Feedback Assessment System Using Sentiment Analysis

Group Members:-

- Nikhil Maurya (1901104)
- Dushyant Rai (1901067)
- Atik K. Nageshwar (1901046)
- Ankit Kumar (1901031)

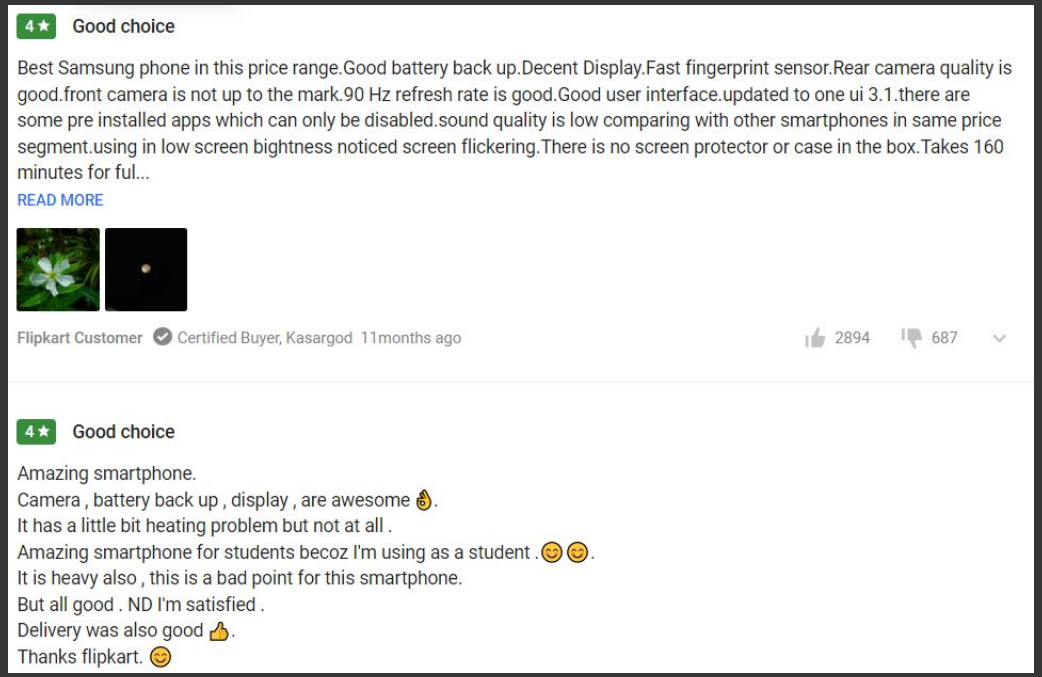
Purpose

Our system aims to provide an alternative way to derive insights from the feedbacks, received from the users of the services.

Why?

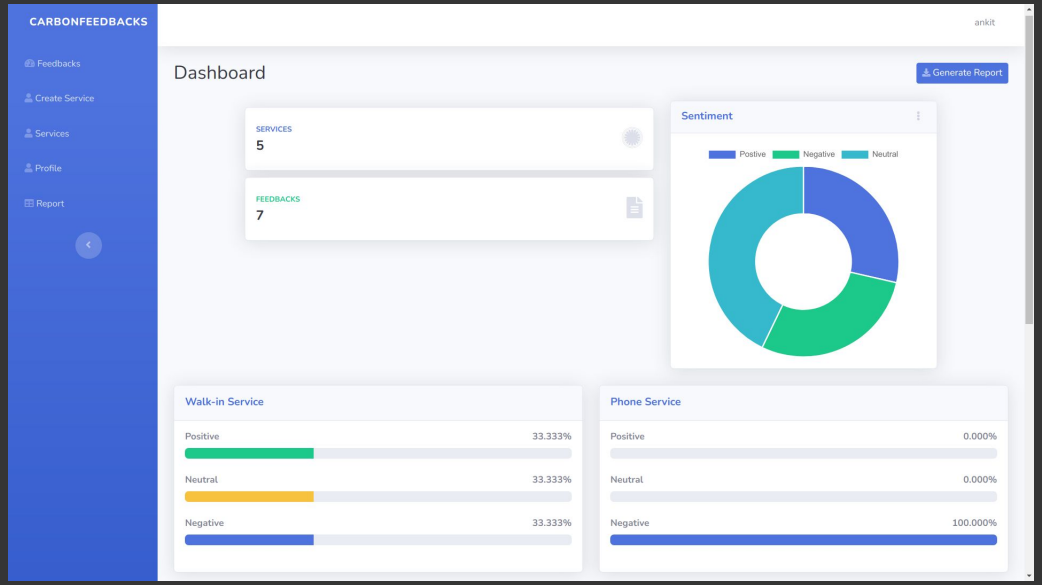
It is inefficient and expensive to analyse the textual feedbacks in large amounts whereas, star based rating systems are information deficient.

Existing Systems



Existing system includes star based / textual feedbacks systems. The textual feedbacks provide a rich insight but is very time-consuming to process in large amounts. In retrospect, star based systems are information deficient and do not provide any useful insight.

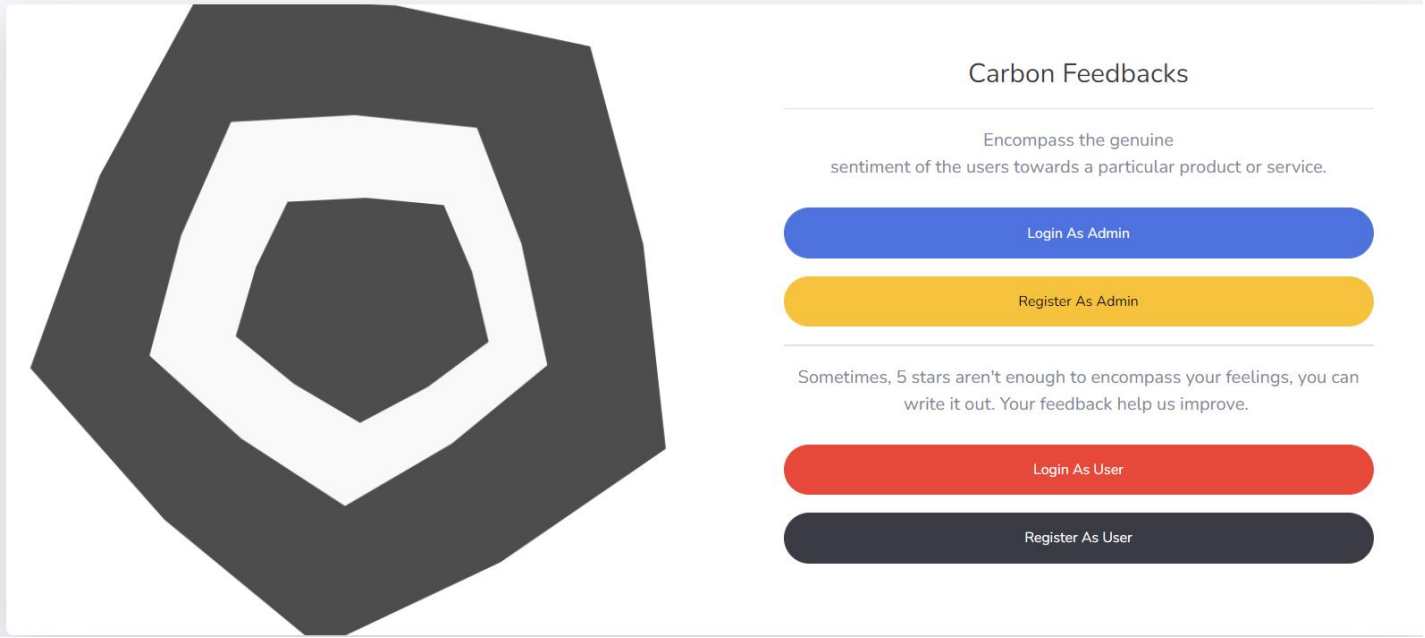
Proposed Systems



Our sentiment analysis based system aims to provide insightful and actionable data. Manual analysis of textual feedbacks is time-consuming and costly process. Our system aims to overcome this limitation by using machine learning techniques.

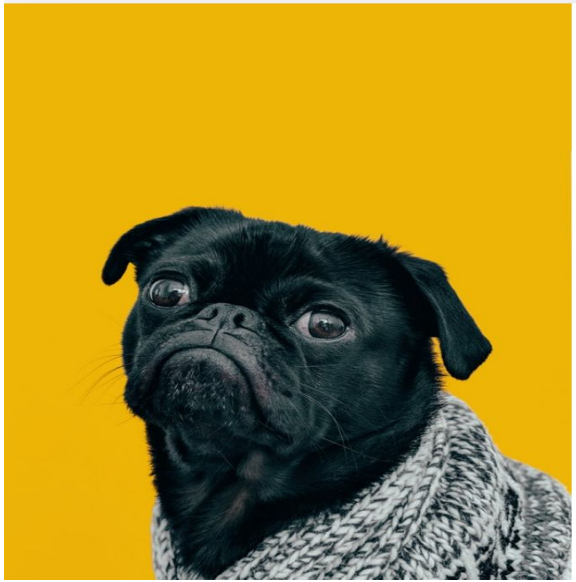
App Flow

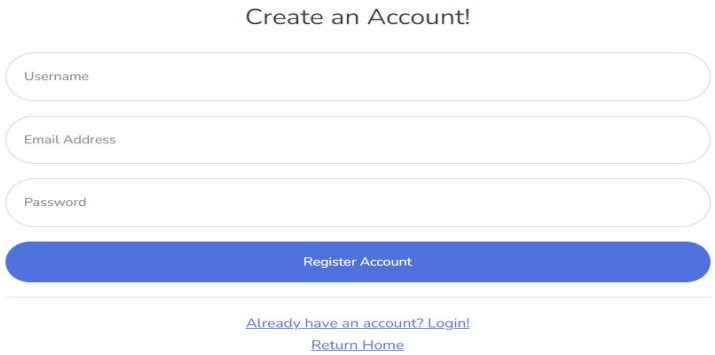
Home Page



An individual can provide feedbacks by registering and logging-in as an user. One can collect feedback for their services by registering as an admin.

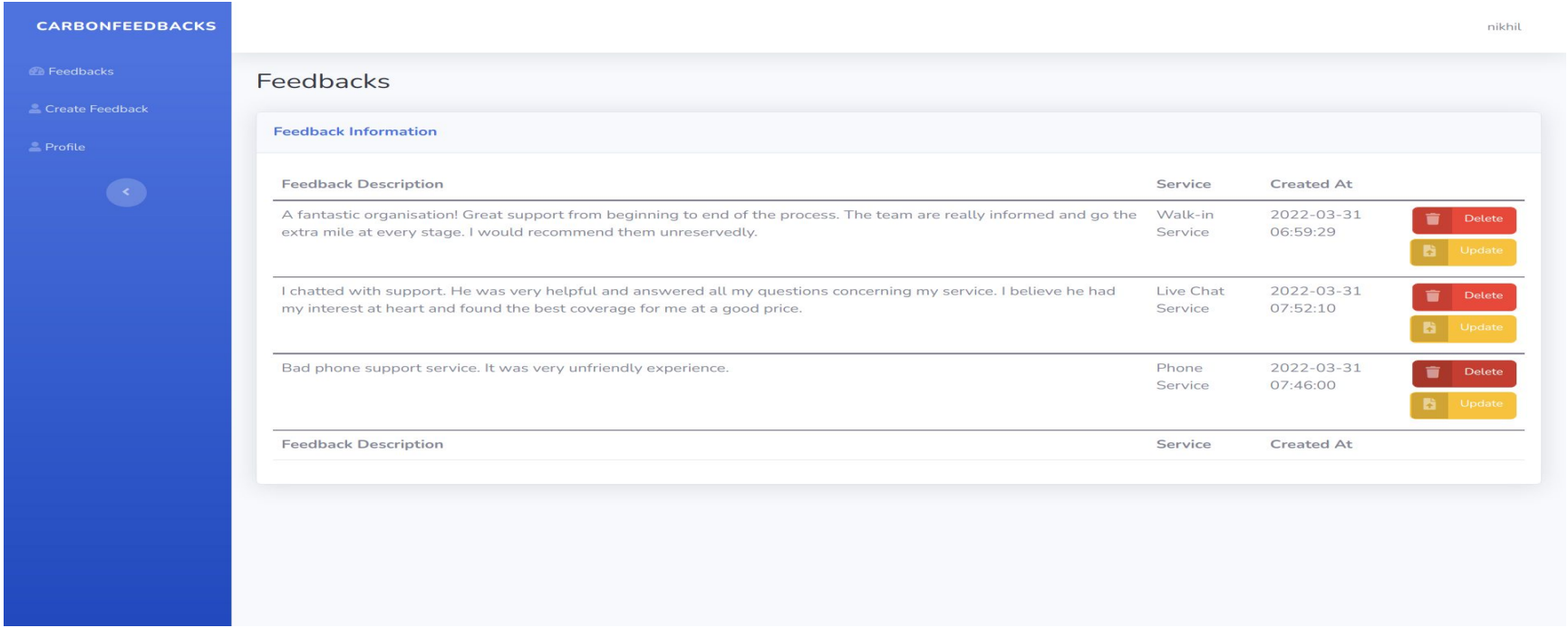
Registration





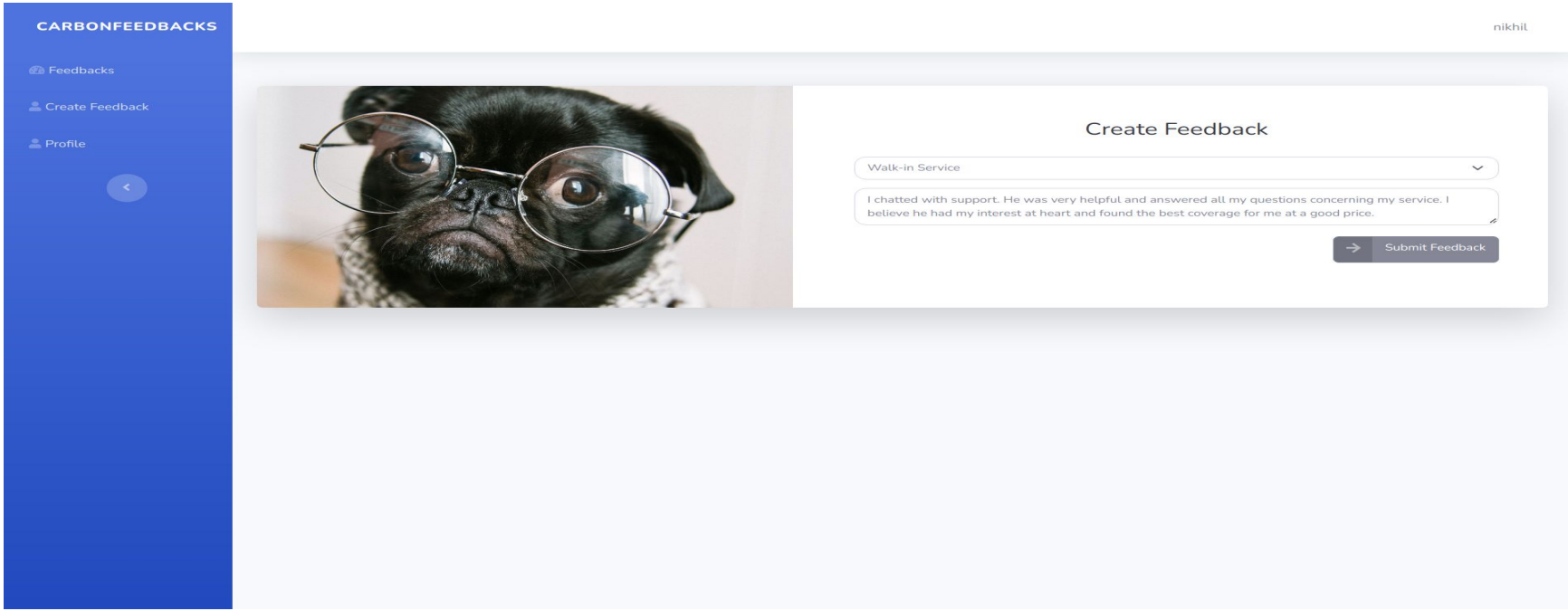
For registration, an individual will have to provide their username, e-mail address and password.

User Dashboard



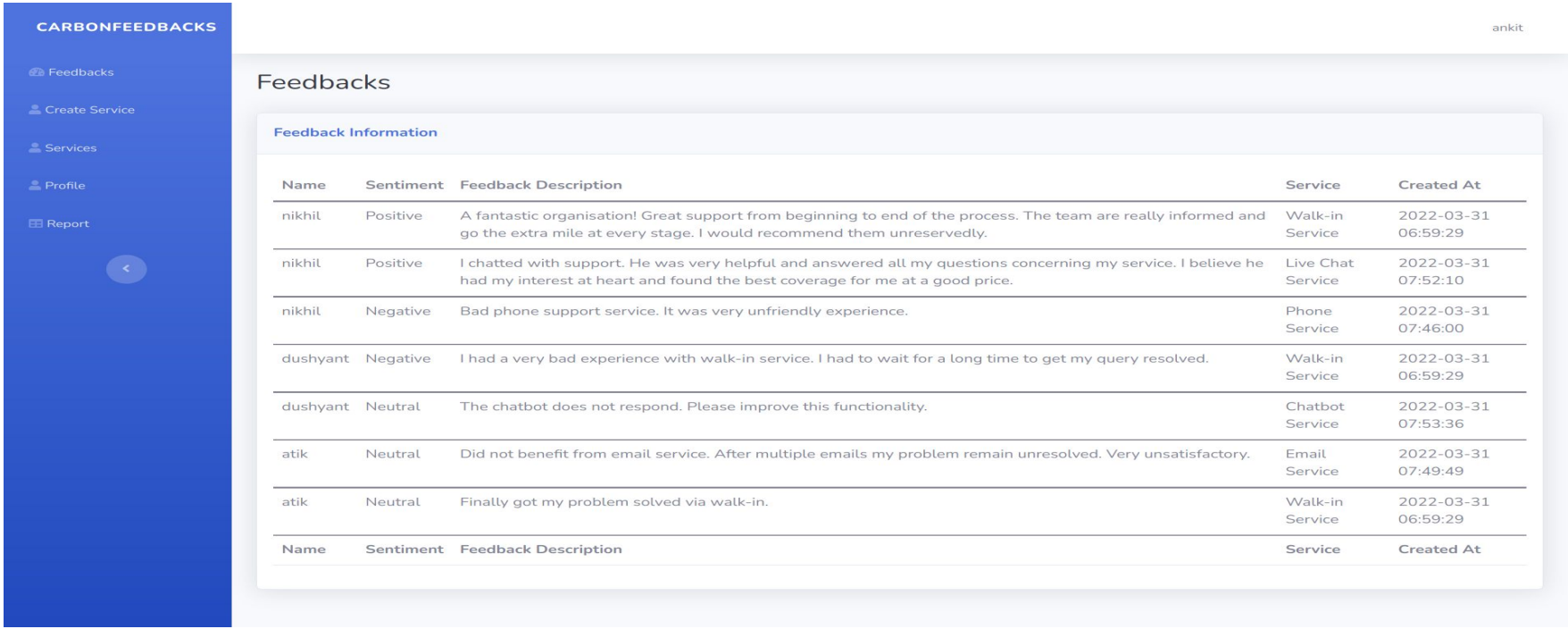
The user can view logged feedbacks associated with respective services on their dashboard.

Create Feedback



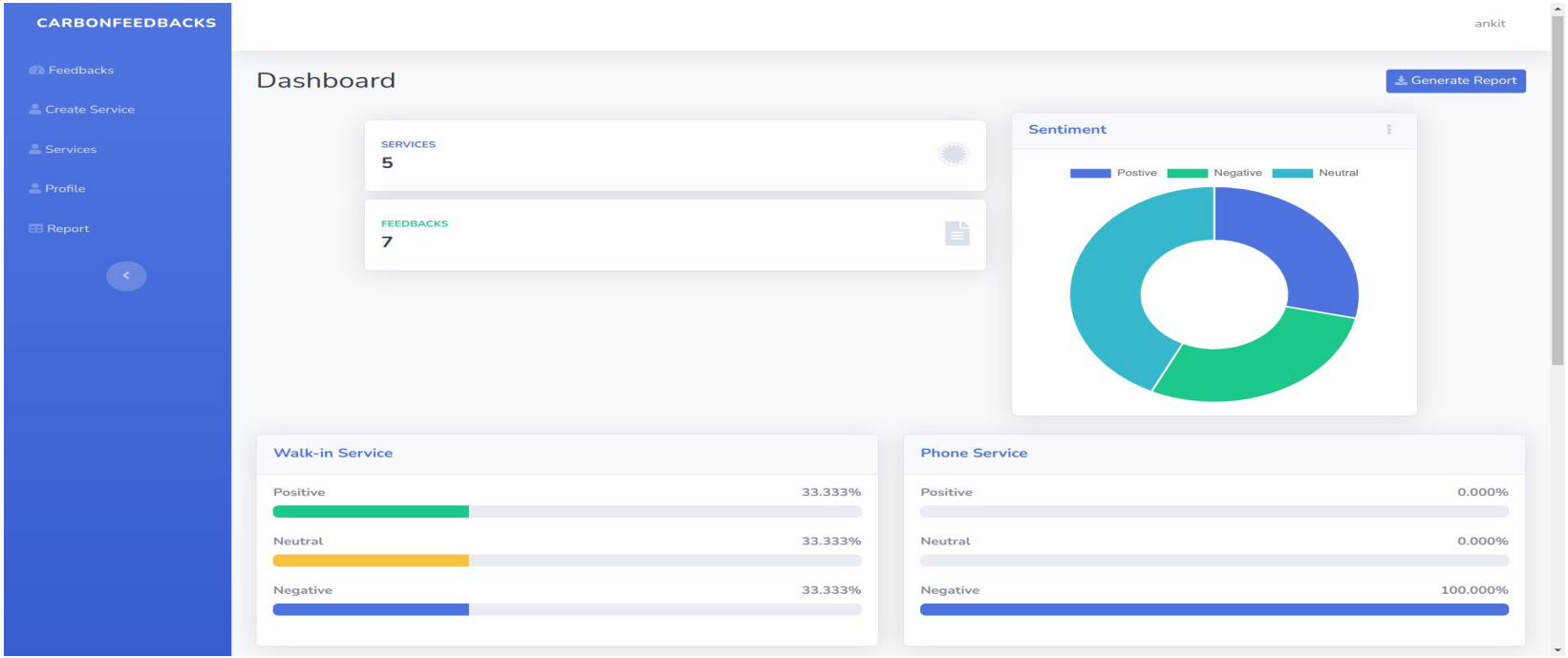
The user can give feedbacks by clicking on the create feedback button listed in the sidebar menu.

Admin Dashboard



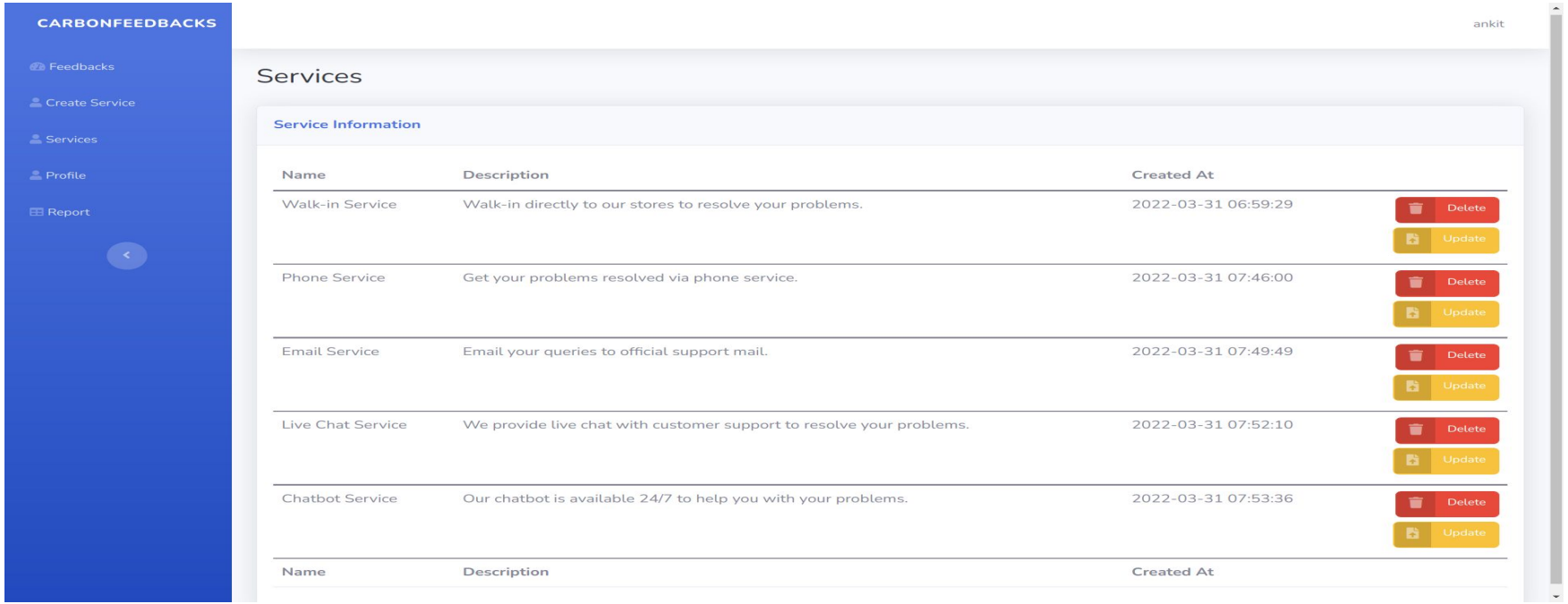
The admin can view user feedbacks associated with their listed services.

Service Report



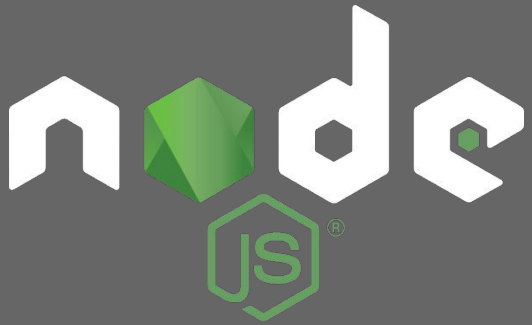
The admin can view the sentiment analysis report associated with their listed services.

Create Services



The admins can list a service by clicking the ‘create service’ button on the sidebar.

Technologies Used



Express

<%= EJS %>

spaCy

Contributions

Nikhil Maurya

1. Designed and implemented frontend using Bootstrap Framework.
2. Implemented authentication routes using passport.js
3. Implemented CRUD routes using expressjs.
4. Implemented rendering logic using EJS.

Dushyant Rai

1. Created SQL tables using DB Browser.
2. Implemented methods to interact with SQLite tables using Knex (SQL query builder).

Atik K. Nageshwar

1. Implemented Sentiment Analysis API using Flask.
2. Deployed the API on heroku.
3. Deployed the web application.

Ankit Kumar

1. Implemented CRUD routes using expressjs.
2. Implemented rendering logic using EJS.
3. Integrated SQL methods with CRUD API routes.

Thank You!