**MỤC LỤC**5

**I. Thông tin chung**5

1.

**Môi trường phát triển ứng dụng dự kiến**5

◦

Liệt kê các công cụ và hệ điều hành dự kiến sử dụng để phát triển phần mềm2. Ví dụ:

▪

Visual Studio [Phiên bản]2

▪

SQL Server [Phiên bản]2

▪

DevExpress [Phiên bản] (nếu có sử dụng các thư viện UI này)2

▪

Windows [Phiên bản]2

2.

**Thông tin về nhóm**25

◦

Liệt kê các thành viên trong nhóm thực hiện đồ án, bao gồm STT, MSSV, Họ và tên, Điện thoại, Email2.

**II. Phát biểu bài toán**5

1.

**Khảo sát hiện trạng**5

◦

1.1. **Mục đích của đề tài**6

▪

Trình bày tầm quan trọng của việc quản lý cộng đồng chia sẻ công thức nấu ăn trong bối cảnh hiện nay7.

▪

Nêu rõ **mục tiêu chính của phần mềm**: tin học hóa các nghiệp vụ quản lý, giảm chi phí, nâng cao hiệu quả tương ứng với các hoạt động quản lý cộng đồng, người dùng và công thức89.

◦

1.2. **Các phần mềm cùng loại đang có**6

▪

Đề cập các ứng dụng, website chia sẻ công thức nấu ăn hiện có trên thị trường. Lưu ý đây là phần mềm ứng dụng, không viết theo đơn đặt hàng cụ thể10.

◦

1.3. **Những hạn chế của những vấn đề đó**6

▪

Phân tích các điểm yếu, thiếu sót của các giải pháp hiện tại hoặc cách quản lý thủ công (sổ sách)7.

◦

1.4. **Hiện trạng tổ chức**9

▪

Mô tả sơ đồ tổ chức (nếu có, ví dụ: các vai trò quản lý, người kiểm duyệt, người dùng) và các bộ phận/vai trò có giao tiếp với bên ngoài.

◦

1.5. **Hiện trạng nghiệp vụ**11

▪

Liệt kê và mô tả các nghiệp vụ chính liên quan đến cộng đồng chia sẻ công thức nấu ăn, bao gồm: tên công việc, biểu mẫu/thông tin liên quan, quy định, người thực hiện, tần suất, thời điểm và cách thức thực hiện11.

◦

1.6. **Hiện trạng tin học**11

▪

Mô tả tổng quan về hệ thống phần cứng, phần mềm, mạng và nguồn nhân lực (người dùng) hiện tại liên quan đến bối cảnh của đề tài.

◦

1.7. **Phương pháp khảo sát**11

▪

Nêu các phương pháp đã sử dụng để thu thập thông tin cho việc khảo sát hiện trạng (phỏng vấn, quan sát, bảng câu hỏi, thu thập tài liệu)11.

2.

**Yêu cầu đặt ra cho hệ thống**68

◦

Xác định các **yêu cầu phần mềm** là các công việc, nghiệp vụ sẽ được hỗ trợ thực hiện trên máy tính với phần mềm12.

◦

**Phân loại yêu cầu phần mềm**1213:

▪

**Yêu cầu nghiệp vụ**: Lưu trữ, Tra cứu, Tính toán, Kết xuất12.

▪

**Yêu cầu hệ thống**: Tính an toàn, Tính bảo mật13.

▪

**Yêu cầu chất lượng**: Tính tiến hóa (dễ nâng cấp, mở rộng), Tính tiện dụng (dễ dùng), Tính hiệu quả (thời gian truy xuất, dung lượng lưu trữ), Tính tương thích13.

◦

2.1. **Quản lý Người dùng**614

▪

2.1.1. Đăng ký tài khoản người dùng14

▪

2.1.2. Sửa thông tin người dùng14

▪

2.1.3. Quản lý công thức đã đăng tải của người dùng

▪

2.1.4. Quản lý công thức yêu thích/đã lưu

▪

2.1.5. Xem lịch sử hoạt động của người dùng

▪

2.1.6. Báo cáo/đóng góp về người dùng khác

◦

2.2. **Quản lý Công thức nấu ăn**1516

▪

2.2.1. Thêm công thức mới

▪

2.2.2. Sửa thông tin công thức

▪

2.2.3. Xóa công thức

▪

2.2.4. Phân loại công thức (theo loại món, nguyên liệu chính, dịp, v.v.)

▪

2.2.5. Quản lý nguyên liệu và định lượng

▪

2.2.6. Quản lý các bước thực hiện

▪

2.2.7. Đánh giá và bình luận về công thức

◦

2.3. **Quản lý Bình luận/Đánh giá**17

▪

2.3.1. Thêm bình luận

▪

2.3.2. Sửa bình luận

▪

2.3.3. Xóa bình luận

▪

2.3.4. Báo cáo bình luận vi phạm

◦

2.4. **Quản lý Danh mục/Phân loại**17

▪

2.4.1. Thêm danh mục mới (ví dụ: loại món ăn, quốc gia, độ khó)

▪

2.4.2. Sửa thông tin danh mục

▪

2.4.3. Xóa danh mục

◦

2.5. **Quản lý Báo cáo/Thống kê**17

▪

2.5.1. Báo cáo thống kê công thức phổ biến nhất (theo lượt xem, lượt thích, đánh giá)

▪

2.5.2. Báo cáo thống kê người dùng tích cực nhất

▪

2.5.3. Báo cáo thống kê số lượng công thức theo danh mục

◦

2.6. **Quản lý Hệ thống (các chức năng chung)**

▪

2.6.1. Đăng nhập

▪

2.6.2. Đăng xuất

▪

2.6.3. Đổi mật khẩu

▪

2.6.4. Quản lý phân quyền người dùng

▪

2.6.5. Quản lý thông báo hệ thống

3.

**Mô tả phân quyền**1819

◦

3.1. **Admin**18

▪

Có toàn quyền quản lý hệ thống, bao gồm người dùng, công thức, bình luận, danh mục và xem báo cáo thống kê19.

◦

3.2. **Người kiểm duyệt (Moderator)**18

▪

Có quyền kiểm duyệt công thức, bình luận, quản lý người dùng (ví dụ: khóa tài khoản) trong phạm vi được chỉ định19.

◦

3.3. **Người dùng (User)**18

▪

Có quyền đăng ký, đăng nhập, quản lý thông tin cá nhân, đăng tải/sửa/xóa công thức của mình, bình luận, đánh giá, quản lý công thức yêu thích19.

◦

3.4. **Khách (Guest)**18

▪

Chỉ có quyền xem công thức và các thông tin công khai khác19.

**III. Mô hình Use Case**20

1.

**Sơ đồ Use-case**2021

◦

Trình bày sơ đồ Use-case tổng thể của hệ thống, minh họa các Actor và các Use-case chính21.

2.

**Danh sách các Actor**20

◦

Liệt kê các Actor chính của hệ thống (ví dụ: Admin, Người kiểm duyệt, Người dùng, Khách) và ý nghĩa của từng Actor21.

3.

**Danh sách các Use-case**20

◦

Liệt kê đầy đủ tất cả các Use-case đã xác định ở mục II.2 và ý nghĩa/ghi chú của chúng21....

4.

**Đặc tả Use-case**20

◦

Đặc tả chi tiết từng Use-case theo mẫu:

▪

**4.X.1 Tóm tắt**: Mô tả ngắn gọn chức năng của Use-case25.

▪

**4.X.2 Dòng sự kiện**:

•

**4.X.2.1 Dòng sự kiện chính**: Mô tả các bước chính khi thực hiện Use-case thành công25.

•

**4.X.2.2 Các dòng sự kiện khác**: Mô tả các tình huống thay thế hoặc ngoại lệ26.

▪

**4.X.3 Các yêu cầu đặc biệt**: Nêu các ràng buộc, quy tắc nghiệp vụ, thông báo lỗi27.

▪

**4.X.4 Trạng thái hệ thống khi bắt đầu thực hiện Use-case**: Điều kiện tiên quyết28.

▪

**4.X.5 Trạng thái hệ thống sau khi thực hiện Use-case**: Kết quả mong muốn28.

▪

**4.X.6 Điểm mở rộng**: Các khả năng mở rộng trong tương lai29.

◦

(Lặp lại cho từng Use-case, ví dụ như **4.1 Đặc tả Use-case “Đăng nhập”**25..., **4.2 Đặc tả Use-case “Đổi mật khẩu”**29..., v.v., áp dụng cho các chức năng của hệ thống chia sẻ công thức nấu ăn).

**IV. Phân tích**36

1.

**Sơ đồ lớp (mức phân tích)**36

◦

Trình bày sơ đồ Class Diagram ở mức phân tích, thể hiện các lớp đối tượng chính của hệ thống (ví dụ: User, Recipe, Ingredient, Step, Comment, Category, Rating) và mối quan hệ giữa chúng37.

◦

Lưu ý về các loại đối tượng: nghiệp vụ, hệ thống, giao diện người dùng38.

2.

**Danh sách các lớp đối tượng và quan hệ**2

◦

Liệt kê các lớp đối tượng đã xác định trong sơ đồ lớp và mô tả các mối quan hệ của chúng2.

3.

**Mô tả chi tiết từng lớp đối tượng**2

◦

Đối với mỗi lớp, mô tả các thuộc tính (dữ liệu) và phương thức (xử lý) của nó38.

4.

**Sơ đồ trạng thái**2

◦

Nếu có các đối tượng có trạng thái thay đổi phức tạp (ví dụ: trạng thái của công thức: chờ duyệt, đã duyệt, bị từ chối), trình bày sơ đồ trạng thái tương ứng2.

**V. Thiết kế dữ liệu**3940

•

Mục tiêu của thiết kế dữ liệu là mô tả cách thức tổ chức lưu trữ dữ liệu của phần mềm bên trong máy tính. Kết quả là Sơ đồ Logic41.

•

Thiết kế cần đảm bảo tính đúng đắn, tính tiến hóa và tính hiệu quả (thời gian truy xuất, dung lượng lưu trữ)41.

1.

**Sơ đồ Logic**41

◦

Bao gồm các bảng dữ liệu và mối quan hệ giữa chúng (1-n, n-n)4243.

◦

Tuân thủ các quy tắc đặt tên bảng (chữ IN HOA) và thuộc tính (chữ IN ký tự đầu mỗi từ, không dấu, không khoảng trắng)44.

2.

**Xác định Khóa chính**45

◦

Nêu rõ các thuộc tính được chọn làm khóa chính cho từng bảng và lý do (đảm bảo tối thiểu, NOT NULL, không trùng lặp, không thay đổi theo thời gian)45.

◦

Đề cập việc sử dụng thuộc tính trừu tượng nếu cần thiết45.

◦

Lưu ý về kiểu dữ liệu cho khóa và khả năng mở rộng46.

3.

**Các kiểu mã hóa**47

◦

Mô tả các kiểu mã hóa có thể được sử dụng (liên tiếp, theo lát, phân đoạn, phân cấp, diễn nghĩa)47.

4.

**Bảng THAMSO**48

◦

Giải thích chức năng của bảng THAMSO (lưu các giá trị quy định không liên quan đến đối tượng dữ liệu khác) và cấu trúc của nó4849.

5.

**Thuật toán thiết kế dữ liệu**50

◦

Mô tả thuật toán thiết kế dữ liệu dựa trên sơ đồ luồng dữ liệu của các yêu cầu phần mềm50.

◦

Tập trung vào:

▪

**Thiết kế dữ liệu với tính đúng đắn**: Dựa trên biểu mẫu và sơ đồ luồng dữ liệu liên quan để xác định các thuộc tính mới, thuộc tính trừu tượng và cập nhật sơ đồ Logic50.

▪

**Thiết kế dữ liệu với tính tiến hóa**: Dựa trên các quy định và sơ đồ luồng dữ liệu về việc thay đổi quy định để xác định các thuộc tính/tham số mới và cập nhật sơ đồ Logic51.

▪

**Thiết kế dữ liệu với yêu cầu hiệu quả**: Xem xét việc thêm thuộc tính tính toán hoặc tách/thêm bảng để cải thiện thời gian truy xuất và dung lượng lưu trữ52.

6.

**Các dạng chuẩn**52

◦

Giới thiệu các dạng chuẩn cơ sở dữ liệu (1NF, 2NF, 3NF, BCNF)52....

7.

**Chuẩn hóa dữ liệu**55

◦

Giải thích khái niệm chuẩn hóa dữ liệu (đưa quan hệ về các dạng chuẩn để loại bỏ dư thừa dữ liệu)55.

◦

Nêu nhược điểm của chuẩn hóa (tăng thời gian truy vấn)55.

**VI. Thiết kế giao diện**3956

•

Nội dung và hình thức trình bày các màn hình giao tiếp của phần mềm, bao gồm cả hệ thống các thao tác người dùng và xử lý tương ứng57.

1.

**Kết quả đạt được**57

◦

1.1. **Sơ đồ liên kết các màn hình**: Trình bày sơ đồ tổng thể các màn hình và quan hệ chuyển điều khiển giữa chúng58.

◦

1.2. **Danh sách các màn hình**: Liệt kê các màn hình chính theo STT, tên, loại và chức năng59.

◦

1.3. **Mô tả chi tiết từng màn hình**:

▪

Mô tả các đối tượng trên màn hình (tên, kiểu, ràng buộc, chức năng)59.

▪

Liệt kê danh sách biến cố và xử lý tương ứng trên màn hình59.

2.

**Phân loại màn hình**60

◦

Xác định các loại màn hình trong hệ thống (Màn hình chính, Màn hình nhập liệu, Màn hình tra cứu, Màn hình thông báo, Báo biểu)60.

3.

**Kiến trúc màn hình**60

◦

Mô tả kiến trúc chung của các màn hình, bao gồm: Thành phần nhập liệu (Text Box, Combo Box, v.v.), Các nút xử lý (thêm, sửa, xóa, tìm kiếm), và Thông tin kết quả (Label, List Box, v.v.)6061.

4.

**Các bước thực hiện**62

◦

**Thiết kế màn hình với tính đúng đắn**: Đảm bảo màn hình hiển thị và xử lý dữ liệu chính xác.

◦

**4.1. Thiết kế màn hình với tính Tiện dụng**: Giao diện quen thuộc, bố trí hợp lý, cung cấp thông tin thêm, cho phép nhập nhiều giá trị đồng thời62.

◦

**4.2. Thiết kế màn hình với tính Hiệu quả**: Chọn control thích hợp, cung cấp giá trị mặc định, hiển thị kết quả trực quan (màu sắc, hình ảnh)63.

5.

**Cách thiết kế các loại màn hình**63

◦

**5.1. Thiết kế màn hình chính**: Phân nhóm công việc (theo chức năng, đối tượng, hoặc nghiệp vụ/quy trình) và lựa chọn hình thức trình bày (thực đơn, biểu tượng, sơ đồ)6364.

◦

**5.2. Thiết kế màn hình Nhập liệu**: Kiểm tra ràng buộc dữ liệu, phát sinh mã tự động, bổ sung nút chuyển điều khiển, hiển thị thông tin tính toán65.

◦

**5.3. Thiết kế màn hình Tra cứu**: Cho phép tra cứu theo nhiều tiêu chuẩn, nút tìm và cập nhật, hỗ trợ xuất báo cáo, xem kết quả đa dạng66.

◦

**5.4. Thiết kế màn hình thông báo**: Thông báo cô đọng, nút chọn rõ ràng, cung cấp phản hồi hệ thống (progress bar)66.

◦

**5.5. Thiết kế Báo biểu**: Thành phần nhập liệu (thời gian), các nút xử lý (Print, Export file), chỉ hiển thị thông tin cần thiết, giữ tiêu đề khi cuộn67.

6.

**Một số nguyên tắc trong thiết kế giao diện**68

◦

Tất cả màn hình phải có tên68.

◦

Số bước để đi đến một màn hình công việc chính phải <=368.

◦

Chỉ trình bày những nội dung thật sự cần thiết, tránh quá nhiều thông tin68.

◦

Chú ý môi trường triển khai ứng dụng68.

◦

Thiết kế phù hợp với đối tượng sử dụng (dễ học, dễ nhớ, quen thuộc)69.

◦

Nhất quán trong toàn bộ hệ thống (cách trình bày, tên nút điều khiển, ý nghĩa biểu tượng)69.

◦

Control: Kích thước cân đối, sắp xếp tối ưu khoảng trống69.

◦

Font chữ: Rõ ràng, cỡ phù hợp, dùng chữ in/hoa đúng trường hợp69.

◦

Màu sắc: Hài hòa, tránh quá nhiều màu, dùng màu nóng để gây chú ý70.

7.

**Công cụ hỗ trợ**70

◦

Liệt kê các công cụ hỗ trợ thiết kế giao diện (MS Word, Paint, Photoshop, Corel) và báo biểu (Crystal Report)70.

**VII. Cài đặt phần mềm**3971

1.

**Phương pháp cài đặt**72

◦

Nêu rõ phương pháp cài đặt được chọn (ví dụ: Cài đặt từ trên xuống và tăng trưởng) và ưu điểm của phương pháp đó (giảm thay đổi, mềm dẻo, phản hồi sớm, thẩm định sớm)7273.

2.

**Môi trường cài đặt**74

◦

Giải thích các vấn đề cần xem xét khi lựa chọn ngôn ngữ lập trình (tính module hóa, cấu trúc dữ liệu, khả năng tích hợp, tính khả chuyển, hỗ trợ hộp thoại)74.

3.

**Phong cách lập trình**74

◦

Mô tả phong cách lập trình của nhóm (cách tổ chức các thành phần như module, class) và cách trình bày (đặt tên biến/hằng/hàm, mô tả đối tượng, chú thích, bố cục thống nhất)7475.

**VIII. Kiểm thử và bảo trì**3976

1.

**Các khái niệm**77

◦

**Kiểm thử phần mềm**: Kiểm tra tính chính xác, an toàn, bảo mật và riêng tư của phần mềm để phát hiện lỗi77.

◦

**Bảo trì phần mềm**: Sửa đổi phần mềm sau khi bàn giao để khắc phục lỗi, nâng cấp tính năng, cải thiện hiệu năng hoặc thích ứng với môi trường thay đổi77.

2.

**Kiểm thử phần mềm**78

◦

2.1. **Các nguyên tắc đảm bảo chất lượng hệ thống**78

▪

**Tính chính xác**: Kiểm tra thông tin nhập/xuất, áp dụng các hình thức kiểm tra (tự động/bằng tay, đầy đủ/đặc trưng, trực tiếp/gián tiếp)7879.

▪

**Tính an toàn**: Khóa các phần cơ sở dữ liệu, sử dụng tệp sao lục (nhật ký, lưu), thực hiện thủ tục phục hồi7880.

▪

**Tính bảo mật**: Nhận diện điểm hở và mối đe dọa, áp dụng các biện pháp bảo mật (vật lý, tài khoản người dùng, mã hóa, truy nhập gián tiếp, tường lửa)7881.

▪

**Tính riêng tư**: Phân loại người dùng, gán quyền truy nhập cụ thể (read, insert, update, delete, expand, drop, index, run), cho phép ủy quyền7882.

◦

2.2. **Bộ kiểm thử**82

▪

Mô tả nội dung của một bộ kiểm thử tốt (tên module/chức năng, thông tin đầu vào, kết quả mong muốn, kết quả thực tế)82.

◦

2.3. **Yêu cầu đối với kiểm thử**83

▪

Đảm bảo kiểm tra hết các trường hợp, lập tài liệu cho quá trình kiểm thử, tiến hành kiểm tra lại lỗi đã sửa83.

◦

2.4. **Tiến trình kiểm thử**84

▪

Kiểm thử đơn vị (unit testing)84.

▪

Kiểm thử tích hợp (integration testing)84.

▪

Kiểm thử hệ thống (system testing) (bao gồm kiểm thử phục hồi, áp lực, thi hành, an ninh)84.

▪

Kiểm thử chấp nhận (acceptance testing) (Alpha, Beta)8485.

◦

2.5. **Các phương pháp và chiến lược**85

▪

Kiểm thử hộp đen (black box testing): dùng để kiểm tra yêu cầu chức năng, không tham khảo cấu trúc bên trong, sử dụng phân hoạch tương đương và đồ thị nhân quả86.

▪

Kiểm thử hộp trắng (white box testing): dựa trên phân tích cấu trúc bên trong, sử dụng đồ thị dòng87.

◦

2.6. **Công cụ kiểm thử tự động**88

▪

Liệt kê các công cụ hỗ trợ kiểm thử tự động (bộ sinh dữ liệu thử nghiệm, bộ xác minh kết quả, bộ kiểm toán mã, mô phỏng module/môi trường, bộ so sánh đầu ra, bộ phân tích dòng dữ liệu)88.

3.

**Bảo trì phần mềm**88

◦

Liệt kê và giải thích 4 loại bảo trì phần mềm: Sửa lại cho đúng (corrective), Thích ứng (adaptive), Hoàn thiện (perfective), Bảo vệ (preventive)89.

**IX. Kết luận và Hướng phát triển**

•

Tóm tắt những kết quả đạt được của đồ án.

•

Đề xuất các hướng phát triển, cải tiến trong tương lai.