TRƯỜNG ĐẠI HỌC KINH TẾ

**KHOA THỐNG KÊ – TIN HỌC**

–––––––––––––––––––––––––––––––



**BÁO CÁO THỰC TẬP NGHỀ NGHIỆP**

**NGÀNH HỆ THỐNG THÔNG TIN QUẢN LÝ**

**CHUYÊN NGÀNH QUẢN TRỊ HỆ THỐNG THÔNG TIN**

**NGHIÊN CỨU VÀ ỨNG DỤNG KIỂM THỬ THỦ CÔNG CHO WEBSITE GURU99 BANK**

Sinh viên thực hiện : Nguyễn Thị Kiều Trinh

Lớp : 45K21.1

Đơn vị thực tập : TMA Solution Bình Định

Cán bộ hướng dẫn : Lê Hoàng Linh

Giảng viên hướng dẫn : Th.S Cao Thị Nhâm

**Đà Nẵng, 8/2022**

**NHẬN XÉT CỦA ĐƠN VỊ THỰC TẬP**

Họ và tên sinh viên: Nguyễn Thị Kiều Trinh

Lớp: 45K21.1 Khoa: Thống kê – Tin học

Trường: Đại học Kinh tế - Đại học Đà Nẵng

Thực tập từ ngày:06/06/ 2022 đến ngày: ........./ .……./ 2022

Tại: TMA Solution Bình Định

Địa chỉ: 12 Đại lộ Khoa học, Thung lũng Sáng tạo, Ghềnh Ráng, thành phố Qui Nhơn, Bình Định

Sau quá trình thực tập tại đơn vị của sinh viên, chúng tôi có một số nhận xét, đánh giá như sau:

**1. Về thái độ, ý thức, đạo đức, kỷ luật**

**2. Kiến thức chuyên môn**

**3. Khả năng hòa nhập và thích nghi với công việc**

**4. Trách nhiệm, sáng tạo trong công việc**

**5. Các nhận xét khác**

**Đánh giá chung:**

**Điểm:**

……….., ngày .......tháng ......năm 2022

**Xác nhận của đơn vị thực tập**

# LỜI CẢM ƠN

# LỜI CAM ĐOAN

Quy định:

Soạn thảo trên trang A4 (trang dọc, lề trái: 3.5cm; trên, phải, dưới: 2.5 cm), Font Times New Roman, canh đều 2 bên, size 13, cách dòng 1.5, cách đoạn trên 6pt, cách đoạn dưới 3pt, hàng đâu tiên lùi vào 1.27 cm; hình và bảng soạn theo caption, chèn trích dẫn chéo (Cross-reference) cho bảng và hình; các danh mục hình, bảng, mục lục làm tự động; đánh số trang như file mẫu (bìa không có số trang, danh mục + mục lục số trang theo i, ii, iii…, nội dung chính theo 1,2,3…)

# MỤC LỤC

[NHẬN XÉT CỦA DOANH NGHIỆP THỰC TẬP ii](#_Toc98339732)

[LỜI CẢM ƠN iii](#_Toc98339733)

[LỜI CAM ĐOAN iv](#_Toc98339734)

[MỤC LỤC vi](#_Toc98339735)

[DANH MỤC HÌNH ẢNH viii](#_Toc98339736)

[DANH MỤC BẢNG BIỂU ix](#_Toc98339737)

[DANH MỤC CÁC TỪ VIẾT TẮT x](#_Toc98339738)

[LỜI MỞ ĐẦU 1](#_Toc98339739)

[CHƯƠNG 1. CƠ SỞ LÝ THUYẾT … 2](#_Toc98339740)

[1.1. Mục 1.1 2](#_Toc98339741)

[1.1.1. Mục 1.1.1 2](#_Toc98339742)

[1.1.2. Mục 1.1.2 2](#_Toc98339743)

[1.2. Mục 1.2 2](#_Toc98339744)

[CHƯƠNG 2. LÝ THUYẾT 3](#_Toc98339745)

[2.1. Mục 2.1 3](#_Toc98339746)

[2.1.1. Mục 2.1.1 3](#_Toc98339747)

[2.1.2. Mục 2.1.2 3](#_Toc98339748)

[2.2. Mục 2.2 3](#_Toc98339749)

[CHƯƠNG 3. TRIỂN KHAI … 4](#_Toc98339750)

[3.1. Mục 3.1 4](#_Toc98339751)

[3.1.1. Mục 3.1.1 4](#_Toc98339752)

[3.1.2. Mục 3.1.2 4](#_Toc98339753)

[3.2. Mục 3.2 4](#_Toc98339754)

[CHƯƠNG 4. KẾT QUẢ 5](#_Toc98339755)

[4.1. Mục 4.1… 5](#_Toc98339756)

[4.2. Mục 4.2… 5](#_Toc98339757)

[KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN 6](#_Toc98339758)

[TÀI LIỆU THAM KHẢO 7](#_Toc98339759)

[PHỤ LỤC 8](#_Toc98339760)

*(Mục lục này chỉ là ví dụ)*

# DANH MỤC HÌNH ẢNH

[Hình 1.1 Ngôn ngữ lập trình Python 2](#_Toc98336120)

[Hình 4.1 Kiến trúc của mô hình 5](#_Toc98336121)

# DANH MỤC BẢNG BIỂU

[Bảng 4.1 Kiến trúc 5](#_Toc74235471)

# DANH MỤC CÁC TỪ VIẾT TẮT

**AI** : Artificial Intelligence

**…**

# LỜI MỞ ĐẦU

1. **Mục tiêu nghiên cứu của đề tài**

* Đề tài này nghiên cứu …

1. **Nhiệm vụ của đề tài**

* Nghiên cứu ...
* ...

1. **Phương pháp nghiên cứu**

* ...

1. **Đối tượng và phạm vi nghiên cứu**

* ...

1. **Kết cấu của đề tài**

Đề tài được tổ chức gồm phần mở đầu, x chương nội dung và phần kết luận...

* Mở đầu
* **Chương 1**:
* **Chương 2**:
* **Chương 3**:
* **Chương 4**:
* Kết luận và hướng phát triển

# CƠ SỞ LÝ THUYẾT

## Tổng quan về kiểm thử phần mềm

### Kiểm thử phần mềm là gì?

Kiểm thử phần mềm là một quá trình thực hiện một chương trình hoặc một ứng dụng với mục đích tìm ra lỗi phần mềm. Được sử dụng để xác định tính đúng đắn, đầy đủ và chất lượng của phần mềm.

### Mục tiêu của kiểm thử phần mềm

* Đảm bảo chất lượng của sản phẩm
* Phòng ngừa và phát hiện lỗi
* Sẵn sàng tích hợp
* Cung cấp thông tin để đưa ra quyết định cho giai đoạn tiếp theo
* Thảo luận về cách viết testcase
* Sự tự tin trong công việc
* Tìm ra lỗi trước khi khách hàng phát hiện ra

### Bảy nguyên tắc của kiểm thử phần mềm

* + - 1. Kiểm thử nhằm chỉ ra lỗi (Testing shows presence of defects)
      2. Kiểm thử toàn diện là không thể (Exhaustive testing is impossible)
      3. Kiểm thử càng sớm càng tốt (Early testing)
      4. Sự tập trung của lỗi (Defect clustering)
      5. Nguyên lý thuốc trừ sâu (Pesticide paradox)
      6. Kiểm thử phụ thuộc vào bối cảnh (Testing is context dependent)
      7. Sai lầm về việc không có lỗi (absence – of – errors fallacy)

### STLC là gì?

Vòng đời kiểm thử phần mềm (STLC) là quá trình kiểm thử được thực hiện theo cách có hệ thống và có kế hoạch. Các hoạt động khác nhau được thực hiện để nâng cao chất lượng của sản phẩm. Mỗi bước đều có tiêu chí đầu vào riêng và có thể phân phối. Được chia thành 6 giai đoạn:

* Phân tích yêu cầu (Requirement analysis)
* Lập kế hoạch kiểm thử (Test planning)
* Phát triển môi trường kiểm thử (Test Case Development)
* Thiết lập môi trường (Enviroment setup)
* Thực hiện kiểm thử (Test Execution)
* Đóng chu trình kiểm thử (Test Cycle Closure)

### Error/Bug/Fault/Failure

* **Error**: Lỗi của con người tạo ra ví dụ như dev code thiếu dấu chấm.
* **Bug**: Dev làm những error đó mà tester thấy được gọi là Bug trong quá trình testing.
* **Fault**: Trong quá trình đưa cho khách hàng mà khách hàng thấy được thì không gọi là Bug mà sẽ gọi là Defect và khách hàng sẽ đánh giá project đó bị Fault, chưa sẵn sàng đưa cho khách hàng.
* **Failure**: kết quả sai lệch với yêu cầu đặc tả và khách hàng sẽ đánh giá sản phẩm thất bại.
* Dev gây ra **Error** 🡪 dẫn đến có **Bug** hoặc **Fault** trong code, tài liệu 🡪 khi thực thi chương trình thì bắt gặp **Failure**.

### Xác minh, xác thực

|  |  |
| --- | --- |
| **Xác minh (Verification)** | **Xác thực (Validation)** |
| * Xác minh là quá trình xác nhận rằng phần mềm đáp ứng được các đặc điểm kỹ thuật của nó. * Trả lời cho câu hỏi: “Sản phẩm có đúng hay không?” | * Xác thực là quá trình xác nhận rằng phần mềm đáp ứng được yêu cầu của người dùng. * Trả lời cho câu hỏi: “Đó có phải là sản phầm phù hợp hay không?” |

*Bảng 1. Phân biệt Xác minh và Xác thực*

### QA, QC

|  |  |
| --- | --- |
| **Đảm bảo chất lượng (QA)** | **Kiểm soát chất lượng (QC)** |
| Đảm bảo chất lượng là tập hợp các hoạt động được lên kế hoạch và có hệ thống để cung cấp sự đảm bảo rằng phần mềm sẽ phù hợp với các yêu cầu được chỉ định và đáp ứng nhu cầu của người dùng. | Kiểm soát chất lượng là quy trình kiểm tra sự hoàn thành của các yêu cầu về chất lượng phần mềm. |
| Ngăn ngừa khiếm khuyết. | Xác định và cải thiện các khiếm khuyết. |
| Là quy trình để tạo phần mềm. | Là quá trình để xác minh phần mềm. |
| Đảm bảo những gì đang làm là đúng điều phải làm. | Đảm bảo kết quả những gì đang làm là những gì mong đợi. |

*Bảng 2. Phân biệt QA và QC*

## Vòng đời phát triển phần mềm

### SDLC là gì?

Vòng đời phát triển phần mềm (SDLC) là một quá trình được ngành công nghiệp phần mềm sử dụng để thiết kế, phát triển và kiểm tra phần mềm chất lượng cao. Mục đích tạo ra một phần mềm chất lượng cao đáp ứng hoặc vượt quá mong đợi của khách hàng, hoàn thành trong thời gian và chi phí ước tính.



Có 6 giai đoạn chính:

* Thu thập và phân tích yêu cầu (Requirement gathering and analysis)
* Thiết kế (Design)
* Thực hiện hoặc mã hóa (Implementation or coding)
* Kiểm thử (Testing)
* Triển khai (Deployment)
* Bảo trì (Maintenance)

### Các mô hình của SDLC

Có nhiều mô hình phát triển phần mềm khác nhau như:

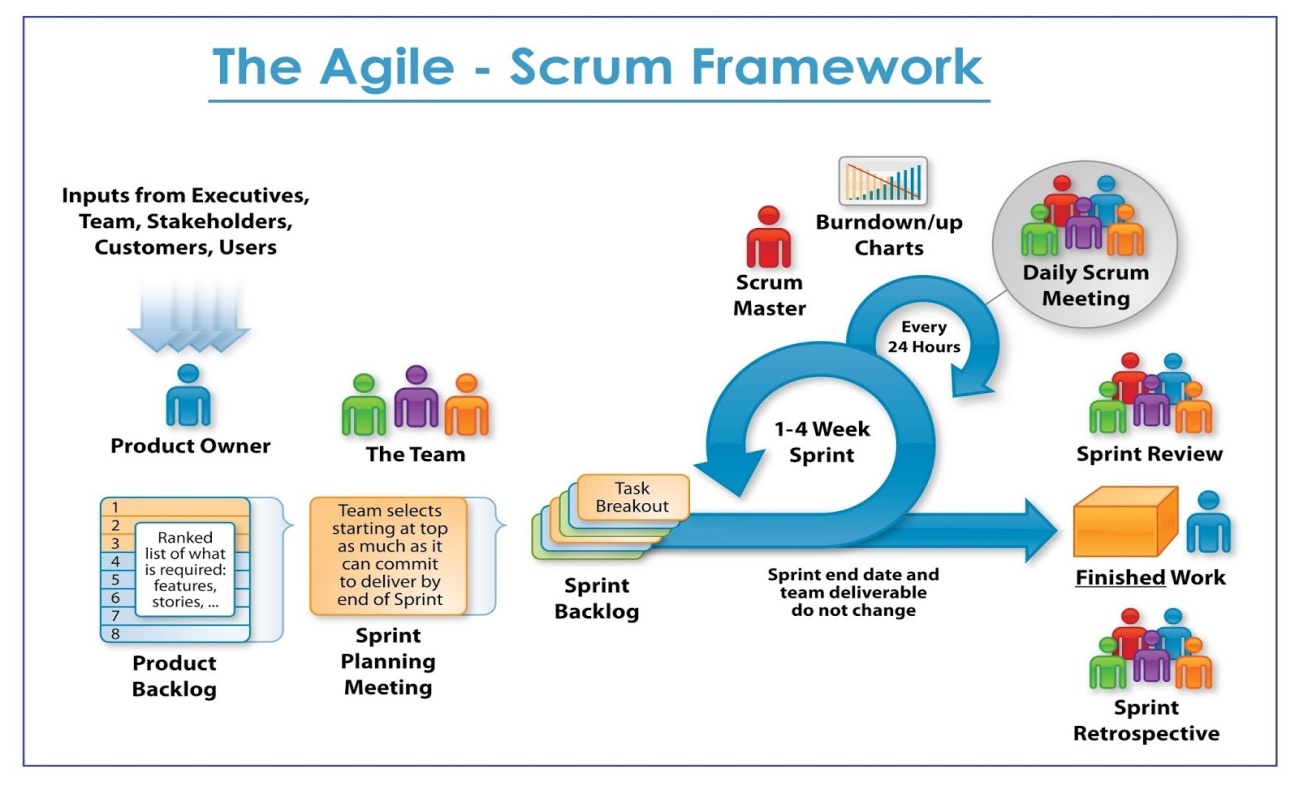
* Waterfall model
* V model
* Incremental model
* RAD model
* Agile model
* Iterative model
* Spiral model

Trong đó mô hình Agile là mô hình phổ biến nhất và dựa trên quy trình phát triển lặp. Mỗi project được chia thành nhiều giai đoạn nhỏ dễ dàng đáp ứng khi có yêu cầu thay đổi từ khách hàng.

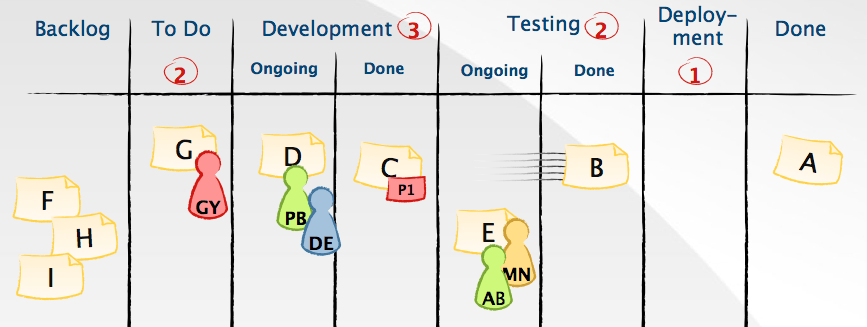
### Phương thức của SDLC

Có 2 phương thức:

* Phương thức Scrum



* Phương thức Kanban



## Các loại và phương pháp kiểm thử phần mềm

### Các loại kiểm thử phần mềm

* **Kiểm thử thủ công (Manual testing)**: là việc kiểm thử một phần mềm hoàn toàn được làm bằng tay bởi tester với mục đích phát hiện ra lỗi phần mềm trong quá trình dev. Tester thực hiện các test case và tạo báo cáo kiểm thử hoàn toàn thủ công mà không có sự trợ giúp của công cụ nào.
* **Kiểm thử tự động (Automation testing)**: là phương pháp kiểm thử tự động, tester sẽ viết các test case sau đó sử dụng các công cụ hỗ trợ để thực hiện kiểm thử, phương pháp này hiệu quả và ít tốn thời gian hơn, giúp chạy các test case lặp lại nhiều lần và task kiểm thử khác khó thực hiện bằng tay như performance và stress tesing.

### Phương pháp kiểm thử phần mềm

* Kiểm thử hộp đen (Black-box testing): là một phương pháp kiểm thử phần mềm mà việc kiểm tra các chức năng của một ứng dụng không cần quan tâm vào cấu trúc nội bộ hoặc hoạt động của nó. Tester không biết gì về kiến trúc hệ thống và không có quyền truy cập vào mã nguồn. Tester tương tác với giao diện người dùng.
* Kiểm thử hộp trắng (White-box testing): là phương pháp kiểm thử phần mềm trong đó các thiết kế, cấu trúc bên trong và việc thực hiện đều được biết đến.

## Các cấp độ kiểm thử phần mềm

### Kiểm thử đơn vị (Unit testing)

Là một loại kiểm thử phần mềm trong đó các đơn vị hay thành phần riêng lẻ của phần mềm được kiểm thử với mục tiêu tách biệt từng phần của chương trình và kiểm tra xem các phần riêng lẻ có hoạt động chính xác hay không.

### Kiểm thử tích hợp (Integration testing)

Là một loại kiểm thử phần mềm trong đó các đơn vị riêng lẻ được kết hợp và kiểm tra như một nhóm với mục đích để tìm ra các lỗi khi tương tác giữa các đơn vị với nhau. Được chia thành 3 loại: bottom up, top down, hybrid.

### Kiểm thử hệ thống (System testing)

Là kiểm thử một hệ thống hay phần mềm tích hợp, hoàn chỉnh với mục tiêu đánh giá sự tuân thủ của hệ thống với các yêu cầu cụ thể. Thực hiện kiểm thử gồm chức năng và phi chức năng.

### Kiểm thử chấp nhận người dùng (User acceptance testing)

Kiểm thử nhằm xác định hệ thống phần mềm có đạt yêu cầu kỹ thuật hay không và được chia làm 2 loại: alpha testing và beta testing.

* Alpha testing: kiểm thử trong môi trường local, xác định tất cả lỗi hoặc vấn đề có thể xảy ra trước khi đưa sản phẩm cho người dùng. Sử dụng kỹ thuật Blackbox và Whitebox.
* Beta testing: là thử nghiệm được thực hiện bởi người dùng thực trong môi trường thực.

## Test case

### Test case là gì?

Test case là một tài liệu trong đó có một tập hợp dữ liệu kiểm thử, điều kiện tiên quyết, kết quả mong đợi và điều kiện hậu kỳ. được phát triển cho một kịch bản kiểm thử cụ thể để xác minh sự tuân thủ theo một yêu cầu cụ thể. Test Case đóng vai trò là điểm bắt đầu cho quá trình thực thi thử nghiệm và sau khi áp dụng một tập hợp các giá trị đầu vào, ứng dụng có một kết quả cuối cùng và rời khỏi hệ thống tại một số điểm kết thúc hoặc còn được gọi là điều kiện hậu thực thi.

### Thông số trường hợp kiểm thử điển hình (Typical Test Case Parameter)

Bao gồm các thông số:

* Test Case ID
* Test Scenario
* Test Case Decription
* Test Steps
* Prerequisite
* Tast Data
* Expected Result
* Actual Result
* Environment Information
* Comments

### Các loại kỹ thuật thiết kế kiểm thử (Types of test design techniques)

Kỹ thuật thiết kế kiểm thử được chia thành hai loại:

* Kỹ thuật kiểm thử tĩnh (Static testing technique): Là việc kiểm tra từng phần của phần mềm, chủ yếu dựa trên các tài liệu của phần mềm hoặc tự phân tích các cú pháp của code để kiểm tra tính logic mà không cần phải chạy phần mềm một cách trực tiếp. Được chia thành 4 bước:
* Informal Reviews
* Technical Reviews
* Walkthrough
* Inspection
* Kỹ thuật kiểm thử động (Dynamic testing technique): được sử dụng để kiểm tra ứng dụng hoặc phần mềm tại thời điểm thực thi, được thực hiện trong quá trình xác nhận. Được chia thành 3 loại:
* Kỹ thuật dựa trên đặc điểm kỹ thuật (Specification-based):
* Phân tích giá trị biên (BVA)
* Phân vùng tương đương (EP)
* Bảng Decision
* Sơ đồ chuyển đổi trạng thái
* Kiểm tra trường hợp sử dụng
* Kỹ thuật dựa trên cấu trúc (Structure-based)
* Kiểm thử câu lệnh (Statement testing)
* Kiểm thử quyết định (Decision testing)
* Kiểm thử điều kiện (Condition testing)
* Kiểm thử đa điều kiện (Multiple condition testing)
* Kỹ thuật dựa trên kinh nghiệm (Experience-based techniques)
* Kiểm thử thăm dò (Exploratory testing)
* Phỏng đoán lỗi (Error guesing)

# LÝ THUYẾT

## Mục 2.1

Nội dung văn bản….

### Mục 2.1.1

### Mục 2.1.2

## Mục 2.2

# TRIỂN KHAI …

## Mục 3.1

### Mục 3.1.1

### Mục 3.1.2

## Mục 3.2

# KẾT QUẢ

## Mục 4.1…

## Mục 4.2…

Kết quả được xây dựng dựa trên ngôn ngữ lập trình thể hiện như (Bảng 4.1).

Bảng . Kiến trúc

|  |  |  |
| --- | --- | --- |
| Lớp | Ý nghĩa | Tham số |
|  |  |  |
|  |  |  |
|  | | |

Hình . Kiến trúc của mô hình

# KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

Đề tài đã thực hiện được …

# TÀI LIỆU THAM KHẢO

* + - 1. Tên tác giả, Tên tác giả… *Tên công trình*. Nơi xuất bản, năm, trang
      2. Tên tác giả, Tên tác giả… *Tên công trình*. Nơi xuất bản, năm, trang

# PHỤ LỤC