Canterbury Institute of Management (CIM)

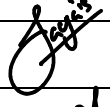
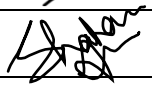
ASSESSMENT COVER SHEET

1. Personal Details			
Student ID	**Given Name(s)**	**Surname**	**Email Address**
1. CIM12137	Ayesh	Jayasekara	cim12137@ciom.edu.au
2. CIM11489		Muskan	cim11489@ciom.edu.au
3. CIM10081	Ahmer	Haseeb	cim10081@ciom.edu.au
4. CIM10988	Shahaan Ali	Khan	cim10988@ciom.edu.au
Campus	Darwin		
Course Title and Code	MSIT401 System Development Methodologies		
Assessment Title	AT3 Group Report and Presentation		
Due Date & Time	Week 11		
Course Lecturer/Tutor Name: Sharad Neupane		Assessment Word Count (if applicable): 3524	

2. Student Declaration

By signing and submitting this coversheet, I/we declare that:

- ✓ This assessment submission is my/our own work unless otherwise acknowledged (including the use of generative AI tools) and is in accordance with the Institute's ***Academic Integrity and Honesty Policy*** available on the website.
- ✓ No part of this assessment has been submitted previously for advanced standing or academic credit in this or any other course.
- ✓ I/we certify that we have not given a copy or have shown a copy of this assessment item to another student enrolled in the course, other than members of this group.
- ✓ I/we are aware that the Lecturer/Tutor of this assessment may, for the purpose of assessing this assessment task communicate a copy of this assessment task to a plagiarism checking service to detect possible breaches of academic integrity, for example, plagiarism, recycling, cheating, contract cheating, or unauthorised use of generative AI (which may then retain a copy of the item on its database for the purpose of future investigation).

Signature:		Date:	13/12/24
Signature:		Date:	13/12/24
Signature:		Date:	13/12/24
Signature:		Date:	13/12/24

Canterbury Institute of Management
63 Oxford Street, DARLINGHURST NSW 2010 | 21 Regent Street, REDFERN NSW 2016 | 2 Printers Place, DARWIN CITY NT 0800 | 399 Lonsdale Street, MELBOURNE VIC 3000
Phone: +61 2 8007 5846 | Website: https://ciom.edu.au/
TEQSA Provider ID: PRV14258 | CRICOS Provider Code: 03809A

MSIT401 System Development Methodologies
Group Case Study Analysis - *Report*

Executive Summary

IMPLEMENTING a *Patient Management System* for a hospital marks a transition point for any health care service provider organization. All stakeholders will gain benefits out of a strategically implemented system in terms of improving efficiency, transparency & ease of business.

This report evaluates a business case presented by one of the transforming healthcare institutions as a part of initial feasibility analysis. The Report includes technical aspects of various functional, non-functional requirements & technical aspects of such requirement implementation.

As a part of evaluation, identification of system use-cases, the proposed database schema, system architecture & some key wireframes for user-interfaces were also documented. While justifying the technical choices made, it also discusses how the new system can be implemented in Agile Development methodology in great detail.

Contents

Executive Summary **1**

Introduction **2**

Use Case Diagram & Descriptions **3**
Usecase Descriptions 6
Usecase 1: Patient Registration 6
Usecase 2: Appointment Scheduling 6
Usecase 3: Billing Management 7

Database Design **8**
Entity Relationship Diagram 8
Normalization 10

System Architecture **13**
Proposed Deployment Architecture 14
Choice of Diagramming: C4 Model 14
Deployment Architecture - Overview 16

System Interfaces Wireframes **17**
Patient Registration Interface 17
Appointment Scheduling Interface 19
Patient Portal Interface 20
Healthcare provider dashboard Interface 21
Billing Interface 22

Reporting **23**

System Development Strategy **25**
Development Strategy: Agile Methodology 25
Agile Development Process 25
Agile Roles and Responsibilities 26
Agile Tools and Techniques 26
Benefits of Agile Methodology 27
Problems and strategies to alleviate the consequences 27

Conclusion **28**

Bibliography **31**

Introduction

The proposed system of Patient Management System is a collective effort to digitalize information and assist healthcare professionals to focus on their duties rather than managing data in conventional method. This also integrates data from existing legacy Electronic Health Record System proving minute-to-date recent data from the external system further centralizing data for ease of access.

The integration of Document Management Module is proposed to ensure the patient's data is made available over the internet few clicks away. Enabling digital payments and automated invoicing capabilities are also considered for the implementation that benefits back office users as well as ever expanding clientele.

In this context of business, keeping close communication with patients is critical for both patients and the organization. Therefore, integration with existing email service also considered as a part of architectural design of the system.

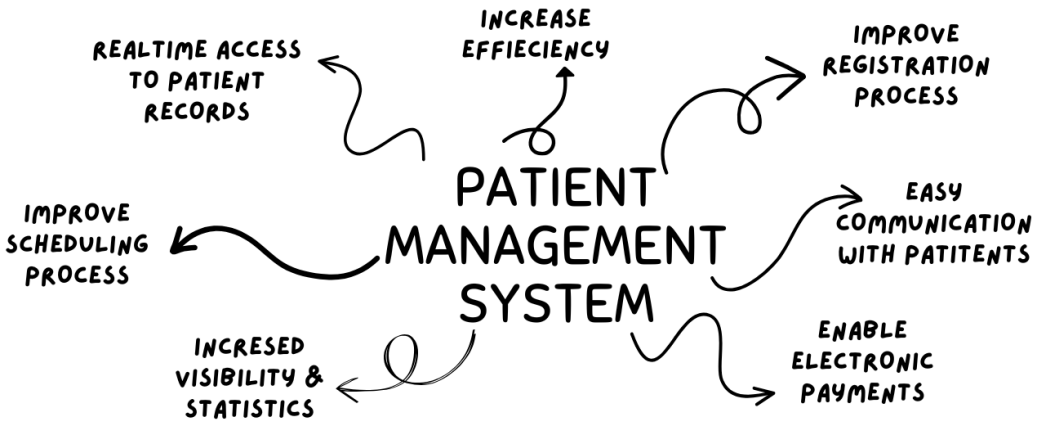


Figure 1: Benefits of Patient Management System

Use Case Diagram & Descriptions

In the context of the Patient Management System (PMS) for the health care organisation, the use case diagram depicts a clear image of how different operations operate in the healthcare organisation. The diagram also represents how different actors interact with the system and their respective roles in the system.

Overview of the Use Case Diagram: The diagram shows how the PMS addresses the core objectives like streamlining patient registration, scheduling appointments, tracking the treatments, managing the bills and access to all the medical records. It also shows the interaction of the key actors such as patients, administrative staff, billing staff, and healthcare providers.

Components of Use Case Diagram:

- **Actors:** The external entities which will interact with the system. The primary actors in this PMS are,
  - **Patient:** uses the system for registering, scheduling appointments, and managing bills.
  - **Administrative staff:** uses the system for appointment scheduling, and document uploads.
  - **Billing staff:** uses the system to handle financial operations, which includes bill generation, and payment processing.
  - **Healthcare provider:** access the system to update patient records, and schedule follow-ups.

- **Use-Cases:** These are the specific actions that are performed by the system in response to the interactions with actors. The uses cases that are used are,
  - **Patient:** Patient registration: this enables to register the patient on the portal and update the details in a secure database, and generate a unique patient id to login
  - **Appointment scheduling:** it facilitates the bookings or rescheduling appointments with availability checks.
  - **Billing management:** Manages all the billing process and payment processes.
  - **Treatment tracking:** Allows healthcare providers to update treatment plans and track patient progress.
  - **Patient notifications:** Allows to send notifications to the patients regarding appointment reminders and follow -ups through e-mail or SMS.

Using Use-Case Diagram; ensures all stakeholders have a clear understanding of the system's functionalities and user interactions. The below use case diagram demonstrates the interactions between the PMS and its actors, with arrows representing relationships. In this, Patients interacts directly with the PMS to register, and schedule appointments, and pay their bills. Administrative staff is managing the back-end process, like registering the patient, uploading documents and confirming the appointments. While the healthcare providers focus on updating treatment records and tracking the treatment process, likewise billing staff manages all the transactions.

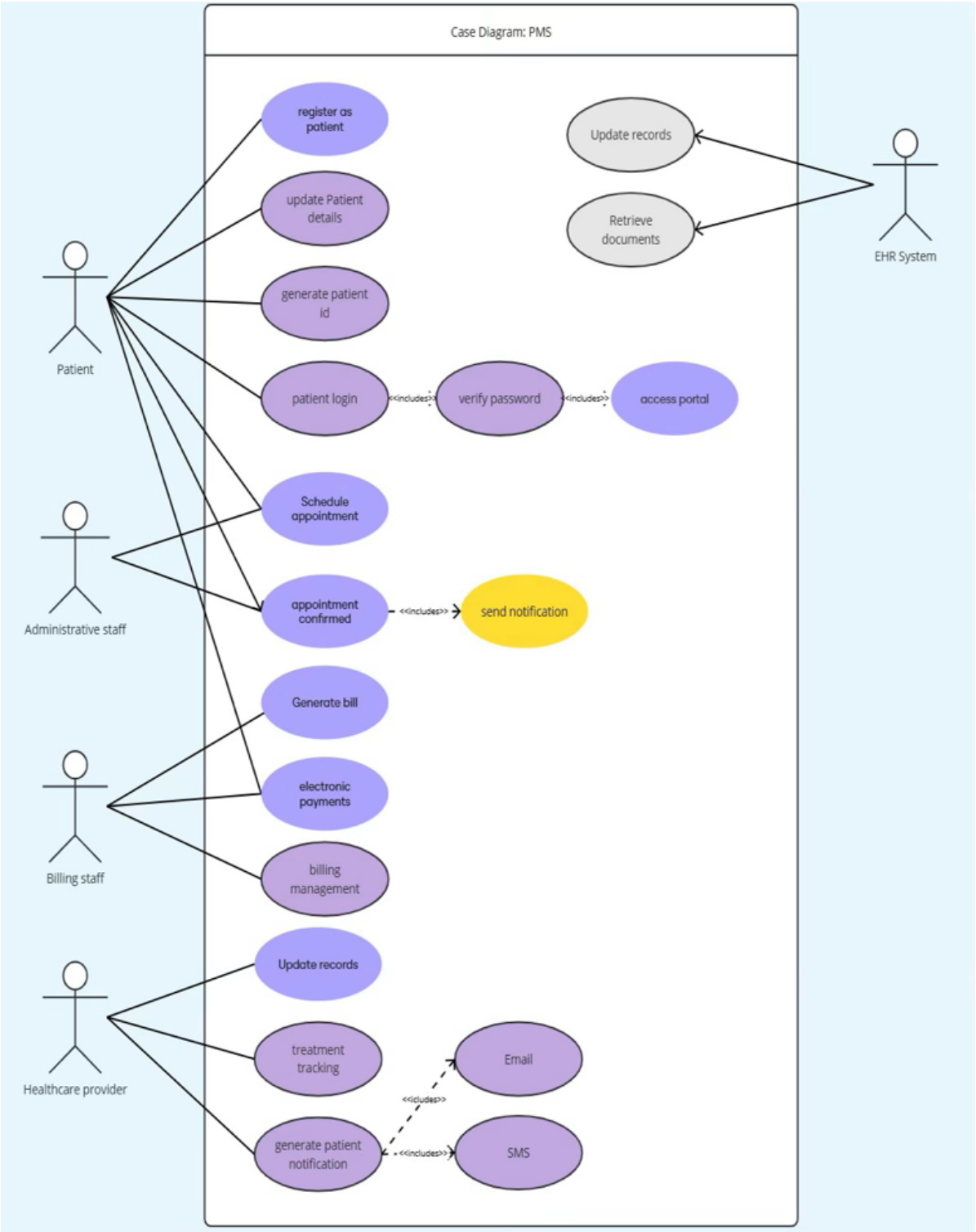


Figure 2: Usecase diagram for the proposed system

Usecase Descriptions

The case description usually describes the required functionality of the use case; also gives the general description of what usually happens, adding a brief description of any minor variations. The use case description is more detailed and (Britton & Doake, 2005; 'Use case diagram for hospital management system', 2024) structured.

Usecase 1: Patient Registration

USE CASE ELEMENT	DETAILS
Use case name	Patient registration
Actors	Patient, administrative staff
Description	Patients enter their details and medical history information to register in the portal.
Preconditions	None
Flow of events	1. Administrative staff input the patient's details 2. The system validates the data which is entered. 3. Confirmation of registration and creation of a unique patient id.
Postconditions	Patient's data is stored and is accessible in all facilities.

Figure 3: Usecase 1: Patient Registration

Usecase 2: Appointment Scheduling

USE CASE ELEMENT	DETAILS
Use case name	Appointment Scheduling
Actors	Patient, administrative staff
Description	Patients schedule appointments online and confirmed by administrative staff
Preconditions	Patient must be registered on the system
Flow of events	1. Patient selects preferred time and date for the appointment. 2. System checks the doctor's availability. 3. Once checked, the appointment is confirmed. 4. Confirmation notification sent to the patient.
Postconditions	Appointment details are updated in the patient's record.

Figure 4: Usecase 2: Appointment Scheduling

Usecase 3: Billing Management

USE CASE ELEMENT	DETAILS
Use case name	Billing management
Actors	Patient, Billing staff
Description	Manages bills and payment for services
Preconditions	Patient record should exist
Flow of events	1. Billing staff generated bill 2. Patient receives the bill and review it 3. Payment is processed electronically 4. Payment confirmation is sent to the patient
Postconditions	Billing status is updated in patient's record

Figure 5: Usecase 3: Billing Management

Database Design

Entity Relationship Diagram

It is important to get the database design of any system up to the expected standard. Software engineers make critical decisions when developing new systems based on data types, relationships & expected volume of data. Most of these design decisions are made referring to Entity Relationship Diagram *(ERD)* made available to them.

If important attributes are mis-identified, rectifying such issues later in the development lifecycle will impose a hefty reengineering cost both in terms of effort and time.

> **An entity is a person, place, thing, or event of informational interest. Attributes are objects that provide descriptive information about entities. Attributes may be unique identifiers or nonunique descriptors. Relationships describe the connectivity between entity instances: one-to-one, one-to-many, or many-to-many.**
>
> (Lightstone et al., 2010)

✍ Similar to Unified Modeling Language, there are multiple notations that are accepted in the industry. Crow's Foot Notation & Chen Notation are popular choices among database engineers. However, in this report Chen Notation was used.

☞ ERD diagrams can be complex diagrams depending on use-case. Therefore, the diagrams are usually read from top-to-bottom & left-to-right fashion. See Figure 6.

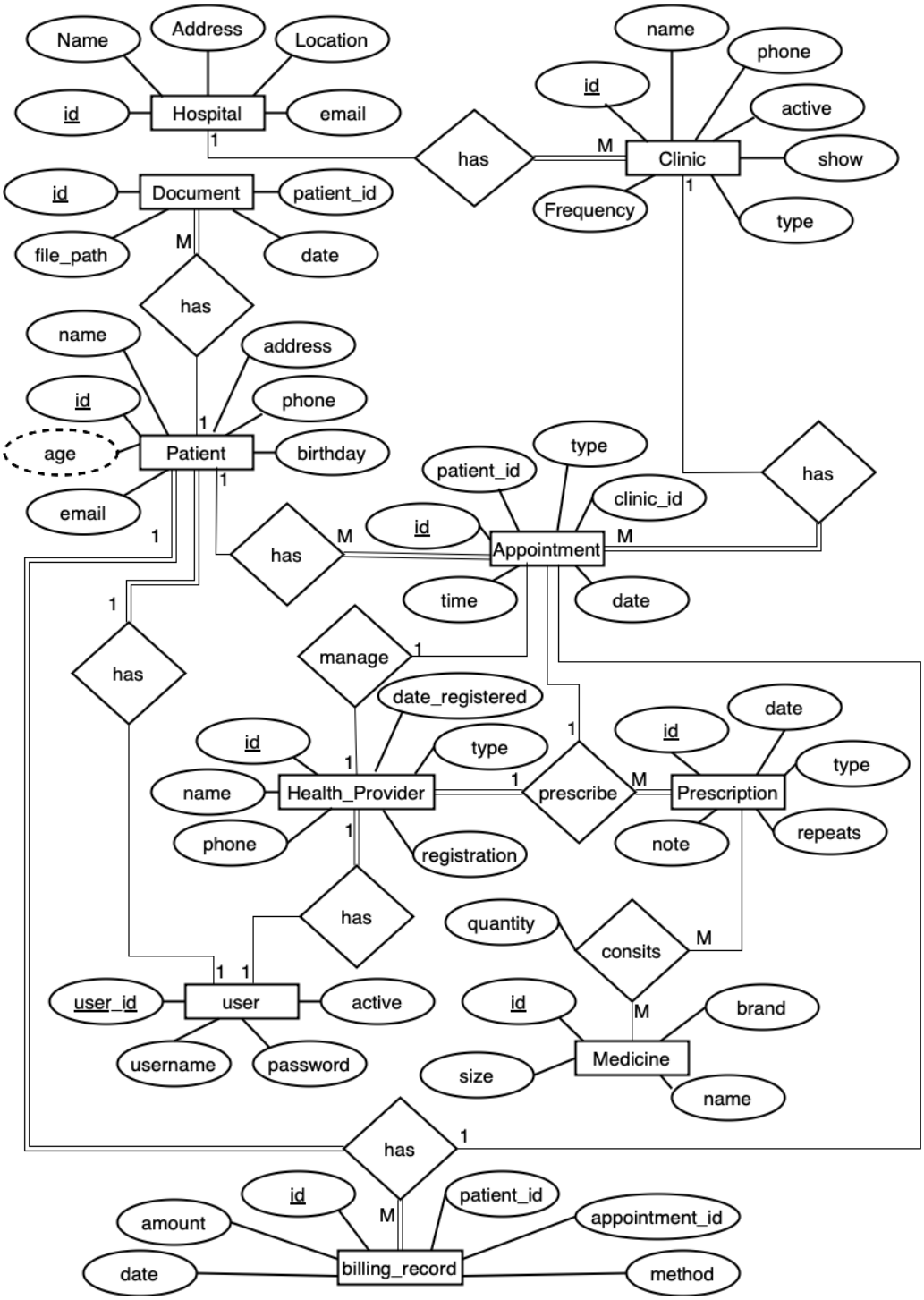


Figure 6: System Entity Relationship Diagram

Normalization

Relational databases concept emphasizes on simplifying attribute composition and enforce referential integrity in its core. This ultimately helps the system to maintain ACID properties *(Atomicity, Consistency, Isolation, Durability)* thus making the datastore efficient and reliable.

The process of achieving this optimal data structure is referred to as normalization. There are multiple levels of normalization in practice, but the first three levels are the most frequently needed in general use-cases.

The three levels of normalization can be summarized as,

- **Level 1 Normalization** - Ensures that attributes are atomic in formation and cannot be simplified any further.
- **Level 2 Normalization** - Ensures there are no non-key dependencies among attributes in a single entity.
- **Level 3 Normalization** - Ensures there are no transitive dependencies among attributes and all non-key attributes depend only on the primary key attribute.

☞ The normalization levels depend on each other. Therefore, each level must be evaluated sequentially and any level cannot be blindly skipped.

Un-normalized Database Schema

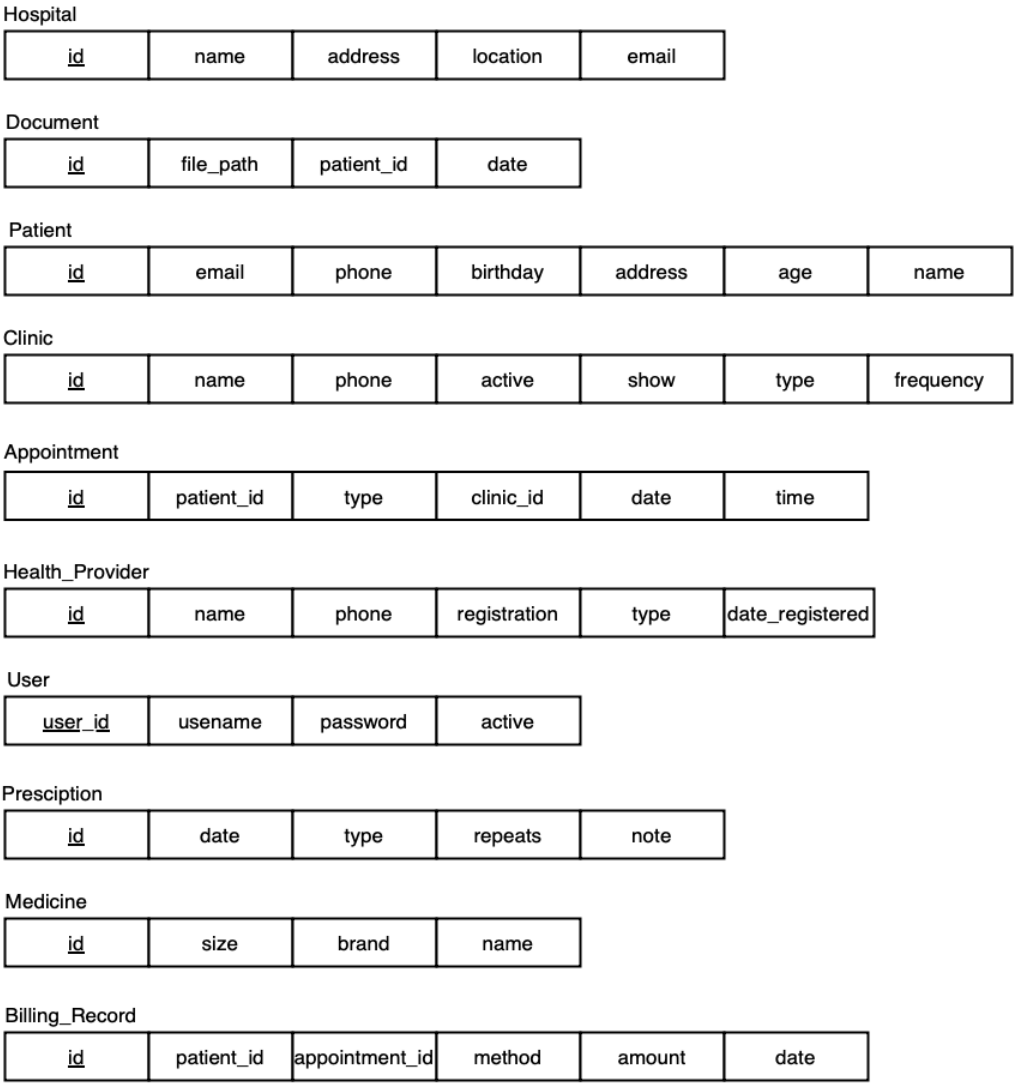


Figure 7: Un-normalized databse schema

Normalized Database Schema

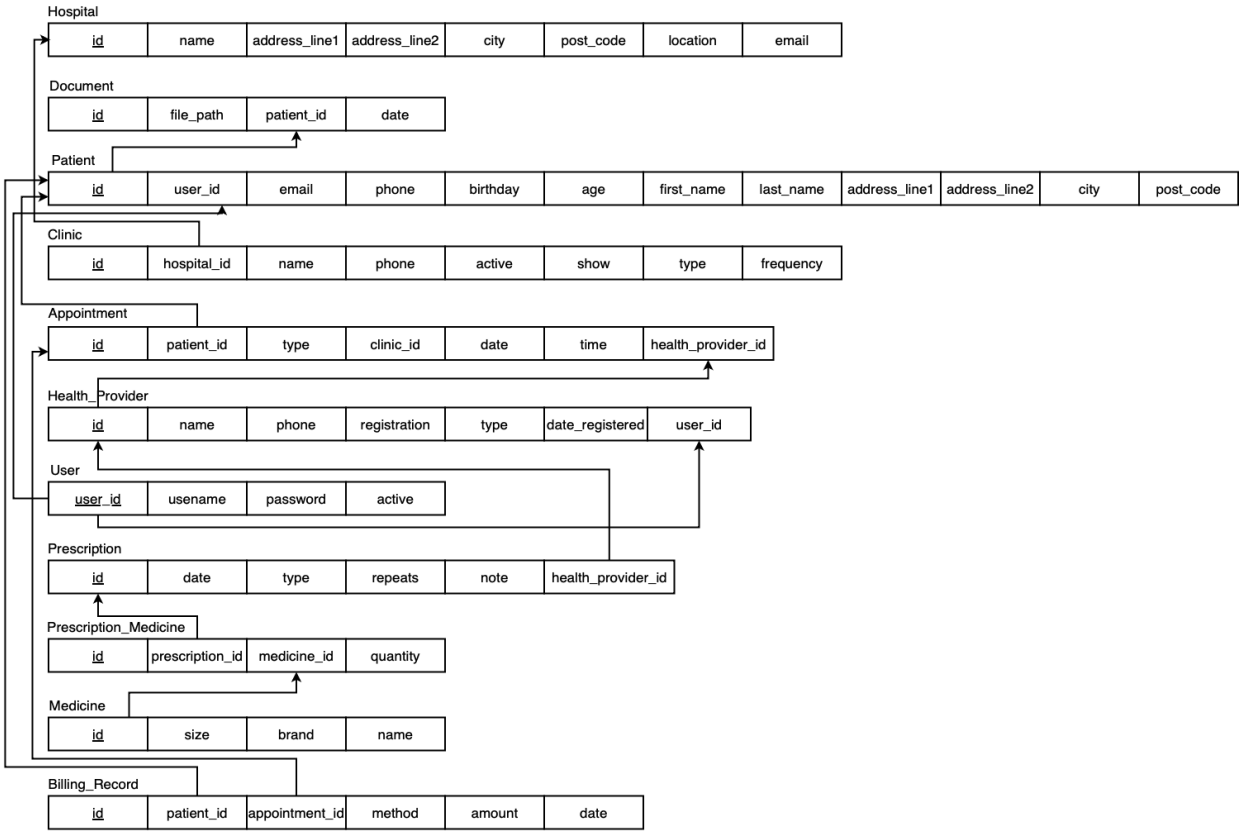


Figure 8: Normalized databse schema [Third Normal Form]

System Architecture

When making architectural decision for new systems there are key drivers that influence the design choices. Few of these important aspects can be listed as,

- **Total Cost of Ownership:** There are multiple potential solutions and deployment architectures for any system implementation. The cost factor is a defining factor that influence the architecture choices based on the organizations ability & willingness to invest on development, execution & maintenance of the system in general.
- **Performance Requirements:** Some systems require high performance low latency solutions in some use cases such as banking systems. Whereas some emphasizes on availability than the performance itself such as static websites. Depending on these requirements, solutions must be proposed with dynamic scalability, load balacing & fail-over options.
- **Functional Requirements:** Given the nature of the system, there are system-specific requirements such as integration with legacy systems or third party software solutions which must be considered to ensure the system is future-proof and able to meet & exceed stakeholder expectations.
- **Non-functional Requirements:** The architecture choices will also be influenced by aspects such as multi-lingual & accessibility features, the ability to cater forecasted amount of clients or produce results in benchmarked time alike.

> **The first step in IT system analysis and design is about finding an appropriate architecture for a particular business situation. The task can be difficult and challenging. It is easy to see why system architects need a deep understanding of the computer system and network components that comprise the modern IT system to make the appropriate design, selections, and trade-offs.**
>
> (Englander & Wong, 2021)

Proposed Deployment Architecture

Selection of system deployment architecture is one of the key milestones of the implementation lifecycle. This decision will impact the actual development strategy & finally how the whole system would behave as a single unit.

Some of the deployment architectures can be listed as,

- Thin/Thick Client
- Client/Server
- Web Services
- Micro-services *(increasingly popular)*

✍ Given the fairly less complex system requirements and considering the growing nature of business; we suggest micro-service based architecture for this system implementation. This choice tables additional benefits of quick go-to-market with minimalistic system functions and future-proof the system with flexibility for changes or expansion. Whilst being able to leverage cloud computing concepts, it will further ensure the recurring maintenance costs to be minimal with pay-as-you-go model of cloud computing.

Choice of Diagramming: C4 Model

According to (Simon Brown, 2024), the C4 Diagramming Model is an extension to Unified Modelling Language & 4+1 Software Modelling Technique. It provides clear overview to both arrangement and function of software system components & their interactions in simpler terms.

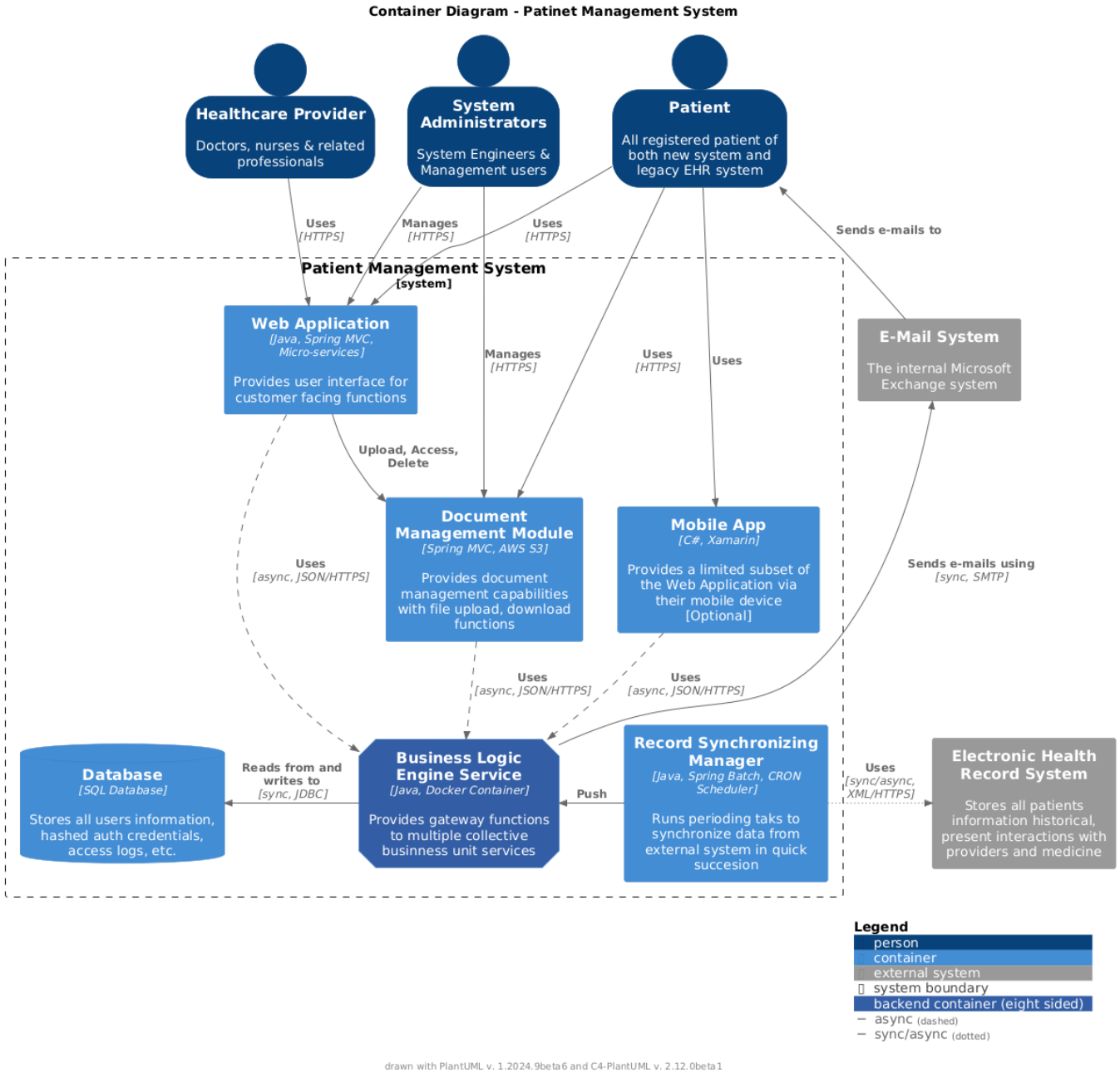


Figure 9: C4 - Context Diagram

Deployment Architecture - Overview

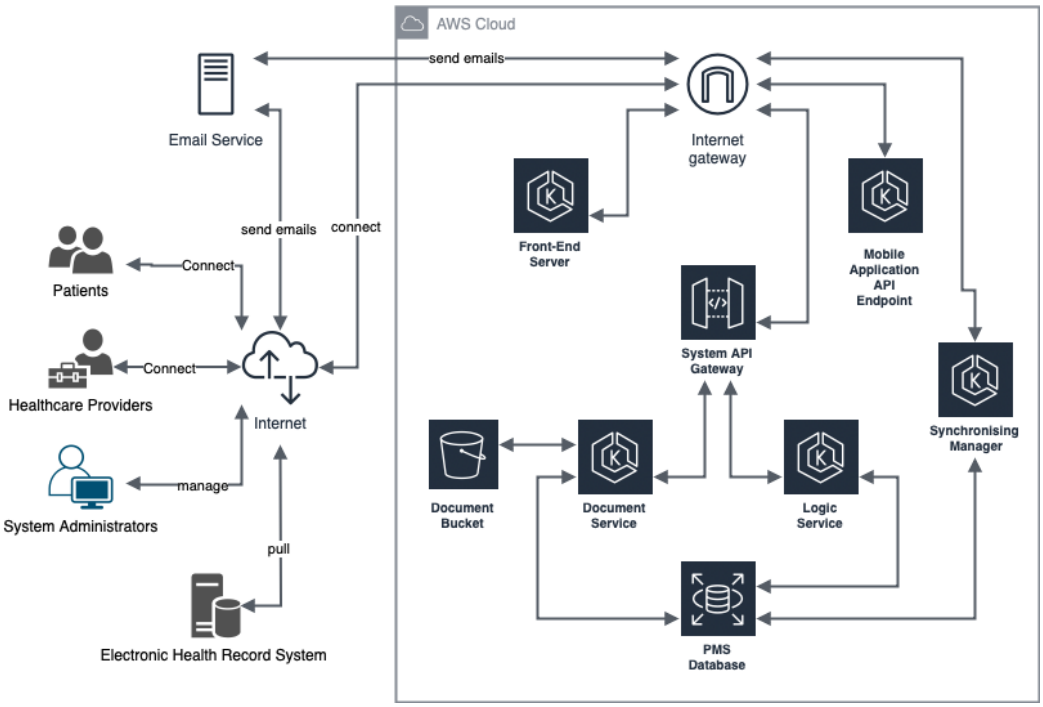


Figure 10: System Deployment Architecture

Figure 10 illustrates how services should be deployed. An assumption was made with electronic health record system is a legacy system hosted on-premise.

Following services can be identified as key micro-services,

- Front-End service
- Mobile API service
- Gateway service
- Document management service
- Synchronizer service
- Core logic service *(can be further decomposed into smaller micro-services)*

System Interfaces Wireframes

In the design of Patient Management system (PMS) and Document Management System (DMS), The user interfaces play an important role to ensure that the administrative staff, patients and healthcare providers smoothly and seamlessly.

For all the users, system must offer simple, easy-to-use and user-friendly interfaces that allow the users to carry out their tasks effectively. Five key interfaces designed for the system are given below which cover patient medical record access, patient appointment scheduling, patient registration, billing and document management.

Patient Registration Interface

This interface allows new patients to register themselves and provide their personal, medical, and insurance details.

Features

This interface includes a section for the patients to input their personal information such as name, date of birth, gender, contact information and address. Another section where patients can input their medical history. Sections for patient's insurance and emergency contact details, and a button to submit the response.

Design Consideration

The design of the interface is clear with logical flow and grouping related fields (personal information, medical history, insurance) together. A progress bar can be included to let the patients know about the registration process. The interface should be mobile friendly to allow the patients to register themselves using smartphones and tablets.

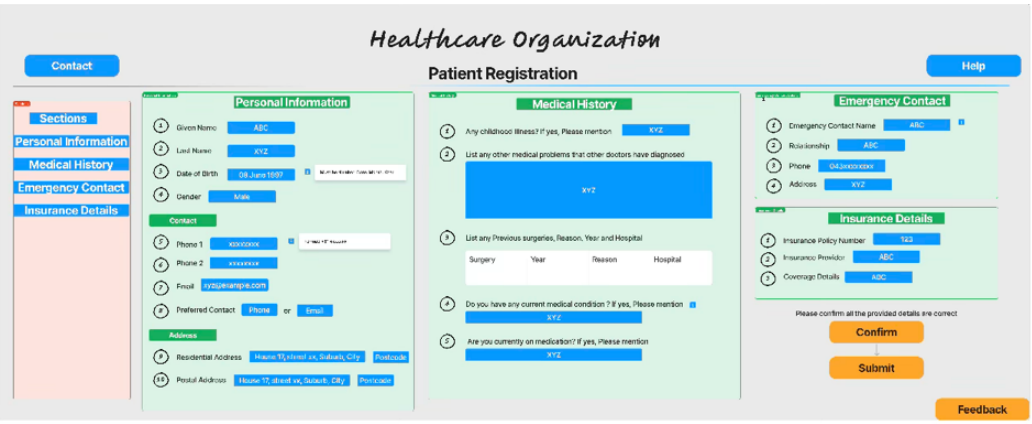


Figure 11: Patient Registration Interface

Appointment Scheduling Interface

This interface allows patients to schedule their appointments with doctors. It should be simple and clearly state the available time slots and available healthcare providers. It allows patients to reschedule and cancel their appointments.

Features

This interface includes search section which allows patients to search healthcare provider either by name or by speciality. A section for patients to select appointment time date. This interface also has a section where patients can provide the reason for their visit and current symptoms. Lastly a section to review the appointment details and confirm appointment.

Design Consideration

Design of the interface is clear and simple. It includes helpful instructions or tooltips for first-time users to guide them through the appointment scheduling process.

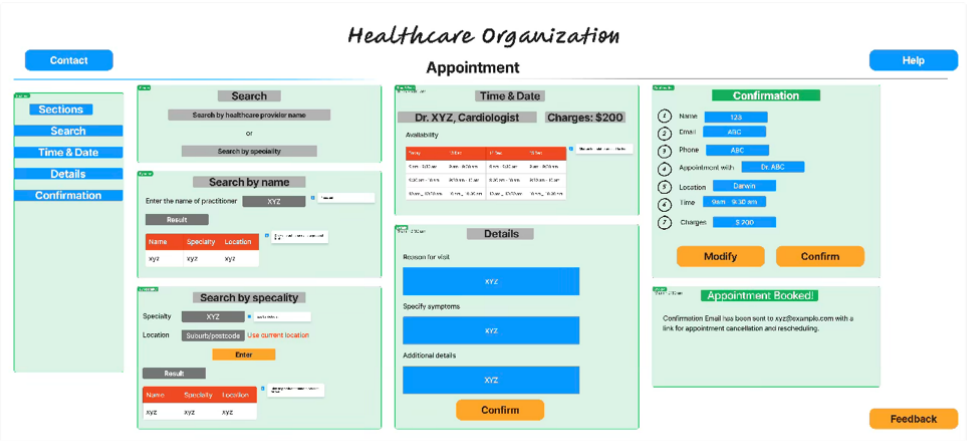


Figure 12: Appointment Scheduling Interface

Patient Portal Interface

This interface allows patients to view manage their details as well as view their records. It allows patients to message the healthcare provider directly. It offers transparency and convenience for patients to manage their healthcare.

Features

A section from where patients can access and review their medical records including diagnoses, treatment plans, test results, and prescription history. A section from where patients can schedule an appointment. A section which can provide patients with their detailed billing information.

Design Consideration

The interface is user-friendly. It provides a clean and organized layout with sections clearly labelled. It also has interactive feature which allow patients to click on sections for more detailed information or to download documents.

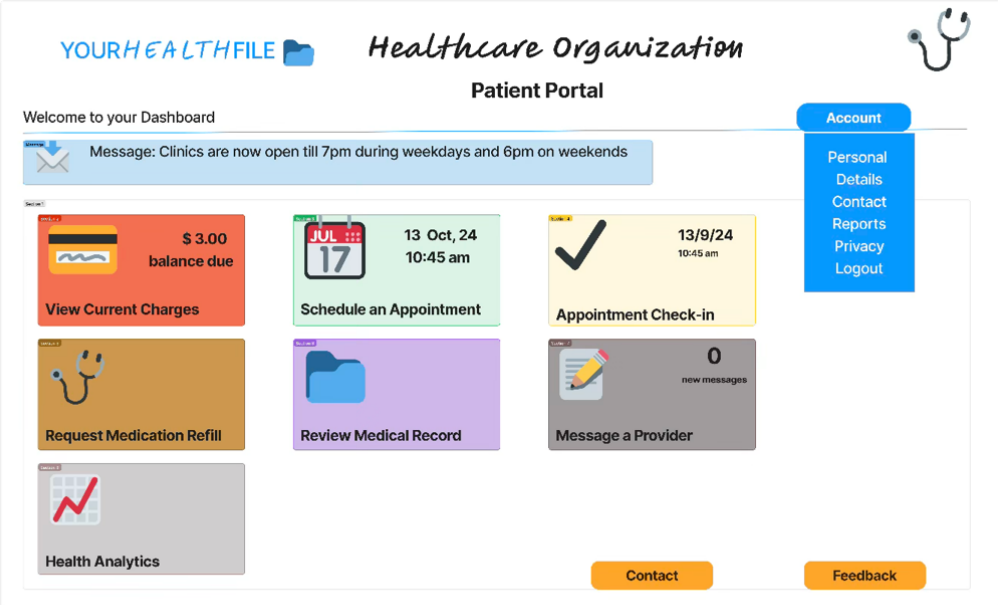


Figure 13: Patient Portal Interface

Healthcare provider dashboard Interface

This interface is designed for healthcare providers (doctors, nurses, specialists) to manage patient care and treatment. The system should allow easy access to patient records, appointment scheduling, and updating medical documents.

Features

On left-hand side, it consists of a section from where the healthcare provider can directly access his/her personal details, patients list etc. It provides details of the number of consultations booked and number of patients consulted in a day. It also provides details regarding the patients who have booked the consultation e.g. their personal details, appointment time, patients medical records and notes.

Design Consideration

Efficient Workflow: The interface allows healthcare providers to perform tasks efficiently with minimal clicks, as they may need to access and update multiple patient records in a short time.

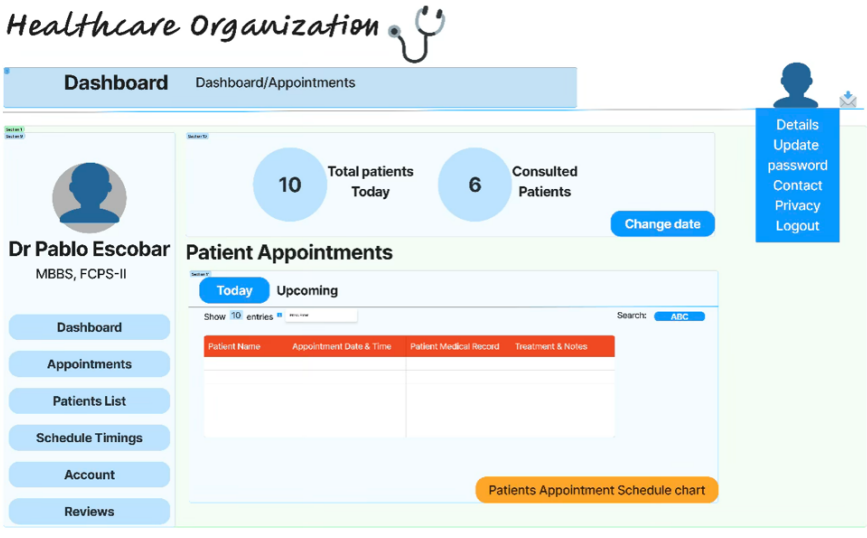


Figure 14: Healthcare provider dashboard Interface

Billing Interface

This interface is designed for staff and patients to manage billing processes. It provides accurate and up-to-date information regarding Total bills, paid amount, due amount and most recent appointment details.

Features

It consists of a section detailing the patients personal information including insurance policy number and insurance provider. It provides details of the most recent consultation with healthcare provider. Payment summary provides information regarding the total amount, how the total amount is divided, paid amount and balance. It also consists of a section from where patients can request to pay the due amount in instalments.

Design Consideration

The billing interface clearly lists everything along with costs, so that patients can easily understand their bill and pay the remaining amount.

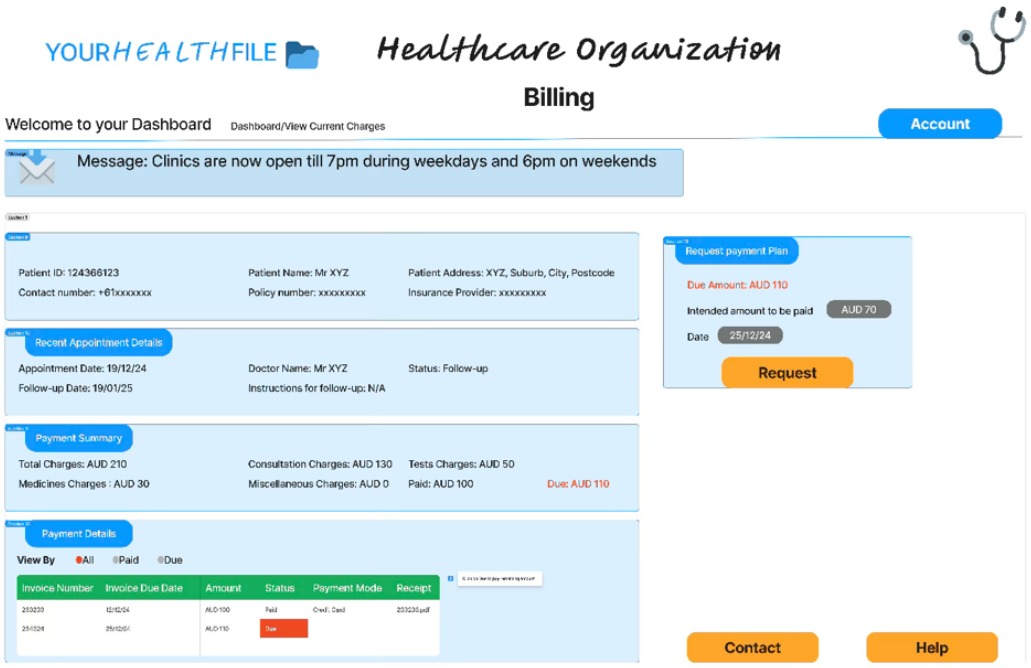


Figure 15: Billing Interface

Reporting

System reporting is a key feature of any Patient Management System (PMS) and Document Management System (DMS), as it helps healthcare administrators, clinicians, and financial teams track and analyse vital data for decision-making, operational improvements, and compliance purposes. Effective reporting can contribute to improving patient outcomes, reducing costs, and enhancing operational efficiency.

The report clearly displays the duration as well as number of patients visited during that duration. It provides detailed information of the patients who booked consultations during that timeframe. It details the patient's name, date of consultation, medical practitioner by whom the consultation was done, total charges of each consultation, amount paid and due amount of each consultation. These details will help the management to keep patient and financial record.

A rough number of future consultations booking with a specific practitioner can be calculated from follow-up column. It also details the previous weeks and current week's operational efficiency of the organization as well as the percentage difference which will help in future planning and decision making. Challenges sections state all the challenges that needs to be overcome to improve patient's experience and increase profits

Figure 16: Reporting Interface

System Development Strategy

Development Strategy: Agile Methodology

Agile methodology is a development strategy tailored to the Patient Management System (PMS) project. This approach emphasizes iterative and incremental development, allowing for flexibility and adaptability throughout the development process.

Key principles of agile methodologies:

- **People and interaction:** Agile prioritizes people and communication over process and tools.
- **Working software:** Agile focuses on delivering working software in short iterations, not perfect documentation. Customer collaboration: Agile encourages close collaboration between developers and stakeholders to ensure the software meets their needs.
- **Responding to change:** Agile is flexible and adaptable, allowing for changing requirements and priorities throughout the development process.

Agile Development Process

The agile development process for a PMS project involves the following steps:

- **Project Planning:** Define the project scope, objectives, and deliverables. Identify stakeholders and their roles. Sprint planning: Break down your project into smaller, more manageable pieces called sprints, each with a defined set of deliverables and timeline. Run sprints: Develop software in short iterations, usually 2-4 weeks long, with the development team tackling high-priority problems first.

- **Daily stand-ups:** Hold daily stand-ups to discuss progress, plans, and possible roadblocks.
- **Sprint reviews:** Hold a review meeting at the end of each sprint to demonstrate to stakeholders how the software works.
- **Sprint retrospective:** Hold a retrospective meeting to discuss what went well and what didn't work and identify areas for improvement.
- **Release planning:** Plan the software release, including testing, deployment, and training.

Agile Roles and Responsibilities

- **Product Owner:** Responsible for defining and prioritizing the product backlog.
- **Scrum Master:** Responsible for facilitating the agile process, removing impediments, and ensuring the team follows agile principles. Development Team: Responsible for developing, testing, and deploying software.
- **Stakeholders:** Responsible for providing feedback, comments, and recommendations throughout the development process.

Agile Tools and Techniques

- **Jira:** A project management tool for tracking issues, prioritizing tasks, and monitoring progress.
- **Trello:** A visual project management tool for organizing tasks, tracking progress, and collaborating with team members.
- **Version control systems:** Tools like Git for managing code changes, tracking versions, and collaborating with team members.
- **Continuous integration/continuous deployment (CI/CD):** Tools like Jenkins for automating software testing, builds, and deployment.

Benefits of Agile Methodology

- **Flexibility:** Agile allows requirements and priorities to change throughout the development process.
- **Faster time to market:** Agile enables rapid development and deployment of software, resulting in faster time to market.
- **Improved collaboration:** Agile fosters closer collaboration between developers, stakeholders, and clients to ensure that the software meets their needs.
- **High quality:** Agile emphasizes continuous testing and integration to ensure that the software is of high quality and meets the required standards.

Problems and strategies to alleviate the consequences

- **The breakdown of communication:** Regular meetings, open communication tools and cooperation can help soften communication.
- **Angels with scale:** Clearly determine the project scale, the priorities of tasks and the regular consideration of progress can help to soften the flipper.

Conclusion

The proposed Patient Management System (PMS) aims to streamline patient registration, appointment scheduling, treatment tracking and billing across 50 clinics and the central hospital. The system will be integrated with the existing Electronic Health Record (EHR) system to improve patient care and streamline administrative processes.

Key Takeaways:

- **Improved patient care:** A PMS gives healthcare providers real-time access to patient records, allowing them to make informed decisions and deliver better care.
- **Improved administrative efficiency:** The system automates many administrative tasks, reducing the workload of medical staff and minimizing errors.
- **Improved patient engagement:** A PMS provides patients with online access to their medical history, appointment schedules, and billing information, thereby promoting patient engagement and empowerment.
- **Improved decision-making:** The system provides healthcare managers with real-time data and analytics, enabling them to make informed decisions on resource allocation and service improvement.

Future Directions

- **Integration with other systems:** PMS can integrate with other healthcare systems such as laboratory information systems and radiology information systems to provide a comprehensive view of patient care.
- **Artificial Intelligence and Machine Learning:** Artificial intelligence and machine learning capabilities can be leveraged to enhance the system to analyze patient data and provide predictive information to improve patient care.
- **Mobile accessibility:** The PMS can be optimized for mobile devices, allowing providers and patients to access the system remotely.
- **Data analytics:** The system can be enhanced with data analytics capabilities to provide insights into patient care, treatment outcomes, and service utilization.

Recommendations

- **Perform in-depth tests:** To carefully test the PMS to ensure that it meets the requirements of medical workers and patients.
- **Provide training and support:** Provide training and support for medical staff to ensure that it is practical for them to use the system.
- **Monitor and Evaluate:** Monitor and evaluate the effectiveness of the PMS in improving patient care and administrative efficiency.
- **Continuously Improve:** Continuously improve the PMS based on feedback from healthcare providers, patients, and other stakeholders.

Limitations

- **Technical issues:** Technical issues such as system integration and data migration may arise during PMS implementation.
- **Resistance to change:** Medical staff may resist changes in workflow and processes, which may affect the acceptance and effectiveness of the PMS.
- **Data Security:** A PMS must ensure the security and privacy of patient data, which can be a challenge given the growing threat of cyber-attacks.
- **Cost:** Implementing and maintaining a PMS can require a significant investment, which can be a challenge for healthcare organizations with limited resources.

By managing these restrictions and improving PMS, healthcare organizations can guarantee that the system meets the development needs of medical and patient service providers and helps to improve patients and management efficiency.

Bibliography

Britton, C., & Doake, J. (2005). *A Student Guide to Object-Oriented Development* [https://www.sciencedirect.com/topics/computer-science/case-description(visited 2024-12-09)]. Butterworth-Heinemann.

Englander, I., & Wong, W. (2021). *The Architecture of Computer Hardware, Systems Software, and Networking - An Information Technology Approach [Sixth Edition]* [https://www.perlego.com/book/3865635(visited 2024-12-09)]. Wiley.

Lightstone, T. T. S., Nadeau, T., & Jagadish, H. (2010). *Database Modeling and Design [Fourth Edition]* [https://www.perlego.com/book/1813726(visited 2024-12-09)]. Morgan Kaufmann.

Simon Brown. (2024). Introduction | C4 Model [Accessed: 2024-12-10]. https://c4model.com/introduction

Use case diagram for hospital management system [Accessed: 2024-12-10]. (2024). https://edrawmax.wondershare.com/templates/use-case-diagram-for-hospital-management.html