Project Progress Report for Eevee’s Retreat

Practice Module for Certificate in Designing Modern Software Systems

1st February 2025 to 14th February 2025

**Group 7**

|  |  |
| --- | --- |
| **Name** | **Email** |
| Ho Zheng Wei | zhengwei.ho@ncs.com.sg |
| Izz Danial Bin Selamat | izzdanial.selamat@ncs.com.sg |
| James Nicolas Tan Cher Wei | james.tan@ncs.com.sg |
| Kahbelan Kalisalvam Kelaver | kahbelan.kelaver@ncs.com.sg |

**Contents**

[1. Introduction 3](#_Toc190439788)

[1.1 Project Name & Description 3](#_Toc190439789)

[1.2 Project Methodology 3](#_Toc190439790)

[1.3 Project Summary 3](#_Toc190439791)

[2. Project Progress Report 5](#_Toc190439792)

[2.1 Reporting Period 5](#_Toc190439793)

[2.2 Sprint Objectives 5](#_Toc190439794)

[2.3 Sprint 1 Accomplishments (Planned vs. Actual) 6](#_Toc190439795)

[2.4 Sprint 1 Burndown Chart 9](#_Toc190439796)

[2.5 Problems encountered, Action Plan, Status 9](#_Toc190439797)

[2.6 Sprint Retrospective 9](#_Toc190439798)

[2.6.1 What went well? 9](#_Toc190439799)

[2.6.2 What could have been done better? 9](#_Toc190439800)

[2.6.3 What will we try next? 10](#_Toc190439801)

[2.7 Sprint 2 Preview 10](#_Toc190439802)

# Introduction

## **Project Name & Description**

**Eevee’s Retreat** is a web-based hotel booking system designed to make the reservation process simple, efficient, and hassle-free for customers. Guests can easily browse available rooms, check availability, book their stay, and manage their reservations all in one place.

The system also includes an admin dashboard, giving hotel staff the tools to manage room availability, pricing, reservations, and facility bookings with ease. With secure authentication, a user-friendly interface, and a streamlined booking engine, Eevee’s Retreat enhances both customer convenience and hotel operations.

## **Project Methodology**

The project follows an **Agile development methodology (SCRUM)**, ensuring an iterative and adaptive approach.

**Sprint Length:** 2 weeks per sprint

**Agile Artifacts:**

* **Sprint Planning:** Defining sprint goals and backlog prioritization
* **Daily Stand-ups:** Quick updates on progress and blockers
* **Sprint Review:** Demonstration of completed work
* **Sprint Retrospective:** Discussion on improvements for the next sprint

**Tracking & Tools:**

* GitHub Kanban Board for product backlog tracking
* GitHub for version control
* Microsoft Teams & Telegram Channel for team communication

## **Project Summary**

**Background:**

In the hospitality industry, providing a seamless and efficient booking experience is crucial for customer satisfaction and business success. Traditional hotel booking methods often lead to inefficiencies such as overbookings, manual errors, and lack of real-time availability updates. To address these challenges, there is a growing need for a modern, automated hotel booking and management system that enhances customer experience while improving hotel operations.

Eevee’s Retreat is designed as a web-based hotel management system that allows customers to browse available rooms, check availability, book their stay, and manage their reservations easily. At the same time, it provides hotel administrators with tools to efficiently manage room availability, pricing, bookings, and customer inquiries.

With an intuitive user interface, secure authentication, and a robust booking engine, the system aims to streamline hotel operations while providing a hassle-free experience for guests.

**Project Scope:**

**Deliverables**

1. A fully functional Eevee’s Retreat web application with core booking features.
2. Admin dashboard for hotel staff to manage rooms, bookings, and customer information.
3. User authentication system with secure role-based access control.
4. Database schemas & UML diagrams detailing system architecture.
5. Test cases & reports ensuring system functionality and reliability.
6. Comprehensive documentation, including user manuals and technical design specifications.

**Exclusions**

1. Integration with third-party payment gateways (payments will be manually processed in this version).
2. Mobile application development, as the focus is on a responsive web-based platform.
3. Multi-hotel chain support, as this version is tailored for a single-hotel system.

**Constraints**

1. Project timeline limitations, requiring us to focus on core booking and management features.
2. Limited familiarity with DevSecOps automation tools, requiring additional learning and setup time.
3. Resource constraints, as the team consists of a limited number of developers working within a fixed time frame.

# Project Progress Report

## **Reporting Period**

The project progress report will reflect the team's progress at the end of each sprint. Since our sprint duration is set to 2 weeks, this report will document the activities and accomplishments during Sprint 1, which began on 1st February 2025 and ended on 14th February 2025.

Serving as the planning and analysis phase, laying groundwork for all subsequent sprints. The primary focus was to establish project objectives, requirements, and system architecture, ensuring that the team had a well-defined roadmap before moving into development in Sprint 2.

## **Sprint Objectives**

Sprint 1 was dedicated to preparing the foundation for the project. The team actively discussed and established the project scope, objectives, and functional requirements, ensuring alignment with our final deliverables. Additionally, this sprint involved setting up the development environment, version control, and collaboration tools to facilitate smooth progress in future iterations.

**Key focus areas during Sprint 1 included:**

* **Defining the project scope and objectives** to ensure clarity in development goals.
* **Requirements gathering and documentation** for both functional and non-functional aspects.
* Establishing **Agile development workflow** by setting up GitHub Kanban Board for backlog management.
* **Setting up the development environment**, including GitHub for version control and **CI/CD pipelines for deployment**.
* **Creating UML diagrams** (use case, sequence, ERD) and initial system architecture design to guide implementation.
* Establishing **communication channels via Microsoft Teams and Telegram** to ensure seamless collaboration.

Beyond technical setup, Sprint 1 also focused on building team synergy and effective collaboration. Daily stand-ups were introduced to maintain transparency and progress tracking. Initial risk assessments were conducted, helping the team identify potential bottlenecks early and formulate mitigation strategies.

Additionally, this sprint allowed the team to gain a deeper understanding of the project domain by discussing use cases, potential challenges, and success criteria. By the end of Sprint 1, the team had a solid foundation and a well-defined approach for development in the upcoming sprints.

## **Sprint 1 Accomplishments (Planned vs. Actual)**

In Sprint 1, the development team successfully established the foundational infrastructure for the Eevee’s Retreat hotel booking system. The project repository was set up on GitHub, implementing a version control strategy to ensure efficient collaboration and code management. The team adopted **Visual Studio Code** (VS Code) as the primary Integrated Development Environment (IDE), optimizing the development workflow.

A structured Agile workflow was introduced, with sprint tracking managed through the **GitHub Kanban Board**, allowing seamless backlog management and task allocation. Key project management activities included defining the project scope, objectives, and system requirements, ensuring alignment with deliverables. Moreover, using the Kanban Board integrated within GitHub enabled the team to track and complete user stories efficiently without the need to reference another tracking tool out of the version control. The team also conducted extensive discussions to document functional and non-functional requirements, providing a clear development roadmap for subsequent sprints.

Within the GitHub Kanban Board, backlog items for Sprint 1 were identified and prioritized based on estimated effort and dependencies. Sprint planning involved breaking down tasks into manageable components, ensuring a balanced workload distribution. Team collaboration was facilitated through **Microsoft Teams** and **Telegram**, enabling real-time communication, task updates, and efficient resolution of blockers.

In addition to project management, significant progress was made in architectural design. The team started work on **UML diagrams**, including Use Case, Sequence, and Entity-Relationship Diagrams (ERD), which serve as the blueprint for system implementation. However, some diagrams are still undergoing refinements based on team feedback.

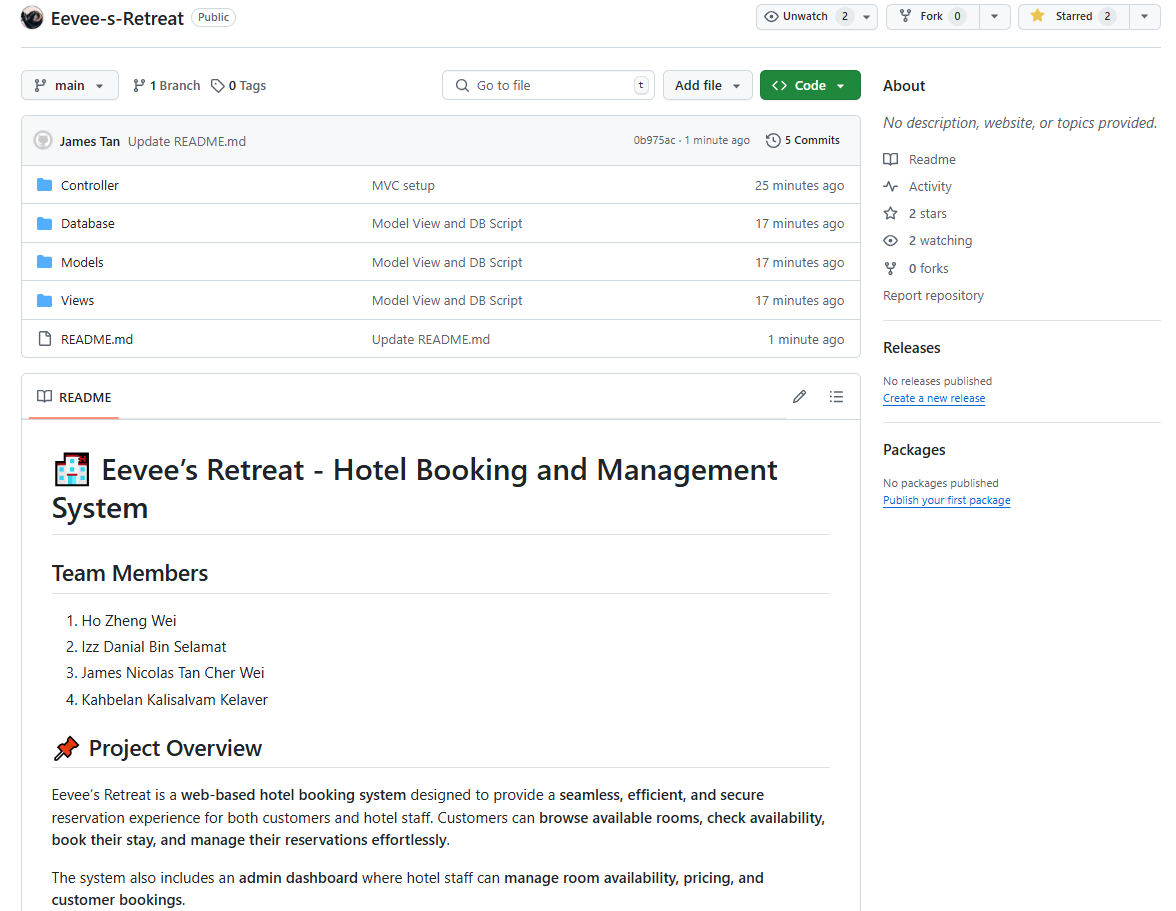
Sprint 1 also focused on risk assessment and mitigation strategies, identifying potential challenges such as timeline constraints and resource limitations. Daily stand-ups were introduced to maintain progress visibility, while sprint retrospectives allowed for continuous improvement discussions.

The team faced some challenges in finalizing the system architecture design, leading to minor delays. Despite this, Sprint 1 laid a solid foundation for upcoming development phases. With a well-defined scope, structured workflow, and established infrastructure, the team is now well-prepared to transition into **Sprint 2**, where core development will commence.

|  |  |  |  |
| --- | --- | --- | --- |
| Task | Planned Completion | Actual Completion | Remarks |
| Define Project Scope & Objectives | ✅ Completed | ✅ Completed | Clear project direction established |
| Gather Functional & Non-Functional Requirements | ✅ Completed | ✅ Completed | Requirements documented in GitHub Projects Kanban Board |
| Setup GitHub Repository & Agile Tools | ✅ Completed | ✅ Completed | Version control and tracking tools ready |
| Create UML Diagrams (Use Case, Sequence, ERD) | 🔄 In Progress | ⚠️Partial | Some diagrams need refinements |
| System Architecture Design | 🔄 In Progress | Delayed | Awaiting team feedback |

The images below are a snapshot of the team’s progress during sprint 1, what was achieved was the initial set up of the GitHub repository, the development environment as well as the product backlog tracking through the kanban board on GitHub Projects.

**Overview of Repository Setup on GitHub:**

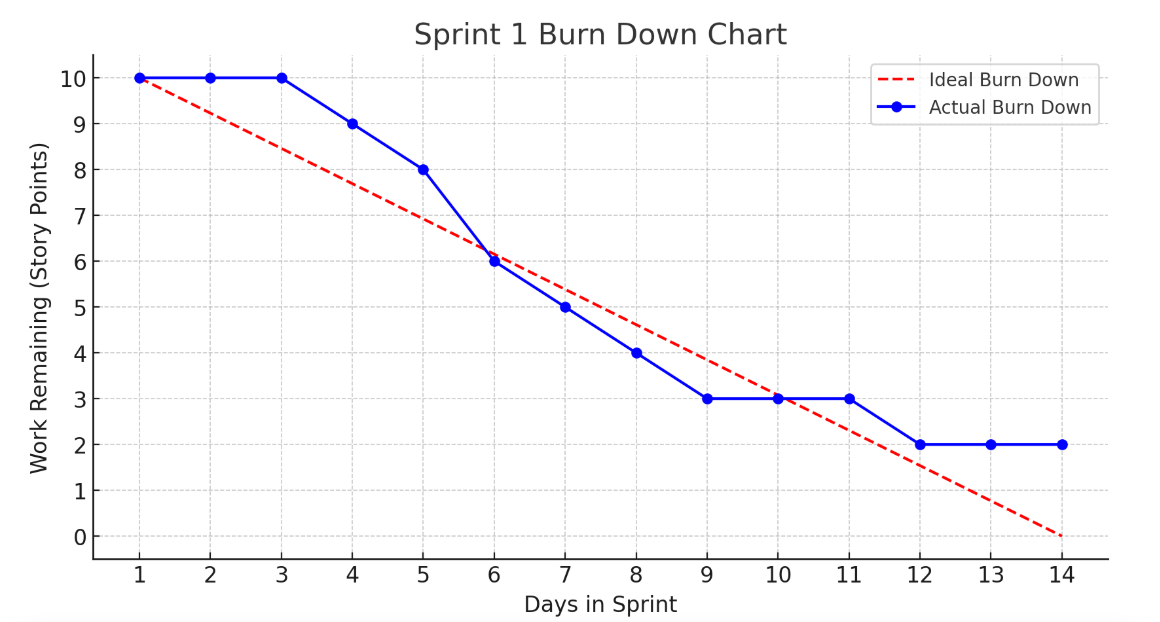


**Overview of Kanban Board using GitHub Projects:**

A screenshot of a computer

Description automatically generated

## **Sprint 1 Burndown Chart**



## **Problems encountered, Action Plan, Status**

|  |  |  |
| --- | --- | --- |
| Problem | Action Plan | Status |
| Lack of clarity in functional requirements | Held additional team meetings for clarification | ✅ Resolved |
| Some UML diagrams incomplete due to time constraint | Assigned dedicated reviewers to finalize diagrams | 🔄 In Progress |
| Minor GitHub conflicts | Created a Git workflow | ✅ Fixed |

## **Sprint Retrospective**

### **What went well?**

The sprint was successful in several aspects, particularly in establishing a clear project direction and well-defined scope, ensuring that all team members were aligned with the objectives. Effective collaboration and communication contributed to a smooth workflow, allowing tasks to be completed efficiently. Agile tools were properly set up, providing a structured approach to project management. Additionally, backlog tracking through the GitHub Kanban Board was efficient, helping the team stay organized and maintain visibility on task progress.

### **What could have been done better?**

Despite the positive outcomes, there were areas that required improvement. The completion of UML diagrams could have been timelier to provide a better visual representation of the system design. Additionally, task delegation could have been more structured, ensuring that responsibilities were distributed more effectively among team members to maximize productivity.

### **What will we try next?**

To improve future sprints, we will allocate more time to reviewing architecture diagrams, ensuring they are comprehensive and align with the project requirements. Furthermore, we will conduct a deeper analysis of sprint planning to refine our approach to task allocation, allowing for better workload distribution and improved efficiency throughout the development process.

## **Sprint 2 Preview**

In the upcoming sprint, our focus will be on completing the UML diagrams and refining the system architecture to ensure a solid foundation for development. This step is crucial for providing clarity on system design and ensuring alignment with project requirements. Additionally, we will begin the core development of the application, implementing essential features and functionalities. This phase marks a significant milestone in the project, as it transitions from planning and design to active development, bringing us closer to delivering a functional system.