Project Progress Report for Eevee’s Retreat

Practice Module for Certificate in Designing Modern Software Systems

17st February 2025 to 28h 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 2](#_Toc1469350587)

[1.1 Project Name & Description 3](#_Toc1164167819)

[1.2 Project Methodology 3](#_Toc851206732)

[1.3 Project Summary 3](#_Toc1204910164)

[2. Project Progress Report 4](#_Toc358948205)

[2.1 Reporting Period 5](#_Toc1085336967)

[2.2 Sprint Objectives 5](#_Toc855025180)

[2.3 Sprint 2 Accomplishments (Planned vs. Actual) 5](#_Toc1706556225)

[2.4 Sprint 2 Burndown Chart 7](#_Toc187187723)

[2.5 Problems encountered, Action Plan, Status 7](#_Toc1216953932)

[2.6 Sprint Retrospective 7](#_Toc1505358590)

[2.6.1 What went well? 7](#_Toc1409528001)

[2.6.2 What could have been done better? 8](#_Toc806386093)

[2.6.3 What will we try next? 8](#_Toc780334271)

[2.7 Sprint 2 Preview 8](#_Toc1580162203)

# 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 2, which began on 17st February 2025 and ended on 28th February 2025.

Sprint 2 marked the transition from the planning and analysis phase to active development. The focus was on refining system architecture, completing outstanding tasks from Sprint 2, and beginning core feature implementation. Ensuring a structured development process was key to maintaining efficiency and meeting project milestones.

## **Sprint Objectives**

The primary goals of Sprint 2 were:

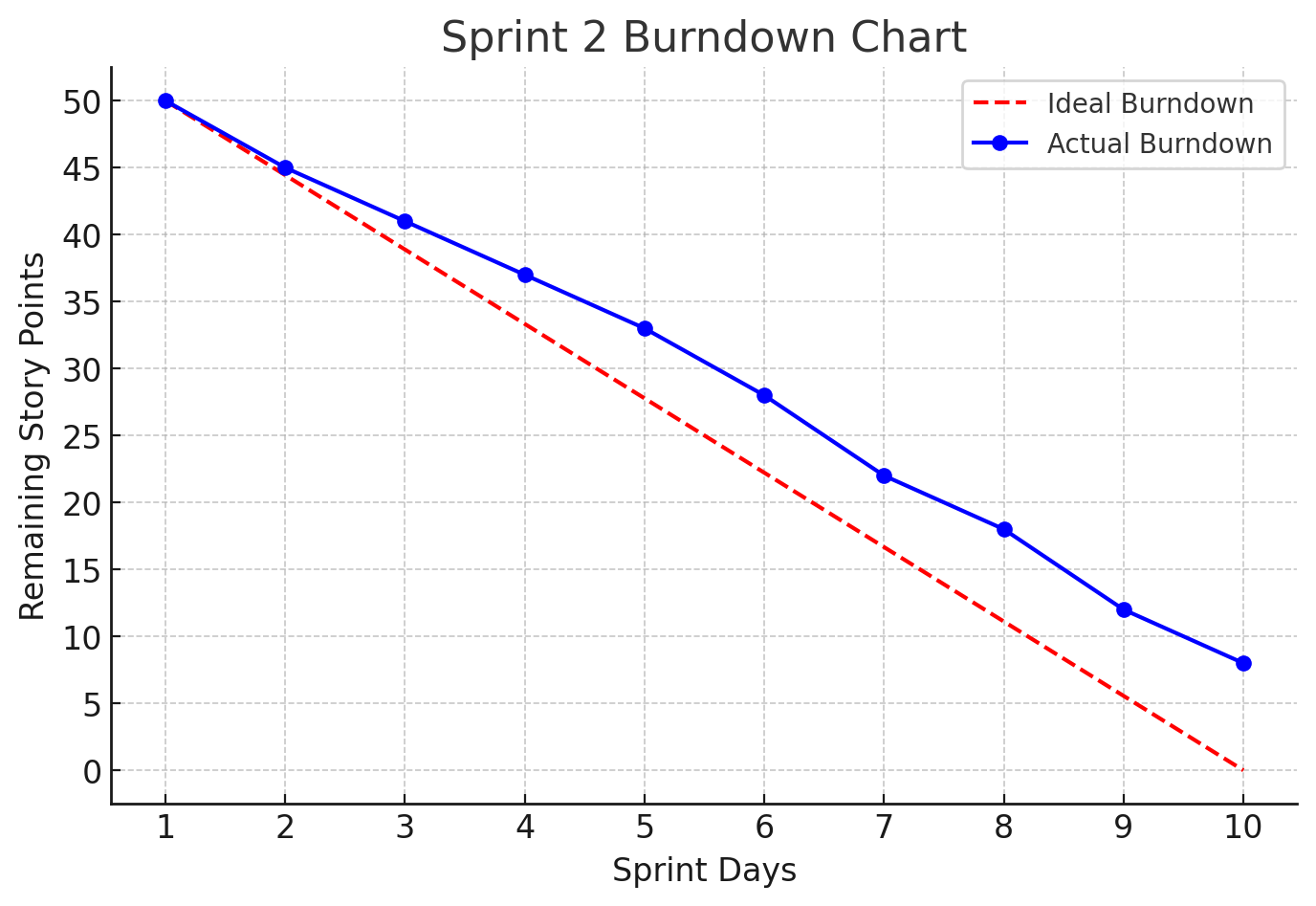
* Completing and refining UML diagrams to finalize system design.
* Finalizing system architecture based on team feedback and requirements.
* Implementing core functionalities of the system.
* Establishing database schema and integrating it with the backend.
* Setting up unit testing and initiating test cases for key functionalities.
* Enhancing team workflow by improving backlog management and refining sprint planning.

## **Sprint 2 Accomplishments (Planned vs. Actual)**

The team successfully refined and finalized the UML diagrams, incorporating necessary modifications based on feedback. The system architecture was approved, providing a solid foundation for ongoing development. Core feature implementation began, with initial modules completed and additional development continuing into the next sprint. The database schema was successfully integrated with the backend, allowing seamless data management.

|  |  |  |  |
| --- | --- | --- | --- |
| **Task** | **Planned Completion** | **Actual Completion** | **Remarks** |
| **Refine UML Diagrams (Use Case, Sequence, ERD)** | ✅ Completed | ✅ Completed | Finalized based on team feedback |
| **Complete System Architecture Design** | ✅ Completed | ✅ Completed | Approved and ready for implementation |
| **Implement Core Functionalities** | 🔄 In Progress | 🔄 In Progress | Initial modules developed; further work needed |
| **Set Up Database Schema & Integration** | ✅ Completed | ✅ Completed | Database structure established and integrated with backend |

## **Sprint 2 Burndown Chart**



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

|  |  |  |
| --- | --- | --- |
| **Problem** | **Action Plan** | **Status** |
| **Complexity in refining UML diagrams** | Conducted focused discussions for clarity | ✅ Resolved |
| **Delays in implementing certain functionalities** | Prioritized key features and reassigned tasks | 🔄 In Progress |
| **Initial difficulties in database integration** | Conducted team debugging sessions | ✅ Resolved |

## **Sprint Retrospective**

### **What went well?**

Sprint 2 was successful in several aspects. The team efficiently finalized the UML diagrams and system architecture, providing a strong foundation for further development. The database schema was implemented and integrated seamlessly with the backend, ensuring smooth data management. Additionally, backlog management and sprint planning were improved, allowing for better organization and alignment with the project timeline.

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

Core feature implementation was delayed due to complex requirements requiring clarification and adjustments. More detailed requirement breakdowns in future sprints can help prevent similar issues. Additionally, a more structured peer review process with clear criteria and scheduled sessions will improve code quality and efficiency.

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

To enhance future sprints, the team will prioritize feature development with clearer milestone targets to prevent delays. More frequent and structured code reviews will be conducted to ensure higher code quality and maintainability. Additionally, allocating additional resources and adjusting task distribution will help speed up feature implementation in Sprint 3, ensuring smoother progress toward project completion.

## **Sprint 3 Preview**

In the upcoming Sprint 3, the focus will be on completing core feature development, refining UI components, and integrating frontend with backend services. Additionally, thorough testing will be conducted to ensure system stability. The team will continue optimizing sprint workflows and refining backlog management for improved efficiency.