

## **Elektronický kalendář**

Projekt ITU, 2019/2020

*Autor:* Monika Rosinská (xrosin00)

*Další členové týmu:* Kryštof Škrdlík (xskrdl01), Petr Volf (xvolfp00)

Datum odevzdání: 08-12-2019

## Analýza zadání

Zadáním projektu je navrhnout aplikaci, která umožní vykreslit kalendář na plochu či do aktuálního obrázku na ploše. Kalendář má podporovat různé typy zobrazení (denní, týdenní, měsíční), má zvýrazňovat aktuální den, týden nebo měsíc a má vypsát číslo týdne dle kalendáře. Aplikace by měla být připravena na případné zobrazování svátků.

Aplikace, které vykreslí kalendář na plochu, jsou dostupné především na mobilních telefonech, proto jsme se rozhodli pro vytvoření podobné aplikace pro počítače, kde je menší konkurence a bude snazší na trhu prorazit.

Jednoduchých kalendářů existuje velké množství, ale zároveň velmi jednoduchých kalendářů je malé množství. Nabízí se tedy vytvořit aplikaci pro seniory, dle mého názoru by o něj nebyl zájem (což se potvrdilo i při průzkumu), jelikož senioři raději používají papírový kalendář, jelikož často nemají počítače a chytré telefony, či je neumí dobře ovládat. Z tohoto důvodu jsme se rozhodli vydat se opačným směrem, a vytvořit kalendář vhodný pro uživatele, kteří budou požadovat komplexní aplikaci, obsahující složitější funkce.

## Existující řešení

Osobně jsem vyzkoušela několik existujících aplikací na telefonu – kalendářů, pracovala jsem s nimi a sepsala, co mi při práci s nimi vyhovovalo či nikoli (konkrétně Bussines Calendar, Google Kalendář, DigiCal, Calendar+ a defaultní kalendář Androidu). Na počítačových systémech takových aplikací existuje velmi málo a nejsou interaktivní, proto jsem neměla možnost žádnou vyzkoušet.

Mezi věci, které mi práci s kalendářem vyhovovaly, patří zejména:

- přidání události jedním kliknutím
- možnost upravit událost (kromě běžných nastavení jako datum, čas, lokace i upozornění na danou dobu před událostí, barvu zobrazení události v kalendáři nebo možnost nastavit pravidelné opakování události)
- možnost vypnout/zapnout české státní svátky nebo jmeniny
- vyhledávání událostí podle klíčových slov
- návrat na aktuální den pomocí jednoho kliknutí

Mezi věci, které mi práci s kalendářem znesnadňovaly či znepříjemňovaly, patří:

- okno s vytvořením/úpravou události se otevírá vždy "přilepené" na okénku daného dne a vždy na jiné straně okénka, pohybuje se tedy po obrazovce a nezůstává na stejném místě
- pod kalendářem je ještě lišta se samostatnými dny (ovládání je příliš citlivé a způsobuje nechtěné posouvání v kalendáři)
- poměrně málo výrazný dnešní den (hůře se orientuje v kalendáři a v plánu na dnešní a budoucí dny)
- neintuitivní rozmístění možností nastavení (nastavení bylo na několika místech, bylo obtížné najít kde změnit přesně, co potřebuji)
- chybí jakákoli změna barvy pozadí (tzv. dark mode – ve večerních hodinách je pro mne bílá barva příliš jasná)

## Reální uživatelé

Sledováním reálných uživatelů jsem získala informace o nejčastěji používaných funkcích aplikací, tedy i informace o tom, které funkce zefektivnit, umožnit k nim rychlý a snadný přístup, nebo jaké funkce naopak zjednodušit.

Mezi nejčastější úkony patří přidávání událostí. Vytvoření nové události by tedy mělo být co nejjednodušší a nejintuitivnější – např. tlačítko „+“ v rohu obrazovky nebo stisknutí a podržení daného dne/dané hodiny v kalendáři.

Dále je častou operací i úprava události – nejjednodušším řešením může být opakované kliknutí na událost.

Jednou z nejméně intuitivních funkcí byla u velkého množství aplikací možnost zobrazit informace o události, kdy se otevřelo okno podobné úpravě události. Většině uživatelů tato funkce vadila a stačil by jim obyčejný náhled, kdy a kde se událost koná.

Funkce, která radě uživatelů chyběla, byla možnost vyhledat událost podle názvu/klíčového slova.

Vypozorovala jsem, že uživatelé se lépe vyznali v kalendáři, který události, které již proběhly, zvýrazňoval méně, než nadcházející události (tedy barva proběhlých událostí byla méně výrazná a sytá). Uživatelé se ale sami nezmínili, že by jim tato funkce u jiných kalendářů chyběla.

Ovládání aplikace pomocí „scrollování“, posouváním nahoru-dolů a do boku by mělo být omezené – příliš možností takového ovládání práci s aplikací znesnadňuje a zvláště nezkušeným lidem a lidem, kteří nepracují s počítačem/mobilním telefonem příliš často, připadá takové ovládání neintuitivní a mohou tyto operace provést neúmyslně.

Vyjma samotného sledování uživatelů probíhal i rozhovor, který vedl k důležitým bodům, zejména:

- Využíváte raději papírový, či elektronický kalendář?
  - Pokud by existovala pro vás vhodná aplikace, používali byste ji místo Vámi oblíbeného papírového kalendáře? – pro uživatele, kteří dávají přednost papírovému kalendáři, zejména senioři
  - Kdyby se na trhu objevila nová aplikace, vyzkoušeli byste ji, nebo jste zcela spokojeni s aktuální a nevyhledáváte nová řešení?
- Co byste od aplikace „kalendář“ očekávali? Jaké jsou podle Vás důležité funkce a prvky?
- Co by Vás přimělo vyzkoušet novou aplikaci, i když již nějakou používáte?
- Podle čeho se rozhodujete, která aplikace je pro Vás vhodná a zda ji budete používat?
- Co Vás odradí od instalování/používání nové aplikace?

## Potřeby uživatele

Z předchozích informací tedy vyplývá, že uživatel potřebuje, aby byl kalendář přehledný, aby do něj mohl snadno přidávat události a upravovat je. Ovládání by mělo být intuitivní a co nejvíce podobné práci s běžným, papírovým kalendářem (metafora s existujícím objektem v reálném světě – např. přepínání jednotlivých dnů/měsíců se ovládá podobně, jako bychom otáčeli stránky u reálného kalendáře).

## Persona typického uživatele



**Tomáš**, 38 let, IT analytik a programátor, vysokoškolské vzdělání, ženatý, 2 děti. Počítač a mobilní telefon používá jak v práci, tak i v osobním životě. Je pracovitý, vede aktivní život, ve volném čase se věnuje rodině a sportu. Aktivně používá aplikace, především e-mail pro komunikaci s kolegy a nadřízenými, aplikace na plánování schůzek a termínů, ale i rodinných záležitostí. Online tráví 14 hodin denně.

## Analýza úkolu

Aplikace podobné nám zadané jsou poměrně běžné v mobilních zařízeních, na počítačích takové aplikace běžné nejsou, většinou se jedná o kalendáře v prohlížeči.

Vykreslení kalendáře na plochu počítače by bylo třeba provést podobně, jak se na telefonech používají widgety – je potřeba zachovat interaktivitu, aby uživatel nemusel při zadávání nebo úpravě události aplikaci zapínat.

## Dekompozice úkolu

Uživatel by se nemusel při používání aplikace „registrovat“ – přidávat e-mailovou adresu, pokud by neměl zájem využívat sdílení kalendáře s ostatními lidmi/skupinami. Mnoho lidí si cení vlastní soukromí, a mohli by tuto možnost uvítat. I když si myslím, že mnoho lidí bude chtít kalendář spíše sdílet, chceme uživatelům dát i tuto možnost.

Vložení kalendáře na plochu by mělo být jednoduché a intuitivní, ale na samotném provedení se s týmem domluvíme, až jej budeme implementovat. Jelikož to je jedna z hlavních funkcí aplikace, tato možnost se objeví jako jedna z prvních a bude uživateli pomáhat si kalendář nastavit.

Ostatní ovládání, jako je přidání nové události, upravení již existující a zobrazení podrobností o události by měla být intuitivní a snadno přístupná, bude tedy zobrazena přímo ve vykresleném okně kalendáře.

Další nastavení jako barvy, vypnutí svátků či jmenin, správa skupin či návrat na dnešní den nejsou tak často potřeba, a tak budou přístupné přes tlačítko nastavení.

Pohybování v kalendáři by mělo evokovat práci s papírovým kalendářem (jak již bylo řečeno).

Kalendář a jeho užívání lze rozložit na následující procesy/úkony (nejdůležitější):

- vykreslení kalendáře na plochu
  - zvolení oblasti pro vložení kalendáře
  - vložení kalendáře
  - popř. posunutí oblasti
- přepínání zobrazení v kalendáři (den/týden/měsíc/rok)
  - zvolení tlačítka na změnu zobrazení
  - zvolení požadovaného zobrazení
- přidání nové události
  - zvolení tlačítka na přidání události
  - nastavení názvu, data, času, lokace, upozornění, opakování, popisu, barvy události
  - uložení a navrácení na zobrazení kalendáře
- upravení již existující události
  - zvolení tlačítka upravení události

- změna nastavení události
  - uložení a navrácení na zobrazení kalendáře
- zobrazení dnešních událostí
  - kliknutí na dnešní den

# Návrh – GUI, backend

## Práce s daty

O uživateli nepotřebujeme sdělit žádné soukromé informace. Jedinou výjimkou je e-mailová adresa v případě, že chce uživatel kalendář s někým sdílet, popřípadě účet kalendáře od Googlu, pokud si jej uživatel přeje propojit.

Uživatel bude především vytvářet nové události a upravovat již existující, tedy pracuje s daty, které se k událostem vztahují. Již existující události se uživateli zobrazují automaticky v rámci jednoho měsíce (popř. týdne/dne/roku). Nové události uživatel přidává pomocí tlačítka "+", kdy se mu zobrazí okno s informacemi o události. Tyto informace se poté uloží (při zvolení možnosti "uložit").

Mezi data, která uživatel ukládá, patří především název, datum a čas události. Může také vyplnit informace o typu události (pracovní / rodinná / ...), kde se událost odehrává, zda chce na událost upozornit a kdy, nebo je možno nastavit barvu zobrazení události v kalendáři nebo vložit poznámku. Informace uživatel buď vypisuje, nebo volí z nabízených možností.

Podobně se události i upravují, je tedy potřeba daná data upravit i v datové části aplikace.

Dále je potřeba uchovávat informace o skupinách (např. práce, rodina, ...), jako názvy, lidé kteří do této skupiny patří, barvu zobrazení daného typu událostí, nebo jak se dané skupině zobrazují události ostatních skupin (ty mohou být skryty, mohou se zobrazovat jen jako "busy" nebo s jiným popisem, nebo mohou být zobrazeny se všemi detaily).

Kromě událostí je třeba uživateli zobrazovat samotný kalendář, a to dle nastavení – kalendář se může zobrazovat jako měsíční, týdenní, denní nebo i roční. Musíme být tedy schopni vykreslit kalendář současného roku (měsíce, dny, datum dne), z toho vychází kalendář měsíční a týdenní, a pro denní kalendář je třeba vykreslit jednotlivé hodiny dne a do nich zanést uložené události.

Data se uchovávají na disku, v případě sdílení i na serveru. Data jsou uložena ve formátu JSON a jsou rozparsovány do objektů. Data na disku nejsou sdílena mezi jednotlivými uživateli počítače, mohou tak mít každý vlastní kalendář, aniž by se jim zobrazovali události druhého uživatele. Spolu je mohou sdílet jako běžní uživatelé.

## Návrh GUI



## Vzhled měsíčního zobrazení kalendáře:

ŘÍJEN 2019						
Po	Út	St	Čt	Pá	So	Ne
30	1	2	3	4	5	6
7	8 Kino	9	10	11	12	13
14	15	16	17	18	19	20
21	22	23 IFJ-přl	24	25	26	27 ITU-od
28	29	30	31	1	2	3
4	5	6	7	8	9	10

Připadalo mi důležité navrhnout měsíční zobrazení kalendáře, protože jej lidé používají jako jedno z nejčastějších (ale poměrně často se používá i týdenní a denní zobrazení, návrh těchto dvou zobrazení má na starost ostatní členové týmu).

Vždy by měl mít uživatel přehled, jaký měsíc má právě zobrazen, proto jsem na horní „lištu“ vypsala název měsíce. Dále je podobně umístěno tlačítko „+“, které umožňuje rychlé přidání nové události. Vedle něj je tlačítko menu, kde se rozvine nabídka různých nastavení, jako např. správa skupin, zobrazení kalendáře (přepínání na týdenní / denní / roční), nastavení barev či další nastavení.

Kalendář jsem navrhla tak, aby měl sedm řádků z toho důvodu, že některé měsíce by se na šesti řádcích nevykreslily a nebylo by vhodné, aby kalendář kvůli těmto výjimkám měnil svou velikost či rozložení, i když by se to dělo jen občas. V okénkách, která patří do jiného měsíce, je datum méně výrazné než u dnů z tohoto měsíce, aby uživatele nerozptylovaly, ale aby si i zachovával pojem o následujících dnech a událostech (které jsou ale také méně výrazné než události tohoto měsíce).

Aktuální den je zobrazen jinou, výraznější barvou, než zbytek dní, aby se v kalendáři lépe orientovalo. Ze stejného důvodu jsou již proběhlé události méně výrazné než ty nadcházející. K tomuto rozhodnutí mě vedly výsledky z mého průzkumu.

Zobrazené události jsou znázorněny barevným proužkem a názvem (pokud se tam název neveleze, je z něj zobrazeno jen tolik, kolik textu se tam vleze), za ním případně čas konání, pokud zbývá volný prostor. Více informací o události se zobrazí, klikne-li na ni uživatel.

Má-li uživatel více událostí v jeden den, zobrazují se pod sebe, případně se (ve výjimečných případech) zobrazí nabídka na rozvinutí více událostí toho dne.

### Návrh přidávání události:

Na'zev události

Datum 22. 10. 2019

Čas 18 : 24 ☒ All day

Kategorie v

Popis...

Uživatel by měl mít možnost uložit si událost tak, aby díky kalendáři věděl všechny důležité informace, a zároveň aby se v událostech dobře orientoval. Proto jsem zvolila poměrně velké zadávací pole pro název události, následuje zvolení data a času, událost může být nastavena jako All-day. Dále uživatel zadá, do jaké kategorie událost patří a jak se má tedy případně zobrazovat členům, se kterými kalendář sdílí (např. pracovní, rodinný, přátelé, ...).

Dále může uživatel přidat popis, kde se událost koná, zda chce být na událost upozorněn, přidat další poznámky k události, či si nastavit pravidelné opakování (každý den, týden, měsíc, ...). Tyto informace jsou méně důležité, proto se zobrazí až po posunutí lišty. Tato funkce umožní vykreslit okno úpravy události menší, a nebude tedy zakrývat velký prostor, který by uživatel mohl chtít využít jinak.

Událost se poté uloží tlačítkem ✓ a přidá se do kalendáře. Vytváření události se ruší křížkem v levém horním rohu.

Podobně bude vypadat i upravování události.

### Testování maket

Jak návrh, tak i makety jsem vytvářela pomocí výsledků výzkumu, kdy jsem pozorovala reálné uživatele používat podobné aplikace, proto nebylo těžké navrhnout takové prostředí, ve kterém se uživatel snadno orientuje a které mu vyhovuje.



Zvolila jsem podobné ovládání, na jaké jsou uživatelé zvyklí z již existujících řešení. Současně jsem navrhovala ovládání pomocí metafory k papírovým kalendářům, jak již bylo zmíněno.

Uživatelé neměli problém „aplikaci“ ovládat, rychle se s ní naučili pracovat a poměrně dobře se v ní vyznali.

Uživatelé ale často podotýkali, že by se aplikace mohla používat jen pomocí klávesnice (základní úkony – jako přidání nové události apod.). Pro skupinu lidí, na kterou jsme se zaměřili, je rychlá a snadná práce poměrně důležitá a my jim v tomto chceme vyhovět. Se samotným návrhem aplikace a rozložením možností byli ale uživatelé spokojeni a nenašli jsme důvod naše návrhy nějak výrazně měnit.

# Architektura aplikace a návrh testování

## Výběr technologií

Aplikaci budeme vytvářet pro počítače s operačním systémem Windows, proto jsme se z několika vhodných technologií rozhodli použít WPF (Windows Presentation Foundation), který využívá XAML, C# a je součástí .NET. Zvolili jsme WPF, protože se snaží poskytnout co nejrozšiřitelnější objektový model, vývojář není vázán na pevnou sadu komponent s pevným vzhledem, můžete si plně graficky upravit jakoukoliv část jakékoliv komponenty nebo vytvořit komponentu úplně novou. Umožňuje jednoduše propojit vzhled, funkčnost a data.

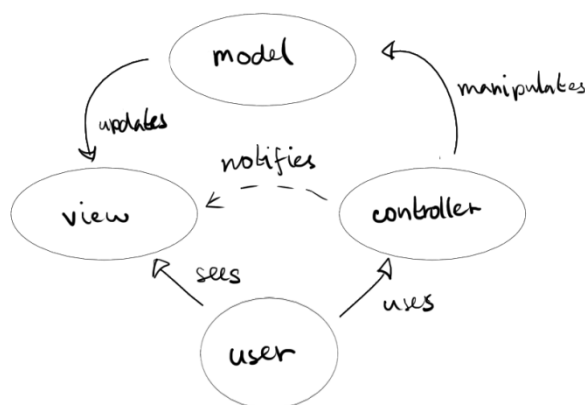
Velmi nám v rozhodování jakou technologii použít pomohla zkušenost ze cvičení, kde jsme zjistili, že všem členům týmu WPF vyhovuje nejvíce, a pro naši platformu je nejvíce vyhovující. Naučit se pracovat s touto technologií je velmi jednoduché, zvláště protože jsme všichni seznámeni s Visual Studiem, s kterým je WPF propojeno, a protože při samotném psaní kódu umožňuje okamžitě graficky zobrazit výsledný vzhled okna.

## Architektura aplikace

Jako architekturu jsme zvolili Model-view-controller (MVC). MVC rozděluje aplikaci na datový model aplikace, uživatelské rozhraní a řídicí logiku. Tyto tři komponenty jsou nezávislé, upravení libovolné z nich nemá vliv na ostatní. Model je datová struktura, která uchovává informace a data, View je zobrazením modelu a dalších prvků uživatelských rozhraní a Controller spojuje model a view a umožňuje jejich komunikaci.

Controller může upravovat data modelu, a view může zobrazovat data z modelu. Model může nepřímo informovat view o změně, ale nemůže samotný view měnit.

Každé zobrazení kalendáře (měsíční, týdenní, denní) má vlastní třídu. Jedna třída se stará o view, o komunikaci mezi view a controllerem. Controller je reprezentován objektem v paměti, data bere buď z disku, nebo z internetu (kalendářový API).



## Co chceme zjistit pomocí testování?

Pomocí testování chceme ověřit, zda je aplikace intuitivní a zda lidé dokáží rychle a efektivně provádět důležité úkony. Je potřeba zjistit, zda se lidé v kalendáři dobře orientují a není těžké se s ním naučit pracovat.

## Testovací protokol

Testování bylo rozděleno do 3 fází.

V první fázi je uživateli předán seznam úkolů, které má vyplnit. V této fázi jsme měřili, jak dlouho uživatelům trvá jednotlivé úkoly splnit, a zároveň jsme pozorovali případná špatná „rozhodnutí“. V druhé fázi jsme uživatele požádali, aby aplikaci používali jako běžný kalendář, například přepsat do ní svůj aktuální kalendář, a pozorovali jsme jejich počínání. V třetí fázi jsme se uživatelů ptali, jak se jim s aplikací pracovalo, zda jim prostředí vyhovovalo atp.

Měřili jsme především dobu, kterou uživatelům trvalo splnit jednotlivé úkoly, a kterými úkony při „běžném“ užívání trávili nejvíce času, včetně informace, jaké úkony dělali nejčastěji. Díky těmto hodnotám jsme získali informace o nejvíce užívaných a nejdéle trvajících funkcích a můžeme díky nim aplikaci lépe optimalizovat, aby uživatel trvaly co nejkratší dobu.

Zaznamenávali jsme také, jak se uživatelé rozhodovali při plnění úkolů a případné odchylky od očekávaného postupu. Pomocí těchto zkušeností jsme mohli upravit umístění funkcí tak, aby jej uživatelé neměli problém nalézt.

Pro tento přístup k testování jsme se rozhodli, protože při první fázi vidíme, jak uživatel reaguje na novou aplikaci, a zda se v něm dobře orientují. Uživatelé by měli být schopni ovládat aplikaci bez návodu, což si ověříme právě první fází. Druhá fáze nám umožní sledovat, jak uživatelé aplikace reálně využívají a zda jsme si správně navrhli. Třetí fáze nám umožní získat zpětnou vazbu od uživatelů.

## Implementace a výsledky testování

Data pro back-end jsme nasimulovali, vložili náhodná data a uměle vygenerované jména atp. Back-end by ale po drobných úpravách mohl být alespoň minimálně funkční, alespoň pro samostatné použití kalendáře bez sdílení.

Provedli jsme testy s uživateli, které měly velmi dobré výsledky. Uživatelé pracovali jak samostatně, tak i s využitím think-loud protokolu, díky kterému jsme si ověřili, že jim ovládání aplikace připadá logické.

Výsledky se nijak nelišily od předchozích testů. Vše uživatelům připadalo logické a neměli problémy úkoly plnit, i think-loud protokol dopadl velmi dobře, kdy uživatelé přemýšleli přesně tak, jak jsme předpokládali a aplikace pro to byla navržena.

Na základě testů jsme provedli ještě drobné úpravy, a vytvořili finální podobu aplikace, kterou jsme ještě krátce otestovali na dvou uživateli.

## **Práce v týmu a nové zkušenosti**

Jelikož tento projekt nebyl mým prvním týmovým projektem, tak jsem měla již nějaké zkušenosti ohledně týmů, které se mi hodily.

V mém týmu se mi pracovalo dobře a téměř na všem jsme se dokázali shodnout. I přesto jsem se naučila trpělivosti, a porozumění názoru jiného člena týmu, neboť ne vždy bylo jednoduché dospět k všeobecné spokojenosti, ale myslím, že se nám podařilo nalézt rozumné kompromisy.

Opět se při práci v týmu osvědčilo zpočátku se zabývat obecnými věcmi – pro koho bude výsledný produkt určen a na jaké platformě jej budeme psát, jaké základní funkce bychom chtěli implementovat – a následně i méně podstatnými rozhodnutími – vzhled, další funkce, atp. Každý člen týmu dostal určité úkoly, které měl splnit, a vždy se tak stalo, což práci usnadňovalo. Velmi nám také pomohla technická zpráva, díky které jsme se drželi podstatných věcí a pracovali průběžně.

- 
- <sup>1</sup> <https://www.usability.gov/how-to-and-tools/methods/personas.html>  
<sup>2</sup> [https://en.wikipedia.org/wiki/Windows\\_Presentation\\_Foundation](https://en.wikipedia.org/wiki/Windows_Presentation_Foundation)  
<sup>3</sup> <https://www.zdrojak.cz/clanky/uvod-do-architektury-mvc/>

## Přílohy