## 1. INRTODUCTION

## 1.1 Purpose

This document contains the complete design description of the Outpatient Records Management System. This includes the architectural features of the system down through details of what operations each code module will perform and the database layout. It also shows how the use cases detailed in the SRS will be implemented in the system using this design.

The primary audiences of this document are the software developers.

This document can also be useful to the users, for the sake of understanding the software, The stakeholders of the project and anyone else who would be interested in knowing how the ORM system works.

## 1.2 Scope

## 1.3 Overview

* Chapter 2 is a Deployment Diagram that shows the physical nodes on which the system resides. This allows a clear explanation of where each design entity will reside. No design unit may straddle two nodes but must have components on each, which collaborate to accomplish the service.
* Chapter 3 is the Architectural Design. This is the heart of the document. It specifies the design entities that collaborate to perform the functionality of the system. Each of these entities has an Abstract Specification and an Interface that expresses the services that it provides to the rest of the system. In turn each design entity is expanded into a set of lower-level design units that collaborate to perform its services.
* Chapter 4 is the basic Data Structure Design, which for this project is a relational database. While it is separated out here for emphasis, it is really the lowest level of the Architectural Design.
* Chapter 5 is on User Interface Design and discusses the methodology chosen, why it was chosen and why it is expected to be effective.
* Chapter 6 describes the structure of the Help System.
* Chapter 7 exhibits the Use Case Realizations. The implementation of each use case identified in the SRS is shown using the services provided by the design objects.

## 1.4 Reference

## 1.5 Definitions and Acronyms

|  |  |
| --- | --- |
| ***Term*** | ***Meaning*** |
| ORM System | Outpatient Records Management System |
| SRS | Software Requirements Specification |
|  |  |

## 2. SYSTEM OVERVIEW

## 2.1. Deployment Diagram

Web Browser

Hospital Server

Internet

Health Care Staff Processor

Intranet

Database Server

Intranet

Figure 1 - Deployment Diagram

A Health Care staff, say doctor, accesses the Outpatient Records Management System through the Internet using a Web browser (not part of this system although Web pages will run on it). The System resides on a dedicated Hospital server with a permanent Web connection. The Doctor manages all of the record update work on his/her personal computer (Health Care staff processor), communicating with the local database on the Database Server when needed, and uploading completed updates to server when they are ready. Every health care staff Processor contains a local file system.

## 3. SYSTEM ARCHITECTURE

## 3.1. Architectural Design

## 3.2. Decomposition Description

## 3.3. Design Rationale

## 4. DATA DESIGN

## 4.1. Data Description

## 4.2. Data Dictionary

## 5. USER INTERFACE DESIGN

## 5.1. Overview of the User Interface

A user interface is a screen which permits the user to interact with the system.

I n the case of ORMS, the user interfaces are developed with official setup of an hospital system in minds of the developers. Every system user’s home page will be a Login Screen which will require the users to access their pages using their credentials. The Login page will have two text fields which will require the user to enter their staff id and their respective passwords so as to login.

## 5.1.1 Doctor interface

The doctors’ dashboard after login, will comprise of a list of patients on queue, waiting for the doctor’s attention. There will also be three text areas for; recording prognosis, recording diagnosis and recording prescriptions respectively. There will also be a submit button which the doctor will press in order to update the data into the database. The doctor will be able to view all records of the patient uploaded into the database.

## 5.1.2 Receptionist interface

After login, the receptionist will be able to capture data for the staff, student or community members using his/her interface. The interface will comprise of Fields requiring the personal details of the aforementioned people. These details comprise of national Identification number in the case of community members, staff identification number in the case of members of the staff and student Registration numbers in the case of students.

## 5.1.3 Physician interface

After login, the physician will be able to update data for patients using his/her interface. The interface will comprise of Fields requiring the specific patient vitals. These comprise of height, name, weight, age, blood pressure if tested, and temperature. There will be a submit button at the end so that the physician can finally update the patient’s records after their encounter.

## 5.1.4 Lab Technician interface

After login, the Lab Technician will be able to view the patients that are on the queue requiring tests. He/she will therefore issue the tests manually and later on record the test results using his/her interface. The interface will comprise of Fields requiring the him/her to capture the test results. There will be a submit button at the end so that the physician can finally update the patient’s records after their encounter.

## 5.1.5 Pharmacist interface

After Login, the pharmacist will be in a position to view the doctors’ prescriptions and administer the requested drugs. He/she will thereafter submit the data into the database to ensure track of those patients that have undergone a complete cycle of treatment.

## 5.2. Screen Images

## 5.3. Screen Objects and Actions

## 6. Help System Structure

## 7. REQUIREMENTS MATRIX

## 8. APPENDICES