Bed management in digital health care

GERARD.O'CONNELL 19192118
Course: B.Sc. in Computer Systems
University of Limerick
Supervisors: Dr Meghana Kshirsagar, Dr Alison O Connor

Acknowledgements

This research was supervised by Dr Meghana Kshirsagar and Dr Alison O Connor, who gratefully acknowledged funding from the Science Foundation Ireland National Challenge Fund grant number 22/NCF/DR/11289.

I want to thank Dr Alison O Conor and Dr Meghana Kshirsaga for their support and direction with this project. I would also like to thank the director of nursing and all the staff at Croom hospital for their input and feedback, along with the wider University Hospital Limerick group staff for their support with data collection and, the information they provided on operational procedures in their respective hospitals.

Table of Contents

Table of Figures

1. Introduction

Ireland's Health Service Executive (HSE) is the national health service. In the mid-west of Ireland, there are six operational hospitals. Collectively, these hospitals are referred to as the University Hospital Limerick Group (UHLG). The current population in the catchment area is half a million people[1]. Each hospital in the ULHG operates at a specific level of medical care. University Hospital Limerick (UHL) is the main hospital operating in the mid-western region. UHL offers 24/7/365 accident, emergency and critical care services and more complex medical procedures. It currently has a capacity of 762.

University Maternity Hospital Limerick is the only maternity hospital in the Mid-West region with a capacity of 110. Croom Orthopaedic Hospital is a dedicated 44-unit hospital that performs elective orthopaedic services for children and adults. It also provides post-acute care, rheumatology, and pain management. Ennis and Nenagh have 50 and 52 in-patient beds, respectively. They provide in-patient and outpatient services but also have medical assessment units (MAUs). MAUs, also known as injury units, treat non-trauma. General practitioner (GP) referrals for such conditions include broken bones, minor burns, dislocations, and wounds. St John's Hospital is similar to Ennis and Nenagh in terms of the services provided, except it is run on a voluntary basis and is a registered charity. Due to that, it is governed differently and provides services to the HSE rather than being an HSE hospital.

Data shows[2] that UHL is chronically overcrowded. From 2018 to today, there has been a steady increase, in the number of patients accessing UHLs emergency department (ED). In December 2018, 51 people were recorded as present in the ED on a trolley. A trolley and hospital bed differ in two main ways, as a trolley is thinner and wider than a hospital bed and is designed for the transportation of patients [3]. In December 2023, the number of patients on trollies rose to 106. The figure of 106 is more than double that recorded for the next hospital on the overcrowding list; (Cork University Hospital [2]). In a National Planning Framework from 2017[4], UHLG outlined that, the Mid-West population was expected to grow from 385,000 to 800,000 by 2040 [5] However, in 2022, the population in the Mid-West region was measured to be 505,369. This indicates that the population growth rate in the Mid-West region has exceeded expectations. It is also noted that planning document [4] did not consider the additional population growth resulting from the war between Ukraine and Russia [6]. Using the central statics offices (CSO) formula for estimates[7]. The Mid-West region indicates the population is closer to 925,000, which is approximately 125,000 more than was forecast for 2017.

Clearly, a large population size means increased demand on hospital resources. While the population accessing hospital resources has increased significantly, hospital capacity has remained fairly stagnant [8] as the data from the department of health highlights in 2009 UHLG had 765 beds and in 2021 the number had climbed to just 807. The result is an insufficient number of hospital beds compared to the number of patients requiring healthcare services. Rather than turn patients away, emergency departments use trollies to cope with patient overflow. While this might alleviate pressure in the short term, it fails to address the wider issues of general capacity. The purpose of the emergency department is to provide critical care only. Many patients, once they have received critical care, still require additional healthcare services. Ideally these patients are transferred from the emergency department to a specialist area within the hospital (i.e. a specific ward). Overcrowding within the emergency department occurs when patients cannot flow out from the emergency department into the wider hospital department. In UHL this is compounded by the closure of the emergency departments at the Ennis and Nenagh hospital [9] forcing all emergency department traffic in the Mid-West region to UHL. At its core overcrowding is a result of demand exceeding supply and this is most clearly demonstrated by analysing the hospital bed capacity.

The current bed management system is a national system called the Integrated Patient Management System (IPMS), which was introduced in 2009 [10]. Since its inception the system has not been updated, nor is there any Application Programming Interface (API), This means IPMS cannot be linked to other software easily resulting in incompatibilities with data management. The result is a fragmented system that operates in a fragmented manner within a specific hospital environment rather than a group level. UHLG also has its own bed management software and patient management software. The bed management software is a copy of the patient management software with a four data points removed. It is overly complex for the average user. As such the system is used for data entry and by the administration side of the hospital and not utilised by the medical staff on the wards. The software also lacks data visualisation a feature that would help staff to see the situation in all wards and hospitals in the UHLG in real time.

1.1 Overview of the project

The project aims to provide interoperable bed management software. The current system is extremely limited, overly complex, and not user-friendly. One key issue is the lack of a ‘live’ system, which hinders communication within the UHLG. The current method of communication is conference calls, in which the state of play in each hospital is discussed alongside potential patient transfers. Patient transfers from UHL to satellite hospitals attempt to alleviate the load on UHL. Representatives from each hospital relays their bed capacity during the conference call. The obvious drawbacks of this method are for the potential for communication errors and no dynamic updates. The software proposed in this research will address both aspects of this problem with a viewable dashboard providing dynamic information all parties, and a reduction in time loss.

Another challenge that needs to be addressed is the system's ability to assign a single person to a bed. Currently, the UHLG bed management system has no functionality to add a patient to a bed in a waiting slot. This is a significant issue for critical wards, such as the Intensive Care Unit (ICU) and dialysis wards. Healthcare staff currently rely on whiteboards to identify whether a bed is in use and/or if another patient is waiting for a bed in that specific ward. This whiteboard is only visible in the nurse's office, and the data is not accessible to other healthcare staff inside the

facility or, indeed, the UHLG. The proposed software integrates both an occupying slot and a waiting slot, enabling each hospital to visualise the status of in-demand wards.

As hospitals are constantly evolving, the proposed software will have a custom administration dashboard. The admin dashboard enables senior nurses to add and remove beds, wards and/or other hospitals. The low no-code features and dependencies locking eliminate the requirement for software engineers. The web-based viewer results in a lightweight system that runs on an internal server making it suitable for many different devices. The microservices architecture makes it easy to link the proposed system with future software through an API toolkit. The overall results is a system that is suitable for HSE digital infrastructure and other significant benefits to HSE staff.

1.2 Digital health care: The big picture

The transition to fully digital health care will be a big step forward in health care and the experience of patients [11]. When a patient checks in, if they are new, a file is created if they are known their file is there. Every citizen with a personal public service number (PPS) is given an Individual Health Identifier (IHI) in Ireland. This unique number links to a patient management file where they are tracked, and all progress and notes are held. The patient management system is the central source that pulls everything together. Digital files will allow seamless transfer of information from multiple sources around healthcare to be centralised, and the loss of files or information due to a paper-based system can be eliminated. Bed management software will create a visual link for hospitals, ambulances, and air ambulances so they can view if beds are free or how many people are waiting in each hospital. It also gives management and ward nurses a view of their patients and makes transfers easy. Tracking patients will be done using the wrist strap patients currently have. Wrist straps will get radio-frequency identification (RFID) that is triggered as a patient goes through each door as doors have sensors in them that records each time a RFID tag passes through it and tracks the patient moves through the hospital, enabling a nurse to locate a patient that has gone missing or wandered off by looking up the room they last entered on the patient management system. While not as accurate as global positioning system. The RFID is a major cost saving and isn't reliant on always having signal [12].

With digital files along with smart hospital beds allowing for the collection of data on patients. Current beds on the market have built-in tablets to view the patients' files and sensors that enable things from monitoring heart rate and movement to sensors aiding healthcare staff in knowing when an intravenous drip (IV) is empty or not delivering the fluid to the patient. The beds also provide assisted driving to aid staff in moving patients around the hospital and climate-controlled mattress to help protect patient skin [13]. Doctors can schedule patients to go for tests and with the aid of the digital files and tracking, healthcare staff can see exactly what has been done, reducing communication errors and confusion for patients and staff. Centralised files will allow medical staff to get the relevant information and the full patient history, providing a clearer picture of the patient.

Technology like near-field communication and 5g are utilised in wards for communication at close range between IOT devices. Allowing for better patient care and data collection for use with using Artificial intelligence (AI) and machine learning (ML). Patients can be screened for things to prevent future complication or to catch health issues they may be unaware of [14]. The collection of data and availability of files will unlock telehealth and allow specialists to see and talk to

patients and control their treatment with confidence [15]. Allowing a specialist to see patients all over the county improves outcomes and reduces barriers for healthcare staff and patients.

1.3 Project Timeline

Bed managemtn system		Semester I				Semester II			
		1	2	3	4	1	2	3	4
Step 1: Identify requirments									
1.1	Identify UHL systems in place								
1.2	Gather requirment for BM								
1.3	Interview Staff								
Step 2: Software design									
2.1	Identify framework								
2.2	Conceptualise design								
2.3	App building								
2.4	Testing								
2.5	Staff testing								
Step 3: Reports and deadlines									
3.1	Presentation								
3.2	interim report								
3.3	Final draft report								
3.4	Demo Day								
3.5	FYP Report								
3.6	FYP Product								

Figure 1 Gantt chart

Key weeks:

Hospital Inspections: Week 3/4 Semester 1.

Presentations: Week 6 Semester 1, R@ise conference week 12 Semester 2.

Interim Report: Week 12 Semester 1.

Demo Day: Week 11 Semester 2.

FYP Product: Week 12 Semester 2.

FYP Report: Week 13 Semester 2.

1.4 Literature review

The UHLG has been suffering from overcrowding for a number of years [2]. Recent examples of how overcrowding has resulted in poor patient outcomes include patients who died after receiving inadequate care. On a number of occasions, the HSE has been forced to pay out compensation by the courts [16]. Five years before one patient died from sepsis[17], a report

outlined 50 recommendations to improve patient care at UHL. One recommendation included that the national policy on sepsis was not followed, and it called for greater staff training in sepsis care [18]. With overcrowding and understaffing, nurses' and doctors' time management skills come into practice. A study found that nurses use routinisation and prioritising for time management [19]. The study showed that ignoring the influence of 'others', the team and the organisation perpetuates a rather individualistic and self-critical time management perspective. This may lead to a failure to address problems in organising work and the coordination of care involving other healthcare workers [19].

One way to reduce pressure on medical staff is to digitalise bed management, which would lead to enhanced communication across teams within a specific hospital environment and, in the case of patient transfers, with external hospitals. A medical team in a hospital in Fortaleza, Brazil, demonstrated a reduced average patient stay from six days to two using a Kanban web-based application [20]. This digital technology advancement resulted in improved patient outcomes and a more effective work environment [21]. In the context of the HSE, implementing digital technologies like this could result in fewer financial payouts for unnecessary patient deaths.

In response to a question from Gino Kelly TD, the HSE provided a costing saying that in 2021 it cost the HSE €938 per day for a bed in a hospital. With a more effective system in place, reducing the need for even one bed would save the HSE €342,370 per year, going off the 2021 figures provided[22]. Financial incentives notwithstanding, patients who receive medical care in a timely manner have a longer expected life span and a better outcome[23].

The software stack and architecture used is an important factor. Hospital information systems must run 24-h a day and 365 days a year. The software stack must be reliable and deal with asynchronous transfers [24]. Using a monolithic software architecture has proven to be cumbersome, and many big tech companies that value uptime and scalability, such as Amazon's e-Commerce Platform, Spotify, and Netflix, have all transitioned to microservices architecture[25]. Such an architecture also prevents one issue from taking down the whole system, as each service runs autonomously with its business logic, database and API.

The advantages of microservices are that each application can be written in a different language with a different framework. It accelerates scalability with the services. It also improves fault isolation as each application is compartmentalised. It enables enhanced team productivity among engineers, allowing smaller teams to focus on a particular service. As new services are required or must be changed, it allows quicker deployment. Unlike monolithic architecture, which requires the entire system to be redeployed for updates, the microservices architecture allows for quick updates and changes as each service evolves and deploys independently of each other. It also allows services to spin up or down as required, saving server resources[26] with Kubernetes and Docker [25].

The software stack in healthcare faces regulatory oversight and standards as it handles sensitive patient data [27]. The EU's security standards concerning healthcare must also be followed [28]. Both sets of regulations cover the software life cycle and the security of software. Engineers with a security-first focus using the appropriate frameworks and tools remain in the regulations quite easily, as frameworks like Django have built-in session security, SQL injection protection and Cross site request forgery (CSRF) protection[29].

The software stack itself is made up of the front end and the back end. The front end is most associated with user interface/user experience (UI/UX). The front end focuses on the

application's client side, creating visually appealing and user-friendly interfaces. The most common programming languages associated with front-end development are hypertext markup language (HTML), cascading style sheets (CSS) and Javascript (JS). The back end is where the developers focus on the application's server side, linking the databases and business logic together and making all the information available to view by the front end via a template, in the case of Django, with its model view template structure[30]. The languages most associated with the backend are Java, Python, and PHP. The actual language used depends on the framework and what the application is to be used, as each language has its advantages and disadvantages[31].

The front end can be a more trivial decision regarding what programming languages and software to choose due to the application being displayed over web pages. Designing a UI/UX that is appealing and usable is still very complex. The back end is where the decisions get much harder concerning what software to use, as the emergency departments can generate huge amounts of data of varying types. The type of data databases can store is also limited, and accessing the database in real-time with no delay becomes an issue as the software stack grows and more applications are required to store and retrieve data from a database. SQL databases are excellent at sorting structured data [32]. It is not good at storing images or trying to store big data, which is a necessity for a hospital. The data generated through hospitals are images, videos, audio, text, metadata, logs and many other data types. This requires NoSQL databases for big data [33]. Within healthcare, the easiest solution is to use something like Amazon Snowflake as a database manager. The advantage of using a major database manager like Amazon or Microsoft is the accessibility of machine learning plugins[34] and the critical security and safety assurances when storing patients' records that you get from big companies with government contracts.

A NoSQL database such as Apache Casandra can be used for companies or projects provided the regulations [27] are followed, and healthcare security and procedures are adhered to [28]. Usually, with a microservices architecture using Django or Spring, the framework will dictate what is used for the business logic layer as they use model view controller (MVC) and model view template (MVT) structures. The framework provides the tools, and the developers are not building new and custom software for every detail. This has the distinct advantage of speed in development, safety, and security. A prime example is Django, which has an authentication system that automatically hashes a user password. When the user tries to log in, it checks the hashes to verify the user credentials. The password raw text is never stored in the database, and the hash is obfuscated [35]. When building custom software with no framework, work requires the developer to build functions to hash passwords and replays on them, never storing the raw text of a password, only the hash.

2 Existing systems within Croom

2.1 The current method of bed management

Croom currently uses two systems for bed management. They have bed management system software and a manual system that uses a whiteboard and magnets, as shown in Figure 1. Their manual system uses different colour magnets to signify different requirements or patient types. The board also lists the bed numbers and each ward's capacity.

Figure 2 Whiteboard used for bed management

The staff does not utilise the current digital system on the ward. This system comprises 16 different informational fields pertaining to each patient. It seems to replicate the functionality of a more comprehensive patient management system, which contains 20 fields of data per patient. This level of complexity and lack of relevant information would appear to be why staff prefer the whiteboards.

2.2 Current information technology systems

Croom staff connect to the internet via ethernet as no WI-Fi is available. A bandwidth issue across the hospital network severely limits internet download and upload speed.

The computers office staff use are intel nuc, or Lenovo think centres. They are extremely small form-factor computers, also known as tiny desktops. There are several downsides to using them. The lack of processing power is the one affecting the staff the most. As was demonstrated by staff, opening an Excel sheet takes an extended period of time. The issue with these small computers is that the hospital is kept at a warm temperature, and the office can be very warm. The tiny computers are not big enough to have powerful fans to cool the internal chips efficiently. This issue is clear on their computers, which have stronger processing power. The computers throttle down to avoid overheating.

The management level staff have laptops that are much more suited to do the job. On-site, they can dock the laptop into a hub and use it with monitors.

The system also uses Microsoft SharePoint as the database. It is unclear if the main database is backed up away from SharePoint or if everything is only stored on SharePoint. In any case, it is not recommended to use SharePoint as the main database even by Microsoft support agents [36] or even to use ShareAcess to access a web-based database and use SharePoint. In the documentation for SharePoint, Microsoft says to use SQL databases. All the documentation is for SharePoint server [37], not SharePoint 365. SharePoint 365 has no documentation relating to database usage with the service[38]. While it is technically possible to have the main database on SharePoint 365, The only evidence I could find with a positive reason to do it was superficial, and the negatives in the same article outweigh the positives [39].

3 Software

3.1 Architecture of the system

The proposed architecture is Django, using Python as the main programming language. Django is a powerful Python web-based framework. It follows the model—views—templates architectural pattern (MVT). The MVT is a well-used and known pattern in software development. It comprises the model that helps handle the database. The template is the presentation layer that handles the user interface, and the view executes the business logic and interacts with the model to update or get information from the database. It then renders the template the user interacts with.

The software follows the open-closed design principle. It is an object-orientated programming design principle where the code is open for extension but closed for modification. This would mean loose coupling, so the system components have minimum dependencies on each other, but it gives you high cohesion as the system has clear responsibilities. Microservices let the software interact with all different software systems through APIs. This makes it a perfect choice as it leaves it open to interact with systems that haven't been invented yet and allows it to interact with current systems with ease on top of its interoperability with current ML libraries. Opening up the opportunity for advanced data collection and use with AI.

3.2 Django Features and Integration

Django comes with a standard SQLite database. SQLite has a bandwidth issue of 2GB in testing, while that limit in the real world may never become a problem. There are other issues, such as a table not being able to have more than 264 rows and no table can have more than 32767 columns. PostgreSQL or MySQL databases would be used in production as they store data magnitudes greater than SQLite, and, in testing, it has not been possible to hit a bandwidth limitation for either database. Other SQL databases can be used. Only two SQL databases, MongoDB and Microsoft SQL, have been ruled out due to security issues[40] [41].

Sonarlint and Sonarcloud are applications from the sonar group. The software is used on the developer's end to ensure clean code free from what is known as code smells. Code smells can be the same code used multiple times in different parts of the program, making it harder to

maintain and possibly introducing a bug or flaw in multiple parts of the program. It also refers to the level of complexity within the code, as code can get very complex very quickly due to computer logic. It also prevents security issues by picking up bad or misused security flaws in the code, such as how HTTP requests are sent or received. As seen from the solarLint plugin below, it picks up on all sorts of faults in code, from simple comments to be deleted to more complex security issues and advice on how to fix them.

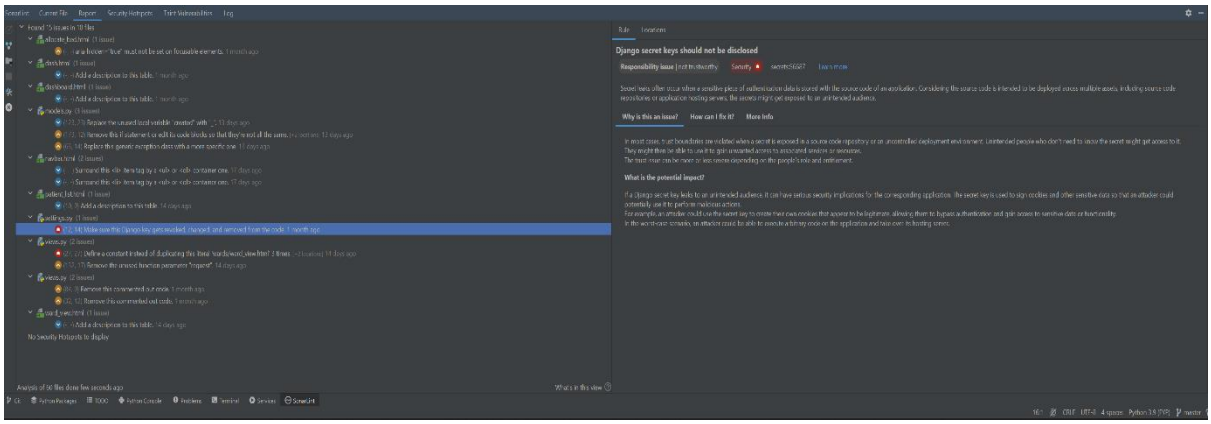

Figure 3 SolarLint Plugin

The Django framework also prevents software engineers from writing insecure or bad code. While it is still possible to write insecure or bad code, the Django framework is set up so that most insecure or bad code will cause an error. It also has built-in security features like authentication and secure HTTP requests. It can handle an array of security features, so there is no need to create security features from scratch, such as user and password authentication and validation. This reduces the risk of hackers breaching the system and stealing passwords and user credentials. It also has the added value of security getting updated with the framework. Security features will be turned on and hardcoded into the libraries by default. To turn anything off, developers must know what they are doing, removing human error in many cases.

For testing as Django runs in a virtual environment, A developer can spin up as many virtual machines or containers as needed and create hundreds or thousands of instances for stress testing the code and running tests in parallel. Because it is Python, its integration with scripts and automation is extremely easy. A bonus is its integration with machine learning applications. Django uses Python, and as most machine learning libraries are written in Python, they can be imported, and the data generated can be fed to machine learning applications. Machine learning can be applied to Django applications with little modification as the framework also supports all the major libraries used with machine learning, such as Pytorch and Keras. While that is currently out of the scope of this software, it is good for future integration into a digital healthcare system with automation and artificial intelligence. This software will integrate seamlessly.

3.2.1 Mobile Accessibility

Django having a web-based view. A mobile version of the software is available as cascading style sheets (CSS) are used to manipulate the web page. The CSS will resize images and content to suit

the device's screen size. It also takes into account the use of tablets. So, when a user logs on via a mobile device, the dashboard will be resized to fit that device. The biggest obstacle to mobile accessibility is the lack of Wi-Fi in hospitals. Despite this, mobile-friendly viewing has been integrated. As can be seen below, when the screen size is shrunk down, the menu at the top turns into a hamburger menu and the text and table change dynamically with the screen size.

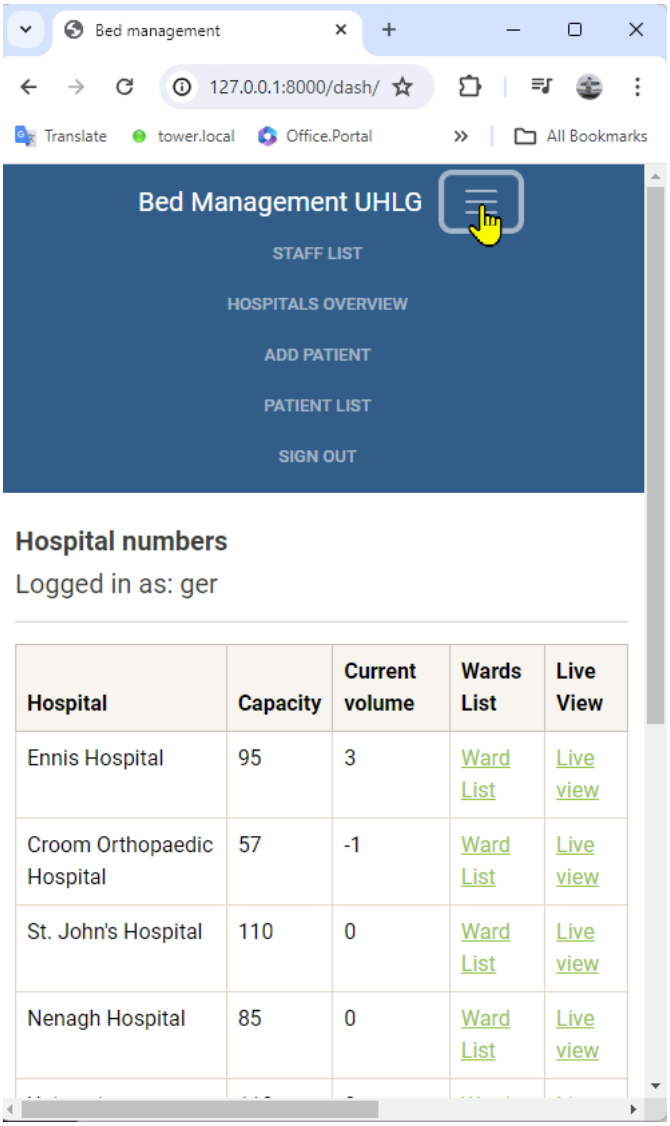

Figure 4 Mobile View

3.2.2 Dashboard

The dashboard is a web application that staff can view. In the nurse station, a monitor or TV can be set up to replace the whiteboard, and the dashboard can be displayed for them to see. The dashboard will be different depending on who is logged into the system. The dashboard provides a display for the staff of the wards, including their capacity and current volume, in a text format, along with an interactive pie chart that shows the capacity of each hospital in relation to each other in the Mid-West region. It also has a bar chart showing the current volume as a percentage compared to the hospital's capacity.

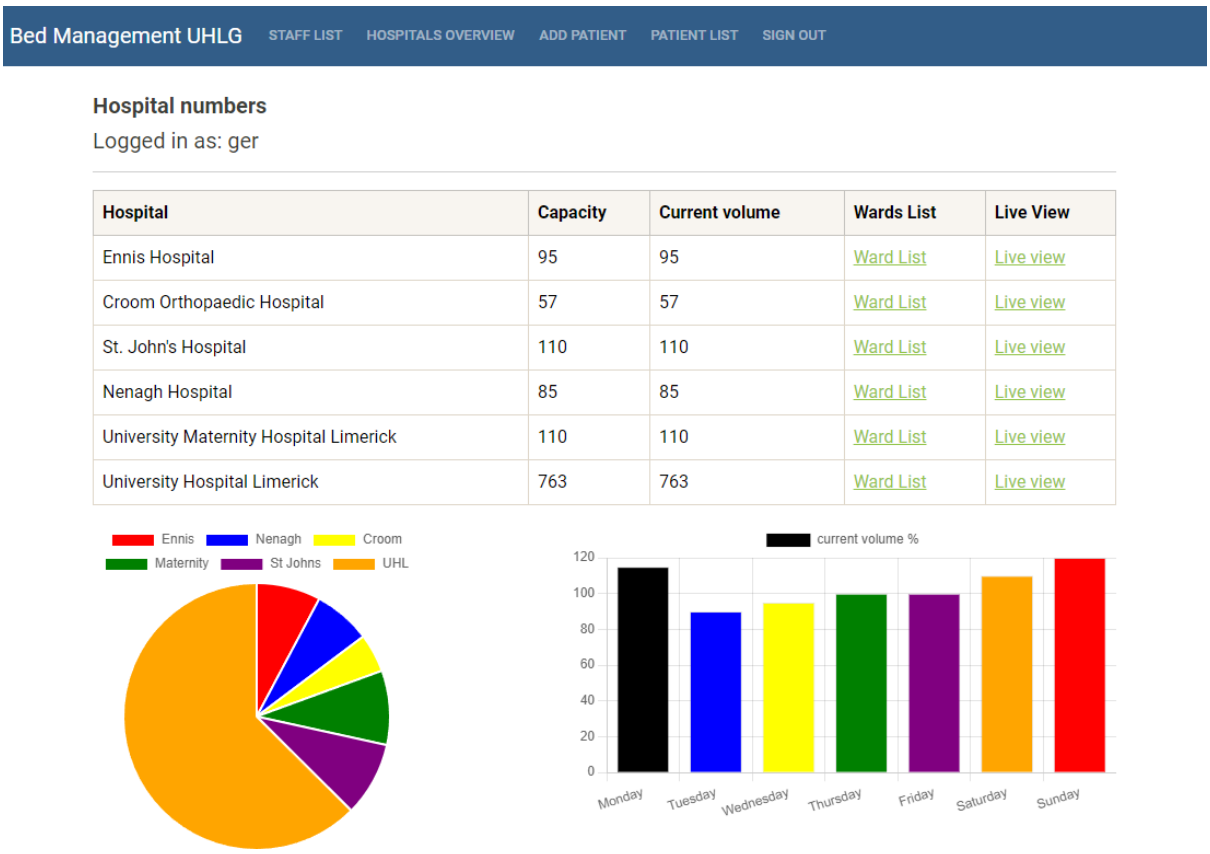

Figure 5 Dashboard

3.2.3 Application

In the application, when the nurse responsible for allocating beds uses the proposed software, they will be in the live view. Each hospital has a waiting section where patients waiting are listed. The nurse can click on their names, allocate them a bed, or set them in the waiting spot to take a designated bed.

The waiting section is populated by an API pulling the details from the patient management system. When patients are checked in, they are available to see in the waiting section. When someone is discharged, that space will be visible to the nurse in charge of allocating beds in the ward. It will eliminate the need for that nurse to ring the ward or to run to check the physical status of the beds. Freeing up their time and reducing stress.

4 Conclusion

In conclusion, With the research and interviews done. The future of fully digital health care is approaching extremely fast, and the benefits will outweigh any negatives. The transition may be costly upfront to put all the infrastructure in place and build out the software stacks needed. Huge improvements in patient care and patient outcomes can be made from the investment.

The proposed software does not solve all the problems, but it can make a difference. In emergency departments, a nurse has to spend one to five minutes ringing around or looking in

wards for beds available. If we can give them the software to view all the beds and the situation in each ward, we can speed up patient processing time and take some pressure off the nurse. The same goes for the dashboard info. By giving management dynamic data for their conference calls, we can reduce information and time loss in the meetings.

No one thing will fix overcrowding, but this software can make a small difference when paired with other software for digital health and other small improvements in alleviating overcrowding and waiting times. With the improvement in patient care and the true value people see and find in digital health care, it will start to snowball effect, and serious momentum will gather in the transition to a fully digital and integrated healthcare system.

5 References

[1] 'Population Usually Resident and Present in the State'. Accessed: Jan. 01, 2024. [Online]. Available: https://data.cso.ie/
[2] 'ED Trolley Watch/Ward Watch Figures below for 31st August 2016'. Accessed: Dec. 11, 2023. [Online]. Available: https://www.inmo.ie/Trolley_Ward_Watch
[3] T. Smith, 'Patient Stretchers vs Hospital Beds; What Is The Difference?', Felgains. Accessed: Apr. 28, 2024. [Online]. Available: https://www.felgains.com/blog/patient-stretchers-vs-hospital-beds-what-is-the-difference/
[4] '2040 national planniong framework UHLG'. Accessed: Apr. 16, 2024. [Online]. Available: https://www.npf.ie/wp-content/uploads/2017/09/0582-UL-Hospitals-Group.compressed.pdf
[5] 'Ireland 2040 National Planning Framework', UL Hospitals Group, Mar. 2017.
[6] S. Pollak, '104,000 Ukrainian arrivals since the beginning of the war', The Irish Times. Accessed: Apr. 28, 2024. [Online]. Available: https://www.irishtimes.com/ireland/2024/03/29/more-ukrainian-refugees-now-leaving-state-accommodation-than-entering-the-country/
[7] 'CSO Standard Report on Population and Migration Estimates'.
[8] 'open bed report 2021', Mar. 2021.
[9] 'Call to reverse Nenagh ED closure', Nenagh Guardian. Accessed: Apr. 28, 2024. [Online]. Available: https://www.nenaghguardian.ie/2022/03/23/call-to-reverse-nenagh-ed-closure/
[10] IMTtemp, '20-plus hospitals now operating IPMS', Irish Medical Times. Accessed: Dec. 19, 2023. [Online]. Available: https://www.imt.ie/news/20-plus-hospitals-now-operating-ipms-19-05-2010/
[11] C. J. Butcher and W. Hussain, 'Digital healthcare: the future', *Future Healthc J*, vol. 9, no. 2, pp. 113–117, Jul. 2022, doi: 10.7861/fhj.2022-0046.
[12] F. Ebrahimzadeh, E. Nabovati, M. R. Hasibian, and S. Eslami, 'Evaluation of the Effects of Radio-Frequency Identification Technology on Patient Tracking in Hospitals: A Systematic Review', *Journal of Patient Safety*, vol. 17, no. 8, p. e1157, Dec. 2021, doi: 10.1097/PTS.0000000000000446.
[13] 'Progress Smart Hospital Bed System', O'Flynn Medical. Accessed: Apr. 28, 2024. [Online]. Available: https://www.oflynnmedical.com/product/progress-smart-bed-system/

[14] Z. I. Attia *et al.*, 'Screening for cardiac contractile dysfunction using an artificial intelligence–enabled electrocardiogram', *Nat Med*, vol. 25, no. 1, pp. 70–74, Jan. 2019, doi: 10.1038/s41591-018-0240-2.
[15] C. Scott Kruse, P. Karem, K. Shifflett, L. Vegi, K. Ravi, and M. Brooks, 'Evaluating barriers to adopting telemedicine worldwide: A systematic review', *J Telemed Telecare*, vol. 24, no. 1, pp. 4–12, Jan. 2018, doi: 10.1177/1357633X16674087.
[16] 'Eve Cleary's family settle action as HSE "expresses regret" over young woman's death', BreakingNews.ie. Accessed: Apr. 22, 2024. [Online]. Available: https://www.breakingnews.ie/ireland/eve-clearys-family-settle-action-as-hse-expresses-regret-over-young-womans-death-1581882.html
[17] M. Sheehan, 'Schoolgirl Aoife Johnston, who died from meningitis, waited in A&E for 12 hours as sepsis went untreated', Independent.ie. Accessed: Dec. 10, 2023. [Online]. Available: https://www.independent.ie/irish-news/revealed-schoolgirl-aoife-johnston-who-died-from-meningitis-waited-in-aande-for-12-hours-as-sepsis-went-untreated/a345024656.html
[18] '"We hoped University Hospital Limerick would have learned lessons from our mother's death" – family of woman who died five years before Aoife Johnston', Irish Independent. Accessed: Apr. 22, 2024. [Online]. Available: https://www.independent.ie/news/we-hoped-university-hospital-limerick-would-have-learned-lessons-from-our-mothers-death-family-of-woman-who-died-five-years-before-aoife-johnston/a1760836247.html
[19] S. Waterworth, 'Time management strategies in nursing practice', *Journal of Advanced Nursing*, vol. 43, no. 5, pp. 432–440, 2003, doi: 10.1046/j.1365-2648.2003.02740.x.
[20] H. A. L. Rocha *et al.*, 'Bed management team with Kanban web-based application', *International Journal for Quality in Health Care*, vol. 30, no. 9, pp. 708–714, Nov. 2018, doi: 10.1093/intqhc/mzy108.
[21] N. Purdy, H. K. Spence Laschinger, J. Finegan, M. Kerr, and F. Olivera, 'Effects of work environments on nurse and patient outcomes', *Journal of Nursing Management*, vol. 18, no. 8, pp. 901–913, 2010, doi: 10.1111/j.1365-2834.2010.01172.x.
[22] S. Anderson, 'Re PQ 46122 21: To ask the Minister for Health the estimated full year cost in 2022 of adding 1,000 beds to the acute hospital network.', Oct. 21, 2021.
[23] J. P. Bunker, H. S. Frazier, and F. Mosteller, 'Improving Health: Measuring Effects of Medical Care', *The Milbank Quarterly*, vol. 72, no. 2, pp. 225–258, 1994, doi: 10.2307/3350295.
[24] N. Sakamoto, 'Availability of software services for a hospital information system', *International Journal of Medical Informatics*, vol. 49, no. 1, pp. 89–96, Mar. 1998, doi: 10.1016/S1386-5056(98)00015-X.
[25] D. Cremins, 'Kubernetes in the age of cloud native', University Of Limerick / Intel.
[26] Atlassian, '5 Advantages of Microservices [+ Disadvantages]', Atlassian. Accessed: Apr. 28, 2024. [Online]. Available: https://www.atlassian.com/microservices/cloud-computing/advantages-of-microservices
[27] 'ISO/TS 82304-2:2021(en), Health software — Part 2: Health and wellness apps—Quality and reliability'. Accessed: Apr. 29, 2024. [Online]. Available: https://www.iso.org/obp/ui/en/#iso:std:iso:ts:82304:-2:ed-1:v1:en
[28] 14:00-17:00, 'IEC 81001-5-1:2021 (Security IT healthcare)', ISO. Accessed: Apr. 29, 2024. [Online]. Available: https://www.iso.org/standard/76097.html
[29] 'Security in Django | Django documentation', Django Project. Accessed: Apr. 29, 2024. [Online]. Available: https://docs.djangoproject.com/en/5.0/topics/security/
[30] 'Django documentation | MVT', Django Project. Accessed: Apr. 29, 2024. [Online]. Available: https://docs.djangoproject.com/en/5.0/
[31] 'How to Choose the Right Programming Language for a New Project', BairesDev. Accessed: Apr. 29, 2024. [Online]. Available: https://www.bairesdev.com/blog/choose-right-programming-language/

[32] ‘SQL Data Types | DigitalOcean’. Accessed: Apr. 28, 2024. [Online]. Available: https://www.digitalocean.com/community/tutorials/sql-data-types
[33] ‘What Is Big Data?’ Accessed: Apr. 28, 2024. [Online]. Available: https://www.oracle.com/ie/big-data/what-is-big-data/
[34] ‘Use Snowflake as a data source to train ML models with Amazon SageMaker | AWS Machine Learning Blog’. Accessed: Apr. 29, 2024. [Online]. Available: https://aws.amazon.com/blogs/machine-learning/use-snowflake-as-a-data-source-to-train-ml-models-with-amazon-sagemaker/
[35] ‘Password management in Django | Django documentation’, Django Project. Accessed: Apr. 29, 2024. [Online]. Available: https://docs.djangoproject.com/en/5.0/topics/auth/passwords/
[36] ‘Using an Access database from SharePoint Online’. Accessed: Apr. 29, 2024. [Online]. Available: https://answers.microsoft.com/en-us/msoffice/forum/all/using-an-access-database-from-sharepoint-online/9f417a1c-5be5-4fa7-b879-5bdce91f829f
[37] SerdarSoysal, ‘Storage and SQL Server capacity planning and configuration (SharePoint Server) - SharePoint Server’. Accessed: Apr. 29, 2024. [Online]. Available: https://learn.microsoft.com/en-us/sharepoint/administration/storage-and-sql-server-capacity-planning-and-configuration
[38] maggierui, ‘Introduction to SharePoint and OneDrive in Microsoft 365 for administrators - SharePoint in Microsoft 365’. Accessed: Apr. 29, 2024. [Online]. Available: https://learn.microsoft.com/en-us/sharepoint/introduction
[39] ‘Can SharePoint Be Used as a Database?’, BairesDev. Accessed: Apr. 29, 2024. [Online]. Available: https://www.bairesdev.com/blog/can-sharepoint-be-used-as-a-database/
[40] ‘Threat Actors Targeting Microsoft SQL Servers’, The Hacker News. Accessed: Jan. 01, 2024. [Online]. Available: https://thehackernews.com/2023/09/threat-actors-targeting-microsoft-sql.html
[41] ‘MongoDB Suffers Security Breach’, The Hacker News. Accessed: Jan. 01, 2024. [Online]. Available: https://thehackernews.com/2023/12/mongodb-suffers-security-breach.html