**TRƯỜNG ĐẠI HỌC BÁCH KHOA TP. HỒ CHÍ MINH**

**KHOA KHOA HỌC KỸ THUẬT VÀ MÁY TÍNH**

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*



BÁO CÁO LAB 1

MÔN HỌC: QUẢN LÝ DỰ ÁN PHẦN MỀM (THÍ NGHIỆM)

HỌC KỲ: HK241

*NỘI DUNG:*

**QUẢN LÝ DỰ ÁN VỚI DEVOPS**

***Người hướng dẫn***: Nguyễn Minh Tâm

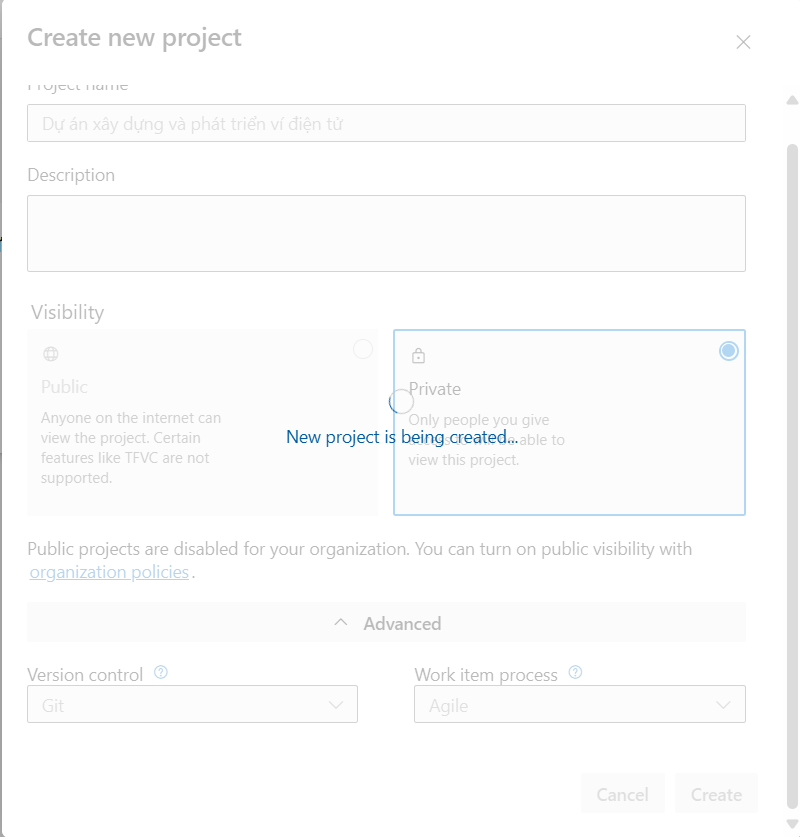
***Nhóm***: L01, ***Ngày hoàn thành***: 20/09/2024

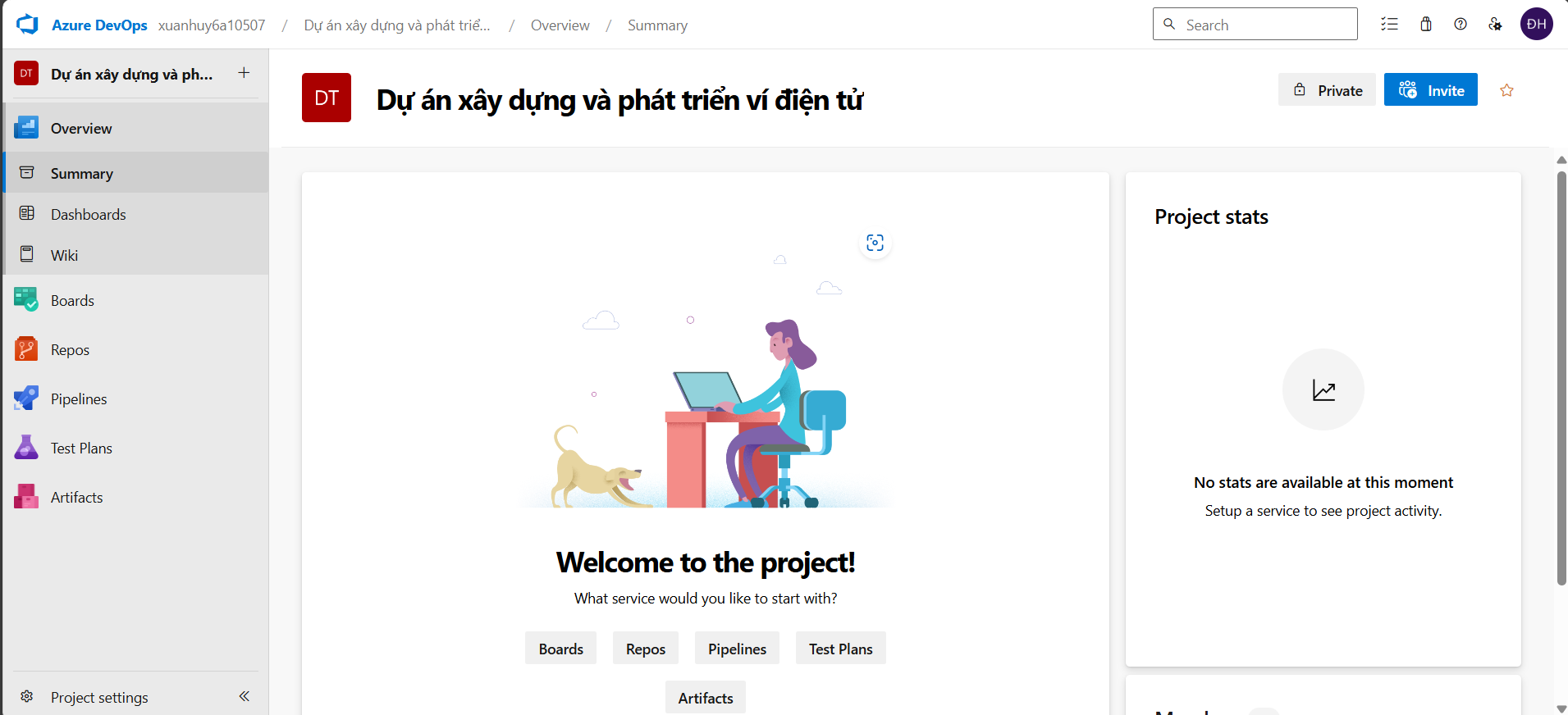
***Họ tên sinh viên:*** Đặng Xuân Huy ***MSSV*:** 2211161

*Thành phố Hồ Chí Minh – 2024*

***Task 1:*** Tạo một Project trong DevOps với ngữ cảnh quản lý quen thuộc của bản thân (ví dụ: phát triển phần mềm, quản lý sản phẩm, hoặc dự án học tập).

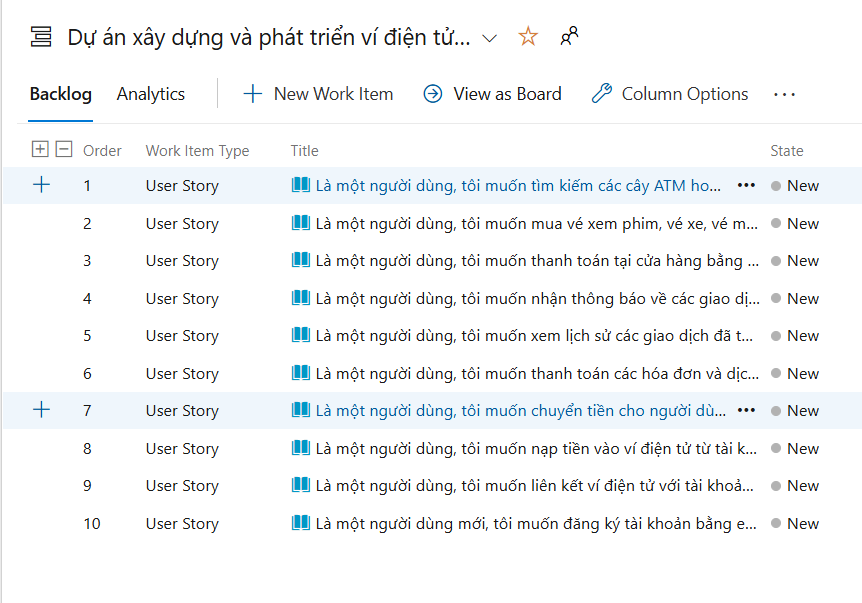
***Bài làm***

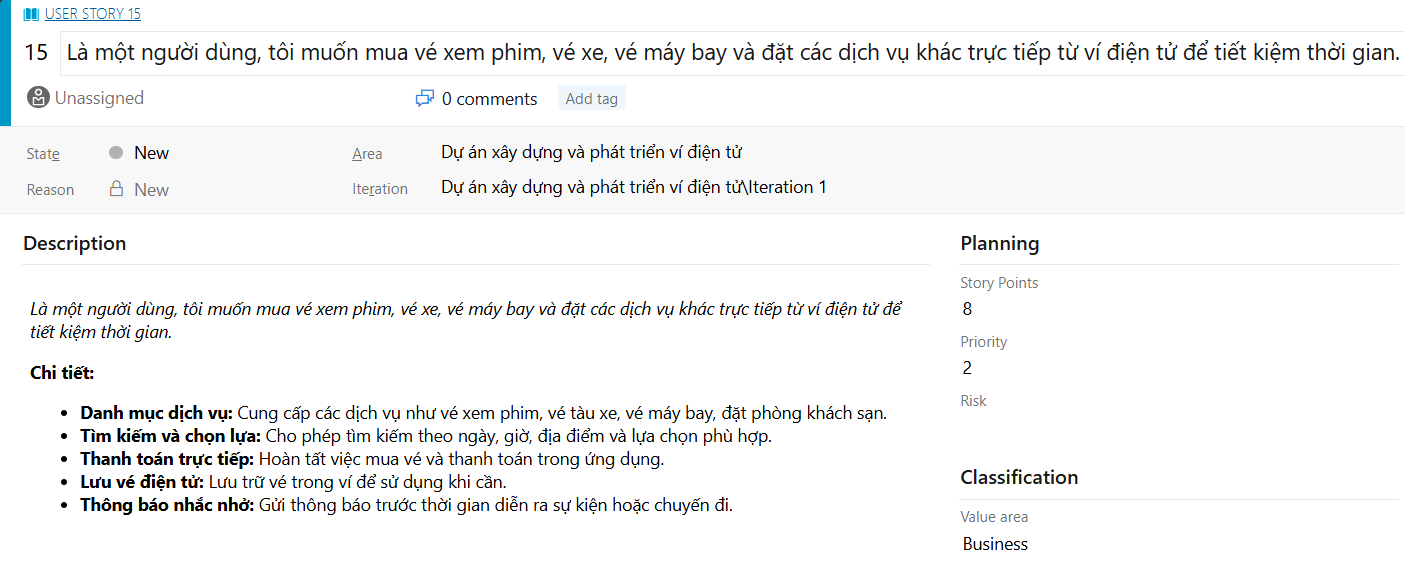
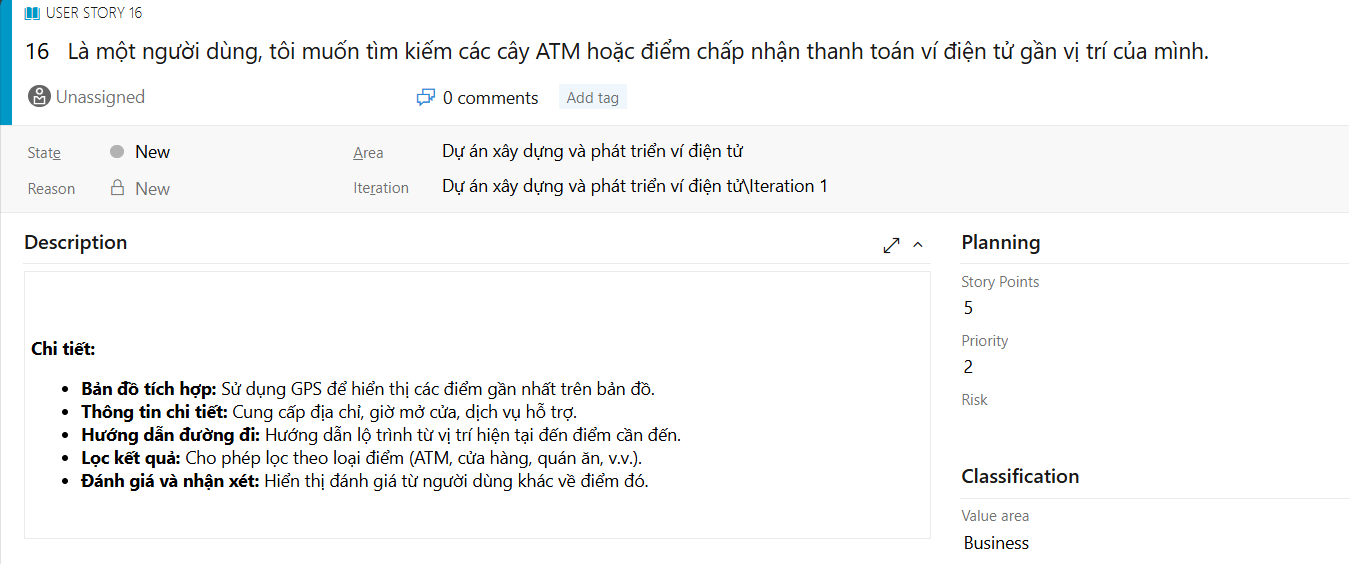


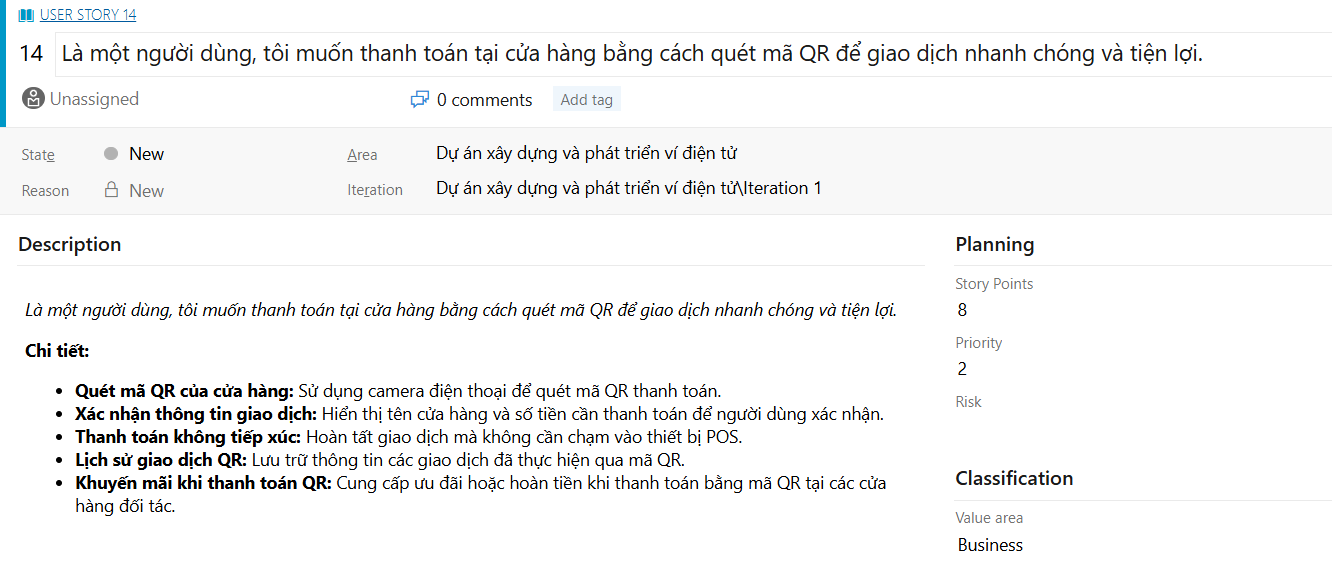


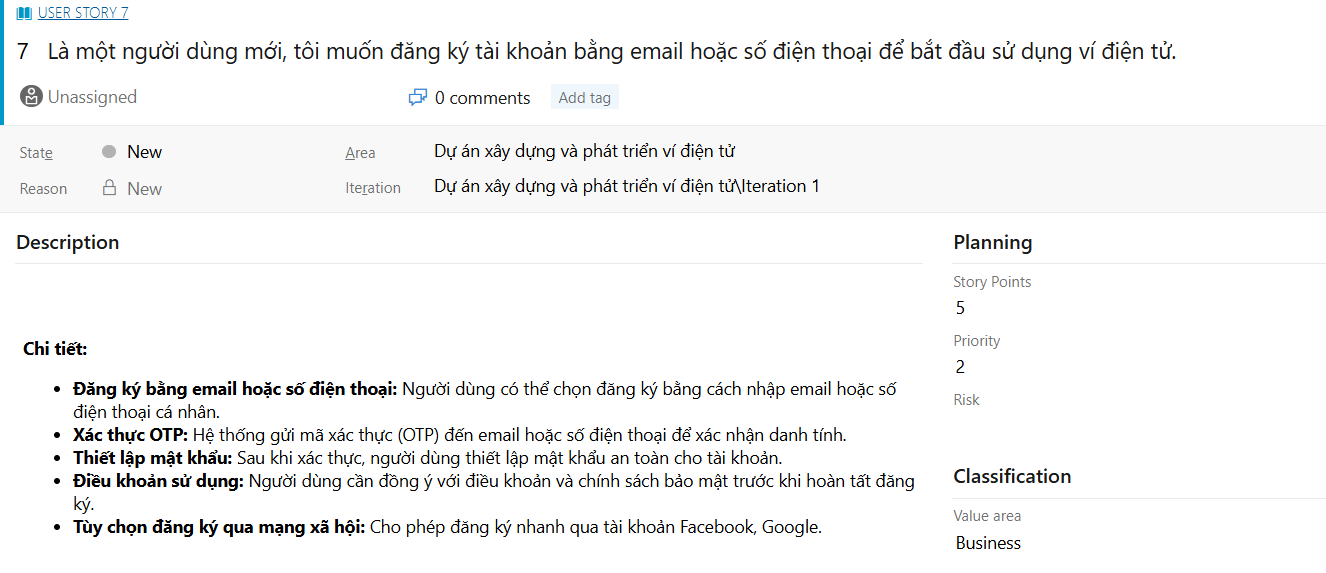
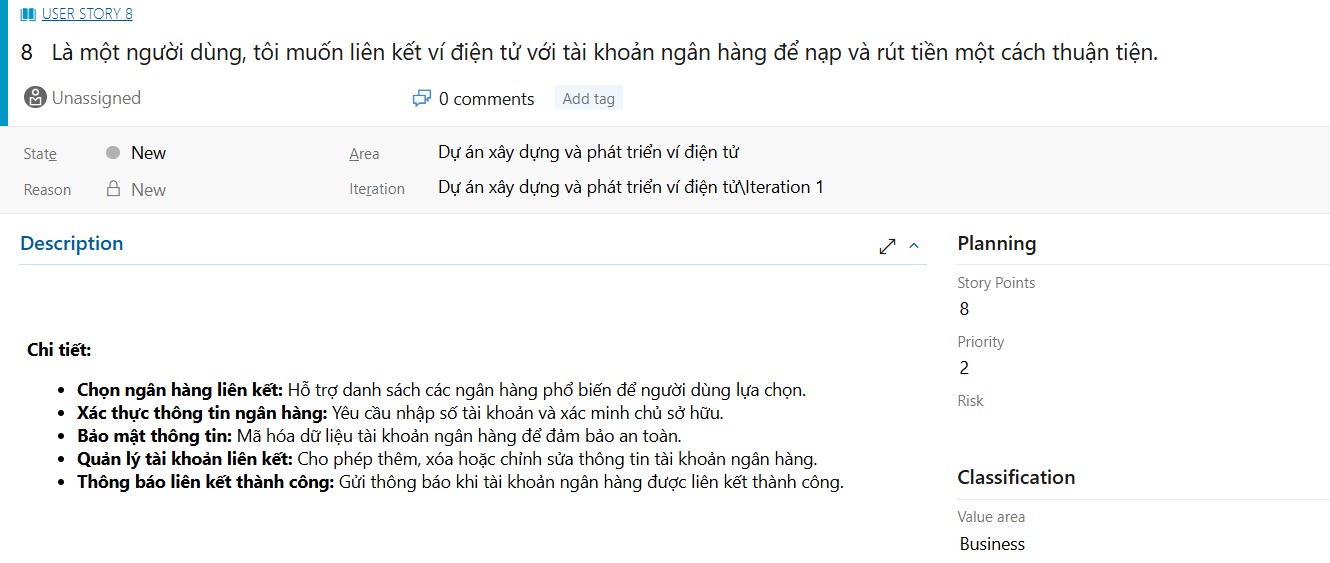
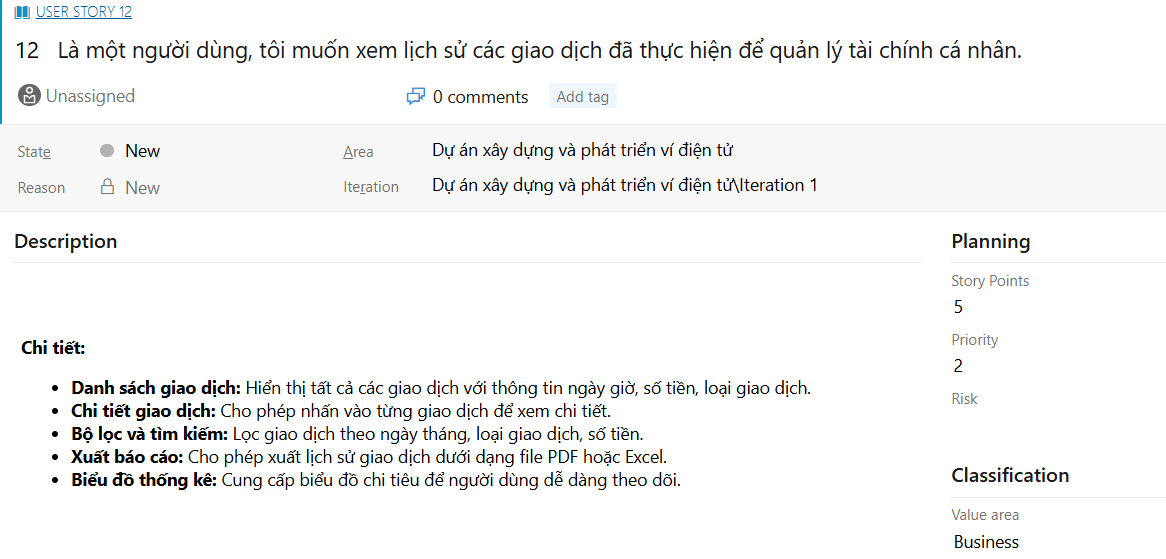
***Task 2:*** Tạo tối thiểu 10 User Stories mô tả các tính năng hoặc yêu cầu cần thực hiện trong dự án. Mô tả nội dung chi tiết cho từng User Story

***Bài làm***







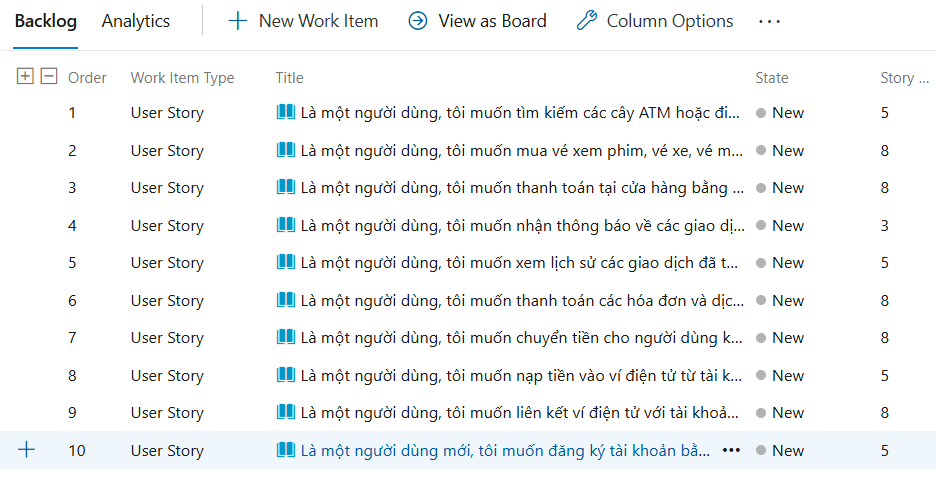


*Tổng quan*

|  |  |
| --- | --- |
| STT | User Story |
| 1 | Là một người dùng mới, tôi muốn đăng ký tài khoản bằng email hoặc số điện thoại |
| 2 | Là một người dùng, tôi muốn tìm kiếm các cây ATM hoặc điểm chấp nhận thanh toán ví điện tử |
| 3 | Là một người dùng, tôi muốn mua vé xem phim, vé xe, vé máy bay và đặt các dịch vụ khác |
| 4 | Là một người dùng, tôi muốn nhận thông báo về các giao dịch và chương trình khuyến mãi |
| 5 | Là một người dùng, tôi muốn thanh toán tại cửa hàng bằng cách quét mã QR |
| 6 | Là một người dùng, tôi muốn thanh toán các hóa đơn và dịch vụ trực tiếp từ ví điện tử |
| 7 | Là một người dùng, tôi muốn chuyển tiền cho người dùng khác trong hệ thống |
| 8 | Là một người dùng, tôi muốn nạp tiền vào ví điện tử từ tài khoản ngân hàng |
| 9 | Là một người dùng, tôi muốn xem lịch sử các giao dịch đã thực hiện |
| 10 | Là một người dùng, tôi muốn liên kết ví điện tử với tài khoản ngân hàng |

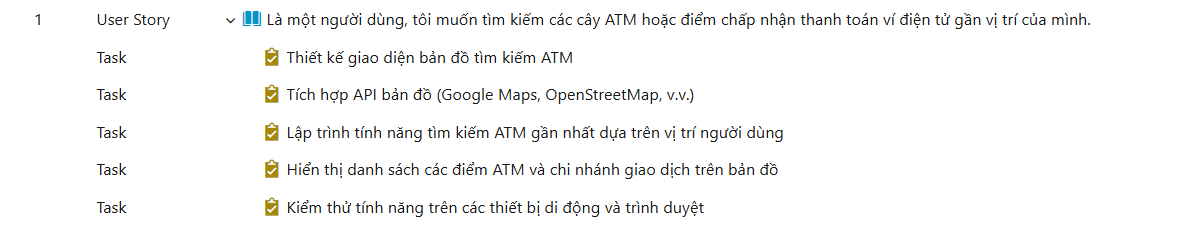
***Task 3****:* Ước lượng Story Points cho từng User Story dựa trên độ khó và thời gian thực hiện

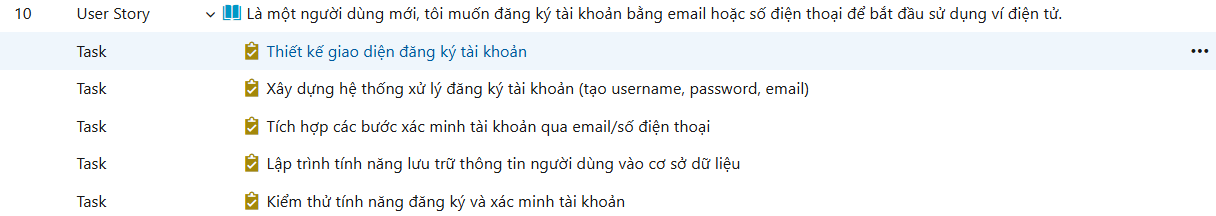
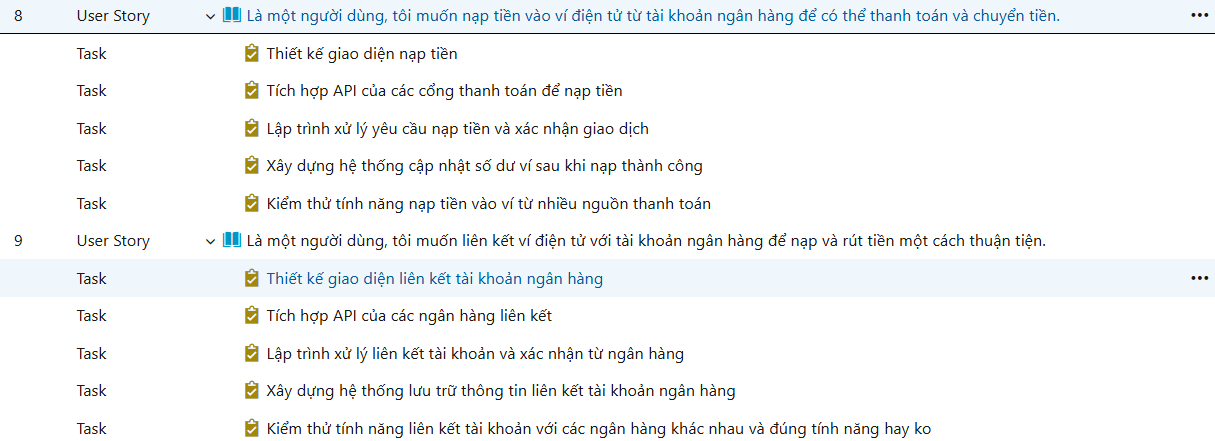
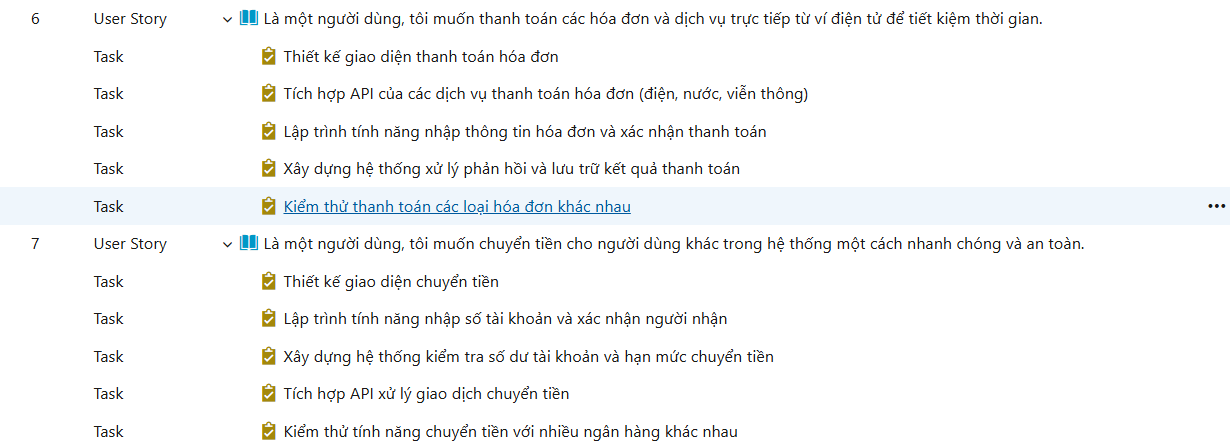
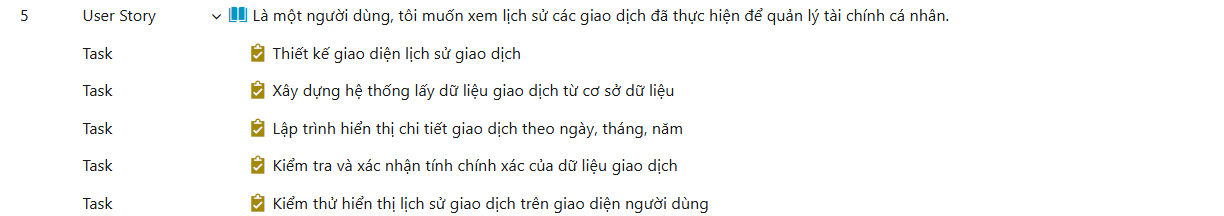
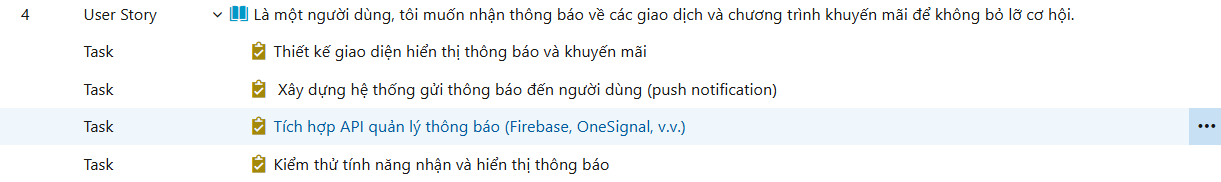
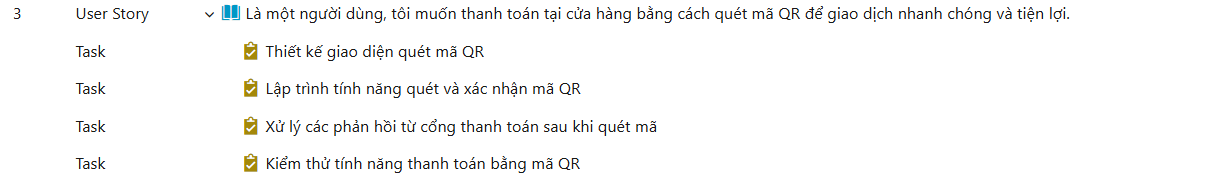
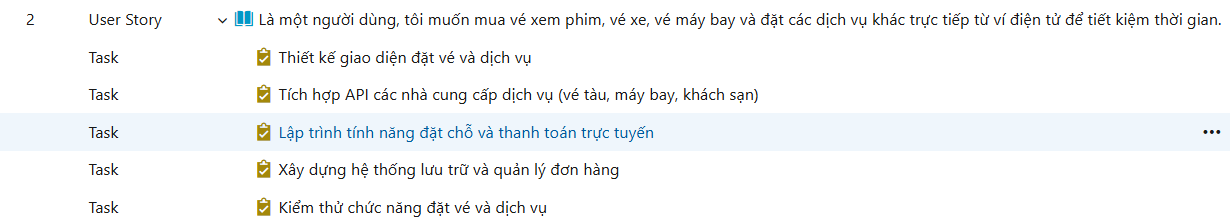
***Bài làm***



***Task 4:*** Phân rã Task cho mỗi User Story, chia nhỏ công việc thành các nhiệm vụ cụ thể.

***Bài làm***





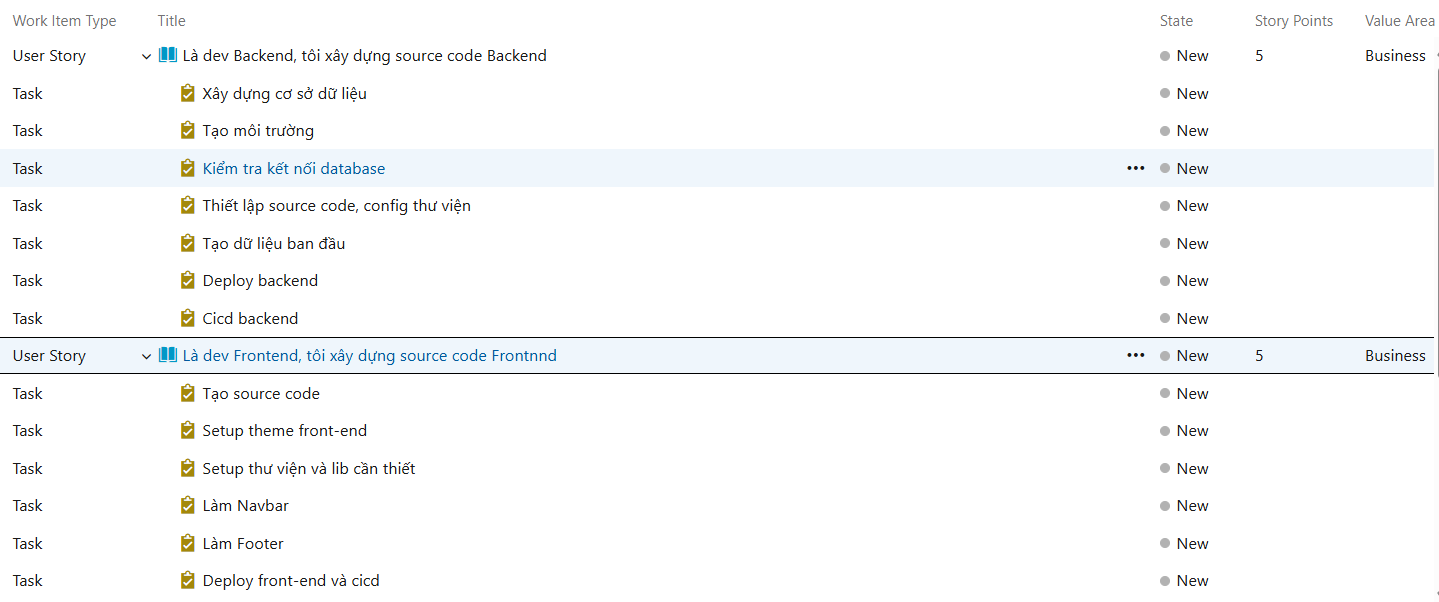
***Task 5:*** Tạo 4 Sprint và lên kế hoạch chi tiết cho mỗi Sprint:

* Sprint 1: Bắt đầu với các User Story đơn giản, làm quen với quy trình.
* Sprint 2: Tăng số lượng User Story / Task thực hiện.
* Sprint 3 & Sprint 4: Ưu tiên các nhiệm vụ quan trọng và tăng tốc hoàn

thành công việc

***Bài làm***

Tại task 5 thực hiện tạo thêm hai user story nữa để dự án hợp lý hơn là:



**Sprint 1**: Làm quen với quy trình (diễn ra trong 2 tuần )

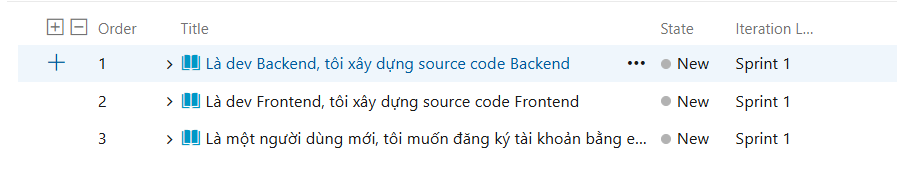
Trong sprint đầu tiên, Dự án xây dựng nền tảng hệ thống bằng việc phát triển source code cho front-end và back-end. Đây là các User Story quan trọng để chuẩn bị *cơ sở* cho những chức năng phức tạp hơn sau này. Vậy sprint 1 sẽ gồm có hai user story là:

**Là dev Backend, tôi xây dựng source code Backend**

**Là dev Frontend, tôi xây dựng source code Frontend**

Ngoài ra bổ sung thêm một user story là

**Là một người dùng mới, tôi muốn đăng ký tài khoản bằng email hoặc số điện thoại để bắt đầu sử dụng ví điện tử.**

****

**Sprint 2**: Tăng tốc độ phát triển tính năng (diễn ra trong 2 tuần )

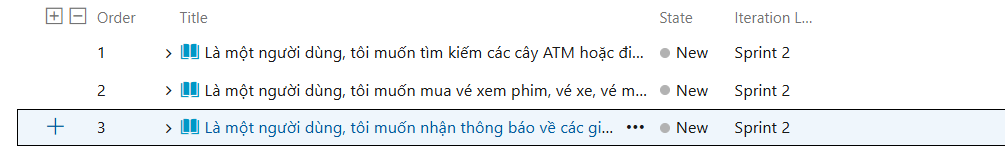
Trong Sprint này tập trung tăng cường xây dựng các tính năng thông thường, không cốt lõi của dự án

Bao gồm:

**Là một người dùng, tôi muốn tìm kiếm các cây ATM hoặc điểm chấp nhận thanh toán ví điện tử gần vị trí của mình.**

**Là một người dùng, tôi muốn mua vé xem phim, vé xe, vé máy bay và đặt các dịch vụ khác trực tiếp từ ví điện tử để tiết kiệm thời gian.**

**Là một người dùng, tôi muốn nhận thông báo về các giao dịch và chương trình khuyến mãi để không bỏ lỡ cơ hội.**

****

**Sprint 3**: Tập trung nhiệm vụ quan trọng và chức năng cốt lõi (diễn ra trong 2 tuần )

Tại sprint này thực hiện xây dựng các chức năng quan trọng, cốt lõi hoặc có thể phát triển từ những chức năng tại sprint 2

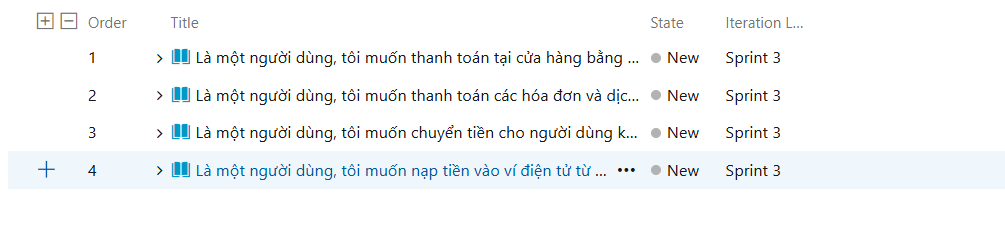
Bao gồm:

**Là một người dùng, tôi muốn thanh toán tại cửa hàng bằng cách quét mã QR để giao dịch nhanh chóng và tiện lợi.**

**Là một người dùng, tôi muốn thanh toán các hóa đơn và dịch vụ trực tiếp từ ví điện tử để tiết kiệm thời gian.**

**Là một người dùng, tôi muốn chuyển tiền cho người dùng khác trong hệ thống một cách nhanh chóng và an toàn.**

**Là một người dùng, tôi muốn nạp tiền vào ví điện tử từ tài khoản ngân hàng để có thể thanh toán và chuyển tiền.**

****

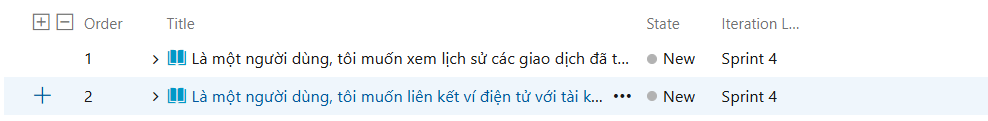
**Sprint 3**: Tăng tốc hoàn thiện và tối ưu hệ thống (diễn ra trong 2 tuần )

Sprint cuối cùng sẽ ưu tiên hoàn thành các chức năng còn lại. Bao gồm:

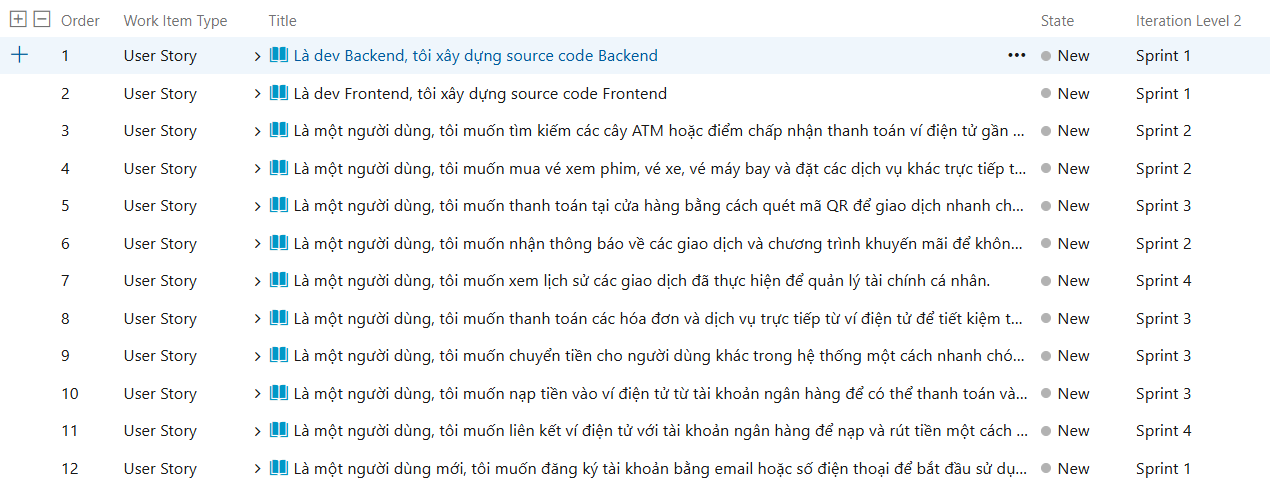
**Là một người dùng, tôi muốn xem lịch sử các giao dịch đã thực hiện để quản lý tài chính cá nhân.**

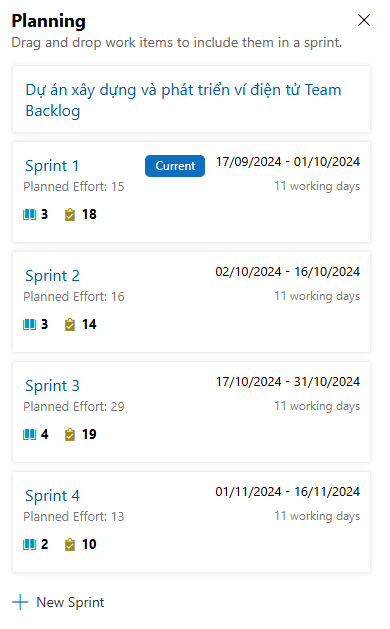
Và

**Là một người dùng, tôi muốn liên kết ví điện tử với tài khoản ngân hàng để nạp và rút tiền một cách thuận tiện.**

****

***Tổng quan***

****

****

***Task 6:*** Lập kế hoạch cho Sprint 1, bao gồm phân công công việc và thời gian ước lượng

***Bài làm***

*Mục tiêu của dự án*: Phát triển và ra mắt phiên bản đầu tiên của ví điện tử trong vòng 3 tháng

*Mục tiêu của sprint 1:* Xây dựng cơ bản được nền tảng của dự án là xây dựng source code cho Frontend và Backend, thông qua đó phát triển chức năng cơ bản đầu tiên là đăng ký tài khoản

***Phân công và ước lượng thời gian***

*Thành viên*

1. Đặng Xuân Huy: Frontend Developer
2. Huy Đặng: Backend Developer

Ước lượng cụ thể

Front-end Developer: Đặng Xuân Huy

* **Thiết lập source code: 4 giờ**
* *Giải thích:* Bước đầu tiên là thiết lập repository, khởi tạo cấu trúc thư mục, và cấu hình các file cơ bản như package.json, webpack (hoặc tương tự) cho dự án ví điện tử. Cấu hình môi trường làm việc và kiểm thử các chức năng cơ bản. Điều này quan trọng để thiết lập nền tảng phát triển chắc chắn.
* **Tích hợp theme và giao diện front-end: 8 giờ**
* *Giải thích:* Tích hợp theme cho ứng dụng ví điện tử bao gồm tùy chỉnh CSS và cấu trúc layout để phù hợp với giao diện người dùng (UI/UX). Các thành phần chính như trang đăng nhập, bảng điều khiển (dashboard) tài khoản, giao dịch, và nạp/rút tiền cần được thiết kế hợp lý, đảm bảo tính tương tác tốt và responsive cho nhiều thiết bị.
* **Cài đặt thư viện cần thiết: 4 giờ**
* *Giải thích:* Cài đặt và cấu hình các thư viện front-end như Redux để quản lý trạng thái, Axios để xử lý các yêu cầu API, và Router để điều hướng. Đảm bảo các thư viện này tích hợp ổn định để xử lý các chức năng quan trọng như hiển thị số dư, lịch sử giao dịch và thông báo.
* **Xây dựng Navbar: 6 giờ**
* *Giải thích:* Navbar của ứng dụng ví điện tử cần có các mục như Trang chủ, Lịch sử giao dịch, Nạp tiền, Rút tiền, và Quản lý tài khoản. Thiết kế phải trực quan và dễ sử dụng, với các biểu tượng phù hợp và khả năng responsive, giúp người dùng dễ dàng điều hướng trên nhiều thiết bị.
* **Xây dựng Footer: 4 giờ**
* *Giải thích:* Footer bao gồm thông tin về công ty, điều khoản sử dụng, chính sách bảo mật, và các liên kết quan trọng. Thiết kế Footer đơn giản nhưng phải đảm bảo tính thẩm mỹ và hiển thị tốt trên các màn hình khác nhau.
* **Triển khai front-end và CI/CD: 8 giờ**
* *Giải thích:* Deploy front-end của ứng dụng ví điện tử lên các nền tảng như Vercel hoặc Netlify, thiết lập quy trình CI/CD tự động hóa deploy khi có commit mới. Đảm bảo môi trường chạy ổn định và xử lý các vấn đề phát sinh khi tích hợp CI/CD.

Backend Developer: Huy Đặng

* **Xây dựng cơ sở dữ liệu: 6 giờ**
* *Giải thích:* Thiết kế cơ sở dữ liệu cho ví điện tử bao gồm các bảng như người dùng (users), giao dịch (transactions), ví tiền (wallets), và lịch sử hoạt động (activity logs). Định nghĩa các mối quan hệ giữa chúng và thiết lập cơ sở dữ liệu với các bảng khóa chính, khóa ngoại. Kiểm tra cấu trúc với dữ liệu mẫu để đảm bảo hoạt động trơn tru.
* **Thiết lập môi trường phát triển: 4 giờ**
* *Giải thích:* Thiết lập môi trường phát triển backend bằng Node.js và Express (hoặc tương tự), cùng với các công cụ hỗ trợ như Docker để mô phỏng môi trường thực tế, Postman để kiểm tra API. Bước này quan trọng để đảm bảo việc phát triển dễ dàng và giảm thiểu lỗi.
* **Kiểm tra kết nối cơ sở dữ liệu: 4 giờ**
* *Giải thích:* Sau khi thiết lập cơ sở dữ liệu, kiểm tra kết nối với backend để đảm bảo các giao dịch như nạp/rút tiền, truy xuất số dư, và lưu trữ lịch sử giao dịch hoạt động chính xác. Viết mã kết nối và thực hiện các yêu cầu nhỏ để đảm bảo mọi thứ đều hoạt động như mong đợi.
* **Thiết lập source code và cấu hình thư viện: 6 giờ**
* *Giải thích:* Cấu hình các thư viện backend như Express, Mongoose (hoặc Sequelize nếu dùng SQL), JWT (cho xác thực token), và thiết lập các API chính như đăng ký, đăng nhập, thực hiện giao dịch, và kiểm tra số dư ví. Đảm bảo routing và middleware được thiết lập chính xác để xử lý luồng dữ liệu và bảo mật.
* **Tạo dữ liệu seed: 4 giờ**
* *Giải thích:* Tạo dữ liệu seed cho các bảng như người dùng, giao dịch mẫu, và ví để thử nghiệm hệ thống. Quá trình này giúp kiểm tra tính năng và độ ổn định của hệ thống trước khi triển khai chính thức. Vì dữ liệu nhỏ nên chỉ tốn khoảng 4 giờ.
* **Deploy backend: 6 giờ**
* *Giải thích:* Deploy backend lên các nền tảng như Heroku, AWS, hoặc DigitalOcean, thiết lập môi trường production và kết nối cơ sở dữ liệu. Kiểm tra hệ thống trong môi trường thực tế để đảm bảo backend xử lý giao dịch và truy xuất dữ liệu một cách chính xác, nhanh chóng.
* **Thiết lập CI/CD cho backend: 8 giờ**
* *Giải thích:* Cấu hình CI/CD cho backend bao gồm tự động hóa quy trình build và deploy khi có commit mới. Điều này giúp hệ thống luôn cập nhật nhanh chóng và giảm thiểu downtime. CI/CD cũng xử lý các trường hợp rollback khi gặp sự cố để đảm bảo hệ thống luôn ổn định.

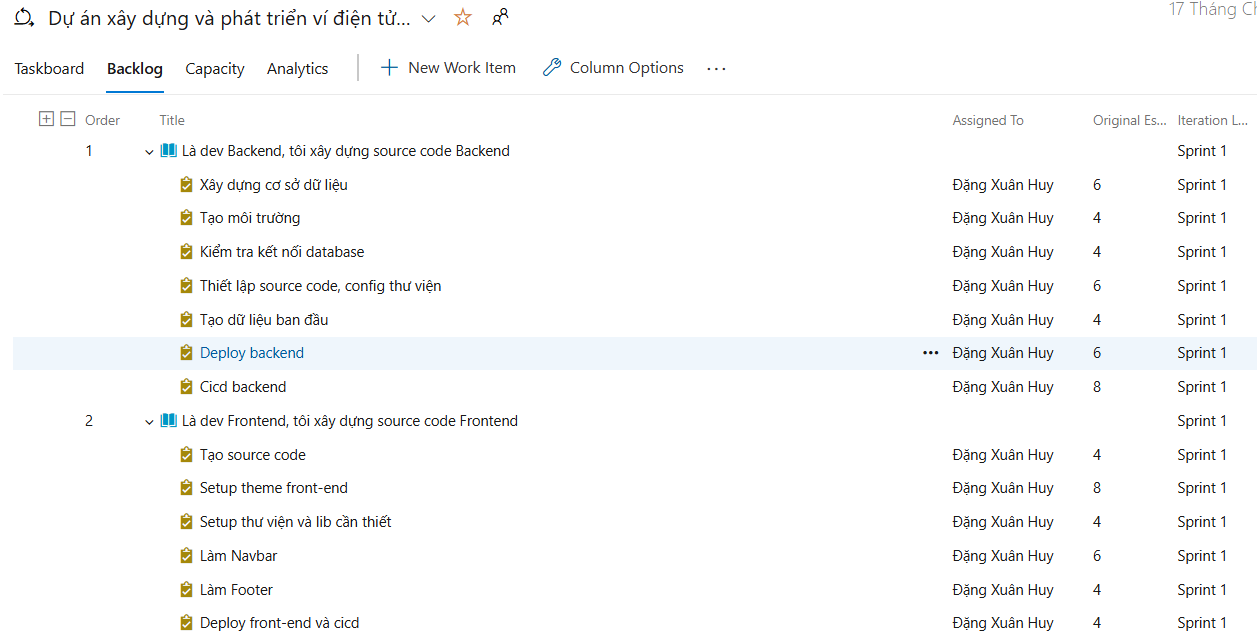
Xây dựng chức năng đăng ký: Đặng Xuân Huy (11 giờ ) và Huy Đặng (14 giờ )

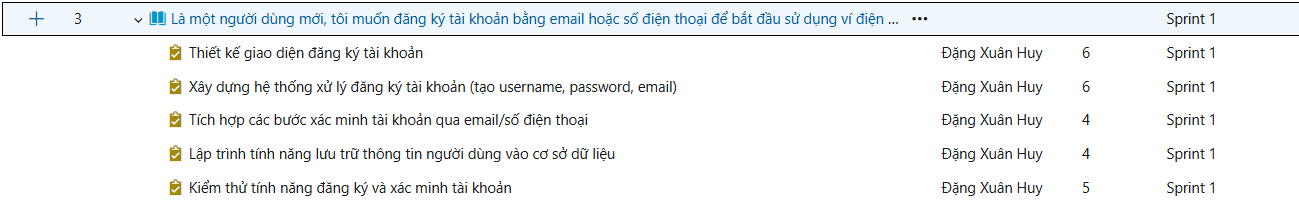
* **. Thiết kế giao diện đăng ký tài khoản: 6 giờ (Đặng Xuân Huy)**
* Giải thích: Tạo mockup giao diện đăng ký với các trường thông tin cần thiết như: họ tên, email, số điện thoại, mật khẩu, xác nhận mật khẩu. Xây dựng trải nghiệm người dùng mượt mà, kiểm tra tính thẩm mỹ và độ thân thiện với người dùng.
* **Xây dựng hệ thống xử lý đăng ký tài khoản: 6 giờ (Huy Đặng)**
* Giải thích**:** Phát triển các API backend xử lý đăng ký tài khoản bao gồm tạo username, mật khẩu đã mã hóa, và lưu trữ email. Kiểm tra tính hợp lệ của dữ liệu đầu vào và xử lý các lỗi liên quan như trùng lặp tài khoản.
* **Tích hợp các bước xác minh tài khoản qua email/số điện thoại: 4 giờ (Huy Đặng)**
* Giải thích**:** Tạo chức năng gửi mã xác minh (OTP) qua email hoặc SMS. Xây dựng giao diện người dùng để nhập mã OTP và xác minh tài khoản. Kiểm tra tính bảo mật và xử lý trường hợp người dùng không nhận được OTP hoặc nhập sai nhiều lần.
* **Lập trình tính năng lưu trữ thông tin người dùng vào cơ sở dữ liệu: 4 giờ (Huy Đặng)**
* Giải thích: Xây dựng cơ sở dữ liệu cho thông tin người dùng, bao gồm các trường như họ tên, email, số điện thoại, mật khẩu đã mã hóa và trạng thái tài khoản (đã xác minh hoặc chưa). Tạo các mối quan hệ giữa bảng người dùng và các bảng liên quan khác như lịch sử giao dịch, hoạt động.
* **5. Kiểm thử tính năng đăng ký và xác minh tài khoản: 5 giờ (Đặng Xuân Huy)**
* Giải thích**:** Thực hiện kiểm thử đơn vị (unit test) cho từng chức năng đăng ký và xác minh tài khoản. Kiểm thử tích hợp (integration test) toàn bộ quy trình đăng ký và xác minh để đảm bảo hệ thống hoạt động trơn tru và không có lỗi.

**Tổng kết thời gian ước lượng**

Đặng Xuân Huy: 45 giờ

Huy Đặng: 52 giờ





Do dự án LAB1 được thực hiện cá nhân, nên giao diện Devops ở trên phần Assigned to không thể phân công hai người là Đặng Xuân Huy và Huy Đặng như báo cáo. Xin được thông cảm cho quá trình này ạ