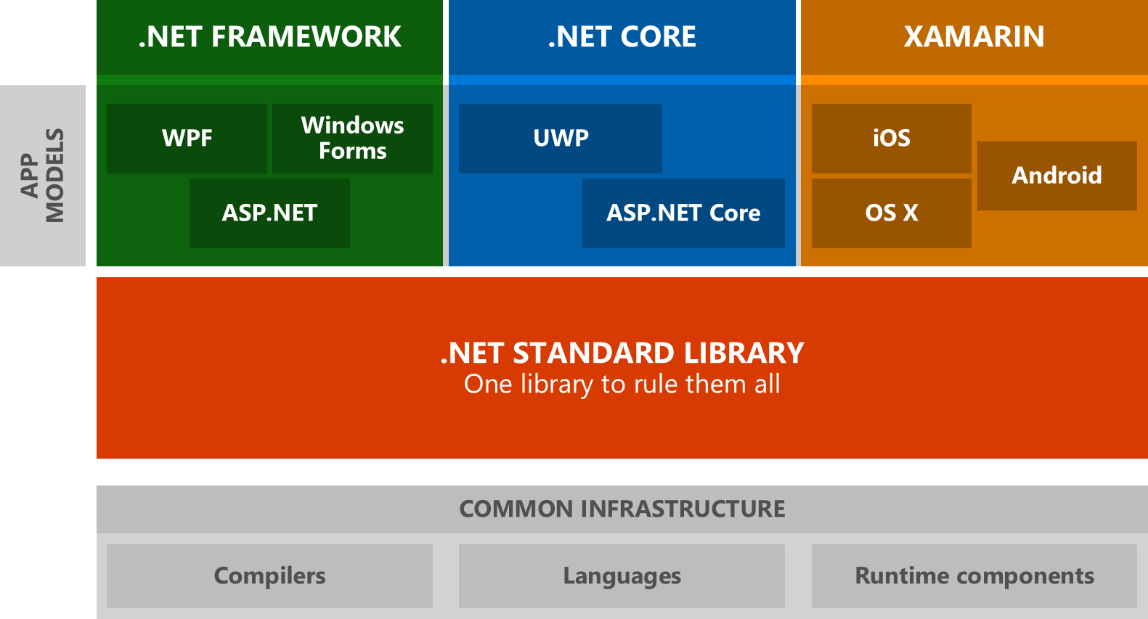
# Platforma .NET

Platforma .NET je rozhraní **zastřešující prostředí pro různé komponenty** vytvořené společností Microsoft.  
Obsahuje veliké množství knihoven, pro **vývoj Desktopových, webových či mobilních aplikací.**  
Zastřešuje také několik programovacích jazyků a umožnuje jejich vzájemné používání.  
Aplikace napsané na této platformě se před během překládají do **Common Language Runtime (CLR)**CLR se poté spouští na architektuře daného počítače.  


## Programovací jazyky v .NET

Platforma disponuje několika jazyky, ale nejpodstatnější z nich je jazyk C#

**C#:**  je **moderní vysokoúrovňový objektově orientovaný programovací jazyk.** syntaxe inspirována jazykem C a základy bere o jazyků C++ a Java.  
 Dalším podstatnou vlastností jazyka je přenositelnost, na kterou se dávalo přednost  
 společně s platformu .NET.

**F#:** je jazyk zaměřen na funkcionální programování s imperativním  
 objektově orientovaným přístupem.

Používá se pro **programování asynchronních operací a více vláknových aplikací** pro  
 víceprocesorové systémy.

**Visual Basic:** je jazyk, který **si zakládá na jednoduchosti**, slouží k rychlému vývoji aplikací,je také objektově orientovaný, není Case sensitive.  
 Pro psaní kódu často využívá normální slovíčka z angličtiny (AND, OR, END…)

## Proč je dobré umět platformu .NET?

**Z pohledu vývojáře:** Umožňuje vývoj **všech věcí v jednom prostředí** konkrétně ve Visual Studiu  
 Celá platforma **.NET je objektově orientovaná.**  
 Programovací jazyk C# je velice rozšířený a jde s ním udělat hodně věcí.  
 Od konzolové kalkulačky až po vývoj 3D her.  
 Vysoce výkonné kompilery.  
 Mnoho aplikací je vyvíjeno na platformě .NET a v jazyku C shapr, což vede k velké  
 rozmanitosti ohledně výběru zaměstnavatele.  
 Microsoft se více a více otevírá open-source, např: NET Core.

**Z pohledu studenta:** Aktivní komunita, řešení problémů je mnohem snazší.  
 Protože C# spadá mezi moderní jazyky, který se také stále vyvíjí, nalezneme  
 nespočet užitečných návodů na internetu.  
 Jedním jazykem můžeme napsat aplikaci pro Windows, pro Android či iOS.  
 Vysoká podpora ze stran škol, tento jazyk se často vyučuje na školách.

## Microsoft Store

Je distribuční platforma, pro aplikace, programy, hry, videí, muziky… pro operační systémy Windows  
Slouží k rychlému a bezpečnému stažení / nákupu.  
// nevím co přesně by zde mělo být napsáno

## Oddělní části designu a části kódu

Při moderním vývoji aplikací **klademe také důraz na efektivitu a znovu použitelnost kódu,**  
to je hlavně potřeba pokud píšeme multiplatformní aplikaci, nejčastěji se bude jednat o Android/iOS aplikaci. Jde o to abychom nemuseli psát celou aplikaci úplně znova.  
Proto nám **.NET umožňuje používat samostatný kód pro logickou část aplikace a pro design  
tzv. XAML**, což je značkovací jazyk používaný k popisu grafického rozhraní.

Např. u Xamarinu (open source platforma, pro vývoj mobilních aplikací) nám stačí napsat jeden kód a bude fungovat jak na Androidu, tak také na iOS, jen stačí upravit vzhled pomocí XAML.

## Windows Forms, WPF, UFW, APS.NET

**Windows Forms:** Je jedna z prvních knihoven platformy .NET, vydána byla již roku 2002  
 Slouží k vývoji grafický aplikací na bázi oken pro operační systém Windows.  
 Není multiplatformní. Nepoužíváme XAML  
 Grafické prostředí vytváříme pomocí drag drop nabídky.  
 Vhodné pro vývoj menších, nenáročných aplikací.

**WPF:** (Windows Presentation Foundation) Jedná se opět o knihovnu na vývoj  
desktopových aplikací pro systém Windows, není multiplatformní.  
Výhody oproti Windows Forms, je lepší podpora grafických prvků  
např: Vektorová grafika, 3D Rendering pomocí Direct3D a práce s médii, animace.  
Lepší podpora různých DPI. Již používáme XAML.  
Velká volnost u vzhledu aplikace.

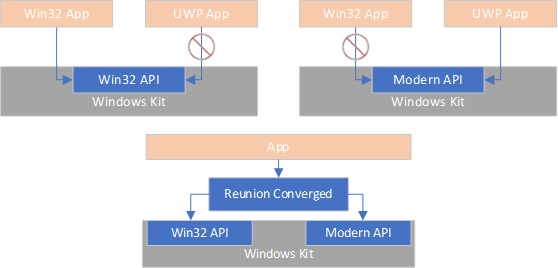
**UWP:** (Universal Windows Platform) Jak již z názvu vypovídá jedná se o vývoj univerzálních aplikací, které bez přepisování kódu fungují na Windows 10, Xbox One, má být částečným nástupcem WPF. Opět používáme XAML pro grafické rozhraní.  
UWP aplikace se dají dobře implementovat do samostatného systému, umí posílat notifikace, přizpůsobují se vzhledu systému (např: Dark Mode) mají jednoduchý a čistý vzhled, starší systémy Windows. Dobře fungují s dotykovými obrazovkami.

**ASP.NET:** Jedná se o framework pro vývoj webových aplikacích a služeb s .NET a C#  
obsahuje již užitečné nástroje a knihovny pro urychlení vývoje.  
Protože kód je kompilovaný je mnohem rychlejší než např. Node.js  
Má výhody pro vývojáře přecházející z Windows, podobný přístup  
takže není těžké přejít a není potřeba se učit nové jazyky např. PHP, Javascript

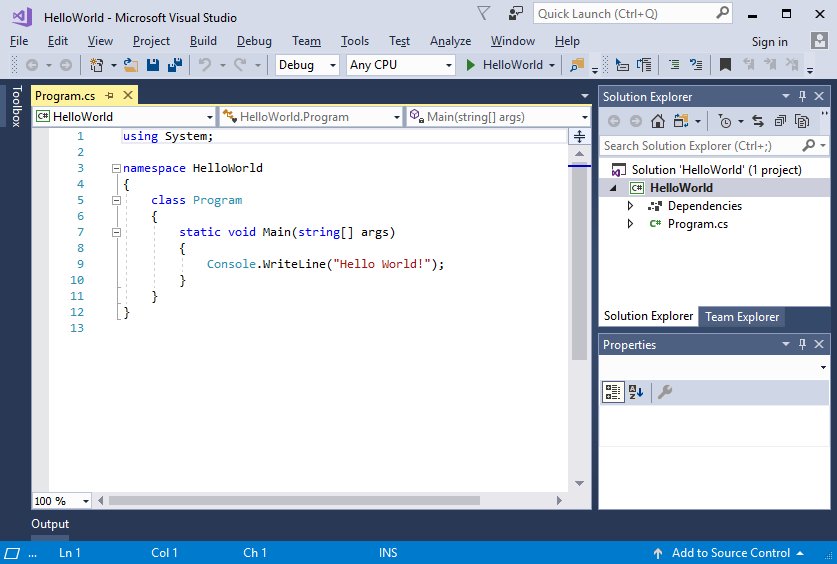
## Win32 vs UWP a Project Reunion

**Win32:** Je původní platforma pro vývoj aplikací, když ještě nebyla platforma .NET  
 Pro vývoj se používá C / C++. Dnes používáme, pokud je rychlost aplikace na  
 první místě, máme přímí přístup k hardwaru počítače, (kompilovaný jazyk = rychlost)

**UWP:** Je nejnovější platforma pro vývoj aplikací, jak již bylo řečeno odděluje  
 kód od vzhledu aplikace, který píšeme samostatně. Máme na výběr oproti Win32  
 více knihoven a nemusíme si většinu věcí psát sami, přicházíme ale o lepší výkon,  
 což ale u dnešních počítačů vůbec nevadí.  
 Pokud aplikace nebude počítat složité matematické vzorce, simulovat, 3D renderovat,  
 tak vždy upřednostníme .NET platformu. Vývoj bude mnohem jednodušší.

 **Project Reunion:** Jedná se o projekt, který má za úkol sjednotit vývojové platformy pro Windows.  
 Sjednotí Win32 a UWP API (Application Programming Interface)  
 Opět ulehčí vývoj Win32 aplikací a zmodernizuje platformu

## Co je potřeba pro vývoj?

Nejpodstatnější nástroj je **Visual Studio je kompletní IDE** (vývojové prostředí)  
které **obsahuje všechny potřebné nástroje pro vývoj** aplikací Win32 až po UWP.  
Také obsahuje Designer pro každou platformu od Windows Forms po UWP.  
Umí také správu databáze, či týmovou práci na projektech.

## Vysvětlení názvů

**SDK:**  (Software development kit)  
 Jedná se o sadu nástrojů pro vývoj daného softwaru, obsahuje knihovny, APIs…  
 Vše potřebné pro vývoj softwaru  
  
**IDE:** (Integrated Development Environment)  
 Jedná se o aplikaci, která pomáhá s vývojem, má nástroje na kontrolu chyb syntaxe,  
 či umí napovídat a při kompilaci programu řekne co je špatně, přehledně spravuje projekt.

**API:** (Application Programming Interface)  
 Jedná se rozhraní, které dovoluje komunikovat mezi různými aplikacemi.  
 Např. existuje API, které nám umožňuje z YouTube brát přesný počet odběratelů  
 a my můžeme tuto funkčnost využít v našem programu. Cílem API je propojení  
 stávajících aplikací aby mohli mezi s sebou jednoduše komunikovat.

**Emulátor:** Jedná se o software, který nám umožňuje běh počítačových programů na jiné platformě  
 např. Chci si spustit starou aplikaci na MS-Dos, použiji tedy emulátor, abych mohl tuto  
 aplikaci spustit. Rozdíl oproti simulátoru je, že emulátor je přesný, software poběží přesně,  
 tak jak by běžel na původním hardwaru.

## Dokumentace:

<https://docs.microsoft.com/cs-cz/documentation/>

<https://docs.microsoft.com/cs-cz/dotnet/>

<https://dotnet.microsoft.com/apps/aspnet>