FYP Report

A Web-Based Approach to Hospital Bed Management

Luke O'Leary - 20264747

Supervisors:

Dr. Meghana Kshirsagar

Dr. Alison O'Connor

Table of Contents

Chapter 1 4

1.1 Introduction 4

1.2 Project objectives 5

Chapter 2 6

2.1 Current Bed Management Processes 6

2.2 Existing Bed Management Software 6

2.3 Impact of Bed Management on throughput of patients 7

2.4 Factors that affect bed assignment. 8

2.5 Security and Privacy 10

2.6 Current technologies used in hospitals. 10

Chapter 3 12

3.1 Users 12

3.2 Functional requirements. 13

3.3 Non-functional requirements 13

3.4 Project Timeline 14

3.5 User Interface/User Experience (UI/UX) 14

3.6 UI/UX Prototypes 15

3.7 Database 17

Chapter 4 19

4.1 Frontend Design 19

4.2 Backend Design 23

4.3 Continuous Integration/Continuous Delivery (CI/CD) 24

4.4 SonarCloud 25

4.5 Unit & Integration Testing....................26
Chapter 5....................27
5.1 Results....................27
5.2 Discussion....................35
5.3 Conclusion....................36
References....................37

Table of Figures

Figure 1: GANTT chart for planned timeline 14
Figure 2: UI: Main Dashboard 15
Figure 3: UI: Administrator Dashboard 16
Figure 4: UI: New Bed Assignment 16
Figure 5: UI: Login 17
Figure 6: Database schema design 18
Figure 7: Stacked bar chart showing bed availability by ward 20
Figure 8: Pie chart showing total bed availability for the hospital 20
Figure 9: Code snippet demonstrating chart implementation 21
Figure 10: Hierarchical Sitemap 22
Figure 11: OpenAPI Swagger 24
Figure 12: GitHub action code snippet 25
Figure 13: User login screen 27
Figure 14: A dynamic dashboard tool that depicts the demand-supply of beds 28
Figure 15: Bed and patient information page 28
Figure 16: Searchable patient and user table screen 29
Figure 17: Patient admission screen 30
Figure 18: Screen to request patient transfers 30
Figure 19: Screen to allow bed managers manage transfer requests 31
Figure 20: Screen to enable system administrators create or delete beds and wards 32
Figure 21: Screen to enable system administrators create or delete users 32
Figure 22: Popup requesting temporary user password for newly created account 33
Figure 23: Example of table used within the application 33

Chapter 1

1.1 Introduction

In the dynamic field of healthcare, the efficient management of hospital resources plays a vital role in ensuring the highest quality of patient care. As healthcare facilities deal with the ever-increasing demand for services, the need for effective bed management systems has become more critical than ever. This final year project, titled "A Web-Based Approach to Hospital Bed Management" aims to address the challenge of resource allocation by developing a web app to enhance the allocation and utilisation of hospital beds, a critical aspect in healthcare facilities.

Hospitals face the constant challenge of balancing the number of incoming patients with the availability of beds, healthcare personnel, and other essential resources. The efficient management of hospital beds is a complex task that requires careful coordination and real-time decision-making [1]. it involves a strategic approach to handle various aspects of patient care and resource allocation effectively.

The primary goal of this project is to design and implement a robust hospital bed management system that optimizes the patient journey within the hospital. The proposed system will leverage software to facilitate the seamless flow of patients from admission to discharge. By enhancing bed allocation and improving resource utilisation, the system aims to maximize efficiency while ensuring the highest standards of patient care.

The consequences of poor bed management serve as motivation for this project. Challenges such as inefficient resource utilisation and slow patient flow can contribute to many serious issues in hospitals such overcrowding and care delays which has consequences for patient treatment, staff/patient satisfaction and many more cascading effects [2]. These issues highlight the importance of effective bed management. The goal of this project is to deliver a suite of features for bed management software to enhance the operational efficiency of bed allocation and patient transfers leading to timely access to healthcare services for patients.

1.2 Project objectives

The goal of this project is to create a web-based hospital bed management system. This goal will be achieved through the following objectives:

1. Design User interface/User experience (UI/UX) templates.
2. Design and create a database for the application.
3. Implement firebase authentication.
4. Create basic interface to create / view admissions, beds, wards and patients.
5. Create administrator panel.
6. Implement transfer functionality.
7. Implement QR code generation for quick access.
8. Create dashboard view.
9. Create mobile friendly version.

Chapter 2

2.1 Current Bed Management Processes

In the context of Ireland’s healthcare system, bed management in hospitals is a multifaceted process, involving various specialized roles and structures management systems. The complexity is evident in Cork University Hospital (CUH) which is within the Cork University Hospital Group (CUHG), where bed management is overseen through three key structures. Specifically, bed management, discharge coordinates and scheduled care/bed all of which are supervised by the head of bed management who reports to the hospital groups CEO.

St. James’s Hospital in Dublin exhibits a comparable structure in managing its beds. Their bed management team consists of the assistant director of nursing, a scheduled care lead for inpatient and day-case patients, administration staff, discharge coordinates, a patient flow manager and a healthcare assistant [3]. This configuration shows the diverse roles required for bed management, including administrative and medical personnel who all have their own responsibilities.

These approaches reflect a broader trend in the Irish healthcare system, where the coordination of patient flow and bed allocation is increasingly reliant on specialized roles and structured management systems. The emphasis on multiple roles, including discharge coordinators and patient flow managers is due to the diverse needs and challenges faced in modern healthcare settings. These settings require structures to manage the flow of patients. It's clear that these hospitals use structured methods to manage their bed allocation, though the impact and effectiveness of these structures is difficult to measure.

2.2 Existing Bed Management Software

The need for software-based bed management systems is evident by the amount of existing solutions currently available on the market [4–6]. However, there is limited information on the exact features of these solutions on their respective websites. Most websites require a scheduled demonstration to gain a better understanding of their solution.

One example is the company “Advanced Data Systems Corporation” which offers the “Medics BedManager” system which advertises real-time bed information, admissions dashboard and a projected occupancy dashboard. “Medics BedManager” claims to maximize occupancy rates, bed revenue, and improve utilization rates [6].

The systems made by “SoftClinic” and “Advanced Data Systems Corporation” have the advantage of being able to integrate with other management systems they have, such as medical/health records to provide a more seamless experience to medical professionals. [5,6]

The market for bed management software is diverse, encompassing a range of systems that offer similar and unique features and tools. These features typically aim to enhance hospital efficiency by providing real-time tracking of bed availability, facilitating smoother patient admissions and discharges, and optimizing bed usage. Despite the apparent benefits the selection of an appropriate system can be challenging for healthcare facilities due to factors like system availability, compatibility, cost, staff training and regulation compliance.

2.3 Impact of Bed Management on throughput of patients

Effective bed management is crucial in the context of patient throughput in hospitals. The significance of this is captured in this statement, related to the UK’s National Health Service (NHS) who state “There is a growing realization that effective management of the flow of inpatients through acute hospitals is fundamental to good quality in patient care and to achieving the relevant NHS Plan Targets” [1]. Here, the focus of management, including bed management, highlights its importance in ensuring efficient patient flow and maintaining high standards of patient care.

In the context of bed management, the challenges in Accident & Emergency (A&E) departments are particularly illustrative. A&E trolley wait times have been attributable to a shortage of beds to move patients into which highlights a direct consequence of bed management inefficiencies [7] . The absence of available beds leads to significant delays in patient admissions from A&E, thereby affecting the overall efficiency of patient care and throughput.

Furthermore, the issue of hospital overcrowding is a widespread problem resulting in constrained bed capacity and slow admissions having negative impacts on patient care and safety [2]. This overcrowding is another facet where bed management plays a crucial role. Overcrowding, often contributed to by inadequate bed management, not only compromises the quality of patient care but also poses risks to safety, emphasizing the need for effective strategies in bed allocation and management.

These insights collectively highlight the crucial role of bed management as a component of hospital management in influencing patient throughput, demonstrating its significance in hospital efficiency and the quality of healthcare.

2.4 Factors that affect bed assignment.

Bed assignment in hospitals is further complicated by various factors, each playing a key role is patient care, safety, and satisfaction. These factors are critical in determining where patients are placed within the hospital.

Care Requirements: The specific medial requirement of a patient is among the biggest factors that influence bed assignment. Patients with similar conditions are generally put in the same wards as these wards have the specialized equipment to best treat the patient. For example, some specialized wards include, Oncology, Cardiology, Postnatal and many more. In addition to equipment different wards typically have different staffing levels. The table below shows the nurse-patient ratio recommendations for California in 2008.

Table 1: Nurse to patient ratio in specific purpose wards. Adapted from [8]

Unit/Location	Nurse to Patient Ratio
Intensive / Critical Care	1:2
Paediatrics	1:4
Neonatal Intensive Care	1:2
Emergency	1:4
Operating Room	1:1
Post Anaesthesia Recovery (PAR)	1:2
Trauma Patients in ER	1:1
Ante partum	1:4
Labour & Delivery	1:2
Postpartum- mother & baby	1:4
Postpartum- women only	1:6
Medical Surgical Wards	1:5
Other speciality care	1:4
Psychiatric	1:6

These ratios ensure that each patient receives the appropriate level of care based on the intensity and complexity of their medical needs.

Gender: In many cultures, segregation of patients by gender in hospital wards is essential to respect privacy and cultural norms. Additionally, patient safety must be considered.

Age: Age is a significant factor in determining the type of care a patient requires. For example, a neonatal unit would not be capable of providing the same level of care to an 80-year-old patient. This extends to the physical aspects of care, such as size of the bed. For instance, paediatric patients need smaller beds tailored to their size, while adults require standard-sized hospital beds. Staffing levels and training can also vary between wards, it has been shown that the safety of children can be improved by having specialised safety training classes for nurses and reducing the work shifts and overtime of nurses in paediatric wards [9].

2.5 Security and Privacy.

The healthcare industry, being a repository of sensitive health information, is a common target to various forms of cyber-attacks. The personal patient information which is stored could have serious consequences if exposed. These various forms of attack can be grouped into four key types: Physical, Software, Network and Encryption attacks each presenting its own difficulties requiring unique solutions [10].

To combat these threats healthcare institutions must always consider the security of their systems and privacy of their patients. It is important that software deployed is secure against attack. Regular training for staff to protect against phishing and social engineering is just as important. A report by ‘Packetlabs’ showed “82% of data breaches contain a human element” in 2023 [11].

2.6 Current technologies used in hospitals.

In addition to bed management systems, modern hospitals incorporate a variety of non-bed management technologies to enhance various aspects of healthcare delivery. These technologies streamline everyday tasks, aiming to improve patient care, and ensuring effective management of health services.

Hardware: The use of mobile computers is essential to support the many software systems deployed in hospitals. These computers can come in many forms such as tablets, laptops, computers, computers of wheels (COW) and even virtual reality (VR) headsets. These mobile devices are essential for work done at the bedside. Nurses and doctors during ward rounds need highly portable devices to perform their clinical tasks. There is a strong preference for COWs in favour of other devices among nurses and doctors for these tasks [12]. Mobile devices enable staff to directly update systems on the go without needing to take careful notes and return to a stationary computer.

Infectious Disease Tracking: A technology in this domain is the tracking of infectious diseases. In Ireland, for example, there is an information system to track, manage and control infectious diseases called “Computerised Infection Disease Reporting” [13]. This system allows for the efficient reporting, tracking and management of infectious diseases across Ireland, playing a critical role in public health monitoring and response strategies.

Digital Medical/Health Records: The adoption of digital medical and health record systems is widespread in hospitals and medicine. Systems provided by companies like “SoftClinic” and “Advanced Data Systems Corporation” offer digital medical/health records [5,6]. These systems enhance medical professionals access to patient information, making it easier for healthcare providers to access and update patient records, thus improving efficiency.

As these systems have not always existed or been utilized the extensive hardcopies of patient’s records are being digitized in many situations. “Vitrosoftware” is one such company offering this solution. This software has been adopted in major Irish hospitals such as Cork University Hospital and LauraLynn Childre’s Hospice. The software is capable of digitising old patient records and other paper processes into their digital system without sacrificing the original look [14,15].

Chapter 3

3.1 Users

Four key types of users were identified as requiring unique access to the bed management system:

- Doctors & Nurses.
- Bed Managers.
- Cleaners.
- Administrators.

The categorization of these users aims to follow the principle of least privilege (POLP). Which states a user should not have more access to a system than is required to complete their task [16]. The table below shows the levels of access users should have within the system.

Table 2: Permissions & System Access

Features	Doctors & Nurses	Bed Managers	Cleaners	Administrators
Assign/Unassign Patients to beds	✓	✓		
Search for patients	✓	✓		
Make Transfer Requests	✓	✓		
Approve Transfer Requests		✓		
Update bed status from requires cleaning to Ready	✓	✓	✓	
Create/Update User Accounts				✓
Create/Update Wards				✓

3.2 Functional requirements.

- **Patient-bed assignment:** Ability to find and assign a bed suitable for a specific patient.
- **Patient Discharge:** Ability to unassign a patient from a bed and mark the bed previously occupied as requiring cleaning.
- **Bed Transfer Requests:** Transfer a patient from one bed to another either in the same hospital or to another.
- **Bed Transfer Management:** Ability to approve transfer requests.
- **Patient Location Tracking:** Ability to find which bed a patient was assigned to.
- **Ward and Bed management:** Ability to create and update the configuration of wards and beds.
- **User Account Management:** Ability to create and update user accounts.
- **QR Code integration:** Ability to create unique QR Codes for every bed and for them to be scanned.

3.3 Non-functional requirements.

- **Security:** The system should be built with cyber-security and patient privacy as a key factor.
- **Scalability:** As hospitals expand or contract the system needs to adapt seamlessly.
- **Adaptability:** As new systems and requirements are introduced the system should be easy to modify and adapt to same.
- **User-friendly:** The system should be built with a user-friendly interface to minimize training required.
- **Reliability:** A robust system that minimises as disruption which could affect overall efficiency.
- **Performance:** The system should operate with minimal computing delays and lag.

3.4 Project Timeline

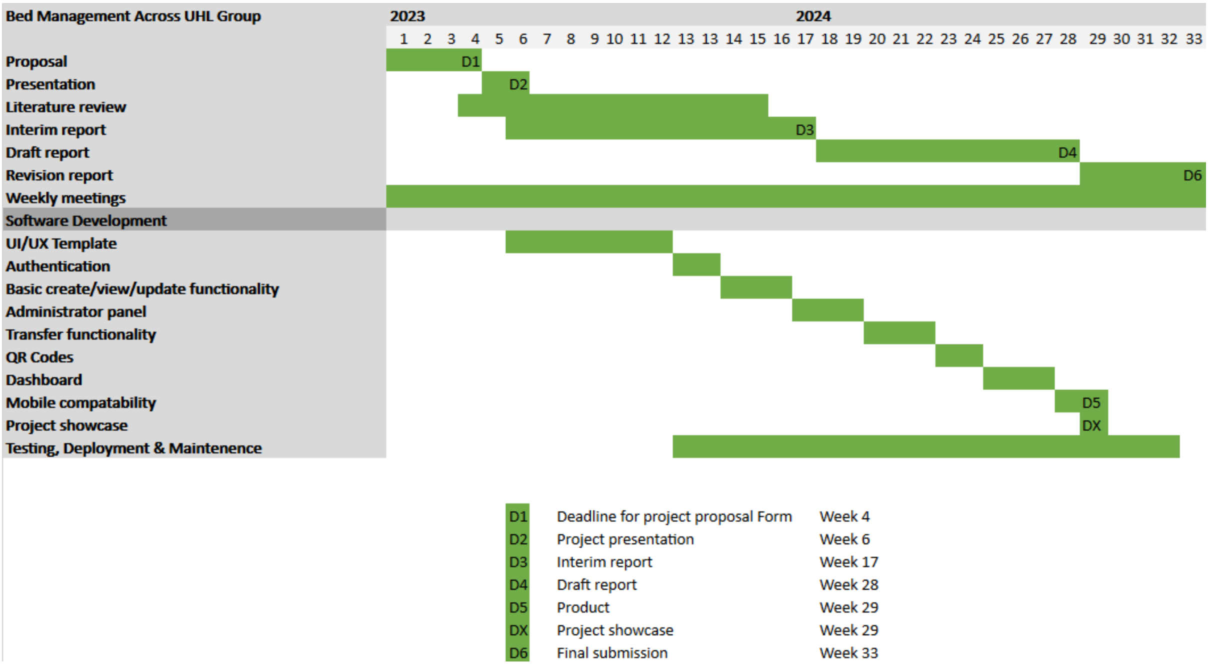

Figure 1: GANTT chart for planned timeline

3.5 User Interface/User Experience (UI/UX)

When designing the UI/UX for the hospital bed management system, several key considerations were made to ensure the system is both functional and user-friendly. This is especially important in the fast paced and stressful hospital environment.

Firstly, the interface has large buttons and interactive elements. This design choice is crucial since healthcare professionals often wear gloves, which can cause issues when trying to navigate with smaller buttons. This aims to help reduce the risk of incorrect user input [17].

Another consideration to reduce the risk of incorrect user input is the integration of QR codes. These codes can be created within the application, assigned to a specific bed and then once scanned, bring the user directly to a specific webpage for assigning or unassigning patient to that bed. This functionality not only makes the process more efficient but also eliminates the potential of making a typo while inputting unique bed identification. This feature is particularly beneficial where speed and accuracy are important.

Furthermore, the system avoids using long unfilterable lists and tables that fill up the user's screen. As the number of patients, beds and wards etc change the system must be able display large sets of information in an organised manner. Filterable lists and tables will allow users to access and view extensive data efficiently without displaying too much unnecessary data.

Overall, these UI/UX considerations are essential for developing a system that is tailor made to the environment in which it will operate. The goal is to have an interface that is quick and easy to navigate while requiring minimal staff training.

3.6 UI/UX Prototypes

The figures 2-5 below showcase the emphasis on large buttons and easy navigation.

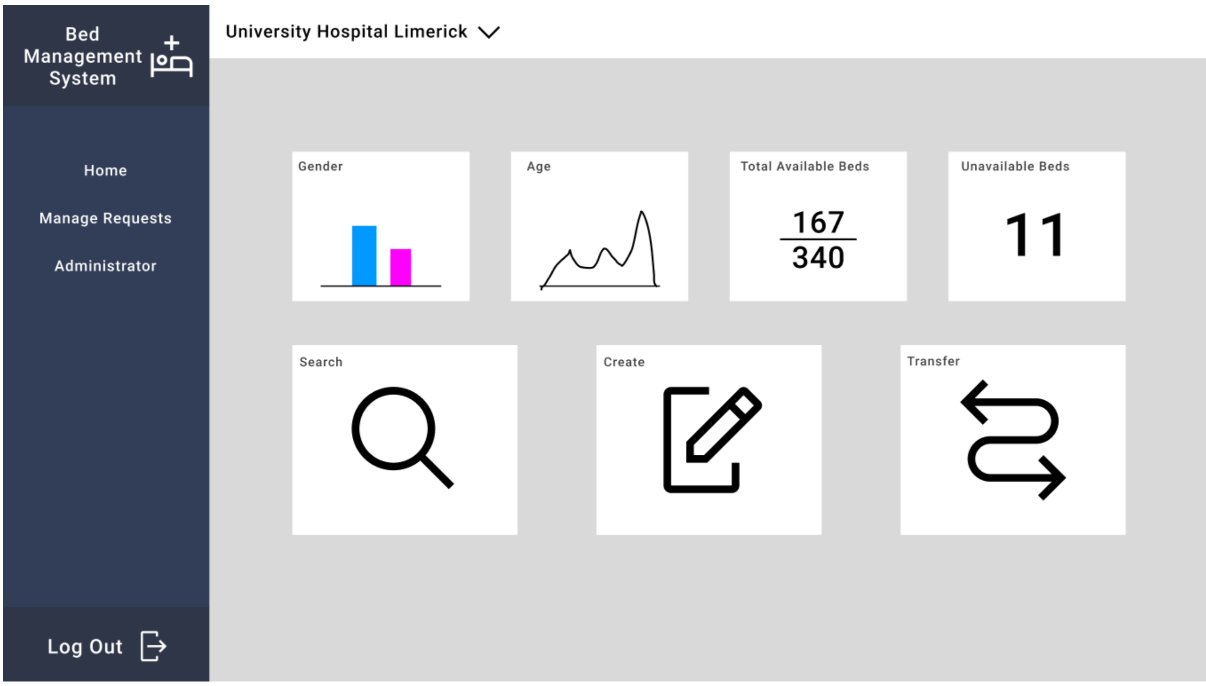

Figure 2: UI: Main Dashboard

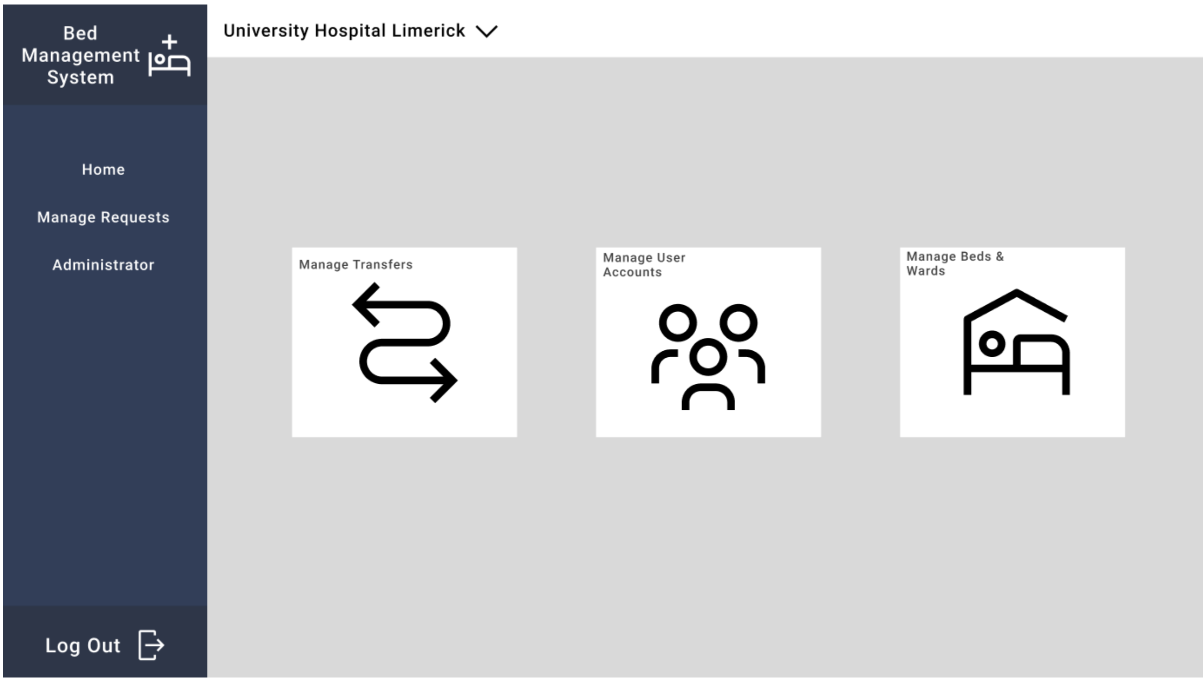

Figure 3: UI: Administrator Dashboard

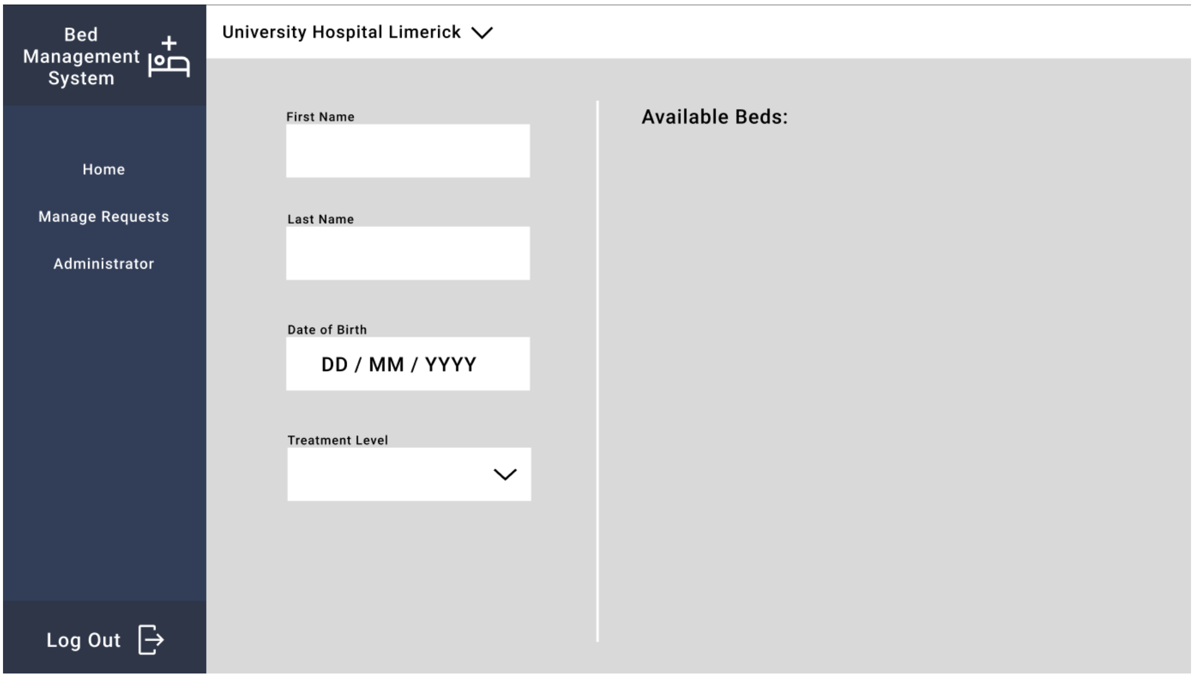

Figure 4: UI: New Bed Assignment

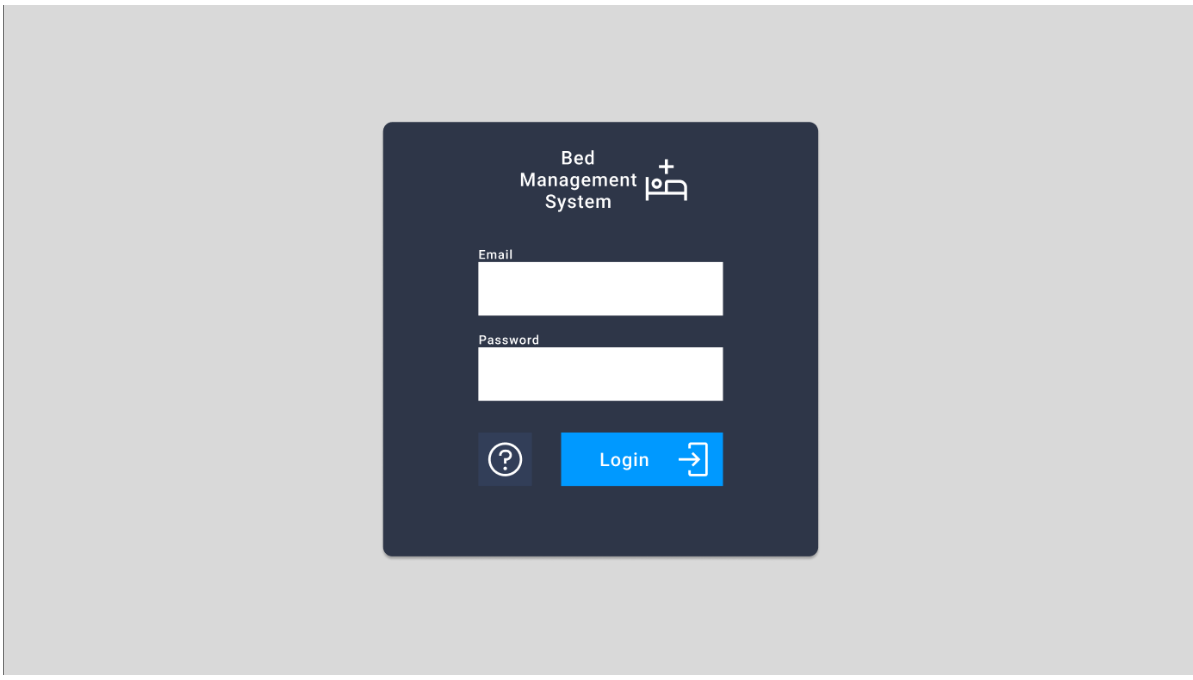

Figure 5: UI: Login

3.7 Database

In this research project a MySQL database was selected as the most appropriate database because of its widespread adoption due to its reliability and efficiency. When designing the MySQL database two key considerations were made: adaptability and traceability. These considerations were critical in ensuring the database design could meet the ever-evolving needs of hospitals while maintaining a record of activities. Figure 6 describes the MySQL database form.

Adaptability: The database structure is designed to be flexible and adaptable to various hospital's needs. For instance, a ward has a "treatment_level", instead of simply naming the treatment level in the `wards` table it links to the "treatment_levels" table instead, to better describe the ward by including the treatment levels name, description and equipment required. Additionally, beds can be disabled for multiple different reasons which can be listed in the `disabled_reasons` table. This allows the system to better reflect the real world such as a bed not being available because it requires repairs vs it needs to be cleaned.

Traceability: Traceability is achieved through record keeping and the use of timestamps and user references in the database. Every significant action, such as bed assignments and booking requests, is recorded with timestamp and references to the users who performed the actions. This leaves an auditable trail of interactions in the system to help with debugging and to add a level of accountability.

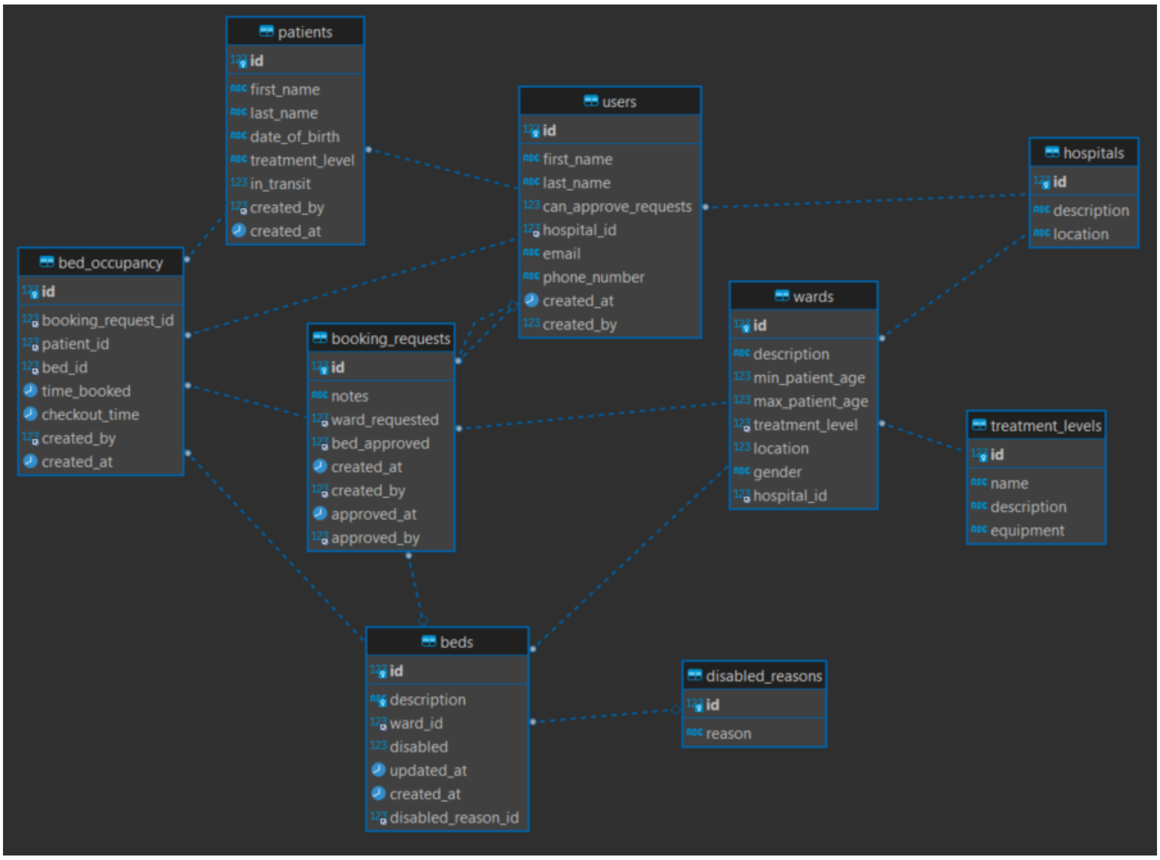

Figure 6: Database schema design

Chapter 4

4.1 Frontend Design

Language and Framework

For frontend development Vue.js and Node.js were selected for creating the single-page application (SPA). Single page applications dynamically rewrite the current web page instead of loading an entire new page. This makes it feel more responsive than a traditional web page. Vue.js has two main competitors that were also considered: Angular and React. Ultimately Vue was selected due to its smaller learning curve and good performance [18,19].

Firebase Authentication

Googles Firebase Authentication was chosen for several key reasons. Firebase provides a comprehensive suite of authentication methods including allowing the integration of Microsoft and Google accounts. Depending on if a hospital has already deployed Microsoft or Google accounts this could simplify user setup. Additionally, it supports two factor authentication (2FA) which adds an extra layer of security on top of a user’s email and password by requiring a one-time use password that is either texted to their mobile or in an authentication app on their phone.

Charts

For data visualisation, Chart.js[20], was chosen. It offers a variety of chart types, including stacked bar charts and pie charts, which are ideal for representing hospital bed availability effectively. One of the key features implemented using Chart.js is the stacked bar chart, which provides a clear visualisation of bed availability across different wards within the hospital. This chart allows staff to quickly assess the availability of beds in each ward.

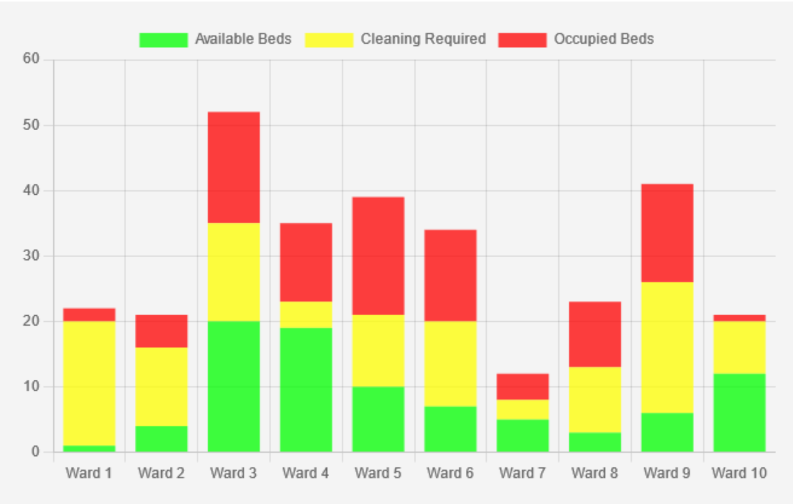

Figure 7: Stacked bar chart showing bed availability by ward

Another key feature used is pie charts. Pie charts are used to display the total bed availability for the whole hospital. This data could be useful to view the overall availability in the hospital.

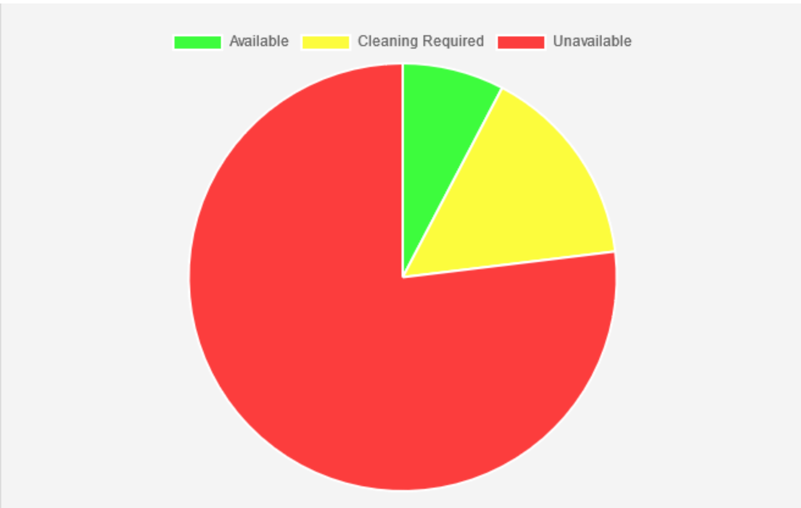

Figure 8: Pie chart showing total bed availability for the hospital

Both the stacked bar chart and the pie chart share some functionalities. By displaying the number of beds requiring cleaning it could help focus cleaning efforts to get more beds available. Additionally, a useful feature provided by chart.js is the ability to toggle the data being displayed by clicking on the legend. For example, if someone selects “Available” on the legend the chart will change to display only “Cleaning Required” and “Unavailable”. This effectively would show the ratio of unavailable beds to cleaning required.

Unfortunately, chart.js is not directly compatible with vue.js. As a result the use of two node packages, the chartjs package and vue-chartjs [21] package. vue-chartjs is a wrapper for chart.js which allows easy implementation of chart.js in the vue project. The figure below shows all the code required to add the stacked bar chart to the dashboard (excluding data).

```
<template>
    <cv-tile id="wardAvailability">
        <Bar id="stackedBarChart" :options="chartOptions" :data="chartData" />
    </cv-tile>
</template>

<script>
import { Bar } from "vue-chartjs";
import {
    Chart as ChartJS,
    Title,
    Tooltip,
    Legend,
    BarElement,
    CategoryScale,
    LinearScale
} from "chart.js";

ChartJS.register(
    Title,
    Tooltip,
    Legend,
    BarElement,
    CategoryScale,
    LinearScale
);
```

Figure 9: Code snippet demonstrating chart implementation

The minimal amount of code required made adding charts to the frontend quick and easy to update. While it is possible to implement chart.js into the project without the vue-chartjs wrapper it would significantly complicate the individual custom components.

UI Component Library

For overall design consistency across the frontend pages the decision to use the Carbon Design System by IBM [22] was made. This library was chosen as it contains large user interfaces which was a key consideration given that users of the software would likely be wearing gloves at times. Carbon is also designed to be accessible by integrating some browser accessibility features such as keyboard shortcuts for navigating some UI components.

Hierarchical Sitemap

The Hierarchical sitemap (Figure 10) shows the hierarchical structure of the frontend site.

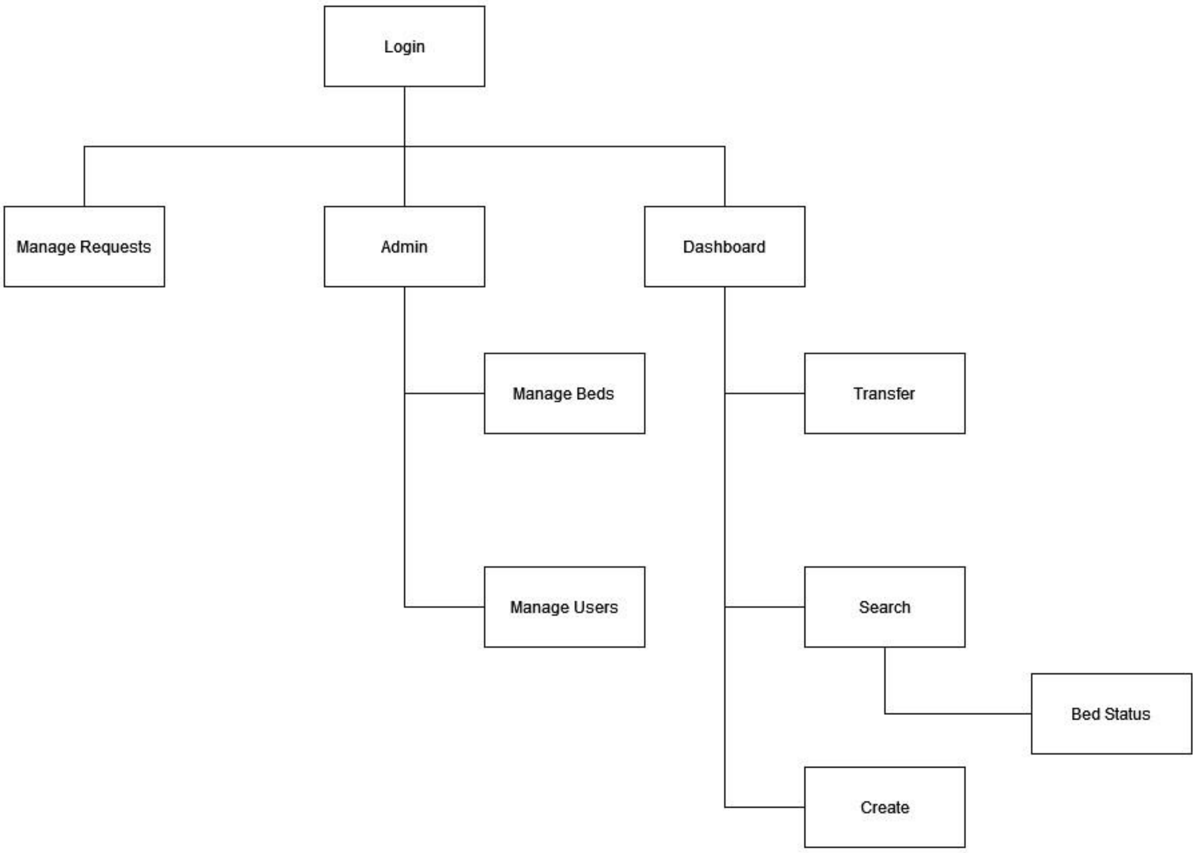

Figure 10: Hierarchical Sitemap

4.2 Backend Design

Language and Framework

For the backend Nest.js and TypeScript was used. The decision for this was due to the backend being primarily for create, read, update and delete (CRUD) operations. While TypeScript is not exclusively an object-oriented language it supports many object-oriented concepts such as class, interfaces and inheritance. This will allow objects such as bed, patients, wards etc to be modelled as classes. Nest.js advertises three of my most important backend non-functional requirements. Nest.js advertises itself as a modern framework for building efficient, reliable and scalable applications [23].

Additionally, the integration of OpenAPI Swagger through the "@nest/swagger" node package significantly enhances the development process. This tool facilitates the automatic generation of API documentation, which is useful for the development process. It offers a convenient and visual way to understand and test the API endpoints without the need for external tools such as Postman, streamlining both development and testing. As seen in figure 11, the generated documentation shows information such as the endpoint, parameters, and expected response.

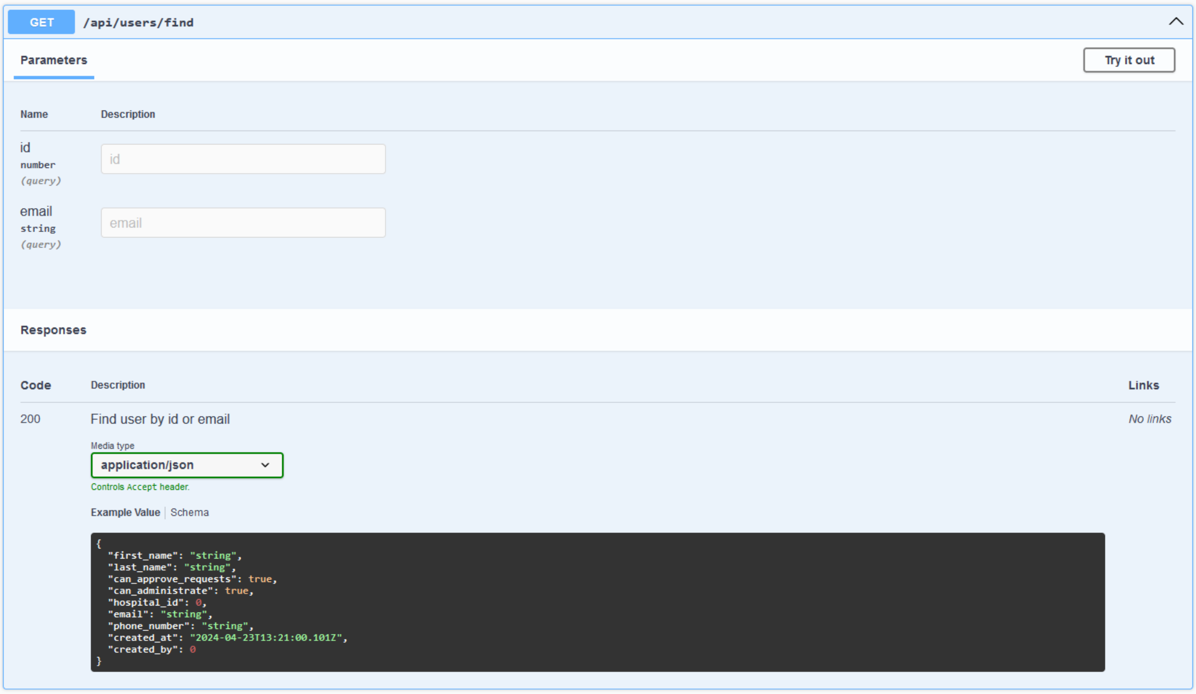

Figure 11: OpenAPI Swagger

MySQL

MySQL was chosen for the database for one key reason which is its ability to be deployed anywhere. MySQL can easily run on commodity consumer grade hardware or using a service such as Amazon Web Services, Microsoft Azure, Google Cloud Platform etc. This can give a hospital or hospital group complete control over where their users data is stored.

4.3 Continuous Integration/Continuous Delivery (CI/CD)

GitHub Actions was chosen for CI/CD for one main reason. As GitHub was already being used for source control it was quick to integrate the repository with GitHub actions. Additionally, SonarCloud which is used for testing also integrates well with GitHub actions. Therefore, by using GitHub Actions with GitHub and SonarCloud a quick and reliable pipeline could be created.

4.4 SonarCloud

Code analysis was implemented using SonarCloud. SonarCloud is a tool that scans your code repository for various issues that can affect the security, reliability and maintainability of the project. SonarCloud was chosen over its competitors because of its comprehensive scan, fast analysis, good integration with GitHub Actions, and free pricing for public repositories.

SonarCloud scans for four types of rules

- Code smell (maintainability domain)
- Bug (reliability domain)
- Vulnerability (security domain)
- Security hotspot (security domain)

And 4 different severities for each

- Blocker (A bug with a high probability of effecting the application)
- Critical (A bug with a low probability of effecting the application)
- Major (Lack of quality which can greatly affect developer productivity)
- Minor (Lack of quality which can slightly affect developer productivity)

Rule and severities adapted from [24] and severities description adapted from [25]

This analysis is trigged in the GitHub action when a pull request is created as seen in Figure 12. and the configuration of the scan can be modified on the SonarCloud website.

```
- name: SonarCloud Scan
  uses: SonarSource/sonarcloud-github-action@master
  env:
    GITHUB_TOKEN: ${{ secrets.GITHUB_TOKEN }}
    SONAR_TOKEN: ${{ secrets.SONAR_TOKEN }}
```

Figure 12: GitHub action code snippet

If there are too many code smells or any bugs, vulnerabilities or security hotspots a branch protection is setup to prevent the code with issues from getting merged into master.

4.5 Unit & Integration Testing

Unit Testing

Unit testing will be implemented on the backend and frontend. This tests individual components such as a single method or function and expects a particular result for a specific input. In the case of the frontend unit testing using the node package `@vue/test-utils`, it also can test entire components or views e.g. a table or dashboard with all the underlying methods meanwhile `jest` will be used for backend testing.

Integration Testing

Integration testing also known as end-to-end testing (E2E) will also be implemented on the backend and to a lesser extent on the frontend. Integration tests works by running the code and interacting with it similar to a user or application. For the backend this will mean testing endpoints and expecting specific results. On the frontend, using the node package `cypress`, we simulate a browser and mouse / keyboard interactions. The reason why integration testing will be used less on the frontend is for three reasons:

- Frontend unit testing overlapping: The package used the unit testing the frontend Vue.js application can yield very similar results to integration testing while providing more logs.
- Increased development time: Developing extensive unit and integration tests can take a long time and maintaining them can be difficult.
- Long runtime: Execution time for E2E tests can take a long time as it must build and run the software, then emulate a browser before finally running the tests.

Chapter 5

5.1 Results

User Interface

The final user interface that was developed achieved many of the initial goals during planning. Many pages such as the login page in figure 13 are similar to the prototypes made during planning.

Figure 13: User login screen

The dashboard page seen in Figure changed significantly compared to the prototypes. The final result is clearer and features a wider variety of charts including a stacked bar chart, horizontal bar chart, pie chart, and line chart. This combination of charts shows more useful information to users than the original prototype without compromising the usability of the dashboard.

Figure 14: A dynamic dashboard tool that depicts the demand-supply of beds

The bed page in Figure 15, successfully integrated QR codes with both an image of the QR code displayed and a button to download the QR code. It features large buttons with clear label indications making it easy for non-technical users to understand the functionality.

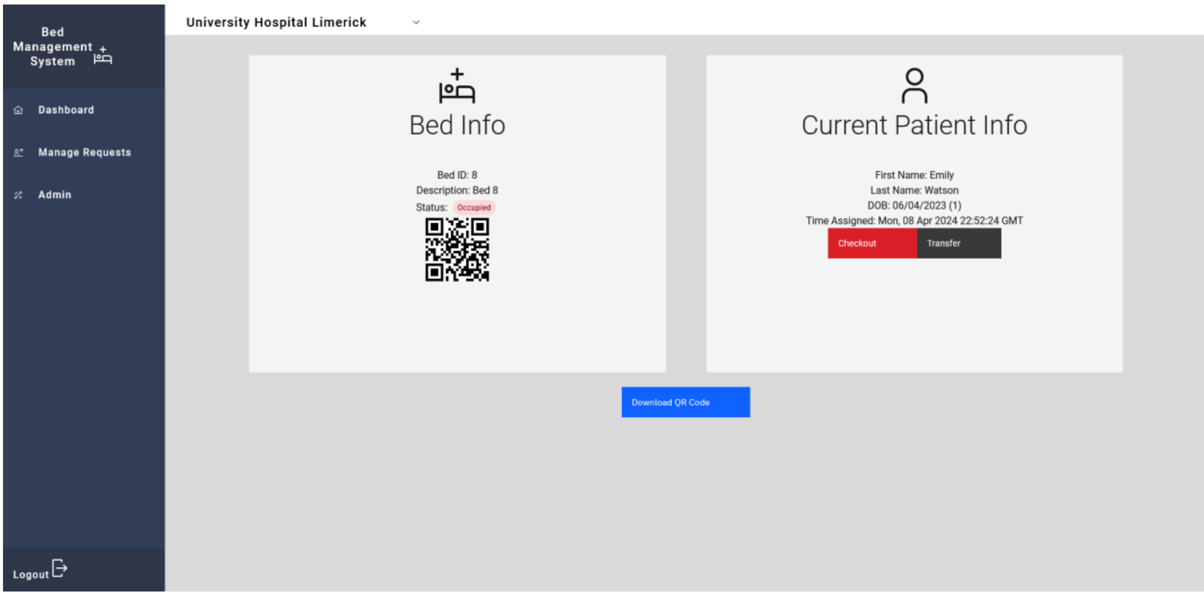

Figure 15: Bed and patient information page

The search page in Figure 16, features a search bar at the top allowing users to search for a ward, patient or bed. Either a patient's name is displayed next to a bed when it is occupied or a tag saying either "Available" or "Cleaning Required" is displayed. From this table users can quickly navigate to more detailed information including the ability to transfer a patient from that specific bed.

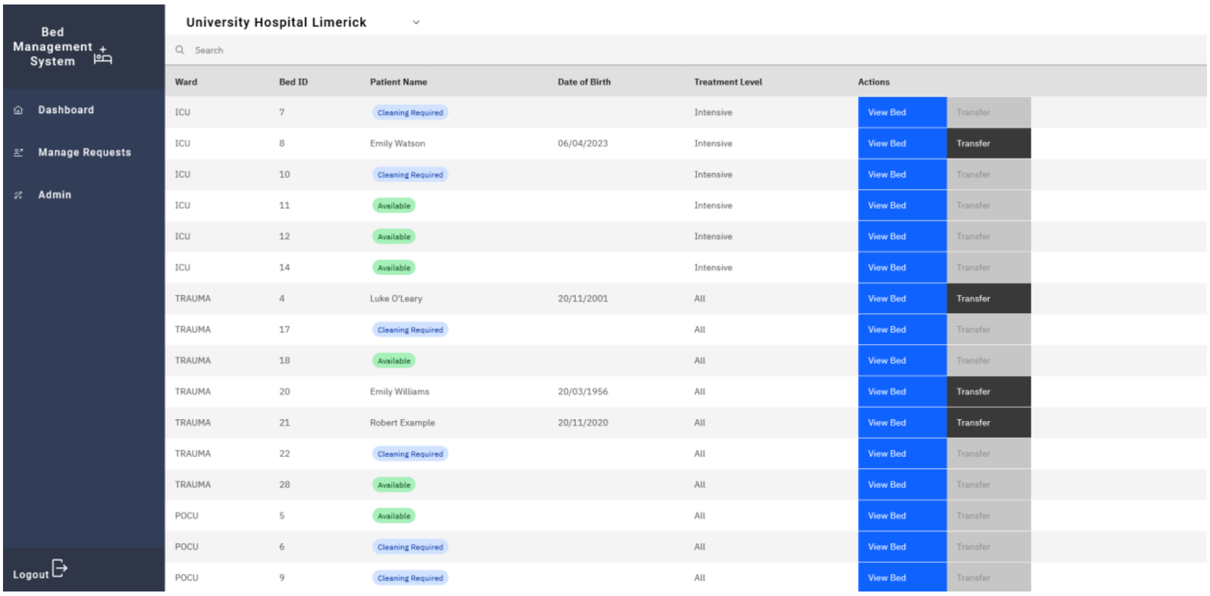

Figure 16: Searchable patient and user table screen

The admissions page seen in Figure 17, is similar to the prototype. Once a patient's information is entered all available beds, given the patient criteria (age and gender), are displayed. This functionality enabled quick patient-bed assignments which reduces the administration burden on healthcare staff compared to traditional methods.

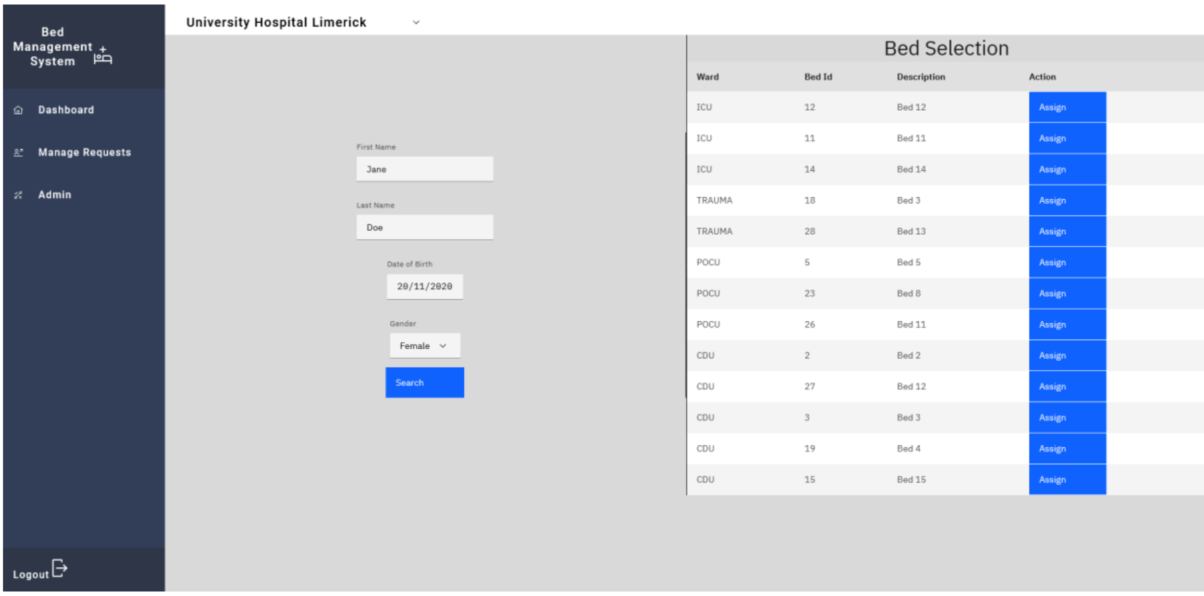

Figure 17: Patient admission screen

The transfer page seen in Figure 18, allows a user to efficiently request patient transfers. The patient may be transferred to another bed in their own hospital, or using the dropdown menu located above the table, a bed in another hospital. The table displays available beds that meet the patients' criteria (age and gender).

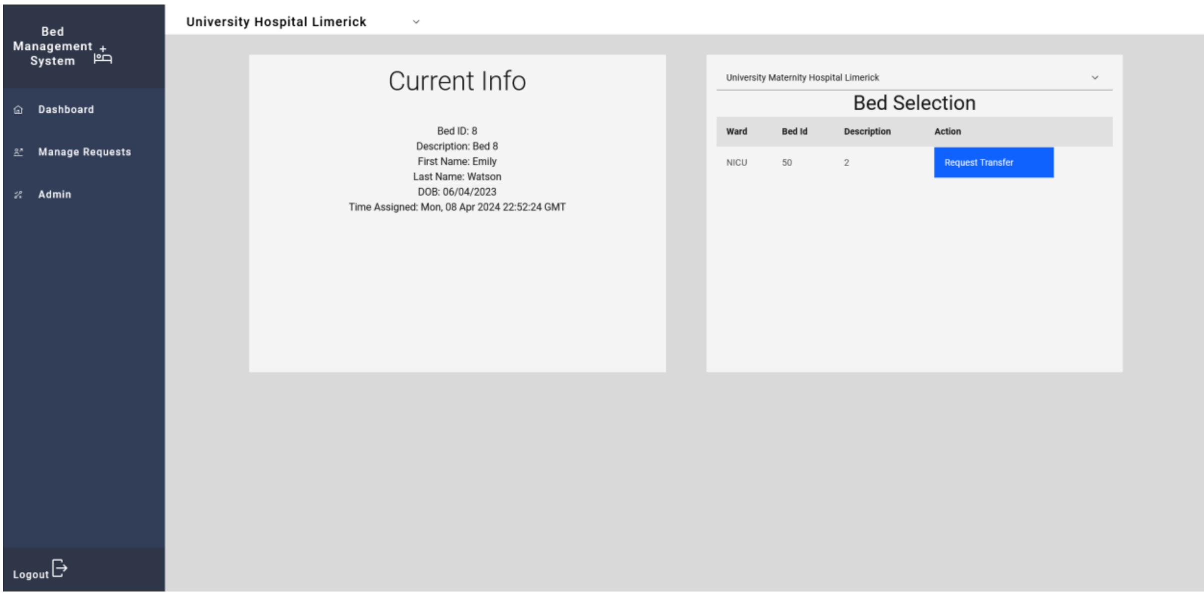

Figure 18: Screen to request patient transfers

The manager requests page seen in Figure 19. This function is visible only to bed managers at the selected hospital. The table (Figure 19) displays all transfer requests for patients within the hospital and external hospital requests. The information displayed includes: the patients name, current bed, ward information and the requested bed and ward information, and the user that requested the transfer. Using the buttons in the table row the bed manager can approve or decline the transfer. If a transfer is declined then it is removed from the table.

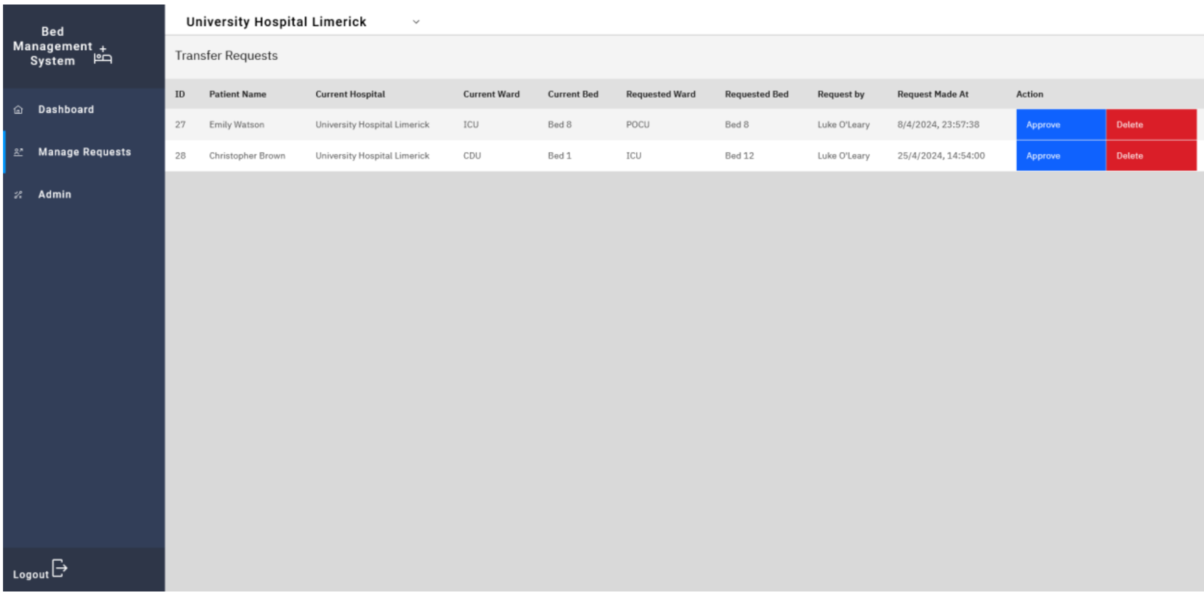

Figure 19: Screen to allow bed managers manage transfer requests

The manage beds page seen in Figure 20, is available to system administrators. It facilitates the viewing, creation, and deletion of wards and beds. When first opened it displays all wards, a user can then “open” a ward to edit all the beds within the ward. Dropdown selectors are used for gender and treatment level to eliminate the possibility of typos and keep the entries uniform.

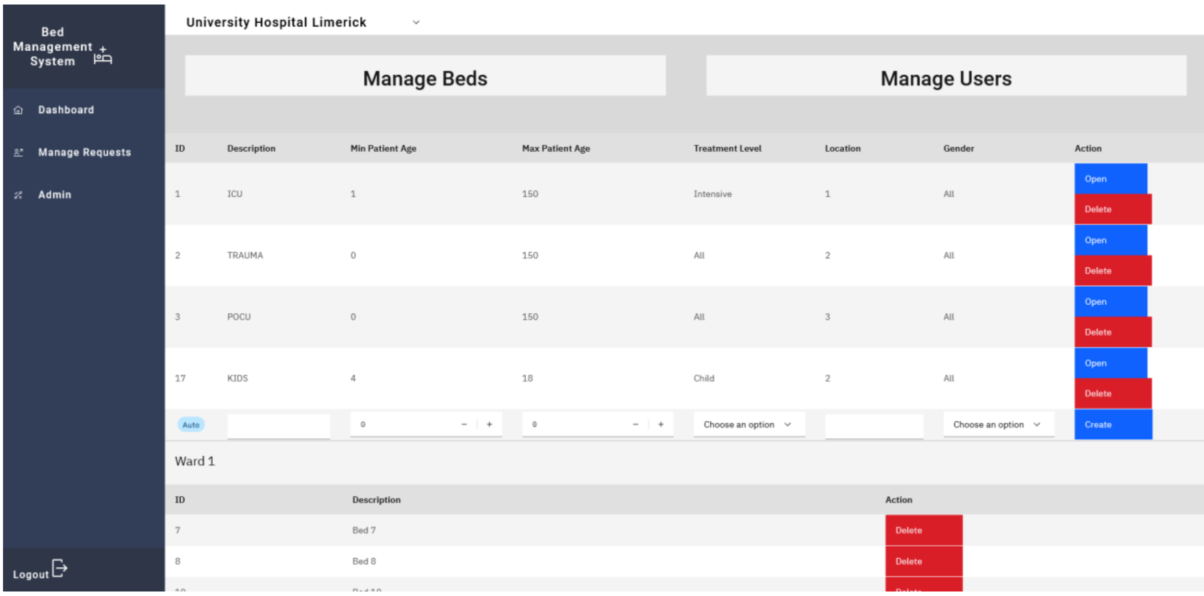

Figure 20: Screen to enable system administrators create or delete beds and wards

The manage users page seen in Figure 21, is only accessible by system administrators. It facilitates the viewing, creation, and deletion of user accounts. System administrators are blocked from deleting their own accounts to prevent all system administrator accounts from being deleted. After users enter details, they may select ‘create’. A popup appears enabling the user to enter a temporary password for the newly created account, seen in Figure 22.

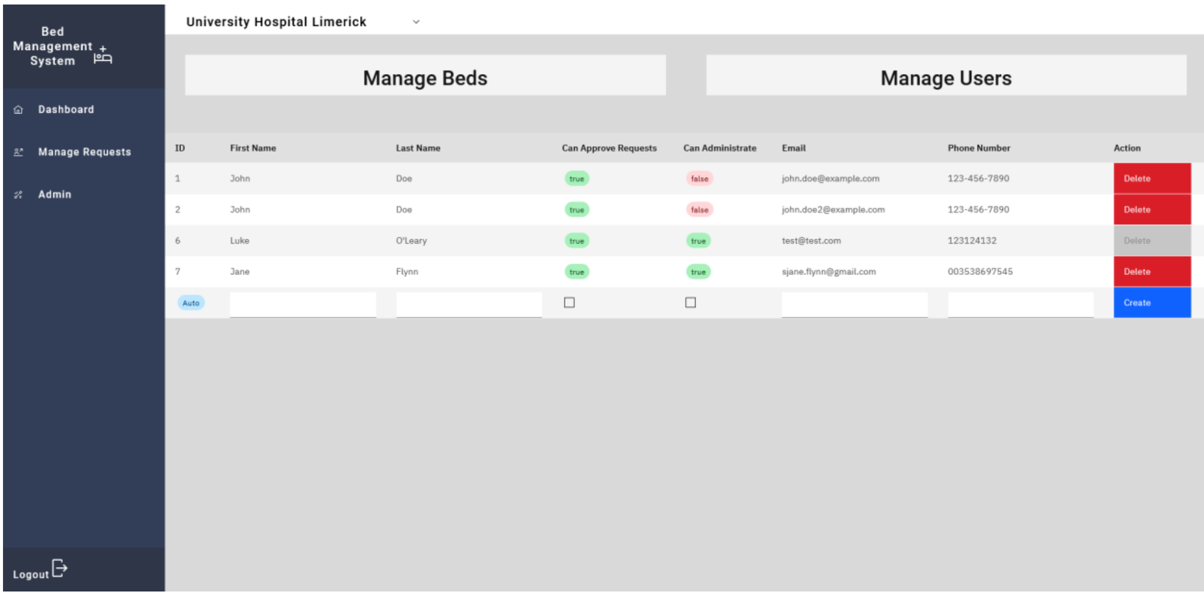

Figure 21: Screen to enable system administrators create or delete users

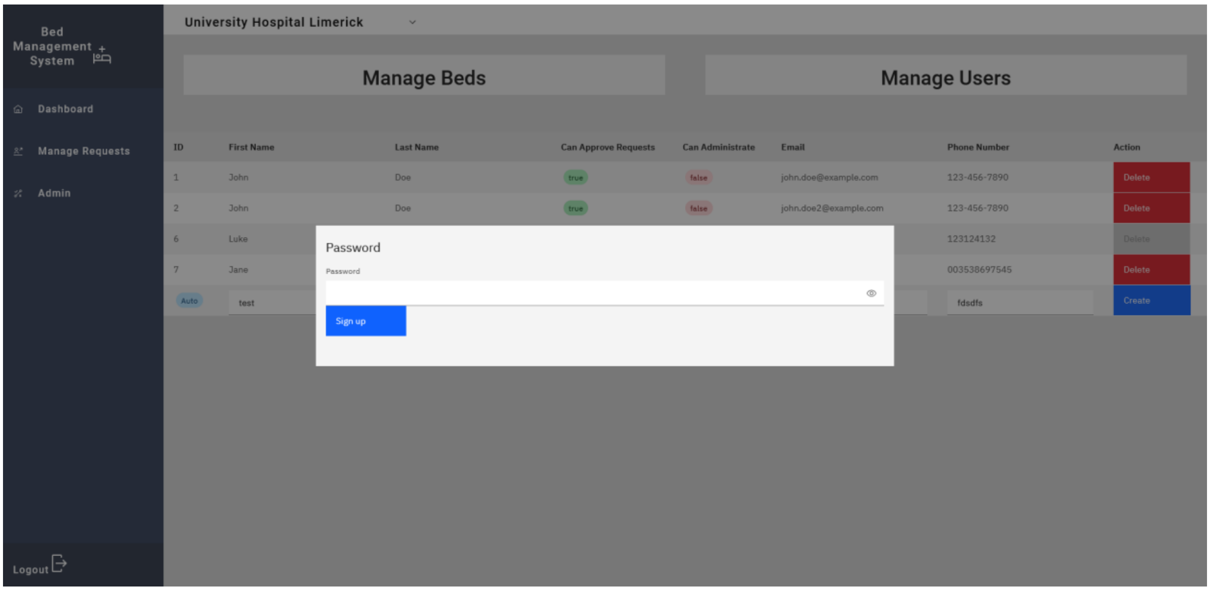

Figure 22: Popup requesting temporary user password for newly created account

Future work

While the software is operational, there are several areas that could be improved further.

- Mobile Compatibility

  The website works on small mobile devices such as smartphones. Mobile devices are limited however to basic functions including:

  - Search for patient or bed.
  - Update bed status.

  Optimising the website for devices smaller than 8 inches is difficult due to information tables which are relatively large. Scaling the table to smaller devices results in difficulties with navigation.

ID	Patient Name	Current Hospital	Current Ward	Current Bed	Requested Ward	Requested Bed	Request by	Request Made At	Action	
27	Emily Watson	University Hospital Limerick	ICU	Bed 8	POCU	Bed 8	Luke O'Leary	8/4/2024, 23:57:38	Approve	Delete
28	Christopher Brown	University Hospital Limerick	CDU	Bed 1	ICU	Bed 12	Luke O'Leary	25/4/2024, 14:54:00	Approve	Delete

Figure 23: Example of table used within the application

For example, reducing the size of the table in Figure 23 would make it unreadable. Using overflow, where the user must horizontally scroll, is not a user-friendly experience.

- Patients in transit

  Currently there is no option on the frontend to mark a patient as being 'in transit' from one location to another. Adding this feature could improve the real time location tracking of a patient. This functionality exists on the backend of the system but was not integrated to the frontend due to time constraints.

- Vue 3

  During development of the system the frontend framework (Vue 2), became deprecated (December 31st, 2023 [26]). Upgrading to Vue 3, prior to any additional frontend development, is considered important for both security and reliability. The changes between Vue 2 and Vue 3 make Vue 2 applications incompatible without refactoring.

- Readmission of same patient

  If a patient is discharged and later returns to the hospital again it is currently necessary to create a "new" patient admission. Adding a feature that enabled a re-admission could reduce administrative burdens further. This feature would also allow better tracking of the patient through digital records that indicate where the patient was previously treated, and the healthcare staff associated with their care. This would enable healthcare providers to communicate effectively across multiple teams and environments.

- Infection tracking

  The location of a patient in terms of their bed and ward is currently stored within the database. Using this data, it would be possible to implement a level of contact tracing. This could stop the spread of infectious diseases through efficient quarantining procedures. In addition, functionality could improve the treatment of patients who in close proximity to the infected individual as they can be quickly identified and receive preventative medical treatment. Many hospitals already track infectious diseases within their hospitals but this functionality is a practical additional level of tracking [13].

5.2 Discussion

The proposed web-based hospital bed management system is positioned to improve how hospital beds are managed. Although current literature on Bed Management (BM) is limited and offers inconclusive results, this project aims to improve the overall efficiency in bed management by enhancing communication. This lack of literature makes it difficult to predict the effectiveness of proposed models and systems [27], which underlines a need for more studies and literature around the topic. Despite limited evidence on how this system can affect bed management it is likely that it would be an improvement.

As previously discussed, bed management is complex. A level of this complexity is finding a bed that meets the patient's needs. A computer system, when given the correct information, can perform this task more efficiently. The role of a bed manager which is typically performed by a nurse, is complex and stressful [1]. A digital system has the potential to alleviate some of this stress and complexity for staff.

Another key area of contention is communication around bed management. Tasks as simple as finding a bed or as complex as transferring a patient to a different hospital can require high levels of communication to ensure patients’ needs are met. To efficiently manage these tasks a high level of communication and teamwork is required [28]. Again, this is an issue that can be aided using a software system specifically designed to manage beds.

In addition to the issues that have always existed, bed management has expanded considerably in the last decade which shows a need for any change to be scalable [1]. While there has been systems, models and architectures put in place to deal with some of this expansion, it has not always met the demand.

5.3 Conclusion

In conclusion the goal of this research is to enhance the management of hospital beds using a web-based system. This project is in response to the complex nature of bed management currently in hospitals where a range of specialized knowledge, training and structured management systems are involved. To address these complexities, this project aims to improve the allocation, utilization and overall efficiency of hospitals beds.

The system leverages Vue.js for frontend development, Nest.js for backend development, Firebase for authentication and MySQL for database management to meet the requirements identified during literature review. These solutions were chosen for a combination of their reliability, scalability and maintainability.

The expected outcomes of this system include an overall improvement in the efficiency of bed allocation and management which has the potential of reducing patient wait times, increase patient throughput and assist with overcrowding [1,2,7]. This work will alleviate the stress of navigating and understanding the complexities of bed management for hospital staff [1]. Furthermore, by streamlining resource management and patient flow within hospitals, patients will have better access to timely care which can reduce mortality and morbidity [29].

As previously identified there is a lack of literature surrounding bed management [27], to develop the system and understand the potential impact of bed management on patient care and hospital efficiency, more research is required. Information such as the cyber-security practices specific hospitals follow could be useful including: the type of user authentication used and what external/internal systems this bed management system may be required to integrate with.

In conclusion, this project is a software system to address a real-world problem, hospital bed management. By addressing the complexities and challenges identified during literature review and integrating modern technology solutions, this project has the potential to make a real impact to the lives of patients and medical professionals. The next steps will involve further research and development to create a system capable of fulfilling the goal.

References

1. Proudlove N, Boaden R, Jorgensen J. Developing bed managers: the why and the how. J Nurs Manag. 2007 Jan;15[1]:34–42.

2. Zepeda-Lugo C, Tlapa D, Baez-Lopez Y, Limon-Romero J, Ontiveros S, Perez-Sanchez A, et al. Assessing the Impact of Lean Healthcare on Inpatient Care: A Systematic Review. IJERPH. 2020 Aug 4;17[15]:5609.

3. St James's Hospital. St James's Hospital. [cited 2023 Dec 18]. Bed Management. Available from: https://www.stjames.ie/services/bedmanagement/

4. Bed management [Internet]. [cited 2023 Dec 29]. Available from: https://www.theaccessgroup.com/en-gb/health-social-care/software/bed-management/

5. Hospital Management Software | Hospital Information Management System (HIMS) | SoftClinic GenX [Internet]. 2020 [cited 2023 Dec 29]. Available from: https://www.softclinicsoftware.com/hospital-information-system-his/

6. Systems AD. Bed Management Software | Advanced Data Systems [Internet]. [cited 2023 Dec 29]. Available from: https://www.adsc.com/bed-management-software

7. Proudlove NC. Can good bed management solve the overcrowding in accident and emergency departments? Emergency Medicine Journal. 2003 Mar 1;20[2]:149–55.

8. Sharma S, Rani R. Nurse-to-patient ratio and nurse staffing norms for hospitals in India: A critical analysis of national benchmarks. J Family Med Prim Care. 2020;9[6]:2631.

9. Nursing (MSN), School of Nursing and Midwifery, Guilan University of Medical Sciences, Rasht, Iran., Pourbayram N, Chehrzad M, Assistant Professor, Department of Nursing, School of Nursing and Midwifery, Guilan University of Medical Sciences, Rasht, Iran., Karkhah S, Nursing (MSN), School of Nursing and Midwifery, Guilan University of Medical Sciences, Rasht, Iran., et al. Compliance With Safety in Nursing Care, Equipment, and Physical Setting in Pediatric Wards. JHNM. 2023 Sep 1;33[4]:278–87.

10. Bhosale KS, Nenova M, Iliev G. A study of cyber attacks: In the healthcare sector. In: 2021 Sixth Junior Conference on Lighting (Lighting) [Internet]. Gabrovo, Bulgaria: IEEE; 2021 [cited 2023 Dec 31]. p. 1–6. Available from: https://ieeexplore.ieee.org/document/9598947/

11. Packetlabs [Internet]. 2023 [cited 2023 Dec 31]. 239 Cybersecurity Statistics (2023). Available from: https://www.packetlabs.net/posts/239-cybersecurity-statistics-2023/

12. Andersen P, Lindgaard AM, Prgomet M, Creswick N, Westbrook JI. Mobile and Fixed Computer Use by Doctors and Nurses on Hospital Wards: Multi-method Study on the

Relationships Between Clinician Role, Clinical Task, and Device Choice. J Med Internet Res. 2009 Aug 4;11[3]:e32.

13. CIDR - Health Protection Surveillance Centre [Internet]. [cited 2023 Dec 18]. Available from: https://www.hpsc.ie/cidr/

14. Vitro Software - Clinical Data Management & Electronic Medical Records (EMR) for Hospitals [Internet]. [cited 2023 Dec 30]. Available from: https://www.vitrosoftware.com/

15. Vitro Software Holdings DAC [Internet]. [cited 2023 Dec 29]. Cork University Hospital. Available from: https://www.vitrosoftware.com/our-clients/healthcare-clients-list/cork-university-hospital

16. Xiaopu Ma, Ruixuan Li, Zhengding Lu, Jianfeng Lu, Meng Dong. Specifying and enforcing the principle of least privilege in role-based access control. 2011 Mar 29;

17. Sun X, Plocher T, Qu W. An Empirical Study on the Smallest Comfortable Button/Icon Size on Touch Screen. In: Aykin N, editor. Usability and Internationalization HCI and Culture [Internet]. Berlin, Heidelberg: Springer Berlin Heidelberg; 2007 [cited 2024 Apr 22]. p. 615–21. [Hutchison D, Kanade T, Kittler J, Kleinberg JM, Mattern F, Mitchell JC, et al., editors. Lecture Notes in Computer Science; vol. 4559]. Available from: http://link.springer.com/10.1007/978-3-540-73287-7_71

18. Cincović J, Punt M. Comparison: Angular vs. React vs. Vue. Which framework is the best choice?

19. Saks E. JavaScript frameworks: Angular vs React vs Vue.

20. Chart.js [Internet]. [cited 2024 Mar 21]. Available from: https://www.chartjs.org/

21. Home | 📈 vue-chartjs [Internet]. [cited 2024 Mar 21]. Available from: https://vue-chartjs.org/

22. Carbon Design System [Internet]. [cited 2024 Mar 21]. Available from: https://www.carbondesignsystem.com

23. NestJS - A progressive Node.js framework [Internet]. [cited 2023 Dec 31]. NestJS - A progressive Node.js framework. Available from: https://nestjs.com

24. Puspaningrum A, Anis Al Hilmi M, Darsih ., Mustamiin M, Ilham Ginanjar M. Vulnerable Source Code Detection Using Sonarcloud Code Analysis: In: Proceedings of the 5th International Conference on Applied Science and Technology on Engineering Science [Internet]. Bandung, Indonesia: SCITEPRESS - Science and Technology Publications; 2022 [cited 2024 Apr 22]. p. 683–7. Available from: https://www.scitepress.org/DigitalLibrary/Link.aspx?doi=10.5220/0011862600003575

25. Rules & SonarCloud [Internet]. [cited 2023 Dec 31]. Available from: https://docs.sonarsource.com/sonarcloud/digging-deeper/rules/

26. Vue 2 is Approaching End Of Life | The Vue Point [Internet]. [cited 2024 Apr 25]. Available from: https://blog.vuejs.org/posts/vue-2-eol

27. Novati R, Papalia R, Peano L. Effectiveness of an hospital bed management model: results of four years of follow-up. annali di igiene medicina preventiva e di comunità. 2017 Apr 30;[3]:189–96.

28. Tortorella F, Ukanowicz D, Douglas-Ntagha P, Ray R, Triller M. Improving Bed Turnover Time With a Bed Management System. JONA: The Journal of Nursing Administration. 2013 Jan;43[1]:37–43.

29. Cooke D, Rohleder T, Rogers P. A dynamic model of the systemic causes for patient treatment delays in emergency departments. Journal of Modelling in Management. 2010 Nov 2;5[3]:287–301.