4/12/2013

**MEDI-LINE**

**YOUR VIRTUAL DOCTOR**

Prof. Mansukh Savaliya

G 45

WEB PROGRAMMING

**Contents**

1. Introduction  
 1.1 Purpose and Scope

1.2 Design Overview  
 1.2.1 Approach

1.2.2 Guiding Principles

1.3 Reference Documents

2. High Level Design  
 2.1 Architectural Design Diagram  
 2.2 Database Design  
 2.2.1 ERD

2.2.2 Relational Model

2.2.3 Sequence Diagrams

3. Low Level Design  
 3.1 Activity Diagram  
 3.2 Class Diagram

4. Interface Description

**1. Introduction**

**1.1 Purpose and Scope**

The purpose of this document is to formalize the design and implementation of the MediLine website. The primary purpose of the document is:

* To Provide the High and Low Level Design of the website.
* To document different modules and components to be implemented.
* To document the algorithm used in building each module or group of modules and show how the various components are integrated.

**1.2 Design Overview**

**1.2.1 Approach**

This document is created when the team moves from the problem domain to the solution domain, from designing to coding domain. This document describes the functions of various modules and how the various modules interact with each other.

A design methodology is a systematic approach to creating a design by applying a particular set of techniques and following a particular set of guidelines. Object Oriented Design views a system as a set of modules with clearly defined behavior that interact with each other in a clearly defined manner to meet the system's requirements.

This document also shows ER Diagram, Class Diagram, Sequence Diagram, Component Diagram and the Activity Diagram.

**1.2.2 Guiding Principles**

Guiding principles provide a foundation upon which to develop the target architecture for our system. These in turn drive design principles that can be used to validate the design. Following are some of the guiding principles that will be followed.

Scalable  
Scalability is the ability of the platform to scale both up and down to support varying numbers of users. The application should be able to scale horizontally (by adding more servers) or vertically (by increasing hardware capacity or software efficiency).

Flexible  
Flexibility is the ability of the application to adapt and evolve to accommodate new requirements without affecting the existing operations. This relies on a modular architecture which isolates one layer from other. Coupling among various modules addressing different functionality must be as low as possible and cohesion among the group of modules to provide functionality must be as high as possible.

Maintainable  
Modules shall be designed with the view for future optimizations and for adding extra functionalities.

Portable  
Modules designed should be portable or platform independent.

Security and Access Privileges  
Modules shall make sure that the correct interface and functionalities are available according to the user type that is using it.

**1.4 Reference Documents**

* IEEE Guide for Developing System Requirements Specifications (IEEE Std 1233, 1998 Edition)

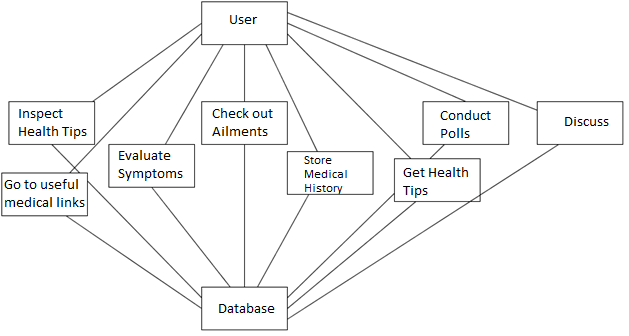
**2. High Level Design**

**2.1 Architectural Design Diagram**

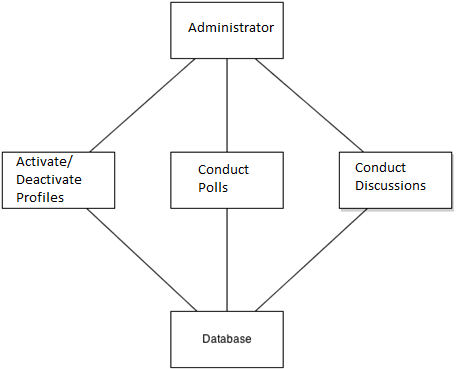
The Architectural Design Diagrams are explained from the viewpoint of the types of users using the system namely, the **Administrator** and the **User**.

The administrator accounts are provided with certain privileges giving them access to special functions in the system.

2.1.1 User View



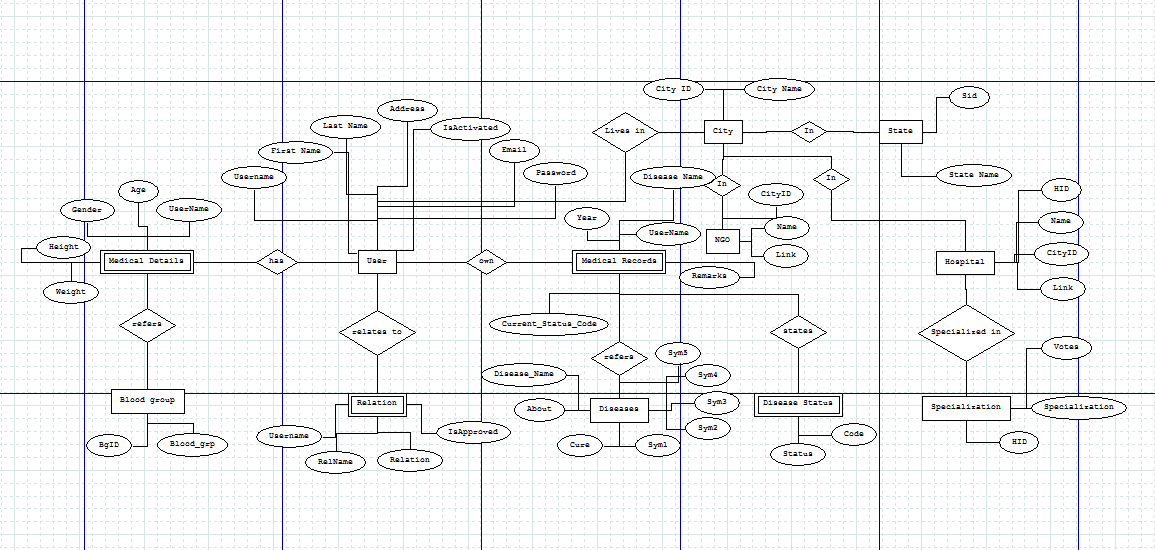
2.1.2 Administrator View



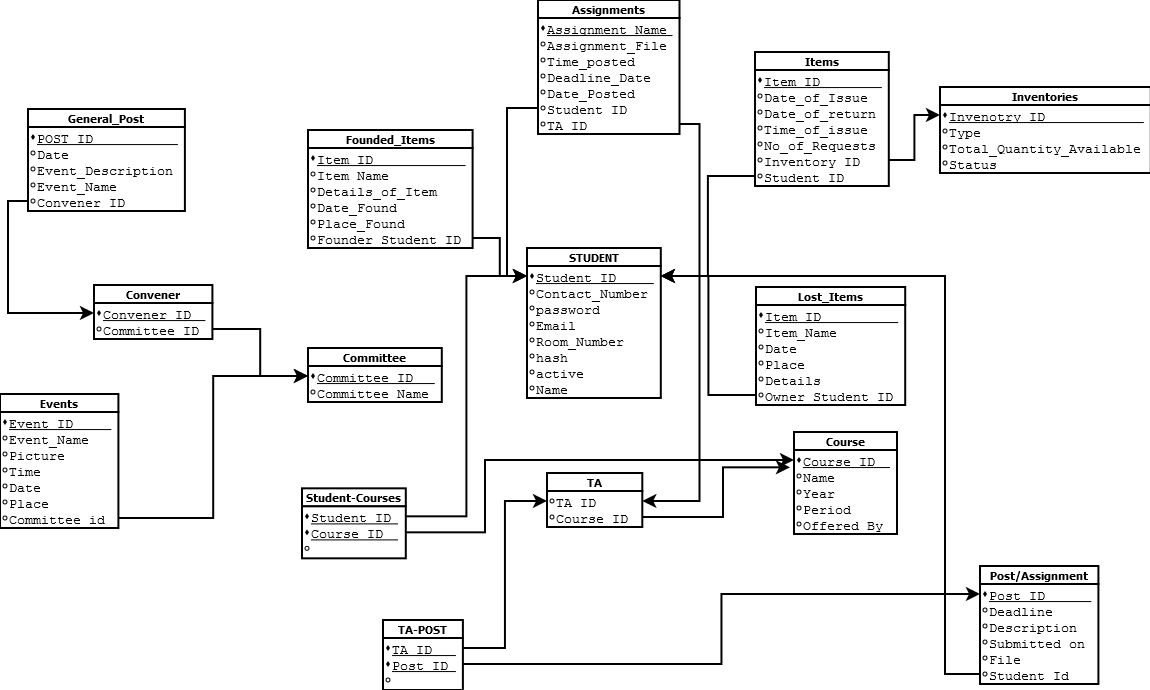
**2.2 Database Design**

2.2.1 ERD

Entity Relationship Diagram i.e. the ERD describes the database implementation in an abstract way, which is really important for the end user or client to understand the working of the system without actually getting into technicalities of the databases.



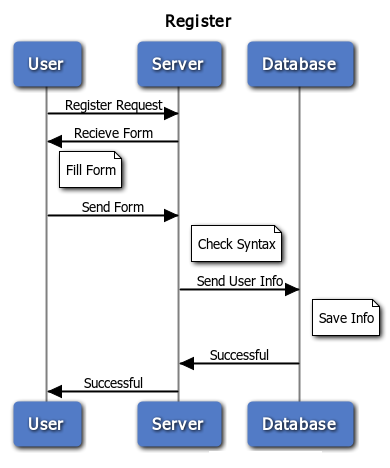
2.2.2 Relational Model



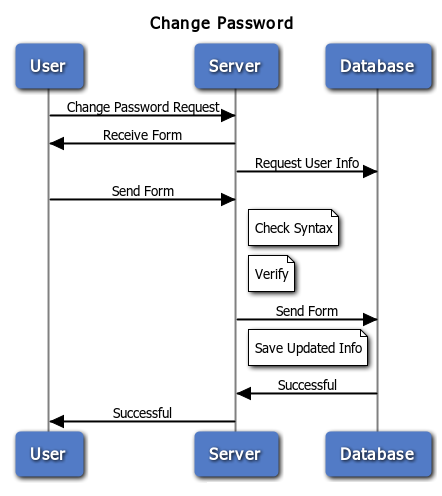
2.2.3 Sequence Diagrams

These are interaction diagrams that show how processes operate with one another and in what order. It shows object interactions on a timeline.

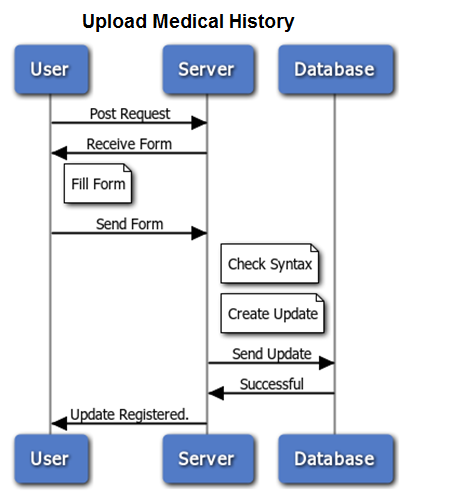
2.2.3.1 Register



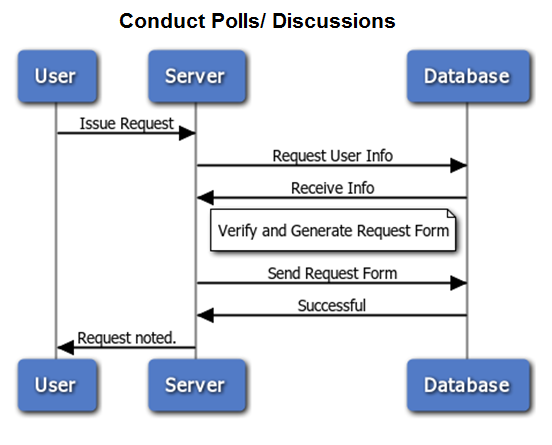
2.2.3.2 Change Password



2.2.3.3 Upload Medical History



2.2.3.4 Conduct Polls/ Discussions

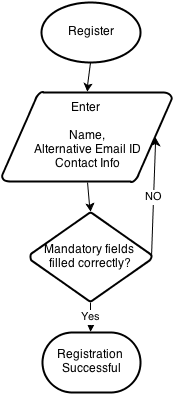


**3. Low Level Design**

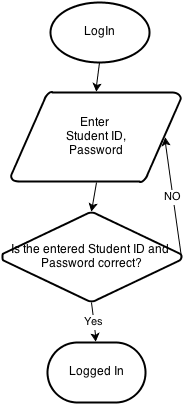
3.1 Activity Diagrams

Activity diagrams are graphical representations of workflows of stepwise activities and actions. These are used to describe the operational step-by-step workflows of components in a system. An activity diagram shows the overall flow of control.

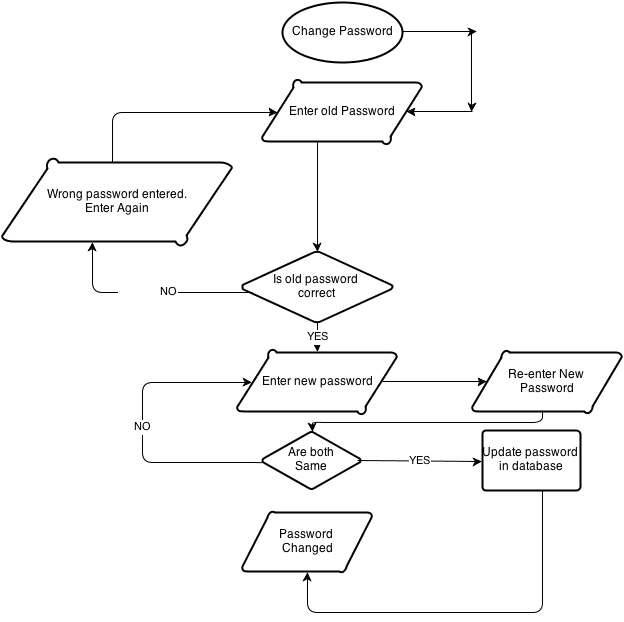
3.1.1 Register



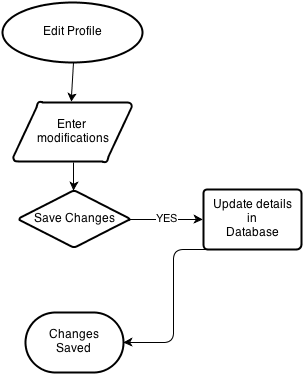
3.1.2 Log In



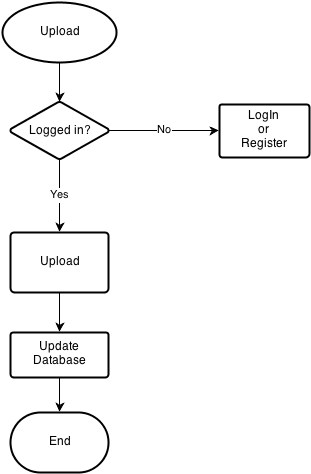
3.1.3 Change Password



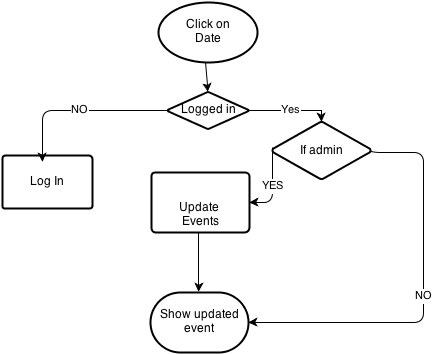
3.1.4 Edit Profile



3.1.5 Upload Medical History

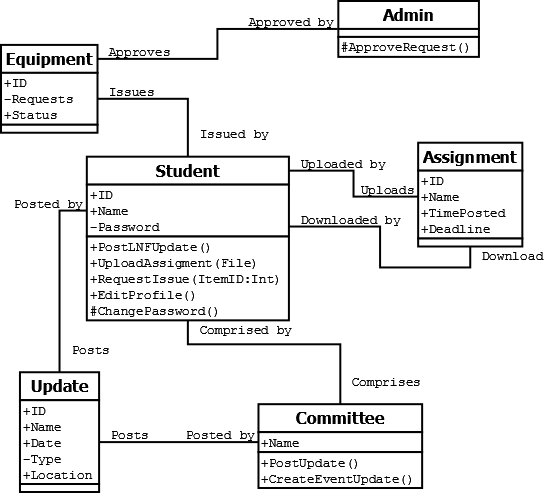


3.1.7 Post Medical Information Updates



3.2 Class Diagram

A class diagram is a type of static structure diagram that describes the structure of a system by showing the system's classes, their attributes, methods, and the relationships among them.



**4. Human Interface Design**

The webpages in the website are divided into 8 sections based on their contents.  
The groups are   
**Evaluate Symptoms,**   
**Ailments,**   
**Health Tips,**   
**First-Aid,**

**Links,**

**Medical History,**

**Polls** and   
**Discussions.**

The webpages are divided into sections namely,  
**Myth Busters** and   
**The** **Central Canvas.**

There are additional webpages for Administrator account which has the functionalities of managing the system.