A

CAPSTONE PROJECT REPORT ON

DOCONNECT

Submitted By

Group-1

C13

BATCH MEMBERS

1. **Sureka A R K** -sureka1039@gmail.com
2. **Sugandha Malhotra** -vansh.sugandha@gmail.com
3. **Arish Ahmad** - arishahmad70@gmail.com
4. **Rama Krishna Dokina** -krishdokina@gmail.com

Mentored by

Shekhar Kumar Hela

PROBLEM STATEMENT

DoConnect is a popular Q and A form in which technical questions are asked and answered.

There are 2 users on the application:-

1. User
2. Admin

FUNCTIONAL REQUIREMENTS:

User –

- As a user I should be able to login, Logout and Register into the application.
- As a user I should be able to post any questions under any topic.
- As a user I should be able to search the questions on any string written in search box.
- As a user I should be able to answer any question.
- As a user I should be able to answer more than one question and more than one time.
- As a user I should be able to chat with other users
- As a user I should be able to upload images for reference.

Admin –

- As an Admin I should be able to login, Logout and Register into the application.
- As an Admin I should be able to get mail as soon as any new Question is posted or any Answers are added.
- As an Admin I should be able to approve the question and Answer.
- As an Admin I should be able to delete inappropriate Questions or Answers.

SYSTEM SPECIFICATIONS

SOFTWARE REQUIREMENTS:

Technologies:

- Angular
- Java Spring Boot

Languages:

- TypeScript
- HTML
- CSS
- Java
- SQLQueries

IDE:

- Eclipse
- Vscode
- MySql

HARDWARE REQUIREMENTS:

Operating System:

- Windows7 or above
- Linux distros

Processor:

- Intel or AMD dual core x86processor.

Ram:

- 2 GB or above.

Hard disk:

- 500 MB of free disk space ormore.

ABSTRACT

- DoConnect is process of posting technical questions and answers similar to just like stackoverflow, Quora.
- 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
- User can browse online sites and can get their relevant answers easily at one go.
- We have used angular for developing the frontend part and java spring boot for backend and to store the data we have used MySQL.

ANGULAR ARCHITECTURE

Angular is a platform or framework to build client-based applications in HTML and TypeScript. It is written in TypeScript.

There are 8 blocks of Angular:

- Module
- Component
- Metadata
- Template
- DataBinding
- Service
- Directive
- DependencyInjection

Module:

Angular apps are modular and Angular has its own modularity system called Angular modules or Ng Modules. Every Angular app has at least one Angular module class, the root module, conventionally named App Module. While the root module may be the only module in a small application, most apps have many more feature modules, each a cohesive block of code dedicated to an application domain, a workflow, or a closely related set of capabilities.

NgModule is a decorator function that takes a single metadata object whose properties describe the module. The most important properties are:

- declarations - the *view classes* that belong to this module. Angular has three kinds of view classes: components, directives, andpipes.
- exports - the subset of declarations that should be visible and usable in the component templates of othermodules.

- imports - other modules whose exported classes are needed by component templates declared in *this*module.
- providers - creators of services that this module contributes to the global collection of services; they become accessible in all parts of theapp.
- bootstrap - the main application view, called the *root component*, that hosts all other app views. Only the *root module* should set this bootstrapproperty.

Component:

Components are the most basic UI building block of an Angular app. An Angular app contains a tree of Angular components. Angular components are a subset of directives, always associated with a template.

Metadata:

Metadata is used to decorate a class so that it can configure the expected behavior of the class. Following are the different parts for metadata. Annotations − These are decorators at the class level. This is an array and an example having both the @Component and @Routesdecorator.

Template:

A template is a form of HTML that tells Angular how to render the component. Views are typically arranged hierarchically, allowing you to modify or show and hide entire UI sections or pages as a unit. The template immediately associated with a component defines that component's hostview.

Data Binding:

Data binding in AngularJS is the synchronization between the model and the view. When data in the model changes, the view reflects the change, and when data in the view changes, the model is updated as well.

Service:

Service is a broad category encompassing any value, function, or feature that your application needs. Almost anything can be a service. A service is typically a class with a narrow, well-defined purpose. It should do something specific and do it well.

Examples include:

- logging service
- dataservice
- messagebus
- taxcalculator
- applicationconfiguration

There isnothing specifically Angular about services. Angular has no definition of a service. There is no service base class, and no place to register a service. Yet services are fundamental to any Angular application. Components are big consumers ofservices.

Directive:

Directives are classes that add new behavior or modify the existing behavior to the elements in the template. Basically directives are used to manipulate the DOM, for example adding/removing the element from DOM or changing the appearance of the DOM elements.

Dependency Injection:

Dependency injection, or DI, is a design pattern in which a class requests dependencies from external sources rather than creating them. Angular's DI framework provides dependencies to a class upon instantiation. Use Angular DI to increase flexibility and modularity in your applications.

SPRINGBOOT ARCHITECTURE

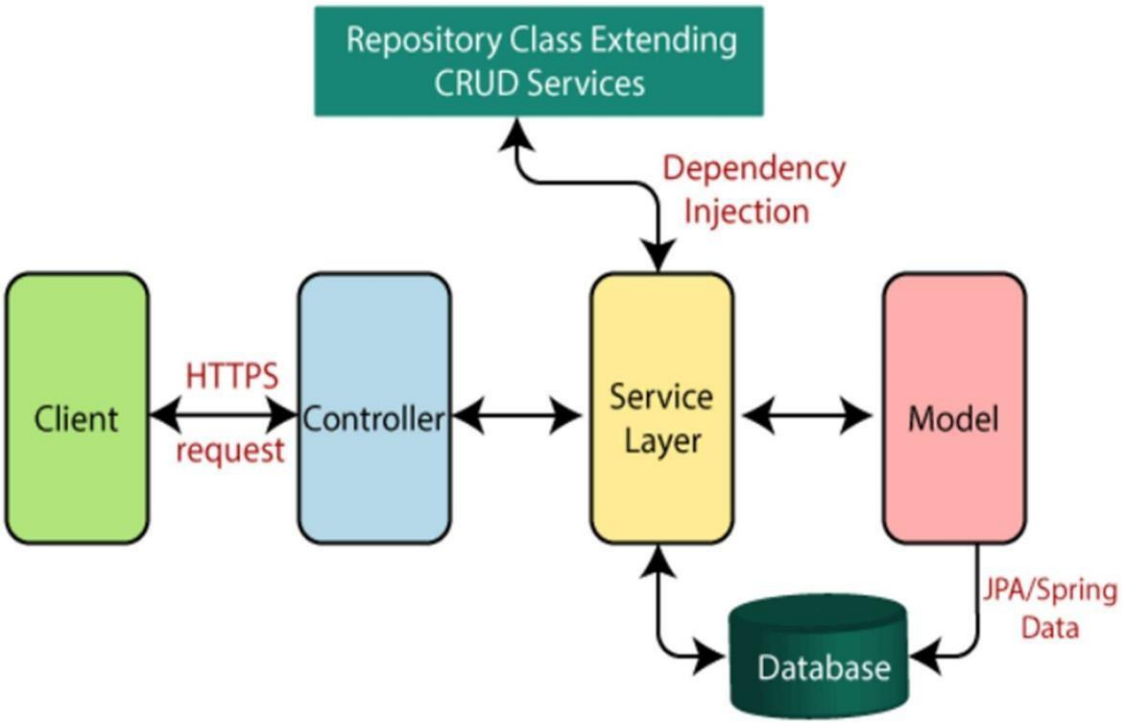
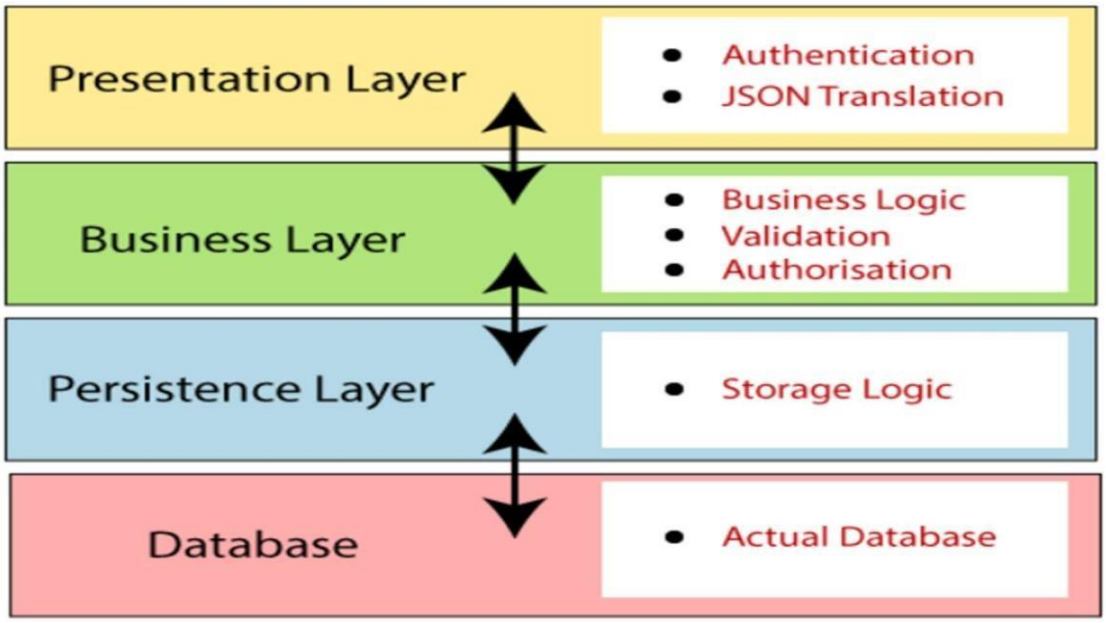


FIG: SPRINGBOOT ARCHITECTURE

The spring boot consists of the following four layers:

- **Presentation Layer** – Authentication & Json Translation.
- **Business Layer** – Business Logic, Validation &Authorization.
- **Persistence Layer** – StorageLogic.
- **Database Layer** – ActualDatabase.



Presentation Layer:

The presentation layer is the top layer of the spring boot architecture. It consists of Views. i.e., the front-end part of the application. It handles the HTTP requests and performs authentication.

Business Layer:

The business layer contains all the business logic. It consists of services classes. It is responsible for validation and authorization. The persistence layer contains all the database storage logic.

Persistence Layer:

Persistence Layer: The persistence layer contains all the storage logic and translates business objects from and to database rows. Database Layer: In the database layer, CRUD (create, retrieve, update, delete) operations are performed.

Database:

The database layer contains all the databases such as MySql, MongoDB, etc. This layer can contain multiple databases. It is responsible for performing the CRUDoperations.

DATABASE ARCHITECTURE

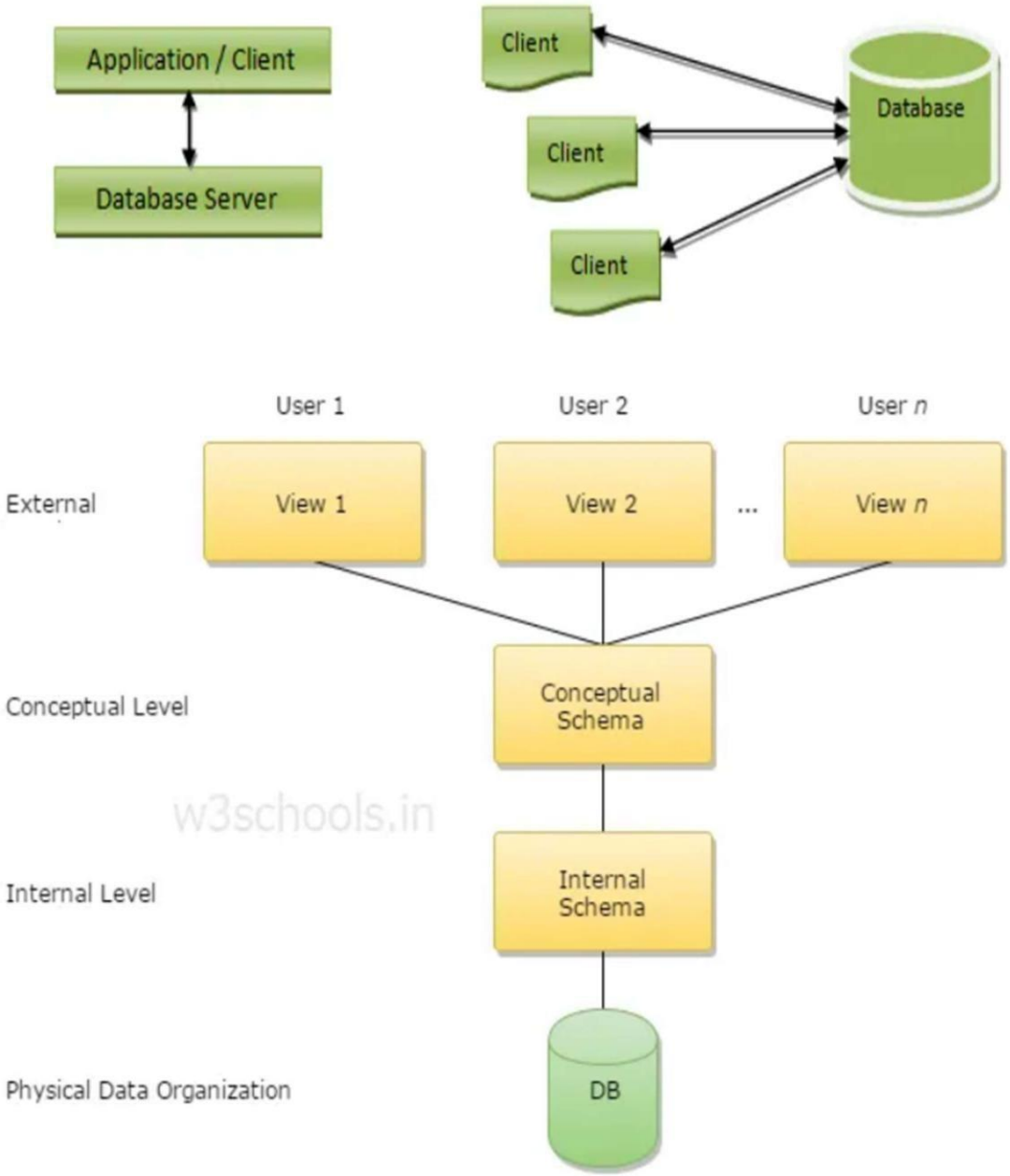


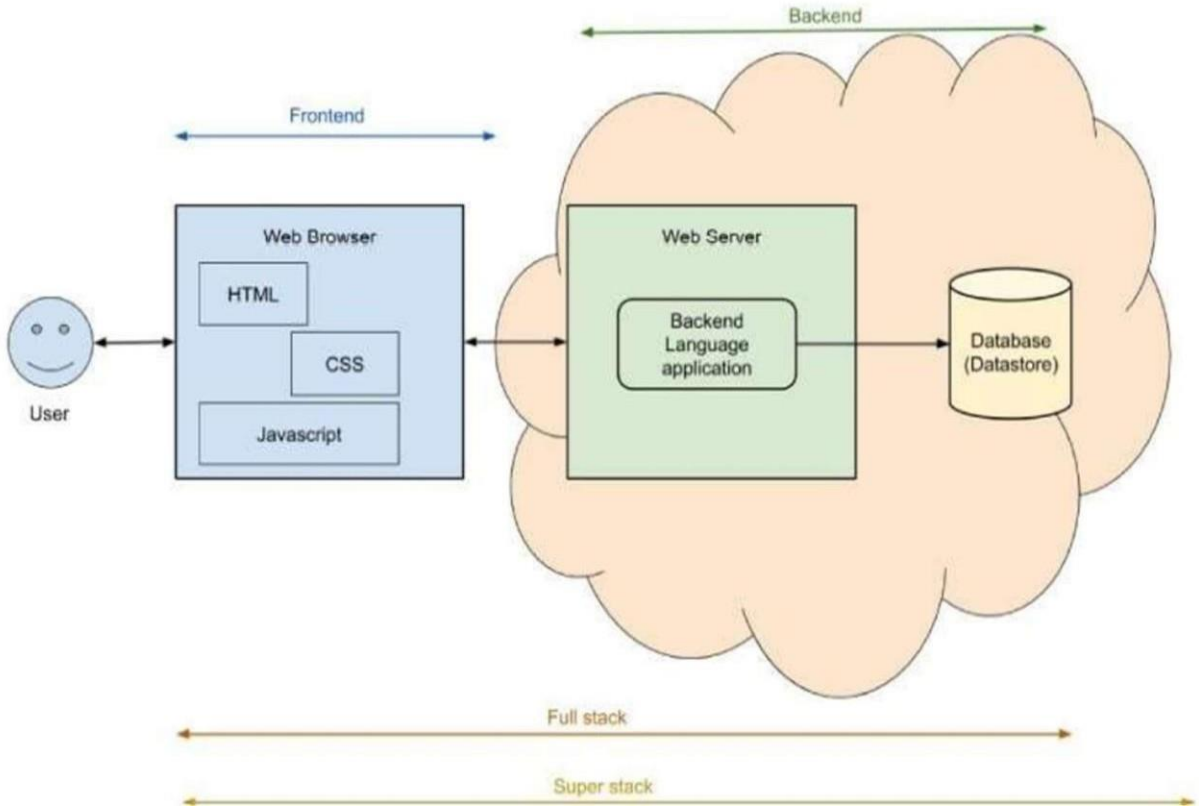
FIG: ARCHITECTURE OF DATABASE SYSTEM

- A Database Architecture is a representation of DBMS design. It helps to design, develop, implement, and maintain the database management system. A DBMS architecture allows dividing the database system into individual components that can be independently modified, changed, replaced, andaltered.
- The Database Management System (DBMS) architecture shows how data in the database is viewed by the users. It is not concerned about how the data are handled and processed by the DBMS. It helps in implementation, design, and maintenance of a database to store and organize information forcompanies.

Entities in the Database include:

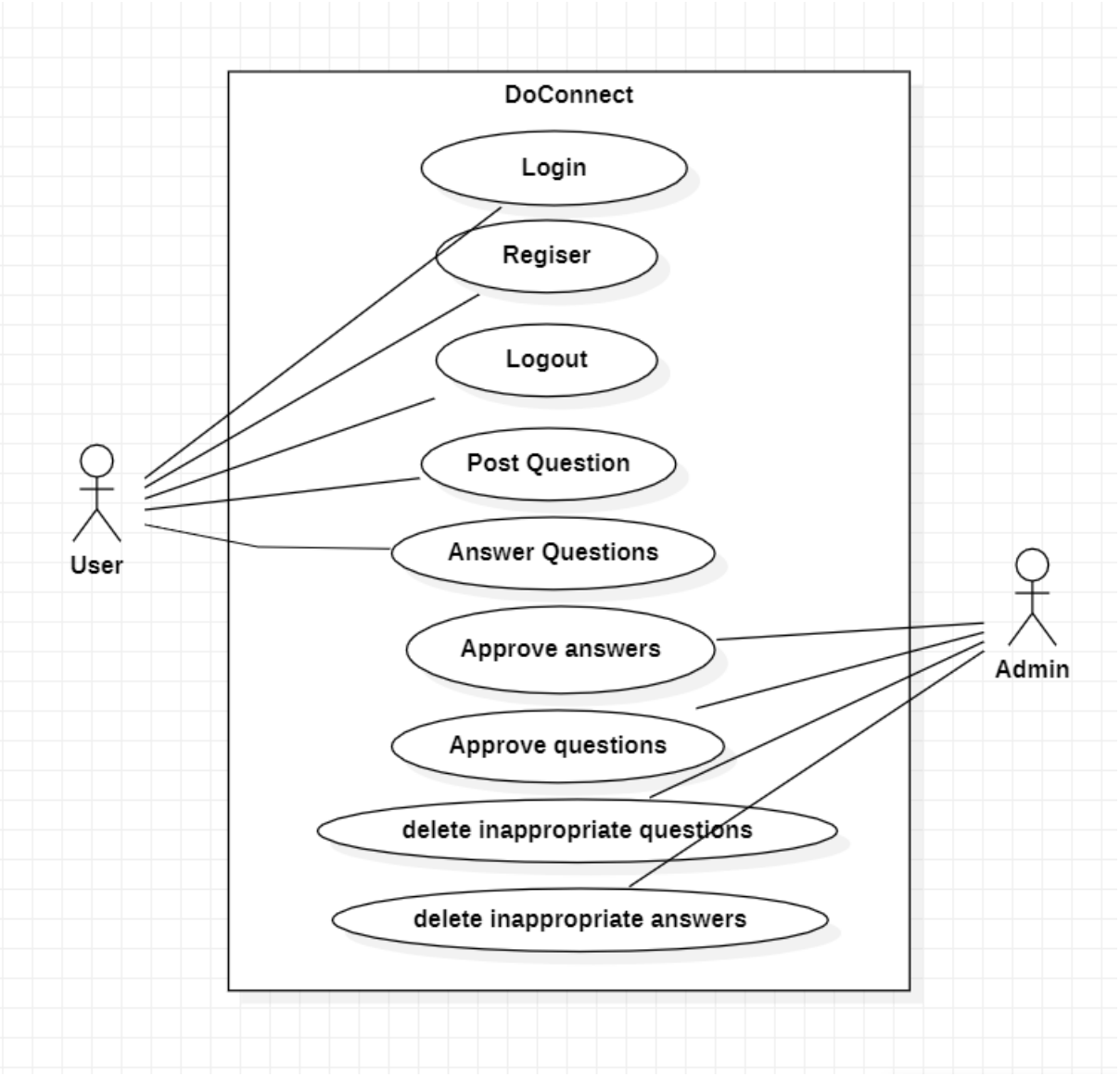
- Admin
- Answer
- Question
- User

TOTAL PROJECT OVERVIEW



Full-stack-web-development entails the creation of both the frontend and backend of a website or application. Front end and back end developers are usually necessary for every web development project; however, a full stack developer performs both.

FUNCTIONALITY OFUSER



- User should be able to login, Logout and Register into the application.
- User should be able to ask any questions under any topic.
- User should be able to search the questions on any string written in search box.
- User should be able to answer any question asked.
- User should be able to answer more than one question and more than one time.
- User should be able to chat with other users
- User should be able to upload images to refer which I want but don’t want to order now.

FUNCTIONALITY OF ADMIN

- Admin should be able to login, Logout and Register into the application.
- Admin should be able to get mail as soon as any new Question is asked or any Answers given.
- Admin should be able to approve the question and Answer.
- As an Admin I should be able to delete inappropriate Questions or Answers

TECHNOLOGIES USED

Front End:

-Angular

Backend:

-Java

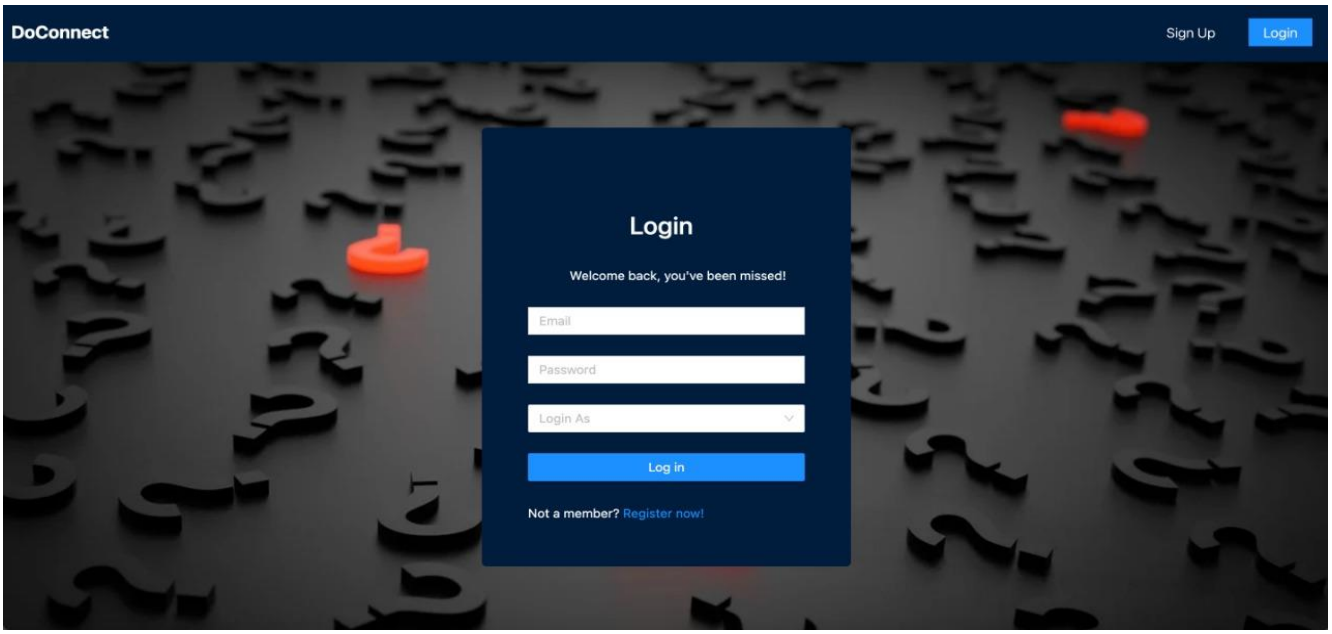
-Spring Boot

-JWT

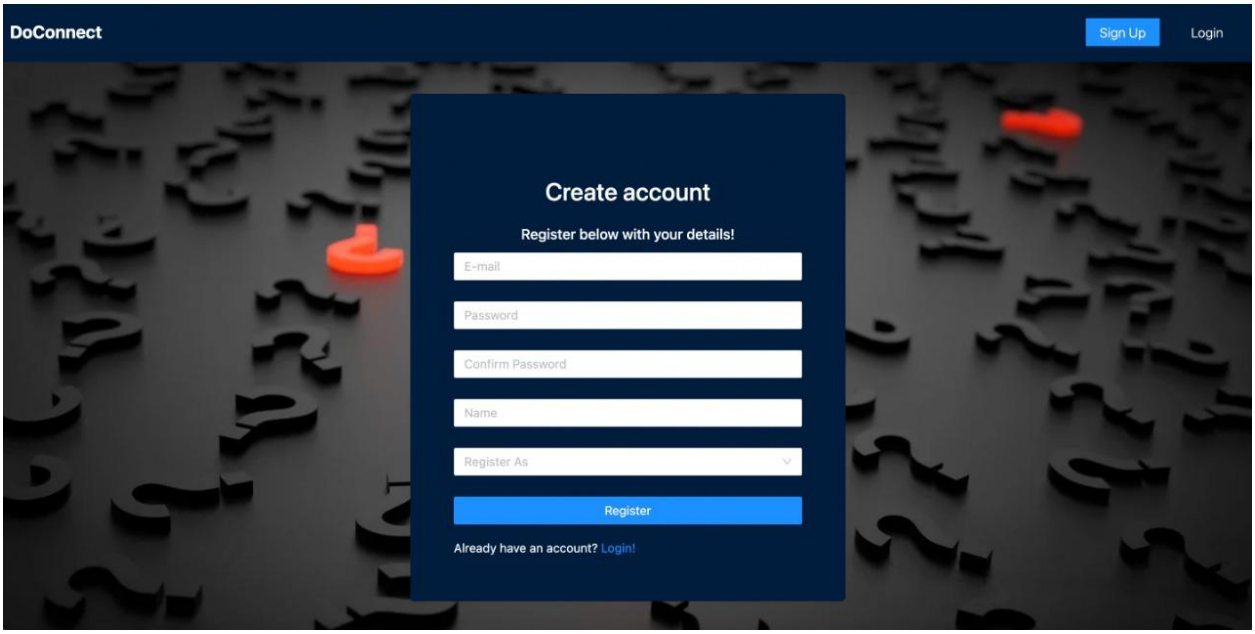
-Spring Security

RESULTS

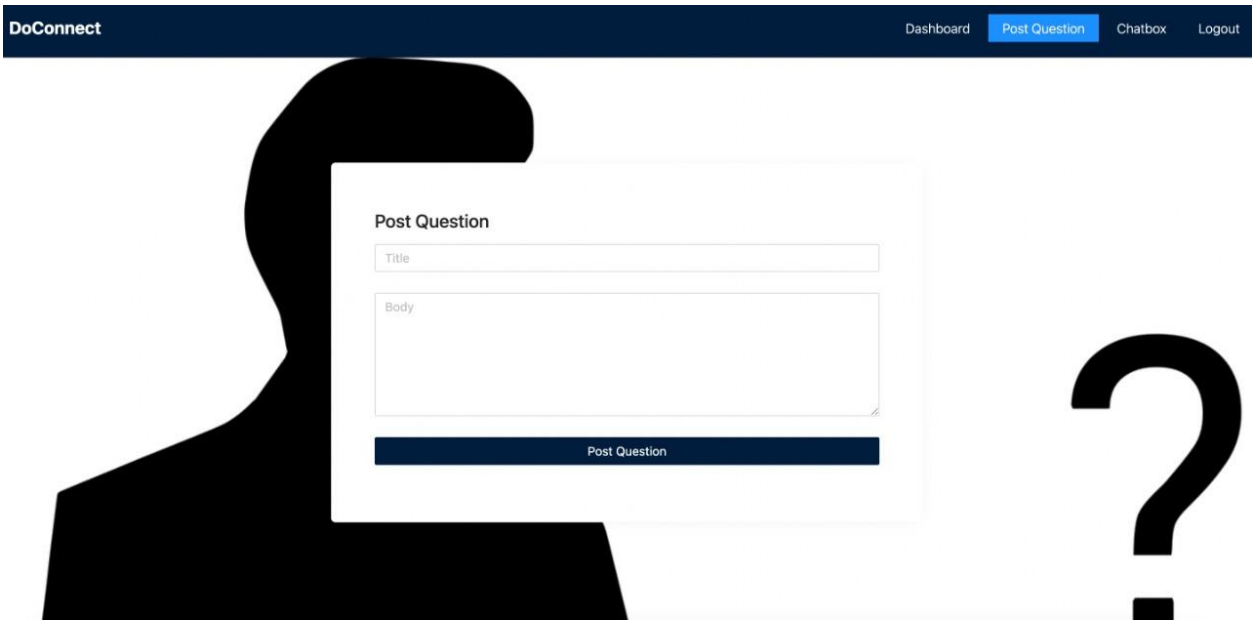
- Login Page



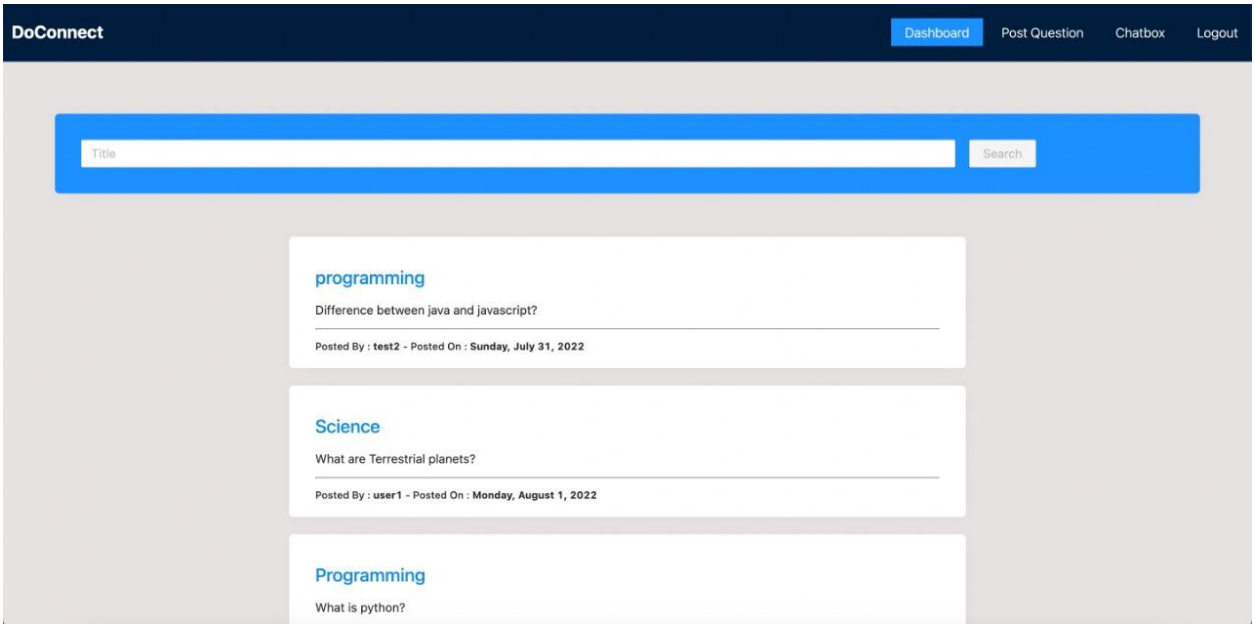
- Register Page



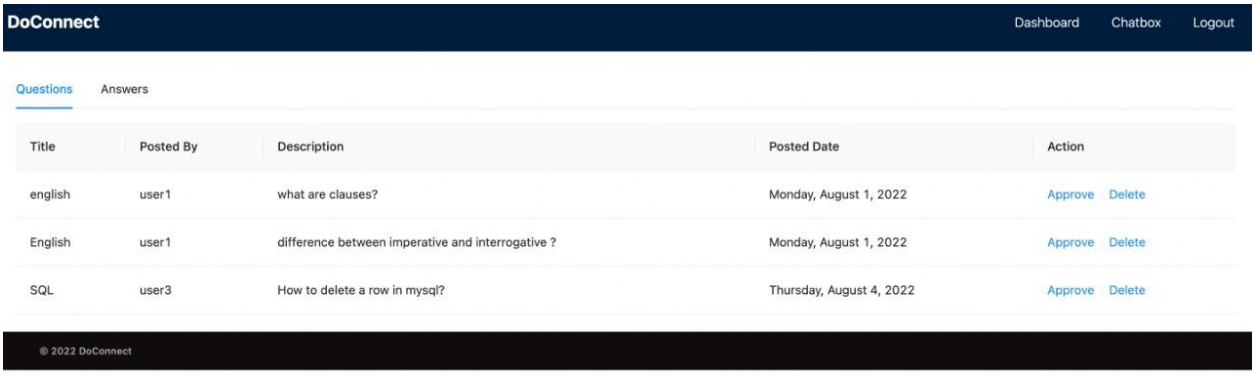
- Post a Question



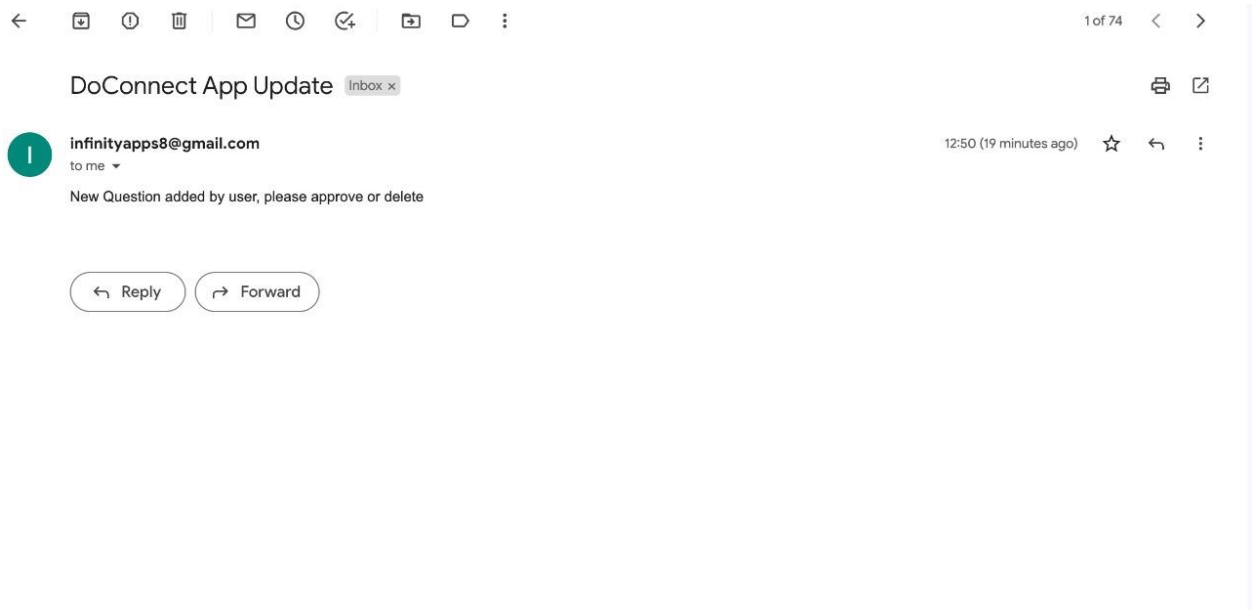
- Dashboard



- Admin Dashboard

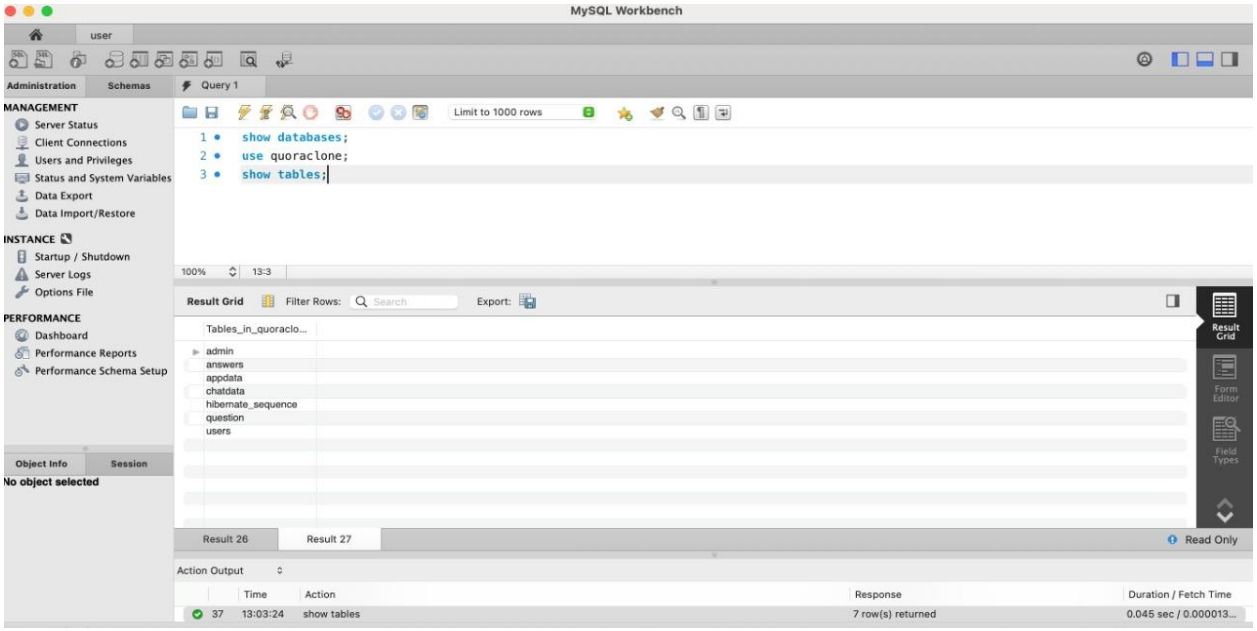


- Update for Admin

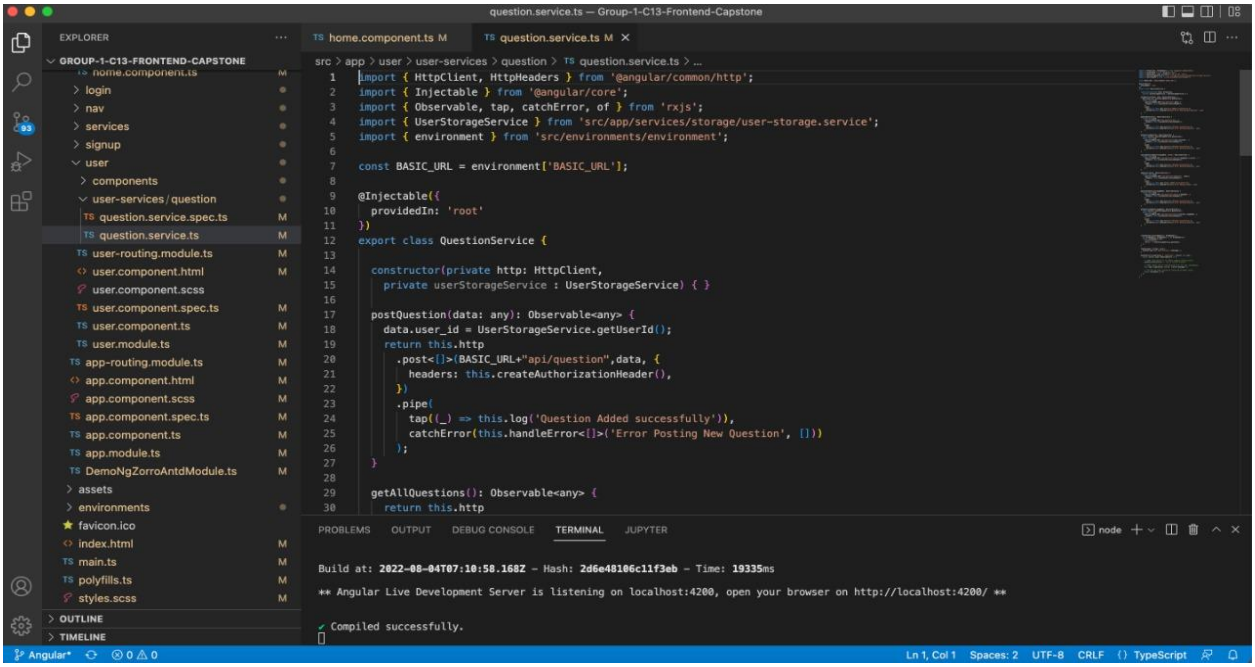
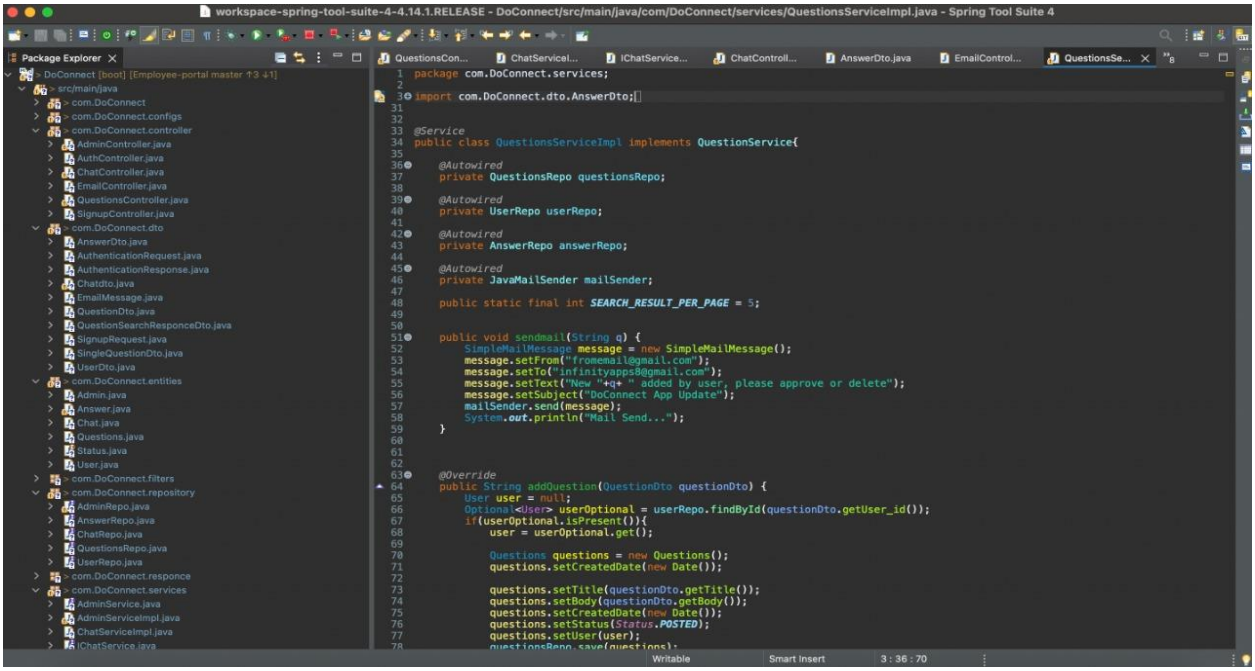


Code Structure

Structure of Database



STRUCTURE OF FRONTEND:



STRUCTURE OF BACKEND: