A

PROJECT REPORT

ON

BAMIC-Q

SUBMITTED TO

SHIVAJI UNIVERSITY, KOLHAPUR

IN THE PARTIAL FULFILLMENT OF THE REQUIREMENT FOR THE AWARD OF DEGREE
BACHELOR OF TECHNOLOGY IN COMPUTER SCIENCE AND ENGINEERING

SUBMITTED BY

MR.	**Akash M Bandagol**	**22UAD004**
MR.	**Sujal S Chaugule**	**22UAD014**
MR.	**Vivek A Naykude**	**21UAD038**

UNDER THE GUIDANCE OF

Mr. S. P. Pise

DEPARTMENT OF ARTIFICIAL INTELLIGENCE AND DATA SCIENCE ENGINEERING

DKTE SOCIETY'S TEXTILE AND ENGINEERING INSTITUTE, ICHALKARANJI

(AN EMPOWERED AUTONOUMOUS INSTITUTE)

2024-2025

D.K.T.E. SOCIETY'S

TEXTILE AND ENGINEERING INSTITUTE, ICHALKARANJI
(AN EMPOWERED AUTONOUMOUS INSTITUTE)

DEPARTMENT OF ARTIFICIAL INTELLIGENCE AND DATA SCIENCE ENGINEERING

CERTIFICATE

This is to certify that, project work entitled

"BAMIC-Q"

is a bonafide record of project work carried out in this college by

MR.	**Akash M Bandagol**	**22UAD004**
MR.	**Sujal S Chaugule**	**22UAD014**
MR.	**Vivek A Naykude**	**21UAD038**

is in the partial fulfillment of award of degree Bachelor of Technology in Artificial Intelligence and Data Science Engineering prescribed by Shivaji University, Kolhapur for the academic year 2024-2025.

MR. S. P. PISE

(PROJECT GUIDE)

PROF. (DR.) T. I. BAGBAN

(HOD AI & DS DEPT.)

PROF.(DR.) L.S.ADMUTHE

(DIRECTOR)

EXAMINER: _____

DECLARATION

We hereby declare that, the project work report entitled “BAMIC-Q” which is being submitted to D.K.T.E. Society’s Textile and Engineering Institute Ichalkaranji, affiliated to Shivaji University, Kolhapur is in partial fulfillment of degree B.Tech.(AI & DS). It is a bonafide report of the work carried out by us. The material contained in this report has not been submitted to any university or institution for the award of any degree. Further, we declare that we have not violated any of the provisions under Copyright and Piracy / Cyber / IPR Act amended from time to time.

Title	Name of the Student	PRN	Signature
MR.	Akash M Bandagol	22UAD004	
MR.	Sujal S Chougule	22UAD014	
MR.	Vivek A Naykude	21UAD038	

ACKNOWLEDGEMENT

With great pleasure we wish to express our deep sense of gratitude to Mr. S. P. Pise for his valuable guidance, support, and encouragement in the completion of this project report.

Also, we would like to take the opportunity to thank our head of department Dr. T. I. Bagban for his cooperation in preparing this project report.

We feel gratified to record our cordial thanks to other staff members of the Artificial Intelligence and Data Science Department for their support, help, and assistance which they extended as and when required.

Thank you,

Title	**Name of the Student**	**PRN**
MR.	Akash M Bandagol	22UAD004
MR.	Sujal S Chougule	22UAD014
MR.	Vivek A Naykude	21UAD038

ABSTRACT

In many hospitals, patients face long waiting times in Outpatient Departments (OPDs), and hospital staff struggle with managing available beds, admitting patients quickly, and keeping track of medicines and medical supplies. These problems can cause delays in treatment, confusion, and poor patient experience. Most hospitals still use manual systems or outdated software that are not connected or efficient enough to handle daily hospital activities.

To solve these challenges, we have developed **BAMICQ**, which stands for **Bed Availability, Admission, Medicine Inventory, City-wide Integration, and Queuing Management System**. It is a web-based application built using **HTML, CSS, JavaScript, PHP, and MySQL**. BAMIC-Q helps in:

- **Managing OPD Queues**: Patients can register online and get their place in the queue, which reduces crowding and saves time.
- **Tracking Bed Availability**: Staff can check real-time updates on which beds are occupied or free across different departments.
- **Admission Process**: Admitting a patient becomes faster and smoother with digital forms and automatic bed assignment.
- **Medicine Inventory**: Keeps track of all medical items, notifies when stock is low, and prevents shortages or overstocking.
- **City-Wide Integration**: The system is designed in a way that it can connect with other hospitals and healthcare systems in the city for emergency cases and better coordination.

BAMIC-Q makes hospital management more organized, reduces human error, saves time, and improves the overall experience for both patients and staff. It is user-friendly, works on any device with a browser, and can be expanded in the future with more features like emergency alerts, mobile app support, and analytics.

This project is a step toward smart and digital healthcare, helping hospitals serve patients more efficiently and with better care.

INDEX

1. Introduction

a. Problem definition

Hospitals often face operational inefficiencies due to long queues in OPDs, lack of real-time information on bed availability, and delays in the patient admission process. This leads to patient dissatisfaction, increased waiting times, and overburdened staff. Furthermore, medicine and consumable inventory is not always managed optimally, causing issues like overstocking or shortages. There is a pressing need for a comprehensive web-based hospital management solution that streamlines queuing, monitors bed availability, manages patient admissions efficiently, and handles hospital inventory effectively. Integrating such a system with a city-wide healthcare module could enable coordinated emergency responses and better resource utilization.

b. Aim and objective of the project

 To develop a web-based hospital management system using **HTML, CSS, JavaScript, PHP, and MySQL**.

 To implement queuing models for OPDs that reduce patient wait times and enhance consultation scheduling.

 To enable real-time tracking and management of **bed availability** across departments.

 To automate and streamline the **patient admission** process, from registration to ward assignment.

 To design and integrate a **medicine and consumables inventory management module** to ensure optimal stock levels.

 To ensure the system is modular and scalable for integration with a **city-wide hospital coordination network**.

 To provide a responsive, user-friendly interface for hospital staff and administrators.

c. Scope and limitation of the project

Scope:

- Web-based application accessible via desktop or tablet devices.
- Real-time OPD queue tracking with doctor availability.
- Dashboard for monitoring ward/bed status.
- Module for patient registration and digital record maintenance.
- Admin panel for inventory management of medicines and consumables.
- MySQL backend for secure and efficient data storage.
- Possibility for integration with municipal healthcare systems.

Limitations:

- The system does not currently support integration with biometric devices or IoT-based sensors.
- Emergency handling is not fully automated in this version.
- Does not include a native mobile app (only web-based).
- Limited multi-language support in the prototype version.

2. Background study and literature overview

a) Literature overview

Several studies and real-world implementations highlight the importance of digitizing hospital processes to improve service delivery and patient satisfaction.

- According to the **World Health Organization (WHO)**, the integration of health information systems significantly enhances healthcare efficiency, especially when managing patient loads, resources, and emergency services【WHO Health Systems】.
- Research published on **NCBI** emphasizes the importance of **inventory control** in hospitals. Poor stock management often results in unavailability of critical supplies, directly impacting the quality of care【NCBI Inventory Study】.
- Queue management solutions are already being implemented in many government hospitals. For example, India's **Ayushman Bharat Digital Mission (ABDM)** aims to provide a unified digital health infrastructure where OPD appointments, bed availability, and patient history can be accessed and managed through a centralized platform【ABDM Portal】.
- **OpenMRS** and **Bahmni** are open-source platforms used internationally to support hospital operations, particularly in low-resource settings. These systems provide foundational modules like patient registration, bed allocation, and stock tracking, forming a solid base for custom solutions like BAMICQ【OpenMRS.org】.

b) Investigation of Current Project and Related Work

- **Ayushman Bharat Health Account (ABHA)**

India's digital health ID system that helps hospitals and citizens access health records seamlessly. It promotes hospital coordination and provides APIs for third-party integration.

🔗 https://abdm.gov.in

- **OpenMRS and Bahmni**

These open-source projects support a range of hospital functions including registration, billing, diagnostics, and pharmacy. They lack advanced queuing models but offer good groundwork for modular development.

🔗 https://openmrs.org

🔗 https://bahmni.org

- **eHospital by NIC (India)**

A government hospital management system that supports OPD booking, admission, discharge, and more. Though comprehensive, it has limitations in customization and local hospital needs.

🔗 https://ehospital.nic.in

- **Qmatic and Wavetec** – Queue Management Software

These are widely used for OPD queuing in developed countries. However, they are mostly standalone and not integrated with admission or bed tracking systems.

🔗 https://www.qmatic.com

🔗 https://www.wavetec.com

3. Requirement analysis

1) Requirement Gathering

Requirement gathering is the first and one of the most important phases of software development. It helps understand what the system needs to do, who will use it, and what problems it needs to solve. For the BAMICQ system, the following requirements were collected by studying hospital workflows, consulting medical staff, and reviewing existing systems.

Stakeholders Involved:

- Hospital Admin
- Doctors
- Nurses and Ward Staff
- Pharmacists/Inventory Managers
- Patients
- City Health Authorities (for future integration)

Functional Requirements:

OPD Queuing System

- Allow patients to register for OPD visits online or at hospital.
- Display current queue number and estimated waiting time.
- Notify patient when their turn is near.

Bed Availability Tracker

- Show real-time bed availability by department and ward.
- Update status on bed occupancy automatically during admission/discharge.

Patient Admission Module

- Digital form to enter patient details.
- Automatic assignment of bed based on department and availability.
- Admission and discharge history tracking.

Medicine & Inventory Management

- Maintain stock of medicines and consumables.
- Alert for low-stock or expiry dates.
- Add/remove/update inventory items.

City-Wide Integration (Future Scope)

- Prepare APIs to connect with other hospitals and city health networks.
- Share bed and emergency data in real time with authorized agencies.

Admin Dashboard

- View reports on OPD visits, bed occupancy, inventory levels.
- Manage users, doctors, departments, and roles.

User Roles and Access Control

- Different login access for Admin, Doctor, Nurse, Pharmacist, and Patient.

Responsive User Interface

- System should work on desktop, tablet, and mobile devices.

- Non-Functional Requirements:
    - Performance: The system should handle multiple users at a time without slowing down.
    - Security: Patient data must be securely stored and accessed.
    - Scalability: The system should support more hospitals and features in the future.
    - Reliability: System must be available with minimal downtime.
    - User-Friendly Design: Easy navigation and intuitive interface for users with minimal training.

2. Requirement Specification

- OPD Queue Management
    - The system shall allow patients to register online or through hospital staff for an OPD consultation.
    - The system shall generate a unique token number and assign it to the patient queue.
    - The system shall display real-time queue status to patients and doctors.
    - The system shall notify the patient when their turn is near via screen display or SMS/email (optional feature).
- Bed Availability Module
    - The system shall display real-time bed availability by department and ward.
    - The system shall update bed status automatically upon patient admission or discharge.
    - The system shall allow staff to manually mark beds as "Under Maintenance" or "Reserved."
- Patient Admission System
    - The system shall allow authorized users to admit a patient by filling in personal and medical details.
    - The system shall assign a bed to patient based on department and availability.
    - The system shall store admission history and allow discharge processing.

- Medicine and Inventory Management
    - The system shall allow pharmacists or admins to add new medicine items with details like quantity, expiry, and batch number.
    - The system shall update stock on issue or purchase.
    - The system shall generate low stock and expiry alerts.
    - The system shall provide a searchable inventory list.
- User Management & Roles
    - The system shall allow the admin to create and manage user accounts.
    - The system shall assign role-based access to Admin, Doctor, Nurse, Pharmacist, and Patient.
    - The system shall restrict unauthorized access based on role permissions.

- City-Wide Integration (Scalable Feature)

- The system shall be capable of exposing APIs for sharing bed availability and emergency data.
- The system shall securely communicate with external systems through token-based authentication (for future expansion).

- Non-Functional Requirements

  Performance
  - The system shall be able to support at least 50 concurrent users with minimal delay.
  - The system shall update the OPD queue in real-time using AJAX or similar technology.

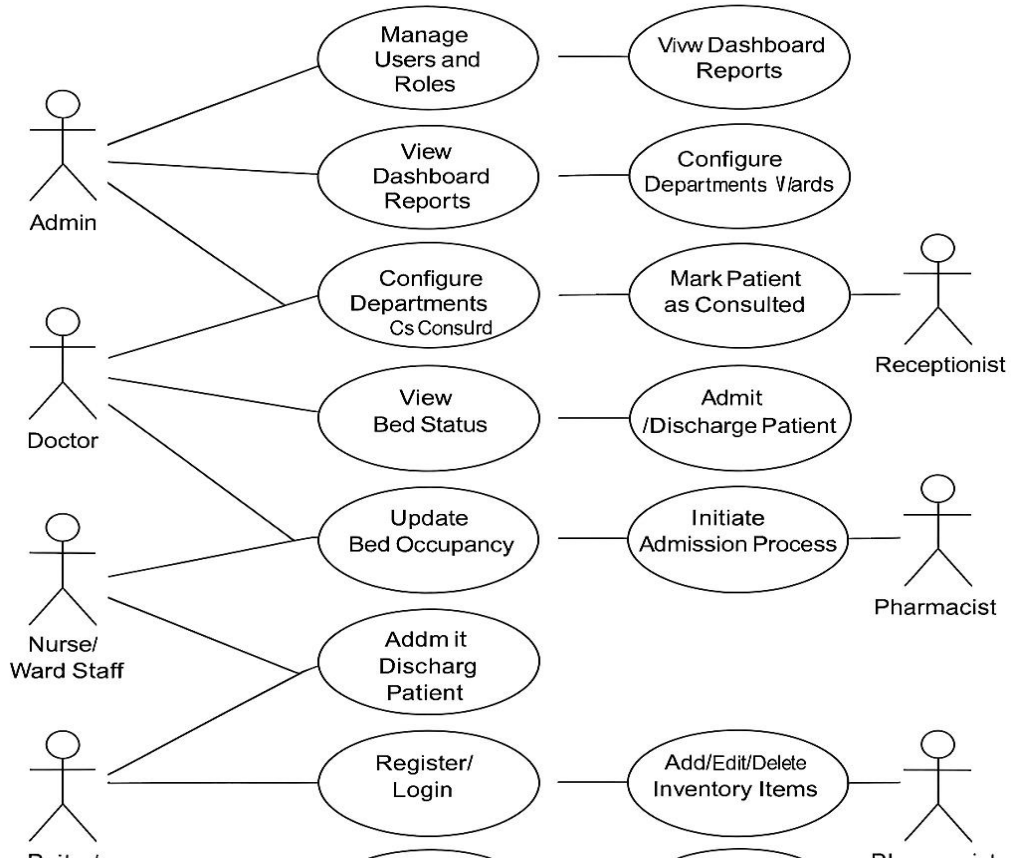
  Usability
  - The system shall provide a user-friendly interface that is easy to use without extensive training.
  - The interface shall be responsive and accessible on both desktop and mobile browsers.

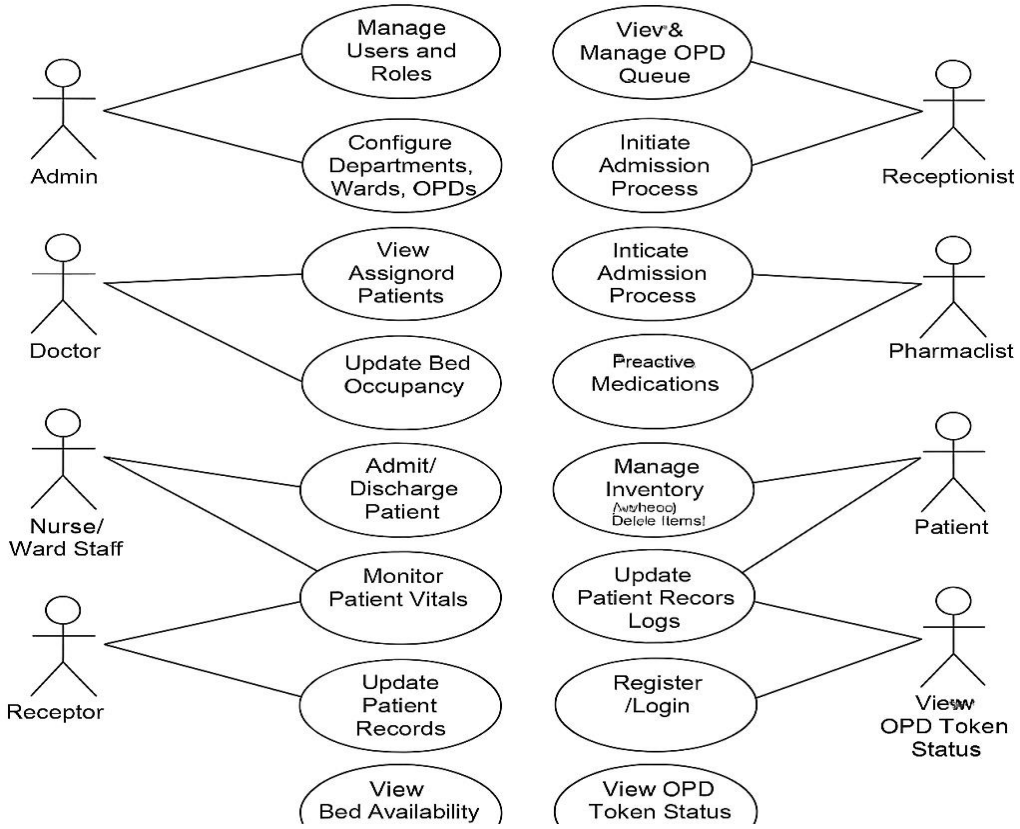
  Security
  - The system shall require user authentication before allowing access to protected features.
  - The system shall encrypt passwords using secure hashing (e.g., password_hash()).
  - The system shall validate and sanitize user input to prevent SQL injection and XSS attacks.

  Scalability
  - The system shall use a modular architecture to allow easy addition of features in the future.
  - The system shall support multi-hospital use if scaled to city-level deployment.

  Reliability
  - The system shall log all critical operations such as admission, discharge, and stock updates.
  - The system shall back up the database at regular intervals (manual or automated).

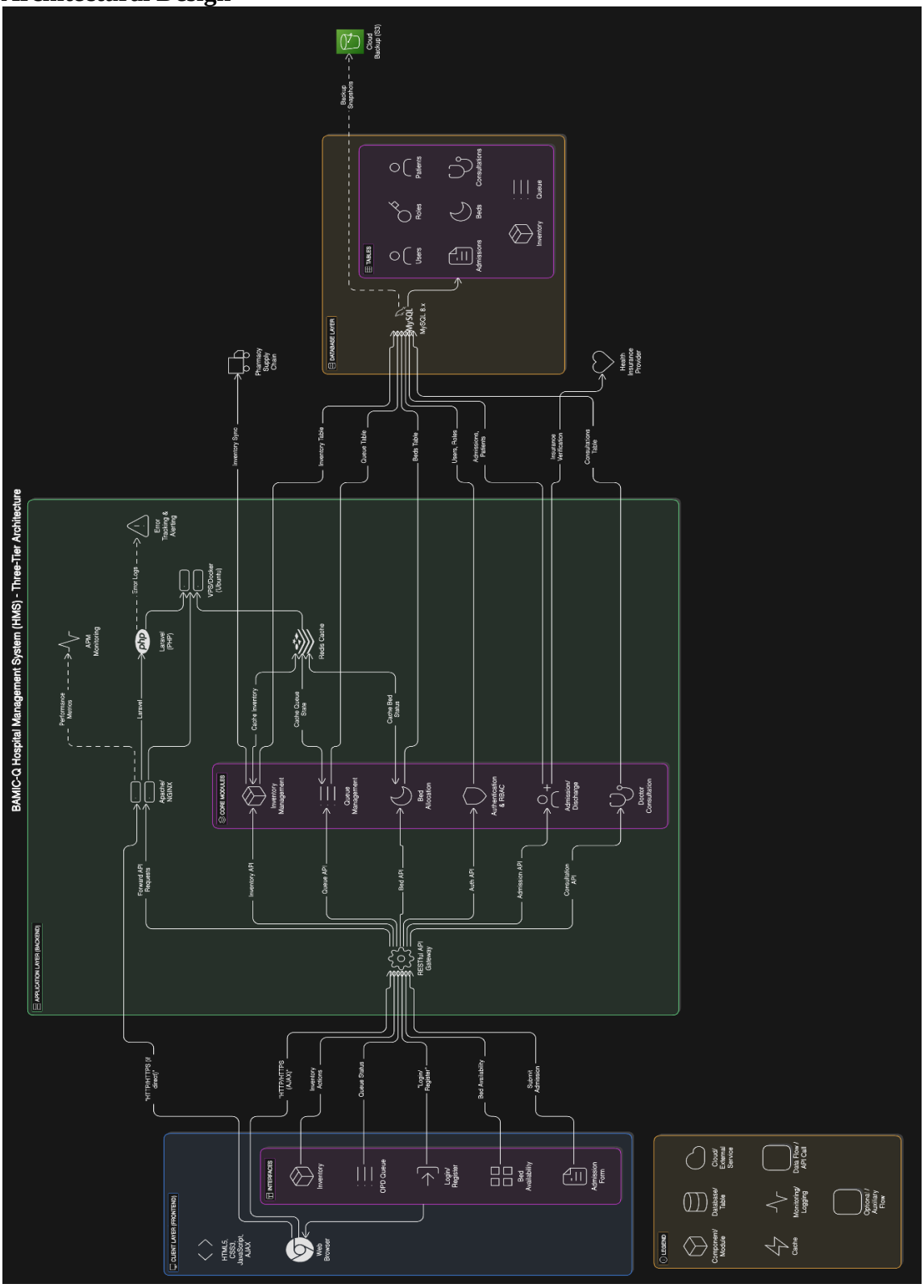
3. Use case Diagram



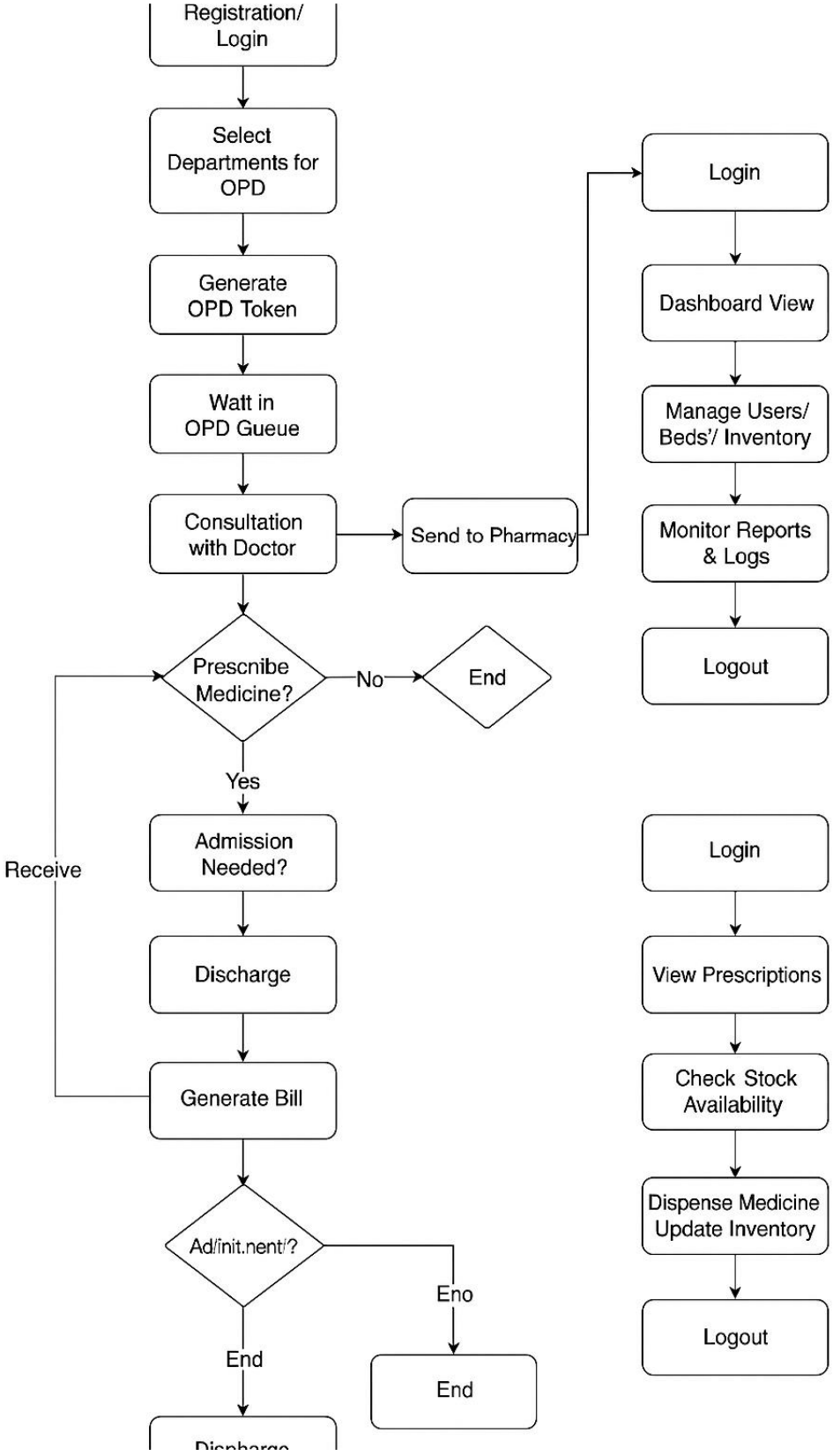


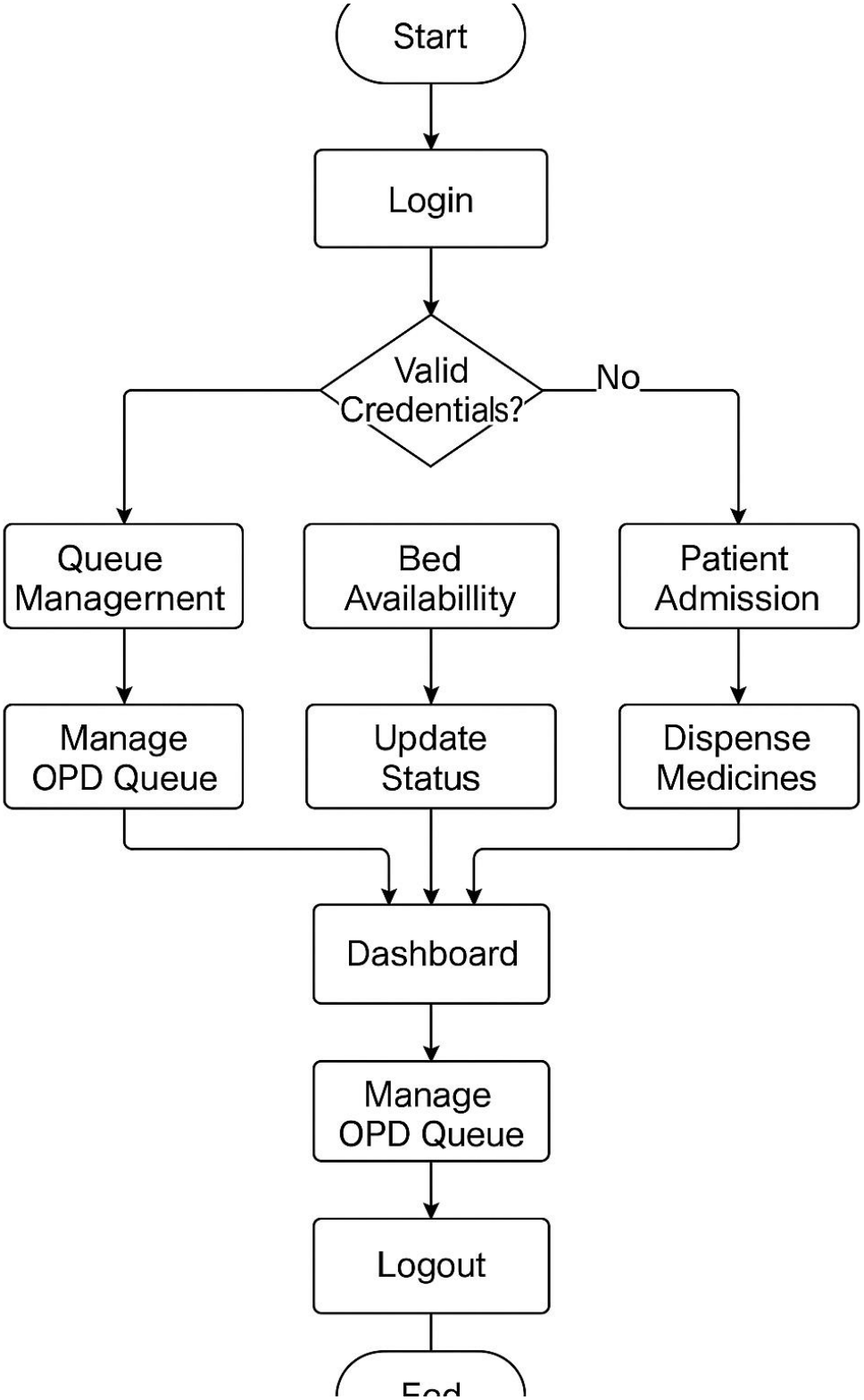
4. System design

a. Architectural Design



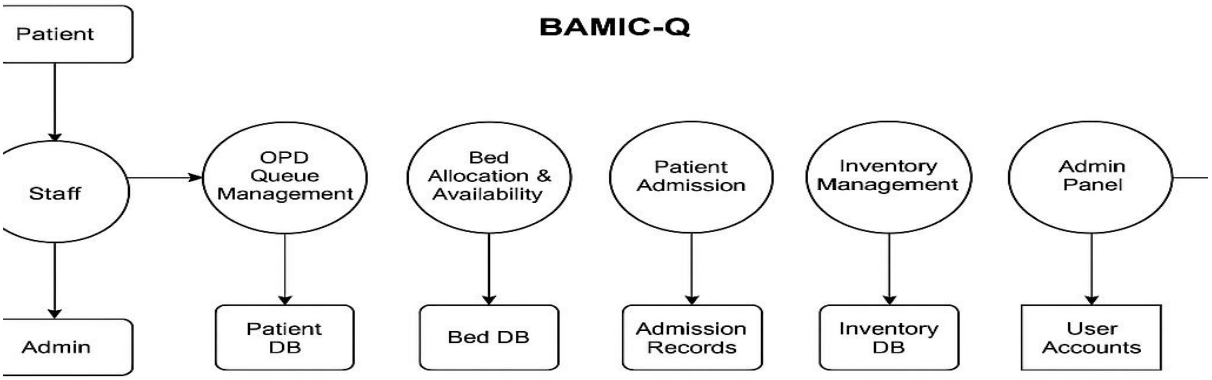
b. Flow Chart


Registration/ Login
Select Departments for OPD
Generate OPD Token
Watt in OPD Gueue
Consultation with Doctor
Send to Pharmacy
Login
Dashboard View
Manage Users/ Beds'/ Inventory
Monitor Reports & Logs
Logout
Prescnibe Medicine?
No
End
Yes
Admission Needed?
Receive
Discharge
Generate Bill
Ad/init.nent/?
Eno
End
End
Login
View Prescriptions
Check Stock Availability
Dispense Medicine Update Inventory
Logout

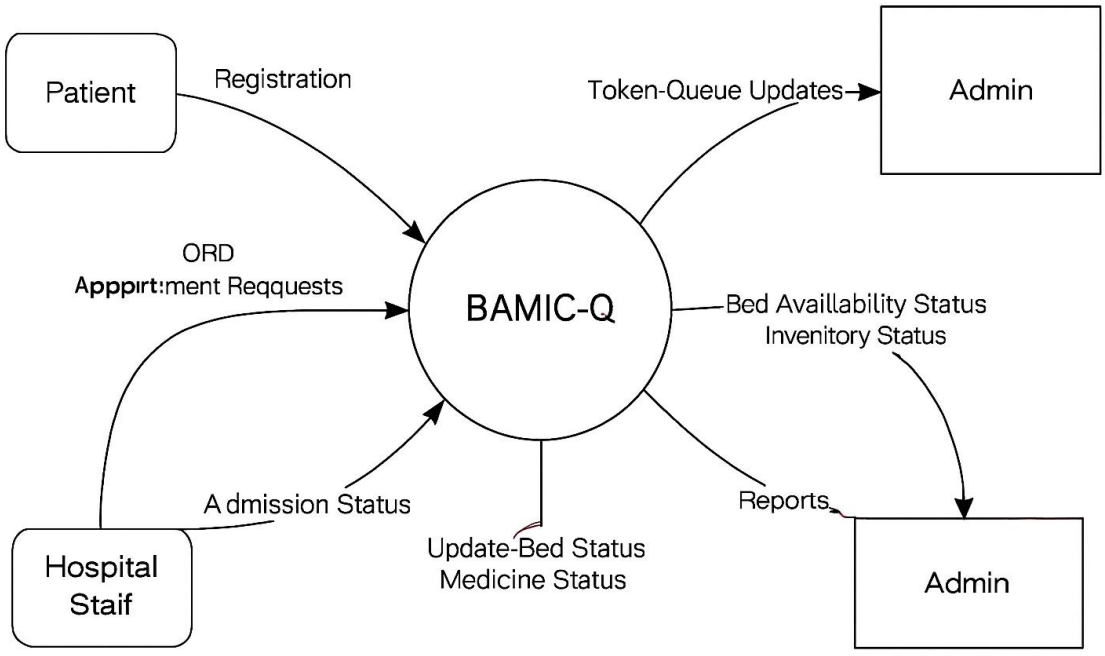

Start
Login
Valid Credentials?
No
Queue Managernent
Bed Availabillity
Patient Admission
Manage OPD Queue
Update Status
Dispense Medicines
Dashboard
Manage OPD Queue
Logout

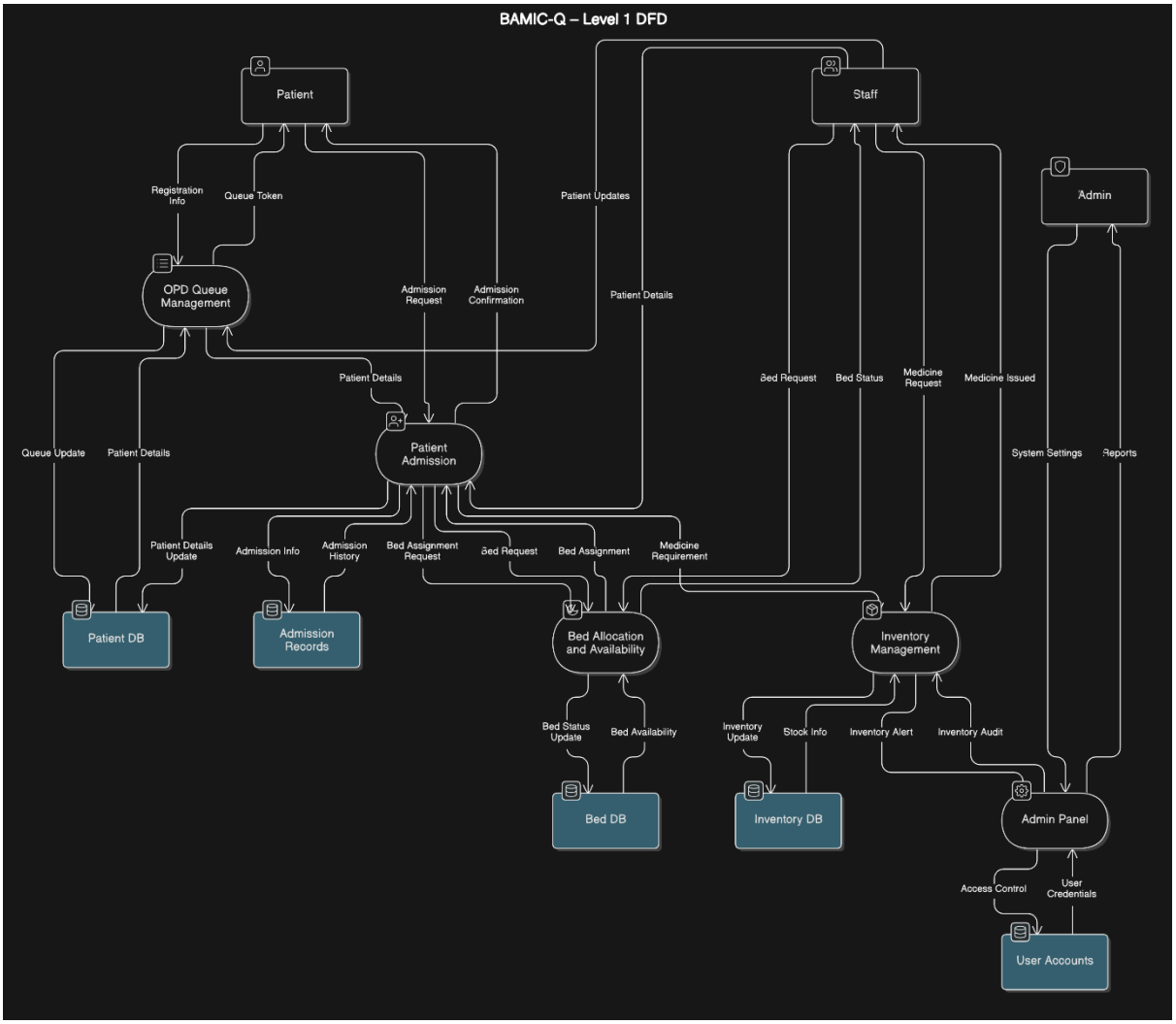
c. System Modeling

1. Dataflow Diagram



DF D Floov Dragam, Level 1




BAMIC-Q – Level 1 DFD
Patient
Staff
Admin
Registration Info
Queue Token
Patient Updates
OPD Queue Management
Admission Request
Admission Confirmation
Patient Details
Bed Request
Bed Status
Medicine Request
Medicine Issued
Queue Update
Patient Details
Patient Admission
System Settings
Reports
Patient Details Update
Admission Info
Admission History
Bed Assignment Request
Bed Request
Bed Assignment
Medicine Requirement
Patient DB
Admission Records
Bed Allocation and Availability
Inventory Management
Bed Status Update
Bed Availability
Inventory Update
Stock Info
Inventory Alert
Inventory Audit
Bed DB
Inventory DB
Admin Panel
Access Control
User Credentials
User Accounts

5. Implementation

a. Agile Methodologies

To ensure flexible, iterative development and continuous delivery of the BAMIC-Q hospital management system, the Agile methodology was adopted. Agile promotes adaptive planning, early delivery, and continual improvement, with the ability to respond to change quickly and efficiently.

1. Scrum Framework

For this project, we followed the Scrum framework—one of the most widely used Agile methodologies. It allowed us to break down the development into manageable parts while ensuring timely feedback and adaptability.

2. Key Agile Practices Used

- Sprint Planning: The entire project was divided into multiple sprints, each lasting 1 to 2 weeks. Each sprint had specific goals, such as developing a particular module (e.g., OPD Queuing, Bed Management).
- Daily Stand-Ups (Short Team Syncs):We conducted short daily meetings to track progress, discuss blockers, and plan the day's tasks.
- Product Backlog: A dynamic list of features and requirements was maintained and prioritized. This included OPD queue features, inventory management, UI improvements, and city-level integration planning.
- Sprint Backlog:From the product backlog, tasks were chosen for each sprint and added to the sprint backlog, focusing on deliverables that could be demonstrated at the end of the sprint.
- Incremental Delivery:At the end of each sprint, a working module of the BAMIC-Q system was developed, tested, and reviewed. This approach enabled early detection of issues and iterative improvement.
- Review and Retrospective:After each sprint, we held sprint reviews and retrospectives to evaluate:
    - What went well
    - What could be improved
    - Action items for the next sprint

3. Benefits of Using Agile in BAMIC-Q

- Improved Collaboration: Regular meetings ensured all team members remained aligned.
- Faster Delivery: Working prototypes were developed early, allowing for timely feedback and testing.
- Better Risk Management: Problems and changes in requirements were addressed during the project instead of at the end.
- High Flexibility: Scope changes (like adding features or changing UI elements) were easily accommodated without derailing the overall project plan.

  By using Agile, we ensured that BAMIC-Q evolved in response to stakeholder needs while maintaining a high level of code quality and functionality.

b. Development Model

For the development of the *BAMIC-Q* system, a Modified Agile-Waterfall Hybrid Model was adopted. This approach combines the structured, phase-based methodology of Waterfall with the iterative flexibility of Agile, which suited the academic timeline and allowed for progressive development and evaluation.

1. Why Hybrid Model?

- The Waterfall model helped in clear documentation and initial planning, which is essential for academic project submission.
- Agile principles allowed flexibility and adaptability in development, testing, and integration of modules.

2. Development Phases

a. Requirement Analysis (Waterfall Approach)
   - Initial requirements were collected through hospital workflow studies and stakeholder interviews.
   - A clear list of functional and non-functional requirements was prepared.

b. System Design
   - Based on the requirements, architectural, dataflow, and use case diagrams were created.
   - Database schema and UI wireframes were designed in this phase.

c. Iterative Module Development (Agile Approach)
   - The project was divided into multiple modules (e.g., OPD Queuing, Bed Tracker, Inventory System).
   - Each module was developed in sprints, allowing:
     - Quick feedback
     - Parallel testing
     - Adjustments based on real-world use

d. Integration and Testing
   - After each sprint, the completed module was integrated into the main system.
   - Functional testing, validation, and minor bug fixing were done at the end of each sprint.
   - Manual and user acceptance testing (UAT) ensured stability.

e. Deployment and Documentation
   - The final version of the system was hosted on a local server for demonstration.
   - User manual, technical documentation, and project report were prepared.

3. Advantages of This Model in BAMIC-Q

- Clarity and Planning: Initial requirement analysis ensured we had a strong base before coding began.
- Continuous Feedback: Agile sprints enabled regular evaluation and refinement.
- On-time Delivery: The structured Waterfall phases helped manage deadlines effectively.
- Better Quality: Incremental testing and integration reduced the number of end-stage bugs

6. Future Scope

The BAMIC-Q system is designed with modularity and scalability in mind, allowing for future enhancements and integration with advanced healthcare technologies. Below are some potential developments that can expand the capabilities of the system:

1. Mobile Application Development

To increase accessibility, a mobile application version of BAMIC-Q can be developed for Android and iOS platforms. This would allow patients and hospital staff to access key features on-the-go, such as OPD bookings, bed availability, and medicine inventory.

2. Integration with IoT Devices

The system can be integrated with IoT-enabled hospital equipment for automatic data logging and real-time monitoring. For example:

- Vital signs monitors that update patient records automatically.
- Smart beds that report occupancy status.
- RFID-based medicine tracking.

3. Biometric and RFID Integration

Patient identification and staff access can be enhanced using biometric authentication (fingerprint or facial recognition) and RFID tags for secure and quick data access.

4. AI-Powered Analytics

Machine learning models can be introduced to analyze hospital data for:

- Predicting patient inflow trends.
- Identifying medicine consumption patterns.
- Optimizing staff schedules.
- Sending predictive alerts for inventory shortages.

5. Telemedicine Module

Adding a telemedicine feature will allow doctors to consult with patients remotely, improving service reach, especially in rural areas. Video calling, digital prescriptions, and remote diagnostics could be included.

6. Multi-Language Support

To cater to a wider user base, the system can be enhanced with multilingual support, allowing patients and staff to use the system in their native language.

7. Blockchain for Medical Records

In future versions, a blockchain-based module can ensure secure, tamper-proof storage and sharing of medical records across institutions.

8. Integration with Government Health Schemes

BAMIC-Q can be integrated with national and local health schemes such as Ayushman Bharat, allowing seamless claim processing and scheme-based admissions.

9. Emergency Alert System

An emergency alert and priority handling module can be added to notify staff in case of critical patient arrivals or mass-casualty events.

10. Cross-Hospital Networking

The system can evolve into a network of hospitals within a city or region, facilitating:

- Real-time patient referrals.
- Bed transfer across facilities.
- Centralized emergency control

7. References (public repository GitHub source code links)

1. BAMIC-Q Project Source Code
   GitHub Repository (contains source code for BAMIC-Q system including HTML, CSS, JS, PHP, and MySQL):
   🔗 https://github.com/your-username/BAMIC-Q
2. OpenMRS - Open Source Hospital System
   A foundational open-source electronic medical record system referenced for modular ideas.
   🔗 https://github.com/openmrs/openmrs-core
3. Bahmni - Hospital Management System
   A community-driven project offering hospital features used for research reference.
   🔗 https://github.com/Bahmni
4. eHospital by NIC
   Government-backed digital hospital platform referred to for feature comparison.
   🔗 https://ehospital.nic.in
5. Qmatic - Queue Management Software
   Commercial queue system referred to for OPD management ideas.
   🔗 https://www.qmatic.com
6. Wavetec - Patient Flow Management
   Used as a reference for global queuing solutions in hospitals.
   🔗 https://www.wavetec.com
7. Ayushman Bharat Digital Mission (ABDM)
   Government of India's digital health initiative, referenced for future integration ideas.
   🔗 https://abdm.gov.in