APPENDIX B

DEVELOPMENT OF 'RED DROP' USING WEB APPLICATION DEVELOPMENT LIFE CYCLE FOR SMALL MEDIUM – SIZED ENTERPRISES PROPOSED BY WEI HUANG USING UML

Development of 'Red Drop' using Web application Development Life Cycle for Small Medium – Sized Enterprises Proposed by Wei Huang Using UML

Wei Huang et al. proposed a development approach for developing web applications for SMEs in 2008. They divided the process in three different parts, two of those are iterative and one is sequential as shown in Figure 1.

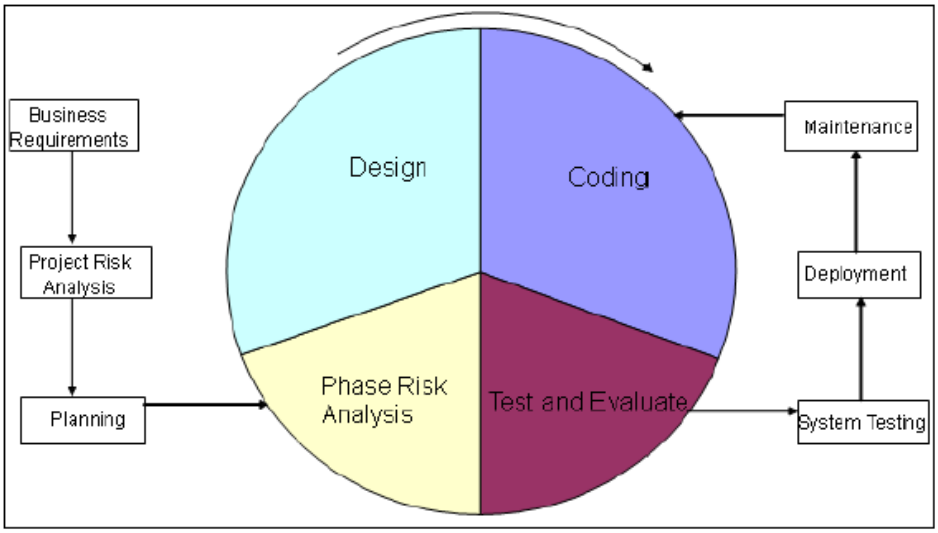

Figure 1 Web Application Development Lifecycle for SMEs [53]

First part i.e., *Sequential Development,* includes requirement analysis, risk analysis and planning for next phases. The overall system development is divided into some phases with each phase giving an executable output. The sequential part is performed for once and for whole system. In *Planning* phase, based on functions and components identified of the project, the development process is divided into some phases with each phase giving executable product. Next part is iterative development process including phase risk analysis, design, coding, and unit testing. Once the module identified for this cycle gets completed, it is fed for system testing which is part of last phase, maintenance phase. In this phase, after successful completion of system testing, module is deployed for use by user. The important thing is that modules are divided in such a manner that after completion of both cycles, executable product could be got. If requirement gets changed or need of change is identified then that need is directly coded without making changes in design documents and without analysing phase risk in order to reduce time consumption in the phase iteration.

SEQUENTIAL PROCESS:

1. BUSINESS REQUIREMENTS

Requirements for *Red Drop* are gathered with use of use-case diagrams. The output of this step is following three documents: ER Diagram, Use Case Diagrams, and SRS (Software Requirement Specification).

ER DIAGRAM

ER Diagram for *Red Drop* shows two entities *admin and donor* and a relationship *manages.* Admin have all rights for accessing the entries in database.

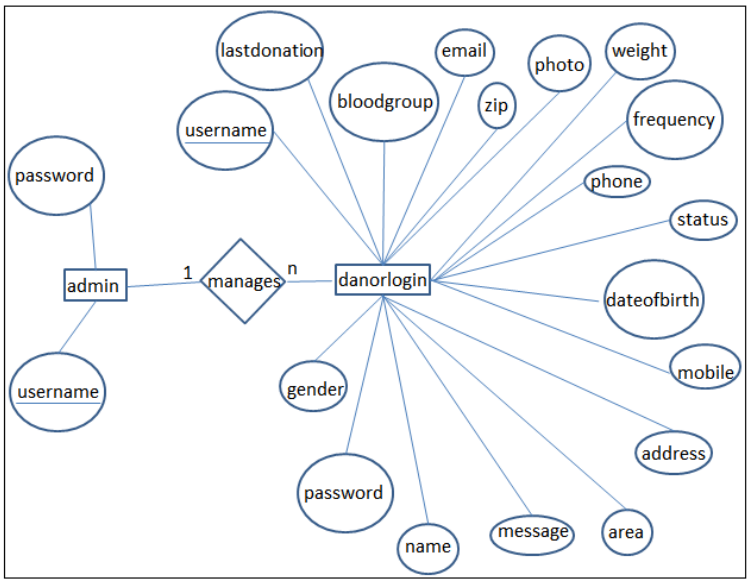

Figure 2 ER Diagram for *Red Drop*

USE CASE DIAGRAMS

In order to start eliciting the requirements, actors are identified. And for this purpose, team formation is done as shown in figure 3.

Actors identified are as follows:

1. Registered User
2. Unregistered User
3. HomePage Development Team
4. Team 1
5. Team 2
6. Team 3
7. Database Management Team
8. Data Collection Team

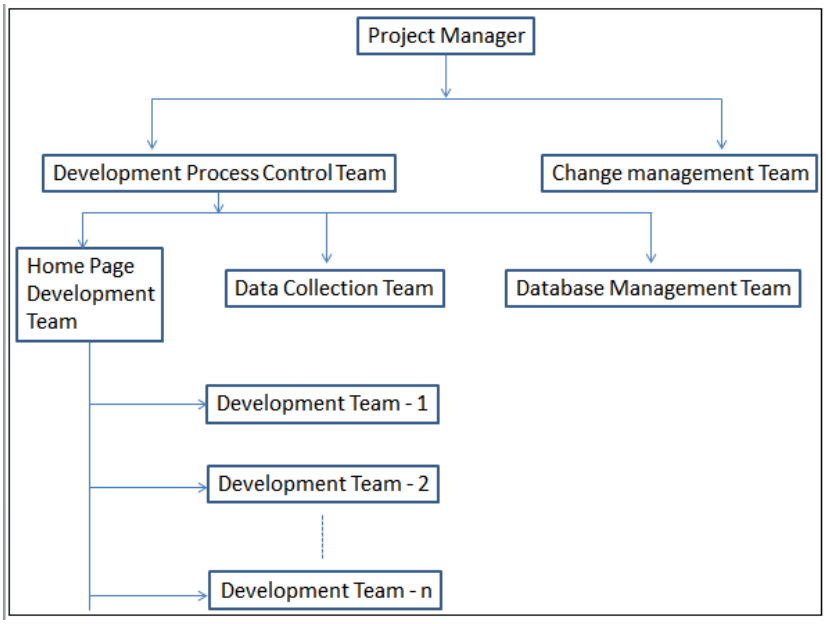

Figure 3 Team Management Chart

Use Case Diagram 1: HomePage

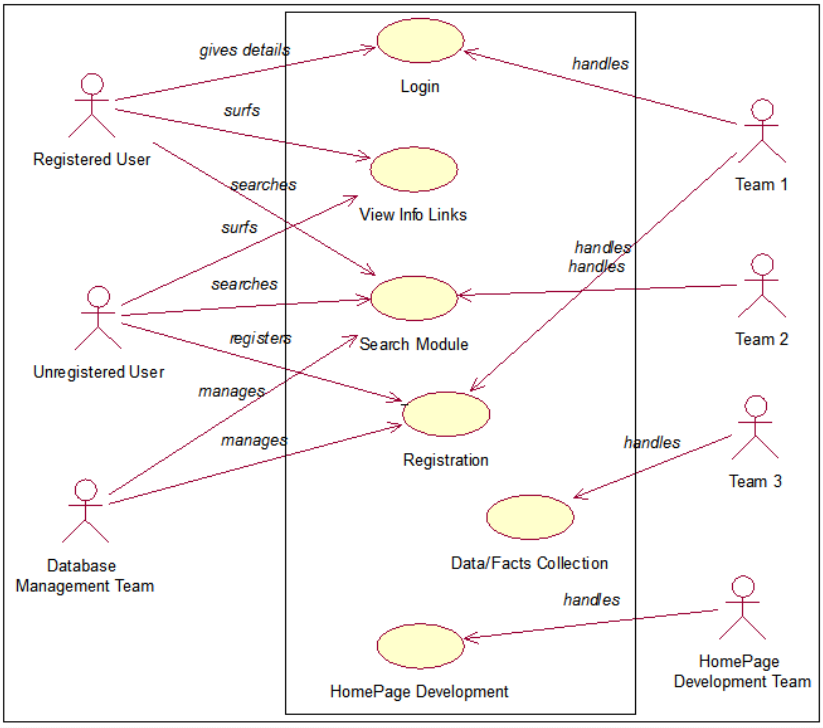

Figure 4 HomePage Use Case Diagram

Use Case Diagram 2: Registration

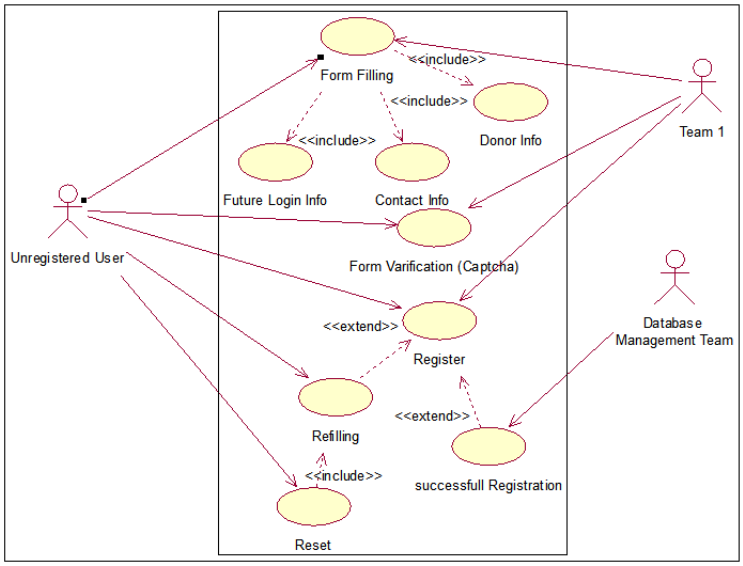

Figure 5 Registration Use Case Diagram

Use Case Diagram 3: Login

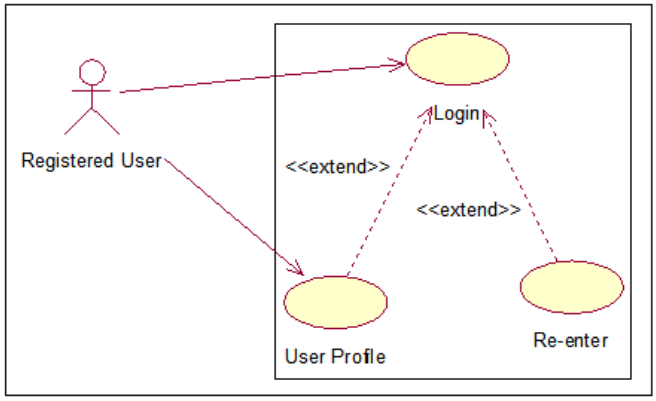

Figure 6 Login Use Case Diagram

Use Case Diagram 4: Search Module

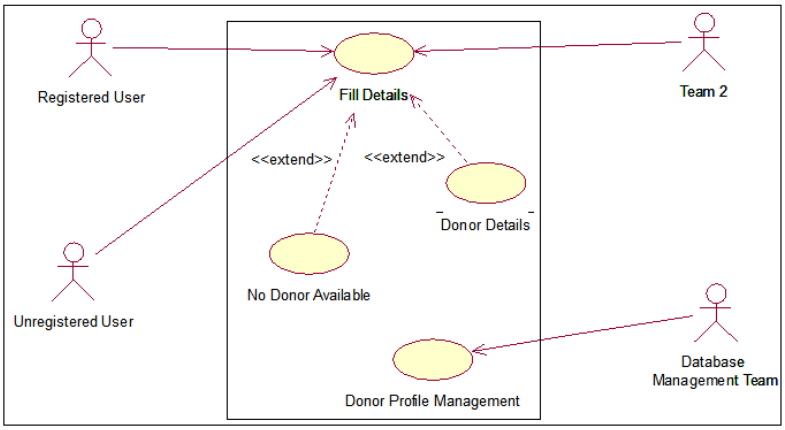

Figure 7 Search Module Use Case Diagram

Use Case Diagram 5: HomePage Development

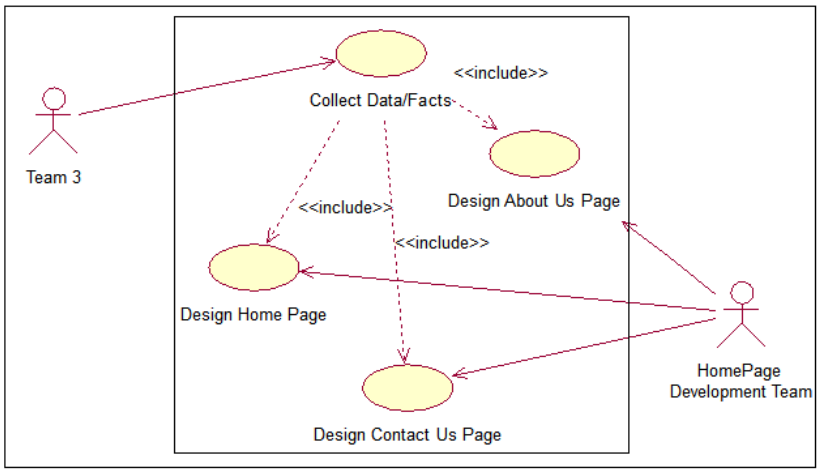

Figure 8 HomePage Development Use Case Diagram

SOFTWARE REQUIREMENT SPECIFICATIONS (SRS)

SRS is a way to represent requirements in a consistent format. The SRS is a specification for a particular product that performs certain functions in a specific environment. SRS could be written by the user or the developer itself. User provides his basic needs and requirements to the developer in form of SRS written in his language. SRS written by the developer serves as a contract between the developer and the user. SRS written by the developer team of *Red Drop* is of interest here.

1. *Introduction*

   This document defines overall software requirements for online blood information website *Red Drop*. This section describes purpose and scope of the project.

   1) *Purpose* – The document describes the capabilities that *Red Drop* will provide. The various constraints are also stated by which the application will abide.
   2) *Scope* – The website *Red Drop* will be informative website providing information about blood donors in different locations and also, it will aware people to donate blood by letting them be known to the blood donation facts.
   3) *Definitions, Acronyms, and Abbreviations* – NONE
   4) *References* – IEEE Recommended Practice for Software Requirement Specifications (IEEE standard 830-1993)
   5) *Overview* – The rest of the SRS describes various system requirements along with interfaces, features, and functionalities in details.

2. *Overall Description* – *Red Drop* maintains donors' information by letting them create their profile and filling information in the form. Another module is Admin module which has all rights regarding maintaining the website. Unregistered users can also view the donors'

information and can search for specific blood group donor in some specific area using Search module.

1) *Product Perspective – Red Drop* will be an online web application requiring a web browser on client side.

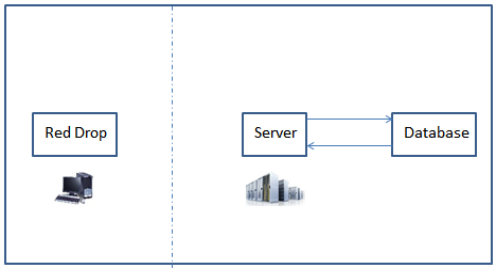

Figure 9 Product Perspective

   i. *System Interfaces* – Admin will have username and password and will have right to add/ delete/ modify the database as well as website content. (*Admin module interface is shown in Appendix C having screenshots of the website.)*
   ii. *User Interfaces* – (i) Registered user will have to enter their username and password details in the Login field. (ii) After being logged on, they will be able to view / modify their profile. (iii) There will be a field for searching the blood donors in specific areas. This field will be available for both unregistered user as well as registered users. (iv) For unregistered users, registration form will be available which on successful fill creates the profile for the user.
   iii. *Hardware Interfaces* – (i) All screen resolutions will be compatible with the website. (ii) Printer support will be provided.
   iv. *Software Interfaces* – (i) Any operating system supporting web browser. (ii) PHP. (iii) HTML. (iv) MySql.
   v. *Communication Interfaces* – NONE.
   vi. *Memory Constraints* – No additional space on hard disk / RAM required except that web browser must run.
   vii. *Operations* – NONE.
   viii. *Site Adaption Requirements*

2) *Product Functions* – In *Red Drop*, three types of users are possible, Un-registered user, Registered user, and Admin having different level access permitted. The website will perform following main functions: (i) Registration form for un-registered users, (ii) Login form for registered users, (iii) Admin page, (iv) Search module of blood donors, (v) Contact information page, (vi) Blood donation facts.

3) *User Characteristics – (i) User should be comfortable with English language, (ii) User* should know to use internet.

4) *Constraints* – NONE

5) *Assumptions and Dependencies* – (i) Information provided by donors is corrects.

6) *Apportioning of Requirements* – UNKNOWN TILL PLANNING PHASE

3. *Specific Requirements*

1) *External Interface Requirements*

i. *User Interfaces* –

(i) Registration Form: Username cannot be left empty. Also Username must start with an alphabet. Password cannot be left empty and must be 6 to 32 characters long. Password must be retyped and also must be matched to Password. Donor name, Gender, Date of birth, Weight, Blood group, and Contact information cannot be left blank. Use of captcha is mandatory.

(ii) Search Module: Blood group and Location makes key for searching.

(iii) Login Module: Username and Password must be matched.

(iv) Donor's Profile: Profile must not be opened by directly giving address of the profile in address bar. It must ask for password every time profile is opened without verification.

ii. *Hardware Interfaces* – As stated in section 2.1.3.

iii. *Software Interfaces* – As stated in section 2.1.4.

iv. *Communication Interfaces* – NONE

2) *System Features* – System will keep information of various blood donors who make registration on the website. Also, it will provide blood facts which will help in awaking people for donating the blood.

3) *Performance Requirements* – Website should respond fast and should not hang in between while large number of users would be accessing it at a single time.

4) *Design Constraints* – NONE

5) *Software System Attributes*

i. *Security* – Admin module will be password protected.

ii. *Maintainability* – Designing will be done in flexible manner. Making changes in one page would not affect other one or would not require knowledge of other one.

iii. *Portability* – It should run on all popular web browsers like Mozilla, Chrome, Safari etc.

6) *Logical Database Requirements* – (i) ADMIN: username, and password. (ii) DONOR: username, password, name, gender, photo, date of birth, weight, blood group, last

donation, frequency, area, mobile number, landline number, email address, postal address, ZIP, message, status.

7) Other Requirements – NONE

2. PROJECT RISK ANALYSIS

NONE

3. PLANNING

Overall development can be divided into two parts: Registration module, and Login module and Search module. Implementation of both of the parts will go on through two iterative parts of the life cycle, iterative development process, and maintenance phase. Figure 3 shows team distribution for this purpose. Registration module will be handled by Team 1. Login and search module will be handled by Team 1 and Team 2. Rest of the teams participate throughout the whole process i.e., throughout the development of both of the parts. Figure 10 shows development of whole system.

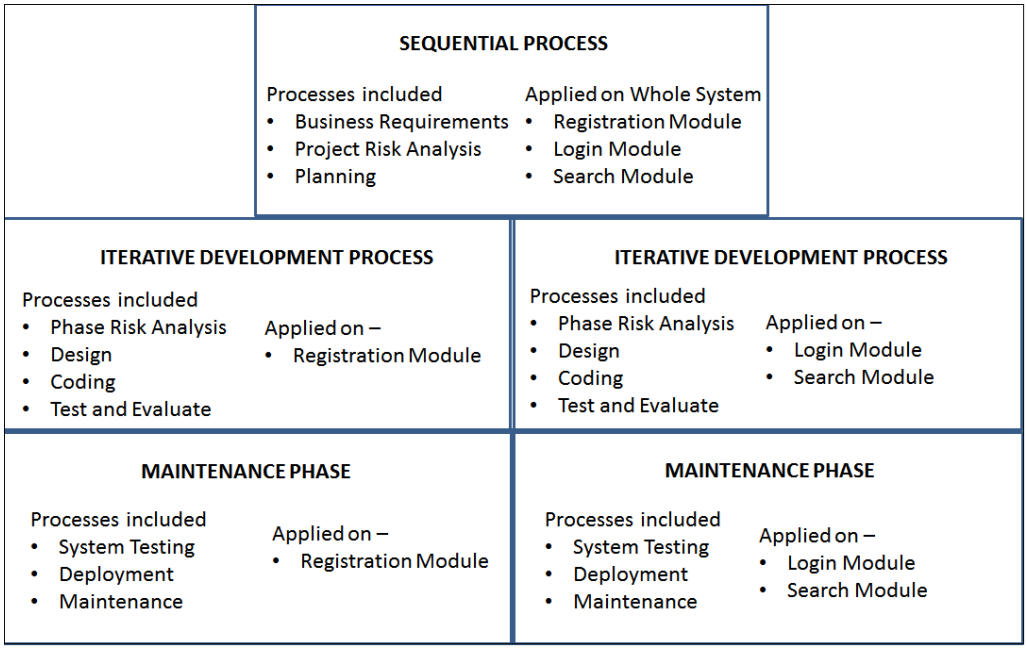

Figure 10 Development Process of *Red Drop*

ITERATIVE DEVELOPMENT PROCESS: Registration Module

A. *Phase Risk Analysis*

NONE

B. Designing

Designing will be done by the help of UML diagrams using object oriented techniques. The following steps will be followed for the designing of the *Registration Module.*

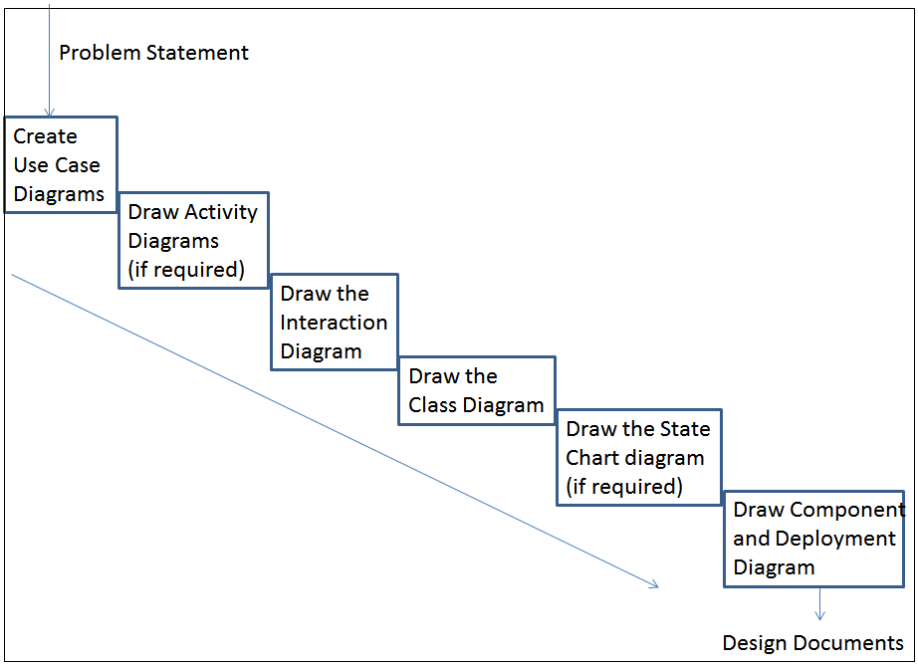

Figure 11 Designing Process

1) *Use Case Diagram:* These are already drawn in Requirement Analysis Phase.

2) *Activity Diagram:* Figure 12 presents Activity Diagram for Registration Module.

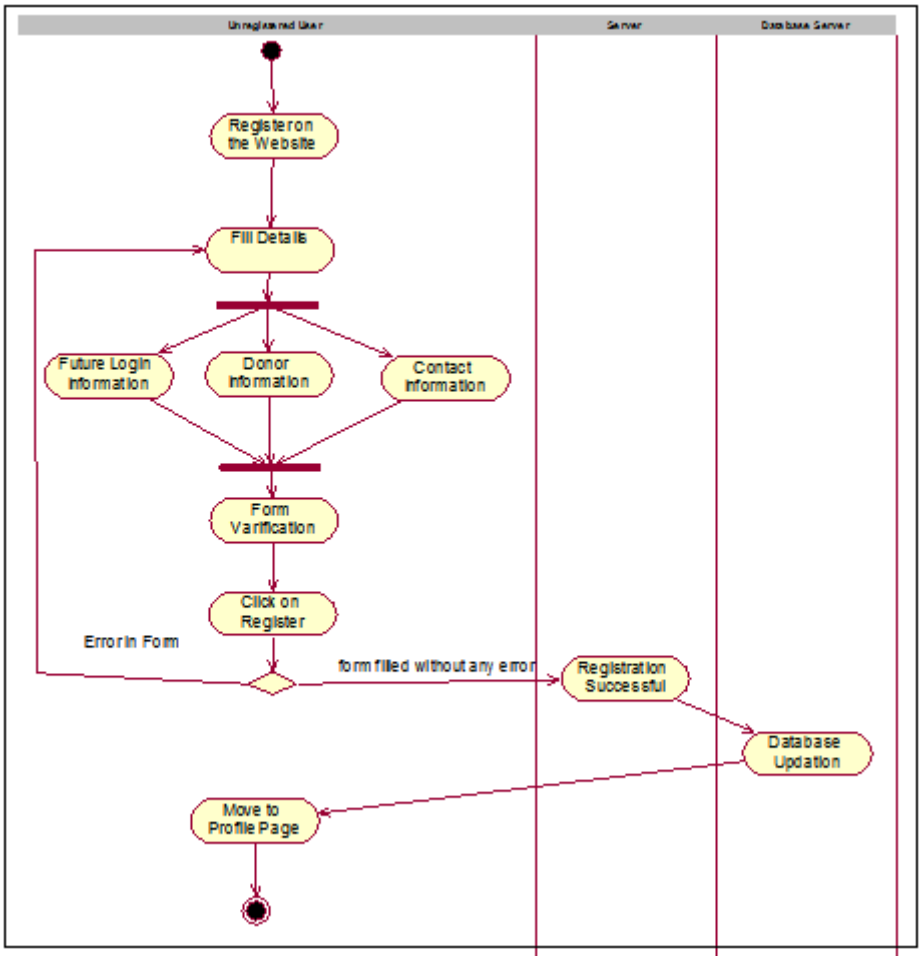

Figure 12 Activity Diagram for Registration Module

3) *Interaction Diagram:*

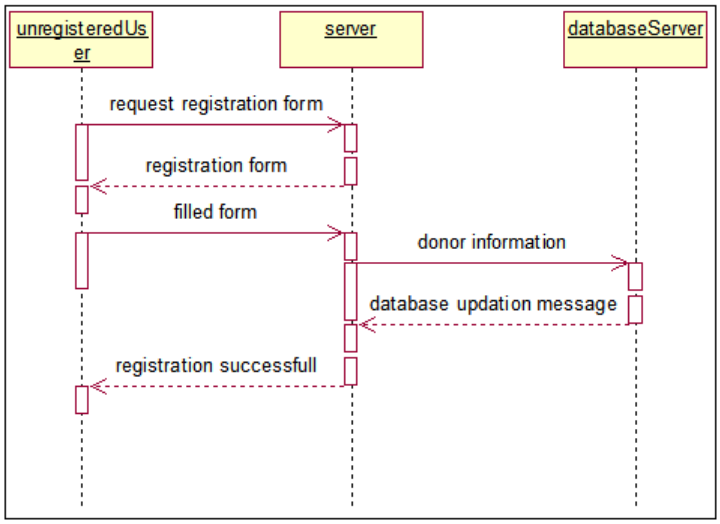

Figure 13 Sequence Diagram for Successful Registration

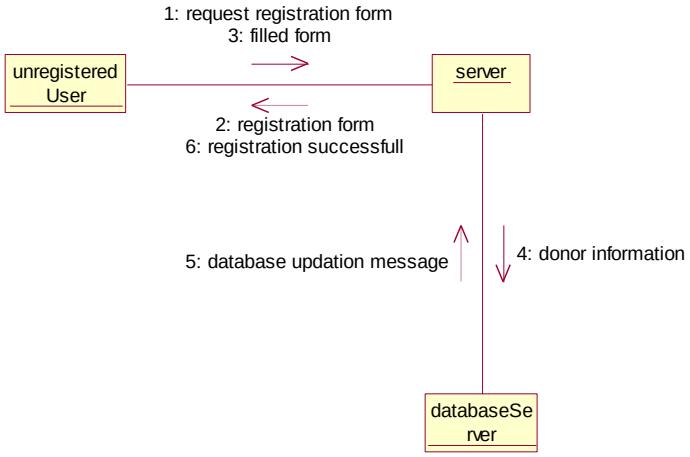

Figure 14 Collaboration Diagram for Successful Registration

4) Class Diagram:

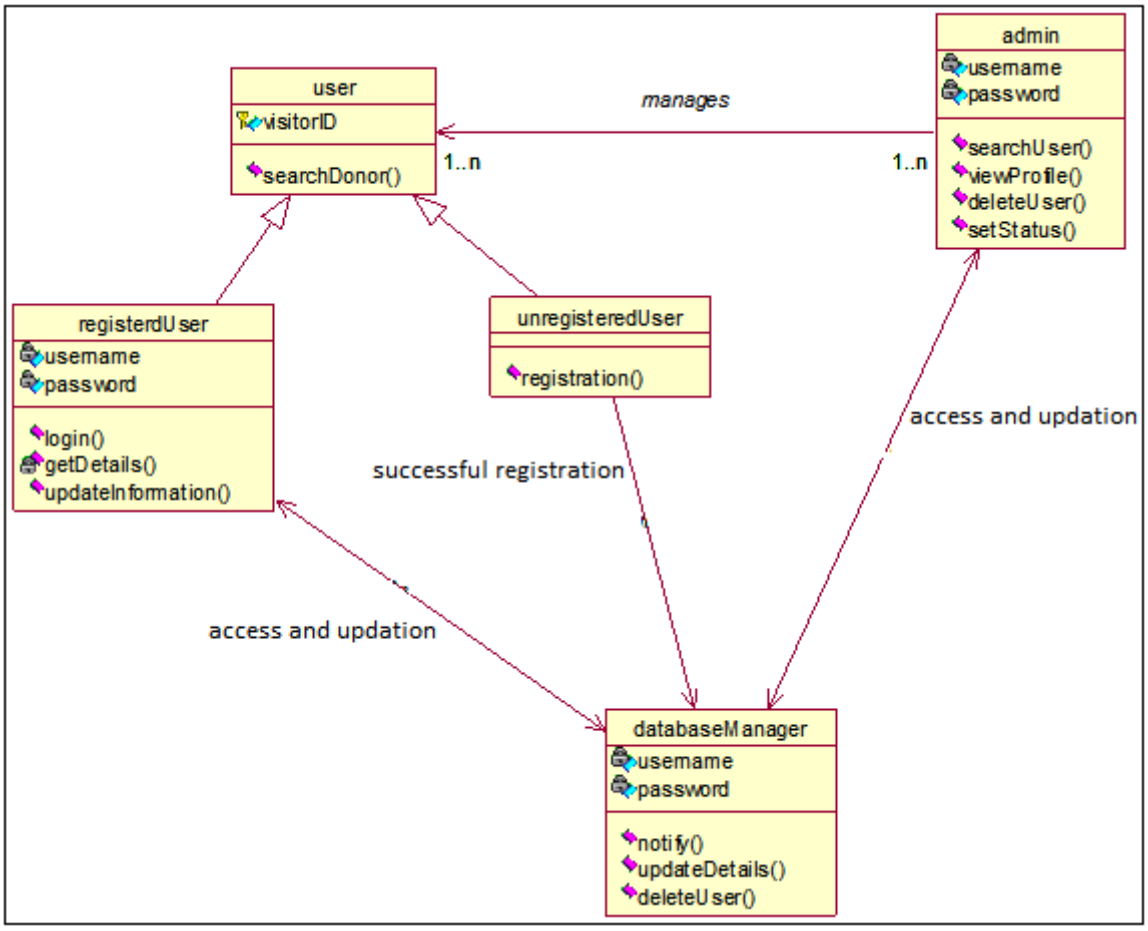

Figure 15 Class Diagram of *Red Drop*

5) *State Chart Diagram*

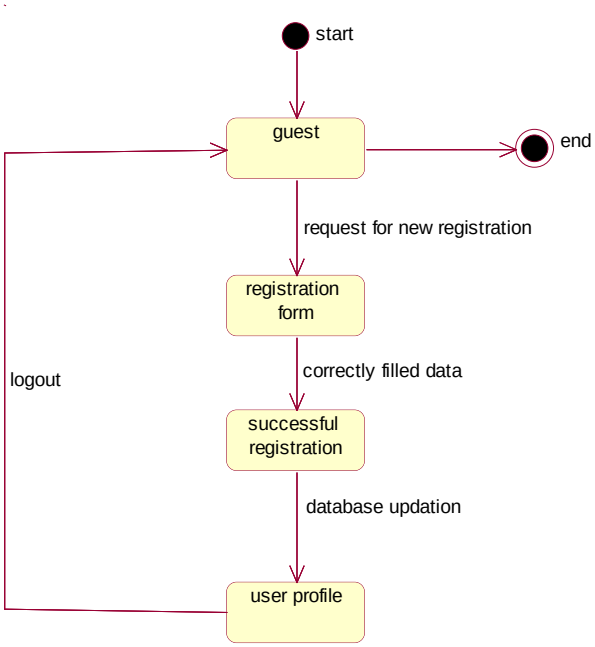

Figure 16 State Chart Diagram of Registration

C. *Coding*

'*For coding, HTML has been used for page designing and PHP has been used for server side coding, MySQL has been used for database*'.

D. *Test and Evaluate*

For registration module, following test cases will be formed.

1) Test Case 1: Valid Username – Username must start with an alphabet.
2) Test Case 2: Valid Password – Password must be 6 to 32 characters.
3) Test Case 3: Password Match – Password and Re-type password must match.
4) Test Case 4: Valid Format of DOB – DOB must be entered in YYYY-MM-DD format.
5) Test Case 5: 10 Digit Mobile/Phone Number – Mobile/Phone number must be 10 digit.
6) Test Case 6: Valid E-Mail ID – E-mail ID must be valid (xxx@yyy.zzz).
7) Test Case 7: Form Verification – Captcha must work correctly.
8) Test Case 8: Mandatory fields – Form must not be filled successfully without filling mandatory fields.
9) Test Case 9: Database Updation – On successful registration, database must be successfully updated.

ITERATIVE RELEASE PROCESS (MAINTENANCE PHASE): Registration Module

This phase includes system testing followed by deployment. System testing is performed by the person who were not involved in development process so far and the main motive of this testing is evaluating the whole system (all modules selected during *planning* phase in sequential process being treated as a standalone executable system) for its design and behaviour both against customers' requirement specifications having destructive thought in mind (try for finding maximum bugs). If system testing is being performed due to modifications in the system then *side effects* are also noted separately. Next stage is deployment of the website on the server letting it to go live for use by the users. After deployment of the website, its maintenance processing starts which might have to be initiated due to change in requirements.