CP 301:
22x7 dashboard

Beneficiary: DC Office, Ropar

Dept. of Computer Science and Engineering

Group: A7

Sparsh Agarwal (2018CSB1126)
Arnav Bansal (2018CSB1075)

Mentor: Dr. Puneet Goyal

Daksh Sharma (2018CSB1082)
Ritvik (2018CSB1115)

Problem Statement

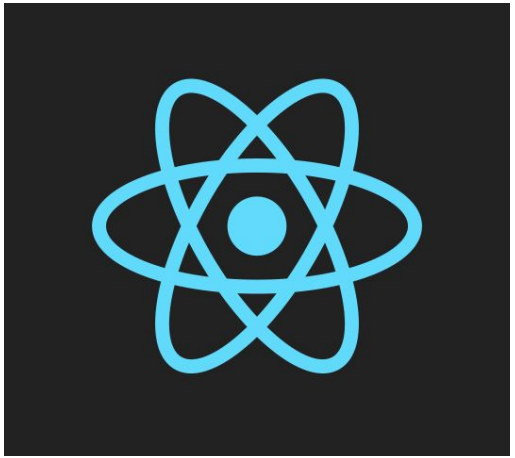
- To develop a web-based dashboard where all the data management, visualization, and storage needs of the DC Office can be taken care of
- Should have the ability to upload spreadsheets
- Data visualization of the uploaded data in the form of various graphs must be possible
- Data can come in various types, like healthcare, taxation, government policies, agriculture etc. Hence, the dashboard has to have a separate provision for them

Challenges currently faced by the beneficiaries

- The Employees at the DC Office currently have to maintain a large number of spreadsheets to keep track of data for various departments of the district
- It is difficult to manually map such a large amount of data into a visually understandable format (graphs)
- With the 2nd wave of COVID-19, the employees are facing a difficulty in maintaining multiple and analysing multiple spreadsheets
- Without any visual representation, it becomes difficult to easily understand and compare the data

Technologies used

React is an open-source front-end JavaScript library for building user interfaces or UI components. It is the base of the frontend of our application



React JS

Libraries Used with React

Ant Design

- Library that includes many pre-built reusable components

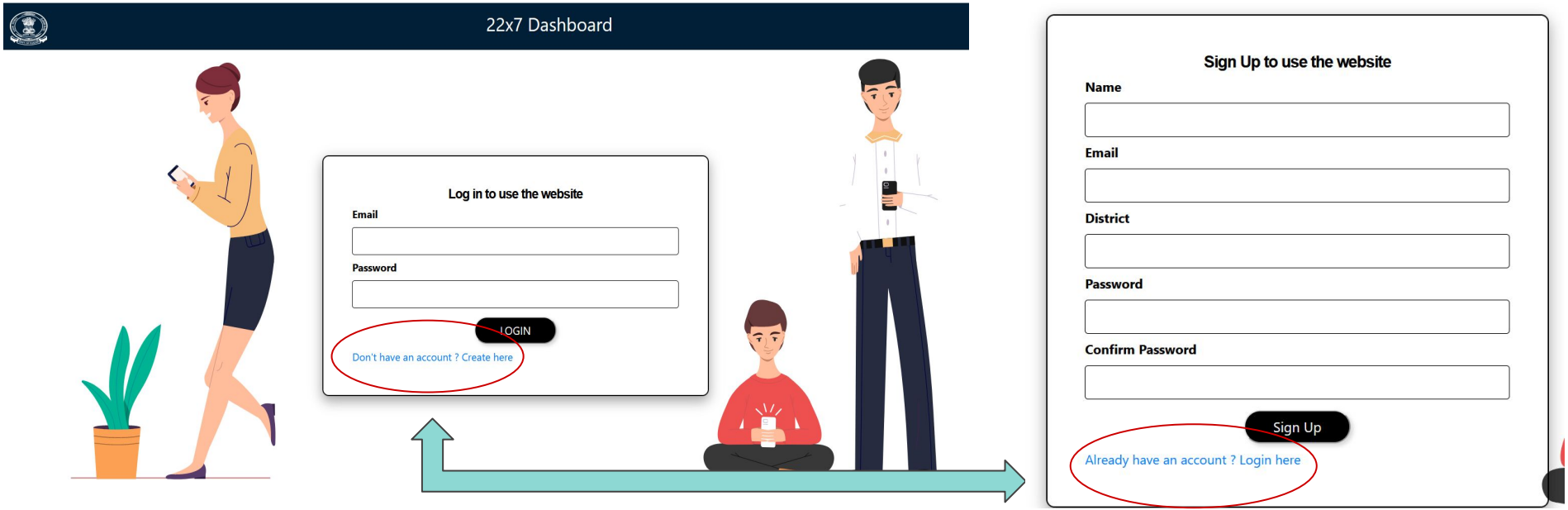
Chart.js

- Library Used for creating all the charts used in the application

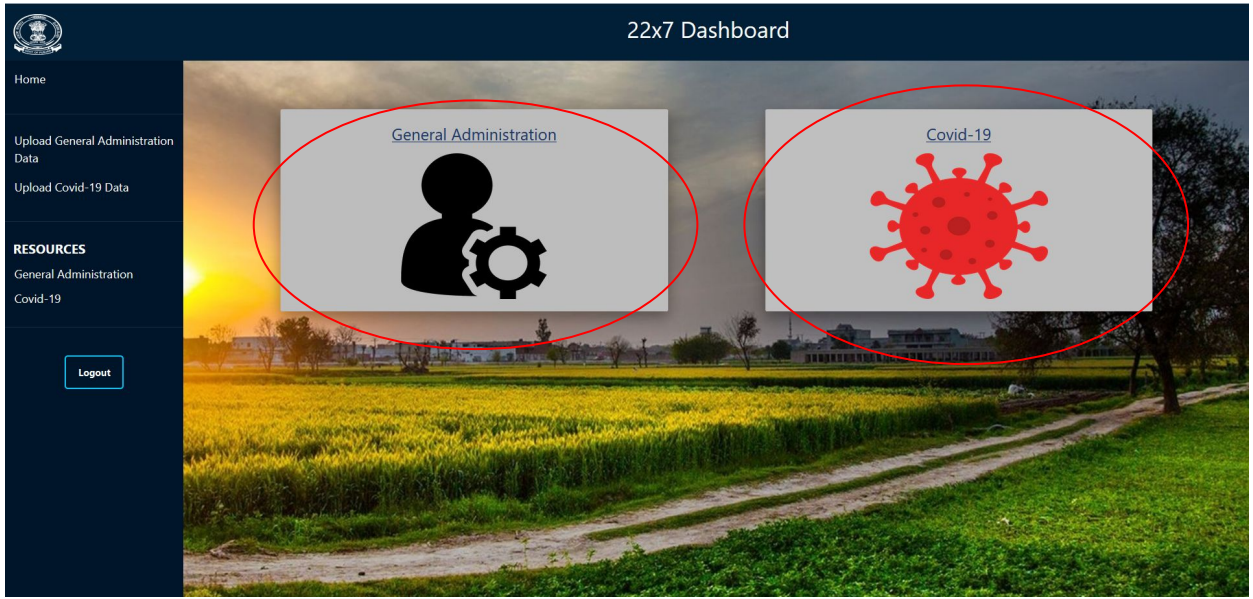
Technologies used

Django is a Python-based free and open-source web framework that follows the model-template-views architectural pattern. Under Django we have used REST framework to create the APIs as required by the frontend. REST is a loosely defined protocol for listing, creating, changing, and deleting data on your server over HTTP

Login And Sign-Up Pages



Homepage



Two Main sections, General Administration and Covid-19

Side-Bar

- Made for easier navigation inside the application
- Present in every page and takes the user to the desired page with a single button click
- Also contains a button, which can be used to logout of the application whenever needed

General Administration Dashboard

₹ Revenue | Court | Healthcare | Drug Abuse | Smart Ration

Data Type	District	From		To			
Collection of Excise	Amritsar	January	2021	January	2021	GO	Reset

Revenue Collection in 2020
₹ 20,18,30,670

Revenue Collection in 2021
₹ 5,46,68,640

Expected Collection in 2021
₹ 21,02,64,000

Percentage collected so far
26%

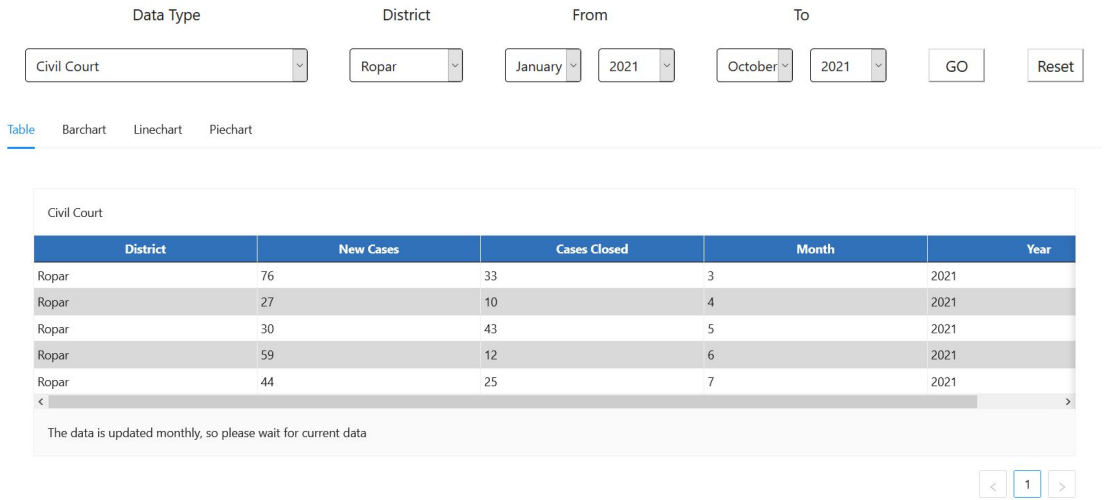
General Administration Dashboard

- Various tabs present under General Administration for easy navigation across different data types

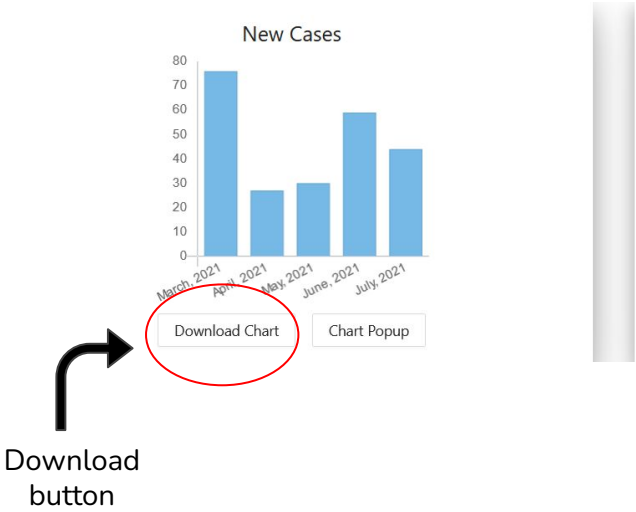
- Important data displayed under each data tab as highlights

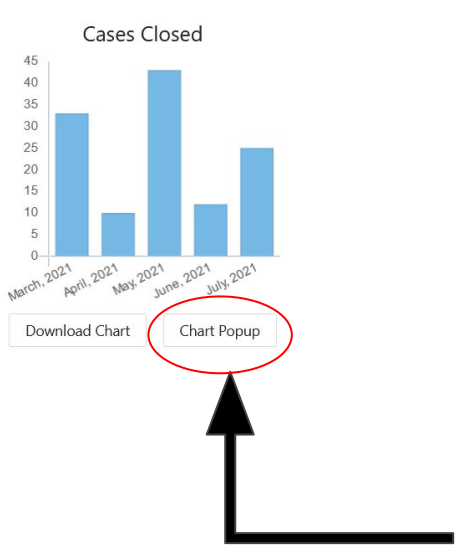
- Search for data relevant to you via various filters

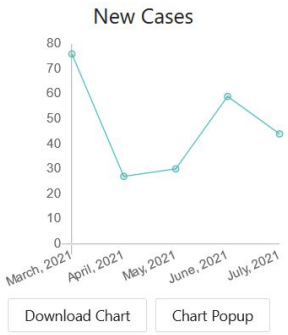
Search Query Output



- Query Output results in a table, containing relevant data
- Four choosable options - Table, BarChart, LineChart, Pie-Chart
- The Charts show visual representation of each column in the table







- Each graph has a download button which can be used to download the graph as a .png file
- Chart Popup button shows the graph in a popped up Modal (better for viewing charts with many data points)

Covid-19 Dashboard

Contact Tracing | Covid Positivity | Deaths

From: dd / mm / yyyy

To: dd / mm / yyyy

District and Sub-Division: Ropar | Ropar

Search: GO

Reset

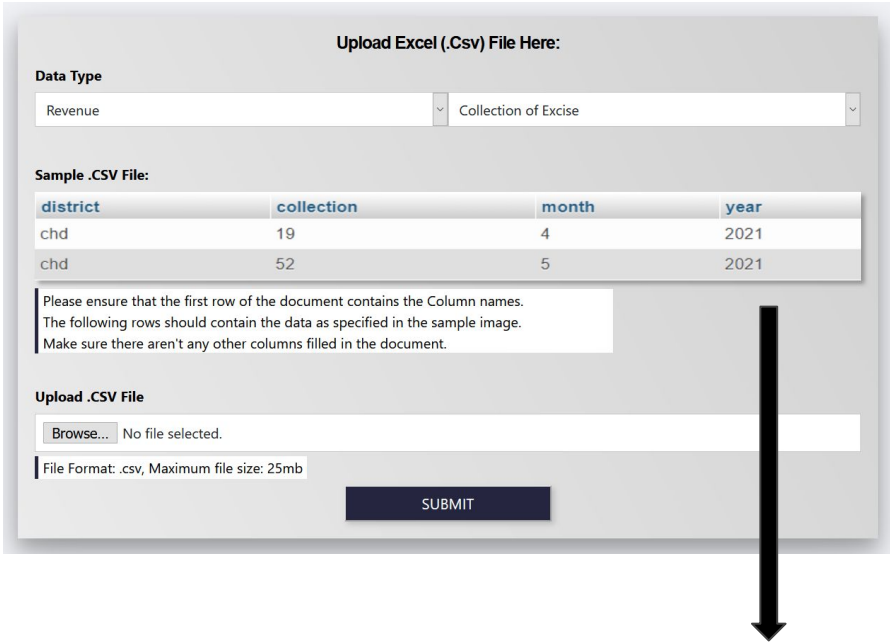
Total Contacts Traced in last Month
29,230

Total Positive Cases Traced in Last Month
2,460

Percentage of Contacts Traced
88.3%

Average Contacts per Case
22.5

Upload page for spreadsheets



- Select the type of data you want to update via the drop down menu
- A sample of the format of the sheet that is to be strictly followed has been shown

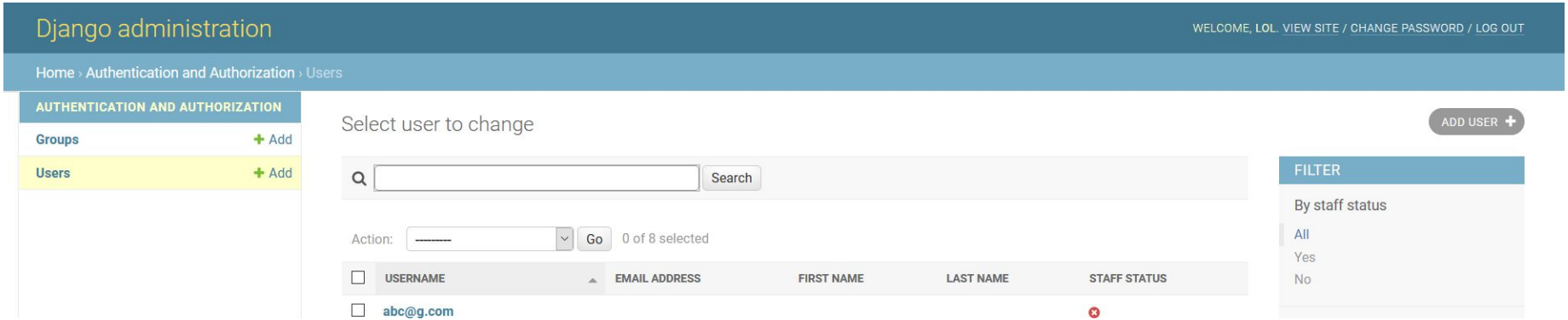
district	collection	month	year
chd	19	4	2021
chd	52	5	2021

The Database

Here are our tables as stored in our database via mySQL.

Below is an example of how data is stored in individual tables.

	District	GST Collected	GST Target	VAT Collected	VAT Target	Month	Year
Delete	Ropar	100	130	120	150	2	2021
Delete	Ropar	220	250	200	230	3	2021



Used In-Built Django User-authentication models as it satisfied the use-case perfectly and allowed for faster development

Individual Contributions

Sparsh Agarwal (2018CSB1126) -:

- Route Handling and Route Protection on Front-End
- Backend to Frontend integration to make the application truly dynamic
- Component-to-Component connection for correct flow of data
- Created Sidebar and some other components

Daksh Sharma (2018CSB1082) -:

- Added CSS Styling to make components look visually appealing
- Created the layout of most of the application's web pages
- Added Design changes to all of application webpages

Ritvik (2018CSB1115) and

Arnav (2018CSB1075):

- Worked on the entire backend
- Designing and setting up of the entire database via mySQL
- Created various Django REST endpoints for the API in the backend for the frontend to access
- Set up important functions such as parsing and storing the CSV file received from the user
- Created and processed different queries according to the filters selected by the user