HOSPITAL PATIENT MANAGEMENT SYSTEM PROJECT

Name – Sonaal Rawat

Student ID – 23223310

Email – Sonaal.rawatucdconnect.ie

Word Count – 1500

Course – MSc Information Systems

Module – Project Management (IS40890)

Table of Contents

Software Development Lifecycle Methodologies for a Hospital Patient Management System (HPMS)

Introduction

The Development of a Hospital Patient Management System (HPMS) is a complex, multi-million-euro project requiring careful planning and execution. Selecting The appropriate Software Development Lifecycle (SDLC) methodology is critical to ensuring efficiency, quality, and compliance with healthcare regulations. This report evaluates various SDLC methodologies and recommends the most suitable approach for this project.

Important Factors for HPMS Development.

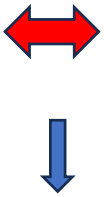
User-Centric Design	Medical professionals (e.g., doctors, nurses) and administrators need intuitive interfaces that minimize errors and streamline daily workflows.
Security	HPMS holds patients' sensitive data (e.g., **medical history, prescription**). Strong protocols - like encryption, access control, and multi-factor authentication are essential.
Patient Safety & System Reliability	HPMS must **ensure 24/7 availability**, accurate patient data, and fault tolerance to prevent service interruption.
Scalability	The system should handle growing volumes of patient data, new hospital branches, and future technologies.
Adherence to Regulatory Standards	Ensuring compliance with healthcare regulations like **GDPR** and **HIPPA** to protect patient data and privacy.

Risk Mitigation and Change Resilience	HPMS must handle regulatory changes, emerging technologies, and unexpected risk (e.g., **pandemics**).

Table: 1 Important Factors

Benefits and drawback of Important HPMS factors

Factor	Drawback If not Implemented
System Reliability & Patient Safety	Medical errors, Patient Risk
Security	Data breaches, loss of trust and liabilities
Scalability	Higher Cost, Inflexibility and Increase work load
Adherence to Regulatory Standards	Reputation Damage, severe penalties
User-Centric Design	User frustration, Operational inefficiency
Risk Mitigation and Change Resilience	Unaddressed Failures, Cost Overruns



Benefits
Error Reduction, Continuous access to patient Information during emergencies
Data protection, Access Control and Reduce threat from malware
Future proofing, Efficient Growth and Cost effective
Legal protection, Patient Trust
Reduced Training, Fewer errors
Change flexibility, continuity

Table:2 Benefits and Drawback

Overview of Key SDLC Methodologies

Diagram: 1 SDLC Methodologies

Several SDLC Methodologies are widely used in software development for their suitability in the context to of developing A HPMS.

.

SDLC Model	Features	Common use cases
Waterfall	A step-by-step process that is strict and docus heavily on documentation.	Large, well-defined projects (e.g., government systems)
Agile	Fast feedback, iterative, flexible	Continuous updates (e.g., web apps)
Scrum	Sprint based, frequent changes	Rapidly evolving (e.g., startups, SaaS
V-model	Extension of waterfall, with testing	High-reliability systems (e.g., aerospace, healthcare)
Spiral	Early risk management, risk-driven	Large scale projects with updates required
DevOps	Automation, Collaboration between dev & ops	Cloud-based application

Table: 3 Types of Methodologies

Comparative Assessment of SDLCs for HPMS Development.

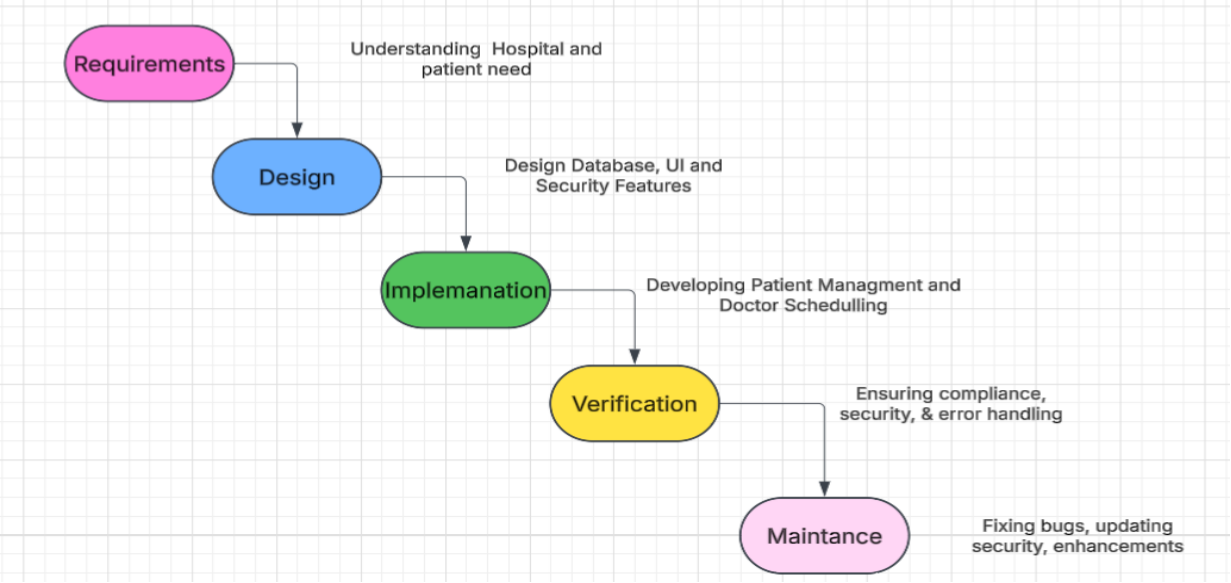
Each Methodology has strength and weakness in the context of HPMS Project, It is important to assess each model based on how properly it fits the vital factors of healthcare software systems, consisting user-centric design, patient safety and reliability, security, scalability and regulatory standards. Below we assess the pros and cons of each SDLC model in the context of an HPMS

Waterfall Model.

Waterfall method divides the project into distinct phases, each of which must be completed before the next one begins. One phase is completed, it is difficult to go back and make changes.

Pros:	Cons:
Clear structure and Documentation make it easy to document each step in detail, which is crucial.	It is difficult to make changes once the phase is started, For an HPMS, where users need might evolve this can be limitation
Each phase is clearly defined, so the budget is more predictable	There is little room for early feedback or changes

Table: 4 Waterfall pros & cons



Suitability for HPMS:

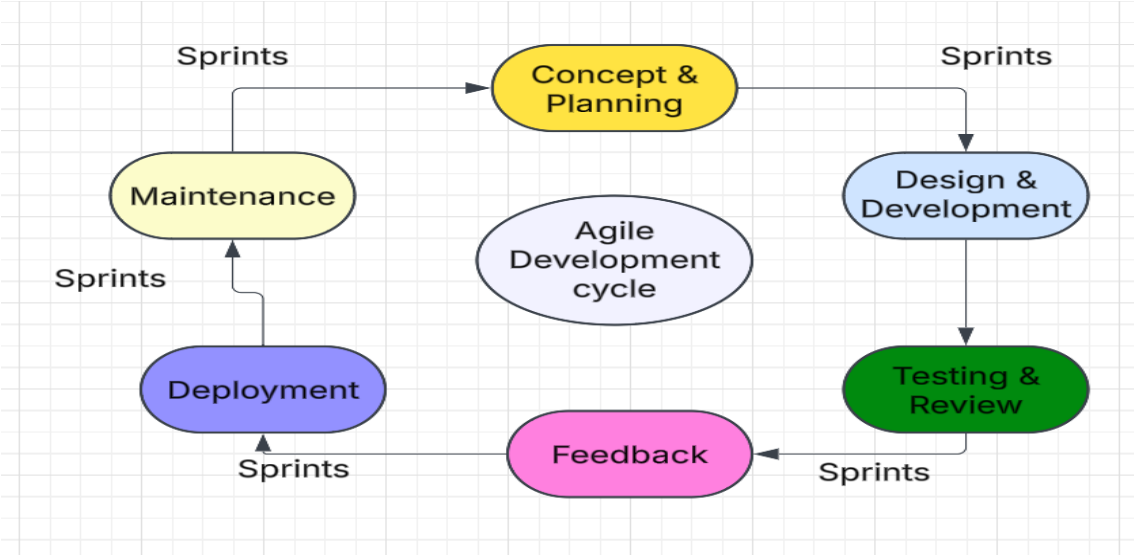
The Waterfall method is not the best method for project like HPMS due to its long development cycles. Healthcare systems like HPMS need regular feedback and flexibility, which Waterfall doesn't provide effectively.

Agile Method

The Agile Model is an Iterative approach and the development is done in small cycles(sprints), with frequent releases and updates, allowing for constant feedback and improvement.

Pros	Cons
Changes can be made throughout the development and is highly adaptable	Focus more on software over detailed documentation, which can make it challenging to meet Healthcare regulations.
Frequent iterations mean that stakeholder can provide input throughout the project, ensuring that the system meets their needs.	Less effective for large projects like HPMS that require more structure and predictability

Table:5 Agile pros & cons



Suitability for HPMS:

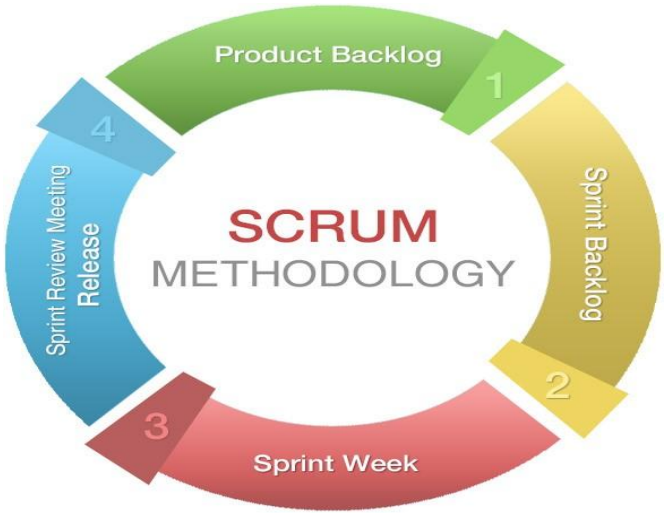
The lack of detailed documentation makes Agile less suitable for HPMS, while it offers flexibility and rapid iteration but is not suitable for large, regulatory heavy healthcare systems.

Scrum Model

Scrum Focus on team collaboration, daily meetings (standups) and is a subset of Agile with a focus on short, fixed length sprints that deliver incremental improvements.

Pros	Cons
Fast Iterations and Frequent Deliverables	Hard to manage for large projects that requires coordination across multiple teams
Encourages Cross functional team collaboration	Stakeholders must be consistently involved and available, which might be difficult in a busy healthcare setting.

Table: 6 Scrum pros & cons



Suitability for HPMS:

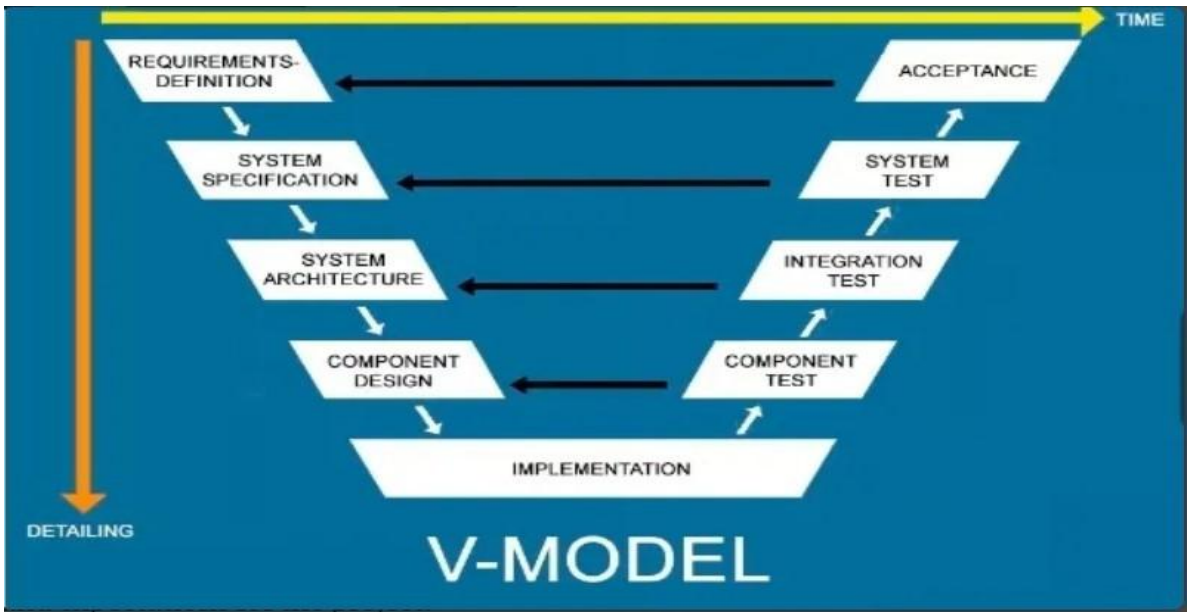
Scrum might work for smaller subprojects or specific components of an HPMS, but is not idea for the entire project. Scrum is best for smaller project like web development or software startups.

V Model

V Model requires verification and validation through each phase of development. Each development phase a separate testing phase, ensuring that testing and validation are integral to the process.

Pros	Cons
Each phase is tested ensuring errors are detected early and that the systems meets all specification	Long development timelines
Ideal for high reliability like HPMS, where safety and data	V-Model is rigid and struggle to make changes late.

Table: 7 V-Model pros & cons



Suitability for HPMS:

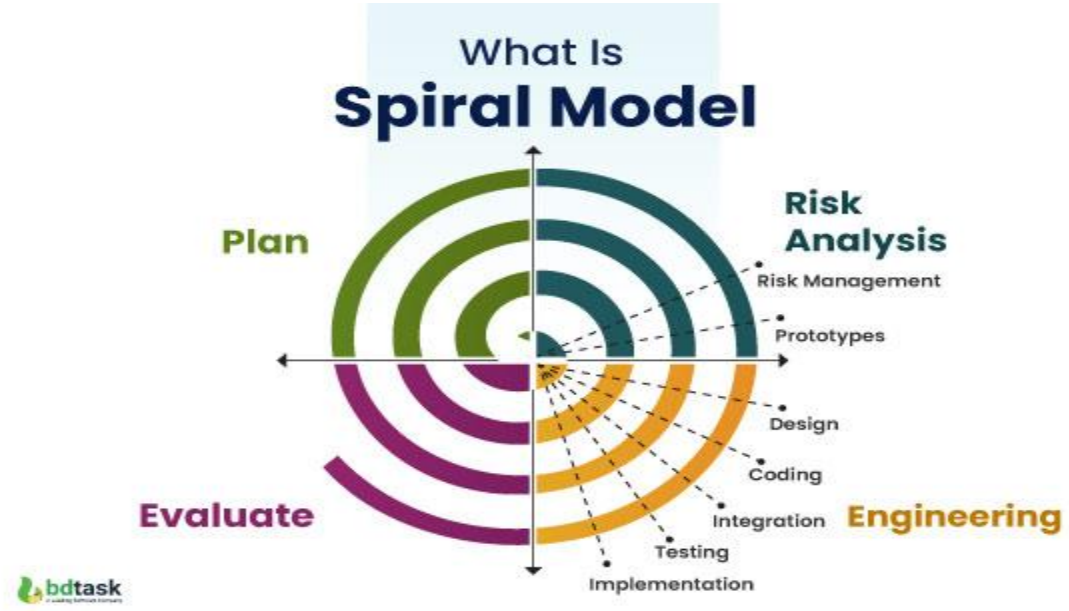
V-Model is best suitable for HPMS project because it focuses on rigorous testing and validation. HPMS requires high reliability, security and regulatory compliance which the V-Model supports effectively.

Spiral Model

The Spiral Model combines iterative development with a focus on risk analysis. Each iteration involves planning, design, prototyping, and risk assessment, which helps to discover potential risks in early development process.

Pros	Cons
Major focus on risk management makes it well suited for large project	Require careful management and highly experienced teams
Requirement can be refined through each iteration, ensuring systems evolves as new information becomes available	It can be more expensive than other methodologies

Table: 8 Spiral pros & cons



Suitability for HPMS:

The Spiral Model is suitable for large scale HPMS project with risk, evolving requirement involved or those that need frequent refinements. However, its complexity and cost may make it less ideal for smaller organization or those with limited budgets.

DevOps Model

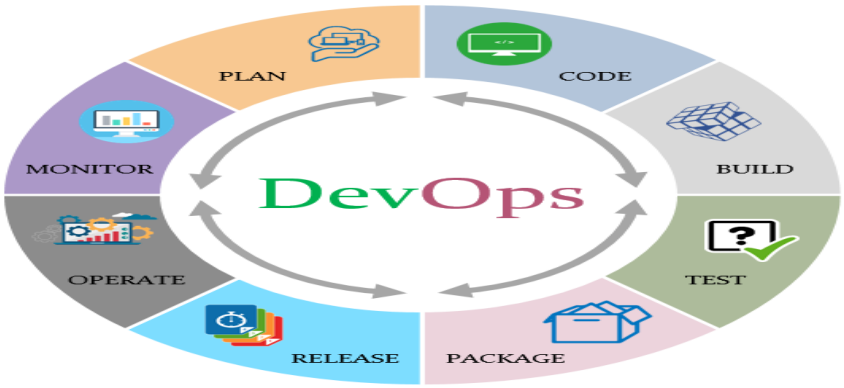
DevOps is a collaboration between software developers and Its operations to enable continuous integration delivery, and monitoring of software. It promotes automation and fast feedback.

Pros	Cons
Collaborating between development and operations teams, leading to faster issue resolution and more reliable systems.	Requires Significant investment in infrastructure and automation tools.
DevOps supports rapid delivery of software by automating the integration and deployment process, ensures updates and bug fixes are deployed quickly	Less Suited for projects with strict regulatory oversight.

Table: 9 DevOps pros & cons

Suitability for HPMS:

DevOps is not suitable for HPMS because rapid peace of updates may conflict with the rigorous testing and regulatory compliance requires in healthcare sector.



SDLC recommended for HPMS Development

Given important factors for an HPMS and comparative evaluation of the SDLC, V Model is the most appropriate function for the project.

Justification for choosing V-Model

Regulatory compliance: The structured approach ensures detailed documentation, which is necessary for regulatory approval.

The reliability of the systems: increase parallel verification and strengthening of the confirmation systems.

Scalability: While changes can be expensive, planning in advance ensures even scalability.

Stakeholder belief: clear project mile stone and test pages provides openness to hospital administrators and investors.

Waterfall model lacks flexibility and **DevOps** is well suited for cloud-based application and while **Agile** and **Scrum** provide flexibility, their iterative nature may lead to regulatory and security challenges.

Supporting Evidence

Studies indicate that structured SDLCs such as V Model and waterfall prefer to emphasize compliance and reliability in IT projects in the health care system (smith et al,2023), the research from healthcare information and management system society (HIMSS) highlights the importance of verification and documentation in software development at the hospital. Case studies from NHS and HSE suggest that strict function reduces the risk in large – scale health project.

Conclusion

It is important to choose the right SDLC method for the successful development of HPMS. Depending on the important project factors, V-Model approach is best recommended due to its structured development process, emphasis on verification and suitability for compliance with regulation. While other methodologies offer advantage, V-Model provides the necessary stiffness and forecasts required for a healthcare system.

Reference

- Smith, J., Brown, K., & Taylor, L (2023). Software development In Healthcare: Best Practices and Case Studies, Oxford University Press.
- Healthcare Information and management Systems society (HIMSS) (2022). Healthcare Its development and compliance Standards.
- National Health Service (NHS) (2021). Digital Transformation in Hospital Systems: Lessons Learned.
- Lucid chart for Diagrams https://www.lucidchart.com