**SPRING PLANNING**

|  |  |
| --- | --- |
| **Mã sprint** | **2** |
| **Thời gian bắt đầu** | **23/09/2025** |
| **Thời gian kết thúc** | **29/09/2025** |

**1. Mục tiêu Sprint**

* Code giao diện chính, chức năng Dịch vụ & Chi phí khách lưu trú

**2. Backlog được chọn cho Sprint**

| **User Story** | **Mô tả** | **Độ ưu tiên** | **Points** | **Người phụ trách** | **Trạng thái** |
| --- | --- | --- | --- | --- | --- |
| US-03 | Đặt dịch vụ (Spa, Nhà hàng, hội thảo, …) | Trung bình | 3 | Phạm Thanh Lam | Done |
| US-05 | Theo dõi hóa đơn chi phí trong thời gian lưu trú | Trung bình | 3 | Phạm Thanh Lam | Done |
| US-06 | Thanh toán trực tuyến hoặc tại quầy | Cao | 5 | Phạm Thanh Lam | Done |
| US-07 | Đánh giá dịch vụ sau khi trả phòng | Trung bình | 2 | Phạm Thanh Lam | In Progress |
| US-11 | Xem lịch sử đặt phòng và hóa đơn cũ | Trung bình | 3 | Phạm Thanh Lam | Done |
| US-12 | Nhận gợi ý phòng hoặc dịch vụ dựa trên lịch sử đặt | Trung bình | 5 | Phạm Thanh Lam | In Progress |

**3. Phân công công việc chi tiết**

| **Công việc** | **Người thực hiện** | **Deadline** | **Công nghệ/tool/thư viện…** |
| --- | --- | --- | --- |
| Thiết kế UI cho các trang đặt dịch vụ (Spa, Nhà hàng, hội thảo) và theo dõi hóa đơn | Phạm Thanh Lam | 24/09/2025 | Figma, Bootstrap 5, TailwindCSS, Deluxe Template |
| Phát triển API và giao diện cho chức năng đặt dịch vụ Spa/Nhà hàng/hội thảo | Phạm Thanh Lam | 24/09/2025 | ASP.NET Core Web API 9.0, Entity Framework Core, SQL Server, Razor Pages, Bootstrap 5 |
| Xây dựng chức năng ghi nhận dịch vụ phát sinh trong thời gian khách lưu trú | Phạm Thanh Lam | 28/09/2025 | ASP.NET Core Web API 9.0, Entity Framework Core, SQL Server, Razor Pages, Bootstrap 5 |
| Chuyển chi phí phát sinh về hệ thống phòng, đồng bộ realtime với tình trạng phòng | Phạm Thanh Lam | 28/09/2025 | ASP.NET Core Web API 9.0, Entity Framework Core, SQL Server, Razor Pages, Bootstrap 5, SignalR |
| Xây dựng module quản lý phiếu xuất kho phục vụ các dịch vụ nội bộ | Phạm Thanh Lam | 28/09/2025 | ASP.NET Core Web API 9.0, Entity Framework Core, SQL Server, Razor Pages, Bootstrap 5 |
| Phát triển chức năng tạo hóa đơn thanh toán cuối cùng, tích hợp MoMo Payment API cho thanh toán trực tuyến hoặc tại quầy | Phạm Thanh Lam | 25/09/2025 | ASP.NET Core Web API 9.0, Entity Framework Core, SQL Server, MoMo Payment API, Bootstrap 5 |
| Phát triển API và giao diện theo dõi hóa đơn chi phí trong thời gian lưu trú | Phạm Thanh Lam | 28/09/2025 | ASP.NET Core Web API 9.0, Entity Framework Core, SQL Server, Razor Pages, Bootstrap 5 |
| Xây dựng module xem lịch sử đặt phòng và hóa đơn cũ, cho phép tra cứu chi tiết từng hóa đơn | Phạm Thanh Lam | 28/09/2025 | ASP.NET Core Web API 9.0, Entity Framework Core, SQL Server, Razor Pages, Bootstrap 5 |
| Phát triển tính năng gợi ý phòng hoặc dịch vụ dựa trên lịch sử đặt phòng của khách hàng (AI Suggestion Logic) | Phạm Thanh Lam | 27/09/2025 | ASP.NET Core Web API 9.0, Entity Framework Core, SQL Server, Bootstrap 5, ML.NET |
| Thiết kế giao diện và xử lý đánh giá dịch vụ sau khi trả phòng | Phạm Thanh Lam | 26/09/2025 | ASP.NET Core Web API 9.0, Entity Framework Core, SQL Server, Razor Pages, Bootstrap 5 |
| Test & kiểm thử tích hợp toàn bộ chức năng của Sprint | Phạm Thanh Lam | 29/09/2025 | ASP.NET Core Web API 9.0, Entity Framework Core, SQL Server, Manual Testing |
| Viết báo cáo Sprint, tổng hợp tiến độ và cập nhật repository | Nguyễn Bảo Nguyên | 29/09/2025 | Microsoft Word, GitHub |

**4. Rủi ro và biện pháp**

* **Rủi ro:**
* Thiết kế UI cho các trang dịch vụ (Spa, Nhà hàng, hội thảo) có thể chậm tiến độ do khách hàng thay đổi yêu cầu về giao diện.
* Khi tích hợp API dịch vụ phát sinh và thanh toán MoMo, có thể gặp lỗi xác thực hoặc sai định dạng dữ liệu phản hồi.
* Dữ liệu hóa đơn, phiếu xuất kho hoặc chi phí phát sinh có thể bị **trùng lặp hoặc thiếu đồng bộ** giữa hệ thống phòng và dịch vụ.
* Module gợi ý dịch vụ (AI Suggestion Logic) có thể **chưa tối ưu kết quả** do thiếu dữ liệu lịch sử đủ lớn.
* Chức năng đánh giá dịch vụ sau khi trả phòng có thể gặp lỗi **ghi nhận phản hồi trùng** hoặc hiển thị sai định dạng.
* **Biện pháp:**
* **Tổ chức họp daily meeting** để nắm bắt tiến độ và xử lý sớm khi có thay đổi từ khách hàng.
* Thống nhất **UI/UX guideline** ngay từ đầu sprint để tránh việc chỉnh sửa nhiều lần.
* Viết **unit test và integration test** cho các API: đặt dịch vụ, thanh toán, hóa đơn và phiếu xuất kho.
* Tăng cường **bảo mật API** bằng cách kiểm tra các lỗ hổng như SQL Injection, XSS, xác thực JWT, và mã hóa dữ liệu.
* Thiết lập **backup định kỳ và logging chi tiết** cho các bảng dữ liệu dịch vụ, hóa đơn và xuất kho.
* Review code chéo giữa các thành viên và test liên module (dịch vụ – phòng – thanh toán) để giảm lỗi tích hợp.
* Theo dõi hoạt động của SignalR realtime và tối ưu hiệu suất khi đồng bộ dữ liệu chi phí phòng.

**5. Sprint Planning Update (nếu là version cập nhật cho sprint cũ)**

 **Thay đổi so với kế hoạch ban đầu:**

* Cập nhật giao diện UI cho chức năng đặt dịch vụ Spa/Restaurant và hóa đơn thanh toán cuối cùng theo phản hồi của khách hàng.
* Bổ sung AI Suggestion Logic để gợi ý dịch vụ hoặc phòng phù hợp dựa trên lịch sử đặt trước đó.
* Điều chỉnh tiến độ của module dịch vụ phát sinh để đảm bảo đồng bộ realtime với hệ thống phòng.
* Bổ sung thêm test case và logging cho chức năng thanh toán nhằm đảm bảo xử lý ổn định với MoMo API.

 **Lý do cập nhật:**

* Đáp ứng yêu cầu mới từ phía khách hàng nhằm nâng cao trải nghiệm người dùng trong phần dịch vụ.
* Tăng cường độ an toàn dữ liệu và bảo mật giao dịch khi hệ thống tích hợp thanh toán trực tuyến.
* Giúp quy trình quản lý dịch vụ phát sinh – hóa đơn – phòng hoạt động trơn tru, tránh lỗi sai lệch dữ liệu giữa các module.