



**BÁO CÁO CHUYÊN ĐỀ**  
**CHUYÊN ĐỀ 2: THỰC HIỆN DỰ ÁN**

**Người thực hiện:** .....  
.....

**Hà Nội – ..../202...**

## MỤC LỤC

1. Định nghĩa yêu cầu phần mềm.....	1
1.1. Phát triển yêu cầu dựa trên câu chuyện người dùng.....	1
1.2. Xác định tiêu chí chấp nhận cho các câu chuyện người dùng.....	1
1.3. Ước tính độ phức tạp câu chuyện người dùng.....	2
1.4. Xác định mức độ ưu tiên của user story theo MoSCoW.....	2
1.5. Tạo bản đồ câu chuyện và tinh chỉnh danh sách câu chuyện người dùng .....	3
1.6. Phân bổ câu chuyện người dùng cho các chu trình phát triển phần mềm .....	4
1.7. Danh sách các câu chuyện người dùng trong các chu trình phát triển phần mềm.....	4
2. Phát triển phiên bản sản phẩm phần mềm .....	4
2.1. Phiên bản phần mềm V1.0.....	4
2.1.1. Mục tiêu và kế hoạch thực hiện .....	5
2.1.2. Xây dựng kịch bản câu chuyện người dùng.....	6
2.1.3. Phát triển tính năng trên câu chuyện ...ID của user story... ..	7
2.1.x. Phát hành phiên bản sản phẩm phần mềm .....	7
2.2. Phiên bản phần mềm V2.0.....	8
2.2.1. Mục tiêu và kế hoạch thực hiện .....	8
2.2.2. Xây dựng kịch bản câu chuyện người dùng.....	8
2.2.3. Phát triển tính năng trên câu chuyện ...ID của user story... ..	8
2.2. x. Phát hành phiên bản sản phẩm phần mềm .....	8
3. Kết luận .....	8

## 1. Định nghĩa yêu cầu phần mềm

Lưu ý: Khi trình bày dữ liệu liên quan đến user story, các loại dữ liệu này cần liên kết được với nhau. Điều này tránh được trường hợp khi cập nhật (tinh chỉnh) làm dữ liệu không thống nhất và khó kiểm soát.

### 1.1. Phát triển yêu cầu dựa trên câu chuyện người dùng

Sử dụng định dạng tiêu chuẩn:

"As a **[user]**, I want **[goal]** so that **[reason]**"

để đảm bảo rằng mỗi user story rõ ràng và dễ hiểu. Với:

- User (tác nhân): Định nghĩa rõ ràng về người dùng hoặc vai trò cụ thể.
- Goal (mục tiêu): Mô tả rõ ràng về những gì người dùng muốn đạt được.
- Reason (lý do): Giải thích tại sao mục tiêu này quan trọng.

Danh sách user story viết dưới dạng bảng, gồm các cột: Theme; Epic; UID; User story (Bảng 1.x).

Bảng 1.x. Danh sách các user story

TT	Theme	Epic	UID	User story

### 1.2. Xác định tiêu chí chấp nhận cho các câu chuyện người dùng

Danh sách user story viết dưới dạng bảng, gồm các cột: UID; As a...; I want to be able to ...; So that...; Acceptance Criteria; Testing (Bảng 1.x).

Bảng 1.x. Danh sách user story cùng tiêu chí chấp nhận

UID	As a...	I want to be able to...	So that ...	Acceptance Criteria	Testing

Acceptance Criteria xác định các điều kiện cụ thể mà mỗi user story phải đáp ứng để được coi là hoàn thành. Các tiêu chí phải cụ thể, có thể đo lường và kiểm thử được.

Các trường hợp kiểm thử của mỗi user story được xác định từ các tiêu chí chấp nhận của user story tương ứng.

### 1.3. Ước tính độ phức tạp câu chuyện người dùng

Sử dụng kỹ thuật Planning Poker để ước tính story point và Estimation Effort.

Danh sách user story viết dưới dạng bảng, gồm các cột: UID; User story, Story point và Estimation Effort (Bảng 1.x).

Bảng 1.x. Danh sách user story với các giá trị ước tính Story point và Estimation Effort

UID	User story	Story point	Estimation Effort

### 1.4. Xác định mức độ ưu tiên của user story theo MoSCoW

MoSCoW là một kỹ thuật phổ biến được sử dụng trong quản lý dự án và phát triển phần mềm để ưu tiên các yêu cầu, bao gồm cả user story trong các dự án Agile. MoSCoW là viết tắt của bốn loại mức độ ưu tiên: Must have, Should have, Could have, và Won't have. Đây là cách tiếp cận giúp các nhóm phát triển và các bên liên quan xác định những yêu cầu nào là thiết yếu và những yêu cầu nào có thể được thực hiện sau hoặc thậm chí không cần thực hiện.

- **Must have.** Đây là các user story/yêu cầu bắt buộc phải có cho sản phẩm hoặc hệ thống. Nếu thiếu bất kỳ yêu cầu nào trong nhóm này, dự án có thể bị coi là thất bại.
- **Should have.** Đây là các user story/yêu cầu quan trọng nhưng không bắt buộc. Những yêu cầu này có ảnh hưởng lớn đến thành công của dự án nhưng hệ thống vẫn có thể hoạt động mà không có chúng.
- **Could have.** Đây là các user story/yêu cầu không quá quan trọng và chỉ nên được thực hiện nếu còn đủ thời gian và nguồn lực sau khi hoàn thành các yêu cầu "must have" và "should have".
- **Won't have.** Đây là các yêu cầu sẽ không được thực hiện trong khoảng thời gian hiện tại nhưng có thể được xem xét cho tương lai. Các yêu

cầu này không ảnh hưởng đến thành công của dự án trong giai đoạn hiện tại.

- Danh sách user story viết dưới dạng bảng, gồm các cột: UID; User story, Story point và Estimation Effort (Bảng 1.x).

Bảng 1.x. Danh sách user story với các giá trị ước tính MoSCoW

UID	User story	MoSCoW
		Must have
		Should have
		Could have
		Won't have

### ***1.5. Tạo bản đồ câu chuyện và tinh chỉnh danh sách câu chuyện người dùng***

Bản đồ câu chuyện (Story map) được sắp xếp các user stories theo thứ tự ưu tiên và dòng chảy nghiệp vụ trên product backlog. Các giá trị được sử dụng khi thực hiện viết bản đồ câu chuyện:

- Dựa trên giá trị nghiệp vụ: Ưu tiên các user story dựa trên giá trị nghiệp vụ mà chúng mang lại cho người dùng.
- Độ khó kỹ thuật: Cân nhắc độ khó kỹ thuật và nguồn lực cần thiết.
- Phản hồi khách hàng: Sử dụng phản hồi từ khách hàng để điều chỉnh thứ tự ưu tiên.

Danh sách user story viết dưới dạng bảng, gồm các cột: UID; User story, Story point, Estimation Effort (Bảng 1.x).

Bảng 1.x. Danh sách user story với các giá trị ước tính Story point và Estimation Effort sau tinh chỉnh

UID	User story	Story point	Estimation Effort

## 1.6. Phân bổ câu chuyện người dùng cho các chu trình phát triển phần mềm

Lưu ý, điểm story point và ước tính Effort cho các Sprint phải tương đối cân bằng nhau. Mỗi Sprint thực hiện tối đa không quá 4 tuần.

Danh sách các user story được phân bổ cho các Sprint trên Product backlog gồm các cột: UID, User story, Story point, Estimation Effort, Sprint, Status. Lưu ý, mỗi Sprint lập thành một phiên bản sản phẩm phần mềm.

USID	User story	Story point	Estimated Effort	Sprint	Status
<b>V 1.0 (dd/mm/yyyy)</b>					
US1.1	Là một người quản lý, tôi muốn nhập thông tin nhân viên để quản lý hồ sơ nhân viên.	5	20	1	Done
US1.2	Là một người quản lý, tôi muốn chỉ định phòng ban cho nhân viên mới để họ được phân loại chính xác.	3	30	1	WIP
US1.3	Là một người quản lý, tôi muốn thêm thông tin liên lạc cho nhân viên mới để có thể liên hệ khi cần.	3	25	1	Done
US2.1	Là một người quản lý, tôi muốn cập nhật chức danh công việc của nhân viên để phản ánh chính xác vai trò hiện tại của họ.	3	15	1	Done
US2.2	Là một người quản lý, tôi muốn cập nhật thông tin liên lạc của nhân viên để có được thông tin mới nhất.	3	20	1	Done
US2.3	Là một người quản lý, tôi muốn cập nhật phòng ban của nhân viên để họ được phân loại chính xác.	2	20	1	Done
<b>V 2.0 (dd/mm/yyyy)</b>					
US3.1	Là một người quản lý, tôi muốn xóa hồ sơ nhân viên khi họ rời khỏi công ty để hệ thống chỉ chứa những nhân viên hiện tại.	2	25	2	To do
US3.2	Là một người quản lý, tôi muốn lưu trữ hồ sơ nhân viên thay vì xóa nó để có thể giữ lại dữ liệu lịch sử.	3	30	2	To do
US4.1	US4.1: Là một người quản lý, tôi muốn xem danh sách tất cả nhân viên để có thể nhanh chóng thấy tên và vai trò của họ.	3	25	2	To do
US4.2	US4.2: Là một người quản lý, tôi muốn lọc danh sách nhân viên theo phòng ban để có thể thấy các nhân viên trong một đội cụ thể.	2	25	2	To do
US4.3	US4.3: Là một người quản lý, tôi muốn tìm kiếm nhân viên theo tên để có thể nhanh chóng tìm thấy thông tin của họ.	2	30	2	To do

## 1.7. Danh sách các câu chuyện người dùng trong các chu trình phát triển phần mềm

Mỗi Sprint backlog lập thành 1 bảng, các cột dữ liệu trên bảng gồm: UID, User story, Story point, Estimation Effort, Status, Acceptance Criteria, Testing, Task owner, Work package, Task, Subtask. Lưu ý: Nếu nội dung bảng Sprint backlog quá lớn thì có thể bỏ bớt cột (ví dụ: Work package).

## 2. Phát triển phiên bản sản phẩm phần mềm

### 2.1. Phiên bản phần mềm V1.0

Lưu ý: Mỗi user story cần trình bày thành một đề mục con trong đề mục 2.1, bắt đầu từ đề mục 2.1.3.

### 2.1.1. Mục tiêu và kế hoạch thực hiện

- Trình bày mục tiêu của Sprint:
  - **Xác định phạm vi sprint:** phạm vi của sprint được viết dựa trên Product backlog, bao gồm các user stories và tasks/subtasks cụ thể mà đội phát triển sẽ hoàn thành trong sprint. Điều này giúp xác định rõ ràng các mục tiêu cần đạt được.
  - **Đặt mục tiêu cụ thể, rõ ràng:** mục tiêu của sprint nên cụ thể và tập trung vào một hoặc một vài tính năng chính. Mục tiêu nên rõ ràng để cả đội phát triển và các bên liên quan đều hiểu được.
  - **Đảm bảo mục tiêu có thể đo lường:** mục tiêu của sprint cần phải có thể đo lường được để xác định khi nào mục tiêu đã hoàn thành. Điều này giúp đánh giá được kết quả của sprint.
  - **Phù hợp với mục tiêu nghiệp vụ:** mục tiêu của sprint nên phù hợp với các mục tiêu nghiệp vụ tổng thể của dự án hoặc sản phẩm. Điều này đảm bảo rằng sprint đang đóng góp vào giá trị cốt lõi của dự án.
  - **Đặt mục tiêu thực tế:** Mục tiêu của sprint cần phải thực tế và có thể hoàn thành trong thời gian quy định của sprint (thường là 2 đến 4 tuần). Điều này giúp đảm bảo rằng đội phát triển có thể đạt được mục tiêu mà không bị quá tải hãy trễ thời gian phát hành sản phẩm.
  - **Đưa vào các yêu cầu quan trọng:** Chọn các yêu cầu hoặc user stories quan trọng nhất từ Product backlog để đưa vào Sprint backlog. Điều này đảm bảo rằng những tính năng có giá trị cao nhất được phát triển trước.
  - **Xác định tiêu chí hoàn thành:** định nghĩa rõ ràng các tiêu chí để một công việc được coi là hoàn thành. Điều này bao gồm cả kiểm thử và tài liệu liên quan.
- Trình bày Sprint backlog dưới dạng bảng
- Có thể sử dụng Gantt Chart để lập kế hoạch cho Sprint





**Feature:** Nhập thông tin nhân viên mới

**Scenario:** Thêm thông tin nhân viên mới thành công

**Given** tôi mở biểu mẫu nhập thông tin nhân viên mới

**When** tôi nhập vào các thông tin hợp lệ:

Trường	Giá trị
Tên	Nguyễn Văn A
Ngày sinh	01/01/1990
Địa chỉ	123 Đường ABC
Email	example@gmail.com
Số điện thoại	0912345678
Vị trí	Kỹ sư phần mềm

**And** tôi nhấn nút "Gửi"

**Then** hệ thống hiển thị thông báo "Thông tin đã được lưu thành công"

**And** thông tin nhân viên mới được lưu vào cơ sở dữ liệu

**Scenario:** Thông báo lỗi khi nhập email không hợp lệ

**Given** tôi mở biểu mẫu nhập thông tin nhân viên mới

**When** tôi nhập vào email không hợp lệ "example@com"

**And** tôi nhấn nút "Gửi"

**Then** hệ thống hiển thị thông báo lỗi "Email không hợp lệ"

### 2.1.3. Phát triển tính năng trên câu chuyện ...ID của user story...

Các nội dung cần trình bày trong mỗi user story:

- Nhiệm vụ
- Mô tả nhiệm vụ: mô tả cụ thể đến Task và SubTask
- Thiết kế giao diện và triển khai thiết kế giao diện (sử dụng Hình là các form để minh họa)
- Thiết kế logic và triển khai thiết kế logic (sử dụng các biểu đồ để minh họa)
- Thiết kế cơ sở dữ liệu
- Kiểm thử.

### 2.1.x. Phát hành phiên bản sản phẩm phần mềm

Trình bày các tính năng đã hoàn thành. Các đánh giá quá trình làm việc, nhận diện các điểm mạnh và điểm yếu.

## ***2.2. Phiên bản phần mềm V2.0***

### ***2.2.1. Mục tiêu và kế hoạch thực hiện***

Trình bày mục tiêu của Sprint

Trình bày Sprint backlog dưới dạng bảng

### ***2.2.2. Xây dựng kịch bản câu chuyện người dùng***

Sử dụng cấu trúc kịch bản Gherkin để viết kịch bản cho các user story.

### ***2.2.3. Phát triển tính năng trên câu chuyện ...ID của user story...***

Các nội dung cần trình bày trong mỗi user story:

- Nhiệm vụ
- Mô tả nhiệm vụ: mô tả cụ thể đến Task và SubTask
- Thiết kế giao diện và triển khai thiết kế giao diện (sử dụng Hình là các form để minh họa)
- Thiết kế logic và triển khai thiết kế logic (sử dụng các biểu đồ để minh họa)
- Thiết kế cơ sở dữ liệu
- Kiểm thử.

### ***2.2. x. Phát hành phiên bản sản phẩm phần mềm***

Trình bày các tính năng đã hoàn thành. Các đánh giá quá trình làm việc, nhận diện các điểm mạnh và điểm yếu.

## **3. Kết luận**

