

A Project Report on

‘SPREAD THE SMILE- DONATION PLATFORM’

By

Ms. Banshika Gupta

(240310120004)

Ms. Jyoti Choudhary

(240310120017)

Mr. Uday Walia

(240310120014)

Guide

Mr. Prashant Kumar

Project Engineer, CDAC Delhi



A Project Report Submitted

at

Centre for Development of Advanced Computing

In partial fulfillment of the requirements

of

PG Diploma in Advanced Computing

[March 2024-August 2024]

Centre for Development of Advanced Computing
ACT'S, Delhi



CERTIFICATE

This is to certify that the project entitled “**SPREAD THE SMILE-DONATION PLATFORM**” is a bonafide work of “**Banshika Gupta (240310120004), Jyoti Choudhary (240310120017)**” **Uday Walia (240310120014)** submitted to C- DAC Delhi in partial fulfillment of the requirement for the award of the Post Graduate Diploma in Advanced Computing.

Mr. Prashant Kumar

Guide

ACKNOWLEDGEMENT

Presentation inspiration and motivation have always played a key role in the success of any venture. We express our sincere thanks to Mr. Ankit Khurana, Center for Development of Advance Computing (C-DAC), Delhi.

We pay our deep sense of gratitude to C-DAC, Delhi to encourage us to the highest peak and to provide us the opportunity to prepare the project. We are immensely obliged to our colleague for their elevating inspiration, encouraging guidance and kind supervision in the completion of our project.

We feel to acknowledge our indebtedness and deep sense of gratitude to our guide Mr. Prashant Kumar whose valuable guidance and kind supervision given to us.

CONTENTS

Abstract.....	1
1. Introduction	
1.1 Purpose.....	2
1.2 Scope.....	2
1.3 Problems Domain.....	3
1.4 Solution Domain	3
1.5 System Requirement	3
2. System Requirement Analysis	
2.1 Information Gathering.....	5
2.2 System Feasibility.....	6
3. System Analysis	
3.1 Use Case Diagram.....	7
3.2 Activity Diagram	9
4. Design	
4.1 Data Modeling.....	11
4.1.1 ER Diagram/Object Diagram.....	11
4.1.2 Flow Chart.....	12
4.1.3 Database.....	13
5. Implementation	
5.1 Frontend code.....	16
5.2 Backend Code.....	26
5.3 Result Analysis.....	33
6 Testing	
6.2 Testability.....	46
6.2.1 Observability.....	46
6.2.2 Decomposability.....	46
6.2.3 Stability.....	46
6.3 Testing Method Case.....	46
7 Future Work	47
8 Conclusion.....	47
9 Appendix	48
10 References	48

LIST OF TABLES

4.1.3 Database

4.1.3.1 Admin13

4.1.3.2 Donor 13

4.1.3.3 Donation Details 14

4.1.3.4 Recipient 14

4.1.3.5 Volunteer 15

ABSTRACT

The **Spread the Smile: Donation Platform** is designed to revolutionize charitable giving in India by addressing the current fragmentation in the donation process. NGOs, donors, and recipients often operate in isolation, leading to inefficiencies and missed opportunities. Spread the Smile aims to unify these efforts by providing a centralized, standardized platform that connects all parties seamlessly.

For donors, the platform offers guidance throughout the donation process, helping them make informed decisions about where their contributions will have the most impact. By providing insights into verified needs and ongoing projects, the platform ensures that donations are meaningful and effective.

On the recipient side, the platform verifies demands and coordinates with volunteers to ensure that aid is delivered quickly and accurately. This not only speeds up the process but also enhances the credibility and trustworthiness of the donation system.

Ultimately, Spread the Smile seeks to bring greater transparency, efficiency, and collaboration to the charitable sector, ensuring that resources are directed where they are most needed and making a significant difference in the lives of those in need.

1. INTRODUCTION

The '**Spread the Smile: Donation Platform**' aims to transform charitable giving in India by bridging the gaps between donors, NGOs, and recipients. Addressing the inefficiencies caused by fragmented efforts, this platform offers a unified solution that guides donors through a transparent and informed donation process, while ensuring that verified recipients receive aid swiftly through coordinated volunteers. By enhancing collaboration and streamlining operations, Spread the Smile seeks to create a more effective and trustworthy donation system, ensuring that resources are utilized where they are needed most.

1.1. Purpose

The purpose of the '**Spread the Smile: Donation Platform**' is to create a unified, efficient system for charitable giving in India. It aims to connect donors, NGOs, and recipients through a single, standardized platform, eliminating fragmentation and improving coordination. By guiding donors in making impactful contributions and ensuring that verified recipients receive aid swiftly through volunteers, the platform seeks to enhance the overall effectiveness, transparency, and credibility of the donation process.

1.2. Scope

The scope of the Spread the Smile: Donation Platform includes:

- a. **Integration of Stakeholders:** Bringing together donors, NGOs, and recipients into a single, cohesive system to facilitate efficient interactions and transactions.
- b. **Donation Process Management:** Providing tools and resources for donors to make informed decisions, track their contributions, and ensure their impact.
- c. **Recipient Verification:** Implementing processes to verify the needs of recipients and ensure that aid is directed to those who genuinely require it.
- d. **Volunteer Coordination:** Organizing and managing volunteers to ensure the prompt and accurate delivery of donations.
- e. **Transparency and Accountability:** Offering real-time tracking and updates to enhance trust and credibility in the donation process.
- f. **Data Analysis and Reporting:** Collecting and analyzing data to assess the

effectiveness of donations, identify areas for improvement, and report on the impact of contributions.

- g. **Scalability and Adaptability:** Designing the platform to accommodate growth and adapt to changing needs within the charitable sector.

1.3 Problem Domain

The **Spread the Smile: Donation Platform** addresses several critical issues in the charitable sector in India. Currently, the sector suffers from fragmentation, with NGOs, donors, and recipients operating in isolation, leading to inefficiencies and a lack of coordination. There is also a lack of standardization, resulting in inconsistent donation processes and difficulties in tracking contributions. Donors often face uncertainty about where to give and how their donations will be utilized, impacting their confidence in the process. Additionally, the absence of reliable verification systems for recipient needs can lead to misallocation or delays in aid. Inefficient aid delivery further exacerbates these problems, while limited transparency reduces accountability and trust. Spread the Smile aims to solve these challenges by offering a unified platform that enhances collaboration, provides clear guidance for donors, verifies recipient needs, and ensures prompt and accurate aid delivery, thereby creating a more effective and trustworthy charitable ecosystem..

1.4 Solution Domain

It offers a comprehensive solution to the inefficiencies and challenges faced in India's charitable sector. By integrating donors, NGOs, and recipients into a single, standardized system, the platform eliminates fragmentation and enhances coordination. It standardizes the donation process, providing a consistent and transparent framework for tracking and managing contributions, which simplifies operations and improves accountability. The platform also guides donors with detailed information about recipient needs and projects, increasing donor confidence and engagement. It includes robust mechanisms for verifying recipient needs, ensuring that aid is directed accurately and efficiently. Through coordination with volunteers, it facilitates swift and precise delivery of donations. Additionally, real-time tracking and updates enhance transparency, building trust among all parties involved. Overall, Spread the Smile creates a more effective, transparent, and reliable donation system, addressing existing inefficiencies and maximizing the impact of charitable efforts.

1.5 System Requirements

1.5.1 Hardware Requirements

- Processor: i3 and above
- RAM: 8GB

1.5.2 Software Requirements

- Client on Internet: Web Browser, Operating System (any).
- Internet Connection: (any).
- Data Base : MySQL Workbench
- Development End: Spring Boot
- Designing End: React
- Tools : MySQL Workbench 8.0 CE, VS Code, Spring Tool Site4Browse

2. SYSTEM REQUIREMENT ANALYSIS

2.1 Information Gathering

For the **Spread the Smile: Donation Platform**, comprehensive information gathering is essential to develop a solution that effectively meets the needs of all users and stakeholders. This process involves collecting detailed requirements from donors, NGOs, recipients, and volunteers. Donors' preferences for donation types and communication, NGOs' needs for support and project details, recipients' specific needs and verification documents, and volunteers' availability and task preferences must all be documented.

2.1.1 Functional Requirements

Functional requirements are those which specify what a system should do.

- Donor, Recipient and Volunteer can Register
- Donor, Recipient ,Volunteer and Admin can Login
- Donor can submit the available resources like cloths, food to admin.
- Recipient can request the resources like cloths, food to admin .
- Volunteer can help in transportation of the resources assigned by admin.
- Amin received the resources from recipient and send to the donor by the help of volunteer.

2.1.2 Non-functional Requirements

Those requirements which are not the functionalities of a system but are the characteristics of the system are called non-functional requirements.

- Secure access of confidential data with the use of encryptions.
- Full time available.
- Data Consistency.
- Data Reliability.

- Better component design to get better performance at peak time.
- Flexible service based on architecture will be highly desirable for future extensions.

2.2 System Feasibility

2.2.1 Technical Analysis

- It is the study of configuration of the system. It considers the technical requirements of the project.
- We have used Reactjs, MySQL workbench as the developer tool and spring boot components to develop the project.
- For storage of data in databases we have used MySQL.

2.2.2 Economical Analysis

It could also be referred to as cost/benefit analysis. It is used for evaluating the effectiveness of a system. In economic analysis the procedure is to determine the benefits and savings that are expected from a system and compare them with costs.

2.2.3 Operational Feasibility

It is dependent on human resources available for the project. This application will be able to handle the all Functionality efficiently. This application will save time and provide an all-time availability to the users. It is secure, time saving and allows users to access the website even from remote locations.

3. SYSTEM ANALYSIS

3.1 Use-case Model

Use Cases are used here to describe the most important sequence of scenarios between the User and the System.

3.1.1 Admin Use Case:

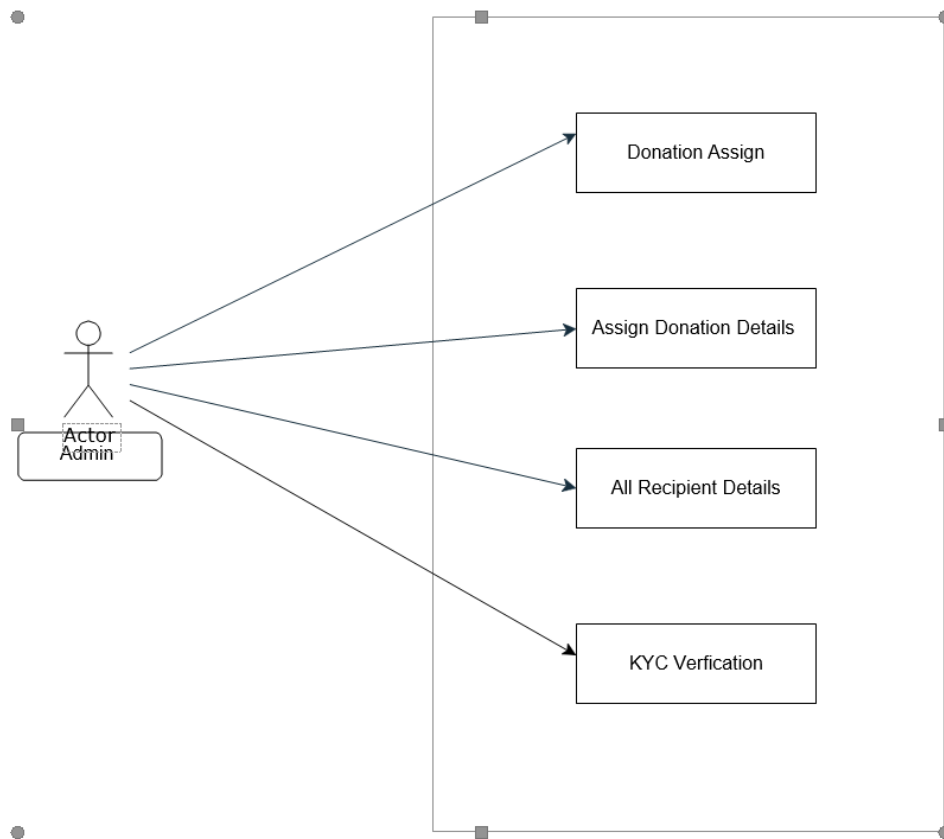


Fig . 1 Admin User Case Diagram

Fig.3.1.1 Admin Use Case

3.1.2 Recipient Use Case:

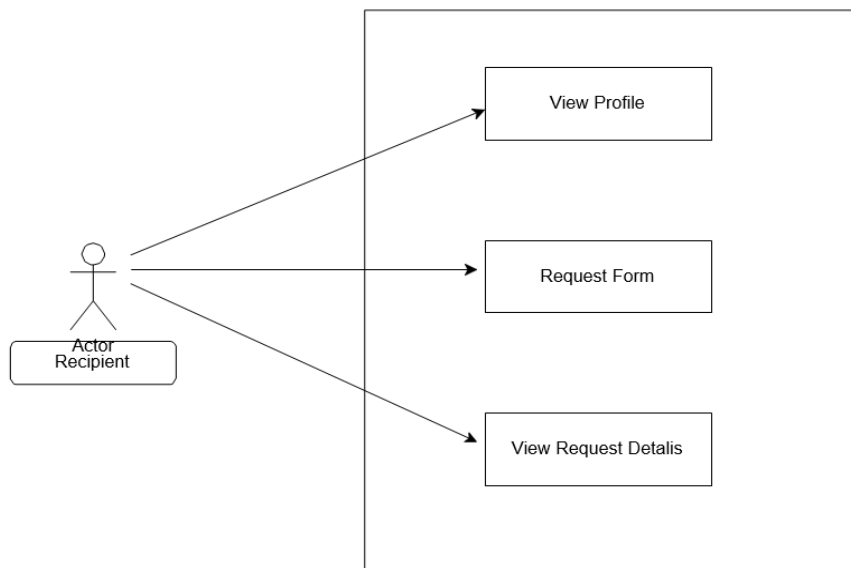


Fig.3.1.2 Recipient Use Case

3.1.3 Volunteer Use Case

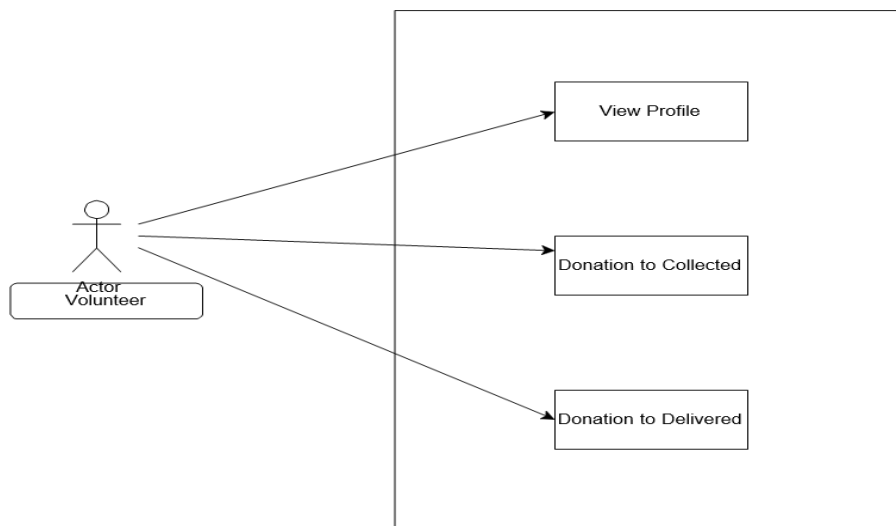


Fig.3.1.3 Volunteer Use Case

3.1.3 Donor Use Case

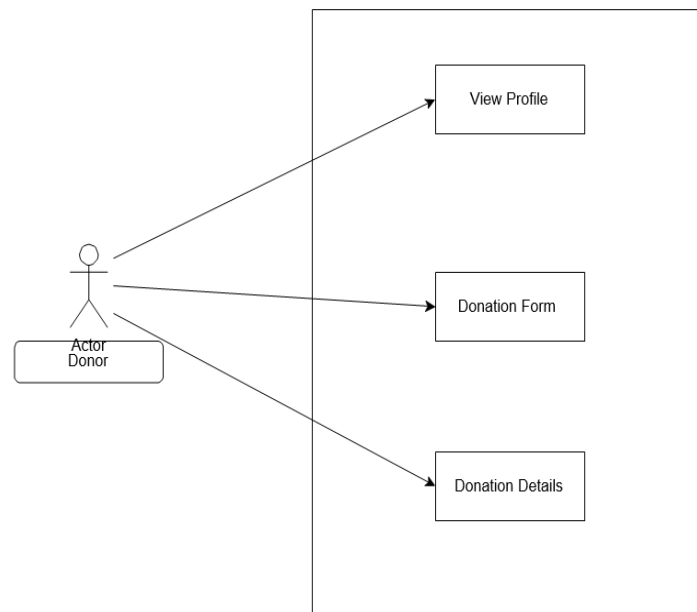


Fig.3.1.4 Donor Use Case

3.2 Activity Diagrams:

Activity diagram is another important behavioral diagram in UML diagram to describe dynamic aspects of the system. Activity diagram is essentially an advanced version of flow chart that modeling the flow from one activity to another activity.

3.2.1 Sign Up Activity

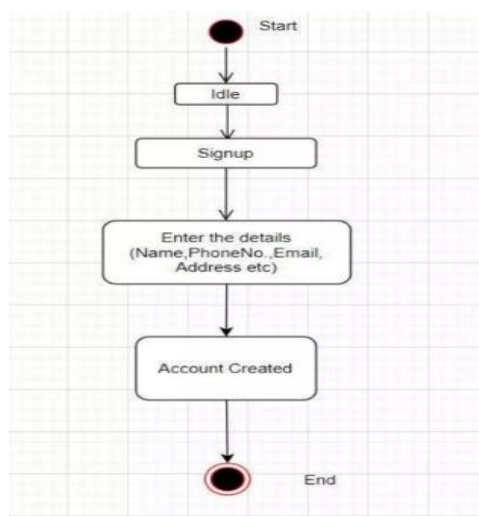


Fig.3.3.1 Sign Up Activity

3.2.2 Sign in Activity:

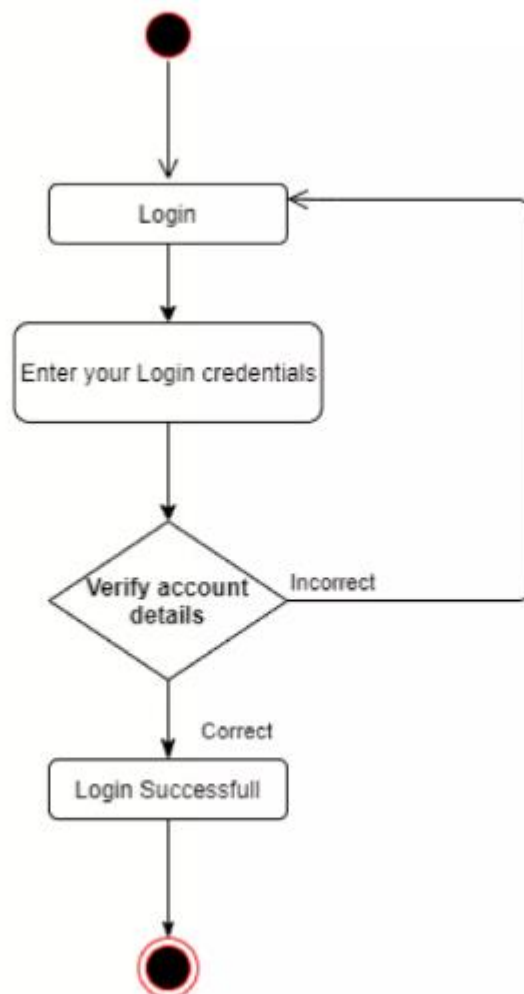


Fig.3.3.2 Sign in Activity

4. DESIGN

4.1 Data Modeling:

A Conceptual Data Model is an organized view of database concepts and their relationships. The purpose of creating a conceptual data model is to establish entities, their attributes, and relationships.

4.1.1 ER Diagram:

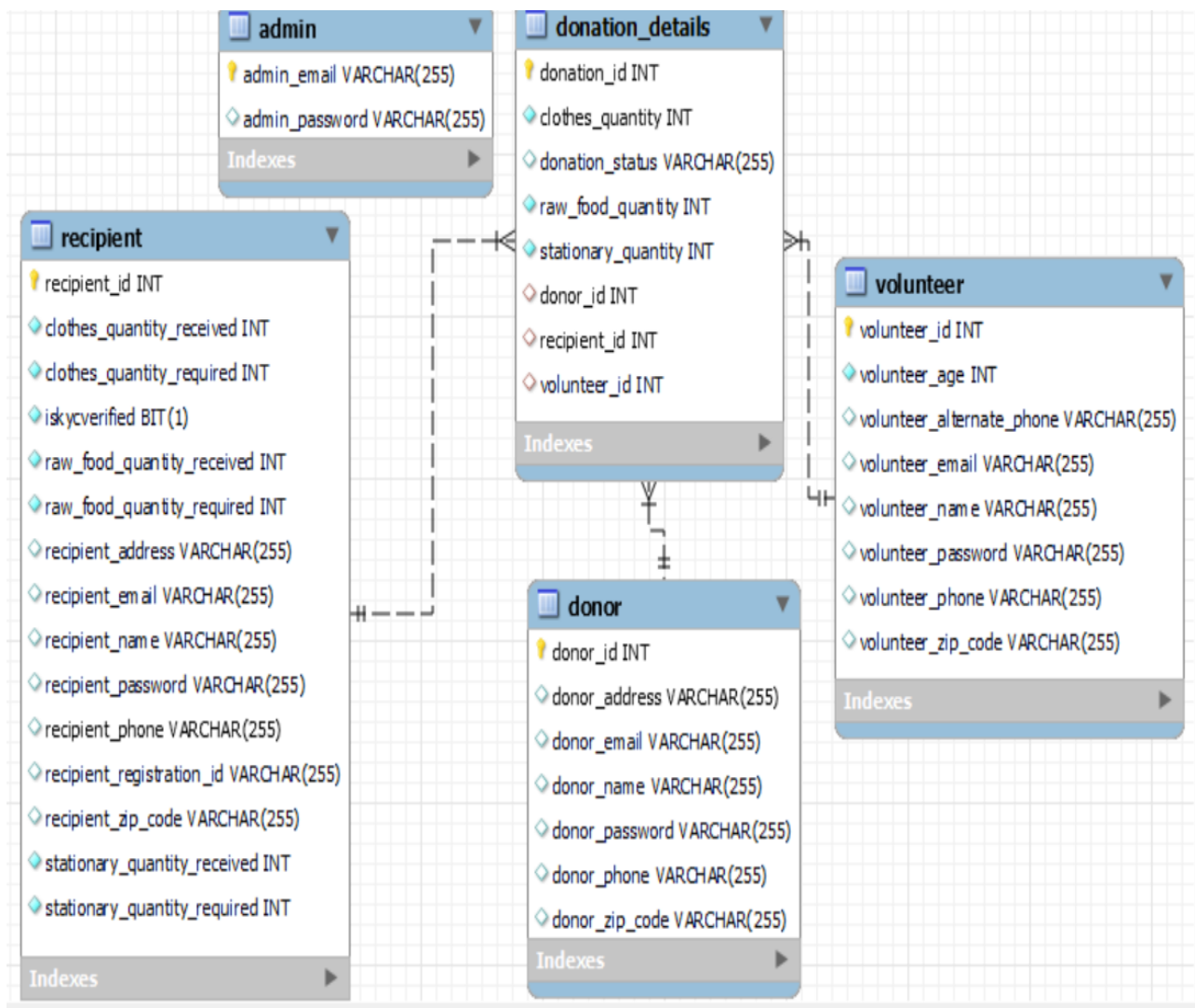
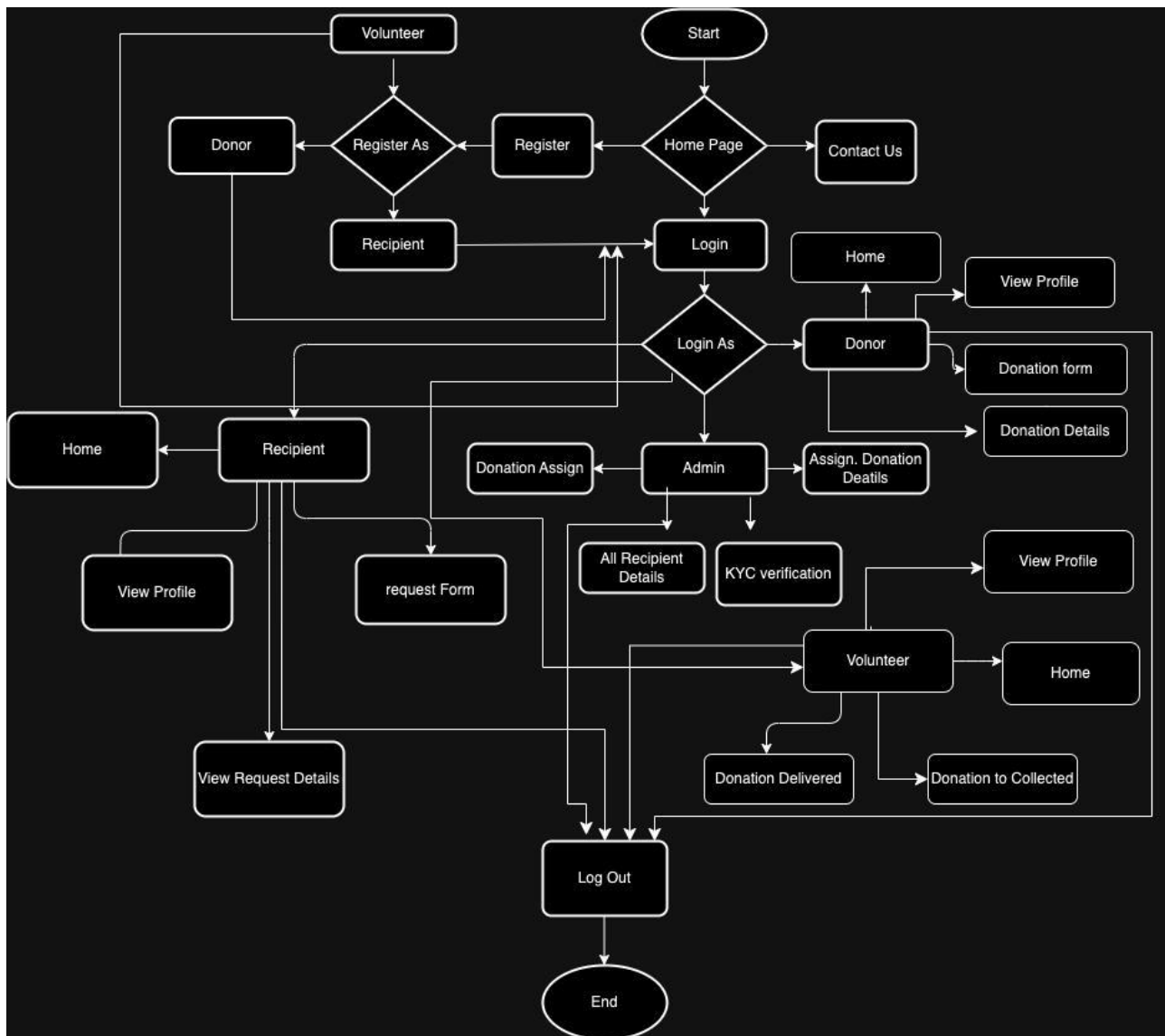


Fig.4.1.1 ER Diagram:

4.1.2 Flow Chart



4.1.4 Database:

Admin_email	Admin_password
admin@gmail.com	\$2a\$10\$Z1iKCO5hrNs0IHJfwt86oOGARVY03X8ksvICGIQWw2WJLB F0cHh3O

4.1.2.1 Table for Admin

```
mysql> select * from donor;
+-----+-----+-----+-----+-----+
| donor_id | donor_address          | donor_email          | donor_name | donor_password          |
| donor_phone | donor_zip_code |
+-----+-----+-----+-----+-----+
| 1 | Sarita vihar          | donor1@gmail.com     | donorone   | $2a$10$nia.98Q4CCAflbro6iB8Q.jz3LeC4Nb66BRb.fWUjrBo9PdA/XB4i | 9315028007 | 400076 |
| 2 | Ashok Park Main , New Delhi | nikhil128@gmail.com | Nikhil Kumar | $2a$10$o0zRUTJwsGhEHCN.BZfz0.ruX25dr.cY55dRZP18/ZgRRkDLam3oi | 8764589657 | 110026 |
| 3 | Sarita vihar          | vsingh89@gmail.com   | Varun Singh | $2a$10$HI6f/6qLzLz560W70vvo.u/3HJ7fqLPul7tLYTVYbIv9MDCSk2HSG | 9786543563 | 110076 |
| 4 | Gokulpuri, New Delhi | chhavivishoni435@gmail.com | Chhavi Vishoni | $2a$10$BUxoFLttqsjeF0rHRakkDuCMma5/8yIVCbQj7ZEwIcXFyTe3azwd2 | 6784537683 | 110094 |
| 5 | Shahdara, New Delhi | akkubhardwaj28@gmail.com | Akash Bhardwaj | $2a$10$mfeUqkhRqEs9ALu415MmNuQoawbAz4oLy0fRpIxQRz.yZJkptbjKy | 9845326798 | 110002 |
+-----+-----+-----+-----+-----+
5 rows in set (0.01 sec)
```

4.1.2.2 Table for Donor

```
mysql> select * from donation_details;
```

donation_id	clothes_quantity	donation_status	raw_food_quantity	stationary_quantity	donor_id	recipient_id	volunteer_id
1	20	isDelivered	10	19	1	1	1
6	18	toBeCollected	17	10	2	1	4
7	40	toBeCollected	36	40	4	NULL	NULL
8	30	toBeCollected	19	19	5	NULL	NULL

```
4 rows in set (0.00 sec)
```

4.1.2.2 Table for Donation Details

```
mysql> select * from volunteer;
```

volunteer_id	volunteer_age	volunteer_alternate_phone	volunteer_email	volunteer_name	volunteer_password
		volunteer_phone	volunteer_zip_code		
1	23		volunteer1@gmail.com	volunteerone	\$2a\$10\$w8pudK0.xn562ey3WtAhxeCmI/4uZmCIBe5E0gOuRdmTFxDyTNkea
2	30	7459342014	kumarvivek@gmail.com	Vivek Kumar	\$2a\$10\$MOYHA1X/c2mmrv5yoMI/GeaGCrDx0QvqQgTbUGuwAUQH1Be2wz6Jq
3	28	9436284737	yb4567@gmail.com	Yash Bbhavsar	\$2a\$10\$Pa5wleHc6yzBvKGARekPJJeDrn4oHUByH4fWTOQixd7uw/cMhvRqam
4	35	7392539467	kumaramar56@gmail.com	Amar Kumar	\$2a\$10\$sLb7naf06RpCENZCfni190NXALZJbXs4nMsb19mJiTJHuK9DfQJ.0
		8956342815			

```
4 rows in set (0.00 sec)
```

4.1.2.4 Table for Volunteer

```
mysql> select * from recipient;
```

recipient_id	clothes_quantity_received	clothes_quantity_required	iskvcverified	raw_food_quantity_received	raw_food_quantity_required	recipient_address	recipient_email	recipient_name	recipient_password	recipient_phone	recipient_registration_id	recipient_zip_code	stationary_quantity_received	stationary_quantity_required
1	0	9	0x01	0	10	delhi	nt1@gmail.com	Recipient	\$2a\$10\$2uicv0BmkAFbg/ssXX3U60urChz9eRoF0LbShwZhyZ.eVvW0LJ9.	9987654321	1000000000	400040	10	
2	10	10	0x00	0	0	Sarita vihar	nt2@gmail.com	Recipienttwo	\$2a\$10\$qRKn05bNzqIEgbw/2kHrqeaiQckXh0M4uAba0j3mQc9yqj06Pl16i	9876543234	1023456789	400076	10	
3	20	19	0x00	10	20	Munshi Ram Colony New Delhi	gmail.com	NGO raHI	\$2a\$10\$z3byxWUXT4HRYI1qEZNP/.DewX13GNe9jDHgy/7NhtuoBxAQeyjECO	9588894599	5678463527	110009	20	
4	0	0	0x00	0	0	Monahar Kunj, Gautam Nagar New Delhi	6@gmail.com	CHETNA NGO	\$2a\$10\$mitzP/nnpdMt3ow7wk2fc.amUcrYpXUF61Qz3JYUSuqxHOGACKv9G	8765437846	5643782547	110049	0	
5	18	20	0x00	23	20	Kokrola Dwaraka, New Delhi	hunt@gmail.com	Helping Hunt	\$2a\$10\$gaY3yWtHspzmAN7DYSEyk..8D/EmHzizg09Lf7W5FF3Ys4w4scSuG	8974956294	7865385678	110078	10	

5 rows in set (0.00 sec)

4.1.2.5 Table for Recipient

5. IMPLEMENTATION

5.1 Fronted End Code

5.1.1 MainHome Page:-

```
import { Container, Row, Card, Col } from "react-bootstrap";
import { NavigationBar } from "../NavigationBar";
import { CircularProgressbar } from "react-circular-progressbar";
import "react-circular-progressbar/dist/styles.css";
import { Link } from "react-router-dom";
import {
  fetchClothesPercentage,
  fetchRawFoodPercentage,
  fetchStationaryPercentage,
} from "../services/RecipientApiService";
import { useEffect, useState } from "react";
import Fade from 'react-reveal/Fade'; // Importing react-reveal animation

// Import the image
import myImage from "../components/images/homeBg.jpg";

export function MainHome() {
  const [rawFood, setRawFood] = useState(0);
  const [clothes, setClothes] = useState(0);
  const [stationary, setStationary] = useState(0);

  const fetchAllQuantities = async () => {
    let response = await fetchRawFoodPercentage();
    setRawFood(response.data);

    response = await fetchClothesPercentage();
    setClothes(response.data);

    response = await fetchStationaryPercentage();
    setStationary(response.data);
  };

  useEffect(() => {
    fetchAllQuantities();
  }, []);

  return (
    <>
    <NavigationBar />
    <Container className="bg-light text-center mt-0 p-3 rounded">
      <Fade top>
        <h1 className="display-4 text-primary mb-4 font-weight-bold">
          SPREAD THE SMILE-DONATION PLATFORM
        </h1>
      </Fade>
      <Fade top>
        <p className="lead">
          Our mission is to help you make a positive impact on the world by
          enabling you to easily donate to causes you care about.
        </p>
      </Fade>

      <Fade bottom>
        <img src={myImage} alt="A description of the image" />
      </Fade>
    </Container>
    </>
  );
}
```

```

</Fade>

<Container className="mt-4">
  <Row>
    <Col lg={4}>
      <Fade left>
        <Card className="border-0 shadow-lg rounded-lg">
          <div className="p-3 bg-primary text-white">
            <h3 className="text-center mb-0">Food Needs</h3>
          </div>
          <Card.Body className="text-center">
            <div
              style={{
                display: "flex",
                flexDirection: "column",
                alignItems: "center",
                marginTop: 20,
              }}
            >
              <CircularProgressbar value={rawFood} strokeWidth={8} />
              <div
                style={{
                  marginTop: 10,
                  fontSize: 24,
                  fontWeight: "bold",
                  color: "blue",
                }}
              >
                {rawFood}% of food needs met
              </div>
            </div>
            <Card.Text className="mt-4">
              Are you aware that 194 million individuals residing in India
              are malnourished and encounter difficulties in obtaining
              healthy and nutritious food? By donating, you can make a
              positive impact on their lives. Your contribution can assist
              us in eliminating hunger and ensuring that no one goes to bed
              without a meal.
            </Card.Text>
          </Card.Body>
        </Card>
      </Fade>
    </Col>

    <Col lg={4}>
      <Fade top>
        <Card className="border-0 shadow-lg rounded-lg">
          <div className="p-3 bg-primary text-white">
            <h3 className="text-center mb-0">Stationery Needs</h3>
          </div>
          <Card.Body className="text-center">
            <div
              style={{
                display: "flex",
                flexDirection: "column",
                alignItems: "center",
                marginTop: 20,
              }}
            >
              <CircularProgressbar value={stationary} strokeWidth={8} />
              <div

```

```

        style={{
            marginTop: 10,
            fontSize: 24,
            fontWeight: "bold",
            color: "blue",
        }}
    >
        {stationary}% of stationery needs met
    </div>
</div>
<Card.Text className="mt-4">
    A large number of children and adults do not have access to
    basic stationery items such as notebooks, pens, and pencils.
    Your donation can help in providing access to education and
    learning opportunities for those in need. Let's come
    together to create a brighter future for all.
</Card.Text>
</Card.Body>
</Card>
</Fade>
</Col>

<Col lg={4}>
    <Fade right>
        <Card className="border-0 shadow-lg rounded-lg">
            <div className="p-3 bg-primary text-white">
                <h3 className="text-center mb-0">Clothing Needs</h3>
            </div>
            <Card.Body className="text-center">
                <div
                    style={{
                        display: "flex",
                        flexDirection: "column",
                        alignItems: "center",
                        marginTop: 20,
                    }}
                >
                    <CircularProgressbar value={clothes} strokeWidth={8} />
                    <div
                        style={{
                            marginTop: 10,
                            fontSize: 24,
                            fontWeight: "bold",
                            color: "blue",
                        }}
                    >
                        {clothes}% of clothing needs met
                    </div>
                </div>
                <Card.Text className="mt-4">
                    Did you know that approximately 40% of India's population
                    lives below the poverty line? Many of these individuals lack
                    access to basic necessities like warm clothing. Your
                    generous donation to support our cause for clothing drive
                    can help provide essential clothing items to those in need.
                </Card.Text>
            </Card.Body>
        </Card>
    </Fade>
</Col>
</Row>

```

```

    <Fade bottom>
    <Container>
    <Link
      to="/login"
      className="btn btn-primary rounded-pill bg-gradient my-4"
    >
      Donate Now
    </Link>
  </Container>
</Fade>
</Container>
</Container>
</>
);
}

```

5.1.2 About US Page:-

```

import React from "react";
import { Container, Row, Col, Image, Card } from "react-bootstrap";
import { NavigationBar } from "./NavigationBar";
import Footer from "./Footer";
import { Fade, Slide } from "react-awesome-reveal"; // Import animations

const AboutUs = () => {
  return (
    <>
      <NavigationBar></NavigationBar>
      <Container className="bg-light text-center mt-2 p-2 rounded">
        <Fade cascade>
          <h1 className="display-4 text-primary mb-2 font-weight-bold">
            SPREAD THE SMILE-DONATION PLATFORM
          </h1>
          <p className="lead">
            Our mission is to help you make a positive impact on the world by
            enabling you to easily donate to causes you care about.
          </p>
        </Fade>
      </Container>
      <Container className="mt-5 bg-white rounded shadow-lg">
        <Fade>
          <h1 className="text-center mb-5">About Us</h1>
        </Fade>
        <Row className="align-items-center mb-5">
          <Col md={6}>
            <Slide direction="left">
              <Image
                src="https://c0.wallpaperflare.com/preview/287/860/54/team-ethnicity-group-hands.jpg"
                alt="About Us"
                fluid
              />
            </Slide>
          </Col>
          <Col md={6}>
            <Slide direction="right">
              <h2 className="mb-4">SPREAD THE SMILE-DONATION PLATFORM</h2>
              <p>
                In India, we often see NGOs and donors working in silos lacking a
                common standardized platform to handle end-to-end donation
                processes. Spread the Smile-Donation Platform aims to address this critical issue by
                bridging this gap between donors and recipients. In this system,
                donors will be assisted throughout the donation decision-making
                process. Verified recipients' demands will be fulfilled swiftly
              </p>
            </Slide>
          </Col>
        </Row>
      </Container>
    </>
  );
};

```


through volunteers. This platform promises to bring credibility and efficiency in the existing system.

```
</p>
</Slide>
</Col>
</Row>
<Row className="justify-content-center">
  <Col md={3}>
    <Fade>
      <Card className="p-3 mb-3 bg-secondary text-white rounded shadow">
        <Card.Body>
          <i className="fas fa-bullseye fa-3x mb-3"></i>
          <Card.Title>Vision</Card.Title>
          <Card.Text>
            The idea is to create self-sustained chapters across the world
            who will look after their local community. And in the process,
            inspire people around us to give back to those who need it
            most.
          </Card.Text>
        </Card.Body>
      </Card>
    </Fade>
  </Col>
  <Col md={3}>
    <Fade delay={200}>
      <Card className="p-3 mb-3 bg-success text-white rounded shadow">
        <Card.Body>
          <i className="fas fa-users fa-3x mb-3"></i>
          <Card.Title>Who We Are</Card.Title>
          <Card.Text>
            The Spread the Smile-Donation Platform is a volunteer-based, zero-funds
            organization that works to get surplus food, stationary,
            clothes from society and the community to serve them to the
            less fortunate people.
          </Card.Text>
        </Card.Body>
      </Card>
    </Fade>
  </Col>
  <Col md={3}>
    <Fade delay={400}>
      <Card className="p-3 mb-3 bg-info text-white rounded shadow">
        <Card.Body>
          <i className="fas fa-bullhorn fa-3x mb-3"></i>
          <Card.Title>Mission</Card.Title>
          <Card.Text>
            The idea is to create self-sustained chapters across the world
            who will look after their local community. And in the process,
            inspire people around us to give back to those who need it
            most.
          </Card.Text>
        </Card.Body>
      </Card>
    </Fade>
  </Col>
</Row>
</Container>
</>
);
};
```

```
export default AboutUs;
```

5.1.3 User Login Page:-

```
import React, { useState, useEffect } from "react";
import { Button, Col, Container, Form, Row } from "react-bootstrap";
import { Link, useNavigate } from "react-router-dom";
import { donorLogin } from "../services/DonorApiService";
import { NavigationBar } from "../NavigationBar";
import ReCAPTCHA from "react-google-recaptcha";
import { motion } from "framer-motion";

export function UserLoginForm() {
  // Define state for form inputs and errors
  const [donorDetails, setDonorDetails] = useState({
    donorEmail: "",
    donorPassword: "",
  });
  const [verified, setVerified] = useState(true);
  const [loading, setLoading] = useState(false);
  const [donorEmailError, setDonorEmailError] = useState("");
  const [donorPasswordError, setDonorPasswordError] = useState("");
  const [errors, setErrors] = useState("");
  const [isVisible, setIsVisible] = useState(false); // For animation
  const navigate = useNavigate();

  useEffect(() => {
    // Show the form with a fade-in effect
    setIsVisible(true);
  }, []);

  const handleChange = (e) => {
    setDonorDetails({ ...donorDetails, [e.target.name]: e.target.value });
  };

  function onChange(value) {
    console.log("Captcha value:", value);
    setVerified(false);
  }

  const handleSubmit = async (e) => {
    e.preventDefault();
    setErrors("");
    if (!validate()) {
      return;
    } else {
      setLoading(true);
      try {
        let obj = {
          donorEmail: donorDetails.donorEmail.toLowerCase(),
          donorPassword: donorDetails.donorPassword,
        };

        const response = await donorLogin(obj);
        console.log(response);
        if (response.status === 200) {
          if (response.data !== "") {
            if (response.data.adminEmail) {
              localStorage.setItem("admin", JSON.stringify(response.data));
              console.log(JSON.parse(localStorage.getItem("admin")));
              navigate("/adminDonationAssign");
            } else if (response.data.donorId) {
              localStorage.setItem("donor", JSON.stringify(response.data));
            }
          }
        }
      } catch (error) {
        console.log(error);
      }
    }
  };
}
```

```

        console.log(JSON.parse(localStorage.getItem("donor")));
        navigate("/donorHome");
    } else if (response.data.recipientId) {
        localStorage.setItem("recipient", JSON.stringify(response.data));
        console.log(JSON.parse(localStorage.getItem("recipient")));
        navigate("/recipientHome");
    } else if (response.data.volunteerId) {
        localStorage.setItem("volunteer", JSON.stringify(response.data));
        console.log(JSON.parse(localStorage.getItem("volunteer")));
        navigate("/volunteerHome");
    }
    } else {
        setErrors("Invalid Login Credentials");
    }
    }
    } catch (error) {
        setErrors("An error occurred. Please try again.");
    } finally {
        setLoading(false);
    }
    }
    };

const validate = () => {
    let isValid = true;
    if (donorDetails.donorEmail.trim() === "") {
        setDonorEmailError("Email is required");
        isValid = false;
    } else if (!/^[a-z]{1}[a-z0-9]{0,}@[a-z]{1,}\.[a-z]{1,}$/i.test(donorDetails.donorEmail)) {
        setDonorEmailError("Email is not valid");
        isValid = false;
    } else {
        setDonorEmailError("");
    }

    if (donorDetails.donorPassword.trim() === "") {
        setDonorPasswordError("Password is required");
        isValid = false;
    } else if (donorDetails.donorPassword.length < 8) {
        setDonorPasswordError("Password must be at least 8 characters long");
        isValid = false;
    } else {
        setDonorPasswordError("");
    }
    return isValid;
};

return (
    <>
    <NavigationBar />
    <Container className="bg-light text-center mt-0 p-3 rounded">
        <h1 className="display-4 text-primary mb-4 font-weight-bold">
            SPREAD THE SMILE-DONATION PLATFORM
        </h1>
        <p className="lead">
            Our mission is to help you make a positive impact on the world by
            enabling you to easily donate to causes you care about.
        </p>
    </Container>
    <Container
        fluid

```

```

        className="d-flex justify-content-center align-items-center pb-3"
        style={{ backgroundColor: "#f7f7f7" }}
    >
    <motion.div
        initial={{ opacity: 0, scale: 0.9 }}
        animate={{ opacity: isVisible ? 1 : 0, scale: isVisible ? 1 : 0.9 }}
        transition={{ duration: 0.5 }}
    >
    <Form
        onSubmit={handleSubmit}
        className="p-5 rounded"
        style={{
            backgroundColor: "rgba(255, 255, 255, 0.8)",
            boxShadow: "0px 0px 10px rgba(0, 0, 0, 0.1)",
        }}
    >
    <h1 className="mb-4 text-center mt-2 p-3">Login</h1>
    <Row>
    <Col lg={12}>
    <Form.Group className="mb-3">
    <Form.Label>User Email</Form.Label>
    <Form.Control
        type="text"
        placeholder="Enter email"
        name="donorEmail"
        value={donorDetails.donorEmail}
        onChange={handleChange}
        className={
            donorEmailError
            ? "form-control-danger"
            : "form-control-primary"
        }
    >
    </Form.Control>
    {donorEmailError && (
    <Form.Text className="text-danger">
    {donorEmailError}
    </Form.Text>
    )}
    </Form.Group>
    </Col>
    <Col lg={12}>
    <Form.Group className="mb-3">
    <Form.Label>Password</Form.Label>
    <Form.Control
        type="password"
        placeholder="Enter password"
        name="donorPassword"
        value={donorDetails.donorPassword}
        onChange={handleChange}
        className={
            donorPasswordError
            ? "form-control-danger"
            : "form-control-primary"
        }
    >
    </Form.Control>
    {donorPasswordError && (
    <Form.Text className="text-danger">
    {donorPasswordError}
    </Form.Text>
    )}
    </Form.Group>

```

```

    </Col>
  </Row>

  <Col className="text-center">
    {errors && <Form.Text className="text-danger">{errors}</Form.Text>}
  </Col>
  <div className="d-flex justify-content-center mb-3">
    <ReCAPTCHA
      sitekey="6LdTH_UkAAAAAHKD0qtLrAlrb1CFvsYb1C4XpZfD"
      onChange={onChange}
    />
  </div>
  <div className="text-center">
    <Button
      type="submit"
      variant="primary"
      disabled={verified}
      className="mb-3"
      style={{ backgroundColor: "#007bff", borderColor: "#007bff" }}
    >
      {loading ? "Logging in..." : "Login"}
    </Button>
  </div>
  <div className="text-center">
    <Link to={"/forgotPassword"}>Forget password?</Link>
  </div>
</Form>
</motion.div>
</Container>
</>
);
}

```

5.1.4 Contact US Page:-

```

import React from "react";
import { Container, Row, Col, Card } from "react-bootstrap";
import { NavigationBar } from "../NavigationBar";
import Footer from "../Footer";
import { Fade, Slide } from "react-awesome-reveal"; // Import animations

export function Contact() {
  return (
    <>
      <NavigationBar />
      <Container className="bg-light text-center mt-2 p-4 rounded shadow-sm">
        <Fade>
          <h1 className="display-4 text-primary mb-4 font-weight-bold">
            SPREAD THE SMILE-DONATION PLATFORM
          </h1>
          <p className="lead">
            Our mission is to help you make a positive impact on the world by
            enabling you to easily donate to causes you care about.
          </p>
        </Fade>
      </Container>
      <Container
        className="bg-light mt-4 p-5 rounded shadow-sm"
        style={{ marginBottom: "100px" }}
      >
        <Row className="justify-content-center">
          <Col md={8}>
            <Slide direction="up">

```

```

<h2 className="text-center font-weight-bold mb-4">
  Contact Us
</h2>
</Slide>
<Slide direction="up" delay={100}>
  <p className="text-center lead mb-4">
    In India, we often see NGOs and donors working in silos lacking
    a common standardized platform to handle end-to-end donation
    processes.
    <span className="text-primary"> Spread the Smile-Donation Platform</span> aims to
    address this critical issue by bridging this gap between donors
    and recipients. This platform promises to bring credibility and
    efficiency to the existing system.
  </p>
  <p className="text-center lead mb-5">
    If you have any questions, feedback, or want to get involved with
    <span className="text-primary"> Spread the Smile-Donation Platform</span>, please
    don't hesitate to reach out to us.
  </p>
</Slide>
<Slide direction="up" delay={200}>
  <Row className="justify-content-center">
    <Col md={6}>
      <Card className="shadow">
        <Card.Body>
          <h2 className="text-primary font-weight-bold mb-4">
            Contact Information:
          </h2>
          <p className="mb-0">
            <i className="fas fa-map-marker-alt mr-2"></i>
            Address: XYZ street, ABC city, India
          </p>
          <p className="mb-0">
            <i className="fas fa-envelope mr-2"></i>
            Email: info@charityxchange.com
          </p>
          <p className="mb-0">
            <i className="fas fa-phone-alt mr-2"></i>
            Phone: +91-1234567890
          </p>
        </Card.Body>
      </Card>
    </Col>
    <Col md={6}>
      <Card className="shadow">
        <Card.Body>
          <p className="text-primary font-weight-bold mb-4">
            For inquiries or partnership opportunities, please
            contact us:
          <br />
          Email: { " " }
          <a
            href="mailto:info@charityxchange.com"
            style={{ color: "black" }}
          >
            info@charityxchange.com
          </a>
          <p>Phone: +91-1234567890</p>
        </Card.Body>
      </Card>
    </Col>
  </Row>
</Slide>

```

```

        </Col>
    </Row>
</Slide>
</Col>
</Row>
</Container>
<Footer />
</>
);
}

```

5.2 BackEnd code:

5.2.1 AdminRestController:

```

6  package com.DonationBackend.cntr;
7
8
9  import org.mindrot.jbcrypt.BCrypt;
10 import org.springframework.beans.factory.annotation.Autowired;
11 import org.springframework.web.bind.annotation.CrossOrigin;
12 import org.springframework.web.bind.annotation.PostMapping;
13 import org.springframework.web.bind.annotation.RequestBody;
14 import org.springframework.web.bind.annotation.RestController;
15
16 import com.DonationBackend.model.Admin;
17 import com.DonationBackend.model.Donor;
18 import com.DonationBackend.model.Recipient;
19 import com.DonationBackend.model.Volunteer;
20 import com.DonationBackend.service.AdminService;
21 import com.DonationBackend.service.DonorService;
22 import com.DonationBackend.service.RecipientService;
23 import com.DonationBackend.service.VolunteerService;
24
25 @CrossOrigin
26 @RestController
27 public class AdminRestController {
28
29     @Autowired
30     private DonorService donorService;
31
32     @Autowired
33     private AdminService adminService;
34
35     @Autowired
36     private RecipientService recipientService;
37
38     @Autowired
39     private VolunteerService volunteerService;
40
41
42     @PostMapping("/adminRegister")
43     public Admin registerAdmin(@RequestBody Admin admin) {
44         // Hash the admin password before saving
45         String hashedPassword = BCrypt.hashpw(admin.getAdminPassword(),
BCrypt.gensalt());
46         admin.setAdminPassword(hashedPassword);
47
48         // Set the admin email
49         String adminEmail = admin.getAdminEmail();
50         admin.setAdminEmail(adminEmail);

```

```

51
52     return adminService.registerAdmin(admin);
53 }
54
55 //to check password with encrypted password
56 static private boolean checkPass(String plainPassword, String hashedPassword) {
57     if (BCrypt.checkpw(plainPassword, hashedPassword)) {
58         System.out.println("The password matches.");
59         return true;
60     }
61     else {
62         System.out.println("The password does not match.");
63         return false;
64     }
65 }
66
67 //admin table check for logged in email and password
68 public Admin adminCheck(String email, String plainPassword ) {
69
70     Admin admin = adminService.getById(email);
71     if (admin != null) {
72         String hashedPassword=admin.getAdminPassword();
73         if(checkPass(plainPassword, hashedPassword)) {
74             return admin;
75         }
76     }
77     return null;
78 }
79
80 //donor table check for logged in email and password
81 public Donor donorCheck(String email, String plainPassword) {
82
83     Donor donor=donorService.getByEmail(email);
84     if (donor != null) {
85         String hashedPassword=donor.getDonorPassword();
86         System.out.println(plainPassword);
87         System.out.println(hashedPassword);
88         if(checkPass(plainPassword, hashedPassword)) {
89             System.out.println("2");
90             System.out.println(donor);
91             return donor;
92         }
93     }
94     return null;
95 }
96
97 //recipient table check for logged in email and password
98 public Recipient recipientCheck(String email, String plainPassword) {
99
100     Recipient recipient = recipientService.getByEmail(email);
101     if (recipient != null) {
102         String hashedPassword=recipient.getRecipientPassword();
103         if(checkPass(plainPassword, hashedPassword)) {
104             System.out.println("Recipient");
105             System.out.println(recipient);
106             return recipient;
107         }
108     }
109     return null;
110 }
111

```



```

112 //volunteer table check for logged in email and password
113 public Volunteer volunteerCheck(String email, String plainPassword) {
114
115     Volunteer volunteer = volunteerService.getByEmail(email);
116     if (volunteer != null) {
117         String hashedPassword=volunteer.getVolunteerPassword();
118         System.out.println(plainPassword);
119         System.out.println(hashedPassword);
120         if(checkPass(plainPassword, hashedPassword)) {
121             return volunteer;
122         }
123     }
124     return null;
125 }
126
127 @PostMapping(value= {"/donorLogin"})
128 public Object userLogin(@RequestBody Donor donorObj ) {
129
130     String email = donorObj.getDonorEmail();
131     String plainPassword=donorObj.getDonorPassword();
132
133     if(adminCheck(email, plainPassword)!=null) {
134         return adminCheck(email, plainPassword);
135     }
136     else if(donorCheck(email, plainPassword)!=null) {
137         return donorCheck(email, plainPassword);
138     }
139     else if(recipientCheck(email, plainPassword)!=null) {
140         return recipientCheck(email, plainPassword);
141     }
142     else if(volunteerCheck(email, plainPassword)!=null) {
143         return volunteerCheck(email, plainPassword);
144     }
145     else
146         return null;
147 }
148
149 //all the above code is for login check in different roles across all tables
150
151 @PostMapping(value= {"/checkEmail"})
152 public boolean checkEmail(@RequestBody Donor donorObj) {
153     String email=donorObj.getDonorEmail();
154     System.out.println("emailCheck method");
155     System.out.println(email);
156     if(adminService.getId(email)!=null) {
157         return true;
158     }
159     else if(donorService.getByEmail(email)!=null) {
160         return true;
161     }
162     else if(recipientService.getByEmail(email)!=null) {
163         return true;
164     }
165     else if(volunteerService.getByEmail(email)!=null) {
166         return true;
167     }
168     else {
169         return false;
170     }
171 }
172

```

```

173 @PostMapping(value= {"/userPasswordUpdate"})
174 public boolean userPasswordUpdate(@RequestBody Donor donorObj) {
175     String email=donorObj.getDonorEmail();
176
177     if(donorService.getByEmail(email)!=null) {
178         Donor donorobj =donorService.getByEmail(email);
179         donorobj.setDonorPassword(donorObj.getDonorPassword());
180         donorService.add(donorobj);
181         return true;
182     }
183     else if(recipientService.getByEmail(email)!=null) {
184         Recipient recipientobj=recipientService.getByEmail(email);
185         recipientobj.setRecipientPassword(donorObj.getDonorPassword());
186         recipientService.add(recipientobj);
187         return true;
188     }
189     else if(volunteerService.getByEmail(email)!=null) {
190         Volunteer volunteerobj=volunteerService.getByEmail(email);
191         volunteerobj.setVolunteerPassword(donorObj.getDonorPassword());
192         volunteerService.add(volunteerobj);
193         return true;
194     }
195     else {
196         return false;
197     }
198
199 }
200
201 }

```

5.2.2 DonationDetailsDao:

```
package com.DonationBackend.dao;
```

```
import java.util.List;
```

```
import org.springframework.data.jpa.repository.JpaRepository;
import org.springframework.data.jpa.repository.Query;
import org.springframework.data.repository.query.Param;
import org.springframework.stereotype.Repository;
```

```
import com.DonationBackend.model.DonationDetails;
import com.DonationBackend.model.Donor;
import com.DonationBackend.model.Volunteer;
```

```
@Repository
```

```
public interface DonationDetailsDao extends JpaRepository<DonationDetails, Integer>{
```

```
    @Query(value = "select p from DonationDetails p where p.donor = :donor")
    public List<DonationDetails> selectByDonor(@Param(value = "donor") Donor donor);
```

```
    @Query(value = "select p from DonationDetails p where p.volunteer = :volunteer")
    public List<DonationDetails> selectByVolunteer(@Param(value = "volunteer") Volunteer
        volunteer);
```

```

// //not needed thiis method
// @Query(value = "select p from DonationDetails p where p.recipient = :recipient")
// public List<DonationDetails> selectByRecipient(@Param(value = "recipient") Recipient
//     recipient);

@Query(value = "select p from DonationDetails p where p.volunteer is NULL")
public List<DonationDetails> selectNotAssignedDonations();

@Query(value = "select p from DonationDetails p where p.volunteer is NOT NULL")
public List<DonationDetails> selectAssignedDonations();

@Query(value = "select p from DonationDetails p where p.volunteer = :volunteer and
    p.donationStatus = 'toBeCollected'")
public List<DonationDetails> selectByDonationToBeCollected(@Param(value =
    "volunteer") Volunteer volunteer);

@Query(value = "select p from DonationDetails p where p.volunteer = :volunteer and
    p.donationStatus = 'isCollected'")
public List<DonationDetails> selectByDonationIsCollected(@Param(value = "volunteer")
    Volunteer volunteer);

}

```

5.2.3 DonationDetails:

```

package com.DonationBackend.model;

import javax.persistence.Entity;
import javax.persistence.GeneratedValue;
import javax.persistence.GenerationType;
import javax.persistence.Id;
import javax.persistence.JoinColumn;
import javax.persistence.ManyToOne;

import org.hibernate.annotations.ColumnDefault;

@Entity
public class DonationDetails {

    @Id
    @GeneratedValue(strategy = GenerationType.IDENTITY)
    private int donationId;
    @ManyToOne
    @JoinColumn(name = "donorId")
    private Donor donor;
    private int rawFoodQuantity;
    private int clothesQuantity;
    private int stationaryQuantity;

```

```

@ManyToOne
@JoinColumn(name = "volunteerId")
private Volunteer volunteer;

@ManyToOne
@JoinColumn(name = "recipientId")
private Recipient recipient;

@ColumnDefault("toBeCollected")
private String donationStatus;

public int getDonationId() {
return donationId;
}
public void setDonationId(int donationId) {
this.donationId = donationId;
}
public Donor getDonor() {
return donor;
}
public void setDonor(Donor donor) {
this.donor = donor;
}
public int getRawFoodQuantity() {
return rawFoodQuantity;
}
public void setRawFoodQuantity(int rawFoodQuantity) {
this.rawFoodQuantity = rawFoodQuantity;
}
public int getClothesQuantity() {
return clothesQuantity;
}
public void setClothesQuantity(int clothesQuantity) {
this.clothesQuantity = clothesQuantity;
}
public int getStationaryQuantity() {
return stationaryQuantity;
}
public void setStationaryQuantity(int stationaryQuantity) {
this.stationaryQuantity = stationaryQuantity;
}
public Volunteer getVolunteer() {
return volunteer;
}
public void setVolunteer(Volunteer volunteer) {
this.volunteer = volunteer;
}
public Recipient getRecipient() {
return recipient;
}
}

```

```

public void setRecipient(Recipient recipient) {
this.recipient = recipient;
}
public String getDonationStatus() {
return donationStatus;
}
public void setDonationStatus(String donationStatus) {
this.donationStatus = donationStatus;
}

}

```

5.2.4 AdminServiceImpl:

```

package com.DonationBackend.service;

import java.util.Optional;

import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.stereotype.Service;

import com.DonationBackend.dao.AdminDao;
import com.DonationBackend.dao.DonorDao;
import com.DonationBackend.dao.RecipientDao;
import com.DonationBackend.dao.VolunteerDao;
import com.DonationBackend.model.Admin;
import com.DonationBackend.model.Donor;

@Service
public class AdminServiceImpl implements AdminService{

    @Autowired
    private AdminDao adminDao;

    @Autowired
    private DonorDao donorDao;

    @Autowired
    private RecipientDao recipientDao ;

    @Autowired
    private VolunteerDao volunteerDao;

    @Override
    public Admin getById(String adminEmail) {

        Optional<Admin> opt = adminDao.findById(adminEmail);
        if(opt.isPresent())
            return opt.get();
        else
            return null;
    }
}

```

```

    }

    @Override
    public Admin registerAdmin(Admin admin) {
        return adminDao.save(admin);
    }

    @Override
    public void add(Admin admin) {
        adminDao.save(admin);
    }

    @Override
    public Admin getByEmail(String email) {
        Optional<Admin> optionalAdmin = adminDao.findById(email);
        return optionalAdmin.orElse(null);
    }

    // to check if registering email is already present
    @Override
    public boolean doesEmailExist(String email) {

        if(adminDao.findById(email).isPresent()) {
            return true;
        }
        else if(donorDao.selectByDonorEmail(email)!=null) {
            return true;
        }
        else if(recipientDao.selectByEmail(email)!=null) {
            return true;
        }
        else if(volunteerDao.selectByVolunteerEmail(email)!=null) {
            return true;
        }
        return false;
    }
}

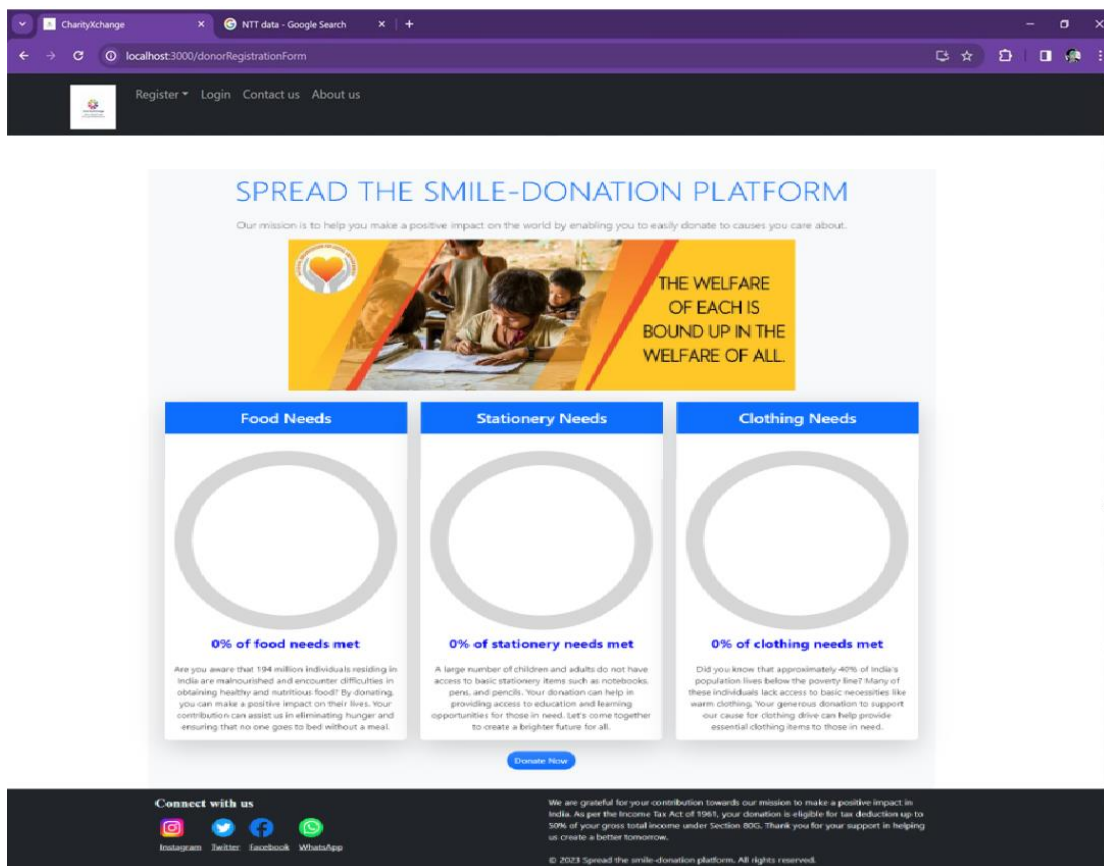
```

201.1 Result Analysis:-

We have 4 module in our project
Admin , Donor , Recipient , Volunteer

- Donor can donate the clothes ,food like that
- Recipient that is a NGO or old age home who can receive the donation
- Volunteer is simply the medium of delivery boy who can take donation from donor and give to the recipient.

Home Page



Donor Registration Form

CharityXchange × NTT data - Google Search × | +

localhost:3000/donorRegistrationForm

Register Login Contact us About us

Donor Registration Form

Donor Name:	Donor Email
<input type="text" value="Enter name"/>	<input type="text" value="Enter email"/>
Name is required	Email ID is required
Donor Phone Number:	Donor Address
<input type="text" value="Enter phone"/>	<input type="text" value="Enter address"/>
Phone number is required	Address is required
Donor ZIP Code	Donor Password
<input type="text" value="Enter ZIP code"/>	<input type="text" value="Enter Password"/>
ZIP code is required	Password is required
Confirm Donor Password	
<input type="text" value="Confirm Password"/>	
Confirm Password is required	

Register Donor

Connect with us


We are grateful for your contribution towards our mission to make a positive impact in India. As per the Income Tax Act of 1961, your donation is eligible for tax deduction up to

Recipient Registration form

CharityXchange

NTT data - Google Search

localhost:3000/recipientRegistrationForm





[Register](#) [Login](#) [Contact us](#) [About us](#)

Recipient Registration Form

Recipient Name: <input type="text" value="Enter name"/>	Recipient Email <input type="text" value="Enter email"/>
Recipient Phone Number <input type="text" value="Enter phone"/>	Recipient Address <input type="text" value="Enter address"/>
Recipient RegistrationId <input type="text" value="Enter RegistrationId"/>	Recipient ZIP Code <input type="text" value="Enter ZIP code"/>
Recipient Password <input type="password" value="Enter Password"/>	Confirm Recipient Password <input type="password" value="Confirm Password"/>

Register recipient

Connect with us

[Instagram](#) [Twitter](#) [Facebook](#) [WhatsApp](#)

We are grateful for your contribution towards our mission to make a positive impact in India. As per the Income Tax Act of 1961, your donation is eligible for tax deduction up to 50% of your gross total income under Section 80G. Thank you for your support in helping us create a better tomorrow.


© 2023 CharityXchange. All rights reserved.

Volunteer Registration Form

CharityXchange

NTT data - Google Search

localhost:3000/volunteerRegistrationForm





[Register](#) [Login](#) [Contact us](#) [About us](#)

Volunteer Registration Form

Volunteer Name: <input type="text" value="Enter name"/>	Volunteer Email <input type="text" value="Enter email"/>
Volunteer Phone Number: <input type="text" value="Enter phone"/>	Volunteer Alternate Phone Number: (optional) <input type="text" value="Enter alternate phone"/>
Volunteer ZIP Code <input type="text" value="Enter ZIP code"/>	Volunteer Age <input type="text" value="Enter age"/>
Volunteer Password <input type="password" value="Enter Password"/>	Confirm Volunteer Password <input type="password" value="Confirm Password"/>

Register Volunteer

Connect with us

[Instagram](#) [Twitter](#) [Facebook](#) [WhatsApp](#)

We are grateful for your contribution towards our mission to make a positive impact in India. As per the Income Tax Act of 1961, your donation is eligible for tax deduction up to 50% of your gross total income under Section 80G. Thank you for your support in helping us create a better tomorrow.

© 2023 CharityXchange. All rights reserved.

Admin Login

CharityXchange ChatGPT

localhost:3000/login

Register Login Contact us About us

SPREAD THE SMILE-DONATION PLATFORM

Our mission is to help you make a positive impact on the world by enabling you to easily donate to causes you care about.

Login

User Email

Password

☐ I'm not a robot

reCAPTCHA

Login

[Forgot password?](#)

Admin dashboard he can assign volunteer from here to collect donation.

CharityXchange ChatGPT

localhost:3000/adminDonationAssign

Donation Assign Assigned Donation Details All Recipient Details Recipient KYC verification Logout

Table for Assigning Volunteers to Collect Donations

Serial ID	Raw Food Quantity	Clothes Quantity	Stationary Quantity	Choose Volunteer	Choose Recipient	Save Changes
-----------	-------------------	------------------	---------------------	------------------	------------------	--------------

Connect with us



[Instagram](#)



[Twitter](#)



[Facebook](#)



[WhatsApp](#)

We are grateful for your contribution towards our mission to make a positive impact in India. As per the Income Tax Act of 1961, your donation is eligible for tax deduction up to 50% of your gross total income under Section 80G. Thank you for your support in helping us create a better tomorrow.


© 2023 CharityXchange. All rights reserved.

List of volunteer he can see whom to assigned and the status can see.

CharityXchange

ChatGPT

localhost:3000/adminDonationAlreadyAssign





[Donation Assign](#) [Assigned Donation Details](#) [All Recipient Details](#) [Recipient KYC verification](#) [Logout](#)


List of Volunteers Assigned to Collect Donations


Serial ID	Donor Assigned	Volunteer Assigned	Recipient Assigned	Donation Status
1	donarone	volunteerone	recipientone	Donation is delivered
2	donarone	volunteerone	recipientone	Donation is delivered
3	donortwo	volunteerone	recipienttwo	Donation is delivered
4	donortwo	volunteerone	recipienttwo	Donation is delivered
5	donorfive	volunteerone	mohit	Donation is delivered
6	donorfive	volunteerone	recipientfive	Donation is delivered
7	donorfive	volunteerfive	recipientfive	Donation is delivered

Connect with us

[Instagram](#)

[Twitter](#)

[Facebook](#)

[WhatsApp](#)

We are grateful for your contribution towards our mission to make a positive impact in India. As per the Income Tax Act of 1961, your donation is eligible for tax deduction up to 50% of your gross total income under Section 80G. Thank you for your support in helping us create a better tomorrow.


© 2023 CharityXchange. All rights reserved.

From here he can see the recipient details and the quantity of donation

CharityXchange

ChatGPT

localhost:3000/adminRecipientsList





[Donation Assign](#) [Assigned Donation Details](#) [All Recipient Details](#) [Recipient KYC verification](#) [Logout](#)


All Recipient Details


Basic Info			Quantity Required			Quantity Received		
Serial ID	Recipient Name	Recipient KYC Status	Raw Food	Clothes	Stationary	Raw Food	Clothes	Stationary
1	recipientone	Verified	150	100	100	250	150	150
2	recipienttwo	Verified	150	150	150	200	200	200
3	recipientfive	Verified	200	200	200	300	300	300
4	mohit	Verified	70	70	70	50	50	50

Connect with us

[Instagram](#)

[Twitter](#)

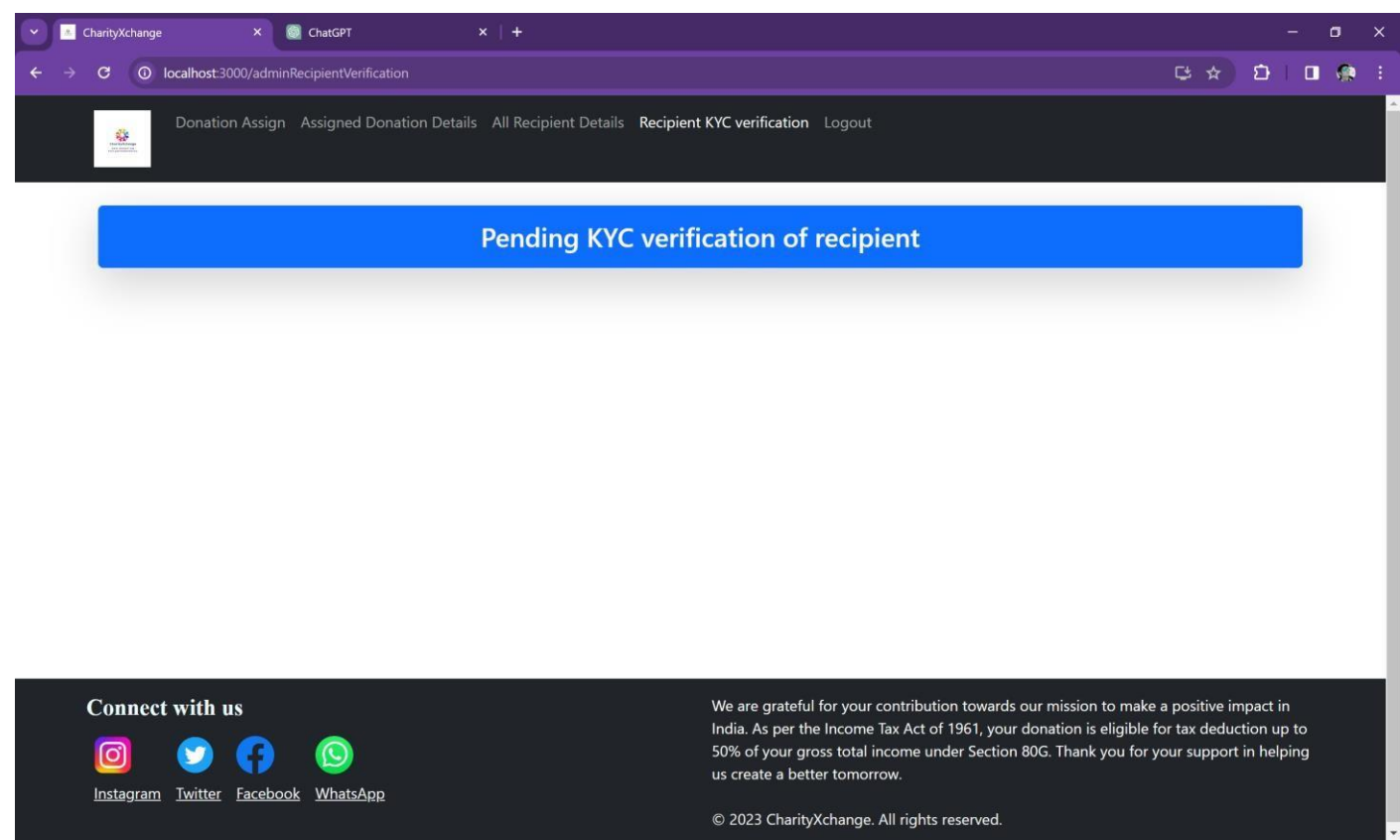
[Facebook](#)

[WhatsApp](#)

We are grateful for your contribution towards our mission to make a positive impact in India. As per the Income Tax Act of 1961, your donation is eligible for tax deduction up to 50% of your gross total income under Section 80G. Thank you for your support in helping us create a better tomorrow.


© 2023 CharityXchange. All rights reserved.

When first time recipient register he need of kyc that is verify by admin.




Donor Dashboard

CharityXchange
ChatGPT
localhost:3000/donorHome


[Home](#)
[View Profile](#)
[Donation form](#)
[Donation Details](#)
[Logout](#)

Welcome to the Donor Interface!







Donate to Support Your Favorite Causes

The Donor Interface makes it easy to support the causes that matter most to you. With our platform, you can donate to organizations focused on providing stationary, clothing, and raw food to those in need. And with our donation tracking system, you can keep track of all your donations in one place.

When you sign up for a Donor account, you'll have access to your own profile page where you can view your donation history. Our platform is designed to make giving easy, so you can focus on making a difference in the world.

Connect with us


[Instagram](#)
[Twitter](#)
[Facebook](#)
[WhatsApp](#)

We are grateful for your contribution towards our mission to make a positive impact in India. As per the Income Tax Act of 1961, your donation is eligible for tax deduction up to 50% of your gross total income under Section 80G. Thank you for your support in helping us create a better tomorrow.


© 2023 CharityXchange. All rights reserved.

Donor profile

CharityXchange
ChatGPT
localhost:3000/donorProfile


[Home](#)
[View Profile](#)
[Donation form](#)
[Donation Details](#)
[Logout](#)





Donor Profile



Name	donorfive
Phone Number	8007592194
Email	donorfive@gmail.com
Address	pune
ZIP code	431005

Edit Profile

Connect with us

[Instagram](#)
[Twitter](#)
[Facebook](#)
[WhatsApp](#)


We are grateful for your contribution towards our mission to make a positive impact in India. As per the Income Tax Act of 1961, your donation is eligible for tax deduction up to 50% of your gross total income under Section 80G. Thank you for your support in helping us create a better tomorrow.

Donation form

CharityXchange

ChatGPT

localhost:3000/donationForm



week's worth of food, offer a homeless person a safe shelter, and also provide stationary, clothing, and other essentials to those in need. Your generosity can make a significant impact on people's lives and provide them with the basic necessities they need to thrive. Imagine the joy on a child's face when they receive new clothes, or the gratitude a struggling family feels when they receive your help. Your donation can be the catalyst for change and create a ripple effect of positivity in your community. Join us in making a difference and donate today!

Donate Now

Enter raw food quantity (in kg.)

0

Enter stationary quantity (in nos.)

0

Enter clothing quantity (in nos.)

0


Submit

New donation will be added but not assigned yet

CharityXchange

ChatGPT

localhost:3000/donordonationdetails



Home

View Profile

Donation form


Donation Details

Logout


My Donations

Serial ID	Raw Food Quantity	Clothes Quantity	Stationary Quantity	Volunteer Assigned	Volunteer Contact Details	Volunteer Alternate Contact Details
1	100	100	100	volunteerfive	8007592194	9284926333
2	100	100	100	volunteerone	8007592194	8007592194
3	100	100	100	volunteerone	8007592194	8007592194
4	200	200	200	Not Assigned	--	--


Connect with us




Instagram



Twitter



Facebook



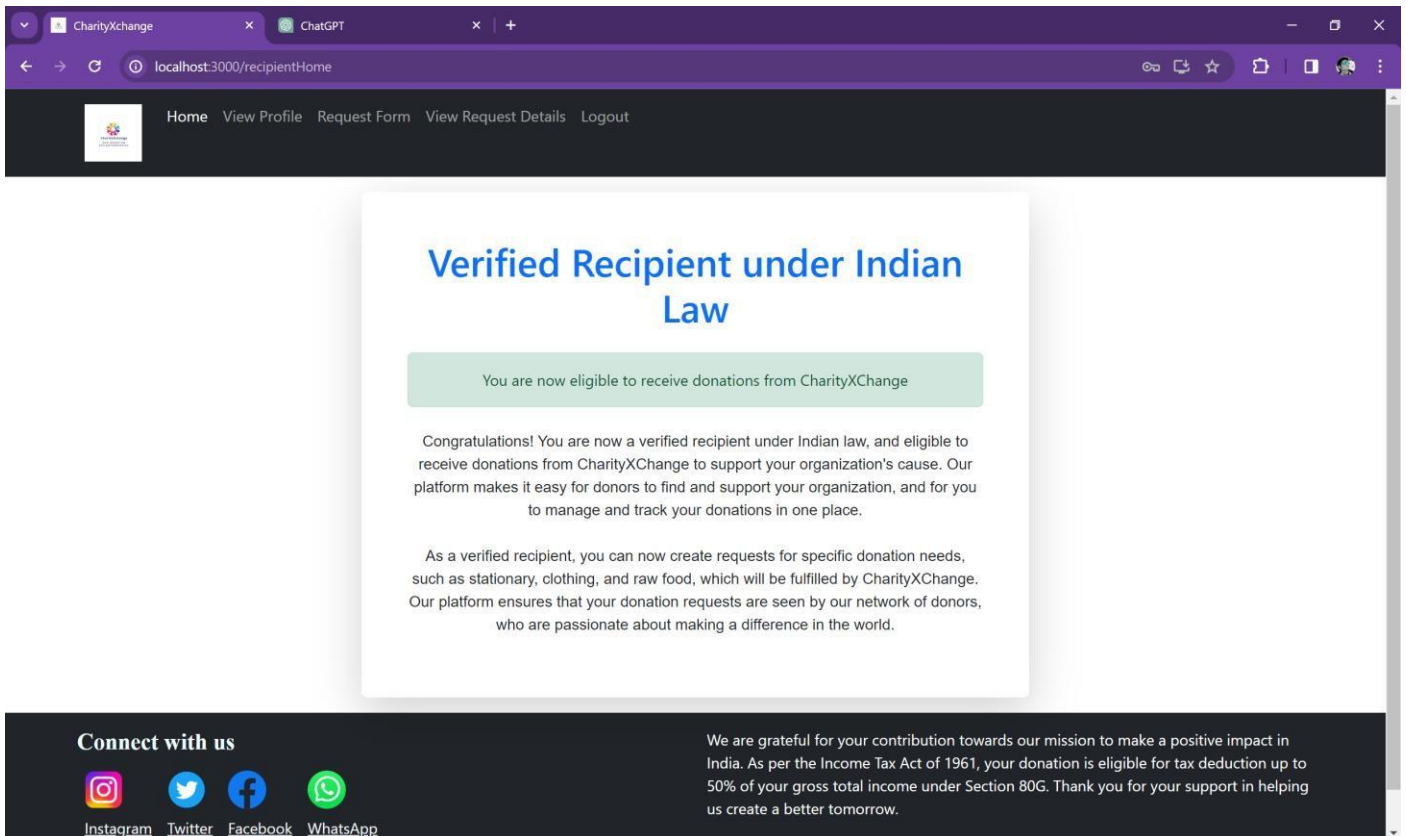
WhatsApp

We are grateful for your contribution towards our mission to make a positive impact in India. As per the Income Tax Act of 1961, your donation is eligible for tax deduction up to 50% of your gross total income under Section 80G. Thank you for your support in helping us create a better tomorrow.

© 2023 CharityXchange. All rights reserved.

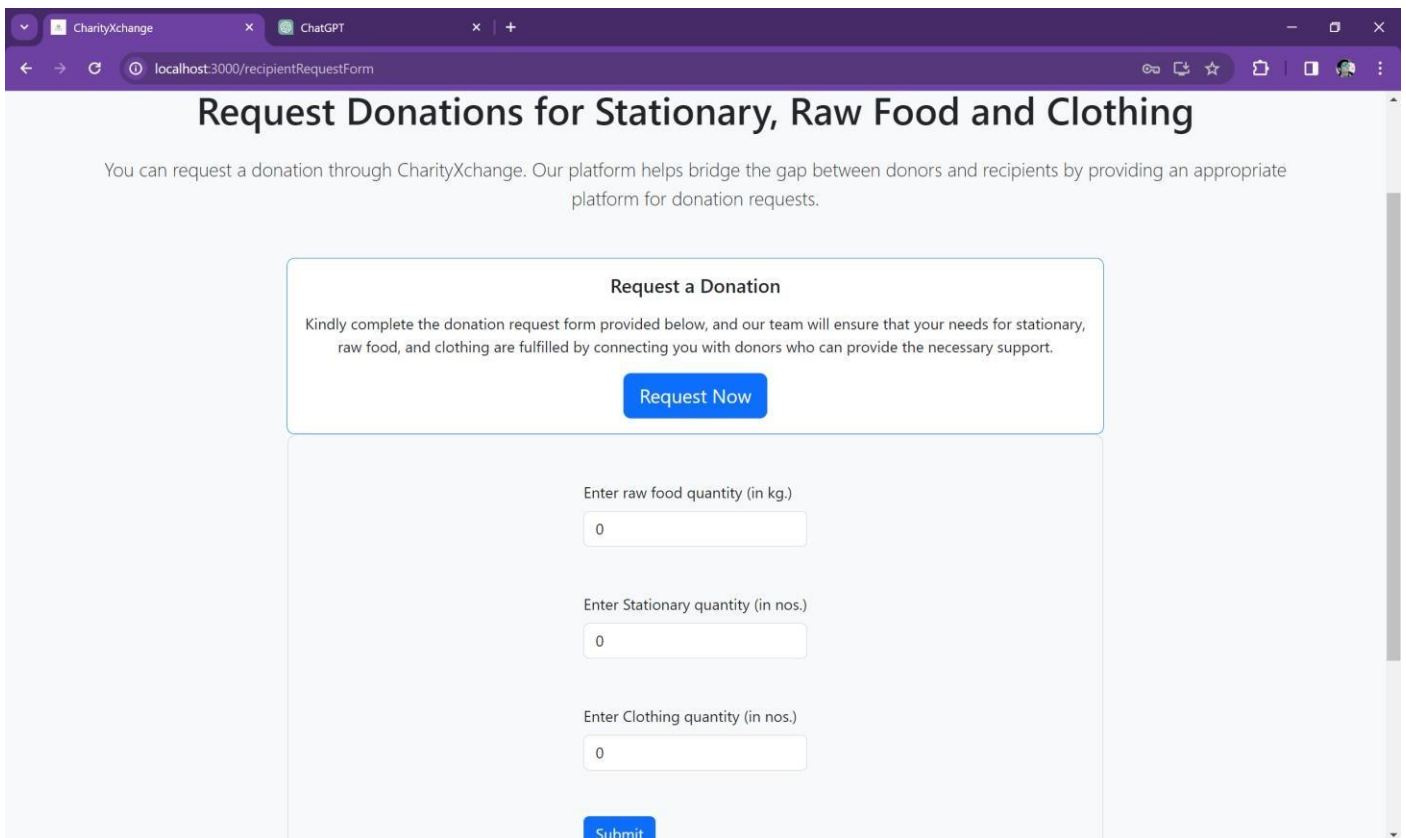
41

Recipient Dashbooard



The screenshot shows a web browser window with the URL `localhost:3000/recipientHome`. The page has a dark purple header with a navigation bar containing links: Home, View Profile, Request Form, View Request Details, and Logout. The main content area is white and features a large blue heading "Verified Recipient under Indian Law". Below this, a green box states: "You are now eligible to receive donations from CharityXChange". The text continues: "Congratulations! You are now a verified recipient under Indian law, and eligible to receive donations from CharityXChange to support your organization's cause. Our platform makes it easy for donors to find and support your organization, and for you to manage and track your donations in one place." It then explains: "As a verified recipient, you can now create requests for specific donation needs, such as stationary, clothing, and raw food, which will be fulfilled by CharityXChange. Our platform ensures that your donation requests are seen by our network of donors, who are passionate about making a difference in the world." At the bottom, there is a dark purple footer with social media links (Instagram, Twitter, Facebook, WhatsApp) and a paragraph of text: "We are grateful for your contribution towards our mission to make a positive impact in India. As per the Income Tax Act of 1961, your donation is eligible for tax deduction up to 50% of your gross total income under Section 80G. Thank you for your support in helping us create a better tomorrow."

Recipient can request so admin can donate to him when he get donation from donor




The screenshot shows a web browser window with the URL `localhost:3000/recipientRequestForm`. The page has a dark purple header with the same navigation bar as the previous screenshot. The main content area is white and features a large black heading "Request Donations for Stationary, Raw Food and Clothing". Below this, a paragraph states: "You can request a donation through CharityXchange. Our platform helps bridge the gap between donors and recipients by providing an appropriate platform for donation requests." The form is titled "Request a Donation" and contains the text: "Kindly complete the donation request form provided below, and our team will ensure that your needs for stationary, raw food, and clothing are fulfilled by connecting you with donors who can provide the necessary support." There is a blue "Request Now" button. Below this, there are three input fields: "Enter raw food quantity (in kg.)" with a value of 0, "Enter Stationary quantity (in nos.)" with a value of 0, and "Enter Clothing quantity (in nos.)" with a value of 0. At the bottom of the form is a blue "Submit" button.

CharityXchange
ChatGPT
localhost:3000/recipientViewRequestDetails
Home View Profile Request Form View Request Details Logout

Table for the recipient made donation requests

Quantity Required			Quantity Received		
Raw Food	Clothes	Stationary	Raw Food	Clothes	Stationary
200	200	200	300	300	300

Connect with us



[Instagram](#)
[Twitter](#)
[Facebook](#)
[WhatsApp](#)

We are grateful for your contribution towards our mission to make a positive impact in India. As per the Income Tax Act of 1961, your donation is eligible for tax deduction up to 50% of your gross total income under Section 80G. Thank you for your support in helping us create a better tomorrow.

© 2023 CharityXchange. All rights reserved.

Volunteer dashboard

CharityXchange
ChatGPT
localhost:3000/volunteerHome
Home View Profile Donations to be collected Donations to be delivered Logout

Thank You, Volunteer!

Your help is essential in making a positive impact on our community. By using our platform, you can see the details of the donations that need to be collected, verify them, and deliver them to assigned recipients.

View Donations

Browse through the list of donations that need to be collected and delivered.


Verify Donations

Make sure that the donations you collect are in good condition and match the details listed in the platform.

Deliver Donations

Deliver the donations to the assigned recipients and make a difference in their lives.

Connect with us



[Instagram](#)
[Twitter](#)
[Facebook](#)
[WhatsApp](#)

We are grateful for your contribution towards our mission to make a positive impact in India. As per the Income Tax Act of 1961, your donation is eligible for tax deduction up to 50% of your gross total income under Section 80G. Thank you for your support in helping us create a better tomorrow.


© 2023 CharityXchange. All rights reserved.

Admin now assign the volunteer

CharityXchange

ChatGPT

localhost:3000/adminDonationAssign







[Donation Assign](#) [Assigned Donation Details](#) [All Recipient Details](#) [Recipient KYC verification](#) [Logout](#)

Table for Assigning Volunteers to Collect Donations

Serial ID	Raw Food Quantity	Clothes Quantity	Stationary Quantity	Choose Volunteer	Choose Recipient	Save Changes
1	200	200	200	volunteerfive	recipientfive	Save

Connect with us



[Instagram](#) [Twitter](#) [Facebook](#) [WhatsApp](#)

We are grateful for your contribution towards our mission to make a positive impact in India. As per the Income Tax Act of 1961, your donation is eligible for tax deduction up to 50% of your gross total income under Section 80G. Thank you for your support in helping us create a better tomorrow.


© 2023 CharityXchange. All rights reserved.

Volunteer go to donor and collect the donation

CharityXchange

ChatGPT

localhost:3000/volunteerDonationToBeCollected







[Home](#) [View Profile](#) [Donations to be collected](#) [Donations to be delivered](#) [Logout](#)

Donation to be collected

Serial ID	Donor Name	Donor Contact Details	Donor Address	Donor ZIP Code	Donation Status	Click To Update
1	donorfive	8007592194	pune	431005	Donation To Be Collected	Save

Connect with us

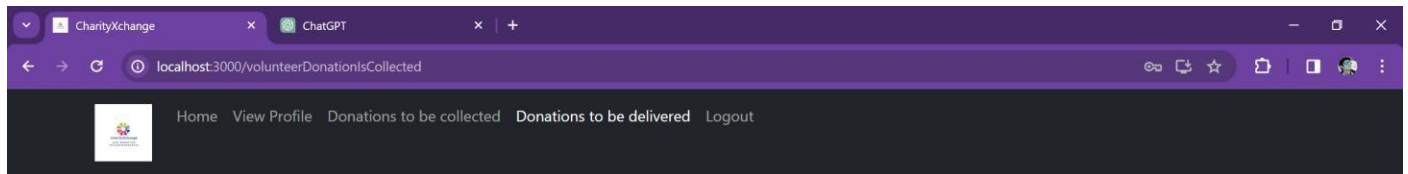


[Instagram](#) [Twitter](#) [Facebook](#) [WhatsApp](#)

We are grateful for your contribution towards our mission to make a positive impact in India. As per the Income Tax Act of 1961, your donation is eligible for tax deduction up to 50% of your gross total income under Section 80G. Thank you for your support in helping us create a better tomorrow.

© 2023 CharityXchange. All rights reserved.

Volunteer go to recipient and donate



Donation to be delivered by the Volunteer

Serial ID	Recipient Name	Recipient Contact Details	Recipient Address	Recipient ZIP Code	Donation Status	Click To Update
1	recipientfive	8007592194	pune	431005	<div>Donation is Collected</div> <div>Donation is Collected</div> <div>Donation is Delivered</div>	<div>Save</div>

Connect with us

[Instagram](#)[Twitter](#)[Facebook](#)[WhatsApp](#)

We are grateful for your contribution towards our mission to make a positive impact in India. As per the Income Tax Act of 1961, your donation is eligible for tax deduction up to 50% of your gross total income under Section 80G. Thank you for your support in helping us create a better tomorrow.

© 2023 CharityXchange. All rights reserved.

6. TESTING

It is a process of ensuring acceptable degree of quality attributes of the software, of finding errors and its purpose is to ensure correctness, robustness, reliability and many other quality attributes.

6.1 Testability

Software testability is the degree to which a software system or module supports testing in a given test context. This cannot be measured directly. A lower degree of testability results in increased test efforts.

6.1.1 Observability

It refers to how easy it is to observe the behavior of a program in terms of its output, effect on the environment and other hardware and software components. In this the incorrect input and output is easily identified and verified.

6.1.2 Decomposability

By controlling the scope of testing, problems can be isolated quickly and smarter testing can be performed. The software system is built from independent modules which can be tested independently.

6.1.3 Stability

In this, fewer the changes, fewer disruptions to testing, that is changes to the software must be infrequent and in a controlled manner. It means that changes to the software do not invalidate existing tests in software testing, due to this the software recovers from failures in software testing.

6.2 Testing Method Used

The Testing method which we have used for our project is black box testing. We used this method because in this approach to testing, the tests are derived on the basis of the requirements or specifications of the module and the internals of the module are not considered. This involves only observation of the output for some input values and there is no attempt to analyze the code which produces the output. The internal structure of the program is ignored. The tester gives inputs to the system and checks the corresponding outputs. If outputs are not satisfactory, then test has successfully detected a problem with the software

7. FUTURE SCOPE

- **Enhanced Mobile Features:** Develop dedicated mobile apps for iOS and Android to offer a more optimized user experience on mobile devices.
- **Multi-Language Support:** Add support for multiple languages to cater to a broader audience across different regions in India.
- **Advanced KYC Verification System:** Allow recipients to upload essential documents (e.g., government-issued ID, proof of address, and income certificate). Use AI to automatically verify these documents for authenticity and flag any discrepancies.

8. CONCLUSION

At the end it is conclude that, we have made effort on the following points.

- Made statement of the aim and objective of the project.
- We define the problem on which we are working in this project.
- We designed the simple user interface and resolved security issues of the system.
- Finally, system is implemented and tested accordingly to the test cases.
- At personal level we learned to work with team, implement software development technology and principles and good programing practice.

9. APPENDIX

Appendix I

- GUI: Graphical User Interface
- Platform dependent operating system which should be compatible with our software.
- SQL: Structured Query Language

10. REFERENCE

- J2EE: The complete Reference, Published by McGraw Hill Education in 2002.
- Google for problem solving
- <https://www.w3schools.com/>
- <https://www.javatpoint.com/spring-boot-tutorial>
- <https://www.javatpoint.com/spring-mvc-tutorial>
- <http://www.google.com>
- <https://www.getbootstrap.com>
- <https://khetibuddy.com/>
- <https://enam.gov.in/web/>

