|  |  |
| --- | --- |
| logomobile.png | BỘ GIÁO DỤC VÀ ĐÀO TẠO  **TRƯỜNG ĐẠI HỌC CÔNG NGHỆ TP. HCM** |

**BÁO CÁO MÔN HỌC**

**BÁO CÁO GIỮA KÌ MÔN**

**HỆ THỐNG BẢO MẬT THÔNG TIN**

SINH VIÊN THỰC HIỆN: **LÊ NGUYỄN THÀNH LONG**

MSSV: 1811060436 LỚP: 18DTHB3

**ĐẶNG PHƯỚC TÀI**

MSSV: 1811061015 LỚP: 18DTHB3

TP. HỒ CHÍ MINH, 12/2020

|  |  |
| --- | --- |
| logomobile.png | BỘ GIÁO DỤC VÀ ĐÀO TẠO  **TRƯỜNG ĐẠI HỌC CÔNG NGHỆ TP. HCM** |

**BÁO CÁO MÔN HỌC**

**BÁO CÁO GIỮA KÌ MÔN**

**HỆ THỐNG BẢO MẬT THÔNG TIN**

NGÀNH: **CÔNG NGHỆ THÔNG TIN**

MÔN HỌC: **HỆ THỐNG BẢO MẬT THÔNG TIN**

SINH VIÊN THỰC HIỆN: **LÊ NGUYỄN THÀNH LONG**

MSSV: 1811060436 LỚP: 18DTHB3

**ĐẶNG PHƯỚC TÀI**

MSSV: 1811061015 LỚP: 18DTHB3

GVHD: **THS. TỐNG THANH VĂN**

TP. Hồ Chí Minh, 12/2020

Mục Lục

[Lời mở đầu 4](#_Toc68770806)

[**Chương 1** **Giới thiệu chung về ứng dụng** 4](#_Toc68770807)

[**1.1** **Giới thiệu chương trình làm ứng dụng** 4](#_Toc68770808)

[**1.1.1** **Giới thiệu Visual Studio** 4](#_Toc68770809)

[Một số tính năng của phần mềm Visual Studio 5](#_Toc68770810)

[Biên tập mã 5](#_Toc68770811)

[Trình gỡ lỗi 5](#_Toc68770812)

[Thiết kế 5](#_Toc68770813)

[Windows Forms Designer 5](#_Toc68770814)

[**1.2** **Giới thiệu về SQL sever** 6](#_Toc68770815)

[Khái niệm SQL 6](#_Toc68770816)

[**Lịch sử ra đời và các ấn bản của SQL server** 7](#_Toc68770817)

[**Chương 2** **Cơ chế hoạt động** 9](#_Toc68770818)

[**2.1** **Giới thiệu về mật mã Vigenère** 9](#_Toc68770819)

[**2.1.1** **Khái niệm** 9](#_Toc68770820)

[**2.1.2** **Mô tả** 9](#_Toc68770821)

[**2.2** **Giới thiệu về phân quyền** 15](#_Toc68770822)

[**2.2.1** **Khái niệm** 15](#_Toc68770823)

[**2.2.2** **Cách phân quyền** 15](#_Toc68770824)

[**Chương 3** **Chương trình thực nghiệm** 16](#_Toc68770825)

[**3.1** **Đăng nhập** 16](#_Toc68770826)

[**3.2** **Đăng ký** 17](#_Toc68770827)

[**3.3** **Màn hình chính** 18](#_Toc68770828)

[**3.4** **Quản lý dự án** 19](#_Toc68770829)

[**3.5** **Quản lý nhân viên** 20](#_Toc68770830)

[**3.6** **Quản lý phòng ban** 21](#_Toc68770831)

[**3.7** **Quản lý tài khoản** 22](#_Toc68770832)

Lời mở đầu

Đồ án môn học với những ngôn ngữ lập trình khá phổ biến với sinh viên khoa Công Nghệ Thông Tin như ngôn ngữ C# để tự thống kê kiến thức và lên ý tưởng cho những đồ án của mình. Đó là yêu cầu và cũng như thách thức đối với sinh viên khi làm quen với thực hành từ nhữnh lý thuyết sách vở.

Để cho em có thể nắm chắc kiến thức và tiếp cận thực tế với một sản phẩm phần mềm, em đã nhận được sự trợ giúp tận tình từ thầy Tống Thanh Văn giảng viên hướng dẫn, cùng với sự góp ý của các thành viên trong lớp để giúp em hoàn thành đồ án này.

Khi nhìn vào định nghĩa mã hóa thông tin, chúng ta có thể thấy một trong những lĩnh vực rất cần thiết để áp dụng mã hóa trong quá khứ: trao đổi liên lạc giữa các bên trong chiến tranh. Hai đồng minh cần phải trao đổi thông tin mà đảm bảo kẻ thù không thể đọc được, vì vậy việc mã hóa dữ liệu khi trao đổi là một trong những yếu tố quyết định trong việc truyền tin mà không thể tạo được một kênh kết nối an toàn và bí mật.

Xuất phát từ những lý do trên, cùng với sự hướng dẫn của Giảng viên Tống Thanh Văn, chúng em thực hiện 1 số chức năng như mã hóa mật khẩu, phân quyền account, theo dõi người dùng … để vận dụng những kiến thức đã được học vào thực tiễn, xây dựng một hệ thống bảo mật cho người dùng.

Chúng em xin chân thành cảm ơn!

Sinh viên thực hiện

**Lê Nguyễn Thành Long**

**Đặng Phước Tài**

1. **Giới thiệu chung về ứng dụng**
   1. **Giới thiệu chương trình làm ứng dụng**
      1. **Giới thiệu Visual Studio**

**Visual Studio là một trong những phần mềm lập trình không thể thiếu cho những bạn theo hướng lập trình VB+ và C#. Tuy nhiên không phải ai cũng có thể hiểu rõ được phần mềm này. Chính vì thế, chúng tôi sẽ giúp các bạn tìm hiều Visual Studio là gì? Cũng như các tính năng của phần mềm này.**



Visual studio là một trong những công cụ hỗ trợ [lập trình website](https://monamedia.co/dich-vu/thiet-ke-website/) rất nổi tiếng nhất hiện nay của Mcrosoft và chưa có một phần mềm nào có thể thay thế được nó. Visual Studio được viết bằng 2 ngôn ngữ đó chính là C# và VB+. Đây là 2 ngôn ngữ lập trình giúp người dùng có thể lập trình được hệ thống một các dễ dàng và nhanh chóng nhất thông qua Visual Studio.

Visual Studio là một phần mềm lập trình hệ thống được sản xuất trực tiếp từ [Microsoft](https://www.microsoft.com/vi-vn/). Từ khi ra đời đến nay, Visual Studio đã có rất nhiều các phiên bản sử dụng khác nhau. Điều đó, giúp cho người dùng có thể lựa chọn được phiên bản tương thích với dòng máy của mình cũng như cấu hình sử dụng phù hợp nhất.

Bên cạnh đó, Visual Studio còn cho phép người dùng có thể tự chọn lựa giao diện chính cho máy của mình tùy thuộc vào nhu cầu sử dụng.

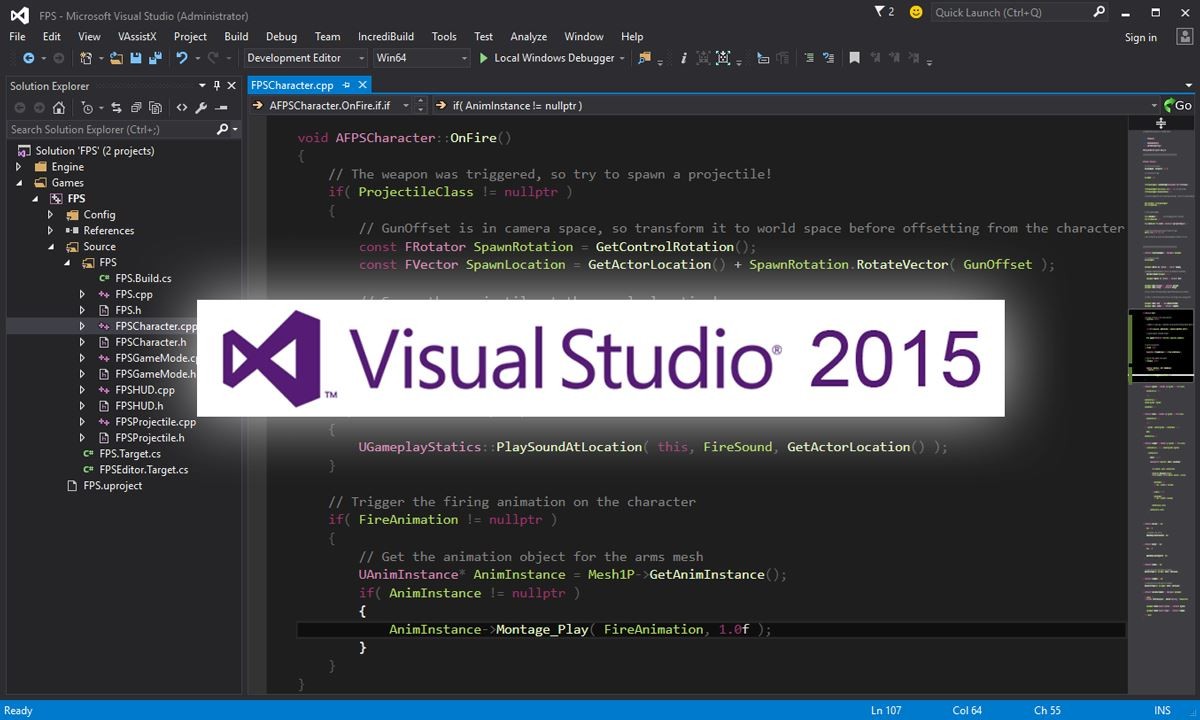
Một số tính năng của phần mềm Visual Studio

**Biên tập mã**

Giống như bất kỳ một IDE khác, Visual Studio gồm có một trình soạn thảo mã hỗ trợ tô sáng cú pháp và hoàn thiện mả bằng các sử dụng [IntelliSense](https://msdn.microsoft.com/en-us/library/hcw1s69b.aspx) không chỉ cho các hàm, biến và các phương pháp mà còn sử dụng cho các cấu trúc ngôn ngữ như: Truy vấn hoặc vòng điều khiển.

Bên cạnh đó, các trình biên tập mã Visual Studio cũng hỗ trợ cài đặt dấu trang trong mã để có thể điều hướng một cách nhanh chóng và dễ dàng. Hỗ trợ các điều hướng như: Thu hẹp các khối mã lệnh, tìm kiếm gia tăng,…

Visual Studio còn có tính năng biên dịch nền tức là khi mã đang được viết thì phần mềm này sẽ biên dịch nó trong nền để nhằm cung cấp thông tin phản hồi về cú pháp cũng như biên dịch lỗi và được đánh dấu bằng các gạch gợn sóng màu đỏ.

Phiên bản Visual Studio 2015 đang được sử dụng phổ biến nhất hiện nay.

**Trình gỡ lỗi**

Visual Studio có một trình gỡ lỗi có tính năng vừa lập trình gỡ lỗi cấp máy và gỡ lỗi cấp mã nguồn. Tính năng này hoạt động với cả hai mã quản lý giống như ngôn ngữ máy và có thể sử dụng để gỡ lỗi các ứng dụng được viết bằng các ngôn ngữ được hỗ trợ bởi Visual Studio.

**Thiết kế**

Windows Forms Designer

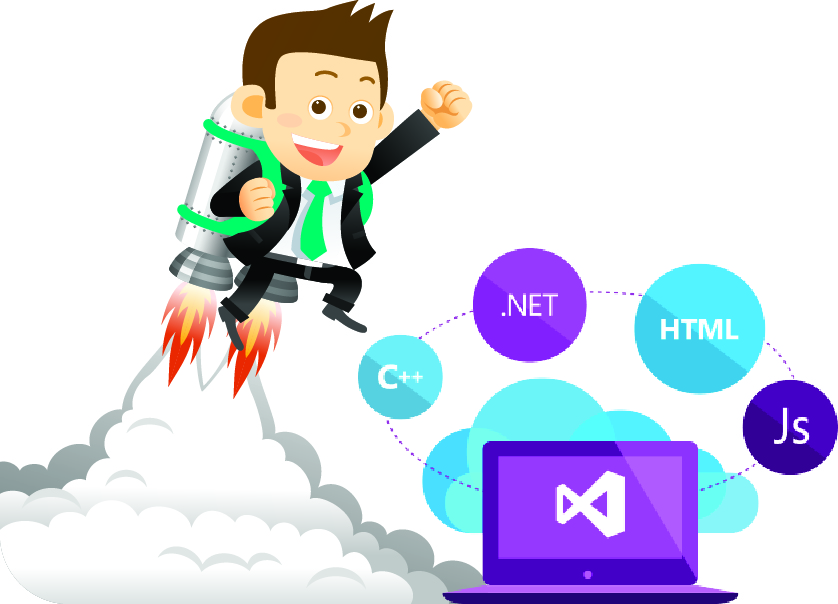
Được sử dụng với mục đích xây dựng GUI sử dụng Windows Forms, được bố trí dùng để xây dựng các nút điều khiển bên trong hoặc cũng có thể khóa chúng vào bên cạnh mẫu. Điều khiển trình bày dữ liệu có thể được liên kết với các nguồn dữ liệu như: Cơ sở dữ liệu hoặc truy vấn.

**WPF Designer**

Tính năng này cũng giống như Windows Forms Designer có công dụng hỗ trợ kéo và thả ẩn dụ. Sử dụng tương tác giữa người và máy tính nhắm mục tiêu vào [Windows Presentation Foundation](https://vi.wikipedia.org/wiki/Windows_Presentation_Foundation).

**Web designer/development**

Visual Studio cũng có một trình soạn thảo và thiết kế website cho phép các trang web được thiết kế theo tính năng kéo và thả đối tượng. Mục đích là để hỗ trợ người dùng tạo trang web dễ dàng hơn, những yêu cầu đơn giản như [thiết kế web du lịch](http://thietkewebsitedulich.com/) hay các trang giới thiệu của công ty có thể sử dụng tính năng này vì nó vẫn đảm bảo cho bạn sở hữu được một website hoàn chỉnh.

Visual Studio là trình soạn thảo nhiều ngôn ngữ lập trình.

* 1. **Giới thiệu về SQL sever**

**Khái niệm SQL**

SQL Server chính là một hệ quản trị dữ liệu quan hệ 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, datase engine và các chương trình ứng dụng dùng để quản lý các bộ phận trong RDBMS và những dữ liệu khác.

**Lịch sử ra đời và các ấn bản của SQL server**

**+ Lịch sử ra đời 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 website**](https://monamedia.co/dich-vu/thiet-ke-website/).

Cho đến ngày nay thì phiên bản mới nhất đó là SQL Server 2016 hỗ trợ bộ vi xử lý 64 bit ra đời vào ngày 1 tháng 6 năm 2016.

**+ Một vài ấn bản SQL Server**

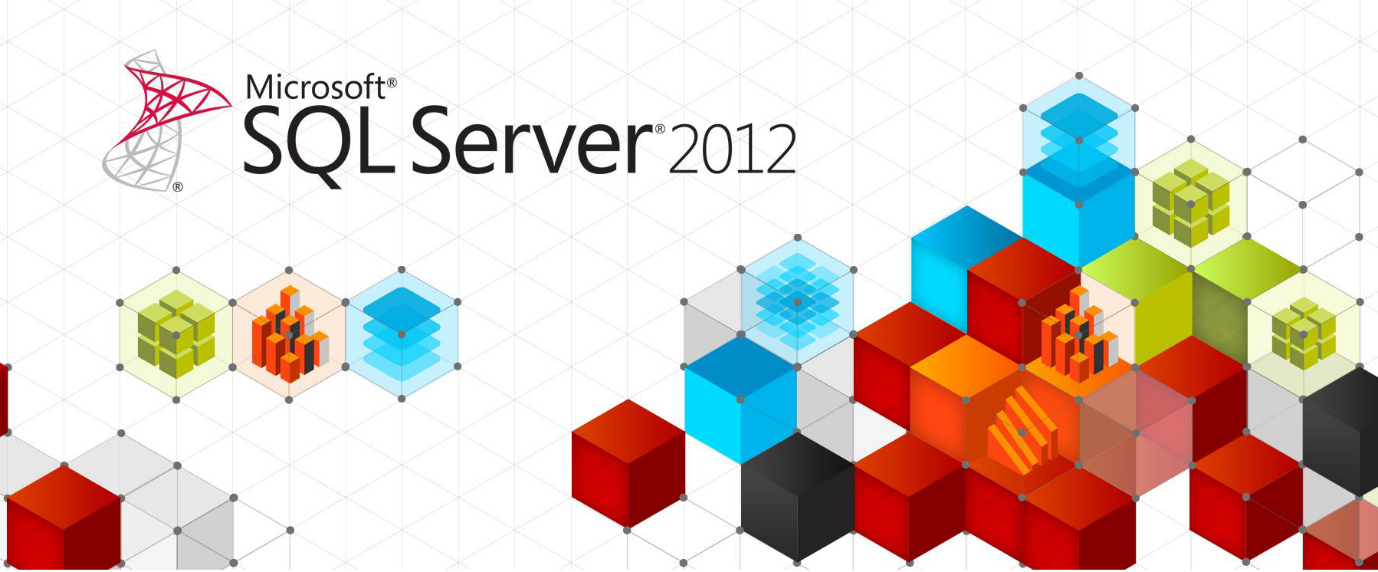
Enterprise: là một ấn bản chứa tất cả các đặc điểm nổ bật của SQL Server như: các công cụ cho tạo và quản lý phân cụm SQL Server, nhân bộ máy cơ sở dữ liệu và một số dịch vụ đi kèm. Nó có thể đánh địa chỉ 12 terabytes và quản lý cơ sở dữ liệu lên tới 524 petabytes.

Standard: Ấn 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: Ấn 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](http://freelancervietnam.vn/), [website spa](http://websitespa.vn/),…

Workgroup: ấn 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 ấn bản này.

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



**Các thành phần cơ bản trong SQL Server**

Các thành 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 ệ 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 dimendion 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 dịch thời đến hàng ngàn người dăng 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.

1. **Cơ chế hoạt động**
   1. **Giới thiệu về mật mã Vigenère**
      1. **Khái niệm**

Mật mã Vigenère là một phương pháp mã hóa văn bản bằng cách sử dụng xen kẽ một số phép [mã hóa Caesar](https://vi.wikipedia.org/wiki/M%E1%BA%ADt_m%C3%A3_Caesar) khác nhau dựa trên các [chữ cái](https://vi.wikipedia.org/wiki/Ch%E1%BB%AF_c%C3%A1i) của một từ khóa. Nó là một dạng đơn giản của mật mã thay thế dùng nhiều bảng chữ cái.

* + 1. **Mô tả**

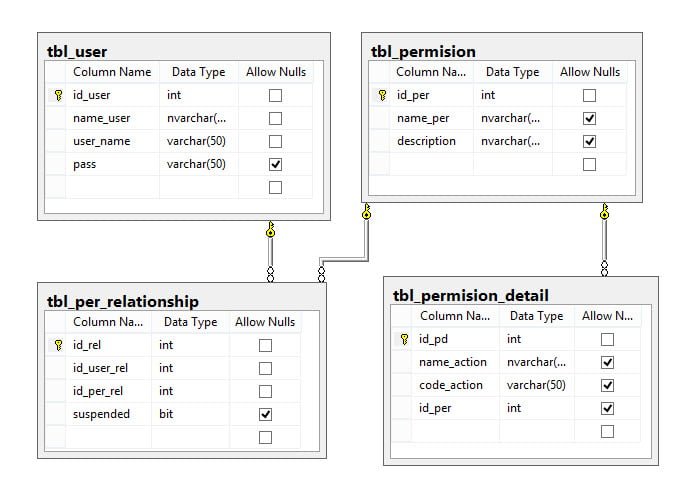
|  |  |
| --- | --- |
| Mật mã Vigenère là một phương pháp mã hóa chữ văn bản tiếng Anh, lần đầu tiên được mô tả bởi Giovan Battista Bellaso vào năm 1553. Phương pháp mã hóa mật mã Vigenère dễ hiểu và dễ thực hiện, nhưng chỉ đến năm 1863 với nhiều nỗ lực suốt ba thế kỷ, Friedrich Kasiski mới xuất bản một phương pháp chung để giải mã mật mã Vigenère.  Mật mã Vigenère là tập hợp các quy tắc thay thế chữ cái đơn trong bảng chữ cái tiếng Anh qua việc sử dụng 26 mật mã Caesar với các bước dịch chuyển từ 0 đến 25 tương ứng từ chữ ‘a’ đến chữ ‘z’. Cụ thể, bản mã Vigenère được lập theo công thức sau:  ci = (pi + ki) mod 26, i=1,…,l  trong đó, C = {c1…cl} là bản mã, P={p1…pl} là bản gốc, K = {k1…kl} là dãy khóa và l là độ dài bản rõ. Tương tự, bản gốc P có thể được tính nếu biết khóa và bản mã theo công thức:  pj = (cj – kj) mod 26, j = 1,…,l  Mật mã Vigenère không thể phá vỡ trong trường hợp sử dụng các khóa đủ dài. Nhưng với các khóa ngắn hoặc nếu nhà thám mã có đủ nhiều bản mã so với độ dài khóa thì khá dễ để phá vỡ. Việc thám mật mã Vigenère thường tiến hành theo hai bước là: xác định độ dài chu kỳ của khóa trước, sau đó tìm khóa cụ thể. ****Tìm chu kỳ khóa của mật mã Vigenère**** Đầu tiên cần lưu ý là chu kỳ của khóa tìm được có thể không đúng với thực tế được sử dụng. Nếu bản mã đủ dài thì có thể là chính xác, các phương pháp được cung cấp ở đây là gần đúng.  Mật mã Vigenère áp dụng các mật mã Caesar khác nhau cho các chữ cái liên tiếp. Ví dụ một bản mã Vigenère như sau:  Thám mật mã Vigenère  Hình 1. Bản mã hóa sử dụng mật mã Vigenère  Mật mã Caesar là một dạng của mật mã thay thế, theo đó mỗi ký tự trong bản rõ được thay thế bằng một ký tự cách nó một đoạn trong bảng chữ cái để tạo thành bản mã. Giả sử với khóa là 3 (dịch 3 vị trí trong bảng chữ cái), thì chữ ‘a’ sẽ được thay bằng chữ ‘d’, chữ ‘b’ sẽ được thay bằng ‘e’ và cứ thế đến hết bản rõ. Phương pháp này được đặt tên là Caesar, vị Hoàng đế đã sử dụng loại mật mã này thường xuyên trong công việc.  Nếu mật mã Vigenère sử dụng khóa có chu kỳ 3 là 'PUB', thì chữ cái rõ đầu tiên được mã hóa bằng mật mã Caesar với khóa là 16 (P là chữ cái thứ 16 của bảng chữ cái), chữ cái thứ hai được mã với khóa là 21 (chữ cái U) và chữ cái thứ ba được mã với khóa là 2 (chữ cái B). Chữ cái rõ thứ 4 được mã hóa quay lại bằng chữ khóa thứ nhất (khóa 16). Kết quả là, các chữ cái ở các vị trí 1,4,7,10,... đều được mã hóa bằng cùng một mật mã Caesar với chữ khóa là P. Các chữ cái ở các vị trí 2,5,8,11,... và 3,6,9,12,... được mã hóa bằng mật mã Caesar với khóa tương ứng là chữ U và B.  Như vậy, trình tự chính xác sẽ phụ thuộc vào chu kỳ của khóa mật mã, tức là độ dài khóa, như với ví dụ trên thì độ dài chu kỳ khóa là 3. ****Phương pháp tìm chu kỳ khóa theo sự lặp lại của nhóm chữ cái**** Để xác định chu kỳ của khóa mật mã Vigenère, phương pháp Kasiski xem xét sự lặp lại của các nhóm chữ cái như Hình 2.  Thám mật mã Vigenère  Hình 2. Sự lặp lại của nhóm chữ cái  Đoạn lặp lại loạt **VHVS** gồm 18 ký tự, gợi ý rằng độ dài khóa có thể là 18, 9, 6, 3, 2. Còn đoạn lặp lại loạt **QUCE** là 30 ký tự, gợi ý độ dài khóa là 30, 15, 10, 6, 5, 3, 2. Kết hợp lại, độ dài khóa có thể là 6, 3 hoặc 2. ****Chỉ số trùng hợp (Index of coincidence - I.C. còn được ký hiệu là****Ic()****)**** Nếu trong bản mã không có sự lặp lại của một loạt chữ cái nào, người thám mã sẽ sử dụng đến chỉ số trùng hợp Ic.  Giả sử X là một chuỗi ký tự trong tiếng Anh, ký hiệu xác suất xuất hiện của các chữ a, b,…, z lần lượt là p0, p1,…, p25. Khi đó:  Thám mật mã Vigenère  • Khi đó:  Ic (x) = ∑  = 0.0822+0.0152+…+0.0012 = 0.065  Chỉ số trùng hợp đôi khi được gọi là tỷ lệ lặp lại. Nếu bản mã cụ thể có độ dài n, na là tần số xuất hiện của chữ ‘a’, nb là tần số xuất hiện của chữ ‘b’…, thì chỉ số trùng hợp gần đúng được tính theo công thức sau:  Thám mật mã Vigenère  Chỉ số trùng hợp (Ic) là một kỹ thuật thống kê giúp xác định một đoạn văn bản có đáp ứng quy luật ngôn ngữ của tiếng Anh. Một tính chất quan trọng của kỹ thuật là giá trị Ic không thay đổi nếu áp dụng mật mã thay thế đơn cho văn bản. Điều này là do Ic dựa trên tần số xuất hiện của chữ cái và mật mã thay thế đơn không làm thay đổi tần số của bộ chữ cái riêng lẻ. Với văn bản tiếng Anh sẽ có giá trị Ic làm tròn là 0.06, nếu các ký tự có phân phối đồng đều thì Ic gần hơn với 0,03 - 0,04.  Phương pháp dùng chỉ số trùng hợp để xác định chu kỳ của khóa mật mã Vigenère thực hiện như sau. Trước tiên, giả sử độ dài khóa là 2, thực hành trích xuất hai chuỗi tại các vị trí 1, 3, 5, 7,... và 2, 4, 6, 8,... từ bản mã như Hình 1 (lưu ý rằng Ic được tính bằng cách sử dụng toàn bộ chuỗi mã, không chỉ là phần được hiển thị).  Thám mật mã Vigenère  Tương tự với trường hợp độ dài khóa là 3 sẽ có 3 chuỗi, tương ứng với các giá trị Ic như sau:  Thám mật mã Vigenère  Như vậy, Ic trung bình đối với trường hợp chu kỳ 2 là khoảng 0,048 và đối với trường hợp chu kỳ 3 là khoảng 0,047.  Quy trình này sẽ được lặp lại cho tất cả các độ dài khóa muốn kiểm tra. Ví dụ tiếp tục tính với chu kỳ khóa lên đến 15 sẽ có các giá trị trung bình I.C (avg I.C.) tương ứng như Hình 3.  Thám mật mã Vigenère  Hình 3. Các giá trị trung bình I.C.  Theo cột giá trị, hoặc theo biểu đồ, có 2 giá trị trung bình I.C cao đột biến đã gợi ý rằng khóa mật mã có thể có độ dài 7 hoặc 14. Cả hai xác suất này phải được tiếp tục kiểm tra. ****Tìm khóa mật mã theo kỹ thuật thống kê khi bình phương**** Ví dụ thám mã với khóa chu kỳ 7 (sử dụng 7 mật mã Caesar) cho bản mã ở Hình 1, việc tìm khóa khá dễ dàng. Thám mã sẽ so sánh giá trị thống kê Khi bình phương của dãy phá mã với giá trị phân phối tần số xuất hiện chữ cái tiếng Anh.  Lập   chuỗi   chữ   cái   lấy   từ   các   vị   trí   1, 8, 15, 22,… của bản mã ở Hình 1 (vurzjugrggugvgjqkeoagugkkqvwqp…). Đây là chuỗi được mã hóa với cùng một mật mã Caesar.  Giải mã chuỗi này với cả 26 mật mã Caesar có thể, lập bảng so sánh phân phối tần số của văn bản được giải mã với phân phối tần số tiếng Anh cho mỗi khóa. Tương ứng, sẽ thu được 26 giá trị thống kê Khi bình phương. Khóa chính xác sẽ tương ứng với văn bản được giải mã với thống kê Khi bình phương thấp nhất. Kết quả cụ thể như Hình 4 đã tìm được chữ cái khóa đầu tiên, theo đó giá trị Khi bình phương nhỏ nhất là 41.22, tương ứng với khóa là 2 (chữ cái ‘c’).  Thám mật mã Vigenère  Hình 4. Giá trị thống kê Khi bình phương của chuỗi giải mã  Tiếp tục tìm 6 chữ cái khóa còn lại theo cách cực tiểu Khi bình phương tương tự để tìm các khóa tương ứng sẽ thu được chuỗi khóa 2,8,0,7,4,17,18. Chuyển về dạng chữ cái là chuỗi 'CIAHERS', chuỗi khóa này bị sai một vị trí. Điều này cho thấy không thể hoàn toàn dựa vào kỹ thuật thám mã này trừ khi thu được bản mã đủ dài. Khóa chính xác trong trường hợp này là 'CIPHERS' và thực tế kiểm tra Khi bình phương có hai giá trị  rất thấp cho dãy con thứ 3. Thật không may, giá trị nhỏ nhất lại không đúng, giá trị khóa đúng có giá trị Khi bình phương lớn hơn giá trị nhỏ nhất một chút.  Thực tế trong kiểm tra Khi bình phương cũng như I.C, xác suất xuất hiện của các chữ cái không phải luôn luôn đúng, chúng gần đúng. Hơn nữa, tần số của các chữ cái trong bản mã không phản ánh chính xác phân phối xác suất các chữ cái trên văn bản mã. Đó là lý do kết quả trên cho ra chữ khóa ‘A’ mà lẽ ra phải là ‘P’. Do đó, việc xem xét thêm các khía cạnh khác như là dựa vào quy luật ngôn ngữ để chỉnh sửa kết quả là rất cần thiết. |  |
|  |  |

* 1. **Giới thiệu về phân quyền**
     1. **Khái niệm**

**Phân quyền** là xu hướng phân tán quyền quyết định trong cơ cấu tổ chức của một doanh nghiệp, phân quyền còn là quá trình chuyển giao nhiệm vụ hay quyền hạn cho bộ phận hay cá nhân có trách nhiệm hoàn thành nhiệm vụ đó. Vậy phân quyền cho nhân viên là việc bạn giao cho nhân viên của mình 1 phần quyền hạn để quyết định và hoàn thành một số công việc nào đó trong bộ phận của doanh nghiệp.

* + 1. **Cách phân quyền**

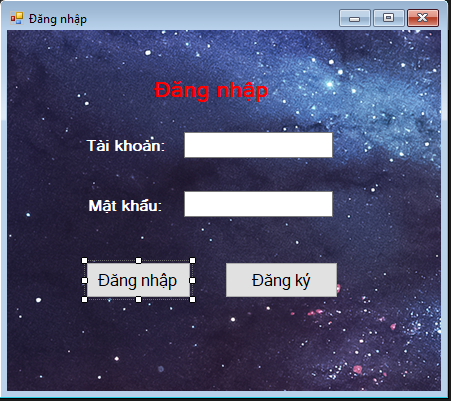
Trong bài viết này mình chỉ hướng dẫn về phân quyền nên mình sẽ làm một csdl nhỏ, chứa các bảng liên quan đến phân quyền như bảng User, bảng quyền hạn, bảng chi tiết quyền hạn và bảng quan hệ với quyền hạn.



Cơ sở dữ liệu phân quyền đơn giản

* tbl\_user: bảng lưu người dùng bao gồm các thuộc tính như ID, Name, Username, Password. Bảng không có khóa ngoại.
* tbl\_permision: bảng chứa nhóm quyền hạn. bao gồm các thuộc tính, ID nhóm quyền hạn, tên nhóm quyền hạn, và mô tả nhóm quyền hạn.
* tbl\_per\_relationship: là bảng lưu mối liên hệ giữa người dùng và nhóm quyền hạn. Mục đích của bảng này không phải là để một người dùng có nhiều nhóm quyền mà để không phải truy vấn lại bảng user chứa thông tin nhạy cảm như username và password.
* tbl\_permision\_detail: là bảng sẽ chứa những quyền hạn cụ thể dành cho nhóm quyền hạn.

1. **Chương trình thực nghiệm**
   1. **Đăng nhập**



Khi đăng nhập người dung cần phải nhập tài khoản và mật khẩu. Hệ thống sẽ lấy thông tin đăng nhập đem so sánh với dữ liệu đã được lưu trong Database.

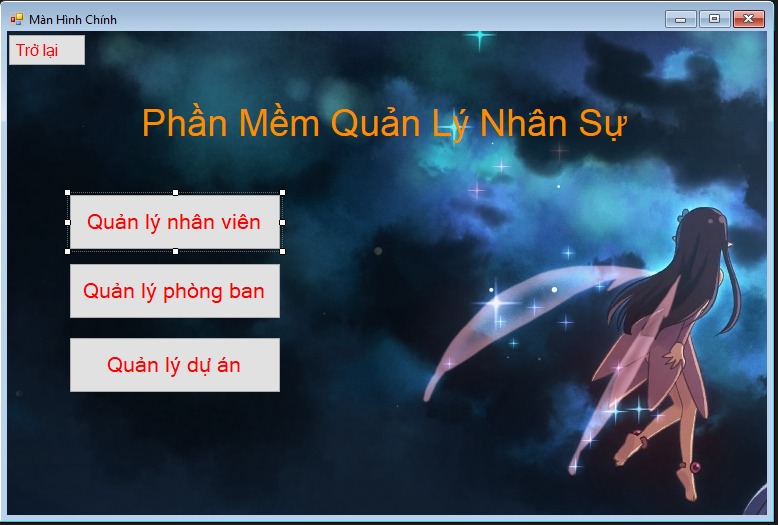
Nếu đúng hệ thống sẽ thông báo đăng nhập thành công và chuyển sang giao diện chính. Nếu người dung đăng nhập sai hệ thống sẽ báo đăng nhập không thành công. Nếu người dung không có tài khoản có thể click vào button Đăng ký để chuyển qua form Đăng ký

* 1. **Đăng ký**



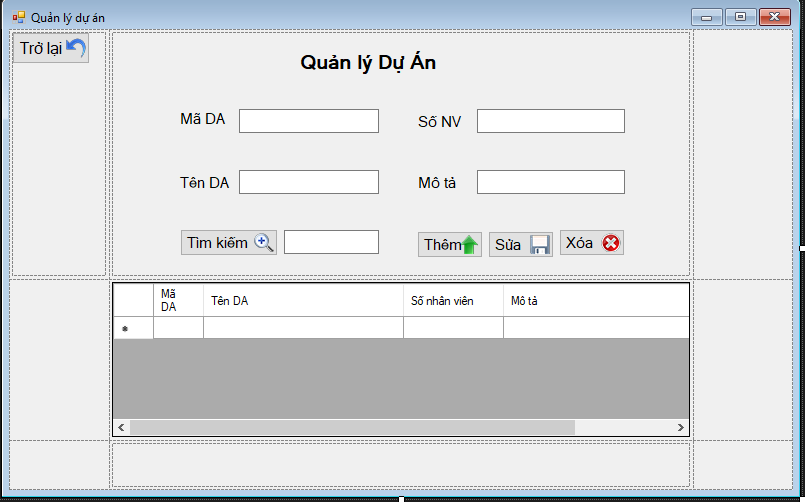
Muốn đăng ký người dùng cần phải điền đầy đủ thông tin tài khoản, mã nhân viên, mật khẩu và nhập lại mật khẩu một lần nữa. Người dùng cần mật khẩu và nhập lại giống nhâu nếu khác nhâu hệ thống sẽ thong báo Mật khẩu nhập lại không đúng va bắt người dùng nhập lại. Nếu tên tài khoản bị trùng với tài khoản dã có hệ thống sẽ thông báo và bắt nhập lại. Khi người dùng đăng ký thành công hệ thống sẽ lưu thông tin người dùng và mật khẩu đã được mã hóa và database.

* 1. **Màn hình chính**



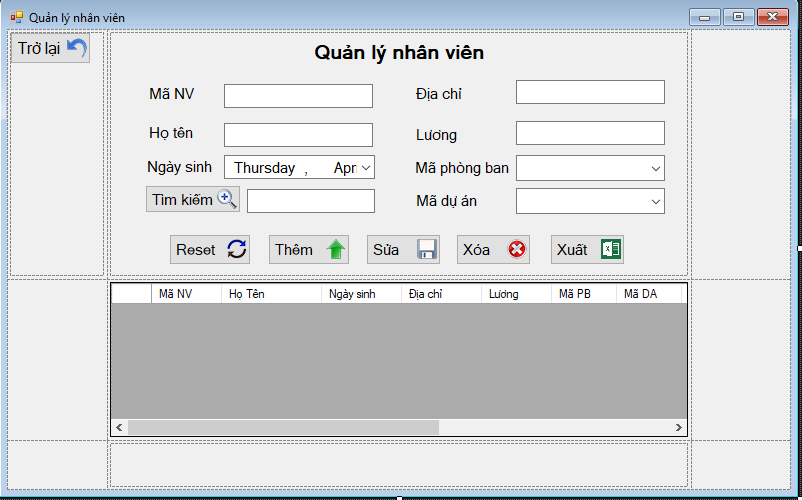
Khi người dùng là addmin hệ thống sẽ hiển thị tất cả các chức năng Quản lý nhân viên, Quản lý phòng ban, Quản lý dự án.còn nếu người dùng là nhân viên thì giao diện sẽ khóa chúc năng Quản lý nhân viên, Quản lý phòng ban lại.

* 1. **Quản lý dự án**



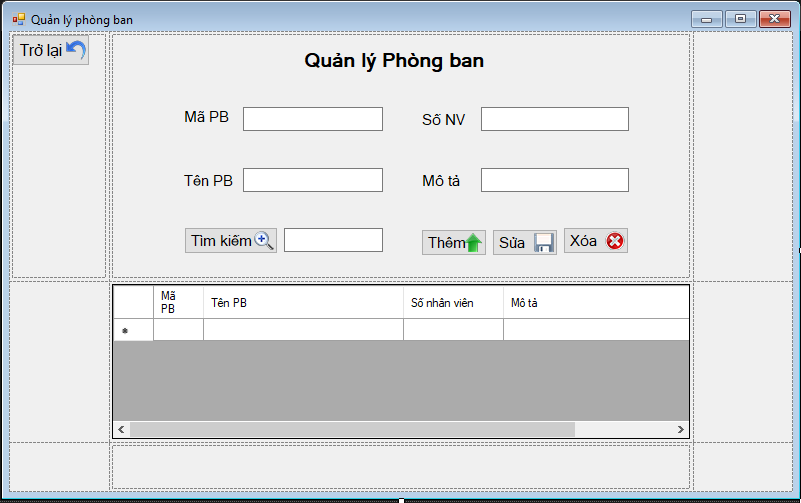
Người dùng có thể xem thông tin mã dự án, tên dự án, số nhân viên làm đự án. Người dùng củng có thể tìm kiếm dự án theeo tên dự án hoặc mã dự án. Người dungf củ có thêm, xóa, sửa ccacs dự án.

* 1. **Quản lý nhân viên**



Addmin có khả năng truy cập và quản lý nhân viên của mình. Addmin có khả năng xem thông tin và sửa thông tin của nhân viên. Khi có nhân viên nghic việc Addmin có thể xóa nhân viên đó ra khổi hệ thống. Khi cần thông tin dữ liệu trên giấy addmin có thể xuất dữ liệu nhân viên ra excel.

* 1. **Quản lý phòng ban**



Addmin có thể xem được các phong ban khác nhau. Addmin cũng có thể tìm kiếm theo mã phòng ban, tên phòn ban. Addmin được phép thêm, xóa, sữa nội dung phòng ban.

* 1. **Quản lý tài khoản**



Addmin có thể kiểm tra lịch sữ hoạt động của các tài khoản. Addmin có thẻ tìm kiếm nhân viên theo mã nhân viên, chức vụ để dễ quản lý. Addmin củng có thể thay đổi chức vụ của nhân viên trong hệ thống