

<b>S.NO</b>	<b>INDEX</b>	<b>Page No</b>
<b>1.</b>	<b>INTRODUCTION</b>	<b>3</b>
	1.1 About the Project	
	1.2 Company Profile	
<b>2.</b>	<b>SYSTEM STUDY</b>	<b>5</b>
	2.1. Existing System	
	2.2. Disadvantages of Existing System	
	2.3. Proposed System	
	2.4. Advantages of Proposed System	
	2.5. Problem Definition and Description	
<b>3.</b>	<b>SYSTEM ANALYSIS</b>	<b>7</b>
	3.1. Resources Required (HW & SW)	
	3.2. Feasibility Study	
	3.3. Use Case Diagram	
	3.4. Data Flow Diagram	
<b>4.</b>	<b>SYSTEM DESIGN</b>	<b>11</b>
	4.1. Architectural Design	
	4.2. I/O Form Design	
	4.3. Tables	
	4.4. Normalization	
<b>5.</b>	<b>CODING &amp; DEBUGGING</b>	<b>22</b>
	5.1. Functional Documentation	
	5.2. Special Features of Language / Utility	
	5.3. Pseudo code/ Algorithm	
<b>6.</b>	<b>TESTING</b>	<b>26</b>
	6.1. Types of Testing Done	
<b>7.</b>	<b>USER MANUAL</b>	<b>26</b>
	7.1. Hardware Requirements	
	7.2. Software Requirements	
	7.3. Installation Procedures	
<b>8.</b>	<b>CONCLUSION</b>	<b>28</b>
	8.1. Summary of the Project	

- 8.2. New Enhancements
- 8.3. Future Possibilities

## **BIBLIOGRAPHY**

29

Article Bibliography

Book Bibliography

Websites References

## 1. INTRODUCTION

### 1.1 About the Project

The project entitled as “**DONATION MANAGEMENT SYSTEM FOR DONATION COLLECTION**” and it is an python application which is designed for VENBA INFORMATION TECHNOLOGIES(P) LTD which is situated in gowrivakkam,chennai. The application will be more useful for the outsource for all the donor,volunter data are stored and maintained by the administrative members or employees. The purpose of this application is to make the administrative facilities to be more easy and efficient to donate money.The application is designed in order to give proper user interface to the end user. The module describes about the administrative details of the employees,donor,volunter. It has 4 modules which describes about the staff,volunter,donor and details about the projects.

The staff module has the process of entering the project details such as project name, start date, end state and details like needed amount etc., the details entered by the staff will be stored in database and will be used for further use. The volunter module will be view the project details module and donate money with separte module then added to project fund raising module.

The volunter or donor module has the process of donating money toto the project with details fo donor name,address,mail\_id and so on and then entered data will be used for future use. both volunter or donor module has the details of entering the some necessary information and the stored data will be used for many other modules if necessary. Finally application will give comfort to NGO especially for social welfare groups.

## 1.2 Company Profile

**Company Name** : Versions Software Solutions

**Address** : plot No:6,Rakshi Sai Building,  
Buhavenswari Nagar,  
Tamabram to Velachery Main Road,  
Sai Baba Kovil Street,  
Gowrivakkam,  
Chennai -600023  
Email:info@venbainfotech.com

**External Guide Name : VENBA KANNAN (HEAD)**

## **2. SYSTEM STUDY**

System Study is used to deliver information about the donors in which describes about both the personal and company details of each donors. The details have to be entered by the staffs and those details will be stored in database and will be used for further use. The application is designed mainly for the administrative members in order to maintain the details of donors.

### **2.1 EXISTING SYSTEM**

Existing System defines the system which is already exist anywhere and may have merits and demerits of the project. The existing project of this application is web application named “Web based System”. It is a web application in which the administrative details will be entered by the staffs. It is an only application where all the details of the donors,volunter will be entered and managed by the administrators. It may lead to a problem and it will be difficult for both staffs and donors and only support web based not developed by mobile based. Also the application may have bugs and errors.

### **2.2 DISADVANTAGES OF EXISTING SYSTEM**

- Human Error
- Connectivity issues
- Less responsive
- Inaccuracy
- Time consumption
- Slow processing Speed
- Data inconsistency

## **2.3 PROPOSED SYSTEM**

Proposed system refers to the system that enhances the existing features to the advanced level and makes the application more accuracy. Here the existing web application is designed in an python application as proposed system in order to give the clear user interface to the end user. The purpose of this application is to give an accuracy of data about the donors. The application will be more interactive. The application reduces the time and increases the performance while entering the data.

## **2.4 ADVANTAGES OF PROPOSED SYSTEM**

- Easy to access
- High performance
- Accuracy of data
- User friendly Interface
- Retrieval of information
- Accurate results

## **2.5 PROBLEM DEFINITION AND DESCRIPTION**

NGO staffs have to enter donor details in order to maintain the record of each and every staff. Not only the staff also the volunteer and donor have to maintain the details and the work is done by the administrative members. The entered details are stored in database and will be used for further use.

For those purpose the application has been designed by using the following modules

- Staff
- Donor
- Volunteer

## **3. SYSTEM ANALYSIS**

### **3.1 RESOURCES REQUIRED**

#### **3.1.1 Hardware Requirements**

Processor : ANY CORE PROCESSOR

RAM : 4 GB (Minimum)

Hard Disk : 320 GB (Minimum)

Other Devices : Android smart device (Smart phone, Tablet)

#### **3.1.2 Software Requirements**

Operating System : Windows 7 or Higher

Linux Ubuntu 16

#### **3.1.3 CLIENT SIDE DEVELOPMENT**

Pycharm version 18.1(python IDE)

Django version 1.2.11

Pip installer

#### **3.1.4 SERVER SIDE DEVELOPMENT**

Script : Python

Database : SQLITE3

Server : Apache http

## **3.2 FEASIBILITY STUDY**

Feasibility is a system proposal according to its work ability, impact on the organization, ability to meet user and effective use of resources.

Three types of feasibility,

1. Technical feasibility.
2. Operational feasibility.
3. Economic feasibility.

Feasibility analysis is necessary to determine whether the proposed system is feasible considering the technical, operational and economic factors. By having detailed feasibility study the management will have clear-cut view of the proposed system with benefit and drawbacks.

### **3.2.1 Technical Feasibility**

The proposed System is to be implemented with MySQL as a backend tool running under local server. Since the system is being developed in running in hand held devices, the system is easy-to-users. Thus the proposed system is technically feasible. The most important criteria for a system are that it must be technically feasible. The proposed system is going to be the part of much bigger system and hence its implementation are designed in such a way that it is going to be faster and efficient.

### **3.2.2 Operational Feasibility**

The user friendly interface, which makes all operation easy to use and no extra training, is needed in this regard, the user doesn't need extra thing and thus lots of time is saved. This makes the project operationally feasible.

### **3.2.3 Economic Feasibility**

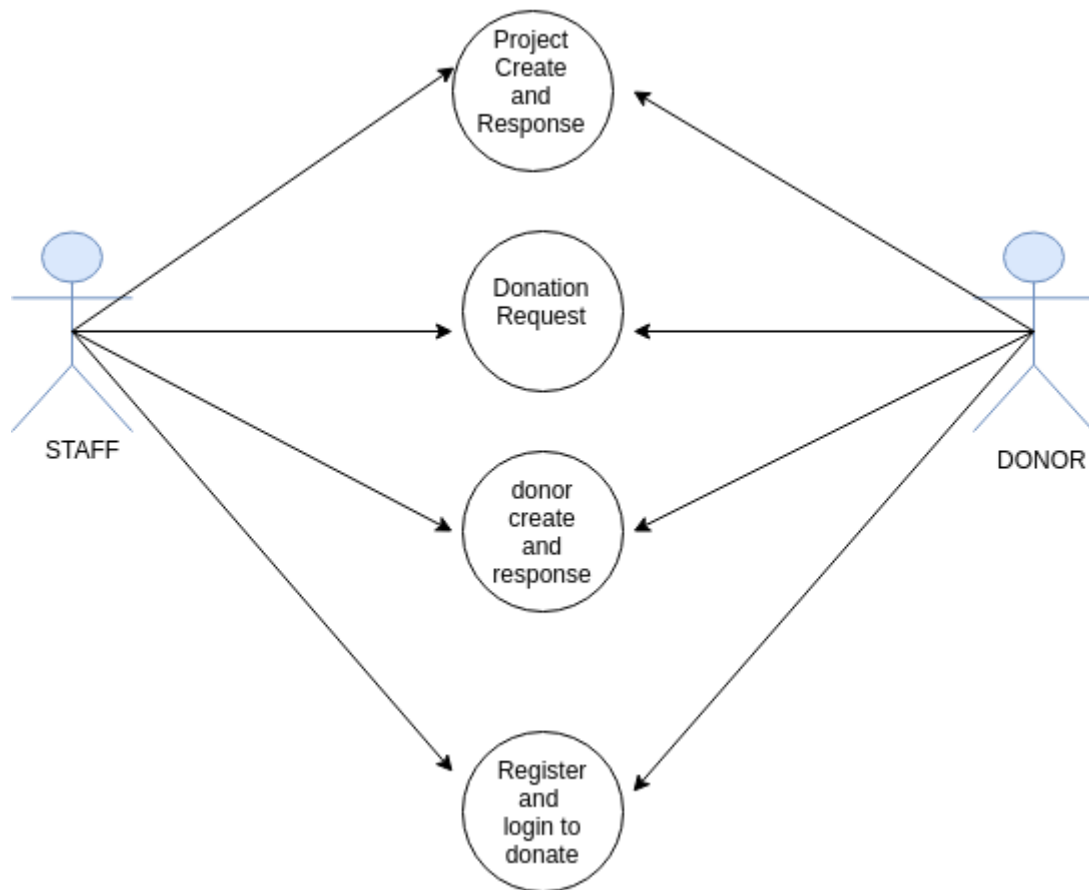
Since the proposed system deals with the mobiles and tablets, it is worth to purchase while needed. Thus the proposed system is economically feasible.



### 3.3 USE CASE DIAGRAM

The use case diagram at its simplest representation of a user's interaction with the system that shows the relationship between the user and the different use cases in which the user is involved. A use case diagram can identify the different types of users of a system and the different use cases and will often be accompanied by other types of diagrams as well.

Use case diagrams are used to gather the requirements of a system including internal and external influences. These requirements are mostly design requirements. So when a system is analysed to gather its functionalities use cases are prepared and actors are identified. Now when the initial task is complete use case diagrams are modelled to present the outside view.



**Fig:3.1 Use Case Diagram**

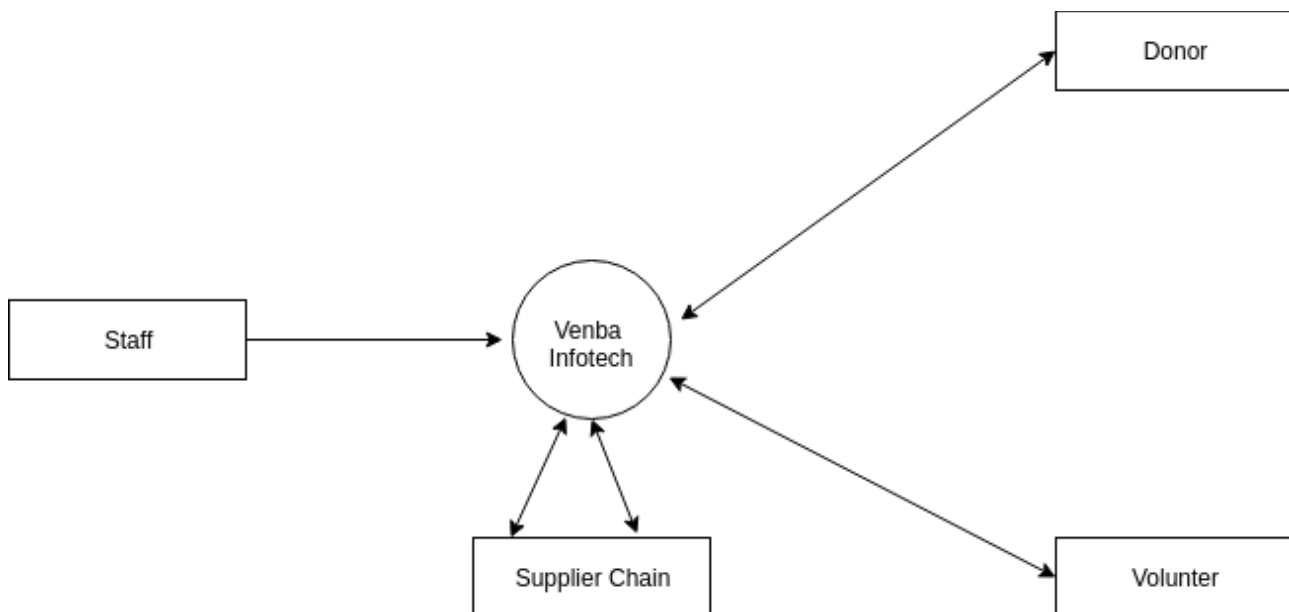
### 3.4 DATA FLOW DIAGRAM

A Data Flow Diagram (DFD) is a graphical representation of the "flow" of data through an information system, modelling its process aspects. Often they are a preliminary step used to create an overview of the system which can later be elaborated. DFDs can also be used for the visualization of data processing (structured design).

A DFD shows what kinds of information will be input to and output from the system, where the data will come from and go to, and where the data will be stored. It does not show information about the timing of processes, or information about whether processes will operate in sequence or in parallel.

Data flow diagrams can be used to provide the end user with a physical idea of where the data they input ultimately has an effect upon the structure of the whole system from order to dispatch to report. How any system is developed can be determined through a data flow diagram model.

#### DATA FLOW DIAGRAM CONTEXT LEVEL



**Fig.3.2 Context Level**

## 4.SYSTEM DESIGN

### 4.1 ARCHITECTURAL DESIGN

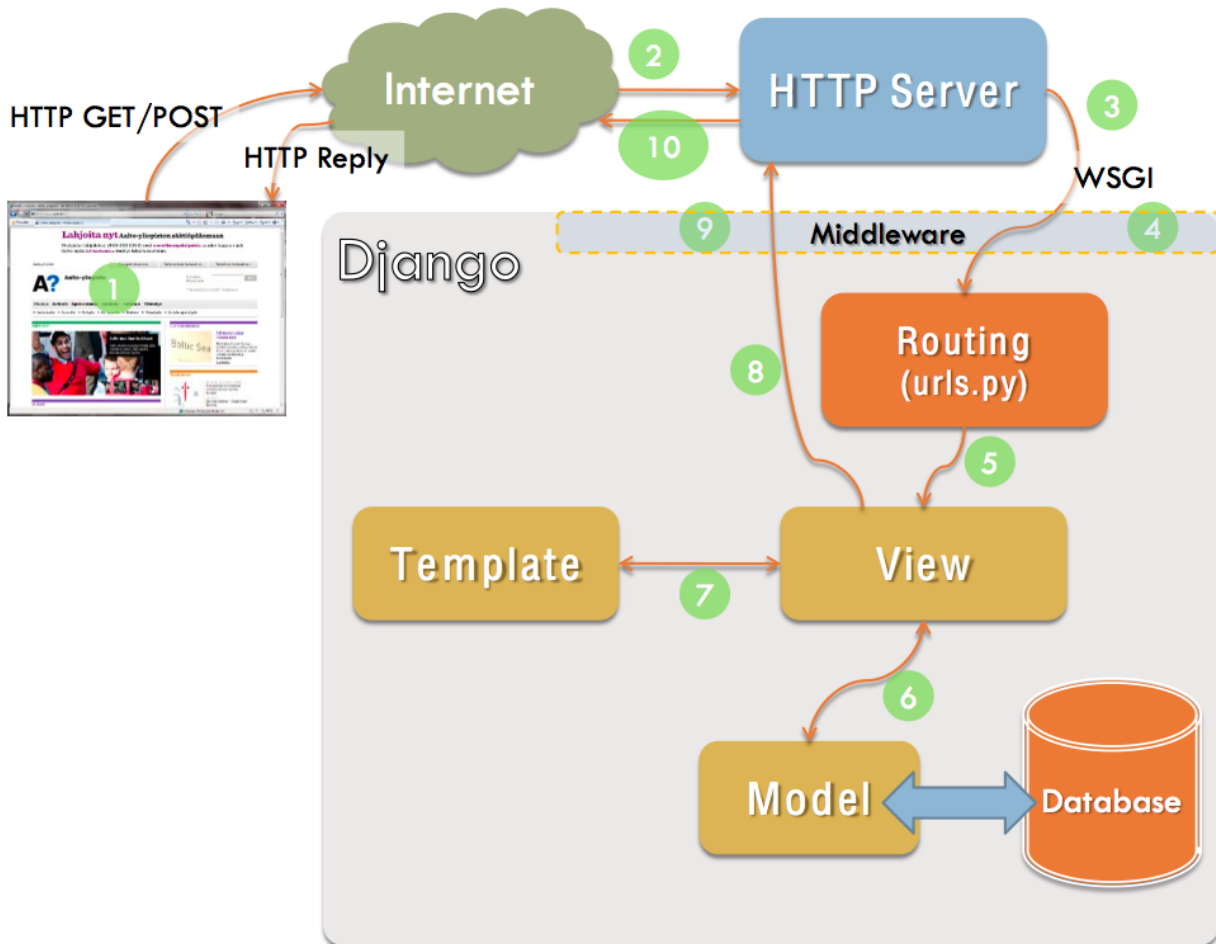


Fig 4.1:Architectural Design

## **4.2 INPUT/OUTPUT DESIGN**

### **4.2.1 Input Design**

The input data are collected into groups of similar data. While entering the input data the operators must know the space allocated for each field, the data types in which the data fields are entered. The input design is the link that ties information system into the world of its users. It consists of developing specification and procedure for data preparation. The inaccurate data won't be accepted by this system. For example entering text in place of telephone number field is inappropriate. So this type of wrong input will not be allowed. In this manner the input design allows only valid and required relevant data to be stored.

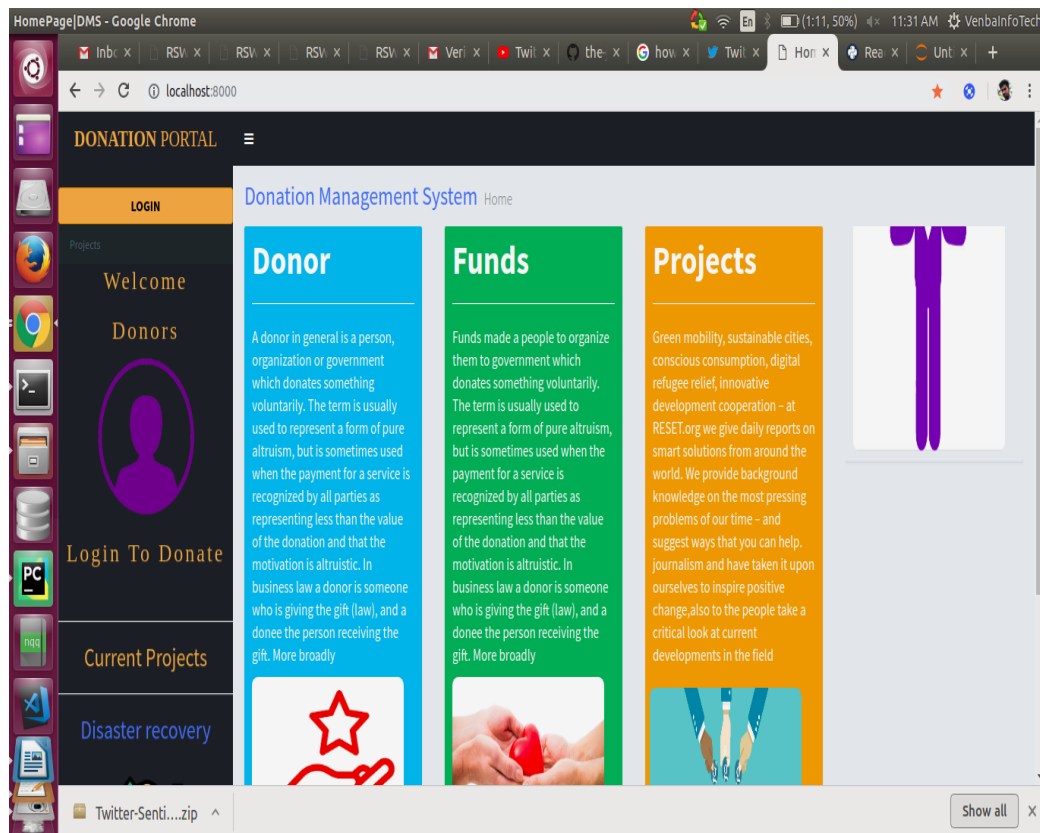


Fig:4.2 Home Page

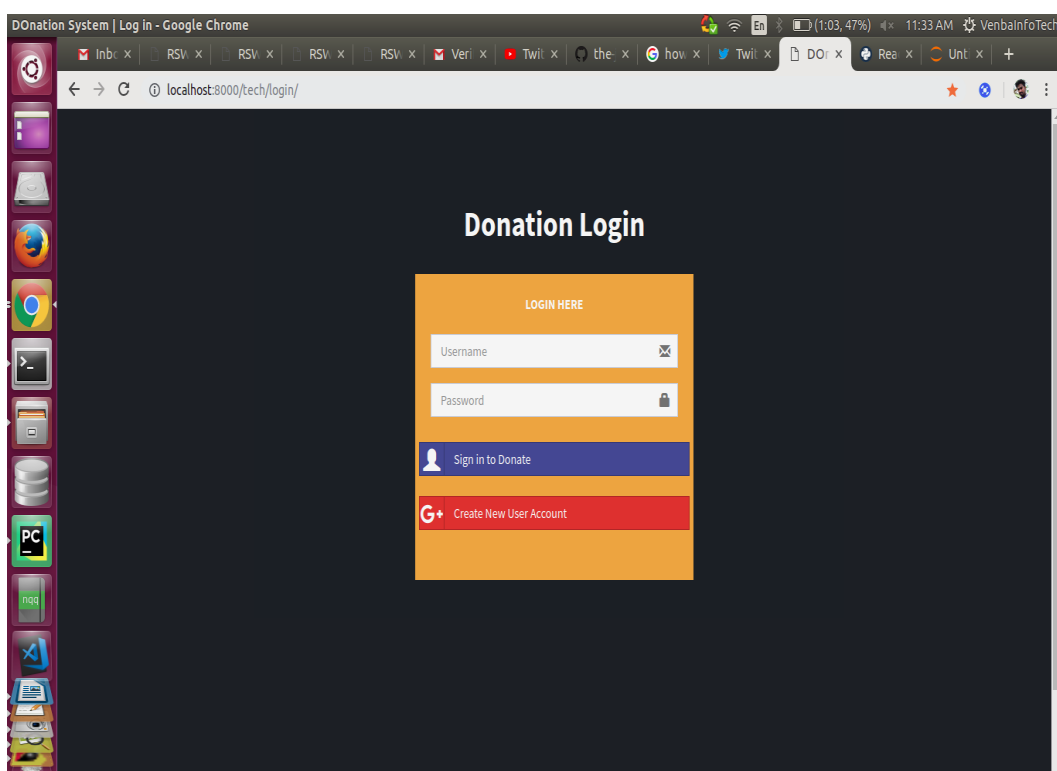


Fig:4.3 :Donation Login Page

Registration[DMS - Google Chrome]

localhost:8000/tech/login/register\_savefile\_retrieve

## Create an New User

Register a new membership

joemicks

\*\*\*\*\*

\*\*\*\*\*

Select Type:

volunteer

volunteer

employee

donors

☐ I agree to the terms

Register

Fig 4.4 Registration Page

Donor System

Welcome

LogOut

### PROJECTS STATUS VIEW

PROJECTS

Show 10 entries

Search:

Project Name	Category	Start Date	Estimated Amount	Mobile Number	Action	Invoice Generate
Disaster recovery	Natural Issues	21/02/2018	4300	7708567876	View Inprogress	Invoice
Water Management	water	21/03/2018	5000	7878676565	View Inprogress	Invoice
xxxxxx	asasas	23/02/2032	200000	8899778877	View Inprogress	Invoice

Showing 1 to 3 of 3 entries

Previous 1 Next

Fig 4.5 :Project View Page

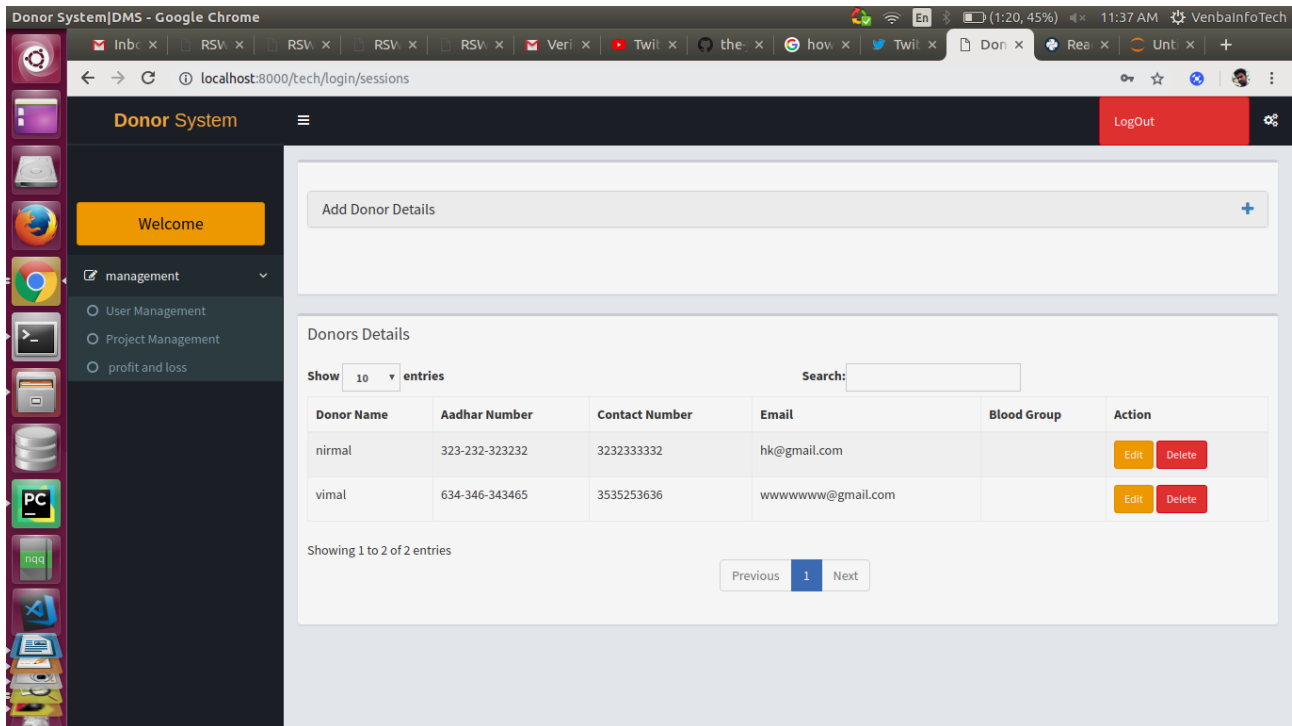


Fig 4.6 : Donor View Page

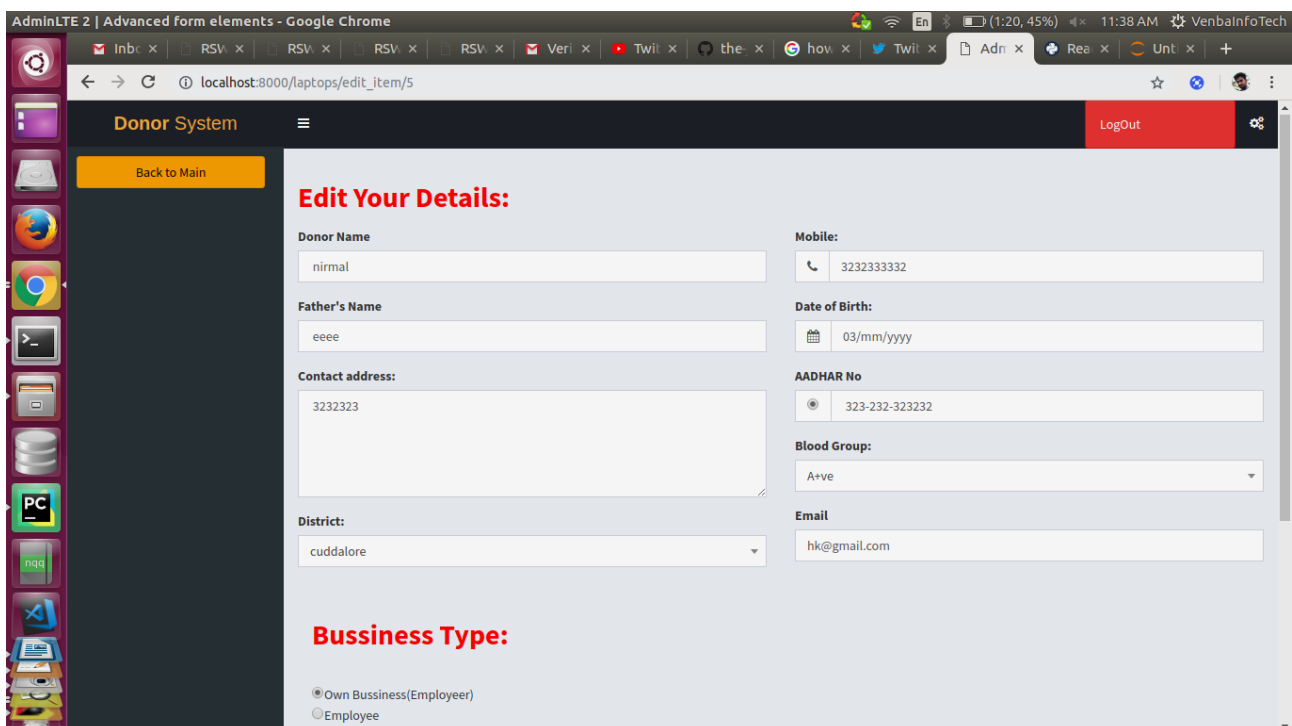


Fig 4.7 Edit Donor Details Page

**Donor System**

[Back to Main](#)

**PROJECT DETAILS**

Title: Disaster recovery

Category: Natural Issues

Start Date: 21/02/2018

End Date:

Fund Amount: 4300

Contact phone: 7708567876

Description: needy to supply food faster

Status: (select project status)

☒ Completed

☐ Inprogress

Upload image: Choose file No file chosen

Email: joe@gmail.com

[Update project Details](#)

Fig 4.8: Edit project Details

**Donor System**

[Back to projects](#)

**DONATION DETAILS**

[Donate Money](#)

category: Natural Issues

Start Date: 21/02/2018

End Date: 22/02/2022

Fund: 4300

email Id: joe@gmail.com

Contact Number: 7708567876

see the description of the project

project summary

needy to supply food faster

[Donation Details](#)

[Income and Expenditure](#)



Fig 4.9 Donation \_view page

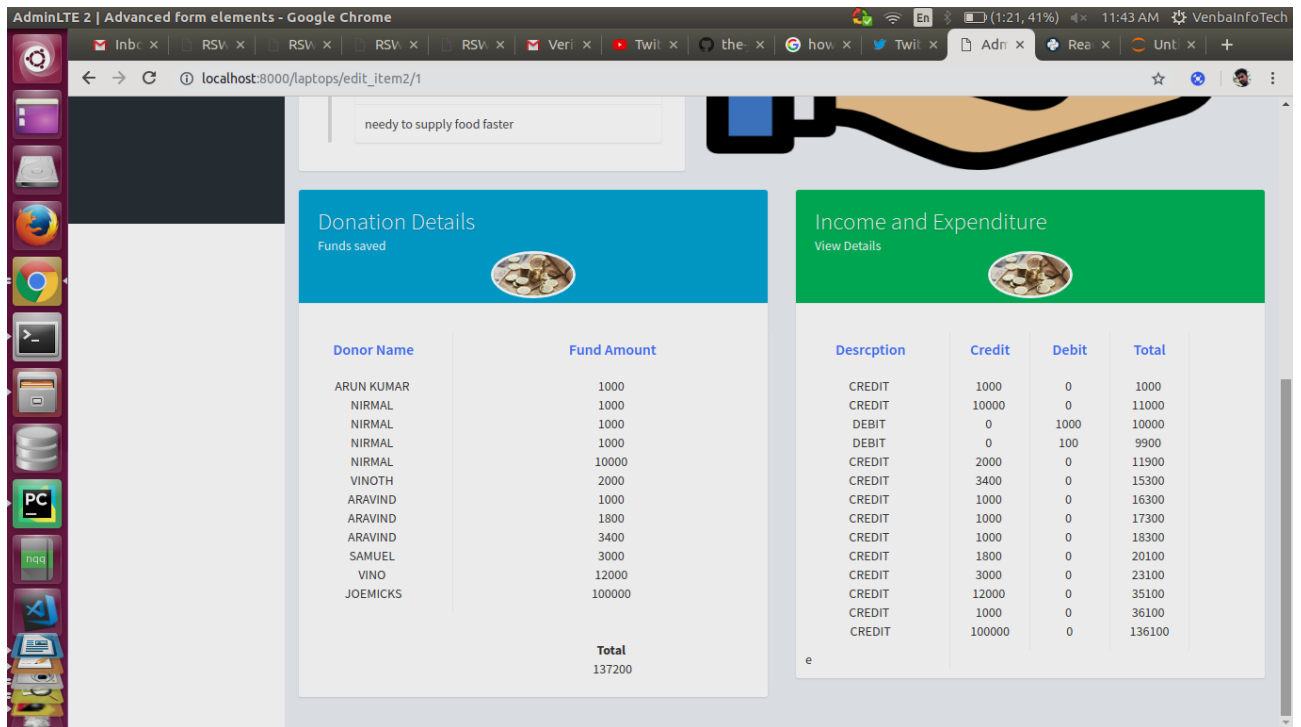


Fig :4.10 Donation view page 2

The screenshot displays the 'Donation view page 2' in a web browser. The page has a dark sidebar with various icons and a main content area. At the top of the main area, there is a 'Donor System' header with a 'Back to projects' button. Below this, the page is titled 'DONATION PORTAL' in red. The form contains the following fields:

- Fund Amount:** A text input field.
- Transaction type:** Radio buttons for 'Cash', 'Online' (selected), and 'Chequee'.
- Bank Name:** A text input field.
- Bank Account Number:** A text input field.
- Transaction Date:** A date picker showing '03/20/2019'.
- Pay Amount:** A blue button.

At the top right of the main area, there is a 'LogOut' button.

Fig 4.11 Donation Portal

**Donor System**

[Back to projects](#)

**Invoice Report** [Print Invoice](#)

**Invoice:**  
Generate invoice form and print for the further progression

**Bill Details** Date: 03/20/2019

<b>From</b> Donor Organisation Email: donortn19@gmail.com	<b>To</b> aravind Phone: 7708567876 Email: joe@gmail.com	<b>Invoice</b> Order ID:1
---	---	------------------------------

Project Name	Description	Sub-total
Disaster recovery	needy to supply food faster	6200

**Section 80G:** Invoice Date: 03/20/2019

Contributions made to certain relief funds and charitable institutions can be claimed as a deduction under Section 80G of the Income Tax Act. All donations, however, are not eligible for deductions under section 80G. Only donations made to prescribed funds qualify as a deduction.

The deduction is allowed to all types of taxpayers: This deduction can be

<b>Total Summary:</b>	Rs.6200
<b>GST Number</b>	81253036446
<b>Pancard Number:</b>	190873636262

Fig 4.12 : Invoice Page

**Print**  
Total: 1 page

[Cancel](#) [Save](#)

**Destination** [Save as PDF](#)  
[Change...](#)

**Pages** ☒ All  
☐ e.g. 1-5, 8, 11-13

**Layout** [Portrait](#)

[More settings](#)

[Print using system dialogue... \(Ctrl+Shift+P\)](#)

**Invoice Report for Donation**

**Bill Details** 20/03/2019

<b>From</b> Donor Organisation Email: donortn19@gmail.com	<b>To</b> aravind Phone: 7708567876 Email: joe@gmail.com	<b>Invoice</b> Order ID:1
---	---	------------------------------

Project Name	Description	Sub-total
Disaster recovery	needy to supply food faster	6200

**Section 80G:** Invoice Summary

Contributions made to certain relief funds and charitable institutions can be claimed as a deduction under Section 80G of the Income Tax Act. All donations, however, are not eligible for deductions under section 80G. Only donations made to prescribed funds qualify as a deduction.

The deduction is allowed to all types of taxpayers: This deduction can be claimed by any taxpayer - individuals, company, firm or any other person.

**Mode of Payment:** This deduction can only be claimed when the contribution has been made via a cheque or a draft or in cash. But the deduction is not allowed for donations made in cash exceeding Rs 10,000. In-kind contributions such as food material, clothes, medicines etc. do not qualify for deduction under section 80G.

**From Financial Year 2017-18 onwards:** Any donations made in cash exceeding Rs 2,000 will not be allowed as deduction. The donations above Rs 2,000 should be made in any mode other than cash to qualify as a deduction under section 80G.

**How to claim the deduction**

Name of the Donee  
PAN of the Donee  
Address of the Donee  
Amount of Contribution

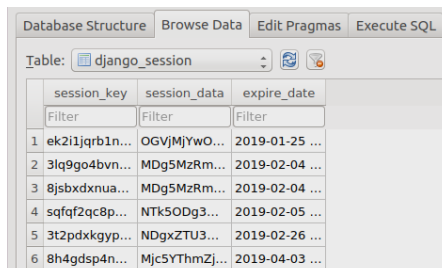
[http://localhost:8000/invoice/print/1](#) 1/1

Fig 4.13 :Generate Invoice page

## 4.3 TABLE DESIGN

A table is a collection of related data held in a structured format within a database. It consists of columns, and rows. In relational databases and flat file databases, a table is a set of data elements using a model of vertical columns and horizontal rows, the cell being the unit where a row and column intersect. A table has a specified number of columns, but can have any number of rows. Each row is identified by one or more values appearing in a particular column subset. The columns subset which uniquely identifies a row is called the primary key

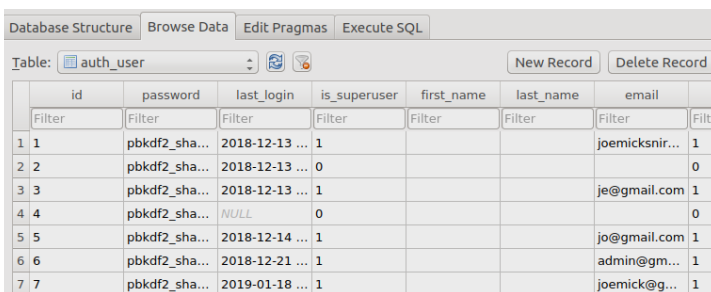
**TABLE 4.3.1 :django\_session**



The screenshot shows a database browser interface with tabs for Database Structure, Browse Data, Edit Pragmas, and Execute SQL. The 'Browse Data' tab is active, displaying the 'django\_session' table. The table has three columns: session\_key, session\_data, and expire\_date. Below the column headers are filter boxes. The data is presented in a list of rows, each with an index number, a truncated session\_key, a truncated session\_data, and a truncated expire\_date.

	session_key	session_data	expire_date
1	ek2iljrb1n...	OGVjMjYwO...	2019-01-25 ...
2	3lq9go4bvn...	MDg5MzRm...	2019-02-04 ...
3	8jsbxdxnua...	MDg5MzRm...	2019-02-04 ...
4	sqfqf2qc8p...	NTk5ODg3...	2019-02-05 ...
5	3t2pdxkgyp...	NDgxZTU3...	2019-02-26 ...
6	8h4gdsp4n...	Mjc5YThmZj...	2019-04-03 ...

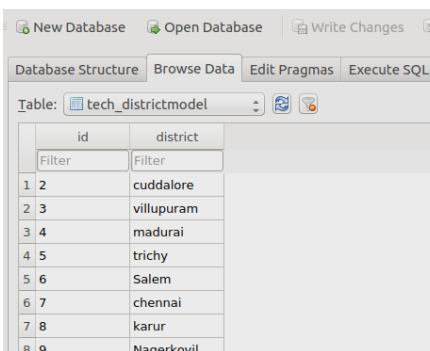
**TABLE 4.3.2 :auth\_user**



The screenshot shows a database browser interface with tabs for Database Structure, Browse Data, Edit Pragmas, and Execute SQL. The 'Browse Data' tab is active, displaying the 'auth\_user' table. The table has seven columns: id, password, last\_login, is\_superuser, first\_name, last\_name, and email. Below the column headers are filter boxes. The data is presented in a list of rows, each with an index number, an id, a truncated password, a truncated last\_login, a boolean is\_superuser, a first\_name, a last\_name, and an email address.

	id	password	last_login	is_superuser	first_name	last_name	email
1	1	pbkdf2_sha...	2018-12-13 ...	1			joemicksnir... 1
2	2	pbkdf2_sha...	2018-12-13 ...	0			0
3	3	pbkdf2_sha...	2018-12-13 ...	1			je@gmail.com 1
4	4	pbkdf2_sha...	NULL	0			0
5	5	pbkdf2_sha...	2018-12-14 ...	1			jo@gmail.com 1
6	6	pbkdf2_sha...	2018-12-21 ...	1			admin@gm... 1
7	7	pbkdf2_sha...	2019-01-18 ...	1			joemick@g... 1

**TABLE 4.3.3 :tech\_districtmodel**



The screenshot shows a database browser interface with tabs for Database Structure, Browse Data, Edit Pragmas, and Execute SQL. The 'Browse Data' tab is active, displaying the 'tech\_districtmodel' table. The table has two columns: id and district. Below the column headers are filter boxes. The data is presented in a list of rows, each with an index number, an id, and a district name.

	id	district
1	2	cuddalore
2	3	villupuram
3	4	madurai
4	5	trichy
5	6	Salem
6	7	chennai
7	8	karur
8	9	Nagerkovil

**TABLE 4.3.4 :tech\_dontation\_portal**

	id	fund_amount	bank_name	ink_acc_numb	requeue_numb	ansaction_dat	aign_donation	
1	116	1000	HDFC	452632546...		03/12/2019	1	56
2	117	10000	23000	12374843		03/12/2019	1	56
3	118	2000				03/12/2019	1	57
4	119	3400				03/12/2019	1	58
5	120	2000				03/12/2019	2	58
6	121	2000				03/12/2019	2	58
7	122	1000	CDSD	2737474757		03/12/2019	1	56
8	123	1000	SAD	SADds		03/12/2019	1	56
9	124	234000				03/13/2019	2	56
10	125	1200				03/13/2019	4	58
11	126	1000	HDFC	212121212...		03/13/2019	1	55
12	127	1800				03/13/2019	1	58
13	128	3000	HDSA	323232378...		03/13/2019	1	60
14	129	10000	aasaas	21121232323		03/18/2019	4	56
15	130	12000				03/18/2019	1	62
16	131	1000				03/18/2019	1	58
17	132	100000				03/18/2019	1	63
18	133	10				03/18/2019	2	63

**TABLE 4.3.5 :tech\_profit\_account**

	id	/pe_transactio	bill_name	amount	project_id	date	xexpense_name
1	46	CREDIT		1000	1	03/12/2019	
2	47	CREDIT		10000	1	03/12/2019	
3	48	Debit		1000	1	21/02/2012	expendtinue
4	49	Debit		100	1	22/02/2021	petrol
5	50	CREDIT		2000	1	03/12/2019	
6	51	CREDIT		3400	1	03/12/2019	
7	52	CREDIT		2000	2	03/12/2019	
8	53	CREDIT		2000	2	03/12/2019	
9	54	Debit		1000	2		deer
10	55	Credit			3		aaaa
11	56	CREDIT		1000	1	03/12/2019	
12	57	CREDIT		1000	1	03/12/2019	
13	58	CREDIT		234000	2	03/13/2019	
14	59	CREDIT		1200	4	03/13/2019	
15	60	CREDIT		1000	1	03/13/2019	
16	61	CREDIT		1800	1	03/13/2019	
17	62	CREDIT		3000	1	03/13/2019	
18	63	Debit	logo.jpg	10000	4	21/02/2012	hhh
19	64	CREDIT		10000	4	03/18/2019	

**TABLE 4.3.7 :userprofilemodel**

Database StructureBrowse DataEdit PragmasExecute SQL

Table: tech\_userprofilemodelNew RecordDelete Record

	id	donor	father	address	district	mobile	dob	
	Filter	Filter	Filter	Filter	Filter	Filter	Filter	Filter
1	3	vimal	ssss	deeteeeve...	5	3535253636	21/02/2012	634
2	5	nirmal	eeee	3232323	2	3232333332	3d/mm/yyyy	323

1 - 2 of 2Go to: 1

TABLE 4.3.8 :table structure

Database StructureBrowse DataEdit PragmasExecute SQL

Create TableCreate IndexModify TableDelete Table

Name	Type	Schema
Tables (23)		
auth_group	CREATE TABLE "auth_group" ("id" integer NOT NULL	
auth_group_permissions	CREATE TABLE "auth_group_permissions" ("id" inte	
auth_permission	CREATE TABLE "auth_permission" ("id" integer NO	
auth_user	CREATE TABLE "auth_user" ("id" integer NOT NULL	
auth_user_groups	CREATE TABLE "auth_user_groups" ("id" integer N	
auth_user_user_permissions	CREATE TABLE "auth_user_user_permissions" ("id"	
django_admin_log	CREATE TABLE "django_admin_log" ("id" integer N	
django_content_type	CREATE TABLE "django_content_type" ("id" integer	
django_migrations	CREATE TABLE "django_migrations" ("id" integer N	
django_session	CREATE TABLE "django_session" ("session_key" var	
sqlite_sequence	CREATE TABLE sqlite_sequence(name,seq)	
tech_bloodmodel	CREATE TABLE "tech_bloodmodel" ("id" integer NC	
tech_bussiness	CREATE TABLE "tech_bussiness" ("id" integer NOT	
tech_districtmodel	CREATE TABLE "tech_districtmodel" ("id" integer N	
tech_donation	CREATE TABLE "tech_donation" ("id" integer NOT	
tech_dontation_portal	CREATE TABLE "tech_dontation_portal" ("id" integ	
tech_logindatas	CREATE TABLE "tech_logindatas" ("id" integer NOT	
tech_master	CREATE TABLE "tech_master" ("id" integer NOT NU	
tech_profit_account	CREATE TABLE "tech_profit_account" ("id" integer	
tech_profit_loss	CREATE TABLE "tech_profit_loss" ("id" integer NO	
tech_project_employee	CREATE TABLE "tech_project_employee" ("id" inte	
tech_registrationdatas	CREATE TABLE "tech_registrationdatas" ("id" integ	
tech_userprofilemodel	CREATE TABLE "tech_userprofilemodel" ("id" integ	
Indices (15)		
auth_group_permissions_group...	CREATE INDEX "auth_group_permissions_group_id	
auth_group_permissions_group...	CREATE UNIQUE INDEX "auth_group_permissions_	
auth_group_permissions_permi...	CREATE INDEX "auth_group_permissions_permis	
auth_permission_content_type...	CREATE INDEX "auth_permission_content_type_id	

## 4.4. Normalization

Normalization is the process of efficiently organization data in a database. There are two goals of the normalization process: eliminating redundant data (for example, storing related data in more than one table) and ensuring data dependencies make sense (only storing related data in table). If the database design is not perfect, it may contain anomalies, which are like a bad dream for any database administrator. Managing a database with anomalies is next to impossible.

**Update anomalies** – If data items are scattered and are not linked to each other properly, then it could lead to strange situations.

For example, when we try to update one data item having its copies scattered over several places, a few instances get updated properly while a few others are left with old values. Such instances leave the database in an inconsistent state. Insert anomalies – we tried to insert data in a record that does not exist at all. Normalization is a method to remove all these anomalies and bring the database to a consistent state.

### **First Normal Form (1NF)**

- Eliminate duplicative columns from the same table.

### **Second Normal Form (2NF)**

- Meet all the requirement of the first normal form.
- Each group of related data and identify each row with a unique column or set of columns (the primary key). Diagram The class diagram is a static diagram. It represents the static view of an application.

## 5. CODING AND DEBUGGING

### 5.1 Functional Documentation

The system maintains all the administrative activities of all the donors, volunteers and staff. The entered information will be stored in database and will be used for further use.

#### 5.1.1 Staff

- New entry
- Report of staff details
  - Code wise
  - Name wise
  - project wise
  - Location wise
- Updating the details
- Password Validation

The module describes the details of all the donors. Staff members have to enter their own details and this will be similar to registration form. The details will be used for future use.

### 5.2 Special Features of Language / Utility

#### PYTHON

Python is an open source programming language . Python is a server side, user interactive, programming language, works nearly in on all platforms like Unix,Linux,Windows It is a general purpose scripting language. It can be embedded into html. Python is a server side scripting language that is embedded in HTML. It is used to manage dynamic content, databases, session tracking, even build entire e-commerce sites. It is integrated with a number of popular databases, including MySQL, PostgreSQL, Oracle, Sybase, Informix, and Microsoft SQL Server. The MySQL server, once started, executes even very complex queries with huge result sets in record-setting time. Also contains many server interfaces. Open source is one of the best specifications of Python.

Especially Web Based Developement it could be use for the django,flask framework more secured sites with cross platform accesibilty conjunction has made.python similary used for different fields of studies like machine learning,deep learning,data science Now a days python play an vital role in the computer fields.

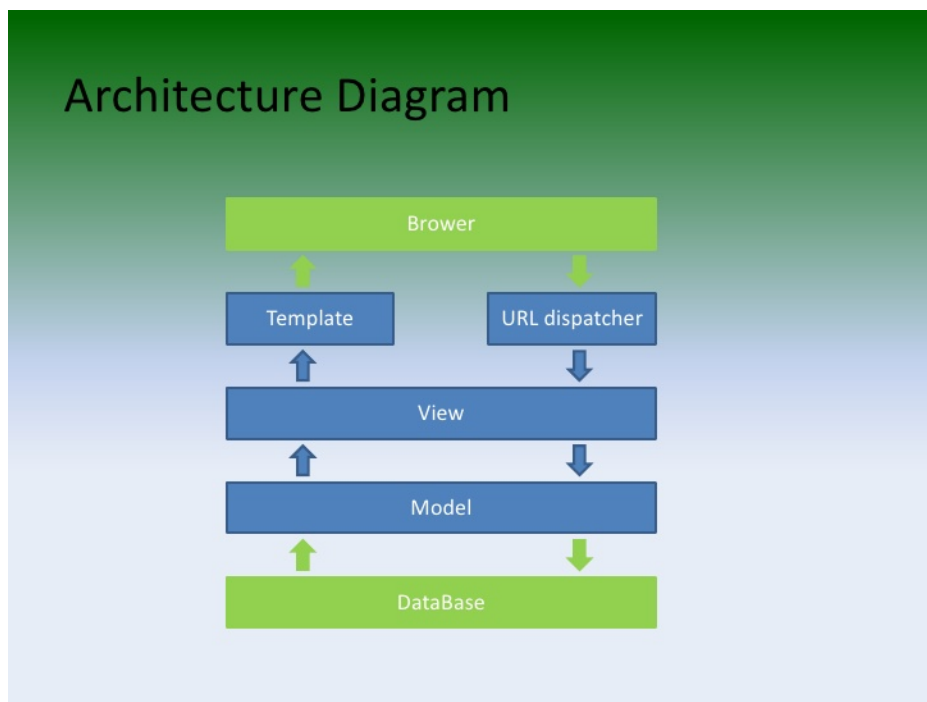
#### DJANGO

Django is a free and open source web application framework written in Python. A framework is nothing more than a collection of modules that make development easier. They are grouped

together, and allow you to create applications or websites from an existing source, instead of from scratch.

This is how websites - even simple ones designed by a single person - can still include advanced functionality like authentication support, management and admin panels, contact forms, comment boxes, file upload support, and more. In other words, if you were creating a website from scratch you would need to develop these components yourself. By using a framework instead, these components are already built, you just need to configure them properly to match your site.

### Application Framework:



## 5.3 Pseudo code / Algorithm

### General

Step 1: Start

Step 2: Click the login button in the home page.

Step 3: insert data in the required fields in the form "login page".



Step 4: login redirect based on id user can redirect .

Step 5: Also register a usertype can access the module differently.

Step 6: End

### **Donor/Volunter**

Step 1: Start

Step 2: Click the view button in the page of redirection.

Step 3: view data in the projects also to donate.

Step 4: Access prohibited for create project/donor .

Step 5: Also generate invoice on a button on viewpage.

Step 6: End

### **Staff**

Step 1: Start

Step 2: Click the Menubar in the page of redirection.

Step 3: create or manipulate data in the projects or donors details.

Step 4: Access project/donor .

Step 5: Also remove any details instantly.

Step 6: End

## **6. TESTING**

### **6.1 DEFINITION**

The importance of software testing and its impact on software cannot be underestimated. Software testing is a fundamental component of software quality assurance and represents a review of specification, design and coding. The greater visibility of software systems and the cost associated with failure are motivating factors for planning, through testing.

#### **6.1.1. Types of Testing Done**

The following are the different types have been carried out:

##### **Integration Testing**

Integration testing is the phase in software testing in which individual software modules are combined and tested as a group. It occurs after unit testing and before system testing. Integration testing takes as its input modules that have been unit tested, groups them in larger aggregates, applies tests defined in an integration test plan to those aggregates and delivers as its output the integrated system ready for system testing.

##### **Validation Testing**

Validation Testing can be defined in many ways, but a simple definition is that validation succeeds when the software functions in a manner that can reasonably expected by a customer. After validation test has been conducted, one of the following two possible conditions exists. The functions or performance characteristics confirm to specification and are accepted.

##### **User Acceptance Testing**

User acceptance of a system is a key factor of any system. The system under consideration is tested for the acceptance by constantly keeping in touch with the prospective system users at the same time of developing and marketing changes whenever required.

## **7. USER MANUAL**

### **7.1 Hardware Requirements**

Processor : ANY CORE PROCESSOR

RAM : 4 GB (Minimum)

Hard Disk : 320 GB (Minimum)

Other Devices : Android smart device (Smart phone, Tablet)

## 7.2 Software Requirements

Operating System : Windows 7 or Higher 7.3.

## 7.3 Installation Procedures

### Install PyCharm

Setting up Pycharm IDE takes just a few clicks. (You should have already downloaded Pycharm) To install Pycharm on Windows, Linux (use commands) proceed as follows:

#### Step 0: Pre-Installation Check List

1. Before that install python 2.7 or above version on the system
2. install python package Manager (pip installer)
3. install Django 1.0 or more on a system for the framework to access libraries
4. After the completion of the installments check it correctly installed or not

## **8. CONCLUSION**

### **8.1 Summary of the Project**

The overall application demonstrates the donation details of all the staff, volunteers, donors who are all interrelated with this project. It is necessary to make entry for each and every one to place orders from the company or to supply the goods. The administrator maintains all the details of the donation and donors details. The process of insertion and Updation will be made through this application. The details entered by the user will be stored in database and will be used for further use.

### **8.2 New Enhancement**

The system is enhanced from web application to python application with responsive support for mobile, tab and it has features of making entry and updating their own profile. By selecting the staff name and staff no the staff can update their details.

### **8.3 Future Possibilities**

In future, the application will be designed in android and also the application will be further developed using new technology called Data Science(Data Analysis) and will be great impact for the company and also to the users.

## BIBLIOGRAPHY

### Web References

<https://docs.djangoproject.com/en/2.1/>  
<https://docs.djangoproject.com/en/dev/releases/2.2/>

### References

1. Dave Smith and Jeff Friesen, “Light Wiegth Django”, Rakmo Press (P) Ltd, New Delhi, 2011. Computer Vision: Algorithms and Applications By Richard Szeliski.
2. Abraham Silberschatz, “ Database Systems”, McGraw Hill International, 1997. 3. Roger S Pressman, “Software Engineering”, McGraw Hill, International 6th Edn, New York, 2006.