1. **What is the purpose of this project?**
   The purpose of this document is to gather requirements, outline the technical design and provide a comprehensive architectural overview for Online Medical Center Management System version 1.0 (SUST medical center automation). The objective of this document is to give a detailed understanding of the system which can be understood by managers and programmers.

2. **What is the PROBLEM your project is trying to solve?**
   Currently, Our Medical Centre uses manual (primitive) Management System for maintaining the patient demography and distributing medicine to the patient. In the existing system, doctors and other employees have to spend a lot of time providing services to the patient because a lot of paper is used to record information. Here, prescribing patients, delivering medicine, maintaining medicine stock, retrieving records etc. everything is tremendously manual and paper dependent. So an automated online management system needs to be developed.

3. **Give an overview of the project.**
   The proposed system will provide a graphical user interface to maintain the total system, including prescribing patients, delivering medicine, maintaining medicine stock etc. Moreover, the new system will be accessible from terminals within the Medical Center and also through the internet from computers outside the Medical Center. Besides, the patients (both students and staff) can view their prescription through the internet from anywhere.

4. **Explain about different users and related services.**
   There are two generic types of user in the system.
   - General user (Patient): There are two types of general user. a) Staff: All valid employees of the university who will be familiar with web browsing. b) Student: All current students of the university who will be familiar with web browsing.
   - Administrative user: There are four types of administrative user who will interact with the system both using the Medical Center terminal and the Internet.
     - Level 1: Doctors (Medical Officer/Senior Medical Officer/Junior Medical Officer): All doctors who have a little bit of knowledge in web browsing.
     - Level 2 : Senior Pharmacist/Store Officer An employee who has knowledge on web browsing and medicine. Senior pharmacists have little technical knowledge about the system.
     - Level 3: Medicine Distributor. An employee who has knowledge on web browsing. Also have some knowledge about medicine.

5. **Explain about some of the important product features:**

1. Stock Management (for medicine): a. Due to the maintenance simplicity and management security stock of medicine is divided in two sub categories Central Stock and Sub Stock. All records of stock medicines are maintained through two subcategory central- store and sub-store. b. At first, newly bought medicines are added to the Central Stock and then transferred into Sub Stock as needed.

```
/*
 * To change this template, choose Tools | Templates
 * and open the template in the editor.
 */

/**
 *
 * @author Administrator
 */
package medicalcenter;

import java.sql.ResultSet;

public class StockLedgerEntry {
    ResultSet rs;
    String query;

    public StockLedgerEntry() {
        rs=null;
        query=null;

    }


        public ResultSet getInfo(String fld,String tbl){
            query="select distinct("+fld+") from "+tbl;
            try{
            database db=new database();
            rs=db.executeQuery(query);
            }catch(Exception e){

                }

            return rs;
        }
```

```
        public ResultSet getInfo(String fld1,String fld2,String tbl){
            query="select " +fld1+ "," +fld2+ " from "+tbl+" order by " +fld2;
            try{
            database db=new database();
            rs=db.executeQuery(query);
            }catch(Exception e){

                }

            return rs;
        }

        public ResultSet getCentralStocInfo(String viewName){
            query="select * from "+viewName;
            try{
            database db=new database();
            rs=db.executeQuery(query);
            }catch(Exception e){

                }
            return rs;
        }
        public ResultSet getUserProfile(String userPk){
            query="SELECT full_name,employee_code,designation_name,dept_name,sex"+
" FROM employee,employee_personal_info,employee_status,designation,department"+
" where employee_pk=employee_personal_pk_fk"+
" and employee_pk= employee_status_pk_fk"+
" and designation_pk=designation_fk"+
" and department_fk=dept_pk"+
" and employee_pk="+userPk;
            try{
            database db=new database();
            rs=db.executeQuery(query);
            }catch(Exception e){

                }
            return rs;
        }

        public ResultSet getPrescriptionDate(String stdPk){
```

```
            query="SELECT std_prescription_pk,prescription_dt FROM
std_prescription_info"
+" where student_pk_fk="+stdPk
+" order by prescription_dt desc";
            try{
            database db=new database();
            rs=db.executeQuery(query);
            }catch(Exception e){

                }
            return rs;
        }
        public ResultSet getDiagnosisDetails(String presPk){
            query="SELECT
prescription_dt,diagnosis_detail,reconsult_dt,general_advice,full_name"+
" FROM std_prescription_info,employee_personal_info"+
" where std_prescription_pk="+presPk+
" and emplaoyee_pk_fk=employee_personal_pk_fk;";
            try{
            database db=new database();
            rs=db.executeQuery(query);
            }catch(Exception e){

                }
            return rs;
        }
        public ResultSet getPatientMedicine(String presPk){
            query="SELECT
med_type,med_com_name,med_weight,no_of_doses,day_duration,med_qty,medication_i
nst_text"+
" FROM
patient_med_info,medicine_gen_info,medicine_com_info,medication_inst_detail"+
" where std_prescription_fk="+presPk+
" and med_com_name_fk=med_com_name_pk"+
" and med_gen_name_fk=med_gen_name_pk"+
" and medication_inst_fk=medication_inst_pk"+
" order by med_type asc";
            try{
            database db=new database();
            rs=db.executeQuery(query);
```

```
            }catch(Exception e){

            }
            return rs;
        }
}
```

2. Entry New Purchased Medicine: a. Pharmacist entry the newly purchased medicine at Central Stock Ledger providing Company Name, Purchased Date, Commission, Medicine Type, Medicine Name, Medicine Quantity, Per piece price, Manufacturing Date, Expiring Date. b. Pharmacists also enter the location of medicine at central stock.

3. Stock Transfer : a) Pharmacists can transfer medicine from Central Stock to Sub Stock as per vacancy of medicine at Sub Stock with the permission of Chief Medical Officer. i. Pharmacists can transfer a single medicine less or equal to the available quantity at Central Stock. ii. S/he can transfer multiple medicines simultaneously less or equal to the available quantity at Central Stock. b) Pharmacists can also update medicine location at sub stock.

```
package servlet;

import java.io.IOException;
import java.io.PrintWriter;
import javax.servlet.ServletException;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
import medicalcenter.database;

/**
 *
 * @author Administrator
 */
public class StocTransfer extends HttpServlet {

    /**
```

```
    * Processes requests for both HTTP <code>GET</code> and <code>POST</code>
methods.
    * @param request servlet request
    * @param response servlet response
    * @throws ServletException if a servlet-specific error occurs
    * @throws IOException if an I/O error occurs
    */
   protected void processRequest(HttpServletRequest request, HttpServletResponse
response)
   throws ServletException, IOException {
       response.setContentType("text/html;charset=UTF-8");
       PrintWriter out = response.getWriter();
       int mId=0,tQty=0;
       String val,cnd=null;

       String [] medId = request.getParameterValues("med_id");
       System.out.println("YES          fg");
       String [] centralQty = request.getParameterValues("central_qty");
       String [] transferQty = request.getParameterValues("transferred_qty");

       try {
           database db=new database();

               for(int i=0; medId != null && i < medId.length; i++)
{//transferQty[i].equals("") || transferQty[i]== null
                   if(transferQty[i].equals("") || transferQty[i]== null)
                       continue;
                   mId=Integer.parseInt(medId[i]);
                   tQty=Integer.parseInt(transferQty[i]);
                   System.out.println("= " +tQty+" "+mId);
                   db.executeProcedure("transfer_stock",mId,tQty);

           //System.out.print(" "+medId[i] );
            //   System.out.print(" "+centralQty[i] );
            //   System.out.print(" "+transferQty[i] );
             //  System.out.println("\n");
               }
```

```
getServletConfig().getServletContext().getRequestDispatcher("/transfer_successful.jsp").
forward(request, response);
        }catch(Exception e){
            System.out.println("Error :"+e);
        }
        finally {
            out.close();
        }
    }

    // <editor-fold defaultstate="collapsed" desc="HttpServlet methods. Click on the +
sign on the left to edit the code.">
    /**
     * Handles the HTTP <code>GET</code> method.
     * @param request servlet request
     * @param response servlet response
     * @throws ServletException if a servlet-specific error occurs
     * @throws IOException if an I/O error occurs
     */
    @Override
    protected void doGet(HttpServletRequest request, HttpServletResponse response)
    throws ServletException, IOException {
        processRequest(request, response);
    }

    /**
     * Handles the HTTP <code>POST</code> method.
     * @param request servlet request
     * @param response servlet response
     * @throws ServletException if a servlet-specific error occurs
     * @throws IOException if an I/O error occurs
     */
    @Override
    protected void doPost(HttpServletRequest request, HttpServletResponse response)
    throws ServletException, IOException {
        processRequest(request, response);
    }

    /**
```

```
     * Returns a short description of the servlet.
     * @return a String containing servlet description
     */
    @Override
    public String getServletInfo() {
        return "Short description";
    }// </editor-fold>

}
```

4. Stock View and Monitor : a. Pharmacists can view both Central Stock and Sub Stock and also monitor the flow of medicine. b. Medicine distributors can only view Sub Stock.

```
 public ResultSet getCentralStocInfo(String viewName){
            query="select * from "+viewName;
            try{
            database db=new database();
            rs=db.executeQuery(query);
            }catch(Exception e){

                }
            return rs;
        }
```

5. Patient Detection and Authentication : a. Students: Doctors authenticate and detect students as legal patients by Digital Image, Registration Number, Department Name, Semester, Address, and Age. b. Staff: Doctors authenticate and detect employees as legal patients by Digital Image, Employee Code, Department Name, Address, Age, and Designation.

```
 public ResultSet getUserProfile(String userPk){
            query="SELECT full_name,employee_code,designation_name,dept_name,sex"+
" FROM employee,employee_personal_info,employee_status,designation,department"+
" where employee_pk=employee_personal_pk_fk"+
" and employee_pk= employee_status_pk_fk"+
" and designation_pk=designation_fk"+
" and department_fk=dept_pk"+
" and employee_pk="+userPk;
            try{
```

```
            database db=new database();
            rs=db.executeQuery(query);
            }catch(Exception e){

            }
            return rs;
        }
```

6. Prescribe Patients : a. Doctors prescribe the patient by providing following information to system i. Diagnosis Details : Chief Complains On Examination Investigation ii. Medicine Details : Medicine Type Medicine Name Time interval Amount per interval Medication instruction Quantity Duration iii. General Advice: General advice can be selected from the drop down list as per demand. iv. Re-consultation Date: The doctor selects the interval in day/month/year for re-consulting and the system will automatically calculate the re- consultation date.

```
public ResultSet getDiagnosisDetails(String presPk){
            query="SELECT
prescription_dt,diagnosis_detail,reconsult_dt,general_advice,full_name"+
" FROM std_prescription_info,employee_personal_info"+
" where std_prescription_pk="+presPk+
" and emplaoyee_pk_fk=employee_personal_pk_fk;";
            try{
            database db=new database();
            rs=db.executeQuery(query);
            }catch(Exception e){

            }
            return rs;
        }
```

7. Distributed Medicine : a. Medicine distributors deliver medicine (suggested by doctor) to the patient which is available to Sub Stock. b. If a particular medicine is not currently available at Sub Stock but available at central Stock then --- i. S/he said to the patient to re-collect the medicine after a certain period. ii. S/he sent a request to Pharmacist to transfer the particular medicine from Central Stock to Sub Stock as early as possible.

8. View Prescription : a. Students: i. All students can view all their prescriptions (from the first prescription to last prescription like an archive) by logging their personal

account. ii. If a particular medicine is not provided by a medical center it is shown specially for purchasing from outside. b. Employee : i. All employees can view all their prescriptions (from the first prescription to last prescription like an archive) by logging into their personal account. ii. All employees can view their payment amount of medicine (which is taken from Medical Center) of the current month as well as the previous all month (from archive). iii. If a particular medicine is not provided by a medical center it is shown specially for purchasing from outside. c. Doctors: Doctors can view all prescriptions prescribed by him/her and also view any prescription by dint of searching feature.

```
public ResultSet getPrescriptionDate(String stdPk){
        query="SELECT std_prescription_pk,prescription_dt FROM
std_prescription_info"
+" where student_pk_fk="+stdPk
+" order by prescription_dt desc";
        try{
        database db=new database();
        rs=db.executeQuery(query);
        }catch(Exception e){

        }
        return rs;
    }
```

9. Searching : a. Medicine Location : The location of a particular medicine at Central Stock. The location of a particular medicine at Sub Stock. b. Medicine Information : Find the quantity, manufacturing date, expiring date of a particular medicine. Find all medicine to a particular generic group. Find all expired medicine with their location both in Central Stock to Sub Stock. c. Search Prescription: In a Graphical User Interface a doctor can search any prescription by selecting some searching criteria. S/he can search by date, diagnosis, patient id, doctor's id.

10. Bill Generation and Payment : a. Pharmacists can generate a bill for newly purchased medicine (which is already enrolled to the system) for individual pharmaceutical companies (i.e. medicine suppliers).

11. Possible Medicine List (for purchasing): a. A proposal or suggestion of a new medicine purchase list (which should be purchased in the upcoming month) with quantity will be generated based on the previous record of database. b. An estimated budget for new medicine list will also be generated based on the previous price rate from database.

12. Expired Medicine : All expired medicine are shown in a new tab so that operator can take necessary steps to process them. In both Central Stock and Sub Stock expired medicines are showed separately and these are not deliverable.

13. Payment for Medicine (only for employee) : a. Employee have to pay for their received medicine at Medical Center. An equivalent amount money of their delivered medicine will be discarded form their monthly salary. b. Employee can view his/her total amount of payment for medicine.

```
<%--
    Document   : transfer_successful
    Created on : Apr 22, 2011, 10:42:59 AM
    Author     : Administrator
--%>

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
   "http://www.w3.org/TR/html4/loose.dtd">

<%
    try
    {
        if(Integer.parseInt(session.getAttribute( "desig_id" ).toString()) != 11 )
        {
%>
    <jsp:forward page="InvalidPage.jsp"></jsp:forward>
        <%}%>
<%}
    catch(Exception e)
    {
%>
    <jsp:forward page="InvalidPage.jsp"></jsp:forward>
<%}%>

<%@page import="java.sql.ResultSet"%>
<jsp:useBean id="medicalcenter" class="medicalcenter.StockLedgerEntry"
scope="session"/>
<%@page contentType="text/html" pageEncoding="UTF-8"%>
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
   "http://www.w3.org/TR/html4/loose.dtd">

<%@include file="include/header.jsp" %>
```

```
<%@include file="include/pharmacist_menu.jsp" %>
<%@include file="include/alwaysinclude.jsp" %>

<div id="page_title">
                    Transfer Successful
</div>
<div id="page_body">
          <br>
<form method="post" action="http://localhost:8084/MedicalCentre/StocTransfer"
onsubmit="return checkValidity();" >
    <table height="300" border="0" width="100%">
        <tr align="center">
            <td align="center">
                <p>
                    Your transfer has been successfully completed.....
                </p>
            </td>
        </tr>
    </table>

</form>
<br><br>
</div>

<%@include file="include/footer.jsp" %>
```

14. Medicine Location : a) Every medicine has a predefine location both in the Central Stock to Sub Stock (e.g. Napa: Room no-203, Almera no-101 , Shelf no-5). b) Operators can change the location of a particular medicine and also be able to define a new location for a newly arrived medicine. c) Medicine distributors will be able to see the location and also be able to find the location of a particular medicine.

15. Notification and Alert : a. When the quantity of a particular medicine falls below a certain amount both in Central Stock and Sub Stock, a notification will be generated automatically. b. When the expiration date of a particular medicine is close, then send an alert to the pharmacist so that s/he can distribute this medicine.

16. Report Generation: a. Different types of reports will be generated based on the criteria (e.g. flow of a particular medicine at a particular period). A third party software (e.g. Crystal Reports) may be used or integrated to the system.