|  |  |
| --- | --- |
| **BỘ GIÁO DỤC VÀ ĐÀO TẠO** | **BỘ QUỐC PHÒNG** |

**HỌC VIỆN KỸ THUẬT QUÂN SỰ**

**\_\_\_\_\_\_\_\_\_\_\_\_\_**

**ĐỀ TÀI**

**XÂY DỰNG ỨNG DỤNG QUẢN LÝ DỰ ÁN PHẦN MỀM**

**CHUYÊN NGÀNH: Công nghệ phần mềm**

**Giáo viên hướng dẫn: Nguyễn Thị Hiền**

**Sinh viên: Nguyễn Thị Hảo**

**HÀ NỘI – NĂM 2017**

**LỤC MỤC**

[**CHƯƠNG 1: TỔNG QUAN VỀ ĐỀ TÀI** 3](#_Toc497995234)

[**1.** **Mô tả bài toán thực tế** 3](#_Toc497995235)

[**2.** **Giới thiệu phần mềm tương tự (Microsoft Project)** 4](#_Toc497995236)

[**3.** **Giới thiệu mô hình phát triển Agile** 6](#_Toc497995237)

[**4.** **Giới thiệu về hệ thống** 9](#_Toc497995238)

[**4.1.** **Đối tượng tham gia hệ thống** 10](#_Toc497995239)

[**4.2.** **Chức năng chính của hệ thống** 10](#_Toc497995240)

[**4.2.1.** **Chức năng quản lý con người** 11](#_Toc497995241)

[**4.2.2.** **Chức năng chính của quản lý thời gian** 13](#_Toc497995242)

[**4.2.3.** **Chức năng chính của quản lý công việc** 15](#_Toc497995243)

[**CHƯƠNG 2: PHÂN TÍCH HỆ THỐNG** 18](#_Toc497995244)

[1. Mô hình Use case 18](#_Toc497995245)

[2. Danh sách các Actor 18](#_Toc497995246)

[3. Danh sách các Use case 18](#_Toc497995247)

[4. Đặc tả các Use case 18](#_Toc497995248)

[5. Biểu đồ lớp 19](#_Toc497995249)

[6. Biểu đồ tuần tự 19](#_Toc497995250)

[**CHƯƠNG 3: THIẾT KẾ HỆ THỐNG** 19](#_Toc497995251)

[**CHƯƠNG 4: KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN** 19](#_Toc497995252)

[**Tài liệu tham khảo** 19](#_Toc497995253)

# **LỜI CẢM ƠN**

# **Bảng các từ viết tắt**

|  |  |  |
| --- | --- | --- |
| STT | Từ viết tắt | Ý nghĩa |
| 1 | CSDL | Cơ sở dữ liệu |
| 2 | PM | Người quản lý dự án phần mềm |
| 3 | QLDA | Quản lý dự án |
| 4 | N/A | Không khả dụng |

**Danh sách các bảng**

# **Danh sách hình ảnh**

**Lời nói đầu**

Trên thế giới, hàng năm có hàng triệu dự án phần mềm được thực hiện mỗi năm. Nhiều dự án trong số này có chất lượng không như kỳ vọng của khách hàng hoặc không cung cấp các phần mềm trong phạm vi ngân sách và thời gian hoàn thành. Tại sao có quá nhiều phần mềm thất bại? Mặc dù có rất nhiều lí do, một trong những lí do quan trọng nhất là quản lý dự án phần mềm không phù hợp. Để giúp nhà quản lý dự án phần mềm có thể nâng cao hiệu quả quản lý dự án tốt hơn, đòi hỏi cần phải có một phần mềm hỗ trợ việc quản lý dự án phần mềm. Phần mềm này giúp cho người quản lý có cái nhìn tổng quan về thời gian, chi phí, công việc trong dự án, quản lý được các rủi ro trong quá trình thực hiện. Từ đó, giúp người quản lý đưa ra các quyết định kịp thời, chính xác và hiệu quả, góp phần vào thành công của dự án phần mềm.

Nội dung của báo cáo đồ án bao gồm:

**Chương 1: Tổng quan về dự án:** Nêu lên thực trạng của quản lý dự án phần mềm hiện nay, mục đích và các yêu cầu cần thiết của ứng dụng quản lý dự án phần mềm. Đánh giá ưu, nhược điểm của các phần mềm tương tự khác. Giới thiệu mô hình phát triển phần mềm Agile. Giới thiệu về môi trường phát triển ứng dụng.

**Chương 2: Phân tích hệ thống:** Phân tích nghiệp vụ của hệ thống, phân tích các use case của hệ thống

**Chương 3: Thiết kế và cài đặt:** Trình bày các bước thiết kế và cài đặt của hệ thống.

**Chương 4: Kết luận và hướng phát triển:** Đánh giá hệ thống và nêu các bước phát triển trong tương lai.

# **CHƯƠNG 1: TỔNG QUAN VỀ ĐỀ TÀI**

Chương này mô tả tổng quan bài toán thực tế tình trạng quản lý phần mềm hiện nay và các nguyên nhân dẫn đến thất bại của một dự án phần mềm, cho thấy tầm quan trọng của việc quản lý dự án phần mềm đối với kết quả thành công của dự án. Trong chương này cũng giới thiệu phần mềm quản lý dự án phần mềm Microsoft Project để thấy được những điểm mạnh cần học hỏi và điểm yếu cần sửa đổi. Từ đó lựa chọn mô hình để phát triển phần mềm, đưa ra mục tiêu và phạm vi của đề tài, giới thiệu môi trường phát triển ứng dụng.

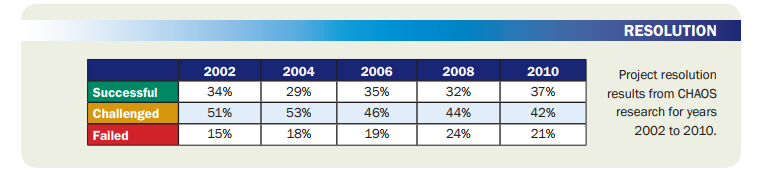
1. **Mô tả bài toán thực tế**

* Định nghĩa về quản lý dự án phần mềm

Không có định nghĩa chung cho quản lý dự án phần mềm. Tuy nhiên, ta có thể hiểu Quản lý dự án phần mềm là tập hợp các công việc được thực hiện bởi một tập thể (có thể có chuyên môn khác nhau, thực hiện công việc khác nhau, thời gian tham gia dự án khác nhau) nhằm đạt được một kết quả như dự kiến, trong thời gian dự kiến, với một kinh phí dự kiến. Trong thuật ngữ của chuyên ngành Công phần mềm, Quản lý dự án phần mềm là các hoạt động trong lập kế hoạch, giám sát và điều khiển tài nguyên dự án (ví dụ như kinh phí, con người), thời gian thực hiện, các rủi ro và quy trình thực hiện dự án nhằm đảm bảo thành công cho dự án. Quản lý dự án phần mềm cần đảm bảo cân bằng giữa ba yếu tố: thời gian, tài nguyên và chất lượng. Ba yếu tố này được gọi là tam giác dự án.[1]

* Thực tế của quản lý dự án phần mềm và tầm quan trọng của quản lý dự án phần mềm

Theo báo cáo thống kê của The Standish Group năm 2014, Hoa Kỳ chi tiêu hơn 250 tỷ đô la mỗi năm cho ứng dụng CNTT phát triển khoảng 175.000 dự án. Chi phí trung bình của một sự phát triển dự án cho một công ty lớn là $ 2,322,000; đối với một công ty trung bình, đó là $ 1,331,000; và đối với một công ty nhỏ, nó là $ 434,000. Nghiên cứu của Standish Group cho thấy một con số đáng kinh ngạc 31,1% số dự án sẽ được hủy bỏ trước khi họ hoàn thành. Các kết quả khác cho thấy có 52,7% dự án sẽ chiếm 189% ước tính ban đầu của họ.[2]



*Bảng: Thống kê kết quả dự án phần mềm qua các năm*

Tiêu chí đánh giá:

* Successful (thành công): hoàn thành đúng thời hạn, theo ngân sách, với các chức năng yêu cầu ban đầu.
* Challenged (thách thức): hoàn thành trễ thời hạn, vượt ngân sách hoặc có ít yêu cầu hơn so với yêu cầu ban đầu
* Failed (thất bại): bị hủy bỏ trước khi hoàn thành hoặc chuyển giao, không bao giờ được sử dụng.

Tại sao có quá nhiều dự án phần mềm thất bại? Một số nguyên nhân dẫn đến sự thất bại của dự án phần mềm theo tổ chức IEEE đưa ra như sau:

* Mất kiểm soát tiến độ dự án, xuất hiện sự sai lệnh và thiếu đồng nhất về dữ liệu giữa tiến độ thực của dự án và tiến độ báo cáo, giữa tiến độ chung và tiến độ các nhóm trong dự án.
* Mục tiêu không rõ ràng
* Ước lượng thiếu chính xác là nguyên nhân của hầu hết các dự án có sai lệch giữa kích thước dự án thực và kích thước ước lượng từ đầu.
* Công nghệ mới
* Không đủ nhân sự [3]

Ba nguyên nhân đầu tiên trong năm nguyên nhân trên liên quan đến quản lý dự án. Hai nguyên nhân sau (công nghệ mới, không đủ nhân sự) có thể xem như rủi ro trong quản lý dự án. Chính vì vậy, quản lý dự án phần mềm hiệu quả có thể cải thiện các cơ hội để thành công.

Quản lý dự án phần mềm có liên quan tới những hoạt động nhằm đảm bảo chuyển giao phần mềm đúng thời hạn, đúng kế hoạch và phù hợp với các yêu cầu của tổ chức phát triển phần mềm. Nếu quản lý tốt thì chưa chắc dự án đã thành công, nhưng nếu quản lý tồi thì chắc chắn dự án sẽ thất bại. Dự án thất bại khi phần mềm chuyển giao chậm hơn so với kế hoạch, chi phí lớn hơn dự tính, và không thoả mãn các yêu cầu đề ra.

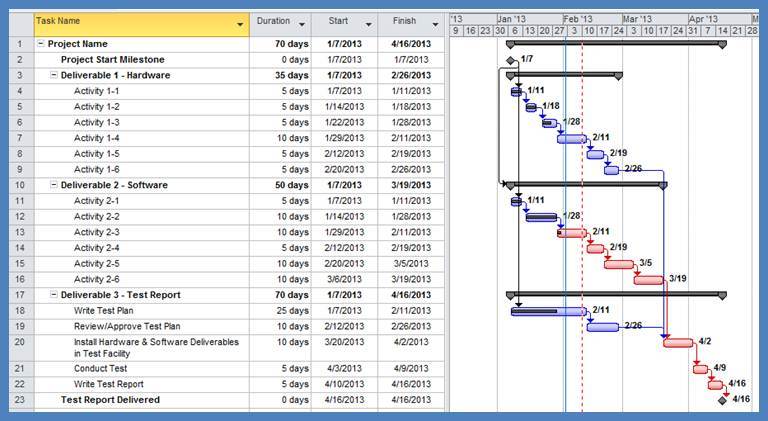
1. **Giới thiệu phần mềm tương tự (Microsoft Project)**

Trên thực tế đã xuất hiện Microsoft Project – là một chương trình của hãng Microsoft giúp bạn có thể lập kế hoạch và quản lý một dự án. Từ thời hạn của một cuộc họp quan trọng cho đến ngân sách thích hợp dành cho nguồn lực của dự án, Microsoft Project cung cấp công cụ làm việc mang lại hiệu quả làm việc tốt hơn. Bạn có thể sử dụng Microsoft Project để:

* Tạo ra các kế hoạch cho dự án ở các cấp độ chi tiết cho một dự án. Làm việc với các thông tin và dữ liệu một cách chi tiết giúp bạn kiểm soát dự án, có thể tự động thiết lập lịch hoặc thiết lập bằng tay.
* Quản lý công việc, chi phí, nguồn lực theo từng cấp độ
* Xem thông tin, dữ liệu của dự án bằng nhiều cách. Áp dụng các nhóm, đánh dấu, sắp xếp, lọc các thông tin mà bạn muốn
* Theo dõi, quản lý kế hoạch trong quá trình thực hiện dự án
* Công tác, chia sẻ những dữ liệu giữa các thành viên trong nhóm để nâng cao năng suất làm việc.[4]

Bên cạnh những chức năng ưu việt đó, phần mềm Microsoft Project còn có những hạn chế sau:

* Chỉ cài đặt được trên hệ điều hành Window
* Phát triển theo mô hình thác nước, mà nhược điểm của mô hình thác nước là khó thay đối, đây là một trở ngại lớn đối với thực tế quản lý dự án phần mềm hiện nay vì thực tế thì hầu hết các dự án phần mềm có tính thay đổi liên tục
* Khó chia sẻ kế hoạch quản lý dự án đến các thành viên trong dự án. Vì chương trình Microsoft Project được cài đặt trên máy tính cục bộ, nên khi có thay đổi về kế hoạch quản lý dự án, các thành viên trong đội khó nắm bắt được thay đổi kịp thời
* Microsoft Project chỉ tập trung chủ yếu vào quản lý thời gian



*Bảng 2.1 Phần mềm Microsoft Project*

Để khắc phục và cải tiến những khuyết điểm của phần mềm Microsoft Project, ứng dụng quản lý dự án phần mềm của em có những cải tiến hơn như sau:

* Phát triển trên nền web nên không phụ thuộc vào hệ điều hành
* Phát triển theo mô hình Agile (ưu điểm của mô hình Agile là: phù hợp với các dự án phần mềm thường xuyên có thay đổi yêu cầu, cải thiện chất lượng phần mềm, rút ngắn thời gian phát triển, sớm đưa ra các sản phẩm làm tăng mức độ hài lòng của khách hàng, kiểm soát dự án tốt và giảm thiểu các rủi do)
* Tập trung vào quản lý thời gian và công việc

1. **Giới thiệu mô hình phát triển Agile Scrum**

* **Giới thiệu chung**

Mô hình Agile Scrum là một trong những mô hình phát triển phần mềm linh hoạt được sử dụng rộng rãi trong các dự án phát triển phần mềm, cung cấp rất nhiều phương pháp luận, quy trình và các thực nghiệm để cho việc phát triển phần mềm trở nên nhanh chóng và dễ dàng. Với nguyên tắc chính là chia nhỏ phần mềm cần sản xuất ra thành các phần nhỏ để phát triển (gọi là Sprint). Mỗi Sprint thường mất 2- 4 tuần để hoàn thành. Vì vậy nó rất phù hợp cho những dự án có nhiều sự thay đổi và yêu cầu tốc độ cao.

Mô hình Scrum vận hành dựa trên đặc tính tự nhiên của người phát triển nên rất dễ hiểu, dễ áp dụng, tạo nên tính tương tác cao giữa những thành viên trong nhóm thay vì chịu sự áp đặt từ bên ngoài.

* **Ưu điểm của mô hình phát triển Agile Scrum**

Agile Scrum đang ngày trở nên phổ biến và được sử dụng rộng rải bởi những nhà phát triển phần mềm. Vậy vì những lợi ích gì mà mô hình Agile Scrum lại được ưa chuộng như vậy?

* Cải thiện chất lượng phần mềm: Framewrok của Scrum giúp nhóm phát triển Scrum nhận phản hồi liên tục và nhanh chóng điều chỉnh để đảm bảo chất lượng phần mềm cao nhất, đồng thời đáp ứng đúng nhu cầu của thị trường luôn thay đổi.
* Rút ngắn thời gian phát hành phần mềm: Scrum đã được chứng minh là cung cấp sản phẩm đến tay khách hàng cuối cùng nhanh hơn 30%-40% so với phương pháp truyền thống. Vì mô hình Scrum làm việc với nguyên tắc chính là chia nhỏ phần mềm cần sản xuất ra thành các phần nhỏ để phát triển gọi là Sprint. Mỗi Sprint thường mất 2- 4 tuần để hoàn thành.
* Nâng cao tinh thần đồng đội: Mô hình Scrum áp dụng cách thức tự quản và tự tổ chức (self-managing & self-organizing ), với mục đích các thành viên trong nhóm Scrum có thể vui vẻ làm việc cùng nhau, khơi dậy sự sáng tạo, chủ động trong họ. Cách thức tự quản lí cũng cho phép mọi thành viên trong nhóm Scrum đều có thể ra quyết định. Trong nhóm Scrum sẽ không có nhóm trưởng mà chỉ có Scrum Master, là người giúp nhóm vượt qua các trở ngại và che chắn cho nhóm khỏi những ảnh hưởng từ nội bộ hay bên ngoài.
* Tăng mức độ hài lòng của khách hành: Sở dĩ như vậy vì nhóm Scrum xem khách hàng là đối tác và giữ khách hàng tham gia vào dự án; thành phần tham gia dự án Scrum còn có Product Owner là người hiểu rõ các yêu cầu (requirements) của dự án và nhu cầu của khách hàng; thời gian cung cấp sản phẩm nhanh hơn
* Kiểm soát dự án tốt: Tất cả các thành viên của nhóm dự án Scrum, Product Ower, Scrum Master và các bên liên quan có rất nhiều cơ hội để kiểm tra và điều chỉnh sản phẩm trong suốt dự án và cuối cùng tạo ra sản phẩm tốt nhất. Vì các framework của mô hình Scrum cho phép nhận các phản hồi liên tục và qua đó có thể nhanh chóng điều chỉnh.
* Giảm thiểu rủi ro: Scrum làm việc theo từng giai đoạn, từng Sprint, nên nhóm dự án có thể thực hiện từng bước, sau đó rút kinh nghiệm hoặc tiếp tục phát huy các ưu điểm của Sprint trước để cải thiện hơn sản phẩm trong Sprint sau tránh gây thất thoát quá lớn trong suốt dự án [5]

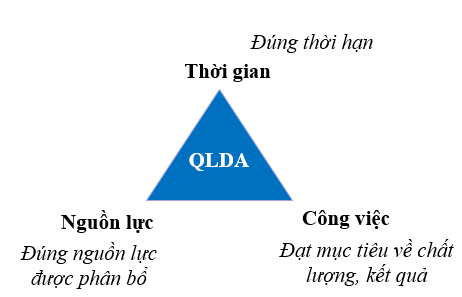
Chính vì những ưu điểm trên nên em đã quyết định lựa chọn mô hình phát triển phần mềm Agile vào ứng dụng của mình.

1. **Mục tiêu của đề tài**
   1. **Mục tiêu tổng quát**

Ứng dụng quản lý dự án phần mềm xây dựng với mục tiêu:

* Phục vụ khai báo, quản lý, thống kê công việc một cách nhanh chóng, chính xác, đồng bộ và đầy đủ thông tin.
* Phục vụ tạo mới, quản lý, điều hành, ra chính sách, quyết định cho người quản lý dự án phần mềm.
  1. **Mục tiêu cụ thể**

Hiểu được tầm quan trọng của quản lý dự án phần mềm trong thành công của dự án là rất quan trọng. Vì vậy mục tiêu đề tài là tạo ra một website giúp người quản lý có một công cụ hỗ trợ trong công việc quản lý dự án phần mềm, làm sao cân bằng giữa các yếu tố trong dự án. Ứng dụng phải đảm bảo các chức năng cơ bản phải có đó là: quản lý con người, quản lý thời gian, quản lý công việc.



Thứ tự ưu tiên của các tiêu chuẩn có thể khác nhau trong các dự án khác nhau. Ví dụ dự án nghiên cứu sản phẩm mới cạnh tranh đòi hỏi thời gian hoàn thành gấp rút và thiết kế riêng biệt trong khi chi phí có thể lớn. Trong khi đó dự án liên quan đến nghiệp vụ thì đặt yêu cầu về chất lượng trên nhất. Việc quản lý dự án phải đảm bảo cân đối được các tiêu chuẩn/ tham số của dự án một cách có hiệu quả nhất. Từ đó có thể nói bản chất của quản lý dự án là quá trình khai thác, cân đối một cách hiệu quả nhất các tham số của một dự án để đạt được kết quả tổng thể tối ưu.

Đề tài xây dựng ứng dụng quản lý dự án phần mềm cung cấp các chức năng chính như:

* Lên kế hoạch phát triển phần mềm, phân chia công việc, nhiệm vụ cho từng thành viên
* Theo dõi hoạt động, đánh giá mức độ ưu tiên trong quá trình phát triển phần mềm
* Tạo ra quy trình làm việc hiệu quả nhất cho toàn đội
* Dễ dàng quản lý nhiều dự án phần mềm cùng lúc
* Cải thiện năng suất làm việc của toàn đội thông qua dữ liệu trực quan và các báo cáo trong thời gian thực hiện

1. **Phạm vi của dự án**

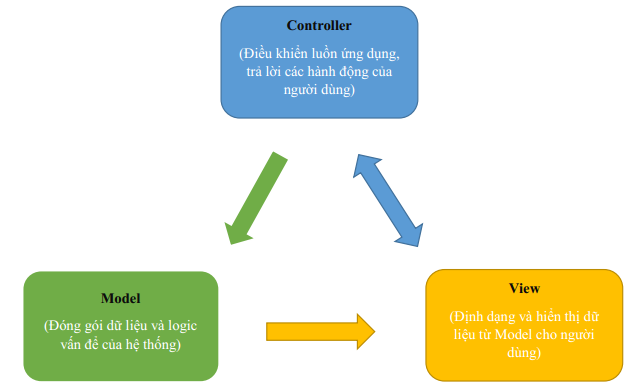
Nguồn lực trong dự án phần mềm bao gồm nhiều yếu tố: Tài chính, con người, máy móc, trình độ của các thành viên tham gia… Đó là các mối quan hệ ràng buộc lẫn nhau. Tuy nhiên, do thời gian thực hiện đề tài còn hạn chế nên đề tài xây dựng ứng dụng quản lý dự án phần mềm của em chỉ xét khía cạnh nguồn lực là con người. Vì vậy, quản lý dự án phần mềm chỉ tập trung vào các chức năng chính: quản lý con người, quản lý thời gian, quản lý công việc.

1. **Môi trường phát triển**
   1. **Giới thiệu mô hình phát triển các ứng dụng web MVC**

* MVC là viết tắt của Model – View – Controller. Là một trong những design pattern. Được vận hành để tách mã lệnh thành 3 phần riêng biệt. Ở mỗi phần MVC sẽ có những chức năng đặc thù. Để xử lý các tác vụ mà request gởi tới. MVC làm cho mã lệnh trở nên trong sáng, dễ phát triển và dễ nâng cấp theo thời gian.

Để làm việc tốt đối với MVC, chúng ta cần nắm thật vững kiến thức OOP. Bản chất của các framework khác cũng được hình thành trên lý thuyết MVC. Do vậy nếu chúng ta nắm tốt MVC. Thì ở những framework khác chắc chắn sẽ không cảm thấy khó hiểu.

* Model: Là thành phần chịu trách nhiệm xử lý các thao tác trên database. Và gởi trả kết quả thông qua view.
* View: Là phần hiển thị thông tin trên website, sau khi đi qua controller và nhận kết quả từ phía model thì view là bước cuối cùng để chuyển thông tin tới người dùng.
* Controller: Là phần điều hướng các request tới những tác vụ tương ứng. Controller là một phần không thể thiếu ở bất cứ framework nào. Vì nó có trách nhiệm gởi và nhận request từ hệ thống tới người sử dụng.



Hình: Mô hình MVC

* Lợi ích của việc sử dụng mô hình MVC trong phát triển website
* Giúp cho ứng dụng dễ bảo trì
* Module hóa các chức năng
* Được xây dựng nhanh chóng
* Dễ dàng thêm các tính năng mới và thay đổi các tính năng cũ
* MCV cho phép các nhà phát triển và thiết kế có thể làm việc đồng thời với nhau
* MVC cho phép thay đổi trong một phần của ứng dụng mà không ảnh hưởng đến các phần khác
* MVC hiện đang là mô hình lập trình tiên tiến bậc nhất hiện nay, điều mà các framework vẫn đang nổ lực để hướng tới sự đơn giản và yếu tố lâu dài cho người sử dụng.
* Một số khuyết điểm của mô hình MVC
* Mặc dù, MVC tỏ ra lợi thế hơn nhiều so với cách lập trình thông thường. Nhưng MVC luôn phải nạp, load những thư viện đồ sộ để xử lý dữ liệu. Chính điều này làm cho mô hình trở nên chậm chạp hơn nhiều so với việc code tay thuần túy.
* MVC đòi hỏi người tiếp cận phải biết qua OOP, có kinh nghiệm tương đối cho việc thiết lập và xây dựng một ứng dụng hoàn chỉnh. Sẽ rất khó khăn nếu OOP của người sử dụng còn yếu.
  1. **Giới thiệu ngôn ngữ lập trình ASP.NET**
* Giới thiệu ngôn ngữ lập trình ASP.NET

ASP.NET là một nền tảng ứng dụng web (web application framework) được phát triển và cung cấp bởi Microsoft, cho phép những người lập trình tạo ra những trang web động, những ứng dụng web và những dịch vụ web.

* Ưu điểm của ngôn ngữ lập trình ASP.NET
* ASP.Net cho phép bạn lựa chọn một trong các ngôn ngữ lập trình mà bạn yêu thích: Visual Basic.Net, J#, C#.
* Thay vì phải đọc và thông dịch mỗi khi trang web được yêu cầu, ASP.Net biên dịch những trang web động thành những tập tin DLL mà Server có thể thi hành nhanh chóng và hiệu quả. Yếu tố này là một bước nhảy vọt đáng kể so với kỹ thuật thông dịch của ASP.
* ASP.Net hỗ trợ mạnh mẽ bộ thư viện phong phú và đa dạng của .Net Framework, làm việc với XML, Web Service, truy cập cơ sở dữ liệu qua ADO.Net
* ASP.Net sử dụng phong cách lập trình mới: Code behide. Tách code riêng, giao diện riêng do vậy dễ đọc, dễ quản lý và bảo trì.
  1. **Giới thiệu hệ quản trị cơ sở dữ liệu SQL server**
* Định nghĩa hệ quản trị cơ sở dữ liệu

Là một hệ quản trị cơ sở dữ liệu quan hệ hoạt động theo mô hình khách chủ cho phép đồng thời nhiều người dùng cùng truy xuất đến dữ liệu, quản lý việc truy nhập hợp lệ và các quyền của từng người dùng trên mạng. Hệ quản trị cơ sở dữ liệu hỗ trợ khả năng lưu trữ, sửa chữa, xóa và tìm kiếm thông tin trong cơ sở dữ liệu (CSDL).

* Ưu điểm của SQL server
* Quản lý dữ liệu dư thừa
* Đảm bảo tính nhất quán của dữ liệu
* Chia sẻ dữ liệu được nhiều hơn
* Đảm bảo tính toàn vẹn cho dự liệu
* Khuyết điểm của SQL server
* Khá phức tạp
* Chiếm nhiều dung lượng bộ nhớ
* Chỉ chạy trên hệ điều hành Windows

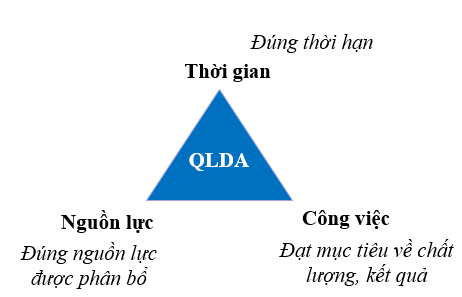
Tóm lại, trong chương này, chúng ta vừa tìm hiểu quản lý dự án phần mềm là rất quan trọng, nó ảnh hưởng đến thành công của dự án. Có ba yếu tố chính trong quản lý dự án phần mềm là: con người, thời gian, công việc. Bên cạnh những ưu điểm của phần mềm Microsoft Project cần học hỏi thì cũng có những khuyết điểm cần bổ sung thay thế. Trong ứng dụng của em quyết định lựa chọn mô hình phát triển phần mềm Agile bởi vì nó có nhiều ưu điểm phù hợp với dự án phần mềm có yêu cầu thường xuyên thay đổi. Chức năng chỉnh của một ứng dụng quản lý dự án phần mềm cần thiết phải có là: quản lý con người, quản lý thời gian và quản lý công việc. Để thực hiện được các yêu cầu chức năng đã nên ở trên, chương 2 sẽ giúp chúng ta phân tích hệ thống, đưa ra các đối tượng sử dụng ứng dụng và các use case của ứng dụng

# **CHƯƠNG 2:** **PHÂN TÍCH HỆ THỐNG**

Trong chương này, tôi sẽ phân tích các chức năng chính cần có của hệ thống, phân tích use case và đặc tả các use case, phân tích các biểu đồ lớp và biểu đồ tuần tự.

1. **Phân tích nghiệp vụ**
   1. **Giới thiệu về hệ thố****ng**

Hiểu được tầm quan trọng của quản lý dự án phần mềm trong thành công của dự án là rất quan trọng. Vì vậy mục tiêu đề tài là tạo ra một website giúp người quản lý có một công cụ hỗ trợ trong công việc quản lý dự án phần mềm, làm sao cân bằng giữa các yếu tố trong dự án. Ứng dụng phải đảm bảo các chức năng cơ bản phải có đó là: quản lý con người, quản lý thời gian, quản lý công việc.



*Bảng 2.2 Mục tiêu của quản lý dự án phần mềm*

Thứ tự ưu tiên của các tiêu chuẩn có thể khác nhau trong các dự án khác nhau. Ví dụ dự án nghiên cứu sản phẩm mới cạnh tranh đòi hỏi thời gian hoàn thành gấp rút và thiết kế riêng biệt trong khi chi phí có thể lớn. Trong khi đó dự án liên quan đến nghiệp vụ thì đặt yêu cầu về chất lượng trên nhất. Việc quản lý dự án phải đảm bảo cân đối được các tiêu chuẩn/ tham số của dự án một cách có hiệu quả nhất. Từ đó có thể nói bản chất của quản lý dự án là quá trình khai thác, cân đối một cách hiệu quả nhất các tham số của một dự án để đạt được kết quả tổng thể tối ưu.

Đề tài xây dựng ứng dụng quản lý dự án phần mềm cung cấp các chức năng chính như:

* Lên kế hoạch phát triển phần mềm, phân chia công việc, nhiệm vụ cho từng thành viên
* Theo dõi hoạt động, đánh giá mức độ ưu tiên trong quá trình phát triển phần mềm
* Tạo ra quy trình làm việc hiệu quả nhất cho toàn đội
* Dễ dàng quản lý nhiều dự án phần mềm cùng lúc
* Cải thiện năng suất làm việc của toàn đội thông qua dữ liệu trực quan và các báo cáo trong thời gian thực hiện
  1. **Đối tượng tham gia hệ thống**

Các đối tượng chính tham gia hệ thống gồm có: **Admin, Project Manager, Member**.

* **Admin**: là người có quyền quản trị hệ thống, duy trì điều hành toàn bộ phần mềm như: tạo ra các danh mục quản lý, có quyền quyết định cao nhất đối với hệ thống.
* **Project Manager**: người quản lý dự án là người sử dụng hệ thống, có quyền tạo một dự án,các giai đoạn, công việc, xem thống kê, nhìn tổng quan dự án và phân quyền cho những người tham gia.
* **Member**: người tham gia vào dự án, thực hiện khai báo công việc mình làm trong dự án, có quyền xem các thống kê liên quan đến công việc được giao. Tùy thuộc theo quyền được Project manager cấp phép, member cũng có thể tạo một công việc mới (team lead).
  1. **Chức năng chính của hệ thống**

Ba yếu tố chính trong quản lý dự án phần mềm, tạo nên tam giác dự án đó là: **Quản lý con người**, **quản lý thời gian** và **quản lý công việc.** 3 yếu tố chính này tạo nên tam giác dự án:

*Bảng 2.3 Ba yếu tố chính của quản lý dự án phần mềm*

Ứng dụng quản lý dự án phần mềm là một hệ thống hỗ trợ những người tham gia dự án phần mềm khai báo kế hoạch và công việc của mình trong dự án thông qua mạng internet. Ứng dụng này được xây dựng trên website, cho phép người dùng có thể sử dụng ở mọi nơi, không hạn chế về khoảng cách địa lý, chỉ cần bạn có kết nối internet. Lý do để chọn phát triển ứng dụng trên website nhằm giúp những người sử dụng không nhất thiết phải ở gần nhau nhưng vẫn có thể theo dõi tiến độ của toàn dự án.

Với ứng dụng này, người quản lý dự án phần mềm có thể dễ dàng lên kế hoạch phát triển và phân chia công việc cho từng thành viên trong dự án, theo dõi hoạt động, đánh giá mức độ ưu tiên trong quá trình phát triển phần mềm, cải thiện năng suất làm việc thông qua dữ liệu trực quan và báo cáo dự án.

Đối với người tham gia dự án, ứng dụng là công cụ giúp họ khai báo công việc thực hiện theo từng ngày, theo dõi công việc được phân công cho mình và có kế hoạch thực hiện theo mức độ ưu tiên của từng công việc.

Để quản lý tốt dự án, hệ thống cần đáp ứng được các chức năng chính sau:

**1.3.1 Chức năng quản lý con người**

* Quản lý danh sách người dùng

- Tạo mới một người dùng: Admin có quyền tạo một tài khoản cho người dùng (người quản lý dự án, thành viên của dự án)

- Chỉnh sửa người dùng: Admin có thể thay đổi thông tin của người dùng.

- Xóa người dùng: Khi tài khoản người dùng không còn hiệu lực, admin có quyền chuyển trạng thái người dùng thành invalid, không xóa hẳn tài khoản người dùng trong database.

* Quản lý chất lượng nguồn lực

Trong một công việc được giao, nguồn lực (cụ thể ở đây là con người) có thể hoàn thành sớm hơn hoặc trễ hơn thời gian hạn định cho công việc đó. Dựa vào chất lượng nguồn lực, có thể đánh giá được việc sử dụng nguồn lực và việc giao công việc cho nguồn lực có hiệu quả không.

Ứng dụng cung cấp chức năng thống kê công việc mà thành viên thực hiện: nhanh hơn so với dự định, chậm hơn so với dự định, kịp tiến độ… Từ đó, giúp người quản lý dự án phần mềm có căn cứ để điều chỉnh công việc cho thành viên này và đánh giá năng lực của thành viên.

* Phân quyền cho nguồn lực

Có nhiều đối tượng tham gia sử dụng hệ thống quản lý dự án. Vì vậy cần phân quyền cho các nguồn lực này để đảm bảo yếu tố bảo mật trong công việc. Ví dụ: Người thuộc đội tester thì không thể xem được công việc của người thuộc đội phát triển. Có các mức độ phân quyền trong ứng dụng đó là:

* Quyền Admintrantor: Quyền cao nhất, có thể thêm mới, sửa đổi trong toàn bộ ứng dụng
* Quyền project manager: Quyền dành cho người quản lý dự án, có quyền tạo mới và sửa đổi một dự án, có quyền phân quyền cho các thành viên trong dự án.
* Quyền team lead: Là người trực tiếp tạo, sửa đổi các công việc cho các thành viên trong đội phát triển của mình, và theo dõi công việc của nhóm mình.
* Quyền member: là quyền dành cho người tham gia vào dự án. Những người này được quyền khai báo công việc mình làm và theo dõi tiến độ công việc của mình.

**1.3.2 Chức năng chính của quản lý thời gian**

* Thiết lập thời gian cho dự án

Thời gian là một yếu tố quan trọng trong quản lý dự án phần mềm. Thiết lập thời gian cho dự án tức là người quản lý sẽ thiết lập thời gian bắt đầu và kết thúc của dự án.

* Khai báo thời gian bắt đầu, kết thúc của dự án
* Chỉnh sửa thời gian bắt đầu, kết thúc của dự án
* Không thể xóa bỏ thời gian bắt đầu, kết thúc của dự án.
* Thiết lập cột mốc quan trọng của dự án

Mỗi dự án đều có một số cột mốc quan trọng như: ngày hoàn thiện đặc tả yêu cầu, ngày bàn giao sản phẩm… Mỗi cột mốc là thời điểm giúp nhà quản lý nhìn lại dự án đang ở tình trạng như thế nào, có chậm tiến độ hay không, từ đó đưa ra những điều chỉnh phù hợp, kịp thời.

* Khai báo cột mốc quan trọng (chính là trạng thái cao nhất của mức độ ưu tiên của công việc)
* Chỉnh sửa cột mốc quan trọng
* Xóa cột mốc quan trọng
* Biểu diễn cột mốc quan trong trên biểu đồ Grantt chart.
* Thiết lập thời gian cho mỗi giai đoạn

Mỗi dự án được chia thành các giai đoạn khác nhau. Chẳng hạn như: Phân tích yêu cầu phần mềm, thiết kế phần mềm, coding, testing… Mỗi giai đoạn nên xác định ngày bắt đầu và kết thúc từ khi bắt đầu lập kế hoạch cho dự án để quản lý tiến độ dự án được tốt nhất. Tương tự như thiết lập thời gian cho dự án (Xem thêm mục 2.2.1)

* Quản lý thời gian của mỗi công việc

Tương tự như thiết lập thời gian cho giai đoạn, dự án, mỗi công việc cũng được thiết lập các mốc thời gian bắt đầu, kết thúc. (Xem thêm mục 2.2.1). Tuy nhiên, quản lý thời gian của mỗi công việc cần thêm:

* Khai báo ngày thực tế bắt đầu công việc
* Chỉnh sửa ngày thực tế bắt đầu công việc
* Khai báo ngày thực tế kết thúc công việc
* Chỉnh sửa ngày thực tế kết thúc công việc.

Mục đích cần có thêm quản lý ngày thực tế của bắt đầu, kết thúc công việc để so sánh với ngày dự kiến bắt đầu, kết thúc công việc. Từ đó, có thể đánh giá được khả năng tạo kế hoạch làm việc của người quản lý dự án, các đội trưởng đội phát triển.

* Thiết lập thời gian những ngày không làm việc

Xác định thời gian làm việc và không làm việc đối với các công việc. Chẳng hạn như ngày làm việc 8 giờ, tuần làm việc 5 ngày.

* Khai báo thời gian làm việc trong tuần
* Chỉnh sửa thời gian làm việc trong tuần
* Quản lý thời gian làm việc của từng cá nhân (khai báo timesheet)

Mỗi cá nhân tham gia dự án phải khai báo thời gian làm việc thực tế từng ngày mà mình thực hiện.

* Khai báo thời gian thực hiện, mô tả công việc đã làm
* Chỉnh sửa thời gian thực hiện, chỉnh sửa mô tả việc đã làm
* Xóa thời gian thực hiện, xóa mô tả việc đã làm
* Xem tổng quan dự án thông qua biểu đồ Grantt Chart

Grantt chart là một biểu đồ trực quan, giúp người quản lý có thể nhìn tổng quát tất cả các giai đoạn của dự án và nắm rõ tiến độ của dự án. Biểu đồ Grantt chart cung cấp những thông tin khối lượng công việc đã hoàn thành, đang thực hiện, chưa thực hiện… trên hình ảnh trực quan, những cột mốc quan trọng cần chú ý, từ đó giúp cho người quản lý đưa ra những điều chỉnh phù hợp và kịp thời.

**1.3.3 Chức năng chính của quản lý công việc**

* Quản lý danh sách dự án
* Tạo mới một dự án: người quản lý dự án phần mềm có quyền tạo một dự án
* Chỉnh sửa các thông tin của dự án: Chỉnh sửa, cập nhật thông tin của dự án khi cần thiết
* Xóa dự án: Không được phép xóa dự án khỏi database mà chỉ có thể thay đổi trạng thái của dự án thành closed khi đã kết thúc dự án.
* Quản lý danh sách giai đoạn của dự án

Tổ chức các công việc thành giai đoạn là một việc làm quan trọng trong kế hoạch của dự án. Mục đích của việc chia các giai đoạn để dễ dàng quản lý hơn. Thời gian thực hiện giai đoạn phụ thuộc vào số lượng công việc trong giai đoạn đó.

* Tạo mới một giai đoạn trong dự án: người quản lý dự án phần mềm có thể tạo các giai đoạn trong dự án để dễ dàng quản lý.
* Chỉnh sửa giai đoạn trong dự án: chỉnh sửa, cập nhật các thông tin của giai đoạn trong dự án khi cần thiết
* Xóa giai đoạn trong dự án: xóa giai đoạn trong dự án khi cần thiết
* Quản lý danh sách công việc

Trong mỗi giai đoạn của dự án lại được chia thành các công việc nhỏ hơn. Chức năng quản lý danh sách công việc tương tự như quản lý danh sách các giai đoạn của dự án. Có thể tham khảo thêm mục 2.3.2

* Quản lý liên kết các công việc

Hầu hết các dự án yêu cầu công việc phải thực hiện theo một trình tự nhất định. Những việc này có mối quan hệ kết thúc – bắt đầu (hay còn gọi là tính phụ thuộc):

* Công việc thứ 2 xảy ra sau khi công việc đầu tiên được thực hiện. Đây là một trình tự.
* Công việc thứ 2 xảy ra chỉ sau khi công việc đầu tiên được hoàn thành. Đây là tính phụ thuộc.

Các chức năng chính:

* Khai báo công việc cần thực hiện trước
* Chỉnh sửa, cập nhật công việc cần thực hiện trước
* Xóa công việc cần thực hiện trước: khi không có công việc cần thực hiện trước thì không có thứ tự thực hiện các công việc, có nghĩa là có thể thực hiện công việc nào trước cũng được.
* Quản lý trạng thái của công việc
* Thêm mới trạng thái của công việc:
* Open: Công việc đã được tạo và đang được chỉ định cho người thực hiện.
* In progress: Công việc đang trong tiến trình thực hiện
* Done: Công việc đã được hoàn thành
* Cancel: Công việc được loại bỏ (Một công việc được tạo ra không thể xóa bỏ, chỉ có thể sửa đổi hoặc chuyển đổi về trạng thái Cancel)
* Sửa đổi trạng thái của công việc: Chỉnh sửa, cập nhật trạng thái của công việc khi cần thiết
* Xóa trạng thái của công việc: Xóa bỏ trạng thái của công việc khi cần thiết
* Quản lý mức độ ưu tiên của công việc

Trong một thời điểm, các công việc sẽ có độ ưu tiên khác nhau. Việc phân loại độ ưu tiên giúp người sử dụng hệ thống dễ dàng quyết định nên thực hiện công việc nào trước, công việc nào sau.

* Thêm mới mức độ ưu tiên: Chẳng hạn như có các mức độ ưu tiên: cột mốc quan trọng của dự án, mức độ ưu tiên cao, mức độ ưu tiên trung bình, mức độ ưu tiên thấp.
* Chỉnh sửa mức độ ưu tiên: thay đổi, cập nhật mức độ ưu tiên khi cần thiết
* Xóa mức độ ưu tiên: xóa các mức độ ưu tiên khi không cần thiết.
* Xem tổng quan dự án

Tùy vào mực độ phân quyền mà người dùng có thể xem được những thông tin nào của dự án. Đối với người quản lý dự án, chức năng xem tổng quan dự án cho phép họ nắm bắt được toàn bộ thông tin liên quan đến dự án, thống kê được có bao nhiêu công việc đã được tạo ra, bao nhiêu công việc đang được thực hiện hoặc đã hoàn thành….

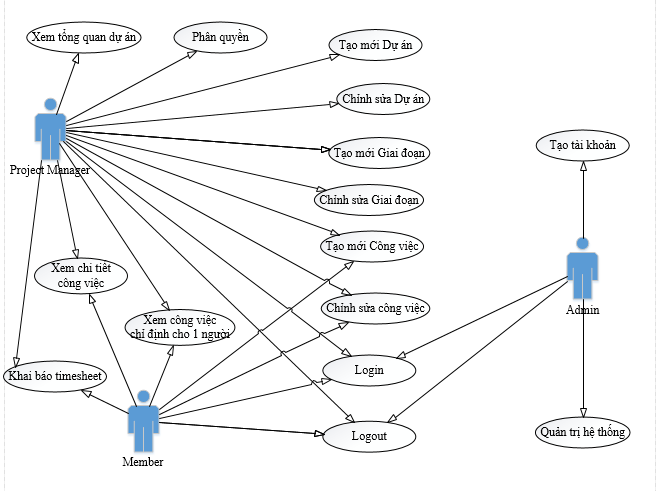
* Chỉ định người thực hiện công việc

Mỗi công việc được giao cho một hoặc một nhóm người thực hiện. Nhìn vào một công việc, ta có thể biết được ai là người thực hiện nó, ta cũng có thể xem danh sách công việc được chỉ định cho mình.

* Chỉ định người thực hiện công việc
* Thay đổi người thực hiện công việc
* Một công việc nhất thiết phải có người được chỉ định thực hiện (không được bỏ trống trường này)
* Xem tổng quan công việc của từng cá nhân

Do cần phải chỉ định người thực hiện cho từng công việc nên mỗi cá nhân có thể xem tổng quan công việc mà mình đã được giao, thống kê số lượng công việc đã được giao, công việc đang thực hiện, công việc đã hoàn thành xong.

1. **Mô hình Use case**



1. **Danh sách các Actor**

|  |  |  |
| --- | --- | --- |
| **STT** | **Actor** | **Ý nghĩa** |
| 1 |  | **Admin**: là người có quyền quản trị hệ thống, duy trì điều hành toàn bộ phần mềm như: Tạo tài khoản ban đầu cho người dùng, tạo ra các danh mục quản lý, có quyền quyết định cao nhất đối với hệ thống. |
| 2 |  | **Project Manager**: người quản lý dự án là người sử dụng hệ thống, có quyền tạo một dự án,các giai đoạn, công việc, xem thống kê, nhìn tổng quan dự án và phân quyền cho những người tham gia. |
| 3 |  | **Member**: người tham gia vào dự án, thực hiện khai báo công việc mình làm trong dự án, có quyền xem các thống kê liên quan đến công việc được giao. Tùy thuộc theo quyền được Project manager cấp phép, member cũng có thể tạo một công việc mới. |

1. **Danh sách các Use case**

|  |  |  |
| --- | --- | --- |
| **STT** | **Use-case** | **Mô tả** |
| 1 | Tạo tài khoản | Admin tạo tài khoản cho các thành viên trong dự án |
| 2 | Login | Đăng nhập vào ứng dụng |
| 3 | Logout | Đăng xuất khỏi ứng dụng |
| 4 | Phân quyền | Phân loại các quyền truy cập của người dùng:  - Quyền Admintrantor  - Quyền Project Manager  - Quyền Team Lead  - Quyền Member |
| 5 | Tạo mới Dự án | PM có quyền tạo mới một dự án |
| 6 | Chỉnh sửa Dự án | PM có thể chỉnh sửa, cập nhật trạng thái của dự án khi cần thiết |
| 7 | Tạo mới Giai đoạn | PM có quyền tạo mới một Giai đoạn trong dự án |
| 8 | Chỉnh sửa giai đoạn | PM có thể chỉnh sửa, cập nhật trạng thái của giai đoạn khi cần thiết |
| 9 | Tạo mới Công việc | Member được phân quyền (team lead) có thể tạo mới một công việc |
| 10 | Chỉnh sửa công việc | Team lead có thể chỉnh sửa, cập nhật trạng thái của công việc khi cần thiết |
| 11 | Khai báo timesheet | Thành viên của dự án có thể khai báo thời gian làm việc, mô tả các công việc đã thực hiện |
| 12 | Quản trị hệ thống | Admin có quyền thêm mới, sửa đổi, cập nhật các thông tin, các trường dữ liệu cần quản lý. |
| 13 | Xem tổng quan dự án | Tùy vào quyền truy cập của người dùng mà có thể xem thông tin tổng quan của dự án (công việc đã hoàn thành, chưa hoàn thành, đang thực hiện, tính hiệu quả của sử dụng nguồn lực…) |
| 14 | Xem chi tiết công việc | Từng thành viên trong dự án có thể xem chi tiết công việc được giao cho mình, tiến độ của công việc và kế hoạch thời gian thực hiện công việc trong bao lâu |
| 15 | Xem công việc được chỉ định cho 1 người | Có thể xem được các công việc được chỉ định cho từng người trong dự án. |

1. **Đặc tả các Use case**
   1. **Đặc tả use case “Tạo tài khoản”**

|  |  |  |  |
| --- | --- | --- | --- |
| **Use case “Tạo tài khoản”** | | | |
| Mục đích | | Admin tạo tài khoản cho các thành viên trong dự án | |
| Tác nhân | | Admin | |
| Điều kiện trước | | Admin đã truy cập vào hệ thống bằng quyền admin | |
| Luồng sự kiện chính (Basic flow) | | Hành động của tác nhân | Hoạt động của hệ thống |
| 1. Admin chọn chức năng tạo tài khoản. |  |
|  | 2. Hiển thị giao diện màn hình tạo tài khoản |
| 3. Nhập thông tin account cần tạo và các trường dữ liệu liên quan |  |
|  | 4. Hệ thống nhận thông tin dữ liệu gửi lên, kiểm tra thông tin dữ liệu  - Nếu đúng thì hiển thị thông báo đã đăng ký tài khoản thành công.  - Nếu thông tin đăng ký lỗi thì chuyển sang luồng sự kiện rẽ nhánh 4.1 |
| Luồng sự kiện phụ (Alternative flow) | | 4.1 Dữ liệu đăng ký không hợp lệ  - Hệ thống đưa ra thông báo lỗi  - Hiển thị lại màn hình đăng ký với các trường thông tin đã nhập, đánh dấu các trường thông tin đang bị lỗi  - Nếu admin muốn tiếp tục đăng ký thì quay về bước 2. | |
| Điều kiện sau | Success | Hiển thị thông báo đã đăng ký tài khoản thành công. Thêm tài khoản mới vào CSDL | |
| Fail | Hiển thị thông báo đăng ký lỗi. CSDL không bị thay đổi. | |

* 1. **Đặc tả use case “Login”**

|  |  |  |  |
| --- | --- | --- | --- |
| **Use case “Login”** | | | |
| Mục đích | | Người dùng đăng nhập vào ứng dụng | |
| Tác nhân | | Người dùng (Admin, PM, team lead, member) | |
| Điều kiện trước | | Người dùng truy cập vào ứng dụng, chức năng tự động ghi nhớ mật khẩu chưa được lựa chọn | |
| Luồng sự kiện chính (Basic flow) | | Hành động của tác nhân | Hoạt động của hệ thống |
| 1. Truy cập ứng dụng |  |
|  | 2. Hiển thị giao diện Login vào ứng dụng |
| 3. Nhập account, password  Lựa chọn có ghi nhớ mật khẩu hay không |  |
|  | 4. Hệ thống nhận thông tin dữ liệu gửi lên, kiểm tra thông tin dữ liệu  - Nếu đúng thì chuyển đến màn hình Trang chủ của ứng dụng  - Nếu thông tin login lỗi thì chuyển sang luồng sự kiện rẽ nhánh 4.1  - Nếu trước đó đã lựa chọn chức năng tự động ghi nhớ mật khẩu thì chuyển sang luồng sự kiện rẽ nhánh 4.2 |
| Luồng sự kiện phụ (Alternative flow) | | 4.1 Dữ liệu đăng ký không hợp lệ   * Hệ thống đưa ra thông báo lỗi * Hiển thị lại màn hình đăng nhập với thông báo tên tài khoản hoặc mật khẩu đang bị sai * Nếu người dùng muốn tiếp tục đăng nhập thì quay về bước 2.   4.2 Trước đó đã lựa chọn chức năng tự động đăng nhập (với account và password đúng) thì chuyển sang màn hình Trang chủ của ứng dụng | |
| Điều kiện sau | Success | Chuyển sang màn hình Trang chủ của ứng dụng. CSDL không bị thay đổi | |
| Fail | Hiển thị thông báo đăng nhập lỗi. CSDL không bị thay đổi. | |

* 1. **Đặc tả use case “Logout”**

|  |  |  |  |
| --- | --- | --- | --- |
| **Use case “Logout”** | | | |
| Mục đích | | Người dùng thoát truy cập khỏi ứng dụng | |
| Tác nhân | | Người dùng | |
| Điều kiện trước | | Người dùng đã login vào ứng dụng | |
| Luồng sự kiện chính (Basic flow) | | Hành động của tác nhân | Hoạt động của hệ thống |
| 1. Admin chọn chức năng logout |  |
|  | 2. Hệ thống nhận yêu cầu, thoát khỏi quyền truy cập của account hiện tại  Hiển thị màn hình login  Nếu xảy ra lỗi, chuyển đến màn hình thông báo lỗi với thông báo không thể đăng xuất khỏi tài khoản |
| Luồng sự kiện phụ (Alternative flow) | | N/A | |
| Điều kiện sau | Success | Hiển thị màn hình login | |
| Fail | Hiển thị màn hình thông báo lỗi không thể đăng xuất khỏi tài khoản | |

* 1. **Đặc tả use case “Phân quyền”**

|  |  |  |  |
| --- | --- | --- | --- |
| **Use case “Phân quyền”** | | | |
| Mục đích | | Phân quyền truy cập cho từng đối tượng người dùng, hạn chế những truy cập không cần thiết để đảm bảo độ bảo mật thông tin cho dự án. | |
| Tác nhân | | PM | |
| Điều kiện trước | | Đăng nhập vào ứng dụng | |
| Luồng sự kiện chính (Basic flow) | | Hành động của tác nhân | Hoạt động của hệ thống |
| 1. Truy cập vào ứng dụng và chọn chức năng phân quyền |  |
|  | 2. Nhận yêu cầu, kiểm tra người dùng có quyền được phân quyền hay không.  - Nếu người dùng có quyền Project manager thì chuyển sang màn hình phân quyền  - Nếu người dùng không có quyền của Project Manager thì chuyển sang luồng sự kiện rẽ nhánh 2.1 |
| 3. Chọn một acccount trong dự án và phân quyền truy cập cho account đó, nhấn button Xác nhận |  |
|  | |  | 4. Nhận dữ liệu gửi lên và kiểm tra thông tin dữ liệu  - Nếu đúng thì hiện lên thông báo cấp quyền thành công  - Nếu thông tin lỗi thì chuyển sang luồng sự kiện rẽ nhánh 4.1 |
| Luồng sự kiện phụ (Alternative flow) | | 2.1 Không có quyền truy cập của Project Manager   * Hệ thống đưa ra thông báo không thể thực hiện phân quyền cho người dùng   4.1 Nếu dữ liệu gửi lên thiếu hoặc sai   * Hệ thống đưa ra thông báo lỗi * Quay trở lại màn hình phân quyền với các trường dữ liệu đã nhập * Nếu người dùng muốn tiếp tục đăng nhập thì quay về bước 3 | |
| Điều kiện sau | Success | Hiển thị thông báo đã cấp quyền truy cập cho account thành công. CSDL bị thay đổi | |
| Fail | Hiển thị thông báo cấp quyền truy cập lỗi. CSDL không bị thay đổi. | |

* 1. **Đặc tả use case “Tạo mới dự án”**

|  |  |  |  |
| --- | --- | --- | --- |
| **Use case “Tạo mới dự án”** | | | |
| Mục đích | | Tạo mới một dự án | |
| Tác nhân | | PM | |
| Điều kiện trước | | Đăng nhập vào ứng dụng | |
| Luồng sự kiện chính (Basic flow) | | Hành động của tác nhân | Hoạt động của hệ thống |
| 1. Truy cập vào ứng dụng và chọn chức năng tạo dự án |  |
|  | 2. Nhận yêu cầu, kiểm tra người dùng có quyền được tạo dự án hay không.  - Nếu người dùng có quyền Project manager thì chuyển sang màn hình tạo dự án  - Nếu người dùng không có quyền của Project Manager thì chuyển sang luồng sự kiện rẽ nhánh 2.1 |
| 3. Điền các trường thông tin của dự án, nhấn button Xác nhận |  |
|  | 4. Nhận dữ liệu gửi lên và kiểm tra thông tin dữ liệu  - Nếu đúng thì hiện lên thông báo tạo mới dự án thành công  - Nếu thông tin lỗi thì chuyển sang luồng sự kiện rẽ nhánh 4.1 |
| Luồng sự kiện phụ (Alternative flow) | | 2.1 Không có quyền truy cập của Project Manager   * Hệ thống đưa ra thông báo không thể thực hiện tạo mới một dự án   4.1 Nếu dữ liệu gửi lên thiếu hoặc sai   * Hệ thống đưa ra thông báo lỗi * Quay trở lại màn hình tạo mới dự án với các trường dữ liệu đã nhập * Nếu người dùng muốn tiếp tục tạo dự án thì quay về bước 3 | |
| Điều kiện sau | Success | Hiển thị thông báo đã tạo mới dự án thành công. CSDL bị thay đổi | |
| Fail | Hiển thị thông báo tạo mới dự án lỗi. CSDL không bị thay đổi. | |

* 1. **Đặc tả use case “Chỉnh sửa dự án”**

|  |  |  |  |
| --- | --- | --- | --- |
| **Use case “Chỉnh sửa dự án”** | | | |
| Mục đích | | Chỉnh sửa thông tin của dự án khi cần thiết | |
| Tác nhân | | PM | |
| Điều kiện trước | | Đã tạo một dự án | |
| Luồng sự kiện chính (Basic flow) | | Hành động của tác nhân | Hoạt động của hệ thống |
| 1. Truy cập vào ứng dụng bằng quyền truy cập của PM |  |
|  | 2. Hiển thị màn hình trang chủ |
| 3. Chọn dự án cần chỉnh sửa |  |
|  | 4. Hiển thị thông tin của dự án |
| 5. Chọn chỉnh sửa thông tin dự án và nhập vào các thông tin cần chỉnh sửa, nhấn button sửa đổi. |  |
|  | |  | 6. Nhận dữ liệu gửi lên và kiểm tra thông tin dữ liệu  - Nếu đúng thì hiện lên thông báo chỉnh sửa dự án thành công  - Nếu thông tin lỗi thì chuyển sang luồng sự kiện rẽ nhánh 6.1 |
| Luồng sự kiện phụ (Alternative flow) | | 6.1 Nếu dữ liệu gửi lên thiếu hoặc sai   * Hệ thống đưa ra thông báo lỗi * Quay trở lại màn hình chỉnh sửa dự án với các trường dữ liệu đã nhập * Nếu người dùng muốn tiếp tục chỉnh sửa dự án thì quay về bước 3 | |
| Điều kiện sau | Success | Hiển thị thông báo đã chỉnh sửa dự án thành công. CSDL bị thay đổi | |
| Fail | Hiển thị thông báo chỉnh sửa dự án lỗi. CSDL không bị thay đổi. | |

* 1. **Đặc tả use case “Tạo mới giai đoạn”**

|  |  |  |  |
| --- | --- | --- | --- |
| **Use case “Tạo mới giai đoạn”** | | | |
| Mục đích | | Tạo mới một giai đoạn của dự án | |
| Tác nhân | | PM | |
| Điều kiện trước | | Đăng nhập vào ứng dụng, đã tạo một dự án | |
| Luồng sự kiện chính (Basic flow) | | Hành động của tác nhân | Hoạt động của hệ thống |
| 1. Truy cập vào ứng dụng và chọn một dự án cụ thể |  |
|  | 2. Hiển thị thông tin của dự án |
| 3. Chọn chức năng thêm mới một giai đoạn |  |
|  | 4. Nhận dữ gửi lên và kiểm tra dữ liệu người dùng có quyền tạo một giai đoạn mới không  - Nếu người dùng có quyền tạo một giai đoạn, chuyển đến màn hình tạo một giai đoạn  - Nếu người dùng không có quyền tạo một giai đoạn, chuyển đến luồng sự kiện 4.1 |
|  | | 5. Nhập các thông tin của giai đoạn mới |  |
|  | 6. Nhận dữ liệu gửi lên, kiểm tra thông tin của dữ liệu nhận được  - Nếu dữ liệu hệ thống nhận được đúng, hiển thị thông báo tạo mới giai đoạn thành công  - Nếu dữ liệu hệ thống nhận được sai hoặc thiếu các trường bắt buộc, hệ thống thông báo lỗi tạo mới giai đoạn, chuyển sang luồng sự kiện 6.1 |
| Luồng sự kiện phụ (Alternative flow) | | 4.1 Không có quyền tạo một giai đoạn mới   * Hệ thống đưa ra thông báo không thể thực hiện tạo mới một giai đoạn * Đưa ra gợi ý có muốn truy cập bằng tài khoản khác hay không.   6.1 Nếu dữ liệu hệ thống nhận được sai hoặc thiếu các trường thông tin bắt buộc.   * Hệ thống đưa ra thông báo tạo mới giai đoạn bị lỗi * Quay về màn hình tạo mới giai đoạn với các trường dữ liệu đã nhập và báo lỗi các trường thông tin bị sai * Nếu người dùng muốn tiếp tục tạo mới một giai đoạn, quay trở về bước 3 | |
| Điều kiện sau | Success | Hiển thị thông báo đã tạo mới giai đoạn thành công. CSDL bị thay đổi | |
| Fail | Hiển thị thông báo tạo mới giai đoạn lỗi. CSDL không bị thay đổi. | |

* 1. **Đặc tả use case “Chỉnh sửa giai đoạn”**

|  |  |  |  |
| --- | --- | --- | --- |
| **Use case “Chỉnh sửa giai đoạn”** | | | |
| Mục đích | | Chỉnh sửa thông tin của giai đoạn khi cần thiết | |
| Tác nhân | | PM | |
| Điều kiện trước | | Đã tạo một giai đoạn của dự án | |
| Luồng sự kiện chính (Basic flow) | | Hành động của tác nhân | Hoạt động của hệ thống |
| 1. Truy cập vào ứng dụng bằng quyền truy cập của PM |  |
|  | 2. Hiển thị màn hình trang chủ |
| 3. Chọn dự án, chọn giai đoạn cần chỉnh sửa |  |
|  | 4. Hiển thị thông tin giai đoạn của dự án |
| 5. Chọn chỉnh sửa thông tin giai đoạn và nhập vào các thông tin cần chỉnh sửa, nhấn button sửa đổi. |  |
|  | |  | 6. Nhận dữ liệu gửi lên và kiểm tra thông tin dữ liệu  - Nếu đúng thì hiện lên thông báo chỉnh sửa giai đoạn thành công  - Nếu thông tin lỗi thì chuyển sang luồng sự kiện rẽ nhánh 6.1 |
| Luồng sự kiện phụ (Alternative flow) | | 6.1 Nếu dữ liệu gửi lên thiếu hoặc sai   * Hệ thống đưa ra thông báo lỗi * Quay trở lại màn hình chỉnh sửa giai đoạn với các trường dữ liệu đã nhập * Nếu người dùng muốn tiếp tục chỉnh sửa giai đoạn thì quay về bước 3 | |
| Điều kiện sau | Success | Hiển thị thông báo đã chỉnh sửa giai đoạn thành công. CSDL bị thay đổi | |
| Fail | Hiển thị thông báo chỉnh sửa giai đoạn lỗi. CSDL không bị thay đổi. | |

* 1. **Đặc tả use case “Tạo mới công việc”**

|  |  |  |  |
| --- | --- | --- | --- |
| **Use case “Tạo mới công việc”** | | | |
| Mục đích | | Tạo mới một công việc | |
| Tác nhân | | PM, Team lead | |
| Điều kiện trước | | Đăng nhập vào ứng dụng, đã tạo một giai đoạn của dự án | |
| Luồng sự kiện chính (Basic flow) | | Hành động của tác nhân | Hoạt động của hệ thống |
| 1. Chọn chức năng tạo mới công việc |  |
|  | 2. Hiển thị giao diện tạo mới công việc |
| 3. Điền các trường thông tin của công việc |  |
|  | 4. Nhận dữ liệu gửi lên, kiểm tra thông tin của dữ liệu nhận được  - Nếu dữ liệu hệ thống nhận được đúng, hiển thị thông báo tạo mới công việc thành công  - Nếu dữ liệu hệ thống nhận được sai hoặc thiếu các trường bắt buộc, hệ thống thông báo lỗi tạo mới công việc, chuyển sang luồng sự kiện 4.1 |
| Luồng sự kiện phụ (Alternative flow) | | 4.1 Nếu dữ liệu hệ thống nhận được sai hoặc thiếu các trường thông tin bắt buộc.   * Hệ thống đưa ra thông báo tạo mới công việc bị lỗi * Quay về màn hình tạo mới công việc với các trường dữ liệu đã nhập và báo lỗi các trường thông tin bị sai * Nếu người dùng muốn tiếp tục tạo mới một công việc, quay trở về bước 3 | |
| Điều kiện sau | Success | Hiển thị thông báo đã tạo mới công việc thành công. CSDL bị thay đổi | |
| Fail | Hiển thị thông báo tạo mới công việc lỗi. CSDL không bị thay đổi. | |

* 1. **Đặc tả use case “Chỉnh sửa công việc”**

|  |  |  |  |
| --- | --- | --- | --- |
| **Use case “Chỉnh sửa công việc”** | | | |
| Mục đích | | Chỉnh sửa thông tin của công việc khi cần thiết | |
| Tác nhân | | PM, Team lead | |
| Điều kiện trước | | Đã tạo một công việc | |
| Luồng sự kiện chính (Basic flow) | | Hành động của tác nhân | Hoạt động của hệ thống |
| 1. Chọn chức năng chỉnh sửa công việc |  |
|  | 2. Hiển thị màn hình chỉnh sửa công việc |
| 3. Thay đổi các thông tin của công việc, nhấn button xác nhận |  |
|  | 4. Nhận dữ liệu gửi lên và kiểm tra thông tin dữ liệu  - Nếu đúng thì hiện lên thông báo chỉnh sửa công việc thành công  - Nếu thông tin lỗi thì chuyển sang luồng sự kiện rẽ nhánh 4.1 |
| Luồng sự kiện phụ (Alternative flow) | | 4.1 Nếu dữ liệu gửi lên thiếu hoặc sai   * Hệ thống đưa ra thông báo lỗi * Quay trở lại màn hình chỉnh sửa công việc với các trường dữ liệu đã nhập * Nếu người dùng muốn tiếp tục chỉnh sửa công việc thì quay về bước 3 | |
| Điều kiện sau | Success | Hiển thị thông báo đã chỉnh sửa giai đoạn thành công. CSDL bị thay đổi | |
| Fail | Hiển thị thông báo chỉnh sửa giai đoạn lỗi. CSDL không bị thay đổi. | |

* 1. **Đặc tả use case “Khai báo timesheet”**

|  |  |  |  |
| --- | --- | --- | --- |
| **Use case “Khai báo timesheet”** | | | |
| Mục đích | | Ghi chép và xác nhận chấm công thời gian làm việc cho thành viên đội dự án | |
| Tác nhân | | Người dùng (PM, Team lead, member) | |
| Điều kiện trước | | Lựa chọn vào một dự án mà mình đang tham gia | |
| Luồng sự kiện chính (Basic flow) | | Hành động của tác nhân | Hoạt động của hệ thống |
| 1. Chọn chức năng khai báo timesheet |  |
|  | 2. Hiển thị giao diện timesheet |
| 3. Thêm mới, chỉnh sửa thời gian timesheet (nếu cần), nhấn submit |  |
|  | 4. Nhận dữ liệu gửi lên, kiểm tra thông tin của dữ liệu nhận được  - Nếu dữ liệu hệ thống nhận được đúng, hiển thị thông tin vừa chỉnh sửa  - Nếu dữ liệu hệ thống nhận được sai hoặc thiếu các trường bắt buộc, hệ thống thông báo lỗi khai báo timesheet, chuyển sang luồng sự kiện 4.1 |
| Luồng sự kiện phụ (Alternative flow) | | 4.1 Nếu dữ liệu hệ thống nhận được sai hoặc thiếu các trường thông tin bắt buộc.   * Hệ thống đưa ra thông báo khai báo timesheet bị lỗi * Quay về màn hình khai báo timesheet với các trường dữ liệu đã nhập và báo lỗi các trường thông tin bị sai * Nếu người dùng muốn tiếp tục khai báo timesheet, quay trở về bước 3 | |
| Điều kiện sau | Success | Hiển thị thông tin đã chỉnh sửa. CSDL bị thay đổi | |
| Fail | Hiển thị thông tin cũ chưa được chỉnh sửa. CSDL không bị thay đổi. | |

* 1. **Đặc tả use case “Quản trị hệ thống”**

|  |  |  |  |
| --- | --- | --- | --- |
| **Use case “Quản trị hệ thống”** | | | |
| Mục đích | | Admin có thể quản lý, thêm mới, chỉnh sửa các trường dữ liệu và CSDL | |
| Tác nhân | | Admin | |
| Điều kiện trước | | Truy cập ứng dụng bằng quyền admin | |
| Luồng sự kiện chính (Basic flow) | | Hành động của tác nhân | Hoạt động của hệ thống |
| 1. Đăng nhập bằng quyền admin |  |
|  | 2. Hiển thị giao diện quản trị hệ thống |
| 3. Thêm mới, chỉnh sửa các bảng, trường dữ liệu trong CSDL |  |
|  | 4. Nhận dữ liệu gửi lên, kiểm tra thông tin của dữ liệu nhận được  - Nếu dữ liệu hệ thống nhận được đúng, hiển thị thông tin vừa chỉnh sửa  - Nếu dữ liệu hệ thống nhận được sai hoặc thiếu các trường bắt buộc, hệ thống thông báo lỗi, chuyển sang luồng sự kiện 4.1 |
| Luồng sự kiện phụ (Alternative flow) | | 4.1 Nếu dữ liệu hệ thống nhận được sai hoặc thiếu các trường thông tin bắt buộc.   * Hệ thống đưa ra thông báo lỗi * Quay về màn hình trước đó với các trường dữ liệu đã nhập và báo lỗi các trường thông tin bị sai * Nếu người dùng muốn tiếp tục thêm mới, chỉnh sửa các bảng, trường dữ liệu trong CSDL, quay trở về bước 3 | |
| Điều kiện sau | Success | Hiển thị thông tin đã chỉnh sửa. CSDL bị thay đổi | |
| Fail | Hiển thị thông tin cũ chưa được chỉnh sửa. CSDL không bị thay đổi. | |

* 1. **Đặc tả use case “Xem tổng quan dự án”**

|  |  |  |  |
| --- | --- | --- | --- |
| **Use case “Xem tổng quan dự án”** | | | |
| Mục đích | | Giúp người quản trị dự án có cái nhìn tổng quan về dự án | |
| Tác nhân | | PM | |
| Điều kiện trước | | Truy cập ứng dụng bằng quyền Project Manager | |
| Luồng sự kiện chính (Basic flow) | | Hành động của tác nhân | Hoạt động của hệ thống |
| 1. Đăng nhập bằng quyền Project Manager, chọn chức năng xem tổng quan dự án |  |
|  | 2. Nhận yêu cầu gửi lên, hiển thị giao diện tổng quan dự án |
| Luồng sự kiện phụ (Alternative flow) | | N/A | |
| Điều kiện sau | Success | Hiển thị giao diện tổng quan dự án | |
| Fail | Không hiển thị giao diện tổng quan dự án | |

* 1. **Đặc tả use case “Xem chi tiết công việc”**

|  |  |  |  |
| --- | --- | --- | --- |
| **Use case “Xem chi tiết công việc”** | | | |
| Mục đích | | Xem chi tiết các thông tin của một công việc | |
| Tác nhân | | Người dùng (PM, Team lead, member) | |
| Điều kiện trước | | Truy cập ứng dụng | |
| Luồng sự kiện chính (Basic flow) | | Hành động của tác nhân | Hoạt động của hệ thống |
| 1. Chọn một công việc cụ thể, lựa chọn chức năng xem chi tiết công việc |  |
|  | 2. Nhận yêu cầu gửi lên, hiển thị giao diện chi tiết công việc |
| Luồng sự kiện phụ (Alternative flow) | | N/A | |
| Điều kiện sau | Success | Hiển thị giao diện chi tiết công việc | |
| Fail | Không hiển thị giao diện chi tiết công việc | |

* 1. **Đặc tả use case “Xem công việc chỉ định cho một người”**

|  |  |  |  |
| --- | --- | --- | --- |
| **Use case “Xem công việc được chỉ định cho một người”** | | | |
| Mục đích | | Xem danh sách các công việc được chỉ định cho một người nào đó | |
| Tác nhân | | member, team lead, PM | |
| Điều kiện trước | | Truy cập ứng dụng | |
| Luồng sự kiện chính (Basic flow) | | Hành động của tác nhân | Hoạt động của hệ thống |
| 1. Chọn chức năng xem danh sách công việc |  |
| 2. Chọn account người cần xem |  |
|  | 3. Nhận yêu cầu gửi lên, hiển thị giao diện danh sách công việc chỉ định cho người dùng |
| Luồng sự kiện phụ (Alternative flow) | | N/A | |
| Điều kiện sau | Success | Hiển thị giao diện danh sách các công việc được chỉ định | |
| Fail | Không hiển thị giao diện danh sách các công việc được chỉ định | |

1. **Biểu đồ lớp**
2. **Biểu đồ tuần tự**

# **CHƯƠNG 3: THIẾT KẾ HỆ THỐNG**

# **CHƯƠNG 4: KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN**

# **Tài liệu tham khảo**

* [1] Định nghĩa về quản lý dự án phần mềm của wikipedia được đăng trên website https://vi.wikipedia.org/wiki/quản\_lý\_dự\_án\_phần\_mềm
* [2] The Standish Group Report CHAOS năm 2014
* [3] E.J. Chikofsky. Changing your endgame strategy. IEEE Software, Nov. 1990
* [4] Tài liệu: Hướng dẫn sử dụng Microsoft Project 2010 của tác giả Dong Nguyen & Team
* [5] Mô hình phát triển phần mềm Agile: <https://itviec.com/blog/agile-la-gi-scrum-la-gi/>