**TRƯỜNG ĐẠI HỌC AN GIANG  
KHOA CÔNG NGHỆ THÔNG TIN**

****

**ĐỒ ÁN**

**QUẢN LÝ DỰ ÁN PHẦN MỀM**

**SOCIAL E-COMMERCE HUB**

**WEBSITE TRUNG TÂM THƯƠNG MẠI ĐIỆN TỬ XÃ HỘI**

**HỌ TÊN SINH VIÊN**

**DTH225650 - Huỳnh Quốc Huy**

**DTH225634 - Phan Chí Hải**

**DTH225647 - Bùi Nguyễn Minh Huy**

**DTH225654 - Nguyễn Phước Huy**

**DTH225671 - Nguyễn Vũ Kha**

**HỌ TÊN GIẢNG VIÊN**

**Ts. Nguyễn Văn Hòa**

**AN GIANG, THÁNG 9 NĂM 2025**

**TRƯỜNG ĐẠI HỌC AN GIANG**

**KHOA CÔNG NGHỆ THÔNG TIN**

****

**ĐỒ ÁN**

**QUẢN LÝ DỰ ÁN PHẦN MỀM**

**SOCIAL E-COMMERCE HUB**

**WEBSITE TRUNG TÂM THƯƠNG MẠI ĐIỆN TỬ XÃ HỘI**

**HỌ TÊN SINH VIÊN**

**DTH225650 - Huỳnh Quốc Huy**

**DTH225634 - Phan Chí Hải**

**DTH225647 - Bùi Nguyễn Minh Huy**

**DTH225654 - Nguyễn Phước Huy**

**DTH225671 - Nguyễn Vũ Kha**

**HỌ TÊN GIẢNG VIÊN HƯỚNG DẪN**

**Ts. Nguyễn Văn Hòa**

**AN GIANG, THÁNG 9 NĂM 2025**

.............................................................................................................

**Giảng viên hướng dẫn**

( Ký và ghi rõ họ tên)

Nội dung nhận xét:

* **Đồng ý** hay **Không đồng ý**  cho sinh viên báo cáo TTCK; Nếu không đồng ý cần ghi rõ lý do.
* Kết quả được so với yêu cầu;
* Ý kiến khác (nếu có)

**MỤC LỤC**

[**Lời mở đầu** 4](#_Toc208834864)

[**1.** **TỔNG QUAN DỰ ÁN** 4](#_Toc208834865)

[**1.1.** **Bối Cánh Và Tầm Nhìn Dự Án** 4](#_Toc208834866)

[**1.1.1.** **Giới thiệu đề tài: Nền tảng E-commerce kết hợp mạng xã hội (Social Commerce)** 4](#_Toc208834867)

[**1.1.2.** **Mục tiêu dự án (Object Objectives)** 5](#_Toc208834868)

[**1.2.** **Phương Pháp Luận Quản Lý Dự Án** 5](#_Toc208834869)

[**1.2.1.** **Giới thiệu về Agile/Scrum** 6](#_Toc208834870)

[**1.2.2.** **Quy trình Scrum và các sự kiện chính**. 6](#_Toc208834871)

[**1.2.3.** **Các vai trò và trách nhiệm trong nhóm 5 người** 6](#_Toc208834872)

[**1.3.** **Các Công Nghệ Và Công Cụ Quản Lý Được Áp Dụng** 6](#_Toc208834873)

[**1.3.1.** **Danh mục công nghệ PHP Laravel Framework** 6](#_Toc208834874)

[**1.3.2.** **Danh mục công cụ** 6](#_Toc208834875)

[**2.** **KHỞI TẠO VÀ LẬP KẾ HOẠCH DỰ ÁN (AGILE PLANNING)** 6](#_Toc208834876)

[**2.1.** **Quản Lý Tích Hợp Dự Án (Project Integration Management)** 6](#_Toc208834877)

[**2.1.1.** **Xây dựng hiến chương dự án** 6](#_Toc208834878)

[**2.2.** **Quản Lý Phạm Vi (Scope Management)** 6](#_Toc208834879)

[**2.2.1.** **Thu thập và phân tích yêu cầu: User Story và Product Backlog** 6](#_Toc208834880)

[**2.2.2.** **Phân rã công việc (Work Breakdown Structure - WBS)** 6](#_Toc208834881)

[**2.3.** **Quản Lý Ước Lượng (Estimation Management)** 6](#_Toc208834882)

[**2.3.1.** **Các kỹ thuật ước lượng trong Agile: Story Points** 6](#_Toc208834883)

[**2.3.2.** **Lập kế hoạch Sprint (Sprint Planning)** 6](#_Toc208834884)

[**2.4.** **Quản Lý Nguồn Nhân Lực (Human Resource Management)** 6](#_Toc208834885)

[**2.4.1.** **Kế hoạch phân công vai trò và trách nhiệm** 6](#_Toc208834886)

[**2.4.2.** **Tương tác và làm việc nhóm trong mô hình Agile** 6](#_Toc208834887)

[**3.** **PHÂN TÍCH, THIẾT KẾ VÀ TRIỂN KHAI KỸ THUẬT** 6](#_Toc208834888)

[**3.1.** **Phân Tích Và Thiết Kế Hệ Thống** 6](#_Toc208834889)

[**3.1.1.** **Sơ đồ luồng dữ liệu (Data Flow Diagram - DFD)** 6](#_Toc208834890)

[**3.1.2.** **Sơ đồ quan hệ thực thể (Entity Relationship Diagram – ERD)** 6](#_Toc208834891)

[**3.1.3.** **Sơ đồ trường hợp sử dụng (Use Case Diagram)** 6](#_Toc208834892)

[**3.1.4.** **Sơ đồ lớp (Class Diagram)** 6](#_Toc208834893)

[**3.2.** **Triển khai phát triển Website** 6](#_Toc208834894)

[**3.2.1.** **Quy trình làm việc theo Sprint** 6](#_Toc208834895)

[**3.2.2.** **Quản lý mã nguồn và làm việc nhóm với Git** 6](#_Toc208834896)

[**3.3.** **Quản lý tài liệu (Documentation Management)** 6](#_Toc208834897)

[**3.3.1.** **Xây dựng kho tài liệu tập trung (Project Wiki)** 6](#_Toc208834898)

[**3.3.2.** **Hướng dẫn sử dụng Notion làm Project Wiki** 6](#_Toc208834899)

[**4.** **KIỂM SOÁT, KIỂM THỬ VÀ BÁO CÁO DỰ ÁN** 6](#_Toc208834900)

[**4.1.** **Quản lý rủi ro (Risk Management)** 6](#_Toc208834901)

[**4.1.1.** **Nhận diện các rủi ro của dự án** 6](#_Toc208834902)

[**4.1.2.** **Rủi ro đặc thù với Laravel** 6](#_Toc208834903)

[**4.1.3.** **Kế hoạch ứng phó với rủi ro** 7](#_Toc208834904)

[**4.2.** **Kiểm soát dự án (Project Control)** 7](#_Toc208834905)

[**4.2.1.** **Theo dõi tiến độ: Daily Scrum và Biểu đồ Burndown** 7](#_Toc208834906)

[**4.2.2.** **Kiểm soát thay đổi trong Agile** 7](#_Toc208834907)

[**4.3.** **Kế hoạch và thực thi kiểm thử (Software Testing)** 7](#_Toc208834908)

[**4.3.1.** **Xây dựng kế hoạch kiểm thử (Test Plan)** 7](#_Toc208834909)

[**4.3.2.** **Viết Test Case cho ứng dụng web** 7](#_Toc208834910)

[**4.3.3.** **Hướng dẫn sử dụng Apace Jmeter cho Performance Testing** 7](#_Toc208834911)

[**4.4.** **Kế hoạch và thực thi kiểm thử (Software Testing)** 7](#_Toc208834912)

[**4.4.1.** **Báo cáo tình trạng dự án** 7](#_Toc208834913)

[**4.4.2.** **Họp Sprint Review và Sprint Retrospective** 7](#_Toc208834914)

[**5.** **PHỤ LỤC** 7](#_Toc208834915)

[**5.1.** **Sơ đồ** 7](#_Toc208834916)

[**5.1.1.** **Sơ đồ luồng dữ liệu (DFD)** 7](#_Toc208834917)

[**5.1.2.** **Sơ đồ quan hệ thực thể (ERD)** 7](#_Toc208834918)

[**5.1.3.** **Sơ đồ trường hợp sử dụng (UseCase Diagram)** 7](#_Toc208834919)

[**5.1.4.** **Sơ đồ lớp (Class Diagram)** 7](#_Toc208834920)

[**5.2.** **Bảng phân công vai trò và trách nhiệm** 7](#_Toc208834921)

[**5.3.** **Ma trận quản lý rủi ro chi tiết** 7](#_Toc208834922)

[**5.4.** **Bảng kế hoạch kiểm thử và Test Case** 7](#_Toc208834923)

[**5.5.** **Hướng dẫn sử dụng các công cụ (Jira,Notion,Git) chi tiết** 7](#_Toc208834924)

[**6.** **TÀI LIỆU THAM KHẢO** 7](#_Toc208834925)

**Lời mở đầu**

Đồ án này trình bày về quy trình quản lý và phát triển dự án phần mềm với đề tài FriendMart - Social Commerce Hub, một website thương mại điện tử xã hội kết hợp hai tính năng chính: bán hàng và mạng xã hội. Dự án được thực hiện bởi một nhóm 5 sinh viên, sử dụng mô hình quản lý linh hoạt Agile với khung quy trình Scrum và công nghệ PHP Laravel. Mục tiêu của báo cáo là trình bày một cách toàn diện quy trình quản lý dự án từ giai đoạn khởi tạo, lập kế hoạch, triển khai, đến kiểm soát và báo cáo dự án, nhằm đảm bảo sản phẩm không chỉ đáp ứng yêu cầu kỹ thuật mà còn có giá trị áp dụng vào thực tế. Báo cáo này không những cung cấp các bước thực hiện dự án cụ thể mà còn đi sâu vào phân tích kế hoạch và đưa ra các quyết định cho việc quản lý dự án, đặc biệt là việc lựa chọn mô hình Agile để giải quyết tính phức tạp và thay đổi liên tục của một nền tảng kết hợp giữa thương mại điện tử và mạng xã hội. Dự án sử dụng các công cụ quản lý dự án hàng đầu như Jira, Notion, quản lý mã nguồn qua Git & GitHub và các công cụ kiểm thử chuyên dụng như PHPUnit, Apache Jmeter một cách có hệ thống, nhằm phản ánh một quy trình quản lý và làm việc logic và hiệu quả.

1. **TỔNG QUAN DỰ ÁN**
   1. **Bối Cánh Và Tầm Nhìn Dự Án**
      1. **Giới thiệu đề tài: Nền tảng E-commerce kết hợp mạng xã hội (Social Commerce)**

Dự án FriendMart được phát triển dựa trên ý tưởng đột phá của "Social Commerce", một hình thức kinh doanh trực tuyến đang trở thành xu hướng chủ đạo bằng cách kết hợp sự kết nối mạnh mẽ của mạng xã hội và thương mại điện tử [1]. Điều này cho phép người dùng mua sắm trực tiếp trên các nền tảng mạng xã hội hoặc một nền tảng chuyên biệt. Từ đó FriendMart được thiết kế để trở thành một trung tâm thương mại điện tử xã hội, nơi người dùng có thể mua sắm, chia sẻ và tương tác trực tiếp trên cùng một nền tảng. Nền tảng này được xây dựng nhằm đáp ứng đầy đủ các chức năng bán hàng hay mua sắm như quản lý sản phẩm, giỏ hàng, thanh toán và quản lý đơn hàng mà còn cho phép người dùng tương tác với nhau thông qua các hoạt động như tạo hồ sơ cá nhân, kết bạn, theo dõi shop, chia sẻ sản phẩm, đánh giá và bình luận, ngoài ra còn kết hợp chức năng trang tin qua việc quản lý chủ đề và bài viết . Mục đích chính là biến trải nghiệm mua sắm từ một giao dịch đơn thuần thành một hoạt động mang tính cộng đồng, nơi người dùng có thể chia sẻ và tương tác xoay quanh sở thích mua sắm của họ [2]. Từ đó, việc mua sắm không chỉ là việc lựa chọn và thêm vào giỏ hàng mà còn là nơi mọi người có thể giao tiếp, xây dựng mối quan hệ với nhau.

Việc lựa chọn đề tài này được đánh giá là một thách thức hợp lý đối với một đồ án môn học. Dự án đòi hỏi phải quản lý nhiều luồng dữ liệu phức tạp, từ thông tin sản phẩm và đơn hàng cho đến các tương tác xã hội như bài đăng và tin nhắn. Điều này tạo điều kiện lý tưởng để áp dụng các kỹ thuật quản lý dự án phần mềm tiên tiến, bao gồm phân tích yêu cầu đa luồng và thiết kế cơ sở dữ liệu quan hệ phức tạp

* + 1. **Mục tiêu dự án (Object Objectives)**

Tầm nhìn tổng quát của dự án là "FriendMart - Where Shopping Meets Friendship", dự án hướng tới việc trở thành một nền tảng thương mại điện tử và kết nối xã hội qua việc mua sắm, nơi những trao đổi không chỉ là giao dịch mà còn là một trải nghiệm xã hội, kết nối mọi người. Điều đó có nghĩa ngoài mua và bán, nền tảng này có thể giúp mọi người kết bạn, tương tác với các bài viết, các sản phẩm, tìm ra những món hàng có lượt thích chung với nhau hay chia sẻ những trải nghiệm sản phẩm qua các bài viết, hình ảnh... Ý tưởng độc đáo này đảm bảo một dự án đầy hứa hẹn, một trang web bán hàng đầy sinh động và có tính kết nối mọi người.

Các mục tiêu của dự án được xác định theo mô hình SMART để đảm bảo tính khả thi và đo lường được. Thứ nhất, về tiêu chí Specific (Cụ thể), chức năng và mục đích của trang web đã được xác định rõ ràng. Dự án này sẽ phát triển một website bán hàng tích hợp mạng xã hội, có nhiều chức năng cốt lõi như quản lý sản phẩm, danh mục, thương hiệu, hóa đơn... Ngoài ra, người dùng có thể thích, bình luận và chia sẻ sản phẩm cho bạn bè. Bên cạnh đó là các chức năng mạng xã hội cơ bản như đăng bài viết, bình luận, thích, chia sẻ, kết bạn, theo dõi, và các tính năng mở rộng như vận chuyển và trang tin tức cũng được đưa vào dự án. Tiếp theo, về tiêu chí Measurable (Đo lường được), dự án đặt ra các thước đo rõ ràng. Với đội ngũ 5 thành viên, mục tiêu là hoàn thành đầy đủ số lượng chức năng cơ bản đã xác định, đảm bảo sản phẩm chạy trơn tru, không có lỗi nghiêm trọng, và đạt tỷ lệ thành công 95% trong các test case đã được định nghĩa cho các chức năng cốt lõi. Toàn bộ công việc sẽ được hoàn thành trong 3-4 Sprint, mỗi Sprint kéo dài 2 tuần, để đáp ứng tiến độ trước buổi báo cáo đồ án. Về tiêu chí Achievable (Khả thi), mục tiêu này là hoàn toàn khả thi vì đội ngũ 5 thành viên đều có kiến thức nền tảng về công nghệ web PHP Laravel đây là một môn học song song với môn quản lý dự án phần mềm hiện tại và có khả năng cao xây dựng thành công dự án. Tiêu chí Relevant (Liên quan) cũng được đáp ứng, khi dự án phục vụ trực tiếp cho mục tiêu học tập, giúp các thành viên áp dụng kiến thức lý thuyết về quản lý dự án vào thực tế tại môi trường học thuật. Cuối cùng, tiêu chí Time-bound (Có thời hạn) được tuân thủ nghiêm ngặt, với cam kết hoàn thành dự án trong khoảng thời gian từ 6 đến 8 tuần để đáp ứng tiến độ môn học.

* 1. **Phương Pháp Luận Quản Lý Dự Án**
     1. **Giới thiệu về Agile/Scrum**

Agile là một phương pháp phát triển phần mềm linh hoạt, tập trung vào việc thích ứng nhanh với thay đổi từ khách hàng hoặc các bên liên quan và tăng cường sự hợp tác trong nhóm. Phương pháp này khác với các mô hình truyền thống như thác nước (Waterfall), vốn yêu cầu kế hoạch cố định cứng nhắc , hoàn thành từng giai đoạn tuần tự từng bước và không thể quay lại[3]. Thay vào đó, Agile chia dự án thành các chu kỳ lặp lại ngắn, được gọi là Sprint, trong dự án hiện tại được xác định trong khoảng thời gian kéo dài từ 2 đến 4 tuần. Mục tiêu của mỗi chu kỳ là tạo ra một phần mềm hoạt động được, sẵn sàng để bàn giao cho khách hàng hoặc các bên liên quan [4].

Để đạt được mục tiêu này, Agile tuân thủ một bộ 12 nguyên tắc cốt lõi. Nguyên tắc đầu tiên và quan trọng nhất là làm hài lòng khách hàng bằng cách cung cấp sớm và liên tục các sản phẩm có giá trị. Bên cạnh đó, Agile luôn chào đón các yêu cầu thay đổi, thậm chí là ở giai đoạn cuối của quá trình phát triển, vì những thay đổi này có thể mang lại lợi thế cạnh tranh cho sản phẩm [5]. Để đảm bảo sự tiến bộ, Agile ưu tiên việc bàn giao phần mềm hoạt động được cho khách hàng trong các khoảng thời gian ngắn, thường từ vài tuần đến vài tháng. Quá trình này được xây dựng từ các thành viên có năng lực và động lực tiếp nhận thay đổi và được cung cấp môi trường và sự hỗ trợ cần thiết để hoàn thành công việc có kỷ luật và tự chử. Giao tiếp gặp gỡ trực tiếp là phương pháp hiệu quả nhất để truyền đạt thông tin, thúc đẩy sự minh bạch và cộng tác. Qua đó, các doanh nhân, đối tác và các kỹ sư phần mềm cần làm việc cùng nhau thường xuyên trong suốt dự án.

Ngoài các nguyên tắc trên, Agile còn nhấn mạnh tầm quan trọng của việc duy trì sự phát triển bền vững và liên tục. Phần mềm hoạt động tốt chính là có sự tiến độ trong quá trình phát triển, chứ không phải dựa vào các tài liệu. Sự chú ý liên tục đến các yếu tố kỹ thuật và thiết kế sẽ cải thiện sự chỉnh chu và linh hoạt của sản phẩm. Xa hơn nữa Agile đề cao sự đơn giản và tối đa hóa lượng công việc không cần thiết. Các phân tích và thiết kế tốt nhất được tạo ra từ những nhóm có khả năng tổ chức cao. Cuối cùng, để thúc đẩy cải tiến liên tục, nhóm sẽ thường xuyên điều chỉnh cách làm việc của mình để đạt hiệu quả cao hơn [6]. Với tính chất phức tạp và có nhiều luồng dữ liệu của dự án FriendMart - Social Commerce Hub Website, các yêu cầu của dự án có khả năng thay đổi cao hoặc không được xác định rõ ràng ngay từ đầu. Việc áp dụng Agile cho phép nhóm phát triển nhanh chóng điều chỉnh dựa trên phản hồi thực tế, đảm bảo sản phẩm cuối cùng phù hợp hơn với nhu cầu thực tế của người dùng.

Ngoài 12 nguyên tắc, Tuyên ngôn Agile còn đề ra bốn giá trị cốt lõi [7], tập trung vào việc ưu tiên con người và sự tương tác hơn các quy trình và công cụ. Thay vì chỉ dựa vào tài liệu chi tiết, Agile đánh giá tiến độ bằng phần mềm hoạt động tốt [8]. Mối quan hệ hợp tác với khách hàng được xem trọng hơn việc đàm phán hợp đồng. Qua đó, Agile tập trung vào việc phản hồi linh hoạt với sự thay đổi, thay vì bám theo một kế hoạch đã định sẵn. Những giá trị này là phương hướng cho cách thức quản lý dự án, đặc biệt trong một dự án có tính học thuật như FriendMart, nơi sự linh hoạt là chìa khóa để xử lý các vấn đề phát sinh không lường trước [8].

* + 1. **Quy trình Scrum và các sự kiện chính**.

Scrum là một khung quy trình phổ biến và hiệu quả nhất trong Agile, được thiết kế để quản lý các dự án phức tạp thông qua một phương pháp tiếp cận lặp lại và tiệm tiến. Toàn bộ quy trình Scrum xoay quanh một chu kỳ lặp đi lặp lại có thời gian cố định gọi là Sprint, thường kéo dài từ 2-4 tuần [9]. Trong mỗi Sprint, nhóm phát triển sẽ tạo ra một "phần tăng dần" (Increment) có thể chuyển giao, tức là một phần mềm hoạt động được, sẵn sàng cho khách hàng.

Một Sprint bắt đầu bằng cuộc họp Sprint Planning [10]. Đây là một cuộc họp hợp tác giữa Product Owner và Development Team, nơi cả nhóm thảo luận và thống nhất về Mục tiêu của Sprint hay Sprint Goal và chọn một tập hợp các User Story từ Product Backlog để đưa vào Sprint Backlog [11]. Cuộc họp này đóng vai trò là lộ trình cho Sprint sắp tới, cung cấp phương hướng và trọng tâm rõ ràng cho nhóm phát triển. Khi một Sprint đang diễn ra, nhóm sẽ tổ chức cuộc họp ngắn hàng ngày gọi là Daily Scrum [6]. Cuộc họp này kéo dài không quá 15 phút, nhằm mục đích cập nhật nhanh chóng tiến độ công việc và giải quyết các trở ngại. Mỗi thành viên sẽ trả lời ba câu hỏi chính: "Hôm qua tôi đã làm gì?", "Hôm nay tôi sẽ làm gì?", và "Tôi có đang gặp khó khăn nào không?". Đây là cơ chế cốt lõi để duy trì sự minh bạch và đảm bảo mọi người đều biết công việc của nhau, từ đó dễ dàng đưa ra kế hoạch cho ngày làm việc.

Khi Sprint kết thúc, nhóm sẽ tổ chức cuộc họp Sprint Review để trình bày sản phẩm đã hoàn thành cho các bên liên quan. Đây không chỉ là một buổi trình diễn, mà là cơ hội để nhóm nhận phản hồi và xác nhận rằng sản phẩm đang đi đúng hướng. Dựa trên kết quả này, Product Owner có thể điều chỉnh hoặc sắp xếp lại thứ tự ưu tiên của Product Backlog cho các Sprint tiếp theo. Cuối cùng, sau Sprint Review là cuộc họp Sprint Retrospective. Đây là một cuộc họp nội bộ quan trọng của nhóm để cùng nhau nhìn lại quá trình làm việc. Nhóm sẽ thảo luận về những điều đã làm tốt, những điều cần cải thiện, và thống nhất các cam kết thay đổi cho Sprint tiếp theo, đảm bảo sự cải tiến liên tục trong quy trình làm việc.

* + 1. **Các vai trò và trách nhiệm trong nhóm 5 người**

Để áp dụng Scrum hiệu quả,nhóm 5 thành viên chúng tôi phân chia cụ thể các vai trò trong suốt quá trình dự án, đồng thời vẫn duy trì tinh thần làm việc nhóm và hỗ trợ lẫn nhau. Các vai trò này bao gồm Scrum Master, Product Owner và Development Team.

Trong đó Scrum Master hay Trưởng nhóm là vai trò đảm bảo cả nhóm tuân thủ đúng quy trình Scrum, mà còn hoạt động như một người phục vụ lãnh đạo (servant leader). Scrum Master chịu trách nhiệm giải quyết các rào cản, tạo điều kiện cho các cuộc họp và bảo vệ nhóm khỏi các yếu tố gây xao nhãng.  Bên cạnh đó 1 vai trò quan trọng là Product Owner Chủ sản phẩm: Đây là người chịu trách nhiệm về yêu cầu sản phẩm, quản lý và sắp xếp thứ tự ưu tiên cho Product Backlog. Product Owner là cầu nối giữa "khách hàng" trong ngữ cảnh của đồ án hiện tại là giảng viên và nhóm phát triển phải đảm bảo rằng các công việc được thực hiện luôn mang lại giá trị tốt nhất có thể cho sản phẩm. Hơn nữa 1 thành phần không thể thiếu là đội ngũ Development Team hay nhóm phát triển đây là tất cả các thành viên còn lại, chịu trách nhiệm trực tiếp trong việc phân tích, thiết kế, lập trình và kiểm thử sản phẩm. Trong mô hình Scrum, nhóm phát triển là một nhóm tự quản lý và đa chức năng. Điều này có nghĩa là nhóm có khả năng tự tổ chức để xác định cách thức tốt nhất để hoàn thành công việc, và các thành viên có thể hỗ trợ lẫn nhau bất kể vai trò chuyên môn của họ. Cụ thể trong dự án FriendMart, nhóm được chia nhỏ ra các vai trò kỹ thuật khác nhau.

Đầu tiền là Backend Developer phụ trách phát triển phần lõi của ứng dụng, bao gồm logic, API và cơ sở dữ liệu bằng framwork Laravel. Song với đó nhóm Frontend Developer sẽ chịu trách nhiệm xây dựng giao diện người dùng bằng HTML, CSS, và JavaScript tương thích với hạ tầng Backend sau khi đã có một nền backend cơ sở, việc phát triển cả hai sẽ chạy song song và cập nhật chéo cho lẫn nhau. Tester & Database Designer đảm nhận việc thiết kế cơ sở dữ liệu và xây dựng kế hoạch kiểm thử cho các giai đoạn sản phẩm được hoàn thành cho đến khi hoàn tất phần mềm. Mặc dù có sự phân công chuyên môn, các thành viên được khuyến khích làm việc chéo chức năng, hỗ trợ nhau khi cần để đảm bảo tiến độ chung của dự án.

* 1. **Các Công Nghệ Và Công Cụ Quản Lý Được Áp Dụng**
     1. **Danh mục công nghệ PHP Laravel Framework**

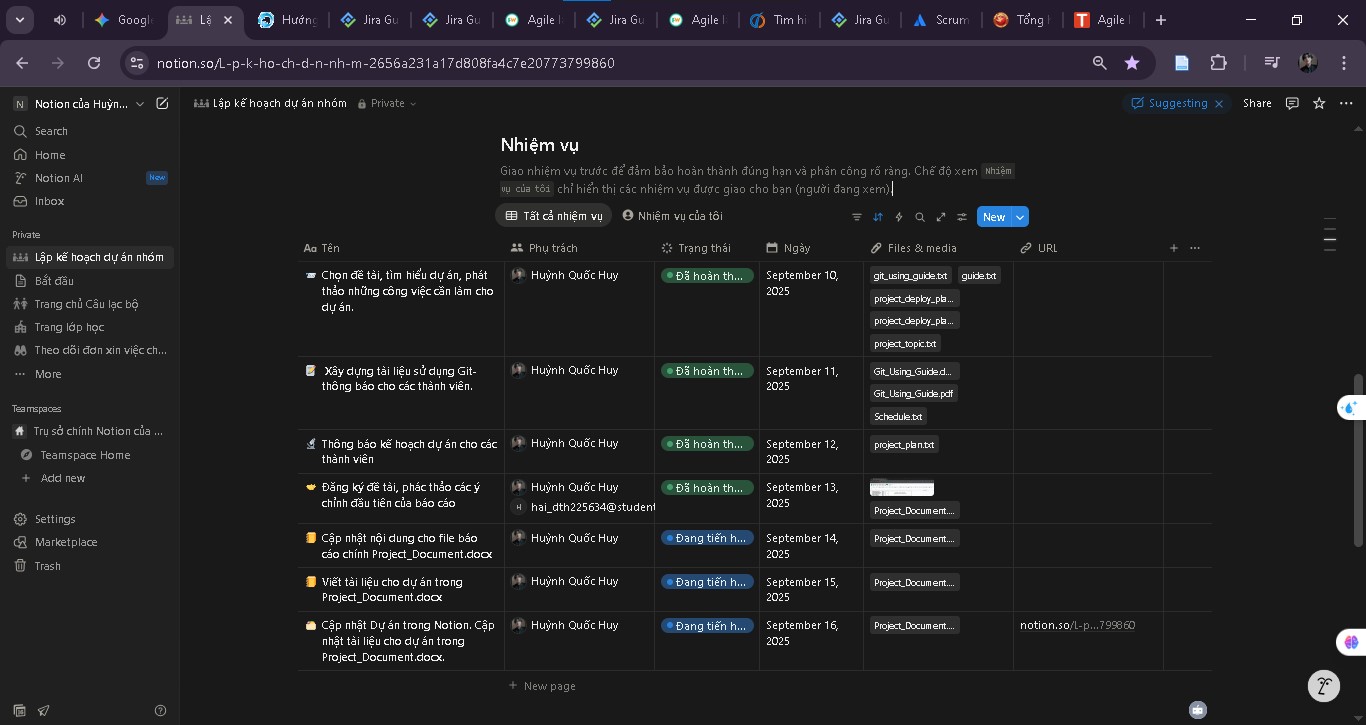
Dự án FriendMart được xây dựng trên nền tảng PHP sử dụng những ưu điểm của framework Laravel, một lựa chọn chiến lược phù hợp với kiến thức và kinh nghiệm sẵn có của toàn bộ đội ngũ phát triển. Laravel không chỉ là một framework phát triển web thông thường mà còn là một hệ sinh thái mạnh mẽ, được thiết kế để xây dựng các ứng dụng phức tạp một cách hiệu quả và có cấu trúc. Laravel sử dụng kiến trúc Model-View-Controller (MVC) để phân tách logic ứng dụng, giúp tổ chức mã nguồn một cách có hệ thống và dễ dàng bảo trì. Framework này cũng cung cấp nhiều công cụ tích hợp sẵn để quản lý cơ sở dữ liệu (Database Migration và Eloquent ORM), giúp nhóm dễ dàng tương tác với dữ liệu và thực hiện các thay đổi schema. Hơn nữa, Laravel được trang bị các tính năng bảo mật mạnh mẽ ngay từ ban đầu, chẳng hạn như CSRF token để bảo vệ các form và Eloquent ORM để ngăn chặn các cuộc tấn công SQL Injection. Với những tính năng này, Laravel tạo ra một nền tảng vững chắc và an toàn để phát triển một ứng dụng đa tính năng như FriendMart..

* + 1. **Danh mục công cụ**

Để quản lý một dự án phức tạp và nhiều tính năng như FriendMart, cần áp dụng một số công cụ để phát triển hiệu quả hơn. Các công cụ này được lựa chọn để hỗ trợ từng khía cạnh của quy trình quản lý Agile, từ lập kế hoạch, phát triển, đến kiểm thử và theo dõi tiến độ.

Đầu tiên, Jira Software đóng vai trò là công cụ quản lý dự án trung tâm, ở đây toàn bộ công việc được tập trung và theo dõi. Jira không chỉ giúp nhóm quản lý các danh sách công việc tồn đọng Product Backlog và Sprint Backlog mà còn cho phép theo dõi tiến độ của từng nhiệm vụ một cách trực quan thông qua các bảng (Board) và biểu đồ (như Burndown Chart) ,[12], [13],[14].

Tiếp theo, công cụ Notion được sử dụng như một không gian làm việc đa năng, trở thành kho tài liệu tập trung của dự án hay còn được gọi là Project Wiki. Tại đây, nhóm sẽ lưu trữ tất cả các tài liệu quan trọng, từ ghi chú cuộc họp, tài liệu thiết kế hệ thống, đến các quyết định kỹ thuật, giúp mọi thành viên dễ dàng truy cập và chia sẻ thông tin [15].



Về khía cạnh quản lý mã nguồn, Git được chọn làm hệ thống kiểm soát phiên bản phân tán. Git cho phép cả 5 thành viên trong nhóm cộng tác trên cùng một codebase một cách an toàn và hiệu quả. Nhóm sẽ tuân thủ một quy trình làm việc chuẩn mực, trong đó mỗi thành viên tạo một nhánh riêng cho tính năng hoặc sửa lỗi của mình, sau đó sử dụng Pull Request (PR) để yêu cầu hợp nhất code vào nhánh chính, đảm bảo chất lượng code và giảm thiểu xung đột.

Bên cạnh đó, các công cụ khác cũng được áp dụng cho từng mục đích chuyên biệt. WBS (Work Breakdown Structure) không phải là một phần mềm mà là một kỹ thuật quản lý giúp phân rã toàn bộ phạm vi công việc của dự án thành các gói công việc nhỏ hơn và dễ quản lý hơn, là cơ sở cho việc lập kế hoạch và ước lượng.

MS Visio là công cụ chuyên nghiệp được sử dụng để trực quan hóa cấu trúc hệ thống, cụ thể là để vẽ các sơ đồ quan trọng như Sơ đồ Luồng dữ liệu (DFD) và Sơ đồ Thực thể-Mối quan hệ (ERD) cho website FriendMart. Cuối cùng, hai công cụ kiểm thử cốt lõi là

PHPUnit và Apache Jmeter. PHPUnit là một framework kiểm thử đơn vị và tính năng tích hợp sẵn trong Laravel, cho phép lập trình viên tự động hóa việc kiểm thử mã nguồn. Trong khi đó,

Apache Jmeter là một công cụ kiểm thử hiệu năng, được sử dụng để mô phỏng một lượng lớn người dùng truy cập cùng lúc, nhằm đánh giá khả năng chịu tải của website và xác định "điểm phá vỡ" của hệ thống trước khi đưa vào hoạt động chính thức. Sự kết hợp có hệ thống của tất cả các công cụ này tạo ra một hệ sinh thái quản lý dự án mạnh mẽ, phản ánh đúng tinh thần của Agile, nơi sự tương tác và cộng tác được ưu tiên hàng đầu, đồng thời đảm bảo sự minh bạch và nhất quán trong toàn bộ quá trình phát triển.

1. **KHỞI TẠO VÀ LẬP KẾ HOẠCH DỰ ÁN (AGILE PLANNING)**
   1. **Quản Lý Tích Hợp Dự Án (Project Integration Management)**
      1. **Xây dựng hiến chương dự án**
      2. **Khởi tạo dự án trên Jira**
   2. **Quản Lý Phạm Vi (Scope Management)**
      1. **Thu thập và phân tích yêu cầu: User Story và Product Backlog**
      2. **Phân rã công việc (Work Breakdown Structure - WBS)**
   3. **Quản Lý Ước Lượng (Estimation Management)**
      1. **Các kỹ thuật ước lượng trong Agile: Story Points**
      2. **Lập kế hoạch Sprint (Sprint Planning)**
   4. **Quản Lý Nguồn Nhân Lực (Human Resource Management)**
      1. **Kế hoạch phân công vai trò và trách nhiệm**
      2. **Tương tác và làm việc nhóm trong mô hình Agile**
2. **PHÂN TÍCH, THIẾT KẾ VÀ TRIỂN KHAI KỸ THUẬT**
   1. **Phân Tích Và Thiết Kế Hệ Thống**
      1. **Sơ đồ luồng dữ liệu (Data Flow Diagram - DFD)**
      2. **Sơ đồ quan hệ thực thể (Entity Relationship Diagram – ERD)**
      3. **Sơ đồ trường hợp sử dụng (Use Case Diagram)**
      4. **Sơ đồ lớp (Class Diagram)**
   2. **Triển khai phát triển Website**
      1. **Quy trình làm việc theo Sprint**
      2. **Quản lý mã nguồn và làm việc nhóm với Git**
   3. **Quản lý tài liệu (Documentation Management)**
      1. **Xây dựng kho tài liệu tập trung (Project Wiki)**
      2. **Hướng dẫn sử dụng Notion làm Project Wiki**
3. **KIỂM SOÁT, KIỂM THỬ VÀ BÁO CÁO DỰ ÁN**
   1. **Quản lý rủi ro (Risk Management)**
      1. **Nhận diện các rủi ro của dự án**
      2. **Rủi ro đặc thù với Laravel**
      3. **Kế hoạch ứng phó với rủi ro**
   2. **Kiểm soát dự án (Project Control)**
      1. **Theo dõi tiến độ: Daily Scrum và Biểu đồ Burndown**
      2. **Kiểm soát thay đổi trong Agile**
   3. **Kế hoạch và thực thi kiểm thử (Software Testing)**
      1. **Xây dựng kế hoạch kiểm thử (Test Plan)**
      2. **Viết Test Case cho ứng dụng web**
      3. **Hướng dẫn sử dụng Apace Jmeter cho Performance Testing**
   4. **Kế hoạch và thực thi kiểm thử (Software Testing)**
      1. **Báo cáo tình trạng dự án**
      2. **Họp Sprint Review và Sprint Retrospective**
4. **PHỤ LỤC**
   1. **Sơ đồ** 
      1. **Sơ đồ luồng dữ liệu (DFD)**
      2. **Sơ đồ quan hệ thực thể (ERD)**
      3. **Sơ đồ trường hợp sử dụng (UseCase Diagram)**
      4. **Sơ đồ lớp (Class Diagram)**
   2. **Bảng phân công vai trò và trách nhiệm**
   3. **Ma trận quản lý rủi ro chi tiết**
   4. **Bảng kế hoạch kiểm thử và Test Case**
   5. **Hướng dẫn sử dụng các công cụ (Jira,Notion,Git) chi tiết**
5. **TÀI LIỆU THAM KHẢO**

[1] “Social Commerce Là Gì? Cách Kinh Doanh Với Social Commerce.” Accessed: Sept. 15, 2025. [Online]. Available: https://mona.media/social-commerce-la-gi/

[2] “Social Commerce là gì? Làm thế nào để người kinh doanh online bắt đầu với Social Commerce? | Advertising Vietnam.” Accessed: Sept. 15, 2025. [Online]. Available: https://advertisingvietnam.com/social-commerce-la-gi-lam-the-nao-de-nguoi-kinh-doanh-online-bat-dau-voi-social-commerce-p16623

[3] Thư L. U., “Tổng quan về nguyên tắc Agile và cách áp dụng trong quản lý dự án.” Accessed: Sept. 15, 2025. [Online]. Available: https://hungviet.com.vn/tong-quan-nguyen-tac-agile-va-ap-dung

[4] Huế N. T., “Agile là gì? Hiểu về mô hình Agile để triển khai hiệu quả,” Base Blog. Accessed: Sept. 15, 2025. [Online]. Available: https://base.vn/blog/agile-la-gi/

[5] B. tập viên 01, “Phương pháp Agile là gì? Ưu nhược điểm của phương pháp Agile trong quản lý dự án,” Công ty Tư vấn Quản lý OCD. Accessed: Sept. 15, 2025. [Online]. Available: https://ocd.vn/agile-la-gi-uu-nhuoc-diem-cua-phuong-phap-agile-trong-quan-ly-du-an/

[6] “Tìm hiểu về mô hình Agile và quy trình Scrum.” Accessed: Sept. 15, 2025. [Online]. Available: https://stanford.com.vn/kien-thuc-lap-trinh/tin-chi-tiet/cagId/27/id/22564/tim-hieu-ve-mo-hinh-agile-va-quy-trinh-scrum

[7] Huyền F. H., “Agile là gì? Kiến thức đầy đủ nhất về Agile trong quản lý dự án,” Fastwork - Nền tảng quản trị & điều hành doanh nghiệp toàn diện. Accessed: Sept. 16, 2025. [Online]. Available: https://fastwork.vn/agile-la-gi/

[8] T. Blog, “Agile là gì? Scrum là gì? Các công cụ quản lý dự án theo Agile,” TopDev. Accessed: Sept. 15, 2025. [Online]. Available: https://topdev.vn/blog/agile-la-gi-scrum-la-gi/

[9] FMIT V., “Tổng hợp phương pháp Agile hiệu quả trong quản lý dự án,” Tổng hợp phương pháp Agile hiệu quả trong quản lý dự án. Accessed: Sept. 16, 2025. [Online]. Available: https://fmit.vn/tin-tuc/cac-phuong-phap-trong-quan-ly-du-an-agile-la-gi

[10] “Jira Guru | Hướng dẫn về Jira Sprint Planning - Agile Professional.” Accessed: Sept. 16, 2025. [Online]. Available: https://devsamurai.vn/jira-guru-huong-dan-ve-sprint-planning-trong-jira-agile-professional/

[11] Atlassian, “What is Scrum? [+ How to Start],” Atlassian. Accessed: Sept. 16, 2025. [Online]. Available: https://www.atlassian.com/agile/scrum

[12] Emily.Ly, “Jira Guru | Hướng dẫn về Jira Sprint Planning - Agile Professional,” DevSamurai Vietnam. Accessed: Sept. 16, 2025. [Online]. Available: https://devsamurai.vn/jira-guru-huong-dan-ve-sprint-planning-trong-jira-agile-professional/

[13] admin, “Jira Guru | Thực hành Scrum với Jira Software,” DevSamurai Vietnam. Accessed: Sept. 16, 2025. [Online]. Available: https://devsamurai.vn/thuc-hanh-scrum-voi-jira-software/

[14] “Hướng dẫn các thao tác làm việc trên Jira.” Accessed: Sept. 16, 2025. [Online]. Available: https://www.bacs.vn/vi/blog/ky-nang/huong-dan-cac-thao-tac-lam-viec-tren-jira-24233.html

[15] “Hướng dẫn cơ bản và Hướng dẫn từng bước – Cách sử dụng Notion.” Accessed: Sept. 16, 2025. [Online]. Available: https://www.notion.com/vi/help/guides