University Transilvania
Faculty IESC
Study program IEC(english)

PROJECT 20 <HOSPITAL MANAGING SYSTEM >
<Andrei Darius-Constantin,1, 4LF521>
<30.05.2023 >

1. Specification defining the problem

1.1. Description of the theme

The theme of the project is to develop a medical managing system for a hospital. The system should allow for scheduling appointments, viewing appointments, registering patients, and viewing patient records. The objective is to create an efficient and user-friendly system that helps healthcare professionals manage appointments and patient information effectively.

1.2. Description of requirements

The project requirements include:

-Classes for Patient, Doctor, and Appointment.

-Functions for scheduling appointments, viewing appointments, registering patients, and viewing patient records.

-Data storage and retrieval mechanisms, such as file handling, to store and retrieve patient and appointment data.

-Error handling and validation to ensure the accuracy and integrity of the data.

1.3. Functional specification

The functional requirements of the program can be described as follows:

Input data:

Patient information (name, CNP, age, gender, address, phone number)

Doctor information (name, specialization)

Appointment details (patient name, patient CNP, doctor name, date, time)

Output data:

Appointment details (patient name, patient CNP, doctor name, date, time)

Patient records (name, CNP, age, gender, address, phone number)

Doctor records (name, specialization)

Functional diagram:

The system should include the following functionalities:

Patient class:

Member variables: name, CNP, age, gender, address, phone number

Constructor and getter functions

```
registerPatient() function:
    -Prompts for patient information
    -Creates a Patient object and saves the data to a file

viewPatientRecords() function:
    -Reads patient records from a file
    -Displays patient information

searchPatient() function:
    -Prompts for a patient name
    -Reads patient records from a file
    -Searches for matching patients based on the name
    -Displays patient details

Doctor class:
Member variables: name, specialization, patient information, program
Constructor and getter functions

viewDoctors() function:
    -Reads doctor records from a file
    -Displays doctor information


Appointment class:
Member variables: patient name, patient CNP, doctor name, date, time
Constructor and getter functions

scheduleAppointment() function:
    -Prompts for patient name, CNP, doctor name, date, and time
    -Checks if the doctor has reached the maximum number of appointments for the day
    -Saves the appointment details to a file

viewAppointment() function:
    -Prompts for a patient name
    -Reads appointment records from a file
    -Searches for matching appointments based on the patient name
    -Displays the appointment details
```

1.4. User Interface

The project only has one simple menu:

```
=== Hospital Management System ===
1. Register a new patient
2. Search for a patient
3. View patient records
4. Schedule an appointment
5. View appointments
6. Edit patient records
7. Delete patient records
8. Generate statistics
9. Change password
10. View doctors
11. Help
12. Exit
==================================
```

2. Project Development

2.1. General objectives

The general objectives of the project are to develop a functional medical managing system that simplifies appointment scheduling, patient registration, and record management. The system should improve the efficiency of healthcare professionals and enhance patient care.

2.2. Description of data processed by the program

The program will process various types of data, including patient information (name, CNP, age, gender, address, phone number), doctor information (name, specialization), and appointment details (patient name, patient CNP, doctor name, date, time). The data will be stored in appropriate data structures or files.

2.3. Description of program modules

The program can be organized into modules, including:

-Patient class module
-Doctor class module
-Appointment class module
-File handling module (to read and write data to files)
-User interface module (to handle user interactions and display information)

3. Conclusions

Based on the project development, the medical managing system has been implemented to address the specified requirements. It provides functionalities for scheduling appointments, viewing appointments, registering patients, and viewing patient records. The code follows an object-oriented approach, ensuring data encapsulation and modularity.

Proposals for future developments of the project:

-Enhance the user interface by implementing a graphical user interface (GUI) for a more intuitive and user-friendly experience.

-Implement data validation and error handling mechanisms to ensure the integrity and correctness of the entered data.

-Incorporate additional features such as appointment reminders, doctor availability management, and billing functionalities.

-Integrate the system with a database management system for efficient data storage and retrieval.

4. References

-C++ Tutorials on www.cplusplus.com

-“W SQUARE ACADEMY” (www.youtube.com) channel

-Stack Overflow (www.stackoverflow.com): A popular community-driven platform for programming-related questions and answers.