**CHAPTER – I**

**Introduction:**

**Overview:**

***Blood donation website*** is a online web based project. Today you can easily connect with anything through internet services. So online platform is a best choice for my project. Blood Bank website is aims to serve for human welfare. This website has all the information you ever need about blood. Many people are here for you to help you, willing to donate the blood for you anytime. I have done all the job, rest is yours. Anybody can search for blood group they required at any time if it is present then they can collect it give to person who needs blood in emergencies.

This website stores the some basic information of person who donates the blood. And also the blood that is going to be stored in blood bank. It also provides the basic information about blood donation such as which type of person can donate to which type of needy person. This is website is built completely from scratch with user friendly interface so that it is very easy to operate with this website if he/she knows some basic knowledge of English.

**Problem Statement:**

Blood is very important for human being. In present scenario searching for blood in Emergency situation is very difficult it is the matter of life and death. So it is very important to get the blood as soon as possible so that persons life can be saved. There is no platform to provide the information of Blood Bank and information of the Blood. This website is here to solve this very important and matter of life and death problem. Even though there is government website it is not user friendly Interface so that anybody can operate it is very complicated to handle. So I have created the user friendly website so that even the person who knows basic English can operate with this website.

**Objective:**

The objectives of this websites are

* Provide a Basic Information about blood donation.
* Collect the basic information of the donors and also about blood and store it in the database.
* Giving the option to search by location to know the information of the donors in blood bank
* To get to know the how much quantity of blood is present in Blood Bank.
* To provide the information to the acceptor that if blood is not present in the given location then where can he/she can get the required blood.
* Creating Friendly user interface so that any body can handle the website.
* Collecting the basic information of the Acceptor person.

**Benefits of the System:**

* Storing the information in a right order so that we can easily analyse what is going on with the Blood Donation.
* Storing the information and also giving the information about blood in the same website.
* Saving the life of a person who is fighting in between life and death.
* Gives the other options such as where he/she can collect the blood if it is not present in the desired location.
* Manage the database efficiently.
* Delete the information efficiently from database whenever the acceptor collects the blood.
* Showing the information of the donors based on the selected location.
* Provides friendly options.

**Chapter-II**

**System Analysis:**

**Existing system:**

There are lot of existing system in web. The main website I found is the website created by the government that is eraktkosh.in which is very wonderful web that is by Government. One of the major drawback I have found in this website is it is very complex and it is not friendly usable website and also this website does not shows details of person who donated the blood. This website does not have lots of draw backs because it is been created by government. The link for this website is

* [www.eraktkosh.in](http://www.eraktkosh.in)



**Proposed System:**

Solving Above two major problems is main motive of creating this website.

* Providing the user a friendly user interface. No Complex website is created.
* The user can see the information of the donors who have donated the blood
* As this system is created from scratch it is easy to handle this website if any problem occurs.
* Creating the system that immediately deletes from the total quantity with respect to the quantity of blood that was taken from the acceptor.
* Providing Complete security to user data. Creating the system that also collects the basic information of acceptors.
* In my current proposed there is no need of any money it is built completely using the some line of code and acceptor will get the free information regarding the blood.
* Collecting necessary information about the donor and acceptor and storing the data base.
* Creating a understandable Data base Structure.
* Linking the id to blood donor and blood in Blood Bank.

**System Requirements:**

The hardware requirements

* Operating system like windows/Linux.
* The Computer/PC with good storage and good ram.
* It should be connected with internet.
* Any IDLE Should be there In computer.

The Software requirements

* The main requirement of software is Python should be installed.
* Flask Should be installed.
* Any browser with the latest version.
* Mysql should be installed.
* Python mysql-connector should be installed.
* All the basics module of python should be installed.
* Python random module should be installed.

**Chapter-III**

**System Design:**

The system design involves the production of technical and visual prototypes. This stage has some non-technical aspects such as gathering of web content. For the server side programming and other technical aspects of the design emphasis will be laid on such design concepts and principles as effective modularity (high cohesion and low coupling), information hiding and stepwise elaboration. The goal is to make the system easier to adapt, enhance, test and use.

**Producing HTML:**

There are basically 4 methods of producing HTML –

1. Coding by hand using a simple text editor
2. Translation in which content produced in a tool such as notepad is saved as a HTML document.
3. Using a tagging editor that helps fill in the required tags and
4. Using a “What you see is what you get editor” (WYSIWYG) such as Adobe Dreamweaver.

All these methods have their advantages and disadvantages.

While coding by hand may be slow and error prone, it does provide great control over mark-up, as well as help address bugs and new HTML/XHTML elements immediately. At the other extreme, ‘What You See Is What You Get’ (WYSIWYG) editors provide visual representation of a page and require no significant knowledge of HTML or CSS. However these kind of editors give the efficient developers much more potentiality. Putting all these into consideration, a WYSIWYG editor, Visual Studios was chosen for this work.

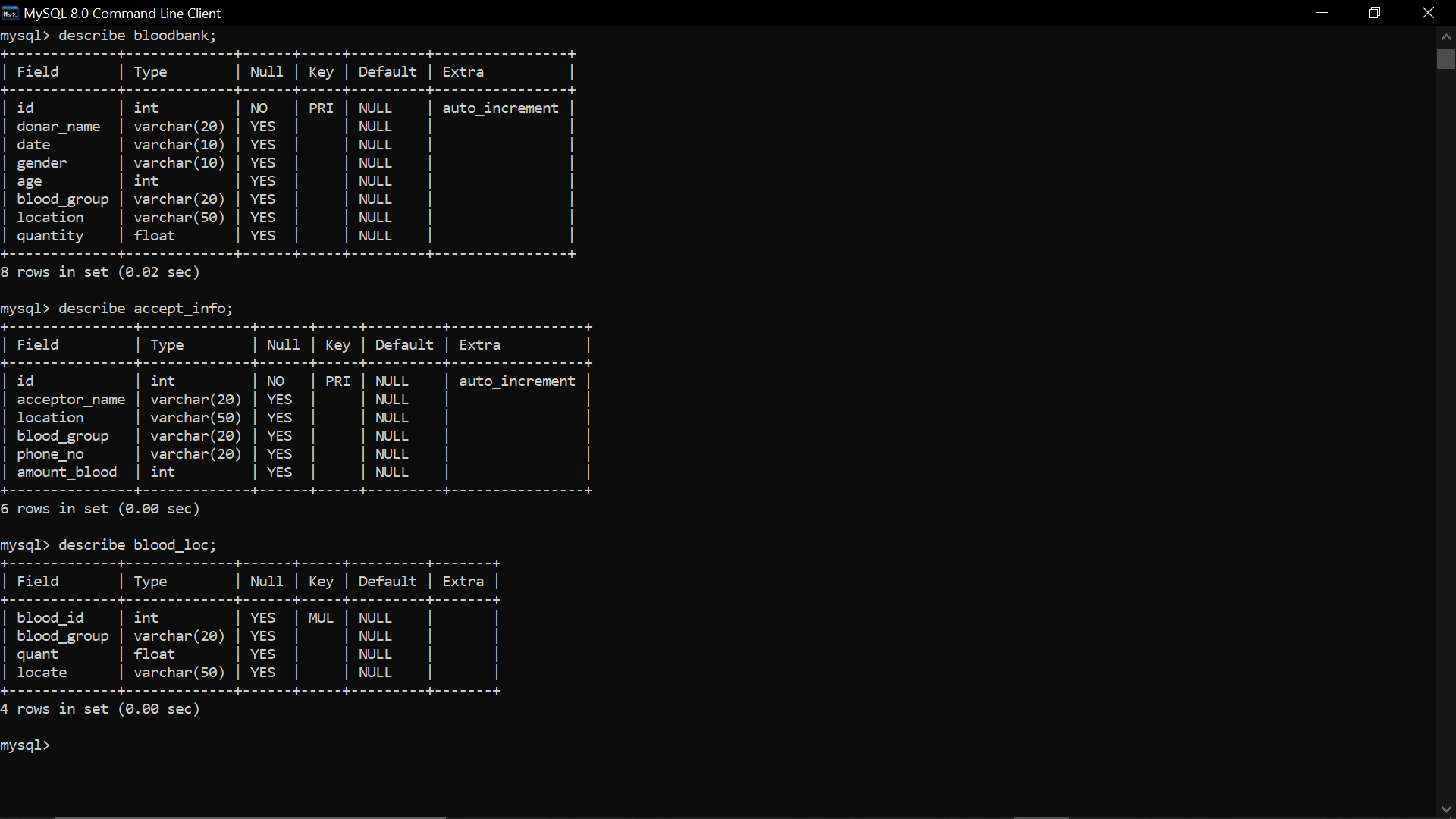
**Database Design:**

Database design involves the production of a model of the data to be stored in the database. A data model is a diagram of the database design that documents and communicates how the database is structured. The design process is divided into three main stages such as conceptual, logical and physical database design. The purpose of the conceptual database design is to decompose the design into more manageable tasks, by examining user perspectives of the system. That is, local conceptual data models are created that are a complete and accurate representation of the enterprise as seen by different users. Each local conceptual data model is made up of entity types, relationship types, attributes and their domains, primary keys and integrity constraints. For each user view identified a local conceptual data model would be built. In building the conceptual data model, a data dictionary is built to identify the major entities in the system.

An entity relationship (ER) diagram is used to visualize the system and represent the user’s requirements. The ER diagram is used to represent entities and how they relate to one another. The ER diagram also shows the relationships between the entities, their occurrence (multiplicities) and attributes.

**Data Base Tables:**

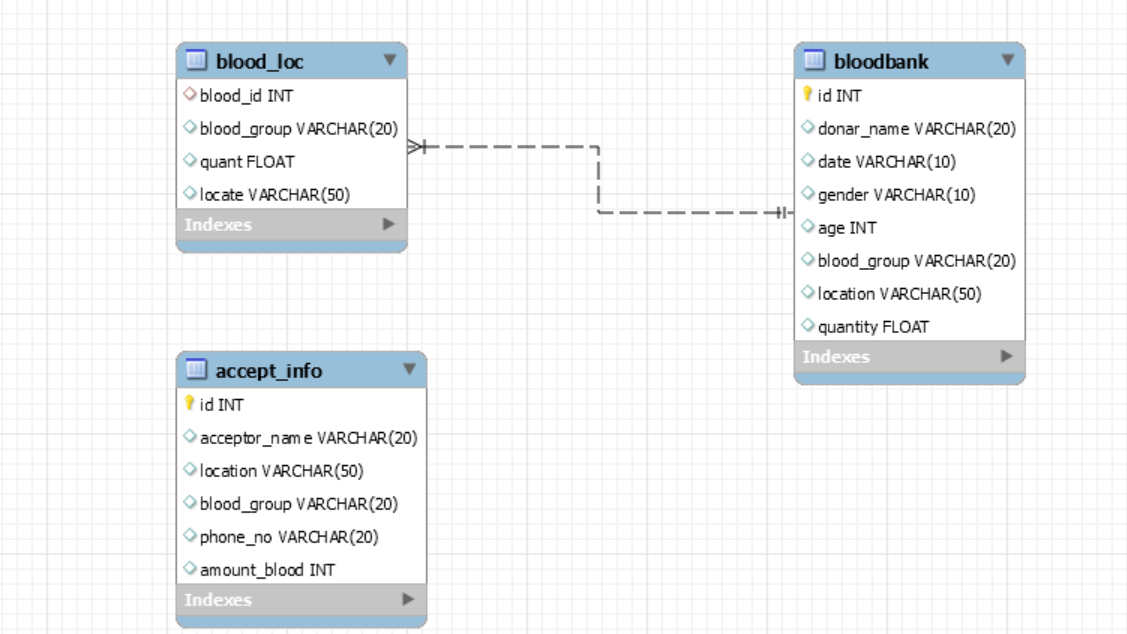
The Description about the database tables are given in below images



There are totally 3 tables are used they are namely.

* bloodbank
* blood\_loc
* accept\_info

**ER Diagram and Explanation:**

****

We can see in above picture it is a ER diagram which is created in mysql workbench.

Entities: bloodbank,blood\_loc,accept\_info

Attributes: bloodbank(id,donar\_name,date,gender,age,blood\_group,location,quantity)

blood\_loc(blood\_id,blood\_group,quant,locate)

accept\_info(id,acceptor\_name,location,blood\_group,phone\_no,amount\_blood)

Relationship: The id in bloodbank and blood\_id in blood\_loc are same

**Chapter – IV**

**System Implementation:**

**Environment:**

Implementation environment is in Python and software name is Visual Studious provides the user friendly environment.

The Functions that this system will be implementing are:

* Collects the information of the donors and store it in mysql
* Fetch the data from database and displays its whenever the Acceptor wants the blood
* After acceptor collecting the blood hi/her information also collected and stored in database
* Fetch the data of donors from the database whenever the user request for it.
* Shows proper messages.

To implement the all the above functions we need some front end, back end techniques those are discussed below

**Front End:**

**HTML:**



Hypertext Mark-up Language (HTML) is the standard mark-up language for documents designed to be displayed in a web browser. It can be assisted by technologies such as Cascading Style Sheets (CSS) and scripting languages such as JavaScript. Web browsers receive HTML documents from a web server or from local storage and render the documents into multimedia web pages. HTML describes the structure of a web page semantically and originally included cues for the appearance of the document.

HTML elements are the building blocks of HTML pages. With HTML constructs, images and other objects such as interactive forms may be embedded into the rendered page. HTML provides a means to create structured documents by denoting structural semantics for text such as headings, paragraphs, lists, links, quotes and other items. HTML elements are delineated by tags, written using angle brackets. Tags such as <img /> and <input /> directly introduce content into the page. Other tags such as <p> surround and provide information about document text and may include other tags as sub-elements. Browsers do not display the HTML tags, but use them to interpret the content of the page.

HTML can embed programs written in a scripting language such as JavaScript, which affects the behaviour and content of web pages. Inclusion of CSS defines the look and layout of content. The World Wide Web Consortium (W3C), former maintainer of the HTML and current maintainer of the CSS standards, has encouraged the use of CSS over explicit presentational HTML since 1997.

**CSS:**

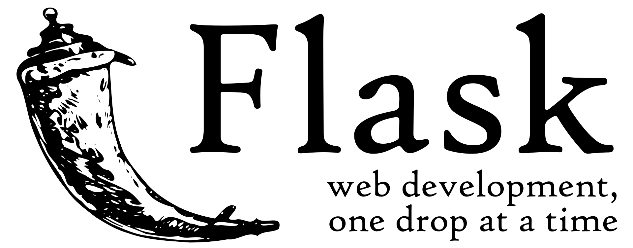


Cascading Style Sheets (CSS) is a style sheet language used for describing the presentation of a document written in a mark-up language like HTML.CSS is a cornerstone technology of the World Wide Web, alongside HTML and JavaScript.CSS is designed to enable the separation of presentation and content, including layout, colours, and fonts. This separation can improve content accessibility, provide more flexibility and control in the specification of presentation characteristics, enable multiple web pages to share formatting by specifying the relevant CSS in a separate .CSS file, and reduce complexity and repetition in the structural content.

CSS information can be provided from various sources. These sources can be the web browser, the user and the author. The information from the author can be further classified into inline, media type, importance, selector specificity, rule order, inheritance and property definition. CSS style information can be in a separate document or it can be embedded into an HTML document. Multiple style sheets can be imported. Different styles can be applied depending on the output device being used; for example, the screen version can be quite different from the printed version, so that authors can tailor the presentation appropriately for each medium. The style sheet with the highest priority controls the content display. Declarations not set in the highest priority source are passed on to a source of lower priority, such as the user agent style. The process is called cascading.

One of the goals of CSS is to allow users greater control over presentation. Someone who finds red italic headings difficult to read may apply a different style sheet. Depending on the browser and the web site, a user may choose from various style sheets provided by the designers, or may remove all added styles and view the site using the browser's default styling, or may override just the red italic heading style without altering other attributes.

**Flask:**



**Flask** is a web framework written in Python. It is classified as a microframework because it does not require particular tools or libraries. It has no database abstraction layer, form validation, or any other components where pre-existing third-party libraries provide common functions. However, Flask supports extensions that can add application features as if they were implemented in Flask itself. Extensions exist for object-relational mappers, form validation, upload handling, various open authentication technologies and several common framework related tools.

**History**

Flask was created by Armin Ronacher of Pocoo, an international group of Python enthusiasts formed in 2004.[[6]](https://en.wikipedia.org/wiki/Flask_(web_framework)#cite_note-6) According to Ronacher, the idea was originally an April joke that was popular enough to make into a serious application.

When Ronacher and Georg Brandl created a bulletin board system written in Python, the Pocoo projects Werkzeug and Jinja were developed.

Flask has become popular among Python enthusiasts. As of October 2020, it has second most stars on GitHub among Python web-development frameworks, only slightly behind Django, and was voted the most popular web framework in the Python Developers Survey 2018.

**Features:**

* Development server and debugger
* Integrated support for unit testing
* RESTful request dispatching
* Uses Jinja templating
* Support for secure cookies (client-side sessions)
* 100% WSGI 1.0 compliant
* Unicode-based
* Extensive documentation
* Google app engine compatibility
* Extensions available to enhance features desired

**Back End:**

**MySql:**



MySQL is an open source relational database management system (RDBMS) based on Structured Query Language (SQL). It is one part of the very popular LAMP platform consisting of Linux, Apache, My SQL, and PHP. Currently My SQL is owned by Oracle. My SQL database is available on most important OS platforms. It runs on BSD Unix, Linux, Windows, or Mac OS. Wikipedia and YouTube use My SQL. These sites manage millions of queries each day. My SQL comes in two versions: My SQL server system and My SQL embedded system.

RDBMS TERMINOLOGY

Before we proceed to explain MySQL database system, let's revise few definitions related to database.

* **Database:**A database is a collection of tables, with related data.
* **Table:**A table is a matrix with data. A table in a database looks like a simple spreadsheet.
* **Column:**One column (data element) contains data of one and the same kind, for example the column postcode.
* **Row:**A row (= tuple, entry or record) is a group of related data, for example the data of one subscription.
* **Redundancy:**Storing data twice, redundantly to make the system faster.
* **Primary Key:**A primary key is unique. A key value cannot occur twice in one table. With a key, you can find at most one row.
* **Foreign Key:**A foreign key is the linking pin between two tables.
* **Compound Key:**A compound key (composite key) is a key that consists of multiple columns, because one column is not sufficiently unique.
* **Index:**An index in a database resembles an index at the back of a book.
* **Referential Integrity:**Referential Integrity makes sure that a foreign key value always points to an existing row

**Chapter – V:**

**System Testing:**

System testing of software or hardware is testing conducted on a complete, integrated system to evaluate the system's compliance with its specified requirements. System testing falls within the scope of black-box testing, and as such, should require no knowledge of the inner design of the code or logic. As a rule, system testing takes, as its input, all of the "integrated" software components that have passed integration testing and also the software system itself integrated with any applicable hardware system(s). The purpose of integration testing is to detect any inconsistencies between the software units that are integrated together (called assemblages) or between any of the assemblages and the hardware. System testing is a more limited type of testing; it seeks to detect defects both within the "inter-assemblages" and also within the system as a whole.

System testing is performed on the entire system in the context of a Functional Requirement Specification(s) (FRS) and/or a System Requirement Specification (SRS). System testing tests not only the design, but also the behaviour and even the believed expectations of the customer. It is also intended to test up to and beyond the bounds defined in the software/hardware requirements specification(s).

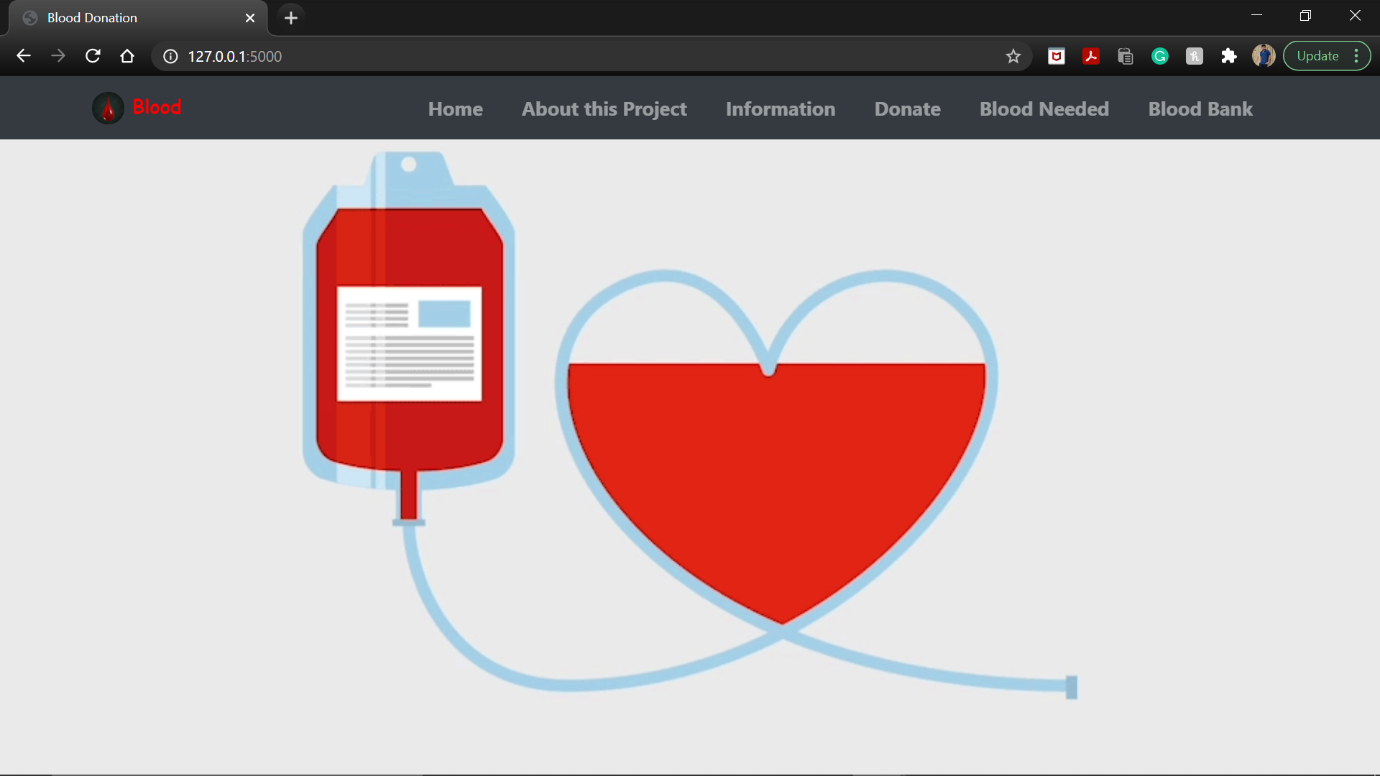
Details of extensive testing done to verify the system functionality:

* The Complete system was tested by creating a environment using conda virtual environment so that we can get to know all the requirement.
* The options can be found in top Navbar to operate with the system.
* The user can get the data by selecting the specific condition from the given option.
* While entering the data of donors it is compulsory to enter all the section.
* While entering some of data it is required selecting a input from given set of option such that no spelling change is made to it.
* The required information for person to donate blood is given in website.
* All Data from the user is collected in the form of String later it is inserted to mysql database table to required form.
* Id is given to all the row so that no duplicate values should not occur.
* The required data is shown on front page in beautiful way.
* The data is collected from the database for given condition correctly and displayed on front end.
* If no data is found from the database for given specific condition then relevant message is shown on screen.
* If some of the user did not get the blood from their location then he/she can get the data related to that blood type like where else he/she can find the required blood type.
* The system is created in a such a way that data is stored in right order.
* The system is handled in such a way that no error should be found.

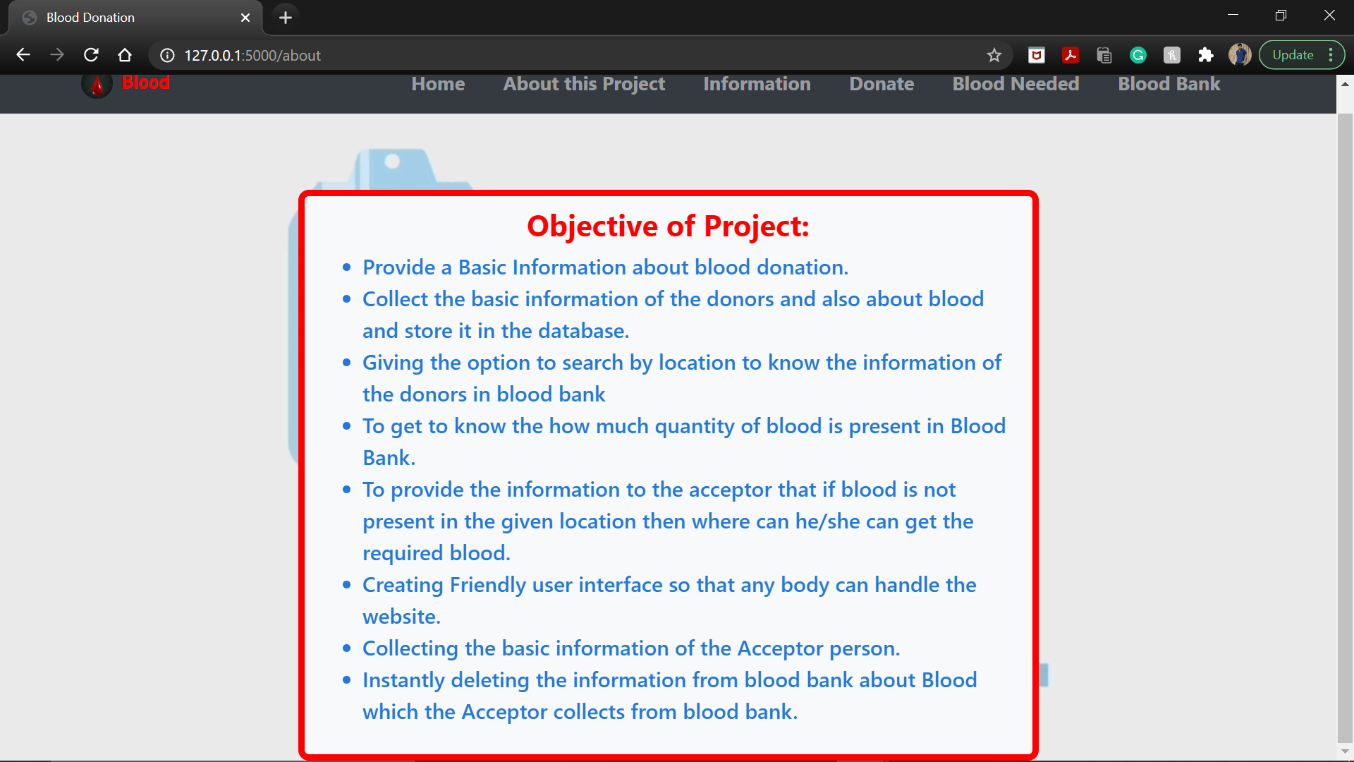
**Chapter – VI:**

**Results:**

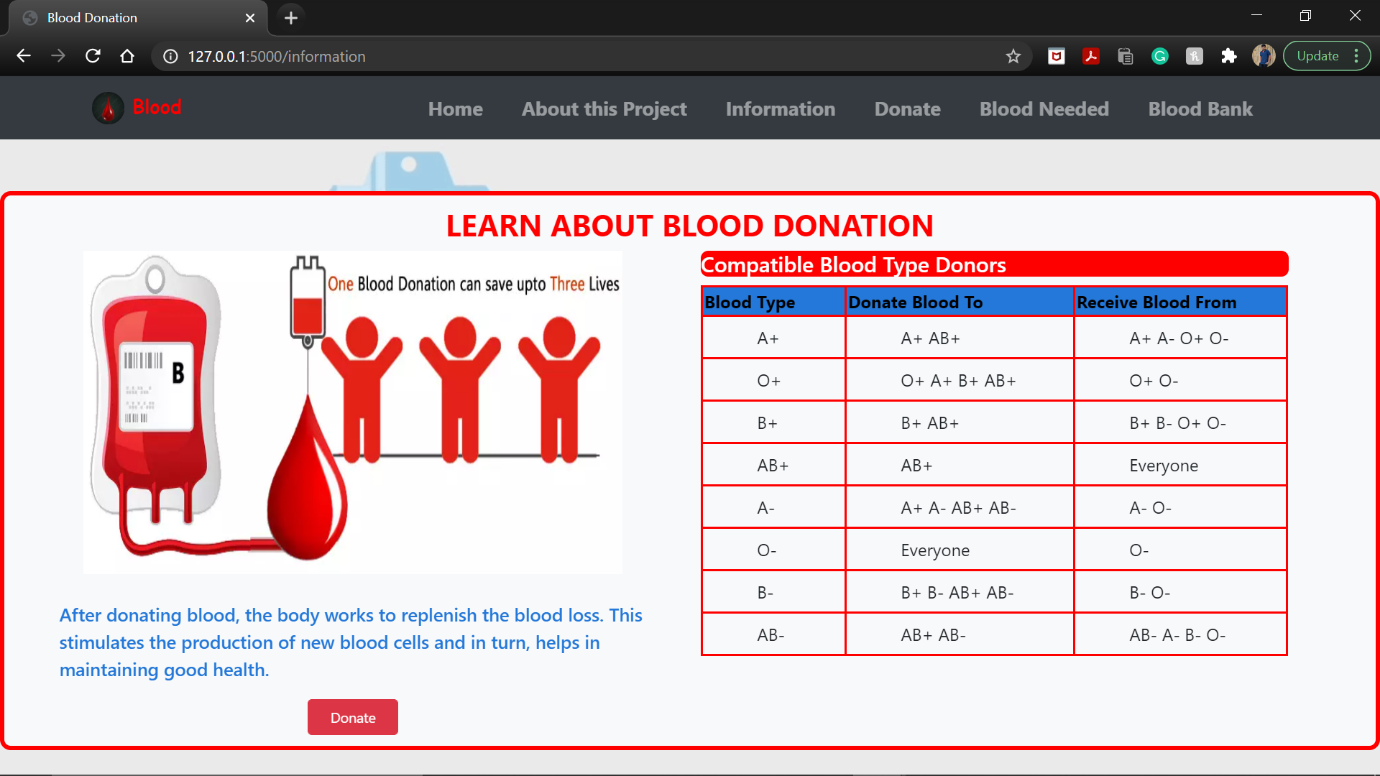
The Screen shots of the pages are show and explained in below pages.



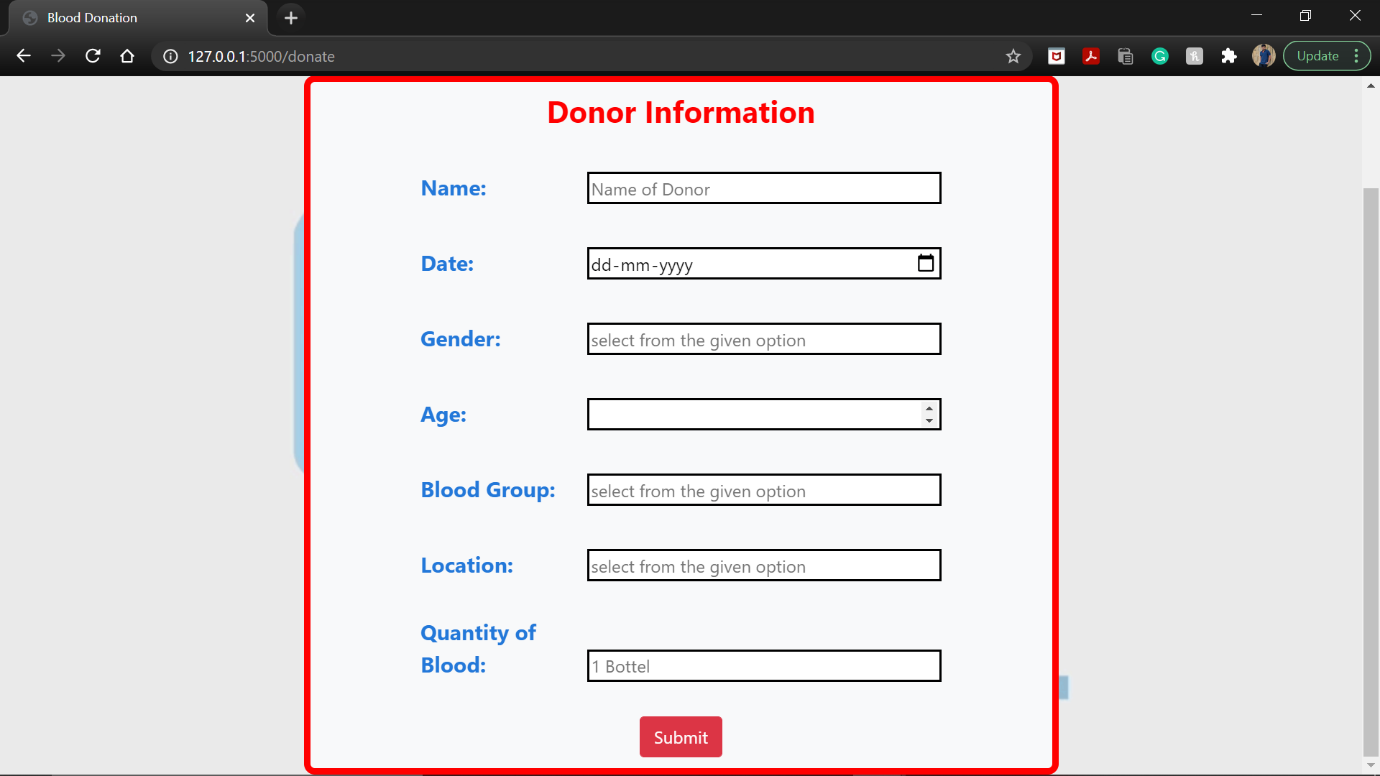
The Home Page of the website.



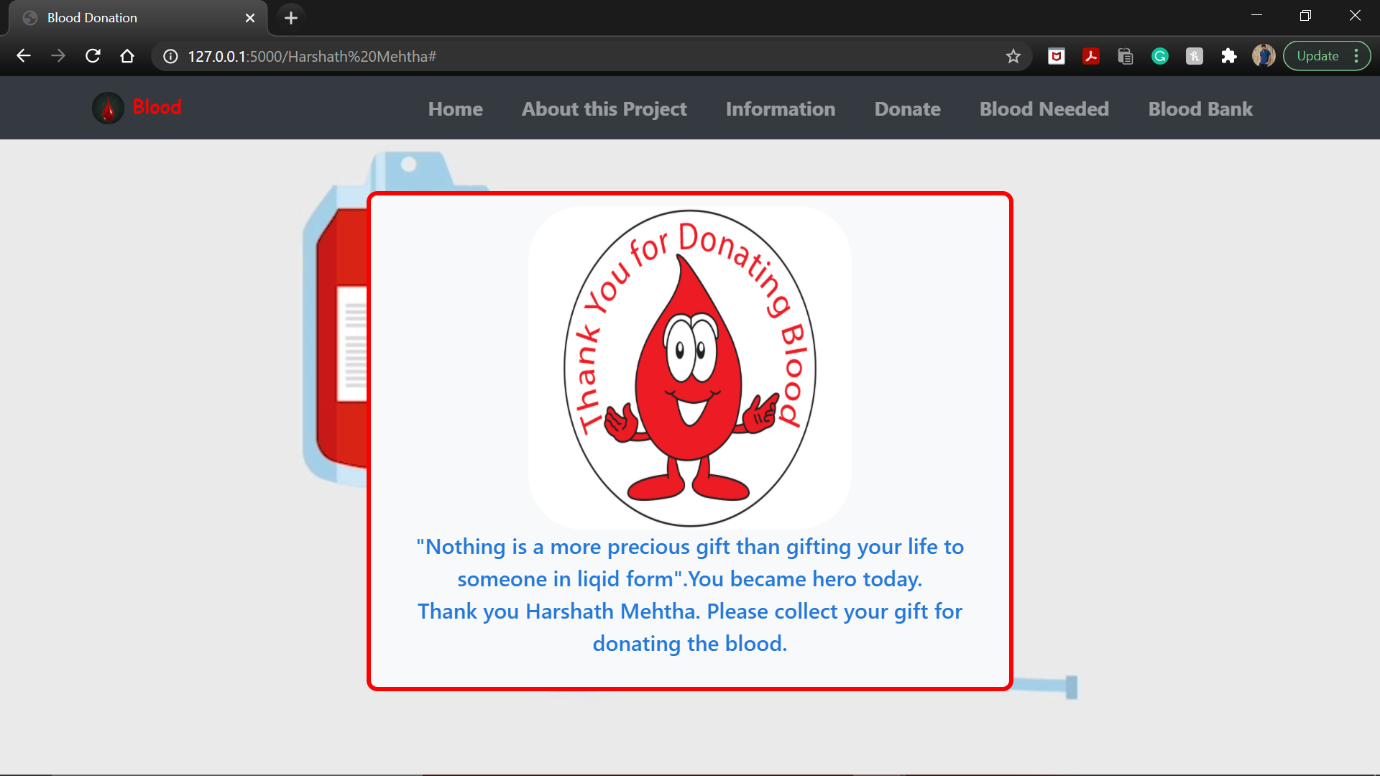
To Give information about objective of the project.



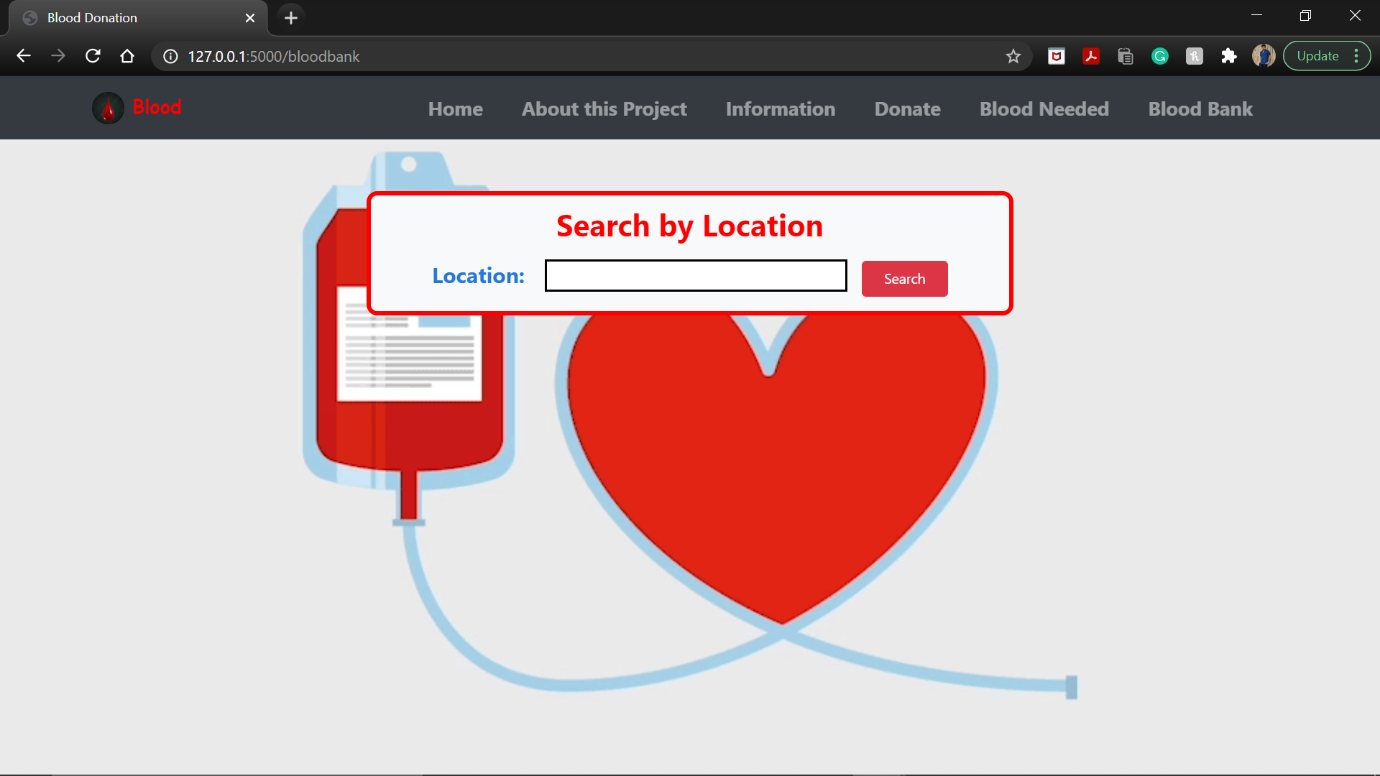
This Page Provides all the basic information about blood donation to donors.



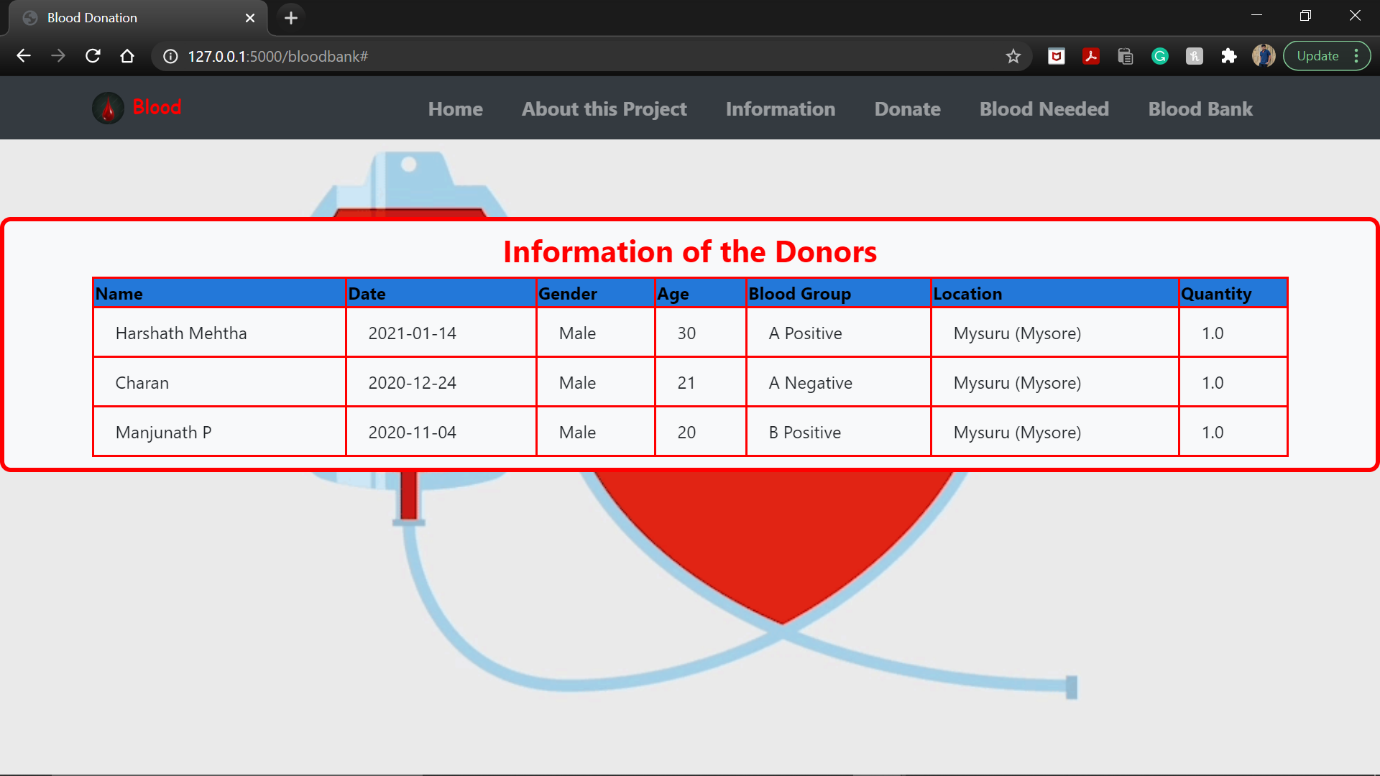
This Page is a form that collects all the basic information about donor. And store it in database.



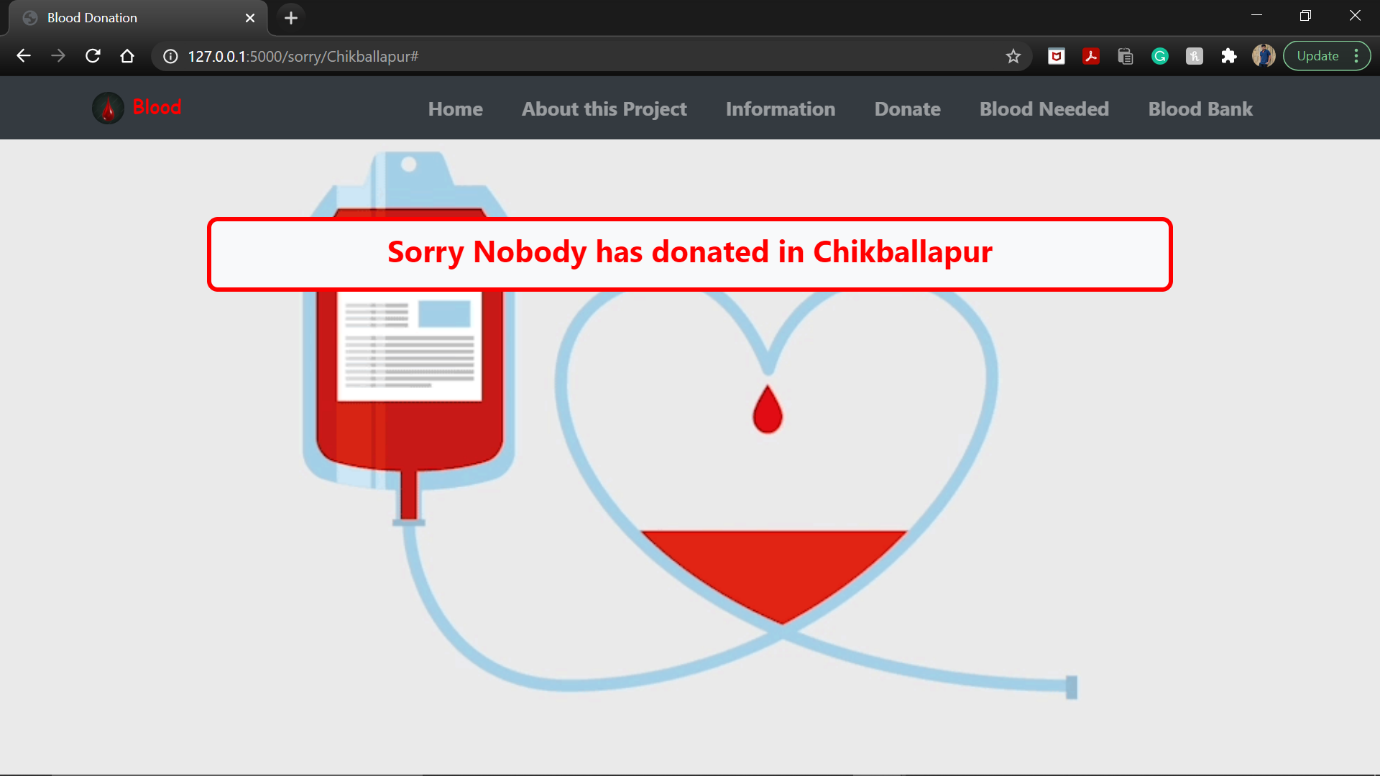
This Page is Displayed after submitting the information about donor.



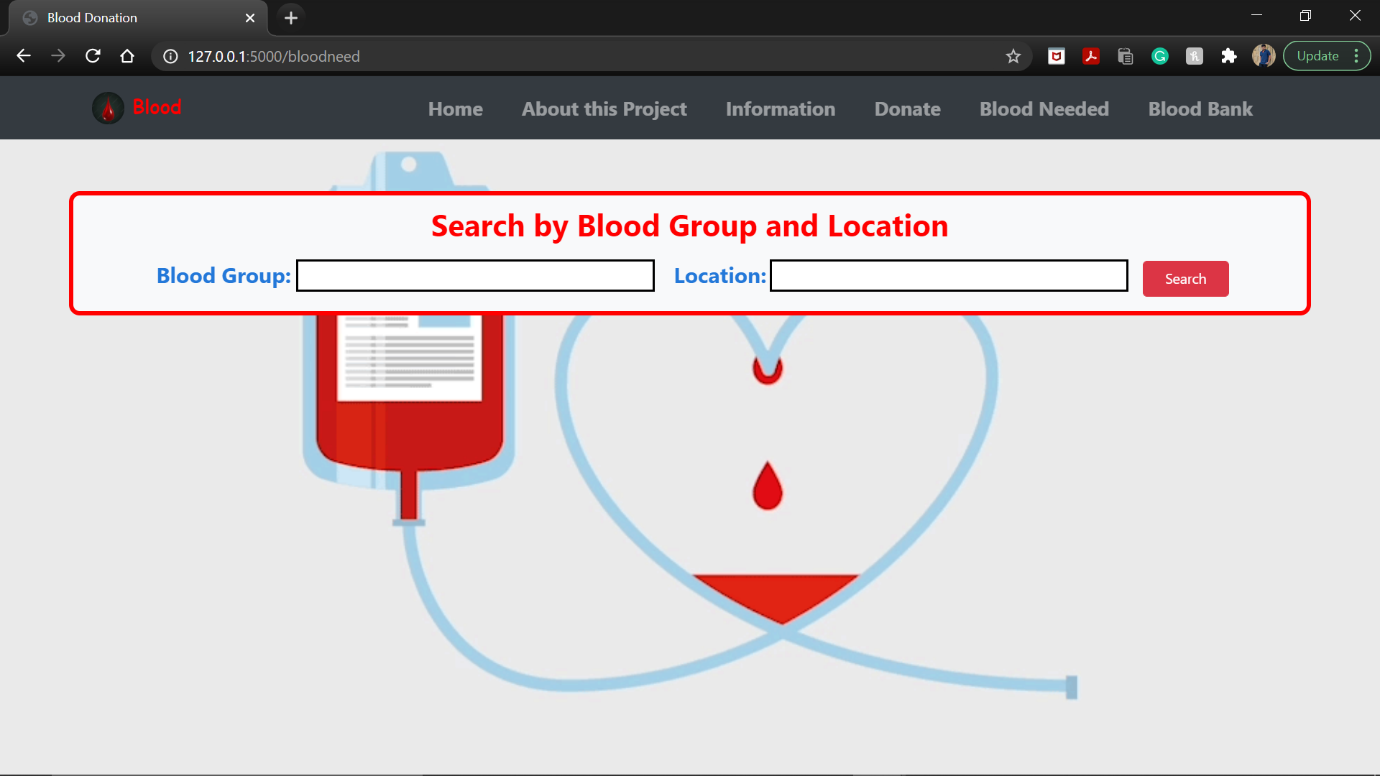
This is a page where user will give a location as input and get the all the information about donors who have submitted the blood in that location.

****

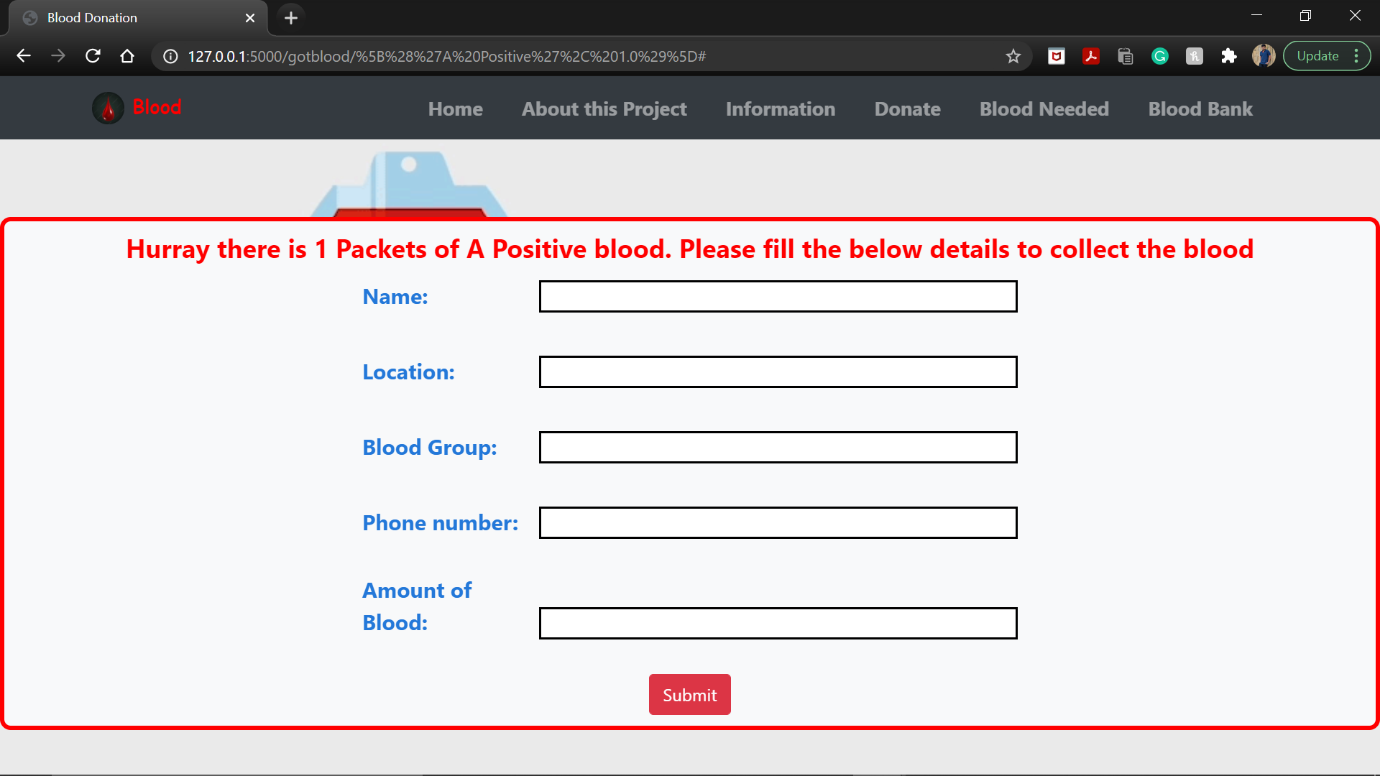
After submitting the location the user can get the information as shown in the above image.



If no person has donated the blood the message is shown as above.



This Page is for acceptor it collects 2 inputs that is location and blood type.



If blood is present in given location the next page will be like this. Acceptor has to fill the form to collect the blood.

After giving the information and collecting the data the amount of blood in that location is deleted. The acceptor data is also collected and stored in the database.

Also some other pages are added such that it shows relevant messages when required.

**Conclusion:**

This Project was completely created from scratch without taking any source code from any other website. This website is created for social wellbeing. This website is can lot of people who are fighting between life and death if future enhancement and other options are provided. Since this is a DBMS mini project I have added few little option like creating a data base and updating it inserting it and deleting it. This website was created using more than 1000 lines of codes. This website can create functions which are high on demand. This website can display the information of the donors who have donated the blood and admin can also see the details of the acceptor in this website.

**Future Enhancement:**

The Future Enhancement can be done to this by adding more animation to website. Collecting more information donor and acceptor like their photo copy and Aadhar card copy and storing it in large data bases. While Entering the input purification can be done so that no spelling mistakes can be corrected while entering into table so that data is stored in proper order. If this website is created very complexly like the website of government then it can be became a competitor to that website. This website can be hosted so that every one can be used. As this website can be run on local host this website cannot be used by others if they want to use right setup has to be made.

**References:**

This are the website I have referred for learning purpose

<https://www.udemy.com/course/python-and-flask-bootcamp-create-websites-using-flask/>

<https://www.w3schools.com/html/>

Most of the references are through stack overflow

<https://stackoverflow.com/>