**TABLE OF CONTENTS**

|  |  |  |
| --- | --- | --- |
| **S.No** | **CONTENTS** | **Page No** |
| **1.** | **ABSTRACT** | 1 |
| **2.** | **SOFTWARE REQUIREMENT SPECIFICATION** | 2 |
| 2.1. Introduction | 2 |
| 2.2. Current System | 3 |
| 2.3. Proposed System | 3 |
| **3.** | **UML DESIGN** | 8 |
|  | 3.1. Use Case Diagram | 9 |
|  | 3.2. Class Diagram | 13 |
|  | 3.3. Sequence Diagram | 14 |
|  | 3.4. Collaboration Diagram | 17 |
|  | 3.5. State Chart Diagram | 20 |
|  | 3.6. Activity Diagram | 23 |
| **4.** | **SYSTEM DESIGN DOCUMENT** | 27 |
|  | 4.1. Introduction | 27 |
|  | 4.2. Current Software Architecture | 28 |
|  | 4.3. Proposed Software Architecture | 28 |
|  | 4.4. Subsystem Services | 29 |
|  | 4.5. Database Design | 31 |
| **5.** | **IMPLEMENTATION SCREENS** | 33 |
| **6.** | **SYSTEM TESTING** | 35 |
| **7.** | **CONCLUSION** | 39 |
| **8.** | **REFERENCES** | 40 |

**LIST OF FIGURES**

|  |  |  |
| --- | --- | --- |
| **Figure no** | **Name of Figure** | **Page no** |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

**LIST OF ABBREVIATIONS**

* **HTTP:** Hypertext Transfer Protocol.
* **JDBC:** Java Database Connectivity.
* **HTML:** Hypertext Markup Language.
* **JSP:**  Java Server Pages.

1. **ABSTRACT**

Our project online medicine system deals with medicines. The main aim of the project is to supply required medicines to the users. Here the advantage is the customer need not to go for the medical shop to buy medicines.

The registered user can give money when he receives the medicines (cash on delivery). The administrator is responsible to receive money and transfer medicines to the users. The administrator is responsible to post the current available medicines on the online medicine system and need to update the information.

The user also gives suggestions to the online medicine system about the newly available best medicines in the market. This online medicine system provides best medicines to the user and help to cure diseases.

1

**1.INTRODUCTION**

1. **SOFTWARE REQUIREMENT SPECIFICATION**

**2.1 Introduction:**

**Teachforfriend**  is basically a website which is useful to all category of students in all schools

and colleges by sharing the knowledge of top grade students voluntarily via this website eiher by making an video and publish it on our webiste or by sharing the materials via our website.

**2.1.1 Purpose of the System:**

The purpose of the system is mainly to reduce the time that a top grade student to teach his friends one aftet the other.It takes so much time for an top grade student to teach the subject he/she knows at the time of examinations. And another problem is though the average and below average students since they have some fear to ask their doubts and other subject related issues. So they approch their friend whoo is cofortable to him in learning a particular topic,but teaching one guy may be easy for a top grade student.If the number increases per person at different timings the person who teaches to his/her friends has to teach the same content again and again.He feels frustrating at some point of time,but he/she may be unable to avoid his/her friends .By utilising the same time without teaching the same topic again and again the top grade students may be able to gain extra knowledge to excel in their parrticular studies.

our website **teachforfriend** is basically allowing its users to publish their tutorial or material in this website once it was completed the other users who want to learn from their friends searches their friend or search for a particular topic to get the same lecture any number of times he/she want and we provide discussion form so that the users can discuss among them get the dought clarified.

By this we can ensure that it is beneficial to all sections of students.

**2.1.2 Scope of the System:**

* Types of users-student-staff-admin
* Types of Modules

**Login module:**

In this the user is able to login using google account.

**Profile module:**

In this module the user is able to view their own tutorials and personal info.

**Add tutorial module:**

In this module the user is able to add tutorial by stating the particular subject,chapter, topic, topic name,topic url(youtube embed url of the users youtube channel).

**Search module:**

In this module the user can able to search for other tutorials by specifying the details of either the the other user or specifying the details of the tutorial(ie.. subject or chapter or topic name etc).

**Help module:**

In this module the user is able to see the details about the website and some videos about how to use the website effectively(ie.. how to upload a tutorial into the website?,how to search for an tutorial etc.).

**2.1.3 Objective :**

The main objective of this application is to enhance the progress of all sections of students via sharing their knowledge among themselves.

**2.1.4 Definitions, acronyms and abbreviations:**

**User**: A person who may be a student who want teach to others or student who want to learn from his friend.

**Admin:**A person who have all privilages in accessing the website and can access and manipulate the info about the other users.

**2.2 Current System:**

At present in this system the user can only be able to add tutorials and view the tutorials made by others or by himself/herself and he/she can search for a particular tutorial using the search box. The user can be able to edit the personal info and info about the tutorials etc

**2.3 Proposed System:**

In future enhancement of the system chat system.

**2.3.1 Functional Requirements:**

Functional requirements are the requirements which deals with the operational requirements of the system and the requirements that are requested by the user.

**User**:

User can add tutorials and can view other tutorials.

**Administrator**:

Administrator is the one who can manages the entire site and prevents unauthorised access of database.

**2.3.2 Non Functional Requirements:**

Non functional requirements describe user visible aspects of the system that are not directly related with the functional behaviour of the system. Non functional requirements include quantitative constraints such as response time or accuracy.

**2.3.2.1 Hardware considerations:**

* Minimum Requirements

|  |  |  |  |
| --- | --- | --- | --- |
| Software | Processor | RAM | Disk Space |
| Internet Explorer – 6 | Intel Pentium III  (or)  AMD – 800MHz | 128 MB | 100 MB |
|  |  |  |  |

* Recommended Requirements:

|  |  |  |  |
| --- | --- | --- | --- |
| Software | Processor | RAM | Disk Space |
| Internet Explorer – 8 | Intel I3 (higher)  (or)  AMD – 1 GHz | 1GB | 40GB |
|  |  |  |  |

**Software considerations**:

* Programming language : php
* UML : Rational rose
* Operating system :Windows,ios,liinux.
* DBMS : mysql.
* Web technology : php,bootstrap.
* Web server : Apache Tomcat 6.0
  + 1. **SYSTEM MODELS:**

Object oriented design is concerned with developing an object oriented model of a software system to implement the identified requirements. It is the process of defining components, interfaces,objects, classes, attributes, and operations that will satisfy therequirements. We typically start with candidate objects defined during analysis, but add much more rigor to their definitions. Then we add or change objects as needy to refine a solution.

Object oriented design can yield the following benefits.

**Maintainability:**

Through simplified mapping to the problem domain, which provides for less analysis effort, less complexity in system design, and easier verification by the user.

**Reusability:**

Reusability of the design saves time and cost.

**Productivity:**

Productivity gains through direct mapping to features of object oriented programming.

**2.3.4 Usecase Diagram:**

a use case is a list of actions or event steps typically defining the interactions between a Actor and a system to achieve a goal. The actor can be a human or other external system**.**

**2.3.4.1 Identification of Actors:**

Actors represent system users. They help delimit the system and give a clearer picture of what the system should do. It is important to note that an actor interacts with, but has no control over the use cases.

An actor is someone or something that:

• Interacts with or uses the system.

• Provides input to and receives information from the system.

• Is external to the system and has no control over the use cases.

An actor can be represented as shown below:

1.A class rectangle with the label <<actor>>

|  |
| --- |
| <<actor>>  Administrator |

1. The stick figure with the name of the actor below.

Administrator user

Actors identified are:

1. Administrator.

2. User.

**2.3.4.2 Identification of Use cases:**

In its simplest form, a use case can be described as a specific way of using the system from a user’s (actors) perspective.

A more detailed description might characterize a use case as:

• A pattern of behavior the system exhibits.

• A sequence of related transactions performed by an actor in the system.

• Delivering something of value to the actor.

Use cases provide a means to:

• Capture system requirements.

• Communicate with the end users and domain experts.

• Test the system.

Use cases are the best discovered by examining the actors and defining what the actor will be do with the system. Since all the needs of a system typically cannot be covered in one use case, it is usual to have a collection of use case. Together this use case collection specifies all the ways of using the system.

* + - 1. **Constructions of Use case diagrams:**

Use-case diagrams graphically depict the system behavior (use cases).These diagram’s present a high level view of how the system is used as viewed from an outsider’s(actor’s) perspective.

A use-case diagram can contain:

• Actors (“things” outside the system)

• Use cases (system boundaries identifying what the system should do).

• Interactions or relationships between actors and use cases in the System include the associations, dependencies, and generalizations.

**2.3.4.4 Use Case Description** :

Use case name : Main use case

Participating actors : User, Administrator

Entry conditions : User enter into the website by using the google login.

Flow of events : 1)User can view his/her profile.

2)User can add tutorial.

3)User can search his friend.

4)User can view the tutorials.

5)User can participate in the discussion forum.

Admin can be able to check the statastics about the users who are using this website.

Exit condition : The system stores the information in the database and respond with profile modified successfully.

Special requirements :The response should be obtained within 20sec.

**2.4. GLOSSARY:**

HTTP:

Hypertext Transfer Protocol is a transaction oriented client orserver protocol between web browser and web server.

HTML:

Hypertext Mark up Language. It is a mark up language used to design static web pages. An HTML file can be created using text editor.

**BOOTSTRAP**: Bootstrap is an open source toolkit for developing with HTML, CSS, and JS. Quickly prototype your ideas or build your entire app with our Sass variables and mixins, responsive grid system, extensive prebuilt components, and powerful plugins built on jQuery.

**PHP:** PHP is a server scripting language, and a powerful tool for making dynamic and interactive Web pages.PHP is a widely-used, free, and efficient alternative to competitors such as Microsoft's ASP.

**MYSQL:** It  enables users to meet the **database** challenges of next generation web, cloud, and communications services with uncompromising scalability, uptime and agility.