Covid Support Center

SOAD Group 28

1. Utkarsh Aditya   - S20180010182
2. Vipul Rawat   - S20180010192
3. Pradum Singh   - S20180010136
4. Sumanth Bhat   - S20180010171
5. Sushant Bondle - S20180010030

Problem Statement

- Coronavirus disease (COVID-19) is an infectious disease caused by a newly discovered coronavirus.
- Shortage of beds to Remdesivir doses, the sudden surge in Covid cases across the country has left government and private hospitals scrambling.
- Amid the crisis, people on social media have come together, trying to help Covid patients and those in need about availability of resources.
- People have been using vague platform like Twitter, Instagram, and Telegram.

Problem Solution - Covid Support Center

- So instead of a vague platform like Twitter, Instagram, and Telegram, we have created a dedicated web app towards the Covid-19 pandemic support.
- People can post questions on the website about any topic they wish related to the pandemic.
- People can answer questions posted by others on the website and provide important information.
- Users will have to register and login to post questions or answers. They can maintain their profile on the website.

Covid Support Center Features

- Authentication - (**Login** or **Register** using username and password)
- CRUD for modules of Questions, Answers and Comments
- Verify answers by voting
- Tags for organizing questions
- Search Questions
- Pagination of questions
- Proper Responsive UI for all screens.
- User and Profile Module.
- Documentation using Apollo Client

04

Unique selling points

- **Faster Search Process**

  Reduced turnaround time by allowing the probable recipients to search and find their required bed, oxygen cylinder, vaccines, etc.

- **Tag based Organization**

  It will be easy to search provided the tags, for example any medical organization can just enter a city name as tag and find out all the questions regarding their tag.

Potential Users

- **The Medical Organisations**

  Using the application medical organizations can search for patients on our website or potential vaccine buyers.

- **Covid Patients**

  Using the application patients can post questions regarding their requirements of bed or vaccine and also find posts about other medical organisations and their healthcare features.

Stack Details

- **ReactJS** - Frontend Framework and To Create interactive UI
- **Apollo Client** - State management library to manage both local and remote data
- **Node.js** - Runtime environment for JS
- **Apollo Server** - To build a self-documenting GraphQL API server
- **MongoDB** - Database to store document-based data
- **Nodemon** - For Dynamic Building
- **React Hook Form** - For flexible forms
- **Dotenv** - To load environment variables from a .env file

EAD features implemented - Scalability

- Mongoose Unique Validator - Plugin for better error handling of unique fields within Mongoose schema
- Dockerizing MERN stack application for scalability
- Utilizing Cloud Computing platform for Load Balancing

```
>>> print([requests.get('http://20.193.244.10/').text for _ in range(100)])
['S20180010182', 'VM2\r\n', 'S20180010182', 'VM2\r\n', 'S20180010182', 'VM2\r\n', 'S20180010182', 'VM2\r\n', 'S20180010182', 'VM2\r\n', 'S20180010182', 'VM2\r\n', 'S20180010182', 'VM2\r\n'
, 'S20180010182', 'VM2\r\n', 'VM2\r\n', 'VM2\r\n', 'S20180010182', 'VM2\r\n', 'S20180010182', 'VM2\r\n', 'S20180010182', 'VM2\r\n', 'S20180010182', 'VM2\r\n', 'S20180010182', 'VM2\r\n', 'S
20180010182', 'VM2\r\n', 'S20180010182', 'VM2\r\n', 'S20180010182', 'VM2\r\n', 'VM2\r\n', 'S20180010182', 'VM2\r\n', 'VM2\r\n', 'S20180010182', 'VM2\r\n', 'S20180010182', 'VM2\r\n', 'S2018
0010182', 'VM2\r\n', 'S20180010182', 'VM2\r\n', 'S20180010182', 'VM2\r\n', 'S20180010182', 'VM2\r\n', 'S20180010182', 'VM2\r\n', 'S20180010182', 'VM2\r\n', 'S20180010182', 'VM2\r\n', 'S201
80010182', 'VM2\r\n', 'S20180010182', 'VM2\r\n', 'S20180010182', 'VM2\r\n', 'S20180010182', 'VM2\r\n', 'S20180010182', 'VM2\r\n', 'S20180010182', 'VM2\r\n', 'S20180010182', 'VM2\r\n', 'S20
180010182', 'VM2\r\n', 'S20180010182', 'VM2\r\n', 'S20180010182', 'VM2\r\n', 'S20180010182', 'VM2\r\n', 'S20180010182', 'VM2\r\n', 'S20180010182', 'VM2\r\n', 'S20180010182', 'VM2\r\n', 'S2
0180010182', 'VM2\r\n', 'S20180010182', 'VM2\r\n', 'S20180010182', 'VM2\r\n', 'S20180010182', 'VM2\r\n', 'S20180010182', 'VM2\r\n', 'S20180010182', 'VM2\r\n', 'S20180010182', 'VM2\r\n', 'S
20180010182', 'VM2\r\n']
>>>
```

EAD features implemented - Security

- Yup - For form validation
- Bcrypt.js - For hashing passwords
- JSON Web Token - A standard to secure/authenticate HTTP requests
- ASP.NET - to generate anti-CSRF security tokens

Contributions

Name	Feature
Utkarsh Aditya	Scalability
Vipul Rawat	Security
Pradum Singh	Scalability
Sumanth Bhat	Security
Sushant Bondle	Security

Work Distribution

- The entire workload of the enterprise application was split in such a way that every member will have to work equal number of hours on their task.
- Every Component was completed by two or more group members so they have been mentioned in their individual contributions separately.
- This was done to promote understanding of the entire application by each group member.

Contributions - Utkarsh Aditya (S20180010182)

- Schema Design using GraphQL and Apollo Client
- Models - Question and Answer Module
- Documentation - GraphQL Playground
- Mongoose Unique Validator - Plugin for better error handling of unique fields within Mongoose schema
- Bcrypt.js - For hashing passwords
- Setting up the environment for mongoDB connection.
- Frontend design - Logo setting and images across application

Contributions - Pradum Singh (S20180010136)

- Model - User Module
- Authorization and login using JWT
- Dockerizing MERN stack application for scalability
- Utilizing Cloud Computing platform for load balancing
- Users Component integration in the application
- Frontend and backend - Users List page along with integration
- GraphQL

13

Contributions - Vipul Rawat (S20180010192)

- Yup - For form validation
- Bcrypt.js - For hashing passwords
- JSON Web Token - A standard to secure/authorize HTTP requests
- ASP.NET - to generate anti-CSRF security tokens
- Model - Comment Module
- Frontend and backend - Tags List page along with integration

14

Contributions - Sumanth Bhat (S20180010171)

- Frontend and backend - Profile page along with integration.
- Frontend Designing - Color scheme, size, logos, pictures, etc.
- Schema Design
- Helped with User Module
- Navbar component and search box

Contributions - Sushant Bondle (S20180010030)

- Frontend and backend - Detail question view page with integration.
- Frontend Designing - Color scheme, size, logos, pictures, etc.
- Worked on the utility functions for error management
- Helped with Question Module
- Pagination

16

References

- YUP api - https://www.npmjs.com/package/yup#api
- Hook forms - https://react-hook-form.com/
- Material UI - https://material-ui.com/
- GraphQL for js - https://graphql.org/graphql-js/
- Json web token - https://jwt.io/
- Tutorials and web sources -
  - https://www.youtube.com/watch?v=Quk1-LhG03E
  - https://www.youtube.com/watch?v=78r0MgH0u0w

Thank You