

KHOA KỸ THUẬT VÀ CÔNG NGHỆ
BỘ MÔN CÔNG NGHỆ THÔNG TIN



ĐỒ ÁN KẾT THÚC MÔN
CÔNG NGHỆ PHẦN MỀM

ĐỀ TÀI:
**PHÁT TRIỂN ỨNG DỤNG
QUẢN LÝ CỬA HÀNG HOA TƯƠI
THEO PHƯƠNG PHÁP AGILE**

Giảng viên hướng dẫn:

TS. Nguyễn Bảo Ân

Nhóm sinh viên thực hiện:

Trần Khánh Duy - 110121150

Lê Thị Nguyễn Xuyên - 110121174

Lê Trục Tín - 110121137

Mã Lớp DA21TTC

Trà Vinh, tháng 06 năm 2024

LỜI CẢM ƠN

Em xin chân thành gửi lời cảm ơn đến quý thầy cô trường Đại Học Trà Vinh nói chung và các thầy cô bộ môn trong khoa kỹ thuật và công nghệ nói riêng đã tạo điều kiện cho chúng em cơ hội thực hành, tiếp xúc để chúng em có thể tránh được những vướng mắc và bỏ ngõ trong môi trường công việc thời gian tới.

Em xin chân thành cảm ơn Ts.Nguyễn Bảo Ân. Nhờ sự giúp đỡ tận tình và những chỉ bảo của Thầy từ lúc bắt đầu cho tới lúc kết thúc đồ án mà em đã hoàn thành đúng thời hạn quy định và tích lũy được cho mình một lượng nền tảng kiến thức quý báu.

Mặc dù đã cố gắng hoàn thành đề tài tốt nhất nhưng do thời gian và kiến thức chuyên ngành còn hạn chế nên em vẫn còn nhiều thiếu sót khi tìm hiểu, đánh giá, và trình bày về đề tài. Rất mong nhận được sự quan tâm, góp ý của các thầy cô giảng viên bộ môn để đề tài của em được hoàn chỉnh và đầy đủ hơn.

Em xin chân thành cảm ơn

Trà Vinh, ngày tháng năm

Nhóm thực hiện:

Trần Khánh Duy - 110121150

Lê Thị Nguyễn Xuyên - 110121174

Lê Trực Tín - 110121137

MỤC LỤC

CHƯƠNG 1: GIỚI THIỆU	8
1.1 Lý do chọn đề tài.....	8
1.2 Mô tả bài toán và đặc tả đề tài	8
1.2.1 Mô tả bài toán	8
1.2.2 Đặc tả đề tài	8
1.3. Mục tiêu nghiên cứu	9
CHƯƠNG 2: CƠ SỞ LÝ THUYẾT	10
2.1 Tổng quan về Agile.....	10
2.1.1 Agile là gì?.....	10
2.1.2 Agile Software Development là gì?.....	10
2.1.3 Các phương pháp của Agile.....	11
2.2 Lợi ích của Agile.....	12
2.3 Đặc điểm của phương pháp Agile.....	13
2.4 Tổng quan về Scrum	14
2.4.1 Scrum là gì?	14
2.4.2 Ba trụ cột của Scrum.....	15
2.4.3 Đặc điểm của nhóm Scrum	16
2.4.4 Vai trò trong nhóm Scrum	17
2.4.5 Các sự kiện trong Scrum.....	17
2.4.6 Các tạo tác trong Scrum.....	19
2.4.7 Các giá trị của Scrum.....	19
2.5 Tổng quan về công nghệ, kiến trúc	21
2.5.1 Tổng quan về Git	21
2.5.1.1 Git là gì?	21
2.5.1.2 Lợi ích của Git.....	21
2.5.2 Tổng quan về Figma	22
2.5.2.1 Figma là gì ?	22
2.5.2.2 Ưu điểm nổi bật của Figma	22
2.5.3 Tổng quan về Jira.....	24
2.5.3.1 Jira là gì?	24
2.5.3.2 Các tính năng cơ bản của Jira.....	25

2.5.3.3 Ưu điểm của Jira.....	25
2.5.3.4 Nhược điểm của Jira.....	26
2.5.3.5 Các thuật ngữ cần biết khi sử dụng Jira	26
2.5.4 Tổng quan Scrum Master	30
2.5.4.1 Scrum Master là gì?.....	30
2.5.4.2 Vai trò và công việc của một Scrum Master	30
CHƯƠNG 3: XÁC ĐỊNH NHU CẦU.....	32
3.1 Product Backlog.....	32
3.2 Sprint Backlog	34
3.3 User stories.....	34
3.4 Technical Backlog	35
3.5 Support Backlog.....	36
CHƯƠNG 4: LẬP KẾ HOẠCH SCRUM	37
CHƯƠNG 5: HIỆN THỰC HÓA KẾ HOẠCH	40
5.1 Thiết kế xử lý	40
5.1.1 Sơ đồ phân rã chức năng.....	40
5.1.2 Sơ đồ luồng dữ liệu.....	40
5.2 Thiết kế database.....	42
5.2.1 Mô hình thực thể kết hợp (ERD)	42
5.2.2 Mô hình vật lý.....	43
5.3 Cơ sở dữ liệu	43
5.4 SPRINT	47
5.4.1 Thiết kế các giao diện bằng figma.....	47
5.4.2 Commit code cho từng công việc theo bảng phân công	52
5.4.3 Burndown Chart.....	54
CHƯƠNG 6: KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN.....	55
6.1.Kết luận	55
6.2. Hướng phát triển.	55
DANH SÁCH TÀI LIỆU THAM KHẢO	56

MỤC LỤC HÌNH

Hình 1. Sơ đồ phân rã chức năng	40
Hình 2. Sơ đồ Mức 0	40
Hình 3. Sơ đồ Mức 1	41
Hình 4. Sơ đồ ERD.....	42
Hình 5. Sơ đồ vật lý.....	43
Hình 6. Thiết kế giao diện trang Đăng nhập	47
Hình 7. Thiết kế giao diện Trang chủ	47
Hình 8. Thiết kế giao diện trang Nhà Cung Cấp	48
Hình 9. Thiết kế giao diện Tìm kiếm hóa đơn.....	48
Hình 10. Thiết kế giao diện trang Sản Phẩm.....	49
Hình 11. Thiết kế giao diện trang Khách Hàng.....	49
Hình 12. Thiết kế giao diện trang Nhân Viên	50
Hình 13. Thiết kế giao diện trang Hóa Đơn Bán Hàng.....	51
Hình 14. Commit của Lê Trực Tín theo nhánh "tinle"	52
Hình 15. Commit của Trần Khánh Duy theo nhánh "khanhduy"	53
Hình 16. Commit của Lê Thị Nguyễn Xuyên theo nhánh "nguyensexuyen"	53
Hình 17. Burndown Chart Sprint 1	54
Hình 18. Burndown Chart Sprint 2.....	54

[illegible]

Giảng viên hướng dẫn

5

NHẬN XÉT CỦA GIÁO VIÊN PHẢN BIỆN

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

BẢNG PHÂN CÔNG CÔNG VIỆC

STT	Nội dung thực hiện	Người thực hiện
1	Phần 1. Giới thiệu	Trần Khánh Duy
2	Phần 2. Cơ sở lý thuyết	Cả nhóm
	Lý thuyết về Agile và SCRUM	Trần Khánh Duy & Lê Thị Nguyễn Xuyên
	Lý thuyết về công nghệ, kiến trúc được sử dụng	Lê Trực Tín
3	Chương 3. Xác định nhu cầu	Cả nhóm
4	Chương 4. Lập kế hoạch SCRUM	Lê Trực Tín
5	Chương 5. Hiện thực hóa kế hoạch	Cả nhóm
6	Chương 6. Kết luận	Cả nhóm

CHƯƠNG 1: GIỚI THIỆU

1.1 Lý do chọn đề tài

Trong những năm gần đây, CNTT đang phát triển rất mạnh mẽ. Bên cạnh đó CNTT cũng giúp cho các ngành kinh doanh đều phát triển theo và giải quyết rất nhiều khó khăn và việc kinh doanh trở nên dễ dàng và thuận lợi hơn. Nhằm thay thế một số công việc mà trước đó phải thao tác bằng tay trên giấy tờ đạt hiệu quả không cao, mất nhiều thời gian và quản lý cửa hàng trở nên nhanh chóng và chính xác hơn. Từ những yêu cầu trên, nhóm em quyết định chọn đề tài kết thúc môn là "Xây dựng quản lý hệ thống cửa hàng hoa".

1.2 Mô tả bài toán và đặc tả đề tài

1.2.1 Mô tả bài toán

Xây dựng quản lý hệ thống cửa hàng hoa nhằm quản lý cửa hàng trở nên hiệu quả hơn. Hệ thống giúp quản lý các thông tin về nhân viên, khách hàng, nhập hàng, hóa đơn, các nhà cung cấp,...

1.2.2 Đặc tả đề tài

Một ứng dụng quản lý cửa hàng hoa có tính ứng dụng cao, cung cấp giá trị thực cho doanh nghiệp và người dùng cuối. Việc sử dụng phương pháp Agile giúp đảm bảo rằng sản phẩm được phát triển phản ánh đúng nhu cầu của người dùng và thị trường thông qua việc phát triển theo từng giai đoạn, cập nhật linh hoạt và phản hồi nhanh chóng.

Mỗi một phân hệ được thiết kế liên quan chặt chẽ như các mắt xích tạo ra tính chuyên nghiệp và khả năng tự động hoá cao. Chức năng cơ bản của hệ điều hành:

- **Quản lý thông tin nhà cung cấp**

Thiết lập các mã nhà cung cấp, tên nhà cung cấp, thêm, sửa thông tin nhà cung cấp,... nhằm đảm bảo việc kiểm soát và thông tin của nhà cung cấp để tiện liên lạc.

- **Quản lý thông tin khách hàng**

Cập nhật những thông tin về khách hàng đặt hoa. Từ đó lưu thông tin khách hàng cho cửa hàng để thuận tiện trong việc giao hoa hoặc trao đổi thắc mắc của khách hàng. Cửa hàng sẽ lưu lại với các thông tin: Họ và tên khách hàng, Địa chỉ, số điện thoại,...

1.3. Mục tiêu nghiên cứu

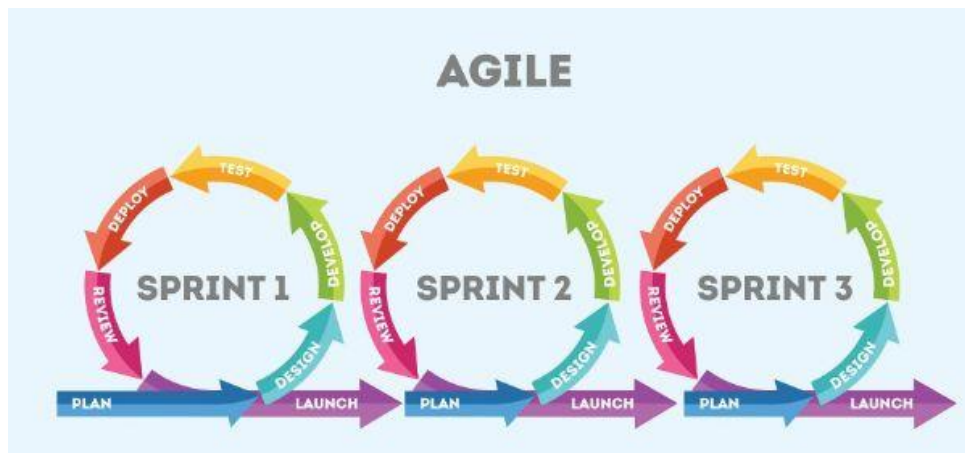
Mục tiêu chính của nghiên cứu là xây dựng một ứng dụng quản lý cửa hàng hoa linh hoạt và có khả năng đáp ứng nhanh chóng với các yêu cầu và thay đổi từ phía người dùng cũng như từ thị trường. Phương pháp Agile sẽ được áp dụng để chia dự án thành các giai đoạn ngắn, mỗi giai đoạn sẽ tập trung vào việc phát triển một phần nhỏ của ứng dụng và thu thập phản hồi từ người dùng.

CHƯƠNG 2: CƠ SỞ LÝ THUYẾT

2.1. Tổng quan về Agile

2.1.1 Agile là gì?

Agile thực chất là một triết lý hay một khung tư duy để nhanh chóng thích ứng và phản hồi với thay đổi, từ đó đạt được thành công trong một môi trường liên tục biến động và không chắc chắn.



Triết lý **Agile** xuất phát từ ngành công nghệ và được mô tả bằng 4 giá trị và 12 nguyên lý cốt lõi trong Tuyên ngôn phát triển phần mềm linh hoạt hay Tuyên ngôn Agile (The Manifesto for **Agile** Software Development)

2.1.2 Agile Software Development là gì?

Agile Software Development là một thuật ngữ chung chỉ tất cả các kỹ thuật và phương pháp phát triển phần mềm theo triết lý Agile.

Triết lý Agile được mô tả sơ bộ trong bản Tuyên ngôn Agile (The Manifesto for Agile Software Development) thông qua những giá trị cốt lõi và nguyên tắc có tính phổ quát, tuy nhiên không ghi rõ thực hiện những giá trị và nguyên tắc ấy như thế nào. Vì vậy, các phương pháp Agile sẽ làm nhiệm vụ định nghĩa rõ hơn để các cá nhân và tập thể dễ dàng vận dụng vào bối cảnh công việc của mình. Các phương pháp này đều khuyến khích việc lập kế hoạch thích ứng, phát triển tăng dần, chuyển giao sớm và cải tiến liên tục nhằm thích ứng nhanh với sự thay đổi – một điểm yếu cố hữu của các phương pháp phát triển phần mềm truyền thống (Waterfall).

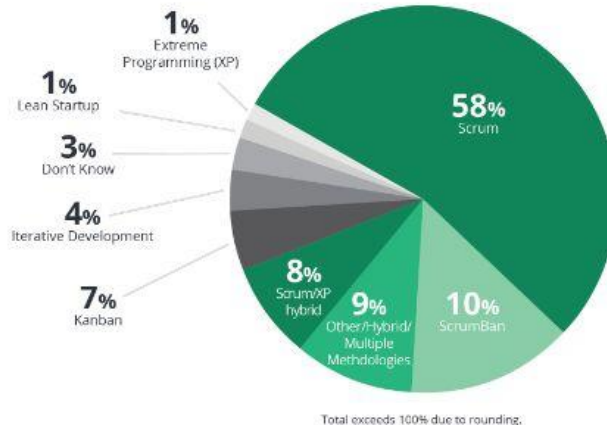
2.1.3 Các phương pháp của Agile

Agile có thể có nhiều phương pháp để áp dụng thực hành khác nhau, nhưng triết lý chung thì giống nhau. Theo khảo sát của VersionOne năm 2020, tỉ lệ áp dụng các phương pháp Agile được mô tả trong biểu đồ dưới đây:

AGILE METHODS AND PRACTICES

AGILE METHODOLOGIES USED

Scrum and related variants continue to be the most common Agile methodologies used by respondents' organizations.



Chúng ta cùng điểm qua về một số phương pháp **Agile** (gọi chung với phạm vi rộng hơn để chỉ cả phương pháp, khung quản trị, kỹ thuật thực hành) phổ biến nhất trong số này:

- **Scrum**: theo Tài liệu Hướng dẫn Scrum (The Scrum Guide) được 2 nhà đồng sáng lập Ken Schwaber and Jeff Sutherland định nghĩa, là một khung làm việc (framework) để phát triển bền vững các sản phẩm phức tạp. Có thể nói Scrum là một trong những phương pháp Agile quan trọng nhất sử dụng cơ chế lặp (iterative) và tăng trưởng (Incremental) để tối ưu hóa hiệu quả cũng như kiểm soát rủi ro. Chúng ta sẽ tìm hiểu chi tiết về Scrum ở phần II của series bài viết về nhập môn **Agile** và Scrum.
- **Kanban**: là một phương pháp **Agile** dựa trên Phương thức Sản xuất Toyota với bốn nguyên lý: Trực quan hóa công việc, giới hạn công việc đang làm, tập trung vào luồng làm việc, cải tiến liên tục. Mô hình Kanban phù hợp cho việc hỗ trợ sản xuất trong quá trình làm việc.
- **Scrumban**: là một phương pháp được Corey Ladas giới thiệu vào năm 2009 trong cuốn sách với tựa đề “Scrumban – Essays on Kanban Systems for Lean Software Development”. Scrumban kết hợp được những ưu điểm của Scrum

và Kanban để cho phép nhóm liên tục cải tiến quy trình và khả năng xử lý công việc.

- **Lean Software Development (LSD):** hay Phát triển phần mềm tinh gọn là hình thức áp dụng Tư duy tinh gọn (Lean Thinking) và các nguyên lý đặc trưng của Tinh gọn (xuất phát từ ngành sản xuất ô tô – Lean Manufacturing) cho lĩnh vực phát triển phần mềm. Thuật ngữ Lean Software Development có nguồn gốc từ một cuốn sách cùng tên của Mary Poppendieck và Tom Poppendieck. Trong đó, bảy nguyên lý diễn giải tư duy Tinh gọn bao gồm: Loại bỏ lãng phí, Khuếch trương việc học, Quyết định càng muộn càng tốt, Chuyển giao càng nhanh càng tốt, Trao quyền cho nhóm, Tạo ra tính toàn vẹn tự thân, Thấy toàn cảnh là linh hồn cho quá trình phát triển phần mềm tinh gọn. Tìm hiểu thêm về Lean Software Development tại đây.
- **XP (Extreme Programming)** – Hay lập trình cực hạn là một phương pháp phát triển phần mềm thuộc họ **Agile** được phát minh bởi Ken Beck – một kỹ sư phần mềm người Mỹ. XP hướng đến việc nâng cao chất lượng phần mềm và khả năng đáp ứng với thay đổi yêu cầu người dùng. XP chủ trương đưa ra các bản phát hành thường xuyên thông qua các chu trình phát triển ngắn. Một số các thực hành của XP như: Lập trình cặp (Pair programming), Tái cấu trúc mã nguồn (Refactoring), Kiểm thử đơn vị (Unit Testing), Tích hợp liên tục (Continuous Integration), Các bản phát hành nhỏ (Small Release)...

Có thể nhận thấy, trong số các phương pháp **Agile**, Scrum thuộc loại phổ biến nhất bởi sự hiệu quả và tối ưu của nó. Theo khảo sát ở trên, Scrum và các phương pháp lai với Scrum như Scrumban, Scrum và XP chiếm gần ¾ mức độ phổ biến. Đó là lý do rất nhiều nhóm bắt đầu quá trình tiếp nhận Agile với việc sử dụng Scrum.

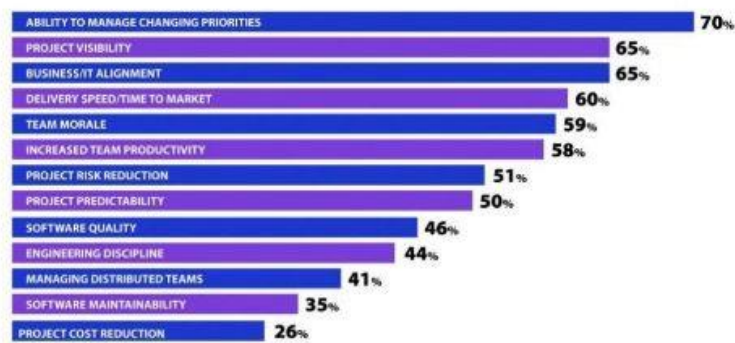
2.2 Lợi ích của Agile

Agile là triết lý với các phương pháp mới thay thế cho phương pháp theo mô hình truyền thống (Waterfall) đã khẳng định vị thế khi đem đến cho cá nhân và tổ chức những lợi ích nhất định. Vậy những lợi ích đó là gì, tại sao thế giới đang chuyển mình rất nhanh để thích ứng với Agile? Khảo sát của VersionOne năm 2020 về việc triển khai Agile đã cho thấy có sự cải thiện trong các lĩnh vực sau:

BENEFITS OF AGILE

BENEFITS OF ADOPTING AGILE

We continue to see many benefits realized by companies adopting Agile. The theme of the top 5 reported benefits is speed and adaptability. This corresponds with the top reported reasons for adopting Agile.



2.3 Đặc điểm của phương pháp Agile

- **Tính lặp (Iterative):** Trong khi dự án thực hiện, các phân đoạn sẽ được lặp đi lặp lại (Iteration hoặc Sprint). Các phân đoạn này diễn ra trong thời gian ngắn (thường từ một đến bốn tuần). Trong mỗi phân đoạn này, nhóm phát triển thực hiện đầy đủ các công việc cần thiết như lập kế hoạch, phân tích yêu cầu, thiết kế, triển khai và kiểm thử để có được phần nhỏ của sản phẩm. Các phương pháp Agile sẽ không lập kế hoạch dài hạn, thay vào đó sẽ phân chia thành những quá trình lập kế hoạch nhỏ, đơn giản và gọn nhẹ.
- **Tính tăng trưởng (Incremental):** Cuối mỗi phân đoạn (Sprint), nhóm phát triển thường cho ra các phần nhỏ của sản phẩm cuối cùng. Các phần nhỏ này thường đáp ứng được các yêu cầu, có khả năng chạy tốt do đã được kiểm thử cẩn thận và có thể sử dụng được ngay. Theo thời gian, các phân đoạn sẽ tiếp nối nhau và tích lũy dần tới khi toàn bộ yêu cầu của khách hàng được thỏa mãn. Khác với mô hình Waterfall – vốn chỉ cho phép nhìn thấy sản phẩm tới khi gần hoàn thành dự án, sản phẩm trong dự án Agile sẽ được phát triển lớn dần theo thời gian, tăng trưởng cho tới khi đạt được trạng thái đủ để phát hành.

Vòng phản hồi ngắn và thích ứng thường xuyên: Do các phân đoạn chỉ kéo dài trong một khoảng thời gian ngắn, việc lập kế hoạch hay có những điều chỉnh, thay đổi trong quá trình phát triển đều có thể đáp ứng nhanh để phù hợp. Ngoài ra, việc khách hàng được tham gia vào các quy trình phát triển cũng sẽ giúp ích cho việc đáp ứng và thay đổi ngay những yêu cầu khác từ phía khách hàng.

Giao tiếp thường xuyên và hiệu quả: Trong các nhóm Agile luôn đề cao việc giao tiếp thường xuyên và trực diện hơn là việc trao đổi qua tài liệu, giấy tờ. Các nhóm phát triển cũng thường chỉ ở quy mô nhỏ (đối với Scrum là từ 3-9 người), từ đó sẽ đơn giản hóa được quá trình giao tiếp và thúc đẩy hợp tác hiệu quả hơn.

Hướng chất lượng: Đảm bảo chất lượng tuyệt đỉnh luôn là một yêu cầu quan trọng trong triết lý Agile. Rất nhiều kỹ thuật và công cụ được sử dụng để hướng đến việc nâng cao chất lượng sản phẩm, chẳng hạn như: Tích hợp Liên tục, Kiểm thử Đơn vị Tự động, Lập trình cặp, Phát triển Hướng Kiểm thử, Mẫu Thiết kế, Tái cấu trúc mã nguồn, v.v..

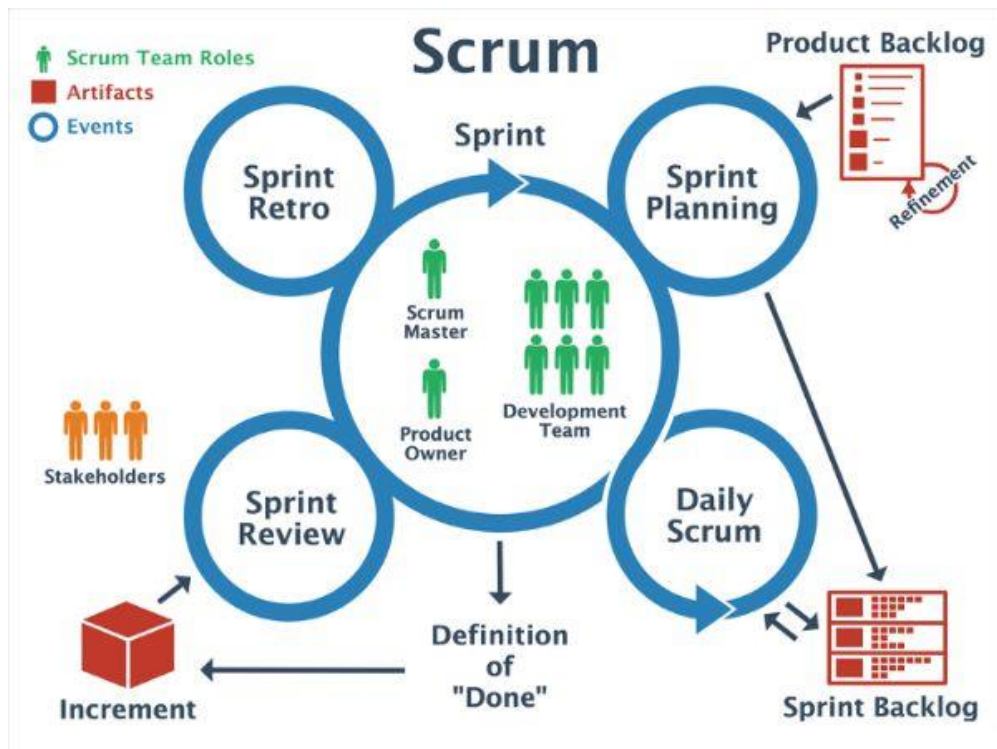
Phát triển dựa trên giá trị: Một trong những nguyên tắc cơ bản của Agile chính là “phần mềm chạy tốt là thước đo chính của tiến độ”. Nguyên tắc này giúp nhóm luôn cố gắng để đạt được kết quả cuối và có thể bỏ đi những công việc dư thừa không trực tiếp đem lại giá trị cho sản phẩm.

Theo cách tiếp cận truyền thống, phạm vi công việc sẽ cố định, thời gian và chi phí sẽ thay đổi để hoàn thành được phạm vi công việc. Theo cách tiếp cận của các phương pháp Agile, thời gian và chi phí sẽ là những phần cố định, khi đó các nhóm Agile luôn cộng tác trực tiếp và thường xuyên với khách hàng để liên tục ưu tiên những hạng mục tạo ra nhiều giá trị nhất. Nguyên tắc này giúp nhóm dám loại bỏ đi các công việc dư thừa không trực tiếp mang lại giá trị cho sản phẩm và rút ngắn thời gian để đi đến sản phẩm cuối cùng. Nhờ đó, các dự án Agile luôn gia tăng được sự hài lòng của khách hàng và cho ra những sản phẩm tối ưu nhất.

2.4 Tổng quan về Scrum

2.4.1 Scrum là gì?

Scrum là một phương pháp Agile dùng cho phát triển sản phẩm, đặc biệt là phát triển phần mềm. Scrum là một khung quản lý dự án được áp dụng rất rộng rãi, từ những dự án đơn giản với một nhóm phát triển nhỏ cho đến những dự án có yêu cầu rất phức tạp với hàng trăm người tham gia, và kể cả những dự án đòi hỏi khung thời gian cố định.



Trong Scrum, công việc được thực hiện bởi Nhóm Scrum thông qua từng phân đoạn lặp liên tiếp nhau được gọi là Sprint. Để hiểu được Scrum thì cần hiểu nguyên lý của Scrum, các Vai trò, Tạo tác, Sự kiện và sự vận hành của một vòng đời Scrum.

2.4.2 Ba trụ cột của Scrum

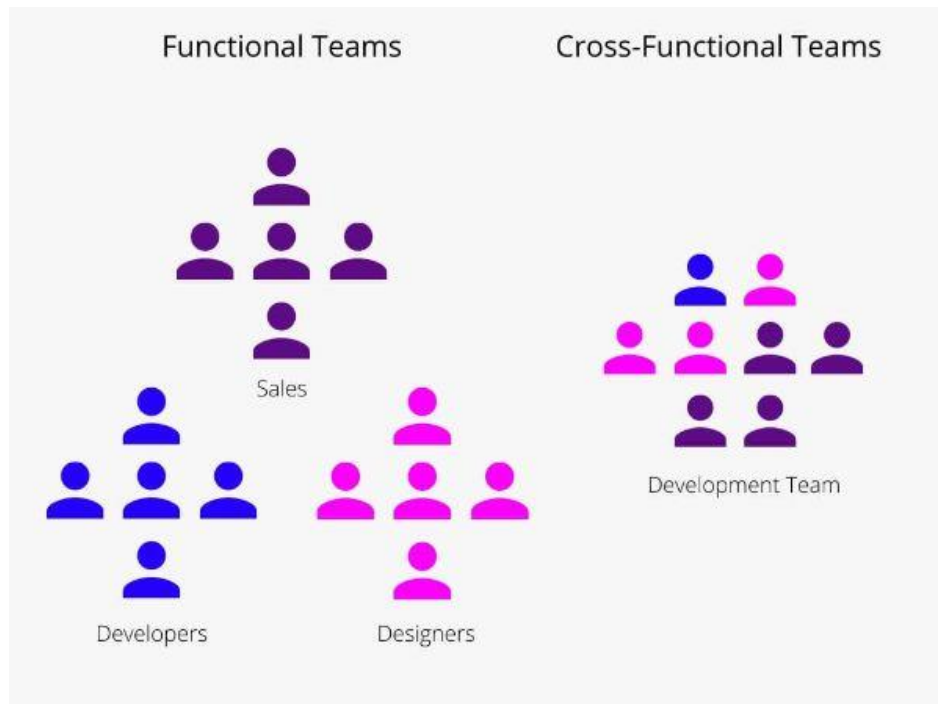
Ba trụ cột (hay ba chân) của **Scrum** là Tính minh bạch, Sự thanh tra và Sự thích nghi. Đây chính là phần lõi của khung làm việc **Scrum**, thiếu bất cứ trụ cột nào trong số này đều khiến khung **Scrum** không còn hoạt động đúng nữa.

- **Minh bạch (transparency):** Đầu tiên, thông tin liên quan tới quá trình phát triển phải minh bạch và thông suốt. Các thông tin đó có thể là: tầm nhìn (vision) về sản phẩm, yêu cầu khách hàng, tiến độ công việc, các khúc mắc rào cản,... Từ đó mọi người ở các vai trò khác nhau có đủ thông tin cần thiết để tiến hành các quyết định có giá trị nhằm nâng cao hiệu quả công việc. Các công cụ và cuộc họp trong **Scrum** luôn đảm bảo thông tin được minh bạch cho các bên.
- **Thanh tra (inspection):** Công tác thanh tra liên tục các hoạt động trong **Scrum** đảm bảo cho việc phát lộ các vấn đề cũng như giải pháp để thông tin đa dạng và hữu ích đến được với các bên tham gia quá trình phát triển. Truy xét kỹ càng và liên tục là cơ chế khởi đầu cho việc thích nghi và các cải tiến liên tục trong **Scrum**.

- Thích nghi (adaptation): Dựa trên các thông tin minh bạch hóa từ các quá trình thanh gia và làm việc, Scrum có thể phản hồi các thay đổi một cách tích cực, nhờ đó mang lại thành công cho sản phẩm. Các nỗ lực minh bạch và thanh tra đều hướng tới hành động thích ứng nhanh chóng và hiệu quả.

2.4.3 Đặc điểm của nhóm Scrum

Nhóm Scrum có 2 đặc điểm đó là tự quản (self-managing) và liên chức năng (cross-functional).



- Tự quản (self-managing): Đây là một thuật ngữ mới thay thế cho thuật ngữ cũ (*self-organized*) được cập nhật trong tài liệu Hướng dẫn **Scrum** mới nhất năm 2020. Điều này có nghĩa là nhóm sẽ cùng ra quyết định sẽ làm gì, ai sẽ làm và làm như thế nào mà không bị sự chỉ đạo bởi ai đó bên ngoài nhóm. Các Nhóm **Scrum** được trao quyền để quản lý công việc của họ nhằm hướng tới một mục tiêu chung là giúp tổ chức giải quyết các vấn đề phức tạp nhanh nhẹn hơn và tạo ra kết quả chất lượng hơn.
- Liên chức năng (*cross-functional*): Một nhóm liên chức năng bao gồm nhiều cá nhân với các chuyên môn khác nhau đủ năng lực được kết hợp lại cùng làm việc hướng tới một mục tiêu chung. Trong dự án, các cá nhân có thể đến từ nhiều phòng ban chức năng khác nhau, cũng có thể xuất phát từ bên ngoài. Nhưng khi đã thành một nhóm (team), thì các cá nhân làm việc tập trung cho

đội như là một đơn vị (unit) để hoàn tất mục tiêu chung. Bên trong nhóm liên chức năng không có các nhóm nhỏ khác.

2.4.4 Vai trò trong nhóm Scrum

Trong Scrum, có ba vai trò: Product Owner, Nhà Phát triển, và Scrum Master. Tất cả hợp thành Nhóm Scrum.

- **Product Owner:** là một trong ba vai trò trong nhóm Scrum. Vai trò này chịu trách nhiệm tối ưu hóa lợi nhuận trên đầu tư (ROI – Return On Investment) thông qua việc quyết định các tính năng của sản phẩm, đánh giá và sắp xếp độ ưu tiên của từng hạng mục, những hạng mục có độ ưu tiên cao thì sẽ được đưa vào phát triển trước, những hạng mục có độ ưu tiên thấp hơn thì sẽ được phát triển sau. Product Owner thường khác với một Giám đốc Sản phẩm truyền thống ở chỗ đó là Product Owner tham gia tích cực vào quá trình phát triển sản phẩm, thay vì chỉ quản lý và ủy quyền cho những người khác thực hiện các quyết định liên quan đến sản phẩm.
- **Scrum Master:** là một vai trò then chốt giúp nhóm Scrum làm việc hiệu quả bằng cách tuân thủ nguyên lý, các kỹ thuật và quy tắc của Scrum. Scrum Master không phải là người quản lý của Nhóm mà là một lãnh đạo theo phong cách phục vụ (Servant Leader). Scrum Master làm tất cả những gì trong thẩm quyền phục vụ Product Owner, Nhóm Phát triển, và Tổ chức đi đến thành công.
- **Nhóm phát triển:** là đội ngũ trực tiếp làm ra sản phẩm, họ bao gồm các chuyên gia có nhiệm vụ chuyển giao phần tăng trưởng ở cuối mỗi Sprint. Các Nhà phát triển không có sự phân chia các chức danh chuyên môn đặc thù cho từng thành viên, ví dụ như: kiểm thử viên, lập trình viên, chuyên gia thiết kế, chuyên gia cơ sở dữ liệu,... mà tất cả đều được gọi chung là Nhà phát triển. Việc này giúp nâng cao tính sở hữu tập thể, trách nhiệm tập thể và bình đẳng giữa các thành viên.

2.4.5 Các sự kiện trong Scrum

- **Sprint:**

Có thể nói Sprint là trái tim của **Scrum** và là khoảng thời gian cố định mà ở đó các Nhà Phát triển thực hiện công việc phát triển sản phẩm.

Sprint được đóng khung thời gian không dài hơn 1 tháng và thường thì không ngắn hơn một tuần. Các Sprint có độ dài như nhau và diễn ra liên tiếp nhau mà không bị gián đoạn. Sprint kết thúc khi thời gian đóng khung kết thúc, bất kể các công việc trong đó đã được hoàn thành hết hay chưa.

- **Lập kế hoạch Sprint (Sprint Planning):**

Là sự kiện diễn ra đầu Sprint để lên kế hoạch làm việc cho toàn bộ Sprint. Sự kiện này được chia làm 3 phần với 3 mục đích rõ ràng:

Phần 1: nhằm trả lời câu hỏi Why – “Tại sao chúng ta làm những thứ này?” – điều này được trình bày bởi Product Owner và sau đó cả nhóm sẽ thống nhất để xác định rõ Mục tiêu của Sprint (Sprint Goal)

Phần 2: trả lời cho câu hỏi What – “Chúng ta sẽ hoàn thành những gì?”. Thông qua việc trao đổi với Product Owner, các Nhà phát triển sẽ lựa chọn những hạng mục từ Product Backlog để phát triển trong Sprint hiện tại.

Phần 3: sẽ trả lời câu hỏi How – “Chúng ta sẽ làm như thế nào?”. Đối với mỗi hạng mục được lựa chọn, các Nhà phát triển sẽ lên kế hoạch các công việc cụ thể để hoàn thành được mục tiêu Sprint (Sprint Goal).

3 câu hỏi Why – What – How sẽ giúp 1 Sprint diễn ra hiệu quả và rõ ràng hơn. Các Nhà Phát triển có quyền quyết định lựa chọn những hạng mục mà mình sẽ làm, không ai được phép can thiệp và gán công việc cho nhóm, kể cả Product Owner hay các lãnh đạo khác. Kết quả của buổi Lập kế hoạch Sprint là: Mục tiêu Sprint và Sprint Backlog.

- **Scrum Hằng ngày (Daily Scrum):**

Là buổi gặp mặt ngắn 15 phút hằng ngày của tất cả các thành viên Nhóm Phát triển để thanh tra và tái lập kế hoạch cho nhóm. Để giữ đơn giản và tạo thói quen thì các buổi Scrum Hằng ngày phải diễn ra tại cùng một địa điểm vào cùng một khung thời gian. Scrum Master không bắt buộc tham dự nhưng phải đảm bảo Nhóm Phát triển đang thực hiện tốt sự kiện này.

- **Sơ kết Sprint (Sprint review):**

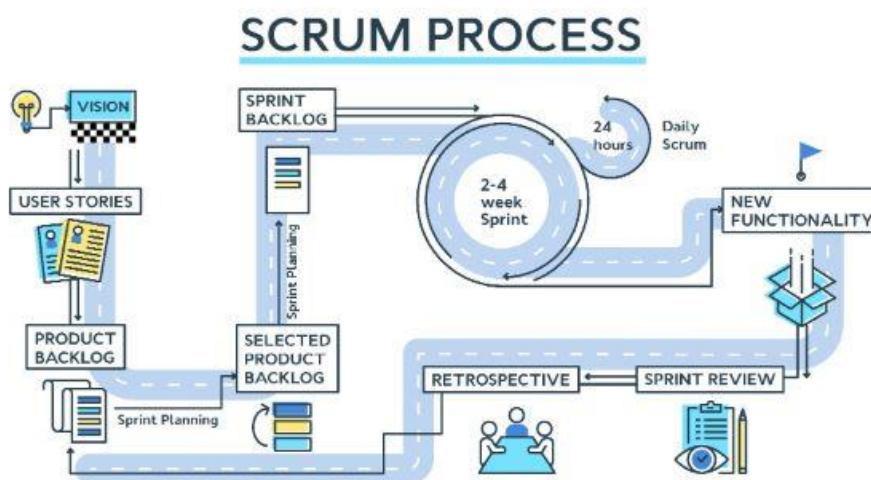
Là sự kiện diễn ra ở cuối Sprint nhằm thanh tra và thích nghi sản phẩm đang được xây dựng. Toàn bộ Nhóm **Scrum** (bao gồm Product Owner, Scrum Master và Nhóm Phát triển) tham dự sự kiện này. Product Owner có thể mời thêm những người khác cùng tham gia.

Sự kiện này bao gồm 2 hoạt động chính đó là dùng thử sản phẩm và thảo luận về tình hình của sản phẩm, hướng đi tiếp theo và những điều chỉnh đối với sản phẩm nếu cần thiết. Product Backlog và Kế hoạch Phát hành có thể được điều chỉnh sau sự kiện này.

- **Cải tiến Sprint (Sprint Retrospective):**

Diễn ra sau sự kiện Sơ kết Sprint nhằm thanh tra và thích nghi quy trình làm việc. Nói ngắn gọn, sự kiện này là để cải tiến cách làm việc.

Nhóm Phát triển và Scrum Master bắt buộc tham gia sự kiện này. Product Owner có thể tham gia hoặc không. Nhóm Phát triển có thể mời thêm những người khác tham dự. Kết quả của buổi làm việc này là một danh sách các thay đổi về cách làm việc được đưa vào áp dụng ngay trong Sprint tiếp theo.



2.4.6 Các tạo tác trong Scrum

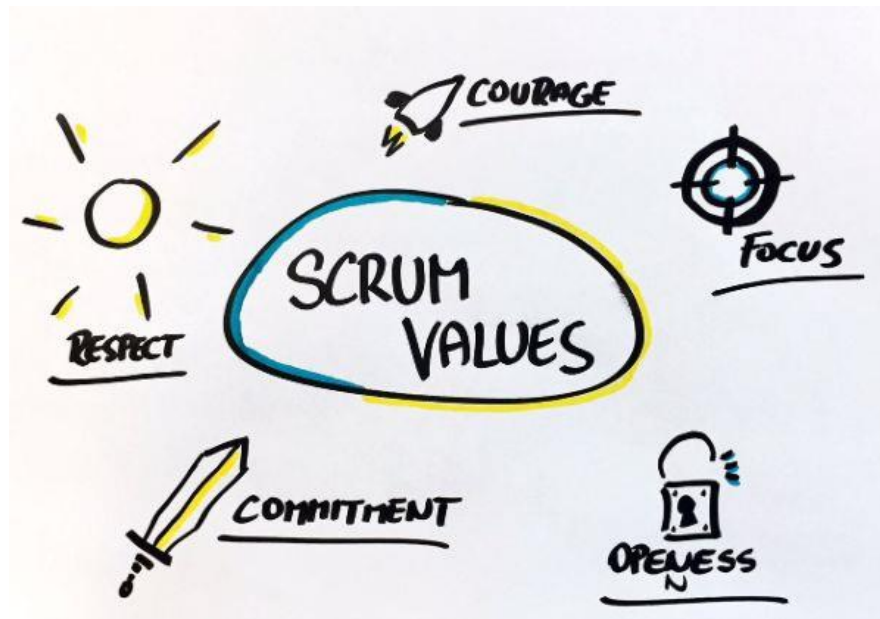
- **Product Backlog:** là nơi lưu trữ danh sách các tính năng mong muốn của sản phẩm và liên tục được cập nhật trong suốt vòng đời của sản phẩm. Mỗi Product Backlog sẽ được gắn với mục tiêu (Product Goal) mô tả đích đến của sản phẩm mà mỗi hạng mục trong Product Backlog phải hỗ trợ mục tiêu này. Product Owner là người chịu trách nhiệm quản lý và bảo trì Product Backlog. Việc này bao gồm xác định nội dung (các hạng mục cần phát triển), đánh giá độ ưu tiên và sắp xếp các hạng mục, làm mịn các hạng mục, làm rõ và giải thích tất cả mọi thắc mắc liên quan đến sản phẩm.
- **Sprint Backlog:** Sprint Backlog là bảng công việc được các Nhà Phát triển sử dụng để quản lý quá trình phát triển trong một Sprint và được cập nhật trong suốt Sprint. Sprint Backlog chứa danh sách các hạng mục và các công việc cần làm tương ứng với từng hạng mục để hoàn thành mục tiêu của Sprint đó
- **Incremental (Phần tăng trưởng):** là phần sản phẩm các Nhà Phát triển tạo ra cuối mỗi Sprint. Đây là một khái niệm quan trọng trong **Scrum** tạo ra sự khác biệt lớn về mặt sản phẩm so với các phương pháp truyền thống. **Scrum** không chỉ đơn giản tách quá trình phát triển thành các Sprint nhỏ liên tiếp nhau, mà cuối mỗi Sprint đòi hỏi nhóm phải chuyển giao một phần tính năng “hoàn chỉnh” của sản phẩm. Hoàn chỉnh ở đây được hiểu theo nghĩa được Product Owner chấp nhận dựa theo Định nghĩa Hoàn thành đã được thống nhất trước đó. Có thể nói, việc chuyển giao được một Phần tăng trưởng ở cuối mỗi Sprint là nhiệm vụ cốt lõi và không hề dễ dàng đối với các Nhà Phát triển.

Bên cạnh những khái niệm đề cập ở trên, các nhóm **Scrum** một khái niệm cũng thường xuyên xuất hiện trong **Scrum** đó là biểu đồ Burndown Chart. Đây là biểu đồ hiển thị khối lượng cần thiết còn lại để hoàn tất công việc. Burndown Chart có thể được dùng để theo dõi tiến độ của Sprint (được gọi là Sprint Burndown Chart) hoặc của cả dự án (Project Burndown Chart).

2.4.7 Các giá trị của Scrum

Để áp dụng **Scrum** và tận dụng hiệu quả những lợi ích mà **Scrum** mang lại phụ thuộc rất nhiều vào mức độ theo đuổi và sống với những giá trị sau đây:

- Tập trung: Mọi người tập trung vào công việc trong Sprint và Mục tiêu Sprint của Nhóm. Khi Nhóm Phát triển đã cam kết với những việc trong Sprint, họ cần phải tập trung để hoàn thành những gì mà mình đã cam kết.
- Dũng cảm: Để một người dám nói ra vấn đề của mình và chấp nhận rất nhiều loại rủi ro khi thay đổi, cam kết, họ cần là người dũng cảm. Về cơ bản, những giá trị khác không thể có nếu bạn không có sự dũng cảm.
- Cam kết: Ngoài ra, chúng ta liên tục cải tiến tức là thay đổi để trở thành một cá nhân tốt hơn, nhóm tốt hơn và tổ chức tốt hơn. Chúng ta luôn phải thay đổi để giữ vững lợi thế cạnh tranh và phục vụ khách hàng. Thực hiện thay đổi bao giờ cũng rất khó khăn, do đó chỉ với sự cam kết của chúng ta mới có thể làm được.
- Cởi mở: Mọi thứ cần phải rõ ràng, minh bạch để mọi người có thể làm việc hiệu quả. Công việc phát triển sản phẩm ngày nay rất phức tạp, một người không thể nhìn và hiểu được hết tất cả mọi vấn đề. Do đó, nếu mọi người không cởi mở với nhau, thông tin bị che giấu rất nhiều và hiệu quả công việc khó có thể nâng cao.
- Tôn trọng: Khi thiếu tôn trọng, mọi người khó thành thật trong chia sẻ. Ví dụ, khi một người không biết một điều gì đó và đi hỏi lại người khác. Người trả lời thay vì mong muốn giúp đỡ để người hỏi trở nên tốt hơn, độc lập hơn lại phàn nàn, đánh giá người hỏi thì lần sau người hỏi sẽ khó mà cởi mở và nói sự thật được. Không có tôn trọng, khó có sự cởi mở. Những công ty có văn hóa đổ lỗi khó có sự cởi mở.



2.5 Tổng quan về công nghệ, kiến trúc

2.5.1 Tổng quan về Git

2.5.1.1 Git là gì?

Git là một hệ thống quản lý phiên bản phân tán (Distributed Version Control System – DVCS), nó là một trong những hệ thống quản lý phiên bản phân tán phổ biến nhất hiện nay. Git cung cấp cho mỗi lập trình viên kho lưu trữ (repository) riêng chứa toàn bộ lịch sử thay đổi.

2.5.1.2 Lợi ích của Git

Các dự án thực tế thường có nhiều lập trình viên làm việc song song. Vì vậy, một hệ thống kiểm soát phiên bản như Git là cần thiết để đảm bảo không có xung đột code giữa các lập trình viên.

Ngoài ra, các yêu cầu trong các dự án như vậy thay đổi thường xuyên. Vì vậy, một hệ thống kiểm soát phiên bản cho phép các nhà phát triển revert và quay lại phiên bản cũ hơn của code.

Cuối cùng, đôi khi một số dự án đang được chạy song song liên quan đến cùng một cơ sở code. Trong trường hợp như vậy, khái niệm phân nhánh trong Git là rất quan trọng.

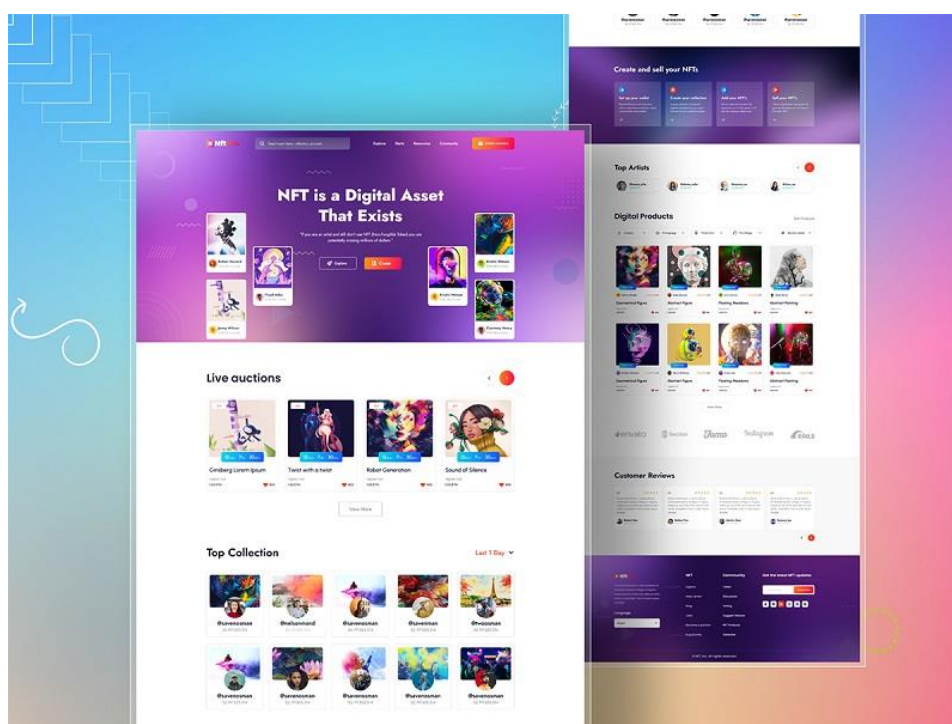
- Dễ sử dụng, thao tác nhanh, gọn, lẹ và rất an toàn.

- Dễ dàng kết hợp các phân nhánh (branch), có thể giúp quy trình làm việc code theo nhóm đơn giản hơn rất nhiều.
- Chỉ cần clone mã nguồn từ kho chứa hoặc clone một phiên bản thay đổi nào đó từ kho chứa, hoặc một nhánh nào đó từ kho chứa là bạn có thể làm việc ở mọi lúc mọi nơi.
- Deployment sản phẩm của bạn một cách không thể nào dễ dàng hơn.

2.5.2 Tổng quan về Figma

2.5.2.1 Figma là gì ?

Figma là một phần mềm thiết kế đồ họa và giao diện người dùng trực tuyến mạnh mẽ. Nó được sử dụng để tạo ra các bản thiết kế UI/UX cho ứng dụng web và di động một cách sáng tạo và không giới hạn. Bạn cũng có thể sử dụng Figma để thiết kế bài đăng trên các mạng xã hội và các dự án thiết kế khác.



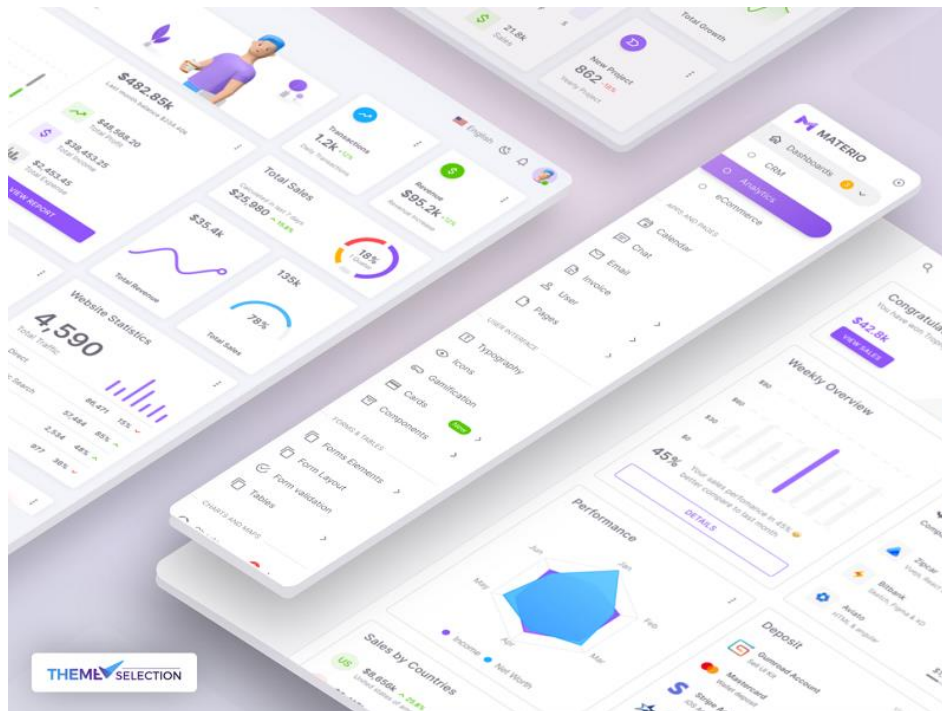
Một điểm đặc biệt của Figma là các công cụ được lưu trữ thông qua hệ thống lưu trữ đám mây. Các designer có thể thiết kế giao diện website và ứng dụng trực tuyến mà không cần phải tải về máy. Cách thức hoạt động tương tự như trên google driver

2.5.2.2 Ưu điểm nổi bật của Figma

Là một công cụ thiết kế đồ họa và giao diện người dùng trực tuyến mạnh mẽ và được sử dụng rộng rãi trong cộng đồng thiết kế. Nó mang lại nhiều ưu điểm nổi bật so với các công cụ thiết kế truyền thống, hãy cùng điểm qua những ưu điểm đó.

- **Thiết kế đa nền tảng và đa thiết bị**

Figma là một phần mềm thiết kế đa nền tảng. Bạn có thể sử dụng trên các nền tảng khác nhau như Windows, MacOS hay Linux. Điều này giúp cho các thiết kế viên có thể làm việc và chia sẻ dự án một cách dễ dàng và linh hoạt hơn.



- **Công cụ tạo và chỉnh sửa vector tuyệt vời**

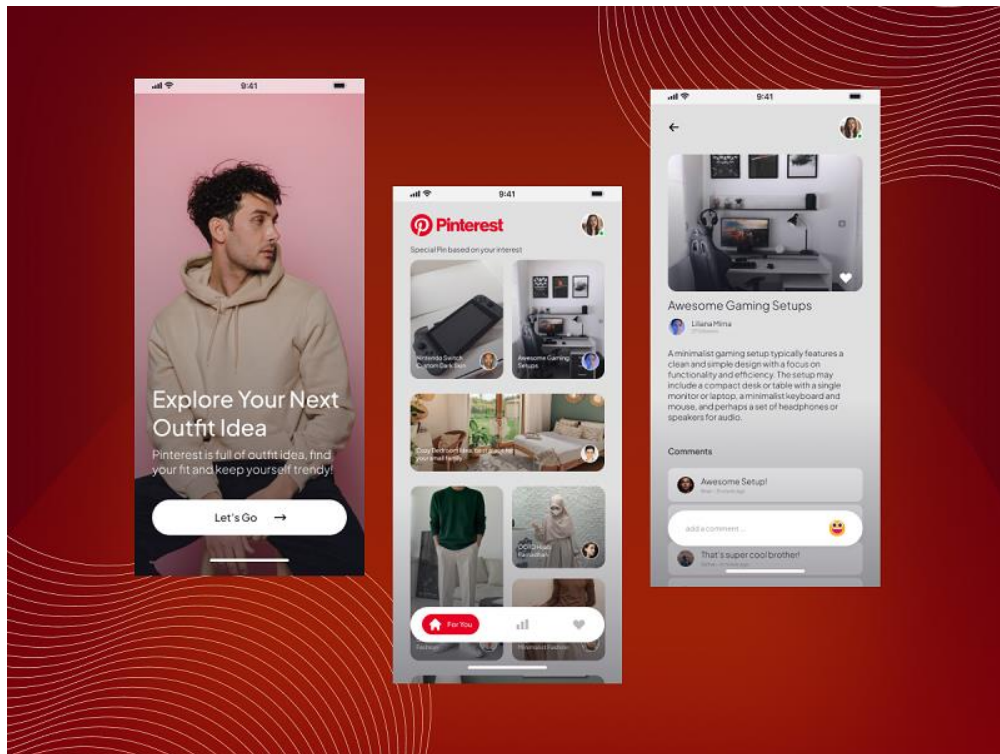
Cung cấp rất nhiều các công cụ thiết kế hữu ích để giúp cho việc thiết kế trở nên dễ dàng và chuyên nghiệp hơn. Bạn có thể tạo ra các biểu tượng, các hình ảnh vector, các hiệu ứng chữ và rất nhiều thứ khác. Ngoài ra, Figma cũng hỗ trợ tính năng thử nghiệm giao diện người dùng trực tiếp trên ứng dụng hay website mà bạn đang thiết kế.

- **Cộng tác và chia sẻ thiết kế dễ dàng**

Figma cho phép nhiều người cùng làm việc trên cùng một dự án. Bạn có thể chia sẻ dự án với đội ngũ thiết kế và các thành viên khác trong công ty để họ có thể theo dõi và đóng góp ý kiến. Tính năng này giúp cho quá trình làm việc của đội ngũ thiết kế trở nên hiệu quả và nhanh chóng hơn.

- **Thanh công cụ tiện lợi**

Thanh công cụ tiện lợi giúp người dùng có thể truy cập nhanh chóng đến các công cụ và tính năng quan trọng. Thanh công cụ này cung cấp các công cụ thiết kế đồ họa, các công cụ làm việc với định dạng văn bản, hiệu ứng, phác thảo, và nhiều hơn nữa.



- **Dễ dàng sử dụng cho người mới bắt đầu**

Figma là một công cụ thiết kế dựa trên đám mây. Các dữ liệu của bạn được lưu trữ và quản lý trên đám mây, giúp cho việc lưu trữ và chia sẻ dữ liệu trở nên tiện lợi và an toàn hơn. Đồng thời, tính năng này cũng giúp cho bạn có thể làm việc từ bất cứ đâu mà không cần phải cài đặt phần mềm trên máy tính

Giao diện dễ sử dụng và thân thiện với người dùng, đặc biệt là với những người mới bắt đầu với thiết kế đồ họa. Các tính năng và công cụ được sắp xếp một cách rõ ràng và dễ dàng để truy cập, giúp cho người dùng có thể tập trung vào việc tạo ra các thiết kế chất lượng cao.

2.5.3 Tổng quan về Jira

2.5.3.1 Jira là gì?

Jira là một công cụ đám mây để theo dõi các công việc và quản lý dự án. Công cụ này được thiết kế để tăng cường phối hợp nhóm trong phát triển phần mềm Agile.

Ngoài ra, Jira cũng cung cấp một bộ theo dõi bug và tiến độ toàn diện trong toàn bộ vòng đời phát triển phần mềm.

2.5.3.2 Các tính năng cơ bản của Jira

- Quản lý và theo dõi tiến độ của dự án.
- Quản lý các tasks, bugs, sự cải tiến, những tính năng mới hoặc bất kỳ vấn đề xảy ra.
- Tạo ra và lưu trữ lại những bộ lọc có cấu hình cao xuyên suốt mọi vấn đề trong hệ thống.
- Chia sẻ bộ lọc với người sử dụng khác hoặc đăng ký và nhận kết quả qua hệ thống email định kỳ.
- Xây dựng quy trình làm việc tương thích với từng yêu cầu của dự án và quy trình của doanh nghiệp
- Bảng dashboard cung cấp cho người sử dụng một không gian riêng. Nhóm xem mọi thông tin liên quan đến cá nhân.
- Cung cấp nhiều loại báo cáo thống kê với những biểu đồ khác nhau, phù hợp với nhiều loại hình dự án và đối tượng người sử dụng.
- Dễ dàng tích hợp với các hệ thống ứng dụng khác (như Email, Excel, RSS,...).
- Có thể chạy trên hầu hết các nền tảng, hệ điều hành và cơ sở dữ liệu.

2.5.3.3 Ưu điểm của Jira

- Jira giúp bạn lập biểu đồ ý tưởng của mình và truyền đạt chúng cho nhóm của bạn thông qua Roadmap. Tính năng này cho phép bạn chia sẻ các kế hoạch lớn và phân công nhiệm vụ cá nhân hiệu quả.
- Ứng dụng này có tất cả các tính năng bạn cần để giúp các team Agile/Scrum đi đúng hướng. Đối với các user story riêng lẻ, bạn có thể tạo các báo cáo như biểu đồ tổng hợp và đối với quản lý nhóm, bạn cũng có thể theo dõi khối lượng công việc cực kì chi tiết.

- Khả năng tích hợp mạnh mẽ . Ứng dụng cho phép tích hợp dễ dàng với các phần mềm bên thứ 3 khác. Ví dụ: với Hipchat và Sack để nhận thông báo và giao tiếp. Hơn 3000 add-ons/plugins có sẵn, giúp customize Jira với các nhu cầu và nhiệm vụ khác nhau.
- Jira cho phép người dùng tùy chỉnh trải nghiệm của họ bằng cách tạo và tùy chỉnh các phần tử khác nhau, chẳng hạn như bảng, biểu mẫu, tiến trình, báo cáo, trường và hơn thế nữa.
- Phần mềm được sử dụng cho một số nhiệm vụ khác nhau của các loại người dùng khác nhau. Cho dù bạn là nhà phát triển, người quản lý, người quản lý dự án hay kỹ sư, bạn sẽ có thể sử dụng Jira để đáp ứng nhu cầu kinh doanh của mình.

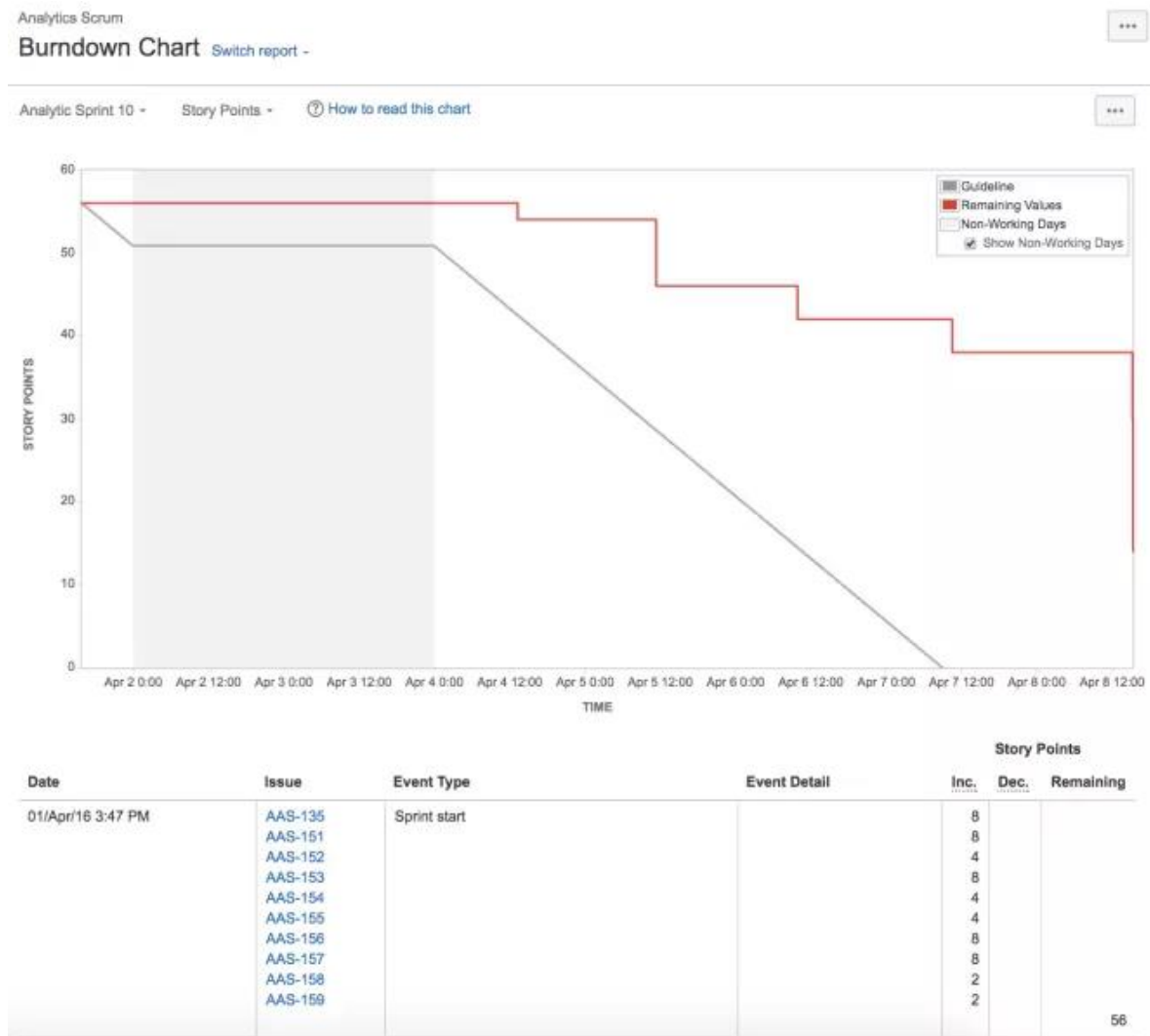
2.5.3.4 Nhược điểm của Jira

- Ứng dụng Jira trên điện thoại không phải là cách tốt nhất để làm việc với Jira khi bạn đang di chuyển. Nhiều người dùng nói rằng mặc dù mobile app của Jira đem đến một trải nghiệm tốt, nhưng họ thích làm việc bằng ứng dụng trên web hơn.
- Tích hợp với các hệ thống khác là khá phức tạp. Di chuyển một dự án từ Microsoft Team Foundation Server sang Jira là một quá trình rất tốn thời gian. Nếu bạn gặp vấn đề như thế này và cần tư vấn và hỗ trợ thì hãy liên hệ đội ngũ chuyên gia Atlassian của Biplus để được hỗ trợ tốt nhất.
- Không thể tải xuống báo cáo. Không thể tải xuống báo cáo Jira dưới dạng hình ảnh. Sử dụng các phương pháp khác làm thay đổi độ phân giải của hình ảnh, dẫn đến chất lượng hình ảnh kém. Jira đã tải lên kích thước tệp giới hạn để giữ cho công cụ này load nhanh hơn. Do đó, không được phép tải lên ảnh, video hoặc tài liệu có kích thước lớn hơn 10MB.
- Cấu hình của một số phần trong Jira đôi lúc hơi khó hiểu.

2.5.3.5 Các thuật ngữ cần biết khi sử dụng Jira

Jira là phần mềm với khá nhiều thuật ngữ công nghệ. Đôi khi chúng có thể là phần khó nhất của lộ trình học tập khi bắt đầu với Jira (và phương pháp Agile). Dưới đây, chúng tôi đã tổng hợp danh sách một số thuật ngữ liên quan đến Jira.

- **Backlog** – đây chính là danh sách các công việc cần làm. Thông thường, backlog trong Jira sẽ chứa danh sách các story, issue và user story cho một sản phẩm hoặc sprint.



- **Board** – bảng biểu: Công cụ mà các nhóm sử dụng để hình dung các đơn vị công việc đang được thực hiện trong quy trình làm việc cụ thể. Nó có thể được điều chỉnh cho các phong cách phát triển phần mềm theo Agile khác nhau (ví dụ: bảng Scrum hiển thị các mục công việc chuyển từ product backlog sang sprint backlog trong khi bảng Kanban thường có quy trình làm việc ba bước: To do, In Progress và Done).

- **Burndown chart** – Biểu đồ Burndown: cho thấy khối lượng công việc thực tế và ước tính phải thực hiện trong một sprint. Đồng thời cho phép theo dõi tốc độ "burn" việc của nhóm.
- **Control chart** – Biểu đồ kiểm soát: Biểu đồ kiểm soát có thể hiển thị thời gian chu kỳ hoặc thời gian dẫn đầu cho sản phẩm, phiên bản hoặc sprint của bạn.
- **Cycle time**: Thời gian chu kỳ: Thời gian chu kỳ là thời gian dành cho việc giải quyết một issue (vấn đề) – thường là thời gian tính từ khi bắt đầu giải quyết một vấn đề đến khi hoàn thành công việc, nhưng cũng bao gồm bất kỳ thời gian nào khác dành cho việc giải quyết vấn đề. Ví dụ: nếu một vấn đề được mở lại, khắc phục và hoàn thành một lần nữa, thì thời gian cho công việc bổ sung này được thêm vào thời gian chu kỳ.
- **Daily stand-up** – họp hàng ngày – Daily là một cuộc họp nhỏ kéo dài 15 phút mỗi ngày để các nhóm đồng bộ các công việc đang được thực hiện.
- **Epic**: Epic ghi lại một khối lượng lớn công việc cần được chia nhỏ thành một số story nhỏ hơn. Có thể mất vài sprint để hoàn thành một epic. Hệ thống phân cấp cho các đơn vị công việc trong phần mềm Jira như sau: Project > Epics/Components > Stories > Tasks > Subtasks.
- **Filter** – Bộ lọc: Bộ lọc xác định những gì hiển thị trên mỗi bảng của bạn. Sử dụng trình tạo truy vấn đơn giản (JQL – Jira Query Language) của Jira, bạn có thể tùy chỉnh chính xác vấn đề (issue) nào được hiển thị trên bảng của bạn.
- **Issue**: Một issue chỉ đơn giản là một đơn vị công việc trong Jira sẽ được theo dõi thông qua một quy trình làm việc, từ khi được tạo đến khi hoàn thành. Nó có thể bao gồm đại diện cho một đơn vị công việc, như một nhiệm vụ đơn giản hoặc một bug, đến một hạng mục công việc lớn hơn (parent work) cần được theo dõi, như một story hoặc một epic.
- **Kanban**: Kanban là một hệ thống để hình dung luồng công việc và giới hạn công việc đang thực hiện. Kanban không được định hướng theo hướng sprint, giống như phương pháp luận phát triển Scrum, vì nó theo hướng tiếp diễn liên tục.

- **Scrum:** Scrum là một phương pháp luận phát triển Agile trong đó sản phẩm được xây dựng trong một chuỗi các lần lặp có độ dài cố định được gọi là sprint. Nó cung cấp cho các nhóm một khuôn khổ để vận chuyển phần mềm theo nhịp đều đặn.
- **Scrum of Scrums:** Scrum of Scrums là một phương tiện mở rộng Scrum tới các dự án lớn, nhiều nhóm. Một kỹ thuật để mở rộng quy mô Scrum, các dự án đa đội – theo truyền thống gọi là program management.
- **Sprint:** là một khoảng thời gian ngắn (lý tưởng là từ hai đến bốn tuần) trong đó nhóm phát triển triển khai và cung cấp một phần của sản phẩm hoặc một chức năng có thể sử dụng được.
- **Sprint planning:** Một cuộc họp lập kế hoạch nhóm xác định những gì cần hoàn thành trong sprint sắp tới.
- **Sprint retrospective:** Cuộc họp nhằm xem xét lại những gì đã làm, đánh giá tốt hoặc không tốt cùng với các kế hoạch hành động để làm cho sprint tiếp theo tốt hơn.
- **Story:** Story hoặc User story là một yêu cầu hệ thống phần mềm được thể hiện bằng một vài câu ngắn, lý tưởng là sử dụng ngôn ngữ phi kỹ thuật.
- **Story point:** Story point là một ước tính về mức độ phức tạp tương đối của một story.
- **Swimlane:** Phân loại các công việc để xem xét công việc nào nên tiến hành trước.
- **Subtask:** Một nhiệm vụ phụ có thể là “tập con” của bất kỳ loại vấn đề nào, tùy thuộc vào Issue Type Scheme (Sơ đồ loại issue) của dự án.
- **Task:** Một task (tác vụ) là một loại issue có sẵn trong Jira.
- **Velocity:** Velocity là thước đo mức độ công việc mà nhóm có thể xử lý trong một khoảng thời gian cụ thể, tức là nhóm có thể hoàn thành bao nhiêu product backlog của sản phẩm trong một lần sprint. Velocity có thể được tính toán dựa trên story point, giá trị kinh doanh, giờ, số lượng issue hoặc bất kỳ trường số (numeric field) nào bạn chọn.

- **Workflow** – quy trình làm việc là logic thúc đẩy chuyển động của một story / subtask dọc theo hành trình phát triển của chúng trên bảng scrum hoặc kanban.

2.5.4 Tổng quan Scrum Master

2.5.4.1 Scrum Master là gì?

Scrum Master là một chuyên gia lãnh đạo nhóm sử dụng phương pháp quản lý dự án Agile xuyên suốt quá trình của một dự án. Scrum Master tạo điều kiện thuận lợi cho mọi giao tiếp và hợp tác giữa ban lãnh đạo và thành viên trong nhóm để đảm bảo đạt được kết quả và chỉ tiêu đề ra.

Phương pháp này có tính cộng tác cao và yêu cầu các quy trình hiệu quả. Kết quả của quy trình phụ thuộc vào chuyên môn của Scrum Master. Các phương pháp thuộc Agile chỉ phổ biến ở các công ty công nghệ. Nhưng Scrum Master có thể được tìm thấy ở mọi ngành nghề, lĩnh vực và công ty trên toàn thế giới.



2.5.4.2 Vai trò và công việc của một Scrum Master

Scrum Master đóng vai trò khác nhau trong từng chức năng của mình. Đối với đối nhóm Scrum mà mình quản lý, một Scrum Master có những đầu việc sau:

- Huấn luyện các thành viên trong nhóm về tự quản lý và chức năng của Scrum

- Giúp nhóm Scrum tập trung vào việc tạo ra các gia tăng có giá trị cao đáp ứng yêu cầu của sản phẩm
- Loại bỏ các trở ngại đối với tiến trình của nhóm Scrum
- Đảm bảo rằng tất cả các sự kiện Scrum diễn ra và tích cực, hiệu quả và được lưu giữ trong timebox.

Ngoài ra, Scrum Master còn cộng tác với Product Owner theo một số cách sau:

- Giúp tìm ra các kỹ thuật để xác định Mục tiêu Sản phẩm hiệu quả và quản lý Product Backlog
- Giúp nhóm Scrum hiểu được sự cần thiết của các hạng mục Product Backlog rõ ràng và ngắn gọn
- Giúp thiết lập kế hoạch sản phẩm thực nghiệm cho một môi trường phức tạp
- Tạo điều kiện cho sự hợp tác của các bên liên quan khi được yêu cầu hoặc cần thiết.

Cuối cùng, vai trò của Scrum Master trong một tổ chức có thể bao gồm:

- Lãnh đạo, đào tạo và huấn luyện tổ chức trong việc áp dụng Scrum
- Lập kế hoạch và tư vấn triển khai Scrum trong tổ chức
- Giúp nhân viên và các bên liên quan hiểu và đưa ra phương pháp tiếp cận theo kinh nghiệm cho các công việc phức tạp
- Loại bỏ các rào cản giữa các bên liên quan và nhóm Scrum.



CHƯƠNG 3: XÁC ĐỊNH NHU CẦU

3.1 Product Backlog

ID	Nội dung
PB1	Đăng nhập: Có thể đăng nhập tài khoản để truy cập vào hệ thống.
PB2	Trang chủ: Trang chủ thể hiện trực quan các trang khác có thể truy cập vào.
PB3	Quản lý nhà cung cấp: Người quản trị có thể thêm, sửa, xóa các nhà cung cấp trên ứng dụng.
PB4	Quản lý khách hàng: Người quản trị có thể thêm, sửa, xóa các khách hàng trên ứng dụng.
PB5	Quản lý sản phẩm: Người quản trị có thể thêm, sửa, xóa các loại sản phẩm.

PB6	Quản lý nhân viên: Người quản trị có thể thêm, sửa, xóa các thông tin của nhân viên bao gồm (tài khoản, mật khẩu).
PB7	Quản lý hóa đơn: Người quản trị có thể thêm, sửa, xóa các hóa đơn bán hàng.
PB8	Thống kê: Thống kê số lượng bán theo ngày, tháng, năm
PB9	Tìm kiếm: Nhân viên có thể tìm kiếm thông tin hóa đơn.

Hình 3. 1 *Product Backlog*

3.2 Sprint Backlog

ID	Nội dung
SPB1	Sprint 1 – Xây dựng CSDL và các trang: Xây dựng giao diện cơ bản cho trang Đăng nhập và các trang Trang chủ, Nhà cung cấp, Nhân Viên.
SPB2	Sprint 2 – Xây dựng các trang còn lại và kiểm tra cuối cùng, phát hành: Hoàn thành giao diện cho trang Sản phẩm, Khách hàng, Hóa đơn bán hàng, kết nối với cơ sở dữ liệu để hiển thị dữ liệu và thao tác các chức năng của các trang. Kiểm tra và fix bugs cuối cùng trên toàn bộ ứng dụng. Triển khai ứng dụng lên môi trường sản phẩm.

Hình 3. 2 *Sprint Backlog*

3.3 User stories

- **User Story 1:** Là một khách hàng, tôi muốn có thể tìm kiếm sản phẩm theo danh mục và tên sản phẩm để dễ dàng tìm thấy sản phẩm mình cần.
- **User Story 2:** Là một chủ cửa hàng, có thể theo dõi số lượng hàng tồn kho để quản lý hàng hóa một cách hiệu quả.
- **User Story 3:** Là một chủ cửa hàng, tôi muốn có báo cáo chi tiết về doanh thu và đơn hàng để có thể đưa ra quyết định kinh doanh đúng đắn.
- **User Story 4:** Là một nhân viên chăm sóc khách hàng, tôi muốn có thể truy cập thông tin đơn hàng nhanh chóng để hỗ trợ khách hàng kịp thời.

3.4 Technical Backlog

ID	Nội dung
TB1	Front-end: <ul style="list-style-type: none">• Thiết kế và xây dựng giao diện cho các trang (Đăng nhập, Trang chủ, Khách hàng, Nhà cung cấp, Sản phẩm, nhân viên, Hóa đơn bán hàng và Tìm kiếm hóa đơn bán hàng).• Xây dựng các thành phần giao diện tái sử dụng như button, form, và menu.• Tối ưu hóa trải nghiệm người dùng trên desktop.• Kiểm thử giao diện để đảm bảo tính nhất quán và hiệu quả.
TB2	Back-end: <ul style="list-style-type: none">• Thiết kế và xây dựng các event cho mỗi chức năng của trang (quản lý khách hàng, quản lý sản phẩm, quản lý nhân viên, quản lý nhà cung cấp, tìm kiếm) và thêm các ràng buộc (thông tin khách hàng, giá bán sản phẩm và tài khoản, mật khẩu của nhân viên).• Tạo và quản lý cơ sở dữ liệu (SQL Server) để lưu trữ thông tin về sản phẩm, khách hàng, nhà cung cấp, phiếu nhập, nhân viên, tài khoản đăng nhập)• Đảm bảo bảo mật dữ liệu và xử lý lỗi một cách an toàn.
TB3	Testing: <ul style="list-style-type: none">• Thực hiện kiểm thử chức năng để đảm bảo mọi tính năng hoạt động đúng như yêu cầu đặt ra.• Thực hiện kiểm thử tương tác (interaction testing) để đảm bảo giao diện và back-end hoạt động một cách nhất quán.• Kiểm tra các ràng buộc về ngày tháng, mệnh giá tiền để quản lý hóa đơn, sản phẩm một cách ổn định.
TB4	Triển khai và Monitor: <ul style="list-style-type: none">• Triển khai ứng dụng lên môi trường sản phẩm (production environment) để người dùng có thể truy cập.• Thiết lập các công cụ theo dõi (monitoring tools) để theo dõi hiệu

	suất ứng dụng và phản ứng kịp thời với các vấn đề.
--	--

3.5 Support Backlog

ID	Nội dung
SB1	Cập nhật tài liệu hướng dẫn sử dụng cho người dùng và quản trị viên
SB2	Xây dựng chức năng Thống kê và Báo cáo để quản trị viên có thể theo dõi hoạt động của ứng dụng và nhân viên.
SB3	Cải thiện khả năng tương thích của ứng dụng trên các nền tảng Windows 7, Windows 8.1, Windows 10, và Windows 11. Linux hoặc MacOS (Wine).

CHƯƠNG 4: LẬP KẾ HOẠCH SCRUM

ID	Issue	Personas	Est. Story Point	Sprint	Start	End
1	Thiết kế và triển khai cơ sở dữ liệu SQL	Lê Trực Tín	8	Sprint 1	8:00 AM 28/05/2024	6:00 PM 28/05/2024
2	Thiết kế giao diện form Đăng nhập	Trần Khánh Duy	3	Sprint 1	8:00 AM 28/05/2024	2:00 PM 28/05/2024
3	Xử lý logic xác thực người dùng	Trần Khánh Duy	5	Sprint 1	2:00 PM 29/05/2024	8:00 PM 29/05/2024
4	Kết nối cơ sở dữ liệu để kiểm tra thông tin đăng nhập	Trần Khánh Duy	3	Sprint 1	8:00 AM 30/05/2024	2:00 PM 30/05/2024
5	Thiết kế giao diện form Trang chủ và xây dựng các nút điều hướng	Lê Trực Tín	5	Sprint 1	2:00 PM 30/05/2024	8:00 PM 30/05/2024
6	Thiết kế giao diện form Quản lý nhà cung cấp	Lê Thị Nguyễn Xuyên	3	Sprint 1	8:00 AM 31/05/2024	2:00 PM 31/05/2024
7	Chức năng thêm, sửa, xóa và tìm kiếm nhà cung cấp	Lê Thị Nguyễn Xuyên	5	Sprint 1	2:00 PM 31/05/2024	8:00 PM 31/05/2024
8	Kết nối cơ sở dữ liệu để quản lý nhà cung cấp	Lê Thị Nguyễn Xuyên	3	Sprint 1	8:00 AM 01/06/2024	2:00 PM 01/06/2024
9	Thiết kế giao diện form Quản lý nhân viên	Lê Trực Tín	5	Sprint 1	2:00 PM 01/06/2024	8:00 PM 01/06/2024

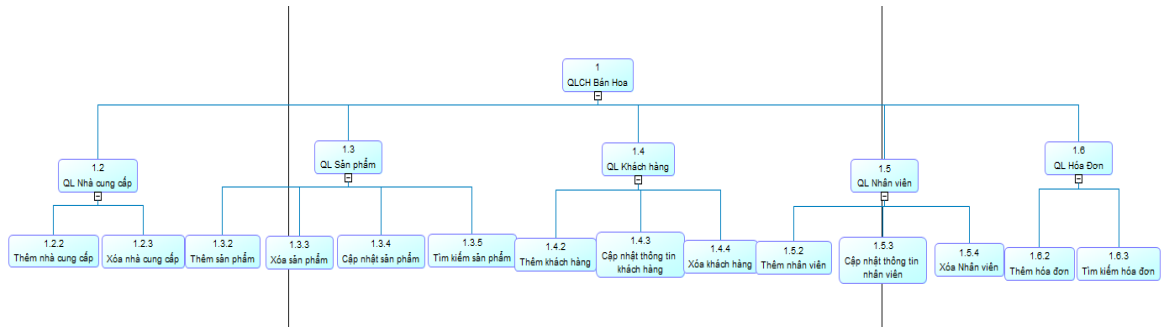
10	Chức năng thêm, sửa, xóa, tìm kiếm NV.	Lê Trực Tín	3	Sprint 1	8:00 AM 02/06/2024	2:00 PM 02/06/2024
11	Kết nối cơ sở dữ liệu để quản lý nhân viên	Lê Trực Tín	5	Sprint 1	8:00 AM 03/06/2024	2:00 PM 03/06/2024
12	Kiểm thử tổng thể các form đã hoàn thành trong Sprint 1	Lê Trực Tín	8	Sprint 1	2:00 PM 03/06/2024	8:00 PM 03/06/2024
13	Thiết kế giao diện form Quản lý sản phẩm	Lê Thị Nguyễn Xuyên	3	Sprint 2	8:00 AM 04/06/2024	2:00 PM 04/06/2024
14	Chức năng thêm, sửa, xóa và tìm kiếm sản phẩm	Lê Thị Nguyễn Xuyên	5	Sprint 2	2:00 PM 04/06/2024	8:00 PM 04/06/2024
15	Kết nối cơ sở dữ liệu để quản lý sản phẩm	Lê Thị Nguyễn Xuyên	3	Sprint 2	8:00 AM 05/06/2024	2:00 PM 05/06/2024
16	Thiết kế giao diện form Quản lý khách hàng	Trần Khánh Duy	5	Sprint 2	2:00 PM 05/06/2024	8:00 PM 05/06/2024
17	Chức năng thêm, sửa, xóa và tìm kiếm khách hàng	Trần Khánh Duy	3	Sprint 2	8:00 AM 06/06/2024	2:00 PM 06/06/2024
18	Kết nối cơ sở dữ liệu để quản lý khách hàng	Trần Khánh Duy	5	Sprint 2	2:00 PM 06/06/2024	8:00 PM 06/06/2024
19	Thiết kế giao diện form Quản lý hóa đơn bán hàng	Lê Trực Tín	3	Sprint 2	8:00 AM 07/06/2024	2:00 PM 07/06/2024
20	Chức năng tạo, sửa, xóa và tìm kiếm hóa đơn	Lê Trực Tín	5	Sprint 2	2:00 PM 07/06/2024	8:00 PM 07/06/2024

21	Kết nối cơ sở dữ liệu để quản lý thông tin hóa đơn	Lê Trực Tín	3	Sprint 2	8:00 AM 08/06/2024	2:00 PM 08/06/2024
22	Thiết kế giao diện form Tìm kiếm hóa đơn bán hàng	Lê Thị Nguyễn Xuyên	5	Sprint 2	2:00 PM 08/06/2024	8:00 PM 08/06/2024
23	Kiểm thử tổng thể các form đã hoàn thành trong Sprint 2	Lê Trực Tín	3	Sprint 2	8:00 AM 09/06/2024	2:00 PM 09/06/2024
24	TESTING cả dự án từ Sprint 1 và 2	Lê Trực Tín	5	Sprint 2	2:00 PM 09/06/2024	8:00 AM 10/06/2024
25	Phát hành sản phẩm	Lê Trực Tín	8	Sprint 2	8:00 AM 10/06/2024	8:00 PM 10/06/2024

CHƯƠNG 5: HIỆN THỰC HÓA KẾ HOẠCH

5.1 Thiết kế xử lý

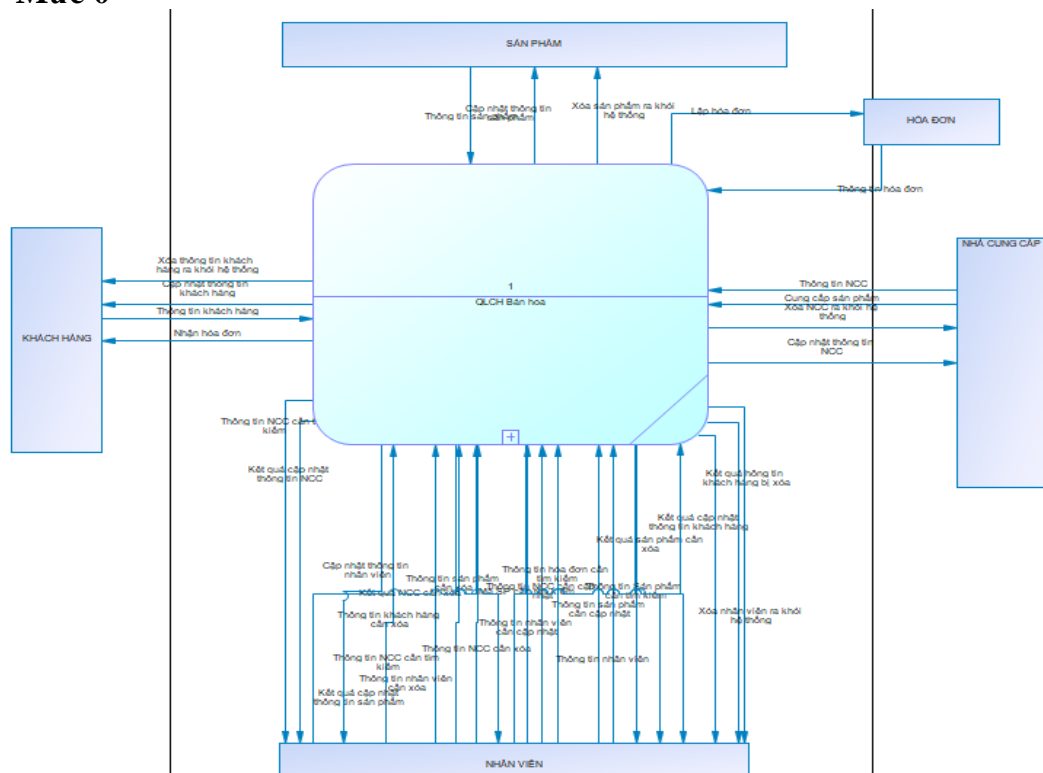
5.1.1 Sơ đồ phân rã chức năng



Hình 1. Sơ đồ phân rã chức năng

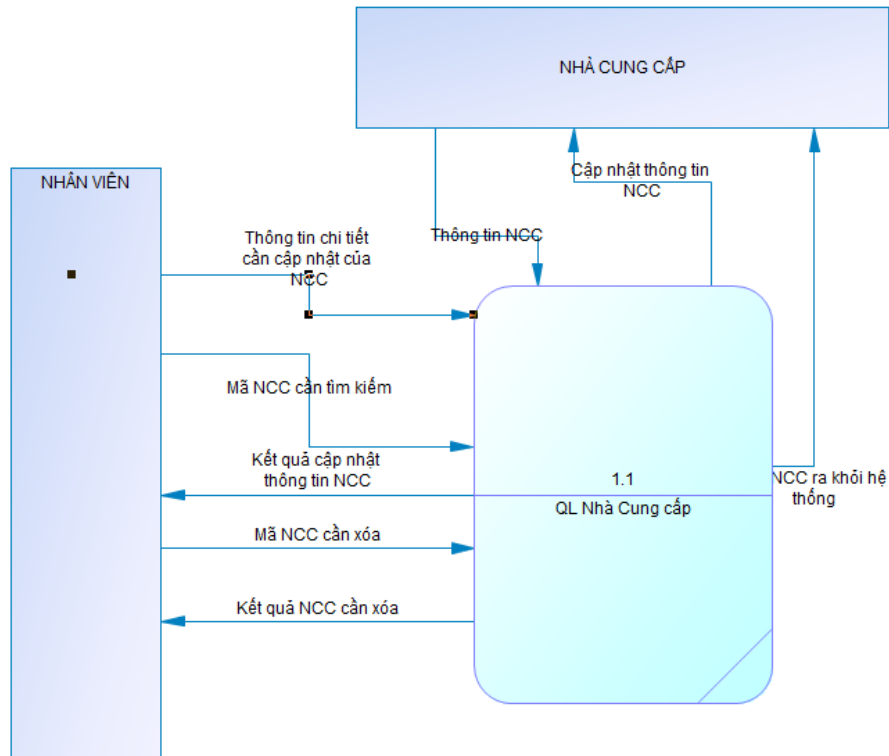
5.1.2 Sơ đồ luồng dữ liệu

Mức 0



Hình 2. Sơ đồ Mức 0

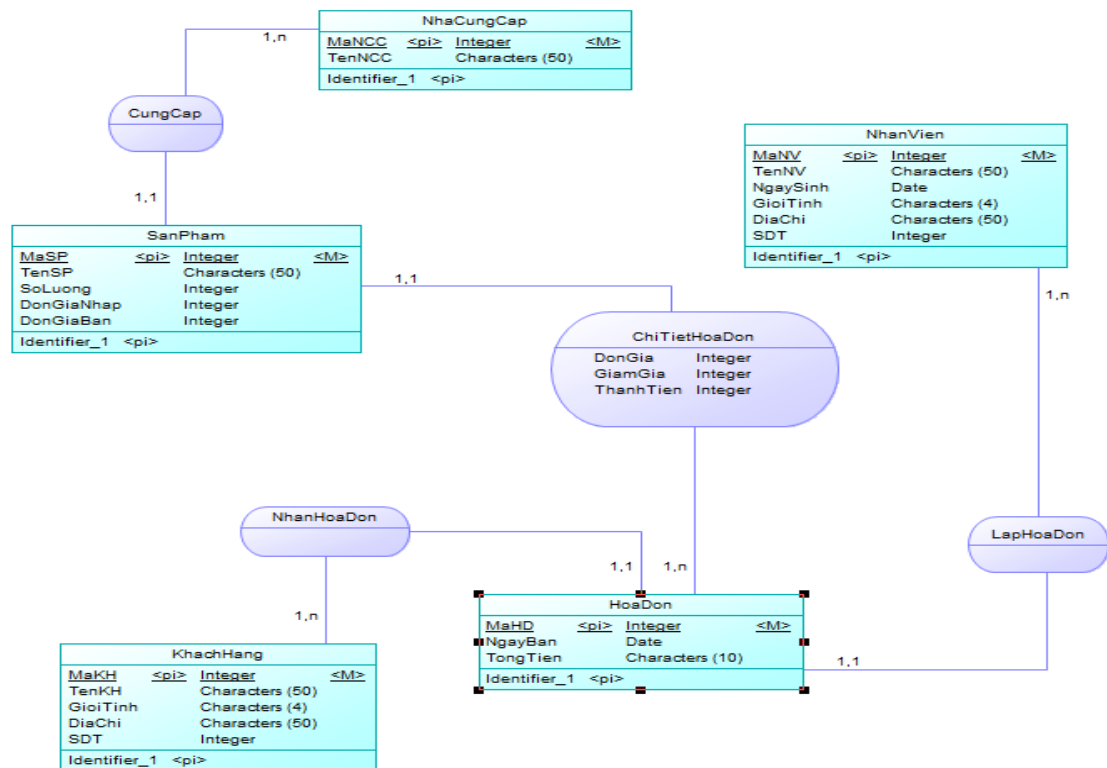
Mức 1



Hình 3. Sơ đồ Mức 1

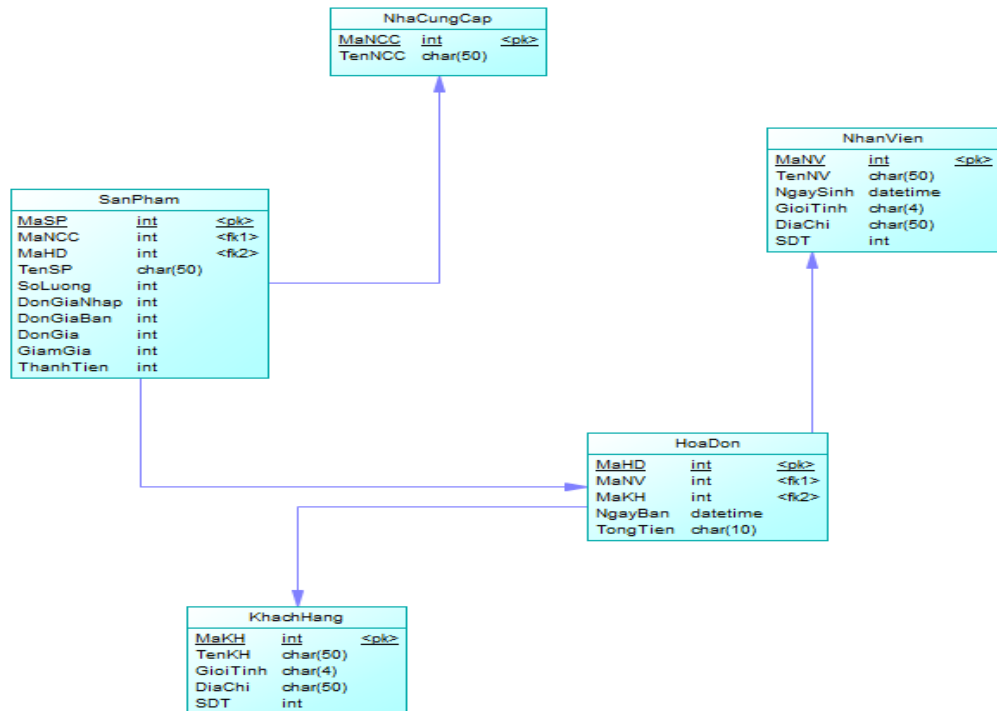
5.2 Thiết kế database

5.2.1 Mô hình thực thể kết hợp (ERD)



Hình 4. Sơ đồ ERD

5.2.2 Mô hình vật lý



Hình 5. Sơ đồ vật lý

5.3 Cơ sở dữ liệu

Từ mô hình ERD ta phân rã được các quan hệ sau (mô hình dữ liệu mức logic):

- NhaCungCap(MaNCC, TenNCC)
- SanPham(MaSP, MaNCC, MaHD, TenSP, SoLuong, DonGiaNhap, DonGiaBan, GiaBan, GiamGia, ThanhTien)
- KhachHang(MaKH, TenKH, GioiTinh, DiaChi, SDT)
- NhanVien(MaNV, TenNV, NgaySinh, DiaChi, SDT)
- HoaDon(MaHD, MaNV, MaKH, NgayBan, TongTien)

Đặc tả cơ sở dữ liệu:

STT	Tên thuộc tính	Mô tả	Kiểu dữ liệu	Ràng buộc
	MaNCC	Mã nhà cung cấp	int	Khóa chính
	TenNCC	Tên nhà cung cấp	nvarchar(50)	

Bảng 1. *NhaCungCap*

STT	Tên thuộc tính	Mô tả	Kiểu dữ liệu	Ràng buộc
	MaSP	Mã sản phẩm	int	Khóa chính
	MaNCC	Mã nhà cung cấp	int	Khóa ngoại
	TenSP	Tên sản phẩm	nvarchar(50)	
	SoLuong	Số lượng	int	RB5
	DonGiaNhap	Đơn giá nhập	int	RB2
	DonGiaBan	Đơn giá bán	int	RB2

Bảng 2. *SanPham*

STT	Tên thuộc tính	Mô tả	Kiểu dữ liệu	Ràng buộc
	MaKH	Mã khách hàng	int	Khóa chính
	TenKH	Tên khách hàng	Nvarchar(50)	
	GioiTinh	Giới tính	Nvarchar(4)	
	DiaChi	Địa chỉ	Nvarchar(50)	
	SĐT	Số điện thoại	int	RB1

Bảng 3. *KhachHang*

STT	Tên thuộc tính	Mô tả	Kiểu dữ liệu	Ràng buộc
	MaNV	Mã nhân viên	int	Khóa chính
	TenNV	Tên nhân viên	Nvarchar(50)	
	NgaySinh	Ngày sinh	date	RB3
	GioiTinh	Giới tính	Nvarchar(4)	
	DiaChi	Địa chỉ	Nvarchar(50)	
	SDT	Số điện thoại	int	RB1

Bảng 4. *NhanVien*

STT	Tên thuộc tính	Mô tả	Kiểu dữ liệu	Ràng buộc
	MaHD	Mã hóa đơn	int	Khóa chính
	MaNV	Mã nhân viên	int	Khóa ngoại
	MaKH	Mã khách hàng	int	Khóa ngoại
	NgayBan	Ngày bán	date	
	TongTien	Tổng tiền	int	RB4

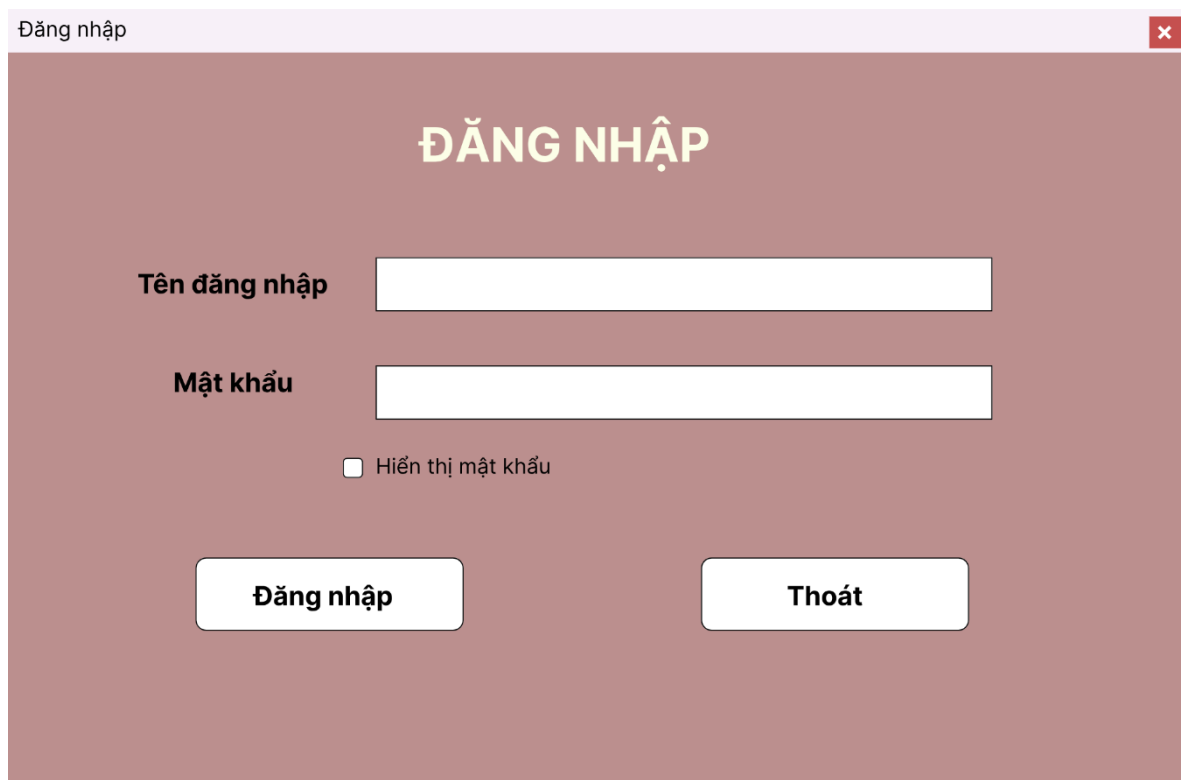
Bảng 5. *HoaDon*

STT	Tên thuộc tính	Mô tả	Kiểu dữ liệu	Ràng buộc
	MaHD	Mã hóa đơn	int	Khóa ngoại
	MaSP	Mã sản phẩm	int	Khóa ngoại
	SoLuong	Số lượng	int	RB4
	GiaGiam	Địa chỉ	int	RB5
	ThanhTien	Thành tiền	int	RB4

Bảng 6. *ChiTietHoaDon*

5.4 SPRINT

5.4.1 Thiết kế các giao diện bằng figma



The image shows a login form titled "ĐĂNG NHẬP" (Login) in a light purple header bar. The form is set against a dark red background. It includes two input fields: "Tên đăng nhập" (Username) and "Mật khẩu" (Password). Below the password field is a checkbox labeled "Hiển thị mật khẩu" (Show password). At the bottom, there are two buttons: "Đăng nhập" (Login) and "Thoát" (Exit).

Đăng nhập

ĐĂNG NHẬP

Tên đăng nhập

Mật khẩu

☐ Hiển thị mật khẩu

Đăng nhập Thoát

Hình 6. Thiết kế giao diện trang Đăng nhập



Hình 7. Thiết kế giao diện Trang chủ

NHÀ CUNG CẤP

Mã Nhà Cung Cấp

Tên Nhà Cung Cấp

Thêm Xóa Sửa Lưu Bỏ qua Đóng

Hình 8. Thiết kế giao diện trang Nhà Cung Cấp

TÌM KIẾM HÓA ĐƠN

Mã Hóa Đơn Mã Khách Hàng

Tháng Năm Mã Nhân Viên

Tìm kiếm Tìm Lại Đóng

Hình 9. Thiết kế giao diện Tìm kiếm hóa đơn

Sản phẩm

SẢN PHẨM

Mã Sản Phẩm	<input type="text"/>	Số Lượng	<input type="text"/>
Mã Nhà Cung Cấp	<input type="text"/>	Đơn Giá Nhập	<input type="text"/>
Tên Sản Phẩm	<input type="text"/>	Đơn giá bán	<input type="text"/>

Hình 10. Thiết kế giao diện trang Sản Phẩm

KHÁCH HÀNG

Mã Khách Hàng	<input type="text"/>	Địa Chỉ	<input type="text"/>
Tên Khách Hàng	<input type="text"/>	Số Điện Thoại	<input type="text"/>

Giới Tính
 ☐ Nam
 ☐ Nữ

Hình 11. Thiết kế giao diện trang Khách Hàng

NHÂN VIÊN

Mã Nhân Viên

Địa Chỉ

Tên Nhân Viên

Điện Thoại

Giới Tính

☐ Nam

☒ Nữ

Ngày Sinh

+ Thêm

Xóa

Sửa

Lưu

Bỏ qua

Đóng

Hình 12. Thiết kế giao diện trang Nhân Viên

HÓA ĐƠN BÁN HÀNG

Thông tin chung

Mã Hóa Đơn

Mã Khách Hàng

Ngày Bán

Tên Khách Hàng

Mã Nhân Viên

Địa Chỉ

Tên Nhân Viên

Điện Thoại

Thông tin các sản phẩm

Mã Sản Phẩm

Tên Sản Phẩm

Đơn Giá

Số Lượng

Giảm Giá (%)

0

Thành Tiền

0

Mã sản phẩm	Tên sản phẩm	Số lượng	Đơn giá	Giảm giá %	Thành tiền

Nháy đúp một dòng để xóa sản phẩm trong hóa đơn

Tổng tiền

Thêm hóa đơn

Lưu hóa đơn

Xóa hóa đơn

Đóng

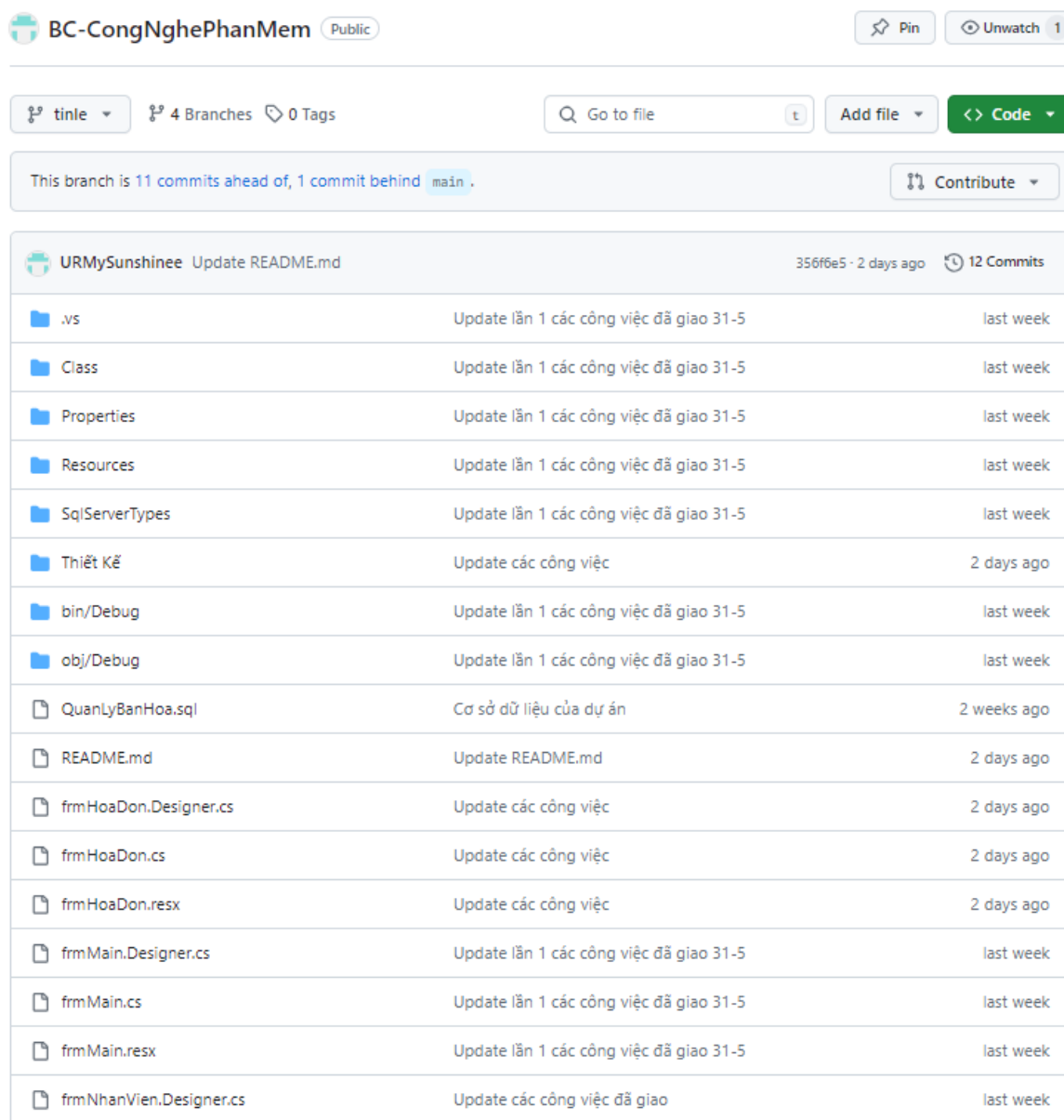
Mã Hóa Đơn

Tìm kiếm

Hình 13. Thiết kế giao diện trang Hóa Đơn Bán Hàng

5.4.2 Commit code cho từng công việc theo bảng phân công

-Commit được phân công của Lê Trực Tín



The screenshot shows the GitHub interface for the repository 'BC-CongNghePhanMem'. At the top, it indicates the repository is public and has 4 branches and 0 tags. The current branch is 'tinle', which is 11 commits ahead of and 1 commit behind the 'main' branch. Below this, a table lists the commit history for the 'tinle' branch, showing the commit message, the commit hash, the time since the commit, and the number of commits.

Commit Message	Commit Hash	Time	Commits
Update README.md	356f6e5	2 days ago	12
Update lần 1 các công việc đã giao 31-5		last week	
Update lần 1 các công việc đã giao 31-5		last week	
Update lần 1 các công việc đã giao 31-5		last week	
Update lần 1 các công việc đã giao 31-5		last week	
Update lần 1 các công việc đã giao 31-5		last week	
Update các công việc		2 days ago	
Update lần 1 các công việc đã giao 31-5		last week	
Update lần 1 các công việc đã giao 31-5		last week	
Cơ sở dữ liệu của dự án		2 weeks ago	
Update README.md		2 days ago	
Update các công việc		2 days ago	
Update các công việc		2 days ago	
Update các công việc		2 days ago	
Update lần 1 các công việc đã giao 31-5		last week	
Update lần 1 các công việc đã giao 31-5		last week	
Update lần 1 các công việc đã giao 31-5		last week	
Update các công việc đã giao		last week	

Hình 14. Commit của Lê Trực Tín theo nhánh "tinle"

-Commit được phân công của Trần Khánh Duy:

The screenshot shows the GitHub repository page for 'BC-CongNghePhanMem'. The repository is public and has 4 branches and 0 tags. The current branch is 'khanhduy', which is 5 commits ahead of and 1 commit behind the 'main' branch. The commit history for the 'khanhduy' branch is displayed, showing 6 commits by user 'khanhduy1910'. The commits are:

Commit Message	Commit Hash	Time Ago
Update chức năng thêm xóa sửa và kết nối cơ sở dữ liệu để quản lý khách...	dee31d2	4 days ago
Sound code		4 days ago
Thiết kế Figma		5 days ago
Initial commit		3 weeks ago

Hình 15. Commit của Trần Khánh Duy theo nhánh "khanhduy"

-Commit được phân công của Lê Thị Nguyễn Xuyên

The screenshot shows the GitHub repository page for 'BC-CongNghePhanMem'. The repository is public and has 4 branches and 0 tags. The current branch is 'nguyensexuyen', which is 4 commits ahead of and 1 commit behind the 'main' branch. The commit history for the 'nguyensexuyen' branch is displayed, showing 5 commits by user 'LeThiNguyenXuyen'. The commits are:

Commit Message	Commit Hash	Time Ago
Update file thiết kế tìm kiếm hóa đơn	26ca004	2 days ago
File Thiết Kế		2 days ago
Initial commit		3 weeks ago
frmNhaCungCap.Designer.cs		2 days ago
frmNhaCungCap.cs		2 days ago
frmNhaCungCap.resx		2 days ago
frmSanPham.Designer.cs		2 days ago
frmSanPham.cs		2 days ago
frmSanPham.resx		2 days ago

Hình 16. Commit của Lê Thị Nguyễn Xuyên theo nhánh "nguyensexuyen"

5.4.3 Burndown Chart



Hình 17. Burndown Chart Sprint 1



Hình 18. Burndown Chart Sprint 2

CHƯƠNG 6: KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

6.1.Kết luận

Kết quả đạt được:

Hoàn thiện đầy đủ chức năng cơ bản của một phần mềm : thêm, sửa, xóa, cập nhật thông tin dựa theo phương pháp Agile.

Giao diện chương trình thân thiện, dễ sử dụng, trực quan và linh hoạt.

Hạn chế:

Do thời gian nghiên cứu còn hạn chế, nên việc nghiên cứu các vấn đề còn chưa đầy đủ.

Kiến thức thực tế và kiến thức lập trình còn hạn chế, nên tính chuyên nghiệp của chương trình chưa cao, bắt lỗi chưa kỹ ở một số chỗ và một số tính năng chưa được tối ưu.

Chương trình chưa thực sự đầy đủ các tính năng như mong muốn.

6.2. Hướng phát triển.

Để chương trình có thể hoạt động hiệu quả hơn cần dữ liệu phải được lưu trữ bằng các hệ quản trị cơ sở dữ liệu có tính bảo mật cao hơn. Cần có sự kết hợp một cách chặt chẽ hơn để việc theo dõi và quản lý cửa hàng bán hoa được tốt hơn.

Để đáp ứng tốt với nhu cầu thực tế thì cần phải nghiên cứu thêm nhu cầu của người dùng để phát triển hệ thống một cách linh hoạt và đơn giản hơn để có thể sử dụng tốt hơn.

DANH SÁCH TÀI LIỆU THAM KHẢO

DANH SÁCH TÀI LIỆU

[1] **Nguyễn Văn Ba**, *Phân tích và thiết kế hệ thống thông tin*, NXB Đại học Quốc gia Hà Nội, 2009.

Link: <https://thuvien.tvu.edu.vn/opac/index.asp>

[2] **Nguyễn Mậu Huân**, *Phân tích thiết kế hệ thống thông tin*.

Link: <https://voer.edu.vn/m/phan-tich-thiet-ke-he-thong-thong-tin/84987530>

[3] **Nguyễn Quang Thông**, *Phân tích thiết kế hệ thống thông tin quản lý*.

Link: <https://voer.edu.vn/c/khao-sat-so-bo-va-xac-lap-du-an/92751615/6e3398f0>

[4] **Valacich**, *Essentials of Systems Analysis & Design*, Pearson, 2012.

Link:

https://drive.google.com/file/d/1bDHGEcq_rfmazEI22B7ZG2PSAAKRiPYI/view?usp=sharing

DANH SÁCH WEBSITE

[5] <https://ngotuongdan.wordpress.com/2012/11/27/chuyen-doi-hinh-thanh-chuoi-vanguoc-lai-Hinh-to-base64-string-and-base64-string-to-Hinh/> (05/2017)

[6] <http://diendan.congdongcviet.com/threads/t42946::do-du-lieu-tu-1-table-of-sql-vaotrong-combobox.cpp> (05/2017)

[7] <https://voer.edu.vn/m/tong-quan-ve-dbms-va-sql-sever/df67417f> (05/2017)

[8] <https://code.msdn.microsoft.com/101-LINQ-Samples-3fb9811b> (05/2017)