**Mục lục**

[**PHẦN 1 - CƠ SỞ LÝ THUYẾT** 2](#_Toc75094261)

[**1.** **Giới thiệu chung về lập trình mạng máy tính** 2](#_Toc75094262)

[**1.1** **Khái quát chung về mạng máy tính** 2](#_Toc75094263)

[**1.2** **Phân loại mạng máy tính** 4](#_Toc75094264)

[**1.3** **Bản chất Lập Trình Mạng** 5](#_Toc75094265)

[**1.4** **Giao thức TCP và giao thức UDP** 6](#_Toc75094266)

[**1.5** **Ngôn ngữ C#** 7](#_Toc75094267)

[**1.6** **Ứng dụng Visual Studio** 8](#_Toc75094268)

[**2.** **Kiến thức cơ bản về Socket** 9](#_Toc75094269)

[**3.** **Sử dụng Mail Service thông qua giao thức SMTP** 10](#_Toc75094270)

[**4.** **Định dạng thông thường của một gói tin** 11](#_Toc75094271)

[**5.** **CSMA/CD – Phương thức đa truy cập phát hiện xung đột** 12](#_Toc75094272)

[**PHẦN 2 - THIẾT KẾ VÀ XÂY DỰNG ỨNG DỤNG** 13](#_Toc75094273)

[**1.** **Tổng quan về đề tài** 13](#_Toc75094274)

[**2.** **Khái quát chung về ứng dụng** 14](#_Toc75094275)

[**2.1** **Chức năng** 14](#_Toc75094276)

[**2.2** **Xây dựng và thiết kế chi tiết của ứng dụng** 15](#_Toc75094277)

[**a)** **FormMain** 15](#_Toc75094278)

[**b)** **FormLogin** 19](#_Toc75094279)

[**c)** **FormRegister** 21](#_Toc75094280)

[**d)** **FormExamPrepare** 22](#_Toc75094281)

[**e)** **FormQuestion** 23](#_Toc75094282)

[**f)** **FormExamQues** 24](#_Toc75094283)

[**g)** **FormExamQuestion** 25](#_Toc75094284)

[**h)** **FormPrepareThi** 26](#_Toc75094285)

[**i)** **FormThi** 27](#_Toc75094286)

[**j)** **FormResult** 30](#_Toc75094287)

[**3.** **Tổng kết** 32](#_Toc75094288)

[**4.** **Danh mục tài liệu tham khảo** 33](#_Toc75094289)

# **PHẦN 1 - CƠ SỞ LÝ THUYẾT**

1. **Giới thiệu chung về lập trình mạng máy tính**
2. **Khái quát chung về mạng máy tính**

- Tập hợp các máy tính kết nối với nhau dựa trên một kiến trúc nào đó để có thể trao đổi dữ liệu

* Máy tính: máy trạm, máy chủ, bộ định tuyến
* Kết nối bằng một phương tiện truyền
* Theo một kiến trúc mạng

- Phương tiện truyền, đường truyền vật lý:

* Hữu tuyến: cáp đồng, cáp quang
* Vô tuyến: sóng hồng ngoại, sóng radio

- Kiến trúc mạng:

* Hình trạng mạng: cách thức các máy tính kết nối bằng đường truyền vật lý với nhau
* Giao thức mạng: xác định cách thức các máy tính trao đổi dữ liệu với nhau như thế nào

- Hoạt động cơ bản trên hệ thống mạng máy tính: truyền thông tin từ máy tính này sang máy tính khác

* Tương tự như con người trao đổi thư tín qua hệ thống bưu điện
* Máy nguồn: gửi dữ liệu
* Máy đích: nhận dữ liệu

- Phân tầng

* Mỗi tầng có thể có một hoặc nhiều chức năng
* Triển khai dịch vụ để thực hiện các chức năng: cung cấp dịch vụ cho tầng trên, sử dụng dịch vụ tầng dưới, độc lập với các tầng còn lại
* Mỗi dịch vụ có thể có một hoặc nhiều cách triển khai khác nhau, cho phép tầng trên lựa chọn dịch vụ phù hợp
* Lợi ích: dễ dàng thiết kế, triển khai; Dễ dàng tái sử dụng; Dễ dàng nâng cấp

- Truyền thông trong kiến trúc phân tầng

* Bên gửi: thêm tiêu đề chứa thông tin phục vụ cho việc xử lý dữ liệu tại tầng tương ứng và chuyển cho tầng dưới (Đóng gói dữ liệu – Encapsulation)
* Bên nhận: xử lý dữ liệu theo thông tin trong phần tiêu đề, tách tiêu đề và chuyển dữ liệu cho tầng trên

- Chồng giao thức (Protocol stack)

* Giao thức: Là tập hợp các quy tắc quy định khuôn dạng, ngữ nghĩa, thứ tự các thông điệp được gửi và nhận giữa các nút mạng và các hành vi khi trao đổi các thông điệp đó
* Mỗi tầng có nhiều cách thức để thực hiện các chức năng sinh ra các giao thức khác nhau

🡺chồng giao thức: ngăn xếp các giao thức truyền thông trên kiến trúc phân tầng

- Định danh trên kiến trúc phân tầng:

* Tầng ứng dụng : tên miền định danh cho máy chủ cung cấp dịch vụ

Tên miền: chuỗi ký tự dễ nhớ với người dùng. Thiết bị mạng không dùng tên miền khi truyền tin

Ví dụ: mps.gov.vn (máy chủ Web của Bộ CA)

* Tầng giao vận: số hiệu cổng định danh cho các dịch vụ khác nhau

Số hiệu cổng: từ 0-65535

Ví dụ: Web-80, DNS-53, Email(SMTP-25, POP-110, IMAP-143)

* Tầng mạng: địa chỉ IP định danh cho các máy trạm, máy chủ, bộ định tuyến

Có thể dùng trong mạng nội bộ và mạng Internet

Địa chỉ IPv4: 4 số có giá trị từ 0-255, các nhau bởi 1 dấu ‘.’

Ví dụ: 123.30.9.222 (máy chủ Web của Bộ CA)

* Tầng liên kết dữ liệu: địa chỉ MAC định danh cho các máy trạm, máy chủ, thiết bị mạng

Chỉ dùng trong mạng nội bộ

1. **Phân loại mạng máy tính**

- Mạng cá nhân (PAN – Personal Area Network)

* Phạm vi kết nối: vài chục mét
* Số lượng người dùng: một vài người dùng
* Thường phục vụ cho cá nhân

- Mạng cục bộ (LAN – Local Area Network):

* Phạm vi kết nối: vài ki-lô-mét
* Số lượng người dùng: một vài đến hàng trăm nghìn
* Thường phục vụ cho cá nhân, hộ gia đình, tổ chức

- Mạng đô thị (MAN – Metropolitian Area Network)

* Phạm vi kết nối: hàng trăm ki-lô-mét
* Số lượng người dùng: hàng triệu
* Phục vụ cho thành phố, khu vực

- Mạng diện rộng (WAN – Wide Area Network)

* Phạm vi kết nối: vài nghìn ki-lô-mét
* Số lượng người dùng: hàng tỉ

- GAN – Global Area Network: phạm vi toàn cầu (Ví dụ: Internet)

- Mạng internet và mạng máy tính không giống nhau. Theo đó, mạng internet là mạng kết nối một số các máy tính đang hoạt động lại cùng nhau. Để truy cập vào mạng internet, cần có một địa chỉ IP công khai (IP Public).

- Mạng internet tồn tại được gồm có cả các thiết bị máy tính, thiết bị truy cập internet và máy chủ. Theo đó, công dụng lớn nhất của mạng internet chính là truy cập vào tin tức và nguồn thông tin kiến thức khổng lồ.

- Một điểm khác nhau nữa là một người dùng mạng máy tính chắc chắn sẽ cần kết nối và khả năng trao đổi trực tiếp giữa 2 máy tính. Tuy nhiên, đối với mạng internet, việc trao đổi có thể thực hiện bằng cách trực tiếp hoặc thông qua máy chủ web.

1. **Bản chất Lập Trình Mạng**

- Lập trình mạng (hay còn gọi là lập trình socket) là nền tảng về kỹ thuật để phát triển các ứng dụng hoạt động trên mạng (LAN hoặc Internet)

- .NET Framework cũng có những hỗ trợ khác cho lập trình mạng như:

* Giao diện luồng dữ liệu (Stream)
* Trình tự hóa dữ liệu (Serialization)
* Biến đổi dữ liệu (Data Conversion)
* Lập trình bất đồng bộ (Asynchronous Programming)
* Lập trình đa luồng (Multi-Threading Programming)
* Tạo bộ đệm (Catching)
* Bảo mật (Socket Security, Cryto Stream)

- Kiến thức về phân tích thiết kế hệ thống, kiến thức hệ thống mạng, mô hình xây dựng chương trình ứng dụng mạng, kiến thức về cơ sở dữ liệu…

- Mô hình lập trình mạng: Đây là một phần không thể thiếu trong lập trình mạng. Hiện nay, các kiểu mô hình được sử dụng rộng rãi nhất là mô hình OSI, mô hình Client/Server, TCP/IP.

- Kiến thức mạng truyền thông: Để có thể lập trình mạng, bạn cần nắm vững các kiến thức về mạng truyền thông, bao gồm hệ thống mạng truyền thông số, truyền bit nối tiếp…

- Ngôn ngữ lập trình mạng: Đây là một tập con của ngôn ngữ máy tính. Nó có dạng số nhị phân, được thiết kế và chuẩn hóa để truyền chỉ thị cho máy tính chứa bộ xử lý. Các ngôn ngữ lập trình phổ biến nhất bao gồm Java, .Net, C/C++, Delphi, Javascript…

1. **Giao thức TCP và giao thức UDP**

|  |  |
| --- | --- |
| Giao thức TCP | Giao thức UDP |
| Giao thức hướng kết nối | Giao thức hướng gói tin |
| Phải có sự đồng thuận kết nối giữa 2 bên | Không cần sự đồng thuận kết nối |
| Tạo ra 1 stream, vào 1 đầu và ra 1 đầu, tốn kém tài nguyên | Lấy dữ liệu đóng gói, dán nhãn, rồi truyền đi bất kỳ |
| Nếu không bị hư hỏng, thì đảm bảo không mất gói | Có thể bị mất gói |
| Chậm hơn | Nhanh hơn |
| Header 20 bytes | Header 8 bytes |
| Gửi đi có cơ chế báo nhận | Không có cơ chế báo lỗi |
| Có khả năng đa check lỗi | Không có cơ chế báo gửi nhận, có thể bị nhiễu do từ trường |

- Điểm giống nhau giữa 2 giao thức: TCP và UDP đều là các giao thức được sử dụng để gửi các bit dữ liệu - được gọi là các gói tin - qua Internet. Cả hai giao thức đều được xây dựng trên giao thức IP. Nói cách khác, dù bạn gửi gói tin qua TCP hay UDP, gói này sẽ được gửi đến một địa chỉ IP. Những gói tin này được xử lý tương tự bởi vì chúng được chuyển tiếp từ máy tính của bạn đến router trung gian và đến điểm đích.

- Cách thức hoạt động của TCP: TCP là giao thức được sử dụng phổ biến nhất trên Internet. Khi bạn yêu cầu một trang web trong trình duyệt, máy tính sẽ gửi các gói tin TCP đến địa chỉ của máy chủ web, yêu cầu nó gửi lại trang web. Máy chủ web phản hồi bằng cách gửi một luồng các gói tin TCP, mà trình duyệt web của bạn kết hợp với nhau để tạo thành trang web. Khi click vào liên kết, đăng nhập, đăng nhận xét hoặc làm bất kỳ điều gì khác, trình duyệt web của bạn sẽ gửi gói tin TCP tới máy chủ và máy chủ gửi lại các gói tin cho TCP.

- Cách thức hoạt đông của UDP: Giao thức UDP hoạt động tương tự như TCP, nhưng nó bỏ qua quá trình kiểm tra lỗi. Khi một ứng dụng sử dụng giao thức UDP, các gói tin được gửi cho bên nhận và bên gửi không phải chờ để đảm bảo bên nhận đã nhận được gói tin, do đó nó lại tiếp tục gửi gói tin tiếp theo. Nếu bên nhận bỏ lỡ một vài gói tin UDP, họ sẽ mất vì bên gửi không gửi lại chúng. Do đó thiết bị có thể giao tiếp nhanh hơn.

1. **Ngôn ngữ C#**

- C# là một ngôn ngữ lập trình **đơn giản, hiện đại, mục đích tổng quát, hướng đối tượng** được phát triển bởi Microsoft và được phê chuẩn bởi European Computer Manufacturers Association (ECMA) và International Standards Organization (ISO).

- C# được thiết kế cho Common Language Infrastructure (CLI), mà gồm Executable Code và Runtime Environment, cho phép chúng ta sử dụng các ngôn ngữ high-level đa dạng trên các nền tảng và cấu trúc máy tính khác nhau.

- C# là một ngôn ngữ lập trình hướng đối tượng được phát triển bởi Tập đoàn Microsoft. Ngôn ngữ này được xem là ngôn ngữ lập trình hướng đối tượng trong sáng và thuần nhất.

- Là ngôn ngữ được Microsoft phát triển dựa trên 2 ngôn ngữ huyền thoại đó là C++ và Java. Và nó cũng được miêu tả là loại ngôn  ngữ có được sự cân bằng giữa C++, Visual Basic, Delphi và Java.

- C# làm việc chủ yếu trên bộ khung .NET (.NET framework)

-Bộ khung .NET (.Net Framework – được phát âm là Dot Net) là một nền tảng thực thi ứng dụng chủ yếu trên hệ điều hành Microsoft Windows được phát triển bởi Microsoft. Nó cho phép các lập trình viên sử dụng lại một số mã thường dùng trong các ứng dụng của họ. Mà không phải lo lắng về việc gõ đi gõ lại nhiều lần.

- Các đặc điểm chỉnh của ngôn ngữ C#:

* **C# là ngôn ngữ hiện đại, xử lý ngoại lệ, thu gom bộ nhớ tự động, những kiểu dữ liệu mở rộng, và bảo mật mã nguồn**
* **C# là ngôn ngữ hướng đối tượng*,*** những đặc điểm chính của ngôn ngữ hướng đối tượng (Object-oriented language) là sự đóng gói (**Encapsulation**), sự kế thừa (**Inheritance**), và đa hình (**Polymorphism**)
* **C# là ngôn ngữ mạnh mẽ và mềm dẻo*,*** ngôn ngữ này không đặt những ràng buộc lên những việc có thể làm.
* **C# là ngôn ngữ có ít từ khóa,** C# là ngôn ngữ sử dụng giới hạn những từ khóa. Phần lớn các từ khóa được sử dụng để mô tả thông tin
* **C# là ngôn ngữ hướng module,** Mã nguồn C# có thể được viết trong những phần được gọi là những lớp, những lớp này chứa các phương thức thành viên của nó
* Nó là một phần của .Net Framework

1. **Ứng dụng Visual Studio**

- Là con đẻ của Microsoft, Visual Studio được biết đến như một phần mềm lập trình website nổi tiếng có một không hai và đến nay chưa có công cụ nào có thể thay thế được chúng

- Visual Studio được lập trình bằng ngôn ngữ VB+ và C# - vốn là các ngôn ngữ lập trình cho phép người dùng xây dựng hệ thống website dễ dàng và nhanh chóng nhất

- Với lợi thế được xây dựng và phát triển trực tiếp từ ông lớn Microsoft, Visual Studio từ khai khai sinh cho đến nay đã được cập nhật rất nhiều phiên bản sử dụng khác nhau. Điều này cho phép người dùng thoải mái lựa chọn phiên bản tốt nhất, có cấu hình phù hợp nhất với đa dạng dòng máy của mình.

- Visual Studio được sáng tạo bởi Microsoft. Vào năm 1997, ông lớn này đã cho ra mắt hai phiên bản của Visual Studio là Professional  và Enterprise. Trải qua nhiều lần phát triển, đến nay, Visual Studio sở hữu nhiều phiên bản khác nhau như VS 2005, VS 2008, VS 2010, VS 2012, VS 2013.

- Trong đó, phiên bản mới nhất là Visual Studio 2015 được chính thức phát hành vào tháng 7 năm 2015. Theo đó, với mỗi phiên bản phát hành của mình, Visual Studio lại tích hợp thêm nhiều tính năng, công nghệ mới mẻ và hấp dẫn hơn.

-Một số tính năng quan trọng của Visual Studio:

* Hỗ trợ đa nền tảng
* Hỗ trợ nhiều ngôn ngữ lập trình
* Biên tập mã
* Thiết kế
* Cung cấp kho tiện ích mở rộng
* Bảo mật kho lưu trữ hiệu quả
* Lưu trữ dữ liệu ở dạng phân cấp
* Hỗ trợ Git

1. **Kiến thức cơ bản về Socket**

- Socket chính là một điểm cuối (end-point) của liên kết truyền thông hai chiều (two-way communication) giữa hai chương trình chạy trên mạng. Các lớp Socket được sử dụng để biểu diễn kết nối giữa client và server, được ràng buộc với một cổng port để các tầng TCP có thể định danh ứng dụng mà dữ liệu sẽ được gửi tới

- Lập trình Socket là lập trình cho phép các người dùng kết nối các máy tính truyền tải và nhận dữ liệu từ máy tính thông qua mạng

- Socket chính là cửa giao tiếp giữa tầng ứng dụng và tầng giao vận (Transport layer). Nói cách khác, Socket là giao diện do ứng dụng tạo ra trên máy trạm, quản lí bởi hệ điều hành qua đó các ứng dụng có thể gửi/nhận thông điệp đến/từ các ứng  dụng khác

- Socket sẽ được ràng buộc với một mã số cổng (Port Number) để giúp tầng giao vận định danh được ứng dụng nhận/gửi thông điệp.

- Các bước sau xảy ra khi thành lập một kết nối TCP giữa hai máy tính sử dụng Socket:

* Server khởi tạo một đối tượng ServerSocket, biểu thị số hiệu cổng (port) nào để xuất hiện giao tiếp
* Server gọi phương thức accept() của lớp ServerSocket. Phương thức này đợi tới khi một Client kết nối với Server trên cổng đã cho
* Sau khi Server đang đợi, một Client khởi tạo một đối tượng Socket, xác định tên Server và số hiệu cổng để kết nối tới
* Constructor của lớp Socket cố gắng để kết nối Client tới Server và số hiệu cổng đã xác định. Nếu giao tiếp được thành lập, bây giờ Client có một đối tượng Socket có khả năng giao tiếp với Server
* Trên Server-side, phương thức accept() trả về một tham chiếu tới một socket mới trên Server mà được kết nối với socket của Client

- Với UDP socket, chúng ta không cần thiết lập liên kết “handshaking” giữa client và server trước khi truyền thông điệp, thông điệp được gửi đi một cách độc lập. Bên gửi sẽ chỉ rõ địa chỉ IP và số hiệu cổng (port number) của bên nhận. Sau khi thông điệp được truyền đến bên nhận, bện nhận có thể dựa vào địa chỉ IP và số hiệu cổng tương ứng của bên gửi được gắn trên gói tin để gửi lại response

1. **Sử dụng Mail Service thông qua giao thức SMTP**

- SMTP (Simple Mail Transfer Protocol - Giao thức truyền thư đơn giản) là một phần của tầng ứng dụng thuộc giao thức TCP/IP. Nó là một chuẩn Internet cho việc truyền thư điện tử (email).  Mặc định cổng TCP được SMTP sử dụng là cổng 25 và nếu SMTP được kết nối được bảo mật với SSL, hay được biết đến là SMTPS sẽ sử dụng cổng 465 làm mặc định.

- SMTP cung cấp một bộ giao thức đơn giản hóa việc giao tiếp email giữa các Email server. Hầu hết các tên máy chủ SMTP được viết dưới dạng tên miền như "smtp.domain.com" hoặc "mail.domain.com".

- Xác thực SMTP (SMTP Authentication hoặc SMTP AUTH), là một phần mở rộng của SMTP. Theo đó, các client (có thể hiểu là những phần mềm sử dụng email hoặc là chương trình mà bạn tự viết) có thể đăng nhập bằng cơ chế xác thực được SMTP server hỗ trợ

- Và điều kiện bắt buộc là phải Cho phép Quyền truy cập của ứng dụng kém an toàn trong thiết lập tài khoản của Google

- Framework Microsoft .Net cung cấp namespace là System.Net và System.Net.Sockets để phục vụ cho các giao thức mạng các mà ứng dụng có thể sử dụng để gửi hoặc nhận dữ liệu qua mạng.

- C# sử dụng namespace System.Net.Mail để làm việc với email. Mình có viết một đoạn code mẫu dưới đây để minh họa làm sao để gửi email từ tài khoản Gmail. SMTP của Gmail là smtp.gmail.com và sử dụng cổng 587 để gửi email.

1. **Định dạng thông thường của một gói tin**

- Field là 1 vùng có cấu trúc, hình dạng rõ ràng

- Một Frame gói tin gồm các Field sau:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| A | B | C | D | E | F |
| Start Frame Field | Address Field | Type/Length Field | Data Field | FCS Field | Stop Frame Field |

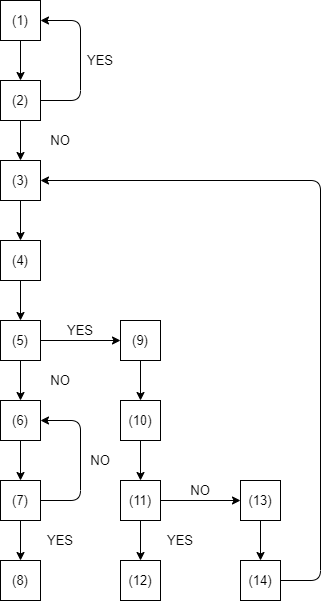
* Start Frame Field: phát ra tín hiệu trên đường dây, nếu không có tín hiệu vọng lại thì đường truyền thông thoáng
* Address Field: lưu địa chỉ máy gửi, máy nhận (địa chỉ MAC)
* Type/Length Field: phân biệt dịch vụ ứng dụng khác nhau, hướng dẫn cách mở, độ dài của gói tin, có thể để rỗng, đánh dấu thứ tự gói
* Data Field: dữ liệu gửi
* FCS Field: kiểm tra lỗi, hàng nguyên đai nguyên kiện hay đã bị xâm phạm, sử dụng thuật toán CRC, nếu sai 1 bit thì coi như dữ liệu bị hỏng, yêu cầu gửi lại.

- Một Frame tầm 1,5 MB

- Chi phí Protocol = %)

* Kích thước Packet là kích thước thật của một gói tin
* Kích thước Payload là kích thước phần dữ liệu

- Hiệu suất của Protocol = (%)

1. **CSMA/CD – Phương thức đa truy cập phát hiện xung đột**
2. Chờ để gửi gói
3. Chờ và lắng nghe
4. Đóng gói gói tin
5. Bắt đâug gửi
6. Kiểm tra có xung đột hay không?
7. Tiếp tục truyền gửi gói
8. Kiểm tra đã truyền xong?
9. Gửi hoàn thành, kết thúc
10. Gửi broadcast tắt nghẽn
11. Báo lần thứ attemp = attemp +1
12. Gửi quá nhiều cảnh báo?
13. Nếu quá nhiều cảnh báo, đứt đường truyền
14. Tính toán thời gian quay lại
15. Quay lại hàng chờ

- Trong phương pháp truy cập CSMA/CD, có thiết bị mạng có dữ liệu để truyền, hoạt động ở chế độ nghe trước khi truyền. Điều này có nghĩa là khi một nút muốn gửi dữ liệu, trước tiên nó phải kiểm tra xem phương tiện mạng có bận hay không. Nếu lúc xác định mạng đang bận, nút sẽ đợi một khoảng thời gian ngẫu nhiên trước khi thử lại. Nếu nút xác định phương tiện mạng không bận, nút sẽ bắt đầu truyền và lắng nghe. Nút lắng nghe để đảm bảo không có trạm nào khác đang truyền tải cùng một lúc. Sau khi hoàn tất quá trình truyền dữ liệu thiết bị sẽ trở về chế độ nghe

# **PHẦN 2 - THIẾT KẾ VÀ XÂY DỰNG ỨNG DỤNG**

1. **Tổng quan về đề tài**

- Ta thường thấy có hai dạng hình thức thi phổ biến nhất hiện nay đó là thi trắc nghiệm và thi tự luận. Sau khi Bộ Giáo dục và Đào tạo quyết định áp dụng dạng đề thi trắc nghiệm vào các kì thi có quy mô lớn như thi THPT quốc gia, kỳ thi tuyển sinh Đại học hay các kì thi cấp chứng chỉ khác, thì việc các **phần mềm thi trắc nghiệm** được cho ra đời ngày càng nhiều nhằm đáp ứng nhu cầu tạo đề cũng như giải đề ôn thi cho học viên. Bài thi trắc nghiệm thể hiện tính khách quan, có khả năng tổng hợp nhiều kiến thức , giúp giáo viên đưa ra đánh giá một cách chính xác.

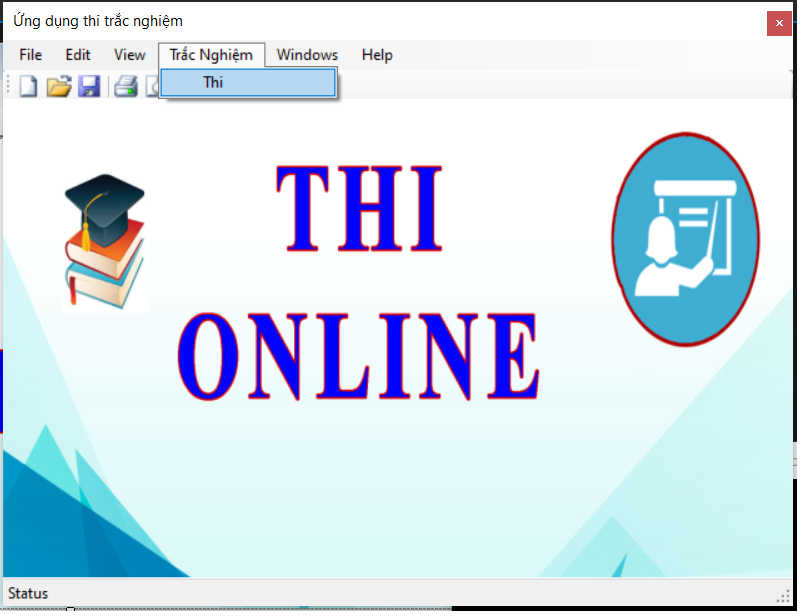
- Đặc biệt đối với hình thức thi trắc nghiệm online không cần phải thực hiện trên giấy, mà chúng ta sẽ có những công cụ phần mềm hiện đại để hỗ trợ soạn đề thi cho thí sinh, có thể thực hành trên đó và thậm chí có cả tính năng tự động chấm điểm hoặc báo điểm, kết quả ngay cho thí sinh sau khi hoàn thành bài thi. Đa số các công cụ hỗ trợ thi trắc nghiệm trực tuyến đều được cài đặt online, họ sẽ cài đặt thêm một phần mềm quản lý thi trực tuyến khi bạn thiết kế website cho trường học, giúp tổ chức sử dụng một nền tảng duy nhất cho giới thiệu và quản lý nghiệp vụ.

- Và nguyên nhân đặc biệt hơn cả là tình hình đại dịch Covid-19 đang có chuyển biến phức hiện nay, để bảo đảm sức khỏe an toàn cho tất cả mọi người, việc học online đã không còn quá xa lạ với tất cả mọi người. Để ứng phó lại hoàn cảnh khó khăn trong thời kỳ này, nhiều ứng dụng online, chuyển đổi số được ra đời.

- Tóm lại từ tất cả nguyên do trên, việc một ứng dụng thi trắc nghiệm trực tuyến ra đời không phải là điều quá bất ngờ.

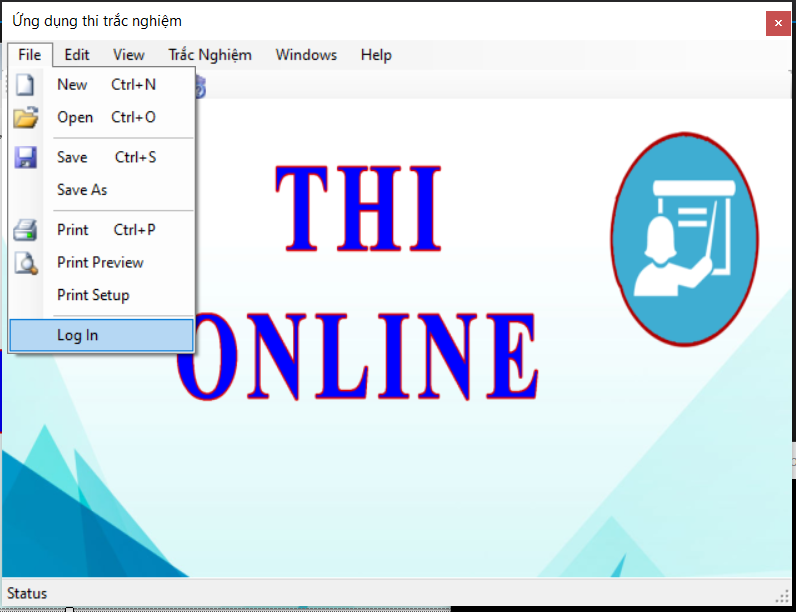
1. **Khái quát chung về ứng dụng**
2. **Chức năng**

|  |  |
| --- | --- |
| Client | Server |
| Đăng ký và Đăng nhập tài khoản dành cho giảng viên và quản lý | Lưu trữ, so sánh và trả về kết quả đăng ký hoặc đăng nhập |
| Xác người dùng thông qua mã OTP gửi qua email | Trả về mật khẩu để so sánh. Nếu đúng sẽ kích hoạt chức năng gửi mã OTP xác thực qua email |
| Phân quyền người dùng bằng cách đăng nhập | Nếu đăng nhập thì thực hiện với quyền giảng viên, còn không thì thực hiện với quyền học sinh |
| Soạn cậu hỏi cho một bộ ngân hàng đề thi | Kiểm tra mã câu hỏi đã tồn tại chưa, nếu chưa thì sẽ lưu lại |
| Ra đề thi từ ngân hàng đề có sẵn | Tải về tất cả câu hỏi trong ngân hàng đề thi và lưu lại kết quả ra đề của người dùng |
| Thi | Tính giờ theo cài đặt đã đặt sẵn khi ra đề. Nếu hết giờ hoặc chủ động Submit trước khi hết giờ sẽ tính điểm cho phiên thi đó và lưu vào kho lưu trữ dữ liệu |
| Bảng kết quả | Hiện thị tất cả thông tin hoặc tìm kiếm thông tin về kết quả thi của một đối tượng |

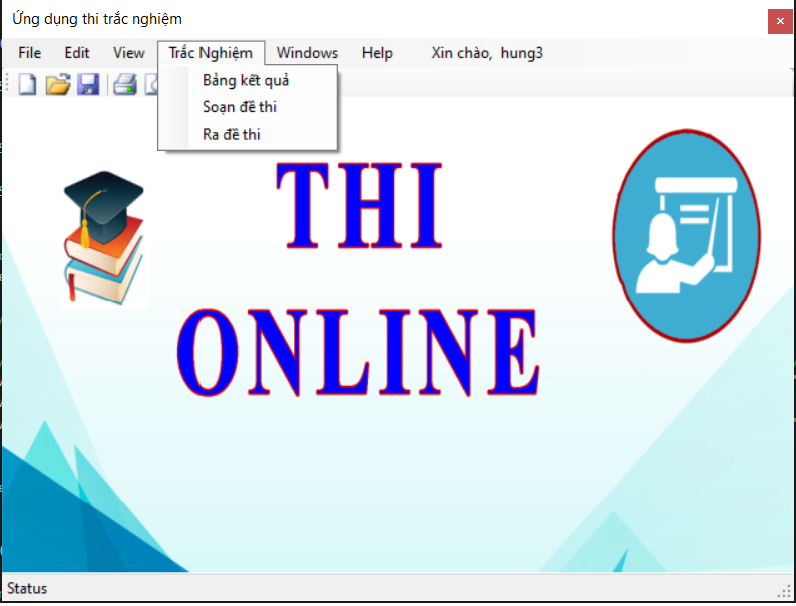
1. **Xây dựng và thiết kế chi tiết của ứng dụng**
2. **FormMain**

*Hình 1.1. FormMain khi vừa mở ứng dụng*

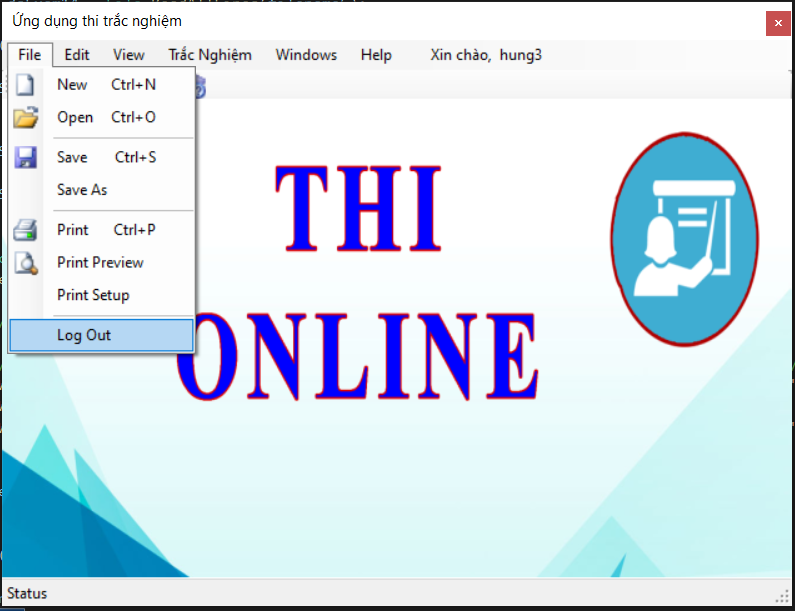
- Khi mở ứng dụng lên, hệ thống sẽ tự động đăng nhập với quyền sinh viên: chỉ được thực hiện chức năng Thi. Nhấn vào Thi để vào chức năng thi trắc nghiệm của ứng dụng

*Hình 1.2. FormMain chức năng Log In*

- Mở File, chọn Log In để chuyển qua Form Login để thực hiện chuyển sang quyền cao hơn, đó là quyền giảng viên

*Hình 1.3. FormMain khi đăng nhập với quyền giảng viên*

- Chọn vào Trắc nghiệm, sẽ hiện thị ra các chức năng chính của quyền giảng viên: xem Bảng kết quả, Soạn đề thi, Ra đề thi

*Hình 1.4. FormMain chức năng Log Out*

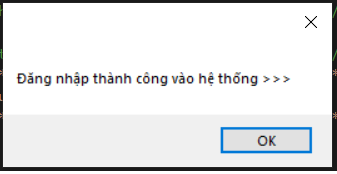
- Khi bấm vào File/ Log Out để đăng xuất khỏi tài khoản giảng viên hiện tại và chuyển qua Form Login

1. ****FormLogin**

*Hình 2.1. FormLogin khi thử chức năng đăng nhập, gửi OTP*

- Nhập Username, Password vào lần lượt các textbox cùng tên. Sau đó, click vào button Login để xác thực. Đợi một thời gian để mã hóa và so sánh kết quả lưu trên Server. Nếu quá trình kết thúc thành công thì sẽ hiện thị “Hash Login Successful>>>>>>” và đợi mã OTP từ email đến.

- Sau khi nhập mã OTP đã nhận vào ô ID. Nếu thành công thì sẽ chuyển lần lượt qua giao diện hình 2.2 rồi tới hình 1.3

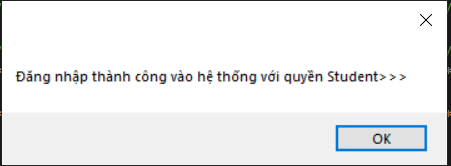
*Hình 2.2. FormLogin sau khi nhập thành công Username, Password, mã OTP*

*Hình 2.3. FormLogin chức năng Student và Registration*

- Bấm vào button Registration để chuyển sang FormRegister để đăng ký một tài khoản quyền giảng viên mới

- Bấm vào button Student để tiếp tục đăng nhập với quyền sinh viên

- Bấm Exit để thoát khỏi ứng dụng

*Hình 2.4. FormLogin sau khi nhập thành công với quyền sinh viên*

1. **FormRegister**

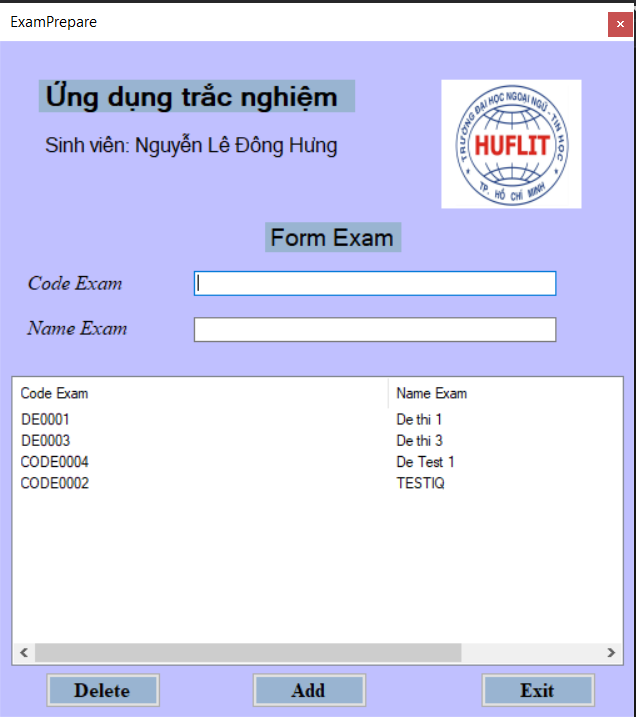
*Hình 3.1. FormRegister và các chức năng chính của nó*

- Nhập đầy đủ các trường Username (chưa tồn tại), Password, Confirm Password (bắt buộc phải trùng với Password), Email (đúng quy tắc email thông thường) và sau đó nhấn Save để lưu vào Server

- Nhập mã Username đã tồn tại, điền đầy đủ thông tin còn lại và nhấn Update để cập nhật thông tin mới cho account này.

- Nhập một Username đã tồn tại và nhấn Delete để xóa mọi thông tin về Username đó

1. **FormExamPrepare**

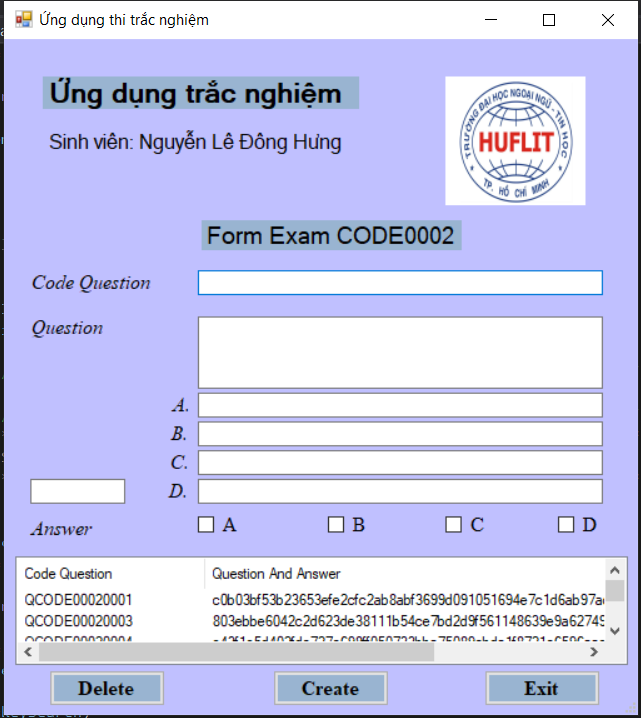
*Hình 4.1. FormExamPrepare thực hiện chức năng Soạn đề thi của ứng dụng*

- Nhập Code Exam (phải là con số nguyên), Name Exam cho bộ đề và nhấn Add để lưu vào Server

- Nhập chọn 1 dòng muốn xóa trên ListView và nhấn Delete để quyết định xóa

- Nhấy đúp vào 1 dòng bất kỳ trên ListView để chuyển qua form thêm câu hỏi cho bộ đề đó

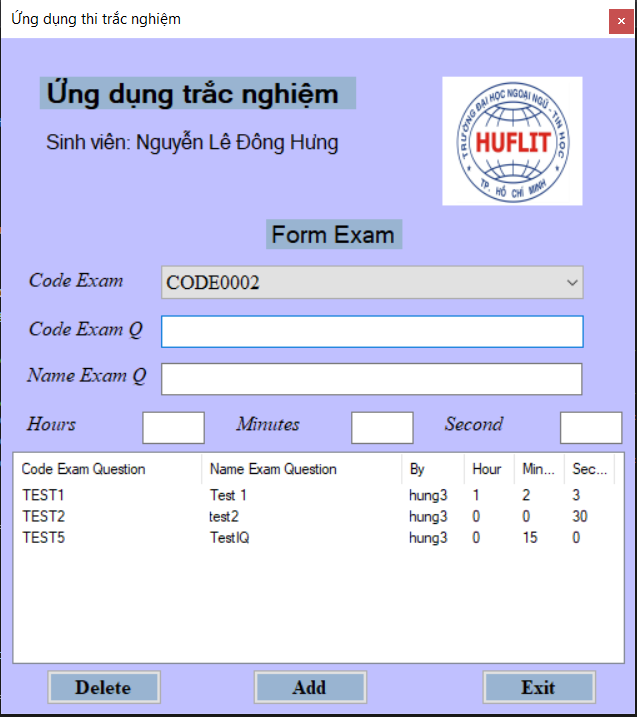
1. **FormQuestion**

*Hình 5.1. FormQuestion để thực hiện chức năng thêm câu hỏi vào bộ đề*

- Đầu tiên chọn Create để bắt đầu thêm câu hỏi

- Nhập Code Question (phải là 1 dãy số nguyên), Question, 4 câu trả lời A B C D và chọn check ít nhất một trong 4 đáp án. Cuối cùng nhấn Save để kết thúc quá trình tạo một câu hỏi

- Nháy vào một câu hỏi bất kỳ trên ListView và nhấn button Delete để xóa câu hỏi đó khỏi bộ để

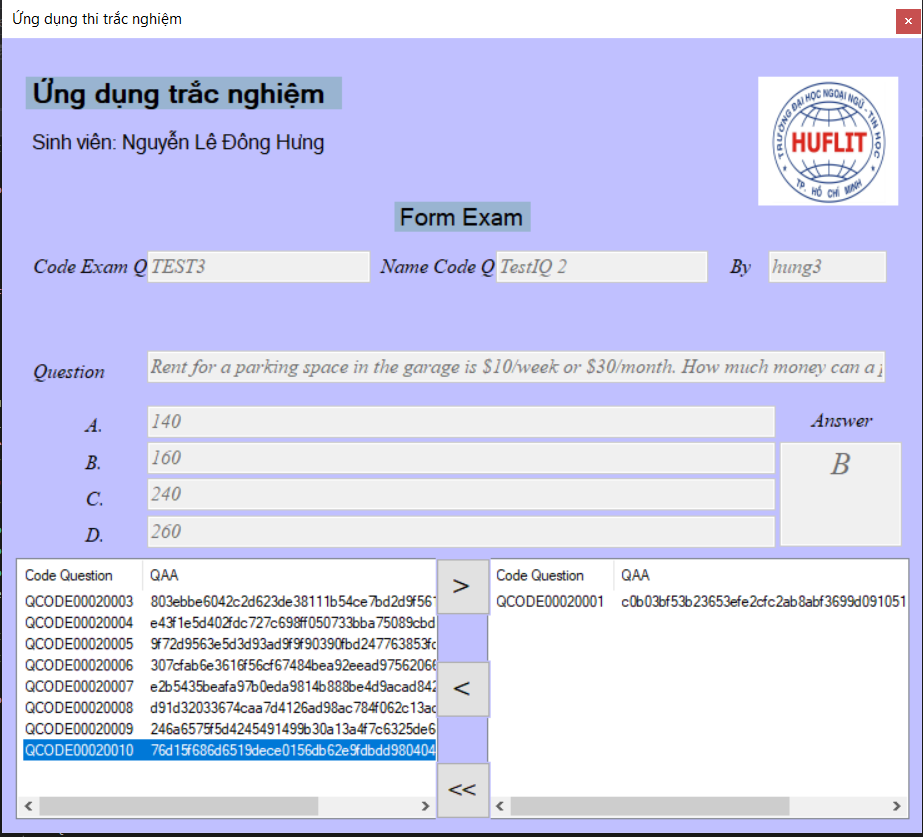
1. **FormExamQues**

*Hình 6.1. FormExamQues và chức năng ra đề từ ngân hàng đề sẵn có*

- Khi vừa tải form lên thì vị trí Code Exam sẽ tải lên ngày những bộ đề đang có sẵn để người dùng có thể sử dụng

- Người dùng nháy vào 1 dòng bất kỳ trên ListView và chọn button Delete để thực hiện xóa dòng đề thi đó ra khỏi danh sách đề thi

- Lựa chọn Code Exam mong muốn từ combox, nhập CodeExamQ, NameExamQ, điền Hours, Minutes, Second (giờ, phút, giây là các số nguyên), sau đó bấm Add để thêm 1 đề mới vào Server

1. **FormExamQuestion**

*Hình 7.1. FormExamQuestion và các chức năng của nó*

- Các ô textbox Code Exam Q, Name Exam Q, By sẽ không thể nhập liệu và chỉnh sửa (enble), nó lần lượt hiện thị mã đề, tên đề và Username người tạo đề

- Nháy đúp vào 1 dòng bất kỳ ở ListView1 để giải mã câu hỏi đó ra ngôn ngữ tự nhiên

- Chọn 1 dòng ở ListView 1 và nhấn button “>” để thêm câu hỏi đó vào đề thi

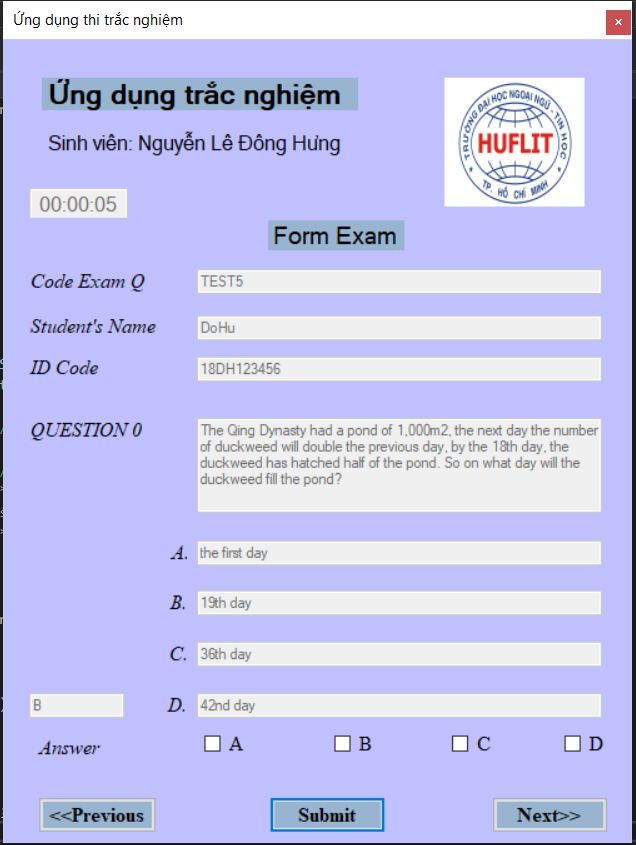
- Chọn 1 dòng ở ListView 2 và nhấn button “<” để xóa câu hỏi đó khỏi đề thi

- Nhấn button “<<” để thoát khỏi chức năng và lưu lại toàn bộ kết quả

1. **FormPrepareThi**

*Hình 8.1. FormPrepareThi và một số thông tin yêu cầu phải cung cấp để vào thi*

- Nhập Code Exam Q đã tồn tại để vào thi bộ đề đó (nếu không có thì không thể vào thi), Tên học sinh (Student’s Name) và mã báo danh (Identification Code) là những thông tin bắt buộc của thí sinh. Cuối cùng thì nhấn button Enter để vào thi

1. **FormThi**

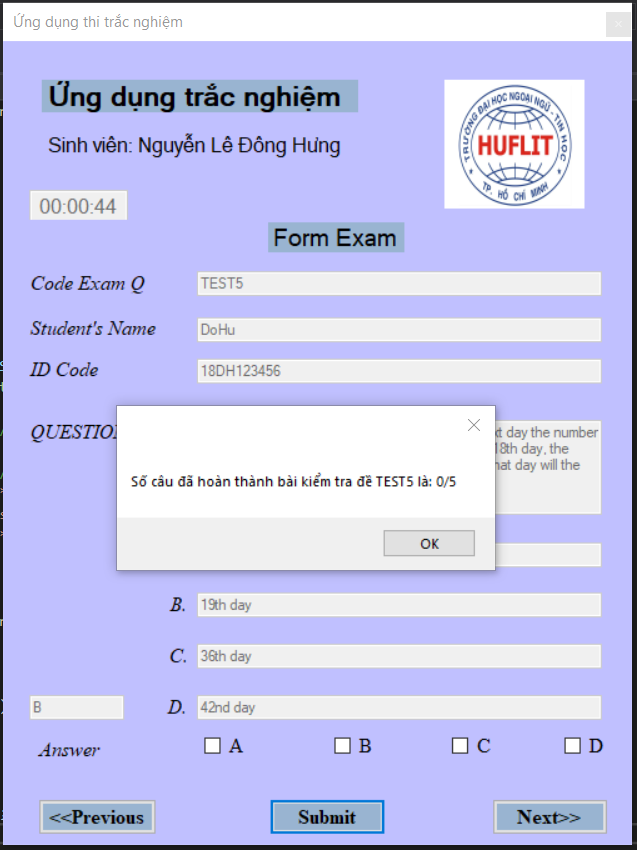
*Hình 9.1. FormThi và thông tin sẽ hiện thị khi vừa mới tải về*

- Sau khi tải dữ liệu câu hỏi thành công, đồng hồ đếm giờ trên góc trái sẽ tự động chạy cho đến thời gian đã cấu hình sẵn khi tạo đề thi. Khi kết thúc thời gian, chương trình sẽ tự động nộp bài và tính điểm cho bài thi này

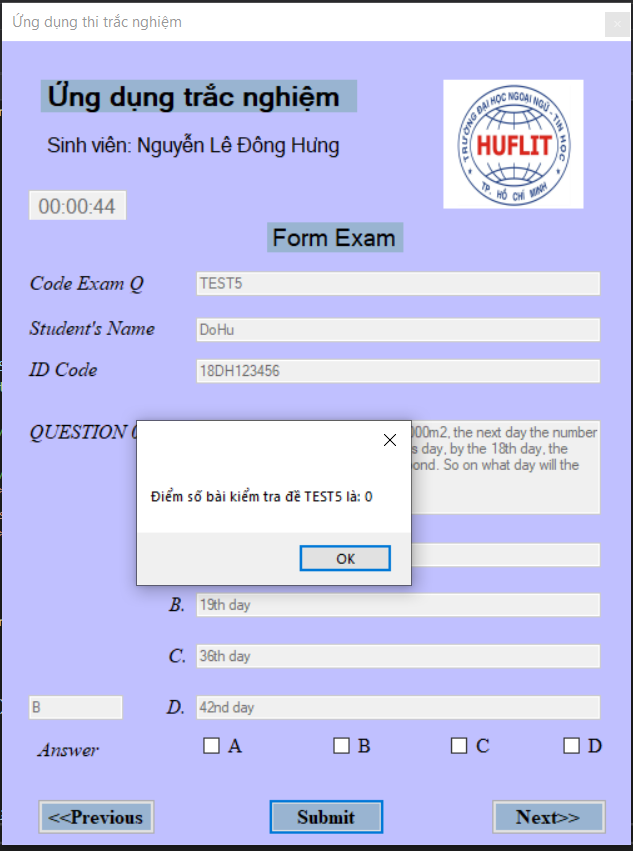
- Nhấn button “<<Previous” để chuyển qua câu hỏi ở phía trước

- Nhấn button “Next>>” để chuyển qua câu hỏi ở phía sau

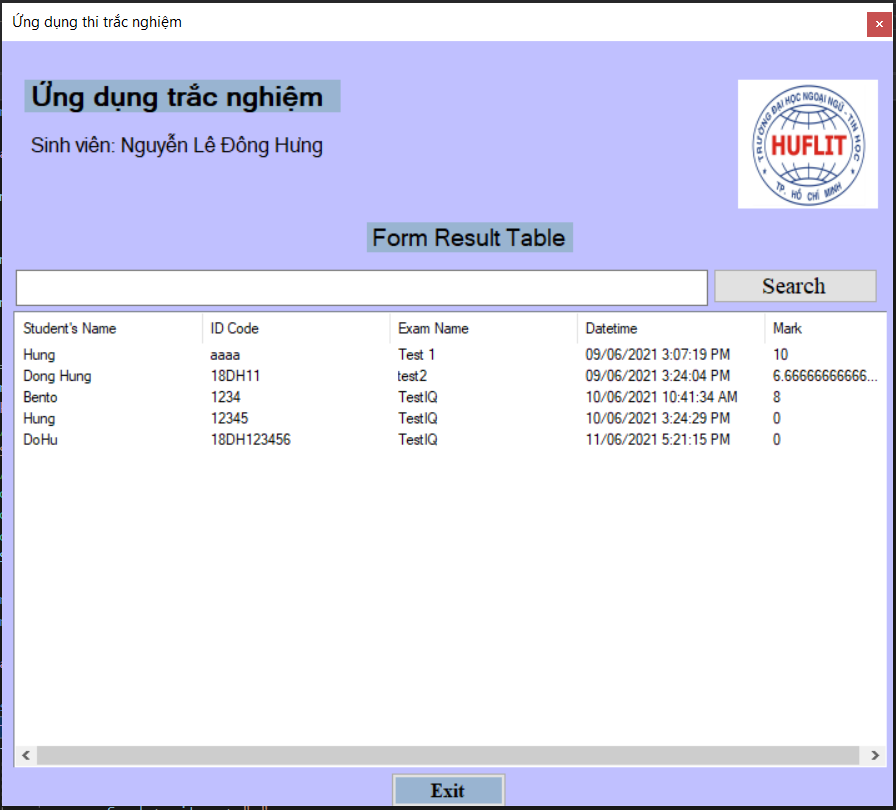
- Nhấn button Submit để nộp bài và tính điểm

*Hình 9.2. FormThi sau khi nhấn button Submit*

- Tính toán số câu hỏi đã làm đúng trên tổng số câu hỏi (0/5 nghĩa là làm đúng 0 câu trên tổng số 5 câu)

*Hình 9.3. FormThi sau khi bấm OK ở hình 9.2*

- Tính toán điểm theo thang điểm 10, sau khi bấm OK ở đây thì các thông tin về Code Exam Q, Student’s Name, ID Code, điểm bài thi này, thời gian khi kết thúc thúc thi

1. **FormResult**

*Hình 10.1. FormResult khi vừa tải về*

- Dữ liệu của tất cả lần thi sẽ được tải về ở ListView

*Hình 10.2. FormResult khi thực hiện chức năng Search*

- Ví dụ nhập keySearch là TestIQ, sau đó nhấn button Search thì ListView sẽ trả về tất cả dữ liệu có liên quan đến TestIQ sẽ được hiển thị. Nếu keySearch không có đối tượng nào trùng khớp thì ListView sẽ không trả về gì cả. Nếu Search “” thì ListView sẽ gửi về tất cả dữ liệu có trong Server về thông tin thi

1. **Tổng kết**

**-** Thông qua việc thực hiện đề tài kết thúc học phần này, nhóm em thực hiện những điều sau đây:

* Nắm rõ được đã được các nội dung lý thuyết của bộ môn, có được vốn kiến thức nền tảng về Lập trình C# kết hợp với Lập trình Mạng (Lập trình Socket).
* Làm được các chức năng cơ bản của một ứng dụng theo mô hình Client-Server: các thao tác giao tiếp giữa Client và Server, đọc/ghi file, thêm xóa sửa dữ liệu trong một file, đặc biệt là tìm kiếm keyword hoặc đối tượng trùng lặp trong file.
* Về mặt ứng đụng, chúng em đã hoàn thành cơ bản các chức năng chính của một “Ứng dụng thi trắc nghiệm trực tiếp”, đảm bảo tính logic của dữ liệu, phân quyền người dùng, tổ chức vào thi thực tế với thời gian thi được cấu hình trước, xây dựng nên bảng lưu trữ kết quả điểm của thí sinh khi thi.

- Ngoài ra, trong quá trình thực hiện cũng còn một vài mặt hạn chế như:

* Phần báo cáo đề tài vẫn còn tồn tại một số khuyết điểm nhỏ liên quan đến yếu tố trình bày và trích dẫn
* Về phần ứng dụng, chưa xử lý được chuỗi Tiếng Việt có dấu, chưa nắm bắt được kỹ thuật xử lý phân luồng để có thể tối ưu hóa tốc độ xử lý của Server và ứng dụng, chưa phân quyền những đối tượng có thể đăng ký tài khoản để tham gia ứng dụng với quyền cao cấp

- Mong muốn của nhóm sẽ có thể phát triển trong thời gian sắp tới, cải tiến và phát triển một số chức năng còn thiếu hoặc chưa tối ưu, thay đổi giao diện ứng dụng để thân thiện với người dùng hơn

1. **Danh mục tài liệu tham khảo**

*About us: CODEGYM*. (2018, December 22). Retrieved from CODEGYM Web Site: https://codegym.vn/blog/2018/12/22/lap-trinh-c-la-gi/

CÔNG TY TNHH IT SYSTEMS VIỆT NAM. (2018). *About Us: IT SYSTEMS.* Retrieved from IT SYSTEMS Web Site: https://itsystems.vn/mang-may-tinh-la-gi-phan-loi-mang-may-tinh-hien-nay/

DZR-Web, T. k. (n.d.). *DZR-Web*. Retrieved from DZR-Web Web Site: http://www.dzr-web.com/top-9-phan-mem-thi-trac-nghiem-online/

Nhi, H. (n.d.). *Kiến Thức Hosting*. Retrieved from TINOHOST Web Site: https://blog.tinohost.com/socket-la-gi/

*Tin Học Sóc Trăng*. (2019, May 16). Retrieved from Tin Học Sóc Trăng Web Site: https://www.tinhocsoctrang.com/2019/05/huong-dan-gui-email-bang-c-sharp-qua-giao-thuc-smtp.html

TrisDo. (2020, August 9). *About Us: CODELEARN*. Retrieved from CODELEARN Web Site: https://codelearn.io/sharing/lap-trinh-socket-co-voi-tcpip-ava

Tùng, B. T. (2010). *IT4060: Lập trình mạng.* Retrieved from IT4060: Lập trình mạng Web Site: https://users.soict.hust.edu.vn/tungbt/it4060/Lec01.Overview.pdf