

TRƯỜNG ĐẠI HỌC TRÀ VINH
KHOA KỸ THUẬT VÀ CÔNG NGHỆ



BÁO CÁO MÔN HỌC
CÔNG NGHỆ PHẦN MỀM

TÊN ĐỀ TÀI
XÂY DỰNG WEBSITE GIỚI THIỆU ĐẶC SẢN
TỈNH TRÀ VINH

Giáo viên hướng dẫn: TS. Nguyễn Bảo Ân

Sinh viên thực hiện:

110121013	Võ Thị Diễm	DA21TTB
110121074	Diệp Tú Như	DA21TTB
110121094	Trần Văn Sang	DA21TTB

Trà Vinh, tháng 6 năm 2024

TRƯỜNG ĐẠI HỌC TRÀ VINH
KHOA KỸ THUẬT VÀ CÔNG NGHỆ



BÁO CÁO MÔN HỌC
CÔNG NGHỆ PHẦN MỀM

TÊN ĐỀ TÀI
XÂY DỰNG WEBSITE GIỚI THIỆU ĐẶC SẢN
TỈNH TRÀ VINH

Giáo viên hướng dẫn: TS. Nguyễn Bảo Ân

Sinh viên thực hiện:

110121013	Võ Thị Diễm	DA21TTB
110121074	Diệp Tú Như	DA21TTB
110121094	Trần Văn Sang	DA21TTB

Trà Vinh, tháng 6 năm 2024

MỤC LỤC

CHƯƠNG 1	GIỚI THIỆU.....	6
1.1	Mô tả bài toán	6
1.2	Đặc tả các yêu cầu chức năng.....	6
1.2.1	Yêu cầu chức năng	6
1.2.2	Yêu cầu phi chức năng	7
1.3	Sơ đồ kiến trúc	7
CHƯƠNG 2	CƠ SỞ LÝ THUYẾT	8
2.1	Lý thuyết về Agile	8
2.1.1	Khái niệm về Agile	8
2.1.2	Lịch sử hình thành và phát triển của Agile	8
2.1.3	Đặc trưng của Agile	9
2.1.4	Tôn chỉ của Agile	10
2.1.5	Nguyên tắc trong Agile	11
2.1.6	Ưu, nhược điểm của Agile	12
2.2	Lý thuyết về Scrum.....	13
2.2.1	Khái niệm về Scrum.....	13
2.2.2	Các vai trò trong Scrum	14
2.2.3	Tạo tác của Scrum.....	14
2.2.4	Quy trình Scrum.....	15
2.2.5	Các yếu tố giúp triển khai Scrum thành công	16
2.3	Tìm hiểu về Figma	17
2.3.1	Khái niệm	17
2.3.2	Lịch sử phát triển.....	18
2.3.3	Ưu, nhược điểm của Figma.....	18
2.4	Công nghệ, kiến trúc được sử dụng	19
2.4.1	W3.CSS	19
2.4.2	Laravel framework	20
2.4.3	PHP.....	21
2.4.4	Kiến trúc MVC.....	22
2.4.5	Docker	24
CHƯƠNG 3	XÁC ĐỊNH YÊU CẦU.....	26
3.1	Đặc tả mục tiêu của dự án.....	26

3.2	Xác định personas	26
3.2.1	Người dùng thông thường (User).....	26
3.2.2	Quản trị viên (Admin).....	26
3.3	Xác định user stories	27
3.3.1	Người dùng thông thường (User).....	27
3.3.2	Quản trị viên (Admin).....	27
3.4	Xác định các product backlog.....	28
3.5	Xác định các nhu cầu phi tính năng	29
CHƯƠNG 4	LẬP KẾ HOẠCH SCRUM.....	30
CHƯƠNG 5	HIỆN THỰC HÓA KẾ HOẠCH	32
5.1	Giao diện thiết kế bằng figma.....	32
5.2	Burndown chart.....	35
5.3	Mã nguồn	36
5.3.1	Dockerfile.....	36
5.3.2	Docker compose	37
CHƯƠNG 6	KẾT LUẬN	39
6.1	Kết quả đạt được	39
6.2	Hướng phát triển	39
TÀI LIỆU THAM KHẢO		40

DANH MỤC HÌNH ẢNH

Hình 5.1 Giao diện trang chủ của trang người dùng	32
Hình 5.2 Giao diện trang giới thiệu	32
Hình 5.3 Giao diện trang món ăn	33
Hình 5.4 Giao diện trang địa điểm phục vụ	33
Hình 5.5 Giao diện trang chi tiết món ăn	34
Hình 5.6 Giao diện trang chi tiết điểm phục vụ	34
Hình 5.7 Sprint 1: Thu thập và thiết kế giao diện	35
Hình 5.8 Sprint 2: Xây dựng và cài đặt trang quản trị	35
Hình 5.9 Sprint 3: Xây dựng và cài đặt trang người dùng	35
Hình 5.10 Sprint 4: Xây dựng và cài đặt các chức năng	36

CHƯƠNG 1 GIỚI THIỆU

1.1 Mô tả bài toán

Bài toán có thể được mô tả như sau: Xây dựng một trang web giới thiệu các món đặc sản của tỉnh Trà Vinh trong đó sử dụng W3.CSS để thiết kế giao diện và Laravel để hiển thị dữ liệu ra giao diện về danh sách các món đặc sản và cung cấp thông tin chi tiết cho mỗi món ăn khi người dùng chọn. Website gồm có bốn trang chính: Trang chủ dùng để hiển thị sơ lược các món ăn để người dùng có thể hình dung được nội dung mà trang web hướng đến. Trang giới thiệu dùng để giới thiệu khái quát về văn hóa ẩm thực ở Trà Vinh để người dùng có thể hiểu rõ hơn. Ở trang món ăn người dùng có thể lọc các món ăn theo danh mục các món ăn, có thể tìm hiểu những thông tin cụ thể hơn bằng cách click vào bất kì một món ăn nào đó sẽ được chuyển đến trang chi tiết món ăn, ngoài ra trang món ăn cũng có thêm chức năng tìm kiếm giúp người dùng có thể tìm kiếm một món ăn nào đó nhanh hơn. Trang địa điểm phục vụ sẽ hiển thị các địa điểm phục vụ theo từng món ăn khác nhau, người dùng cũng có thể click vào đó để xem chi tiết hơn về các địa điểm đó. Trang chi tiết món ăn có thể hiển thị các thông tin như hình ảnh, tên, mô tả, giá bán tham khảo, địa điểm tham khảo và các nguyên liệu chính của món ăn. Trang chi tiết địa điểm cũng sẽ hiển thị các thông tin chi tiết về địa điểm như địa chỉ, giờ mở cửa, giờ đóng cửa,... W3.CSS sẽ tạo ra một giao diện đẹp mắt để hiển thị dữ liệu, Laravel giúp hiển thị dữ liệu lên trang web một cách đơn giản, nhanh chóng và hiệu quả, giúp trải nghiệm của người dùng được tốt hơn.

1.2 Đặc tả các yêu cầu chức năng

1.2.1 Yêu cầu chức năng

Hiển thị danh sách và chi tiết món ăn, chi tiết địa điểm: Hiển thị danh sách các món ăn và người dùng có thể xem chi tiết thông tin của từng món ăn đó.

Chức năng tìm kiếm: Cho phép người dùng tìm kiếm các món ăn theo tên.

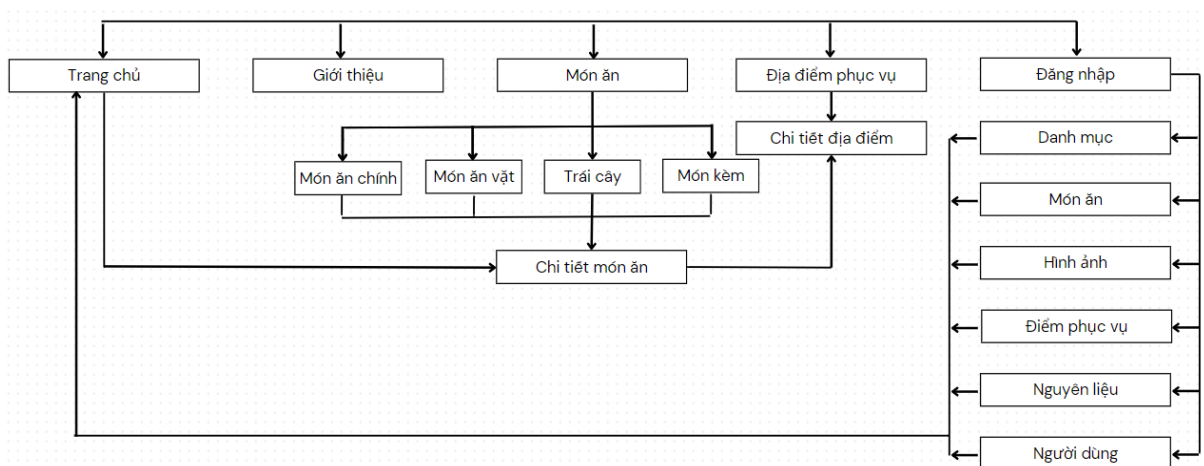
Quản lý các món ăn và các địa điểm phục vụ (Admin): Thêm mới các món ăn và các địa điểm phục vụ vào danh sách, xóa các món ăn và các địa điểm phục vụ ra khỏi danh sách, sửa và cập nhật thông tin các món ăn và các địa điểm phục vụ trong danh sách.

1.2.2 Yêu cầu phi chức năng

Giao diện thân thiện với người dùng: Giao diện dễ đọc và dễ sử dụng, người dùng có thể tìm kiếm món ăn một cách nhanh chóng. Các nút điều hướng cần được thiết kế một cách rõ ràng và dễ sử dụng để người dùng có thể dễ dàng điều hướng giữa các trang.

Quản lý nội dung dễ dàng: nội dung và hình ảnh dễ được cập nhật mới một cách thuận lợi.

1.3 Sơ đồ kiến trúc



CHƯƠNG 2 CƠ SỞ LÝ THUYẾT

2.1 Lý thuyết về Agile

2.1.1 Khái niệm về Agile

Agile (Agile Software Development) là một phương thức phát triển phần mềm linh hoạt, được thực hiện bằng cách sử dụng các bước lặp ngắn từ 1 đến 4 tuần. Mục tiêu của Agile là giúp rút ngắn thời gian phát triển sản phẩm, đưa sản phẩm đến với tay khách hàng càng sớm càng tốt.

Về bản chất, Agile giống như một phương pháp luận, một triết lý dựa trên nguyên tắc phân đoạn vòng lặp (iterative) và tăng trưởng (incremental) nên sở hữu tính linh hoạt cao. Tính chất này đi ngược lại với các phương pháp quản lý dự án truyền thống – vốn dĩ triển khai các giai đoạn một cách tuyến tính và vô cùng bị động trước các thay đổi bất ngờ.

Ngày nay, triết lý Agile đã vượt xa khỏi khu vực truyền thống của mình là phát triển phần mềm để đóng góp sự thay đổi trong cách thức làm việc, quản lý, sản xuất ở các ngành khác như sản xuất, dịch vụ, sales, marketing, giáo dục,... và trở thành một phương pháp quản lý dự án phổ biến nhất hiện nay với nhiều đại diện được gọi là các phương pháp “họ Agile”.

2.1.2 Lịch sử hình thành và phát triển của Agile

Trước khi có sự xuất hiện của Agile, mô hình Waterfall được xem như tiêu chuẩn vàng để phát triển phần mềm. Thông thường, quy trình sẽ bắt đầu bằng việc một nhà phân tích kinh doanh viết tài liệu yêu cầu nghiệp vụ, mô tả những yêu cầu mà họ cần từ phần mềm. Những tài liệu này thường rất dài và chi tiết, chứa đựng tất cả mọi thông tin từ chiến lược tổng thể đến các thông số kỹ thuật. Sau khi tất cả các tài liệu hoàn tất, các nhà phát triển mới bắt đầu code phần mềm, hoàn thiện nó và test. Tất cả quy trình này có thể mất đến một vài năm để kết thúc và để phần mềm được đưa vào sử dụng.

Vào năm 2001, một nhóm gồm 17 nhà phát triển phần mềm kinh nghiệm, bao gồm Kent Beck, Martin Fowler, Ron Jeffries, Ken Schwaber và Jeff Sutherland đã cùng nhau nghiên cứu và xuất bản Tuyên ngôn Agile với 4 giá trị quan trọng và 12 nguyên tắc hướng dẫn cách tiếp cận lặp đi lặp lại và tập trung vào con người để phát

triển phần mềm. Tuyên ngôn Agile được xem như là niềm tin của họ vào cách vận hành hoàn hảo mà quá trình phát triển phần mềm hiện đại nên có.

Với Tuyên ngôn Agile, các nhà phát triển nhấn mạnh vào tầm quan trọng của sự hợp tác, những sự thích ứng và cải tiến liên tục thay vì bị bó buộc vào các phương pháp triển khai cứng nhắc và khối lượng tài liệu cần chuẩn bị công kênh của phương pháp Waterfall truyền thống.

2.1.3 Đặc trưng của Agile

Tính lặp (Iterative): Dự án sẽ được thực hiện trong các phân đoạn lặp đi lặp lại, thường có khung thời gian ngắn (từ 1 - 4 tuần). Trong mỗi phân đoạn đó, nhóm phát triển dự án sẽ thực hiện đầy đủ các công việc cần thiết như lập kế hoạch, phân tích yêu cầu, thiết kế, triển khai, kiểm thử để cho ra các phần nhỏ của sản phẩm.

Tính tăng trưởng và tiến hóa (Incremental & Evolutionary): Cuối các phân đoạn, nhóm cho ra các phần nhỏ của sản phẩm cuối cùng, thường là đầy đủ, có khả năng chạy tốt, được kiểm thử cẩn thận và có thể sử dụng. Theo thời gian, phân đoạn này tiếp nối phân đoạn kia, các phần chạy được này sẽ được tích lũy, lớn dần lên cho tới khi toàn bộ yêu cầu của khách hàng được thỏa mãn.

Tính thích nghi (adaptive): Do các phân đoạn chỉ kéo dài trong một khoảng thời gian ngắn và việc lập kế hoạch cũng được điều chỉnh liên tục, nên các thay đổi trong quá trình phát triển (yêu cầu thay đổi, thay đổi công nghệ, thay đổi định hướng về mục tiêu,...) đều có thể được đáp ứng theo cách thích hợp.

Nhóm tự tổ chức và liên chức năng: Các cấu trúc nhóm này tự phân công công việc mà không dựa trên các mô tả cứng nhắc về chức danh hay một sự phân cấp rõ ràng. Nhóm tự tổ chức đã đủ các kỹ năng cần thiết để có thể được trao quyền tự ra quyết định, tự quản lý và tổ chức lấy công việc của chính mình để đạt được hiệu quả cao nhất.

Quản lý tiến trình thực nghiệm (Empirical Process Control): Các nhóm Agile ra các quyết định dựa trên các dữ liệu thực tiễn (data-driven) thay vì tính toán lý thuyết hay các tiên giả định. Agile rút ngắn vòng đời phản hồi để dễ dàng thích nghi và gia tăng tính linh hoạt nhờ đó có thể kiểm soát được tiến trình, và nâng cao năng suất lao động.

Giao tiếp trực diện (face-to-face communication): Agile không phản đối việc tài liệu hóa, nhưng đánh giá cao hơn việc giao tiếp trực diện thay vì thông qua giấy tờ. Agile khuyến khích nhóm phát triển trực tiếp nói chuyện để hiểu rõ hơn về những gì khách hàng thực sự cần. Trong giao tiếp giữa nội bộ nhóm, Agile khuyến khích trực tiếp trao đổi và thống nhất với nhau về thiết kế của hệ thống và cùng nhau triển khai thành các chức năng theo yêu cầu.

Phát triển dựa trên giá trị (value-based development): Một trong các nguyên tắc cơ bản của Agile là “sản phẩm chạy tốt chính là thước đo của tiến độ”. Nhóm Agile thường cộng tác trực tiếp và thường xuyên với khách hàng để biết yêu cầu nào có độ ưu tiên cao hơn, mang lại giá trị hơn sớm nhất có thể cho dự án.

2.1.4 Tôn chỉ của Agile

Cá nhân và sự tương tác quan trọng hơn các quy trình và công cụ

Trước đây, các nhóm phát triển sản phẩm thường có xu hướng tập trung vào các công cụ hay quy trình tốt nhất để xây dựng sản phẩm. Tuy nhiên, Agile cho rằng những điều đó quá thật quan trọng, nhưng những người đứng sau các quy trình đó mới là nhân tố quan trọng nhất.

Sở hữu một nhóm các cá nhân phù hợp cho dự án là yếu tố quan trọng nhất để thành công. Công cụ tốt mà rơi vào tay của những người không có năng lực cũng trở nên vô dụng. Ngoài ra, sự tương tác và kết nối giữa các thành viên trong nhóm cũng là chìa khóa then chốt để làm việc nhóm và giải quyết mọi vấn đề phát sinh.

Sản phẩm dùng được tốt hơn tài liệu đầy đủ

Trước đây, các kỹ sư phần mềm sẽ dành nhiều thời gian để viết ra các tài liệu chi tiết về thiết kế sản phẩm, thông số,... trước khi bắt tay vào code thực tế. Tài liệu chi tiết rõ ràng là một điểm cộng, tuy nhiên, có những yếu tố khác mà họ cần tập trung vào hơn, đó chính là việc cung cấp một phần mềm có thể hoạt động tốt được.

Tuyên ngôn Agile đặt việc cung cấp một sản phẩm thiết thực cho khách hàng của bạn ở mức ưu tiên cao nhất.

Cộng tác với khách hàng quan trọng hơn đàm phán hợp đồng

Trước đây chủ dự án sẽ soạn thảo hợp đồng trước, sau đó khách hàng mới là người hoàn thiện bức tranh chi tiết về thành phẩm yêu cầu. Điều này rất dễ dẫn đến sự mâu thuẫn giữa những gì hợp đồng đã viết, những gì sản phẩm có thể làm và những gì mà khách hàng thực sự yêu cầu.

Với Tuyên ngôn Agile, sự tập trung được đặt vào sự phát triển liên tục. Nhóm thực hiện sẽ xây dựng một vòng lặp tương tác và phản hồi với khách hàng để đảm bảo rằng sản phẩm tạo ra phù hợp với họ.

Phản hồi với sự thay đổi quan trọng hơn làm theo kế hoạch

Chúng ta luôn kỳ vọng xây dựng nên một lộ trình triển khai dự án hoàn hảo và không bao giờ có sự thay đổi, tuy nhiên, điều đó hầu như chẳng bao giờ xảy ra. Quá trình triển khai sẽ luôn phát sinh những vấn đề mới không nằm trong kế hoạch, nhu cầu và yêu cầu của khách hàng cũng sẽ luôn thay đổi.

Tuyên ngôn Agile yêu cầu nhóm thực hiện có được khả năng xoay vòng và thay đổi hướng đi bất cứ lúc nào, với một lộ trình linh hoạt, thay đổi theo từng tuần, tháng hoặc quý.

2.1.5 Nguyên tắc trong Agile

1. Ưu tiên cao nhất của Agile là làm hài lòng khách hàng thông qua việc cung cấp sớm và liên tục các sản phẩm có giá trị.
2. Agile luôn chào đón các yêu cầu thay đổi, kể cả khi nó xuất hiện ở giai đoạn cuối của quá trình triển khai, khi mọi thứ đã gần như hoàn thiện. Quy trình Agile luôn khai thác sự thay đổi để tạo nên lợi thế cạnh tranh cho khách hàng.
3. Giao phần mềm chạy được cho khách hàng một cách thường xuyên, từ vài tuần đến vài tháng.
4. Các nhà kinh doanh và nhà phát triển phần mềm phải làm việc cùng nhau hàng ngày trong suốt dự án.
5. Tập trung xây dựng dự án xung quanh những cá nhân có động lực, cung cấp cho họ môi trường và sự hỗ trợ cần thiết, cùng với niềm tin vào việc hoàn thành tốt công việc.
6. Phương pháp hiệu quả nhất để truyền đạt thông tin là trao đổi trực tiếp.

-
7. Phần mềm hoạt động tốt chính là thước đo của tiến độ triển khai Agile.
 8. Quy trình Agile thúc đẩy sự phát triển bền vững.
 9. Sự chú ý liên tục đến yếu tố kỹ thuật và thiết kế sẽ cải tiến sự linh hoạt.
 10. Nghệ thuật tối đa hóa lượng công việc chưa xong: Sự đơn giản là cần thiết.
 11. Những thiết kế và yêu cầu tốt nhất chỉ xuất hiện tại những nhóm làm việc có khả năng tự tổ chức tốt.
 12. Nhóm thực thi cần suy nghĩ về việc làm việc hiệu quả một cách thường xuyên để có sự điều chỉnh hành vi phù hợp.

2.1.6 Ưu, nhược điểm của Agile

Ưu điểm của Agile

Mô hình Agile ngày càng trở nên phổ biến, được nhiều doanh nghiệp áp dụng để phát triển sản phẩm của mình bởi nó sở hữu những ưu điểm vượt trội so với những phương pháp truyền thống.

Dễ dàng thực hiện thay đổi ở bất kỳ giai đoạn nào của dự án, thích nghi nhanh và hiệu quả với sự thay đổi, bao gồm cả yêu cầu sửa lại sản phẩm, sự biến động từ thị trường,...

Phát triển và bàn giao sản phẩm nhanh hơn, bởi việc chia nhỏ dự án cho phép đội ngũ có thể tiến hành kiểm tra theo từng phần, phát hiện sự cố và sửa chữa vấn đề nhanh hơn.

Sản phẩm đạt chất lượng tốt hơn, do luôn nhận được phản hồi ngay lập tức từ phía khách hàng và được tối ưu lại ngay sau đó.

Lãng phí ít tài nguyên hơn vì nhóm luôn thực hiện các công việc đã được cập nhật, giảm tải thời gian chờ, giảm tải thời gian sửa lỗi sản phẩm, giảm tải số lượng lớn giấy tờ tài liệu,...

Người tham gia dự án không cần phải nắm mọi thông tin ngay từ đầu, phù hợp với những dự án chưa xác định rõ ràng được mục tiêu cuối cùng.

Nhược điểm của Agile

Bên cạnh những ưu điểm vượt trội, cũng không khó để nhận ra một số nhược điểm còn tồn tại khi triển khai mô hình Agile trong thực tế:

Khó lên kế hoạch dự án, đặc biệt là khó xác định rõ ràng thời gian bàn giao sản phẩm cuối cùng, không nắm rõ được chi phí thực sự của dự án là bao nhiêu,... vì dự án được chia nhỏ thành các vòng lặp khác nhau.

Bắt buộc phải hướng dẫn và đào tạo chi tiết, thì thành viên dự án mới có thể hiểu rõ được mô hình Agile và thực hiện theo nó một cách rõ ràng, đặc biệt là trong thời gian đầu.

Ít tài liệu hướng dẫn về dự án và không xác định rõ được kỳ vọng và thành phẩm ngay từ đầu, do người thực thi mô hình Agile thường tin rằng mọi thứ sẽ thay đổi rất nhiều, không quá cần thiết phải ghi chép và lưu trữ tài liệu.

Đòi hỏi sự cam kết về thời gian và công sức từ các bên hơn, vì tất cả mọi người cần phải liên tục tương tác với nhau trong suốt quá trình thực thi dự án.

Chi phí thực hiện dự án theo mô hình Agile thường cao hơn so với các phương pháp phát triển khác.

2.2 Lý thuyết về Scrum

2.2.1 Khái niệm về Scrum

Scrum được xem là mô hình phổ biến nhất trong các phương pháp vận hành doanh nghiệp theo phương pháp luận Agile là một Agile framework giúp đội ngũ dự án quản lý công việc của học thông qua các giá trị, nguyên tắc và thực tiễn.

Scrum được hình thành bởi 3 nguyên tắc cốt lõi: Minh bạch – Kiểm tra – Thích ứng, do vậy mô hình này được các nhà quản lý ưa chuộng áp dụng mỗi khi cần nâng cao hiệu quả làm việc nhóm.

Scrum khuyến khích đội ngũ tự học hỏi thông qua kinh nghiệm, tự tổ chức trong giải quyết vấn đề và tự suy ngẫm về thành tựu cũng như hậu quả để liên tục cải thiện.

Trọng tâm của Scrum là Sprint. Một dự án Scrum chia thời gian dự án thành nhiều Sprint, mỗi Sprint kéo dài khoảng 1 - 4 tuần và trong thời gian đó sẽ hoàn thành một khối lượng công việc nhất định.

2.2.2 Các vai trò trong Scrum

Product Owner: là người chịu trách nhiệm tối ưu giá trị tạo ra bằng cách đưa ra quyết định về tính năng của sản phẩm, đánh giá và sắp xếp ưu tiên cho từng hạng mục tính năng. Product Owner cũng là người tạo ra Product Backlog, được dùng để lưu trữ các hạng mục tính năng của sản phẩm. PO là người thường xuyên tiếp xúc và trao đổi với khách hàng nhằm đảm bảo rằng dự án sẽ mang lại nhiều lợi ích nhất cho khách hàng bằng cách tạo ra một sản phẩm phù hợp với thị hiếu và nhu cầu của khách hàng.

Scrum Master: là người chịu trách nhiệm về hiệu quả làm việc của Development Team đạt hiệu quả cao cũng như theo đúng nguyên lý và quy tắc của Scrum framework. Chính xác thì công việc của SM là dẫn dắt Team đi theo các quy trình của Scrum và hỗ trợ Team trong việc hoàn thành dự án. Tại Việt Nam thì SM còn thường đảm nhận thêm nhiều công tác quản lý truyền thống như lên kế hoạch, giám sát nên SM còn được gọi là PM Agile (phân biệt với PM truyền thống).

Development Team: Development Team hay đội ngũ phát triển bao gồm tất cả các thành viên tham gia phát triển dự án như developer, tester, designer, ... Họ là những người trực tiếp xây dựng các tính năng của sản phẩm và có nhiệm vụ chuyển giao các thành quả tăng trưởng vào mỗi cuối Sprint.

2.2.3 Tạo tác của Scrum

Tạo tác của Scrum hay Scrum Artifacts là những công cụ trong Scrum được sử dụng để giải quyết vấn đề và quản lý dự án. Ngoài ra chúng còn được sử dụng để giúp cho các bên liên quan hiểu về chi tiết dự án và dự án đang diễn ra như thế nào.

Product Backlog: là một danh sách bao gồm các tính năng, yêu cầu, cải tiến và chỉnh sửa cần hoàn thành để dự án thành công. Các hạng mục trong Product Backlog được sắp xếp theo thứ tự ưu tiên và quản lý bởi Product Owner. Product Backlog là không cố định, hoàn toàn có thể bị thay đổi từ nội dung đến thứ tự ưu tiên. Do đó, PO có nhiệm vụ bảo trì và làm mịn nó. Ngoài ra, PO phải giải thích được chi tiết của từng hạng mục để các thành viên có hiểu biết chung về sản phẩm, tránh việc làm sai yêu cầu của dự án.

Sprint Backlog: là một hạng mục cần hoàn thành trong Sprint, có thể coi như là tập hợp con của Product Backlog và có tính chất tương tự. Điểm khác biệt giữa

Sprint Backlog và Product Backlog là Sprint Backlog được quản lý bởi Development Team còn Product Backlog được quản lý bởi Product Owner. Với mỗi Sprint, đội ngũ dự án chọn ra các hạng mục từ Product Backlog và đưa vào Sprint Backlog, điều này sẽ diễn ra trong buổi Sprint Planning. Sprint Backlog có thể linh hoạt nhưng các mục tiêu cơ bản thì đội ngũ dự án phải hoàn thành và không thể nhượng bộ.

Product Increment: là sản phẩm có một phần tính năng “hoàn chỉnh” được đội ngũ bàn giao vào mỗi cuối sprint. Còn về định nghĩa “hoàn chỉnh” thì được PO xác định ở đầu sprint (thuật ngữ là Definition of Done – DoD). Scrum chia dự án thành các Sprint nhưng các đầu ra của từng sprint phải liên quan tới nhau, sau mỗi sprint thì sản phẩm lại có sự tăng trưởng. Bạn có thể hiểu rằng thay vì đợi đến khi hoàn thiện dự án mới bàn giao cho khách hàng thì với Scrum, mỗi sprint đều bàn giao 1 phần của sản phẩm cuối cùng cho khách hàng. Chúng ta có thể cải thiện liên tục thông qua các phản hồi thực tế từ khách hàng. Product Increment là thuật ngữ tạo tác ít xuất hiện nhất vì nó có thể được gọi với nhiều tên khác nhau như sprint goal, milestone. Tuy nhiên thực tế thì nó lại là phần quan trọng nhất của Scrum và chính nó đã khiến Scrum trở nên phổ biến với lợi ích của mình.

2.2.4 Quy trình Scrum

Khi một dự án bắt đầu, Product Owner sẽ làm việc với khách hàng và đưa các yêu cầu vào trong Product Backlog. Sau đó sẽ lên kế hoạch Sprint, với mỗi sprint lại lên kế hoạch một lần và đưa các mục tiêu vào Sprint Backlog. Liên tục lặp lại Sprint cho tới khi kết thúc dự án.

Trong buổi Sprint Planning cũng sẽ thống nhất thời gian một Sprint kéo dài là bao lâu. Sau khi đã có kế hoạch Sprint, sẽ có Sprint Backlog chứa các hạng mục công việc trong Sprint đó. Mỗi ngày sẽ có một buổi họp ngắn hàng ngày (Daily Scrum/ Daily Meeting) để kiểm tra tiến độ hoàn thành sprint goal và điều chỉnh sprint backlog nếu cần thiết. Cũng trong buổi họp này thì chúng ta sẽ thống nhất làm gì trong 24 giờ tiếp theo. Vào mỗi cuối Sprint, sẽ có hai cuộc họp nữa là Sprint Review và Sprint Retrospective.

Sprint Review là cuộc họp để demo sản phẩm của sprint và trình bày cho Product Owner các hạng mục đã hoàn thành. Chính PO là người quyết định xem sản

phẩm đã đạt và có thể chuyển giao hay chưa. Vào cuối cuộc họp Sprint Review, PO cũng dành thời gian xem lại Product Backlog và dự định những gì sẽ làm trong Sprint tiếp theo. Cuối cùng là Sprint Retrospective. Cuộc họp này là dành cho Development Team, là thời gian dành cho các thành viên ngồi lại với nhau để xem xem điều gì làm tốt/ chưa tốt. Sau cuộc họp thì có thể đưa ra các cải tiến trong sprint tiếp theo. Các sprint sẽ được lặp lại như vậy liên tục cho tới khi dự án kết thúc.

2.2.5 Các yếu tố giúp triển khai Scrum thành công

2.2.5.1 Ba trụ cột của Scrum

Ba trụ cột của Scrum là những gì làm cho Scrum thành công. Nếu không có chúng, có nguy cơ lãng phí thời gian của và đi lòng vòng, không tiến bộ được gì.

Trong Scrum Guide cũng có định nghĩa khá rõ về 3 trụ cột này (3 Scrum Pillar):

Minh bạch (Transparency): “Quy trình và công việc nổi bật phải được hiển thị cho những người thực hiện công việc cũng như những người tiếp nhận công việc”.

Thanh tra (Inspection): “Các thành phần Scrum và tiến trình hướng tới các mục tiêu đã thống nhất phải được kiểm tra thường xuyên và siêng năng để phát hiện các vấn đề hoặc sự khác biệt tiềm ẩn không mong muốn.”

Thích nghi (Adaptation): “Nếu bất kỳ khía cạnh nào của một quá trình đi lệch ra ngoài các giới hạn có thể chấp nhận được hoặc nếu sản phẩm tạo ra là không thể chấp nhận được, thì quá trình đang được áp dụng hoặc các vật liệu đang được sản xuất phải được điều chỉnh”

2.2.5.2 Năm giá trị của Scrum

Các giá trị của Scrum được hiểu là các phẩm chất mà những người trong team Scrum nên có thậm chí là cần có để triển khai Scrum dễ dàng hơn.



Dũng cảm (Courage): Đương đầu với thay đổi, dám đưa ra các ý tưởng cải tiến hay suy nghĩ của mình, tất cả đều cần có sự dũng cảm.

Tập trung (Focus): Scrum yêu cầu sự tập trung vào mục tiêu chứ không phải là bị phân tán nguồn lực đi khắp nơi, cho dù mục tiêu cũng được tập trung theo thứ tự ưu tiên.

Cam kết (Commitment): Tự quản lý là một trong những điểm nổi trội của Scrum. Để có thể tự quản lý một cách hiệu quả thì người đó phải có sự cam kết về đầu ra công việc một cách ổn định.

Tôn trọng (Respect): Tôn trọng lẫn nhau là chìa khoá để các thành viên trong nhóm có thể làm việc một cách thoải mái và hiệu quả.

Cởi mở (Openness): Cải tiến liên tục cần ý tưởng liên tục, nếu bị bó buộc trong một khuôn nào đó thì điều này sẽ rất khó.

2.3 Tìm hiểu về Figma

2.3.1 Khái niệm

Figma là một công cụ thiết kế đồ họa và giao diện người dùng (UI) dựa trên web, hỗ trợ thiết kế và cộng tác trực tuyến, cho phép các nhà thiết kế tạo ra và chỉnh sửa các bản thiết kế trong thời gian thực với sự tham gia của nhiều người. Figma tích hợp nhiều tính năng mạnh mẽ như công cụ vẽ vector, quản lý prototyping, và khả năng

quản lý thiết kế theo nhóm. Được ưa chuộng nhờ tính tiện lợi và hiệu suất cao, Figma là lựa chọn phổ biến cho các nhóm thiết kế hiện đại. Ngoài ra, Figma còn hỗ trợ làm việc trên nhiều nền tảng khác nhau mà không cần cài đặt phần mềm.

2.3.2 Lịch sử phát triển

Figma bắt đầu cung cấp phiên bản xem trước miễn phí dành cho người dùng muốn đăng ký trải nghiệm vào ngày 3 tháng 12 năm 2015. Sau đó, phần mềm được phát hành công khai đầu tiên ngày 27 tháng 9 năm 2016. Ngày 22 tháng 10 năm 2019, Figma đã giới thiệu Figma Community, một mạng xã hội cho phép các nhà thiết kế xuất bản tác phẩm của mình để người khác xem và phối lại.

Tính đến tháng 4 năm 2020, Figma được định giá hơn hai tỷ đô la Mỹ và đạt mức 10 tỷ đô la vào cuối tháng 5 năm 2021. Ngày 15 tháng 09 năm 2022, tập đoàn Adobe đã mua lại nền tảng thiết kế này với giá trị ước tính khoảng 20 tỷ đô la Mỹ. 2

2.3.3 Ưu, nhược điểm của Figma

2.3.3.1 Ưu điểm

Khả năng tương thích cao: Là một phần mềm trên nền tảng website, Figma có thể được truy cập và sử dụng trên hầu hết các hệ điều hành phổ biến hiện nay như Windows, MacOS hay thậm chí là Linux, chỉ cần thiết bị được kết nối Internet. Hiện tại, Figma còn có các phiên bản ứng dụng trên máy tính và điện thoại để người dùng có thể sáng tạo hay theo dõi các dự án ở bất cứ nơi đâu.

Tính cộng tác cao: Ưu điểm hàng đầu của Figma phải kể đến tính cộng tác trong thời gian thực. Với Figma, làm việc nhóm từ xa trở nên dễ dàng khi nó cho phép những người khác trong dự án có thể cộng tác về thiết kế, cung cấp phản hồi, kiểm tra tiến độ công việc hoặc gắn cờ các vấn đề có thể xảy ra theo thời gian thực.

Trang bị nhiều Plugin mạnh mẽ: Figma cung cấp nhiều plugin hữu ích giúp người sử dụng dễ dàng khắc phục các lỗi nhỏ và tăng hiệu suất làm việc. Việc quản lý màu sắc, nội dung hình ảnh hay hoạt ảnh đều trở nên dễ dàng hơn bao giờ hết.

2.3.3.2 Nhược điểm

Yêu cầu kết nối internet liên tục: Figma hoạt động dựa trên nền tảng web, do đó người dùng cần có kết nối internet liên tục để sử dụng. Điều này có thể gây khó khăn trong các tình huống mất mạng hoặc có mạng chậm.

Hiệu suất có thể bị ảnh hưởng: Khi làm việc với các dự án lớn hoặc có nhiều chi tiết, Figma có thể gặp vấn đề về hiệu suất, khiến cho công việc thiết kế trở nên chậm chạp và kém mượt mà.

Hạn chế tính năng ngoại tuyến: Dù đã có phiên bản desktop, nhưng nhiều tính năng vẫn yêu cầu kết nối internet, làm giảm tính tiện dụng khi làm việc ngoại tuyến.

Hạn chế tùy chỉnh giao diện: So với một số công cụ thiết kế khác, Figma có ít tùy chọn tùy chỉnh giao diện người dùng, khiến cho một số người dùng cảm thấy khó khăn khi muốn cá nhân hóa môi trường làm việc.

Chi phí: Mặc dù có phiên bản miễn phí, nhưng để sử dụng đầy đủ các tính năng và cho các nhóm làm việc chuyên nghiệp, người dùng cần trả phí hàng tháng hoặc hàng năm, có thể gây ra chi phí đáng kể cho các công ty nhỏ hoặc cá nhân.

2.3.3.3 Đối tượng sử dụng

Là một công cụ thiết kế nhưng Figma không chỉ dành riêng cho người thiết kế. Với tính tương tác cao, Figma thường được lựa chọn để làm “giao diện chung” cho các dự án thiết kế UX/UI. Do đó, bất kỳ ai tham gia vào dự án, từ designer, front-end/back end developer cho đến cả CTO cũng có thể học cách sử dụng Figma để quá trình làm việc nhóm mượt mà hơn.

2.4 Công nghệ, kiến trúc được sử dụng

2.4.1 W3.CSS

W3.CSS là một framework CSS nhẹ nhàng và dễ sử dụng, được phát triển bởi W3Schools. Nó được thiết kế để giúp tạo ra giao diện người dùng đẹp mắt và linh hoạt cho các trang web và ứng dụng web mà không cần quá nhiều kiến thức về CSS.

Đặc điểm

Lightweight: W3.CSS có dung lượng nhỏ và tải nhanh, giúp tối ưu hóa hiệu suất của trang web.

Responsive: Framework này hỗ trợ responsive design, tự động điều chỉnh giao diện để phù hợp với các kích thước màn hình khác nhau, từ điện thoại di động đến máy tính bảng và desktop.

Easy to Use: W3.CSS có cú pháp dễ hiểu và dễ sử dụng, ngay cả cho những người mới bắt đầu với CSS. Các class được đặt tên rõ ràng và mô tả chức năng của chúng, giúp việc lập trình trở nên dễ dàng hơn.

Customizable: Mặc dù W3.CSS cung cấp một loạt các thành phần giao diện sẵn có, nhưng vẫn có thể tùy chỉnh chúng để phù hợp với nhu cầu cụ thể thông qua việc sử dụng các class và CSS tùy chỉnh.

Các Thành Phần Giao Diện: W3.CSS cung cấp một loạt các thành phần giao diện người dùng sẵn có, bao gồm: Buttons, Forms, Navigation Bars, Grid System, Alerts, Tables, Cards, ...

Hỗ Trợ Tích Hợp: W3.CSS được hỗ trợ trên hầu hết các trình duyệt hiện đại và có thể dễ dàng tích hợp vào các dự án sử dụng HTML, CSS và JavaScript.

Cộng Đồng và Tài Liệu: Với sự hỗ trợ từ W3Schools và cộng đồng lập trình viên, chúng ta có thể tìm thấy rất nhiều tài liệu và ví dụ về cách sử dụng W3.CSS để tạo ra giao diện người dùng đẹp mắt và chuyên nghiệp.

2.4.2 Laravel framework

Định nghĩa: Laravel là một framework PHP mã nguồn mở, được thiết kế để giúp các nhà phát triển tạo ra các ứng dụng web một cách nhanh chóng và dễ dàng.

Lịch sử: Laravel được tạo ra bởi Taylor Otwell vào năm 2011 với mục tiêu cung cấp một framework có cú pháp đẹp và dễ sử dụng.

Đặc điểm chính:

Kiến trúc MVC: Laravel tuân thủ mô hình Model-View-Controller (MVC), giúp tách biệt phần logic xử lý và giao diện người dùng.

Eloquent ORM: Một hệ thống ORM mạnh mẽ giúp tương tác với cơ sở dữ liệu một cách dễ dàng và trực quan.

Artisan CLI: Công cụ dòng lệnh mạnh mẽ của Laravel, giúp tự động hóa nhiều nhiệm vụ phát triển.

Blade Templating Engine: Một công cụ xử lý template mạnh mẽ và dễ sử dụng.

Bảo mật: Laravel cung cấp nhiều tính năng bảo mật như bảo vệ CSRF, mã hóa mật khẩu và bảo vệ chống tấn công SQL injection.

Quản lý phiên bản: Hỗ trợ quản lý phiên bản và triển khai ứng dụng một cách dễ dàng.

Quản lý gói với Composer: Laravel sử dụng Composer để quản lý các thư viện và gói bổ sung, giúp dễ dàng mở rộng và nâng cấp.

Các tính năng chính của Laravel:

Route và Middleware: Quản lý các route và lớp trung gian (middleware) một cách linh hoạt.

Controller: Dễ dàng tạo và quản lý các controller để xử lý logic nghiệp vụ.

Migration và Seeding: Hỗ trợ di chuyển và khởi tạo cơ sở dữ liệu.

Event và Listener: Hệ thống sự kiện và listener giúp xây dựng ứng dụng phản ứng nhanh.

Queues: Hỗ trợ xử lý hàng đợi giúp nâng cao hiệu suất ứng dụng.

Task Scheduling: Lên lịch thực hiện các tác vụ định kỳ một cách dễ dàng.

2.4.3 PHP

Định nghĩa: PHP (Hypertext Preprocessor) là một ngôn ngữ lập trình kịch bản phía máy chủ, được thiết kế chủ yếu để phát triển ứng dụng web động.

Lịch sử: PHP được tạo ra bởi Rasmus Lerdorf vào năm 1994 và phát triển thành một trong những ngôn ngữ phổ biến nhất cho lập trình web.

Đặc điểm chính

Mã nguồn mở: PHP là phần mềm mã nguồn mở, miễn phí và có cộng đồng phát triển rộng lớn.

Đa nền tảng: PHP có thể chạy trên nhiều hệ điều hành như Windows, Linux, macOS.

Dễ học: Cú pháp PHP tương đối đơn giản và dễ học, đặc biệt cho những người đã có kinh nghiệm với các ngôn ngữ lập trình khác.

Tích hợp tốt với HTML: PHP có thể dễ dàng nhúng vào trong mã HTML, giúp việc tạo trang web động trở nên thuận tiện hơn.

Các tính năng của PHP

Xử lý biểu mẫu: PHP có thể thu thập dữ liệu từ biểu mẫu, gửi email, và xử lý dữ liệu từ các trang web khác.

Quản lý phiên làm việc: PHP hỗ trợ quản lý phiên làm việc và cookie.

Kết nối cơ sở dữ liệu: PHP cung cấp nhiều tùy chọn để kết nối và thao tác với cơ sở dữ liệu như MySQL, PostgreSQL, SQLite.

Hỗ trợ các thư viện và framework: PHP có nhiều thư viện và framework phong phú như GD Library cho xử lý ảnh, cURL cho truyền tải dữ liệu qua các giao thức khác nhau, và nhiều framework phát triển web mạnh mẽ như Laravel, Symfony, CodeIgniter.

2.4.4 Kiến trúc MVC

Kiến trúc MVC (Model-View-Controller) là một mô hình kiến trúc phổ biến được sử dụng trong phát triển phần mềm, đặc biệt là trong việc phát triển các ứng dụng web. Dưới đây là chi tiết hơn về các phần cấu thành của kiến trúc MVC

2.4.4.1 Model

Chức năng: Model là phần của ứng dụng chịu trách nhiệm xử lý dữ liệu và logic liên quan đến dữ liệu. Nó đại diện cho dữ liệu của ứng dụng và chứa các phương thức để tương tác với cơ sở dữ liệu.

Nhiệm vụ:

Truy cập và lấy dữ liệu từ cơ sở dữ liệu.

Thực hiện các phương thức để thêm, sửa đổi, và xóa dữ liệu từ cơ sở dữ liệu.

Chứa logic xử lý dữ liệu và các phương thức xử lý nghiệp vụ.

2.4.4.2 View

Chức năng: View là phần của ứng dụng chịu trách nhiệm hiển thị dữ liệu cho người dùng cuối. Nó là giao diện người dùng mà người dùng nhìn thấy và tương tác.

Nhiệm vụ:

Hiển thị dữ liệu cho người dùng dưới dạng trang web hoặc giao diện người dùng khác.

Đảm bảo rằng giao diện người dùng phản ánh đúng dữ liệu từ Model và trạng thái của ứng dụng.

Thực hiện các tính năng giao diện như biểu mẫu nhập liệu, nút bấm, và liên kết.

2.4.4.3 Controller

Chức năng: Controller là phần của ứng dụng chịu trách nhiệm điều hướng yêu cầu từ người dùng và tương tác với Model và View tương ứng.

Nhiệm vụ:

Tiếp nhận các yêu cầu từ người dùng thông qua các route.

Xử lý các yêu cầu bằng cách gọi các phương thức trong Model để lấy hoặc xử lý dữ liệu.

Chuyển dữ liệu từ Model cho View để hiển thị kết quả cho người dùng.

Quản lý luồng điều hướng của ứng dụng bằng cách quyết định xem yêu cầu nào cần được xử lý bởi đâu.

2.4.4.4 Lợi Ích của Kiến trúc MVC

Tách biệt logic: Kiến trúc MVC tách biệt logic ứng dụng thành các phần riêng biệt, làm cho mã nguồn dễ đọc, dễ hiểu và dễ bảo trì.

Tính tái sử dụng: Các thành phần của kiến trúc MVC có thể tái sử dụng được, giúp giảm thiểu việc lặp lại mã nguồn.

Dễ quản lý: Với việc chia ứng dụng thành các phần nhỏ hơn, kiến trúc MVC giúp quản lý ứng dụng dễ dàng hơn.

2.4.5 Docker

Định nghĩa: Docker là một nền tảng mã nguồn mở giúp tự động hóa việc triển khai các ứng dụng bên trong các container phần mềm. Container cho phép các nhà phát triển đóng gói ứng dụng với tất cả các phần phụ thuộc cần thiết vào một gói duy nhất, đảm bảo rằng ứng dụng có thể chạy một cách nhất quán trên mọi môi trường.

Lịch sử: Docker được phát triển bởi Solomon Hykes và công ty Docker, Inc. và ra mắt lần đầu vào năm 2013.

Các khái niệm cơ bản

Container: Một container là một gói phần mềm độc lập và nhẹ, bao gồm tất cả những gì cần thiết để chạy một ứng dụng, bao gồm mã nguồn, runtime, hệ thống thư viện và thiết lập hệ thống.

Image: Image là một khuôn mẫu bất biến để tạo ra container. Image bao gồm hệ điều hành, phần mềm cài đặt, cấu hình, và ứng dụng.

Dockerfile: Dockerfile là một tệp văn bản chứa các chỉ dẫn để xây dựng Docker image. Dockerfile mô tả các bước cần thiết để cài đặt và thiết lập môi trường cho ứng dụng.

Docker Hub: Docker Hub là một dịch vụ lưu trữ và chia sẻ Docker image, cho phép người dùng tải lên và tải xuống các image công khai hoặc riêng tư.

Orchestration: Docker có thể tích hợp với các công cụ như Kubernetes và Docker Swarm để quản lý và điều phối nhiều container trên các máy chủ khác nhau.

Các tính năng chính của Docker

Portable (Di động): Container Docker có thể chạy nhất quán trên các môi trường khác nhau, từ máy phát triển của nhà phát triển đến môi trường sản xuất.

Lightweight (Nhẹ): Docker container sử dụng chung nhân hệ điều hành, vì vậy chúng nhẹ hơn và khởi động nhanh hơn so với máy ảo truyền thống.

Isolation (Cách ly): Mỗi container được cách ly hoàn toàn với các container khác và với hệ thống chủ, đảm bảo môi trường chạy độc lập.

Scalable (Có thể mở rộng): Docker hỗ trợ mở rộng ứng dụng dễ dàng thông qua việc tạo ra nhiều container và phân phối chúng trên các máy chủ khác nhau.

Version Control (Kiểm soát phiên bản): Docker image có thể được phiên bản hóa, giúp theo dõi các thay đổi và quay lại phiên bản trước nếu cần.

Các thành phần chính của Docker

Docker Engine: Là nền tảng chính của Docker, bao gồm Docker daemon, API REST và CLI.

Docker Daemon: Chạy trên máy chủ và chịu trách nhiệm xây dựng, chạy, và quản lý container.

Docker CLI: Giao diện dòng lệnh cho phép người dùng tương tác với Docker daemon.

Docker API: Cung cấp các API REST để tương tác với Docker daemon từ các ứng dụng khác.

Docker Compose: Một công cụ cho phép định nghĩa và quản lý nhiều container Docker như một ứng dụng duy nhất. Sử dụng tệp YAML để cấu hình dịch vụ của ứng dụng.

Docker Swarm: Một công cụ tích hợp trong Docker Engine cho phép triển khai và quản lý một nhóm các Docker Engine dưới dạng một cluster.

Lợi ích của việc sử dụng Docker

Hiệu quả phát triển: Docker giúp tạo ra môi trường phát triển đồng nhất, giảm thiểu các vấn đề phát sinh do khác biệt môi trường.

Tính nhất quán: Docker container đảm bảo rằng ứng dụng chạy nhất quán trên mọi môi trường, từ phát triển đến thử nghiệm và sản xuất.

Bảo mật: Container cung cấp sự cách ly giữa các ứng dụng, giảm thiểu nguy cơ bảo mật.

Khả năng mở rộng: Docker giúp dễ dàng mở rộng và quản lý ứng dụng thông qua các container, cho phép xử lý tải cao một cách hiệu quả.

CHƯƠNG 3 XÁC ĐỊNH YÊU CẦU

3.1 Đặc tả mục tiêu của dự án

Dự án này nhằm mục đích xây dựng một trang web để cung cấp thông tin chi tiết về các món ăn và địa điểm phục vụ. Trang web sẽ giúp cho người dùng dễ dàng tìm kiếm, khám phá thông tin, tìm hiểu chi tiết về các món đặc sản. Đối với quản trị viên sẽ có quyền thêm mới các món ăn và các địa điểm phục vụ vào danh sách, xóa các món ăn và các địa điểm phục vụ ra khỏi danh sách, sửa và cập nhật thông tin các món ăn và các địa điểm phục vụ trong danh sách, đồng thời trang web còn cung cấp một giao diện thân thiện và dễ sử dụng.

3.2 Xác định personas

3.2.1 Người dùng thông thường (User)

Mục tiêu:

- Tìm kiếm các món ăn mới để trải nghiệm.
- Tìm địa điểm ăn uống phù hợp.

Hành vi:

- Thường xuyên sử dụng điện thoại để tìm kiếm thông tin về món ăn và địa điểm phục vụ
- Ưa thích các ứng dụng có giao diện thân thiện và dễ sử dụng.

3.2.2 Quản trị viên (Admin)

Mục tiêu:

- Quản lý và cập nhật thông tin về các món ăn và địa điểm phục vụ.

Hành vi:

- Sử dụng máy tính để bàn và đôi khi là máy tính bảng để quản lý thông tin.
- Cần một hệ thống quản lý nội dung dễ sử dụng, cho phép cập nhật nhanh chóng và chính xác.

3.3 Xác định user stories

3.3.1 Người dùng thông thường (User)

Tìm kiếm món ăn: có thể nhập tên món ăn vào thanh tìm kiếm, để nhanh chóng tìm thấy thông tin về món ăn đó.

Tiêu chí:

- Thanh tìm kiếm hiển thị ngay trên trang chủ.
- Người dùng nhập tên món ăn và nhấn nút tìm kiếm.
- Kết quả tìm kiếm hiển thị danh sách các món ăn phù hợp với từ khóa.

Xem chi tiết món ăn: xem chi tiết thông tin của từng món ăn, có thể biết nguyên liệu, cách chế biến và địa điểm phục vụ.

Tiêu chí:

- Danh sách món ăn hiển thị với hình ảnh và mô tả ngắn.
- Người dùng nhấp vào món ăn để xem chi tiết.
- Trang chi tiết món ăn hiển thị thông tin đầy đủ bao gồm nguyên liệu, cách chế biến, giá cả và địa điểm phục vụ.

Xem chi tiết địa điểm: là một người dùng, tôi muốn xem thông tin chi tiết về các địa điểm phục vụ món ăn, để tôi có thể quyết định nơi mình sẽ đến.

Tiêu chí:

- Danh sách địa điểm hiển thị với tên và mô tả ngắn.
- Người dùng nhấp vào địa điểm để xem chi tiết.
- Trang chi tiết địa điểm hiển thị thông tin như địa chỉ, giờ mở cửa, menu và đánh giá từ người dùng.

3.3.2 Quản trị viên (Admin)

Thêm mới món ăn: có thể thêm mới các món ăn vào danh sách, để cập nhật thông tin cho người dùng.

Tiêu chí:

- Giao diện quản lý hiển thị nút "Thêm món ăn mới".
- Quản trị viên nhập các thông tin cần thiết (tên, mô tả, hình ảnh, nguyên liệu, cách chế biến, địa điểm phục vụ).
- Món ăn mới được thêm vào danh sách và hiển thị trên ứng dụng.

Thêm mới địa điểm phục vụ: có thể thêm mới các địa điểm phục vụ vào danh sách, để cập nhật thông tin cho người dùng.

Tiêu chí:

- Giao diện quản lý hiển thị nút "Thêm địa điểm mới".
- Quản trị viên nhập các thông tin cần thiết (tên, địa chỉ, mô tả, hình ảnh, menu).
- Địa điểm mới được thêm vào danh sách và hiển thị trên ứng dụng.

Xóa món ăn và địa điểm phục vụ: có thể xóa các món ăn và địa điểm phục vụ khỏi danh sách, để quản lý thông tin chính xác và cập nhật.

Tiêu chí:

- Giao diện quản lý hiển thị danh sách các món ăn và địa điểm với nút "Xóa".
- Quản trị viên nhấp vào nút "Xóa" để xóa món ăn hoặc địa điểm khỏi danh sách.
- Món ăn hoặc địa điểm bị xóa không còn hiển thị trên ứng dụng.

3.4 Xác định các product backlog

Nội dung công việc:

1. Thu thập dữ liệu đặc sản tỉnh Trà Vinh
2. Thiết kế giao diện website
3. Thiết kế cơ sở dữ liệu
4. Xây dựng giao diện trang quản trị
5. Cài đặt cơ sở dữ liệu

-
6. Cài đặt hiển thị dữ liệu, các chức năng thêm, xóa, sửa dữ liệu
 7. Hoàn thiện và kiểm thử trang quản trị
 8. Xây dựng giao diện trang chủ
 9. Cài đặt hiển thị dữ liệu lên trang chủ
 10. Xây dựng giao diện trang món ăn, trang chi tiết
 11. Cài đặt hiển thị dữ liệu lên trang món ăn, trang chi tiết
 12. Xây dựng giao diện trang giới thiệu, trang địa điểm phục vụ
 13. Kiểm thử giao diện người dùng
 14. Cài đặt chức năng tìm kiếm
 15. Xây dựng và cài đặt chức năng trang đăng nhập
 16. Phân trang cho sản phẩm
 17. Hoàn thiện và kiểm thử chức năng

3.5 Xác định các nhu cầu phi tính năng

Giao diện thân thiện với người dùng: Giao diện dễ đọc và dễ sử dụng, người dùng có thể tìm kiếm món ăn một cách nhanh chóng. Các nút điều hướng cần được thiết kế một cách rõ ràng và dễ sử dụng để người dùng có thể dễ dàng điều hướng giữa các trang.

Quản lý nội dung dễ dàng: nội dung và hình ảnh dễ được cập nhật mới một cách thuận lợi.

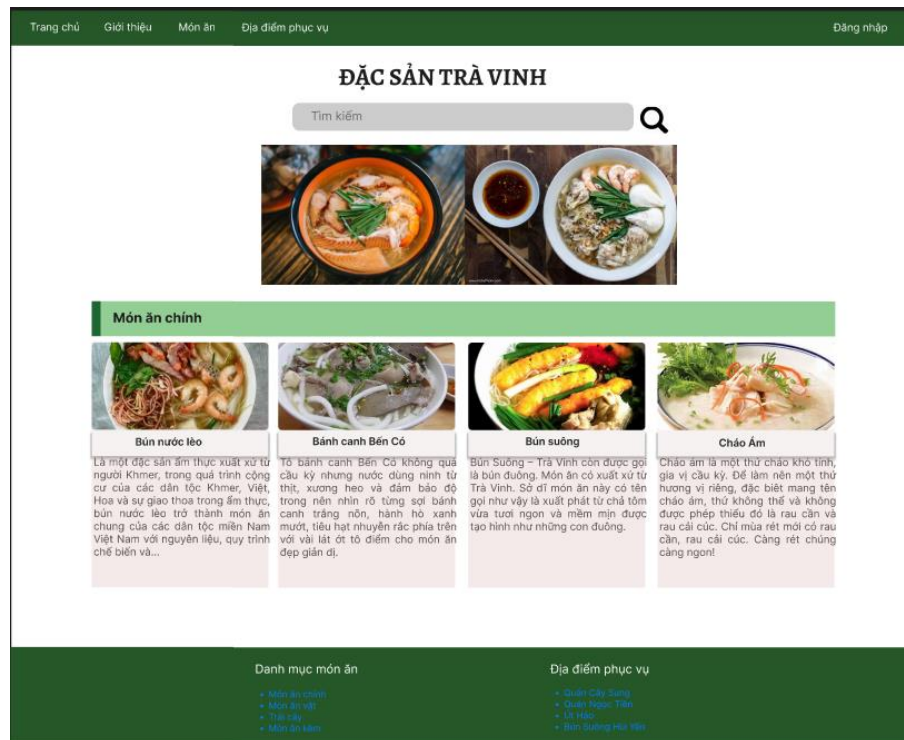
CHƯƠNG 4 LẬP KẾ HOẠCH SCRUM

Sprint	Nội dung công việc	Điểm	Thời gian bắt đầu	Thời gian kết thúc	Người thực hiện
Thu thập và thiết kế giao diện	Thu thập dữ liệu đặc sản tỉnh Trà Vinh	4	03/05/2024	04/05/2024	Diệp Tú Như
	Thiết kế giao diện website	4	05/05/2024	06/05/2024	Trần Văn Sang
	Thiết kế cơ sở dữ liệu	4	05/05/2024	06/05/2024	Võ Thị Diễm
Xây dựng và cài đặt trang quản trị	Xây dựng giao diện trang quản trị	4	07/05/2024	08/05/2024	Diệp Tú Như
	Cài đặt cơ sở dữ liệu	4	07/05/2024	08/05/2024	Võ Thị Diễm
	Cài đặt hiển thị dữ liệu, các chức năng thêm, xóa, sửa dữ liệu	6	09/05/2024	11/05/2024	Võ Thị Diễm
	Hoàn thiện và kiểm thử trang quản trị	4	12/05/2024	13/05/2024	Trần Văn Sang
Xây dựng và cài đặt trang người dùng	Xây dựng giao diện trang chủ	4	14/05/2024	15/05/2024	Diệp Tú Như
	Cài đặt hiển thị dữ liệu lên trang chủ	4	16/05/2024	17/05/2024	Võ Thị Diễm
	Xây dựng giao diện trang món ăn, trang chi tiết	4	18/05/2024	19/05/2024	Trần Văn Sang
	Cài đặt hiển thị dữ liệu lên trang món ăn, trang chi tiết	4	20/05/2024	21/05/2024	Diệp Tú Như
	Xây dựng giao diện trang giới thiệu, trang địa điểm phục vụ	4	20/05/2024	21/05/2024	Trần Văn Sang
	Kiểm thử giao diện người dùng	4	21/05/2024	22/05/2024	Võ Thị Diễm

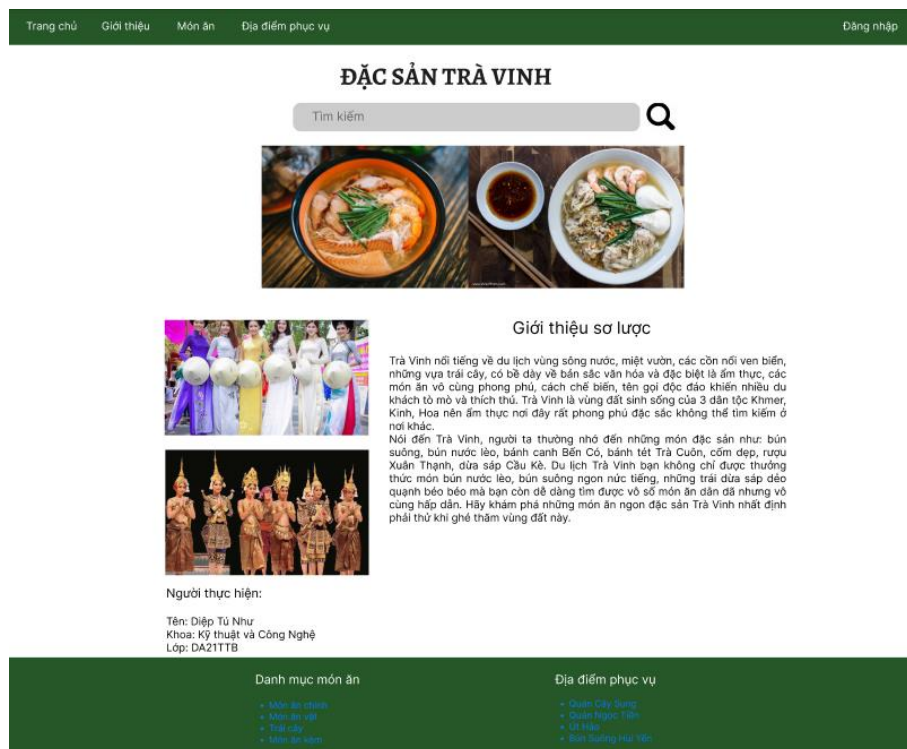
Xây dựng và cài đặt các chức năng	Cài đặt chức năng tìm kiếm	4	23/05/2024	24/05/2024	Diệp Tú Như
	Xây dựng và cài đặt chức năng trang đăng nhập	6	25/05/2024	26/05/2024	Võ Thị Diễm
	Phân trang cho sản phẩm	4	27/05/2024	28/05/2024	Võ Thị Diễm
	Hoàn thiện và kiểm thử chức năng	4	29/05/2024	30/05/2024	Trần Văn Sang

CHƯƠNG 5 HIỆN THỰC HÓA KẾ HOẠCH

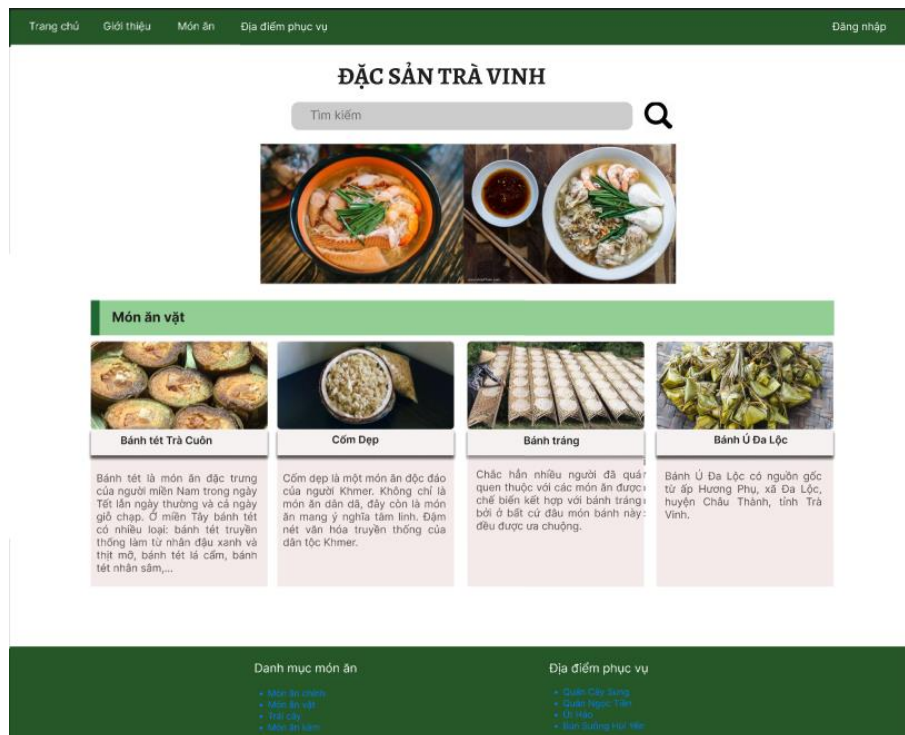
5.1 Giao diện thiết kế bằng figma



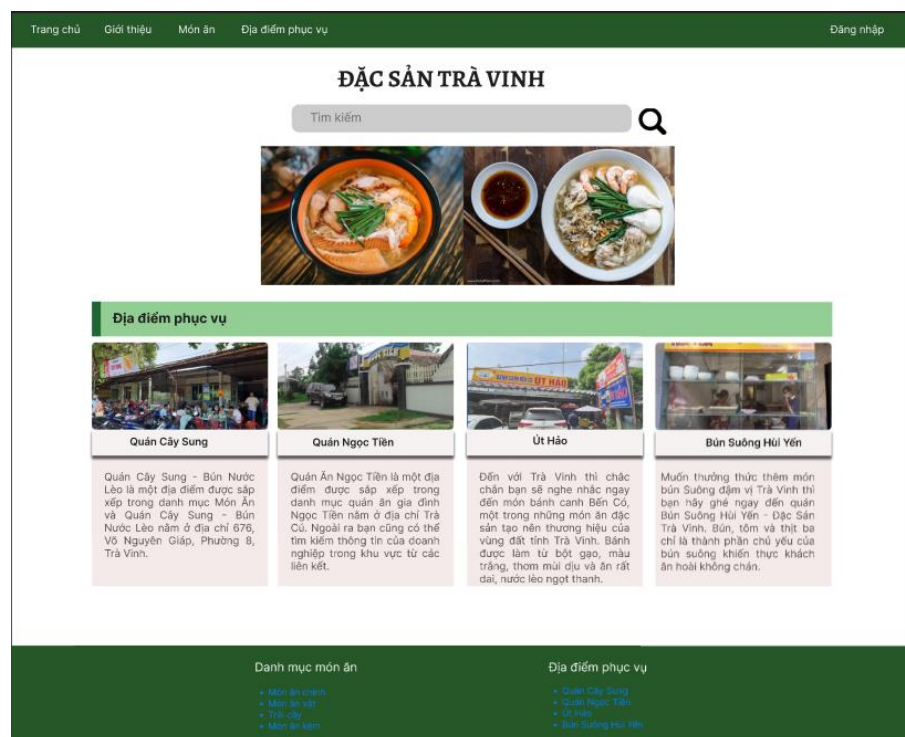
Hình 5.1 Giao diện trang chủ của trang người dùng



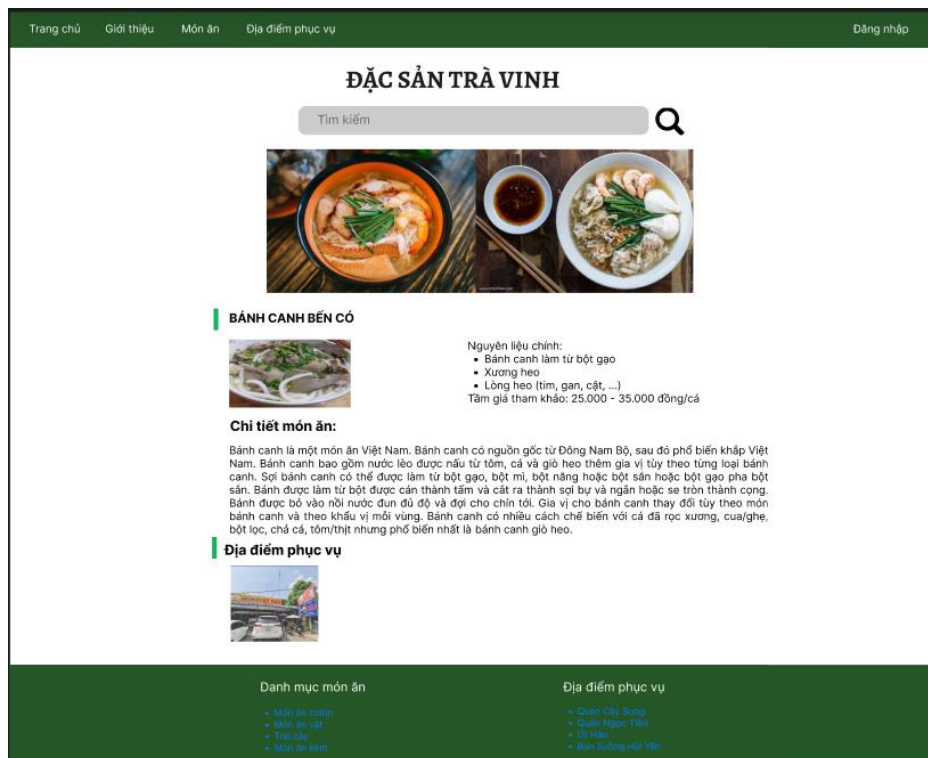
Hình 5.2 Giao diện trang giới thiệu



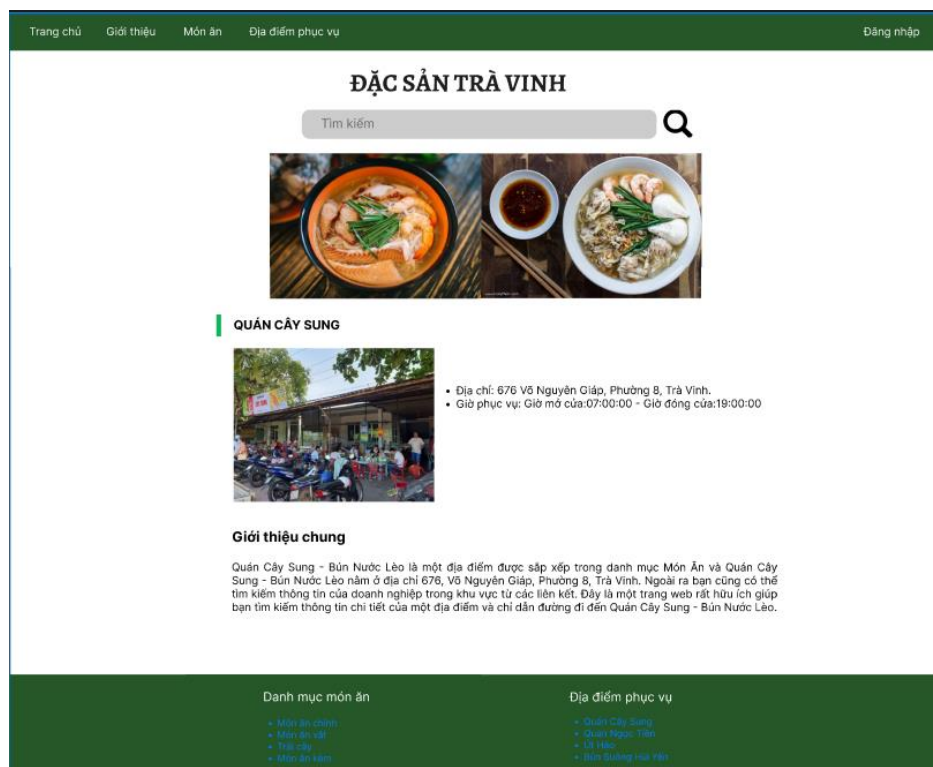
Hình 5.3 Giao diện trang món ăn



Hình 5.4 Giao diện trang địa điểm phục vụ

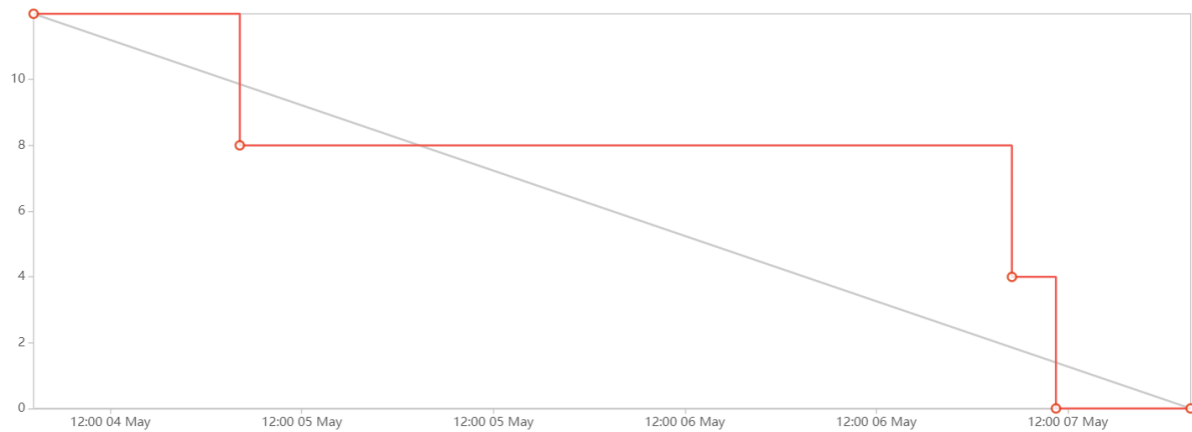


Hình 5.5 Giao diện trang chi tiết món ăn



Hình 5.6 Giao diện trang chi tiết điểm phục vụ

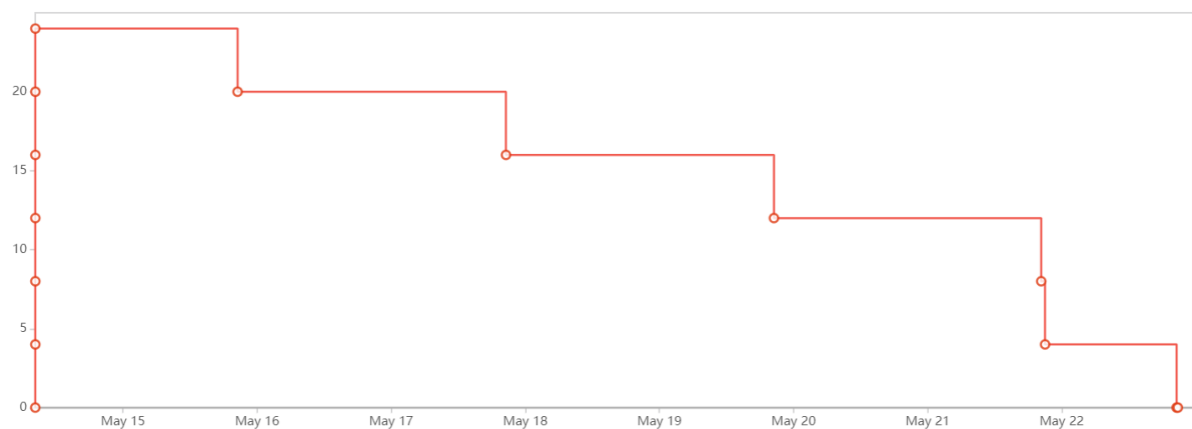
5.2 Burndown chart



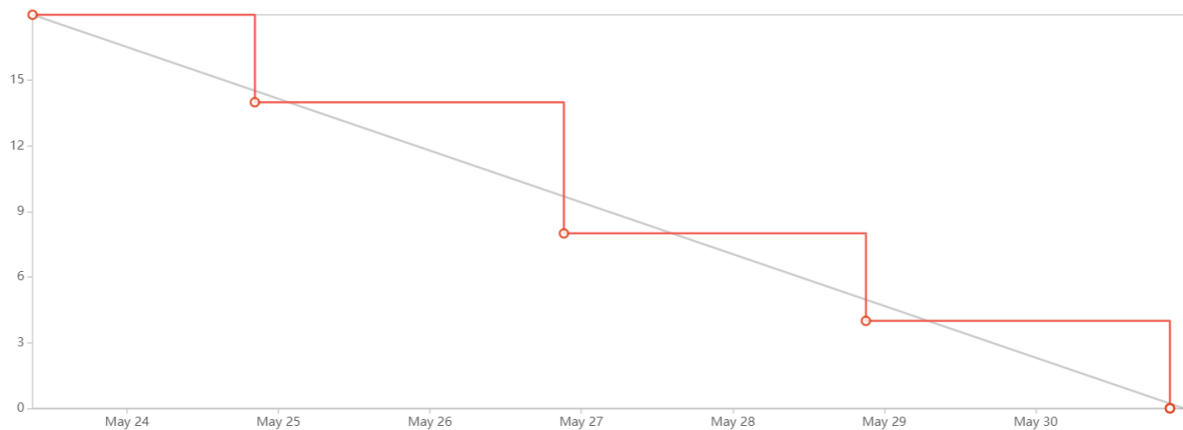
Hình 5.7 Sprint 1: Thu thập và thiết kế giao diện



Hình 5.8 Sprint 2: Xây dựng và cài đặt trang quản trị



Hình 5.9 Sprint 3: Xây dựng và cài đặt trang người dùng



Hình 5.10 Sprint 4: Xây dựng và cài đặt các chức năng

5.3 Mã nguồn

5.3.1 Dockerfile

```
FROM php:8.1-apache
# Cập nhật các gói và cài đặt các tiện ích cần thiết
RUN apt-get update && apt-get install -y \
    build-essential \
    libpng-dev \
    libjpeg62-turbo-dev \
    libfreetype6-dev \
    libonig-dev \
    libxml2-dev \
    zip \
    unzip \
    git \
    curl
# Cài đặt các phần mở rộng PHP cần thiết
RUN docker-php-ext-install pdo_mysql mbstring exif pcntl
# Cài đặt Composer
COPY --from=composer:latest /usr/bin/composer /usr/bin/composer
# Cài đặt các gói của Laravel
WORKDIR /var/www/html
COPY ./Software-technology /var/www/html
RUN composer install
```

```
# Thiết lập quyền cho các thư mục storage và cache
RUN chown -R www-data:www-data /var/www/html/storage
/var/www/html/bootstrap/cache
# Bật mod_rewrite cho Apache
RUN a2enmod rewrite
# Copy file cấu hình Apache
COPY ./apache/laravel.conf /etc/apache2/sites-available/000-default.conf
# Expose port 80
EXPOSE 80
```

5.3.2 Docker compose

```
services:
  laravel-docker:
    container_name: laravel-docker
    build: .
    volumes:
      - ./Software-technology:/var/www/html
    ports:
      - 8000:80
    networks:
      - laravel
    depends_on:
      - mysql_db

  mysql_db:
    container_name: mysql_db
    image: mysql:latest
    ports:
      - 3307:3306
    environment:
      MYSQL_ROOT_PASSWORD: root
      MYSQL_DATABASE: db_specialty
    networks:
      - laravel

  phpmyadmin:
    image: phpmyadmin:latest
```

```
ports:
```

```
  - 8001:80
```

```
environment:
```

```
  - PMA_ARBITRARY=1
```

```
networks:
```

```
  - laravel
```

```
networks:
```

```
  laravel:
```

```
    driver: bridge
```

CHƯƠNG 6 KẾT LUẬN

6.1 Kết quả đạt được

Thu thập dữ liệu về một số loại đặc sản tỉnh Trà Vinh

Thiết kế cơ sở dữ liệu lưu trữ

Cài đặt các chức năng cho phép thêm xóa, sửa dữ liệu, tìm kiếm dữ liệu

Xây dựng Website giới thiệu về một số loại đặc sản tỉnh Trà Vinh với ngôn ngữ HTML, CSS, thư viện W3.CSS

Hoàn thành được theo tiến độ Sprint đã đề ra

6.2 Hướng phát triển

Mở rộng các tính năng mới cho website như: đăng ký, bình luận, chat. Tìm hiểu Google API để bổ sung tính năng cung cấp các thông tin địa điểm phục vụ thông qua Google Map.

Tối ưu hóa trang web cho trải nghiệm người dùng tốt hơn.

TÀI LIỆU THAM KHẢO

- [1] <https://base.vn/blog/scrum-la-gi/>
- [2] <https://hocvienagile.com/quan-tri-du-an-cong-nghe-thong-tin-agile-scrum/>
- [3] <https://base.vn/blog/agile-la-gi/>
- [4] <https://www.scrum.org/learning-series/what-is-scrum/>
- [5] <https://www.projectmanager.com/blog/scrum-artifacts>