**Software Requirements Specification**

**Hospital management system**

Yasas Ranawaka 2015/cs/111 15001113

# Introduction

## Purpose

## The purpose of this document is to describe both functional and non-functional requirements of the Hospital Management System. The main purpose of our system is to make hospital task easy and to develop software that replaces the manual hospital system into automated hospital management system.

## 1.2Project Scope

The system is labeled as the Hospital Management System (HMS). The Hospital Management System is manage patients’ registration and to assign doctors to patients in designated wards as need arises. HMS is keep the details of the treatments that are prescribed for the patients, and its results, details of the test and prescribe test for patients and its results. The HMS is manage the employees’ registration and keep the track of attendance. The HMS keeps the records of payment details. Doctors also use the system to view the details of tests, view the details of the wards and view the details of prescribed treatments for patients.

**1.3 Acronyms, and Abbreviations:**

a. HMS: Hospital Management System  
b. SRS: Software Requirements Specification.  
c. GUI: Graphical User Interface.

# Overall System Description

## Product Perspective

This Hospital Management System is self-contained system that manages some activities of the hospital.

**2.2 Product Features**

The project maintains two levels of users:-

• Administrator Level

• Doctor Level

The system functions can be describes as follows:

Registration:

Patients: When a patient is registered, there are should be a ward and an assigned consultant. Also when a patient is registered, there should be a patient’s name and contact details definitely.

Employee: When a nurse is registered, there are should be a ward. Also when an employee is registered, there should be a employee’s name and contact details definitely.

Generation: The System generates the following information

List of detailed information regarding the patients who are admitted in the hospital.

List of detailed information regarding the employees who are registered in the hospital.

List of the employees’ attendance details.

List of detailed information regarding the wards that are in the hospital.

List of detailed information regarding the tests that are exist in the hospital.

List of detailed information regarding the treatments that are prescribed to the patients.

List of detailed information regarding the payments that patients paid.

Searching:

A patient’s details can be searched from the list of patients or entering the patient’s ID.

Also employee’s details, ward details, test details can be searched from their lists.

When enter the patient’s ID, you can see his/her all payments details and its total.

When enter the patient’s ID, you can see his/her all treatment and details.

**2.3 User Characteristics**

Every user should be:

• Comfortable of working with computer.

• He must have knowledge in medical field.

• He must also have basic knowledge of English too.

**2.4 Design and Implementation Constraints**

\* Login and password is used for identification of user and there is no facility for guest.

\* The system use the MYSQL Database which is open source and free.

\* The system is stand-alone application.

# System Features

**3.1 Functional Requirements**

**Use case**: Login of Admin

++ Primary Actor: Administrator

++Pre-Condition: Application should be started

++ Main Scenario:

\*Admin provides the username and the password.

\*Click the ‘Login’ command button.

\*System does authentication.

\*Main application relevant to admin is displayed.

**Use case**: Register/Delete/Update the Patient

++ Primary Actor: Administrator

++Pre-Condition: Application should be started and login

++ Main Scenario:

\*Click the ‘Patient’ command button from Admin Home page.

\*Main application relevant to patient is displayed.

\*Enter patient’s details, click ‘Register/Delete/Update’ button

**Use case**: Search the Patient’s Details

++ Primary Actor: Administrator

++Pre-Condition: Application should be started and login

++ Main Scenario:

\*Click the ‘Patient’ command button from Admin Home page.

\*Main application relevant to patient is displayed.

\*Enter patient Id details, click ‘Search button OR

Click the Patient in the patients table

**Use case**: Add/Delete/Update the Ward

++ Primary Actor: Administrator

++Pre-Condition: Application should be started and login

++ Main Scenario:

\*Click the ‘Ward’ command button from Admin Home page.

\*Main application relevant to ward is displayed.

\*Enter details of ward, click ‘Add/Delete/Update’ button

**Use case**: Search the Details of Ward

++ Primary Actor: Administrator

++Pre-Condition: Application should be started and login

++ Main Scenario:

\*Click the ‘Ward’ command button from Admin Home page.

\*Main application relevant to ward is displayed.

\* Click the Ward in the patients table.

**Use case**: Register/Delete/Update the Employee

++ Primary Actor: Administrator

++Pre-Condition: Application should be started and login

++ Main Scenario:

\*Click the ‘Employee’ command button from Admin page.

\*Main application relevant to employee is displayed.

\* Click the type of the occupation

\*Enter employee’s details, click ‘Register/Delete/Update’

**Use case**: Search the Employee’s Details

++ Primary Actor: Administrator

++Pre-Condition: Application should be started and login

++ Main Scenario:

\*Click the ‘Patient’ command button.

\*Main application relevant to employee is displayed.

\* Click the employee in the Employees table.

**Use case**: Add/Delete/Update the Test

++ Primary Actor: Administrator

++Pre-Condition: Application should be started and login

++ Main Scenario:

\*Click the ‘Test’ command button from Admin Home page

\*Main application relevant to Test is displayed.

\*Enter details of Test, click ‘Add/Delete/Update’ button

**Use case**: Search the Details of Test

++ Primary Actor: Administrator

++Pre-Condition: Application should be started and login

++ Main Scenario:

\*Click the ‘Test’ command button.

\*Main application relevant to Test is displayed.

\*Click the test in the Test table.

**Use case**: Add/Delete/Update the Test for relevant patient

++ Primary Actor: Administrator

++Pre-Condition: Application should be started and login

++ Main Scenario:

\*Click the ‘Test’ command button.

\*Main application relevant to Test is displayed.

\*Enter the relevant patient ID, relevant Test ID and other

-details, click ‘Add/Delete/Update’ button.

**Use case**: Search the patient’s tests information

++ Primary Actor: Administrator

++Pre-Condition: Application should be started and login

++ Main Scenario:

\*Click the ‘Patient’ command button.

\*Main application relevant to Test is displayed.

\* Enter the relevant patient ID and relevant Test ID,

\*Click ‘Search’ button.

**Use case**: Add the Treatment for relevant patient

++ Primary Actor: Administrator

++Pre-Condition: Application should be started and login

++ Main Scenario:

\*Click the ‘Treatment’ button.

\*Main application relevant to Treatment is displayed.

\*Enter the relevant patient ID and other details

\*Click ‘Add’ button.

**Use case**: Update /Delete the Treatment for relevant patient

++ Primary Actor: Administrator

++Pre-Condition: Application should be started and login

++ Main Scenario:

\*Click the ‘Treatment’ button.

\*Main application relevant to Treatment is displayed.

\*Enter the relevant patient ID and relevant Treatment No

\*Click ‘Add/Delete/Update’ button.

**Use case**: Search the patient’s treatments information

++ Primary Actor: Administrator

++Pre-Condition: Application should be started and login

++ Main Scenario:

\*Click the ‘Patient’ command button.

\*Main application relevant to Treatment is displayed.

\* Enter the relevant patient ID and relevant Treatment No,

\*Click ‘Search’ button.

**Use case**: Add the Payment for relevant patient

++ Primary Actor: Administrator

++Pre-Condition: Application should be started and login

++ Main Scenario:

\*Click the ‘Payment’ button.

\*Main application relevant to Payment is displayed.

\*Enter the relevant patient ID and other details

\*Click ‘Add’ button.

**Use case**: Update /Delete the Payment for relevant patient

++ Primary Actor: Administrator

++Pre-Condition: Application should be started and login

++ Main Scenario:

\*Click the ‘Treatment’ button.

\*Main application relevant to Payment is displayed.

\*Enter the relevant patient ID and relevant Payment No

\*Click ‘Add/Delete/Update’ button.

**Use case**: Search the patient’s Payments information

++ Primary Actor: Administrator

++Pre-Condition: Application should be started and login

++ Main Scenario:

\*Click the ‘Patient’ command button.

\*Main application relevant to Payment is displayed.

\* Enter the relevant patient ID and Click ‘Search’ button.

**Use case**: Add the Attendance for relevant employee

++ Primary Actor: Administrator

++Pre-Condition: Application should be started and login

++ Main Scenario:

\*Click the ‘Payment’ button.

\*Main application relevant to Attendance is displayed.

\*Enter the relevant Employee ID and other details

\*Click ‘Add’ button.

**Use case**: Update /Delete the Attendance for relevant Employee

++ Primary Actor: Administrator

++Pre-Condition: Application should be started and login

++ Main Scenario:

\*Click the ‘Treatment’ button.

\*Main application relevant to Attendance is displayed.

\*Enter the relevant Employee ID and relevant Payment No

\*Click ‘Add/Delete/Update’ button.

**Use case**: Search the Employee’s Attendances information

++ Primary Actor: Administrator

++Pre-Condition: Application should be started and login

++ Main Scenario:

\*Click the ‘Patient’ command button.

\*Main application relevant to Attendance is displayed.

\* Enter the relevant Employee ID and Click ‘Search’ button.

**Use case**: Login of Doctor

++ Primary Actor: Doctor

++Pre-Condition: Application should be started

++ Main Scenario:

\*Doctor provides the username and the password.

\*Click the ‘Login’ command button.

\*System does authentication.

\*Main application relevant to Doctor is displayed.

**Use case**: Search the Details of Ward

++ Primary Actor: Doctor

++Pre-Condition: Application should be started and login

++ Main Scenario:

\*Click the ‘Ward’ command button from Admin Home page.

\*Main application relevant to ward is displayed.

\* Click the Ward in the patients table.

**Use case**: Search the Details of Test

++ Primary Actor: Doctor

++Pre-Condition: Application should be started and login

++ Main Scenario:

\*Click the ‘Test’ command button.

\*Main application relevant to Test is displayed.

\*Click the test in the Test table.

**Use case**: Search the patient’s treatments information

++ Primary Actor: Doctor

++Pre-Condition: Application should be started and login

++ Main Scenario:

\*Click the ‘Patient’ command button.

\*Main application relevant to Treatment is displayed.

\* Enter the relevant patient ID and relevant Treatment No,

\*Click ‘Search’ button.

**3.2 Non-Functional Requirements**

**Security**

Any user who uses the system shall have a User Name and  
Password. The project maintains two levels of users:-

• Administrator Level

• Doctor Level

Administrators shall be able to add, view and modify all  
information in system.

Doctors shall be able to only view Wards, Tests and Treatment  
information in system

**Maintainability**

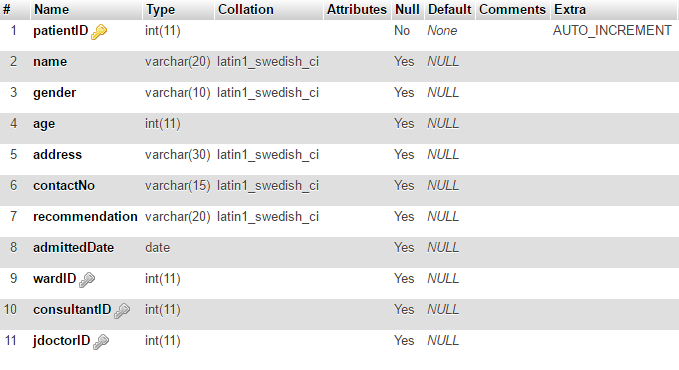
There are data in database. Then data can be maintain anytime.

**Availability**

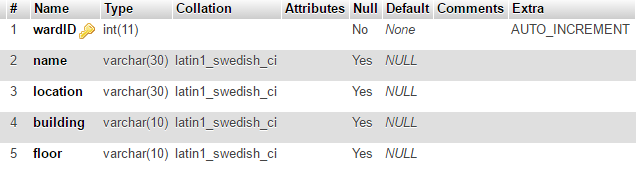
The system shall be available all the time.

**3.2 Database Tables**

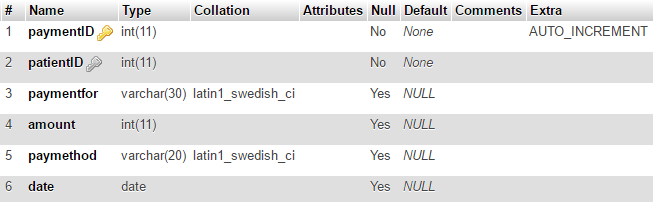
**1. Patient Table**



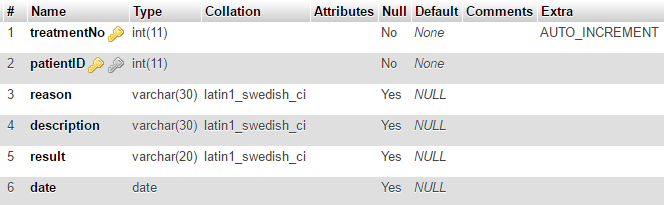
**2. Ward Table**



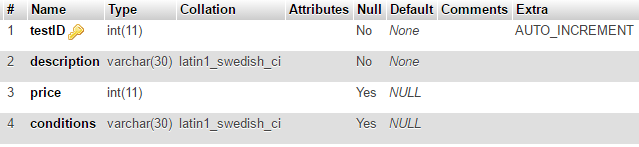
**3. Payment Table**



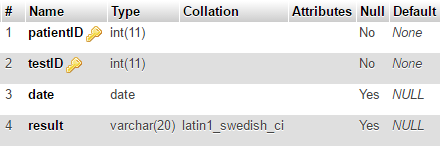
**4. Treatment Table**



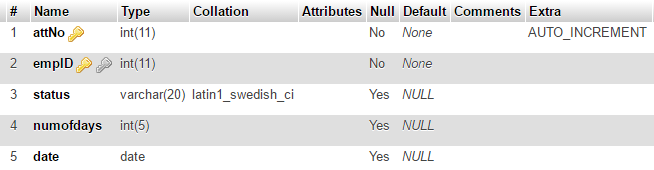
**5. Test Table**



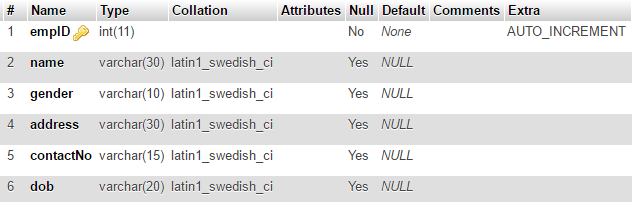
**6. Patient Test Table**



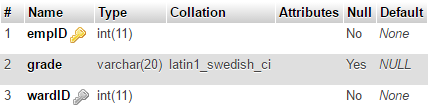
**7. Attendance Table**



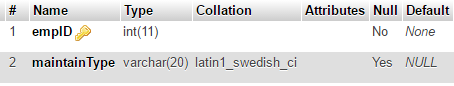
**8. Employee Table**



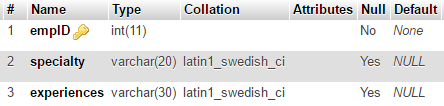
**9. Nurse Table**



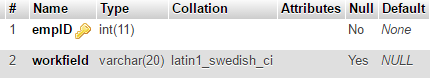
**10. Maintain Staff Table**



**11. Consultant Table**

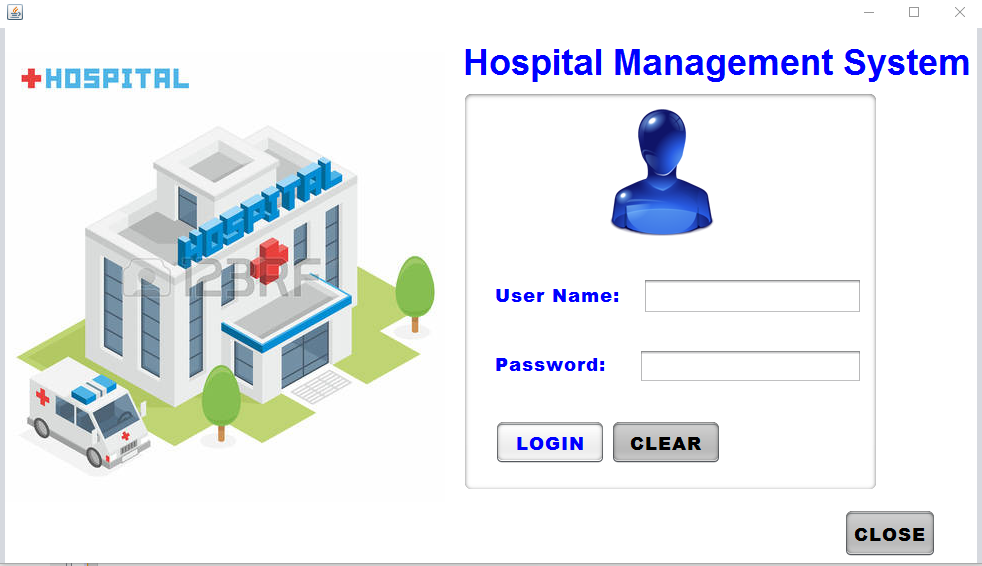


**12. Junior Doctor Table**



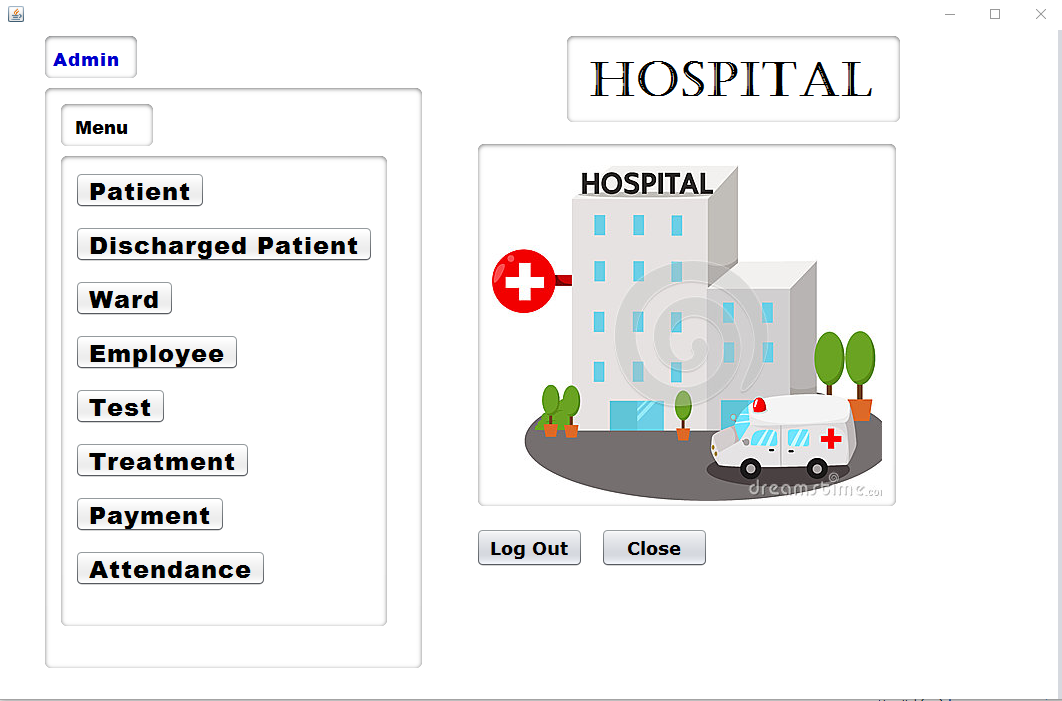
# Interface Requirements

**4.1. Login Interface**

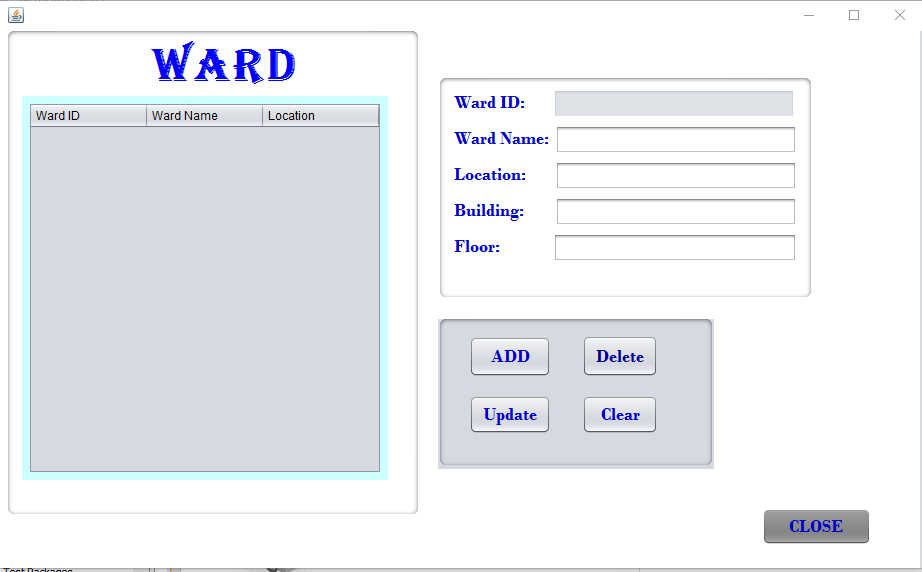


**4.2. User Interface for Admin**

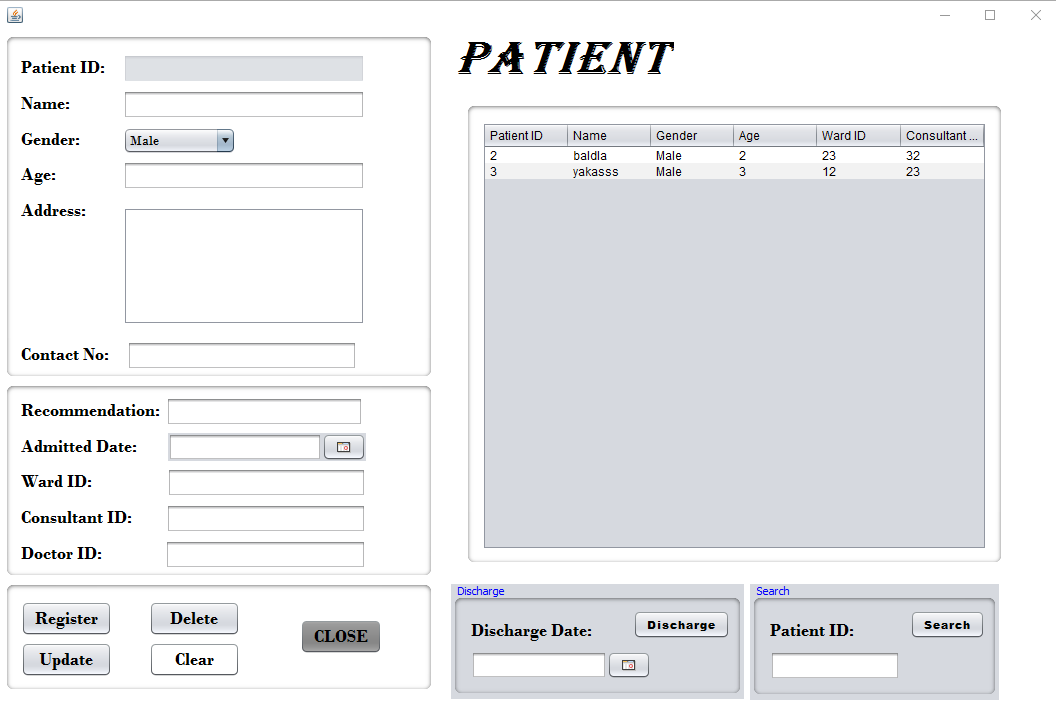
**4.2.1. Admin Home page**



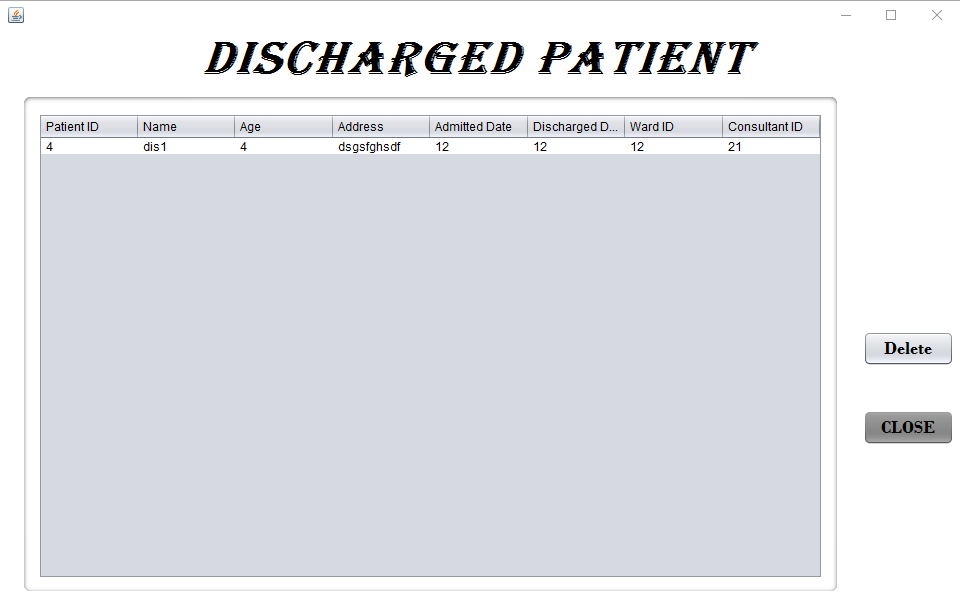
**4.2.2. Ward**



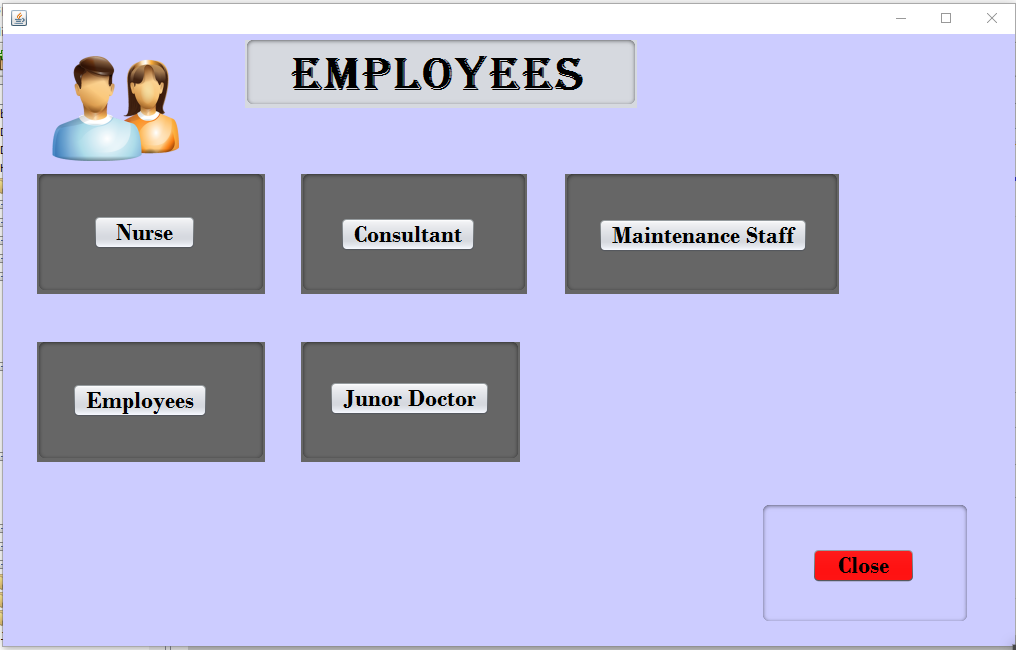
**4.2.3. Patient**



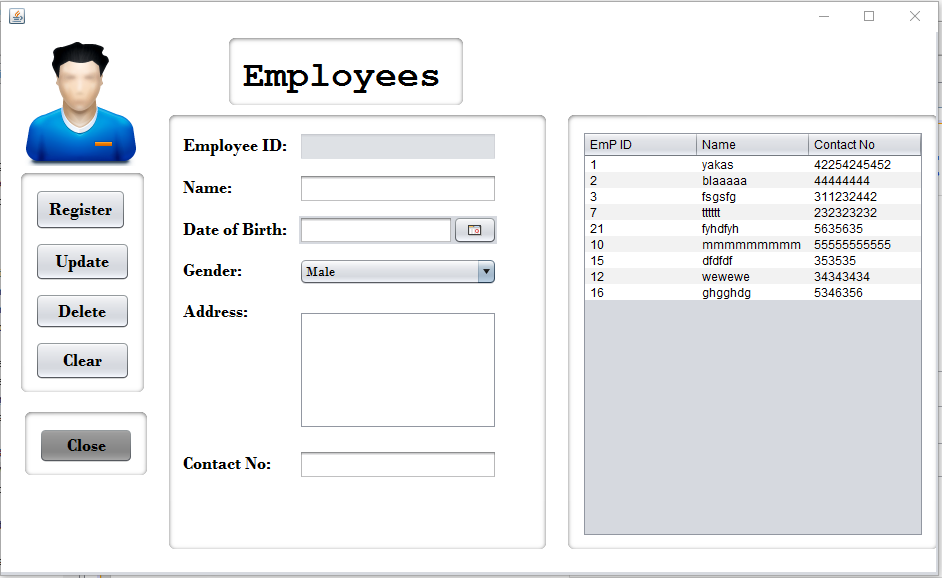
**4.2.4. Discharged Patient**



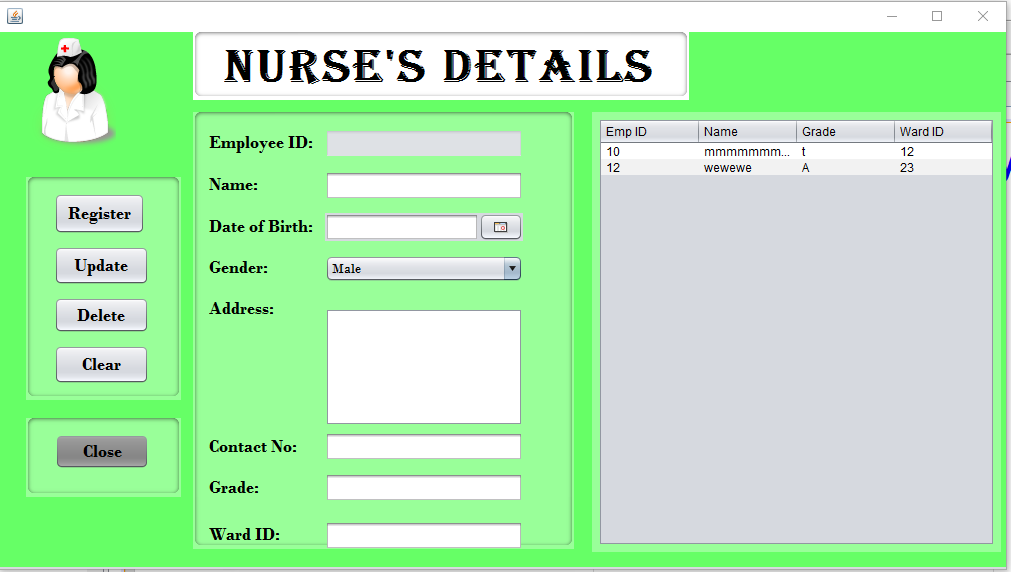
**4.2.4. Employee 1st page**



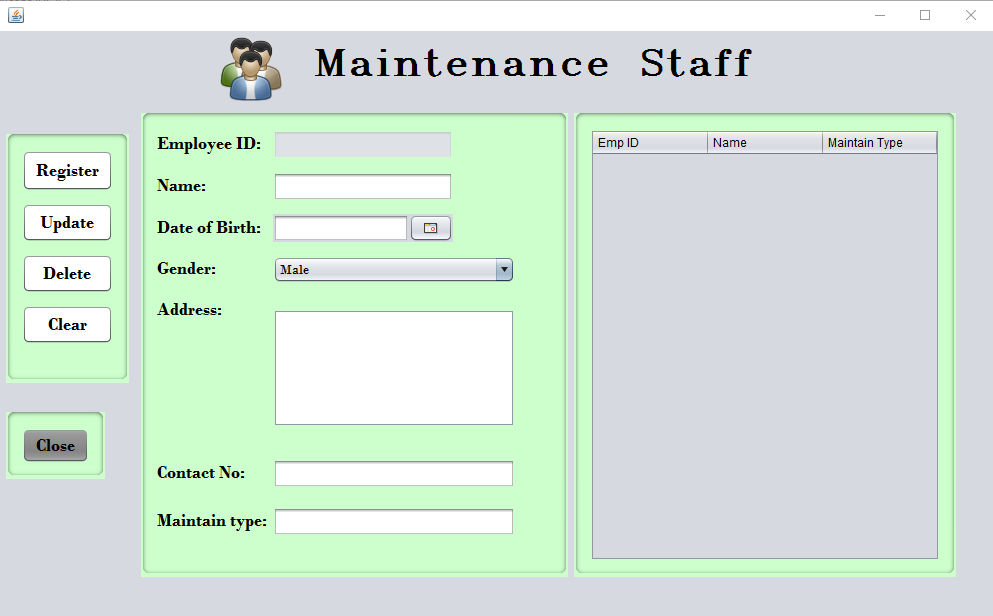
**4.2.5. Employee**



**4.2.6. Nurse**



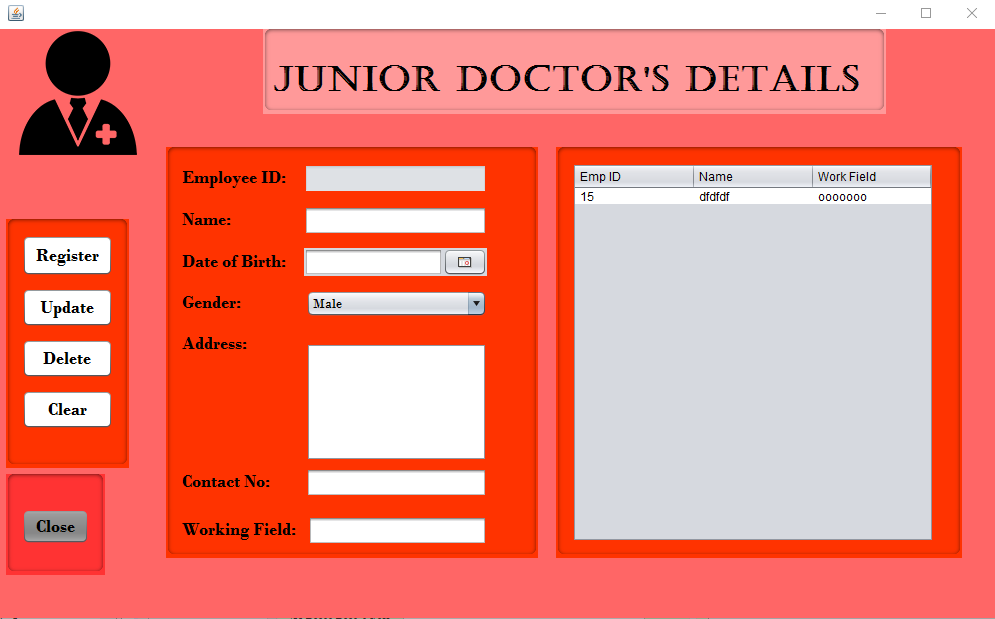
**4.2.7. Maintaining Staff**



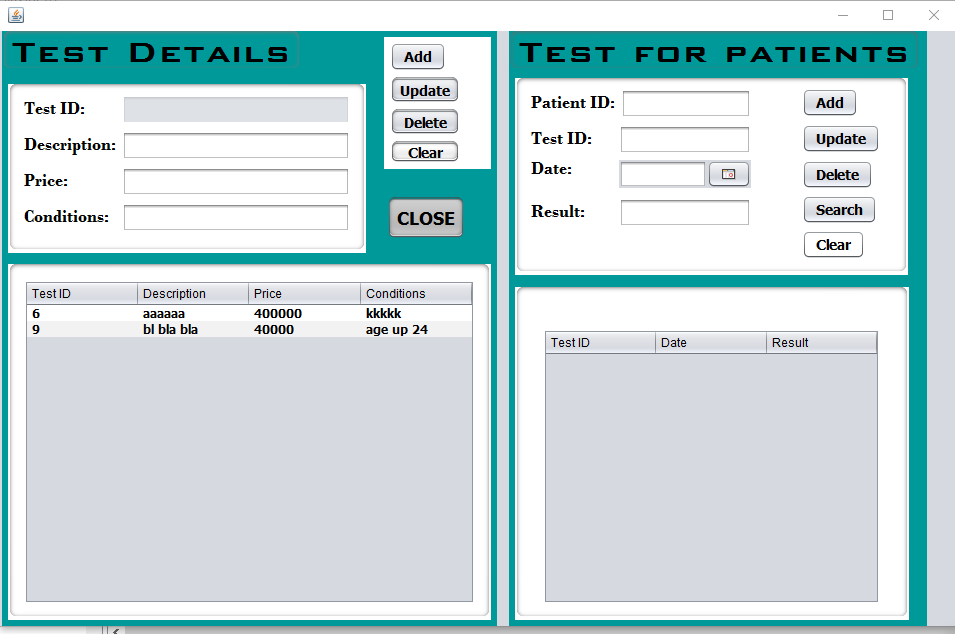
**4.2.8. Consultant**



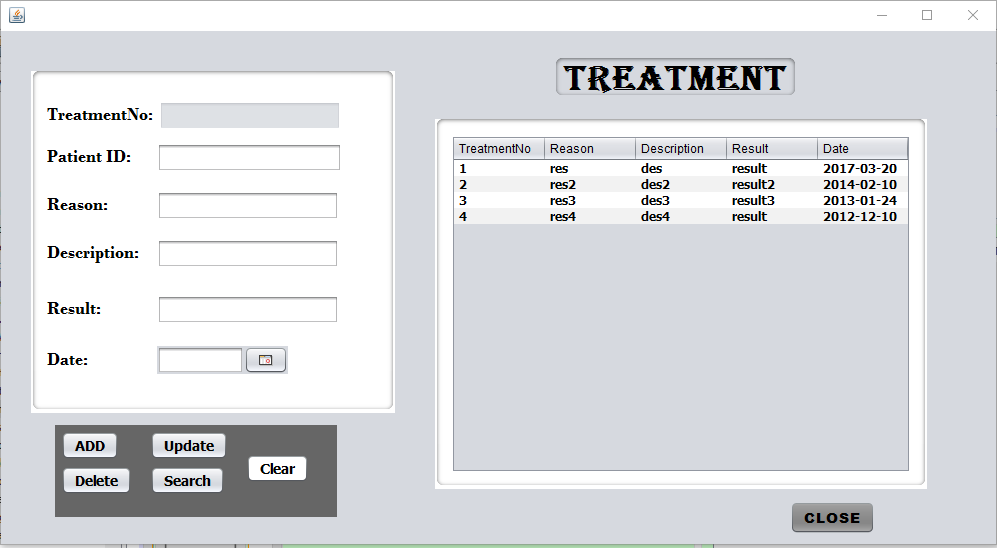
**4.2.9. Junior Doctor**



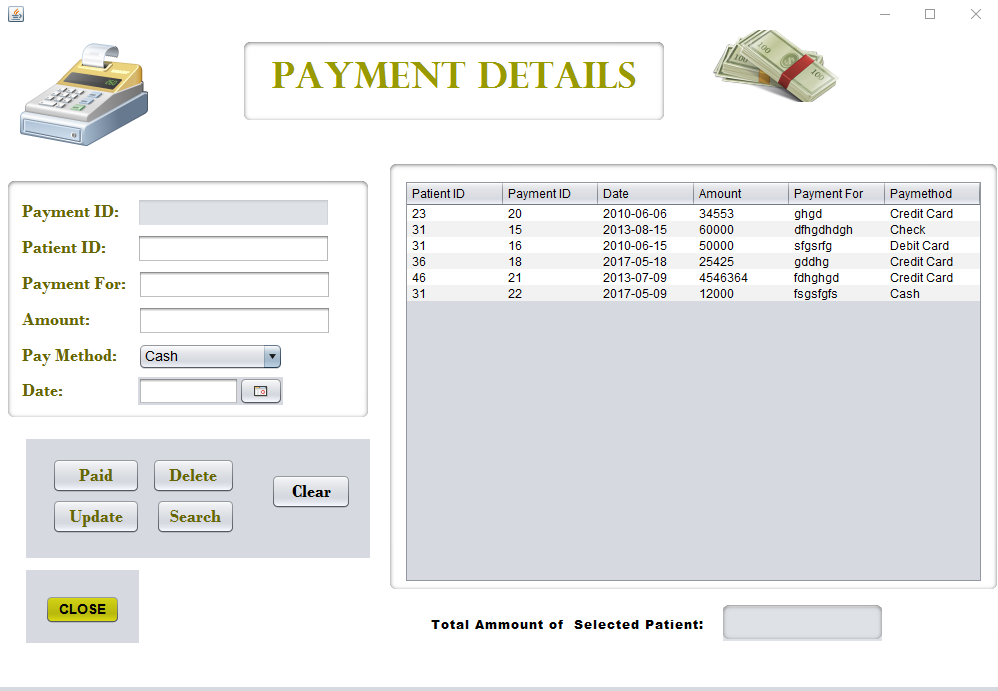
**4.2.10. Test**



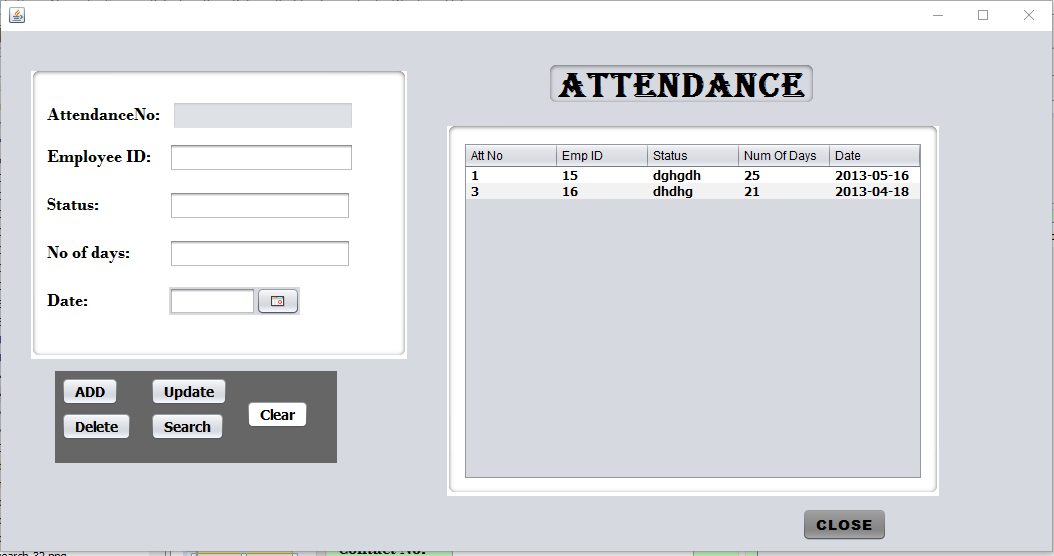
**4.2.11. Treatment**



**4.2.12. Payment**



**4.2.13. Attendance**



**4.3. User Interface for Doctor**

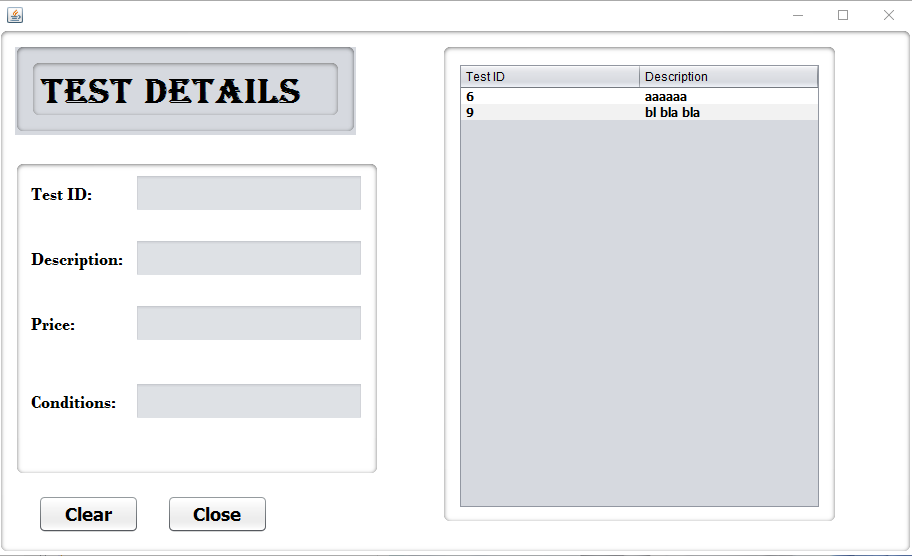
**4.3.1. Doctor Home Page**



**4.3.2. Doctor Ward**



**4.3.3. Doctor View Test Details**



**4.3.4. Doctor View Treatment Details**

