|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| TRƯỜNG ĐẠI HỌC BÁCH KHOA HÀ NỘI  **VIỆN ĐIỆN TỬ - VIỄN THÔNG**  logo_128  **BÁO CÁO**  **THỰC TẬP TỐT NGHIỆP**  **CÔNG TY CỔ PHẦN PHẦN MỀM LUVINA**   |  |  | | --- | --- | | **Tên sinh viên** | **Phạm Hồng Đạt** | | **MSSV** | **20172459** |   Hà Nội, 6-2022 |

**LỜI NÓI ĐẦU**

Ngoài việc học các kiến thức trên trường thì một phần không thể thiếu của mỗi sinh viên trên giảng đường là những quảng thời gian thực tập. Thực tập đem lại cho sinh viên không chỉ kiến thức về chuyên môn mà qua đó sinh viên còn được rèn luyện về kĩ năng mềm như kĩ năng giao tiếp, kĩ năng làm việc nhóm… Hiểu rõ được tầm quan trọng đó nên Viện Điện Tử Viễn Thông cũng như trường Đại học Bách Khoa Hà Nội đã phối hợp cùng các công ty đã tổ chức kĩ thực tốt nghiệp cho sinh viên.

Thực tập tốt nghiệp được diễn ra vào kì học cuối cùng của sinh viên, đây là khoảng thời gian vô cùng quan trọng của sinh viên khi mà các em chuẩn bị kết thúc quảng thời gian 5 năm học đại học và sắp đi làm tại các công ty. Kì thực tập như một bước đệm quan trọng giúp em tự tin trên con đường đầy thử thách phía.

Trong khoảng thời gian thực tập em đã học hỏi được rất nhiều, được làm việc dưới sự hướng dẫn tận tình của các cán bộ và công nhân trong ty. Khoảng thời gian thực tập tại công ty, em được quan sát quá trình làm phần mềm, học các kiến thức cơ bản về lập trình web. Vì thời gian thực tập là có hạn và kiến thức còn hạn chế nên bản báo cáo này không thể tránh khỏi những sai sót nên em mong thầy cô góp ý để bản báo cáo này được hoàn thiện hơn. Em xin chân thành cảm ơn ban giám đốc, cán bộ kỹ thuật, nhân viên các anh chị nhân viên đã giúp đỡ và tạo điều kiện cho em một kỳ thực tập thành công.

Quá trình thực tập thực sự có ý nghĩa đối với mỗi sinh viên như em, đó là khoảng thời gian bổ ích và vui vẻ. Đó cũng là giỏ tri thức làm hành trang cho mỗi sinh viên trên con đường tìm kiếm sự nghiệp sau này.

Và cuối cùng em xin cám ơn cô **Phùng Thị Kiều Hà** đã tận tình giúp đỡ em mỗi khi gặp khó khăn trong kì thực tập vừa qua. Em cũng xin cảm ơn các thầy cô trong Viện Điện Tử Viễn Thông trường Đại học Bách Khoa Hà Nội và công ty cổ phần Luvina đã tạo điều kiện và giúp đỡ em trong quá trình thực tập. Em xin chân thành cảm ơn!

**LỜI CAM ĐOAN**

Em là Phạm Hồng Đạt, mã số sinh viên 20172459, sinh viên lớp ĐTVT 09-K62. Em xin cam đoan toàn bộ nội dung được trình bày trong báo cáo **BÁO CÁO THỰC TẬP TỐT NGHIỆP CÔNG TY CỔ PHẦN PHẦN MỀM LUVINA**là kết quả quá trình thực tập và tổng kết của em. Các dữ liệu được nêu trong đồ án là hoàn toàn trung thực, phản ánh đúng kết quả thực tập. Em xin chịu hoàn toàn trách nhiệm với những nội dung được viết trong báo cáo này.

|  |  |
| --- | --- |
|  |  |

Hà Nội Ngày 14 tháng 6 năm 2022

**Phạm Hồng Đạt**

**MỤC LỤC**

[CHƯƠNG 1. GIỚI THIỆU CHỨC NĂNG, NHIỆM VỤ, CƠ CẤU CỦA CÔNG TY CỔ PHẦN PHẦN MỀM LUVINA 1](#_Toc107182859)

[1.1 Giới thiệu về Luvina 1](#_Toc107182860)

[1.2 Tầm nhìn 3](#_Toc107182861)

[1.3 Sứ mệnh 3](#_Toc107182862)

[CHƯƠNG 2. NỘI DUNG THỰC TẬP 4](#_Toc107182863)

[2.1 Các quy định chung của công ty 4](#_Toc107182864)

[2.1.1 Giao tiếp 4](#_Toc107182865)

[2.1.2 Lưu ý khi họp, báo cáo 4](#_Toc107182866)

[2.1.3 Vấn đề bảo mật 4](#_Toc107182867)

[2.1.4 Quy trình làm việc 4](#_Toc107182868)

[2.1.5 Thái độ, trách nhiệm trong công việc 5](#_Toc107182869)

[2.2 Nội dung chính của quá trình thực tập 5](#_Toc107182870)

[2.2.1 HTML, CSS, JavaScript 5](#_Toc107182871)

[2.2.2 Java core 6](#_Toc107182872)

[2.2.3 Database 8](#_Toc107182873)

[2.2.4 Web security 9](#_Toc107182874)

[2.2.5 Java web 9](#_Toc107182875)

[CHƯƠNG 3. ỨNG DỤNG CÁC KIẾN THỨC TRONG KÌ THỰC TẬP VÀO ĐỒ ÁN TỐT NGHIỆP 10](#_Toc107182876)

[3.1 Thực trạng và giải pháp 10](#_Toc107182877)

[3.2 Thiết kế hệ thống 12](#_Toc107182878)

[3.3 Các yêu cầu cần đạt được khi code sau quá trình thực tập 13](#_Toc107182879)

[KẾT LUẬN 14](#_Toc107182880)

# CHƯƠNG 1. GIỚI THIỆU CHỨC NĂNG, NHIỆM VỤ, CƠ CẤU CỦA CÔNG TY CỔ PHẦN PHẦN MỀM LUVINA

## Giới thiệu về Luvina

Công ty cổ phần phần mềm Luvina chuyên thực hiện thiết kế, phát triển, bảo trì và vận hành các sản phẩm công nghệ thông tin chuyên cho thị trường Nhật Bản. Công ty đã hợp tác và là đối tác lâu dài với nhiều khách hàng trên nhiều lĩnh vực khác nhau, bao gồm cả các công ty lớn của Nhật. Công ty hoạt động chủ yếu như 1 công ty outsource

Sau 17 năm phát triển Luvina đã đạt được những thành tựu nhất định:

* Danh hiệu TokouDaihatsu Venture (01/2005)
* ISO 9001 (07/2006)
* ISO / IEC 27001 (02/2011)
* CMMI 3 (12/2012)
* Giải thưởng Kuramae Venture (11/2014)

Quá trình phát triển của công ty đi kèm với việc quy mô của công ty không ngững mở rộng, đến nay đã được mở rộng với gần 800 nhân viên với 40% nhân sự có năng lực tiếng Nhật. Công ty hiện đang có 3 cở sở:

* 1 trụ sở chính nằm tại Hà Nội nơi chủ yếu nhân lực của công ty, lĩnh vực hoạt động là phát triển phần mềm, bảo trì và vận hành hệ thống phần mềm.



Hình 1.1 Trụ sở chính tại Cầu Giấy, Hà Nội

* 1 pháp nhân tại Nhật Bản, lĩnh vực hoạt động là cung cấp nhân lực - phát triển phần mềm - bảo trì và vận hành hệ thống cho thị trường Nhật Bản



Hình 1.2 Trụ sở tại Nishi - Kanagawa Science Park

* 1 văn phòng đại diện tại đã nẵng



Hình 1.3 Văn phòng đại diện tại số 2 Quang Trung, quận Hải Châu, Đà Nẵng

Có thể thấy không chỉ ở trong nước mà Luvina đang vươn mình mạnh mẽ ra thế giới. Dự đoán ngành IT sẽ vẫn còn phát triển rất nhanh và mạnh trong tương lai do sự đáp ứng về nhân lực và hơn nữa nguồn nhân lực chất lượng cao cũng vô cùng vẫn còn thua xa so với nhu cầu thực tế. Con số này vẫn không ngừng tăng trong các năm qua.



Hình 1.4 Số lượng công việc cần tuyển và còn thiếu trong IT ở Việt Nam

Đơn vị mà em được thực tập là Công ty cổ phần phần mềm Luvina tại Hà Nội. Công ty nằm ở nội thành đường Hoàng Quốc Việt, quận Cầu Giấy Hà Nội. Tuy là nội thành nhưng không gian ở đây khá trong lành, yên tĩnh, giúp em khá thoải mái trong quá trình thực tập.

## Tầm nhìn

Dịch vụ phần mềm của Luvina, được xây dựng trên nền tảng là các giá trị cốt lõi của Luvina lan rộng đến từng ngõ ngách trên thế giới. Khách hàng không chỉ được tận hưởng sự tiện lợi của dịch vụ phần mềm mang lại mà còn cảm nhận được niềm hạnh phúc khi được chia sẻ các giá trị cốt lõi của Luvina. Chúng ta sẽ cùng nhau sống một cuộc sống thịnh vượng cả về vật chất lẫn tinh thần, cùng tạo dựng một thế giới hạnh phúc mà ở đó chúng ta có thể tin cậy và được tin cậy.

## Sứ mệnh

Công ty khát khao được cống hiến cho nền văn minh thế giới bằng những dịch vụ và sản phẩm phần mềm giá trị cao. Công ty muốn phát huy hết trí tuệ và tài năng của những con người trong tổ chức cho sứ mạng cao cả này. Bằng cách tạo ra những giá trị thực được công nhận bởi quốc tế. Bên cạnh đó, công ty muốn tạo ấn tượng mạnh mẽ với thế giới về trình độ kỹ thuật và sự chuyên nghiệp của người Việt trong dịch vụ phần mềm, ngày càng nâng cao vị thế của Việt Nam trên thế giới.

# NỘI DUNG THỰC TẬP

## Các quy định chung của công ty

Cũng giống như các công ty khác, Luvina cũng có những quy tắc, quy định chung của công ty và các nhân viên nói chung hay các lập trình viên nói riêng cần phải tuân thủ. Quan điểm của Luvina là 1 lập trình viên giỏi không chỉ có chuyên môn giỏi mà còn phải giao tiếp, làm việc nhóm tốt và hơn nữa là phải có trách nhiệm trong công việc. Chính vì vậy mà ngay từ những ngày đầu của quá trình thực tập, em đã được training rất nhiều, sau đây em xin tổng hợp lại thành các ý chính như sau:

### 2.1.1 Giao tiếp

Em được ôn lại các quy tắc viết, nói trong Tiếng Việt, điều này vốn không phải thế mạnh của 1 sinh viên kĩ thuật. Việc ôn luyện lại khả năng nói giúp em làm việc nhóm với động nghiệp 1 cách hiệu quả hay quan trọng hơn là trao đổi với cấp trên 1 cách trôi trẩy hơn. Ngoài ra không thể thiếu đến kĩ năng viết cũng rất quan trọng trong những lúc viết email hay báo cáo công việc.

### 2.1.2 Lưu ý khi họp, báo cáo

Trong khi họp luôn phải mang theo sổ bút để note lại các ý chính liên quan đến mình. Khi gặp cấp trên để báo cáo, nói chuyện thì cần phải chuẩn bị các ý chính trước ra sổ để báo không trong quá trình giao tiếp không bị thiếu ý.

### 2.1.3 Vấn đề bảo mật

* Tất cả những tài liệu, source code của dự án đều phải bảo mật, không được phát tán lên mạng internet
* Không được mang các thiết bị có thể lưu trữ dữ liệu như máy tính cá nhân, usb,… đến công ty, không được kết nối điện thoại với máy tính của công ty
* Không được mang tài sản, tài liệu của công ty về nhà

### 2.1.4 Quy trình làm việc

Sau quá trình thực tập em đã hiểu rất kĩ về quy trình để hoàn thiện 1 sản phẩm phầm mềm của Luvina. Em được học các xử lý khi gặp phải những lỗi và cần phải hỏi các anh chị như thế nào. Ngoài ra em còn được học cách phòng tránh trễ deadline bằng cách phải báo cáo công việc thường xuyên cho quản lý của mình.

### 2.1.5 Thái độ, trách nhiệm trong công việc

* Quyết không trễ deadline, đây là phương châm của Luvina. Đối với em thì để không trễ deadline thì không được chủ quan, báo cáo khó khăn ngay khi gặp phải mà không thể giải quyết để tránh lãng phí thời gian, trên hết là niềm đam mê trong công việc.
* Không được đi muộn về sớm. Đi sớm 15 phút cũng là đúng giờ, đi sớm còn giúp chúng ta có được thiện cảm với đồng nghiệp, với khách hàng mà còn giúp chúng ta có thời gian chuẩn bị cho 1 ngày dài làm việc. Em cũng được các anh chị dạy là không được về khi các đồng nghiệp còn đang gặp phải khó khăn. Mỗi khi xong việc của mình thì không được ngồi chờ hết giờ xong đi về mà phải báo cho cấp trên để đi hỗ trợ mọi người. Điều này vừa giúp cấp trên có đánh giá tốt về mình, vừa giúp bản thân đi nhanh hơn, tăng kĩ năng trong công việc, vì dạy người khác sẽ giúp mình hiểu vấn đề nhất.

## Nội dung chính của quá trình thực tập

### 2.2.1 HTML, CSS, JavaScript

* HTML (viết tắt của từ HyperText Markup Language, hay là "Ngôn ngữ Đánh dấu Siêu văn bản") là một ngôn ngữ đánh dấu được thiết kế ra để tạo nên các trang web trên World Wide Web. Với HTML ta hoàn toàn có thể tạo lên các trang web tĩnh rất đơn giản
* CSS là chữ viết tắt của Cascading Style Sheets, nó là một ngôn ngữ được sử dụng để tìm và định dạng lại các phần tử được tạo ra bởi. Nói ngắn gọn hơn là ngôn ngữ tạo phong cách cho trang web. Bạn có thể hiểu đơn giản rằng, nếu HTML đóng vai trò định dạng các phần tử trên website như việc tạo ra các đoạn văn bản, các tiêu đề, bảng,…thì CSS sẽ giúp chúng ta có thể thêm style vào các phần tử HTML đó như đổi bố cục, màu sắc trang, đổi màu chữ, font chữ, thay đổi cấu trúc…
* JavaScript là ngôn ngữ lập trình website phổ biến hiện nay, nó được tích hợp và nhúng vào HTML giúp website trở nên sống động hơn.

Có thể thấy làm chủ được các kiến thức này sẽ là nên tảng tốt cho 1 lập trình viên lập trình web

### 2.2.2 Java core

Java là một một ngôn ngữ lập trình hiện đại, bậc cao, hướng đối tượng, bảo mật và mạnh mẽ. Java được biết đến với tính năng viết 1 lần dùng ở mọi nơi vì nó có thể hoạc động ở bất cứ môi trường phần cứng hoặc phần mềm nhờ môi trường runtime riêng cho mình (JRE) và API, Java được gọi là Platform.

Java đã phát triển rất nhiều, có thể nói để học, hiểu hết được tất cả mọi kiến thức về Java gần như là không thể, ta chỉ nên học sâu về 1 số kiến thức phục vụ cho 1 công việc nhất định. Tuy nhiên để làm được điều đó thì đều cần phải kĩ các kiến thức cơ bản nhất của Java (Java core). Có thể điểm qua các kiến thức mà em đã nắm vững:

* Các kiến thức cơ bản
* 2 loại biến trong java: nguyên thuỷ (primitive) và tham chiếu (reference). Khi so sánh 2 dữ liệu nguyên thuỷ thì Java sẽ so sánh 2 giá trị trực tiếp, khi so sánh 2 giá trị nguyên thuỷ thì Java sẽ so sánh 2 địa chỉ của biến.
* Auto boxing khi khai báo 1 kiểu dữ liệu tham chiếu.
* Auto unboxing khi so sánh 1 kiểu dữ kiệu nguyên thuỷ với 1 kiểu dữ liệu tham chiếu
* Lớp String trong java. Lớp không chỉ dùng để lưu chữ 1 chuỗi mà còn cung cấp các phương thức giúp cho thao tác với 1 chuỗi trở nên rất dễ dàng
* Lớp StringTokenizer giúp việc phân tách 1 chuỗi trở nên dễ dàng hơn
* Calendar và Date trong java
* Regular expression cung cấp các biểu thức chính quy để kiểm tra 1 chuỗi có đúng với định dạng mong muốn hay không. Điều này rất hữu ích với các hệ thống cần validate dữ liệu đầu vào của người dùng
* Mảng trong java
* Hướng đối tượng trong Java
* Tính đóng gói và bảo mật bên trong. Tức là các thuộc tính của các đối tượng được bảo vệ, không thể thay đổi từ bên ngoài 1 cách trực tiếp, việc này đảm bảo sự toàn vẹn, bảo mật của đối tượng trong Java. tính đóng gói được thể hiện thông qua phạm vi truy cập (access modifier)
* Tính kế thừa trong Java là khả năng cho phép ta xây dựng một lớp mới dựa trên các định nghĩa của một lớp đã có. Lớp đã có gọi là lớp Cha, lớp mới phát sinh gọi là lớp Con và đương nhiên kế thừa tất cả các thành phần của lớp Cha, có thể chia sẻ hay mở rộng các đặc tính sẵn có mà không phải tiến hành định nghĩa lại.
* Tính đa hình:
* Nạp chồng (Overloading): Đây là khả năng cho phép một lớp có nhiều thuộc tính, phương thức cùng tên nhưng với các tham số khác nhau về loại cũng như về số lượng. Khi được gọi, dựa vào tham số truyền vào, phương thức tương ứng sẽ được thực hiện
* Ghi đè (Overriding): là hai phương thức cùng tên, cùng tham số, cùng kiểu trả về nhưng lớp con viết lại và dùng theo cách của nó, và xuất hiện ở lớp cha và tiếp tục xuất hiện ở lớp con. Khi dùng override, lúc thực thi, nếu lớp Con không có phương thức riêng, phương thức của lớp Cha sẽ được gọi, ngược lại nếu có, phương thức của lớp Con được gọi
* Tính trừu tượng là một tiến trình ẩn các chi tiết trình triển khai và chỉ hiển thị tính năng tới người dùng. Tính trừu tượng cho phép bạn loại bỏ tính chất phức tạp của đối tượng bằng cách chỉ đưa ra các thuộc tính và phương thức cần thiết của đối tượng trong lập trình. Trong Java, chúng là sử dụng abstract class và abstract interface để có tính trừu tượng
* Cấu trúc dữ liệu và giải thuật
* Sắp xếp chọn: duyệt từ đầu đến phần tử kề cuối danh sách, duyệt tìm phần tử nhỏ nhất từ vị trí kế phần tử đang duyệt đến hết, sau đó đổi vị trí của phần tử nhỏ nhất đó với phần tử đang duyệt và cứ tiếp tục như vậy
* Sắp xếp chèn: ta có mảng ban đầu gồm phần tử A[0] xem như đã sắp xếp, ta sẽ duyệt từ phần tử 1 đến n – 1, tìm cách chèn những phần tử đó vào vị trí thích hợp trong mảng ban đầu đã được sắp xếp
* Sắp xếp nổi bọp: Duyệt qua danh sách, làm cho các phần tử lớn nhất hoặc nhỏ nhất dịch chuyển về phía cuối danh sách, tiếp tục lại làm phần tử lớn nhất hoặc nhỏ nhất kế đó dịch chuyển về cuối hay chính là làm cho phần tử nhỏ nhất (hoặc lớn nhất) nổi lên, cứ như vậy cho đến hết danh sách
* Queue: được hiểu là hàng đợi và là cấu trúc dữ liệu hoạt động theo nguyên tắc: vào trước ra trước (First in first out - FIFO)
* Stack: là một cấu trúc dữ liệu hoạt động theo nguyên tắc Last In First Out. Hiểu đơn giản là phần tử sẽ được thêm vào cuối stack và khi lấy ra ta cũng sẽ lấy phần tử cuối stack (phần tử được thêm vào gần nhất)
* Hash Table: là một cấu trúc dữ liệu sử dụng hàm băm để ánh xạ từ giá trị xác định, được gọi là khóa (ví dụ như tên của một người), đến giá trị tương ứng (ví dụ như số điện thoại của họ). Hàm băm mang có ưu điểm là độ phúc tạp của thuật toán tìm kiếm là O(1) thay vì O(n) như tìm kiếm trong mảng như bình thường.
* Các kiến thức khác
* Exception: xử lý ngoại lệ trong java
* Lớp Collection trong Java: cung cấp rất nhiều các các tổ chức dữ liệu trong Java, tuỳ từng bài toán mà ta lên chọn sử dụng các collection phù hợp, các yếu tố cần quan tâm:
* Tốc độ truy cập các phần tử
* Tốc độ thêm, sửa, xoá các phần tử
* Đặc tính khi lưu collection (thứ tự, độ ưu tiên…)
* Equal, ToString trong java: mọi Object đầu kế thừa class Object, class này có sẵn 2 phương thức Equal để so sánh 2 Object, ToString để nêu các hiển thị Object ra màn hình console. Vì vậy ta hoàn toàn có thể định nghĩa lại 2 phương này 1 cách từ ý
* Comparable, comparator thường được dùng với lớp Collection để sắp xếp các phần tử trong Collection

### 2.2.3 Database

Em cũng được rèn luyện lại quá trình thiết kế database và các câu lệnh sql nâng cao.

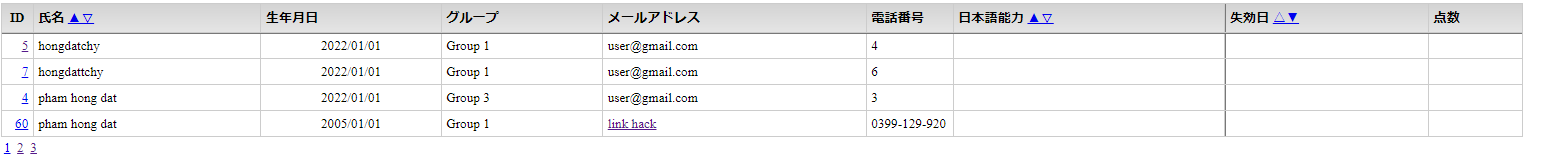
* Các chuẩn trong thiết kế cơ sở dữ liệu: chuẩn NS1, NS2, NS3
* Các quan hệ 1-1, 1-n, n-n
* Thực hành thiết kế Database cho các yêu cầu của dự án
* Các câu lệnh Join khi cần lấy thông tin từ nhiều bảng: Inner join, Left join, Right join
* Các câu lệnh truy vấn lồng nhau, tìm kiếm theo điều kiện ở nhiều bảng

### 2.2.4 Web security

Bảo mật là 1 việc không thể thiếu trong quá trình làm phần mềm thực tế, em đã được các anh chị dạy về các lỗi bảo mật và lập trình viên hay mắc phải có thể dẫn tới việc hệ thống bị hacker tấn công:

#### Phòng chống được lỗi XSS

* Đây là lỗi tấn công dựa vào giao diện, thường được dùng trong các màn hình có hiển thị dữ liệu nhập vào từ người dùng. Để hiểu rõ ta tìm hiểu qua ví dụ sau



###### Hình 2.1 Ví dụ về lỗi XSS

* Đây là giao diện 1 trang hiển thị danh sách người dùng, ở cột ngoài cùng bên phải là hiển thị email mà người dùng đăng kí, nhưng nếu người dùng nhập 1 chuỗi kí tự đặc biệt thì hoàn toàn có thể hiển thị 1 link như hàng thứ 4 trong hình trên. Nếu click đó độc hại và 1 người dùng khác click phải thì rất nguy hiểm.



###### Hình 2.2 Ví dụ sau khi phòng chống lỗi XSS

* Để phòng chống thì ta cần mã hoá các kí tự đặc biệt của dữ liệu trước khi trả về cho trình duyệt. Sau khi xử lý thì ta sẽ đươc kết quả như trên, dữ liệu hiển thị đúng với dữ liệu nhập vào dưới dạng text.

#### Phòng chống được lỗi SQL Injection

* Đây là lỗi rất nguy hiểm, nếu không phòng chống tốt hacker hoàn toàn có thể lấy giữ liệu người dùng hay xoá bỏ toàn bộ database. Lỗi thường bị gây ra khi trang web cần có các mệnh đề truy vấn Where Like hoặc mệnh đề Order By, bản chất là các câu lệnh này đều lấy dữ liệu người dùng nhập vào và hoàn toàn có thể gây ra lỗi.

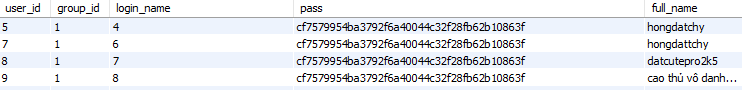


###### Hình 2.3 Ví dụ về lỗi SQL Injection

* Trong ví dụ trên tại ô tìm kiếm nếu ta nhập 1 chuỗi như sau : **';drop table tbl\_user;** thì hoàn toàn bảng tbl\_user sẽ bị xoá vì chức năng tìm kiếm có sử dụng câu SQL có mệnh đề Where Like. Để phòng chống thì cần xử lý các kí tự đặc biệt trước khi thực hiện câu truy vấn. Mệnh đề Order By thường được dùng trong các trường hợp sắp xếp dữ liệu khi lấy ra theo tên 1 cột nào đó, để bảo mật thì trước khi thực hiện câu truy vấn ta phải kiếm tra dữ liệu nhập vào có phải là tên 1 cột nào đó trong Database không.

#### Mã hoá dữ liệu nhậy cảm

* Các dữ liệu nhậy cảm cần được mã hoá, ví dự như thông tin mật khẩu người dùng. Đây là hình ảnh mật khẩu sau khi đã mã hoá, tránh được việc lộ mật khẩu khi những người quản lý server có hành vi muốn chiếm đoạt tài khoản.



###### Hình 2.4 Mã hoá dữ liệu nhậy cảm

### 2.2.5 Java web

Mặc dù hiện nay có rất nhiều những công nghệ mới nhưng công ty vẫn quyết định trainning cho em về kiến thức cơ bản nhất về java web. Việc này không quá xa lạ với cách học, làm việc của em từ trước tới nay, muốn làm được việc tốt thì phải hiểu được bản chất của công nghệ, kĩ thuật. Sau quá trình này thực sự em học được rất nhiều thứ như:

#### Servlet - JSP

Là chương trình Java thực thi trên Web Server và JVM. Lớp trung gian giữa Web Browser hay HTTP Client với các ứng dụng hay CSDL trên HTTP server. Thực hiện bằng cơ chế Web Application. Servlet đón nhận request từ client thực hiện xử lý theo các phương thức/services được implement trong servlet và kết xuất response trả về theo định dạng của trang HTML. Chức năng: đọc dữ liệu HTML form, kết nối CSDL, vận hành cookies, sessions, bảo mật ... Cung cấp các thành phần API để thực hiện tương tác với các ứng dụng Web.

***Nhược điểm:*** Phức tạp; hoạt động chậm

#### Java coding convention

Code Convention (Coding Standards) được hiểu đơn giản là “quy ước viết code”. Nói một cách dễ hiểu, Code Convention là một nhóm các quy ước, quy tắc chung được áp dụng khi viết Code. Đó là các quy tắc số lượng, quy tắc xuống hàng, quy tắc định danh cho class, hằng, biến… Khi xây dựng code theo những quy ước này, các khối code sẽ trở nên “sạch đẹp, dễ nhìn và dễ hiểu hơn

# ỨNG DỤNG CÁC KIẾN THỨC TRONG KÌ THỰC TẬP VÀO ĐỒ ÁN TỐT NGHIỆP

## Thực trạng và giải pháp

Hiện nay, hệ thống giao thông công cộng như xe buýt, xe buýt nhanh, tàu đường sắt trên cao đang thử nghiệm, vẫn chưa đáp ứng được nhu cầu tăng trưởng của thành phố. Bên cạnh đó, với xu hướng năng lượng xanh, giảm phát thải CO2, xe đạp là phương tiện giao thông đang được chú ý tại nhiều thành phố. Đặc biệt, nếu phương tiện xe đạp được cho thuê – chia sẻ như một phương tiện giao thông công cộng, tính linh động, gọn nhẹ, phù hợp với các quãng đường ngắn, kết nối các điểm trong trung tâm sẽ đem lại rất nhiều tiện ích cho người dùng.

A group of people riding bikes on a street

Description automatically generated with low confidence

Hình 3.1: Hình ảnh 1 người dân thành phố Amsterdam đi xe đạp đi làm

Thực tế, với các độ thi lớn (megacity), việc sử dụng phương tiện cá nhân (ô tô con) đi từ các vùng ngoại ô vào vùng rìa và sau đó dùng phương tiện công cộng (bus, metro, tram…) hoặc các phương tiện thân thiện môi trường như xe đạp sẽ giúp giảm lượng xe cá nhân trong vùng trung tâm, vùng lõi đô thị. Như vậy, vừa giúp giảm ách tắc giao thông, vừa tăng thêm tiện ích, khuyến khích sử dụng phương tiện xanh để giảm ô nhiễm môi trường. Đồ án tốt nghiệp này sẽ hướng đến thiết kế, phát triển database và backend, frontend và mobile app để phục vụ nhu cầu thuê xe đạp cho khách hàng.

## Thiết kế hệ thống

### 3.2.1 Thiết kế sơ đồ hệ thống

Diagram

Description automatically generated

Hình 3.3: Sơ đồ khối của hệ thống

#### Khối Device

Khối Device sẽ có trách nhiệm mô phỏng các thiết bị khoá gắn trên xe thật. Khối Device gồm 2 phần chính:

* Website có chức năng mô phỏng các thiết bị thật
* MQTT client cho phép trao đổi thông tin với Server qua giao thức MQTT, giao tiếp với với Website qua giao Rest API và Websocket

#### Khối IoT Messaging server và khối IoT Messaging client

MQTT là một giao thức nhắn tin tiêu chuẩn OASIS cho Internet of Things (IoT). Nó được thiết kế như một phương tiện truyền tải tin nhắn publish / subscriber cực kỳ nhẹ, lý tưởng để kết nối các thiết bị từ xa và cần tiêu thụ ít năng lượng. MQTT gồm 2 thành phần MQTT broker, MQTT client. MQTT broker nhận tất cả các thông báo từ các MQTT client và sau đó định tuyến các thông báo đến các MQTT client đích thích hợp. Vậy nên MQTT là 1 giao thức rất thích hợp cho hệ thống để liên lạc giữa backend với device

#### Khối backend

Backend cần đạp được các yêu cầu:

* Cung cấp các Rest api cho user app và admin web hoạt động
* Cung cấp khả năng thông báo tin nhắn cho ser app
* Tích hợp được MQTT client vào backend

Spring boot là là một dự án phát triển bởi JAV (ngôn ngữ java) trong hệ sinh thái Spring framework. Nó giúp cho các lập trình viên chúng ta đơn giản hóa quá trình lập trình một ứng dụng với Spring, chỉ tập trung vào việc phát triển business cho ứng dụng. Spring boot hoàn toàn là sự lựa chọn tốt để đáp ứng các yêu cầu của backend ở phía trên

#### Khối database

Trong suốt các năm qua mySQL luôn là một sự lựa chọn hàng đầu để dùng làm database cho các ứng dụng web-app.

#### User app

Android là hệ điều hành có mã nguồn mở dựa trên nền tảng Linux do chính Google phát hành được thiết kế dành cho các nhà phát triển thiết bị, các nhà mạng, lập trình viên có thể tiếp cận nhanh chóng, điều chỉnh tự do trong mã nguồn mở đó. Vậy nên em quyết định làm user app trên hệ điều hành android.

#### Admin web

Admin web sẽ cung cấp khả năng quản lý các thông tin trong hệ thống như thông tin user, lịch sự các contract, thông tin các station… Em quyết định chọn react js để làm frontend cho web này vì framework này có những ưu điểm sau:

* React cho phép các nhà phát triển sử dụng các phần riêng lẻ của ứng dụng của họ ở cả phía máy khách và phía máy chủ, điều này cuối cùng giúp tăng tốc độ của quá trình phát triển.
* So với các frontend framework khác, mã React dễ bảo trì hơn và linh hoạt hơn do cấu trúc mô-đun của nó. Do đó, sự linh hoạt này giúp tiết kiệm rất nhiều thời gian và chi phí cho doanh nghiệp
* React JS được thiết kế để cung cấp hiệu suất cao. Cốt lõi của khung cung cấp chương trình DOM ảo và kết xuất phía máy chủ, giúp các ứng dụng phức tạp chạy cực nhanh

#### Khối Google Map Platform

User app cần hiển thị bản đồ cho phép người dùng có thể để tìm kiếm các station hay là hiển thị vị trí khi user thuê xe trên bản đồ, chính vì vậy user app cần kết nốt với Google Map Platform để làm được điều đó

### 3.2.1 Database của hệ thống

Diagram

Description automatically generated

Hình 3.4: Database của hệ thống

**Bảng user**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Tên column** | **Kiểu dữ liệu** | **Not null** | **Tự tăng** | **Mô tả** |
| id | INT | ⚫ | ⚫ | * khoá chính |
| email | VARCHAR(100) | ⚫ |  | * email người dùng * dùng để đăng nhập * không trùng lặp |
| id\_number | VARCHAR(20) |  |  | * số chứng minh thư hoặc căn cước công dân |
| password | VARCHAR(64) | ⚫ |  | * mật khẩu người dùng được mã hoá bằng hàm băm SHA-256 |
| phone | VARCHAR(20) | ⚫ |  | * số điện thoại |
| firstname | VARCHAR(45) | ⚫ |  | * tên người dùng |
| lastname | VARCHAR(45) | ⚫ |  | * họ và tên đệm |
| credit\_card | VARCHAR(45) |  |  | * mã tài khoản ngân hàng |
| birthday | DATE | ⚫ |  | * sinh nhật |
| city\_id | INT | ⚫ |  | * mã tỉnh/thành phố * khoá ngoại tham chiếu đến bảng city |
| district\_id | INT | ⚫ |  | * mã quận/huyện * khoá ngoại tham chiếu đến bảng district |
| ward\_id | INT | ⚫ |  | * mã phường/xã * khoá ngoại tham chiếu đến bảng ward |
| last\_time\_access | DATETIME |  |  | * thời gian cuối cùng truy cập vào tài khoản |
| image | VARCHAR(255) |  |  | * link đường dẫn chưa file ảnh chân dung của người dùng |
| gender | VARCHAR(4) | ⚫ |  | * giới tính người dùng |
| reset\_code | VARCHAR(20) |  |  | * mã code cần dùng khi người dùng reset lại mật khẩu |
| active\_code | VARCHAR(20) |  |  | * mã code dùng khi active tài khoản |
| is\_active | BIT(1) | ⚫ |  | * trạng thái người dùng * nếu là 0 thì là chưa active, 1 là đã active |

**Bảng City**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Tên column** | **Kiểu dữ liệu** | **Not null** | **Tự tăng** | **Mô tả** |
| id | INT | ⚫ |  | * khoá chính |
| name | VARCHAR(45) | ⚫ |  | * tên thành tỉnh/thành phố |

**Bảng District**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Tên column** | **Kiểu dữ liệu** | **Not null** | **Tự tăng** | **Mô tả** |
| id | INT | ⚫ |  | * khoá chính |
| name | VARCHAR(45) | ⚫ |  | * tên quận/huyện |
| city\_id | INT | ⚫ |  | * mã city * khoá ngoại tham chiếu đến bảng city |

**Bảng Ward**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Tên column** | **Kiểu dữ liệu** | **Not null** | **Tự tăng** | **Mô tả** |
| id | INT | ⚫ |  | * khoá chính |
| name | VARCHAR(45) | ⚫ |  | * mã district * khoá ngoại tham chiếu đến bảng district |

**Bảng Admin**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Tên column** | **Kiểu dữ liệu** | **Not null** | **Tự tăng** | **Mô tả** |
| id | INT | ⚫ | ⚫ | * khoá chính |
| name | VARCHAR(100) | ⚫ |  | * email admin * dùng tài khoản đăng nhập * không trùng lặp |
| password | VARCHAR(64) | ⚫ |  | * mật khẩu người dùng được mã hoá bằng hàm băm SHA-256 |

**Bảng Black list**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Tên column** | **Kiểu dữ liệu** | **Not null** | **Tự tăng** | **Mô tả** |
| id | INT | ⚫ | ⚫ | * khoá chính |
| token | VARCHAR(64) | ⚫ |  | * mã token sau khi đã logout |

**Bảng Station**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Tên column** | **Kiểu dữ liệu** | **Not null** | **Tự tăng** | **Mô tả** |
| id | INT | ⚫ | ⚫ | * khoá chính |
| name | VARCHAR(45) | ⚫ |  | * tên station |
| slot\_quantity | INT | ⚫ |  | * số slot để xe tối đa |
| current\_number\_car | INT | ⚫ |  | * số xe hiện tại trong station |
| location | VARCHAR(45) | ⚫ |  | * đại chỉ của station |
| longitude | DOUBLE | ⚫ |  | * kinh độ của station |
| latitude | DOUBLE | ⚫ |  | * vĩ độ của station |

**Bảng Bike**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Tên column** | **Kiểu dữ liệu** | **Not null** | **Tự tăng** | **Mô tả** |
| id | INT | ⚫ | ⚫ | * khoá chính |
| station\_id | INT | ⚫ |  | * mã station * khoá ngoại tham chiếu đến bảng station |
| frame\_number | VARCHAR(45) | ⚫ |  | * số khung của xe |
| product\_year | VARCHAR(45) |  |  | * năm sản xuất của xe |

**Bảng Device**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Tên column** | **Kiểu dữ liệu** | **Not null** | **Tự tăng** | **Mô tả** |
| id | INT | ⚫ | ⚫ | * khoá chính |
| bike\_id | INT | ⚫ |  | * mã bike * khoá ngoại tham chiếu đến bảng bike |
| status\_lock | BIT(1) | ⚫ |  | * trạng thái khoá * 0 là đang khoá, 1 là đang mở khoá |
| longitude | DOUBLE |  |  | * kinh độ của bike |
| latitude | DOUBLE |  |  | * vĩ độ của bike |
| battery | INT |  |  | * phần trăm pin còn lại của device |

**Bảng Contract Bike**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Tên column** | **Kiểu dữ liệu** | **Not null** | **Tự tăng** | **Mô tả** |
| id | INT | ⚫ | ⚫ | * khoá chính |
| bike\_id | INT | ⚫ |  | * mã bike * khoá ngoại tham chiếu đến bảng bike |
| user\_id | INT | ⚫ |  | * mã user * khoá ngoại tham chiếu đến bảng bike |
| payment\_method | VARCHAR(45) | ⚫ |  | * tên phương thức thanh toán |
| start\_time | DATETIME | ⚫ |  | * thời gian bắt đầu của chuyến đi |
| end\_time | DATETIME |  |  | * thời gian kết thúc của chuyến đi |

**Bảng Path**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Tên column** | **Kiểu dữ liệu** | **Not null** | **Tự tăng** | **Mô tả** |
| id | INT | ⚫ | ⚫ | * khoá chính |
| contract\_id | INT | ⚫ |  | * mã contract * khoá ngoại tham chiếu đến bảng contract |
| distance | DOUBLE |  |  | * khoảng các mà xe đã đi được trong 1 lần thuê |
| routes | LONGTEXT | ⚫ |  | * các toạ độ mà xe đã đi trong 1 lần thuê |

## Các yêu cầu cần đạt được khi code sau quá trình thực tập

* Không được dùng “hash code**”**. Các hằng số trong hệ thống phải khai báo ra 1 class constant
* Phòng chống được lỗi XSS
* Phòng chống được lỗi SQL Injection
* Thiết kế sequence diagram trước khi code các tính năng để có thể tối ưu code

# KẾT LUẬN

Sau quá trình hoàn thành kỳ thực tập thiết thực và bổ ích, em đã tích lũy cho mình những kiến thức thực tế cơ bản về lập trình web. Được học hỏi trau dồi những kỹ năng, những công việc mới mẻ mà chưa bao giờ được thực hành.

Em sẽ cố gắng vận dụng tốt những kiến thức đã học được vào công việc trong thời gian sắp tới.