

WESHARE
(SOCIAL SERVICE AND PUBLIC AWARENESS-SSPA)

PROJECT REPORT

Submitted by

RANIMOL ALEX

Reg.No.: SJC17MCA-D034

to

APJ Abdul Kalam Technological University

In partial fulfillment of award of the degree of

MASTER OF COMPUTER APPLICATION



Department of Computer Applications

ST. JOSEPH'S COLLEGE OF ENGINEERING & TECHNOLOGY, PALAI
Choondacherry P.O,Bharananganam-Pravithanam Road,Palai,Kottayam-689 579

JUNE 2019

D E C L A R A T I O N

I hereby declare that the project report "**WeShare (Social Service & Public Awareness-SSPA)**", submitted for partial fulfillment of the requirements for the award of the degree of Master of Computer Applications of the APJ Abdul Kalam Technological University, Kerala is a bonafide work done by me under supervision of **Mrs. Liz George**, Asst. Professor, MCA Dept. This submission represents my ideas in my own words and where ideas or words of others have been included, I also declare that I have adhered to ethics of academic honesty and integrity and have not misrepresented or fabricated any data or idea or fact or source in my submission. I understand that any violation of the above will be a cause for disciplinary action by the institute and/or the University and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been obtained. This report has not been previously formed the basis for the award of any degree, diploma or similar title of any other University.

RANIMOL ALEX

Reg. No:SJC17MCA-D034

**ST. JOSEPH'S COLLEGE OF ENGINEERING AND
TECHNOLOGY,PALAI**



C E R T I F I C A T E

This is to certify that, the project report titled **WESHARE (SOCIAL SERVICE & PUBLIC AWARENESS-SSPA)** is a bonafide record of the **Main Project** presented by **RANIMOL ALEX (Reg.No .SJC17MCA-D034)**, Fourth Semester MCA(Lateral Entry) student, under our guidance and supervision, in partial fulfillment of the requirements for the award of the degree, **MCA** of **APJ Abdul Kalam Technological University, Thiruvananthapuram.**

Mrs. Liz George
(Guide)

Mrs. Soumya George
(Project Coordinator)

Dr. Jainendrakumar T.D
(Head of the Dept.)

Viva-voce held on:

External Examiner 1 :

External Examiner 2 :

Acknowledgement

Behind this successful venture is the blessings and guidance of many people. First and the foremost, I thank the **God Almighty**, who gave me the inner strength, resource and ability to complete the work successfully, without which all my efforts would have been in vain.

I would like to convey my special gratitude to **Dr. J. David**, Principal, SJCET, Palai, for the moral support he provided.

I express my thankfulness to **Dr. Jainendrakumar TD**, Head of MCA Department, for his co-operation and valuable suggestions. Also I express my heartfelt thanks to my project co-ordinators **Mrs. Rinu Mathew** and **Mrs. Soumya George** for their helpful feedback and timely assistance.

I would like to thank sincerely my guide **Mrs. Liz George**, Assistant Professor, MCA Department, for her invaluable guidance, constant assistance and support in all stages of my work.

I also thank my all teachers and faculties who helped me by proper guidance and advices. I also extend my thanks to all my friends who helped me by giving motivation. Their smallest piece of advice was really valuable.

Once again I convey my gratitude to all those persons who had directly or indirectly influenced my work as a whole.

RANIMOL ALEX

Abstract

Social services are a range of public services provided by the different organizations. Social services include the benefits and facilities such as education, food subsidies, health care, police, fire service, job training and subsidized housing, adoption, community management, policy research, and lobbying. One of the most critical standard in running a charity organization is raising the required funds, which are practically necessary in conducting the associated social service. The major hurdle that is practically executed in running a charity system is identifying the donors who have been regularly funding charity organization for different purposes. Whenever the request for a new fund arises the authorities within the system have to make extensive search for finding the right donors for that specific charity request.

Public awareness is the public's level of understanding about the importance and implications of socially relevant topics. The entire system is divided into 4 major modules:

- Admin module
- Visitor module
- Organization module
- Public Awareness Module

Contents

1	INTRODUCTION	1
1.1	Need for the Project	1
1.2	Outline of the Project	2
1.3	Motivation	2
1.4	Scope of the project	2
1.5	About the Organization	2
2	REQUIREMENT ANALYSIS & SPECIFICATION	4
2.1	System study	4
2.1.1	Existing System	4
2.1.2	Proposed System	4
2.2	System Specification	5
2.2.1	Specification for Development	5
2.2.2	Specification for Implementation	6
2.3	Software Tools	6
2.3.1	Microsoft Visual Studio	6
2.3.2	ASP.NET	7
2.3.3	Scripting Language: C SHARP(C#)	9
2.3.4	SQL SERVER 2012	10
3	SYSTEM MODELING	12
3.1	Introduction	12
3.2	Module Description	12

3.2.1	Admin module	12
3.2.2	Organization module	13
3.2.3	Visitor module	13
3.2.4	Public Awareness module	13
3.3	Data Flow Diagram	13
3.3.1	Context Level	17
3.3.2	Level 1 DFD for Admin	18
3.3.3	Level 1 DFD for Organization	19
3.3.4	Level 1 DFD for Visitor	20
3.4	UML Diagram	21
3.4.1	Usecase Diagram	21
3.4.2	Activity Diagram	23
3.4.3	Sequence Diagram	28
4	SYSTEM DESIGN	31
4.1	Introduction	31
4.1.1	Logical design	32
4.1.2	Physical design	32
4.1.3	OUTPUT DESIGN	33
4.2	FORM DESIGN	34
4.2.1	Goal of form design	34
4.3	ARCHITECTURAL DESIGN	34
4.4	SYSTEM MODULES	35
4.5	DATA BASE DESIGN	36
4.5.1	Design Consideration	37
4.5.2	Identifying Keys	37
4.5.3	Tables	38
4.6	USER-INTERFACE DESIGN	47
4.6.1	The UI Screen shots	48

5 SYSTEM TESTING	55
5.1 Introduction	55
5.2 Unit Testing	55
5.3 Integration Testing	56
5.4 Acceptance Testing	56
6 SYSTEM IMPLEMENTATION	60
6.1 Implementation Methods	60
6.2 Implementation Plan	60
7 CONCLUSION AND FUTURE SCOPE	62
8 SAMPLE CODE	63
8.1 LoginPage	63
8.2 Create Request Page	65
8.3 Class File:Visitor-Activities	68
REFERENCES	77

List of Figures

3.1	Usecase diagram:Admin	22
3.2	Usecase diagram:Organization	22
3.3	Usecase diagram:Visitor	23
3.4	Activity Diagram: Login	24
3.5	Activity Diagram: Request	25
3.6	Activity Diagram: Add Vacancy	25
3.7	Activity Diagram: Apply Job	26
3.8	Activity Diagram: Conduct Activities	27
3.9	Activity Diagram: Share Post	27
3.10	Activity Diagram: View Blood Donors Details	27
3.11	Sequence Diagram: for Admin	28
3.12	Sequence Diagram: for Organization	29
3.13	Sequence Diagram: for Visitor	30
4.1	Screenshot 1:HomePage(First Page)	48
4.2	Screenshot 2:About	48
4.3	Screenshot 3:Login	49
4.4	Screenshot 4:Visitor Registration	49
4.5	Screenshot 5:Organization Registration	50
4.6	Screenshot 6:Admin Home Page	50
4.7	Screenshot 7:Approve Organization	51
4.8	Screenshot 8:Approve Visitor	51

4.9 Screenshot 9:Visitor Home Page	52
4.10 Screenshot 10:Create	52
4.11 Screenshot 11:View Requests	53
4.12 Screenshot 12:View blood donors Available	53
4.13 Screenshot 13:Share your thoughts	54

List of Tables

4.1	Tbl_userreg	38
4.2	Tbl_oreg	39
4.3	Tbl_Login	39
4.4	Tbl_Profile	40
4.5	Tbl_ProfileJob	40
4.6	Tbl_Accademics	41
4.7	Tbl_Requests	42
4.8	Tbl_Blood	42
4.9	Tbl_Post	43
4.10	Tbl_JobVaccancy	43
4.11	Tbl_ConductActivities	44
4.12	Tbl_AppliedJobSeeker	44
4.13	Tbl_Account	45
4.14	Tbl_Bank	45
4.15	Tbl_Branch	46
5.1	Test case:1(Login Page)	57
5.2	Test case 2: Share Thoughts Form	57
5.3	Test case 3: Add Request Form	58
5.4	Test case 4: Conduct Activities	59

Chapter 1

INTRODUCTION

1.1 Need for the Project

Social Service serves as a constant reminder that we should do well and share what we have with those who are under-privileged. Social services are a range of public services provided by the different organizations. Social services include the benefits and facilities such as education, food subsidies, health care, police, fire service, job training and subsidized housing, adoption, community management, policy research, and lobbying. One of the most critical standards in running a charity organization is raising the required funds, which are practically necessary in conducting the associated social service. The major hurdle that is practically executed in running a charity system is identifying the donors who have been regularly funding charity organization for different purposes. Whenever the request for a new fund arises the authorities within the system have to make extensive search for finding the right donors for that specific charity request.

Social Service serves as a constant reminder that we should do well and share what we have with those who are under-privileged. Public Awareness and Social Service is a society run by a group of Colleges and Schools, aiming to promote public awareness and community involvement among students. Public Awareness is an essential attribute of a responsible citizen in a democratic society and it is also a building block for a successful career in the future. The main Aim of our website is to co-ordinate everyone to help poor students with their studies, build house for homeless peoples etc.

1.2 Outline of the Project

The Requirement analysis and Specification is included in chapter 2. It provides a model of system information, function and behavior. System modeling is described in chapter 3. Chapter 4 has the review of overall system design. Testing methods and system testing reports are included in chapter 5. System implementation and implementation methods are discussed in chapter 6. The conclusion and future scope of the project is summarized in chapter 7. The concluding discussions include recommendations for future investigations.

1.3 Motivation

The main motivation to go ahead with this project is that in the commercial world the importance of social service. Social Service serves as a constant reminder that we should do well and share what we have with those who are under-privileged.

I hope that this tool will help organizations as well as individuals to help others and also helps the needy to reach their requirement through few clicks.

1.4 Scope of the project

This software mainly focuses on services. It is developed as a charity software. So it can be used in any organization for providing helps to the needy, add vacancy, etc.

1.5 About the Organization

St. Joseph's College of Engineering and Technology, Palai (also referred to as SJCET Palai) is an Indian private college managed by the Syro-Malabar Catholic Diocese of Palai, located in Palai, Kerala, India. It is an AICTE approved college offering professional degree programs in engineering and management courses.

The college was founded by a group of well-known academics. They are pioneering educators, having unmatched experience in the field of education with a belief that the

continuous search for knowledge is the sole path to success. The Primary focus of the institution is to expose the young minds to be world of technology, instilling in them confidence and fortitude to face new challenges that enables them to excel in their chosen field. The college inculcates the development of all facets of the mind culminating in an intellectual and balanced personality. Our team of dedicated and caring faculty strives to widen the students horizon of learning thereby achieving excellent results for every student.

St. Josephs College Of Engineering And Technology, right from inception, has been maintaining high levels of standard in academic and extracurricular realms of activities. We offer BTECH degree courses in 6 engineering disciplines, and Masters Degree courses in Engineering, Computer Application and Business Administration. In the short span of a decade of its existence and among the six batches of students that have graduated, the college bagged several university ranks and has a remarkably high percentage of pass. The students of batch of MCA bagged ranks in the university. The college is also the venue of national and state level seminars and symposiums and has emerged as the hub of technical education in the state. The placement scenario is also quite commendable, with several premier industries visiting for St. Josephs College Of Engineering And Technology placement and recruitment.

Chapter 2

REQUIREMENT ANALYSIS & SPECIFICATION

2.1 System study

To develop a computer based system, system analysis is to be performed for an enhanced data model. Firstly studied the existing system in detail for preparing the data model.

Analysis involves requirement determination and specification. Basically it involves establishing requirements for all system elements and then mapping these requirements to the software form. The purpose is to provide an understanding and to enable a communication about the system between the developers and the people establishing requirements.

2.1.1 Existing System

Usually, the one(poor) who need some requests are send their requests to a person/organization. Or the requests are shared to a group by means of mouth-to-mouth as well as social medias. There is actually no platform for performing almost all kind of social services in a single platform.

2.1.2 Proposed System

Social services are a range of public services provided by the different organizations. Social services include the benefits and facilities such as education, food subsidies, health care, police, fire service, job training and subsidized housing, adoption, community man-

agement, policy research, and lobbying. WeShare is a website run by a group of Organizations and peoples, aiming to promote public awareness and community involvement among individuals. The major hurdle that is practically executed in running a charity system is identifying the donors who have been regularly funding charity organization for different purposes. Whenever the request for a new fund arises the authorities within the system have to make extensive search for finding the right donors for that specific charity request.

The aim of my website is to coordinate individuals and organizations to help poor students with their studies, to help poor in health care, to help unemployed persons in finding jobs etc. Public awareness is the public's level of understanding about the importance and implications of socially relevant topics.

2.2 System Specification

2.2.1 Specification for Development

Hardware Specification

- Processor : Intel Pentium IV and above
- RAM : 256 MB
- Hard Disk : 200 GB or higher
- Display: 14.1 Colour Monitor(CRT,LCD or LED)

Software Specification

- Operating System : Windows XP or higher
- Web Browser : Google chrome or other web browsers
- Front-End : HTML, Javascript, C
- IDE : Visual Studio 2017
- Framework : ASP.NET

- Back-End Tool : SQL SERVER 2012

2.2.2 Specification for Implementation

Hardware Specification

- Processor : Intel Pentium IV and above
- RAM : 256 MB or above
- Hard Disk : 40 GB or higher
- Display: 14.1 Colour Monitor(LCD,CRT or LED)

Software Specification

- Operating System :Linux or Windows XP or higher.
- Web Browser : Internet Explorer or other web browsers.

2.3 Software Tools

2.3.1 Microsoft Visual Studio

Microsoft Visual Studio is an integrated development environment (IDE) from Microsoft. It is used to develop computer programs, as well as websites, web apps, web services and mobile apps. Visual Studio uses Microsoft software development platforms such as Windows API, Windows Forms, Windows Presentation Foundation, Windows Store and Microsoft Silverlight. It can produce both native code and managed code.

Visual Studio includes a code editor supporting IntelliSense (the code completion component) as well as code refactoring. The integrated debugger works both as a source-level debugger and a machine-level debugger. Other built-in tools include a code profiler, forms designer for building GUI applications, web designer, class designer, and database schema designer. It accepts plug-ins that enhance the functionality at almost every level—including adding support for source control systems (like Subversion and Git) and adding

new toolsets like editors and visual designers for domain-specific languages or toolsets for other aspects of the software development lifecycle (like the Team Foundation Server client: Team Explorer).

Visual Studio supports 36 different programming languages and allows the code editor and debugger to support (to varying degrees) nearly any programming language, provided a language-specific service exists. Built-in languages include C,[7] C++, C++/CLI, Visual Basic .NET, C, F,[8] JavaScript, TypeScript, XML, XSLT, HTML, and CSS. Support for other languages such as Python,[9] Ruby, Node.js, and M among others is available via plug-ins. Java (and J) were supported in the past.

The most basic edition of Visual Studio, the Community edition, is available free of charge. The slogan for Visual Studio Community edition is "Free, fully-featured IDE for students, open-source and individual developers".

2.3.2 ASP.NET

Microsoft .NET technology you will have access to a new generation of advanced software joining the best of computing and communications in a revolutionary new way. The effect will be to totally transform the Web and every other aspect of the computing experience. .NET enables developers, businesses, and consumers to harness technology on their terms. .NET will allow the creation of truly distributed Web services that will integrate and collaborate with a range of complementary services to help customers in ways that todays dotcoms can only dream of. The fundamental idea behind .NET is that the focus is shifting from individual Web sites or devices connected to the Internet to constellations of computers, devices, and services that work together to deliver broader, richer solutions. People will have control over how, when, and what information is delivered to them. Computers, devices and services will be able to collaborate with each other to provide rich services, instead of being isolated islands where the user provides the only integration. Businesses will be able to offer their products and services in a way that lets customers seamlessly embed them in their own electronic fabric. Microsoft .NET will make computing and communicating simpler and easier than ever. It will spawn a new generation

of Internet services, and enable tens of thousands of software developers to create revolutionary online services and businesses. It will put you back in control, and enable greater control of your privacy, digital identity, and data. And software is what makes it all possible. However, Microsoft's .NET technology will only succeed if others adopt this new standard.

The ASP.NET worker process is a distinct worker process, `aspnet_wp.exe`, *separate from inetinfo.exe (Microsoft Platforms also).*

We are no longer constrained to the two scripting languages available in traditional ASP: Any fully compliant .NET language can now be used with ASP.NET, including C and VB.NET

In contrast, ASP.NET pages are always compiled into .NET classes housed within assemblies. This class includes all of the server-side code and the static HTML, so once a page is accessed for the first time (or any page within a particular directory is accessed), subsequent rendering of that page is serviced by executing compiled code. This eliminates all the inefficiencies of the scripting model of traditional ASP. There is no longer any performance difference between compiled components and server-side code embedded within a page they are now both compiled components. There is also no performance difference between interspersing server-side code blocks among static HTML elements, and writing large blocks of server-side code and using `Response.Write()` for static HTML content. Also, because the `.aspx` file is parsed into a single code file and compiled, it is not possible to use multiple server-side languages within a single `.aspx` file.

But in ASP.NET In addition to improved performance over the interpreted model, pages that are compiled into classes can be debugged using the same debugging tools available to desktop applications or component developers. Errors with pages are generated as compiler errors, and there is a good chance that most errors will be found at compilation time instead of runtime, because VB.NET and C# are both strongly typed languages. Plus, all the tools available to the .NET developer are applicable to the `.aspx` developer

In ASP.NET it is no longer possible to include executable code outside the scope of a function within a script block marked as `runat= server`, and conversely, it is no longer possible

to define a function within a pair of server-side script tags. But in ASP.NET, you are now required to place the Language directive with a Page directive, as follows: You can have as many lines of directives as you need. Directives may be located anywhere in your .aspx file but standard practice is to place them at the beginning of the file. Several new directives have been added in ASP.NET

2.3.3 Scripting Language: C SHARP(C#)

C# is a general purpose programming object oriented language invented around 1999 or 2000 by Anders Hejlsberg at Microsoft. It is very similar to Java in its syntax with a major difference being that all variable types are descended from a common ancestor class. The purpose of C# is to precisely define a series of operations that a computer can perform to accomplish a task. Most of these operations involve manipulating numbers and text, but anything that the computer can physically do can be programmed in C#. Computers have no intelligence they have to be told exactly what to do and this is defined by the programming language you use. Once programmed, they can repeat the steps as many times as you wish at very high speeds. Modern PCs are so fast they can count to a billion in a second or two.

Features

- It has no global variables or functions. All methods and members must be declared within classes. Static members of public classes can substitute for global variables and functions.
- Local variables cannot shadow variables of the enclosing block, unlike C and C++. Variable shadowing is often considered confusing by C++ texts.
- C# supports a strict Boolean data type
- Managed memory cannot be explicitly freed; instead, it is automatically garbage collected. Garbage collection addresses the problem of memory leaks by freeing the programmer of responsibility for releasing memory that is no longer needed.

- In addition to the try...catch construct to handle exceptions, C# has a try...finally construct to guarantee execution of the code in the finally block.
- Multiple inheritance is not supported, although a class can implement any number of interfaces. This was a design decision by the language's lead architect to avoid complication and simplify architectural requirements throughout CLI.
- C#, like C++ (but unlike Java), supports operator overloading.
- C# currently (as of version 4.0) has 77 reserved words

2.3.4 SQL SERVER 2012

Microsoft SQL Server is a relational database management system developed by Microsoft. As a database server, it is a software product with the primary function of storing and retrieving data as requested by other software applications which may run either on the same computer or on another computer across a network (including the Internet).

SQL SERVER is a database management system

A database is a structured collection of data. It may be anything from a simple shopping list to a picture gallery or the vast amounts of information in a corporate network. To add, access, and process data stored in a computer database, you need a database management system such as SQL Server. Since computers are very good at handling large amounts of data, database management systems play a central role in computing, as standalone utilities, or as parts of other applications.

SQL SERVER databases are relational

A relational database stores data in separate tables rather than putting all the data in one big storeroom. The database structures are organized into physical files optimized for speed. The logical model, with objects such as databases, tables, views, rows, and columns,

offers a flexible programming environment. You set up rules governing the relationships between different data fields, such as one-to-one, one-to-many, unique, required or optional, and pointers between different tables. The database enforces these rules, so that with a well-designed database, your application never sees inconsistent, duplicate, orphan, out-of-date, or missing data.

Chapter 3

SYSTEM MODELING

3.1 Introduction

System modeling implies doing modeling for the system, which means creating models or prototypes of the system. They are helpful for further development. Unified Modeling Language(UML) helps in creating diagrammatic representation of models. System models may be used in different ways as part of a process for improving understanding of a situation, identifying problems or formulating opportunities and supporting decision making.

3.2 Module Description

The system after careful analysis has been identified to be presented with the following modules. The Modules involved are:

1. Admin Module
2. Organization Module
3. Visitor Module

3.2.1 Admin module

This module takes the overall responsibility upon the systems transactions and operations. The administrative module takes the sole responsibility of managing the reliability

and consistency of the data. The system takes care of the data collection, updating and deletion of the actual data at the centralized level.

3.2.2 Organization module

This module handles the organizations who acts as donors who have come forward to participate in the charity funding. It also integrates itself to the charity request expenditures, and the charity clients information. This module also manages all the transactions and operations that relate to the donors.

3.2.3 Visitor module

This module manages the clients who got registered onto the system; for the request of a charity fund. It gets integrated with the standards of mail management that exists that exists upon the system. The clients also integrate themselves with the beneficiaries who are targeted upon the charity fund that is raised for the purpose.

3.2.4 Public Awareness module

This module manages the details of all the clients who got registered onto the system; for the request of a charity fund. It gets integrated with the standards of mail management that exists that exists upon the system. The clients also integrate themselves with the beneficiaries who are targeted upon the charity fund that is raised for the purpose.

3.3 Data Flow Diagram

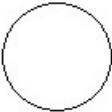
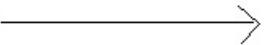
A data flow diagram is a graphical technique that depicts information flow and transforms that are applied as data move from input to output. The DFD is used to represent increasing information flow and functional details. A level 0 DFD also called a fundamental system model represents the entire software elements as a single bible with input and output indicated by incoming and outgoing arrows respectively.

A data-flow diagram (DFD) is a way of representing a flow of a data of a process or a system (usually an information system). The DFD also provides information about the outputs and inputs of each entity and the process itself. A data-flow diagram has no control flow, there are no decision rules and no loops. Specific operations based on the data can be represented by a flowchart.

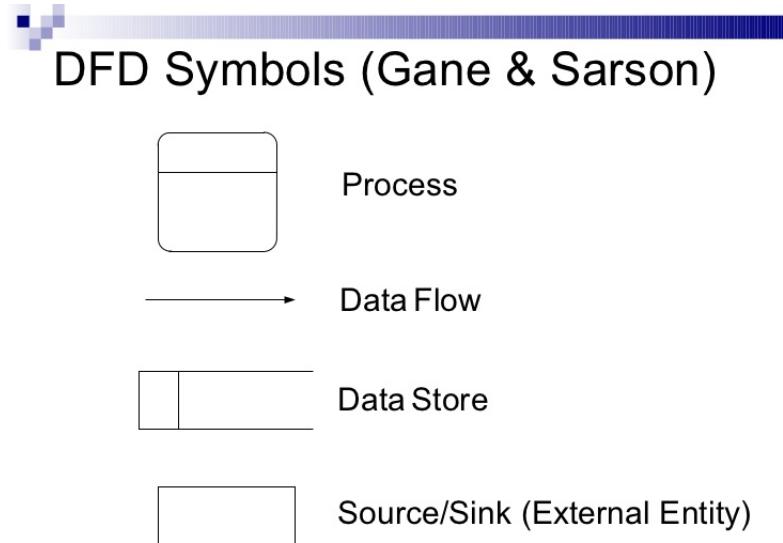
Larry Constantine first developed the DFD as a way of expressing system requirements in a graphical form, this lead to the modular design. A DFD is also known as a bubble Chart has the purpose of clarifying system requirements and identifying major transformations that will become programs in system design. So it is the starting point of the design to the lowest level of detail. A DFD consists of a series of bubbles joined by data flows in the system. There are mainly two models for drawing DFD:

- Gane Sarsson Model
- Yorden Codd Model

The notations for displaying data-flow diagrams in Yorden Codd are:

Symbol	Meaning
Process 	Single Process : A circle is used to represent the entire system.
Data Flow 	Data Flow : An arrow is used to represent the flow of data between the process and external entities.
External Entity 	External Entity : A square or rectangle represents any person or organization that sends data to or receives data from the system.
Data Store 	Data Store : An open rectangle represents the location where the data is stored. It could be a filing cabinet, harddisk.

The notations for displaying data-flow diagrams in Gane Sarsson Model are:



Data flow diagrams present the logical flow of information through a system in graphical or pictorial form. Data flow diagrams have only four symbols, which makes it useful for communication between analysts and users. Data flow diagrams (DFDs) show the data used and provided by processes within a system. DFDs make use of four basic symbols.

- **External Entity**

An external entity is a source or destination of a data flow which is outside the area of study. Only those entities which originate or receive data are represented on a business process diagram. The symbol used is an oval containing a meaningful and unique identifier.

- **Process**

A process shows a transformation or manipulation of data flows within the system. The symbol used is a rectangular box which contains 3 descriptive elements:

Firstly an identification number appears in the upper left hand corner. This is allocated arbitrarily at the top level and serves as a unique reference.

Secondly, a location appears to the right of the identifier and describes where in the system the process takes place. This may, for example, be a department or a piece of

hardware. Finally, a descriptive title is placed in the centre of the box. This should be a simple imperative sentence with a specific verb, for example 'maintain customer records' or 'find driver'.

- Data Flow

A data flow shows the flow of information from its source to its destination. A data flow is represented by a line, with arrowheads showing the direction of flow. Information always flows to or from a process and may be written, verbal or electronic. Each data flow may be referenced by the processes or data stores at its head and tail, or by a description of its contents.

- Data Store

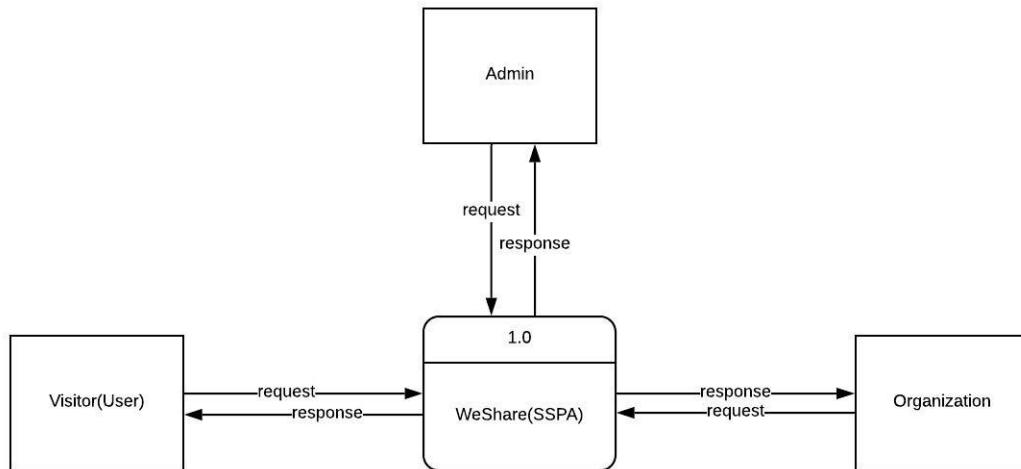
A data store is a holding place for information within the system:

It is represented by an open ended narrow rectangle. Data stores may be long-term files such as sales ledgers, or may be short-term accumulations: for example batches of documents that are waiting to be processed. Each data store should be given a reference followed by an arbitrary number.

DFD for WeShare using Gane Sarsson Model

3.3.1 Context Level

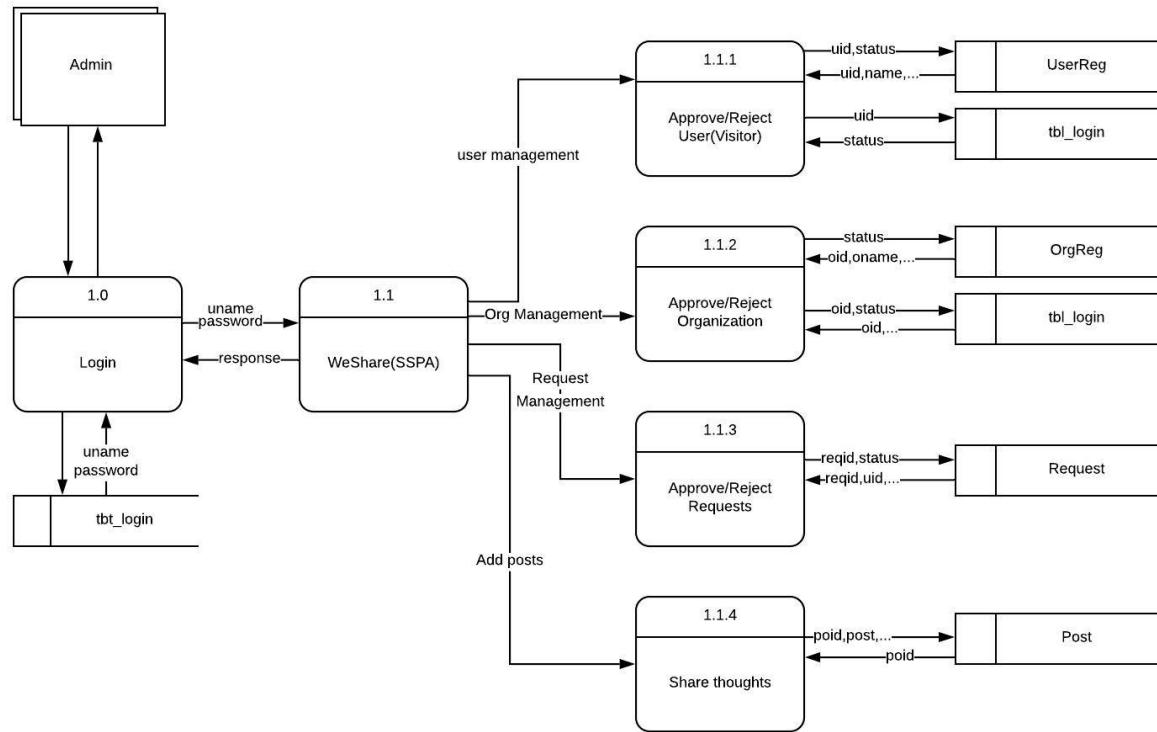
A context diagram is a top level (also known as "Level 0") data flow diagram. It only contains one process node ("Process 0") that generalizes the function of the entire system in relationship to external entities. Here the function of WeShare in relation to external entities Admin, Visitor(user) and Organization are defined.



3.3.2 Level 1 DFD for Admin

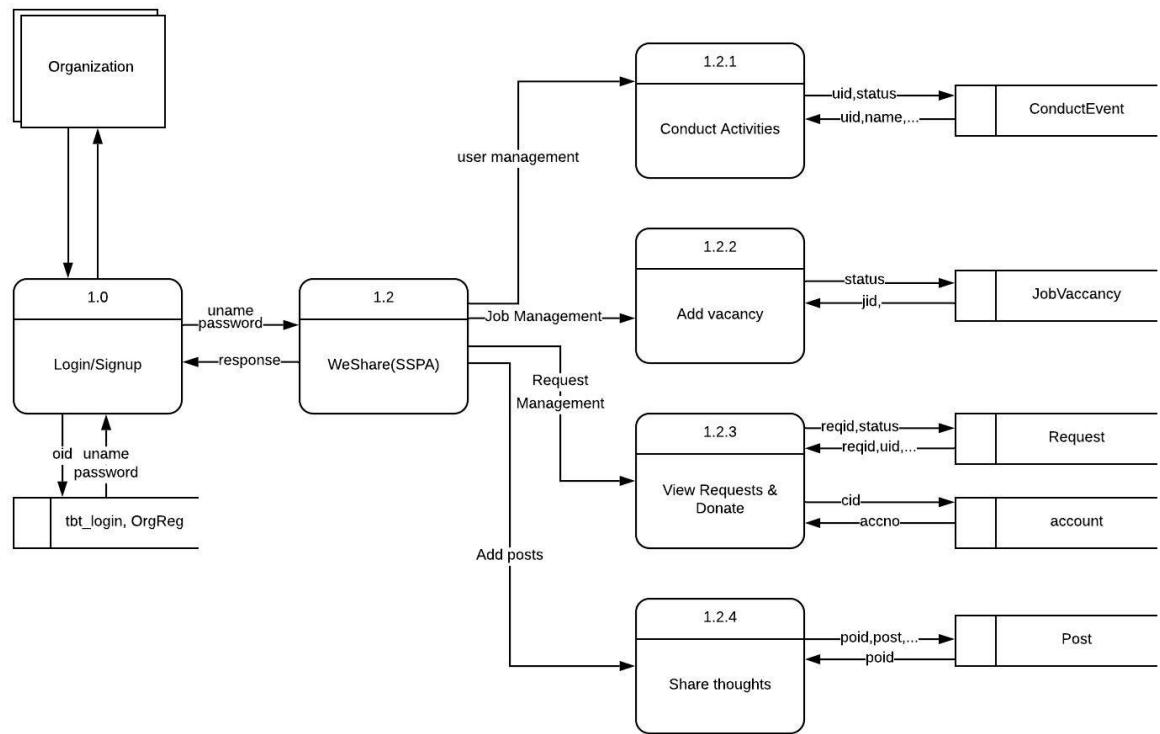
A level 1 DFD notates each of the main sub-processes that together form the complete system. We can think of a level 1 DFD as an exploded view of the context diagram.

Here the entire sub processes that the admin performs are depicted.



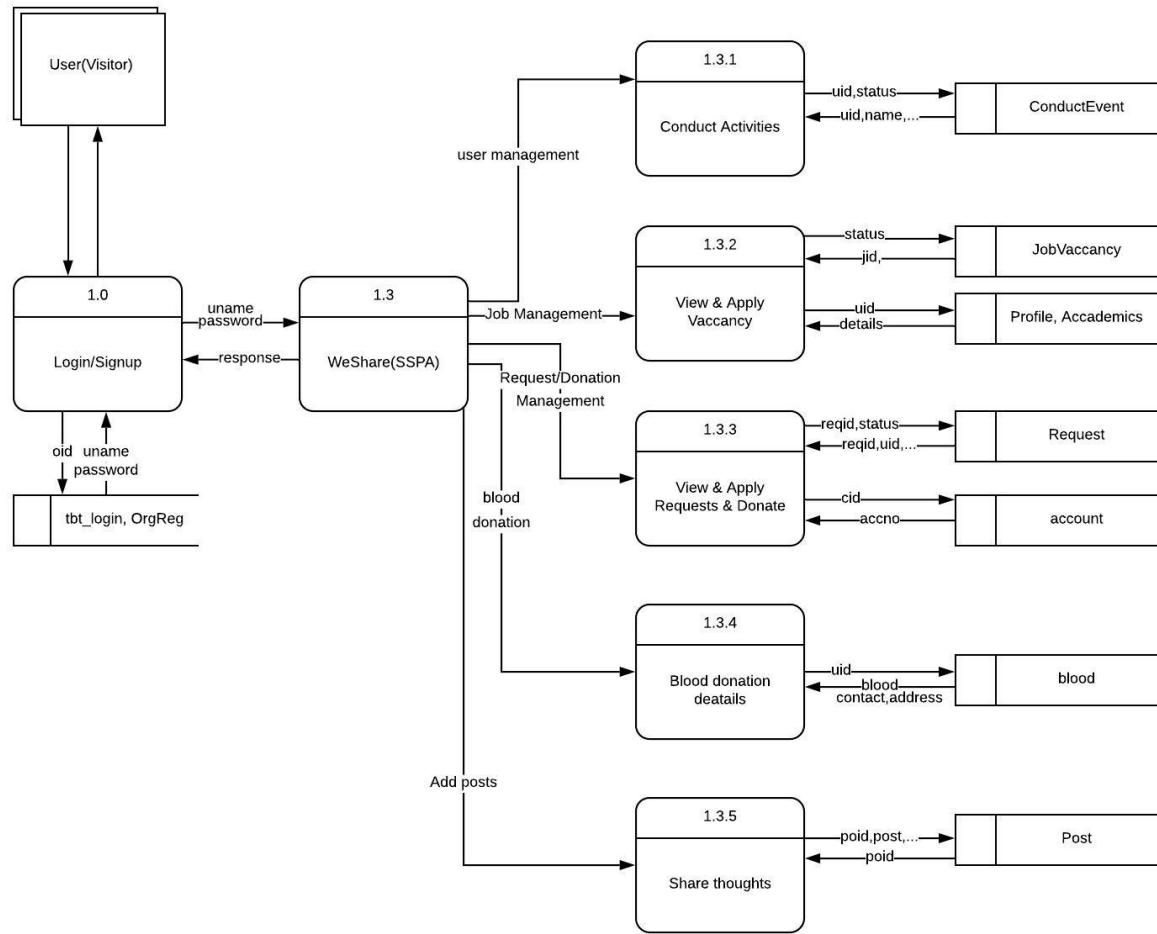
3.3.3 Level 1 DFD for Organization

Here the entire sub process that the organization can perform is depicted.



3.3.4 Level 1 DFD for Visitor

Here the entire sub process that the Visitor(user) can perform is depicted.



3.4 UML Diagram

A UML diagram is a diagram based on the UML (Unified Modeling Language) with the purpose of visually representing a system along with its main actors, roles, actions, artifacts or classes, in order to better understand, alter, maintain, or document information about the system.

3.4.1 Usecase Diagram

A Use Case diagram illustrates a set of use cases for a system, i.e. the actors and the relationships between the actors and use cases. A use case diagram at its simplest is a 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. The use cases are represented by either circles or ellipses.

Symbol	Reference Name
	Actor
	Use case
	Relationship

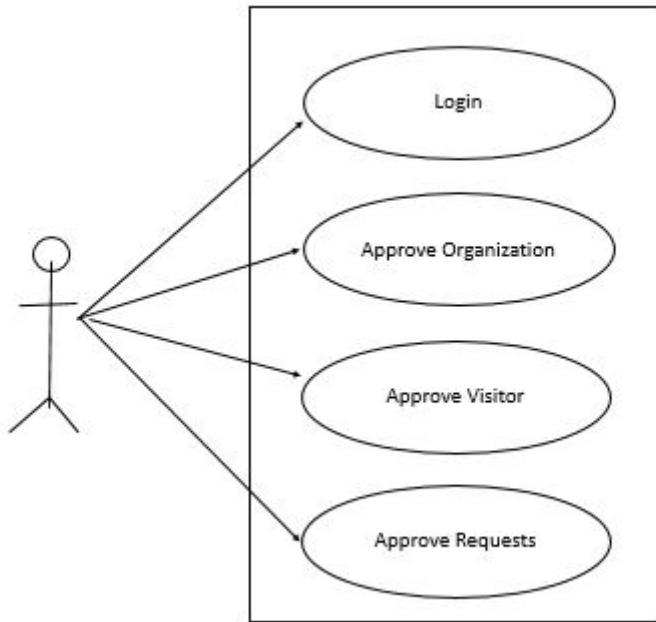


Figure 3.1: Usecase diagram:Admin

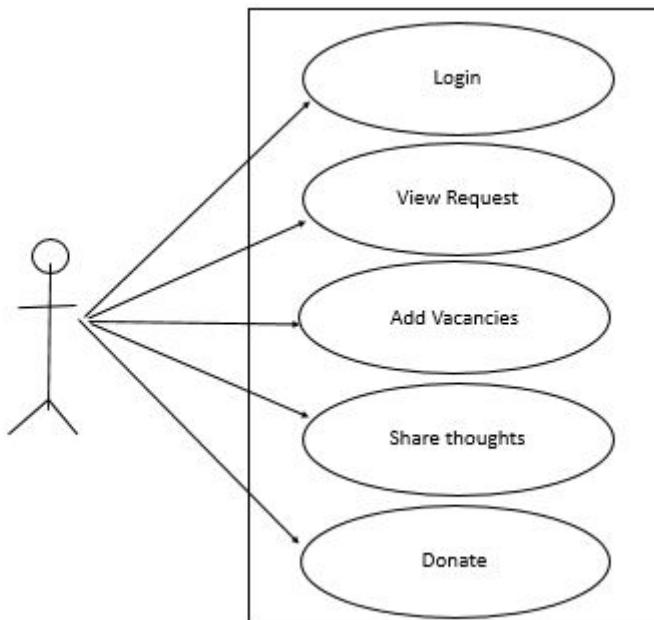


Figure 3.2: Usecase diagram:Organization

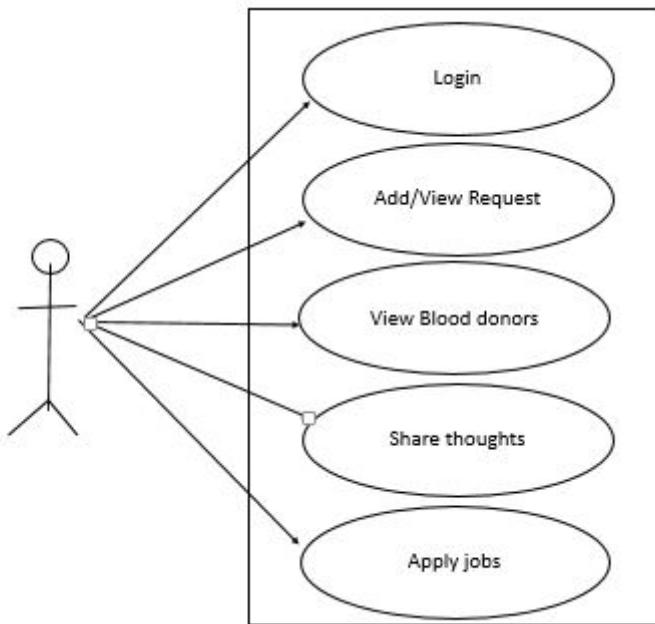


Figure 3.3: Usecase diagram:Visitor

3.4.2 Activity Diagram

Activity diagrams are graphical representations of work flows of stepwise activities and actions with support for choice, iteration and concurrency. In the Unified Modeling Language, activity diagrams are intended to model both computational and organizational processes (i.e., work flows), as well as the data flows intersecting with the related activities. Although activity diagrams primarily show the overall flow of control, they can also include elements showing the flow of data between activities through one or more data stores.

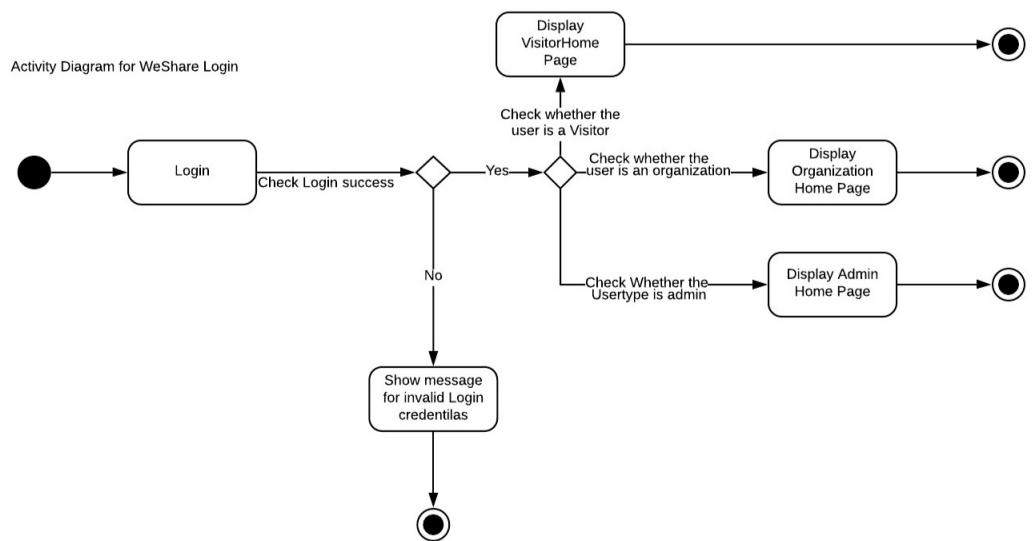
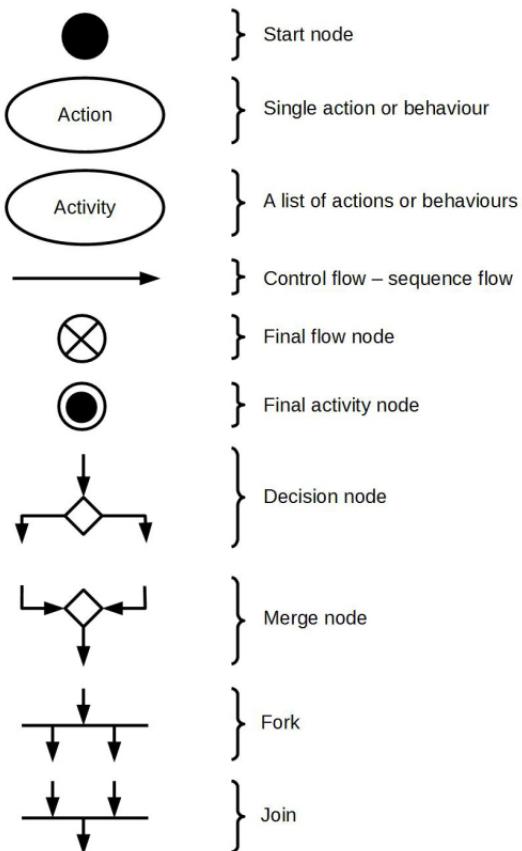
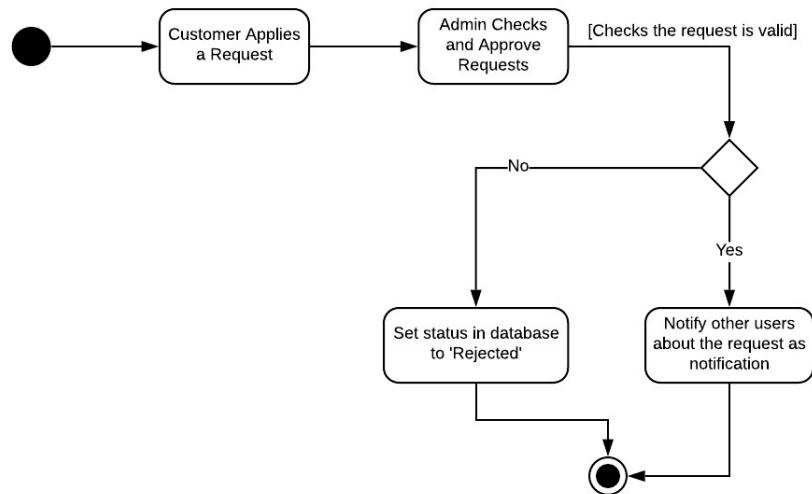


Figure 3.4: Activity Diagram: Login

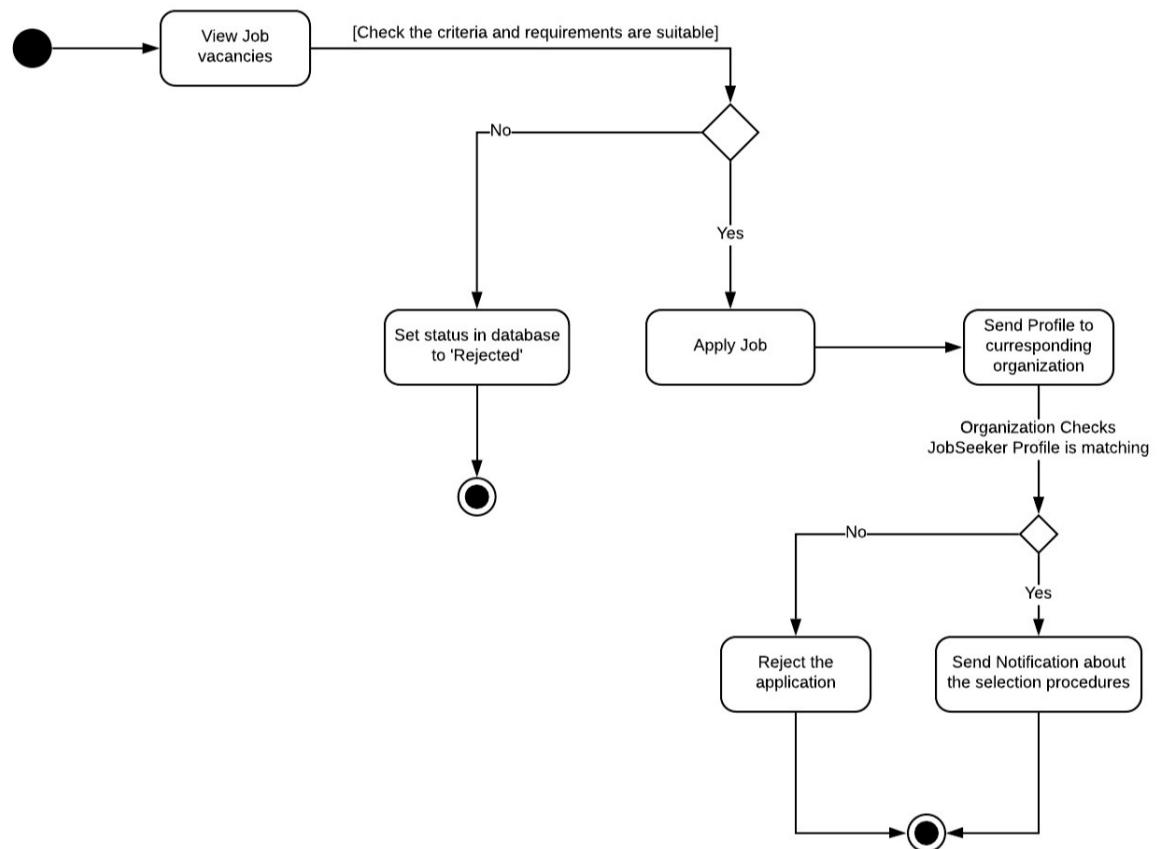
Activity Diagram for Requests Application

**Figure 3.5:** Activity Diagram: Request

Activity diagram for Vacancy Adding

**Figure 3.6:** Activity Diagram: Add Vacancy

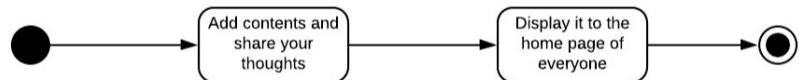
Activity Diagram for Applying Jobs

**Figure 3.7:** Activity Diagram: Apply Job

Activity Diagram for Conduct Activities

**Figure 3.8:** Activity Diagram: Conduct Activities

Activity Diagram for Share Post

**Figure 3.9:** Activity Diagram: Share Post

Activity diagram for View Blood donors

**Figure 3.10:** Activity Diagram: View Blood Donors Details

3.4.3 Sequence Diagram

A sequence diagram shows object interactions arranged in time sequence. It depicts the objects and classes involved in the scenario and the sequence of messages exchanged between the objects needed to carry out the functionality of the scenario. Sequence diagrams are typically associated with use case realizations in the Logical View of the system under development. Sequence diagrams are sometimes called event diagrams or event scenarios.

A sequence diagram shows, as parallel vertical lines (lifelines), different processes or objects that live simultaneously, and, as horizontal arrows, the messages exchanged between them, in the order in which they occur. This allows the specification of simple runtime scenarios in a graphical manner.

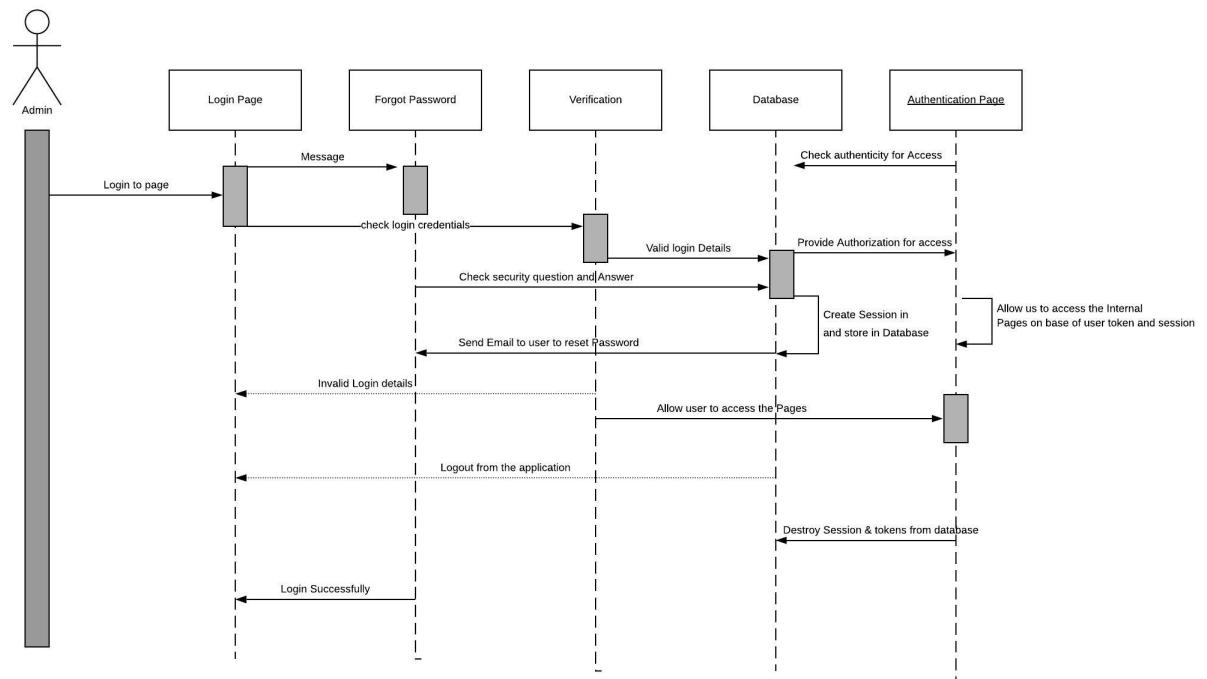
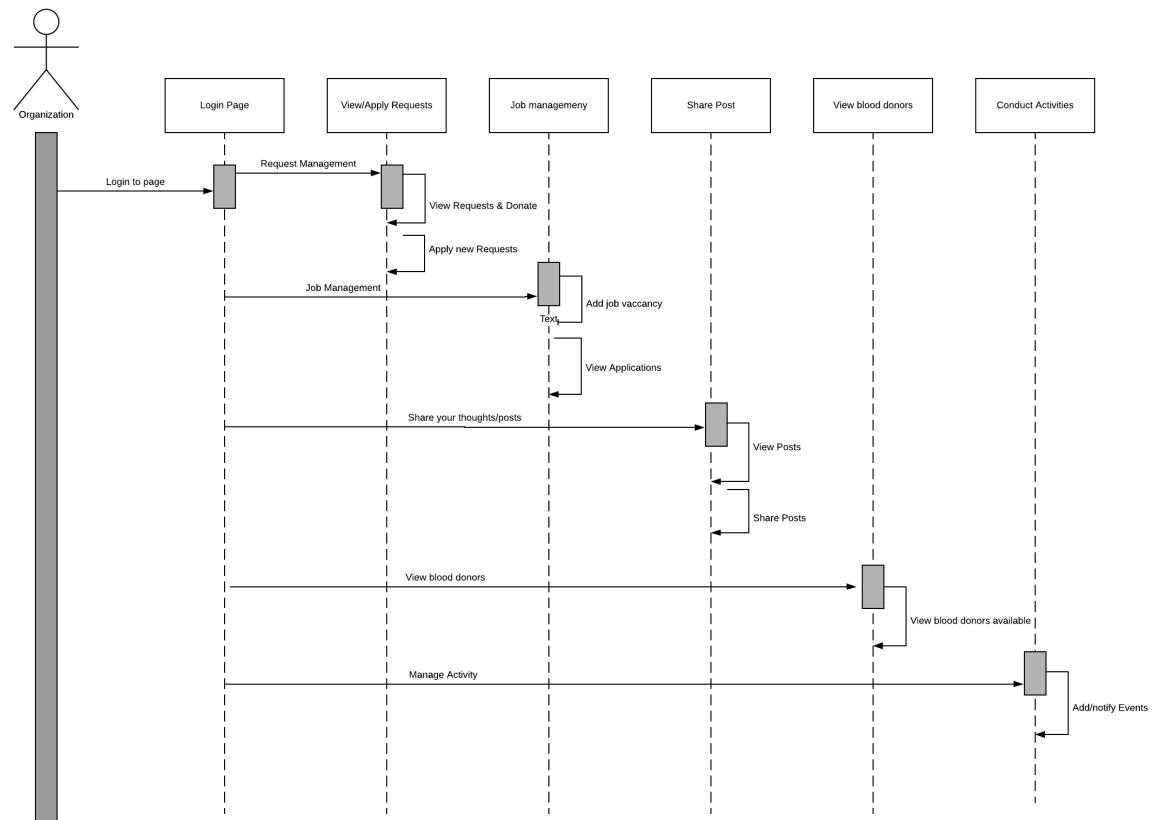
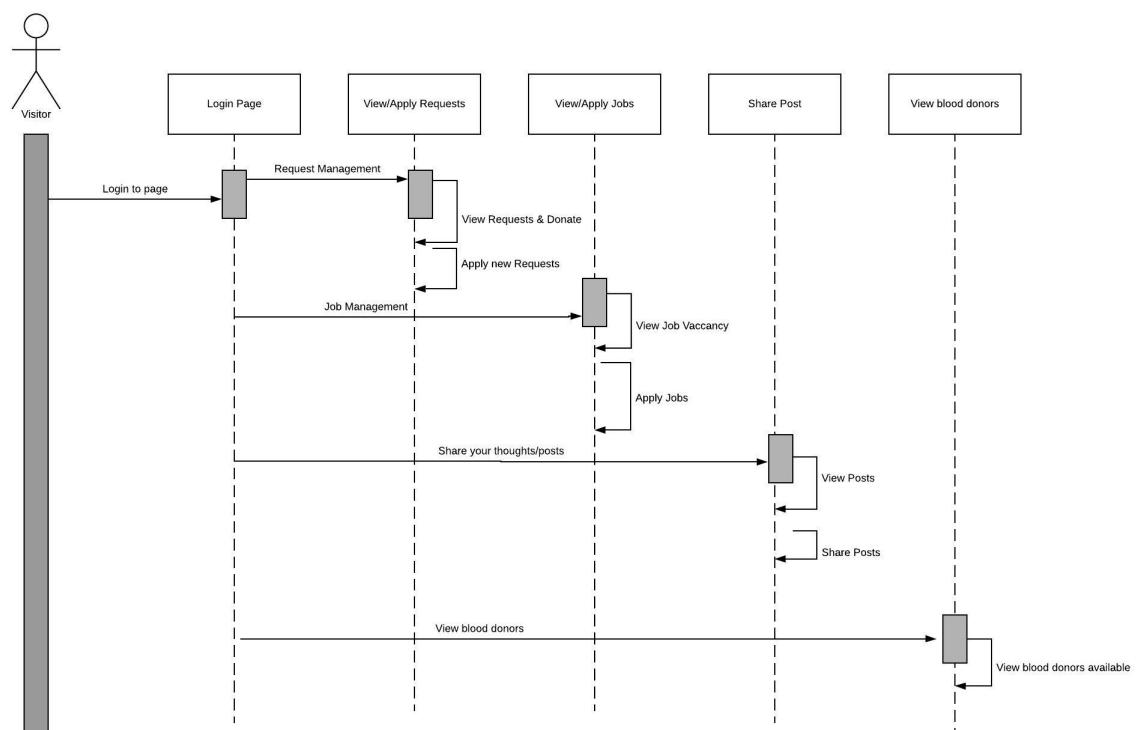


Figure 3.11: Sequence Diagram: for Admin

**Figure 3.12:** Sequence Diagram: for Organization

**Figure 3.13:** Sequence Diagram: for Visitor

Chapter 4

SYSTEM DESIGN

4.1 Introduction

The system design is the most creative and challenging phase of system development life cycle. It is an approach for the creation of proposed system, in which the logic and details structure of the proposed system is designed, which will help the system coding. The most creative and challenging phase of the system development process is design phase it is a solution, a how to approach to the creation of the proposed system. Design is the first step in the development of the engineered product is initiated only after a clear exposition of expected product is available. System design is vital for efficient database management. It provides the understanding of procedural detail necessary for implementing the system. A number of sub-system is to be identified which constitutes the whole system.

From a project management point of view, software design is conducted in two steps. Preliminary design is concerned with the information of requirements into data and software architecture. Details design focuses on refinement to the architectural representation. Design starts with the system requirement specification and converts it to a physical reliability, response time, throughput of the system, maintainability, expandability should be taken into account.

The database tables are designed by using the entire necessary field in a compact and correct manner. Care has been taken to be avoiding redundant and duplicated field. The important of software design can be started with word quality .Design is place where

quality is fostered in software development. Designs are the only way where requirements are actually translated in to a finished software product or system

4.1.1 Logical design

The logical design of a system pertains to an abstract representation of the data flows, inputs and outputs of the system. This is often conducted via modeling, using an over-abstract (and sometimes graphical) model of the actual system. In the context of systems design are included. Logical design includes ER Diagrams i.e.Entity Relationship Diagrams.

4.1.2 Physical design

The physical design relates to the actual input and output processes of the system. This is laid down in terms of how data is input into a system, how it is verified/authenticated, how it is processed, and how it is displayed as In Physical design, following requirements about the system are decided.

1. Input requirement.
2. Output requirements.
3. Storage requirements.
4. Processing Requirements.

System control and backup or recovery. Put another way, the physical portion of systems design can generally be broken down into three sub-tasks:

1. User Interface Design.
2. Data Design.
3. Process Design.

User Interface Design is concerned with how users add information to the system and with how the system presents information back to them. Data Design is concerned with how the data is represented and stored within the system. Finally, Process Design is concerned with how data moves through the system, and with how and where it is validated, secured and/or transformed as it flows into, through and out of the system. At the end of the systems design phase, documentation describing the three sub-tasks is produced and made available for use in the next phase.

Physical design, in this context, does not refer to the tangible physical design of an information system. To use an analogy, a personal computer's physical design involves input via a keyboard, processing within the CPU, and output via a monitor, printer, etc. It would not concern the actual layout of the tangible hardware, which for a PC would be a monitor, CPU, motherboard, hard drive, modems, video/graphics cards, USB slots, etc. It involves a detailed design of a user and a product database structure processor and a control processor. The H/S personal specification is developed for the proposed system.

4.1.3 OUTPUT DESIGN

Outputs from computer systems are required primarily to communicate the results of processing to users. They are also used to provide a permanent copy of the results for later consultation. Intelligent output designs will improve systems relationships with the users and help much in decision making. Various onscreen messages are also displayed for effective communication with the user.

The output design is simple and easy to understand. The significant points that have to be considered for the output design are as follows.

- Determine what information to be present.
- Decide whether to display, print or voice the information and select the medium of output.
- Arrange the presentation information in an acceptable format.
- Decide how to describe the output to intended represents.

4.2 FORM DESIGN

The process of designing forms involves clarifying the specific needs of your application, identifying the information you want to work with, and then devising a design that best meets your needs. This section briefly describes the process.

4.2.1 Goal of form design

The goal of form design is to display and obtain the information you need in an accessible, efficient manner. The form should encapsulate data so that it may be run without affecting other forms that use the same data. Dbase Plus makes this simple. It's important for your design to provide users with the information they need and clearly tell them what they need to do to successfully complete a task. A well-designed form has visual appeal that motivates users to use your application. In addition, it should use limited screen space efficiently.

4.3 ARCHITECTURAL DESIGN

Architectural design of a computing system is the structure of the system, which comprises software element, the externally visible properties of those elements, and the relationship among them. Architectural design defines the relationship between major structural elements of the program. Large system can be decomposed into subsystems that provide some related set of services.

Architectural design is the process of identifying these subsystems and establishing a frame work for the subsystem controls and communication. The primary objective is to develop a modular program structure and represent the control relationship between modules. In addition architectural design modules program structure and data flow through the program. Architectural design is aimed at trying to refine the conceptual view of the software product. It permits a systems quality attributes such as performance or reliability.

Architectural design is the back born for building successful software intensive system. A good software structure design results in the development of a project working

system. For the development of the software, modules were identified .This software was developed into separate named and addressable components that are integrated to satisfy the problem requirements. The modular design reduces the complexity, facilitates change and results in easier implementation by encouraging parallel development of different part of the system. The procedural design transforms structural elements of program architecture into a procedural description of software components. The architectural design considers architecture as the most important functional requirements.

The system is based on three - tier architecture. The first level is the user interface (presentation Logic), which displays controls, and receives and validates user inputs .All the event handlers in the web form are in the first level. The second level is the business layer (Business Logic) where the application-specific logic takes place. The third level is the data layer where the application information is stored in files or database. It contains logic about to retrieve and update data. The important feature about three tier design is that information only travels from one level to an adjacent level.

4.4 SYSTEM MODULES

A module is a separate unit of software or hardware. Typical characteristics of modular components include portability, which allows them to be used in a variety of systems, and interoperability, which allows them to function with the components of other systems.In software, a module is a part of a program. Programs are composed of one or more independently developed modules that are not combined until the program is linked. A single module can contain one or several routines. In this system we use 4 modules

1. Administration:-Admin is the person who administrates the entire system. He publishes the details about social service activities in the site. Administrator controls the events done by visitor and organizations. They can register into the system. But they need approval from the admin to login to the system. Admin should approve client request details and add it as a notification to other users.
2. Organization level : - Firstly organization register into the system. After the ap

proval by the admin the organization can login into the system using username and password. organization can add notification such as vacancy details, for the visitors. They can also donate their share to the needy on seeing the requests as notification if they want. They can conduct events such as seminar, marathon etc. for the promotion of their activities too. They can share their views and thoughts as a post which can be seen by other users in their home page.

3. Visitor level : - Firstly visitor register in the site. After the approval by the admin the visitor can login into the system using username and password. Social service activities are done by following the notification from the other users. Visitor can add their requirement as request and admin checks it. If admin approves, the other users can see this as a notification.
4. Public Awareness : - This module comprises the social awareness. The homepage of each user is filled with current news and the posts shared by other users of the system. The user can post their view and thought that can be viewed by others in the system.

4.5 DATA BASE DESIGN

A database is a collection of interrelated data stored with minimum redundancy to serve many users quickly and efficiently. The general objective is to make information access easy, quick, inexpensive and flexible for the users. The general theme behind a database is to integrate all information. Database design is recognized as a standard of management information system and is available virtually for every computer system. In database design several specific objectives are considered:

- Ease of learning and use
- Controlled redundancy
- Data independence
- More information at low cost

- Accuracy and integrity
- Recovery from failure
- Privacy and security
- Performance

A database is an integrated collection of data and provides centralized access to the data. Usually the centralized data managing the software is called RDBMS. The main significant difference between RDBMS and other DBMS is the separation of data as seen by the program and data has direct access to stores device. This is the difference between logical and physical data.

4.5.1 Design Consideration

The system is analyzed to the requirements and possible tables and fields are identified.

4.5.2 Identifying Keys

Once we have drawn up the lists of possible tables and fields, the next step in there logic database is to identify and foreign keys for each tables

- Primary Keys

A primary key (pk) consist of a field or set of fields that uniquely identify each record in that table. The primary field defines the primary key.

- Foreign Keys

A foreign key (fk) comprises a field, or multiple field that to the primary key of another table. For any database application data is stored in tables. So table designing is a most important part of back-end designing. Steps are taken to avoid unnecessary replication of data and to achieve maximum data consistency and integrity. The

appropriate data type is chosen for the effective storage and manipulation of information. Another important matter about table designing is the field width. Each field has enough space to accommodate the data from domain, which has maximum width.

4.5.3 Tables

Table 4.1: Tbl_userreg

This table used to store the basic details of user to create an account in WeShare.

FIELD NAME	DATA TYPE	CONSTRAINTS	DESCRIPTION
uid	varchar(50)	Primary Key	login-id of a user(Visitor)
fname	varchar(50)	NOT NULL	firstname of the visitor
lname	varchar(50)	NOT NULL	lastname of visitor
occupation	varchar(50)	NOT NULL	occupation of the visitor
gender	varchar(50)	NOT NULL	gender of visitor
email	varchar(50)	NOT NULL	Email id of the visitor
mob	varchar(50)	NOT NULL	contact Number of visitor
uname	varchar(50)	NOT NULL	username of the visitor
pass	varchar(50)	NOT NULL	password of visitor
status	varchar(max)	NOT NULL	status for the visitor

Table 4.2: Tbl.orgreg

This table used to store the basic details of organization to create an account in WeShare.

FIELD NAME	DATA TYPE	CONSTRAINTS	DESCRIPTION
oid	varchar(50)	Primary Key	login-id of a Organization
oname	varchar(50)	NOT NULL	Name of the Organization
otyp	varchar(50)	NOT NULL	Category of organization
oloc	varchar(50)	NOT NULL	Location of the Organization
ocontact	varchar(50)	NOT NULL	Contact of organization
email	varchar(50)	NOT NULL	Email id of the Organization
website	varchar(50)	NOT NULL	Website of organization
uname	varchar(50)	NOT NULL	username of the organization
pass	varchar(50)	NOT NULL	password of the organization
logo	varchar(50)		Logo of the organization
certificates	varchar(max)		proofs for the organization

Table 4.3: Tbl.Login

It is used to store the login credentials and the status of user accounts of all users of WeShare to login WeShare.

FIELD NAME	DATA TYPE	CONSTRAINTS	DESCRIPTION
uid	varchar(50)	Primary Key	login-id of a user
username	varchar(50)	NOT NULL	Username of the user
password	varchar(50)	NOT NULL	Password of the user
usertype	varchar(50)	NOT NULL	Type of user
LOGG	varchar(50)	NOT NULL	status of user

Table 4.4: Tbl_Profile

This table used to store the basic Profile details of user(visitors) in WeShare.

FIELD NAME	DATA TYPE	CONSTRAINTS	DESCRIPTION
pid	varchar(50)	Primary Key, Foreign Key(Tbl_login)	login-id of a user
prefix	varchar(50)		prefix of the visitor
name	varchar(50)	NOT NULL	fullname of visitor
address	varchar(50)	NOT NULL	address of the visitor
dob	varchar(50)	NOT NULL	date of birth of visitor
occupation	varchar(50)	NOT NULL	occupation of visitor
nation	varchar(50)	NOT NULL	nationality
blood	varchar(50)	NOT NULL	blood group of visitor
contact	varchar(50)	NOT NULL	contact number of visitor
email	varchar(50)	NOT NULL	email id of visitor
propic	varchar(max)		profile picture of visitor

Table 4.5: Tbl_ProfileJob

This table used to store the Job details of user(visitors) in WeShare.

FIELD NAME	DATA TYPE	CONSTRAINTS	DESCRIPTION
proid	varchar(50)	Primary Key, Foreign Key(Tbl_Login)	Login-id of a user
job	varchar(50)	NOT NULL	job of the visitor
category	varchar(50)	NOT NULL	category of visitor
company	varchar(50)	NOT NULL	company of visitor
salary	varchar(50)	NOT NULL	salary of visitor
location	varchar(50)	NOT NULL	Location of company situates
experience	varchar(50)	NOT NULL	work experience of the visitor

Table 4.6: Tbl_Accademics

This table used to store the basic Academic details of user(visitors) in WeShare.

FIELD NAME	DATA TYPE	CONSTRAINT	DESCRIPTION
pid	Varchar(50)	PK, FK	Id
Highqual	Varchar(50)		Highest qualification
Pg_course	Varchar(50)		Pg course
Pg_clg	Varchar(50)		College of pg course studied
Pg_status	Varchar(50)		Status of pg course
pg_yop	Varchar(50)		Year of passing
pg_perc	Varchar(50)		Percentage of pg course
ug_course	Varchar(50)		UG course
ug_clg	Varchar(50)		College of study
ug_yop	Varchar(50)		Year of passing ug course
ug_perc	Varchar(50)		Aggregate percentage og ug
ug_university	Varchar(50)		university
dip_course	Varchar(50)		Diploma course
dip_clg	Varchar(50)		College of diploma studied
dip_university	Varchar(50)		University of diploma
dip_yop	Varchar(50)		Year of completion
dip_perc	Varchar(50)		Aggregate percentage
hse	Varchar(50)		HSE course
hse_board	Varchar(50)		HSE board
hse_school	Varchar(50)		HSE school
hse_yop	Varchar(50)		HSE year of completion
hse_perc	Varchar(50)		Aggregate percentage
sslc_board	Varchar(50)		SSLC Board
sslc_school	Varchar(50)		SSLC school
sslc_yop	Varchar(50)		Year of completion
sslc_perc	Varchar(50)		Aggregate percentage

Table 4.7: Tbl_Requests

This table used to store the basic requests details of user(visitors) who is in need.

FIELD NAME	DATA TYPE	CONSTRAINTS	DESCRIPTION
reqid	varchar(50)	Primary Key	Request-id of a user
subject	varchar(50)		subject of the request
reqdate	varchar(50)	NOT NULL	date of request applied
request	varchar(50)	NOT NULL	content of the request
reqtype	varchar(50)	NOT NULL	category of the requests
cost	varchar(50)		amount required
closingdate	varchar(50)	NOT NULL L	deadline of the request
status	varchar(50)		status of the request processing
clientid	varchar(50)	Foreign Key(Tbl_Login)	id of user who is requesting
attachments	varchar(max)		supporting proofs for requests

Table 4.8: Tbl_Blood

This table used to store the basic blood details of user(visitors) interested in blood donation.

FIELD NAME	DATA TYPE	CONSTRAINTS	DESCRIPTION
proid	varchar(50)	Primary Key, Foreign Key(Tbl_Login)	User-id of a user
name	varchar(50)	NOT NULL	Name of the user
blood	varchar(50)	NOT NULL	Blood group of the user
email	varchar(50)	NOT NULL	Email id of the user
mob	varchar(50)	NOT NULL	Contact number of user
address	varchar(50)	NOT NULL	Address of the user

Table 4.9: Tbl_Post

This table used to store the post details of user(visitors) needs to post in the visitor's home-page.

FIELD NAME	DATA TYPE	CONSTRAINTS	DESCRIPTION
poid	varchar(50)	Primary Key	Post-id of a userpost
name	varchar(50)		Name of the user
uid	varchar(50)	Foreign Key(Tbl_Login)	userid of the user
[content]	varchar(50)		Content of the post
files	varchar(50)		images posted by the user
date	varchar(50)	NOT NULL	date of the post posted

Table 4.10: Tbl_JobVaccancy

This table used to store the details of job vaccancy that the organization posts for user(visitors) in WeShare.

FIELD NAME	DATA TYPE	CONSTRAINTS	DESCRIPTION
Oid	varchar(50)	Foreign Key(Tbl_Login)	Organization-id
Vid	varchar(50)	PK	Vacancy-id of the organization
job	varchar(50)		job title
post	varchar(50)		post of the vaccancy
category	varchar(50)		category of the vacancy
Address	varchar(50)		Address where company situates
Exp	varchar(50)		Experience Required
Salary	varchar(50)		Salary scale
Requirement	varchar(max)		Job requirements
noofvacancy	int		No. of vacancies available
status	varchar(50)		Status of the vacancy applied

Table 4.11: Tbl_ConductActivities

This table used to store the details of activities that tries to conduct.

FIELD NAME	DATA TYPE	CONSTRAINTS	DESCRIPTION
evntid	varchar(50)	Primary Key	Event-id of the user
eventname	varchar(50)		Name of the Event
eventtyp	varchar(50)		Type of the Event
Eventdetails	varchar(50)		Details of the Event
eventdate	varchar(50)		date of the event
eventplace	varchar(50)		place where event occurs
eventcoordinator	varchar(50)		Coordinator of the event
eventcoordmob	varchar(50)		Contact number of event co-ordinator
eventimages	varchar(max)		images to show

Table 4.12: Tbl_AppliedJobSeeker

This table used to store the details of user(visitors) who are applied for the job vacancies announced by the organizations in WeShare.

FIELD NAME	DATA TYPE	CONSTRAINTS	DESCRIPTION
userid	varchar(50)	Primary Key, Foreign Key(Tbl_Login)	User-id of a Visitor
orgid	varchar(50)	Foreign Key(Tbl_Login)	Id of the organization
jid	varchar(50)		job id of the organization
status	varchar(50)		status of the application

Table 4.13: Tbl_Account

This table used to store the basic bank account details of user(visitors) in WeShare.

FIELD NAME	DATA TYPE	CONSTRAINTS	DESCRIPTION
id	varchar(50)	Primary Key, Foreign Key(Tbl_Login)	User-id of a Visitor
bankid	varchar(50)	Foreign Key(Tbl_Bank)	Id of the bank
branchid	varchar(50)	Foreign Key(Tbl_Branch)	id of the branch
cardno	varchar(50)	NOT NULL	Card number
cardholder	varchar(50)	NOT NULL	Card holder
cvv	varchar(50)	NOT NULL	cvv of account
expmonth	varchar(50)	NOT NULL	expiry month of the card
expyear	varchar(50)	NOT NULL	expiry year of the card
amount	varchar(50)	NOT NULL	balance amount in the account

Table 4.14: Tbl_Bank

This table used to store the bank details so that to create a dummy banking process.

FIELD NAME	DATA TYPE	CONSTRAINTS	DESCRIPTION
bankid	varchar(50)	Primary Key	id of the bank
bankname	varchar(50)	NOT NULL	NAme of the bank
banklogo	varchar(50)	NOT NULL	Logo of the bank

Table 4.15: Tbl_Branch

This table used to store the bank branch details so that to create a dummy banking process.

FIELD NAME	DATA TYPE	CONSTRAINTS	DESCRIPTION
branchid	varchar(50)	Primary Key	id of the branch
bankid	varchar(50)	Foreign Key(Tbl_Bank)	id of the bank
branchname	varchar(50)	NOT NULL	Name of the branch
address	varchar(50)	NOT NULL	address of the branch
phone	varchar(50)	NOT NULL	contact number of the branch
ifsc	varchar(50)	NOT NULL	ifsc code

4.6 USER-INTERFACE DESIGN

User interface is the front-end application view to which user interacts in order to use the software. User can manipulate and control the software as well as hardware by means of user interface. Today, user interface is found at almost every place where digital technology exists, right from computers, mobile phones, cars, music players, airplanes, ships etc.

User interface is part of software and is designed such a way that it is expected to provide the user insight of the software. UI provides fundamental platform for human-computer interaction.

UI can be graphical, text-based, audio-video based, depending upon the underlying hardware and software combination. UI can be hardware or software or a combination of both.

The software becomes more popular if its user interface is:

- Attractive
- Simple to use
- Responsive in short time
- Clear to understand
- Consistent on all interfacing screens

4.6.1 The UI Screen shots

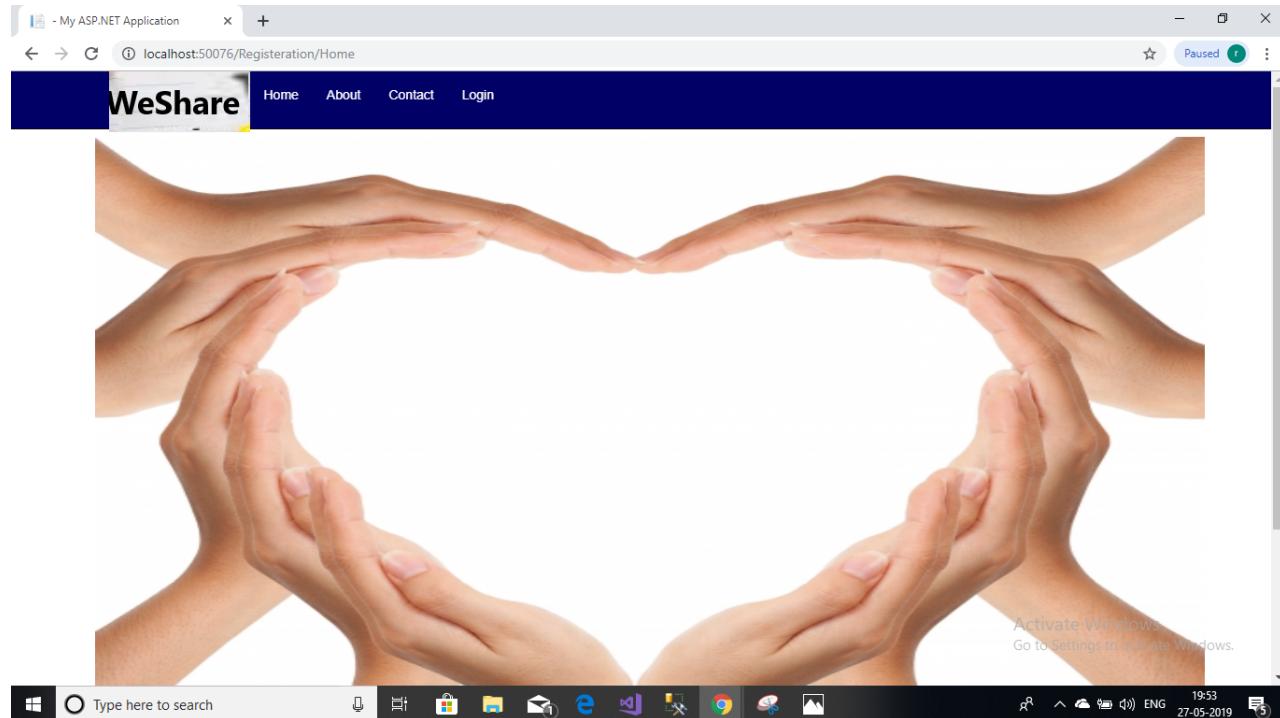


Figure 4.1: Screenshot 1:HomePage(First Page)

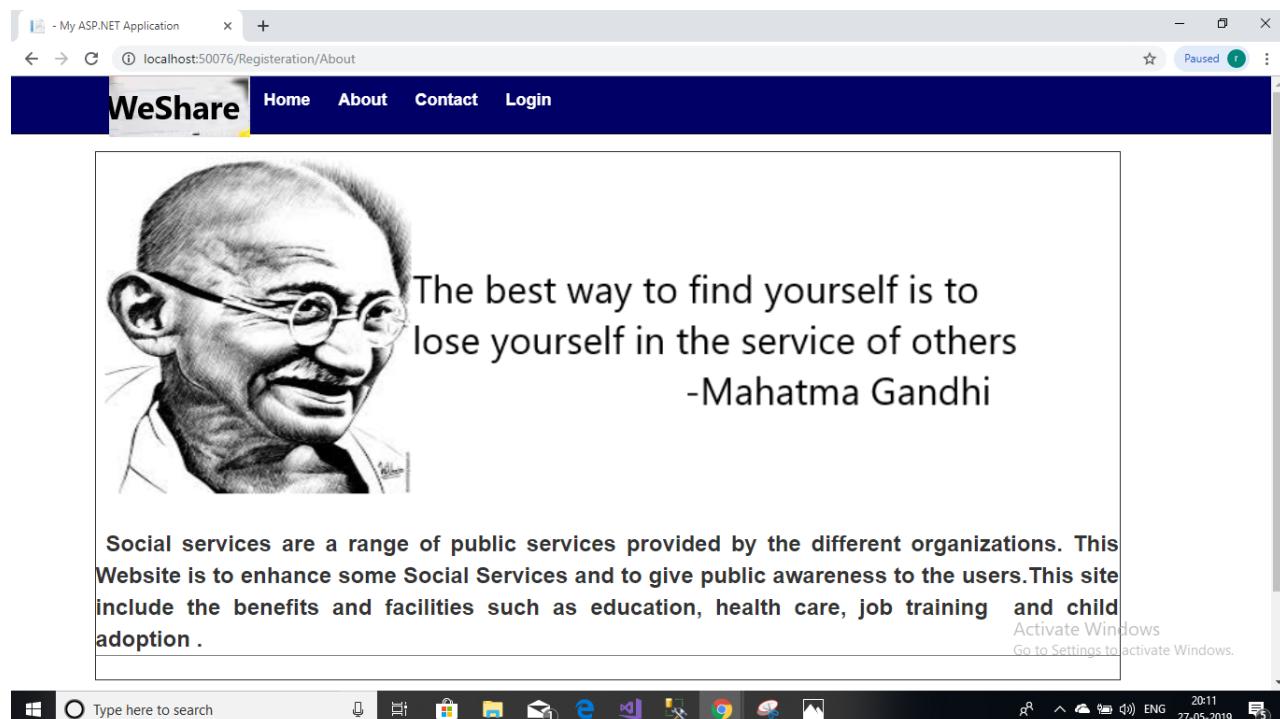


Figure 4.2: Screenshot 2:About

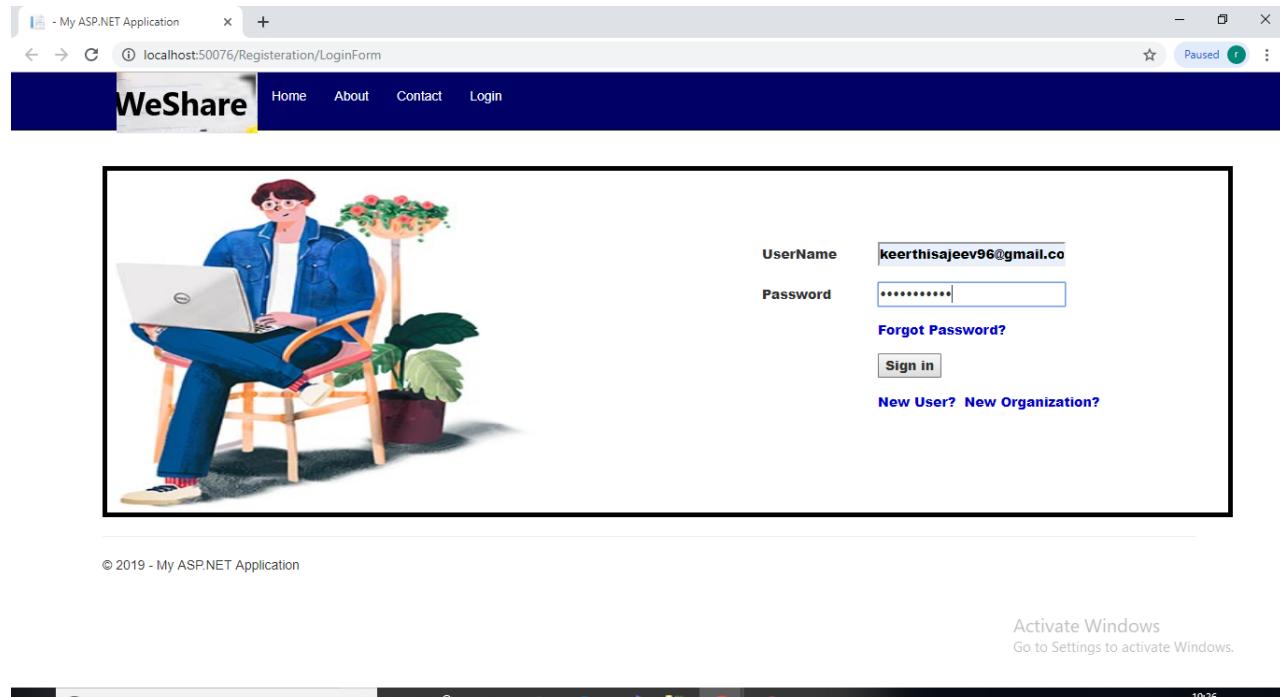


Figure 4.3: Screenshot 3:Login

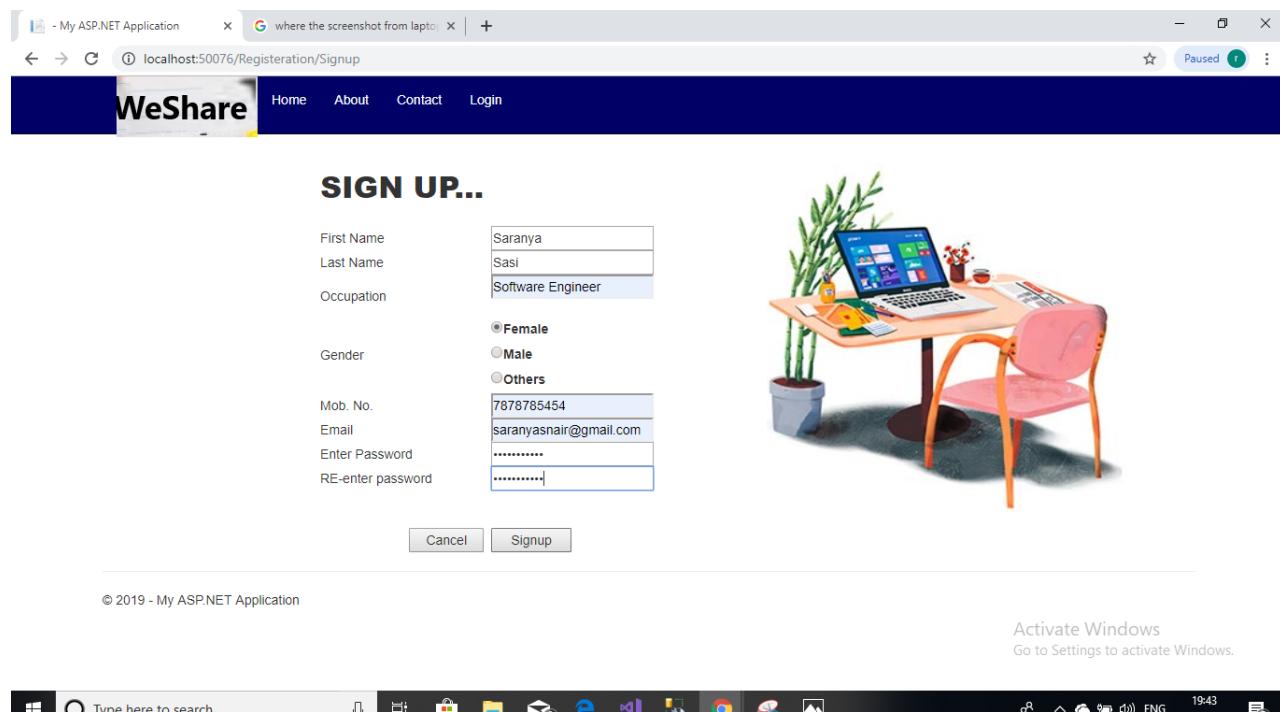


Figure 4.4: Screenshot 4:Visitor Registration

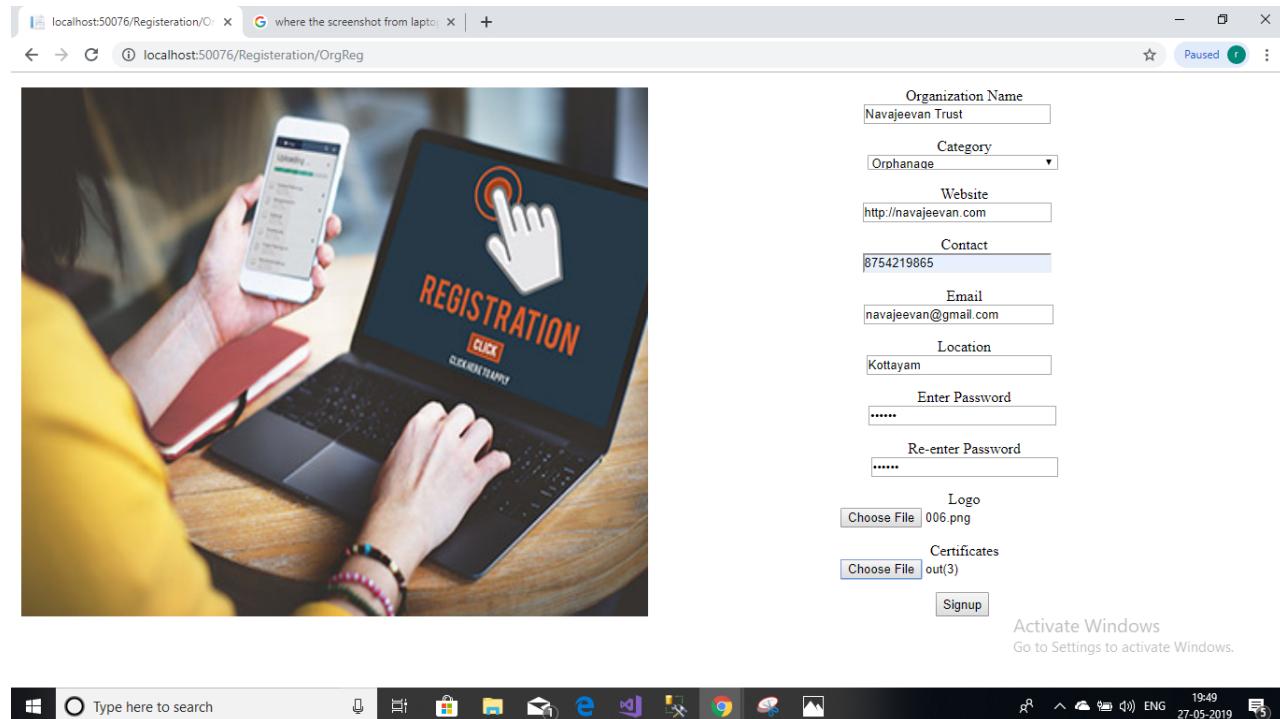


Figure 4.5: Screenshot 5:Organization Registration

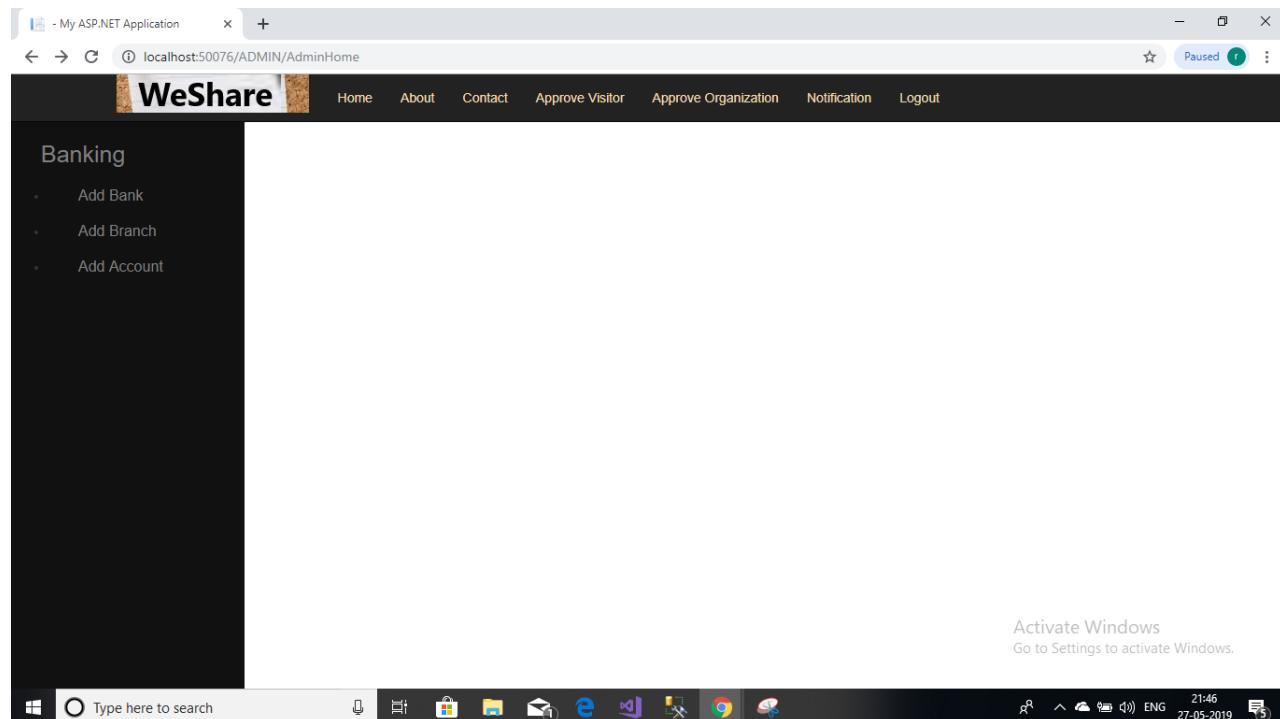
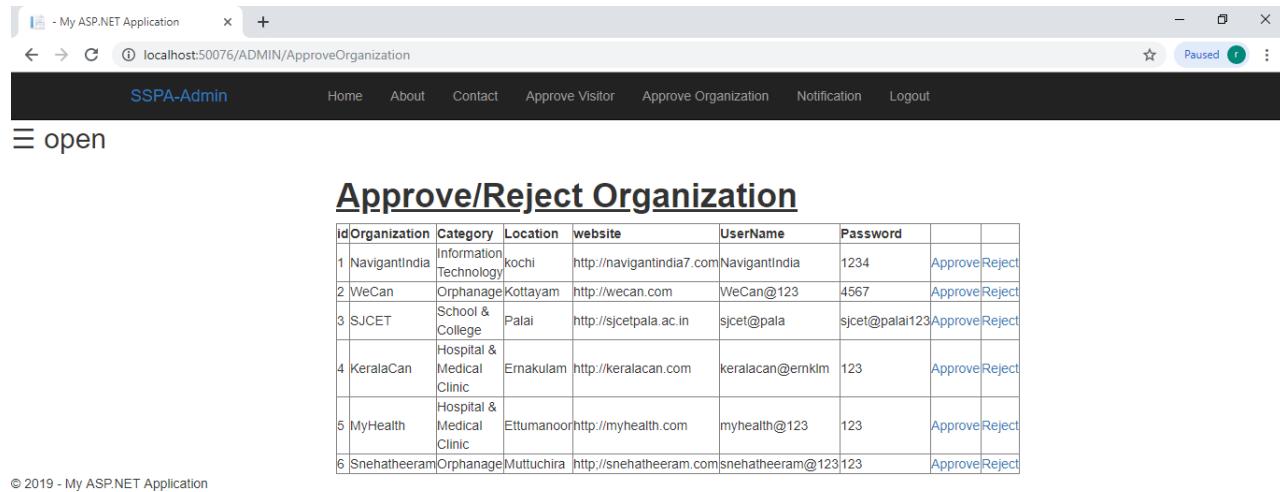
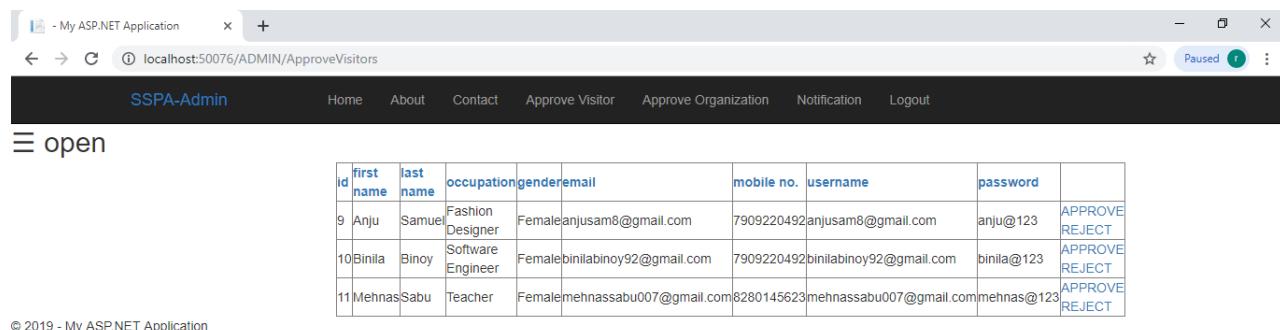


Figure 4.6: Screenshot 6:Admin Home Page

**Figure 4.7:** Screenshot 7:Approve Organization**Figure 4.8:** Screenshot 8:Approve Visitor

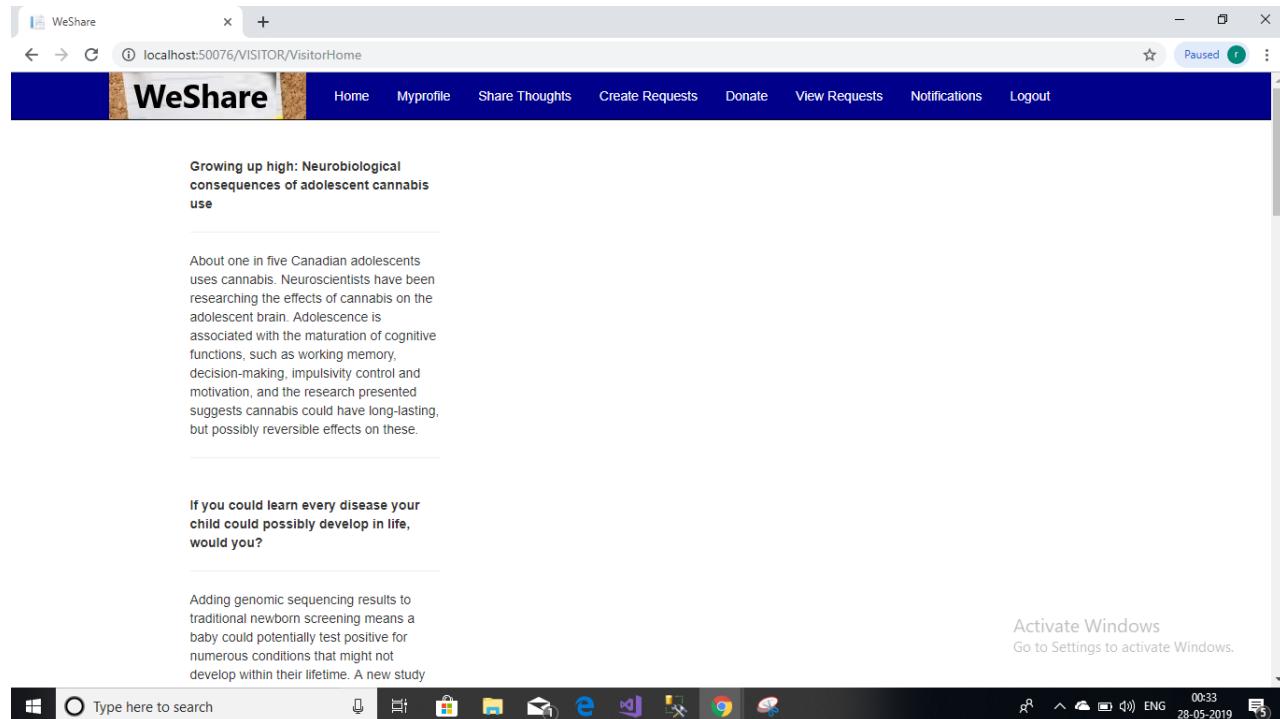


Figure 4.9: Screenshot 9:Visitor Home Page

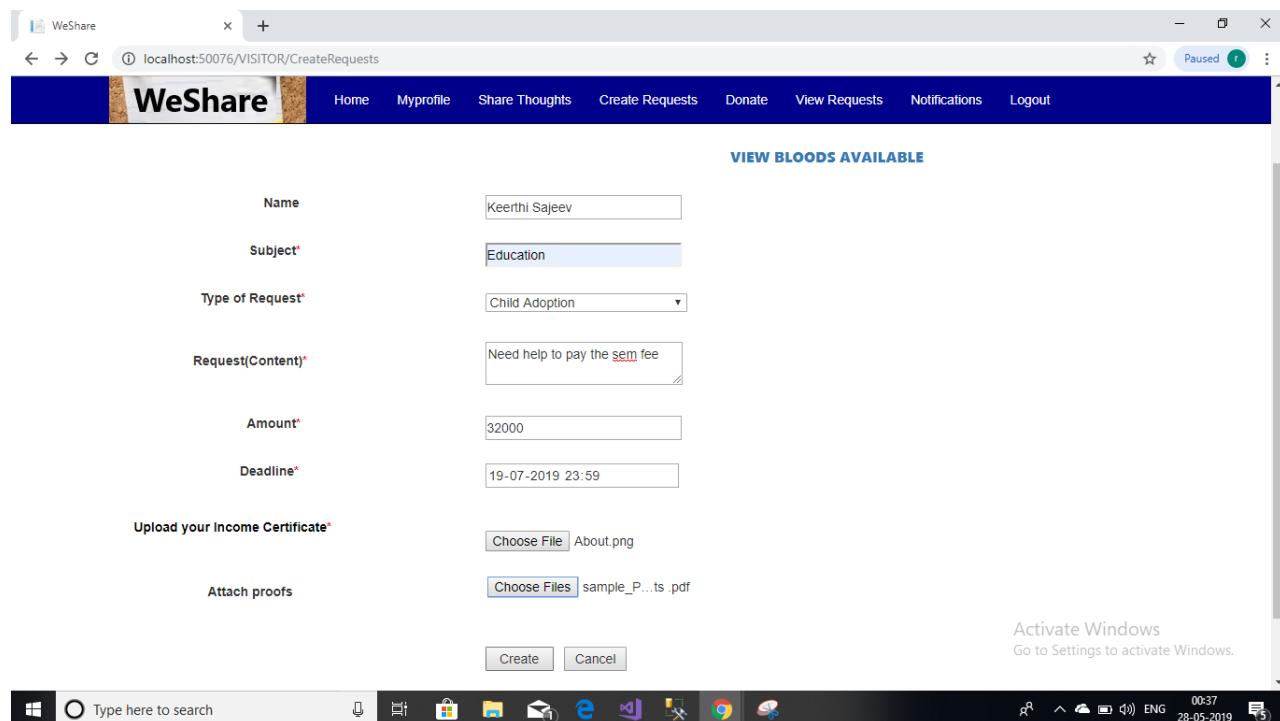


Figure 4.10: Screenshot 10:Create

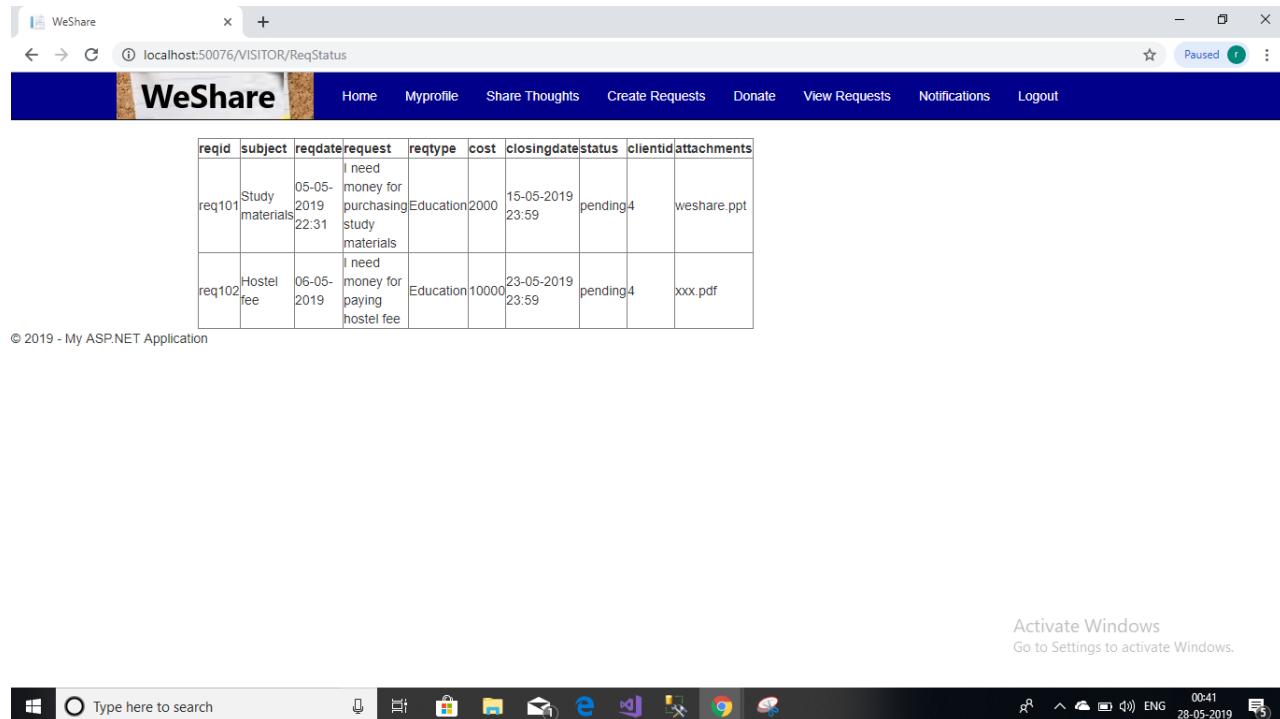


Figure 4.11: Screenshot 11:View Requests

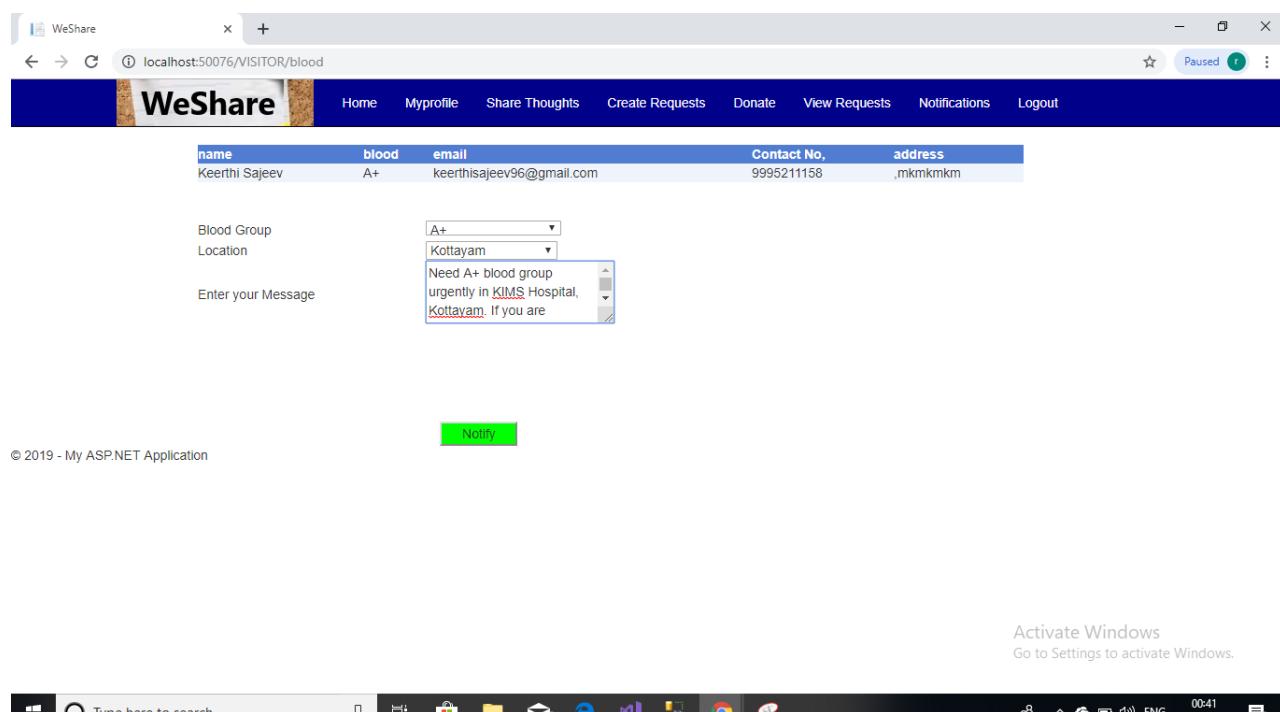


Figure 4.12: Screenshot 12:View blood donors Available

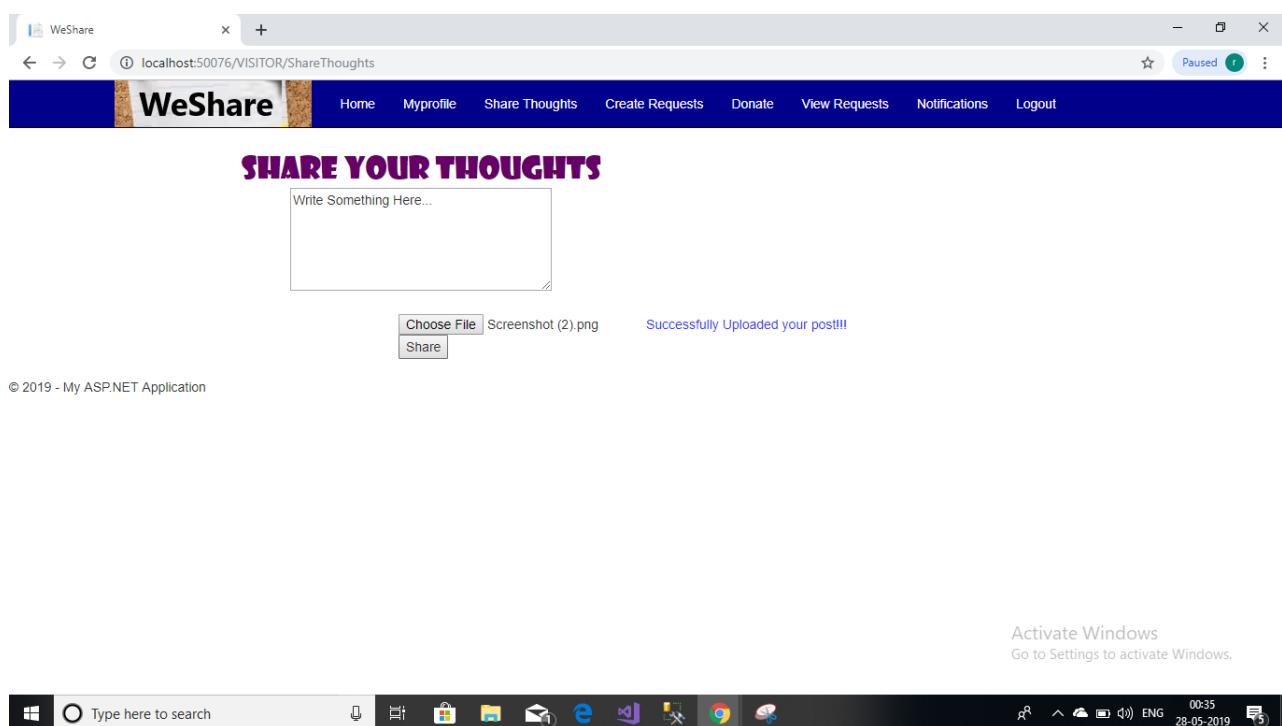


Figure 4.13: Screenshot 13:Share your thoughts

Chapter 5

SYSTEM TESTING

5.1 Introduction

After design and development of the web application,it is relevant to test with the real data to pullout the errors,mistakes and problems hidden with the website.Outcomes of the website are evaluated and compared with the target outputs.While testing system,a clear picture about how the application is going to be used and what kind of issues it can face in real time can be really helpful and able rectify problems so it can implement properly.The entire testing can be divided into 3 phases.

5.2 Unit Testing

Unit testing focuses verification efforts on the smallest of software design,the module.This is also known as module testing.The four modules Login, Employee,Notification, Task and Reports are tested separately.These testing are carried out during programming stage itself.The outcomes are compared with the requirements and was found satisfactory.This,it is possible to conclude that every program in the software was functioning correctly such as data correctly stored in the database,proper working of mail function,working of combo boxes,proper working of navigation through links,proper generation of pdf documents etc.I followed a sequential order in the unit testing process.Whenever a function implemented ,it must be verified and only that move on to the next task.Also whenever a

module completed, its all individual functions tested separately. Functions of user interface design tested at last iteration.

5.3 Integration Testing

In integration testing systems ability to maintain data integrity and operation in coordination with other systems in same and different environment is being checked. Some are: the website opens properly with all the relevant pages and images, checks login properly, whether the number of users can access the website simultaneously, if the website works properly in all the major browsers and their latest versions, if the process are being done on the website via specific user are secure enough, if the content of pages are properly aligned, and well managed if session is implemented and working as expected.

5.4 Acceptance Testing

User acceptance testing is a key factor that counts the success of the system. Web apps are now very familiar to majority of users. So its acceptance were tested by admin at the time of developing and making changes whenever required. This done with regard to the following points.

- Input and output screen design
- Speed and accuracy
- Simplicity.
- Menu driven system.

All these have been successfully accepted by the users, because system itself keep the format and navigation of the requirement of user.

Test Cases

Table 5.1: Test case:1(Login Page)

Name of control	Validation	Inputs	Response
text box(User name)	It should be a registered email id	ranimol123@gmail.com	success
		abc.gmail.com	Invalid Entry
text box(Password)	Same as that send to mail	ranimol123	success
		aaa	Invalid Entry

Table 5.2: Test case 2: Share Thoughts Form

Name of Control	Validation	Input	Response
Text Box(Content)	Write the thoughts	null	Mandatory
		Public awareness of the environment means the ability to emotionally understand the surrounding world, including the laws of the natural environment, sensitivity to all the changes occurring in the environment, understanding of cause-and-effect relationships between the quality of the environment and human behaviour,...	Success
File upload(Images)	Select one or more images	image.pdf	Invalid
		image.jpg	Success

Table 5.3: Test case 3: Add Request Form

Name of Control	Validation	Input	Response
Text Box(Name)	Alphabets only	***	Not a valid name
		Arya	Success
Text Box(Subject)	Required	null	Mandatory Field
		Education Accessories	Success
Drop Down List(Type of Request)	Required	null	Mandatory Field
		Education	Success
Text Box(Request)	Required	null	Mandatory
		I want help for the education needs....	Success
Text Box(Amount)	Must enter an amount	null	Mandatory
		1000	Success
Text Box(Deadline)	Must select a date	monday	Invalid
		null	Mandatory
		22/08/2019	Success
FileUpload(Proofs)	Must select a file	null	Mandatory
		proof.pdf	Success

Table 5.4: Test case 4: Conduct Activities

Name of Control	Validation	Input	Response
Text Box(EventName)	Required	null	Mandatory
		Event@xyz	Success
Text Box(Event type)	Required	null	Mandatory Field
		Education	Success
Text Box(Event date)	Required	null	Mandatory Field
		Education	Invalid
		22/07/2019 23:59:59	Success
Text Box(Event details)	Required	null	Mandatory Field
		Education.....	Success
Text Box(Event Location)	Required	null	Mandatory Field
		Kochi	Success
Text Box(Event coordinator)	Required	null	Mandatory Field
		Felix Alex	Success
Text Box(Contact)	Valid phone number	123456789	Invalid Field
		9958565225	Success
File Upload(image)	Select image only	123456789.pdf	Invalid Format
		ranimol.jpg	Success

Chapter 6

SYSTEM IMPLEMENTATION

6.1 Implementation Methods

Implementation is the stage in the project where the theoretical design is turned into a working system and is giving confidence on the new system for the users that it will work efficiently and effectively. Web hosting simply means making a website available on the world wide web such that people are able to view the content you have on your website. A web site can be developed on any computer including your personal computer if you install and configure the right software. However, that will not make a website available via the internet. Therefore, to make it available via the internet, the files that make up the website must be uploaded or copied to a special type of computer called a web server. Once this is done, the website will be available via the internet and is called web hosting. The first thing to consider when starting a website is to choose a web hosting provider (Eg. Go daddy.com, HostGator etc.). The web hosting provider provides the web space (i.e. special computers called web servers) where your website files are stored, as well as the technologies and services needed for your website to be viewed on the Internet.

6.2 Implementation Plan

The implementation process begins with preparing a plan for the implementation of the system. According to this plan, the activities are to be carried out, discussions made

regarding the equipment and resources and the additional equipment has to be acquired to implement the new system. In network backup system no additional resources are needed. Implementation is the final and the most important phase. The most critical stage in achieving a successful new system is giving the users confidence that the new system will work and be effective. The system can be implemented only after thorough testing is done and if it is found to be working according to the specification. This method also offers the greatest security since the old system can take over if the errors are found or inability to handle certain type of transactions while using the new system. The implementation process consist of the following steps:

- List all files required for implementation
- Identify all data required to build new files
- List all documents and procedures that go in to the new system.

Chapter 7

CONCLUSION AND FUTURE SCOPE

The project is a charity based solution for poor persons. Currently most of the social services are facing difficulties in reaching to the right needy ones. This system can speed up the needy to send their needs to others faster than by vocal medium. The project is helpful for an individual at the same time for an organization too. They can add their vacancies, etc.

It is the first phase of the project WeShare. As a future scope, it can add more features such as providing money lending facility, approve and check requests and users using data mining techniques etc. The Social Service can be extended to all kind of users inside and outside an organization in the future.

The future of WeShare can include artificial intelligence, big data technologies. AI has the potential to make every transaction smarter. AI will power up our application to learn from our interactions through data, help us find answers to our questions, predict our needs, and it will take care of tedious and forgotten tasks so we can focus the important ones.

Chapter 8

SAMPLE CODE

SAMPLE CODE

8.1 LoginPage

```
using System;
using System.Collections.Generic;
using System.Linq;
using WeShare_SSPA_Class;
using System.Data;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
namespace WeShare_SSPA_.Registration {
    public partial class LoginForm : System.Web.UI.Page {
        Signin log = new Signin();
        protected void Page_Load(object sender, EventArgs e) {
        }
        protected void BtnSigninClick1(object sender, EventArgs e)
    {
```

```
DataTable dtReg = new DataTable();
if(Page.IsValid){
    log.Uname = TextBox1.Text;
    log.Paswd = txt_pass.Text;
    dtReg = log.CommonLogin();
    intRowCount = dtReg.Rows.Count;
    for(int i = 0; i < RowCount; i++)
    {
        string usernam = dtReg.Rows[i]["username"].ToString();
        stringpass = dtReg.Rows[i]["password"].ToString();
        stringtyp = dtReg.Rows[i]["usertype"].ToString();
        stringstatus = dtReg.Rows[i]["LOGG"].ToString();
        if(usernam == log.Uname& pass == log.Paswd)
        {
            Session["userid"] = dtReg.Rows[i]["userid"].ToString();
            if(status != "0")
            {
                if(dtReg.Rows[i]["usertype"].ToString() == "Admin")
                {
                    Response.Redirect("~/ADMIN/AdminHome.aspx");
                }
                elseif(dtReg.Rows[i]["usertype"].ToString() == "User.Organization")
                {
                    Response.Redirect("~/ORGANIZATION/OrgHome.aspx");
                }
            }
        }
    }
    Response.Redirect("~/VISITOR/VisitorHome.aspx");
}
```

```
else
{
if (dtReg.Rows[i]["usertype"].ToString() == "Admin")
{
Response.Write("Script\;alert('Access Denied')\;Script\;");
}
else if
(dtReg.Rows[i]["usertype"].ToString() == "User.Organization")
{
Response.Write("Script\;alert('Access Denied')\;Script\;");
}
else
{
Response.Redirect("~/VISITOR/Profile.aspx");
}
}
}
}
}
}
}
```

8.2 Create Request Page

```
using System;
using System.Data;
using WeShareSSPA Class;
namespace WeShareSSPA.VISITOR
```

```
{  
    public partial class CreateRequests : .Web.UI.Page  
{  
    protected void PageLoad(object sender, EventArgs e)  
{  
        VISITORActivities act = new VISITORActivities();  
        DataTable dataTab = new DataTable();  
        string name, id;  
        if (Session["userid"].ToString() != null)  
        {  
            id = (string)Session["userid"];  
            dataTab = act.ViewPersonalProfile(id);  
            int rc = dataTab.Rows.Count;  
            for (int j = 0; j < rc; j++)  
            {  
                if (dataTab.Rows[j]["pro_id"].ToString() == (string)Session["userid"])  
                {  
                    name = dataTab.Rows[j]["name"].ToString();  
                    txtclient.Text = name;  
                }  
            }  
        }  
        protected void Button1Click(object sender, EventArgs e)  
        {  
            VISITORActivities act = new VISITORActivities();  
            DataTable dataTab = new DataTable();  
            if (filuplproof.HasFiles)  
            {  
                ...  
            }  
        }  
    }  
}
```

```
filuplproof.SaveAs(Server.MapPath(" /proofs/" + filuplproof.FileName));
act.Proofs = filuplproof.FileName.ToString();
}

act.Clsdate = txtDeadline.Text;
act.Cost = txtamnt.Text;
act.Reqdate = System.DateTime.Now.ToString();
act.Subject = txtreqsubject.Text;
act.Reqtype = DropDownList1.SelectedValue;
act.Status = "Waiting for Approval";
act.Request = txtreq.Text;
act.Clntid = Session["userid"].ToString();
act.CreateRequest();
Response.Write("";script;alert('Request uploaded!');script;");

}

protected void DropDownList1_SelectedIndexChanged(object sender, EventArgs e)
{
if(DropDownList1.SelectedItem.Text == "Education Needs")
{
LabelHelp.Text = "Proofs that must be attached are 1. Certificate from authority 2. Documents";
}
else if(DropDownList1.SelectedItem.Text == "Health Care Needs")
{
LabelHelp.Text = "Proofs that must be attached are
1. Doctor's / Physician's proof
2. Doctor's / Physician's Contact 3. Income certificate";
}
else if(DropDownList1.SelectedItem.Text == "Child Adoption")
{
LabelHelp.Text = "";
}
```

```
    }
    else
    {
        LabelHelp.Text = "1. Documents that can prove the address, income, education details";
    }
}
}
}
}
```

8.3 Class File:Visitor-Activities

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Data;
using System.Data.SqlClient;
using System.Web.UI.WebControls;
namespace WeShareSSPA.Class
{
    public class VISITORActivities
    {
        string ConnectionString = System.Configuration.ConfigurationManager.ConnectionStrings["SqlConnectionString"];
        SqlConnection con;
        public SqlCommand cmd;
        public SqlDataReader dr;
        public SqlDataAdapter ad = new SqlDataAdapter();
        public DataTable dt = new DataTable();
        public DataSet ds = new DataSet();
        public void OpenConection()
```

```
{  
con = new SqlConnection(ConnectionString);  
con.Open();  
}  
  
public void CloseConnection()  
{  
con.Close();  
}  
  
public void ExecuteQueries(string Query)  
{  
SqlCommand cmd = new SqlCommand(Query, con);  
cmd.ExecuteNonQuery();  
}  
  
protected void GetConnection()  
{  
if (con.State == ConnectionState.Open)  
{  
con.Close();  
}  
con.ConnectionString = System.Configuration.ConfigurationManager.ConnectionStrings["myConn"].  
ConnectionString;  
con.Open();  
}  
  
public void ReadData(string sql)  
{  
GetConnection();  
cmd = new SqlCommand(sql, con);  
cmd.ExecuteReader();  
}
```

```
public void WriteData(string sql)
{
    GetConnection();
    cmd = new SqlCommand(sql, con);
    cmd.ExecuteNonQuery();
}

public void FillDataList(DataList datlst)

DataTable forreq = new DataTable();
DataTable fordon = new DataTable();

string SQL = "select subject,reqdate,request,reqtype,cost,closingdate from Request
where status='pending' orderby reqdate desc";
GetConnection();
cmd = new SqlCommand(SQL, con);
ad.SelectCommand = cmd;
ad.Fill(forreq);
if (forreq.Rows.Count > 0)
{
    datlst.DataSource = forreq;
    datlst.DataBind();
}
private string reqid;
private string subject;
private string reqdate;
private string request;
private string reqtype;
private string cost;
private string clsdate;
```

```
private string status;
private string clntid;
private string proofs;
    private string uid;
private string name;
private string content;
private string files;
private string poid;
private string date;
    private string blood;
private string add;
    public string Reqid { get =< reqid; set =< reqid = value; }
public string Subject { get =< subject; set =< subject = value; }
public string Reqdate { get =< reqdate; set =< reqdate = value; }
public string Request { get =< request; set =< request = value; }
public string Reqtype { get =< reqtype; set =< reqtype = value; }
public string Cost { get =< cost; set =< cost = value; }
public string Clsdate { get =< clsdate; set =< clsdate = value; }
public string Status { get =< status; set =< status = value; }
public string Clntid { get =< clntid; set =< clntid = value; }
public string Proofs { get =< proofs; set =< proofs = value; }
public string Uid { get =< uid; set =< uid = value; }
public string Name { get =< name; set =< name = value; }
public string Content { get =< content; set =< content = value; }
public string Files { get =< files; set =< files = value; }
public string Date { get =< date; set =< date = value; }
public string Poid { get =< poid; set =< poid = value; }
public string Blood { get =< blood; set =< blood = value; }
public string Add { get =< add; set =< add = value; }
```

```
public SqlDataAdapter da = new SqlDataAdapter();
public void adapter(string sql)
{
    dt.Clear();
    OpenConection();
    SqlCommand cmd = new SqlCommand(sql, con);

    da.SelectCommand = cmd;
    da.Fill(dt);
}

public void FillDropDownList(string Text, string Value, string Table, string Condition,
DropDownList ddlst)
{
    if (Condition == "")
    {
        string sql = "select " + Value + "," + Text + " from " + Table;
        adapter(sql);
        if (dt.Rows.Count > 0)
        {
            ddlst.DataSource = dt;
            ddlst.DataTextField = Text;
            ddlst.DataValueField = Value;
            ddlst.DataBind();
        }
    }
    else
    {
        string sql = "select " + Text + "," + Value + " from " + Table + " where " + Condition + "";
    }
}
```

```
adapter(sql);
if (dt.Rows.Count < 0)
{
    ddlst.DataSource = dt;
    ddlst.DataTextField = Text;
    ddlst.DataValueField = Value;
    ddlst.DataBind();
}
}
ddlst.Items.Insert(0, "—Select—");
}

public void FillGrid(string sql, GridView grdwv)
{
    adapter(sql);
    if (dt.Rows.Count < 0)

    {
        grdwv.DataSource = dt;
        grdwv.DataBind();
    }
}

public void CreateRequest()
{
    OpenConection();
    SqlCommand command = new SqlCommand("select reqid from Request order by reqid
desc ", con);
    object cMax = command.ExecuteScalar();
    if (cMax != DBNull.Value)
    {
```

```
reqid = (string)cMax;
int pos = reqid.IndexOf("req");
if (pos > 0)
{
    string p = reqid.Remove(pos);
    int id = Convert.ToInt32(p);
    id++;
    Reqid = "req" + id.ToString();
}
else
{
    Reqid = "req" + 101;
}

SqlCommand cmd = new SqlCommand("insert into Request values(@reqid,@sub,
@rdqte,@req,@type,@cost,@deadline,@status,@clientid,@proofs where clientid=@clntid;)",
con);
cmd.Parameters.AddWithValue("@reqid", Reqid);
cmd.Parameters.AddWithValue("@sub", Subject);
cmd.Parameters.AddWithValue("@rdate", Reqdate);
cmd.Parameters.AddWithValue("@req", Request);
cmd.Parameters.AddWithValue("@type", Reqtype);
cmd.Parameters.AddWithValue("@cost", Cost);
cmd.Parameters.AddWithValue("@deadline", Clsdate);
cmd.Parameters.AddWithValue("@status", Status);
cmd.Parameters.AddWithValue("@clientid", Clntid);
cmd.Parameters.AddWithValue("@proofs", Proofs);
cmd.ExecuteNonQuery();
}
```

```
public void Donate()
{
    OpenConection();
    } DataTable ViewPersonalProfile(string id)
{
    DataTable Profile = new DataTable();
    OpenConection();
    SqlCommand cmd = new SqlCommand("select * from Profile where pro_id = @proid", con);
    cmd.Parameters.AddWithValue("@proid", id);
    cmd.ExecuteNonQuery();
    SqlDataAdapter da = new SqlDataAdapter(cmd);
    da.Fill(Profile);
    CloseConnection();
    return Profile;
}

public DataTable BloodDetails()
{
    DataTable blood = new DataTable();
    OpenConection();
    SqlCommand cmd = new SqlCommand("select * from blood where blood=@blood and
address like
cmd.Parameters.AddWithValue("@blood",Blood);
cmd.Parameters.AddWithValue("@address", Add);
cmd.ExecuteNonQuery();
    SqlDataAdapter da = new SqlDataAdapter(cmd);
    da.Fill(blood);
    CloseConnection();

    return blood;
```

```
    }

    public void SharePost()
    {
        OpenConection();
        SqlCommand command = new SqlCommand("select max(poid) from post ", con);
        int poid;
        object cMax = command.ExecuteScalar();
        if (cMax != DBNull.Value)
        {
            poid = (int)cMax;
            poid++;
        }
        else
        {
            poid = 1101;
        }
        SqlCommand cmd = new SqlCommand("insert into post values(@id,@name,@uid,@content,
        @files,@date)", con);
        cmd.Parameters.AddWithValue("@id", poid);
        cmd.Parameters.AddWithValue("@name", Name);
        cmd.Parameters.AddWithValue("@uid", Uid);
        cmd.Parameters.AddWithValue("@content", Content);
        cmd.Parameters.AddWithValue("@files", Files);
        cmd.Parameters.AddWithValue("@date", Date);
        cmd.ExecuteNonQuery();
    }
}
```

REFERENCES

- [1] *Software Engineering* , Rodger S. Pressman,Tata McGraw Hill,Fifth Edition,Year 2004.
- [2] *Database Management Systems*, Reghu Ramakrishnan,Johness Gehrke.
- [3] "Agile Software Development,Principles Patterns and Practices",Robert C Martin,Prentice Hall Imprint,Pearson Education,2nd Edition,Year 2002
- [4] https://en.wikipedia.org/wiki/Web_hosting_service[online]
- [5] "ASP.NET: The Complete Reference"by, Matthew MacDonald
- [6] *Pro C# 7: With .NET and .NET Core* by Andrew Troelsen, Philip Japikse
- [7] *Microsoft SQL Server 2016: A Beginner's Guide, Sixth Edition* ,by Dusan Petkovic,6th Edition.
- [8] <http://www.quora.com>
- [9] *SQL : The Complete Beginner's Guide - Step By Step Instructions (The Black Book)* Kindle Edition by Byron Francis
- [10] *An Introduction to Database Systems*,Eighth edition by C.J. Date