

Project Report

CS 203

DATABASE SYSTEMS

NAIKI

Members

Ovaiz Ali 18K-0137

Zaeem Ahmed 18K-0166

Muhammad Hassan 18K-0294

Teacher: Dr. Zulfiqar

20th December 2020

1) DESIGN PHASE:

a) Proposal:

i) Introduction and brief description.

As per the latest poverty estimates, 24% of Pakistan's population lives below the national poverty line; which includes 31% in rural areas and 13% in urban areas. Hence, there is a desperate need for an automated efficient system for Pakistan which connects privileged with the underprivileged people of Pakistan.

This project will provide a platform for people to donate whatever they are fortunate with (blood, money, clothes, time, knowledge), for the betterment of the lower and underprivileged class people who are living hand to mouth conditions or have lost their home due to natural disasters or any reason and for the betterment of the environment of the country. Moreover, it also encourages people to donate more and more as the user will also receive points after each donation depending upon the amount and type of donation made which can later be redeemed for free or discounted items at the selected store (or these points can be donated again in the form of cash).

Therefore, NAIKI is a system which takes Donors and the Help seekers on board in order to support those in need.

ii) What functions should the system perform? That is, inventory control, billing, ordering, etc.

A needy person will call on the NGO's customer care or visit them in person with their request of the need. The staff at NGO will lodge their needs into the system and then search it in the donation database whether any of the donors is donating what is required by the needy. This will lead to 2 outcomes; (1) The request is matched with the donation and the donor is sent a message that their donation has been accepted, (2) The request is not matched as none of the donors is donating what is needed. In this case the request will be uploaded in the database and when any match is available, the needy person will be contacted by the NGO staff.

A donor will login into their account and then publish the ad/status of their donation. After it's published, the database will automatically search for matching needs as per the donation. (1) If it is matched the donor will be prompted and asked if he wants to donate to the person prompted. If he chooses yes, the NGO staff will be notified that the request for the need has been accepted by the donor. If he chooses no, then the ad/status of donation will be published. (2) If it is not matched then the ad/status will automatically be published and whenever the matching requests will be made, the donor will be notified if he wants to donate to this person or not.

(A donor will login and search for the available requests on the database and then he can choose to donate to any kind of request that is being posted)

iii) Front-End and Back-End Technologies

Front-End Technologies

- HTML
- CSS
- JavaScript

Back-End Technologies

- Node Js
- Express Js
- MySQL

b) Normalization:

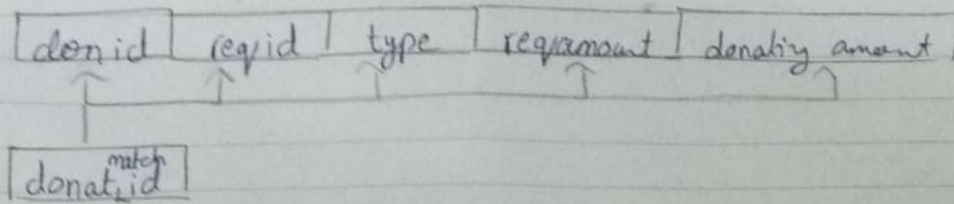
FIRST NORMAL FORM:-

USER:-

userid	name	cnic	contact	email	password	city	donorid	seekerid
PK							FK	FK
admin id	donid	reqid	ngoname	donationtype	amount	ngoloc		
FK								

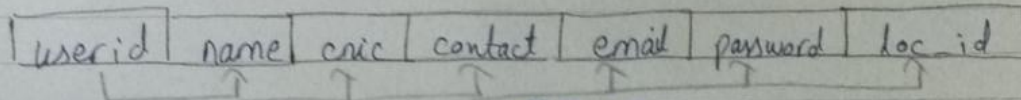
Date

DONATION MATCH:-



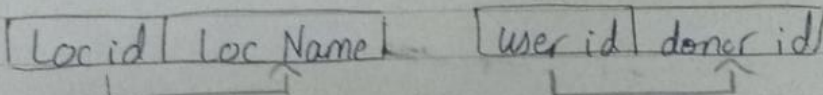
THIRD NORMAL FORM:-

User:-



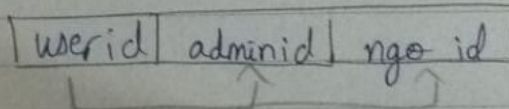
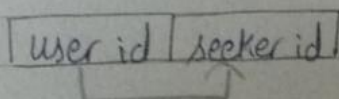
LOCATION:-

DONOR:-

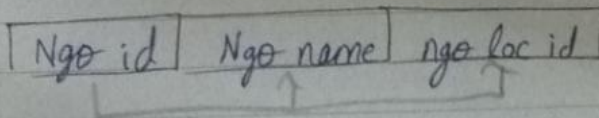


SEEKER:-

NGO ADMIN:-



NGO:-



Date _____

DONATION:-

don_id	donat type id	donating amount	donat_id
--------	---------------	-----------------	----------

Help Request:-

req_id	seeker_id	req type id	req amount
--------	-----------	-------------	------------

Type:-

type id	type name:-
---------	-------------

Donation Match:-

matchid	donid	seekerid	donatamount	remaining amount
---------	-------	----------	-------------	------------------

c) Entity Relationship Design:

i) Describe your entities. You must have enough entities to insure your project is not a "toy" system. GENERAL RULE: you should have about 5 entities and 7 or 8 tables.

◆ Entities

- Donor (Person who donates on our portal)
- Help Seeker (Person who receives donation from our portal)
- NGO (Registered NGO's on our portal, in order to place donation request or help out us if we are short of resources)
- Sys_User (General users of the portal, it covers all of them donors, help seekers and NGO admins)
- Location (To keep track of the location of stakeholders, in order to carry out further proceedings)
- NGO Employee/Admin (Representative from registered NGO's on our portal which manages all the donation requests and donations accumulated)
- Donation Request (Donation request being placed are tracked to fulfill our help seekers necessity)
- Donation Type (Specified types of donations which our portal deals in)

❖ Tables



- Sys_user (all details of user)

[illegible]

- Location (data of cities)

idLoc	INT	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
LocName	VARCHAR(45)	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

- Ngo (data of NGOs in the system)

 idNGO	INT	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
 NGOName	VARCHAR(45)	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
 Loc_id	INT	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>


- Ngo_emp (data of employees working in ngos)

 idNGO_Emp	INT	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
 user_id	INT	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
 NGO_id	INT	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>



- Donor (data of all the donors)

 idDonor	INT	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
 user_id	INT	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

- Seeker (data of all the help seekers)

 idSeeker	INT	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
 user_id	INT	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

- Donat_type (data of all the things donation can be made)

 type_id	INT	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
 type_name	VARCHAR(40)	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

- Donation (data of donations made by donor)

 donat_id	INT	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
 donor_id	INT	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
 type_id	INT	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
 quantity	INT	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
 loc_id	INT	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

- Donation_req (data of all help requests made by help seeker)

 don_id	INT	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
 seeker_id	INT	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
 type_id	INT	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
 quantity	INT	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
 loc_id	INT	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

- Don_details (data of all donations allocated to the help seeker)

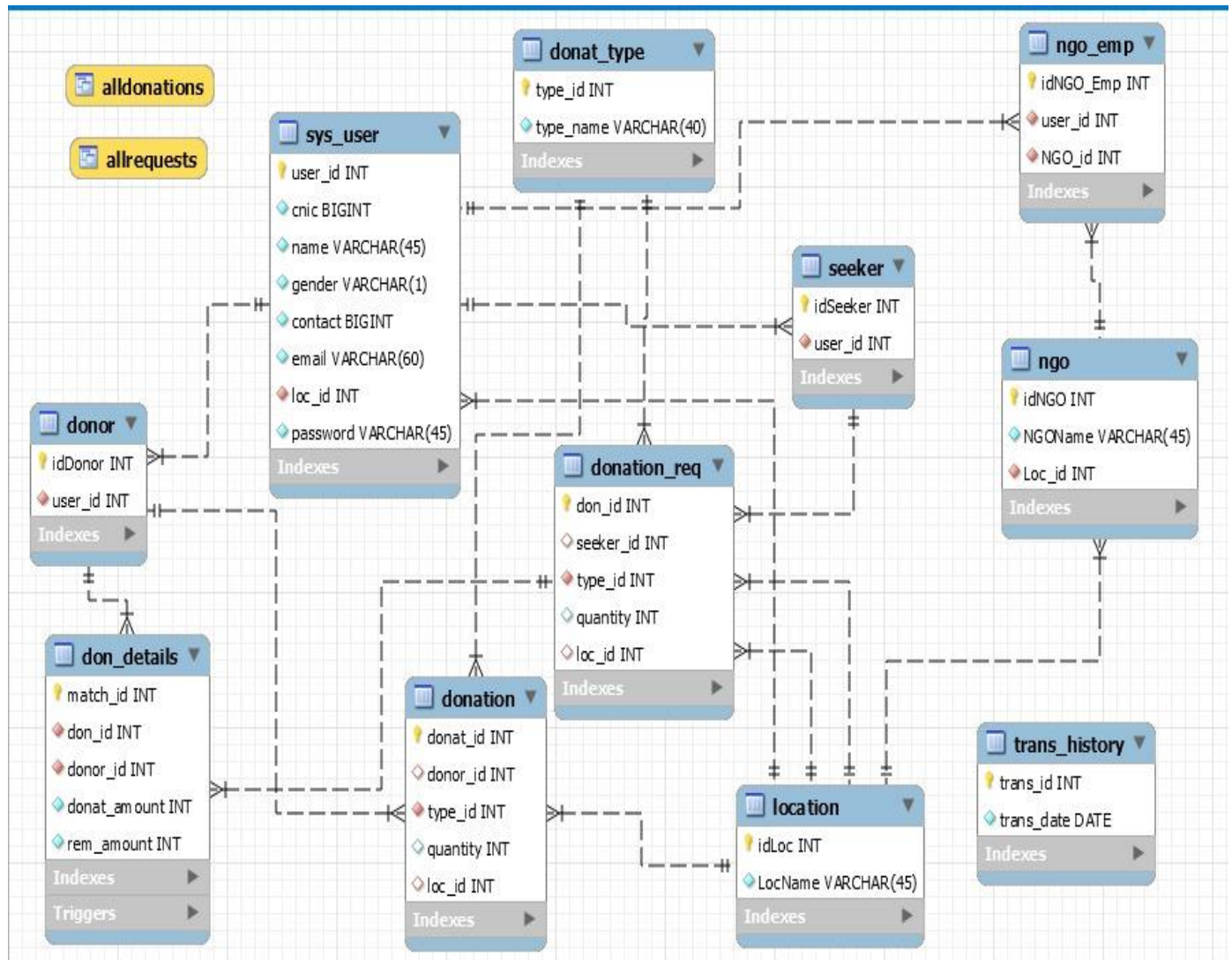
[illegible]

- Trans_history (data of the time when a donation is allocated)

[illegible]

ii) Draw the E--R Diagram for your database.

❖ Extended Entity Relationship Diagram:



2) IMPLEMENTATION PHASE:

a) Joins:

```
async getDonationData() {
  try {
    const response = await new Promise((resolve, reject) => {
      const query = "select dt.type_id, type_name from donat_type dt Join donation_req dr on dt.type_id = dr.type_id;";
```

```
async dispDonor(type) {
  try {
    const response = await new Promise((resolve, reject) => {
      const sql = 'select type_id from donat_type where type_name = ?'
      connection.query(sql, type,(err, results)=> {
        if(err) throw err;
        var typ = results[0].type_id;
        // console.log(typ);
        const query = `select distinct u.cnid from sys_user u join donor d join donation don where u.user_id = d.user_id and d.idDonor = don.donor_id and don.type_id = ${typ} and don.quantity>0`;
```

```
async dispSeeker(type,dcnid) {
  try {
    const response = await new Promise((resolve, reject) => {
      var typ;
      const sql = 'select type_id from donat_type where type_name = ?'
      connection.query(sql, type,(err, results)=> {
        if(err) throw err;
        typ = results[0].type_id;
        console.log(typ);
        const query = `select distinct u.cnid from sys_user u join seeker s join donation_req req where u.user_id = s.user_id and s.idSeeker = req.seeker_id and req.type_id = ${typ} and u.cnid <> ${dcnid} and req.quantity>0`;
```

```

async dispAmount(cnic) {
  try {
    const response = await new Promise((resolve, reject) => {
      const query = `select don.quantity from sys_user u join donation don join donor d
        where don.donor_id = d.idDonor and d.user_id = u.user_id and u.cnic = ${cnic} and don.quantity>0 limit 1`;

```

```

async disp_Req_Amount(cnic) {
  try {
    // console.log(cnic);
    const response = await new Promise((resolve, reject) => {
      const query = `select req.quantity from sys_user u join donation_req req join seeker s
        where req.seeker_id = s.idSeeker and s.user_id = u.user_id and u.cnic = ${cnic} and req.quantity>0 limit 1`;

```

```

CREATE DEFINER=`root`@`localhost` PROCEDURE `user_donation`(IN dcnic BIGINT)
BEGIN
select don.donat_id, u.name, t.type_name, don.quantity, u.contact
from donation don join sys_user u join donat_type t join donor d
where d.idDonor = don.donor_id and t.type_id = don.type_id and d.user_id = u.user_id and u.cnic =
END

```

```

async getAllUserDonatData(cnic) {
  try {
    const response = await new Promise((resolve, reject) => {
      const query = `call user_donation(?)`;

```

```

CREATE DEFINER=`root`@`localhost` PROCEDURE `user_request`(IN scnic BIGINT)
BEGIN
select req.don_id, u.name, t.type_name, req.quantity, u.contact
from donation_req req join sys_user u join donat_type t join seeker s
where s.idSeeker = req.seeker_id and t.type_id = req.type_id and s.user_id = u.user_id and u.cnic
END

```

```

async getAllUserReqData(cnic) {
  try {
    const response = await new Promise((resolve, reject) => {
      const query = `call user_request(?)`;

```

b) Advanced SQL Queries:

❖ LIMIT

```
async dispAmount(cnic) {
  try {
    const response = await new Promise((resolve, reject) => {
      const query = `select don.quantity from sys_user u join donation don join donor d
        where don.donor_id = d.idDonor and d.user_id = u.user_id and u.cnic = ${cnic} and don.quantity>0 limit 1;`;

```

```
async disp_Req_Amount(cnic) {
  try {
    // console.log(cnic);
    const response = await new Promise((resolve, reject) => {
      const query = `select req.quantity from sys_user u join donation_req req join seeker s
        where req.seeker_id = s.idSeeker and s.user_id = u.user_id and u.cnic = ${cnic} and req.quantity>0 limit 1;`;

```

```
query = `select donor_id, donat_id from donation where type_id = (select type_id from donat_type where
  type_name = "${type}") and quantity = ${don_amount} limit 1;`;
connection.query(query,(err,result)=>{
```

❖ SUB QUERIES

```
async insertMatch(type, don_cnic, seek_cnic, don_amount, req_amount){
  try{
    const response = await new Promise((resolve, reject) => {
      var s_id;
      var d_id;
      var dona_id;
      let query = `select don_id from donation_req req where req.seeker_id in (select s.idSeeker from seeker s
        where s.user_id = ( select u.user_id from sys_user u where u.cnic = ${seek_cnic})) and req.type_id = (select
        t.type_id from donat_type t where t.type_name = "${type}") and quantity = ${req_amount};`;
      connection.query(query,(err,result)=> {

```

```
var donor = `select idDonor from donor where user_id in (select user_id from sys_user where cnic
  = ?)`;
connection.query(donor,cnic, function(err,result){
```

```
var seeker = `select idSeeker from seeker where user_id in (select user_id from sys_user where
cnic = ?)`;
connection.query(seeker,cnic, function(err,result){
    if(err) throw err;
    console.log(result);
});
```

```
async checkseeker(){
    try {
        const response = await new Promise((resolve, reject) => {
            const query = "SELECT cnic from sys_user where user_id in (select user_id from seeker);";
```

❖ DISTINCT

```
async dispDonor(type) {
    try {
        const response = await new Promise((resolve, reject) => {
            const sql = 'select type_id from donat_type where type_name = ?'
            connection.query(sql, type,(err, results)=> {
                if(err) throw err;
                var typ = results[0].type_id;
                // console.log(typ);
                const query = `select distinct u.cnic from sys_user u join donor d join donation don where u.user_id = d.
user_id and d.idDonor = don.donor_id and don.type_id = ${typ} and don.quantity>0`;
                connection.query(query, (err, results) => {
                    if(err) throw err;
                    console.log(results);
                });
            });
        });
    } catch (err) {
        console.log(err);
    }
}
```

```
async dispSeeker(type,dcnic) {
    try {
        const response = await new Promise((resolve, reject) => {
            var typ;
            const sql = 'select type_id from donat_type where type_name = ?'
            connection.query(sql, type,(err, results)=> {
                if(err) throw err;
                typ = results[0].type_id;
                console.log(typ);
                const query = `select distinct u.cnic from sys_user u join seeker s join donation_req req where u.
user_id = s.user_id and s.idSeeker = req.seeker_id and req.type_id = ${typ} and u.cnic <> ${dcnic} and
req.quantity>0`;
                connection.query(query, (err, results) => {
                    if(err) throw err;
                    console.log(results);
                });
            });
        });
    } catch (err) {
        console.log(err);
    }
}
```


❖ PROCEDURES

```
let sql = `call new_donor_seeker(?)`;
connection.query(sql,uid, function(err,result)
{
```

```
CREATE DEFINER=`root`@`localhost` PROCEDURE `new_donor_seeker`(IN uid int)
> BEGIN
    insert into seeker(user_id) value(uid);
    insert into donor(user_id) value(uid);
END
```

```
async getSignInDetails() {
    try {
        const response = await new Promise((resolve, reject) => {
            const query = "call login()";
```

```
CREATE DEFINER=`root`@`localhost` PROCEDURE `login`()
> BEGIN
    select cnic, password from sys_user;
END
```

```
async getAllUserDonatData(cnic) {
    try {
        const response = await new Promise((resolve, reject) => {
            const query = `call user_donation(?)`;
```

```
CREATE DEFINER=`root`@`localhost` PROCEDURE `user_donation`(IN dcnic BIGINT)
BEGIN
select don.donat_id, u.name, t.type_name, don.quantity, u.contact
from donation don join sys_user u join donat_type t join donor d
where d.idDonor = don.donor_id and t.type_id = don.type_id and d.user_id = u.user_id and u.cnic =
```

```
async getAllUserReqData(cnic) {
    try {
        const response = await new Promise((resolve, reject) => {
            const query = `call user_request(?)`;
```

```
CREATE DEFINER=`root`@`localhost` PROCEDURE `user_request`(IN scnic BIGINT)
BEGIN
select req.don_id, u.name, t.type_name, req.quantity, u.contact
from donation_req req join sys_user u join donat_type t join seeker s
where s.idSeeker = req.seeker_id and t.type_id = req.type_id and s.user_id = u.user_id and u.cnic
END
```

- ```
CREATE DEFINER=`root`@`localhost` PROCEDURE `commit_trans`()
BEGIN
 commit;
END
```

## ❖ VIEWS

```
async getAllDonatData() {
 try {
 const response = await new Promise((resolve, reject) => {
 const query = "select * from alldonations; ";
 });
 }
}
```

You, 21 minutes ago • Uncommitted changes

```

CREATE
 ALGORITHM = UNDEFINED
 DEFINER = `root`@`localhost`
 SQL SECURITY DEFINER
VIEW `alldonations` AS
 SELECT
 `don`.`donat_id` AS `donat_id`,
 `u`.`name` AS `name`,
 `t`.`type_name` AS `type_name`,
 `don`.`quantity` AS `quantity`,
 `u`.`contact` AS `contact`
 FROM
 (((`donation` `don`
 JOIN `sys_user` `u`)
 JOIN `donat_type` `t`)
 JOIN `donor` `d`)
 WHERE
 ((`d`.`idDonor` = `don`.`donor_id`)
 AND (`t`.`type_id` = `don`.`type_id`)
 AND (`d`.`user_id` = `u`.`user_id`))

```

```

async getAllReqData() {
 try {
 const response = await new Promise((resolve, reject) => {
 const query = "select * from allrequests;";

```

```

CREATE
 ALGORITHM = UNDEFINED
 DEFINER = `root`@`localhost`
 SQL SECURITY DEFINER
VIEW `allrequests` AS
 SELECT
 `req`.`don_id` AS `don_id`,
 `u`.`name` AS `name`,
 `t`.`type_name` AS `type_name`,
 `req`.`quantity` AS `quantity`,
 `u`.`contact` AS `contact`
 FROM
 (((`donation_req` `req`
 JOIN `sys_user` `u`)
 JOIN `donat_type` `t`)
 JOIN `seeker` `s`)
 WHERE
 ((`s`.`idSeeker` = `req`.`seeker_id`)
 AND (`t`.`type_id` = `req`.`type_id`)
 AND (`s`.`user_id` = `u`.`user_id`))

```

## TRIGGER

```
DELIMITER $$
```

```

CREATE TRIGGER commit_trans
 AFTER INSERT
 ON don_details FOR EACH ROW
BEGIN
 insert into trans_history(trans_date) value(CURDATE());
END$$

```

```
DELIMITER ;
```

### c) Built-in Functions:

```
async countRequest() {
 try {
 const response = await new Promise((resolve, reject) => {
 const query = `select count(don_id) from donation_req`;
```

```
async countDonation() {
 try {
 const response = await new Promise((resolve, reject) => {
 const query = `select count(donat_id) from donation`;
```

```
async countMatch() {
 try {
 const response = await new Promise((resolve, reject) => {
 const query = `select count(match_id) from don_details`;
```

## d) User Input:

### ❖ SIGNUP

Sign In Sign Up

Username

Cnic

Contact

E-mail

Choose City ▼

Password Hide

☐ Male ☐ Female

☐ I agree to the [Terms](#)


Create account

```
async insertUser(name,cnic,gender,contact,email,city,password){
 try{
 const response = await new Promise((resolve, reject)=>{
```



## ❖ DONATION FORM FOR DONOR

Millions of people's life has changed, Million more are awaiting for your donations

[Home](#) [About](#) [NAIKI Services ▾](#) [Donation ▾](#) [Contact Us](#) [Logout](#) [Donate Now](#)


### Donation Form

|                                                        |                                        |                                     |
|--------------------------------------------------------|----------------------------------------|-------------------------------------|
| Name                                                   | CNIC No.                               |                                     |
| <input type="text" value="Mark"/>                      | <input type="text" value="13 digit"/>  |                                     |
| City                                                   | Donation Type                          | Quantity                            |
| <input type="text" value="Choose..."/>                 | <input type="text" value="Choose..."/> | <input type="text" value="digits"/> |
| <input type="checkbox"/> Agree to terms and conditions |                                        |                                     |
| <a href="#">Submit form</a>                            |                                        |                                     |


**NAIKI**  
Here to create a better world using our utmost potential. We are here because we believe in achieving goals beside setting meagre limits.

**Quick Links**  
[Home](#)  
[About Us](#)  
[Services](#)  
[Contact Us](#)

**Contact Us**  
St-4 Sector 17-D On National Highway  
Karachi, Pakistan.  
+92-332-2847298


[info@domain.com](mailto:info@domain.com)  


Copyright NAIKI © 2020. All rights reserved.



```
async setDonation(Name, cnic, city, type, quantity) {
 try {
```

## ❖ REQUEST FORM FOR HELP SEEKER



HomeAboutNAIKI Services ▾Donation ▾Contact Us

LogoutDonate Now

Millions of people's life has changed, Million more are awaiting for your donations


### Request Form

|                                                        |                                        |                                     |
|--------------------------------------------------------|----------------------------------------|-------------------------------------|
| Name                                                   | CNIC No.                               |                                     |
| <input type="text" value="Mark"/>                      | <input type="text" value="13 digit"/>  |                                     |
| City                                                   | Donation Type                          | Quantity                            |
| <input type="text" value="Choose..."/>                 | <input type="text" value="Choose..."/> | <input type="text" value="digits"/> |
| <input type="checkbox"/> Agree to terms and conditions |                                        |                                     |
| <input type="button" value="Submit form"/>             |                                        |                                     |

**NAIKI**  
Here to create a better world using our utmost potential. We are here because we believe in achieving goals beside setting meagre limits.

**Quick Links**  
[Home](#)  
[About Us](#)  
[Services](#)  
[Contact Us](#)

**Contact Us**  
St-4 Sector 17-D On National Highway  
Karachi, Pakistan.  
+92-332-2847298

[info@domain.com](mailto:info@domain.com)  



Copyright NAIKI © 2020. All rights reserved.



```
async setDonationReq(Name, cnic, city, type, quantity) {
 try {
 You, an hour ago • Uncommitted changes }
}
```

## ❖ DONATION ALLOCATION FORM FOR NGO EMPLOYEE

Millions of people's life has changed, Million more are awaiting for your donations

 [Home](#) [About](#) [NAIKI Services](#) [Donation](#) [Contact Us](#) [Logout](#) [Donate Now](#)

### Allocate Donations

Donation Type:

Donor CNIC No.:

Help Seeker CNIC No.:

Amount/Donation Available:

Amount/Donation Required:

☐ Agree to terms and conditions

[Allocate Donation](#)

---

**NAIKI**  
Here to create a better world using our utmost potential. We are here because we believe in achieving goals beside setting meagre limits.

**Quick Links**  
[Home](#)  
[About Us](#)  
[Services](#)  
[Contact Us](#)

**Contact Us**  
St-4 Sector 17-D On National Highway  
Karachi, Pakistan.  
+92-332-2847298  
[info@domain.com](mailto:info@domain.com)  
[in](#) [f](#) [t](#) [@](#)

Copyright NAIKI © 2020. All rights reserved.

```
async insertMatch(type, don_cnic, seek_cnic, don_amount, req_amount){
 try{
```

### e) Insert:

#### ❖ NEW USER

```
let sql = `insert into sys_user (name,cnic,gender,contact,email,loc_id,password) values ("${name}","${cnic}","${gender}","${contact}","${email}","${1}","${password}")`;
connection.query(sql, function(err,result)
```

### ❖ NEW DONOR AND SEEKER

```
CREATE DEFINER=`root`@`localhost` PROCEDURE `new_donor_seeker`(IN uid int)
BEGIN
 insert into seeker(user_id) value(uid);
 insert into donor(user_id) value(uid);
END
```

### ❖ NEW DONATION

```
let sql = `insert into donation (donor_id,type_id,quantity,loc_id) values (${d},${t},"${quantity}",${l})`;
connection.query(sql, function(err,result)
{
```

### ❖ NEW HELP REQUEST

```
let sql = `insert into donation_req (seeker_id,type_id,quantity,loc_id) values (${s},${t},"${quantity}",${l})`;
connection.query(sql, function(err,result)
{
```

### ❖ NEW DONATION ALLOCATION

```
let sql = `insert into don_details(don_id, donor_id, donat_amount, rem_amount) values(${s_id}, ${d_id}, ${don_amount}, ${rem_s});`;
connection.query(sql, (err,result)=>{
```

## ❖ ALLOCATION TIME

```
CREATE TRIGGER commit_trans
 AFTER INSERT
 ON don_details FOR EACH ROW
BEGIN
 insert into trans_history(trans_date) value(CURDATE());
END$$
```

---

### f) Update:

```
in function insertMatch
```

```
sql = `update donation set quantity = ${rem_d} where donat_id = ${dona_id}`;
connection.query(sql, function(err,result)
```

```
sql = `update donation_req set quantity = ${rem_s} where don_id = ${s_id}`;
connection.query(sql, function(err,result)
```

### g) Delete

```
async deleteDonation(dcnic) {
 try {
 const response = await new Promise((resolve, reject) => {
 const query = `delete from donation where donor_id = (select donor_id from donor where user_id = (select user_id from sys_user where cnic = ${dcnic}))`;
 });
 }
}
```

```
async deleteRequest(scnic) {
 try {
 const response = await new Promise((resolve, reject) => {
 const query = `delete from donation_req where seeker_id = (select seeker_id from seeker where user_id =
 (select user_id from sys_user where cnic = ${scnic}))`;
```

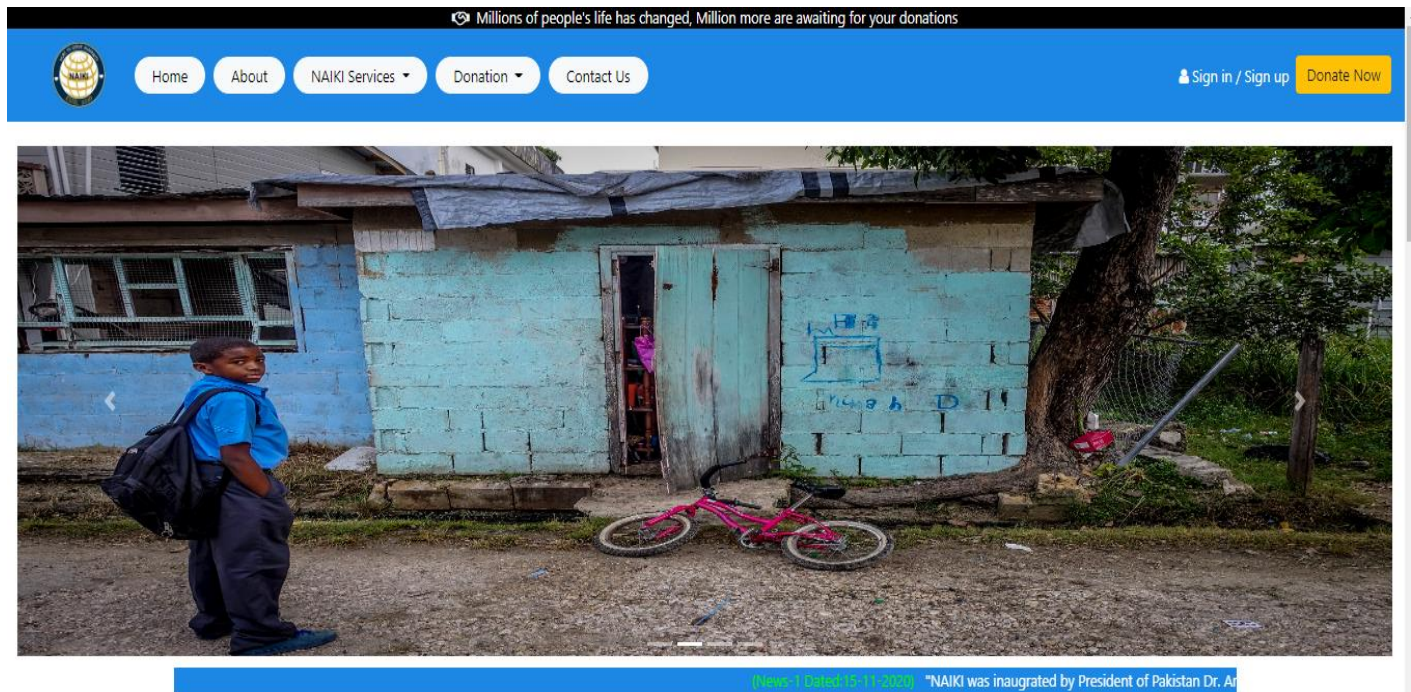
## h) Transaction:

```
CREATE DEFINER=`root`@`localhost` PROCEDURE `commit_trans`()
BEGIN
 commit;
END
```



# SCREENSHOTS OF PROJECT

## ❖ HOME PAGE



# INTRODUCTION

NAIKI is a platform which connects potential Donors and deserving Donation Seekers in order to achieve our organisation's foremost goal of poverty alleviation from developing countries.

Hence, we are working as an independent, non-governmental, non-profit and charitable organization. It works under its own Memorandum of Articles of Association and leadership.



## WHAT ARE WE DOING ?

NAIKI proudly presents the streams of services offered within it's platform to the Donors and Donation Seekers within few clicks.





#### STEP 1

Reachout to our online portal (NAIKI) or any NGO's booth to fill out your request for Donations.



#### STEP 2

After filling out request or forms your requirements would be made visible as an anonymous to the Donors and vice versa.



#### STEP 3

As soon as the deserved candidate for the placed request in either cases is available their credentials are sent to the stakeholders.



#### FINISH

After the confirmation and verification all existing means are used to fulfill the needs of Donation seekers.



#### NAIKI

Here to create a better world using our utmost potential. We are here because we believe in achieving goals beside setting meagre limits.

#### Quick Links

[Home](#)  
[About Us](#)  
[Services](#)  
[Contact Us](#)

#### Contact Us

📍 St-4 Sector 17-D On National Highway  
Karachi, Pakistan.  
☎ +92-332-2847298

[info@domain.com](mailto:info@domain.com)



# ❖ LOGIN

Millions of people's life has changed, Million more are awaiting for your donations

Home About NAKI Services

Sign In / Sign Up Donate Now

Sign In

Sign Up

4220106749441

test123

Hide

☒ Donor ☐ Help Seeker ☐ NGO Admin

☒ Remember me

Login

[Forgot your password?](#)

— — — — —

"NAKI was inaugurated by President of Pakistan Dr. Arif Alvi"

"NAKI announced it's Donation Application Form."

INTRODUCTION

# ❖SIGN UP

Sign In

Sign Up

Username

Cnic

Contact

E-mail

Choose City

Password

Hide

☐ Male

☐ Female

☐ I agree to the [Terms](#)

Create account

Home

About

NAIKI Services

ANYTHING


Hungry

HELPS

GA

INTRODUCTION

## ❖ DONOR

Millions of people's life has changed, Million more are awaiting for your donations

[Home](#) [About](#) [NAIKI Services](#) [Donation](#) [Contact Us](#) [Logout](#) [Donate Now](#)

## Welcome to NAIKI!

[View My Donations](#) [Give Donation](#)

### NAIKI

Here to create a better world using our utmost potential. We are here because we believe in achieving goals beside setting meagre limits.

### Quick Links


[Home](#)  
[About Us](#)  
[Services](#)  
[Contact Us](#)

### Contact Us

📍 St-4 Sector 17-D On National Highway  
Karachi, Pakistan.  
☎ +92-332-2847298

[info@domain.com](mailto:info@domain.com)  
[in](#) [f](#) [t](#) [@](#)

Copyright NAIKI © 2020. All rights reserved.

Millions of people's life has changed, Million more are awaiting for your donations

[Home](#) [About](#) [NAIKI Services](#) [Donation](#) [Contact Us](#) [Logout](#) [Donate Now](#)

## My Donations

| ID | Name        | Donation Type | Amount/Quantity | Contact no. |
|----|-------------|---------------|-----------------|-------------|
| 1  | Zaeem Ahmed | money         | 0               | 3311355294  |
| 3  | Zaeem Ahmed | money         | 3000            | 3311355294  |

### NAIKI

Here to create a better world using our utmost potential. We are here because we believe in achieving goals beside setting meagre limits.

### Quick Links

[Home](#)  
[About Us](#)  
[Services](#)  
[Contact Us](#)

### Contact Us

📍 St-4 Sector 17-D On National Highway  
Karachi, Pakistan.  
☎ +92-332-2847298

[info@domain.com](mailto:info@domain.com)  
[in](#) [f](#) [t](#) [@](#)

Copyright NAIKI © 2020. All rights reserved.



[Home](#)[About](#)[NAIKI Services](#)[Donation](#)[Contact Us](#)[Logout](#)[Donate Now](#)

## Donation Form

Name

Mark

CNIC No.

13 digit

City

Choose...

Donation Type

Choose...

Quantity

digits

☐ Agree to terms and conditions

Submit form

### NAIKI

Here to create a better world using our utmost potential. We are here because we believe in achieving goals beside setting meagre limits.

### Quick Links

[Home](#)  
[About Us](#)  
[Services](#)  
[Contact Us](#)

### Contact Us

St-4 Sector 17-D On National Highway  
Karachi, Pakistan.  
+92-332-2847298

[info@domain.com](mailto:info@domain.com)



## Welcome to NAIKI!

View My Help Requests

Place Help Requests

## My Help Requests

| ID | Name      | Donation Type  | Amount/Quantity | Contact no. |
|----|-----------|----------------|-----------------|-------------|
| 2  | Ovaiz Ali | ration         | 0               | 3234532567  |
| 3  | Ovaiz Ali | clothes (male) | 2               | 3234532567  |

[Home](#)[About](#)[NAIKI Services](#)[Donation](#)[Contact Us](#)[Logout](#)[Donate Now](#)

## Request Form

Name

Mark

CNIC No.

13 digit

City

Choose...

Donation Type

Choose...

Quantity

digits

☐ Agree to terms and conditions

Submit form

### NAIKI

Here to create a better world using our utmost potential. We are here because we believe in achieving goals beside setting meagre limits.

### Quick Links

[Home](#)  
[About Us](#)  
[Services](#)  
[Contact Us](#)

### Contact Us

St-4 Sector 17-D On National Highway  
Karachi, Pakistan.  
+92-332-2847298

[info@domain.com](mailto:info@domain.com)



# ❖ NGO ADMIN

## Welcome to Admin Panel!

- View Help Requests
- View Donation Received
- Allocate Donations

## Help Requests Received

| ID | Name        | Donation Type  | Amount/Quantity | Contact no. |
|----|-------------|----------------|-----------------|-------------|
| 1  | Zaeem Ahmed | money          | 10000           | 3311355294  |
| 2  | Ovaiz Ali   | ration         | 0               | 3234532567  |
| 3  | Ovaiz Ali   | clothes (male) | 2               | 3234532567  |
| 4  | Sarah       | money          | 45000           | 3321456842  |

## Donations Received

| ID | Name        | Donation Type | Amount/Quantity | Contact no. |
|----|-------------|---------------|-----------------|-------------|
| 1  | Zaeem Ahmed | money         | 0               | 3311355294  |
| 2  | Sarah       | ration        | 0               | 3321456842  |
| 3  | Zaeem Ahmed | money         | 3000            | 3311355294  |

## Allocate Donations

Donation Type Donor CNIC No. Help Seeker CNIC No.

money 4220106749441 4220106749444

Amount/Donation Available Amount/Donation Required

3000 45000

☒ Agree to terms and conditions

Allocate Donation