Hospital Information System -

Project Summary and Documentation

Jeet Kumar

Chintan Patel

Caitlin Miller

**Project Summary**

The purpose of our project is to create an information system which can be used in hospitals to share patient's personal data with different departments of the hospital. It will have multiple modules with each module giving and receiving different pieces of data corresponding to their branch in the hospital. For our project we created a program that consists of three modules. The three modules are billing, personal information and staff. The purpose of the personal info module is to store, print and update data in text files assigned to each patient. The purpose of the billing module is to print hospital costs and print the amount the patient owes based on the amount of days they stayed at the hospital. The purpose of the staff module is to see which nurses, doctors, and other staff members are employed in the hospital and add any staff members.

**Team Member Contributions**

Jeet Kumar - Brainstorming, Coding of menu and functions, Meeting Coordination, Final Report

Chintan Patel - Brainstorming, Coding of class, array, and text file function

Caitlin Miller - Brainstorming, Coding of functions, Coordination of Members

**Original Four Cases**

This project can be used to share personal data between hospital employees about the patients like birth date, gender, weight, height, patient history, insurance info, etc. It can also be used to share test results for the patients between patients, doctors, and nurses. They can also store valuable information about each patient based on the conversations with the patient like patient’s allergies to certain medicines, etc. This project can also be used by the billing department to access the patient’s insurance information, medical records, payment information, etc. The billing department can use the medical records to verify insurance information, supply an insurance plan, etc. It can also be used by the hospital pharmacy to access the patient’s medical diagnosis to distribute accurate medications to the patient.

**Final Implementations**

The final implementations are a bit different from the original cases. The final project has three modules. The first module is the patient’s personal information. In this module the hospital staff member can create a new patient file with all their information, they can also view an existing patient files for any information they need, and finally they can update the patient files. The second module is the billing module. The billing module has two options. Total cost for the patient, and the breakdown of the hospital cost. The final module is the staff module. This module has the option of viewing existing staff member such as the doctors, nurses and other staff members. This module also has the option of adding new staff member.

**Foreseen Limitations**

There are some limitations in regards to the security of the project. Since we have not learned enough about Java, it will be difficult to implement a security system that will have restrictive access.  The program also cannot be used as a database for storage. Since we do not have enough knowledge about Java, we cannot store mass information in files or create files every time a user inputs data.

**Problems / Solutions**

We encountered several problems during the preliminary phases of our software. For our program we wanted to be able to call information from different patient’s files to each module. In order to do this, we first decided to use arrays in order to create storage for multiple pieces of data. However, this created complications because we could only store data for one patient and not multiple patients as we had originally planned. For this problem we choose to change the array into a class file which allowed us to have multiple patients.

**Testing**

After the coding was complete we tested our project by running the code in jGrasp. When the program ran it asks the user to select a module. We selected the the first module which asked the user to enter new patient information. After adding the new patient information we checked if the information was stored properly. We also tried to update the information option. It updated the file without overwriting the file. The second test that we ran was for the billing module. In this test we input how many days did the patient stay in the hospital. When the user inputs an integer value the billing class calculates and prints out the total hospital cost. The final test we ran was for the staff module. In this test we checked to see if lists and prints out different doctors, nurses, and other staff. The test also consisted of adding new staff members.

**Future Enhancements**

If we were able to make any changes to this software in the future, we would like to add more modules or modules that are more specific to the patients’ and the departments’ needs. In these modules we would have liked to implement more options that would give the user a greater understanding of how our program could help them. If we knew more about programming in Java, then we could implement more animated modules and options that could better cater to needs of specialized patients.

**Documentation**

For our software, we have a Main file. In our main file we have our main function that will branch out to our 3 main modules, a class section for the billing module, and text file creator for the personal information module. In order to run this software the user can use any java editor. In order for this program to run successfully the computer system must uses the latest java 8.

The user would open the main.java file in a java editor, and just simply run the program. If the program is being used for the first time the user will be asked to create a new patient file in this file the user will be asked to input all of the patient’s information that is relevant to the hospital staff. After the user has input the information the program will assign the patient a random patient number. This patient number will also be used in naming the patient file in the directory. After the user has created few patient files there will be an option to view any patient file the user wants. The program will list all the patient files, the user just has to input the patient number to open that patient’s file. The user will also have the option to add new information to the existing patient files. The user can also use this program for billing purpose. When the user runs the billing module they will be asked to input how many days the patient had stayed at the hospital. When the user enters a value the program calculates the total and prints it out for the user. This program can also be used to view the existing staff members such as doctors, nurses, and other staff members. The user can also add new staff member. When the user wants to add new staff member all they have to do is enter if the new member is a doctor, a nurse, or other staff member. When they enter that they will be asked to enter the new staff member’s name. After entering the name the program will print out the new list of staff members with the new member's name on it.   

In this program there is only one class called billing. The class consists of four variables medicine, room, food, and days. The default value for these variables is set for 1 day. So the default cost of a patient stay in the hospital is one day. The class has a constructor for the variable days. When the user run the billing module the user is asked to input how many days the patient was in the hospital. After the input has been entered the class takes that value and multiplies it with the default value to produce the total cost for the patient. The total cost is then printed out in the main program.     