# **Титулна страница**

**Име на училището:** Професионална Техническа Гимназия „Цар Симеон Велики“

**Населено място:** град Търговище

**Тема на дипломния проект:** Графично приложение за складиране на градивни елементи

**Трите имена на ученика:** Стелиан Росенов Христов

**Професия | специалност:** Компютърен техник | Компютърна техника и технологии

**Име и фамилия на ръководителя:** Нина Михайлова

# **Увод**

1. Описание на средата за разработка на Microsoft за програмиране на приложения с графичен интерфейс Visual Studio.
2. Описание на програмния език за дипломния проект ( C# ).
3. Описание за създаването на приложения с графичен интерфейс на C#.
4. Описание на елементите използвани в графичното приложение.
5. Посочване и обяснение на условни оператори, включително вложени условни оператори.
6. Описание как се работи с приложението.
7. Списък с градивни елементи и тяхното действие.

# **Основна част**

**1. Описание на средата за разработка на Microsoft за програмиране на приложения с графичен интерфейс Visual Studio.**

Средата за разработка, която се използва е създадена от Microsoft. Microsoft Visual Studio е IDE(интегрирана среда за разработка). Тя се използва за разработка на компютърни програми, също така уебсайтове, уеб приложения, уеб услуги, мобилни приложения и игри. Може да произвежда както собствен код, така и управляван код.

Visual Studio включва редактор на код, поддържащ IntelliSense(помага за завършването на код и става по-бързо писането), също и рефакторинг на код.

Интегрираният дебъгер работи както като дебъгер на ниво източник така и на ниво източник, така и като дебъгер на ниво машина. Други вградени инструменти са инструмент за профилиране на код, дизайнер за изграждане на графични приложения(GUI), уеб дизайнер, дизайнер на класове и дизайнер на схеми на бази данни.

Той също така може да приема всякакви добавки, които разширяват функционалноста на почти всяко ниво – включително и добавяне на инсрументи като редактори и виуални дизайнери за специфични за домейна езици или набори от инструменти за други аспекти от жизнения цикъл на разработка на софтуер.

Microsoft Visual Studio поддържа 36 различни езика за програмиране и позволява на редактора на код и инструмента за остраняване на грешки да поддържат в различна степен почти всеки език за програмиране, при условие че съществува специфична за езика услуга. Вградените езици включват C, C++/CLI, Visual Basic .NET, C#, F#, JavaScript, TypeScript, XML, XSLT, HTML и CSS. Поддръжка на други езици като Python, Ruby, Node.js и M, наред с други е достъпен чрез плъгини. Java се е поддържала в миналото.

Най-основното издание на Visual Studio, изданието на Общността, е достъпно безплатно. Безплатното издание се казва Visual Studio Community, което е предназначено за студенти, отворен код и индивидуални разработчици.

**Архиктектура**

Visual Studio не поддържа нито един език за програмиране, решение или инструмент по същество, вместо това позволява включването на функционалност, кодирана като VSPackage. Когато тя е инсталирана, функционалността е достъпна като услуга. Интегрираната среда за разработка (IDE) представя три услуги: SVsSolution, която предоставя възможност за изброяване на проекти и решения. SVsUIShell, който предоставя функционалност за прозорци и потребителски интерфейс (включително раздели, ленти с инструменти и прозорци с инструменти). И последно SVsShell , който се занимава с регистрация на VSPackages. В допълнение, IDE също така отваря за координирането и позволяването на комуникация между услугите. Всички редактори, дизайнери, типове проекти и други инструменти се изпълняват като VSPackages. Visual Studio използва COM за достъп до VSPackages. SDK на Visual Studio включва също така рамката за управлявани пакети (MPF), която е набор от управлявани обвивки около COM-интерфейсите, които позволяват на пакетите да бъдат написани на всеки съвместим с CLI език. Въпреки това, MPF не предоставя цялата функционалност , предоставяна от COM интерфейсите на Visual Studio. След това услугите могат да се използват за създаване на други пакети, които добавят функционалност към IDE.

Поддръжката на езици за програмиране се добавя чрез използване на специфичен VSP пакет, наречен езикова услуга. Езиковата услуга дефинира различни интерфейси, които реализацията на VSPackage може да приложи, за да добави поддръжка за различни функционалности. Функционалностите, които могат да бъдат добавен по този начин, включват оцветяване на синтаксиса, завършване на изрази,съвпадание на скоби, подсказки за информация за параметри, списъци с членове и маркери за грешки за фонова компилация. Ако интерфейсът е внедрен, функционалността ще бъде налична за езика. Езиковите услуги могат да бъдат внедрени или в собствен код или управляван код. За собствен код могат да се използват или собствените COM интерфейси, или Babel Framework (част от Visual Studio SDK). За управляван код MPF включва обвивки за писане на управлявани езикови услуги.

Visual Studio не включва вградена поддръжка за контрол на източника, но дефинира два алтернативни начина за интегриране на системите за контрол на източника с IDE. Source Control VSPackage може да предостави свой собствен персонализиран потребителски интерфейс. За разлика от тях, плугин за контрол на източника, използващ MSSCCI(интерфейс за управление на изходния код на Microsoft) представя набор от функции, които се използват за първи път за интегриране на Visual SourceSafe с Visual Studio 6.0, но по-късно беше отворен чрез Visual Studio SDK. Visual Studio .NET 2002 използва MSSCCI 1.1, а Visual Studio .NET 2003 използва MSSCCI 1.2. Visual Studio 2005, 2008 и 2010 използват MSSCCI версия 1.3, която добавя поддръжка за преименуване и разпространение за изтриването, както и асинхронно отваряне.

Visual Studio поддържа изпълнение на множество екземпляри на средата (всяка със собствен набор от VSPackages). Екземплярите използват различни кошери на системния регистър (вижте дефиницията на MSDN за термина " кошер на регистъра " в смисъла, използван тук), за да съхраняват състоянието си на конфигурация и се различават по своя AppId (идентификатор на приложение). Инстанциите се стартират от специфичен за AppId .exe, който избира AppId, задава основния кошер и стартира IDE. VSPackages, регистрирани за един AppId, се интегрират с други VSPackages за този AppId. Различните продуктови издания на Visual Studio са създадени с помощта на различните AppIds. Visual Studio ExpressПродуктите на изданието са инсталирани със собствени AppId, но продуктите Standard, Professional и Team Suite споделят един и същ AppId. Следователно, можете да инсталирате Express изданията рамо до рамо с други издания, за разлика от другите издания, които актуализират същата инсталация. Професионалното издание включва супернабор от VSPackages в стандартното издание, а екипният пакет включва супернабор от VSPackages и в двете други издания. Системата AppId се използва от Visual Studio Shell във Visual Studio 2008.

**Характеристика**

***Редактор на кодове***

Visual Studio (както всяка друга IDE ) включва редактор на код , който поддържа подчертаване на синтаксиса и завършване на код с помощта на IntelliSense за променливи , функции , методи , цикли и LINQ заявки. IntelliSense се поддържа за включените езици, както и за XML , каскадни стилови таблици и JavaScript при разработване на уеб сайтове и уеб приложения . Предложенията за автоматично довършване се появяват в aбезмоделно списъчно поле над прозореца на редактора на код, в близост до курсора за редактиране. Във Visual Studio 2008 нататък може да се направи временно полупрозрачен, за да се види препятстваният от него код. Редакторът на код се използва за всички поддържани езици.

Visual Studio Code Editor също така поддържа настройка на отметки в кода за бърза навигация. Други помощни средства за навигация включват свиващи кодови блокове и инкрементално търсене , в допълнение към нормалното търсене на текст и търсенето на регулярни изрази . Редакторът на код също включва клипборд с множество елементи и списък със задачи. Редакторът на код поддържа фрагменти от код, които са запазени шаблони за повтарящ се код и могат да бъдат вмъкнати в кода и персонализирани за проекта, върху който се работи. Вграден е и инструмент за управление на кодови фрагменти. Тези инструменти се показват като плаващи прозорци, които могат да бъдат настроени да се скриват автоматично, когато не се използват или закачени отстрани на екрана. Редакторът на код на Visual Studio също поддържа рефакторинг на код , включително пренареждане на параметри, преименуване на променливи и методи, извличане на интерфейс и капсулиране на членове на класа в свойствата, наред с други.Visual Studio разполага с фонова компилация (наричана още инкрементална компилация). Докато кодът се пише, Visual Studio го компилира във фонов режим, за да предостави обратна връзка за грешки в синтаксиса и компилацията, които са маркирани с червено вълнообразно подчертаване. Предупрежденията са маркирани със зелено подчертаване. Фоновата компилация не генерира изпълним код, тъй като изисква различен компилатор от този, използван за генериране на изпълним код. Фоновата компилация първоначално беше въведена с Microsoft Visual Basic , но сега е разширена за всички включени езици.

***Програма за отстраняване на грешки***

Visual Studio включва средство за отстраняване на грешки , което работи както като средство за отстраняване на грешки на ниво източник, така и като средство за отстраняване на грешки на ниво машина. Той работи както с управляван код , така и с естествен код и може да се използва за отстраняване на грешки в приложения, написани на всеки език, поддържан от Visual Studio. В допълнение, той може също да се свързва към работещи процеси, да наблюдава и да отстранява грешки в тези процеси. Ако изходният код за изпълнявания процес е наличен, той показва кода, докато се изпълнява. Ако изходният код не е наличен, той може да покаже разглобяването . Дебъгерът на Visual Studio може също да създава дъмпове на паметта , както и да ги зарежда по-късно за отстраняване на грешки. Поддържат се и многонишкови програми. Програмата за отстраняване на грешки може да бъде конфигурирана да се стартира, когато приложение, работещо извън средата на Visual Studio, се срине.Visual Studio Debugger позволява задаване на точки на прекъсване (които позволяват изпълнението да бъде спряно временно на определена позиция) и наблюдава (които следят стойностите на променливите с напредване на изпълнението). Точките на прекъсване могат да бъдат условни, което означава, че се задействат, когато условието е изпълнено. Кодът може да се премине през , т.е. да се изпълнява един ред (от изходния код) наведнъж. Може или да влезе във функции за отстраняване на грешки в него, или да премине през него, т.е. изпълнението на тялото на функцията не е достъпно за ръчна проверка. Инструментът за отстраняване на грешки поддържа Edit and Continue, т.е. позволява кодът да бъде редактиран, докато се отстранява грешки. При отстраняване на грешки, ако показалецът на мишката се задържи над която и да е променлива, текущата й стойност се показва в подсказка („подсказки за данни“), където също може да бъде променена, ако желаете. По време на кодирането дебъгерът на Visual Studio позволява ръчно извикване на определени функции от Immediateпрозореца с инструменти. Параметрите на метода се предоставят в прозореца Immediate.

***Дизайнер***

Visual Studio включва множество визуални дизайнери, които да помогнат при разработването на приложения. Тези инструменти включват:

***Windows Forms Designer***

Дизайнерът на Windows Forms се използва за изграждане на GUI приложения с помощта на Windows Forms . Оформлението може да се контролира чрез поставяне на контролите в други контейнери или заключване отстрани на формуляра. Контролите, които показват данни (като текстово поле, списъчно поле и изглед на мрежа), могат да бъдат обвързани с източници на данни като бази данни или заявки . Обвързаните с данни контроли могат да бъдат създадени чрез плъзгане на елементи от прозореца Източници на данни върху повърхност за проектиране. Потребителският интерфейс е свързан с код, използвайки програмен модел, управляван от събития. Дизайнерът генерира C# или VB.NET код за приложението.

***WPF дизайнер***

WPF дизайнерът с кодово име Cider беше представен с Visual Studio 2008. Подобно на дизайнера на Windows Forms, той поддържа метафората за плъзгане и пускане. Използва се за създаване на потребителски интерфейси, насочени към Windows Presentation Foundation . Той поддържа всички функции на WPF, включително обвързване на данни и автоматично управление на оформлението. Той генерира XAML код за потребителския интерфейс. Генерираният XAML файл е съвместим с Microsoft Expression Design , ориентираният към дизайнера продукт. XAML кодът е свързан с код, като се използва код зад модел.

***Уеб дизайнер/разработка***

Visual Studio също така включва редактор и дизайнер на уеб сайтове, които позволяват създаване на уеб страници чрез плъзгане и пускане на джаджи. Използва се за разработване на ASP.NET приложения и поддържа HTML , CSS и JavaScript . Той използва модел зад код за свързване с ASP.NET код. От Visual Studio 2008 нататък, машината за оформление, използвана от уеб дизайнера, се споделя с прекратения Expression Web . Има също така поддръжка на ASP.NET MVC за технологията MVC като отделно изтегляне и проект ASP.NET Dynamic Data , наличен от Microsoft.

***Дизайнер на класа***

Дизайнерът на класове се използва за създаване и редактиране на класовете (включително неговите членове и техния достъп) с помощта на UML моделиране. Class Designer може да генерира C# и VB.NET кодови очертания за класовете и методите. Може също така да генерира диаграми на класове от ръчно написани класове.

***Дизайнер на данни***

Дизайнерът на данни може да се използва за графично редактиране на схеми на база данни , включително въведени таблици, първични и външни ключове и ограничения. Може да се използва и за проектиране на заявки от графичния изглед.

***Картографски дизайнер***

От Visual Studio 2008 нататък дизайнерът на картографиране се използва от LINQ to SQL за проектиране на съпоставянето между схемите на базата данни и класовете , които капсулират данните. Новото решение от подхода на ORM, ADO.NET Entity Framework , замества и подобрява старата технология.

**Други инструменти:**

***Отворете браузъра на раздели***

Браузърът на отворени раздели се използва за изброяване на всички отворени раздели и за превключване между тях. Извиква се с помощта на ***CTRL + TAB.***

***Редактор на свойства***

Инструментът Properties Editor се използва за редактиране на свойства в GUI панел във Visual Studio. Той изброява всички налични свойства (както само за четене, така и тези, които могат да бъдат зададени) за всички обекти, включително класове , формуляри, уеб страници и други елементи.

***Браузър на обекти***

Браузърът на обекти е браузър на пространство от имена и библиотека на класове за Microsoft .NET . Може да се използва за разглеждане на пространствата от имена (които са подредени йерархично) в управлявани сборки . Йерархията може или не може да отразява организацията във файловата система.

***Изследовател на решения***

На езика на Visual Studio решението е набор от кодови файлове и други ресурси, които се използват за изграждане на приложение. Файловете в решението са подредени йерархично, което може или не може да отразява организацията във файловата система. Solution Explorer се използва за управление и преглеждане на файловете в решение.

***Team Explorer***

Team Explorer се използва за интегриране на възможностите на Azure DevOps (или Azure DevOps Services или Azure DevOps Server ) в IDE. В допълнение към интеграцията за контрол на версиите, той предоставя възможност за преглед и управление на отделни работни елементи (включително потребителски истории, грешки, задачи и други документи). Той е включен като част от инсталация на Visual Studio и се предлага и като самостоятелно изтегляне.

***Data Explorer***

Data Explorer се използва за управление на бази данни на екземпляр и на Microsoft SQL Server .Позволява създаване и промяна на таблици на база данни (или чрез издаване на T-SQL команди или чрез използване на дизайнера на данни). Може също да се използва за създаване на заявки и съхранени процедури ,като последните в T-SQL или в управляван код чрез SQL CLR . Налична е и поддръжка за отстраняване на грешки и IntelliSense .

***Server Explorer***

Инструментът Server Explorer се използва за управление на връзки към база данни на достъпен компютър. Използва се също за разглеждане на работещи услуги на Windows ,броячи на производителност, журнална събития на Windows и опашки за съобщения и ги използва като източник на данни.

***Dotfuscator Community Edition***

Visual Studio включва безплатна „лека“ версия на Dotfuscator.

***Рамка за генериране на текст***

Visual Studio включва рамка за генериране на пълен текст, наречена T4 ,която позволявана Visual Studio да генерира текстови файлове от шаблони или в IDE, или чрез код.

***Инструмент за администриране на уебсайт ASP.NET***

Инструментът за администриране на уебсайт ASP.NET позволяваконфигуриранетонауебсайтовенаASP.NET .

***Инструменти на Visual Studio за Office***

Visual Studio Tools за Office е комплект за разработване на софтуер (SDK) и добавка за Visual Studio, която включва инструменти за разработване на пакета Microsoft Office .Преди това (за Visual Studio .NET 2003 и Visual Studio 2005) беше отделен SKU, който поддържаш е само Visual C# и Visual Basic езици или беше включен в Team Suite. С Visual Studio 2008 той вече не е отделен SKU, а е включен в Professional и по-високи издания. При внедряване на VSTO решения е необходимо отделно време за изпълнение.

**Разширяемост**

Visual Studio позволява на разработчиците да пишат разширения за Visual Studio, за да разширят неговите възможности. Тези разширения се "включват" в Visual Studio и разширяват неговата функционалност. Разширенията се предлагат под формата на макроси , добавки и пакети . Макросите представляват повтарящи се задачи и действия, които разработчиците могат да записват програмно за запазване, възпроизвеждане и разпространение. Макросите обаче не могат да прилагат нови команди или да създават прозорци с инструменти. Те са написани с помощта на Visual Basic и не са компилирани. Добавките осигуряват достъп до обектния модел на Visual Studio и могат да взаимодействат с инструментите на IDE. Добавките могат да се използват за внедряване на нова функционалност и могат да добавят нови прозорци с инструменти. Добавките се включват в IDE чрезCOM и може да бъде създаден на всякакви COM-съвместими езици. Пакетите се създават с помощта на Visual Studio SDK и осигуряват най-високо ниво на разширяемост. Те могат да създават дизайнери и други инструменти, както и да интегрират други езици за програмиране. SDK на Visual Studio предоставя не управлявани API , както и управляван API за изпълнение на тези задачи. Въпреки това, управляваният API не е толкова изчерпателен, колкото не управлявания. Разширенията се поддържат в стандартните (и по-високи) версии на Visual Studio 2005. Експресните издания не поддържат разширения за хостинг. Visual Studio 2008 представи Visual Studio Shell , който позволява разработването на персонализирана версия на IDE. Visual Studio Shell дефинира набор от VSPackages, които осигуряват функционалността, необходима във всяка IDE. Освен това могат да се добавят и други пакети за персонализиране на инсталацията. Изолираният режим на обвивката създава нов AppId, където са инсталирани пакетите. Те трябва да се стартират с различен изпълним файл. Той е насочен към разработване на персонализирани среди за разработка, за конкретен език или за конкретен сценарий. Интегрираният режим инсталира пакетите в AppId на изданията Professional/Standard/Eeam System, така че инструментите да се интегрират в тези издания. Visual Studio Shell се предлага като безплатно изтегляне. След пускането на Visual Studio 2008, Microsoft създаде Visual Studio Gallery. Той служи като централно място за публикуване на информация за разширения във Visual Studio. Разработчиците от общността, както и търговските разработчици, могат да качват информация за своите разширения във Visual Studio .NET 2002 чрез Visual Studio 2010. Потребителите на сайта могат да оценяват и преглеждат разширенията, за да помогнат при оценката на качеството на публикуваните разширения. Разширението се съхранява във файл VSIX. Вътрешно VSIX файлът е ZIP файл, който съдържа някои XML файлове и вероятно един или повече DLL. Едно от основните предимства на тези разширения е, че не изискват инсталиране на администраторски права. Планирани са и RSS емисии за уведомяване на потребителите за актуализации на сайта и функции за маркиране.

**Поддържани продукти**

**Microsoft Visual C++**

Microsoft Visual C++ е частична реализация на Microsoft на C и пълна реализация на C++ компилатора и свързаните езици-услуги и специфични инструменти за интеграция с Visual Studio IDE. Може да компилира в режим C или C++. За C++, от версия 15.7 той отговаря на C++17 . C реализацията на Visual Studio 2015 все още не поддържа пълния стандарт; по-специално, комплексното число заглавка complex.h , въведено в C99, не се поддържа. Visual C++ поддържа спецификацията C++/CLI за писане на управляван код , както и код в смесен режим (комбинация от собствен и управляван код ). Microsoft позиционира Visual C++ за разработка в собствен код или в код, който съдържа както собствени, така и управлявани компоненти. Visual C++ поддържа COM , както и библиотеката MFC . За разработката на MFC той предоставя набор от съветници за създаване и персонализиране на стандартен код на MFC и създаване на GUI приложения с помощта на MFC. Visual C++ може също да използва дизайнера на форми на Visual Studio за графично проектиране на потребителския интерфейс. Visual C++ може да се използва и с Windows API . Той също така поддържа използването на вътрешни функции ,които са функции, разпознати от самия компилатор и не имплементирани като библиотека. Вътрешните функции се използват за излагане на набора от инструкции SSE на съвременните процесори. Visual C++ също така включва спецификацията OpenMP (версия 2.0).

**Microsoft Visual C#**

Microsoft Visual C#, реализацията на Microsoft на езика C# , е насочена към .NET Framework, заедно с езиковите услуги, които позволяват на Visual Studio IDE да поддържа C# проекти. Докато езиковите услуги са част от Visual Studio, компилаторът се предлага отделно като част от .NET Framework. Компилаторите на Visual C# 2008, 2010 и 2012 поддържат съответно версии 3.0, 4.0 и 5.0 от спецификациите на езика C#. Visual C# поддържа дизайнера на клас Visual Studio, дизайнера на формуляри и дизайнера на данни, наред с други.

**Microsoft Visual Basic**

Microsoft Visual Basic е реализация от Microsoft на езика VB.NET и свързаните с него инструменти и езикови услуги. Той беше представен с Visual Studio .NET (2002). Microsoft позиционира Visual Basic за бърза разработка на приложения . Visual Basic може да се използва за създаване както на конзолни приложения, така и на GUI приложения. Подобно на Visual C#, Visual Basic също поддържа дизайнера на Visual Studio Class, дизайнера на формуляри и дизайнера на данни, наред с други. Подобно на C#, VB.NET компилаторът също е наличен като част от .NET Framework, но езиковите услуги, които позволяват на VB.NET проекти да се разработват с Visual Studio, са достъпни като част от последното.

**Microsoft Visual Web Developer**

Microsoft Visual Web Developer се използва за създаване на уеб сайтове, уеб приложения и уеб услуги с помощта на ASP.NET. Могат да се използват езици C# или VB.NET . Visual Web Developer може да използва Visual Studio Web Designer за графично проектиране на оформления на уеб страници.

**Azure DevOps**

Azure DevOps е предназначен за проекти за съвместна разработка на софтуер и осигурява контрол на версиите , планиране и проследяване на работата, събиране на данни и отчитане . Той също така включва Team Explorer , който е интегриран във Visual Studio. На 10 септември 2018 г. Microsoft обяви ребрандиране на Visual Studio Team Services (VSTS) в Azure DevOps Services и Team Foundation Server (TFS) в Azure DevOps Server.

**Предишни продукти**

**Visual FoxPro**

Visual FoxPro е ориентиран към данни обектно-ориентиран и процедурен език за програмиране, произведен от Microsoft . Той произлиза от FoxPro (първоначално известен като FoxBASE ), който е разработен от Fox Software в началото на 1984 г. Visual FoxPro е тясно интегриран със собствен двигател за релационна база данни, който разширява възможностите на FoxPro xBase, за да поддържа SQL заявки и манипулиране на данни. Visual FoxPro е пълнофункционален динамичен език за програмиране , който не изисква използването на допълнителна среда за програмиране с общо предназначение. През 2007 г. Visual FoxPro беше прекратен след версия 9 Service Pack 2. Поддържаше се до 2015 г.

**Visual SourceSafe**

Microsoft Visual SourceSafe е софтуерен пакет за контрол на източника , ориентиран към малки проекти за разработка на софтуер. Базата данни SourceSafe е многопотребителска, многопроцесова база данни с файлова система, използваща примитиви на базата данни на файловата система на Windows, за да осигури поддръжка за заключване и споделяне. Всички версии са многопотребителски, като използват SMB (файлов сървър) мрежа. Въпреки това, с Visual SourceSafe 2005 бяха добавени други режими клиент-сървър , Lan Booster и VSS Internet (които използваха HTTP / HTTPS ). Visual SourceSafe 6.0 беше наличен като самостоятелен продукт [56]и беше включен в Visual Studio 6.0 и други продукти като Office Developer Edition. Visual SourceSafe 2005 беше наличен като самостоятелен продукт и беше включен в Team Suite 2005. Azure DevOps замени VSS като препоръчана от Microsoft платформа за контрол на източника.

**Microsoft Visual J++ / Microsoft Visual J#**

Microsoft Visual J++ беше реализацията на Microsoft на езика Java (със специфични за Microsoft разширения) и свързаните езикови услуги. Тя беше прекратена в резултат на съдебни спорове от Sun Microsystems и технологията беше рециклирана във Visual J#, Java компилатор на Microsoft за .NET Framework. J# беше наличен с Visual Studio 2005 (поддържа се до 2015 г.), но беше преустановен във Visual Studio 2008.

**Visual InterDev**

Visual InterDev беше използван за създаване на уеб приложения с помощта на технологиите на Microsoft Active Server Pages (ASP). Той поддържа завършване на код и включва инструменти за управление на сървър на база данни . Той е заменен с Microsoft Visual Web Developer.

***Издания***

**Общност**

Изданието на Общността беше обявено на 12 ноември 2014 г. като нова безплатна версия, с подобна функционалност на Visual Studio Professional. Преди тази дата единствените безплатни издания на Visual Studio бяха вариантите Express с ограничени функции. За разлика от експресните варианти, Visual Studio Community поддържа множество езици и осигурява поддръжка за разширения. Индивидуалните разработчици нямат ограничения за използването на изданието на Общността. Следните употреби също позволяват неограничено използване: принос към проекти с отворен код, академични изследвания, в учебна среда в класната стая и за разработване и тестване на драйвери на устройства за операционната система Windows. Цялото друго използване от организацията зависи от това дали сте класифицирани като Предприятие (повече от 250 служители или повече от 1 милион USDв годишни приходи, според Microsoft). Непредприятия могат да използват до 5 копия без ограничение, потребител номер 6 и по-висок изисква търговски лиценз; Корпоративните организации изискват търговски лиценз за използване извън отбелязаните изключения. Visual Studio Community е ориентиран към индивидуални разработчици и малки екипи.

**Професионален**

От Visual Studio 2010, професионалното издание е комерсиалното издание на Visual Studio от начално ниво. (Преди това беше налично стандартно издание с по-ограничени функции.) Той предоставя IDE за всички поддържани езици за разработка. Поддръжката на MSDN е налична като MSDN Essentials или пълната библиотека на MSDN в зависимост от лиценза. Той поддържа XML и XSLT редактиране и може да създава пакети за внедряване, които използват само ClickOnce и MSI . Той включва инструменти като Server Explorer и интеграция с Microsoft SQL Serverсъщо. Поддръжката за разработка на Windows Mobile беше включена в Visual Studio 2005 Standard, но с Visual Studio 2008 тя е достъпна само в Professional и по-високи издания. Поддръжката за разработка на Windows Phone 7 беше добавена към всички издания на Visual Studio 2010. Разработката за Windows Mobile вече не се поддържа във Visual Studio 2010. Тя е заменена от Windows Phone 7 .

**Enterprise**

В допълнение към функциите, предоставени от професионалното издание, изданието Enterprise предоставя нов набор от инструменти за разработка на софтуер, разработка на бази данни, сътрудничество, показатели, архитектура, тестване и отчитане.

**История**

Първата версия на Visual Studio беше Visual Studio 97. [63] Преди това Visual Basic, Visual C++, Visual FoxPro и Visual SourceSafe се продаваха като отделни продукти.

**97**

Microsoft за първи път пусна Visual Studio (с кодово име Бостън ,за едноименния град , като по този начин започва кодовите имена на VS, свързани с места)през 1997 г., обединявайки много от своите инструменти за програмиране заедно за първи път. Visual Studio 97 се предлага в две издания: Visual Studio Professional и Visual Studio Enterprise, професионалното издание има три компактдиска, а корпоративното четири компактдиска. Той включва Visual J++ 1.1 за програмиране на Java и въвежда Visual InterDev за създаване на динамично генерирани уеб сайтове с помощта на Active Server Pages. Имаше един-единствен придружаващ компактдиск, който съдържаше библиотеката на Microsoft Developer Network .Visual Studio 97 беше опит за използване на една и съща среда за разработка за множество езици. Visual J++, InterDev и библиотеката MSDN са използвали една и съща „среда“, наречена Developer Studio. Visual Studio също беше продаден като пакет с отделните IDE, използвани за Visual C++, Visual Basic и Visual FoxPro.

**6.0 (1998)**

Следващата версия, версия 6.0 (с кодово име Aspen , по името на ски курорта в Колорадо), , беше пусната през юни 1998 г. и е последната версия, която работи на платформата Windows 9x . Всяка версия на всеки език отчасти също се установи до v6.0, включително Visual J++, който беше преди v1.1, и Visual InterDev при първата версия. Изданието v6 на Microsoft беше основната среда за следващите четири издания, за да предостави на програмистите интегрирана подобна платформа. Това накара Microsoft да прехвърли разработката на независима от платформата .NET Framework .Visual Studio 6.0 беше последната версия, която включва Visual J++, който Microsoft премахна като част от споразумение със Sun Microsystems , което изисква Microsoft Internet Explorer да не предоставя поддръжка за виртуалната машина Java .Visual Studio 6.0 се предлага в две издания: Professional и Enterprise. Изданието Enterprise съдържаше допълнителни функции, които не се намират в Professional издание, включително:Изследовател на производителността на приложениятаМениджър за автоматизация Microsoft Visual ModelerRemAuto Connection Manager. Visual Studio AnalyzerVisual Studio също беше продаден като пакет с отделните IDE, използвани за Visual C++, Visual Basic и Visual FoxPro.

**.NET 2002**

Microsoft пусна Visual Studio .NET (VS.NET), с кодово име Rainier (за Вашингтон Mount Rainier ), през февруари 2002 г. ( бета версията беше пусната чрез Microsoft Developer Network през 2001 г.). Най-голямата промяна беше въвеждането на управлявана среда за разработка на код, използваща .NET Framework. Програмите, разработени с помощта на .NET, не се компилират на машинен език (както C++ е, например), а вместо това във формат, наречен Microsoft Intermediate Language (MSIL) или Common Intermediate Language (CIL). Когато CIL приложение се изпълнява, това е такакомпилиран, докато се изпълнява на съответния машинен език за платформата, на която се изпълнява, като по този начин прави кода преносим между няколко платформи. Програмите, компилирани в CIL, могат да се изпълняват само на платформи, които имат реализация на обща езикова инфраструктура . Възможно е да се изпълняват CIL програми в Linux или Mac OS X , като се използват не-Microsoft .NET реализации като Mono и DotGNU .Това беше първата версия на Visual Studio, която изисква NT - базирана Windows платформа. Инсталаторът налага това изискване.Visual Studio .NET 2002 се доставя в четири издания: Academic, Professional, Enterprise Developer и Enterprise Architect. Microsoft представи C# (C-sharp), нов език за програмиране, който е насочен към .NET. Той също така представи наследника на Visual J++, наречен Visual J#. Visual J# програмите използват езиковия синтаксис на Java. Въпреки това, за разлика от програмите на Visual J++, програмите на Visual J# могат да са насочени само към .NET Framework, а не към виртуалната машина на Java, към която са насочени всички други инструменти на Java.Visual Basic се промени драстично, за да пасне на новата рамка и новата версия беше наречена Visual Basic .NET. Microsoft също добави разширения към C++, наречени Managed Extensions за C++ , така че .NET програмите могат да се създават на C++.Visual Studio .NET може да създава приложения, насочени към Windows (използвайки Windows Forms част от .NET Framework), уеб (използвайки ASP.NET и уеб услуги ) и, с добавка, преносими устройства (използвайки .NET Compact Framework ).Средата на Visual Studio .NET беше пренаписана, за да използва частично .NET. Всички езици са версии на Visual Studio, той има по-изчистен интерфейс и по-голяма сплотеност.Освен това е по-персонализиран с прозорци с инструменти, които автоматично се скриват, когато не се използват. Докато Visual FoxPro 7 стартира като част от Visual Studio .NET 2002, а ранните бета версии на VS позволяваха отстраняване на грешки в базирани на VFP DLL, той беше премахнат преди пускането му, за да следва своя собствен път за разработка. Вътрешният номер на версията на Visual Studio .NET 2002 е версия 7.0. Microsoft пусна Service Pack 1 за Visual Studio .NET 2002 през март 2005 г.

**.NET 2003**

През април 2003 г. Microsoft въведе незначителна надстройка на Visual Studio .NET, наречена Visual Studio .NET 2003, с кодово име Everett (за град със същото име ).Той включва надстройка до .NET Framework, версия 1.1, и е първата версия, която поддържа разработването на програми за мобилни устройства, използвайки ASP.NET или .NET Compact Framework. Съответствието със стандартите на компилатора на Visual C++ е подобрено, особено в областта на частичната специализация на шаблони . Visual C++ Toolkit 2003 е версия на същия C++ компилатор, доставен с Visual Studio .NET 2003 без IDE, който Microsoft направи свободно достъпен. От 2010гтой вече не е наличен и Express Editions го изместиха. Visual Studio .NET 2003 също поддържа Managed C++, който е предшественик на C++/CLI. Вътрешният номер на версията на Visual Studio .NET 2003 е версия 7.1, докато версията на файловия формат е 8.0. Visual Studio .NET 2003 се доставя в пет издания: Academic, Standard, Professional, Enterprise Developer и Enterprise Architect. Изданието Visual Studio .NET 2003 Enterprise Architect включва внедряване на технологиите за моделиране на Microsoft Visio 2002, включително инструменти за създаване на базирани на Unified Modeling Language визуални представяния на архитектурата на приложението и моделиране на обектно-ролева (ORM) и логическо решение за моделиране на база данни . Бяха въведени и „шаблони за предприятия“, за да помогнат на по-големите екипи за разработка да стандартизират стиловете на кодиране и да наложат политики около използването на компонентите и настройките на свойствата.Сервизен пакет 1 беше издаден на 13 септември 2006 г.

**2005**

Visual Studio 2005, с кодово име Whidbey (препратка към остров Уидби в района на Пюджет Саунд ), беше пуснат онлайн през октомври 2005 г. и в магазините за търговия на дребно няколко седмици по-късно. Microsoft премахна прозвището ".NET" от Visual Studio 2005 (както и всеки друг продукт с .NET в името си), но той все още е насочен основно към .NET Framework, който беше надстроен до версия 2.0. Това е последната версия, налична за Windows 2000 , а също и последната версия, която може да е насочена към Windows 98 , Windows Me и Windows NT 4.0 за C++ приложения. Вътрешният номер на версията на Visual Studio 2005 е 8.0, докато версията на файловия формат е 9.0. Microsoft пусна Service Pack 1 за Visual Studio 2005 на 14 декември 2006 г. Допълнителна актуализация за Service Pack 1, която предлага съвместимост с Windows Vista, беше предоставена на 3 юни 2007 г. Visual Studio 2005 беше надстроен, за да поддържа всички нови функции, въведени в .NET Framework 2.0, включително общи и ASP.NET 2.0. Функцията IntelliSense във Visual Studio беше надстроена за генерични и бяха добавени нови типове проекти, за да поддържат уеб услугите на ASP.NET. Visual Studio 2005 допълнително въвежда поддръжка за нова базирана на задачи платформа за изграждане, наречена Microsoft Build Engine ( MSBuild ), която използва нов XML-базиран файлов формат на проекта. Visual Studio 2005 включва също локален уеб сървър, отделен от IIS , който може да хоства ASP.NET приложения по време на разработка и тестване. Той също така поддържа всички бази данни на SQL Server 2005. Дизайнерите на бази данни бяха надстроени, за да поддържат ADO.NET 2.0, който е включен в .NET Framework 2.0. C++ също получи подобен ъпгрейд с добавянето на C++/CLI , който е планиран да замени използването на Managed C++ . Други нови функции на Visual Studio 2005 включват "Deployment Designer", който позволява валидирането на дизайна на приложения преди внедряването, подобрена среда за уеб публикуване, когато се комбинира с ASP.NET 2.0 и тестване на натоварване, за да се види производителността на приложението при различни видове потребителски натоварвания. Започвайки с изданието от 2005 г., Visual Studio добави и обширна 64-битова поддръжка. Докато самата среда за разработка на хост е достъпна само като 32-битово приложение, Visual C++ 2005 поддържа компилиране за x86-64 (AMD64 и Intel 64), както и IA-64( Itanium ). SDK на платформата включва 64-битови компилатори и 64-битови версии на библиотеките.Microsoft също обяви Visual Studio Tools за приложения като наследник на Visual Basic за приложения (VBA) и VSA (Visual Studio for Applications). VSTA 1.0 беше пуснат за производство заедно с Office 2007 . Той е включен в Office 2007 и също е част от Visual Studio 2005 SDK. VSTA се състои от персонализирана IDE, базирана на Visual Studio 2005 IDE, и среда за изпълнение, която може да бъде вградена в приложения, за да изложи своите функции чрез обектния модел .NET. Приложенията на Office 2007 продължават да се интегрират с VBA, с изключение на InfoPath 2007, който се интегрира с VSTA. Версия 2.0 на VSTA (базирана на Visual Studio 2008) беше пусната през април 2008 г. [142]Тя е значително различна от първата версия, включително функции като динамично програмиране и поддръжка за WPF , WCF , WF , LINQ и .NET 3.5 Framework.

**2008**

Visual Studio 2008, и Visual Studio Team System 2008 с кодово име Orcas (препратка към Orcas Island , също остров в региона Puget Sound , като Whidbey за предишната версия от 2005 г.), бяха пуснати за абонати на MSDN на 19 ноември 2007 г. заедно с .NET Framework 3.5. Изходният код за Visual Studio 2008 IDE е достъпен под лиценз за споделен източник за някои от партньорите на Microsoft и ISV . [96] Microsoft пусна Service Pack 1 за Visual Studio 2008 на 11 август 2008 г. [146]Вътрешният номер на версията на Visual Studio 2008 е версия 9.0, докато версията на файловия формат е 10.0. Visual Studio 2008 е последната версия, която поддържа насочване към Windows 2000 за C++ приложения. Visual Studio 2008 е фокусиран върху разработването на Windows Vista , 2007 Office система и уеб приложения. За визуален дизайн са включени нов визуален дизайнер на Windows Presentation Foundation и нов HTML / CSS редактор, повлиян от Microsoft Expression Web . J# не е включен. Visual Studio 2008 изисква .NET 3.5 Framework и по подразбиране конфигурира компилираните асембли да се изпълняват на .NET Framework 3.5, но също така поддържа множество насочване, което позволява на разработчиците да избират коя версия на .NET Framework (от 2.0, 3.0, 3.5, Silverlight CoreCLR или .NET Compact Framework) сборката работи. Visual Studio 2008 включва и нов анализ на кода инструменти, включително новия инструмент за метрики на кода (само в Team Edition и Team Suite Edition). [148] За Visual C++ Visual Studio добавя нова версия на Microsoft Foundation Classes (MFC 9.0), която добавя поддръжка за визуалните стилове и контролите на потребителския интерфейс, въведени с Windows Vista . За оперативна съвместимост на собствения и управлявания код, Visual C++ въвежда STL/CLR, който е порт на контейнерите и алгоритмите на C++ Standard Template Library (STL) към управлявания код . STL/CLR дефинира подобни на STL контейнери, итератори и алгоритми, които работят върху C++/CLI управлявани обекти. Функциите на Visual Studio 2008 включват XAML - базиран дизайнер (с кодово име Cider ), дизайнер на работни потоци , LINQ to SQL дизайнер (за дефиниране на съпоставяния на типове и капсулиране на обекти за данни на SQL Server), XSLT дебъгер, поддръжка на JavaScript Intellisense , поддръжка за отстраняване на грешки в JavaScript, поддръжка за UAC манифести , система за едновременно изграждане, наред с други. Той се доставя с подобрен набор от UI джаджи, както за Windows Forms , така и за WPF . Той също така включва многонишков механизъм за изграждане ( MSBuild) за компилиране на множество изходни файлове (и изграждане на изпълнимия файл) в проект в множество нишки едновременно. Той също така включва поддръжка за компилиране на ресурси за икони във формат PNG , въведен в Windows Vista. Актуализиран дизайнер на XML схема беше пуснат отделно известно време след пускането на Visual Studio 2008. Visual Studio Debugger включва функции, насочени към по-лесно отстраняване на грешки в многонишкови приложения. В режим на отстраняване на грешки, в прозореца Threads , който изброява всички нишки, задържането на курсора на мишката върху нишка показва проследяването на стека на тази нишка в подсказките. Нишките могат директно да бъдат именувани и маркирани за по-лесно идентифициране от самия прозорец. В допълнение, в кодовия прозорец, заедно с посочването на местоположението на текущо изпълняващата се инструкция в текущата нишка, се посочват и изпълняваните в момента инструкции в други нишки. Инструментът за отстраняване на грешки в Visual Studio поддържа интегрирано отстраняване на грешки на библиотеката с базови класове на .NET 3.5 Framework(BCL), който може динамично да изтегля изходния код на BCL и символите за отстраняване на грешки и да позволява влизане в източника на BCL по време на отстраняване на грешки. От 2010 г. е налично ограничено подмножество от източника на BCL, като за по-късно се планира повече поддръжка на библиотеката.

**2010**

На 12 април 2010 г. Microsoft пусна Visual Studio 2010 с кодово име Dev10 , и .NET Framework 4 . Visual Studio 2010 IDE беше преработен, което според Microsoft изчиства организацията на потребителския интерфейс и „намалява претрупването и сложността“. Новата IDE поддържа по-добре множество прозорци на документи и плаващи прозорци с инструменти, като същевременно предлага по-добра поддръжка на няколко монитора. Обвивката на IDE е пренаписана с помощта на Windows Presentation Foundation (WPF), докато вътрешните елементи са преработени с помощта на Managed Extensibility Framework (MEF), която предлага повече точки за разширяване от предишните версии на IDE, които позволяват добавки за промяна на поведението на IDE. Новият мулти-парадигмен ML -вариант F# е част от Visual Studio 2010. Visual Studio 2010 идва с .NET Framework 4 и поддържа разработване на приложения, насочени към Windows 7 . Той поддържа бази данни IBM DB2 и Oracle , в допълнение към Microsoft SQL Server . Той има интегрирана поддръжка за разработване на приложения на Microsoft Silverlight , включително интерактивен дизайнер. Visual Studio 2010 предлага няколко инструмента за опростяване на паралелното програмиране : в допълнение към паралелните разширения за .NET Framework и библиотеката с паралелни модели за естествен код, Visual Studio 2010 включва инструменти за отстраняване на грешки в паралелни приложения. Новите инструменти позволяват визуализирането на паралелни задачи и техните стекове по време на изпълнение . Инструментите за профилиране на паралелни приложения могат да се използват за визуализиране на времето за изчакване на нишките и миграции на нишки между процесорни ядра. Intel и Microsoft съвместно обещаха поддръжка за ново време за изпълнение на едновременно изпълнение във Visual Studio 2010 и Intel стартира поддръжка на паралелизъм в Parallel Studio като добавка за Visual Studio. Редакторът на код на Visual Studio 2010 вече подчертава препратките; всеки път, когато е избран символ, всички други употреби на символа се маркират. Той също така предлага функция за бързо търсене за постепенно търсене във всички символи в проекти на C++, C# и VB.NET. Бързото търсене поддържа съвпадения на поднизове и търсене на camelCase . Функцията за йерархия на повикванията позволява на разработчика да види всички методи, които са извиквани от текущ метод, както и методите, които извикват текущия. IntelliSense във Visual Studio поддържа първо потреблениережим, в който разработчиците могат да се включат. В този режим IntelliSense не попълва автоматично идентификатори; това позволява на разработчика да използва недефинирани идентификатори (като имена на променливи или методи) и да ги дефинира по-късно. Visual Studio 2010 също може да помогне в това, като ги дефинира автоматично, ако може да изведе типовете им от употребата. Текущите версии на Visual Studio имат известна грешка, която прави IntelliSense неизползваем за проекти, използващи чист C (не C++). Visual Studio 2010 разполага с нова помощна система, която замества визуализатора на MSDN библиотека . Помощната система вече не се базира на Microsoft Help 2 и не използва Microsoft Document Explorer. Динамичната помощ, съдържаща връзки към помощни елементи въз основа на това, което разработчикът е правил по това време, беше премахната в окончателното издание,но може да бъде добавена обратно чрез изтегляне от Microsoft. Visual Studio 2010 вече не поддържа разработка за Windows Mobile преди Windows Phone 7 . Visual Studio 2010 Service Pack 1 беше пуснат през март 2011 г.

**Ultimate 2010**

Visual Studio Ultimate 2010 замества Visual Studio 2008 Team Suite. Включва нови инструменти за моделиране, като Architecture Explorer , който графично показва проекти и класове и връзките между тях. Поддържа UML диаграма на активността, диаграма на компонент, диаграма на (логическа) клас, диаграма на последователност и диаграма на случаите на използване. Visual Studio Ultimate 2010 също включва анализ на въздействието на теста , който предоставя намеци за това кои тестови случаи са повлияни от модификации на изходния код, без реално да се изпълняват тестовите случаи. Това ускорява тестването, като избягва провеждането на ненужни тестови случаи.Visual Studio Ultimate 2010 също включва исторически инструмент за отстраняване на грешки за управляван код, наречен IntelliTrace . За разлика от традиционния дебъгер, който записва само активния в момента стек, IntelliTrace записва всички събития, като предишни извиквания на функции, параметри на метод, събития и изключения. Това позволява изпълнението на кода да бъде пренавито, в случай че не е била зададена точка на прекъсване, където е възникнала грешката. Отстраняването на грешки с IntelliTrace кара приложението да работи по-бавно от отстраняването на грешки без него и използва повече памет, тъй като трябва да бъдат записани допълнителни данни. Microsoft позволява конфигуриране на това колко данни трябва да бъдат записани, което на практика позволява на разработчиците да балансират скоростта на изпълнение и използването на ресурси. Ръководството на лабораториятакомпонентът на Visual Studio Ultimate 2010 използва виртуализация, за да създаде подобна среда за изпълнение за тестери и разработчици. Виртуалните машини са маркирани с контролни точки, които по - късно могат да бъдат разследвани за проблеми, както и за възпроизвеждане на проблема. Visual Studio Ultimate 2010 също така включва възможността за записване на тестове, които улавят специфичното състояние на операционната среда, както и точните стъпки, използвани за провеждане на теста. След това тези стъпки могат да бъдат възпроизведени, за да се възпроизведат проблеми.

**2012**

Окончателната версия на Visual Studio 2012 беше обявена на 1 август 2012 г., а официалното стартиране се проведе на 12 септември 2012 г. За разлика от предишните версии, Visual Studio 2012 не може да записва и възпроизвежда макроси и редакторът на макроси е премахнат. Новите функции включват поддръжка за WinRT и C++/CX (компонентни разширения) и C++ AMP ( програмиране на GPGPU ) семантично оцветяване. На 16 септември 2011 г. на уебсайта на Microsoft беше публикуван пълен „Преглед за разработчици“ на Visual Studio 11. Visual Studio 11 Developer Preview изисква Windows 7, Windows Server 2008 R2, Windows 8 или по-нова операционна система. Версиите на Microsoft Foundation Class Library (MFC) и C runtime (CRT), включени в тази версия, не могат да произвеждат софтуер, който е съвместим с Windows XP или Windows Server 2003, освен чрез използване на естествено многоцелеви и отказ от най-новите библиотеки, компилатори, и заглавки. Въпреки това, на 15 юни 2012 г., публикация в блога в блога на VC++ Team обяви, че въз основа на отзивите на клиентите, Microsoft ще въведе отново естествена поддръжка за целите на Windows XP (макар и не за XP като платформа за разработка) във версия на Visual C++ да бъде пусната по-късно през есента на 2012 г. „Visual Studio 2012 Update 1“ (Visual Studio 2012.1) беше пусната през ноември 2012 г. Тази актуализация добави поддръжка за целите на Windows XP и също добави други нови инструменти и функции (напр. подобрени поддръжка за диагностика и тестване за приложения от Windows Store). На 24 август 2011 г. публикация в блог от Sumit Kumar, програмен мениджър в екипа на Visual C++, изброява някои от функциите на предстоящата версия на Visual Studio C++ IDE:

* Семантично оцветяване: Подобрено оцветяване на синтаксиса, различни цветове, дефинирани от потребителя или цветове по подразбиране за синтаксис на C++, като макроси, изброявания, имена на типове и функции.
* Осветяване на препратка : Изборът на символ подчертава всички препратки към този символ в обхвата.
* New Solution Explorer : Новият Solution Explorer позволява визуализиране на йерархии на класове и файлове в рамките на решение/проект. Може да търси извиквания към функции и употреби на класове.
* Автоматично показване на списъка IntelliSense : IntelliSense се показва автоматично по време на въвеждане на код, за разлика от предишни версии, където трябваше да бъде извикан изрично чрез използване на определени оператори (т.е. оператор на обхват (::)) или клавишни комбинации ( *Ctrl-Space* или *Ctrl -J* ).
* Филтриране на списък с членове : IntelliSense използва размита логика, за да определи кои функции/променливи/типове да се показват в списъка.
* Кодови фрагменти : Кодови фрагменти са включени в IntelliSense за автоматично генериране на подходящ код въз основа на параметрите на потребителя, могат да се създават персонализирани кодови фрагменти.

Изходният код на Visual Studio 2012 се състои от приблизително 50 милиона реда код.

**Отказ на интерфейса**

По време на бета версията на Visual Studio 11, Microsoft елиминира използването на цвят в инструменти, освен в случаите, когато цветът се използва за уведомяване или промяна на състоянието. Въпреки това, използването на цвят беше върнато след обратна връзка, изискваща повече контраст, диференциация, яснота и "енергия" в потребителския интерфейс. Във Visual Studio 2012 RC основна промяна в интерфейса е използването на лента с менюта с всички главни букви, като част от кампанията за поддържане на Visual Studio в съответствие с посоката на други потребителски интерфейси на Microsoft и за предоставяне на добавена структура към горното меню бар зона. Редизайнът беше критикуван, че е труден за четене и противоречи на тенденциите, започнати от разработчиците да използват CamelCase , за да изпъкнат по-добре думите. Някои спекулираха, че основната причина за редизайна е включването на опростения вид и усещане на програмите на Metro. Въпреки това съществува опция за системния регистър на Windows , която позволява на потребителите да деактивират интерфейса с всички главни букви.

**2013**

Визуализацията за Visual Studio 2013 беше обявена на конференцията Build 2013 и беше предоставена на 26 юни 2013 г. Visual Studio 2013 RC (Release Candidate) беше предоставена на разработчиците в MSDN на 9 септември 2013 г. Окончателната версия на Visual Studio 2013 стана достъпна за изтегляне на 17 октомври 2013 г. заедно с .NET 4.5.1. Visual Studio 2013 официално стартира на 13 ноември 2013 г. на събитие за виртуално стартиране, представено от С. Сомасегар и хоствано на събития .visualstudio .com . „Актуализация 1 на Visual Studio 2013“ (Visual Studio 2013.1) беше пусната на 20 януари 2014 г. Visual Studio 2013.1 е целенасочена актуализация, която адресира някои ключови области на обратната връзка с клиентите. „Visual Studio 2013 Update 2“ (Visual Studio 2013.2) беше издаден на 12 май 2014 г. Visual Studio 2013 Update 3 беше пуснат на 4 август 2014 г. С тази актуализация Visual Studio предоставя опция за деактивиране на менютата с главни букви, която беше въведена в VS2012. „Visual Studio 2013 Update 4“ (Visual Studio 2013.4) беше издаден на 12 ноември 2014 г. „Visual Studio 2013 Update 5“ (Visual Studio 2013.5) беше пуснат на 20 юли 2015 г.

**2015**

Първоначално наричан Visual Studio „14“, първият преглед на технологиите на Общността (CTP) беше пуснат на 3 юни 2014 г. , а кандидатът за изданието беше пуснат на 29 април 2015 г.; Visual Studio 2015 беше официално обявено като окончателно име на 12 ноември 2014 г.Visual Studio 2015 RTM беше пуснат на 20 юли 2015 г. Visual Studio 2015 Update 1 беше пуснат на 30 ноември 2015 г. Visual Studio 2015 Update 2 беше пуснат на 30 март 2016 г. Visual Studio 2015 Актуализация 3 беше пусната на 27 юни 2016 г.

**2017**

Първоначално наричан Visual Studio "15", той беше пуснат на 7 март 2017 г. Първият преглед беше пуснат на 30 март 2016. Visual Studio "15" Preview 2 беше пуснат на 10 май 2016. Visual Studio "15" Preview 3 беше пуснат на 7 юли 2016 г. Visual Studio "15" Preview 4 беше пуснат на 22 август 2016 г. Visual Studio „15“ Preview 5 беше пуснат на 5 октомври 2016 г. На 14 ноември 2016 г., за кратък период от време, Microsoft пусна публикация в блог, разкриваща версията на името на продукта на Visual Studio 2017 заедно с предстоящите функции. На 16 ноември 2016 г. „Visual Studio 2017“ беше обявено като окончателно име и беше пуснато Visual Studio 2017 RC . На 7 март 2017 г. Visual Studio 2017 беше пуснат за общодостъпност. На 14 март 2017 г. беше пусната първата корекция за Visual Studio 2017 поради неуспехи по време на инсталиране или отваряне на решения в първата версия. На 5 април 2017 г. беше пуснато Visual Studio 2017 15.1 и беше добавена поддръжка за насочване към .NET Framework 4.7.На 10 май 2017 г. Visual Studio 2017 15.2 беше пуснат и добави ново работно натоварване „Наука за данни и работно натоварване на аналитични приложения“. Актуализация за коригиране на темата с тъмни цветове беше пусната на 12 май 2017 г.На 14 август 2017 г. излезе Visual Studio 2017 15.3 и добави поддръжка за насочване към .NET Core 2.0. Актуализация (15.3.1) беше пусната четири дни по-късно за справяне с уязвимостта на Git с подмодули ( CVE 2017-1000117 ).На 10 октомври 2017 г. излезе Visual Studio 15.4. На 4 декември 2017 г. излезе Visual Studio 15.5. Тази актуализация съдържаше големи подобрения в производителността, нови функции, както и корекции на грешки.На 6 март 2018 г. излезе Visual Studio 15.6. Включва актуализации за тестване на модули и производителност. На 7 май 2018 г. излезе Visual Studio 15.7. Той включваше актуализации навсякъде, включително инсталатора, редактора, инструмента за отстраняване на грешки, наред с други. Почти всички точкови издания, най-новата от които е 15.7.6 , издадена на 2 август 2018 г., включват актуализации на сигурността. С пускането на Visual Studio 2017 15.7, Visual C++ вече отговаря на стандарта C++17 . На 20 септември 2018 г. излезе Visual Studio 15.8.5. Инструментите за Xamarin вече поддържат Xcode 10. Visual Studio 2017 предлага нови функции като поддръжка за EditorConfig (рамка за прилагане на стил на кодиране), поддръжка на NGen , набор от инструменти за .NET Core и Docker (Преглед) и Xamarin 4.3 (Преглед). Освен това има XAML редактор, подобрен IntelliSense, тестване на модули на живо, подобрение за отстраняване на грешки и по-добро IDE изживяване и производителност. Освен това, това е последната версия на Visual Studio, която поддържа поддръжка на проекти за Windows 10 Mobile.

**2019**

На 6 юни 2018 г. Microsoft обяви Visual Studio 2019 (версия 16).

На 4 декември 2018 г. излезе Visual Studio 2019 Preview 1.

На 24 януари 2019 г. беше пуснат Visual Studio 2019 Preview 2.

На 13 февруари 2019 г. излезе Visual Studio 2019 Preview 3.

На 27 февруари 2019 г. излезе Visual Studio 2019 RC.

Той е общодостъпен (GA) от 2 април 2019 г. и е достъпен за изтегляне.

**2022**

На 19 април 2021 г. Microsoft обяви Visual Studio 2022 (версия 17). Това е първата версия, която се изпълнява като 64-битов процес, позволяващ на главния процес на Visual Studio да има достъп до повече от 4 GB памет, предотвратявайки изключения при липса на памет, които могат да възникнат при големи проекти.

На 17 юни 2021 г. беше пуснат Visual Studio 2022 Preview 1.

На 14 юли 2021 г. излезе Visual Studio 2022 Preview 2.

На 10 август 2021 г. излезе Visual Studio 2022 Preview 3.

На 14 септември 2021 г. излезе Visual Studio 2022 Preview 4.

На 12 октомври 2021 г. бяха пуснати Visual Studio 2022 RC и Preview 5, като се настрои 8 ноември 2021 г. за общодостъпност.

Той е общодостъпен (GA) от 8 ноември 2021 г. и е достъпен за изтегляне.

**Свързани продукти**

**Azure DevOps Services**

На 13 ноември 2013 г. Microsoft обяви пускането на софтуер като услуга за предлагане на Visual Studio на платформата Microsoft Azure ; по това време Microsoft го нарича Visual Studio Online. По-рано обявен като Team Foundation Services, той се разшири върху локалния Team Foundation Server (TFS; сега известен като Azure DevOps Server ), като го направи достъпен в Интернет и внедри модел на непрекъснато издание . Клиентите могат да използват портала на Azure, за да се абонират за Visual Studio Online. Абонатите получават хоствана Git -съвместима система за контрол на версиите , услуга за тестване на натоварване, телеметрияуслуга и редактор на код в браузъра с кодово име "Монако". По време на Connect(); Събитие за разработчици 2015 на 18 ноември 2015 г., Microsoft обяви, че услугата е ребрандирана като „Visual Studio Team Services (VSTS)“. На 10 септември 2018 г. Microsoft обяви друго ребрандиране на услугата, този път на „Azure DevOps Services“. Microsoft предлага нива на достъп на заинтересовани страни, основни и Visual Studio на абонати за услугите Azure DevOps. Основният план е безплатен за до пет потребители. Потребители с абонамент за Visual Studio могат да бъдат добавени към план без допълнителна такса.

**Управление на жизнения цикъл на приложението Visual Studio**

Visual Studio Application Lifecycle Management (ALM) е колекция от интегрирани инструменти за разработка на софтуер, разработени от Microsoft. Тези инструменти в момента се състоят от IDE (Visual Studio 2015 Community и по-големи издания), сървър (Team Foundation Server) и облачни услуги (Visual Studio Team Services). Visual Studio ALM поддържа екипно базирано развитие и сътрудничество, Agile управление на проекти, DevOps , контрол на източниците, пакетиране, непрекъснато развитие, автоматизирано тестване, управление на версиите, непрекъсната доставка и инструменти за отчитане за приложения и услуги. Във Visual Studio 2005 и Visual Studio 2008 марката беше известна като Microsoft Visual Studio Team System (VSTS). През октомври 2009 г. марката Team System беше преименувана на Visual Studio ALM с изданието на Visual Studio 2010 (кодово име „Rosario“). Visual Studio Team Services дебютира като Visual Studio Online през 2013 г. и беше преименуван през 2015 г.

**Управление на лабораторията на Visual Studio**

Visual Studio Lab Management е инструмент за разработка на софтуер, разработен от Microsoft за софтуерни тестери за създаване и управление на виртуални среди. Lab Management разширява съществуващата платформа за управление на жизнения цикъл на приложенията на Visual Studio , за да даде възможност за интегрирана тестова лаборатория, базирана на Hyper-V . От Visual Studio 2012 той вече се доставя като част от него; и може да се настрои след интегриране на Azure DevOps и SCVMM.

**Visual Studio LightSwitch**

Microsoft Visual Studio LightSwitch е разширение и рамка, специално пригодени за създаване на бизнес приложения, изградени върху съществуващи .NET технологии и платформи на Microsoft. Създадените приложения са архитектурно 3-степенни : потребителският интерфейс работи на Microsoft Silverlight или HTML 5 клиент или като приложение на SharePoint 2013; нивото на логиката и достъпа до данни е изградено на WCF Data Services и е изложено като OData канал, хостван в ASP.NET; и основното съхранение на данни поддържа Microsoft SQL Server Express , Microsoft SQL Server иMicrosoft SQL Azure . LightSwitch също така поддържа други източници на данни, включително Microsoft SharePoint , OData и WCF RIA Services.LightSwitch включва графични дизайнери за проектиране на обекти и връзки на обекти, заявки за обекти и екрани на потребителския интерфейс. Бизнес логиката може да бъде написана на Visual Basic или Visual C# . LightSwitch е включен в Visual Studio 2012 Professional и по-нова версия. Visual Studio 2015 е последната версия на Visual Studio, която включва инструментите LightSwitch. Слоят на потребителския интерфейс вече е незадължителен компонент при внедряване на решение на LightSwitch, което позволява разгръщане само за услуга.

Първата версия на Visual Studio LightSwitch, издадена на 26 юли 2011 г., имаше много разлики от текущата пускане на LightSwitch. По-специално инструментът е закупен и инсталиран като самостоятелен продукт. Ако Visual Studio 2010 Professional или по-нова версия вече е инсталирана на машината, LightSwitch ще се интегрира в това. Втората основна разлика е, че средното ниво е изградено и изложено с помощта на WCF RIA Services.От 14 октомври 2016 г. Microsoft вече не препоръчва LightSwitch за разработка на нови приложения.

**Код на Visual Studio**

Visual Studio Code е безплатен редактор на изходен код , заедно с други функции, за Linux, Mac OS и Windows. Той също така включва поддръжка за отстраняване на грешки и вграден Git Control . Той е изграден върху отворен код , и на 14 април 2016 г. е пуснат като версия 1.0.

**Visual Studio Team System Profiler**

Visual Studio Team System Profiler (VSTS Profiler) е инструмент за анализиране на производителността на .NET проекти, който анализира пространствената и времева сложност на програмата. Той анализира кода и подготвя доклад, който включва вземане на проби от процесора, инструментариум, разпределение на .NET памет и спорове за ресурси .