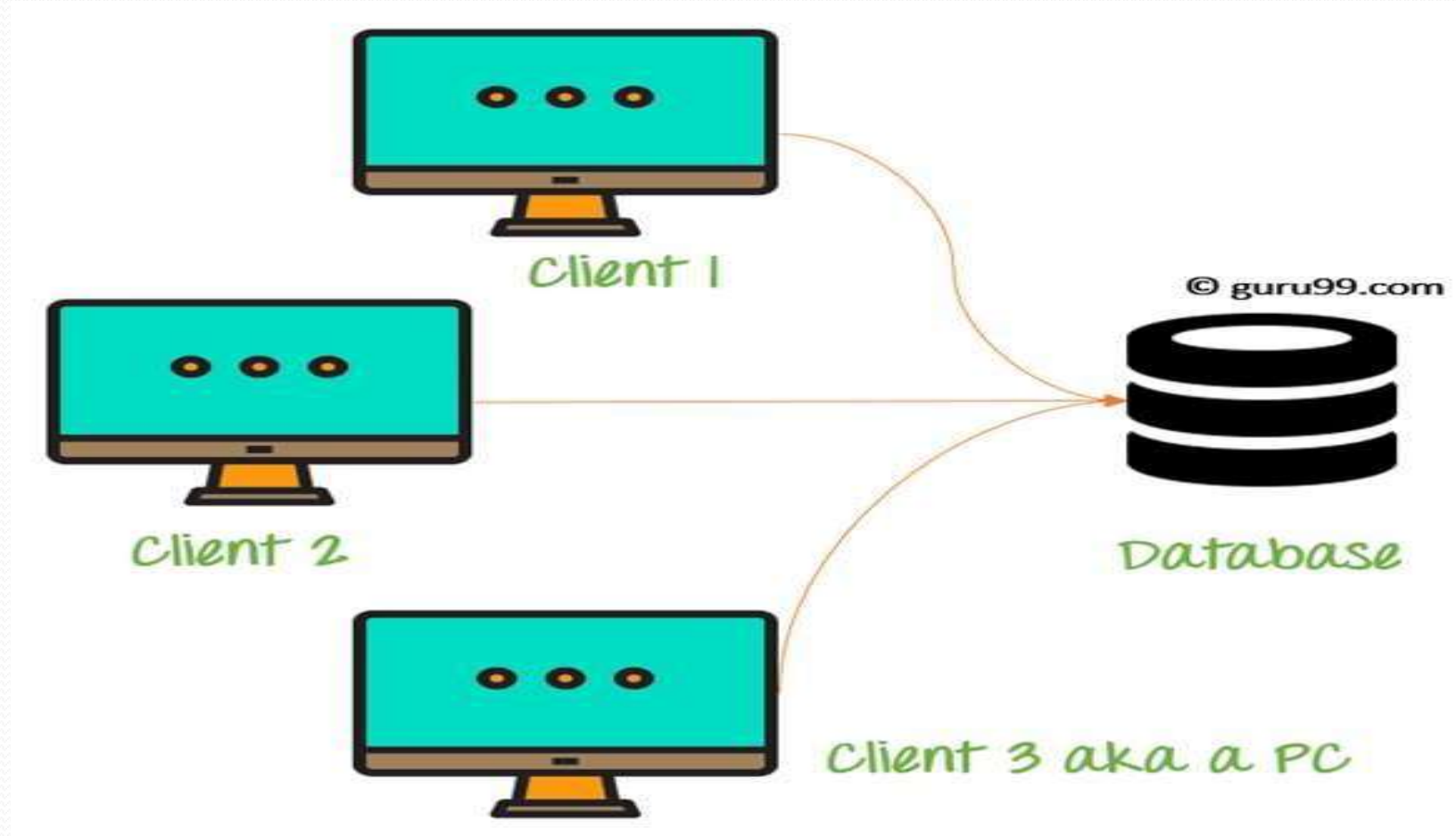
Spring Boot flow architecture



Micro service Architecture



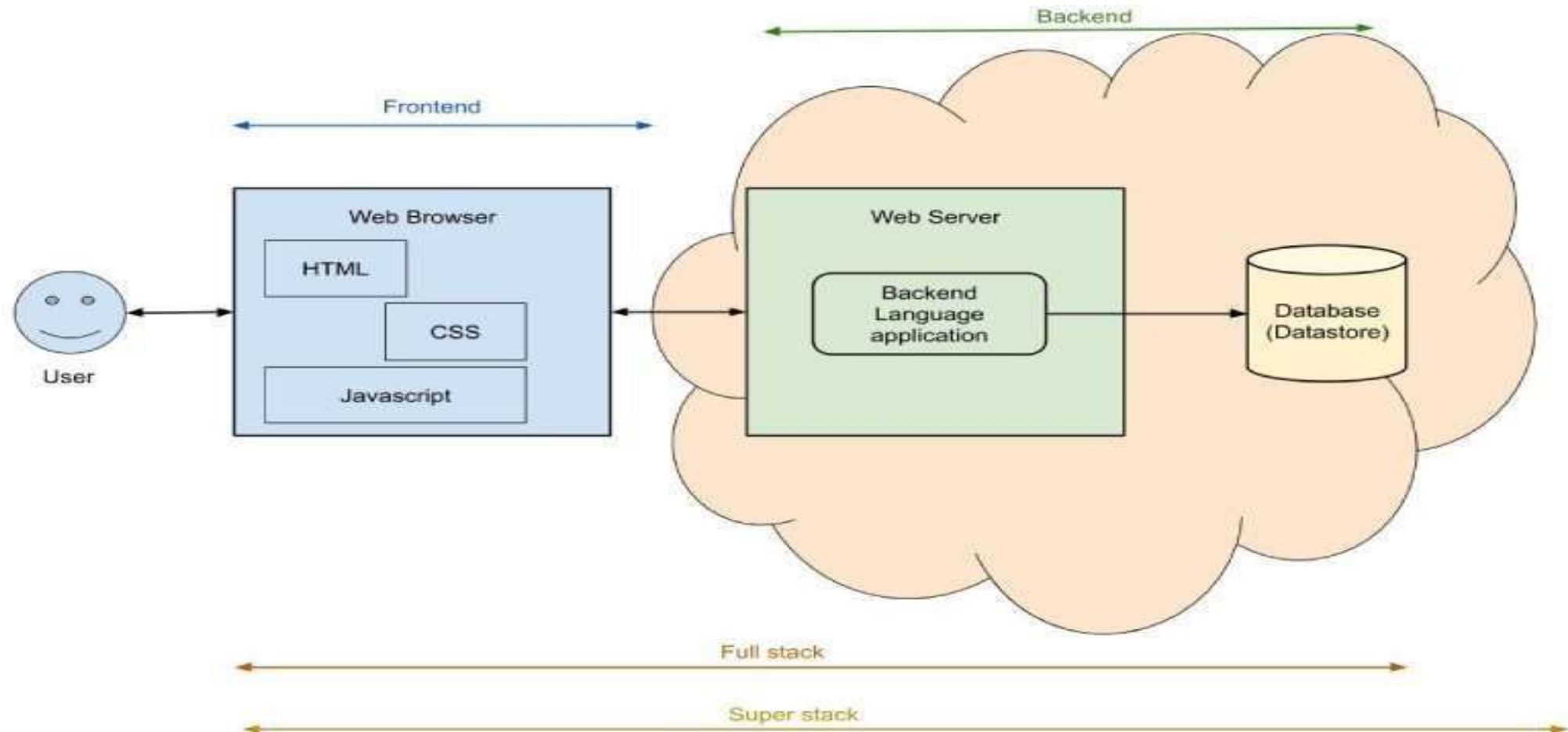
Data Base Architecture



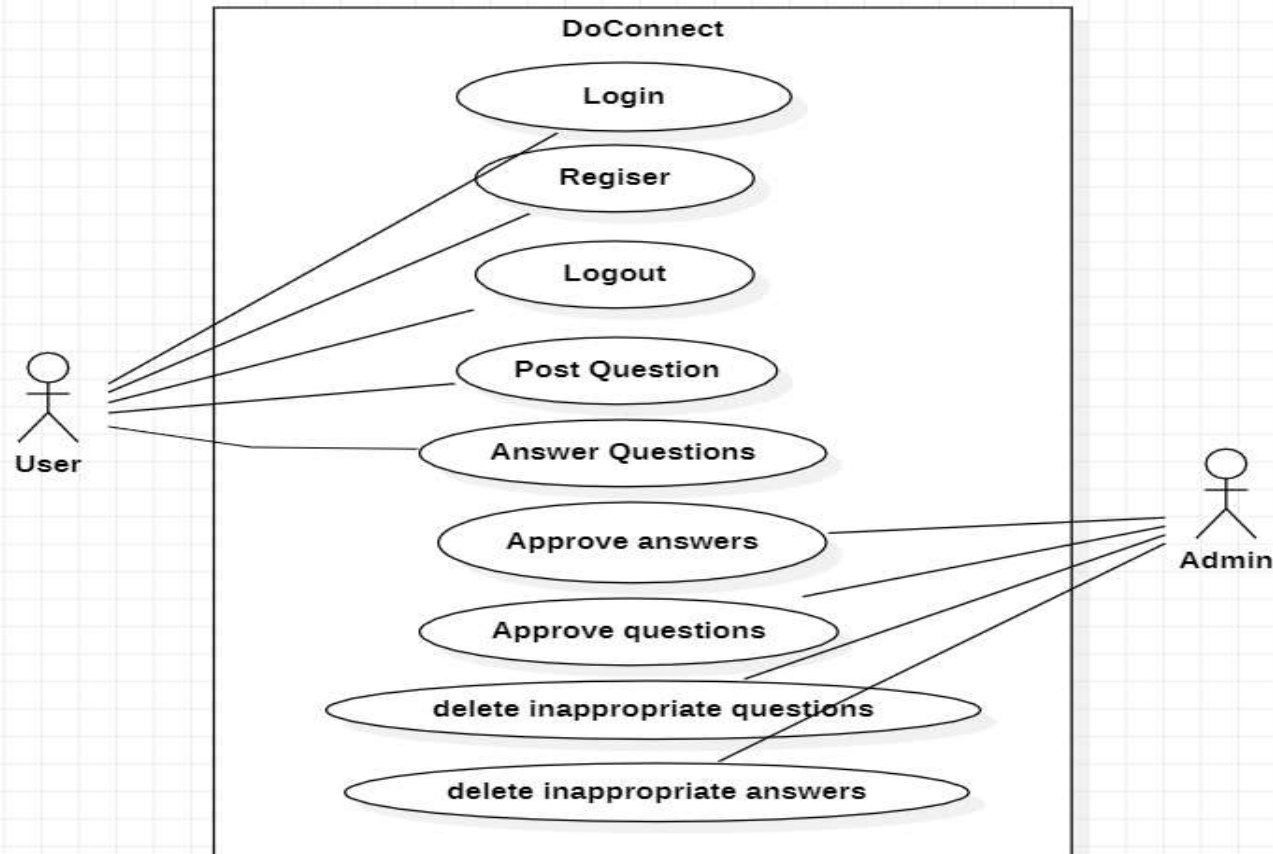
Database Structure

- User
- Admin
- Question
- Answer

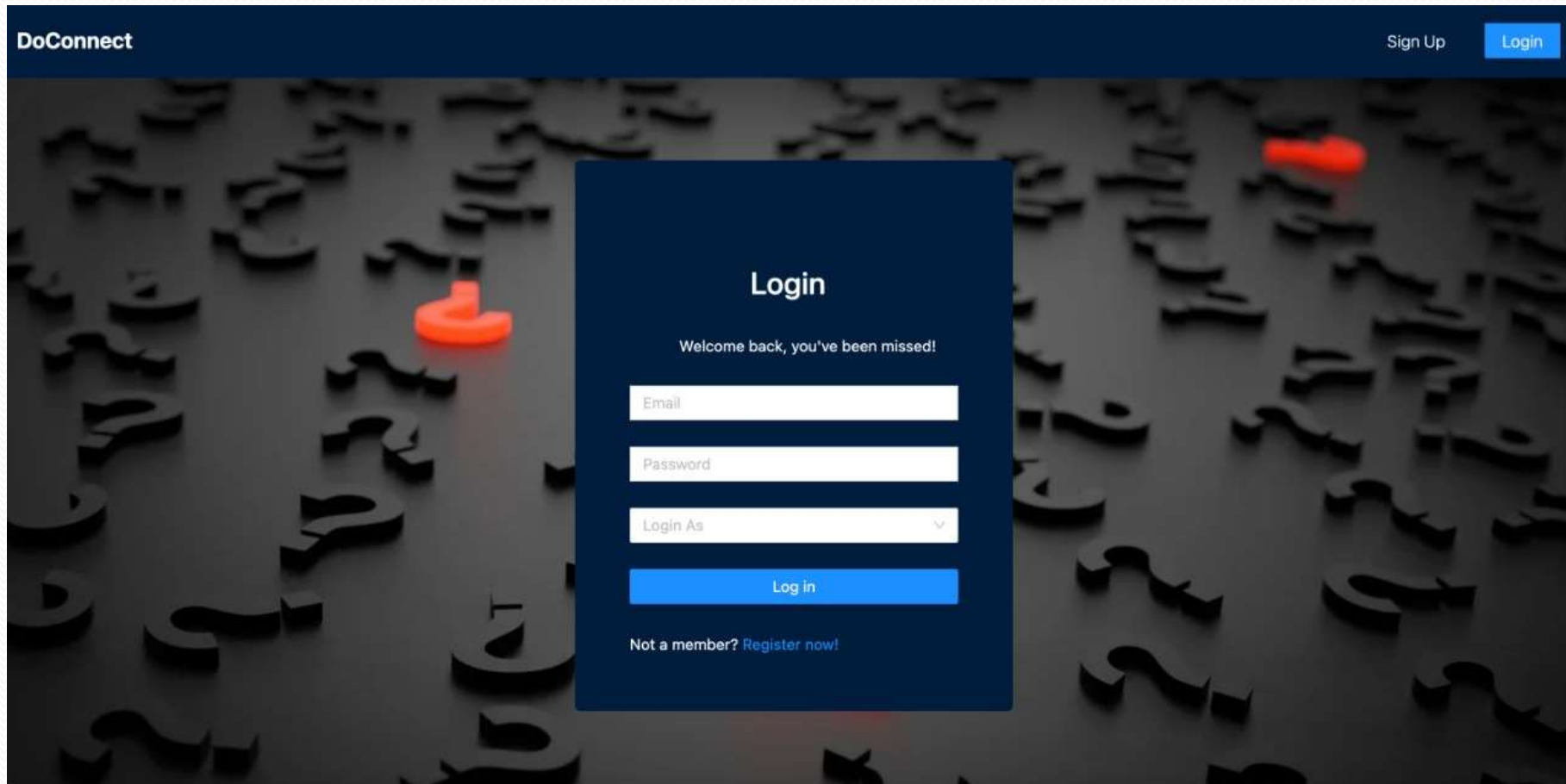
Total Project View



Use Case Diagram



Glimpse of Our Project



The screenshot shows a web application interface for 'DoConnect'. The background is a dark grey with a pattern of raised, 3D question marks. Two orange question marks are highlighted. A dark blue login modal is centered on the screen. The modal has a title 'Login', a welcome message, input fields for email, password, and a dropdown for 'Login As'. A blue 'Log in' button is at the bottom of the modal, and a link to 'Register now!' is at the very bottom.

DoConnect

Sign Up Login

Login

Welcome back, you've been missed!

Email

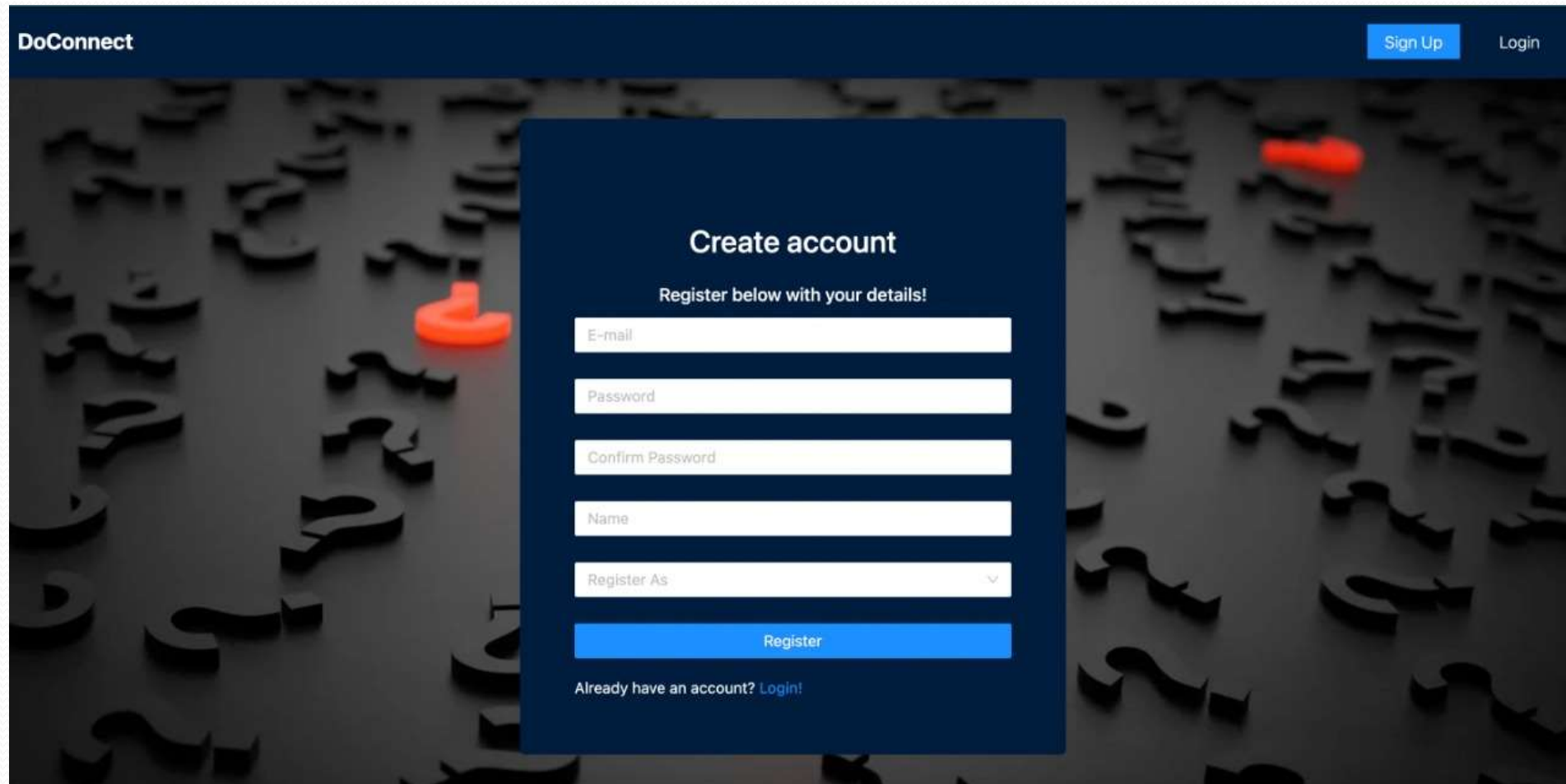
Password

Login As

Log in

Not a member? [Register now!](#)

Register

A screenshot of a web registration form for 'DoConnect'. The form is a dark blue modal box centered on a background of a dark surface with scattered white question marks and two orange pushpins. The modal has a title 'Create account' and a subtitle 'Register below with your details!'. It contains five input fields: 'E-mail', 'Password', 'Confirm Password', 'Name', and 'Register As' (a dropdown menu). Below these is a blue 'Register' button. At the bottom, it says 'Already have an account? [Login!](#)'. The top of the page has a dark blue header with the 'DoConnect' logo on the left, and 'Sign Up' and 'Login' buttons on the right.

DoConnect

Sign Up Login

Create account

Register below with your details!

E-mail

Password

Confirm Password

Name

Register As

Register

Already have an account? [Login!](#)

Dashboard

DoConnect

DashboardPost QuestionChatboxLogout

programming

Difference between java and javascript?

Posted By : test2 - Posted On : Sunday, July 31, 2022

Science

What are Terrestrial planets?

Posted By : user1 - Posted On : Monday, August 1, 2022

Programming

What is python?

Post the Question

DoConnect

DashboardPost QuestionChatboxLogout

Post Question

Title

Body

Post Question

?

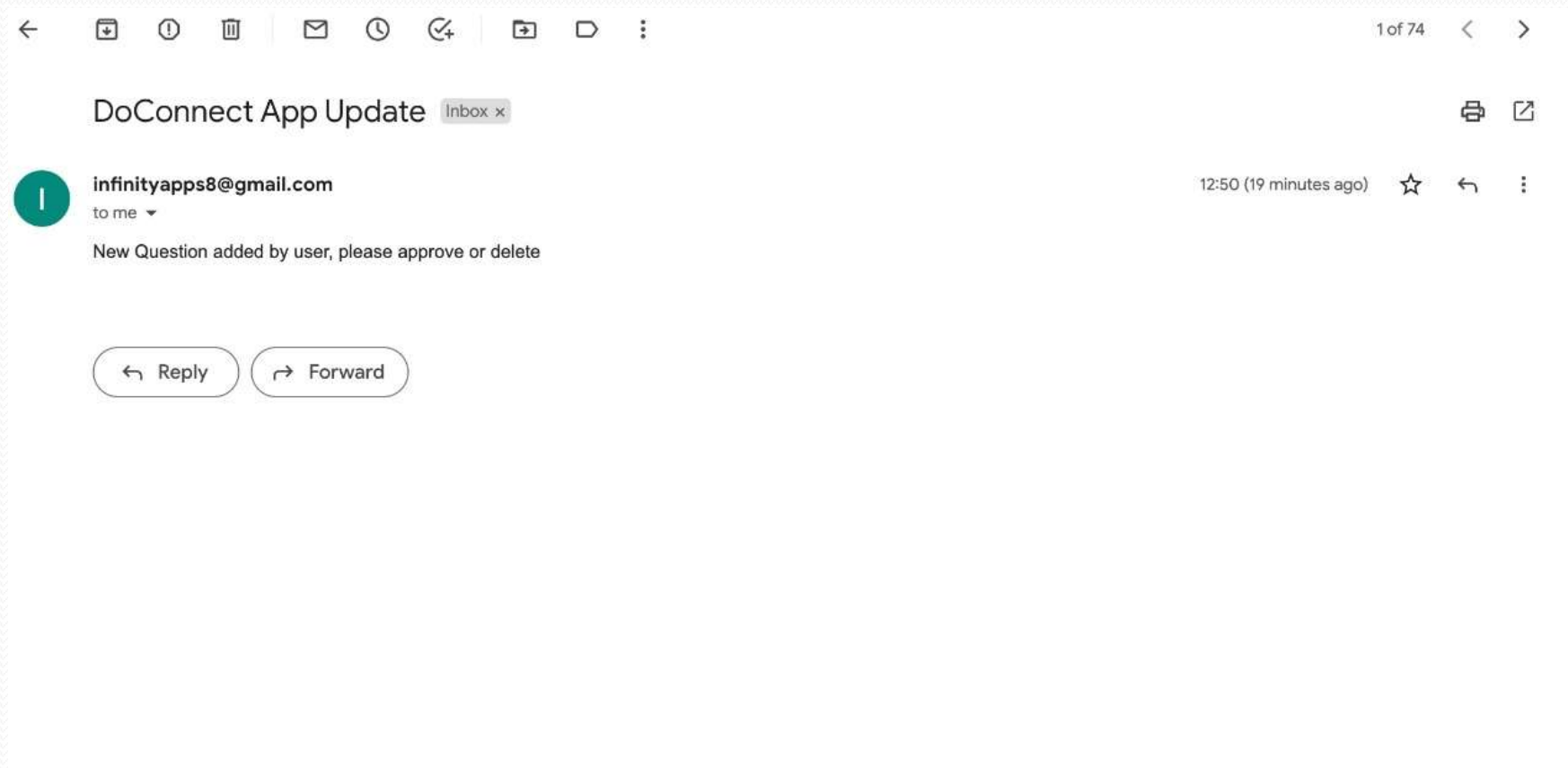
Admin Dashboard

Questions

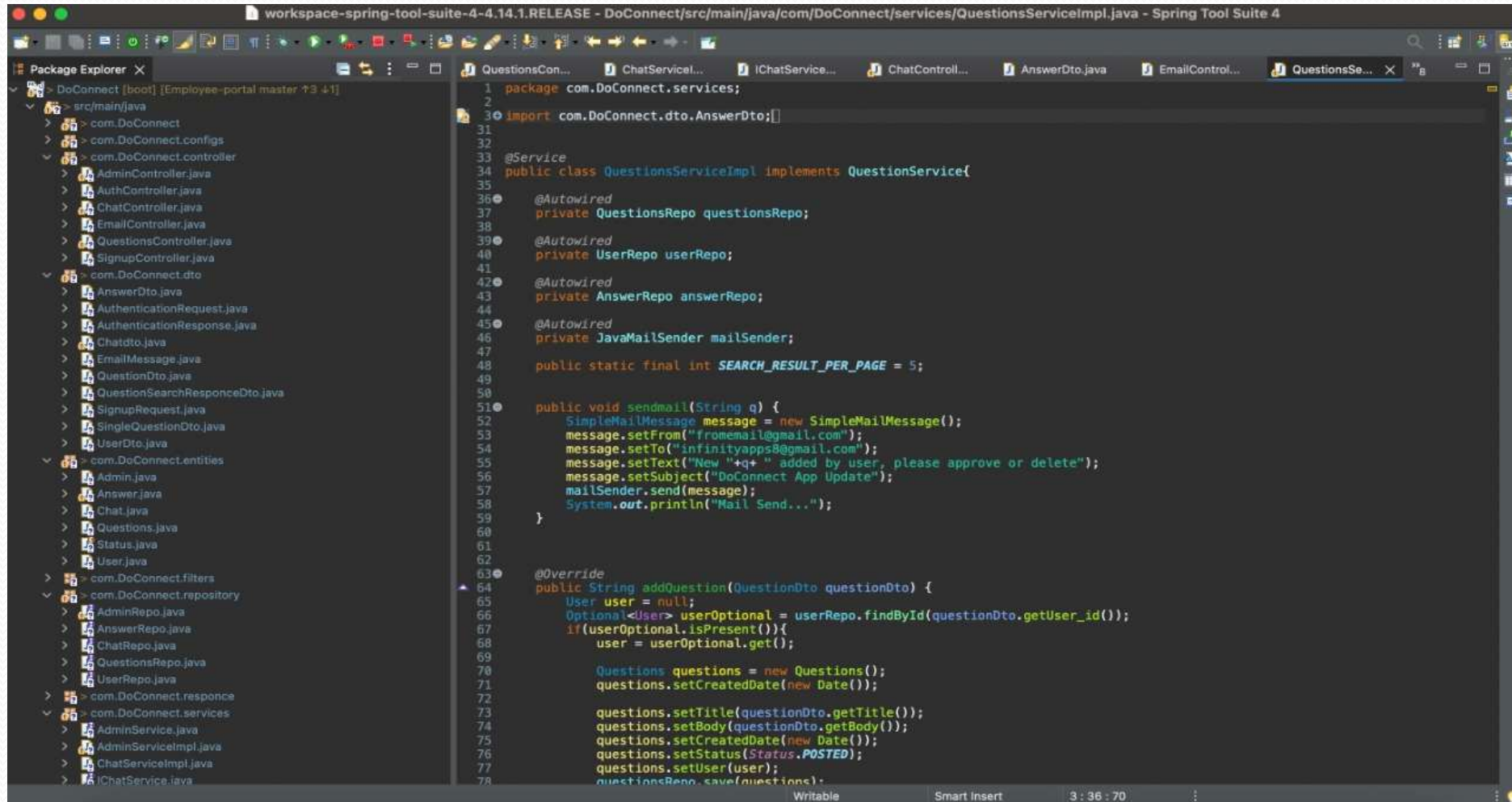
Answers

Title	Posted By	Description	Posted Date	Action
english	user1	what are clauses?	Monday, August 1, 2022	Approve Delete
English	user1	difference between imperative and interrogative ?	Monday, August 1, 2022	Approve Delete
SQL	user3	How to delete a row in mysql?	Thursday, August 4, 2022	Approve Delete

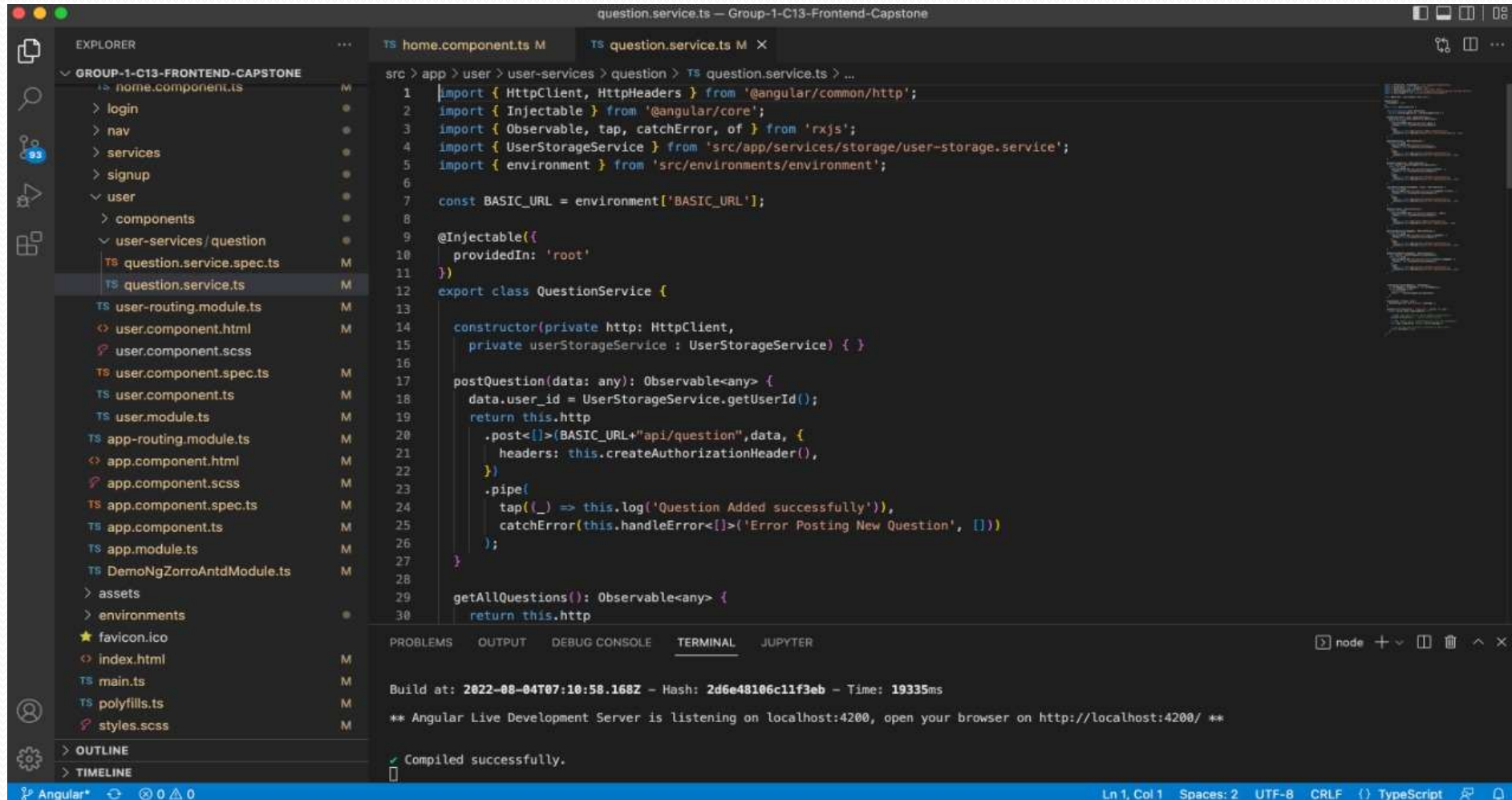
Update for Admin



Structure Of Back-end



Structure Of Front-end



```
question.service.ts — Group-1-C13-Frontend-Capstone

1 import { HttpClient, HttpHeaders } from '@angular/common/http';
2 import { Injectable } from '@angular/core';
3 import { Observable, tap, catchError, of } from 'rxjs';
4 import { UserStorageService } from 'src/app/services/storage/user-storage.service';
5 import { environment } from 'src/environments/environment';
6
7 const BASIC_URL = environment['BASIC_URL'];
8
9 @Injectable({
10   providedIn: 'root'
11 })
12 export class QuestionService {
13
14   constructor(private http: HttpClient,
15     private userStorageService : UserStorageService) { }
16
17   postQuestion(data: any): Observable<any> {
18     data.user_id = UserStorageService.getUserId();
19     return this.http
20       .post<[]>(BASIC_URL+"api/question",data, {
21         headers: this.createAuthorizationHeader(),
22       })
23       .pipe(
24         tap((_) => this.log('Question Added successfully')),
25         catchError(this.handleError<[]>('Error Posting New Question', []))
26       );
27   }
28
29   getAllQuestions(): Observable<any> {
30     return this.http
```

Build at: 2022-08-04T07:10:58.168Z - Hash: 2d6e48106c11f3eb - Time: 19335ms

** Angular Live Development Server is listening on localhost:4200, open your browser on http://localhost:4200/ **

✓ Compiled successfully.

Database

The screenshot displays the MySQL Workbench application window. The title bar reads "MySQL Workbench". The interface is divided into several sections:

- Left Sidebar (Navigation Panel):**
 - MANAGEMENT:** Server Status, Client Connections, Users and Privileges, Status and System Variables, Data Export, Data Import/Restore.
 - INSTANCE:** Startup / Shutdown, Server Logs, Options File.
 - PERFORMANCE:** Dashboard, Performance Reports, Performance Schema Setup.
 - Object Info / Session:** No object selected.
- Top Panel:** "Administration" and "Schemas" tabs. The "Query 1" tab is active, showing a SQL query:

```
1 • show databases;  
2 • use quoraclo;  
3 • show tables;
```
- Query Execution Area:** Below the query, there are icons for execution and a "Limit to 1000 rows" option.
- Result Grid:** Displays the results of the query. The first result, "Tables_in_quoraclo...", shows a list of tables:

Tables_in_quoraclo...
admin
answers
appdata
chatdata
hibernate_sequence
question
users
- Bottom Panel:** "Action Output" section showing the execution log:

	Time	Action	Response	Duration / Fetch Time
✓ 37	13:03:24	show tables	7 row(s) returned	0.045 sec / 0.000013...

