

**LA GRANDEE INTERNATIONAL COLLEGE**

**Simalchaur, Pokhara, Nepal**

A Project Proposal/Report

on

**Patient Record System**

**Submitted to**

Ramesh Chalise

Bachelor of Computer Application (BCA) Program

In partial fulfillment of the requirements for the degree of BCA under

Pokhara University

**Submitted by**

Deven Shrestha

Sandesh Bastola

Nobel Baral

Susil Neupane

BCA, II

April 27, 2024

# Project Summary

In short, Our program “Patient Record System” will assist in easier management of Patient’s record in using a computer system. This program will allow the user to add new patient with their details which are their name, address, disease, age, sex, severity and prescribed medicine and the program will automatically assign the patient with an unique code which can later be used to easily identify a specific user easily within our program. You can also search for an already existing patient using their patient code or general information to either update their data, show their data or remove them from the “ongoing treatment” section and move them to “treated” or if they transferred to another hospital then move them to the “transferred” section. If a patient has already been in the hospital before and has arrived again, you can search within the “treated or transferred” to view their previous data and update the data and move him/her to “ongoing treatment” section and assign them a new patient code. If the user mistakenly typed wrong information to the program, they can even remove the patient’s data although the patient code will only be removed if the wrong data was the latest one and if the wrong data was the older one then the code will not be removed but the code will be associated with the empty data.

# Table of Contents

[Project Summary ii](#_Toc165205083)

[Table of Contents iii](#_Toc165205084)

[List of Tables iv](#_Toc165205085)

[List of Figures v](#_Toc165205086)

[List of Abbreviation vi](#_Toc165205087)

[1. Introduction 1](#_Toc165205088)

[2. Problem Statement 2](#_Toc165205089)

[3. Objective 3](#_Toc165205090)

[4. Methodology 4](#_Toc165205091)

[4.1 Algorithm 4](#_Toc165205092)

[4.2 Flowchart 5](#_Toc165205093)

# List of Tables

# List of Figures

# List of Abbreviation

PRS: Patient Record System

BCA: Bachelor of Computer Application

Repo: Repository

# Introduction

Patient Record System(PRS) is a system in which a user can manage and modify patient’s data stored in a computer. This system will allow the user to add, update, move and remove patient’s data with relative ease and convience. We are writing such program in C programming language which is a middle level language to ensure good performance of the program.

# Problem Statement

A lot of people come in a hospital to get checked and cure their diseases. Of course, the hospital can manage the patient’s data in a traditional system using paper but that way is more risky and time consuming. Papers are prone to damage through fire, water and other components and can easily be destroyed. In addition to that searching through different files just to update one patient’s information can be a time consuming process. Furthermore some files can be lost and cause massive problem to both the patient and the hospital. This program aims to solve such issue by handling the patient’s data through a computer.

# Objective

The objective of creating this program are as follows:

* Provide easy to use and manage interface to record and manage patient’s data
* Save time and resource of a hospital

# Methodology

We are going to use various programming techniques we will learn through the internet and in college to create the PRS system. Here are the current methodology we have planned that we will use to create the system, do note that the given methodology are subject to change as we get more experienced with the process:

* All members of our group will create a github account and commit all their contributions in a repo named Patient-Record-System-in-C
* There will be different folders for each members corresponding to our names in the reposititory, each members will commit only in their corresponding repository and after discussion with each member the approved code will be applied in the main program.
* The program will store previous patient’s data and current patient’s data separately.
* Each patient will have their own unique patient code.
* When the program runs, it will first ask if the patient has been in our hospital before and if he/she has then the program will ask the patient code of the patient while he/she were admitted in the hospital and search through the previous patient’s data and the user can update the data and the patient will be assigned a new code.
* If the patient has not been in our hospital then it will ask the user to enter patient’s name, address, age, sex, disease, severity and prescribed medicine.
* The user can search for current and previous patient records using the patient’s code to update, view or move them.

A program is easier to build when you have laid your plans in a form of algorithms and flowchart. Algorithms and flowcharts can avoid burn out from being overwhelmed by a project.

## Algorithm

* **Add patient to the database:**

Step 1: Start

Step 2: Ask if the patient has been to our hospital before

Step 3: If yes then ask for his/her code, search through the “cured or transferred” section, if the patient is not found then go to step 2, if the patient is found then ask for new age, disease, severity and prescribed medicine, if no then ask to enter patient’s name, age, address, disease, severity and prescribed medicine

Step 4: Assign the patient his/her patient code which will be the array number they happen to have their data stored on

Step 5: Ask if the user wants to add another patient

Step 6: If yes then go to Step 2, if no then go to Step 7

Step 7: End

* **Search/View/Update/Move/Remove patient’s data:**

Step 1: Start

Step 2: Ask to input patient’s code

Step 3: If patient’s code exist, then show the patient’s data, if this doesn’t then print “Wrong or Invalid Code” and go to Step 2

Step 4: Ask if the user wants to update, move or remove the patient’s data

Step 5: If the user chooses to update, then it will ask the user to type the patient’s information again, if the user chooses to move, it will ask whether the user wants to move the data to “cured” or “transferred”, if the user chooses “cured” then replace their name with <patient-name>(CURED) in the “ongoing treatment” section and transfer them to the “cured” section, if the user chooses transferred then replace their name with <patient-name>(TRANSFERRED) in the “ongoing treatment” section and transfer them to the “transferred” section, if the user chooses to remove, then if the user they want to remove is the latest one they can remove the patient’s data with it’s code altogether, if the user they want to remove is the last one then just replace the name with “REMOVED” and put all the value with integer/float numbers 0 and strings with “REMOVED”.

Step 6: End

## Flowchart

* **Add patient to the database**