

HAPPINESS FOUNDATION: A donation site

A Project Report

Submitted by

Ayush Mundra

Jagrit Acharya

Under the Guidance of

Prof. Kamal Mistry

*in partial fulfillment for the award of the
degree of*

BTECH

Computer Engineering



**Mukesh Patel School of Technology Management &
Engineering, Mumbai**

April, 2021

DECLARATION

I, Ayush Mundra, Jagrit Acharya, Roll No. B032, B001 BTech (Computer Engineering), IV semester understand that plagiarism is defined as anyone or combination of the following:

1. Un-credited verbatim copying of individual sentences, paragraphs or illustration (such as graphs, diagrams, etc.) from any source, published or unpublished, including the internet.
2. un-credited improper paraphrasing of pages paragraphs (changing a few words phrases, or rearranging the original sentence order)
3. Credited verbatim copying of a major portion of a paper (or thesis chapter) without clear delineation of who did wrote what. (Source: IEEE, The institute, Dec. 2004)
4. I have made sure that all the ideas, expressions, graphs, diagrams, etc., that are not a result of my work, are properly credited. Long phrases or sentences that had to be used verbatim from published literature have been clearly identified using quotation marks.
5. I affirm that no portion of my work can be considered as plagiarism and I take full responsibility if such a complaint occurs. I understand fully well that the guide of the seminar/ project report may not be able to check for the possibility of such incidences of plagiarism in this body of work.

Signature of the Students: _____, _____, _____, _____

Names: Ayush Mundra, Jagrit Acharya

Roll Nos.: B032, B001

Place: Mumbai

Date:

CERTIFICATE

This is to certify that the project entitled “HAPPINESS FOUNDATION” is the Bonafide work carried out by Ayush Mundra, Jagrit Acharya of BTech, MPSTME (NMIMS), Mumbai, during the IV semester of the academic year, in partial fulfillment of the requirements for the Course Database Management System.

Prof. Kamal Mistry

Internal Mentor

Examiner 1

Examiner 2

ACKNOWLEDGEMENT

We express our gratitude to our professor Kamal Mistry for his constant guidance and support throughout the semester. We consider ourselves to be extremely privileged to have been his students. We benefited enormously from his excellence as a mentor. We are grateful to him for being patient and for all the time he spent in discussing the various concepts of Database Management Systems.

Table of contents

Chapter 1: Introduction to the system

- 1.1 Introduction
- 1.2 Problem Statement
- 1.3 Users of the system:

Chapter 2: System Design and constraints

- 2.1 ER Model
- 2.2 Reduction of ER model to Relational Model
- 2.3 Schema Diagram
- 2.4 Constraints
- 2.5 Normalization techniques applied on relational model

Chapter 3: Implementation

- 3.1 Hardware and Software details (Front end and Back end details)
- 3.2 Tools or library used
- 3.3 Screenshots and description
- 3.4 Database structure

Chapter 4: Conclusion and Future work

Chapter 1: Introduction to the system

1.1 Introduction:

- This is a website where donator can donate their things to other users and NGOs which are not useful for him/her anymore.
- We are allowing users to donate their belonging easily as they see the list of all the NGOs and other users who has posted request for help.
- We have also solved the problem for the NGO as they can also see the products (object which are there for to donate) posted by the other Donors so they can specifically contact the donors and get that product which they need.

1.2 Problem Statement:

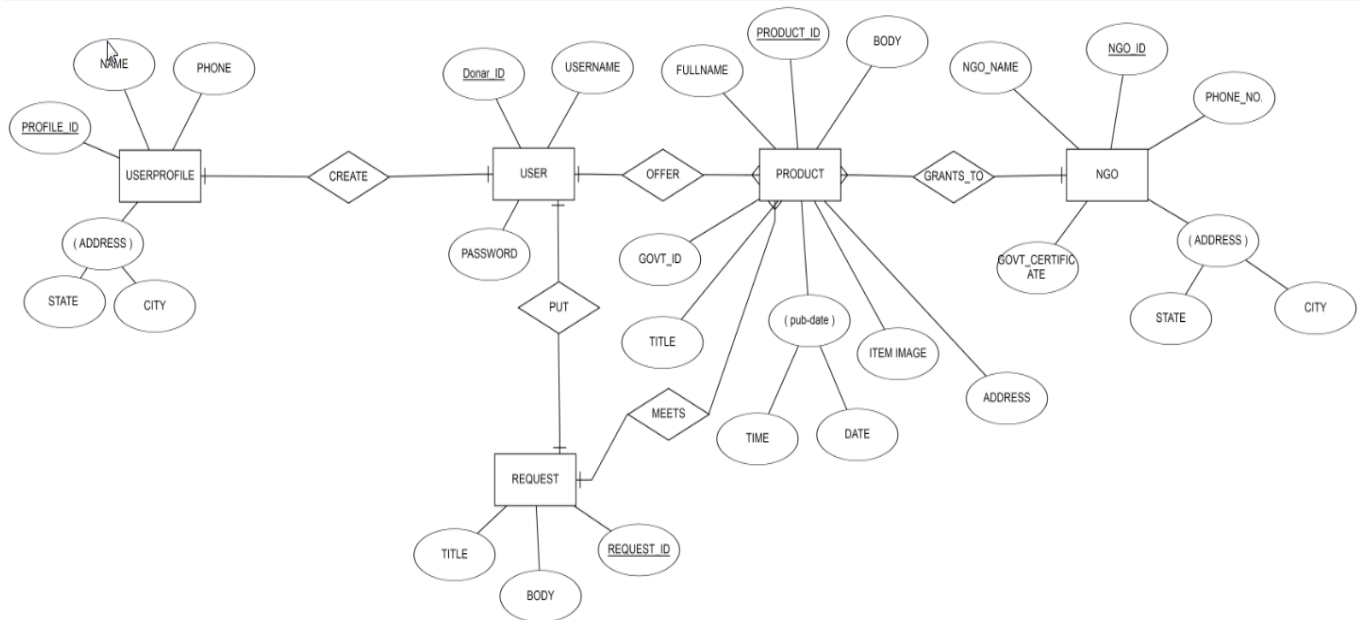
- We see in our daily life that we have goods that can be reused by other people for example (previous classes or semester books, old clothes of our family members etc.), as we don't have any correct place where we can share it with some needy people who could get benefits of those things Instead of just throwing the stuff in the garbage or selling them to other shopkeepers, it is the underutilization of the product which can be used in a better way, like sharing with others.
- NGO is not reachable, and we don't have any idea what they need.

1.3 Users of the system:

Admin, needy, helper

Chapter 2: System Design and constraints

2.1 ER Model



2.2 Reduction of ER model to Relational Model

Userprofile model:







Query Editor Query History						Scratch Pad	✕
<pre>1 SELECT * FROM public."Userprofile_profile" 2 ORDER BY id ASC</pre>							
Data Output Explain Messages Notifications							
id	Name	Phone	city	state	profile_owner_id		
[PK] integer	character varying (200)	character varying (20)	character varying (200)	character varying (200)	integer		
1	Jagrit Acharya	08875091601	Jaipur	Rajasthan	2		
2	Ayush Mundra	09079219168	Ajmer	Rajasthan	1		
3	Ayush Mundra	09079219168	Ajmer	Rajasthan	3		
4	Jagrit Acharya	08875091601	Mumbai	Maharashtra	7		

User model:








Query Editor Query History








```
1 SELECT * FROM public.auth_user
2 ORDER BY id ASC
```

Data Output Explain Messages Notifications

	 id [PK] integer	 password character varying (128)	 last_login timestamp with time zone	 is_superuser boolean	 username character varying (150)	
1	1	pbkdf2_sha256\$150000\$XCFPjx9F3...	2021-03-21 22:07:43.178301+05:30	false	Ayush Mundra	
2	2	pbkdf2_sha256\$150000\$mhtT3ofqv...	2021-03-23 22:44:41.474511+05:30	false	a	
3	3	pbkdf2_sha256\$150000\$rhTEZSHU...	2021-03-23 22:06:20.987168+05:30	true	ayush	
4	4	pbkdf2_sha256\$150000\$JXZ7ezlYy...	2021-03-22 10:35:14.141036+05:30	false	admin	
5	5	pbkdf2_sha256\$150000\$GdzgzKDN...	2021-03-22 10:35:45.322509+05:30	false	s	
6	6	pbkdf2_sha256\$150000\$M0m6im9...	2021-03-22 11:58:08.820036+05:30	false	aa	
7	7	pbkdf2_sha256\$150000\$FlvRKr7e9...	2021-03-23 22:57:27.912551+05:30	false	aaa	

Products model:

Data Output		Explain	Messages	Notifications				
	 id [PK] integer	 title character varying (200)	 pub_date timestamp with time zone	 body text	 itemimage character varying (100)	 govt_id character varying (100)	 Name character varying (200)	
1		1 Shirt	2021-03-22 16:28:21.80968+05:30	Blue shi...	images/python.png	images/Picture1.png	Jagrit Acharya	
2		2 Shoes	2021-03-22 16:28:47.658853+05:30	NIKE sh...	images/python_tkvznG.png	images/Picture1_T5uzZWu.png	Jagrit Acharyan	
3		3 Books class 9	2021-03-22 16:31:24.363904+05:30	English ...	images/as2.png	images/as1.png	Jagrit Acharya	
4		4 bottle	2021-03-27 20:17:32.782881+05:30	steel Bo...	images/WIN_20201128_23_03_59_P...	images/WIN_20210327_12_03_05_P...	Aditi Acharya	

 body text	 itemimage character varying (100)	 govt_id character varying (100)	 Name character varying (200)	 Phone character varying (20)	 address character varying (200)	 product_owner_id integer	
Blue shirt size...	images/python.png	images/Picture1.png	Jagrit Acharya	08875091601	a-41 parshwanath colony ajmer road...		6
NIKE shoes si...	images/python_tkvznG.png	images/Picture1_T5uzZWu.png	Jagrit Acharyan	08875091601	a-41 parshwanath colony ajmer road...		6
English & mat...	images/as2.png	images/as1.png	Jagrit Acharya	08875091601	a-41 parshwanath colony ajmer road...		6
steel Bottle 1 l...	images/WIN_20201128_23_03_59_P...	images/WIN_20210327_12_03_05_P...	Aditi Acharya	08875091601	dadar east		7

Ngoprofile model:

Query Editor

Query History

```
1 SELECT * FROM public.ngo_ngoprofile
2 ORDER BY id ASC
```

Scratch Pad

Query Editor

Query History

Data Output

Explain

Messages

Notifications

	<div><div>Id</div><div>[PK] integer</div></div>	<div><div>Name</div><div>character varying (200)</div></div>	<div><div>Phone</div><div>character varying (20)</div></div>	<div><div>state</div><div>character varying (200)</div></div>	<div><div>city</div><div>character varying (200)</div></div>	<div><div>ngo_creator_id</div><div>integer</div></div>	<div><div>certificates</div><div>character varying (100)</div></div>	
1	1	Jagrit Acharya	08875091601	Rajasthan	Jaipur		1 images/template_main.jpg	
2	2	Ayush Mundra	09079219168	Madhya Pradesh	Baihar		1 images/er-diagram_tOFHbEtE.png	
3	3	Jagrit Acharya	08875091601	Meghalaya	Resubelpara		2 images/aall.png	
4	4	SRINAGAR	09079219168	Rajasthan	Ajmer		3 images/LPG_Policy.pptx	

Request model:

Query Editor

Query History

```
1 SELECT * FROM public.request_needrequest
2 ORDER BY id ASC
```

Scratch Pad

Data Output

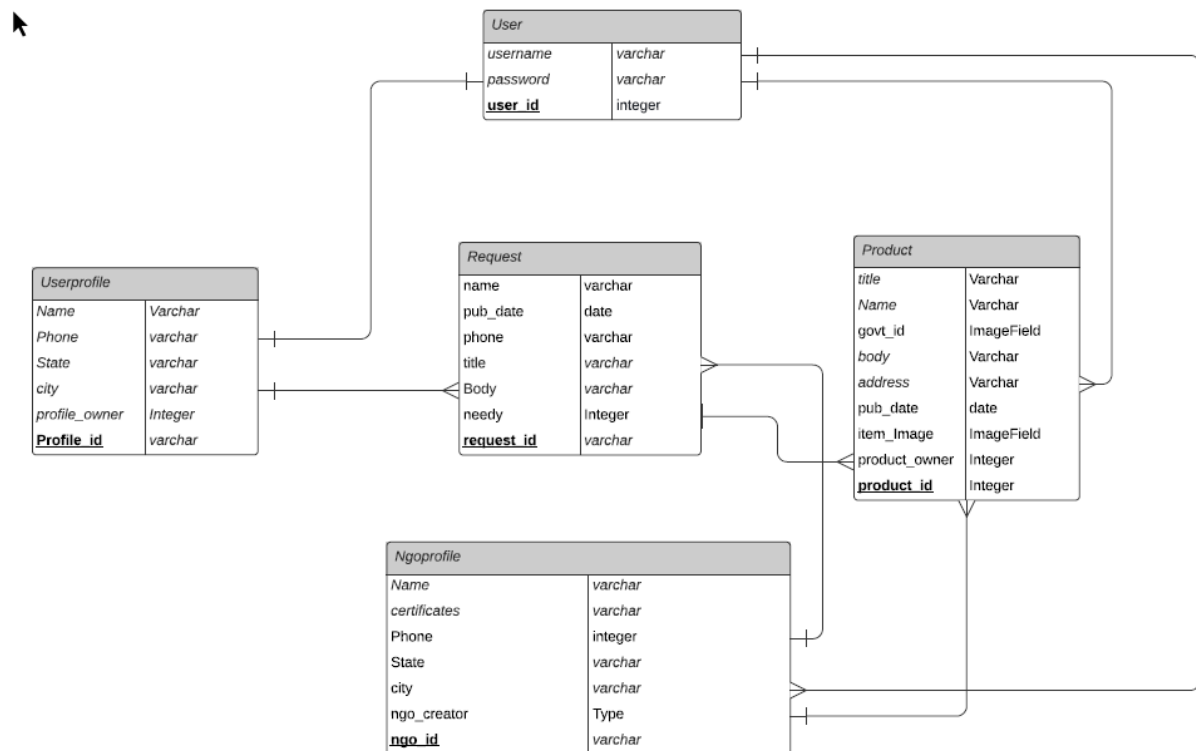
Explain

Messages

Notifications

<div><div></div></div> <div>id</div> <div>[PK] integer</div>	<div><div></div></div> <div>title</div> <div>character varying (200)</div>	<div><div></div></div> <div>pub_date</div> <div>timestamp with time zone</div>	<div><div></div></div> <div>body</div> <div>character varying (200)</div>	<div><div></div></div> <div>needy_id</div> <div>integer</div>	<div><div></div></div> <div>name</div> <div>character varying (200)</div>	<div><div></div></div> <div>phone</div> <div>character varying (200)</div>
1	1 jeans	2021-03-24 04:16:04.649308+05:30	walst 40	2	+918875091601	Jagrit Acharya
2	2 tshirt	2021-03-24 04:23:46.422989+05:30	i need a blue tshirt size small	3	09079219168	Ayush Munda

2.3 Schema Diagram



2.4 Constraints

Userprofile model:

profile_id : primary key not null,

Profile_owner : foreign key, default =1, on delete cascade on update cascade

Product model:

Product_id: primary key not null,
Product_owner: foreign key, default =0, on delete cascade on update cascade

Ngoprofile model:

ngo_id : primary key not null,
Ngo_creator : foreign key, default =1, on delete cascade on update cascade

Request model:

request_id: primary key not null,
needy: foreign key, default =1, on delete cascade on update cascade

2.5 Normalization techniques applied on relational model:

Userprofile model:

- There is no insertion, update or deletion anomaly present in the table as every entity has a separate model.
- Functional dependencies=(profile_id->Name, profile_id->Username, profile_id->Phone, profile_id->state, profile_id->city, profile_id->profile_owner.
- Prime attributes: { profile_id }
- Non-prime attributes : {Name, Username, Phone, state, city, profile_owner }
- It is 1-NF (No multivalued attribute present so no need to think about 1-NF) It is 2-NF (Since profile_id is the only candidate key so no partial dependency exist as there is no composite primary key)
- It is 3-NF (Since on every functional dependency either there is candidate key on LHS or Prime attribute on RHS).

- It is BCNF (Since on every functional dependency there is candidate key on the LHS)

For Product model:

- There is no insertion, update or deletion anomaly present in the table as every entity has a separate model(table).
- Functional dependencies=(product_id ->Name, product_id->Phone, product_id->address, product_id-> pub_date, product_id-> body, product_id-> itemImage, product_id->govt_id, product_id->product_owner)
- Prime attributes: { product_id}
- Non-prime attributes : {Name, Phone, product_owner, address, pub_date, body, itemImage, govt_id}
- It is 1-NF (No multivalued attribute present so no need to think about 1-NF)
- It is 2-NF (Since product_id is the only candidate key, so no partial dependency exist as there is no composite primary key)
- It is 3-NF (Since on every functional dependency either there is candidate key on LHS or Prime attribute on RHS)
- It is BCNF (Since on every functional dependency there is candidate key on the LHS)

For Ngoprofile model:

- There is no insertion, update or deletion anomaly present in the table as every entity has a separate model(table).

- Functional dependencies=(ngo_id ->Name, ngo_id ->Username, ngo_id->Phone, ngo_id->state, ngo_id->city, ngo_id->ngo_creator)
- Prime attributes: {ngo_id}
- Non-prime attributes : {Name, Phone, Username, city, state, owner2}
- It is 1-NF (No multivalued attribute present so no need to think about 1-NF)
- It is 2-NF (Since product_id is the only candidate key so no partial dependency exist as there is no composite primary key)
- It is 3-NF (Since on every functional dependency either there is candidate key on LHS or Prime attribute on RHS)
- It is BCNF (Since on every functional dependency there is candidate key on the LHS)

For Request model:

- There is no insertion, update or deletion anomaly present in the table as every entity has a separate model(table).
- Functional dependencies=(request_id ->name, request_id->Phone, request_id->title, request_id->pub_date, request_id->needy)
- Prime attributes: {request_id}
- Non-prime attributes : {name, title, phone, pub_date, body, needy}

- It is 1-NF (No multivalued attribute present so no need to think about 1-NF)
- It is 2-NF (Since product_id is the only candidate key so no partial dependency exist as there is no composite primary key)
- It is 3-NF (Since on every functional dependency either there is candidate key on LHS or Prime attribute on RHS)
- It is BCNF (Since on every functional dependency there is candidate key on the LHS)

Chapter 3: Implementation

3.1 Hardware and Software details (Front end and Back end details)

Software details:

Text Editor: VS code

Front end languages:

- i. HTML
- ii. CSS
- iii. Bootstrap
- iv. Javascript
- v. SCSS

Backend:

1. Django Framework
2. Python programming language
3. PostgreSQLDatabase

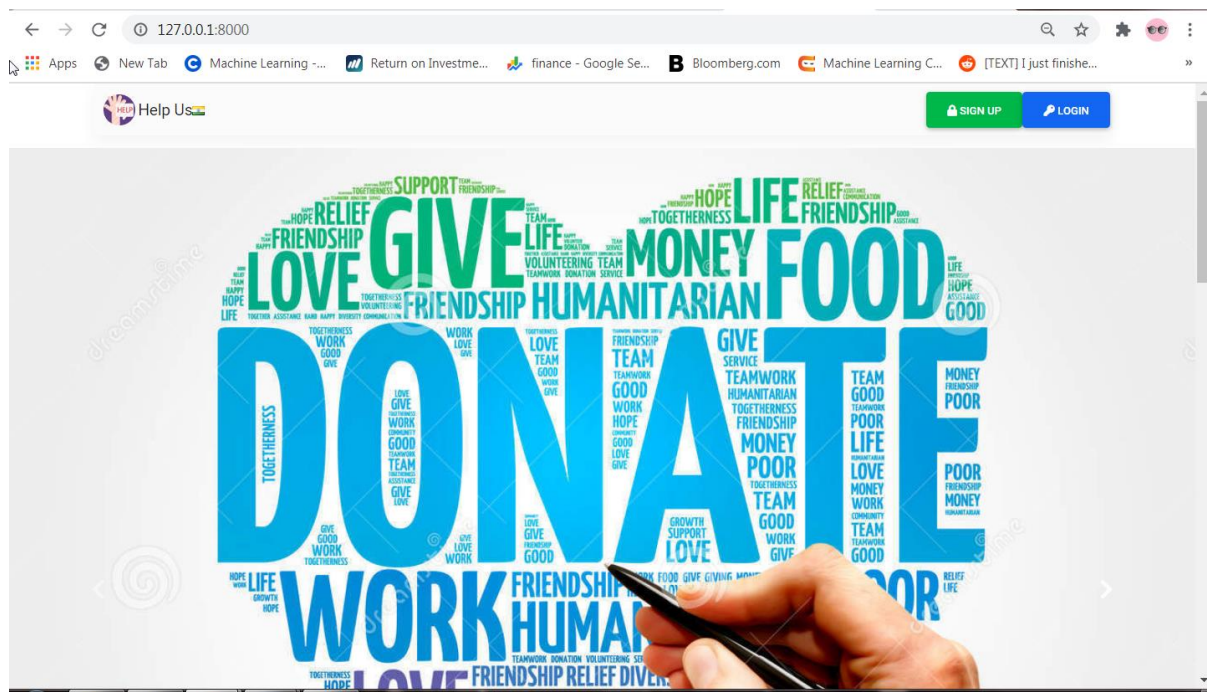
Hardware details:

Windows10, processor i7 4th generation, 8 GB RAM, 256 GB SSD

3.2 Tools or library used

- from django.contrib import admin
- from django.urls import path, include
- from django.contrib.auth.decorators import login_required
- from django.contrib.auth.models import User
- from django.utils import timezone
- from django.shortcuts import render, redirect, get_object_or_404
- from django.db import models
- from django.contrib import auth

3.3 Screenshots and description



Home page

SIGN UP

Please sign Up

Username
Jagrit Acharya

Password

Confirm-Password

Username

Password

Confirm-Password



© Jagrit Acharya 2021

Activate Windows
Go to Settings to activate Windows.

Sign Up

Login

Please Login

Username
ayush

Password

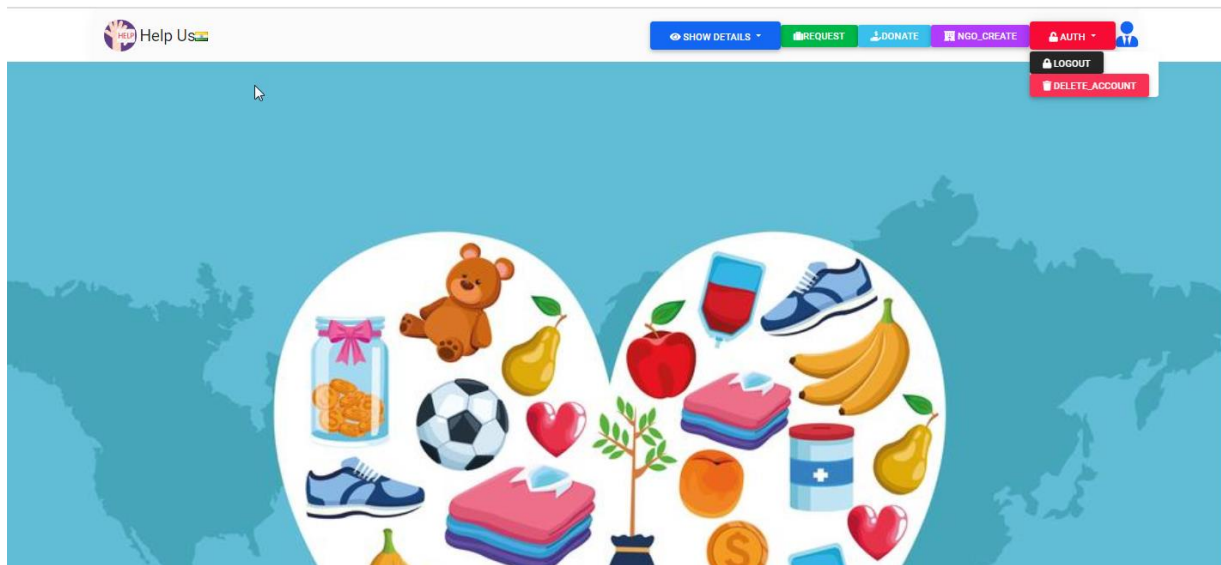
Username


Password



© Jagrit Acharya 2021

Login





Help Us


SHOW DETAILS

REQUEST

DONATE

NGO_CREATE

AUTH



Create

Full Name

Mundra NGO

title

Mundra NGO is a very famous NGO

Body

We are here to help other people

phone

9079219168

Address

Sadar bazar, near shitla mataji temple, Srinagar, Near charbhuja ji temple, Srinagar

PRODUCT IMAGE

Choose File

er-diagram.png


GOVERNMENT ID

Choose File

time table.png

SAVE PRODUCT





product id: 13

**Product Name: Physics books
class 10th**

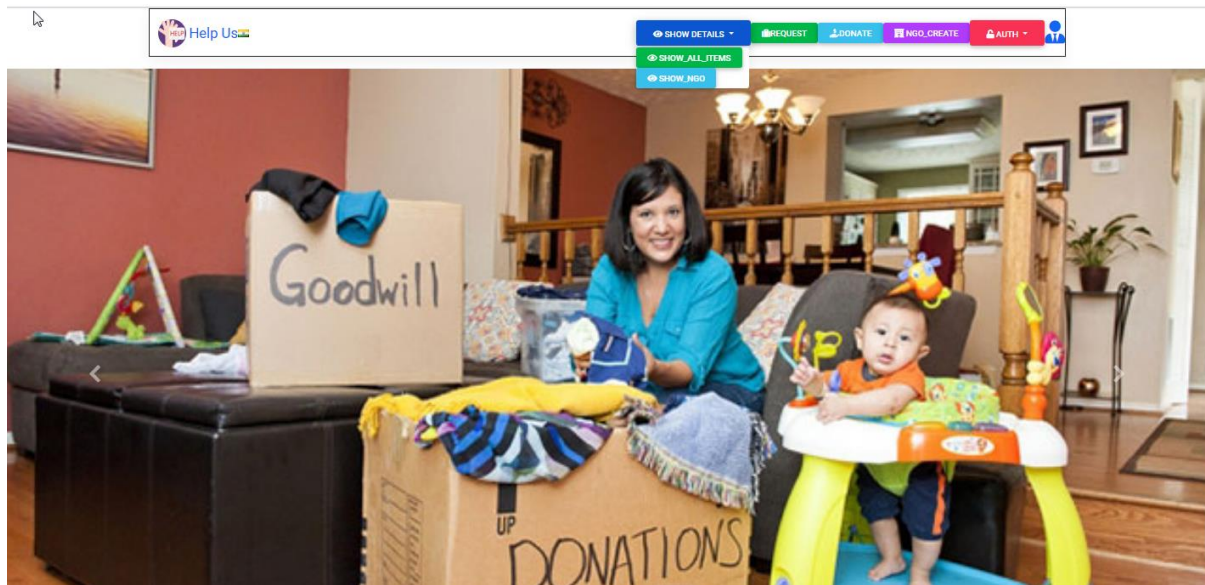
Your Name: 09079219168

Phone Number: 09079219168

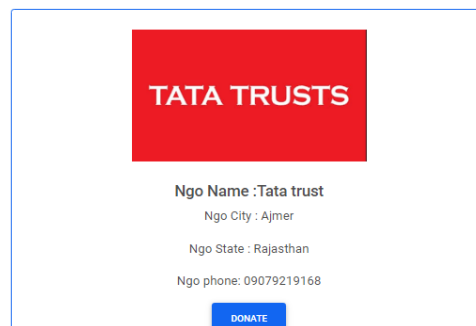
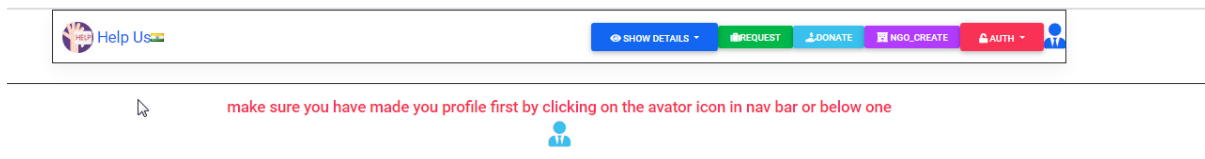
Body: I want to donate this book now it is of no
use to me

Address: banglore

Product after donation



Press show NGO to check NGOs



© Jagrit Acharya 2021

NGOs near you are shown

← → ↻ 127.0.0.1:8000/ngo/help 🔍 ☆ ⚙️ 👤

Help Us 🇮🇳

SHOW DETAILS REQUEST DONATE NGO_CREATE AUTH

Create Your Ngo Profile

Ngo name

Phone Number

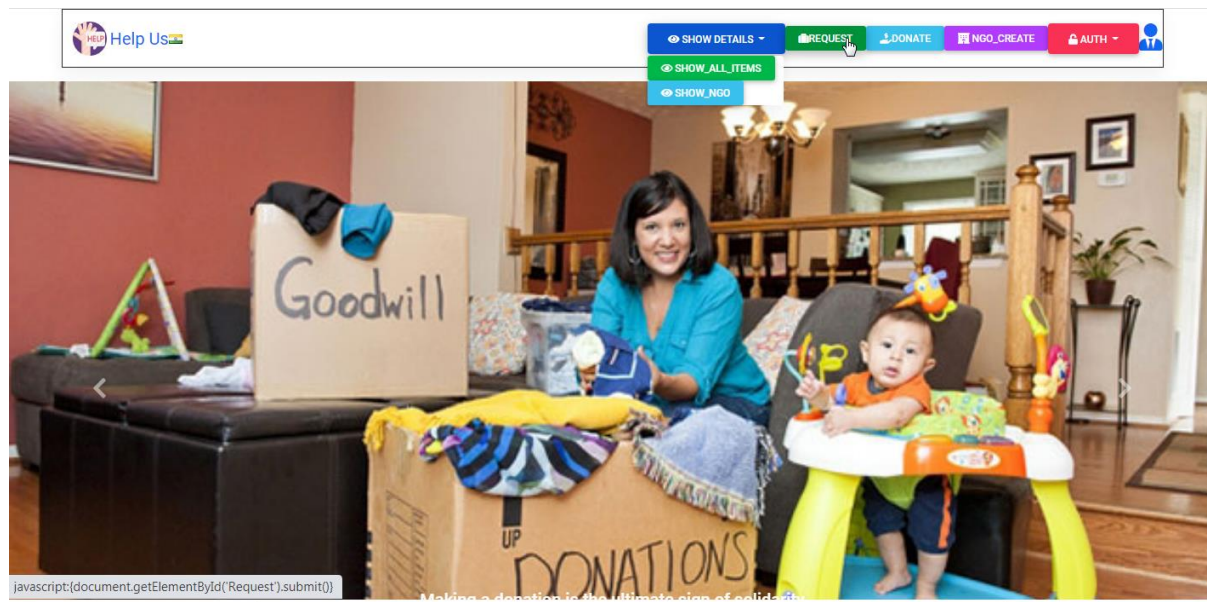
State

City

Add certificate
CERTIFICATE No file chosen

© Jagrit Acharya 2021

Form to create a new NGO account



click on request to post request



All fields are required

Create request

Request form



Create Your Profile

Form to create your profile

Django administration

WELCOME, DELL1. [VIEW SITE](#) / [CHANGE PASSWORD](#) / [LOG OUT](#)

[Home](#) > [Userprofile](#) > [Profiles](#) > [Profile object \(24\)](#)

Change profile

HISTORY

Name:

acharya

Phone:

12334432

State:

Arunachal Pradesh

City:

Changlang

Profile owner:

jagrit2

Delete

Save and add another

Save and continue editing

SAVE

User profile model

Django administration

WELCOME, DELL1. [VIEW SITE](#) / [CHANGE PASSWORD](#) / [LOG OUT](#)

[Home](#) > [Request](#) > [Needrequests](#) > [Needrequest object \(55\)](#)

Change needrequest

HISTORY

Name:

jagrit acharya

Phone:

12243342

Title:

shoes

Pub date:

Date: 2021-04-03 Today

Time: 15:48:46 Now

Note: You are 5.5 hours ahead of server time.

Body:

size 9

Needy:

jagrit2

Delete

Save and add another

Save and continue editing

SAVE

Request model

Django administration
WELCOME, DELL1. VIEW SITE / CHANGE PASSWORD / LOG OUT

Home > Ngo > Ngoprofiles > Ngoprofile object (21)

Change ngoprofile
HISTORY

Name: trust 1

Certificates:
Currently: images/id_card_6KaEU00.jpeg
Change: Choose File No file chosen

Phone: 8875091601

State: Arunachal Pradesh

City: Changlang

Ngo creator: jagrit2

Delete Save and add another Save and continue editing SAVE

Ngo profile model

Home > Donator > Products > Product object (21)

Change product
HISTORY

Title: shoes

Name: jagrit acharya

Phone: 12243342

Address: a-41 parshwanath colony

Pub date:
Date: 2021-04-03 Today
Time: 15:44:46 Now
Note: You are 5.5 hours ahead of server time.

Body: size 9

ItemImage:
Currently: images/shoes.png
Change: Choose File No file chosen

Govt id:
Currently: images/id_card_upJmZKz.jpeg
Change: Choose File No file chosen

Products Model

Django administration
WELCOME, DELL1. VIEW SITE / CHANGE PASSWORD / LOG OUT

Home > Authentication and Authorization > Users

Select user to change
ADD USER

Search

Action: 0 of 23 selected

<input type="checkbox"/>	USERNAME	EMAIL ADDRESS	FIRST NAME	LAST NAME	STAFF STATUS
<input type="checkbox"/>	a				

FILTER
By staff status
All
Yes
No
By superuser status

User model (login details)

Chapter 4: Conclusion and Future work

To conclude, Django and PostgreSQL work well in cohesion & they complement each other's functionalities perfectly as Django provides a concrete frontend & Backend of the website, PostgreSQL provides a reliable, secure and solid DBMS. The connectivity between these elements is fairly simple and can be established easily. DBMS provides a database service with a higher scope considering the scalability of the project since it can manage large data sets easily through its Relational Database Model(RDBMS) Structure.

Future work:

- We would be tracking location through the mobile device.
- Ngo or user login page will be different.
- Payment gateway integration will be also there.