

Tvorba mobilních aplikací

11. seminář

Radek Janošík

Univerzita Palackého v Olomouci

26. 4. 2022

Multiplatformní vývoj – motivace

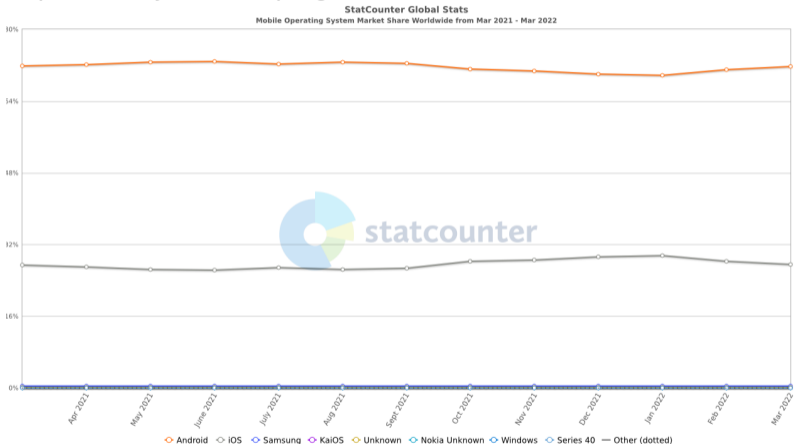
- Představte si, že máte nápad na mobilní aplikaci, která „udělá díru do světa“

Multiplatformní vývoj – motivace

- Představte si, že máte nápad na mobilní aplikaci, která „udělá díru do světa“
 - ▶ Pro jakou platformu ji začnete programovat? Android? iOS? Windows Phone?

Multiplatformní vývoj – motivace

- Představte si, že máte nápad na mobilní aplikaci, která „udělá díru do světa“
 - ▶ Pro jakou platformu ji začnete programovat? Android? iOS? Windows Phone?



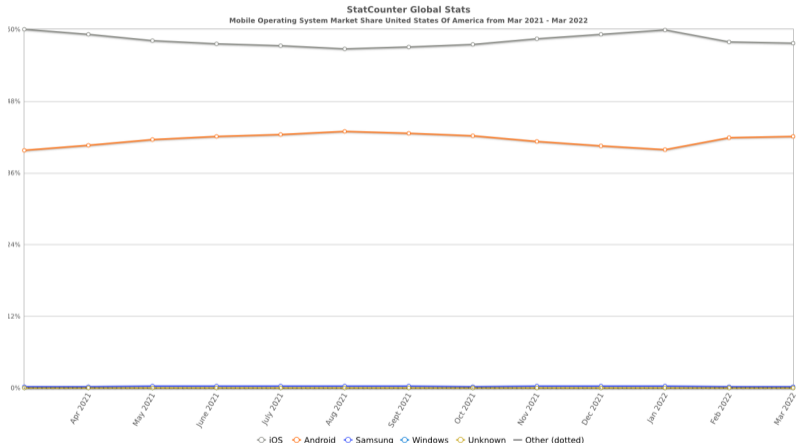
Obrázek: Podíl operačních systémů na mobilních zařízeních ([statcounter.com](https://www.statcounter.com))

Multiplatformní vývoj – motivace

- Z obrázku by se volba pro poměr cena/výkon mohla zdát jasná
 - ▶ Ale...

Multiplatformní vývoj – motivace

- Z obrázku by se volba pro poměr cena/výkon mohla zdát jasná
 - ▶ Ale...



Obrázek: Podíl operačních systémů na mobilních zařízeních v USA ([statcounter.com](https://www.statcounter.com))

Multiplatformní vývoj – motivace

- No dobrá, budeme vyvíjet pro obě platformy
 - ▶ Čím více lidí oslovíme, tím více na nich vyděláme
 - ▶ Jsou uživatelé Apple bohatší? Ochotní vydat více peněz?

Multiplatformní vývoj – motivace

- No dobrá, budeme vyvíjet pro obě platformy
 - ▶ Čím více lidí oslovíme, tím více na nich vyděláme
 - ▶ Jsou uživatelé Apple bohatší? Ochotní vydat více peněz?
- Platformy jsou velmi odlišné, ale v něčem podobné
- ⇒ Potřebujeme 2x tolik lidí a 2.5x tolik peněz

Multiplatformní vývoj – motivace

- No dobrá, budeme vyvíjet pro obě platformy
 - ▶ Čím více lidí oslovíme, tím více na nich vyděláme
 - ▶ Jsou uživatelé Apple bohatší? Ochotní vydat více peněz?
- Platformy jsou velmi odlišné, ale v něčem podobné
- ⇒ Potřebujeme 2x tolik lidí a 2.5x tolik peněz
- Nešlo by napsat jeden kód a *nějak* jej přeložit pro více platform?

Multiplatformní vývoj – motivace

- No dobrá, budeme vyvíjet pro obě platformy
 - ▶ Čím více lidí oslovíme, tím více na nich vyděláme
 - ▶ Jsou uživatelé Apple bohatší? Ochotní vydat více peněz?
- Platformy jsou velmi odlišné, ale v něčem podobné
- ⇒ Potřebujeme 2x tolik lidí a 2.5x tolik peněz
- Nešlo by napsat jeden kód a *nějak* jej přeložit pro více platform?
- Naštěstí řešení existuje
 - ▶ Ale má své pro a i proti

Multiplatformní vývoj (1 / 2)

- Idea: Napišme univerzální kód a spusťme jej „nějak“ na více platformách

Multiplatformní vývoj (1 / 2)

- Idea: Napišme univerzální kód a spusťme jej „nějak“ na více platformách
- Většinou toho nelze 100% dosáhnout
 - ▶ Specifické vlastnosti platforem (přístup k HW, ...)
 - ▶ Vzhled – Chceme stejný vzhled na všech platformách?

Multiplatformní vývoj (1 / 2)

- Idea: Napišme univerzální kód a spusťme jej „nějak“ na více platformách
- Většinou toho nelze 100% dosáhnout
 - ▶ Specifické vlastnosti platforem (přístup k HW, ...)
 - ▶ Vzhled – Chceme stejný vzhled na všech platformách?
 - ▶ Je dobré dodržovat zažité nároky na vzhled

Multiplatformní vývoj (1 / 2)

- Idea: Napišme univerzální kód a spusťme jej „nějak“ na více platformách
- Většinou toho nelze 100% dosáhnout
 - ▶ Specifické vlastnosti platform (přístup k HW, ...)
 - ▶ Vzhled – Chceme stejný vzhled na všech platformách?
 - ▶ Je dobré dodržovat zažité nároky na vzhled
- Reálně např.: 90 % sdíleného kódu, 10 % platform-specific

Multiplatformní vývoj (1 / 2)

- Idea: Napišme univerzální kód a spusťme jej „nějak“ na více platformách
- Většinou toho nelze 100% dosáhnout
 - ▶ Specifické vlastnosti platform (přístup k HW, ...)
 - ▶ Vzhled – Chceme stejný vzhled na všech platformách?
 - ▶ Je dobré dodržovat zažité nároky na vzhled
- Reálně např.: 90 % sdíleného kódu, 10 % platform-specific
- Je možné udělat veškerou funkcionalitu nativní aplikací multiplatformně?

Multiplatformní vývoj (1 / 2)

- Idea: Napišme univerzální kód a spusťme jej „nějak“ na více platformách
- Většinou toho nelze 100% dosáhnout
 - ▶ Specifické vlastnosti platform (přístup k HW, ...)
 - ▶ Vzhled – Chceme stejný vzhled na všech platformách?
 - ▶ Je dobré dodržovat zažité nároky na vzhled
- Reálně např.: 90 % sdíleného kódu, 10 % platform-specific
- Je možné udělat veškerou funkcionalitu nativní aplikací multiplatformně?
 - ▶ Ano, nic tomu nebrání. Možná se trochu nadřeme

Multiplatformní vývoj (1 / 2)

- Idea: Napišme univerzální kód a spusťme jej „nějak“ na více platformách
- Většinou toho nelze 100% dosáhnout
 - ▶ Specifické vlastnosti platform (přístup k HW, ...)
 - ▶ Vzhled – Chceme stejný vzhled na všech platformách?
 - ▶ Je dobré dodržovat zažité nároky na vzhled
- Reálně např.: 90 % sdíleného kódu, 10 % platform-specific
- Je možné udělat veškerou funkcionalitu nativní aplikací multiplatformně?
 - ▶ Ano, nic tomu nebrání. Možná se trochu nadřeme
- Multiplatformní aplikace mohou mít horší UX
 - ▶ Znalý uživatel může být schopen rozpoznat
 - ▶ Laik většinou již ne

Multiplatformní vývoj (1 / 2)

- Idea: Napišme univerzální kód a spusťme jej „nějak“ na více platformách
- Většinou toho nelze 100% dosáhnout
 - ▶ Specifické vlastnosti platform (přístup k HW, ...)
 - ▶ Vzhled – Chceme stejný vzhled na všech platformách?
 - ▶ Je dobré dodržovat zažitá nároky na vzhled
- Reálně např.: 90 % sdíleného kódu, 10 % platform-specific
- Je možné udělat veškerou funkcionalitu nativní aplikací multiplatformně?
 - ▶ Ano, nic tomu nebrání. Možná se trochu nadřeme
- Multiplatformní aplikace mohou mít horší UX
 - ▶ Znalý uživatel může být schopen rozpoznat
 - ▶ Laik většinou již ne
- Dobře udělaná multiplatformní aplikace může poskytnout lepší UX než odbytá nativní aplikace

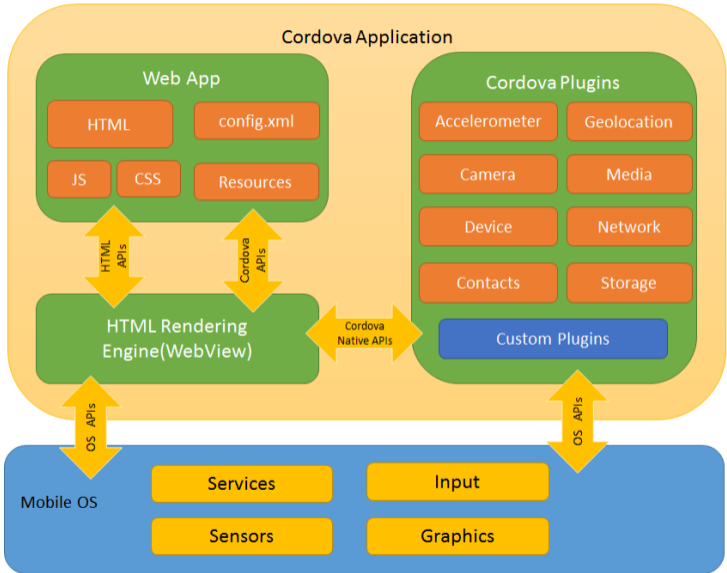
Multiplatformní vývoj (2 / 2)

- Velmi často se dnes používají webové technologie (HTML, CSS, JS)
 - ▶ Mobilní aplikace složena z nativní *obálky* a multiplatformní části
 - ▶ *Obálka* zpřístupní multiplatformní části platform-specific funkcionalitu
- Nebo *obálka* může umožňovat spouštět aplikace z jiného jazyka
 - ▶ Aplikace možné psát v Pythonu, C#, C++ (QT mobile), ...
 - ▶ Složitější přístup, ale také funkční
- Multiplatformních frameworků existuje celá řada
 - ▶ Ale „prvních pár obsadí 90 % trhu“
 - ▶ Popularita velmi proměnlivá v čase, neustálý vývoj
- Dnes si pouze projdeme několik základních možností
 - ▶ Ne do hloubky, pouze přehledově

Apache Cordova

- Open-source framework pro multiplatformní vývoj mobilních aplikací
- Vychází z PhoneGap, zastřešeno Apache, cordova.apache.org
- Běh standardních (téměř libovolných) webových technologií na více platformách
 - ▶ Podpora Androidu, iOS, Desktop (Electron)
 - ▶ Windows (UWP), OSX – zastaralé
- Složeno z nativní *obálky* a webové aplikace
- Stanovené API pro přístup k zařízení (fotoaparát, sensory, poloha, ...)
- Vhodné pro lidi se zkušenostmi s webovým vývojem

Apache Cordova – architektura



Apache Cordova – instalace a spuštění

- Apache Cordova je dostupná přes správce balíčků pro Node.js `npm`
 - ▶ Doporučuji nainstalovat, bude se vám hodit i v dalších kurzech
- Jelikož již máme Android SDK a emulátory, tak je instalace snadná

Apache Cordova – instalace a spuštění

- Apache Cordova je dostupná přes správce balíčků pro Node.js `npm`
 - ▶ Doporučuji nainstalovat, bude se vám hodit i v dalších kurzech
- Jelikož již máme Android SDK a emulátory, tak je instalace snadná
- Nainstalujeme Apache Cordova do systému s právy administrátora
`npm install -g cordova`

Apache Cordova – instalace a spuštění

- Apache Cordova je dostupná přes správce balíčků pro Node.js `npm`
 - ▶ Doporučuji nainstalovat, bude se vám hodit i v dalších kurzech
- Jelikož již máme Android SDK a emulátory, tak je instalace snadná
- Nainstalujeme Apache Cordova do systému s právy administrátora

```
npm install -g cordova
```

- Vytvoříme nový projekt a přidáme Android platformu

```
cordova create MyApplication  
cd MyApplication  
cordova platform add android
```


Apache Cordova – instalace a spuštění

- Apache Cordova je dostupná přes správce balíčků pro Node.js `npm`
 - ▶ Doporučuji nainstalovat, bude se vám hodit i v dalších kurzech
- Jelikož již máme Android SDK a emulátory, tak je instalace snadná
- Nainstalujeme Apache Cordova do systému s právy administrátora

```
npm install -g cordova
```

- Vytvoříme nový projekt a přidáme Android platformu

```
cordova create MyApplication  
cd MyApplication  
cordova platform add android
```

- A můžeme aplikaci rovnou spustit

```
cordova run
```

Apache Cordova – struktura projektu

- `node_modules` – adresář se závislostmi (automaticky instalované)
- `platforms` – adresář pro nativní *obálky* – my máme Android

Apache Cordova – struktura projektu

- `node_modules` – adresář se závislostmi (automaticky instalované)
- `platforms` – adresář pro nativní *obálky* – my máme Android
- `www` – samotná webová aplikace
 - ▶ Prakticky nás může zajímat jen tato část
 - ▶ Zbytek nemusíme měnit

Apache Cordova – struktura projektu

- `node_modules` – adresář se závislostmi (automaticky instalované)
- `platforms` – adresář pro nativní *obálky* – my máme Android
- `www` – samotná webová aplikace
 - ▶ Prakticky nás může zajímat jen tato část
 - ▶ Zbytek nemusíme měnit
- `plugins` – instalované pluginy. Např.:
`cordova plugin add cordova-plugin-camera`
- `package.json` – konfigurace platformem, pluginů, závislostí
- `config.xml` – nastavení názvů, autorů, ... pro jednotlivé platformy
 - ▶ https://cordova.apache.org/docs/en/latest/config_ref/

Apache Cordova – vzhled aplikací

- Umíme tedy vytvořit mobilní aplikaci, ve které budeme provozovat webovou
- Pokud nebude webová aplikace „vypadat jako mobilní“ bude to k ničemu

Apache Cordova – vzhled aplikací

- Umíme tedy vytvořit mobilní aplikaci, ve které budeme provozovat webovou
- Pokud nebude webová aplikace „vypadat jako mobilní“ bude to k ničemu
- Proč bychom nahrazovali prohlížeč nějakou *obálkou*
- Pokud by měl uživatel stejný *zážitek* mohli jsme na webu zůstat

Apache Cordova – vzhled aplikací

- Umíme tedy vytvořit mobilní aplikaci, ve které budeme provozovat webovou
- Pokud nebude webová aplikace „vypadat jako mobilní“ bude to k ničemu
- Proč bychom nahrazovali prohlížeč nějakou *obálkou*
- Pokud by měl uživatel stejný *zážitek* mohli jsme na webu zůstat
- Idea: Udělejme webovou aplikaci, která bude vypadat jako mobilní
 - ▶ Vlastní design tlačítek, sliderů, menu, ...

Apache Cordova – vzhled aplikací

- Umíme tedy vytvořit mobilní aplikaci, ve které budeme provozovat webovou
- Pokud nebude webová aplikace „vypadat jako mobilní“ bude to k ničemu
- Proč bychom nahrazovali prohlížeč nějakou *obálkou*
- Pokud by měl uživatel stejný *zážitek* mohli jsme na webu zůstat
- Idea: Udělejme webovou aplikaci, která bude vypadat jako mobilní
 - ▶ Vlastní design tlačítek, sliderů, menu, ...
- Ano, je to cesta, ale proč znovu vynalézat kolo

Apache Cordova – vzhled aplikací

- Umíme tedy vytvořit mobilní aplikaci, ve které budeme provozovat webovou
- Pokud nebude webová aplikace „vypadat jako mobilní“ bude to k ničemu
- Proč bychom nahrazovali prohlížeč nějakou *obálkou*
- Pokud by měl uživatel stejný *zážitek* mohli jsme na webu zůstat
- Idea: Udělejme webovou aplikaci, která bude vypadat jako mobilní
 - ▶ Vlastní design tlačítek, sliderů, menu, ...
- Ano, je to cesta, ale proč znovu vynalézat kolo
- ⇒ Mobilní webové frameworky

Ionic

- Mobilní webový framework – tvorba webových aplikací cílených na mobilní zařízení
- <https://ionicframework.com/>
- Velké množství hotových UI komponent
- Možnost integrace do aplikací s frameworky Angular, React, Vue.js
- Lze provozovat v Apache Cordova nebo samostatně

Ionic – instalace

- Ionic je opět distribuován pomocí npm
- Nainstalujeme:

```
npm install -g ionic
```

Ionic – instalace

- Ionic je opět distribuován pomocí `npm`
- Nainstalujeme:
`npm install -g ionic`
- Vytvoříme nový projekt se šablonou se záložkami
`ionic start MyApp tabs`
- Zvolíme zamýšlený framework (Angular, React, Vue.js)

Ionic – instalace

- Ionic je opět distribuován pomocí `npm`
- Nainstalujeme:

```
npm install -g ionic
```
- Vytvoříme nový projekt se šablonou se záložkami

```
ionic start MyApp tabs
```
- Zvolíme zamýšlený framework (Angular, React, Vue.js)
- A spustíme

```
cd MyApp  
ionic serve
```
- Otevře se webový prohlížeč s naší aplikací

Další mobilní webové frameworky

- <https://jquerymobile.com/> – Odnož JQuery optimalizované pro dotekové zařízení
 - ▶ Není tolik „mobilní“ spíše mezikrok, „dotykový web“

Další mobilní webové frameworky

- <https://jquerymobile.com/> – Odnož JQuery optimalizované pro dotekové zařízení
 - ▶ Není tolik „mobilní“ spíše mezikrok, „dotykový web“
- <http://mobileangularui.com/> – Bootstrap + Angular
 - ▶ Dobré pokud znáte obojí

Další mobilní webové frameworky

- <https://jquerymobile.com/> – Odnož JQuery optimalizované pro dotekové zařízení
 - ▶ Není tolik „mobilní“ spíše mezikrok, „dotykový web“
- <http://mobileangularui.com/> – Bootstrap + Angular
 - ▶ Dobré pokud znáte obojí
- <https://onsen.io/> – podpora Angular, React, Vue.js, JQuery
 - ▶ Bývá označován za nejpropracovanější

Další mobilní webové frameworky

- <https://jquerymobile.com/> – Odnož JQuery optimalizované pro dotekové zařízení
 - ▶ Není tolik „mobilní“ spíše mezikrok, „dotykový web“
- <http://mobileangularui.com/> – Bootstrap + Angular
 - ▶ Dobré pokud znáte obojí
- <https://onsen.io/> – podpora Angular, React, Vue.js, JQuery
 - ▶ Bývá označován za nejpropracovanější
- A spousty dalších

Xamarin

- Open-source platforma umožňující vyvíjet multiplatformní (Android, iOS, tvOS, watchOS, macOS, Windows) nejen mobilní aplikace v C#
- Obrovská mašinérie nástrojů, knihoven, kompilátorů, . . .
- Dříve populární, dnes na ústupu (i MS přechází na React Native)
- Velká část kódu je totožná pro všechny platformy
- Pro specifické platformy se musí upravit „jen drobnosti“
- Vývoj je docela náročný na HW, na učebnách nejspíš není nainstalováno

Xamarin – instalace

- Potřeba modifikovat instalaci Visual Studia, doporučuji nainstalovat nejnovější (2022)
- Spustit *Visual Studio Installer* a přidat položku *Mobile development with .NET*

Modifying — Visual Studio Enterprise 2019 — 16.9.0

Workloads

Individual components

Language packs

Installation locations



Python development

Editing, debugging, interactive development and source control for Python.



Node.js development

Build scalable network applications using Node.js, an asynchronous event-driven JavaScript runtime.



Desktop & Mobile (5)



.NET desktop development

Build WPF, Windows Forms, and console applications using C#, Visual Basic, and F# with .NET Core and .NET Fr...



Desktop development with C++

Build modern C++ apps for Windows using tools of your choice, including MSVC, Clang, CMake, or MSBuild.



Universal Windows Platform development

Create applications for the Universal Windows Platform with C#, VB, or optionally C++.



Mobile development with .NET

Build cross-platform applications for iOS, Android or Windows using Xamarin.



Mobile development with C++

Build cross-platform applications for iOS, Android or Windows using C++.



Xamarin – sample project

- Vytvoříme nový projekt – Zvolíme jazyk C#, platformu Android
- K dispozici máme několik šablon – mobilní Android aplikace, multiplatformní aplikace, pro hodinky, ...
- Zvolíme *Mobile App (Xamarin.Forms)* – šablona pro iOS i Android
- K dispozici několik šablon
 - ▶ *flyout* – aplikace s postranním menu, na malých displejích se schová
<https://docs.microsoft.com/en-us/xamarin/xamarin-forms/app-fundamentals/shell/flyout>
 - ▶ *tabbed* – aplikace se záložkami, přepínání např. dole <https://docs.microsoft.com/en-us/xamarin/xamarin-forms/app-fundamentals/shell/tabs>
 - ▶ *blank* – prázdná šablona s jednou obrazovkou
- Zvolíme *tabbed*, vytvořená šablona obsahuje `GettingStarted.txt` s hezkým popisem, co dál

Progresivní webové aplikace (PWA)

Progresivní webové aplikace (PWA)

- Proč obalovat webovou stránku nějakou obálkou
- Pojďme udělat webovou aplikaci pořádně pro všechna zařízení

Progresivní webové aplikace (PWA)

- Proč obalovat webovou stránku nějakou obálkou
- Pojďme udělat webovou aplikaci pořádně pro všechna zařízení
- PWA = Webové aplikace schopné fungovat offline
 - ▶ Nezávislé na zařízení (responsivní)
 - ▶ Nezávislé na OS (multiplatformní)
 - ▶ Nemusí se instalovat přes Play/Store/...

Progresivní webové aplikace (PWA)

- Proč obalovat webovou stránku nějakou obálkou
- Pojďme udělat webovou aplikaci pořádně pro všechna zařízení
- PWA = Webové aplikace schopné fungovat offline
 - ▶ Nezávislé na zařízení (responsivní)
 - ▶ Nezávislé na OS (multiplatformní)
 - ▶ Nemusí se instalovat přes Play/Store/...
- Vytvoříme webovou stránku dostupnou přes URL
- Doplníme o *manifest* s informacemi o aplikaci

Progresivní webové aplikace (PWA)

- Proč obalovat webovou stránku nějakou obálkou
- Pojďme udělat webovou aplikaci pořádně pro všechna zařízení
- PWA = Webové aplikace schopné fungovat offline
 - ▶ Nezávislé na zařízení (responsivní)
 - ▶ Nezávislé na OS (multiplatformní)
 - ▶ Nemusí se instalovat přes Play/Store/...
- Vytvoříme webovou stránku dostupnou přes URL
- Doplníme o *manifest* s informacemi o aplikaci
- Ukázka: `https://martin-trnecka.github.io/pwa/`
 - ▶ Možné nainstalovat – „dát na plochu“

Závěr k multiplatformnímu vývoji

- Frameworků pro multiplatformní vývoj mobilních aplikací je celá řada
 - ▶ Velmi rozličné technologie/přístupy
- Dnes pouze ukázka několika základních, vysvětlení principů

Závěr k multiplatformnímu vývoji

- Frameworků pro multiplatformní vývoj mobilních aplikací je celá řada
 - ▶ Velmi rozličné technologie/přístupy
- Dnes pouze ukázka několika základních, vysvětlení principů
- Doporučuji ještě prostudovat:
 - ▶ **React Native** <https://reactnative.dev/>
 - ▶ **Flutter** <https://flutter.dev/>
 - ▶ **NativeScript** <https://nativescript.org/>
 - ▶ **Herní engine:** <https://solar2d.com/>, <https://unity.com/>

Závěr k multiplatformnímu vývoji

- Frameworků pro multiplatformní vývoj mobilních aplikací je celá řada
 - ▶ Velmi rozličné technologie/přístupy
- Dnes pouze ukázka několika základních, vysvětlení principů
- Doporučuji ještě prostudovat:
 - ▶ **React Native** <https://reactnative.dev/>
 - ▶ **Flutter** <https://flutter.dev/>
 - ▶ **NativeScript** <https://nativescript.org/>
 - ▶ **Herní engine:** <https://solar2d.com/>, <https://unity.com/>
- Frameworky nejsou všespásné, vždy musíme brát ohled na specifika platformem
 - ▶ Paměťová, procesorová, energetická náročnost

Závěr k multiplatformnímu vývoji

- Frameworků pro multiplatformní vývoj mobilních aplikací je celá řada
 - ▶ Velmi rozličné technologie/přístupy
- Dnes pouze ukázka několika základních, vysvětlení principů
- Doporučuji ještě prostudovat:
 - ▶ React Native <https://reactnative.dev/>
 - ▶ Flutter <https://flutter.dev/>
 - ▶ NativeScript <https://nativescript.org/>
 - ▶ Herní engine: <https://solar2d.com/>, <https://unity.com/>
- Frameworky nejsou všespásné, vždy musíme brát ohled na specifika platformem
 - ▶ Paměťová, procesorová, energetická náročnost
- Jedno UI pro všechny platformy není cesta

- 1 Vyberte si multiplatformní framework (i z nejmenovaných)
- 2 Vytvořte v něm aplikaci běžící na Androidu s funkcionalitou podobnou s příkladem s poznámkami
 - ▶ Zobrazí seznam uložených poznámek
 - ▶ Umožní přidat novou poznámku
 - ▶ Editovat/smazat stávající
- 3 Poznámky budou (nějak) uloženy v persistentní databázi (přežijí restart aplikace)

Příští hodina

- Příští hodinu proběhne prezentace vašich projektů
- Účast je povinná (pro ty, kteří chtějí zápočet)
- Prezentace potrvá max. 5 minut
 - ▶ Ukážete běžící aplikaci
 - ▶ Představíte její architekturu
 - ▶ V kódu ukážete zajímavé části
- Připravte si to ve spustitelném stavu (ať zbytečně nečekáme)
- Prezentace nejlépe z vlastního zařízení (připojené na projektor – HDMI)
- Pořadí bude losováno náhodně