



MANAGING THE DEVELOPMENT PROCESS OF THE SERVICE MANAGEMENT SYSTEM

Author: Anchalee Homjai

Advisors: Kittipitch Kuptavanich

Company: I GEAR GEEK Co., Ltd.



FACULTY OF SCIENCE
CHIANG MAI UNIVERSITY

ABSTRACT

As a Project Manager (PM) intern, I led the development of a Service Management System, a CRM tool designed for B2B organizations to improve customer relationship tracking while ensuring data confidentiality. My role encompassed requirement analysis, project planning, risk management, team coordination, and stakeholder communication to ensure successful project execution.

To achieve this, a hybrid project management approach (Agile + Waterfall) was implemented, combining Agile's adaptability with Waterfall's structured planning for efficient progress tracking and risk mitigation. The system was developed using Golang, Vue.js, and MySQL, along with frameworks such as Fiber, Quasar, TailwindCSS, and Nuxt.js, ensuring scalability and high performance.

Through effective project management, the system successfully streamlined customer management processes, improved operational efficiency, and enhanced data security. This project highlights the critical role of structured project management in delivering high-impact software solutions within a business environment.

INTRODUCTION

Effective management is crucial to the success of software companies in today's competitive environment. This study focuses on identifying the factors that influence management success in a software company located in Chiang Mai. To address these factors, a Service Management System was developed to streamline internal data management and enhance customer tracking efficiency.

As the project manager, the researcher oversaw the design and implementation of this system, ensuring it met the organization's needs for improving operational workflows and customer relationship management.

METHODOLOGY

1. Agile Practices (2-3 weeks per sprint)

- **Sprint Planning:** Tasks, goals, and priorities were collaboratively defined at the start of each sprint, based on stakeholder feedback and the product roadmap.
- **Daily Stand-Ups (15 minutes):** Brief daily meetings updated progress, addressed roadblocks, and maintained communication.
- **Sprint Reviews:** Sprints concluded with reviews where completed features were demonstrated, and feedback was gathered for the next iteration.
- **Sprint Retrospectives:** Post-sprint reflections using EasyRetro identified areas for improvement, with themed sessions (e.g., Game of Thrones) fostering engagement.

2. User-Centered Design

- **Persona Analysis:** Collaboration with the client to understand user needs and pain points, influencing the system's design.
- **Wireframing:** Visualized the system's structure and interface to ensure alignment on user experience.
- **User Acceptance Testing (UAT - 1 week):** Allowed end-users to test the system, providing feedback for final refinements.

3. Risk Management

- **Risk Identification & Assessment:** Proactively identified and assessed risks, addressing technical challenges and resource constraints.
- **Risk Response:** Mitigation strategies minimized high-priority risks and ensured project continuity.

4. Requirements Management

- **Requirements Gathering:** Collaborated with stakeholders through interviews and workshops to document system requirements.
- **Change Management:** Reviewed and implemented change requests based on their impact on scope and timelines.
- **Prioritization:** Focused on delivering high-value features first, aligned with user needs.

5. Project Phases

- **Requirements Gathering (5 weeks):** Defined user needs and documented requirements, supported by wireframes.
- **Development (12 weeks, 5 Sprints):** Iterative sprints using a tech stack that included Vue.js, Golang, and MySQL.
- **UAT (1 week):** Validated the system with end-users and addressed feedback.
- **Production & Deployment:** Deployed the system, created user documentation, and trained staff.

6. Testing & Quality Assurance

- **Continuous Testing:** Integrated testing throughout development to ensure functionality and reliability.
- **Full System Testing:** Comprehensive testing before deployment, ensuring the system met client expectations.

Tools

- **Instagantt:** For project scheduling and tracking.
- **EasyRetro:** For sprint retrospectives.

RESULT

The Service Management System significantly enhanced the company's internal workflows, improving data management and customer relationship tracking. It streamlined back-office operations, enabling staff to efficiently manage customer information and service requests. The user-friendly interface reduced training time and boosted productivity. Feedback from user testing revealed high satisfaction with the system's reliability, ease of use, and speed. Additionally, the system enabled real-time tracking and prioritization of customer requests, resulting in faster responses and improved customer satisfaction. Overall, the system contributed to more efficient operations and strengthened the company's customer relationships.

CONCLUSION

The development of the Service Management System demonstrated the effectiveness of combining modern web development tools with structured project management methodologies. The system not only streamlined internal operations but also provided a scalable solution for future enhancements. The lessons learned from this project highlight the importance of clear planning, agile responsiveness, and continuous feedback in delivering impactful software solutions.

REFERENCES

- Alert Media. (n.d.). Risk Management Lifecycle: 5 Steps to a Safer Organization. Retrieved from <https://www.alertmedia.com/blog/risk-management-lifecycle/>
- Prismetric. (n.d.). Guide to Website Development. Retrieved from <https://www.prismetric.com/comprehensive-guide-to-website-development/>
- Cambridge Network. (n.d.). Cost, Quality & Time – The Triple Constraint. Retrieved from <https://www.cambridgenetwork.co.uk/news/cost-quality-time-triple-constraint>