

TRƯỜNG ĐẠI HỌC BÁCH KHOA HÀ NỘI
VIỆN ĐIỆN TỬ - VIỄN THÔNG

*



ĐỒ ÁN
TỐT NGHIỆP ĐẠI HỌC

Đề tài:

Thiết Kế Và Xây Dựng Website Thương Mại Điện Tử

Sinh viên thực hiện: **Lê Bá Tuấn Anh**

Mã số sinh viên: **20140088**

Lớp: **ĐT-TT 10 - K59**

Giảng viên hướng dẫn: **TS. Trần Quang Vinh**

Hà Nội, Ngày 26 tháng 12 năm 2018

TRƯỜNG ĐẠI HỌC BÁCH KHOA HÀ NỘI
VIỆN ĐIỆN TỬ - VIỄN THÔNG

*



ĐỒ ÁN
TỐT NGHIỆP ĐẠI HỌC

Đề tài:

Thiết Kế Và Xây Dựng Website Thương Mại Điện Tử

Sinh viên thực hiện: Lê Bá Tuấn Anh

Mã số sinh viên: 20140088

Lớp: ĐT-TT 10 - K59

Giảng viên hướng dẫn: TS. Trần Quang Vinh

Cán bộ phản biện:

Hà Nội, Ngày 26 tháng 12 năm 2018

**Đánh giá quyển đồ án tốt nghiệp
(Dùng cho giảng viên hướng dẫn)**

Giảng viên đánh giá:

TS. Trần Quang Vinh

Họ và tên Sinh viên:

Lê Bá Tuấn Anh

MSSV:

20140088

Tên đồ án: **Thiết Kế Và Xây Dựng Website Thương Mại Điện Tử.**

Chọn các mức điểm phù hợp cho sinh viên trình bày theo các tiêu chí dưới đây:

Rất kém (1); Kém (2); Đạt (3); Giỏi (4); Xuất sắc (5)

Có sự kết hợp giữa lý thuyết và thực hành (20)						
1	Nêu rõ tính cấp thiết và quan trọng của đề tài, các vấn đề và các giả thuyết (bao gồm mục đích và tính phù hợp) cũng như phạm vi ứng dụng của đồ án.	1	2	3	4	5
2	Cập nhật kết quả nghiên cứu gần đây nhất (quốc tế và trong nước).	1	2	3	4	5
3	Nêu rõ và chi tiết phương pháp nghiên cứu/ giải quyết vấn đề.	1	2	3	4	5
4	Có kết quả mô phỏng/ thực nghiệm và trình bày rõ ràng kết quả đạt.	1	2	3	4	5

Có khả năng phân tích và đánh giá kết quả .(15)

5	Kế hoạch làm việc rõ ràng bao gồm mục tiêu và phương pháp thực hiện dựa trên kết quả nghiên cứu lý thuyết một cách có hệ thống.	1	2	3	4	5
6	Kết quả được trình bày một cách logic và dễ hiểu, tất cả kết quả đều được phân tích và đánh giá thỏa đáng.	1	2	3	4	5
7	Trong phần kết luận, tác giả chỉ rõ sự khác biệt (nếu có) giữa kết quả đạt được và mục tiêu ban đầu đề ra đồng thời cung cấp lập luận để đề xuất hướng giải quyết có thể thực hiện trong tương lai.	1	2	3	4	5

Kỹ năng viết (10)

8	Đồ án trình bày đúng biểu mẫu quy định với cấu trúc các chương logic và đẹp mắt (bảng biểu, hình ảnh rõ ràng, có tiêu đề, được đánh số đúng thứ tự và được giải thích hay đề cập trong đồ án, có căn lề, dấu cách sau chấm, dấu phẩy, v.v), có mở đầu chương và kết luận chương, có liệt kê tài liệu tham khảo và trích dẫn theo quy định.	1	2	3	4	5
---	--	---	---	---	---	---

9	Kỹ năng viết xuất sắc (cấu trúc câu văn chuẩn, văn phong khoa học, lập luận logic và có cơ sở, từ vựng sử dụng phù hợp, v.v.).	1	2	3	4	5
---	--	---	---	---	---	---

Thành tựu nghiên cứu khoa học (5)(chọn 1 trong 3 trường hợp)

10a	Có bài báo khoa học được đăng hoặc chấp nhận đăng/đạt giải SVNC khoa học giải 3 cấp Viện trở lên/các giải thưởng khoa học (quốc tế/trong nước) từ giải 3 trở lên/Có đăng ký bằng phát minh sáng chế.	5
10b	Được báo cáo tại hội đồng cấp Viện trong hội nghị sinh viên nghiên cứu khoa học nhưng không đạt giải từ giải 3 trở lên/Dạt giải khuyến khích trong các kỳ thi quốc gia và quốc tế khác về chuyên ngành như TI contest.	2
10c	Không có thành tích nghiên cứu khoa học.	0
Điểm Tổng		.. /50
Điểm tổng quy đổi về thang 10		

Nhận xét thêm của Thầy/Cô (Giảng viên hướng dẫn nhận xét về thái độ và tinh thần làm việc của sinh viên)

.....

.....

.....

.....

.....

Hà Nội, ngày ...tháng ...năm 2018

Người nhận xét

(Ký và ghi rõ họ tên)

Đánh giá quyển đồ án tốt nghiệp (Dùng cho cán bộ phản biện)

Giảng viên đánh giá:

Họ và tên Sinh viên: Lê Bá Tuấn Anh

MSSV: 20140088

Tên đồ án: **Thiết Kế Và Xây Dựng Website Thương Mại Điện Tử.**

Chọn các mức điểm phù hợp cho sinh viên trình bày theo các tiêu chí dưới đây:

Rất kém (1); Kém (2); Đạt (3); Giỏi (4); Xuất sắc (5)

Có sự kết hợp giữa lý thuyết và thực hành (20)						
1	Nêu rõ tính cấp thiết và quan trọng của đề tài, các vấn đề và các giả thuyết (bao gồm mục đích và tính phù hợp) cũng như phạm vi ứng dụng của đồ án.	1	2	3	4	5
2	Cập nhật kết quả nghiên cứu gần đây nhất (quốc tế và trong nước).	1	2	3	4	5
3	Nêu rõ và chi tiết phương pháp nghiên cứu/ giải quyết vấn đề.	1	2	3	4	5
4	Có kết quả mô phỏng/ thực nghiệm và trình bày rõ ràng kết quả đạt.	1	2	3	4	5

Có khả năng phân tích và đánh giá kết quả (15)						
5	Kế hoạch làm việc rõ ràng bao gồm mục tiêu và phương pháp thực hiện dựa trên kết quả nghiên cứu lý thuyết một cách có hệ thống.	1	2	3	4	5
6	Kết quả được trình bày một cách logic và dễ hiểu, tất cả kết quả đều được phân tích và đánh giá thỏa đáng.	1	2	3	4	5
7	Trong phần kết luận, tác giả chỉ rõ sự khác biệt (nếu có) giữa kết quả đạt được và mục tiêu ban đầu đề ra đồng thời cung cấp lập luận để đề xuất hướng giải quyết có thể thực hiện trong tương lai.	1	2	3	4	5

Kỹ năng viết (10)						
8	Đồ án trình bày đúng biểu mẫu quy định với cấu trúc các chương logic và đẹp mắt (bảng biểu, hình ảnh rõ ràng, có tiêu đề, được đánh số đúng thứ tự và được giải thích hay đề cập trong đồ án, có căn lề, dấu cách sau chấm, dấu phẩy, v.v), có mở đầu chương và kết luận chương, có liệt kê tài liệu tham khảo và trích dẫn theo quy định.	1	2	3	4	5

9	Kỹ năng viết xuất sắc (cấu trúc câu văn chuẩn, văn phong khoa học, lập luận logic và có cơ sở, từ vựng sử dụng phù hợp, v.v.).	1	2	3	4	5
---	--	---	---	---	---	---

Thành tựu nghiên cứu khoa học (5)(chọn 1 trong 3 trường hợp)

10a	Có bài báo khoa học được đăng hoặc chấp nhận đăng/đạt giải SVNC khoa học giải 3 cấp Viện trở lên/các giải thưởng khoa học (quốc tế/trong nước) từ giải 3 trở lên/Có đăng ký bằng phát minh sáng chế.	5
10b	Được báo cáo tại hội đồng cấp Viện trong hội nghị sinh viên nghiên cứu khoa học nhưng không đạt giải từ giải 3 trở lên/Dạt giải khuyến khích trong các kỳ thi quốc gia và quốc tế khác về chuyên ngành như TI contest.	2
10c	Không có thành tích nghiên cứu khoa học.	0
Điểm Tổng		.. /50
Điểm tổng quy đổi về thang 10		

Nhận xét thêm của Thầy/Cô

.....

Hà Nội, ngày ...tháng ...năm 2018

Người nhận xét

(Ký và ghi rõ họ tên)

Lời nói đầu

Trong những năm gần đây, sự phát triển mạnh mẽ của thương mại điện tử trên thế giới đã góp phần làm thay đổi cách thức kinh doanh, giao dịch truyền thống và đem lại những lợi ích to lớn cho xã hội.

Sự ra đời của thương mại điện tử đã đánh dấu sự bắt đầu của một hệ thống mới nhằm phát triển kinh tế, một trong những điều kiện cơ bản và có ý nghĩa quyết định trong việc phát triển thương mại điện tử là việc hoàn thiện các dịch vụ thanh toán điện tử.

Thực tế đã chứng minh thanh toán điện tử là một trong những điều kiện cốt lõi để phát triển thương mại điện tử với vai trò là một khâu không thể tách rời của quy trình giao dịch và còn là biện pháp xác thực việc ký kết hợp đồng giữa người bán và người mua. Thương mại điện tử giúp giảm thấp chi phí bán hàng và chi phí tiếp thị. Bằng Internet/web, một nhân viên bán hàng có thể giao dịch được với rất nhiều khách hàng, danh mục bán hàng điện tử trên các trang web không những phong phú hơn mà còn thường xuyên được cập nhật so với các danh mục điện tử in ấn khuôn khổ giới hạn và luôn luôn lỗi thời.

Qua đó thương mại điện tử tạo điều kiện cho việc thiết lập và củng cố mối quan hệ giữa các nhân tố tham gia vào quá trình thương mại. Thông qua mạng, các đối tượng tham gia có thể giao tiếp trực tiếp và liên tục với nhau nhờ đó sự hợp tác lẫn sự quản lý đều được tiến hành nhanh chóng và liên tục.

Trước yêu cầu đó, em quyết định chọn đề tài “**THIẾT KẾ VÀ XÂY DỰNG WEBSITE THƯƠNG MẠI ĐIỆN TỬ**” để hiểu thêm và đóng góp một phần nhỏ vào việc hoàn thiện và phát triển hệ thống thương mại điện tử ở Việt Nam.

Qua đây, em cũng xin chân thành cảm ơn thầy giáo **TS. TRẦN QUANG VINH** đã trực tiếp định hướng, tạo điều kiện cần thiết và tận tình hướng dẫn

để em có thể hoàn thành đồ án của mình. Em cũng xin chân thành cảm ơn các thành viên trong phòng nghiên cứu Hệ thống mạng và các ứng dụng thông minh **SANSLAB** (Smart Applications and Network System Laboratory) đã nhiệt tình hỗ trợ, giúp đỡ em trong quá trình nghiên cứu và thực hiện đồ án.

Trong quá trình làm đồ án, do kiến thức của em còn nhiều hạn chế và hiểu biết chưa rộng nên đồ án không tránh khỏi thiếu sót. Em rất mong nhận được sự chỉ bảo và nhận xét quý báu của các thầy cô.

Em xin cam đoan các kết quả được trình bày trong đồ án là công trình nghiên cứu của em dưới sự hướng dẫn của giáo viên hướng dẫn. Các kết quả trong đồ án là hoàn toàn trung thực, chưa được công bố trong bất kỳ công trình nào trước đây. Các kết quả được dùng để tham khảo đều được trích dẫn đầy đủ và theo đúng quy định.

Em xin chân thành cảm ơn!

Hà Nội, ngày...tháng...năm 2019

Tác giả

Lê Bá Tuấn Anh

Tóm tắt báo cáo

Thương mại điện tử là toàn bộ chu trình và các hoạt động kinh doanh liên quan đến các tổ chức hay cá nhân tiến hành hoạt động thương mại có sử dụng các phương tiện điện tử và công nghệ xử lý thông tin số hoá, bao gồm cả sản xuất, phân phối, marketing, mua – bán, giao hàng hoá và dịch vụ bằng các phương tiện điện tử.

Xuất phát từ những yêu cầu đó, đồ án này tập trung vào việc xây dựng và phát triển website thương mại điện tử để phục vụ nhu cầu về các hệ thống thương mại điện tử ở Việt Nam. Website thương mại điện tử là 1 website cung cấp cho người sử dụng và các nhà cung cấp, phân phối, quản lý hàng hoá quản lý những hoạt động của 1 cửa hàng hay 1 hệ thống cửa hàng vừa và nhỏ. Đối với khách hàng website cũng cấp những chức năng tìm kiếm, xem thông tin sản phẩm, xem địa chỉ và thông tin người bán hàng, cho phép người mua hàng có thể đặt hàng trực tuyến qua nhiều phương thức: giao dịch qua Ngân Lượng, đặt hàng COD, giao dịch thông qua ATM. Đối với các nhà quản lý tương tác với website theo các cấp độ người dùng, những nhà quản lý có thể thêm sửa xoá sản phẩm của mình theo quyền được phân từ người quản trị, họ cũng có thể quản lý các đơn hàng hay xem doanh thu của cửa hàng mình qua biểu đồ.

Qua đó, đề tài của em sẽ được chia ra thành các mục sau:

- CHƯƠNG 1: Tổng quan đề tài.

Trong chương tổng quan đề tài, em nêu ra những mục chính là: Lý do chọn đề tài, mục tiêu của đề tài, các phương pháp sử dụng trong đồ án và cuối cùng là kết luận.

- CHƯƠNG 2: Cơ sở lý thuyết

Để xây dựng và hệ thống website thương mại điện tử, em trình bày một số

cơ sở lý thuyết làm nền tảng để đáp ứng yêu cầu kỹ thuật đề xuất như là:

- Back-end: ASP.NET Core, Entity Framework Core, Identity Framework Core.
 - Front-end: HTML (HyperText Markup Language), CSS (Cascading Style Sheets), Bootstrap, JS (Java Script), Jquery.
- CHƯƠNG 3: Phân tích, thiết kế hệ thống.

Chương này đưa ra những phân tích yêu cầu và thiết kế chi tiết cho website thương mại điện tử.

- CHƯƠNG 4: Triển khai, kiểm thử và kết quả.

Sau khi đưa ra đưa ra các phân tích thiết kế ở chương 3 em tiến hành cài đặt kiểm thử các chức năng đã đề ra. Các kết quả triển khai kiểm thử phần mềm.

Abstract

E-commerce is the entire cycle and business activities that involve organizations or individuals conducting commercial activities using electronic means and digital information processing technology, including production, distribution, marketing, buying - selling, delivery of goods and services by electronic means.

Due to these requirements, this project focuses on building and developing an e-commerce website to meet the demands of e-commerce systems in Vietnam. An e-commerce website is a website that helps users and suppliers, distributors and goods managers manage the activities of a store or a medium and small store system. For customers, the website also provides with search functions, the information of products, the information of addresses and sellers, allowing buyers to order online via various methods: transaction via Ngan Luong (an online payment website), COD order and transaction via ATM. For managers interacting with the website at levels of user, they can add, edit and delete their products according to the rights assigned by the administrator. They can also manage orders or view their store's sales in charts.

There by, my topic is divided into the following sections:

- CHAPTER 1: Overview of the topic.

In this chapter, I mentioned the main sections: Reason for choosing the topic, the target of the topic, the methods used in the project and the conclusion.

- Chapter 2: Theoretical basis.

In order to build a system of an e-commerce website, I will present several theoretical bases as a foundation to meet the proposed technical requirements such as:

-
- Back-end: ASP.NET Core, Entity Framework Core, Identity Framework Core.
 - Front-end: HTML (HyperText Markup Language), CSS (Cascading Style Sheets), Bootstrap, JS (Java Script), and Jquery.
 - Chapter 3: Analysis and system design. This chapter provides the analysis of requirements and detailed design for the e-commerce website.
 - Chapter 4: Deployment, testing and results. After mentioning the design analysis in chapter 3, I proceeded to perform installation testing of the proposed functions. The results of software test deployment are also mentioned.

Mục lục

Lời nói đầu	v
Tóm tắt báo cáo	vii
Abstract	ix
Mục lục	xi
Danh mục hình vẽ	xiii
Danh mục bảng biểu	xiv
Danh mục các từ viết tắt	xv
Chương 1 Đặt vấn đề	1
1.1 Tổng quan về đề tài	1
1.2 Mục tiêu của đề tài	3
1.3 Các phương pháp sử dụng trong thiết kế	4
1.4 Kết luận	5
Chương 2 Cơ sở lý thuyết	6
2.1 Ngôn ngữ lập trình C#	6
2.2 .NET	7
2.2.1 .NET Framework	10
2.2.2 Mono	11
2.2.3 .NET Core	12
2.2.4 Vòng đời yêu cầu trong ASP.NET	13
2.3 CSHTML	16
2.4 Bootstrap	16
2.5 Jquery Ajax	17
2.5.1 Jquery	17

2.5.2 Ajax	19
2.6 Phần mềm hỗ trợ và phát triển website	21
2.6.1 Visual Studio	21
2.6.2 Visual Studio Code	22
2.6.3 Microsoft Sql Server Management Studio	23
2.7 Kết luận	26
Chương 3 Xây dựng thuật toán đa truy nhập	27
3.1 Phân tích yêu cầu bài toán đa truy nhập	27
3.2 Hệ thống BKRES-LoRa	28
3.2.1 Kiến trúc tổng thể hệ thống	28
3.2.2 Nút mạng cảm biến tích hợp module SX1278	29
3.2.3 Mô hình hoạt động của hệ thống BKRES-LoRa	30
3.3 Xây dựng thuật toán đa truy nhập	31
3.3.1 Thuật toán đa truy nhập của Gateway	31
3.3.2 Thuật toán đa truy nhập của nút	32
3.4 Xây dựng chi tiết và đánh giá	34
3.4.1 Bản tin yêu cầu tham gia mạng	34
3.4.2 Bản tin gửi dữ liệu	36
3.5 Kết luận	37
Chương 4 Kết quả và đánh giá	38
4.1 Các kết quả đạt được	38
4.1.1 Phân tích kết quả các thử nghiệm	38
4.1.2 Hệ thống BKRES-LoRa	40
4.2 Đánh giá thuật toán và định hướng phát triển	40
4.3 Kết luận	42
Kết luận	43
Tài liệu tham khảo	43
Phụ Lục	45

Danh mục hình vẽ

2.1	Quy trình biên dịch và chạy chương trình	9
2.2	Kiến trúc của .NET	10
2.3	Kiến trúc .Net Framework	11
2.4	Tổng quan vòng đời trong ASP.NET nói chung	14
2.5	Vòng đời trong ASP.NET MVC	15
2.6	Vòng đời trong ASP.NET Core	16
3.1	Kiến trúc hệ thống	29
3.2	Nút cảm biến tích hợp chip LoRa SX1278	29
3.3	Mô hình hoạt động của mạng	31
3.4	Luồng trao đổi dữ liệu trong quá trình yêu cầu tham gia mạng . .	35
3.5	Luồng dữ liệu trong hai trường hợp	37
4.1	Quá trình xử lý của Gateway	38
4.2	Kết quả thí nghiệm đánh giá độ mất gói	39
4.3	Kit điều khiển BKRES-LoRa	41
4.4	Hiển thị dữ liệu trên ứng dụng web	42

Danh mục bảng biểu

2.1	Các hàm có sẵn trong Jquery [3]	18
3.1	Bảng thông số cấu hình của các nút trong mạng	30
3.2	Cấu trúc bản tin yêu cầu tham gia mạng	35
3.3	Cấu trúc bản tin dữ liệu	36
4.1	Bảng kết quả độ mất gói trong điều kiện thuận lợi	39
4.2	Bảng quả độ mất gói trong điều kiện thách thức	40

Danh mục các từ viết tắt

Ký hiệu	Tiếng Anh	Tiếng Việt
IoT	Internet of Thing	Internet vạn vật
WSN	Wireless Sensor Network	Mạng cảm biến không dây
LoRa	Long range	Truyền thông LoRa
FSK	Frequency Shift Key	Điều chế số theo tần số tín hiệu
CSS	Chirp Spread Spectrum	
IES	Inter-symbol Interference	Nhiều giữa các ký tự
SF	Spreading factor	Hệ số trải phổ
BW	Bandwidth	Băng thông
CR	Coding Rate	Tốc độ mã hóa
CRC	Cyclic Redundancy Check	Kiểm tra vòng
MAC	Medium Access Control	Kiểm soát truy nhập kênh truyền
LAN	Local Area Network	Mạng cục bộ
FDM	Frequency Division Multiplexing	Ghép kênh phân chia theo tần số
FDMA	Frequency Division Multiple Access	Đa truy nhập phân chia theo tần số
TDMA	Time Division Multiple Access	Đa truy nhập phân chia theo thời gian
CDMA	Code Division Multiple Access	Đa truy nhập phân chia theo mã
SDMA	Space Division Multiple Access	Đa truy nhập phân chia theo không gian

CSMA	Carrier Sense Multiple Access	Đa truy nhập dựa vào cảm nhận kênh truyền
CSMA-CA	CSMA with Collision Avoidance	Đa truy nhập CSMA thêm cơ chế tránh va đập
RTT	Round Trip Time	Trẽ hai chiều
MACA	Multiple Access with Collision Avoidance	Đa truy nhập có thêm cơ chế tránh va đập
RTS	Ready-to-Send	Bản tin thông báo sắp gửi tin
CTS	Clear-to-Send	Bản tin thông báo sẵn sàng nhận tin
SPI	Serial Peripheral Interface	Giao tiếp nối tiếp ngoại vi
USB	Universal Serial Bus	Chuẩn kết nối USB
GSM	Global System for Mobile Communications	Hệ thống toàn cầu dành cho việc liên lạc di động
AES	Advanced Encryption Standard	Tiêu chuẩn mã hóa tiên tiến
TCP-TP	Transmission Control Protocol/Internet Protocol	Giao thức kiểm soát truyền tải dùng cho mạng Internet
ID	Identify	Định danh
IC	Intergrated Circuit	Vi mạch tích hợp
IMEI	International Mobile Equipment Identity	Số nhận dạng thiết bị di động

Chương 1

Đặt vấn đề

Trong chương 1, em nêu ra những mục chính là: Lý do chọn đề tài, mục tiêu của đề tài, các phương pháp sử dụng trong đồ án và cuối cùng là kết luận.

1.1 Tổng quan về đề tài

Thương mại điện tử là một khái niệm mới. Mặc dù ra đời chưa lâu nhưng nó đã nhanh chóng khẳng định được vị thế của mình nhờ sức hấp dẫn cũng như trên đà khá ngoạn ngục. Cùng với sự phát triển chóng mặt của Internet, thương mại điện tử đang có những bước tiến rất nhanh với tốc độ ngày càng cao.

Cuối những năm 1900, thương mại điện tử vẫn còn là một khái niệm khá mới mẻ ở Việt Nam. Nhưng dưới sức lan toả rộng khắp của thương mại điện tử, các công ty Việt Nam cũng đang từng bước làm quen với phương thức kinh doanh hiện đại này. Một số dự án thương mại điện tử ở Việt Nam đang phát triển rất mạnh mẽ như:

- thegioididong.vn, chuỗi cửa hàng thegioididong.com được thành lập từ 2004 chuyên bán lẻ các sản phẩm kỹ thuật số di động bao gồm điện thoại di động, máy tính bảng, laptop và phụ kiện điện tử với gần 1000 siêu thị tại 64 tỉnh thành trên khắp Việt Nam. Thế giới di động đã xây dựng được một dịch vụ khách hàng khác biệt vượt trội với văn hoá “Đặt khách hàng làm trung tâm” trong mọi suy nghĩ và hành động của mình với tất cả các nhân viên của công ty. Với uy tín của mình, thế giới di động hiện nay đứng top 1 bán hàng trực tuyến uy tín chất lượng với các mặt hàng di động, máy

tính xách tay các sản phẩm công nghệ cao.

- lazada.vn, là một website mua sắm trực tuyến được thiết kế chuyên nghiệp, hiện đại. Lazada là trang web được nhiều người sử dụng nhất hiện nay tại Việt Nam. Tuy nhiên công ty chủ quản của Lazada không phải của Việt Nam mà là của công ty Singapore có chi nhánh tại Việt Nam. Lazada là công ty bán hàng trực tuyến lớn nhất tại khu vực Đông Nam Á với rất nhiều chi nhánh tại các nước lớn khu vực như Indonesia, Thailand, Philippines, Malaysia, Singapore. Tuy nhiên Lazada không cung cấp tất cả các hàng hóa dịch vụ mà chủ yếu là tạo ra sàn giao dịch online cho các cửa hàng đăng ký bán hàng trên website, công ty chỉ đảm bảo về giao dịch trực tuyến và quản lý cửa hàng, khách hàng.
- tiki.vn, Tiki là một trong những trang web mua sắm trực tuyến hàng đầu Việt Nam sở hữu hơn 800.000 khách hàng và cung cấp đến 120.000 sản phẩm thuộc 10 ngành hàng khác nhau như: Sách, Làm đẹp – Sức khoẻ, Nhà cửa – Đời sống, Điện thoại – Máy tính bảng, Thiết bị số – Phụ kiện số, Điện gia dụng, Thiết bị văn phòng phẩm, Mẹ và Bé, Đồ chơi – Đồ lưu niệm, Thể thao – Dã ngoại với mức doanh số tăng trưởng gấp ba lần mỗi năm. Tiki.vn đã được trao tặng danh hiệu “website TMĐT được yêu thích năm 2014” do người tiêu dùng bình chọn sau 5 năm nỗ lực hoạt động không ngừng nghỉ. Mạng lưới giao hàng của TiKi phục vụ trên toàn quốc, miễn phí cho mọi đơn hàng từ 250.000đ, riêng tại TPHCM và Hà Nội chỉ từ 150.000đ. Dịch vụ vận chuyển trong 24h giúp khách hàng trải nghiệm mua sắm trực tuyến một cách tiện lợi vừa tiết kiệm được thời gian, công sức mà vẫn bảo đảm được các quyền lợi về bảo hành hay đổi/trả dễ dàng trong vòng 30 ngày.
- sendo.vn, Là trang web mua bán trực tuyến của Tập đoàn FPT nhằm kết nối người mua và người bán trên toàn quốc. Ra đời là một dự án Thương mại Điện tử do Công ty CP Dịch vụ Trực tuyến FPT (FPT Online) xây dựng và phát triển, Sendo.vn chính thức ra mắt người dùng vào tháng 9/2012. Ngày 13/5/2014, Công ty CP Công nghệ Sen Đỏ được thành lập, trực thuộc Tập đoàn FPT, là công ty chủ quản của Sendo.vn.
- robins.vn Zalora là trang web mua sắm thời trang quốc tế được đầu tư vào thị trường Việt Nam khá mạnh trong thời gian gần đây. Với sự uy tín

cao và nguồn vốn đầu tư khá mạnh, Zalora không khó để chiếm lĩnh thị trường thời trang online tại Việt Nam. Zalora có một hệ thống mua hàng trực tuyến nhanh gọn, dễ dàng, thuận tiện và tiết kiệm thời gian cho khách hàng. Ở Zalora bạn có thể mua tất cả món hàng thời trang nào mà bạn cần.

- vatgia.com, Khi truy cập chợ mua bán online www.vatgia.com, khách hàng dễ dàng tìm kiếm thông tin với hàng nghìn gian hàng và sản phẩm về điện tử, công nghiệp, xe cộ, xây dựng – nhà cửa, hay các dịch vụ và giải trí. Gian hàng được trình bày một cách khoa học, kết hợp với nhiều công cụ tìm kiếm tiện lợi, dễ dàng cho người tiêu dùng có thể tìm được sản phẩm như mong muốn, rẻ, chính xác và trong thời gian ngắn nhất. Bên cạnh đó, website thương mại điện tử này còn cung cấp cho người tiêu dùng công cụ bình chọn đánh giá chất lượng dịch vụ của người bán, không gian để nhiều người đóng góp ý kiến, trao đổi thông tin về sản phẩm để tìm được những sản phẩm dịch vụ có giá cả và chất lượng tốt nhất.
- chotot.com, Chợ Tốt ra đời vào năm 2012, là một kênh rao vặt trung gian, kết nối người bán và người mua bằng những giao dịch đơn giản, tiện lợi, nhanh chóng, an toàn. Tại Chợ Tốt, người dùng dễ dàng mua bán, mọi mặt hàng, dù đó là đồ cũ hay đồ mới. Các lĩnh vực như bất động sản, xe cộ, đồ dùng cá nhân, đồ điện tử,...[1]

Trong cuộc cách mạng công nghệ 4.0 các công nghệ thay nhau áp dụng dụng vào thương mại để phục vụ đời sống hàng ngày cho con người, thương mại điện tử giúp cho các nhà quản lý một phần nào trong việc mang sản phẩm của mình đến với người sử dụng, cũng giúp cho người tiêu dùng không mất quá nhiều thời gian để tiêu dùng.

1.2 Mục tiêu của đề tài

Trên thực tế các cửa hàng vẫn bán hàng theo phương thức truyền thống người mua hàng phải đến tận nơi để xem hàng hóa và mua, vào thời kì cách mạng công nghệ 4.0 chúng ta phải đưa công nghệ vào thương mại để xoá bỏ phương thức mua bán truyền thống. Dựa vào các lý do trên em đưa ra yêu cầu đối với website thương mại điện tử như sau:

- Đối với người quản lý:

- Phân quyền theo từng chức năng đối với các cấp người sử dụng website.
- Cho phép người quản trị có tất cả các quyền đối với hệ thống.
- Quản lý sản phẩm theo các thuộc tính của sản phẩm: ảnh, số lượng, giá bán, màu sắc, giá bán sỉ, kích cỡ,...
- Quản lý các danh mục sản phẩm để người quản lý có thể mở rộng mặt hàng.
- Quản lý người vận chuyển hàng.
- Quản lý các đơn hàng.
- Xem doanh thu theo tháng, hay các chuỗi ngày.
- Quản lý các quảng cáo, ảnh giao diện, thông tin khách hàng, địa chỉ liên hệ với cửa hàng.
- Quản lý bài đăng cho website.

- Đối với người tiêu dùng:

- Giao diện người dùng thân thiện với người dùng.
- Hiển thị các danh mục sản phẩm để người dùng dễ chọn lựa.
- Tìm kiếm theo gợi ý theo tên sản phẩm có trong hệ thống.
- Hiển thị thông tin sản phẩm rõ ràng và đầy đủ.
- Giao diện thanh toán dễ dàng cho người mua hàng.
- Đăng ký, đăng nhập cho người dùng để lưu thông tin vào đơn hàng để người quản lý dễ dàng giao hàng đến người tiêu dùng.
- Chức năng quên mật khẩu để người dùng có thể lấy lại tài khoản để xem những đơn hàng mà người dùng đã mua ở website.

1.3 Các phương pháp sử dụng trong thiết kế

Các phương pháp mà em sử dụng gồm có:

- Tham khảo tài liệu: tham khảo tài liệu từ sách báo về điện tử, từ mạng internet, kế thừa và phát triển các kết quả nghiên cứu đã có.

- Quan sát, học hỏi: thảo luận cùng thầy và các bạn để đưa ra được hướng đi tốt nhất, đạt hiệu quả cao,
- Thực hành và sửa lỗi: tiến hành viết code và sửa lỗi để đạt được kết quả tối ưu nhất,

1.4 Kết luận

Mục tiêu của đề tài này là xây dựng website thương mại điện tử để thay thế phương thức mua bán truyền thông. Để hoàn thành đề tài cần có những kiến thức về phân tích thiết kế hướng đối tượng, quy trình kiểm thử phần mềm, quy trình thiết kế phần mềm, kiến thức về lập trình hướng đối tượng, và những kiến thức thực tế về thương mại điện tử em tìm hiểu được từ các trang web thực tế. Từ đây em có những kế hoạch cụ thể để hoàn thành mục tiêu đề ra và được em trình bày ở các chương sau.

Chương 2

Cơ sở lý thuyết

Chương này em sẽ trình bày những kiến thức và các công cụ cần có để xây dựng một website

2.1 Ngôn ngữ lập trình C#

C# là một ngôn ngữ lập trình hướng đối tượng được phát triển bởi Microsoft, là phần khởi đầu cho kế hoạch .NET của Microsoft. Tên của ngôn ngữ bao gồm ký tự thăng theo Microsoft nhưng theo ECMA là C#, chỉ bao gồm dấu số thường. Microsoft phát triển C# dựa trên C++ và Java. C# được miêu tả là ngôn ngữ có được sự cân bằng giữa C++, Visual Basic, Delphi và Java.

C# được thiết kế chủ yếu bởi Anders Hejlsberg kiến trúc sư phần mềm nổi tiếng với các sản phẩm Turbo Pascal, Delphi, J++, WFC. Tiêu chuẩn ECMA liệt kê các mục tiêu của việc thiết kế ngôn ngữ C#:

- Ngôn ngữ được dự định là một ngôn ngữ lập trình đơn giản, hiện đại, hướng đến nhiều mục đích sử dụng, và là một ngôn ngữ lập trình hướng đối tượng.
- Ngôn ngữ và việc triển khai đáp ứng các nguyên tắc của ngành kỹ thuật phần mềm như kiểm tra chặt chẽ kiểu dữ liệu, kiểm tra giới hạn mảng, phát hiện các trường hợp sử dụng các biến chưa có dữ liệu, và tự động thu gom rác. Tính mạnh mẽ, sự bền bỉ, và năng suất của việc lập trình là rất quan trọng đối với ngôn ngữ này.

- Ngôn ngữ sẽ được sử dụng để phát triển các thành phần của phần mềm theo hướng thích hợp cho việc triển khai trong các môi trường phân tán.
- Khả năng di chuyển (portability) là rất quan trọng, đặc biệt là đối với những lập trình viên đã quen với C và C++.
- Hỗ trợ quốc tế hóa.
- Ngôn ngữ sẽ được thiết kế để phù hợp với việc viết các ứng dụng cho cả hai hệ thống: hosted và nhúng, từ các phần mềm quy mô lớn, đến các phần mềm chỉ có các chức năng đơn giản.
- Mặc dù các ứng dụng C# có tính kinh tế đối với các yêu cầu về bộ nhớ và chế độ xử lý, ngôn ngữ này không cạnh tranh trực tiếp về hiệu năng và kích thước đối với ngôn ngữ C hoặc assembly
- C# được biên dịch ra mã trung gian MSIL sau đó thực thi bởi Common Language Runtime (CLR) [2].

Em chọn ngôn ngữ C# bởi vì:

- Rất phổ biến và được sử dụng bởi hàng triệu lập trình viên trên toàn thế giới.
- Dễ học và sử dụng.
- So với Java thì nó là đối thủ lớn nhất. Chúng ta không so sánh 2 ngôn ngữ nhưng em thích C# vì nó luôn cải tiến.
- Nền tảng .NET cũng luôn phát triển ngày càng hiện đại trong khi Java phát triển chậm.

2.2 .NET

.NET bao gồm 3 thành phần:

- Runtime
- Libraries

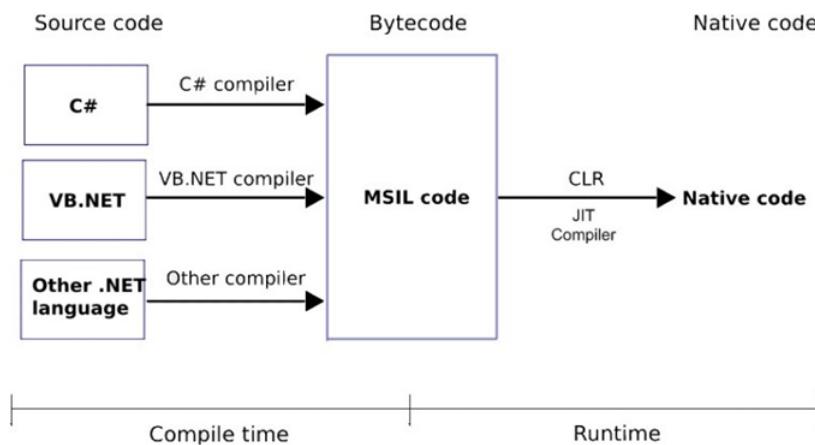
- Toolings

Quy trình biên dịch và chạy chương trình của .NET. Hình 2.1 ta thấy từ source code qua trình biên dịch tương ứng của ngôn ngữ đấy trong hệ sinh thái .NET ví dụ như C# compiler hay VB.NET compiler để sinh ra MSIL code. Trong .Net ngôn ngữ trung gian trong .NET nó khá gần với mã máy những không chứa thông tin cụ thể về CPU, nên ngôn ngữ MSIL giúp cho đoạn code trung gian của chúng ta có thể hoạt động trên nhiều loại CPU (64bit, 32bit), cũng như nhiều loại kiến trúc khác nhau (ARM, Intel...). Trên thực tế một vài ngôn ngữ (Javascript, Python...) không sử dụng đến ngôn ngữ trung gian: Source sẽ được dịch thẳng ra mã máy tại Runtime. Điểm lợi của việc này là quá trình build được đơn giản hóa, tuy nhiên hiệu năng sẽ bị hạn chế. Ngoài việc biên dịch, môi trường hoạt động (Runtime) còn có những công dụng như:

- Tự động quản lý bộ nhớ. Khi làm việc với những ngôn ngữ bậc cao như C# hay Java, chúng ta không cần giải phóng bộ nhớ bằng cách gọi free() như khi làm việc với C/C++. CLR bao gồm một công cụ dọn rác (Garbage collector -GC) sẽ tự động giải phóng những phần bộ nhớ không được sử dụng
- Strong typings: CLR quản lý thông tin về các kiểu dữ liệu đã sử dụng. Điều này giúp cho lập trình viên có thể phân biệt được các định dạng thông tin của từng biến khác nhau (class, structure...)

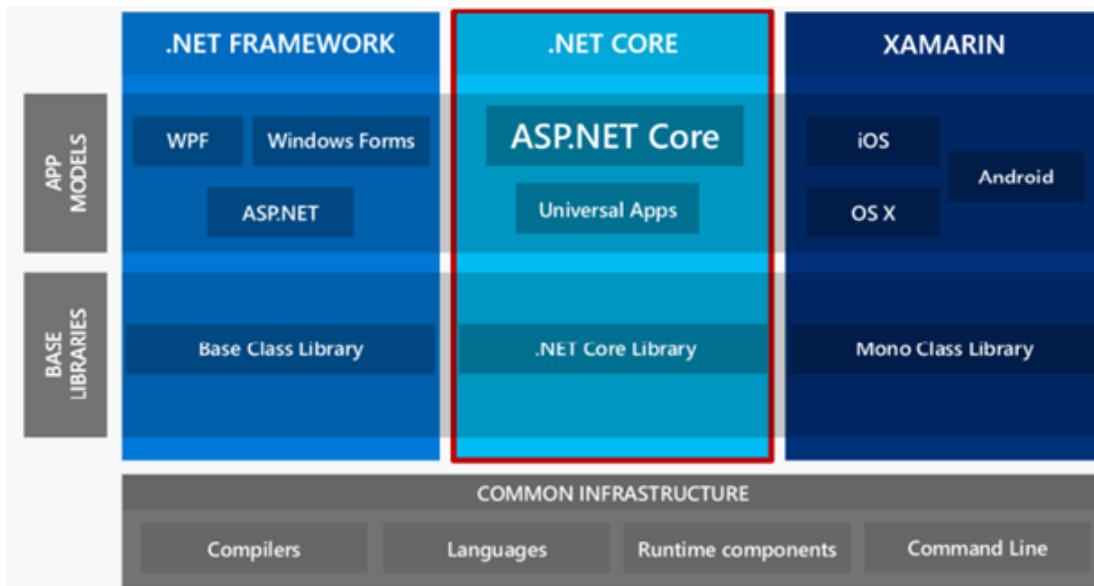
Khi làm việc với .NET, code của chúng ta sẽ tương tác với rất nhiều các class khác nhau. Tất cả những class này được định nghĩa trong hệ thống thư viện cơ bản của .NET được gọi tắt là BCL (Base class libraries). Mã nguồn của BCL, trái với mọi người hay nghĩ, là mã nguồn mở. Chúng ta có thể truy cập mã nguồn này tại sourceof.net. Các công cụ (toolings) của .NET bao gồm compiler và Visual Studio .NET sử dụng hệ thống build của Microsoft gọi là MSBuild. Đối với nền tảng .NET core mới thì chúng ta còn có thêm công cụ dòng lệnh (dotnet cli).

Hình 2.2 chúng ta có thể thấy về cơ bản, .NET Framework, .NET core và Mono là ba phiên bản .NET khác nhau (có nghĩa là mỗi phiên bản có Runtime, Libraries và Toolings riêng).



Hình 2.1: Quy trình biên dịch và chạy chương trình

- .NET Framework được Microsoft đưa ra chính thức từ năm 2002. .NET Framework chỉ hoạt động trên Windows. Những nền tảng ứng dụng như WPF, Winforms, ASP.NET(1-4) hoạt động dựa trên .NET Framework.
- Mono là phiên bản cộng đồng nhằm mang .NET đến những nền tảng ngoài Windows. Mono được phát triển chủ yếu nhằm xây dựng những ứng dụng với giao diện người dùng và được sử dụng rất rộng rãi: Unity Game, Xamarin...
- Cho đến năm 2013, Microsoft định hướng đi đa nền tảng và phát triển .NET core. .NET core hiện được sử dụng trong các ứng dụng Universal Windows platform và ASP.NET Core.
- Tuyệt đối không nên dùng Mono để vận hành web server. Bộ máy dọn rác của Mono không được thiết kế để hoạt động với webserver và sẽ gây ra quá tải nhanh chóng
- Nên lựa chọn .NET Framework hay .NET Core cho các web server .NET Core chạy được đa nền tảng và có hiệu năng cao hơn. Nhược điểm duy nhất của nó là số lượng thư viện hỗ trợ vẫn còn hạn chế. .NET Framework có hệ sinh thái lớn hơn với nhiều các thư viện hỗ trợ hơn.



Hình 2.2: Kiến trúc của .NET

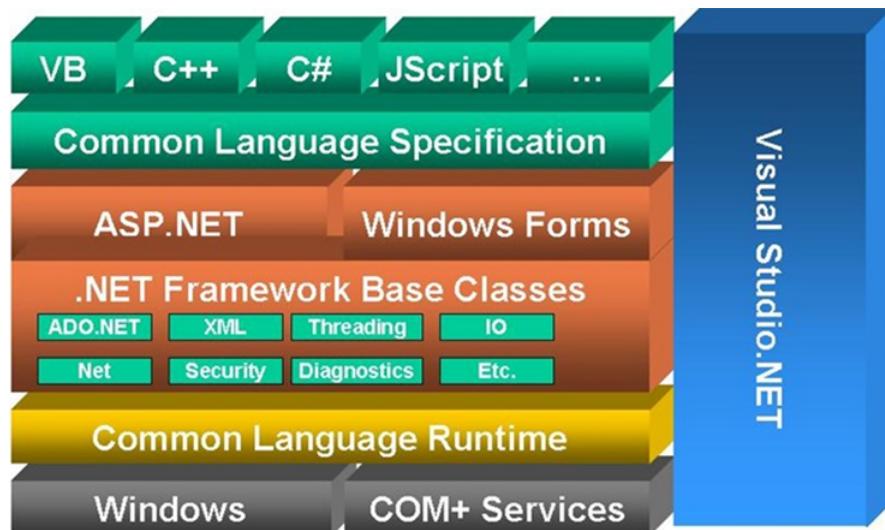
2.2.1 .NET Framework

Như đã nêu ở phần 2.1 ngôn ngữ C# ra đời để phục vụ Microsoft phát triển nền tảng .NET Framework phục vụ phát triển cho nhiều ứng dụng cho nên C# không đứng riêng lẻ mà nó là 1 phần của nền tảng .NET, .NET Framework bao gồm môi trường phát triển, hỗ trợ đa ngôn ngữ mà C# là một trong số đó (ngoài ra có F#, VB.NET, Managed C++, J#). Những thành phần của .NET Framework (Hình 2.3) bao gồm:

- Các ngôn ngữ lập trình .NET (C#, VB.NET...).
- Môi trường thực thi code (CLR) sẽ thực thi chương trình được viết từ ngôn ngữ lập trình.
- Các công cụ phát triển như trình biên dịch csc dùng để biên dịch ngôn ngữ C# sang mã trung gian (MSIL) mà CLR có thể hiểu.
- Tập các thư viện chuẩn (Class Library) như ADO.NET cho phép truy cập database (ví dụ SQL Server hoặc MySQL) và WCF cho phép tạo ra các ứng dụng API theo chuẩn HTTP và trả về JSON, SOAP...

.Net Framework hiện tại là phiên bản hỗ trợ nhiều ứng dụng nhất khi nó có thể dùng để viết các ứng dụng chạy trên nền Windows như: Web App, Win

App hay là nó có thể xây dựng những tầng Web API giàn cho nền tảng Web hay Mobile.



Hình 2.3: Kiến trúc .Net Framework

2.2.2 Mono

Mono là một nền tảng open-source với mục đích chính là tạo những ứng dụng cross-platform trên nền .Net. Chúng ta có thể sử dụng Mono trên các hệ điều hành như Unix, Linux, Mac OS X, Solaris và tất nhiên là Windows. Bất kì ngôn ngữ nào được biên dịch thành mã IL thuần túy đều có thể tương thích với Mono. Ngoài ra, Mono còn cung cấp thư viện hỗ trợ rất nhiều loại ngôn ngữ lập trình khác như Java, PHP, Python, Object Pascal, Cobra... Mono framework được sử dụng để lập trình ứng dụng mobile và game 2D, 3D. Các thành phần chính của Mono Framework:

- Mono runtime: cung cấp trình biên dịch Just-in-Time (JIT), Ahead-of-Time (AOT), thực thi, quản lý các tiến trình và giao tiếp với hệ thống.
- C# Compiler: bao gồm các công cụ
 - mcs: (phiên bản 1.1) hỗ trợ C# 1.0 và C# 3.0 ngoại trừ các tính năng về generic.
 - gmcs: (phiên bản 2.0) hỗ trợ đầy đủ C# 3.0.

- smcs: (phiên bản 2.1) hỗ trợ thêm Silverlight/Moonlight.
- dmcs: (phiên bản 4.0) hỗ trợ C# 4.0.
- Base Class Library: thư viện nền tảng để phát triển ứng dụng, tương thích với .Net framework.
- Mono Class Library: Cung cấp các thư viện lập trình như Gtk+, Zip files, LDAP, OpenGL, Cairo, POSIX,... [4]

2.2.3 .NET Core

.NET Core là một framework mã nguồn mở mới và framework đa nền tảng (cross-platform) cho việc xây dựng những ứng dụng hiện tại dựa trên kết nối đám mây, giống như web apps, IoT và backend cho mobile. Ứng dụng ASP.NET Core có thể chạy trên .NET Core hoặc trên phiên bản đầy đủ của .NET Framework. Nó được thiết kế để cung cấp và tối ưu hệ thống đang và đã phát triển cho những ứng dụng được triển khai trên đám mây (cloud) hoặc chạy on-promise. Nó bao gồm các thành phần theo hướng module nhằm tối thiểu tài nguyên và chi phí phát triển. Chúng ta có thể phát triển và chạy những ứng dụng ASP.NET Core đa nền tảng trên Windows, Mac và Linux. Đồng thời nó đã trở thành một mã nguồn mở. Đây là một thay đổi rất lớn và theo em là quan trọng nhất của ASP.NET Core. Điều mà trước đây khó có một lập trình viên nào có thể nghĩ đến. Có lẽ đó cũng là một xu thế mà các ngôn ngữ lập trình hiện nay đang hướng tới.

Bản phát hành đầu tiên của ASP.NET đã xuất hiện cách đây 15 năm trước, nó là một phần của .NET Framework. Từ đó, hàng triệu lập trình viên đã sử dụng nó để xây dựng những ứng dụng web tuyệt vời, và trên những năm đó Microsoft đã phát triển thêm nhiều tính năng mới. ASP.NET Core có một số thay đổi kiến trúc lớn, đó là kết quả của việc học hỏi rất nhiều từ các framework module hóa khác. ASP.NET Core không còn dựa trên System.Web.dll nữa. Nó được dựa trên một tập hợp các gói, các module hay cũng được gọi là các Nuget packages. Điều này cho phép chúng ta tối ưu ứng dụng của chúng ta để chỉ bao gồm những packages nào cần thiết. Lợi ích của nó là giúp cho ứng dụng nhỏ hơn, bảo mật chặt chẽ hơn, giảm sự phức tạp, tối ưu hiệu suất hoạt động và giảm chi phí, thời gian cho việc phát triển. Với ASP.NET Core chúng ta đạt được những

nền tảng cài tiến dưới đây:

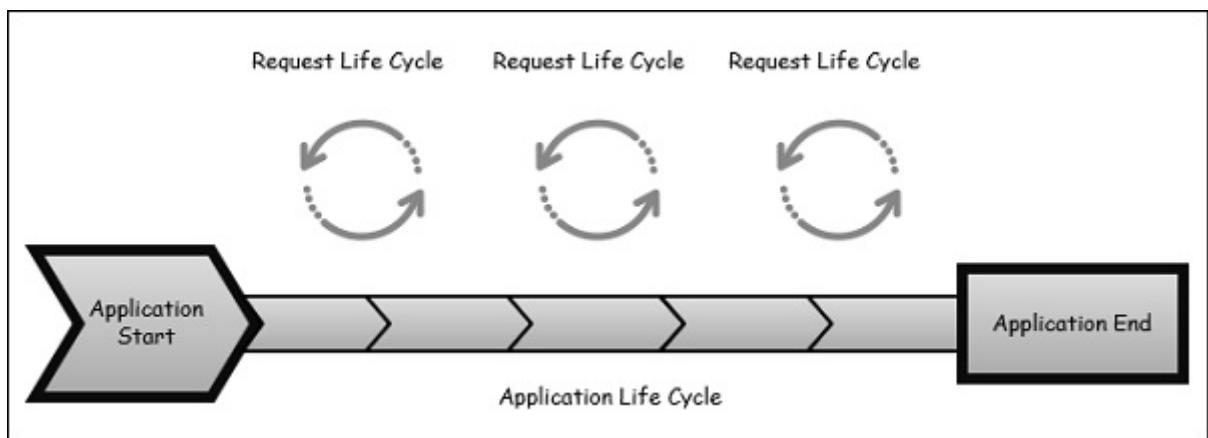
- Hợp nhất việc xây dựng web UI và web APIs.
- Tích hợp những client-side frameworks hiện đại và những luồng phát triển.
- Hệ thống cấu hình dựa trên môi trường đám mây thật sự.
- Dependency injection được xây dựng sẵn.
- HTTP request được tối ưu nhẹ hơn.
- Có thể host trên IIS hoặc self-host trong process của riêng project hiện tại.
- Được xây dựng trên .NET Core, hỗ trợ thực sự app versioning.
- Chuyển các thực thể, thành phần, module như những NuGet packages
- Những công cụ mới để đơn giản hóa quá trình phát triển web hiện đại
- Xây dựng và chạy đa nền tảng(Windows, Mac và Linux)
- Mã nguồn mở và tập trung vào cộng đồng

2.2.4 Vòng đời yêu cầu trong ASP.NET

Trong ASP.NET có 2 loại vòng đời:

- Vòng đời của ứng dụng là quá trình ứng dụng thực sự bắt đầu chạy IIS cho đến khi nó dừng lại. Vòng đời ứng dụng có thể chia thành các giai đoạn sau:
 - Người dùng gửi yêu cầu truy cập vào dữ liệu của ứng dụng và trình duyệt sẽ gửi yêu cầu này đến Web Server
 - Các sự kiện sau đây sẽ được khởi tạo:
 - * Một đối tượng của lớp ApplicationManager được tạo.
 - * Một đối tượng lớp HostingEnvironment được tạo để cung cấp thông tin về nguồn dữ liệu.
 - * Các thành phần đầu của ứng dụng sẽ được biên dịch.

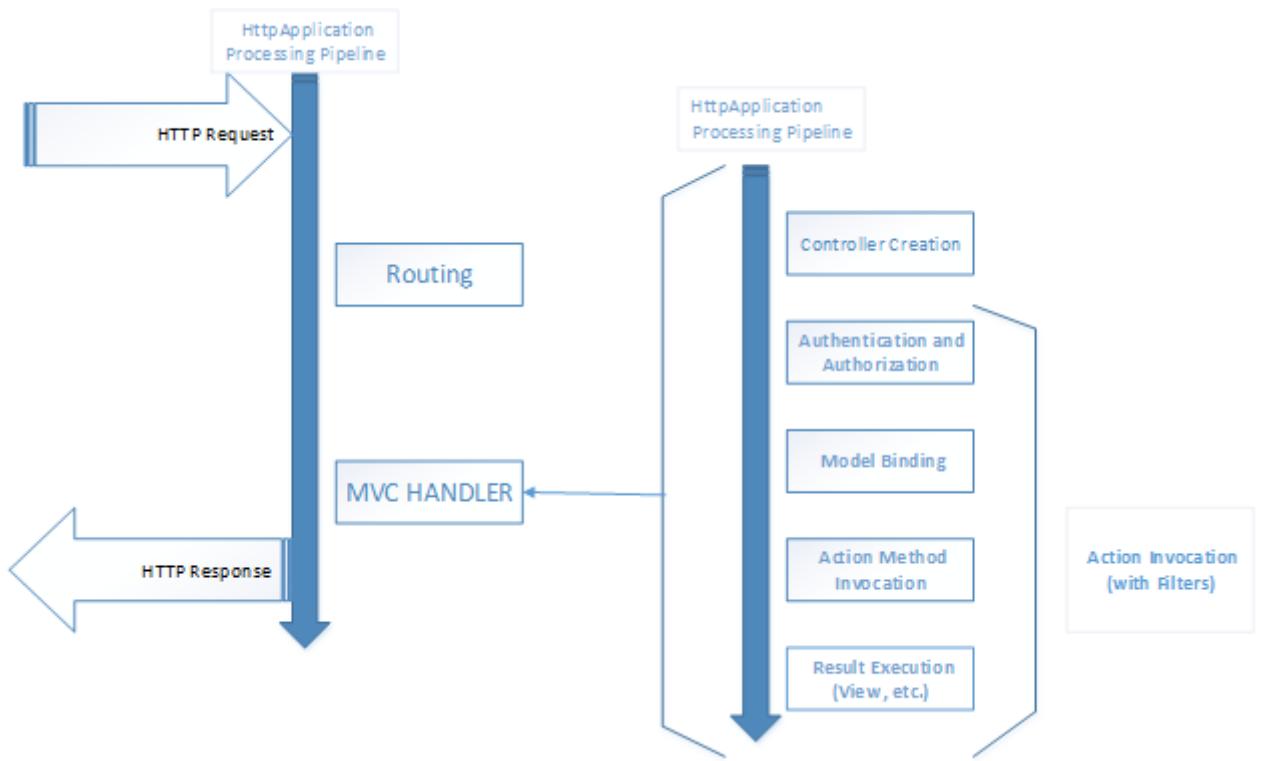
- Các đối tượng HttpContext, HttpRequest và HttpResponse được khởi tạo và cài đặt.
- Một thẻ hiện của đối tượng HttpApplication được tạo và gắn cho yêu cầu.
- Các yêu cầu được xử lý bởi lớp HttpApplication, các sự kiện khác nhau được kích hoạt bởi lớp này để xử lý các yêu cầu.
- Vòng đời của yêu cầu gửi lên server.



Hình 2.4: Tổng quan vòng đời trong ASP.NET nói chung

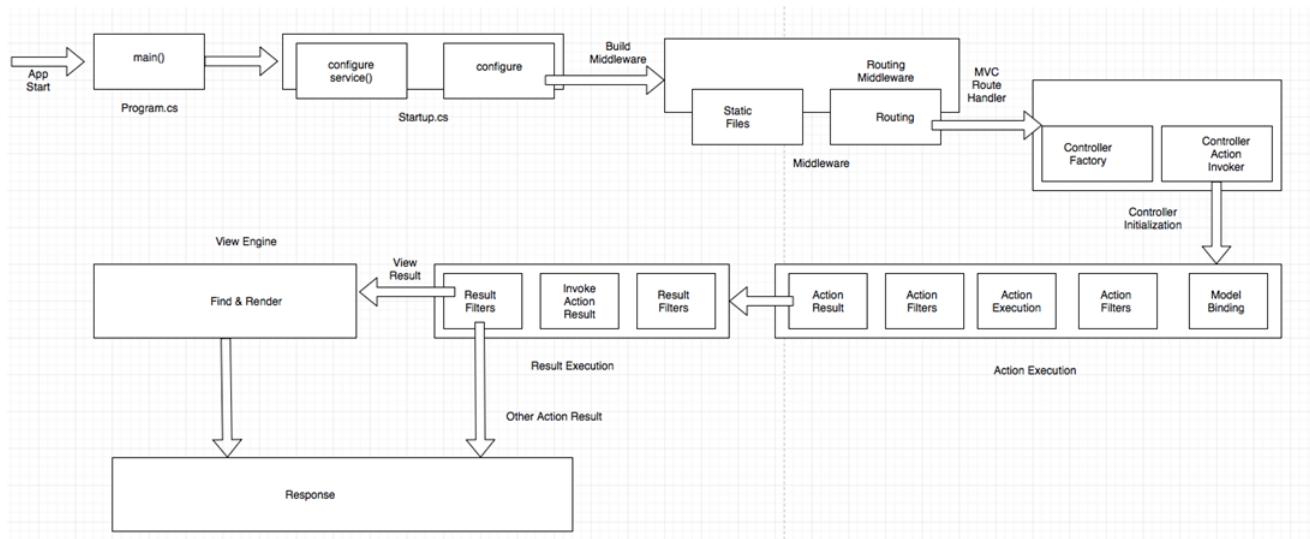
Hình 2.4 là 1 vòng đời của 1 ứng dụng trong nền tảng .NET ta có thể thấy trong 1 vòng đời ứng dụng có rất nhiều các yêu cầu gửi lên từ các user cho server xử lý, mỗi một vòng đời của 1 yêu cầu gửi lên cho server xử lý sẽ chứa rất nhiều sự kiện mà người dùng thao tác và mong muốn server trả kết quả ra ngoài màn hình. Kết thúc 1 vòng đời 1 của yêu cầu hệ thống sẽ tự động đóng kết nối và dọn những thực thể mà yêu cầu đã sử dụng.

Hình 2.5 là hình mô tả chi tiết vòng đời trong ASP.NET MVC ở mức 2 level. Khi có 1 yêu cầu gửi lên ứng dụng sẽ có một chuỗi sự kiện được xử lý ở server, nó sẽ được định tuyến qua 1 đường ống xử lý trong lớp HttpApplication để đến được đúng Controller và Action của nó, khi vào Controller sự kiện đầy phải được kiểm tra quyền đăng nhập và quyền ứng dụng nếu ứng dụng có cài đặt 2 quyền trên, sau đó là bước truyền các tham số vào cho Action xử lý nếu qua được bước kiểm tra quyền cuối cùng Action sẽ trả về 1 kết quả cho Controller gửi ra View và trả về 1 kết quả được hiển thị cho người dùng thông qua View.



Hình 2.5: Vòng đồi trong ASP.NET MVC

Hình 2.6 mô tả chi tiết vòng đồi trong ASP.NET Core. Khi có một yêu cầu gửi đến server nó cũng sẽ được định tuyến qua 1 URL nhất định, vì ASP.NET Core được thiết kế theo dạng module hoá nên các module được khởi tạo và sử dụng được khai báo trong file StartUp.cs trong đó phương thức ConfigureServices() để khai báo những module được sử dụng trong hệ thống còn phương thức Configure() được sử dụng để sử dụng các module đã khai báo. Sau đây qua tầng Middleware là tầng ở giữa để xử lý định tuyến URL và những file tĩnh trong hệ thống như: file css, js,... Ở tầng Middleware này lập trình viên có thể lập trình cho hệ thống kèm một số thứ lên kèm với yêu cầu người dùng như là gửi kèm 1 header để xác thực người dùng chặng hạn. Sau khi định tuyến được yêu cầu thì server xử lý các tham số của yêu cầu vào trong Controller, ở Controller các thẻ hiện sẽ được tiêm vào hàm tạo của Controller theo cơ chế Dependency Injection. Sau khi xử lý thành công ở Controller thì hệ thống sẽ trả lại kết quả đến hiện thị trên View cho người dùng.



Hình 2.6: Vòng đời trong ASP.NET Core

2.3 CSHTML

CSHTML là 1 view engine dùng để xử lý và tạo HTML, nó là 1 view để nhận các dữ liệu mà controller gửi đến, CSHTML cho phép viết mã nguồn C# để xử lý dữ liệu trên view dễ dàng hơn. CSHTML sử dụng Razor để viết mã. Cơ chế Razor giảm thiểu số lượng ký tự và phím nhấn cần để tạo một tập tin View, do đó lập trình viên có thể làm việc nhanh và trôi chảy hơn. Khi chạy trên môi trường web thì file CSHTML sẽ tự động trở thành mã HTML để hiển thị lên các trình duyệt website

2.4 Bootstrap

Bootstrap là 1 framework HTML, CSS, và JavaScript cho phép người dùng dễ dàng thiết kế website theo 1 chuẩn nhất định, tạo các website thân thiện với các thiết bị cầm tay như mobile, ipad, tablet,...

Bootstrap bao gồm những cái cơ bản có sẵn như: typography, forms, buttons, tables, navigation, modals, image carousels và nhiều thứ khác. Trong bootstrap có thêm nhiều Component, Javascript hỗ trợ cho việc thiết kế responsive của chúng ta dễ dàng, thuận tiện và nhanh chóng hơn. Nên dùng bootstrap để viết giao diện front-end vì:

- Bootstrap là một trong những framework được sử dụng nhiều nhất trên thế giới để xây dựng nên một website. Bootstrap đã xây dựng nên 1 chuẩn riêng và rất được người dùng ưa chuộng. Chính vì thế, chúng ta hay nghe tới một cụm từ rất thông dụng "Thiết kế theo chuẩn Bootstrap".
- Rất dễ để sử dụng: Nó đơn giản vì nó được base trên HTML, CSS và Javascript chỉ cần có kiến thức cơ bản về 3 cái đó là có thể sử dụng bootstrap tốt.
- Responsive: Bootstrap xây dựng sẵn responsive css trên các thiết bị Iphones, tablets, và desktops. Tính năng này khiến cho người dùng tiết kiệm được rất nhiều thời gian trong việc tạo ra một website thân thiện với các thiết bị điện tử, thiết bị cầm tay.
- Tương thích với trình duyệt: Nó tương thích với tất cả các trình duyệt (Chrome, Firefox, Internet Explorer, Safari, and Opera). Tuy nhiên, với IE browser, Bootstrap chỉ hỗ trợ từ IE9 trở lên. Điều này vô cùng dễ hiểu vì IE8 không support HTML5 và CSS3.

2.5 Jquery Ajax

2.5.1 Jquery

jQuery là một Framework được xây dựng dựa trên các tính năng của JavaScript. Vì thế trong khi phát triển các ứng dụng sử dụng jQuery, chúng ta có thể sử dụng tất cả các hàm và các tính năng khác được bổ trợ trong JavaScript. jQuery làm đơn giản hóa việc truyền tải HTML, xử lý sự kiện, tạo hiệu ứng động và tương tác Ajax. Với jQuery, khái niệm Rapid Web Development đã không còn quá xa lạ. jQuery là một bộ công cụ tiện ích JavaScript làm đơn giản hóa các tác vụ đa dạng với việc viết ít code hơn. Dưới đây liệt kê một số tính năng tối quan trọng được hỗ trợ bởi jQuery:

- Thao tác DOM: jQuery giúp dễ dàng lựa chọn các phần tử DOM để traverse (duyệt) một cách dễ dàng như sử dụng CSS, và chỉnh sửa nội dung của chúng bởi sử dụng phương tiện Selector mã nguồn mở, mà được gọi là Sizzle.

- Xử lý sự kiện: jQuery giúp tương tác với người dùng tốt hơn bằng việc xử lý các sự kiện đa dạng mà không làm cho HTML code rối tung lên với các Event Handler.
- Hỗ trợ AJAX: jQuery giúp chúng ta rất nhiều để phát triển một site giàu tính năng và phản hồi tốt bởi sử dụng công nghệ AJAX.
- Hiệu ứng: jQuery đi kèm với rất nhiều các hiệu ứng đa dạng và đẹp mắt mà chúng ta có thể sử dụng trong các Website của mình.
- Gọn nhẹ: jQuery là thư viện gọn nhẹ - nó chỉ có kích cỡ khoảng 19KB (gzipped).
- Được hỗ trợ hầu hết bởi các trình duyệt hiện đại: jQuery được hỗ trợ hầu hết bởi các trình duyệt hiện đại, và làm việc tốt trên IE 6.0+, FF 2.0+, Safari 3.0+, Chrome và Opera 9.0+

Bảng 2.1: Các hàm có sẵn trong Jquery [3]

STT	Tên hàm	Mô tả hàm
1	charAt()	Trả về ký tự tại chỉ mục (index) đã cho.
2	concat()	Kết nối hai chuỗi văn bản và trả về một chuỗi mới.
3	forEach()	Gọi một hàm cho mỗi phần tử của một mảng.
4	indexOf()	Trả về chỉ mục về sự xuất hiện đầu tiên bên trong việc gọi đối tượng String với giá trị đã cho, hoặc -1 nếu không tìm thấy.
5	length()	Trả về độ dài của chuỗi.
6	pop()	Gỡ bỏ phần tử cuối của một mảng và trả về phần tử đó.
7	push()	Thêm một hoặc nhiều phần tử tới phần cuối của một mảng và trả về độ dài mới của mảng đó.

8	reverse()	Dảo ngược thứ tự các phần tử trong một mảng – phần tử đầu tiên thành cuối cùng và cuối cùng thành đầu tiên.
9	sort()	Sắp xếp phân loại các phần tử của một mảng.
10	substr()	Trả về các ký tự trong một mảng bắt đầu từ vị trí đã cho từ số các ký tự đã xác định.
11	toLowerCase()	Trả về giá trị chuỗi đang gọi được biến đổi thành kiểu chữ thường.
12	toString()	Trả về sự biểu diễn chuỗi của giá trị số.
13	toUpperCase()	Trả về giá trị chuỗi đang gọi được biến đổi thành chữ hoa.

2.5.2 Ajax

Ajax là một bộ công cụ cho phép load dữ liệu từ server mà không yêu cầu tải lại trang. Nó sử dụng chức năng sẵn có XMLHttpRequest(XHR) của trình duyệt để thực hiện một yêu cầu đến server và xử lý dữ liệu server trả về.

Ví dụ : khi một người dùng viết một nhận xét trên bài viết đăng trên trang Facebook. Sau khi người dùng gửi nhận xét thành công trang Facebook mà người đó đang truy cập cần phải được cập nhật để hiển thị nhận xét vừa mới được tạo ra này. Nếu load lại toàn bộ trang mà người dùng đang truy cập thì sẽ không hiệu quả do tất cả những gì chúng ta muốn là hiển thị nhận xét mới được tạo ra, Ajax được tạo ra để giải quyết vấn đề này, thay vì tải lại toàn bộ trang trình duyệt sẽ chỉ load những phần được thay đổi để tiết kiệm thời gian chờ đợi một lượng thông tin lớn về từ server.

Một số ứng dụng sử dụng Ajax như : Gmail , Google Maps , Youtube , Facebook,...

Jquery cung cấp một số phương thức để thực hiện các chức năng ajax. Chúng ta có thể yêu cầu các text, HTML, XML và JSON từ server sử dụng cả giao thức HTTP GET và HTTP POST, chúng ta cũng có thể lấy dữ liệu từ bên

ngoài trực tiếp vào trong phần tử được chọn.

- Phương thức jquery load()

```
$(selector).load(URL,data,callback);
```

- Phương thức load() lấy dữ liệu từ server và trả dữ liệu cho phần tử được chọn.
- URL: mà chúng ta muốn lấy dữ liệu.
- Data: cặp key/value gửi đi cùng với yêu cầu.
- Callback: tên của hàm sẽ được thực thi sau khi phương thức load hoàn thành.

- Phương thức Post trong JQuery Ajax

```
$(selector).post(URL,data,function(data,status,xhr),dataType)
```

- Có tác dụng lấy dữ liệu từ server bằng phương thức HTTP POST REQUEST
- URL: required , đường dẫn đến file cần lấy thông tin.
- Data: không bắt buộc , là một đối tượng object gồm các key và value sẽ gửi lên server.
- function(data, status , xhr): là function sẽ xử lý khi thực hiện thành công với các parameters:
 - * Data : bao gồm các dữ liệu trả về từ request.
 - * Status : gồm trạng thái request (“success” , “notmodified” , “error” , “timeout” , or “parsererror”).
 - * Xhr : gồm các đối tượng XMLHttpRequest.
- dataType: là dạng dữ liệu trả về. (text, json, script, xml,html,jsonp)

- Phương thức Get trong Jquery Ajax

```
$(selector).post(URL,data,function(data,status,xhr),dataType)
```

- Có tác dụng lấy dữ liệu từ server bằng phương thức HTTP POST REQUEST.
- Các thành phần tương tự như phương thức Post.

2.6 Phần mềm hỗ trợ và phát triển website

2.6.1 Visual Studio

Visual Studio là (IDE – Integrated Development Environment) một bộ công cụ phát triển phần mềm do Microsoft phát triển. Visual Studio cũng là một phần mềm được sử dụng bởi các lập trình viên để xây dựng nên các sản phẩm phần mềm. Nó được sử dụng để phát triển chương trình máy tính cho Microsoft Windows, cũng như các trang web, các ứng dụng web và các dịch vụ web. Visual Studio sử dụng nền tảng phát triển phần mềm của Microsoft như Windows API, Windows Forms, Windows Presentation Foundation, Windows Store và Microsoft Silverlight. Nó có thể sản xuất cả hai ngôn ngữ máy và mã số quản lý.

Visual Studio bao gồm một trình soạn thảo mã hỗ trợ IntelliSense cũng như cải tiến mã nguồn. Trình gõ lỗi tích hợp hoạt động cả về trình gõ lỗi mức độ mã nguồn và gõ lỗi mức độ máy. Công cụ tích hợp khác bao gồm một mẫu thiết kế các hình thức xây dựng giao diện ứng dụng, thiết kế web, thiết kế lớp và thiết kế giản đồ cơ sở dữ liệu. Nó chấp nhận các plug-in nâng cao các chức năng ở hầu hết các cấp bao gồm thêm hỗ trợ cho các hệ thống quản lý phiên bản (như Subversion) và bổ sung thêm bộ công cụ mới như biên tập và thiết kế trực quan cho các miền ngôn ngữ cụ thể hoặc bộ công cụ dành cho các khía cạnh khác trong quy trình phát triển phần mềm.

Em chọn Visual Studio bởi nó có những điểm mạnh sau đây:

- Hỗ trợ lập trình trên nhiều ngôn ngữ như C/C++, C#, F#, Visual Basic, HTML, CSS, JavaScript. Phiên bản Visual Studio 2015 có hỗ trợ ngôn ngữ Python.
- Visual Studio là một công cụ hỗ trợ việc Debug một cách mạnh mẽ, dễ dàng nhất (Break Point, xem giá trị của biến trong quá trình chạy, hỗ trợ debug từng câu lệnh).

- Giao diện Visual Studio rất dễ sử dụng đối với người mới bắt đầu.
- Visual Studio hỗ trợ phát triển ứng dụng desktop MFC, Windows Form, Universal App, ứng dụng mobile Windows Phone 8/8.1, Windows 10, Android (Xamarin), iOS và phát triển website Web Form, ASP.NET MVC và phát triển Microsoft Office.
- Visual Studio hỗ trợ kéo thả để xây dựng ứng dụng một cách chuyên nghiệp, giúp các lập trình viên mới bắt đầu có thể tiếp cận nhanh hơn.
- Visual Studio cho phép chúng ta tích hợp những extension từ bên ngoài như Resharper (hỗ trợ quản lý và viết mã nhanh cho các ngôn ngữ thuộc .Net), hay việc cài đặt thư viện nhanh chóng thông qua Nuget
- Visual Studio được sử dụng rộng rãi bởi lập trình viên trên toàn thế giới.

2.6.2 Visual Studio Code

Visual Studio Code là sản phẩm của Microsoft, ra mắt vào tháng 4 năm 2015 ở hội nghị Build. Đặc điểm nổi bật là đơn giản, gọn nhẹ, dễ dàng cài đặt. Visual Studio Code có thể cài đặt được trên cả Windows, Linux và Mac OS và hỗ trợ nhiều ngôn ngữ. Điểm mạnh của Visual Studio Code:

- VS Code còn hỗ trợ tất cả các ngôn ngữ lập trình trên nhiều nền tảng khác nhau với bộ thư viện Extension phong phú. VS Code cũng là một công cụ phát triển web rất tốt.
- VS Code chỉ tập trung quản trị code trên đơn vị file(như một công cụ text-editor), không như Visual Studio thiêng về project và solution. Đặc điểm này giống với Sublime hay Atom. Cũng dễ hiểu khi VS Code cũng được xây dựng trên nền Electron như Sublime.
- VS Code với trình gợi ý và tính năng auto-completion tích hợp trí tuệ nhân tạo sẽ đem đến cho chúng ta một trải nghiệm mới hơn và tiện lợi hơn.
- VS Code cũng tích hợp Git cũng như bộ command của nó giúp developer có thể nhanh chóng tải và cài đặt các project từ Git repo.
- Đặc biệt VS Code là miễn phí nếu không nói là mã nguồn mở.

2.6.3 Microsoft Sql Server Management Studio

2.6.3.1 Khái niệm về SQL Server

SQL Server chính là một hệ quản trị dữ liệu quản lý sử dụng câu lệnh SQL để trao đổi dữ liệu giữa máy cài SQL Server và máy Client. Một Relational Database Management System – RDBMS gồm có: databases, database engine và các chương trình ứng dụng dùng để quản lý các bộ phận trong RDBMS và những dữ liệu khác.

2.6.3.2 Lịch sử ra đời và các phiên bản của SQL Server

Năm 1989, phiên bản đầu tiên của SQL Server 1.0 ra đời được dùng cho các hệ điều hành 16 bit và được phát triển cho tới ngày nay.

Cho tới khi SQL Server ra phiên bản 6.5 thì được thị trường chấp nhận rộng rãi. Một đột phá cải tiến cho SQL Server 7.0 khi được Microsoft viết lại một engine hoàn toàn mới. Đến khi SQL Server từ phiên bản 7.0 cải tiến lên 8.0 chủ yếu phát triển về tính năng thiết kế web.

Cho đến ngày nay thì phiên bản mới nhất đó là SQL Server 2017 hỗ trợ bộ vi xử lý 64 bit ra đời vào ngày 1 tháng 6 năm 2017. Cho đến nay SQL Server có 5 phiên bản:

- Enterprise: là một phiên bản chứa tất cả các đặc điểm nổi bật của SQL Server như: các công cụ cho tạo và quản lý phân cụm SQL Server, nhân bộ máy cơ sở dữ liệu và một số dịch vụ đi kèm. Nó có thể đánh địa chỉ 12 terabytes và quản lý cơ sở dữ liệu lên tới 524 petabytes.
- Standard: Phiên bản này có thể chạy tốt trên hệ thống lên tới 4 CPU và 2 GB RAM rất thích hợp cho các dịch vụ thiết kế web vừa và nhỏ.
- Developer: Phiên bản này giới hạn số lượng người kết nối với server nhưng có đầy đủ các tính năng của Enterprise Edition. Đây là phiên bản được sử dụng cho kiểm tra và phát triển ứng dụng phù hợp cho các cá nhân trong lĩnh vực web như: freelancer Việt Nam.
- Workgroup: phiên bản SQL Server này có các chức năng lõi cơ sở dữ liệu nhưng không đi kèm các dịch vụ. Ở phiên bản 2012 không có phiên bản này.

- Express: Ân bản này dễ dàng sử dụng và quản trị cơ sở dữ liệu đơn giản.

2.6.3.3 Các thành phần cơ bản trong SQL Server

Các thành phần cơ bản trong SQL Server gồm có: Reporting Services, Database Engine, Integration Services, Notification Services, Full Text Search Service, ... Tất cả kết hợp với nhau tạo thành một giải pháp hoàn chỉnh giúp cho việc phân tích và lưu trữ dữ liệu trở nên dễ dàng hơn.

- Database Engine: Đây là một engine có khả năng chứa dữ liệu ở các quy mô dưới dạng support và table. Ngoài ra, nó còn có khả năng tự điều chỉnh ví dụ: trả lại tài nguyên cho hệ điều hành khi một user log off và sử dụng thêm các tài nguyên của máy khi cần.
- Integration Services: là tập hợp các đối tượng lập trình và các công cụ đồ họa cho việc sao chép, di chuyển và chuyển đổi dữ liệu. Khi bạn làm việc trong một công ty lớn thì dữ liệu được lưu trữ ở nhiều nơi khác nhau như được chứa trong: Oracle, SQL Server, DB2, Microsoft Access, ... và bạn chắc chắn sẽ có nhu cầu di chuyển dữ liệu giữa các server này. Ngoài ra, bạn còn muốn định dạng dữ liệu trước khi lưu vào database. Chắc chắn Integration Services sẽ giúp bạn giải quyết được công việc này dễ dàng.
- Analysis Services: Đây là một dịch vụ phân tích dữ liệu rất hay của Microsoft. Dữ liệu khi được lưu trữ vào trong database mà bạn không thể lấy được những thông tin bổ ích thì coi như không có ý nghĩa gì. Chính vì thế, công cụ này ra đời giúp bạn trong việc phân tích dữ liệu một cách hiệu quả và dễ dàng bằng cách dùng kỹ thuật khai thác dữ liệu – datamining và khái niệm hình khối nhiều chiều – multi dimension cubes.
- Notification Services: Dịch vụ thông báo này là nền tảng cho sự phát triển và triển khai các ứng dụng soạn và gửi thông báo. Ngoài ra, dịch vụ này còn có chức năng gửi thông báo theo lịch thời gian hàng ngày đến nhiều người dùng ký sử dụng trên nhiều loại thiết bị khác nhau.
- Reporting Services: là một công cụ tạo, quản lý và triển khai báo cáo bao gồm: server và client. Ngoài ra, nó còn là nền tảng cho việc phát triển và xây dựng các ứng dụng báo cáo.

- Full Text Search Service: là một thành phần đặc biệt trong việc truy vấn và đánh chỉ mục dữ liệu văn bản không cấu trúc được lưu trữ trong các cơ sở dữ liệu SQL Server.
- Service Broker: là một môi trường lập trình cho việc tạo ra các ứng dụng trong việc nhảy qua các Instance.

2.6.3.4 Tại sao sử dụng SQL Server trong thiết kế website

SQL Server không phải là một hệ quản trị cơ sở dữ liệu độc lập mà nó chỉ là một thành phần với vai trò ngôn ngữ là công cụ giao tiếp giữa hệ cơ sở dữ liệu và người dùng. Chính vì thế nó được sử dụng trong các dịch vụ thiết kế web đep với chức năng giao tiếp với người dùng với các vai trò sau:

- SQL là một ngôn ngữ đòi hỏi có tính tương tác cao: Người dùng có thể dễ dàng trao đổi với các tiện ích thông qua các câu lệnh của SQL đến cơ sở dữ liệu và nhận kết quả từ cơ sở dữ liệu.
- SQL là một ngôn ngữ lập trình cơ sở dữ liệu: Các lập trình viên có thể xây dựng các chương trình ứng dụng giao tiếp với cơ sở dữ liệu bằng cách nhúng các câu lệnh SQL vào trong ngôn ngữ lập trình.
- SQL là một ngôn ngữ lập trình quản trị cơ sở dữ liệu: Người quản trị cơ sở dữ liệu có thể quản lý, định nghĩa và điều khiển truy cập cơ sở dữ liệu thông qua SQL.
- SQL là một ngôn ngữ lập trình cho các hệ thống chủ khách: SQL được sử dụng như là một công cụ giao tiếp với các trình ứng dụng trong hệ thống cơ sở dữ liệu khách chủ.
- SQL là ngôn ngữ truy cập dữ liệu trên Internet: SQL được sử dụng với vai trò tương tác với dữ liệu trong hầu hết các máy chủ web và máy chủ Internet.
- SQL là ngôn ngữ cơ sở dữ liệu phân tán: Với vai trò giao tiếp với các hệ thống trên mạng, gửi và nhận các yêu cầu truy xuất dữ liệu với nhau.

2.7 Kết luận

Từ những lý thuyết về các ngôn ngữ lập trình cũng như các Framework mà em đã tìm hiểu, em sẽ vận dụng chúng để thiết kế và triển khai website được trình bày ở các chương sau

Chương 3

Xây dựng thuật toán đa truy nhập

Từ những cơ sở lý thuyết thu thập ở chương 2, chương này em sẽ trình bày quá trình thiết kế thuật toán đa truy nhập để ứng dụng thuật toán vào hệ thống BKRES-LoRa.

3.1 Phân tích yêu cầu bài toán đa truy nhập

Từ việc phân tích chi tiết yêu cầu và chức năng của thuật toán đa truy nhập cần phải đáp ứng, em đã xác định được yêu cầu chức năng và phi chức năng của hệ thống như sau:

- Yêu cầu chức năng:
 - Có khả năng đa truy nhập ($n - 1$), tức là nhiều nút sẽ gửi dữ liệu đến 1 Gateway,
 - Có khả năng xử lý bài toán join/disjoin mạng,
 - Hoạt động được với số lượng nút lớn,
- Yêu cầu phi chức năng:
 - Các nút truyền dữ liệu chính xác,
 - Nút tiêu tốn ít năng lượng,
 - Thiết bị hoạt động ổn định với thuật toán,

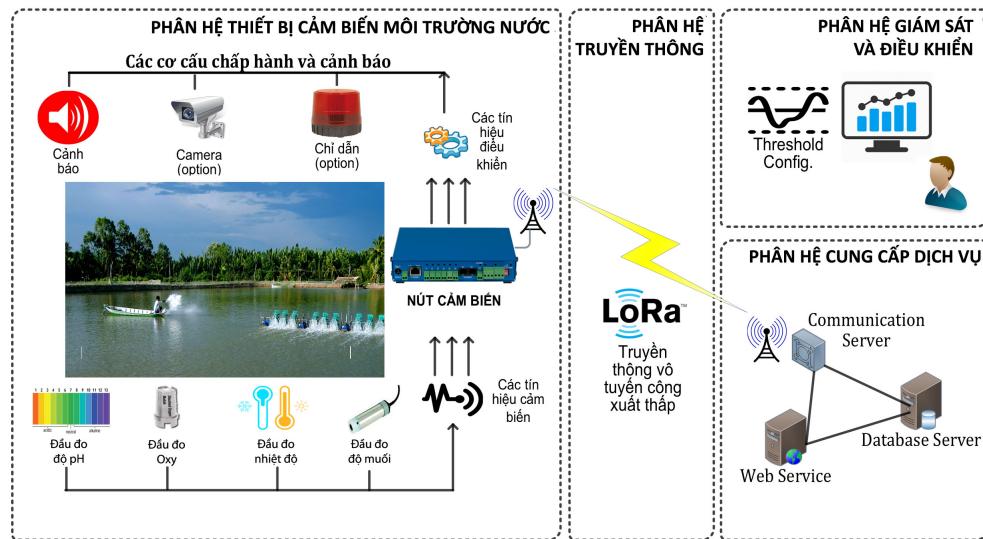
3.2 Hệ thống BKRES-LoRa

3.2.1 Kiến trúc tổng thể hệ thống

Hệ thống giám sát tham số môi trường nước BKRES sau khi được tích hợp thêm module truyền thông LoRa thì dữ liệu thu thập được từ các nút sẽ được gửi đến Gateway thông qua module truyền thông LoRa, còn Gateway sẽ giao tiếp với server thông qua một số giao thức như TCP/IP, GSM,...Hình 3.1 sẽ mô tả rõ hơn kiến trúc gồm 4 phân hệ chính của hệ thống BKRES-LoRa.

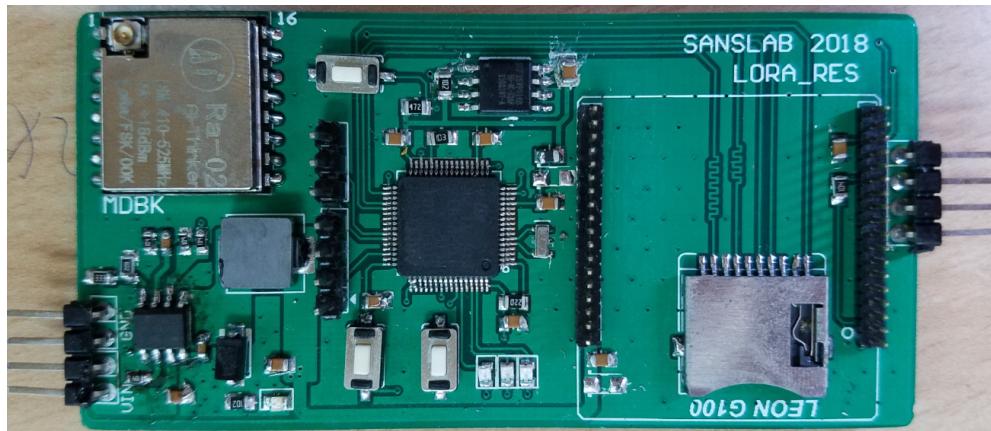
- Phân hệ cảm biến và xử lý dữ liệu: Ở mỗi nút mạng được trang bị các bộ cảm biến ghi đo 4 tham số (hàm lượng Oxy, nhiệt độ, độ pH và nồng độ muối) để thu thập dữ liệu tham số môi trường. Dữ liệu cảm biến được gửi đến bộ điều khiển trung tâm (sử dụng vi điều khiển STM32) để phân tích, xử lý, đóng gói và gửi đến module truyền thông LoRa.
- Phân hệ truyền thông sử dụng LoRa: Hoạt động theo mô hình đơn chặng, có nhiệm vụ gửi dữ liệu từ phân hệ cảm biến đến Gateway và từ đó dữ liệu được gửi đến server trung tâm.
- Phân hệ cung cấp dịch vụ: Sau khi nhận dữ liệu từ phân hệ truyền thông, phân hệ này có nhiệm vụ xử lý dữ liệu và cung cấp các dịch vụ cho phân hệ giám sát và người dùng.
- Phân hệ giám sát và điều khiển: có nhiệm vụ hiển thị dữ liệu cảm biến một cách trực quan thông qua ứng dụng di động, ứng dụng web. Người dùng có thể cấu hình hệ thống, đặt các mức ngưỡng cảnh báo,... trên ứng dụng.

Module LoRa SX1278 được lựa chọn để tích hợp vào thiết bị cảm biến như thể hiện trên Hình 3.2. Nút cảm biến được thiết kế và chế tạo tài phòng nghiên cứu SANSLAB, gồm các thành phần phần cứng chính như: phân hệ cảm biến các tham số môi trường nước, phân hệ truyền thông, phân hệ vi xử lý và điều khiển, phân hệ cấp nguồn và phân hệ hiển thị, cảnh tại chỗ. Phần mềm trên nút cảm biến gồm firmware và drivers điều khiển hoạt động các phân hệ và ngoại vi, thuật toán đa truy nhập cũng được nhúng trong firmware của thiết bị.



Hình 3.1: Kiến trúc hệ thống

3.2.2 Nút mạng cảm biến tích hợp module SX1278



Hình 3.2: Nút cảm biến tích hợp chip LoRa SX1278

Nút cảm biến được thiết kế gồm các khối có nhiệm vụ như sau:

- Khối nguồn: có nhiệm vụ cung cấp nguồn cho toàn bộ mạch. Mức điện áp 3,3 V cung cấp cho vi điều khiển, thiết bị ngoại vi cùng với cảm biến. Nguồn cung cấp cho mạch là pin, nhỏ gọn và có thể sạc được nên rất phù hợp với các thiết bị trong mạng WSN,
- Khối xử lý trung tâm: có nhiệm vụ lấy dữ liệu từ các thiết bị cảm biến, xử lý rồi chuyển qua khối truyền thông, cũng như xử lý các thông tin nhận được từ khối truyền thông. Khối xử lý trung tâm sử dụng vi điều khiển

STM32F1 là một loại vi điều khiển lõi Cortex M3 có tốc độ xử lý cao, cấu hình mạnh, tiết kiệm năng lượng với kích thước nhỏ gọn và giá thành rẻ,

- Khối truyền thông: có nhiệm vụ truyền nhận dữ liệu giữa các thiết bị trong mạng. Khối truyền thông sử dụng module LoRa Ai Thinker SX1278 được kết nối với khối xử lý trung tâm qua giao tiếp SPI,
- Khối lưu trữ dữ liệu: được dùng để lưu trữ cấu hình hệ thống cùng với nhật ký cảm biến. Khối này gồm IC flash NAND SST25VF016B và thẻ nhớ. IC flash có tốc độ xử lý cao, dung lượng bộ nhớ nhỏ thích hợp với việc lưu trữ phần mềm cùng với cấu hình của hệ thống. Thẻ nhớ được sử dụng để lưu trữ nhật ký cảm biến, dễ dàng cho người sử dụng có thể kiểm tra và sao lưu dữ liệu.

3.2.3 Mô hình hoạt động của hệ thống BKRES-LoRa

Thuật toán đa truy nhập đề xuất được phát triển và thử nghiệm trên mô hình gồm 3 nút cảm biến và một Gateway như Hình 3.3. Để đánh giá khả năng truyền dữ liệu, độ ổn định, khả năng tùy biến và tự cấu hình của các nút trong mạng, các thông số cơ bản của các nút sẽ được thiết lập theo như Bảng 3.1.

Bảng 3.1: Bảng thông số cấu hình của các nút trong mạng

Thông số	Giá trị
Channel	11 (433 MHz)
BW	125 kHz
CR	4/5
SF	7
Header	ON
CRC	ON

Hoạt động của mô hình được mô tả như sau. Ban đầu, mỗi nút mới (sau khi tự động cấu hình) sẽ gửi yêu cầu tham gia mạng đến Gateway, sau khi được chấp nhận thì nút bắt đầu quá trình gửi dữ liệu đến Gateway. Trong quá trình này, mỗi nút sẽ nhận dữ liệu đã được xử lý từ vi điều khiển, rồi gửi dữ liệu đến Gateway theo một chu kỳ định sẵn. Trong trường hợp xảy ra lỗi với bộ truyền

thông LoRa, nút sẽ gửi dữ liệu trực tiếp đến server trung tâm qua mạng GSM (có sẵn trong BKRES phiên bản cũ), đến khi hết xảy ra lỗi với bộ truyền thông, nút lại bắt đầu lại quá trình như bình thường.



Hình 3.3: Mô hình hoạt động của mạng

3.3 Xây dựng thuật toán đa truy nhập

Do số lượng nút trong mạng ít, nên em quyết định sử dụng phương pháp đa truy nhập ngẫu nhiên, tức là khi nút có dữ liệu nó sẽ gửi luôn bản tin cho Gateway.

3.3.1 Thuật toán đa truy nhập của Gateway

Trong thuật toán đa truy nhập, Gateway có các nhiệm vụ chính sau:

- Nhận dữ liệu từ các nút,
- Quản lý nút trong mạng.

Sau khi đã tự cấu hình, trong vòng lặp vô hạn, Gateway liên tục mở kênh truyền để nhận bản tin. Nếu bản tin đó là do một nút mới gửi đến (nút chưa tham gia vào mạng) thì gateway sẽ tiến hành quá trình xác thực nút, rồi gửi bản tin phản hồi cho nút. Còn nếu bản tin là bản tin chứa dữ liệu, Gateway sẽ nhận và xử lý dữ liệu và tiếp tục quá trình mở kênh truyền để nhận dữ liệu. Algorithm 1 thể hiện thuật toán xử lý của Gateway.

Algorithm 1 Thuật toán của Gateway

```

1: pass  $\leftarrow$  @Rsanslab
2: Gateway tự động cấu hình  $\triangleright$  Gateway sẽ tự động cài một số thông số như
   ID, SF, CR, tần số, ...
3: while 1 do
4:   if Nhận được gói tin then
5:     packet  $\leftarrow$  packet_receive  $\triangleright$  gói tin nhận được
6:     if packet là gói tin gửi dữ liệu then
7:       data  $\leftarrow$  packet.data
8:       Xử lý dữ liệu  $\triangleright$  Gateway có thể gửi dữ liệu lên Server, hiển thị
   hoặc lưu dữ liệu
9:     else if packet là gói tin yêu cầu tham gia mạng then
10:    ID  $\leftarrow$  packet.src  $\triangleright$  Lấy ID của nút gửi đến
11:    authentication  $\leftarrow$  packet.data  $\triangleright$  Lấy dữ liệu bản tin xác thực
12:    response  $\leftarrow$  true
13:    if authentication  $\neq$  pass then
14:      response  $\leftarrow$  false
15:    end if
16:    gửi bản tin phản hồi response cho nút ID
17:  end if
18: end if
19: end while

```

3.3.2 Thuật toán đa truy nhập của nút

Chức năng chính của nút trong thuật toán đa truy nhập:

- Khi nút chưa tham gia mạng, nút sẽ gửi yêu cầu để tham gia mạng. Khi đó xảy ra một số trường hợp sau:
 - Quá trình yêu cầu không thành công, nút sẽ gửi lại,
 - Nếu bản tin xác thực sai, nút sẽ không hoạt động nữa,
 - Nếu quá trình yêu cầu tham gia mạng thành công, nút sẽ gửi dữ liệu.
- Sau khi tham gia mạng thành công, nút sẽ gửi dữ liệu theo chu kỳ.

Algorithm 2 Thuật toán của Nút

```

1: msgAuthentication ← @Rsanslab
2: Nút tự động cấu hình      ▷ Nút sẽ tự động cài một số thông số như ID, SF,
   CR, tần số, ...
3: idGateway ← Địa chỉ Gateway
4: response ← 2           ▷ response = 0 → bản tin xác thực chính xác; response
   = 1 → bản tin xác thực không chính xác; response = 2 → nút chưa tham
   gia mạng.
5: RequestJoinNetworks(idGateway, msgAuthentication)
6: while 1 do
7:   if response == 0 then
8:     data ← lấy dữ liệu từ cảm biến
9:     cnt = 0
10:    repeat
11:      state ← sendPacket(idGateway, data)
12:      if state > 0 then
13:        Delay()
14:      end if
15:      cnt ← cnt + 1
16:    until state > 0 and cnt < threshold      ▷ threshold là số lần gửi tối đa
17:    else if response == 1 then
18:      Sleep()
19:    else
20:      RequestJoinNetworks(idGateway, msgAuthentication)
21:    end if
22: end while

```

Sau khi được cấu hình, nút sẽ gửi bản tin xác thực cho Gateway (ID của Gateway được thiết lập sẵn). Nút sẽ thực hiện quá trình yêu cầu tham gia mạng (quá trình yêu cầu tham gia mạng được mô tả bởi Algorithm 3) đến khi thành công (nếu bản tin xác thực sai, nút sẽ dừng lại). Sau khi nút nhận được bản tin chấp nhận tham gia vào mạng của Gateway, nút sẽ tiến hành quá trình gửi dữ liệu. Quá trình xử lý của nút được thể hiện bởi Algorithm 2.

Trong quá trình yêu cầu tham gia vào mạng, nút sẽ gửi bản tin xác thực

đến khi nào nhận lại được bản tin phản hồi của Gateway. Khi đó nút sẽ biết được mình đã được chấp thuận tham gia mạng hay chưa. Còn về phần Gateway, sau khi nhận được bản tin xác thực được gửi từ nút, Gateway sẽ kiểm tra bản tin xác thực đó có chính xác không. Kết quả của quá trình kiểm tra bản tin xác thực sẽ được Gateway gửi lại cho nút.

Algorithm 3 RequestJoinNetworks

```

1: procedure REQUESTJOINNETWORKS(idGateway, msgAuthentication)
2:   repeat
3:     state  $\leftarrow$  sendPacket(idGateway, msgAuthentication)     $\triangleright$  state là trạng
   thái của quá trình gửi tin; state = 0 gửi bản tin thành công; state > 0 gửi
   bản tin chưa thành công
4:     if state > 0 then
5:       Delay()
6:     else if Nhận được bản tin hồi đáp từ Gateway then
7:       response  $\leftarrow$  packet_receive.data
8:       if response không chính xác then
9:         state  $\leftarrow$  1
10:      end if
11:    end if
12:  until state > 0
13: end procedure

```

3.4 Xây dựng chi tiết và đánh giá

3.4.1 Bản tin yêu cầu tham gia mạng

Bản tin yêu cầu tham gia mạng sẽ được gửi khi nút đã tự cấu hình. Để giảm số lượng bản tin phải trao đổi giữa nút và Gateway để tham gia mạng, thì bản tin yêu cầu tham gia mạng sẽ chứa luôn bản tin xác thực. Trong quá trình thử nghiệm, chuỗi xác thực sẽ là "@Rsanslab" và được cấu hình sẵn trong firmware của cả nút và Gateway. Gateway sau khi nhận được bản tin yêu cầu tham gia mạng của nút, sẽ tách phần dữ liệu (chứa chuỗi xác thực) để so sánh với chuỗi xác thực của nó rồi tiến hành phản hồi cho nút. Cấu trúc gói yêu cầu tham gia

mạng được mô tả trong bảng 3.2. Trong thí nghiệm, bản tin yêu cầu tham gia

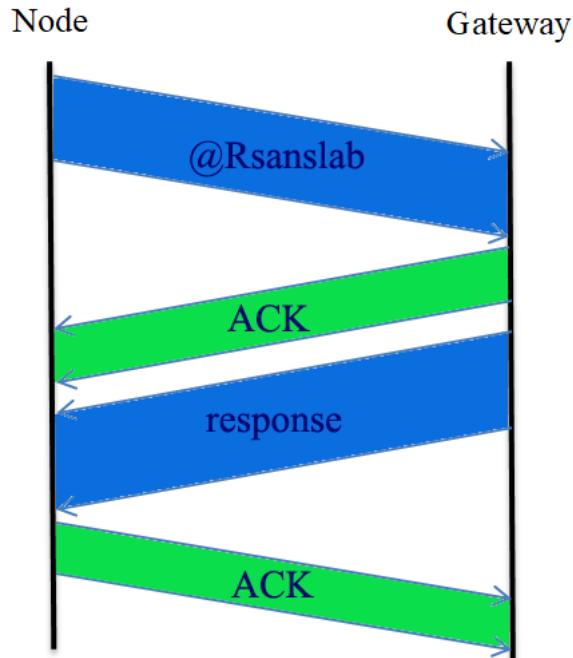
Bảng 3.2: Cấu trúc bản tin yêu cầu tham gia mạng

Header	Data
2 byte	max. 255 byte

mạng tương đối đơn giản mục đích để giảm thời gian xác thực của Gateway. Cấu trúc cụ thể bản tin:

- Header: @R,
- Data: sanslab.

Luồng trao đổi bản tin giữa nút và Gateway sẽ được mô tả trong Hình 3.4. Dữ liệu của bản tin phản hồi chỉ là "1" nếu bản tin xác thực chính xác hoặc là "0" nếu bản tin xác thực không chính xác.



Hình 3.4: Luồng trao đổi dữ liệu trong quá trình yêu cầu tham gia mạng

Thuật toán yêu cầu tham gia mạng sẽ xảy ra 2 trường hợp:

- Trường hợp tốt nhất: Nút gửi yêu cầu tham gia mạng đến Gateway và

nhận luôn được bản tin phản hồi của Gateway. Khi đó độ phức tạp của thuật toán là $O(c)$.

- Trường hợp xấu nhất: Nút gửi yêu cầu tham gia mạng đến Gateway nhưng không nhận được bản tin phản hồi của Gateway (hoặc gửi bản tin không thành công), nút sẽ gửi lại bản tin yêu cầu tham gia mạng. Khi đó độ phức tạp của thuật toán là $O(n)$.

3.4.2 Bản tin gửi dữ liệu

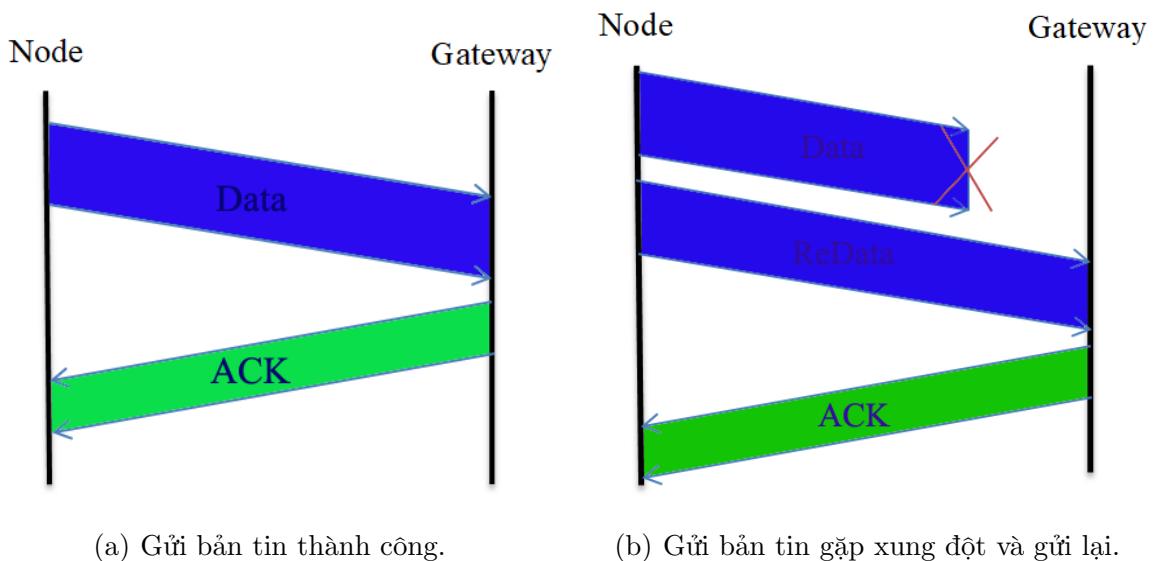
Sau khi đã tham gia mạng thành công, theo chu kỳ, nút bắt đầu gửi bản tin chứa dữ liệu đến Gateway. Do số lượng nút nhỏ (3 đến 4 nút) nên khả năng xảy ra va chạm dẫn đến mất gói là rất ít. Do đó, nguyên nhân chính khiến nút gửi bản tin không thành công là do nút gửi bản tin không đúng vào thời điểm Gateway mở kênh truyền để nhận bản tin. Khi không gửi được bản tin thì nút sẽ trễ một khoảng thời gian, sau đó sẽ gửi lại bản tin. Cũng như bản tin yêu cầu tham gia mạng, bản tin chứa dữ liệu có phần header để nhận diện, ngoài ra bản tin còn chứa các thông số khác. Bản 3.3 mô tả cấu trúc bản tin gửi dữ liệu.

Bảng 3.3: Cấu trúc bản tin dữ liệu

Header	IMEI	Command_ID	Timestamp	Data	End
7 byte	15 byte	1 byte	7 byte	max. 255 byte	2 byte

Luồng trao đổi bản tin dữ nút và Gateway được mô tả trong Hình 3.4.2.

- Header: chứa dấu hiệu nhận diện bản tin chứa dữ liệu ("@DN") và kích thước của bản tin,
- IMEI: là mã nhận dạng giữa các thiết bị với nhau, mỗi thiết bị có một IMEI khác nhau được cấu hình sẵn,
- Command_ID: là loại bản tin,
- Timestamp: chứa thông tin về thời gian của dữ liệu (gồm: ngày, tháng, năm, giờ, phút, giây) được lấy từ vi điều khiển,



Hình 3.5: Luồng dữ liệu trong hai trường hợp

- Data: chứa dữ liệu các thông số môi trường nước, gồm lần lượt các trường sau: pH, Oxy, Salt, Temp, NH_3 , H_2S , NO_2 ,
- End: báo hiệu kết thúc bản tin, là ký tự "\$".

Trong cả hai trường hợp gửi bản tin gấp xung đột và không gấp xung đột độ phức tạp của thuật toán là $O(c)$ do số lần gửi lại của nút không được quá số lần quy định để đảm bảo tính thời gian thực của dữ liệu.

3.5 Kết luận

Sau khi đã xây dựng và đánh giá thuật toán đa truy nhập một cách kỹ lưỡng, thuật toán sẽ được triển khai dưới dạng code và nhúng vào những thiết bị trong hệ thống BKRES-LoRa. Quá trình thử nghiệm cùng với kết quả thử nghiệm sẽ được mô tả trong Chương 4.

Chương 4

Kết quả và đánh giá

Sau khi đã xây dựng thành công thuật toán đã trình bày trong 3, em tiến hành thực nghiệm để phát hiện và điều chỉnh để thuật toán đa truy nhập tốt hơn. Chương này, em xin trình bày về những kết quả đã đạt được và hướng phát triển của đề tài.

4.1 Các kết quả đạt được

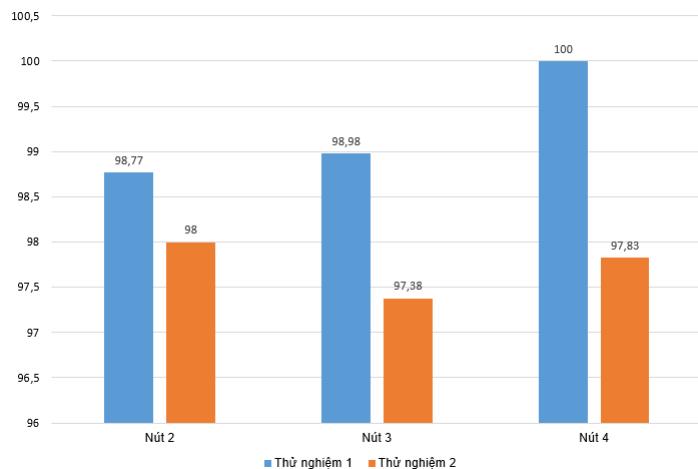
4.1.1 Phân tích kết quả các thử nghiệm

Sau khi tự động cấu hình, nó sẽ gửi yêu cầu tham gia mạng đến Gateway và bắt đầu gửi tin (nếu được Gateway chấp thuận). Hình 4.1 mô tả quá trình kiểm tra yêu cầu tham gia mạng của nút và nhận gói tin từ nút.

```
Received/Sent data
Start configuring LoRa module
LoRa successfully configured

Receive msg
Receive Data form node 9: @DM028|05:13:35 24/05/2018|ID: 354725065508149|Temp: 28.0[oC]|Oxy:0.67[mg/l]|PH:6.8|NH3:0.066[mg/l]|H2S:0.065[mg/l]
|Salt:0.069[]|44
Receive msg
Send accepted Msg to Node 9
Gateway: 1
Receive msg
Receive Data form node 9: @DM028|05:10:02 24/05/2018|ID: 354725065508131|Temp: 28.0[oC]|Oxy:0.67[mg/l]|PH:6.8|NH3:0.066[mg/l]|H2S:0.065[mg/l]
|Salt:0.069[]|44
Receive msg
Receive Data form node 9: @DM028|05:15:08 24/05/2018|ID: 354725065508149|Temp: 28.0[oC]|Oxy:0.67[mg/l]|PH:6.8|NH3:0.066[mg/l]|H2S:0.065[mg/l]
|Salt:0.069[]|44
Receive msg
Receive Data form node 9: @DM028|05:10:14 24/05/2018|ID: 354725065508131|Temp: 28.0[oC]|Oxy:0.67[mg/l]|PH:6.8|NH3:0.066[mg/l]|H2S:0.065[mg/l]
|Salt:0.069[]|44
Receive msg
Receive Data form node 9: @DM028|05:15:20 24/05/2018|ID: 354725065508149|Temp: 28.0[oC]|Oxy:0.67[mg/l]|PH:6.8|NH3:0.066[mg/l]|H2S:0.065[mg/l]
|Salt:0.069[]|44
Receive msg
Receive Data form node 9: @DM028|05:10:27 24/05/2018|ID: 354725065508131|Temp: 28.0[oC]|Oxy:0.67[mg/l]|PH:6.8|NH3:0.066[mg/l]|H2S:0.065[mg/l]
|Salt:0.069[]|44
Receive msg
Receive Data form node 9: @DM028|05:15:33 24/05/2018|ID: 354725065508149|Temp: 28.0[oC]|Oxy:0.67[mg/l]|PH:6.8|NH3:0.066[mg/l]|H2S:0.065[mg/l]
|Salt:0.069[]|44
Receive msg
Receive Data form node 9: @DM028|05:10:39 24/05/2018|ID: 354725065508131|Temp: 28.0[oC]|Oxy:0.67[mg/l]|PH:6.8|NH3:0.066[mg/l]|H2S:0.065[mg/l]
|Salt:0.069[]|44
Receive msg
Receive Data form node 9: @DM028|05:15:46 24/05/2018|ID: 354725065508149|Temp: 28.0[oC]|Oxy:0.67[mg/l]|PH:6.8|NH3:0.066[mg/l]|H2S:0.065[mg/l]
|Salt:0.069[]|44
```

Hình 4.1: Quá trình xử lý của Gateway



Hình 4.2: Kết quả thí nghiệm đánh giá độ mất gói

4.1.1.1 Thử nghiệm 1: Đánh giá măt gói trong điều kiện thuận lợi (PTN)

Thí nghiệm này được triển khai tại phòng nghiên cứu Sanslab, thí nghiệm thực hiện gồm có 3 nút và 1 gateway. Các thông số kỹ thuật được cấu hình như trong Bảng 3.1, các nút nằm trong bán kính 10 m của Gateway. Ngoài ra có hai số thông số được thiết lập trước là ID của nút và ID của Gateway. Kết quả của thực nghiệm được mô tả trong Bảng 4.1 và biểu đồ Hình 4.2.

Bảng 4.1: Bảng kết quả độ măt gói trong điều kiện thuận lợi

	Nút 2	Nút 3	Nút 4
Gửi gói tin thành công	98,77,02 %	98,98 %	100 %
Hủy gói tin	1,23 %	1,03 %	0 %

4.1.1.2 Thử nghiệm 2: Đánh giá độ măt gói trong điều kiện thách thức

Thí nghiệm 2 được thực hiện tại khuôn viên Đại học Bách Khoa Hà Nội, các thông số kỹ thuật của thử nghiệm 2 được thiết lập giống với thử nghiệm 1. Trong thử nghiệm này, Gateway được đặt tại tầng 6 thư viện Tạ Quang Bửu, nút đặt rải rác trong khuôn viên trường với môi trường có vật cản. Kết quả thử nghiệm

2 được ghi lại trong Bảng 4.2 và được biểu diễn trên Hình 4.2.

Bảng 4.2: Bảng quả độ mất gói trong điều kiện thách thức

	Nút 2	Nút 3	Nút 4
Gửi bản tin thành công	98,0 %	97,38 %	97,83 %
Hủy bản tin	2,0 %	2,62 %	2,17 %

4.1.1.3 Đánh giá kết quả thử nghiệm

Trong cả 2 thử nghiệm, độ mất gói là rất nhỏ (nhỏ hơn 3%) chứng tỏ thuật toán đa truy nhập hoạt động tốt với số lượng nút nhỏ (3 nút). Bên cạnh đó, tỷ lệ mất gói của thử nghiệm 2 cao hơn thử nghiệm 1, cho thấy được khoảng cách và vật cản có ảnh hưởng đến độ mất gói trong quá trình truyền dữ liệu.

4.1.2 Hệ thống BKRES-LoRa

Công nghệ LoRa đã được phòng nghiên cứu Sanslab áp dụng vào hệ thống BKRES thành công. Module truyền thông được tích hợp vào kit điều khiển (hình 4.3) với kích thước gọn nhẹ, tiêu tốn ít năng lượng (module sử dụng Pin để cung cấp năng lượng).

Những bản tin Gateway nhận được sẽ được gửi đến server và những dữ liệu đó được hiển thị các ứng dụng (hình 4.4).

4.2 Đánh giá thuật toán và định hướng phát triển

Thuật toán đa truy nhập có những ưu điểm và nhược điểm sau:

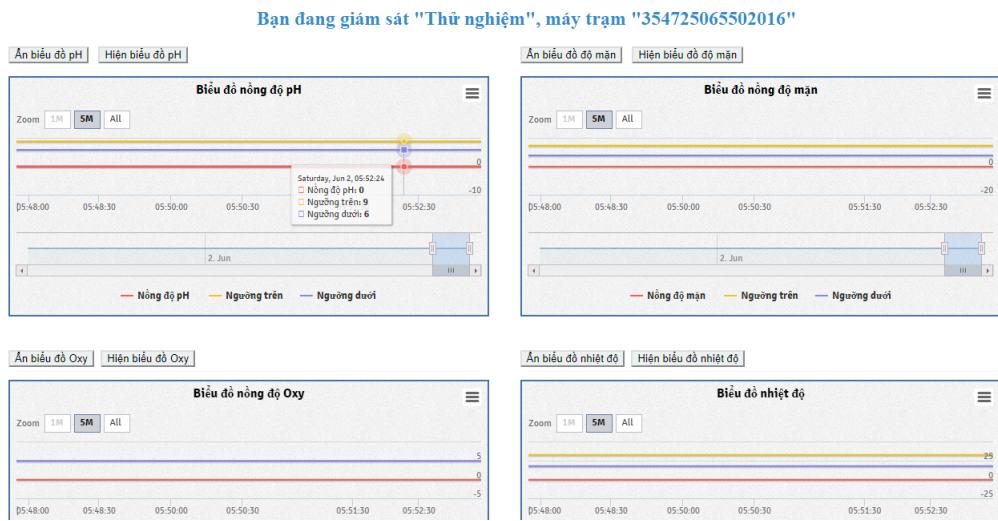
- Ưu điểm:
 - Thuật toán hoạt động tốt,



Hình 4.3: Kit điều khiển BKRES-LoRa

- Hiệu năng truyền gói thành công cao hơn nhiều so với các giao thức truy nhập ngẫu nhiên khác,
- Các thiết bị sử dụng thuật toán hoạt động ổn định, tiêu thụ ít năng lượng (so với sử dụng module sim).
- Nhược điểm:
 - Chưa xác định được độ ảnh hưởng của số lượng thiết bị đến thuật toán,
 - Chưa xác định được các thời gian tiêu tốn để thực hiện các quá trình (tham gia mạng, gửi dữ liệu) để điều chỉnh trễ hợp lý,
 - Quá trình yêu cầu tham gia mạng phức tạp, ảnh hưởng đến quá trình gửi tin của nút khác,
 - Năng lượng tiêu thụ của Gateway lớn (vì Gateway không có khoảng thời gian để nghỉ),
 - Hầu như chưa có sự quản lý của Gateway đối với các nút trong mạng.

Thuật toán cần được phát triển hơn nữa để có thể tăng hiệu năng, độ ổn định của những thiết bị trong mạng và áp dụng cho mạng có số lượng nút lớn. Ngoài ra để tăng khoảng cách truyền, cần phát triển giao thức áp dụng cho mô hình mạng đa chặng để giảm số lượng Gateway trong mạng.



Hình 4.4: Hiển thị dữ liệu trên ứng dụng web

4.3 Kết luận

Qua kết quả của những thử nghiệm đã được tiến hành, chứng tỏ thuật toán đa truy nhập hoạt động tốt với số lượng nút nhỏ (3 nút), do điều kiện về số lượng thiết bị giới hạn nên em chưa xác định được kết quả hoạt động của thuật toán đa truy nhập trong mạng có số lượng nút lớn hơn. Ngoài ra, thuật toán đa truy nhập đã được thích hợp vào hệ thống BKRES-LoRa thành công.

Kết luận

Sản phẩm của đồ án "**Thiết kế, phát triển giao thức đa truy nhập LoRa, tích hợp vào mạng WSN giám sát môi trường**" đã được thử nghiệm và tích hợp vào hệ thống BKRES-LoRa. Hệ thống BKRES-LoRa chuẩn bị được thực nghiệm trong môi trường thực tế. Nhờ đó, thuật toán đa truy nhập có thể phát hiện những hạn chế còn tồn tại và sẽ được sửa chữa, cải thiện thêm. Trong tương lai, em tin hệ thống BKRES-LoRa sẽ được cải tiến ngày một tốt hơn để ngoài phục vụ cho lĩnh vực nghiên cứu, nó có thể trở thành một sản phẩm thương mại. Từ đó, hỗ trợ con người trong việc giám sát các thông số môi trường nói chung, cũng như giúp người nông dân trong quá trình nuôi tôm nói riêng.

Trong quá trình nghiên cứu, xây dựng và thiết kế sản phẩm, em đã vấp phải nhiều khó khăn. Nhưng nhờ có sự hướng dẫn tận tình của TS. Trần Quang Vinh cũng như sự giúp đỡ của các thành viên trong phòng nghiên cứu Sanslab đã giúp em hoàn thiện đồ án này. Qua đó, giúp em củng cố thêm những kiến thức về phần cứng, lập trình; những kỹ năng làm việc nhóm, tìm hiểu nghiên cứu những vấn đề mới; cũng như có được những kiến thức mới và trải nghiệm thực tế.

Tài liệu tham khảo

- [1] “11 trang web thương mại điện tử hàng đầu Việt Nam 2017.” [Online]. Available: <https://giaidieu.com/blog/11-trang-web-thuong-mai-dien-tu-hang-dau-viet-nam-2017>. [Accessed: 25-Dec-2018].
- [2] “Introduction - The complete C# tutorial.” [Online]. Available: <https://csharp.net-tutorials.com/getting-started/introduction/>. [Accessed: 25-Dec-2018].
- [3] “jQuery API Documentation.” [Online]. Available: <https://api.jquery.com/>. [Accessed: 25-Dec-2018].
- [4] “NET Framework Architecture Mono.” [Online]. Available: <https://www.mono-project.com/archived/net-framework-architecture/>. [Accessed: 25-Dec-2018].
- [5] U. Noreen, A. Bounceur và L. Clavier, "A Study of LoRa Low Power and Wide Area Network Technology", 3rd International Conference on Advanced Technologies for Signal and Image Processing - ATSIP, Fez, Morroco, May 22-24, 2017.
- [6] Andrew S. Tanenbaum, David J. Wetherall, "*Computer Network*", Prentice Hall, 2010.
- [7] Module LoRa Ra-02 SX1278 datasheet (<https://www.en.ai-thinker.com>).
- [8] SX1276/77/78 datasheet (<https://www.semtech.com>).
- [9] STM32F1xC, STM32F1xD, STM32F1xE datasheet (<https://www.st.com>).
- [10] STM32Cube initialization code generator.

Phụ lục

Hàm main của Gateway

```
int main(void)
{
    /* Reset of all peripherals, Initializes the Flash interface and the Systick. */
    HAL_Init();

    /* Configure the system clock */
    SystemClock_Config();

    /* Initialize all configured peripherals */
    MX_GPIO_Init();
    MX_DMA_Init();
    MX_SPI3_Init();
    MX_USART1_UART_Init();

    /* USER CODE BEGIN 2 */
    SX1278_Config();
    /* USER CODE END 2 */

    while (1)
    {
        uint8_t e = receivePacketTimeoutACK (&hspi3, 10000);
        if (!e){
            UART_print ("\nReceive msg");
            char msg[40] = {0};
            for (uint8_t i = 0; i < packet_received.length; i++){
                msg[i] = (char)packet_received.data[i];
            }
            int ID = packet_received.src;
            if (msg[1] == 'D'){
                char title[100];
                sprintf(title, "\nReceive Data form node %d: ", ID);
                UART_print(title);
                UART_print(msg);
            }
            else checkNode(ID, msg);
        }
    }
}
```

```
    else UART_print ("\nDon't have msg");
}
}
```

Hàm kiểm tra bản tin xác thực của Gateway

```
void checkNode (int nodeAddr, char msg[]){
    char title[100];
    sprintf(title, "\nSend accepted Msg to Node %d", nodeAddr);
    char *response;
    uint8_t e;
    UART_print(title);
    if (msg[2] == 's' && msg[3] == 'a' && msg[4] == 'n' && msg[5] == 's' &&
        msg[6] == 'l' && msg[7] == 'a' && msg[8] == 'b'){
        response = "1";
        UART_print ("\nGateway: 1");
        arrNodeAddr[numberofNode++] = nodeAddr;
    }
    else{
        response = "0";
        UART_print ("\nGateway: 0");
    }
    uint8_t cnt;
    do{
        e = sendPacketTimeoutACKRetries(&hspi3, nodeAddr, response);
        cnt++;
        HAL_Delay (5);
    }
    while (e > 0 && cnt < 3);
}
```

Hàm main của nút

```
int main(void)
{
    /* MCU Configuration-----*/
    /* Reset of all peripherals, Initializes the Flash interface and the Systick. */
    HAL_Init();

    /* Configure the system clock */
    SystemClock_Config();
    /* Initialize all configured peripherals */
    MX_GPIO_Init();
    MX_DMA_Init();
    MX_SPI3_Init();
    MX_USART1_UART_Init();
    /* USER CODE BEGIN 2 */
    SX1278_Config();
    /* USER CODE END 2 */
    delay = NODE_ADDRESS * 10;
    requestJoinNetwork(GWAY_ADDRESS);
    while (1)
    {
        /* USER CODE END WHILE */
        if (joinedNetwork == 1)
            sendData(GWAY_ADDRESS);
        HAL_Delay(15000);
        /* USER CODE BEGIN 3 */
    }
}
```

Hàm yêu cầu tham gia mạng của nút

```

void requestJoinNetwork (uint8_t GWaddr)
{
    char msgRQ[20] = "@Rsanslab";
    uint8_t e;
    do{
        e = sendPacketTimeoutACK (&hspi3, GWaddr , msgRQ);
        if (e == 0){
            UART_print ("\nSend msg_request successful, waiting
msg_acception");
            uint8_t state = receivePacketTimeoutACK (&hspi3, 10000);
            char msgAccept[10] = {0};
            if (state == 0 && packet_received.src == GWAY_ADDRESS){
                for (uint8_t i = 0; i < packet_received.length; i++){
                    msgAccept[i] = (char) packet_received.data[i];
                }
                if (msgAccept[0] == '1'){
                    UART_print ("\nGateway accepted");
                    joinedNetwork = 1;
                }
                else{
                    UART_print ("\nGateway didn't accept");
                    joinedNetwork = 2;
                }
                //delay = NODE_ADDRESS * 10;
            }
        }
        else{
            UART_print("\nSend msg_request nonsuccessful");
            //int x = rand() % 10;
            HAL_Delay (1000);
            //delay *= 2;
        }
    }
    while (e);
}

```

Hàm gửi dữ liệu của nút

```

void sendData (uint8_t GWAddr){
    char msgData[100] =
        "@DN045<~1Gq!J1,UaD2)R'IOf(0='UY5E$cWE>!!!!d!!!!g!!!!f!!!!c!!!!b!!!!~>";
    //sprintf(msgData, "", counter);
    delay = 10 * NODE_ADDRESS;
    uint8_t e;
    uint8_t cnt = 0;
    do{
        e = sendPacketTimeoutACK(&hspi3, GWAddr, msgData);
        cnt_sent++;
        cnt++;
        if (e > 0){
            //int x = rand() % 10;
            HAL_Delay (delay);
            if (delay *2 < 2000) delay *= 2;
        }
        else
            counter++;
    }
    while(e && cnt < 5);
    char tmp[100];
    if (e == 0){
        sprintf(tmp, "\nNode %d :received %d, retries %d, deleted %d, sent
%d", NODE_ADDRESS, counter, cnt, cnt_huy_goi, cnt_sent);
        UART_print(tmp);
    }
    else
        cnt_huy_goi++;
}

```
