

Mục lục

Danh mục hình vẽ	2
1. Tính cấp thiết	3
2. Mục tiêu	5
3. Cách tiếp cận.....	5
4. Phương pháp nghiên cứu:.....	5
5. Phạm vi nghiên cứu:.....	5
6. Nội dung nghiên cứu và kết quả đạt được:.....	5
6.1. Tổng quan về công cụ lập trình Visual Studio	5
6.1.1 Cấu trúc Visual studio	6
6.1.2 Trình biên dịch mã nguồn	9
6.1.3 Trình gỡ lỗi	10
6.1.4. Thiết kế	11
6.1.5 Lịch sử Visual Studio.....	12
6.2 Tính năng mới trong VS2022.....	16
6.2.1 Cải tiến hiệu suất.....	16
6.2.2 Visual Studio 2022 là 64-bit	16
6.2.3 Tìm trong tệp nhanh hơn.....	16
6.2.3 Xây dựng các ứng dụng hiện đại.....	17
6.2.4 Các công cụ dành cho nhà phát triển tốt hơn cho C ++ và .NET và Hot Reload.....	17
6.2.5 Hỗ trợ nhiều repo với Git trong IDE	17
6.2.6 Cải tiến IntelliCode	17
6.2.7 Thiết kế cho mọi người	18
6.4. Kết luận và kiến nghị	18
Danh sách từ viết tắt.....	19
Tài liệu tham khảo.....	20

Danh mục hình vẽ

Hình 1. Logo của các ngôn ngữ lập trình phổ biến hiện nay (Dammio 2019).....	4
Hình 2. Bảng xếp hạng độ phổ biến của các ngôn ngữ lập trình (root 2019).....	4
Hình 3. Mở rộng Visual Studio cho MAC.....	7
Hình 4. Visual Studio Code remote	8
Hình 5. Giao diện soạn thảo mã nguồn.....	9
Hình 6. Tìm kiếm thông minh.....	Error! Bookmark not defined.
Hình 7. Tái cấu trúc trong VS	Error! Bookmark not defined.

1. Tính cấp thiết

Hiện nay, thay vì phải thực hiện toàn bộ công việc lập trình theo phương pháp thủ công, lập trình viên có thể sử dụng sự hỗ trợ đến từ các loại công cụ lập trình. Những công cụ viết phần mềm này không chỉ đảm bảo chất lượng mà còn giúp bạn rút ngắn thời gian hoàn thiện việc phát triển các phần mềm máy tính.

Về bản chất, các công cụ viết phần mềm hay IDE – môi trường phát triển tích hợp đều dùng để chỉ các phần mềm máy tính có chức năng hỗ trợ lập trình viên tạo ra các phần mềm khác.

Một công cụ lập trình cơ bản thông thường sẽ bao gồm những thành phần sau:

- Trình soạn thảo mã: Sử dụng để viết mã code cho phần mềm;
- Trình biên dịch, thông dịch: Sử dụng để chuyển đổi các câu lệnh từ ngôn ngữ lập trình sang ngôn ngữ mà máy tính có thể hiểu được và đảm bảo các mã lệnh sẽ được thực hiện chính xác;
- Trình gỡ lỗi: Hỗ trợ dò tìm, phát hiện lỗi trong quá trình soạn thảo mã;

Bên cạnh đó, tùy thuộc vào từng loại công cụ viết code và phần mềm IDE khác nhau, phần mềm sẽ còn được tích hợp thêm giao diện người dùng đồ họa GUI, trình duyệt lớp, trình quản lý đối tượng,...

Số lượng và khả năng hoạt động hiệu quả của các công cụ thành phần sẽ là cơ sở để lập trình viên đánh giá, lựa chọn những loại phần mềm viết code chất lượng, nên được ứng dụng cho công việc của chính họ.

Hiện tại có rất nhiều ngôn ngữ lập trình, mỗi ngôn ngữ phù hợp với một môi trường và dự án khác nhau.

Trong phạm vi của báo cáo này, tôi trình bày tóm tắt về các điểm mới đột phá của bộ công cụ Visual Studio 2022.

2. Mục tiêu

Nắm được các kỹ thuật mới của bộ công cụ Visual Studio năm 2022.

3. Cách tiếp cận

Đề tài tiếp cận vấn đề nhằm đạt đến mục tiêu từ các góc độ sau:

- Tiếp cận từ thực tiễn: khảo sát, đánh giá, phát hiện những điểm yếu hiện tại trong các ứng dụng, các phiên bản Visual Studio hiện tại
- Tiếp cận từ cơ sở lý thuyết: khảo sát, đánh giá mức độ tốt của các phương pháp dịch máy, để từ đó cải tiến, đề xuất định dạng mới, phù hợp với mục tiêu đề ra
- Tiếp cận từ những xu hướng phát triển công nghệ hiện đại

4. Phương pháp nghiên cứu:

Phương pháp phân tích lý thuyết: Nghiên cứu Lý thuyết tổng quan về lập trình.

Phương pháp thực nghiệm: thực hiện trực tiếp các tính năng mới, phân tích đánh giá kết quả.

5. Phạm vi nghiên cứu:

Bộ công cụ Visual Studio 2022.

6. Nội dung nghiên cứu và kết quả đạt được:

6.1. Tổng quan về công cụ lập trình Visual Studio

Microsoft Visual Studio là một môi trường phát triển tích hợp (IDE) từ Microsoft. 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.

Visual Studio bao gồm một trình soạn thảo mã hỗ trợ IntelliSense cũng như cải tiến mã nguồn. Trình gỡ lỗi tích hợp hoạt động cả về trình gỡ lỗi mức độ mã nguồn và gỡ 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.

Visual Studio hỗ trợ nhiều ngôn ngữ lập trình khác nhau và cho phép trình biên tập mã và gỡ lỗi để hỗ trợ (mức độ khác nhau) hầu như mọi ngôn ngữ lập trình. Các ngôn ngữ tích hợp gồm có C, C++ và C++/CLI (thông qua Visual C++), VB.NET (thông qua Visual Basic.NET), C# (thông qua Visual C#) và F# (như của Visual Studio 2010). Hỗ trợ cho các ngôn ngữ khác như J++/J#, Python và Ruby thông qua dịch vụ cài đặt riêng rẽ. Nó cũng hỗ trợ XML/XSLT, HTML/XHTML, JavaScript và CSS.

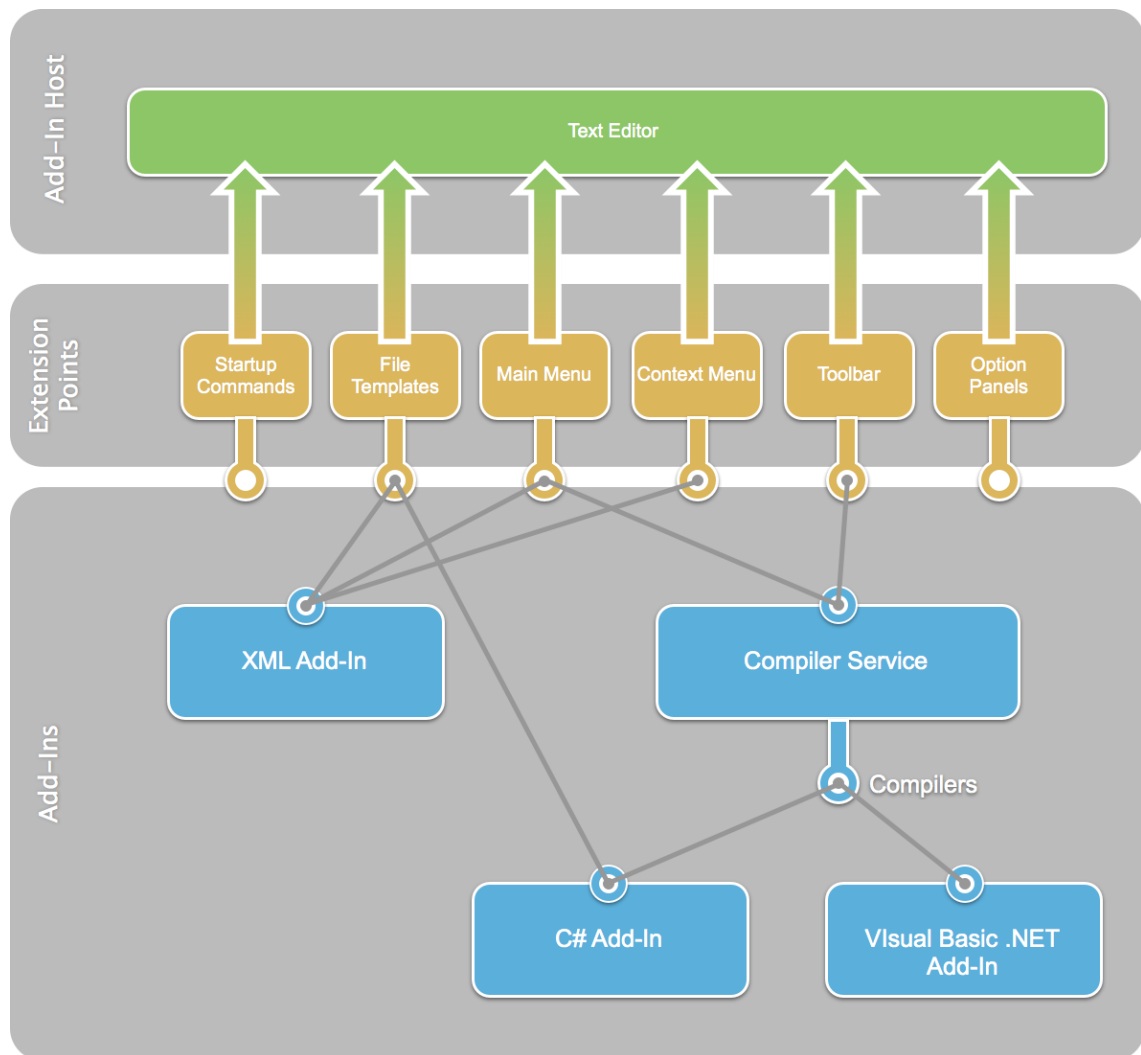
Microsoft cung cấp phiên bản "Express" (đối với phiên bản Visual Studio 2013 trở về trước) và "Community" (đối với bản Visual Studio 2015 trở về sau) là phiên bản miễn phí của Visual Studio.

6.1.1 Cấu trúc Visual studio

Visual Studio không hỗ trợ cho bất kỳ ngôn ngữ lập trình nào về giải pháp hoặc công cụ thực chất, thay vào đó nó cho phép cắm chức năng được mã hóa như là một VSPackage. Khi cài đặt, các chức năng có sẵn như là một dịch vụ. IDE cung cấp ba dịch vụ: SVsSolution cung cấp khả năng liệt kê các dự án và các giải pháp; SVsUIShell cung cấp cửa sổ và giao diện người dùng và SVsShell. Ngoài ra, IDE cũng có trách nhiệm điều phối và cho phép truyền thông giữa các dịch vụ. Tất cả các biên tập viên, nhà thiết kế, các loại dự án và các công cụ khác được thực hiện theo VSPackages. Visual Studio sử dụng COM để truy cập VSPackages. Visual Studio SDK cũng bao gồm Managed Package Framework (MPF) là một tập hợp quản lý bao bọc quanh các COM-interfaces cho phép các gói được viết bằng bất kỳ ngôn ngữ nào. Tuy nhiên, MPF không cung cấp tất cả các chức năng bọc lộ trong Visual Studio COM-interfaces. Các dịch vụ có thể được tiêu thụ để tạo ra các gói khác, để thêm chức năng cho Visual Studio IDE.

Ngoài môi trường truyền thống Window, Visual Studio còn có thể chạy trên môi trường MAC và UNIX.

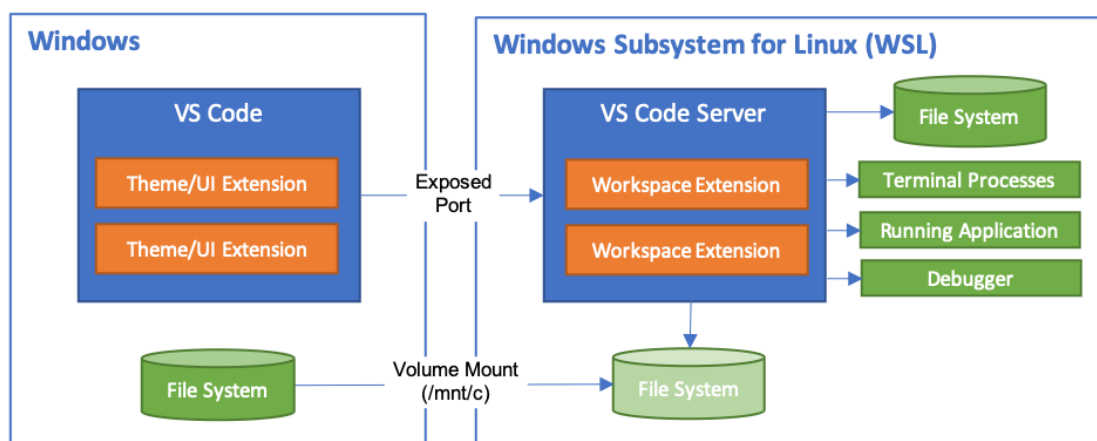
Để tùy chỉnh Visual Studio cho Mac, bạn có thể tạo gói tiện ích mở rộng xây dựng từ các điểm mở rộng có trong máy chủ hỗ trợ trong các thư viện có sẵn trong Visual Studio cho Mac, như được minh họa bằng sơ đồ sau (heiligerdankgesang):



Hình 3. Mở rộng Visual Studio cho MAC

Visual Studio Code Remote - Tiện ích mở rộng WSL cho phép bạn sử dụng Hệ thống con Windows cho Linux (WSL) làm môi trường phát triển toàn thời gian của bạn ngay từ Mã VS. Bạn có thể phát triển trong môi trường dựa trên Linux, sử dụng các công cụ và tiện ích dành riêng cho Linux, đồng thời chạy và gỡ lỗi các ứng dụng dựa trên Linux của bạn một cách thoải mái với Windows.

Tiện ích mở rộng chạy các lệnh và các tiện ích mở rộng khác trực tiếp trong WSL để bạn có thể chỉnh sửa các tệp nằm trong WSL hoặc hệ thống tệp Windows được gắn (ví dụ /mnt/c) mà không phải lo lắng về các sự cố đường dẫn, khả năng tương thích nhị phân hoặc các thách thức hệ điều hành chéo khác.



Hình 4. Visual Studio Code remote

Hỗ trợ cho các ngôn ngữ lập trình được thêm vào bằng cách sử dụng một VSPackage đặc biệt được gọi là một dịch vụ ngôn ngữ. Một dịch vụ ngôn ngữ định nghĩa giao tiếp khác nhau mà việc thực hiện VSPackage có thể thực hiện để hỗ trợ thêm cho các chức năng khác nhau. Các chức năng có thể được thêm vào theo cách này bao gồm cú pháp màu, hoàn thành báo cáo kết quả, kết hợp đôi, công cụ chú giải tham số thông tin, danh sách thành viên và đánh dấu lỗi trên nền biên dịch.[10] Nếu giao diện được thực hiện, các tính năng sẽ có sẵn ngôn ngữ. Dịch vụ ngôn ngữ sẽ được thực hiện trên cơ sở mỗi ngôn ngữ. Việc triển khai có thể tái sử dụng mã từ phân tích cú pháp hoặc trình biên dịch cho ngôn ngữ. Dịch vụ ngôn ngữ có thể được triển khai hoặc trong mã nguồn gốc hoặc mã số quản lý. Đối với mã nguồn gốc, thì cả COM-interfaces gốc hoặc Babel Framework (một phần của Visual Studio SDK) đều có thể được sử dụng. Đối với mã số quản lý thì các MPF sẽ bao hàm các dịch vụ quản lý văn bản.

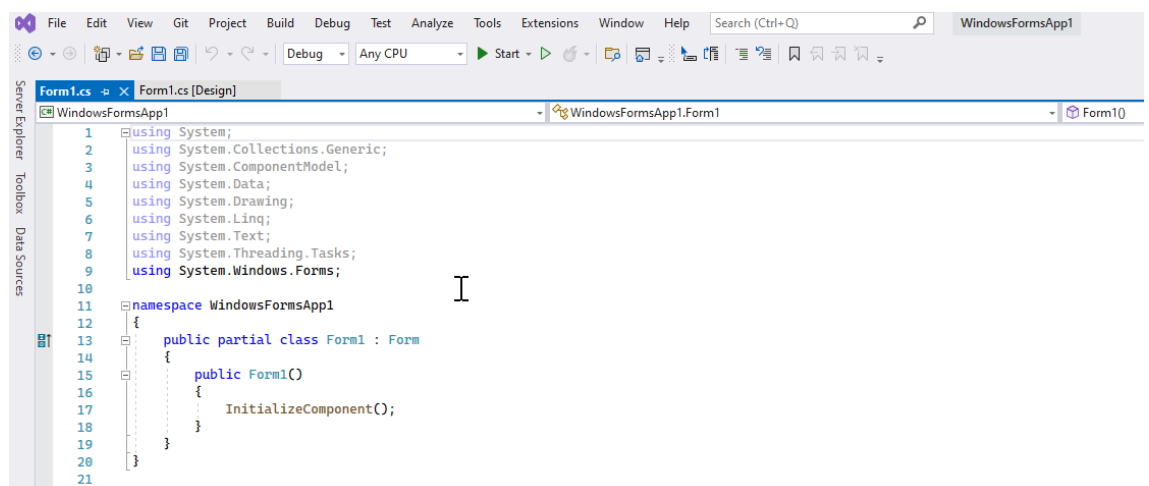
Visual Studio không bao gồm bất kỳ Hệ thống quản lý phiên bản hỗ trợ kiểm soát mã nguồn nhưng nó xác định hai cách thay thế cho các hệ thống kiểm soát mã nguồn để tích hợp với IDE. Một VSPackage kiểm soát mã nguồn có thể cung cấp giao diện người dùng tùy chỉnh của riêng mình. Ngược lại, một plugin kiểm soát mã nguồn bằng cách sử dụng MSSCCI (Microsoft Source Code Control Interface) cung cấp một tập các chức năng được sử dụng để thực hiện chức năng kiểm soát mã nguồn khác nhau, với một giao diện người dùng Visual Studio tiêu chuẩn. MSSCCI lần đầu tiên được sử dụng để tích hợp Visual SourceSafe với Visual Studio 6.0 nhưng sau đó được mở ra thông qua Visual Studio SDK. Visual Studio.NET 2002 dùng MSSCCI 1.1, và Visual Studio.NET 2003 dùng MSSCCI 1.2. Visual Studio 2005, 2008 và 2010 dùng MSSCCI 1.3.

Visual Studio hỗ trợ chạy nhiều cá thể của môi trường (tất cả đều có VSPackages riêng của mình). Những trường hợp sử dụng các registry hives khác nhau để lưu trữ trạng thái cấu hình và được phân biệt bởi AppID (Application ID). Các trường

hợp được đưa ra bởi một AppId-specific.exe cụ thể mà lựa chọn AppID, thiết lập các hive gốc và khởi chạy IDE. VSPackages đăng ký một AppID được tích hợp với VSPackages khác cho AppID đó. Các phiên bản sản phẩm khác nhau của Visual Studio được tạo ra bằng cách sử dụng AppIds khác nhau. Các sản phẩm phiên bản Visual Studio Express được cài đặt với AppIds riêng nhưng với các sản phẩm Standard, Professional và Team Suite chia sẻ cùng AppID. Do đó, người ta có thể cài đặt các phiên bản Express song song với các phiên bản khác, không giống như các phiên bản khác cập nhật các cài đặt tương tự. Phiên bản Professional bao gồm các VSPackages khổng lồ trong phiên bản Standard và Team. Hệ thống AppID được thừa hưởng bởi Visual Studio Shell trong Visual Studio 2008.

6.1.2 Trình biên dịch mã nguồn

Giống như bất kỳ IDE khác, nó bao gồm một trình soạn thảo mã hỗ trợ tô sáng cú pháp và hoàn thiện mã bằng cách sử dụng IntelliSense không chỉ cho các biến, hàm và các phương pháp mà còn các cấu trúc ngôn ngữ như vòng điều khiển hoặc truy vấn. IntelliSense được hỗ trợ kèm theo cho các ngôn ngữ như XML, Cascading Style Sheets và JavaScript khi phát triển các trang web và các ứng dụng web. Các đề xuất tự động hoàn chỉnh được xuất hiện trong một hộp danh sách phủ lên trên đỉnh của trình biên tập mã. Trong Visual Studio 2008 trở đi, nó có thể được tạm thời bán trong suốt để xem mã che khuất bởi nó. Các trình biên tập mã được sử dụng cho tất cả các ngôn ngữ được hỗ trợ.



Hình 5. Giao diện soạn thảo mã nguồn

Các trình biên tập mã Visual Studio cũng hỗ trợ cài đặt dấu trang trong mã để điều hướng nhanh chóng. Hỗ trợ điều hướng khác bao gồm thu hẹp các khối mã

lệnh và tìm kiếm gia tăng, ngoài việc tìm kiếm văn bản thông thường và tìm kiếm Biểu thức chính quy. Các trình biên tập mã cũng bao gồm một bìa kẹp đa mục và một danh sách công việc. Các trình biên tập mã hỗ trợ lưu lại các đoạn mã được lặp đi lặp lại nhằm để chèn vào mã nguồn sử dụng về sau. Một công cụ quản lý cho đoạn mã được xây dựng là tốt. Những công cụ này nổi lên như các cửa sổ trôi nổi có thể được thiết lập để tự động ẩn khi không sử dụng hoặc neo đậu đến các cạnh của màn hình. Các trình biên tập mã Visual Studio cũng hỗ trợ cải tiến mã nguồn bao gồm tham số sắp xếp lại, biến và phương pháp đổi tên, khai thác và đóng gói giao diện các lớp thành viên bên trong những trạng thái giữa những thứ khác.

Visual Studio có tính năng biên dịch nền (còn gọi là biên dịch gia tăng). Như mã đang được viết, Visual Studio biên dịch nó trong nền để cung cấp thông tin phản hồi về cú pháp và biên dịch lỗi, được đánh dấu bằng một gạch dưới gợn sóng màu đỏ. Biên dịch nền không tạo ra mã thực thi, vì nó đòi hỏi một trình biên dịch khác hơn là để sử dụng tạo ra mã thực thi. Biên dịch nền ban đầu được giới thiệu với Microsoft Visual Basic nhưng bây giờ đã được mở rộng cho tất cả các ngôn ngữ.

6.1.3 Trình gỡ lỗi

Visual Studio có một trình gỡ lỗi hoạt động vừa là một trình gỡ lỗi cấp mã nguồn và là một trình gỡ lỗi cấp máy. Nó hoạt động với cả hai mã quản lý cũng như ngôn ngữ máy và có thể được sử dụng để gỡ lỗi các ứng dụng được viết bằng các ngôn ngữ được hỗ trợ bởi Visual Studio. Ngoài ra, nó cũng có thể đính kèm theo quy trình hoạt động và theo dõi và gỡ lỗi những quy trình. Nếu mã nguồn cho quá trình hoạt động có sẵn, nó sẽ hiển thị các mã như nó đang được chạy. Nếu mã nguồn không có sẵn, nó có thể hiển thị các tháo gỡ. Các Visual Studio debugger cũng có thể tạo bãi bộ nhớ cũng như tải chúng sau để gỡ lỗi. Các chương trình đa luồng cao cấp cũng được hỗ trợ. Trình gỡ lỗi có thể được cấu hình sẽ được đưa ra khi một ứng dụng đang chạy ngoài Visual Studio bị treo môi trường.

Trình gỡ lỗi cho phép thiết lập các breakpoint (mà cho phép thực thi được tạm thời dừng lại tại một vị trí nhất định) và watch (trong đó giám sát các giá trị của biến là việc thực hiện tiến bộ). Breakpoint có thể có điều kiện, nghĩa là chúng được kích hoạt khi điều kiện được đáp ứng. Mã có thể được biểu diễn, tức là chạy một dòng (của mã nguồn) tại một thời điểm. Nó có hoặc là bước sang các chức năng để gỡ lỗi bên trong nó, hoặc là nhảy qua nó, tức là, việc thực hiện các chức năng không có sẵn để kiểm tra thủ công. Trình gỡ lỗi hỗ trợ Edit and Continue, nghĩa là, nó cho phép mã được chỉnh sửa khi nó đang được sửa lỗi (chỉ có 32 bit, không được hỗ trợ trong 64 bit). Khi gỡ lỗi, nếu con trỏ chuột di chuyển lên bất kỳ biến, giá trị hiện tại của nó được hiển thị trong phần chú giải ("chú thích dữ liệu"), nơi mà nó cũng có thể được thay đổi nếu muốn. Trong quá trình viết mã, các trình gỡ lỗi của Visual Studio cho phép một số chức năng được gọi ra bằng tay từ cửa sổ công cụ Immediate. Các thông số cho phương thức được cung cấp tại các cửa sổ Immediate.

6.1.4. Thiết kế

Windows Forms Designer: được sử dụng để xây dựng GUI sử dụng Windows Forms; bố trí có thể được xây dựng bằng các nút điều khiển bên trong hoặc khóa chúng vào bên cạnh mẫu. Điều khiển trình bày dữ liệu (như hộp văn bản, hộp danh sách, vv) có thể được liên kết với các nguồn dữ liệu như cơ sở dữ liệu hoặc truy vấn. Các điều khiển dữ liệu ràng buộc có thể được tạo ra bằng cách rê các mục từ cửa sổ nguồn dữ liệu lên bề mặt thiết kế. Các giao diện người dùng được liên kết với mã sử dụng một mô hình lập trình hướng sự kiện. Nhà thiết kế tạo ra bằng C# hoặc VB.NET cho ứng dụng.

WPF Designer: có tên mã là Cider, được giới thiệu trong Visual Studio 2008. Giống như Windows Forms Designer, hỗ trợ kéo và thả ẩn dụ. Sử dụng tương tác người-máy nhằm mục tiêu theo Windows Presentation Foundation. Nó hỗ trợ các chức năng WPF bao gồm kết nối dữ liệu và tự động hóa bố trí quản lý. Nó tạo ra mã XAML cho giao diện người dùng. Các tập tin XAML được tạo ra là tương thích với Microsoft Expression Design, sản phẩm thiết kế theo định hướng. Các mã XAML được liên kết với mã đang sử dụng một mô hình code-behind.

Web designer/development: Visual Studio cũng bao gồm một trình soạn thảo và thiết kế trang web cho phép các trang web được thiết kế bằng cách kéo và thả các đối tượng. Nó được sử dụng để phát triển các ứng dụng ASP.NET và hỗ trợ HTML, CSS và JavaScript. Nó sử dụng mô hình code-behind để liên kết với mã ASP.NET. Từ Visual Studio 2008 trở đi, công cụ bố trí được sử dụng bởi các nhà thiết kế web được chia sẻ với Microsoft Expression Web. Ngoài ra ASP.NET MVC Framework hỗ trợ cho công nghệ MVC là tải xuống riêng biệt và dự án ASP.NET Dynamic Data có sẵn từ Microsoft.

Class designer: Các lớp thiết kế được dùng để biên soạn và chỉnh sửa các lớp (bao gồm cả các thành viên và truy cập của chúng) sử dụng mô hình UML. Các lớp thiết kế có thể tạo ra mã phác thảo C# và VB.NET cho các lớp và các phương thức. Nó cũng có thể tạo ra sơ đồ lớp từ các lớp viết tay.

Data designer: Thiết kế dữ liệu có thể được sử dụng để chỉnh sửa đồ họa giản đồ cơ sở dữ liệu bao gồm các bảng, khóa chính, khóa ngoại và các ràng buộc. Nó cũng có thể được sử dụng để thiết kế các truy vấn từ các giao diện đồ họa.

Mapping designer: Từ Visual Studio 2008 trở đi, thiết kế ánh xạ được dùng bởi Language Integrated Query để thiết kế các ánh xạ giữa các giản đồ cơ sở dữ liệu và các lớp để đóng gói dữ liệu. Các giải pháp mới từ cách tiếp cận ORM, ADO.NET Entity Framework sẽ thay thế và cải thiện các công nghệ cũ.

6.1.5 Lịch sử Visual Studio

Tên chính thức là Visual Studio 97. Vào năm 1997, Visual Studio ra mắt, tên mã là Boston (tên thành phố, Microsoft có thói quen đặt tên mã của Visual Studio theo tên địa danh). Visual Studio 97 là sự kết hợp của nhiều công cụ lập trình với nhau. Visual Studio 97 xuất hiện trong hai phiên bản: Visual Studio Professional và Visual Studio Enterprise, phiên bản chuyên nghiệp chứa trên ba đĩa CD và phiên bản doanh nghiệp chứa trên bốn đĩa CD. Nó bao gồm Visual J++ 1.1 cho các lập trình viên Java, cũng như đi kèm với Visual InterDev để tạo ra các trang web được tạo tự động bằng Active Server Pages. Đi kèm các CD cài đặt là một CD chứa thư viện Mạng Microsoft Developer Network.

Phiên bản kế tiếp có tên chính thức là Visual Studio 6.0, tên mã Aspen (khu nghỉ mát trượt tuyết ở Colorado), được phát hành vào tháng 6 năm 1998 và là phiên bản cuối cùng chạy trên nền Windows 9x. Mỗi phiên bản của mỗi thành phần cũng được nâng lên thành v6.0, bao gồm Visual J++ trước đó là v1.1 và Visual InterDev tại lần phát hành đầu tiên. Phiên bản v6 của Microsoft là môi trường cốt lõi cho bốn phiên bản tiếp theo nhằm cung cấp cho các lập trình một nền tảng lập trình giống nhau. Điều này đã khiến Microsoft tập trung sự phát triển vào nền tảng độc lập .NET Framework.

Vào tháng 2 năm 2002, Microsoft đã phát hành Visual Studio.NET, có tên mã là Rainier (Mount Rainier của Washington). Phiên bản beta đã được phát hành thông qua MSDN vào năm 2001. Thay đổi lớn nhất là việc giới thiệu một môi trường phát triển mã được quản lý bằng .NET Framework. Các chương trình phát triển sử dụng .NET không được biên dịch thành ngôn ngữ máy (như C++ chẳng hạn) mà thay vào đó là một định dạng gọi là Microsoft Intermediate Language (MSIL) hoặc Ngôn ngữ trung gian dùng chung (CIL). Khi một ứng dụng CIL được thực thi, nó được biên dịch vào ngôn ngữ máy phù hợp với nền tảng đang chạy, do đó mã nguồn có thể sử dụng cho nhiều nền tảng khác nhau. Các chương trình biên soạn vào CIL chỉ có thể được thực thi trên các nền tảng có cơ sở ngôn ngữ dùng chung. Có thể chạy các chương trình CIL trong Linux hoặc Mac OS X sử dụng các ứng dụng không phải Microsoft.NET như Mono và DotGNU.

Vào tháng 4/2003, Visual Studio.NET 2003 ra mắt, có tên mã là Everett. Ở bản này, .NET Framework được nâng cấp lên phiên bản 1.1. Đây cũng là phiên bản Visual Studio đầu tiên hỗ trợ phát triển các chương trình cho các thiết bị di động, sử dụng ASP.NET hoặc .NET Compact Framework. Các tiêu chuẩn tuân thủ của Visual C++ được cải thiện. Visual C++ Toolkit 2003 được bán kèm với Visual Studio.NET 2003 mà không có IDE. Tính đến năm 2010, bộ này không còn tồn tại nữa và thay thế nó là Express Editions. Số phiên bản của Visual Studio.NET 2003 là phiên bản 7.1, số phiên bản định dạng tệp là 8.0.

Visual Studio 2005 có tên mã Whidbey (đảo Whidbey ở Puget Sound), được phát hành trực tuyến từ tháng 10 năm 2005. Kể từ phiên bản này Microsoft loại bỏ

tên gọi ".NET" ở tên sản phẩm (cũng như mọi sản phẩm khác có.NET), nhưng Visual Studio vẫn chủ yếu nhắm mục tiêu vào.NET Framework (lúc này đã được nâng cấp lên phiên bản 2.0). Đây là phiên bản mới nhất có sẵn cho Windows 2000 và cũng là phiên bản cuối cùng để có thể phát triển ứng dụng C++ cho Windows 98, Windows Me và Windows NT 4.0.

Visual Studio 2008 và Visual Studio Team System 2008 có tên mã Orcas (đảo Orcas, cũng là một hòn đảo ở Puget Sound), bắt đầu phát hành cho các thuê bao MSDN vào ngày 19 tháng 11 năm 2007 cùng với .NET Framework 3.5. Mã nguồn cho Visual Studio 2008 IDE có sẵn dưới giấy phép nguồn chia sẻ cho một số đối tác của Microsoft. Microsoft phát hành gói dịch vụ 1 cho Visual Studio 2008 vào ngày 11 tháng 8 năm 2008. Số phiên bản nội bộ của Visual Studio 2008 là 9.0, số phiên bản định dạng tệp là 10.0. Visual Studio 2008 là phiên bản cuối cùng hỗ trợ cho các ứng dụng C++ chạy trên Windows 2000.

Visual Studio Ultimate 2010 thay thế Visual Studio 2008 Team Suite. Nó bao gồm các công cụ mô hình hóa mới, như Architecture Explorer, hiển thị đồ họa các dự án và các lớp và các mối quan hệ giữa chúng. Nó hỗ trợ sơ đồ hoạt động UML, sơ đồ thành phần, sơ đồ lớp (logic), sơ đồ trình tự và sơ đồ ca sử dụng. Visual Studio Ultimate 2010 cũng bao gồm Phân tích tác động thử nghiệm cung cấp gợi ý về các trường hợp thử nghiệm bị ảnh hưởng bởi các sửa đổi đối với mã nguồn, mà không thực sự chạy các trường hợp thử nghiệm. Điều này tăng tốc độ kiểm tra bằng cách tránh chạy các trường hợp kiểm tra không cần thiết (2019b).

Bản dựng cuối cùng của Visual Studio 2012 đã được công bố vào ngày 1 tháng 8 năm 2012 và sự kiện ra mắt chính thức được tổ chức vào ngày 12 tháng 9 năm 2012. Không giống như các phiên bản trước, Visual Studio 2012 không thể ghi và phát macro và trình chỉnh sửa macro đã bị xóa. Các tính năng mới bao gồm hỗ trợ WinRT và C++ / CX (Phần mở rộng thành phần) và C++ AMP (lập trình GPGPU).

Bản xem trước cho Visual Studio 2013 đã được công bố tại hội nghị Build 2013 và được cung cấp vào ngày 26 tháng 6 năm 2013. Visual Studio 2013 RC

(Ứng cử viên phát hành) đã được cung cấp cho các nhà phát triển trên MSDN vào ngày 9 tháng 9 năm 2013.

Bản đầu được gọi là Visual Studio "14", Bản xem trước công nghệ cộng đồng (CTP) đầu tiên được phát hành vào ngày 3 tháng 6 năm 2014 và Ứng viên phát hành được phát hành vào ngày 29 tháng 4 năm 2015; Visual Studio 2015 đã chính thức được công bố là tên cuối cùng vào ngày 12 tháng 11 năm 2014.

Visual Studio 2015 RTM được phát hành vào ngày 20 tháng 7 năm 2015. Visual Studio 2015 Update 1 được phát hành vào ngày 30 tháng 11 năm 2015. Visual Studio 2015 Update 2 được phát hành vào ngày 30 tháng 3 năm 2016. Visual Studio 2015 Update 3 được phát hành vào ngày 27 tháng 6 năm 2016.

Phiên bản xem trước đầu tiên được phát hành vào ngày 30 tháng 3 năm 2016 với cái tên là Visual Studio "15". Ngày 14 tháng 11 năm 2016, Microsoft đăng một bài blog tiết lộ tên sản phẩm Visual Studio 2017 cùng với các tính năng sắp tới. Sau đó, vào ngày 16 tháng 11 năm 2016, "Visual Studio 2017" đã được công bố là tên cuối cùng của sản phẩm và Visual Studio 2017 RC. Cuối cùng, vào ngày 7 tháng 3 năm 2017, Visual Studio 2017 RTM chính thức ra mắt công chúng. Visual Studio 2017 tổng hợp các cập nhật của các bản preview, cũng như bổ sung .NET Core và ASP.NET Core. Bản này cũng cung cấp các tính năng mới như hỗ trợ EditorConfig (một khuôn khổ cho việc thực thi mã hóa), hỗ trợ NGen, công cụ .NET Core và Docker và Xamarin 4.3. Ở phiên bản này, XAML Editor và IntelliSense được cải tiến, unit test trực tiếp, gỡ lỗi nâng cao, và cải thiện tổng thể hiệu suất của IDE.

Vào ngày 4 tháng 12 năm 2018 Visual Studio 2019 Preview 1 đã được phát hành. Vào ngày 24 tháng 1 năm 2019 Visual Studio 2019 Preview 2 đã được phát hành. Vào ngày 13 tháng 2 năm 2019 Visual Studio 2019 Preview 3 đã được phát hành. Vào ngày 27 tháng 2 năm 2019 Visual Studio 2019 RC đã được phát hành. VS2019 thường có sẵn (GA) kể từ ngày 2 tháng 4 năm 2019 và có sẵn để tải xuống.

Tháng 11 năm 2021, bản Visual Studio 2022 đã được phát hành².

6.2 Tính năng mới trong VS2022

Với Visual Studio 2022, chúng ta sẽ luôn nhận được các công cụ và dịch vụ tốt nhất hiện có cho bất kỳ nhà phát triển nào, bất kỳ ứng dụng nào và bất kỳ nền tảng nào. Cho dù chúng ta đang sử dụng Visual Studio lần đầu tiên hay là người đã sử dụng nó trong nhiều năm, có rất nhiều điều để thích trong phiên bản mới nhất của Visual Studio 2022.

6.2.1 Cải tiến hiệu suất

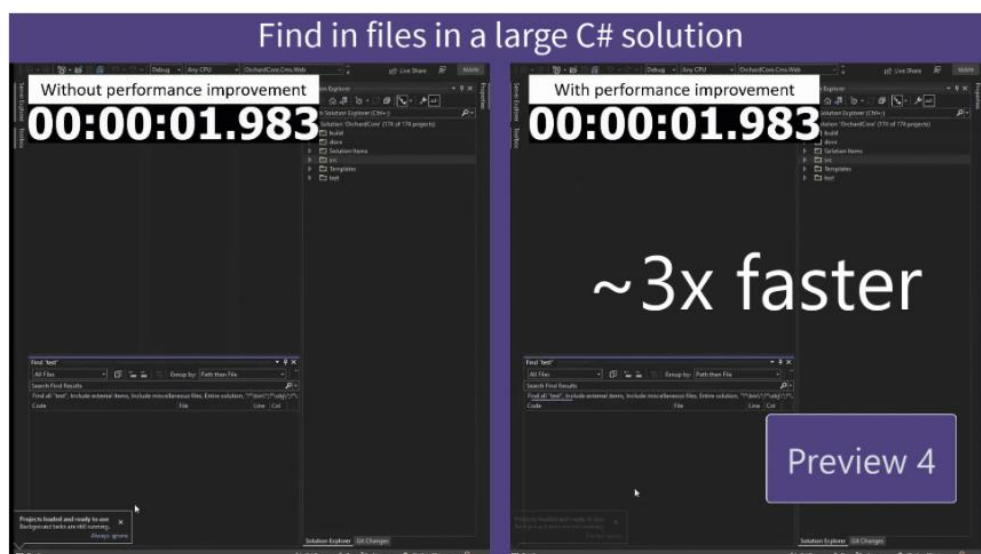
Visual Studio 2022 nhanh hơn, dễ tiếp cận hơn, nhẹ hơn và được thiết kế cho cả người học và những người xây dựng các giải pháp quy mô công nghiệp.

6.2.2 Visual Studio 2022 là 64-bit

Visual Studio 2022 trên Windows hiện là một ứng dụng 64-bit. Điều này có nghĩa là chúng ta có thể mở, chỉnh sửa, chạy và gỡ lỗi ngay cả những giải pháp lớn nhất và phức tạp nhất mà không lo hết bộ nhớ.

6.2.3 Tìm trong tệp nhanh hơn

Việc tìm kiếm trong file đã được tăng tốc độ lên gấp 3 lần.



² <https://devblogs.microsoft.com/visualstudio/visual-studio-2022-now-available/>

6.2.3 Xây dựng các ứng dụng hiện đại

Visual Studio 2022 giúp chúng ta dễ dàng và nhanh chóng xây dựng các ứng dụng dựa trên đám mây, hiện đại với Azure. Ngoài ra, phiên bản mới cũng hỗ trợ đầy đủ cho .NET 6 và khuôn khổ hợp nhất của nó dành cho web, ứng dụng khách và ứng dụng di động cho cả nhà phát triển Windows và Mac. Và, Visual Studio 2022 bao gồm hỗ trợ mạnh mẽ cho khối lượng công việc C++ với các tính năng năng suất mới, công cụ C++ 20 và IntelliSense.

6.2.4 Các công cụ dành cho nhà phát triển tốt hơn cho C++ và .NET và Hot Reload

Visual Studio 2022 bao gồm các công cụ phát triển ứng dụng đa nền tảng tốt hơn và phiên bản mới nhất của công cụ xây dựng C++, bao gồm hỗ trợ C++ 20.

Ngoài ra, VS22 cập nhật Tải lại nóng để bạn có thể chỉnh sửa các dự án C++ hoặc .NET trong khi ứng dụng đang chạy.

6.2.5 Hỗ trợ nhiều repo với Git trong IDE

Nếu đã làm việc với các dự án được lưu trữ trên các kho lưu trữ Git khác nhau, ta có thể đã sử dụng các công cụ bên ngoài hoặc nhiều phiên bản của Visual Studio để kết nối với chúng. Trong Visual Studio 2022, ta có thể làm việc với một giải pháp duy nhất có các dự án trong nhiều kho lưu trữ và đóng góp cho tất cả chúng từ một phiên bản Visual Studio.

6.2.6 Cải tiến IntelliCode

Hoàn thành toàn bộ dòng: Trong Visual Studio 2022, tính năng IntelliCode hiện có thể tự động hoàn thành mã tối đa một dòng cùng một lúc.

Đề xuất Tác vụ nhanh: Tính năng mới trong Visual Studio 2022, IntelliCode hiện có thể phát hiện khi bạn đang thực hiện một tác vụ thông thường và đề xuất Hành động nhanh phù hợp, hoàn thành ngay khi bạn đang nhập.

6.2.7 Thiết kế cho mọi người

VS đang làm mới giao diện người dùng để giúp bạn nắm bắt được quy trình của mình tốt hơn. Một số thay đổi bao gồm các nét thẩm mỹ hiện đại hóa giao diện người dùng.

6.4. Kết luận và kiến nghị

- VS2022 là một công cụ mạnh và hữu ích thực sự cho việc lập trình.
- Tôi đề nghị sinh viên nghiên cứu về VS2022 để học tập tốt trong các môn lập trình Net1, Net2.

Danh sách từ viết tắt

Chữ cái viết tắt/ Ký hiệu	Tiếng Anh	Tiếng Việt
VS	Visual Studio	Lập trình trực quan
VB	Visual Basic	Ngôn ngữ lập trình VB

Tài liệu tham khảo

- Choudhury A (2019) Top 10 Development Environments Used In 2019. In: Anal. India Mag. <https://analyticsindiamag.com/top-10-development-environments-used-in-2019/>. Accessed 31 Dec 2019
- heiligerdankgesang Extending Visual Studio for Mac - Visual Studio 2019 for Mac. <https://docs.microsoft.com/en-us/visualstudio/mac/extending-visual-studio-mac>. Accessed 31 Dec 2019
- top10tphcm (2021) Top 10 ngôn ngữ lập trình phổ biến nhất hiện nay. In: Top10tphcm. <https://top10tphcm.com/top-ngon-ngu-lap-trinh-pho-bien-nhat-hien-nay>. Accessed 27 Dec 2021
- The Most Popular Programming Languages - 1965/2021 - New Update. In: Stat. Data. <https://statisticsanddata.org/data/the-most-popular-programming-languages-1965-2021/>. Accessed 27 Dec 2021
- (2019a) Microsoft Visual Studio. Wikipedia Tiếng Việt
- (2019b) Microsoft Visual Studio. Wikipedia