HOSPITAL MANAGEMENT SYSTEM

COURSE NAME : CSE 115.2

SUBMITTED TO :

Tanvir Yeasin Opy
Lab Instructor

SUBMITTED ON :

13 April 2025

PREPARED BY :

Md. Fahim Hossain	2512847642
Samia Zaman	2511026642
Sidratul Muntahara Audhira	2513009642
Aminur Rashid Sohag	2514160042

ABSTRACT

An extensive Hospital Management System is developed in this project to organize and digitalize the hospital operations. Using a role-based access control mechanism, it allows different stakeholders (doctors, patients, hospital administrators, and pharmacists, for instance) to interact with each other seamlessly.

With features such as patient data management, dispensary tracking, appointment booking, and secure login authentication, the system boosts operational efficiency and reduces manual efforts. The project which is implemented in C language shows how data structures and file handling can be employed to solve real-world healthcare management problems.

OBJECTIVES

- Develop a structured hospital management system
- Implement authentication for staff login
- Manage patient records, appointments, and prescriptions
- Enable dispensary stock tracking and updates
- Automate ID generation for patients and staff
- Store and retrieve medical records efficiently

Methodology

The Hospital Management System is developed using C programming, ensuring modularity and efficiency in handling hospital-related data. The development process follows a structured approach

Programming Languages and Associated Tools

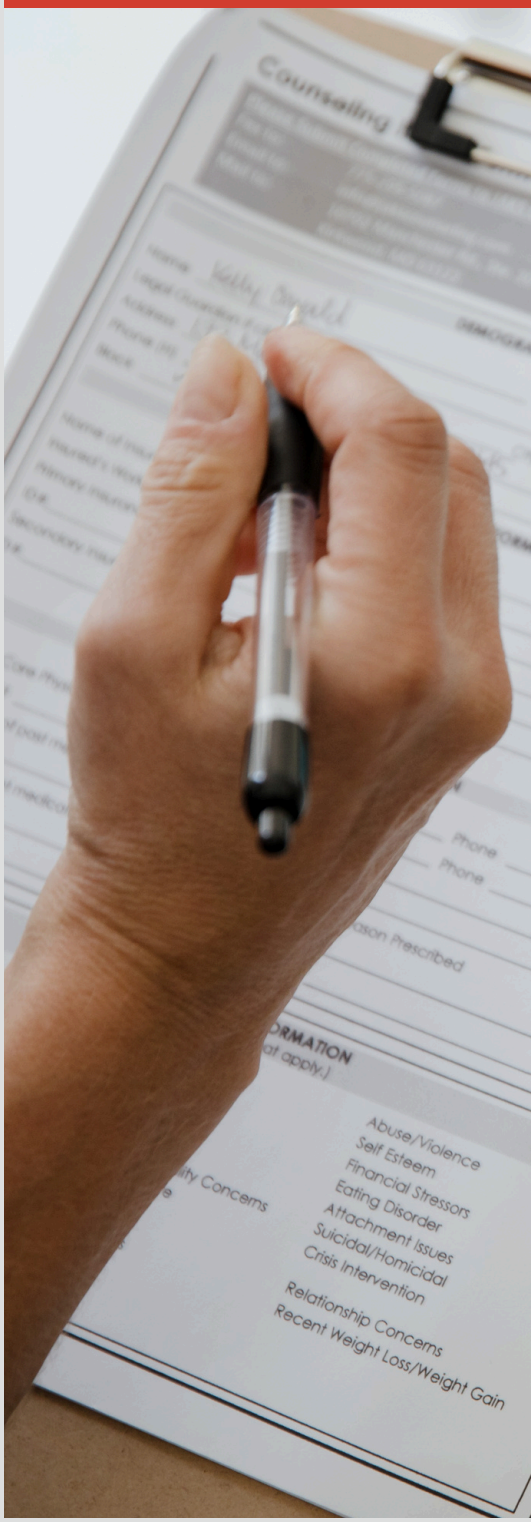
The implementation for the project is carried out in C using its standard libraries: stdio.h, stdlib.h, and string.h for I/O commands, file operations, and strings respectively. The patient records, appointments, and dispensary records are all stored using File handling for data persistence.

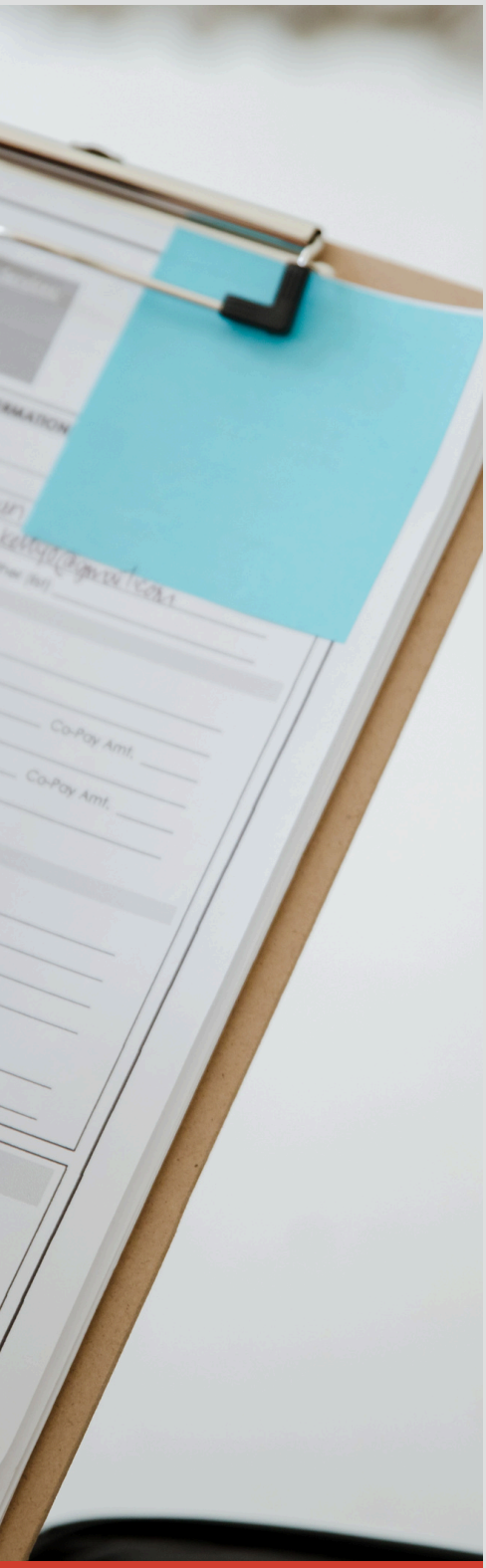
Role-Based Access Control

The system has an authentication feature to enable role-based access restriction for a doctor, patient, hospital admin, and pharmacist. All roles are allowed to have tailored access to different permissions on hospital data.

Data Structures & Storage

The patient records, medical history, and appointments are organized in a way in which data is efficiently maintained utilizing the struct.The system allows for data maintenance through retrieval and updates, storing data in a structured text file.





Methodology

Functionality Implementation

•Authentication System:
Users are required to log in using their credentials in order to access the system.

• Patient Management:Update, view or add patient records. This system is available for admins and doctors.

•Dispensary Management:Pharmacists have the ability to add and update current stock of medicines.

•Appointment Scheduling:Patients are able to book appointments with their preferred doctors and these appointments are based on availability.

Error Handling & Validation

Data validation measures are employed to mitigate the risk of wrong data entry. Unforeseen possibilities such as unintentional duplicate ID generation or invalid login attempts are dealt with utilizing error handling measures to ensure proper functionality.

Optimization & Automation

Without the need for manual input, the system ensures unique identifiers with no patient ID duplication through an automated generation system.The system ensures there’s no overbooking of appointments by checking with the doctors schedule. This enhances the appointment booking experience for patients.

Project Structure Overview

```
HOSPITAL-MANAGEMENT-SYSTEM/
│
├── .github/                                # GitHub configuration files
├── distributed-works/                      # Individual contributions from
                                            each team member
│   ├── aminur-rashid/
│   ├── audhira-muntaha/
│   ├── md-fahim-hossain/
│   └── samia-zaman/
│
├── main/                                   # Core files for the running system
│
│   ├── zen-care.c                          # ✅ Main file to be run by user
│   ├── appointments.csv                    # Appointment data
│   ├── doctors.csv                         # Doctors data
│   ├── patients.csv                        # Patient information
│   ├── record.csv                          # General patient records
│   ├── staffs.csv                          # Hospital staff data
│   ├── department-codes.csv                # Department code data
│   ├── dispensary.csv                      # Dispensary records
│   ├── medicine-characteristics.csv        # Medicine code data
│   └── zen-care.exe                        # Compiled main executable
│
├── LICENSE
└── README.md
```

REQUIREMENTS

GCC COMPILER (LINUX/WINDOWS)
VS CODE OR ANY C-SUPPORTED IDE
TERMINAL (COMMAND PROMPT / BASH)

HOW TO COMPILE AND RUN

```
cd hospital-management-system/main
gcc zen-care.c -o zen-care.exe

./zen-care.exe    # On Bash
zen-care.exe      # On Windows cmd
```

MEET THE TEAM

CONTRIBUTORS BEHIND THE MODULES

MD. FAHIM HOSSAIN

2512847642

SIDRATUL MUNTAHARA AUDHIRA

2513009642

AMINUR RASHID SOHAG

2514160042

SAMIA ZAMAN

2511026642

- Login System: Authenticates users and redirects based on role.
- Add/View/Search Patient: Doctors/Admins can add and retrieve patient data.
- Medical Records: Add/view/delete records associated with patients.
- Prescription Update: Doctors can update prescriptions.
- Dispensary System: Add, update, and view available medicines.
- Appointment Booking: Patients can book/view appointments.
- File Handling: Data stored persistently using structured files in .csv format.
- Automated ID Generation: Ensures unique IDs using increment-based automation.

DISPENSARY

A terminal-based C application for efficient management of medicine data within a hospital's dispensary. It supports unique ID generation for medicines, tracks medicine details and It enables real-time tracking, addition, and update of medicine inventory, improving dispensary operations.

🌟 BENEFITS

1. Scalable System: Structured ID format allows integration with future hospital modules.
2. Data Safety: Temporary file approach protects data during updates.
3. Clear Visualization: Organized display ensures transparency in inventory status.
4. Ready for Expansion: Structure allows easy future integration (e.g., billing, expiration tracking).
5. Cross-Team Compatibility: Simple CSV format supports collaborative development and debugging.

CONTRIBUTOR

MD. FAHIM HOSSAIN

2512847642

🔧CORE FEATURES

1. Add New Medicine Records
   - Captures complete details: name, dosage form, generic name, group, company, and quantity.
   - Automatically generates a unique Medicine ID based on department and subdivision codes.
   - Prevents duplicate entries by verifying existing records before addition.
2. Auto-ID Generation System
   - Encodes department and sub-division info.
   - Adds a unique suffix to ensure uniqueness.
   - Supports scalable and hierarchical categorization of medicines.
3. View Complete Medicine Inventory
   - Displays all medicine records in a structured tabular format.
   - Columns: ID, Name, Dosage Form, Generic Name, Group, Company, Quantity.
4. Update Existing Medicine Quantity
   - Search by Medicine ID.
   - Update through addition or subtraction of stock.
   - Prevents invalid subtraction (below zero).
5. View Medicine Codes
   - Dynamically reads from medicine-characteristics.csv.
   - Shows department and sub-division codes for reference.
   - Supports user-friendly selection during medicine entry.
6. File Handling & Data Persistence
   - Stores records in a persistent CSV file: dispensary.csv.
   - Uses temporary files (temp.csv) for safe data updates.
   - Efficient line-by-line file scanning for fast record access.
7. Interactive CLI Interface
   - Menu-driven system for ease of navigation.
   - User prompts and validation for smooth experience.

CONTRIBUTOR

MD. FAHIM HOSSAIN

2512847642

LOGIN LOGOUT SYSTEM

The system features a Login function that allows authorized users to securely access their accounts, while the Logout function ensures sessions are safely closed and redirected to the login screen. The Add Patient module collects and stores new patient information, including personal, contact, and medical details. With the View Patient function, users can retrieve comprehensive patient records instantly. The View Doctors feature displays doctor profiles, their specialties, and availability. Lastly, View Staffs offers access to non-doctor staff profiles, detailing their roles and assignments within the hospital.

BENEFITS

1. Secures system access with login/logout
2. Allows only authorized user entry
3. Speeds up patient registration
4. Prevents duplicate patient records
5. Provides quick access to patient info
6. Helps manage doctor schedules
7. Improves staff coordination
8. Enhances overall hospital efficiency

CONTRIBUTOR

SAMIA ZAMAN

2511026642

🔧CORE FEATURES

1. Log In
   - Authenticates user credentials.
   - Verifies username and password against saved user records.
   - Grants access based on role (e.g., admin, doctor, staff).
   - Displays appropriate dashboard upon successful login.
2. Log Out
   - Safely ends the current session.
   - Clears any temporary data or session variables.
   - Redirects to the login screen after logout.
3. Add Patients
   - Registers new patient details into the system.
   - Fields include name, age, gender, contact info, medical history, etc.
   - Automatically generates a unique Patient ID.
   - Validates input to prevent duplicate entries.
4. View Patients
   - Displays stored patient information.
   - Searchable by Patient ID or name.
   - Lists details such as medical history, prescriptions, visit records.
5. View Doctors
   - Shows doctor profiles with specialization and availability.
   - Lists name, department, contact info, and schedule.
   - Allows sorting or filtering by department or specialization.
6. View Staffs
   - Displays non-doctor medical and admin staff details.
   - Includes name, role (e.g., nurse, receptionist), shift, and contact info.
   - Helps in managing staff allocation and responsibilities.
7. File Handling & Data Persistence
   - Stores records in a persistent CSV file: dispensary.csv.
   - Uses temporary files (temp.csv) for safe data updates.

CONTRIBUTOR

SAMIA ZAMAN

2511026642

APPOINTMENT SYSTEM

The Appointment feature allows patients to schedule medical appointments efficiently within the system. It collects essential patient information such as ID, name, doctor details, department, and the appointment date. The system automatically generates a unique appointment ID based on the current date, ensuring no duplication. All data is saved in a persistent CSV file for future reference. This module enhances the hospital's appointment management process by maintaining structured records and enabling easy retrieval.

🌟 BENEFITS

1. Ensures accurate and fast appointment booking
2. Automatically generates unique appointment IDs
3. Prevents duplicate scheduling
4. Saves appointment records securely in a file
5. Reduces manual effort in hospital appointment management
6. Makes patient-doctor scheduling more organized
7. Helps staff access appointment data easily
8. Enhances overall service quality and efficiency in the hospital

CONTRIBUTOR

SIDRATUL MUNTAHARA AUDHIRA

2513009642

🔧CORE FEATURES

1. Appointment ID Generation
    - Automatically generates a unique appointment ID based on the current date (Year, Month, Day).
    - Prevents duplication of appointment entries.
2. Patient Information Entry
    - Collects user input for:
    - Patient ID
    - Patient Name
    - Doctor Name
    - Doctor's Department
    - Appointment Date
    - Ensures proper formatting and sequence of data entry.
3. File Handling
    - Opens appointments.csv in append mode to store appointment records.
    - Writes all appointment details to the file.
    - Closes the file securely after writing to ensure data safety.
4. Data Validation
    - Performs basic validation, such as checking if the file opens successfully.
    - Displays the generated Appointment ID for user confirmation.
5. User Interaction
    - Uses terminal-based prompts (scanf, printf) to interact with the user.
    - Shows a success message along with the scheduled appointment date after storing information.
6. Structured Record Keeping
    - Stores all data in a structured format for easy access and future retrieval.
    - Assists hospital administration in maintaining a clean and organized appointment system.

CONTRIBUTOR

SIDRATUL MUNTAHARA AUDHIRA

2513009642

DISPENSARY

It generates unique IDs for each patient, records detailed diagnosis and prescriptions, and securely stores data in structured files. The system ensures privacy, quick access, and improved coordination between patients and doctors.

🌟 BENEFITS

1. Unique Identification: Patients are assigned unique IDs for accurate and easy access to records.
2. Data Privacy: Only authorized personnel can access sensitive patient data.
3. Quick Access: Doctors and patients can quickly retrieve and review medical records.
4. Minimized Paperwork: Reduces manual reporting and physical file handling.
5. Efficient Management: Improves the speed and reliability of patient data management.
6. Supports Informed Decisions: Helps doctors stay updated on the patient's medical condition and history.
7. Future-Ready Format: Structured data storage allows smooth expansion into modules like lab reports, billing, or appointment tracking.

CONTRIBUTOR

AMINUR RASHID SOHAG

2514160042

🔧CORE FEATURES

1. Patient Record ID Generation
   a. Automatically generates a unique Patient ID using current date and time.
   b. Ensures no duplicate entries for the same patient record.
   c. ID structure supports future scalability (e.g., timestamp + department code).
2. Comprehensive Patient Information Entry
   a. Collects detailed patient data:
      i. Patient Name
      ii. Doctor's Name
      iii. Doctor's Department
      iv. Diagnosis
      v. Prescription
   b. Ensures consistency and structure in medical recordkeeping.
3. Efficient File Handling
   a. Opens record.csv in append mode to add new medical records.
   b. Writes patient reports to the file safely.
   c. Closes the file securely after writing to avoid data loss or corruption.
4. Data Validation & Feedback
   a. Validates successful file operations (e.g., file open status).
   b. Displays generated Patient ID and Appointment ID to the user.
   c. Confirms successful data entry with terminal-based feedback messages.
5. Interactive CLI Interface
   a. Uses a terminal-based prompt system for real-time data input.
   b. Provides clear success or error messages to guide users.
   c. Enhances accessibility for medical staff working in a text-based environment.
6. Structured Record Keeping
   a. Stores all patient data in a well-organized CSV format for easy retrieval.
   b. Enables doctors to maintain and track medical histories efficiently.
   c. Supports future implementation of search and analytics features.

CONTRIBUTOR

AMINUR RASHID SOHAG

2514160042

Approach, Problems faced and Solutions

01 Used Libraries

The project uses standard C libraries for input/output operations: stdio.h and stdlib.h for memory management and file handling.

02 Data Storing

Data Files for the patients' records, prescriptions, and appointments are stored using file handling in C. Ensuring data integrity was challenging in this case but adequate measures through validation and error handling above were implemented.

03 Login

The system manages to login staff with the correct provided credentials specific to their role. One of the difficulties encountered was credential storage. This was solved by using encrypted storage.

04 Structure Description

Structs contain all the required fields like patient ID, patient name, age, medical history, and appointment details as well as the relevant data for patients, and their medical records alongside the scheduled appointments.

05 Dispensary Management

This portion of the system enables pharmacists to add, update, and view available stock of medicines. One of the more complicated issues was ensuring the consistency of the data, which was solved using structured file handling techniques.

06 ID Generation using Automation

Patient and staff are issued with one of a kind ID's in an incremental manner. The main issue was ensuring the ID was unique without duplication. This challenge was overcome by keeping a registry of the ID being utilized

07 Book Appointment

Patients have the opportunity to book available doctors for their appointments. Conflicts in scheduling were dealt with by either avoiding duplication in overlapping appointments or automated pre-emptive checking.

Challenges & Solutions:

CHALLENGE: MANAGING LARGE DATA THROUGH FILE I/O.

SOLUTION: USED PROPER STRUCTURING AND FILE MODES FOR CONSISTENT ACCESS.

CHALLENGE: ENSURING UNIQUE PATIENT/MEDICINE IDS.

SOLUTION: IMPLEMENTED AUTO-ID GENERATION MECHANISM.

CHALLENGE: ROLE-BASED FUNCTION ACCESS.

SOLUTION: DEVELOPED A PERMISSION-BASED DASHBOARD AFTER LOGIN

CHALLENGE: INITIAL CODE ORGANIZATION.

SOLUTION: DIVIDED FUNCTIONS BASED ON ROLE AND GROUPED THEM LOGICALLY.

CONCLUSION

THE CONSTRUCTION OF THE HOSPITAL MANAGEMENT SYSTEM IS A DEFINITE STEP TOWARDS PROPELLING THE HEALTHCARE SERVICES INTO NEW TECHNOLOGICAL HEIGHTS.

INCORPORATION OF KEY FEATURES LIKE AUTHENTICATION, PATIENT MANAGEMENT, DISPENSARY PROCESSES, AND APPOINTMENT SCHEDULING GUARANTEES THAT THIS SYSTEM IS EFFECTIVE AND MEETS THE DEMANDS OF THE REAL WORLD.

OUR TEAM PROVIDED UNIQUE SOLUTIONS TO A NUMBER OF TECHNICAL CHALLENGES RELATED TO DATA ORGANIZATION, ACCESS CONTROL, PROCESSES AUTOMATION, AND ERROR HANDLING, ALL INTEGRATING NEW PRACTICAL WAYS INTO THEIR DESIGN.

THE EFFORTS FUELED BY THE C PROGRAMMING LANGUAGE TAUGHT US A LOT, BUT SO DID THE DESIGNING OF THE SYSTEM WITH ITS OPERATIONAL PROCESSES INTERTWINED WITH HEALTHCARE.

IF FURTHER ADVANCEMENT IS DONE, LIKE MOVING TO A DATABASE CONTROLLED BACK END AND ADDING A GRAPHICAL USER INTERFACE, THIS PROJECT WOULD TRANSFORM INTO COMPREHENSIVE HOSPITAL SOFTWARE SYSTEM.

THIS PROJECT IS A CLEAR DEMONSTRATION OF THE TEAMWORK AND DEDICATION OF THE STUDENTS AND SHOWS THE DEVELOPERS' READINESS TO TACKLE THE CHALLENGES THE WORLD POSES.