

# Knižní databázový systém KDFit

Technická zpráva k projektu do předmětu ITU  
FIT VUT v Brně, 2020

## Název týmu

Tým xloren15

## Autoři

Jan Lorenc, xloren15

Dominik Kaláb, xkalab11

Vojtěch Staněk, xstane45

# Obsah

## 1. Zadání a organizace týmu

### 1.1 Cíl

### 1.2 Tým

### 1.3 Roadmapa

### 1.4 Rizika a opatření

## 2. Průzkum a zkušenosti

### 2.1 Existující řešení

[Knihy.ABZ.cz \(xstane45\)](#)

[Databáze portálu Knihy Dobrovský \(xstane45\)](#)

[KDB.cz \(xloren15\)](#)

[ČBDB \(xloren15\)](#)

[DatabazeKnih.cz \(xkalab11\)](#)

[Goodreads \(xkalab11\)](#)

### 2.2 Uživatelské potřeby

### 2.3 Shrnutí

## 3. Architektura řešení

### 3.1 Architektura systému

### 3.2 Architektura aplikace/í

### 3.3 Datový model

### 3.4 Vybrané technologie a implementace

#### 3.4.1 Backend

#### 3.4.2 Web

#### 3.4.3 Desktop

## 4. Návrh GUI - Webová aplikace KDFit [xstane45]

### 4.1 Požadavky na GUI

### 4.2 Makety

### 4.3 Pilotní test

#### 4.3.1 Testování členy týmu

#### 4.3.2 Testování uživateli

##### Uživatel 1

##### Uživatel 2

### 4.4 Vyhodnocení testu a revize návrhu

## 5. Návrh GUI – Rozložení a vybrané části webové a desktopové aplikace KDFit [xloren15]

### 5.1 Požadavky na GUI

### 5.2 Mockupy

#### 5.2.1 Web

#### 5.2.2 Desktop

### 5.3 Pilotní test

- [5.3.1 Web](#)
  - [5.3.2 Desktop](#)
  - [5.3.3 Společné](#)
  - [5.4 Vyhodnocení testu a revize návrhu](#)
- [6. Návrh GUI - Desktopová aplikace KDFit \[xkalab11\]](#)
  - [6.1 Požadavky na GUI](#)
    - [6.1.1 Požadavky na desktopovou aplikaci jako celek](#)
    - [6.1.2. Požadavky na tabulkové zobrazení záznamů](#)
    - [6.1.3 Požadavky na detaily záznamů](#)
  - [6.2 Makety](#)
  - [6.3 Pilotní test](#)
    - [6.3.1 Uživatel číslo 1](#)
    - [6.3.2 Uživatel 2](#)
  - [6.4 Vyhodnocení testu a revize návrhu](#)
- [7. Implementace GUI - Webová aplikace KDFit \[xstane45\]](#)
  - [7.1 Implementace](#)
  - [7.2 Použité nástroje a knihovny](#)
    - [DotVVM 2.0 na platformě ASP.NET 3.1](#)
    - [jQuery, Bootstrap 4.5, CSS3 a Font Awesome 4](#)
  - [7.3 Finální testování](#)
    - [7.3.1 Průběh testování](#)
  - [7.4 Vyhodnocení testu](#)
- [8. Implementace GUI - Rozložení a vybrané části webové a desktopové aplikace KDFit \[xloren15\]](#)
  - [8.1 Implementace](#)
  - [8.2 Použité nástroje a knihovny](#)
  - [8.3 Finální testování](#)
    - [8.3.1 Průběh testování](#)
  - [8.4 Vyhodnocení testu](#)
- [9. Implementace GUI - Desktopová aplikace KDFit \[xkalab11\]](#)
  - [9.1 Implementace](#)
  - [9.2 Použité nástroje a knihovny](#)
    - [9.2.1 Windows Presentation Foundation \(WPF\)](#)
    - [9.2.2 MessageBox](#)
  - [9.3 Finální testování](#)
    - [9.3.1 Uživatel číslo 1](#)
  - [9.4 Vyhodnocení testu](#)
- [10. Zprovoznění systému](#)
- [11. Závěr](#)

## Reference

# 1. Zadání a organizace týmu

## 1.1 Cíl

Cílem projektu je vytvořit databázový systém pro spravování knih a autorů, jejich prohlížení a hodnocení. První částí řešení bude webová aplikace pro uživatele, která umožní uživatelům prohlížet, hodnotit knihy a spravovat seznam oblíbených knih. Druhou částí řešení bude desktopová aplikace pro správce, která umožní knihy a autory spravovat. Inspirací budou již existující systémy s podobným účelem jako např. ČBDB či Databazeknih.cz. Aplikace bude určena pro čtenáře knih, kteří hledají inspiraci pro novou četbu, nebo se chtějí podělit o zkušenosti a názor na již přečtené knihy. Systém pomůže s průzkumem knižního světa vášnivým čtenářům, hledáním budoucí četby a správou jejich oblíbených knih.

## 1.2 Tým

xloren15 - backend, webová aplikace a výpomoc s desktopovou aplikací

xstane45 - webová aplikace

xkalab11 - desktopová aplikace

## 1.3 Roadmapa

25.10. - Dokončené wireframy a mockupy, získaný feedback od alespoň tří potenciálních uživatelů ke každé aplikaci. První 3 části TZ. Připravená seedovací data pro databázi.

1.11. - Individuální dokončení 4. části TZ. Dokončená implementace backendu, včetně testů.

8.11. - Průběžná schůzka k postupu implementace. Feedback od členů týmu.

23.11. - Dokončené implementace aplikací. Aplikace jsou funkční. Testujeme a opravujeme drobné nedostatky, příprava TZ na odevzdání.

6.12. - Týden s předstihem odevzdáváme projekt. Projekt je perfektně otestovaný a funkční.

## 1.4 Rizika a opatření

Nedodržení termínů - v případě nedodržení termínů máme rezervu 1 týden. Část závěrečného testování je také záměrně prodloužená pro případnou kompenzaci časového skluzu.

Výpadek člena týmu (nemoc, osobní či rodinné problémy, smrt) - v případě výpadku člena týmu počítá každý člen s možností nárůstu celkového objemu práce, kterou bude muset vypracovat. Časová rezerva bude také nápomocná.

Nedostatečné schopnosti člena týmu - v případě nedostatečných schopností člena týmu je vedoucí týmu, který je zkušený v používání námi zvolených technologií, ochoten zaškolit a vši svojí schopností pomoci členům týmu k vypracování jejich části projektu.

Ztráta důležitých souborů a dat - díky verzovacímu systému git a vzdálenému repozitáři na GitHubu by k tomuto dojít nemělo. Pokud by tento problém nastal, drží každý člen týmu lokální repozitář, ze kterého je možné obnovit důležité soubory.

Nefunkčnost dílčích částí projektu - každý člen je povinen průběžně testovat a dohlížet na čistotu kódu jeho části projektu. Do master větve repozitáře se nesmí dostat žádný závadný, nefunkční, či nedostatečně zdokumentovaný kód. Tohoto bude docíleno pomocí pull requestů řízených vedoucím týmu a rozpracovanými akceptačními podmínkami pro odevzdání kódu.

Nesouhlas členů týmu - v případě nesouhlasu členů týmu se každý člen týmu zavázal podřídit příkazu vedoucího týmu, který má ve všech oblastech tohoto projektu finální slovo.

## 2. Průzkum a zkušenosti

### 2.1 Existující řešení

V rámci návrhu dobrého uživatelského rozhraní proběhl průzkum, hodnocení a recenze již existujících řešení. Každý z následujících portálů/aplikací byl zkoumán jedním členem týmu.

#### Knihy.ABZ.cz (xstane45)

Kompaktní a nepříliš estetická nabídka v levé části portálu není pro uživatelskou orientaci nejjednodušší. Samotné prohlížení knih je zpracované hezky a uživatelsky přívětivě - položka seznamu nezabírá moc místa, zároveň je ale dostatečně velká aby byly přehledně vidět důležité údaje ještě před rozkliknutím detailu knihy. V detailu knihy lze kromě všech informací najít i ukázkou z dané knihy, což může mít zásadní vliv na případné rozhodování uživatele o tom, zda by si knihu přečetl či ne.

**Přínosy:** Přehledné a zároveň dostatečně informativní náhledy knih, detail knihy je velice podrobný, možnost mnoha různých vyhledávání a řazení knih a to v různých kategoriích - lze snadno najít tu pravou knihu. Možnost recenzí a hodnocení knih, ukázky z knih.

**Omezení:** Menu až příliš kompaktní - nepřehledné. Celkové barevné schéma portálu je "duté", používá podobné barvy pro všechno, takže kromě odkazů, názvů knih a jmen autorů text splývá s nabídkou menu a s pozadím (hůře čitelný, světle šedé písmo na bílém pozadí).

## Databáze portálu Knihy Dobrovský (xstane45)

Příjemné barevné schéma a přehledná vodorovná nabídka menu v horní části stránky zaručuje, že se uživatel neztratí a dostane se do kategorie, která ho zajímá. Kromě papírových knih lze v portálu prohlížet i audioknihy a e-knihy. Všechna tři média jsou dále rozdělena do několika kategorií, podle kterých se knihy řadí. Vyhledávání funguje spíše jen podle názvu knihy nebo autora, nelze vyhledávat např. podle žánru nebo jejich kombinací. Náhledy knih ve výsledcích vyhledávání nebo prohlížení kategorií jsou skromné, není tam vidět např. krátký popis, nicméně ostatní důležité údaje tam jsou. V detailu knihy přibylá kromě popisu knihy i nabídka dalších, podobných knih, nebo knih, které zákazníci kupují zároveň s právě prohlíženou.

**Přínosy:** Přehledné menu a barevné schéma, prohlížení i jiných médií než papírové knihy. Možnost hodnotit knihy. Inteligentní nabídka podobných knih (jak na základě žánru a autora, tak na základě uživatelské zkušenosti).

**Omezení:** Nelze ke knihám psát recenze, takže si je nelze ani přečíst. Animace při ovládání menu uživatele spíše omezují, než aby esteticky vylepšily vzhled stránky, navíc zpomalují méně výkonné přístroje/prohlížeče. Z náhledů knih ve výsledcích vyhledávání/prohlížení kategorií uživatel moc nezjistí.

## KDB.cz (xloren15)

Aplikace má poměrně bohaté menu, což dává potenciál pro různorodé použití. To by však uživatel i čekal, vzhledem k přední příčce návrhu webového prohlížeče. Více pozitiv ale už není vidět. Aktivní část webu zabírá pouze polovinu obrazovky. To má za následek přehnaně nahuštěný obsah. Z třívrstvé hlavičky by bylo záhodno něco přesunout do neexistujícího postranního menu. U registračního formuláře chybí potvrzovací tlačítko, nelze se tedy zaregistrovat, a proto člověk nemůže přidávat knihy či hodnocení. Aktuálně je navíc stránka naprosto nepoužitelná, vzhledem k absenci dat, přesto se jí však daří být extrémně pomalá. Prvky SPA jsou zde zcela absentní, podobně však i responzivní design.

## ČBDB (xloren15)

Nejrozšířenější knižní databáze v Česku a na Slovensku. Lze popsat jako knižní ekvivalent k čsfd. Na první pohled přehledná, intuitivní a neinvazivní hlavička webu umožňuje snadnou orientaci po webu. Knihy lze snadno vyhledávat a třídit podle mnoha parametrů, hodnotit je, ukládat si je do oblíbených, dokonce i nakupovat. Přihlášený uživatel je může i přidávat či editovat. Mimo samotné hodnocení se uživatel může rozhodovat o přečtení knihy i díky statistikám knih, kolik lidí si ji již přečetlo nebo právě čte apod. Nechybí ani fórum pro diskuzi mimo samotné hodnocení či knižní bazar.

**Přínosy:** Dobrovský datový sklad s benefitem toho, že uživatelé mohou přispívat. U knih jsou reference na obchody, kde lze knihu koupit. Možnost diskuze a existence fóra. Nechybí mnoho, aby mohla být aplikace použita jako sociální síť zaměřená na knihy.

**Omezení:** Žádné. Jsem dlouholetý uživatel a osobně aplikace umí vše, co od ní vyžaduji.

## DatabazeKnih.cz (xkalab11)

Jeden z nejpoužívanějších portálů v ČR. Velké množství reklam. Obsah portálu tvoří pouze cca polovinu zobrazení. Databáze má standardní funkcionality evidence autorů a knih. Knihy lze vyhledávat podle názvu, filtrovat podle kategorie, sérií, štítků a řadit podle několika předem definovaných pohledů. Zajímavé sekce této aplikace jsou “Novinky”, kde přibývají zprávy z knižního světa, e-knihy zdarma, kde se dají stáhnout zajímavé knihy a čtenářská výzva, kde lze najít knihy, které jsou zařazeny do některé ze zvláštních kategorií (např. “Knihy jejichž autor letos slaví kulatiny.”) Přihlášený uživatel může posílat zprávy jiným uživatelům, psát recenze na knihy, sledovat aktivitu oblíbených uživatelů, či dostávat upozornění.

**Přínosy:** Velká databáze děl, mnoho recenzí a velký počet aktivních uživatelů, upozornění a novinky. Unikátní a dobrá je sekce “e-knihy zdarma” kde lze stáhnout vskutku zajímavá díla. Pro někoho by mohla být přínosná možnost posílat zprávy, či sekce čtenářská výzva. Mně osobně přijdou ale tyto možnosti zbytečné.

**Omezení:** Obrovské množství reklam. UI je dost neintuitivní a design ne příliš hezký.

## Goodreads (xkalab11)

Anglický portál pro hodnocení knih. Na první pohled čistý, jednoduchý design bez zbytečností. Knihy je možné třídit podle kategorií. Jako přihlášený uživatel mohu dostávat doporučení, hodnotit knihy a sledovat jaké knihy čtou moji přátelé. Také zde lze vyhledávat citáty, či události této komunity. Z této stránky se také lze přímo prokliknout na odkaz k zakoupení knihy. Mimo hodnocení je u knihy také sekce Q&A kde mohou čtenáři pokládat otázky a odpovídat na otázky ke knize.

**Přínosy:** Q&A sekce a možnost vyhledávání citátů a akcí jsou zajímavé nápady. Kladně hodnotím absenci reklam.

**Omezení:** Portál je dost jednoduchý, neresponzivní a nenabízí toho moc nad míru standardních funkcionalit podobných portálů.

## 2.2 Uživatelské potřeby

Jak již bylo zmíněno, aplikace je určena pro čtenáře knih, kteří hledají inspiraci pro novou četbu, chtějí se podělit o zkušenosti a názor na již přečtené knihy, nebo zjistit právě takové názory ostatních čtenářů. Systém pomůže s průzkumem knižního světa vášnivým čtenářům, hledáním budoucí četby a správou jejich oblíbených knih.

Z rozhovoru s potenciálním uživatelem aplikace vyšlo najevo, že se jedná o zajímavý nápad a že by určitě takovou aplikaci minimálně vyzkoušel. Má zkušenosti s dvěma existujícími portály, které zřídka používá pro inspiraci (databazeknih.cz a ČBDB).

Jako přednosti vyzdvihl přehlednost a intuitivnost stránek a skvělé vyhledávání podle všemožných kritérií. Jako nevýhody vyzdvihl malé písmo a obrázky při prohlížení na mobilu a obrovskou míru reklam na databazeknih.cz.



Hodnocení hraje zásadní faktor při vybírání nové četby, recenze čte a i ty mají zásadní vliv na rozhodování. Upozorňuje však, že dobrý autor je mnohem větší ukazatel dobré četby, hodnocení a recenze rozhodují spíše u uživateli neznámých autorů. Účet ani na jednom portálu nemá, protože jej nepotřebuje. Knížky nehodnotí ani k nim nepíše recenze, protože “narozdíl od mladší generace nemá potřebu se ke všemu vyjadřovat”. Na druhou stranu je rád, že lidé svůj názor vyjadřují, protože mu to pomáhá ve výběru kvalitních knih.

## 2.3 Shrnutí

Díky průzkumu existujících řešení vznikla představa o novém řešení.

**Pro webovou aplikaci** je důležitá přehlednost uživatelského rozhraní, nezahlcenost stránky reklamou a možnost řadit knihy a autory podle různých kritérií.

Náhled knihy (např. ve výsledcích vyhledávání) by měl obsahovat dostatečně velký obrázek, název knihy, jméno autora/autorů, zařazení žánru/kategorie a malý úryvek popisu.

Seznam knih nebo autorů v kategoriích či výsledcích vyhledávání by neměl být přehlcený - není potřeba na stránku zobrazit co nejvíce knížek, ba naopak jednoduchý a přehledný. Při pohledu na detail knihy uživatelé ocení kromě delšího popisu i úryvek z knihy. Recenze a hodnocení jsou nedílnou součástí detailu knihy.

Na úvodní stránce je místo pro novinky z knižního světa, a to jak akce s ním spojené, tak např. jubileum autora, nové dílo známého autora apod.

Důležité je, aby byly webové stránky responzivní, protože v dnešní době mnoho uživatelů používá k prohlížení mobily, tablety a podobná zařízení - proto je určitě vhodné použít CSS framework Bootstrap.

Uživatel bude mít možnost označovat knihy jako oblíbené - takovýto seznam pak může sloužit jako tzv. “wishlist”, tj. knihy, které se uživatel chystá přečíst někdy v budoucnu, nebo jako označení opravdu oblíbených knížek, ke kterým se může uživatel vrátit a připomenout si zajímavý žánr nebo autora.

**Desktopová aplikace** je v mnoha ohledech podobná té webové - důležitá přehlednost, možnost řazení apod., kvůli rychlé a efektivní práci s databází - přidávání a upravování nových autorů, knih atd.

Není důležité implementovat funkce, ke kterým slouží webová aplikace. Desktopová aplikace slouží ke správě databáze autorů a knih, přidávání novinek, psaní popisů knih a životopisů autorů a jistě cenzuře nevhodných komentářů (urážlivé, robotem-generované). Zaměřením na tyto funkce unikátní pro desktopovou aplikaci a úplným vypuštěním funkcí, ke kterým slouží pouze webová aplikace (přihlašování uživatelů, seznamy oblíbených), dosáhneme větší přehlednosti.

## 3. Architektura řešení

### 3.1 Architektura systému

Celý systém je složen ze tří částí. První z nich je REST API služba implementující backend logiku, primárně tedy CRUD operace nad databází. Pomocí Swaggeru se ke službě generuje klient, díky kterému mohou jednotlivé aplikace velmi snadno a pohodlně posílat požadavky, což zjednodušuje a minimalizuje kód na frontendu potřebný ke komunikaci s backendem. Oběma aplikacím je tento zpřístupněn jako nuget balíček.

První aplikací je WPF desktopová aplikace, která slouží k administrativním účelům systému. Není nutné zde řešit přihlašování, neboť se předpokládá, že tuto aplikaci bude moci obdržet jenom admin. V této části systému uživatel může přidávat, mazat, upravovat autory knih, knihy, či provádět jistou cenzuru nad uživatelskými komentáři ke knihám, ne však upravovat, pouze mazat. Dále odtud lze psát novinky z knižního světa nebo přijímat návrhy či požadavky o uživateli na nové autory/knihy či právě upozornění na nějakou nadcházející knižní akci, která by stála za to se objevit v sekci "Novinky" pro ostatní uživatele.

Druhou aplikací je webový portál. Uživatel se může registrovat a přihlásit. Neregistrovanému uživateli je povoleno prohlížet novinky, knihy, autory a hodnocení, registrovanému pak i přidávat hodnocení, ukládat si knihy do oblíbených či psát feedback administrátorům. Tento feedback se může týkat například toho, jaké knihy ho zajímají, jestli nejdou přidat, nebo že ví o nějaké nadcházející knižní akci, o které si myslí, že je dostatečně zajímavá na to, aby ji admini dali do "Novinek" i pro ostatní uživatele.

### 3.2 Architektura aplikace/í

Služba pro backend jakožto REST API automaticky implementuje controller a model z MVC architektury. Kontrolerů a tedy i výsledných klientů je pro každou doménu jeden. Interně je služba postavena na návrhovém vzoru mediátor.

Pro webovou i desktopovou aplikaci je použita stejná architektura běžná pro platformu .NET a tedy MVVM (Model-View-ViewModel). Ve vrstvě Model se pouze volají klienti naší služby obalení o zpracování chyb a případnou úpravu přijatých dat pro potřeby View. ViewModel vrstva pak řeší logiku za samotným markup jazykem View vrstvy, ať už se jedná o html nebo xaml.

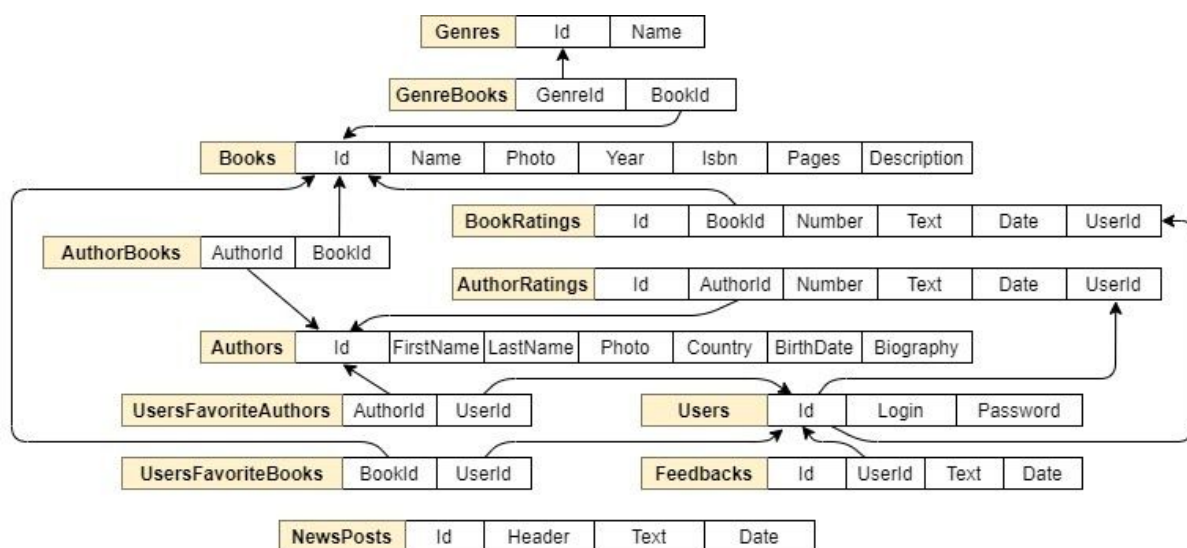
### 3.3 Datový model

Entity:

- Kniha - reprezentuje samotnou knihu v databázi
  - Může mít více autorů
  - Může spadat pod vícero žánrů
  - Neomezený počet uživatelů ji může napsat recenzi

- Neomezený počet uživatelů si ji může přidat do “oblíbených”
- Autor - reprezentuje autora knihy
  - Může napsat více knih
  - Neomezený počet uživatelů mu může napsat recenzi
  - Neomezený počet uživatelů si ho může přidat do “oblíbených”
- Žánr - je výhodné ho modelovat jako entitu a ne atribut, protože:
  - Kniha může mít více žánrů
  - Pod jeden žánr spadá vícero knih
- Recenze knihy - uživatel může ohodnotit knihu formou recenze
- Recenze autora - uživatel může ohodnotit autora formou recenze
- Uživatel - reprezentuje uživatele systému
  - Může si ukládat knihy/autory do “oblíbených”
  - Může psát knihám/autorům hodnocení
  - Může psát zpětné vazby administrátorům systému
- Zpětná vazba - uživatel může napsat administrátorům zpětnou vazbu / návrh na vylepšení / doporučení knihy / tip na knižní akci
- Článek - samostatná entity reprezentující článek do sekce “Novinky”

Následující diagram znázorňuje databázové tabulky:



Primárním úkolem API je provádění CRUD operací nad těmito entitami.

## 3.4 Vybrané technologie a implementace

### 3.4.1 Backend

Rest Api služba v jazyce C# ve frameworku .Net Core 3.1 s využitím Entity Framework Core 3.1 jako ORM pro práci s databází a Swaggerem pro vizualizaci api a generování klientů pro snadné použití.

Implementováno pomocí návrhového vzoru mediátor s využitím knihovny MediatR. Umožňuje snadno a efektivně odstínit kontrolery api od veškeré logiky, jak to má být. Jako výsledek v nich mediátor pouze posílá konkrétní zprávy, nic víc. Jednotlivé handlers zpráv mediátoru zajišťují primárně CRUD operace nad databází, ale i jiné. Služba je otestovaná pomocí unit testů, čímž se zvýší pravděpodobnost bezchybného chodu pro frontend.

### 3.4.2 Web

Použity technologie Html, Css, KnockoutJS, jQuery, Bootstrap na straně klienta s DotVVM na platformě Asp.Net Core 3.1 na straně serveru. DotVVM je de facto C# wrapper nad KnockoutJS, díky čemuž v teorii není nutno napsat jediný řádek v JavaScriptu. Umožňuje .Net vývojářům psát i webové aplikace na architektuře MVVM, jak jsou zvyklí z WPF/Xamarin (desktop/mobil), s minimem JavaScriptu.

Pro jednotlivé klienty zpřístupňující backend jsou napsány třídy, které tyto obalují za účelem výstupu specifickým pro webové potřeby. Těmito jsou například zpracování chyb nebo transformace dat do lépe zobrazitelné podoby, pokud to je třeba. Pro snazší použití těchto tříd je využita Dependency injection, jejíž podporu má Asp.Net Core nativně v sobě a není tedy třeba tento vzor implementovat vlastnoručně, to samé i pro třídy samotného klienta. Ve viewmodelech se pak tyto třídy použijí a celkově se implementuje veškerá zobrazovací logika pro view, včetně zpracování událostí jako jsou třeba kliknutí na tlačítka.

Velká výhoda DotVVM je interní implementace vzoru observer (skrz KnockoutJS), díky kterému lze realizovat propojení hodnot na klientovi a ve viewmodelu (tzv. bindingy), kde se navzájem proměnné upozorňují o změně, čímž ubývá požadavků mezi klientem a serverem. Dále pak nabádá k práci s kontrolkami (v jiných technologiích třeba název subview), díky čemuž lze snadno přepoužít kód ve view. Pro samotný design je využita knihovna Bootstrap.

### 3.4.3 Desktop

Aplikace napsána ve Windows Presentation Foundation (WPF) opět na platformě .Net Core 3.1. Podobně jako na webu jsou vytvořeny wrapper třídy pro klienty služby, jen instanciaci se už neprovádí ručně neboť tu není implementovaná DI a ve viewmodelech se obsluhuje view. Podobně se využívá i bindingů s tím rozdílem, že notifikaci o změně je nutno implementovat, neboť to není nativní chování, jako v případě webu. Vzhledem ke stejnému jazyku, platformě i architektuře je implementace velmi podobná té na webu. Jako markup jazyk pro view se využívá klasický .NETový xaml.

## 4. Návrh GUI - Webová aplikace KDFit [xstane45]

### 4.1 Požadavky na GUI

Webová aplikace slouží k zobrazování novinek z knižního světa (titulní stránka), prohlížení a vyhledávání knih, prohlížení a vyhledávání autorů a zobrazení žebříčku nejlepších (nejlépe hodnocených) knih a autorů. Registrovaný a přihlášený uživatel může také přidávat knihy a autory do seznamu oblíbených, psát recenze ke knihám i autorům a hodnotit jak knihy, tak autory.

### 4.2 Makety

Na základě průzkumu existujících řešení, požadavcích na GUI a následné diskuzi vznikl návrh zachycený na následujících mockupech.



Obr. 1: Prohlížení a vyhledávání knih

# Knižní databáze

Login  
 Heslo  
[Přihlásit se](#)  
[Nemáte účet? Zaregistrujte se!](#)

Novinky  
Knihy  
Spisovatelé  
Žebříček

## Pokročilé vyhledávání

Iméno:   
Příjmení:

Rok narození: od  do   
Národnost:   
Hodnocení: od  do

Řadit dle:
Příjmení A-Z
Příjmení Z-A
Národnost A-Z
Národnost Z-A
Hodnocení ↑
Hodnocení ↓

### Alchymista

Paulo Coelho

Alchymista je román brazilského spisovatele Paula Coelho, poprvé byl vydán Rio de Janeiro roku 1988, v češtině v roce 1999. Kniha vypráví o cestě mladého pastýře za svým snem. Autor čtenáře přesvědčuje o smyslnosti cesty za osobním štěstím...

[Dozvědět se více...](#)

### Pán prstenu

J. R. R. Tolkien

Pán prstenu je epický román záru lidanská fantasy od Johana Ronalda Reuela Tolkienu, napsaný v přestávkách v letech 1937-1949. Patří mezi nejznámější fantasy příběhy, někdy bývá označován za zakládající dílo žánru. Jedná se o třetí nejprodávanější...

[Dozvědět se více...](#)

### Pán prstenu

### Alchymista

Obr. 2: Prohlížení a vyhledávání autorů

# Knižní databáze

Login  
 Heslo  
[Přihlásit se](#)  
[Nemáte účet? Zaregistrujte se!](#)

Novinky  
Knihy  
Spisovatelé  
Žebříček

## 1984

George Orwell

Rok vydání: 1949  
ISBN: 978-80-556-1083-2  
Počet stran: 253  
Žánr: Román, Sci-fi, Dystopie  
Hodnocení: 9,2/10

1984, dnes již klasické dílo antiutopického žánru, je bezesporu jedním z nejpozoruhodnějších románů 20. století a brilantní analýzou všech totalitárních systémů. "Antiutopie" má i svého "antihrdinu": je jím Winston Smith, obyčejný, ne příliš statečný člověk, který si nepteje nic světoborného: chce jen poznat skutečnou, ne zmanipulovanou podobu minulosti, věřit tomu, co vidí na vlastní oči a mít svobodu tvrdit, že dva a dva jsou čtyři. Tyto elementární nároky jsou však prímým ohrožením moci, postavené na lži, a proto je Winstonova vzpoura proti oblidnému organismu, jehož je nepatrnou buňkou, předem odsouzená k nezdaru. Presto mu autor vkládá do úst jistou naději, působivější o to, že vyslovenou v mčném: "Nějak už vám to selže, něco vás zdolá. Život vás zdolá."

### Recenze

[Napsat recenzi](#)

vladislav  
10/10

Tato kniha mě velmi překvapila. Jelikož je to povinná četba, tak jsem ji s jistým odporem začala číst. Nakonec mě úplně dostala. Není to lehké čtení a člověk se nad tím musí trochu zamyslet, ale podle mě pro člověka, který něco ví o historii, to nebude zas tak těžké. Mrazivé ovšem je, že když se zamyslete, kdy toto dílo autor napsal, a jak situace vypadala v roce 1984, tak vám dojde, že se autor docela trefil.

user1984  
0.0/0

Povinná četba. V roce 1984 byl příběh této knihy realitou ve "východním bloku", o dvacet let později v zemi svého vzniku - Velké Británii.

Obr. 3: Detail knihy. Detail autora vypadá stejně.  
(samozřejmě obsahuje obrázek a informace o autorovi)

## Recenze

[Napsat recenzi](#)

Hodnocení  / 10

Text recenze

Obr. 4: Psaní recenze (vyžaduje přihlášeného uživatele)

# Knižní databáze

Login  
 Heslo  
[Přihlásit se](#)  
Nemáte účet? [Zaregistrujte se!](#)

Novinky  
Knihy  
Spisovatelé  
Žebříček ▾

Napište nám...

## Registrace nového uživatele

Přezdivka (přihlašovací jméno):

Heslo:

Potvrdit heslo:

Registrovat

Obr. 5: Registrace uživatele

# Knižní databáze

Přihlášený uživatel:  
ferd00  
[Upravit profil](#)

Novinky  
Knihy  
Spisovatelé  
Žebříček ▾  
Oblíbené ▾

Napište nám...

## Upravit profil uživatele

Přezdivka (přihlašovací jméno):  
ferd00  
Uložit

### Změna hesla

Současné heslo:

Nové heslo:

Potvrdit nové heslo:

Změnit heslo

Obr. 6: Úprava profilu přihlášeného uživatele



Novinky  
Knihy  
Spisovatelé  
Žebříček  
Oblíbené  
Oblíbené knihy  
Oblíbení autoři

## Oblíbené knihy



Obr. 7: Procházení oblíbených knih

## 4.3 Pilotní test

Testování návrhu probíhalo dvěma způsoby. Nejprve byl návrh otestován ostatními členy týmu. Poté byl návrh upraven do podoby, ve které je vyobrazen v mockupech v předchozí části. V této podobě pak byl představen dvěma potenciálním uživatelům a testerům. Ti měli nejprve za úkol provést jednoduché úlohy<sup>1</sup>, u kterých bylo pozorováno, zda je GUI dostatečně intuitivní a uživatelsky pohodlné. Po úspěšném či neúspěšném splnění úkolů pak byla vedena volná diskuse o návrhu a GUI a uživatelé - testéři měli možnost volného pohybu po aplikaci, aby si ji mohli vyzkoušet ve všech ohledech.

### 4.3.1 Testování členy týmu

Po prvním, naprosto jednoduchém návrhu, jsme v týmu diskutovali rozložení stránky, pozici nabídky menu a její položky, barevné schéma, pozici přihlašovacího formuláře apod. Výsledek je zřejmý z mockupů - stránka obsahuje vodorovnou hlavičku ve vrchní části, svislou nabídku menu v levé části a obsah stránek se zobrazuje ve zbytku stránky. Taktéž jsme prodiskutovali a rozhodli pozici přihlašovacího formuláře, způsobu registrace a upravení profilu. Celá stránka má zelené barevné schéma.

### 4.3.2 Testování uživateli

Vzhledem k aktuální pandemické situaci jsme pro testování vybrali dva uživatele - čtenáře z rodinného kruhu.

<sup>1</sup> Usability testing - <https://uxplanet.org/ultimate-guide-to-user-research-bed4a57d260>, <https://www.nngroup.com/articles/first-rule-of-usability-dont-listen-to-users/>



V **první části** testu měli uživatelé za úkol provést následujících pět úloh:

1. Najít knihu 1984
2. Najít všechny spisovatele narozené v roce 1984
3. Přidat knihu do oblíbených
4. Změnit přihlašovací jméno
5. Napsat recenzi ke knize či autorovi

Výchozím bodem každé úlohy byla titulní stránka s novinkami (kterou popisuje kolega xloren15). Kromě zadání úkolu byla uživateli ponechána naprostá volnost v pohybu po stránce. Jedinou výjimkou je přidání knihy do oblíbených, kdy bylo před začátkem úlohy uživateli sděleno, že aplikace obsahuje danou funkcionalitu a on/ona ji chtěl/jí využít.

Důležitým prvkem tohoto testování je neovlivňovat uživatele v pohybu, klikání či psaní. Zároveň je potřeba bedlivě sledovat, co dělá a ideálně ho vyzvat ke komentáři u každého úkonu (kliknutí, psaní apod.). Díky tomu lze zjistit myšlenkové pochody uživatele a jeho způsob pohybu v aplikaci.

V **druhé části** testu měli uživatelé volnost pohybu, komentovali však to, co právě chtěl/jí udělat a proč hledají právě na tom konkrétním místě. Na konec proběhla diskuze k celkovému návrhu a použitelnosti GUI včetně připomínek.

## Uživatel 1

### První část

Najít knihu 1984 - bez problémů, uživatel postupoval dle očekávání

Najít spisovatele narozené v r. 1984 - bez problémů, uživatel postupoval dle očekávání

Přidat knihu do oblíbených - uživatel postupoval dle očekávání, ale hvězdičku značící přidání do oblíbených v detailu knihy hledal několik vteřin, tedy dlouho - hvězdička je málo výrazná. Pro přidání do oblíbených se uživatel musí přihlásit - protože však zatím nemá účet, musel se registrovat. Byl z toho trochu zmatený a proto na to byl dotázán v konečné diskuzi.

Změnit přihlašovací jméno - bez problémů, první pohyb myši (a očima) vedl k místu přihlašovacího formuláře. Místo pro odkaz na upravení profilu je tedy na správném místě.

Napsat recenzi - bez problémů, o recenzích věděl z předchozího vyhledávání knihy.

### Druhá část

Všechny činnosti, které uživatel prováděl probíhaly dle očekávání. V diskuzi na konci testu vyplynuly následující názory uživatele:

- Hvězdička pro přidání do oblíbených je sice velká, ale málo výrazná. *Návrh na řešení: přidat ke hvězdičce popis "Přidat do oblíbených".*
- Registrace - při přesměrování na registrační formulář byl uživatel zmatený a ptal se, proč by se měl registrovat. *Návrh řešení: přidat k registračnímu formuláři důvody, proč se registrovat - psaní recenzí, seznamy oblíbených knih a autorů, psaní zpětné vazby.*
- Psaní zpětné vazby - popis evokuje možnost položit otázku, avšak neexistuje žádný způsob jak na ni uživateli odpovědět. *Návrh řešení: přeformulovat nápis "Napište nám..." na "Napište nám zpětnou vazbu".*

- Recenze - jednotlivé recenze jsou odděleny až moc, graficky tvoří samostatné bloky. Zároveň je málo graficky oddělený formulář pro psaní nové recenze od ostatních, již napsaných recenzí. *Návrh řešení: provést revizi oddělovačů u recenzí.*
- U recenzí by také ocenil viditelné datum. *Návrh na řešení: přidat datum k recenzím.*

## Uživatel 2

### První část

Najít knihu 1984 - bez problémů, uživatel postupoval dle očekávání

Najít spisovatele narozené v r. 1984 - bez problémů, uživatel postupoval dle očekávání

Přidat knihu do oblíbených - uživatel postupoval dle očekávání, ale hvězdičku značící přidání do oblíbených opět hledal dlouho - hvězdička je vskutku málo výrazná. Pro přidání do oblíbených se uživatel musí přihlásit - protože však zatím nemá účet, musel se registrovat. Hned mu došlo proč se musí registrovat, nicméně ho to překvapilo.

Změnit přihlašovací jméno - bez problémů, první pohyb myši (a očima) vedl k místu přihlašovacího formuláře. Místo pro odkaz na upravení profilu je tedy na správném místě.

Napsat recenzi - bez problémů, uživatel postupoval dle očekávání.

### Druhá část

U dalších činností uživatel postupoval dle očekávání. V diskuzi uživatel zmínil:

- Hvězdička pro přidání do oblíbených je málo výrazná. *Návrh na řešení: přidat ke hvězdičce popis "Přidat do oblíbených".*
- Registrace - při přesměrování na registrační formulář uživateli sice došlo, proč se musí registrovat, zároveň by ho však zajímalo, co všechno může s účtem dělat (zda se registrace vyplatí). *Návrh řešení: přidat k registračnímu formuláři důvody, proč se registrovat - psaní recenzí, seznamy oblíbených knih a autorů, psaní zpětné vazby.*

## 4.4 Vyhodnocení testu a revize návrhu

Díky oběma uživatelům testujících webovou aplikaci jsou zřejmé dva nedostatky: málo výrazná hvězdička pro přidání do oblíbených v detailu knihy/autora a matoucí či málo vysvětlené důvody k registraci. Navrhovaným řešením tedy je přidat popis ke hvězdičce, aby byla výraznější a k registračnímu formuláři doplnit krátký popis a seznam důvodů, proč se registrovat.

Dalšími nedostatky aplikace je nedostatečně specifický nápis odkazu pro napsání zpětné vazby, méně přehledný seznam recenzí a nedostatečně oddělený formulář pro psaní nové recenze od recenzí starších, u kterých také není vidět datum jejich vytvoření. Řešení tedy je provést grafickou revizi oddělovačů u recenzí a přidání datumů k nim.

## 5. Návrh GUI – Rozložení a vybrané části webové a desktopové aplikace KDFit [xloren15]

### 5.1 Požadavky na GUI

U konkurenčních aplikací jsme si všimli, že jsou často řešeny stylem obsahu uprostřed a prázdným místem na stranách. Toto vede k přehluštěnému obsahu a celkově jsme zhodnotili, že se jedná spíše o starší typ designu webu, často proto, že se mnoho programátorů bojí responzivního designu a rozložit web na 50% obrazovky již na začátku tento problém řeší vcelku dobře. Z toho důvodu by bylo vhodnější modernější řešení rozložení aplikací s postranním menu, čímž se uvolní obsah hlavičky a stránka jako taková se rozšíří. To nám pak dá více místa a lepší vizuální požitek z aplikace.

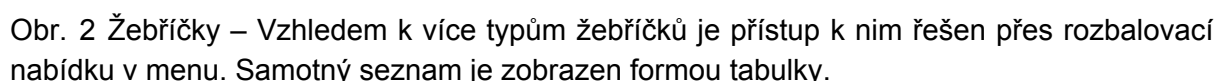
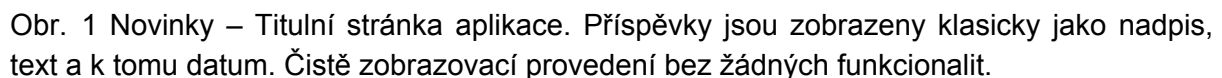
Sekce novinky by neměla obsahovat zvláštní funkcionalitu. Jedná se o titulní stránku s cílem zobrazit nejnovější příspěvky administrátorů systému týkající aktualit z knižního světa, jako jsou například nadcházející autogramiády, nově vydané knihy atd. Cílem stránky je tedy tyto pouze uživateli zobrazit nějakým příjemným způsobem.

Žebříčky by měly sloužit k zobrazení nejoblíbenějších knih a autorů podle uživatelských hodnocení. Takový seznam lze vizualizovat prostřednictvím tabulky.

Feedback obecně slouží k jednoduché jednostranné komunikaci mezi uživatelem a správcem systému. Typicky se jedná o návrh na vylepšení, upozornění na chyby či nepřesnosti nebo obecně o informování o nějaké skutečnosti. Přesně k tomuto účelu by měly sloužit ve webové aplikaci. Vzhledem k tomu, že se jedná jen o kus textu, není potřeba řešit speciální stránkou, nýbrž stačí použít vyskakovací okno či popover. Jedná se o běžnou formu řešení, proto by to mělo být pro uživatele intuitivní.

V desktopové aplikaci se pak feedbacky zobrazují ve formě seznamu s možností feedback skryt, pokud byl již vyřešen nebo smazat, jedná-li se o spam, či jiný nevhodný text.

### 5.2.1 Web





Obr. 3 Feedback – Tento prvek bývá typicky umístěn v některém ze spodních rohů aplikací. Pro zachování zvyklostí jsme tak taktéž učinili. V principu se jedná jen o textové pole s tlačítkem, proto vyřešeno přes vyskakovací okénko.

## 5.2.2 Desktop



Obr. 4 Rozložení – Ovládání aplikace je rozvrhnuto mezi hlavičku a postranní menu. To má za následek větší přehlednost. Dále lze postranní menu schovat, čímž se získá více místa na stránce.



Obr. 5 Feedbacky – Administrátorovi se budou přehledně zobrazovat feedbacky jednotlivých uživatelů. Má možnost je skrýt, pokud je již třeba vyřešil, nebo smazat, pokud mají nevhodný obsah. Aby o již zobrazené navždy nepřišel, má možnost si je znovu zobrazit skrze checkbox. Dále je zde vidět skryté postranní menu.



Obr. 6 Vyhledávání – V aplikaci je globální vyhledávání, ve němž dále uživatel specifikuje, ve které části aplikace chce vyhledávat.

## 5.3 Pilotní test

Výše zmíněné části aplikací byly testovány na dvou uživateli, jednom zručném v práci s počítačem a jednom naopak ne příliš. Tyto testy jsem prováděl s každým zvlášť tím způsobem, že jsem měl mockupy vytisknuté a při uživatelských akcích jsem před ně předkládal následující.

### 5.3.1 Web

#### **Novinky:**

Vzhledem k tomu, že zde není žádná funkcionalita a ještě se jedná o titulní stránku, čili první, která se uživateli zobrazí, nebylo zde moc, co testovat. Testoval jsem alespoň otázkou: „Co bys čekal, že se ti zobrazí, když aplikaci spustíš?“.

Zkušený uživatel by čekal „nějakou“ titulní stránku, přičemž „aktuality“ padly jako jedna z možných forem titulní stránky.

Nezkušený uživatel odpověděl slovem „knížky“, když však po ukázání mockupu viděl sekci novinek, řekl, že „se také jedná o pěknou funkci“.

Dále jsem se ptal, jak dobře se jim příspěvky čtou. Od obou jsem se dozvěděl, že by nejen ocenili větší rozestupy mezi samotnými příspěvkem, ale i větší odsazení textu od krajů bloků a hlavičky.

#### **Žebříčky:**

Úkol zněl následovně: „Řekni mi, která kniha je pátá nejoblíbenější?“

Zkušený rozpoznal rozbalovací menu a po ukázání na něj řekl, že se asi ještě objeví možnosti žebříčků, nečekal tedy rovnou novou stránku. Následně ukázal na „Nejlepší knihy“ a v tabulce snadno našel druhý díl Pána prstenů. Po úkolu sám od sebe prohlásil, že mu to přijde intuitivní.

Méně zkušený taktéž nejprve ukázal na Žebříčky v menu, nicméně pak se hlavou otočil k obsahu, nerozpoznal tedy rovnou rozbalovací menu. Bez problémů pak však ukázal na „Nejlepší knihy“ a našel správnou knihu.

#### **Feedback:**

Tu jsem byl velmi zvědavý, jak bude psaní zpětné vazby pro uživatele snadné. Vzhledem k tomu, že pro to neexistuje stránka a není to ani v menu, zněla úvodní otázka následovně: „Vím o autogramiádě slavného autora X a chci se o to podělit. Jak dám správci systému o této akci vědět?“

Zkušený uživatel ani nehledal, rovnou ukázal na nápis „Napište nám“. Dále ukázal na textové pole popoveru, kam by psal text, a poté na tlačítko odeslat. Dle jeho slov si toho všiml už na prvním obrázku s novinkami, když si udělal přehled, co se vše na stránce nachází.

Druhý uživatel nejdřív hledal v hlavičce stránky, poté se přesunul do postranního menu, kde to dole našel. Následně již neměl s textovým polem a tlačítkem problém.

### 5.3.2 Desktop

#### **Vyhledávání:**

Tato část byla vcelku jednoduchá pro obě strany. Já jsem pouze zadal, aby mi v aplikaci vyhledali Christophera Paoliniho. Oba zcela s přehledem ukázali na vyhledávací box, kam by začali psát a poté zvolili „Hledat v Autoři“.

#### **Feedback:**

Test jsem začal tím, že jsem ukázal mockup mého kolegy se seznamem knih a zadal následující úkoly:

„Přečti mně druhý feedback a označ ho jako zpracovaný.“

„Smaž první feedback.“

„Zapomněl jsem, co jsem si zrovna přečetl, co jak se na to znovu můžu podívat?“

Najít feedbacky v menu nebyl problém ani pro jednoho uživatele a druhý feedback mně přečetli bez problémů. Nad tlačítkem „Zobrazeno“ méně zkušený uživatel chvíli váhal, ale nakonec oba označili správně. S druhým úkolem žádné potíže nebyly. Pro najetí checkboxu na zobrazení všech feedbacků se oba uživatelé museli chvíli zorientovat, ale relativně rychle taktéž oba našli a pomyslně zaklikli. Tu se však oba uživatelé pozastavili nad jeho smyslem, neboť by údajně jen buď provedli požadavek z feedbacku a tento smazali, nebo ho smazali rovnou a požadavek ignorovali. Nedokázali si představit, že by se někdy k nějakým vraceli.

### 5.3.3 Společné

#### **Layout:**

Toto bohužel není věc, kterou lze snadno otestovat, neboť se jedná jen o jakýsi vjem aplikace. Z toho důvodu jsem alespoň pokládal otázky, zda-li jim toto rozložení vyhovuje, jestli by spíše preferovali výše zmíněný vycentrovaný styl aplikace, zda je menu přehledné a intuitivní. Ukázal jsem jim i konkurenční aplikace pro srovnání.

Zpětná vazba byla taková, že rozložení vyhovovalo zkušenějšímu uživateli více než ty s postranním místem, méně zkušenějšímu to bylo jedno, nicméně naše řešení mu vyhovovalo. Menu a hlavičky se jim zdály být intuitivní, líbila se jim i možnost menu rozšiřovat a naopak schovávat, jen ikony pro autora a uživatele by údajně mohly být více odlišitelné.

## 5.4 Vyhodnocení testu a revize návrhu

Vcelku vzato jsem shledal testy úspěšnými. Každodenní uživatel počítačových aplikací bez sebemenších problému či pochybností zcela správně odpověděl na všechny otázky a



provedl všechny úkony. Uživatel nezběhlý v práci s počítačem sice občas hledal nebo i tipoval, ve výsledku však všechny úkony také zvládl.

V návrhu bych proto měnil jen detaily. Ukázalo se, že by bylo vhodné místy trochu zvětšit odsazení pro lepší vizuální dojem, zejména u příspěvků novinek od sebe, jejich textů od okrajů nebo tlačítka „Odeslat“ u psaní feedbacku. V důsledku zpětné vazby testerů u feedbacků v desktopové aplikaci jsme se v týmu shodli, že funkcionality “zobrazeno” je zbytečná, což v návrhu pouze odstraní pár prvků, nicméně rozložení to neovlivní.

## 6. Návrh GUI - Desktopová aplikace KDFit [xkalab11]

### 6.1 Požadavky na GUI

#### 6.1.1 Požadavky na desktopovou aplikaci jako celek

U aplikace pro správce nejsou kladené tak vysoké požadavky na líbivost aplikace jako v aplikaci pro uživatele. Aplikace musí být čistá, rychlá a k relevantním informacím musí vést co nejkratší cesta. Nejčastější prací administrátora bude vytváření a upravování knih, autorů a novinek. Administrátor musí mít možnost cenzurovat (pouze mazáním) jak recenze, tak uživatele samotné. Administrátor také musí mít možnost spravovat zpětnou vazbu od uživatelů. Administrátor je informovaným uživatelem, používá aplikaci s jasně určeným cílem, proto mu musíme poskytnout možnost tohoto cíle co nejdříve dosáhnout.

#### 6.1.2. Požadavky na tabulkové zobrazení záznamů

Tabulky musí zobrazovat relevantní informace pro správce, pojmout velké množství informací a poskytovat možnost rychle a efektivně filtrovat a řadit informace. Samozřejmě nesmí chybět ani stránkování.

#### 6.1.3 Požadavky na detaily záznamů

Detaily záznamů musí být multifunkční jak pro editaci, čtení, tak i vytváření nového záznamu. Jednotný design módů formuláře umožní administrátorovi rychlejší zaučení v používání aplikace. Jelikož recenze knih ani autorů nejsou v aplikaci vedeny samostatně (bez kontextu ztrácí smysl), musí být na detailu těchto záznamů možnost recenze číst a mazat.

## 6.2 Makety

Knížní databáze			
Vyhledat			
<			
+ Nový			
Filtr			
↑ ID	Jméno	Autor	ISBN
141	12 pravidel pro život	Jordan B. Peterson	978-80-257-2792-8
145	Brisingr	Christopher Paolini	80-7200-978-8
150	Zelená míle	Stephen King	80-7306-024-8

< 1 >

Na prvním mockupu vidíme tabulku záznamů knih. Zobrazené jsou nejdůležitější sloupce pro administrátora databáze. Sloupce jsou momentálně seřazené vzestupně podle ID. Řazení tabulky lze změnit kliknutím na sloupec podle kterého chceme řadit. Tato funkcionality je možná méně intuitivní, ale umožňuje rychlou navigaci informovanému uživateli za uchování vysoké hustoty informací na stránku. Nad tabulkou se nachází rychlá možnost založit nový záznam a rozbalitelné menu filtru. Kliknutím na řádek tabulky se dostaneme na detail knihy.

Knížní databáze

Vyhledat

<

Knihy

Autoři

Novinky

Uživatelé

Feedback

+ Nový

ID:

Jméno:

Autor:

ISBN:

Resetovat filtr

Aplikovat filtr

↑ ID	Jméno	Autor	ISBN
141	12 pravidel pro život	Jordan B. Peterson	978-80-257-2792-8
145	Brisingr	Christopher Paolini	80-7200-978-8
150	Zelená míle	Stephen King	80-7306-024-8

< 1 >

Rozbalené menu filtru umožní filtrovat jakýkoliv ze sloupců tabulky.

Knížní databáze

Vyhledat

<

Knihy

Autoři

Novinky

Uživatelé

Feedback

12 pravidel pro život

Upravit

Jordan B. Peterson

Rok vydání: 2018

Počet stran: 424

Žánr: psychologie, filozofie

Vazba: měkká

ISBN: 978-80-257-2792-8

Co by měl znát každý člověk v moderním světě? ptá se renomovaný psycholog Jordan B. Peterson a jeho odpověď na tuto nesnadnou otázku v sobě jedinečně propojuje odvěké pravdy a ohromující objevy nejsoučasnějšího vědeckého výzkumu.

Jarda ★★★★★

„Bravo! Pan Psycholog s velkým P. Rád sleduji jeho přednášky a podle mě je to jeden z předních psychologů západního světa. Jeden z mála, který se nebojí říct pravdu do očí a říct ji takovou jaká je, bez toho, aniž by se snažil podlézat současným trendům a postmodernímu populacionalismu, který

Detail knihy je jednoduchý a přehledný. Přes ikonku u autora se dostaneme na záznam autora. Tato funkcionalita je možná méně intuitivní, ale umožňuje rychlou navigaci informovanému uživateli.

27

Knížní databáze

Vyhledat

<

Knihy
Autoři
Novinky
Uživatelé
Feedback

12 pravidel pro život

Odstranit

Uložit

Jordan B. Peterson

Rok vydání: 2018

Počet stran: 424

Žánr: psychologie, filozofie

Vazba: měkká

ISBN: 978-80-257-2792-8

Co by měl znát každý člověk v moderním světě? ptá se renomovaný psycholog Jordan B. Peterson a jeho odpověď na tuto nesnadnou otázku v sobě jedinečně propojuje odvěké pravdy a ohromující objevy nejsoučasnějšího vědeckého výzkumu.

Jarda ★★★★★

Odstranit

„Bravo! Pan Psycholog s velkým P. Rád sleduji jeho přednášky a podle mě je to jeden z předních psychologů západního světa. Jeden z mála, který se nebojí říct pravdu do očí a říct ji takovou jaká je, bez toho, aniž by se snažil podlézat současným trendům a postmodernímu populacionalismu, který

Jordan B. Peterson
12 pravidel pro život
Pročítáčka pro Kindle

Nahrát

Detail knihy v editaci. V editaci je možné záznam upravit, či kompletně odstranit. Také je možné mazat recenze.

Knížní databáze

Vyhledat

<

Knihy
Autoři
Novinky
Uživatelé
Feedback

Název knihy

Uložit

Autor knihy

Rok vydání:

Počet stran:

Žánr:

Vazba:

ISBN:

Popis

Nahrát

Formulář nového záznamu knihy.

28

Formuláře a tabulky autorů a novinek budou vypadat velmi podobně. Proto jsem jejich mockupy nevytvářel. Seznam uživatelů bude vypadat podobně. Uživatelé ovšem nebudou mít detailové zobrazení a půjdou přímo ze seznamu smazat.

## 6.3 Pilotní test

Nejdůležitější metrikou efektivnosti designu aplikace pro správce je rychlost provedení běžných úkolů správců. Druhou nejdůležitější metrikou je intuitivnost aplikace. Třetí metrikou je líbivost a design aplikace. Metriky těchto testů jsou tedy zvoleny následovně:

- Počet kroků potřebných k provedení cíle
- Počet chybných kroků nového uživatele
- Hodnocení aplikace uživatelem na škále 1-10

Uživatelé dostali následující úkoly:

1. Změnit ISBN knihy „12 pravidel pro život“. Z výchozí pozice seznam knih.
2. Odstranit recenzi s určeným nevhodným slovem pod knihou „12 pravidel pro život“. Z výchozí pozice detail knihy „12 pravidel pro život“.
3. Zobrazit v seznamu knih pouze knihy od Jordana Petersona. Z výchozí pozice detail knihy „12 pravidel pro život“.
4. Slovně ohodnotit design a intuitivnost aplikace a oznámkovat na škále 1-10.

V průběhu tohoto testu měli uživatelé za úkol popisovat svoje myšlení a vysvětlovat svoje kroky.

### 6.3.1 Uživatel číslo 1

První uživatel byla mladá studentka, která pracuje jako technická podpora. Jedná se tedy typově o uživatele, pro kterého je tato aplikace navržena. Test s tímto uživatelem proběhl osobně.

### Průběh testu:

Změnit ISBN knihy 12 pravidel pro život:

1. Uživatel na seznamu knih klikl na položku ISBN knihy 12 pravidel pro život. Toto jej přesměrovalo na detail knihy. *Uživatel klikal přímo do sloupce ISBN. Toto však nijak nevadilo, jelikož odkazem je celý řádek.*
2. Uživatel klikl na tlačítko „Upravit“. Toto jej přeplo do editačního módu knihy.
3. Uživatel upravil ISBN.
4. Uživatel klikl na tlačítko „Uložit“.

První úkol tedy proběhl bezchybně a dle očekávání.

Odstranit recenzi s nevhodným slovem pod knihou 12 pravidel pro život:

1. Uživatel klikl na tlačítko upravit, jelikož si možnosti odstranit recenze na editačním módu všiml v úkolu 1. *Bohužel jistě familiérnosti se v průběhu testování nelze vyhnout. Postupem testování se uživatel rychle učí a brzo ztrácí status prvotního uživatele.*
2. **Uživatel se pokusil vyhledat recenzi s předem určeným nevhodným slovem pomocí globálního vyhledávání.** Toto mu pouze nabídlo vyhledat v knihách, autorech, či novinkách. Uživatel zjistil, že tudy cesta nevede. Jednalo se o chybný krok.
3. **Uživatel se pokusil použít zažitou funkci ctrl+f.** Jelikož o implementaci této funkce jsme neuvažovali, aplikace nereagovala a jednalo se o chybný krok.
4. Uživatel si všiml postranního scrollbaru a recenzi našel.
5. Uživatel použil tlačítko odstranit a recenzi úspěšně smazal.

Tento úkol proběhl s mírnými zádrhly. Uživateli trvalo si všimnout možnosti scrollovat stránkou.

Zobrazit v seznamu knih pouze knihy od Jordana Petersona:

1. Uživatel rozbalil menu filtru kliknutím na ikonku vedle slova filtr.
2. Uživatel do kolonky autor vypsál klíčové slovo „Peterson“.
3. **Uživatel stiskl enter pro aplikování filtru.** Tato funkce nebyla uvážena, tudíž se z našeho pohledu jednalo o chybný krok.

4. Uživatel použil tlačítko aplikovat filtr. Tímto uživatel dokončil úkol.

Hodnocení: 8/10

Uživatel si připadal ve známém prostředí. Aplikace se chovala intuitivně až na pár neimplementovaných (neuvážených) funkcionalit.

### 6.3.2 Uživatel 2

Druhá uživatelka byla žena ve středních letech, která má omezené uživatelské zkušenosti s počítači. Často navštěvuje velmi úzkou skupinu webových stránek, na kterou je zvyklá a umí používat pouze pár programů. Test s tímto uživatelem proběhl přes sdílení obrazovky přes platformu Microsoft Teams.

Průběh testu:

Změnit ISBN knihy 12 pravidel pro život.

1. Uživatel klikl na ISBN „12 pravidel pro život“
2. Uživatel si scrollbarom prohlédl celou stránku, aby dle jeho slov „Měl přehled“.
3. Uživatel klikl na tlačítko „Upravit“.
4. Uživatel klikl do řádku ISBN.
5. Uživatel přepsal ISBN.
6. Uživatel klikl na tlačítko „Uložit“.

První test proběhl bez větších problémů. Překvapil mě uživatelův impuls prvně prohlédnout celou stránku a až poté řešit cíl úkolu.

Odstranit recenzi s nevhodným slovem pod knihou 12 pravidel pro život.

1. Uživatel se přepnul do editačního módu.
2. Uživatel napsal do globálního vyhledávání hledané slovo.
3. Uživatel klikl na hledat v „Knihy“. To jej přesměrovalo na seznam. Uživatel byl zhnusen absencí vyhledávání v recenzích.



4. Uživatel se vrátil na detail knihy 12 pravidel pro život.
5. Uživatel se přepnul do editačního módu.
6. Uživatel scrolloval a našel recenzi.
7. Uživatel recenzi odstranil.

V tomto testu již podruhé vyplynul požadavek na vyhledávání v recenzích.

Zobrazit v seznamu knih pouze knihy od Jordana Peterson:

1. Uživatel klikl na seznam autorů.
2. Uživatel rozklikl autora „Jordan B. Peterson“.
3. Uživatel požadoval seznam knih na detailu autora. *Toto je velmi rozumná funkcionalita, která mi však unikla.*
4. Uživatel vyhledal „Peterson“ v globálním vyhledávání knih. To jej dostalo na seznam knih vyfiltrovaných na jméno „Peterson“.

Druhý uživatel hledal seznam knih autora na jeho záznamu. Poté splnil vyfiltrování seznamu pomocí globálního vyhledávání, což je alternativa k filtrování seznamu samotného.

Hodnocení: 6/10

Uživateli se nelíbí omezenost globálního vyhledávání a nemožnost vyhledávání v hodnoceních. Editace záznamu byla uživateli známá a očekávaná. Uživatel objevil překlep v nadpisu programu. Uživatel nemá rád zelenou barvu, design se mu tedy příliš nelíbí.

## 6.4 Vyhodnocení testu a revize návrhu

Z těchto testů byl znát zřetelný rozdíl mezi použitím aplikace zdatným a nezdatným uživatelem. Zatímco nezdatný uživatel očekával spíše uživatelské prvky jako například líbivější design, aplikaci více prohlížel a byl zmatenější, zdatný uživatel aplikaci procházel rychle, soustředil se na cíl a chyběla mu například integrace klávesnice. Oba uživatelé se shodli na požadavku vyhledávání v recenzích, a to ať přes globální vyhledávání či v rámci detailu knížky. Překlep v nadpisu byla úsměvná chyba. Osobně vidím velkou přidanou hodnotu v možnosti vyhledávat v rámci stránky a tuto funkci zvážím. Integrace klávesnice a klávesových zkratk je další velmi žádanou funkcí a zvážím její implementaci.

Efektivnost aplikace byla vysoká a bezchybné provedení úkolů bylo možné uskutečnit v rámci pár kroků. Výsledky testu jsou vesměs uspokojivé a poskytují několik přínosných poznatků. Nižší intuitivnost obzvláště u uživatelů nezkušených s podobnými aplikacemi byla očekávaná

## 7. Implementace GUI - Webová aplikace KDFit [xstane45]

### 7.1 Implementace

Implementace využívá běžnou .NET architekturu MVVM. Modelová vrstva obsahuje wrappery, které volají klienty naší služby (backendu), obohacené o zpracování chyb a případnou úpravu dat. ViewModelová vrstva obsahuje GUI logiku pro všechny stránky, zpracovává tedy jednoduché požadavky, upravuje viditelnost, obecně uchovává aktuální stav výsledného prezentovaného GUI. View vrstva obsahuje rozložení a vzhled stránek (a GUI), definovaný jazykem HTML rozšířeném o kontrolky vlastní (v podstatě subview) a zabudované v DotVVM (které jsou při sestavování přeloženy na odpovídající HTML entity).

V řešení je také implementováno několik pomocných struktur. Ty obsahují např. **strategii** pro správné směřování mezi jednotlivými stránkami, pomocnou datovou strukturu pro uchování **recenzí** pro snadnější CRUD operace, **dataset** na který je možné iterativně vznášet dotazy, což umožňuje postupné načítání dlouhých seznamů (nenačítá se celý seznam ale pouze jeho část) a možnost vytvoření stránkování pro procházení takovýchto seznamů, atd.

Propojení View a ViewModel vrstvy je implementováno pomocí tzv. binding, tj. propojení hodnot na klientovi a ve ViewModelu. Díky interní implementaci návrhového vzoru Pozorovatel ve frameworku DotVVM se klient a ViewModel navzájem upozorňují o změnách.

### 7.2 Použité nástroje a knihovny

DotVVM 2.0 na platformě ASP.NET 3.1

**Výhody:** Vynikající framework pro tvorbu webových aplikací založený na ASP.NET. I free verze stačí na tvorbu libovolné funkcionality. Interní implementace vzoru observer - automatická synchronizace mezi View a ViewModelem. Překlad konstrukcí do KnockoutJS - některé operace probíhají na klientovi a nezatěžují server. Výsledkem jsou hezké, rychlé (jak na klientovi, tak na serveru) aplikace.

**Nevýhody:** Složitější konstrukce (dependency injection), nevhodné pro malé jednoduché aplikace bez složitější logiky.

jQuery, Bootstrap 4.5, CSS3 a Font Awesome 4

Vytvářet hezké webové aplikace bez kaskádových stylů je v podstatě nemožné. Jakýmsi standardem pro vytváření responzivních aplikací se stala knihovna Bootstrap využívající CSS, jQuery a tedy i Javascript, a proto je i zde využita. Zdrojem piktogramů a ikon pro lepší UX je Font Awesome 4.

## 7.3 Finální testování

Testování probíhalo velice podobně, jako testování webové aplikace v předchozí části. Uživatel měl splnit následující úkoly:

1. Najít knihu *Brisingr*
2. Napsat recenzi ke knize
3. Najít spisovatele narozené mezi lety 1950 a 1984
4. Přidat tři spisovatele do oblíbených a prohlédnout si svůj seznam oblíbených autorů
5. Změnit heslo

Výchozím bodem testování byla titulní stránka s novinkami, po dokončení úkolu však uživatel zůstal tam, kde skončil. Kromě zadání úkolu byla uživateli ponechána naprostá volnost v pohybu po stránce.

I tentokrát byl důraz kladen na neovlivňování uživatele v pohybu, klikání či psaní. Zároveň je potřeba bedlivě sledovat, co dělá a ideálně ho vyzvat ke komentáři u každého úkonu (kliknutí, psaní apod.). Díky tomu lze zjistit myšlenkové pochody uživatele a jeho způsob pohybu v aplikaci.

### 7.3.1 Průběh testování

Najít knihu *Brisingr* - v podstatě bez problémů. Uživatel nejprve vyhledával knihu s využitím pokročilého vyhledávání podle jména autora. Dotazem bylo ověřeno, že špatně rozuměl zadání úkolu a myslel si, že má vyhledat všechny knihy autora jménem *Brisingr*. Po vyjasnění knihu bez sebemenších problémů našel.

Napsat ke knize recenzi - uživatel využil právě rozkliknuté knihy *Brisingr* a chtěl k ní napsat recenzi. Při rozkliknutí "Napsat recenzi" se mu objevilo upozornění, že se pro psaní recenzí musí registrovat. Tedy se bez nejmenších problémů registroval (čímž náhodou ověřil to, že se nemohou registrovat dva uživatelé se stejným jménem - na první pokus totiž zadal uživatelské jméno, které již v systému existovalo)., znovu našel knihu *Brisingr* a napsal k ní recenzi. Pak byl vyzván aby recenzi upravil - hned po prvním pohybu myši našel správné tlačítko, upravil recenzi a uložil ji (nespletl si tlačítka Zrušit a Uložit - primární a sekundární tlačítka jsou tedy také dobře graficky navrhnutá). Celý tento úkol tedy otestoval několik use-case a **všechny proběhly bez problémů**.

Najít spisovatele narozené mezi lety 1950 a 1984 - uživatel opět využil rolovacího filtru, správně zadal hodnoty a bez problémů našel všechny čtyři spisovatele v databázi.

Přidat tři spisovatele do oblíbených a prohlédnout si svůj seznam oblíbených autorů - Bez problémů. Uživatel resetoval filtr, našel tři sobě známé a oblíbené autory a pomocí hvězdičky v detailu přidal (hvězdičku našel sám a bez problémů) do oblíbených. Nakonec intuitivně v menu vyhledal položku Oblíbené, po rozbalení vybral Oblíbení autoři a prohlédl si svůj seznam.

Změnit heslo - bez nejmenších problémů. Nejprve hledal v pravé horní části obrazovky (kde byl přihlašovací formulář a informace o přihlášeném), našel odkaz Upravit profil a po rozkliknutí si úspěšně změnil heslo na nějaké jiné.

## 7.4 Vyhodnocení testu

Z výsledků testu je jasně vidět, že GUI je navrženo dobře, přehledně a intuitivně. To potvrdil i testovaný uživatel sám, protože podle jeho sdělení je aplikace “pěkná, přehledná, má dobré barvy a všechno je tam, kde bych to hledal, tedy z toho, co jsem tady dělal”. Ocenil i dobrou upozornění aplikace - pro psaní recenzí je nutné se registrovat, důvody k registraci (ty ocenil už během plnění úkolu), vhodně zvolené upozornění, že uživatelské jméno již v systému existuje, atd.

Velice kladně uživatel také ocenil responsibilitu aplikace, tj. že aplikaci lze použít i na mobilu. Menu mu však v portrétovém zobrazení přišlo příliš malé a přišlo by mu vhodnější, kdyby bylo širší a mělo větší tlačítka.

## 8. Implementace GUI - Rozložení a vybrané části webové a desktopové aplikace KDFit [xloren15]

Mimo částí GUI systému jsem měl na starost i tvorbu backend REST API služby. S tím souvisí i tvorba klienta a zprovoznění komunikace GUI aplikací se službou. Proto kromě samotného GUI stručně popíši i tento aspekt implementace.

### 8.1 Implementace

Obě aplikace pracují na stejném návrhovém vzoru MVVM. Vrstvy View a ViewModel jsou v obou případech implementovány ve spustitelných projektech svých řešení. Vrstva Model se pak nachází v knihovně s příponou DAL a má na starost zpřístupnění REST API ViewModelům. Klient pro službu je generovaný nástrojem Swagger a zpřístupněný GUI aplikacím jako Nuget balíček. Díky tomu je volání služby velmi snadné a proveditelné na jediném řádku. Není však vhodné tento přímo používat ve ViewModelech, neboť může dojít k serverové chybě, nebo je zapotřebí dotažená data ještě upravit. Proto jsou klienti obaleni třídami v Model vrstvě řešící tyto věci, aby jednotlivé ViewModely byly implementačně co nejčistší a řešili skutečně jen klientskou logiku.

V obou aplikacích jsem vytvářel jejich rozložení. Na webu je toho dosaženo pomocí bootstrapových tříd, zejména flex a container, zaručující pěkný a zároveň responzivní výsledek. Postranní menu je v případě úzké obrazovky předěláno na rozbalovací z hlavičky aplikace. V kódu je toto možné najít ve view souboru MasterPage.dotmaster sloužící jako šablona pro ostatní stránky s korespondujícím ViewModelem MasterPageViewModel, z něž pak ViewModely ostatních stránek dědí. U desktopové aplikace je rozložení a responzivita řešena pomocí Grid elementů, jež zajišťují automatickou změnu velikosti. Pro užší obrazovku nebo pro zvětšení místa lze taktéž zmenšit postranní menu jen na ikonky bez popisků.

Pro obě aplikace jsem řešil i doménu feedbacků. Na webu je jejich psaní provedeno jednoduše skrze popover, jímž je pouze kus markup kódu, jenž se přidává do / odebírá z DOMu. Na straně desktop aplikace pak tyto mají vlastní stránku, tedy xaml markup a ViewModel, v níž jsou zobrazeny jako seznam a administrátor je může mazat.

Žebříčky nejoblíbenějších knih a autorů jsou na webu řešeny pouze nemodifikovatelnými tabulkami obsahující seřazené entity dle jejich hodnocení. Implementačně mají vlastní stránku i ViewModel.

Poslední mojí částí systému je globální vyhledávání v desktopové aplikaci. To uživateli umožňuje se navigovat na knihu/autora/článek bez nutnosti přecházet na jejich stránky a zadávat filtr. Implementováno je to tím způsobem, že každá doména má vlastní vyhledávací tlačítko, jež uživatele přesměruje na patřičný seznam s přednastaveným filtrem na jméno.

## 8.2 Použité nástroje a knihovny

Klíčovým nástrojem pro komunikaci se službou je již zmíněný Swagger, jenž automaticky vygeneruje klienta pro dané api. Bez něj by se muselo volat službu ručně a znásobilo to kód pro klientskou část aplikace.

Pro desktopovou aplikaci bylo využito WPF frameworku .NET Core. Kromě vygenerovaného klienta pro api již aplikace nepoužívá žádné jiné externí knihovny.

Technologie webové aplikace jsou již popsány v sekci [7.2](#). Doplním však ještě knihovnu popper.js, jež doplňuje bootstrap o funkcionalitu vyskakovacích prvků, jakými jsou například dropdowny, popovery atd.

## 8.3 Finální testování

Úkoly testů byly zcela shodné s těmi v kapitole [5.3](#). Jediným rozdílem bylo vynechání funkcionality “zobrazeno” u feedbacků v desktopové aplikaci, jež nakonec nebyla implementována. Z důvodu stejných testů bylo použito jiných testerů, avšak opět jeden více a druhý méně zdatný s prací na počítači.

Pro shrnutí byly jejich úkoly následující:

- Web
  - Najít 5. nejoblíbenější knihu.
  - Napsat administrátorům o autogramiádě slavného autora X.
- Desktop
  - Vyhledat Christophera Paoliniho.
  - Přečíst druhý feedback.
  - Smazat první feedback.

### 8.3.1 Průběh testování

Najít 5. nejoblíbenější knihu - Oba uživatelé prakticky okamžitě zamířili k menu a klikli na žebříčky. Rozbalení submenu je nezaleklo, zvolili knihy a z tabulky vyčetli 5. záznam.

Napsat feedback - Zde měli oba uživatelé podobnou reakci. Psaní zpětné vazby nejdříve hledali vpravo nahoře, zřejmě proto, že se tam již nalézá nějaký text určený uživatelům, byť pouze pro přihlášení/registraci. Když nenalezli, zamířili dolů, kde si již všimli “Napište nám”. Tento proces však zabral pouhou vteřinu, takže sice až druhý pokus, stále však bez výrazného hledání. Při zobrazení popoveru po kliknutí na tlačítko již nebyl problém napsat zprávu a odeslat.

Vyhledat Christophera Paoliniho - Triviální úkol pro oba uživatele. K vyhledávacímu poli zamířili rovnou, napsali do něj jméno, přečetli si, co je za možnosti a klikli na “V Autoři”.

Přečíst druhý feedback - Opět bez rozmýšlení oba klikli na “Feedback” v menu a ze seznamu bez problému přečetli požadovanou zpětnou vazbu.

Smazat první feedback - V sekci feedbacků ani jeden nepřehlédl ikonu koše ke smazání a správně na ni klikl.

## 8.4 Vyhodnocení testu

Žádný z úkolů nezpůsobil uživatelům žádné problémy. Méně zkušený uživatel ani trochu nezaostával za zkušeným v reakcích či hledání v aplikaci. Z toho usuzuji, že je aplikace dostatečně intuitivní pro různé typy uživatelů.

Po skončení testů jsem byl požádán, jestli by si s aplikacemi mohli chvíli hrát, což jsem s radostí dovolil a dál sledoval, jak uživatelé se systémem pracují. Zcela bez popudů vytvářeli nové knížky, zkoušeli, zda se jim skutečně zobrazí na webu apod. Na základě jejich využití plného potenciálu aplikace bez žádných rad či tutoriálů jsem dospěl k závěru, že je systém z pohledu UI zřejmě dobře navržen.



## 9. Implementace GUI - Desktopová aplikace KDFit [xkalab11]

### 9.1 Implementace

V Desktopové aplikaci jsem vyvíjel tabulkové pohledy knih, autorů a novinek, jejich řazení, filtrování, jejich detaily, editaci a vytváření nových záznamů.

Implementace se sestává ze dvou projektů. V knihovně BooksWpf.DAL se nacházejí třídy služeb, které obalují použití naší API a tedy slouží jako model. V aplikaci BooksWpf se pak nachází Views, ViewModely, použité soubory (ikonky atd.), styly.

Většina zobrazování aplikace je tvořena kontrolkami typu Grid, StackPanel, TextBlock a Image. Pro zobrazování záznamů v tabulkách je použita kontrolka ListView. Pro interaktivitu aplikace využíváme kontrolky Button s navázanými ViewModel příkazy přes WPF binding a navázané visibility pro odkrývání a skrývání částí aplikace.

Pro vkládání dat v editačním zobrazení používáme kontrolky TextBox, ComboBox a DatePicker. Nové záznamy využívají stejné zobrazení jako editace.

Filtrování je zobrazitelné kliknutím na tlačítko "Filtrovat" a je řešené přes sérii vstupních textboxů, které na stisk vnořeného tlačítka "Filtrovat" provolají načtení záznamů s argumenty vypsanych filtrů.

Nadpisy sloupců jsou implementované jako tlačítka sloužící k řazení tabulky. Řadí se vždy podle jednoho sloupce a to buď vzestupně, nebo sestupně.

Validování uživatelských vstupů je prováděno pomocí třídy MessageBox z namespace System.Windows.

Za zmínku stojí funkce SetProperty ve třídě ViewModelBase, která když se použije v Setteru property třídy, tak zajistí okamžité propisování jejích změn na view.

Všechny metody, které používají naši API jsou implementovány asynchronně pomocí třídy Task a provolávány pomocí třídy AsyncCommand pro příjemnější UX bez záseků.

### 9.2 Použité nástroje a knihovny

#### 9.2.1 Windows Presentation Foundation (**WPF**)

Grafický subsystém od Windows umožňující rychlý vývoj prezentační vrstvy MVVM aplikací. Funguje na základě vlastního jazyka XAML, založeném na XML, ve kterém se jednotlivými uzly XML vytváří prvky zobrazení.

V našem řešení představuje view a viewmodel vrstvu desktopové aplikace.

**Výhody:** Velmi dobré vývojové prostředí ve Visual Studiu. Poměrně jednoduché na pochopení a použití. Je velmi dobře připraveno na implementaci MVVM aplikací pomocí data bindingů.

**Nevýhody:** Občas příliš rigidní co se designu týče, vývojáři je umožněno upravovat výsledek omezeně. Složitější úpravy jsou často neintuitivní. Staršího data. Nemůže se vizuálně rovnat například moderním javascriptovým knihovnám.

### 9.2.2 MessageBox

Třída zajišťující jednoduché vyskakovací varovné okno. Klasická funkcionálita Windows.

**Výhody:** Jednoduché použití, známá uživatelům, funkční, doprovázená zvukovým efektem.

**Nevýhody:** Rigidní, působí mírně zastarale.

## 9.3 Finální testování

Finální testování proběhlo s podobnými úkoly jako pilotní testování, ale bylo rozšířeno o několik úkolů. Uživatelé byly rozdílní od těch v pilotních testech.

Uživatelé dostali následující úkoly:

1. Změnit ISBN knihy „12 pravidel pro život“. Z výchozí pozice seznam knih.
2. Odstranit recenzi s určeným nevhodným slovem pod knihou „12 pravidel pro život“. Z výchozí pozice detail knihy „12 pravidel pro život“.
3. V seznamu autorů vyfiltrovat autory, které mají ve jméně písmeno “e” a seřadit je sestupně.
4. Vytvořit novinku.
5. Vymazat uživatele “user”.
6. Slovně ohodnotit design a intuitivnost aplikace a oznámkovat na škále 1-10.

### 9.3.1 Uživatel číslo 1

První uživatel byla mladá studentka, která má běžné uživatelské zkušenosti s technologií. Test s tímto uživatelem proběhl osobně.

#### Průběh testu:

##### **Změnit ISBN knihy 12 pravidel pro život:**

1. Uživatel na seznamu knih klikl na položku ISBN knihy 12 pravidel pro život, jelikož si všiml hover podbarvení. Toto jej informovalo o klikatelnosti řádku a přesměrovalo jej na detail knihy.
2. Uživatel klikl na tlačítko „Upravit“. To jej přeplo do editačního módu knihy.
3. Uživatel upravil ISBN.
4. Uživatel klikl na tlačítko „Uložit“.

**Odstranit recenzi s nevhodným slovem pod knihou 12 pravidel pro život:**

1. Uživatel klikl na tlačítko "Upravit", jelikož si všiml, že nemá možnost mazat recenze na zobrazení. V editaci se mu objevila možnost mazat recenze.
2. Uživatel instinktivně scrolloval.
3. Uživatel našel recenzi.
4. Uživatel použil tlačítko "Odstranit" a recenzi úspěšně smazal.

**V seznamu autorů vyfiltrovat autory, kteří mají ve jméně písmeno "e" a seřadit je sestupně:**

1. Uživatel otevřel záložku autorů.
2. Rozbalil tlačítko "Filtrovat", jelikož viděl chevron a to mu intuitivně napovědělo o rozbalitelnosti filtru.
3. Uživatel vyplnil písmeno e do textboxu jména a stiskl "Filtrovat".
4. Uživatel si po chvíli hledání všiml kliknutelnosti nadpisu sloupce.
5. Poté již intuitivně klikl znovu na nadpis sloupce pro seřazení sestupně.

**Vytvořit novinku:**

1. Uživatel šel do seznamu novinek.
2. Klikl na tlačítko "Nový".
3. Vypsal nadpis.
4. Datum neměnil.
5. Vypsal text, který chtěl.
6. Klikl na tlačítko "Uložit".

**Smazat uživatele user:**

1. Uživatel jde do seznamu uživatelů.
2. Uživatel smazal uživatele tlačítkem koše.

**Hodnocení: 9/10**

Uživatel si připadal ve známém prostředí. Aplikace se chovala intuitivně a uživatel neměl žádný problém s jejím užíváním. Bod dolů je za nedokonalý design.

### 9.3.2 Uživatel 2

Druhý uživatel byl muž ve středních letech, která má omezené uživatelské zkušenosti s počítači. Často navštěvuje velmi úzkou skupinu webových stránek, na kterou je zvyklý a umí používat pouze pár programů. Je také občasným hráčem videoher. Test s tímto uživatelem proběhl osobně

**Průběh testu:****Změnit ISBN knihy 12 pravidel pro život.**

1. Uživatel po chvilce hledání klikl na řádek knihy „12 pravidel pro život“
2. Uživatel klikl na tlačítko „Upravit“.
3. Uživatel klikl do řádku ISBN.
4. Uživatel přepsal ISBN.
5. Uživatel klikl na tlačítko „Uložit“.

**Odstranit recenzi s nevhodným slovem pod knihou 12 pravidel pro život:**

1. Uživatel hledal tlačítko smazání na recenzi.
2. Uživatel po chvíli přemýšlení klikl na tlačítko editovat
3. Uživatel scrolloval a našel recenzi.
4. Uživatel recenzi odstranil.

**V seznamu autorů vyfiltrovat autory které mají ve jméně písmeno “e” a seřadit je sestupně:**

1. Uživatel otevřel záložku autorů.
2. Rozbalil tlačítko “Filtrovat” a čekal co se stane.
3. Po rozbalení intuitivně vyplnil pole “Celé jméno” písmenem “e”.
4. Uživatel stiskl tlačítko “Filtrovat”.
5. Uživatel si po delším hledání všiml kliknutelnosti nadpisu sloupce.
6. Poté již intuitivně klikl znovu na nadpis sloupce pro seřazení sestupně.

**Vytvořit novinku:**

1. Uživatel šel do seznamu novinek.
2. Klikl na tlačítko “Nový”.
3. Vypsal nadpis.
4. Datum změnil v DatePickeru bez problémů.
5. Vypsal text který chtěl.
6. Klikl na tlačítko uložit.

**Smazat uživatele user:**

1. Uživatel jde do seznamu uživatelů.
2. Uživatel smazal uživatele ikonkou koše.

**Hodnocení: 7/10**

Uživateli se nelíbí skrytost funkcionality řazení, nelíbí se mu absence nadpisu polí ve vytváření novinek. Design jej také neuchvátil.

## 9.4 Vyhodnocení testu

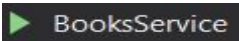
Osobně jsem byl s výsledky velmi spokojen. Uživatelé neměli s používáním systému žádné výraznější potíže až na třídění, které bylo záměrně neinvazivně skryté, jelikož desktopová aplikace počítá s informovanými uživateli. Pár připomínek bylo k nemodernímu designu, ale jelikož desktopová aplikace není určena běžným uživatelům, již volba frameworku nepočítala s oslnivým designem. Výsledky testu tedy hodnotím velmi uspokojivě.

## 10. Zprovoznění systému

Odevzdaný soubor obsahuje vše potřebné kromě knihoven třetích stran pro webovou aplikaci. Těmi jsou [Bootstrap v4.5.2](#), [popper v2](#), [jQuery](#), [Font Awesome Free 5.15.0](#). Ty lze jednoduše stáhnout z adresy <http://www.stud.fit.vutbr.cz/~xloren15/itu-web-libs.zip>. Zip je třeba rozbalit do složky /Web/BooksWeb/wwwroot/lib.

Dále je třeba získat nuget balíčky služby pro klientské aplikace a pak už jen spustit.

Postup pro Visual Studio:

1. Otevřte BooksService.sln v IDE Visual Studio.
2. Pravým tlačítkem klikněte na "Solution 'BooksService'" a zvolte "Build Solution".
3. Pravým tlačítkem klikněte na projekt "BooksService.Client" a zvolte "Pack".
4. Pravým tlačítkem klikněte na projekt "BooksService.Common" a zvolte "Pack".
5. Ze složky bin/Debug či bin/Release (v závislosti na nastaveném sestavovacím módu) jednotlivých projektů přesuňte soubory BooksService.Client[Common].1.0.0.nupkg do libovolné složky, která bude zdrojem nuget balíčků.
6. V Tools->Options->NuGet Package Manager->Package Sources přidejte lokální složku s vašimi nuget balíčky.
7. Spustěte službu tlačítkem Run 
8. Otevřte BooksWeb.sln a BooksWpf.sln v IDE Visual Studio a taktéž spustěte (V případě problémů sestavení z důvodu nenalezených nuget balíčků může pomoci Visual Studio vypnout a zapnout. Je možné, že si ještě nenačetlo nový zdroj balíčků.)

Příkazový řádek (vyžaduje mít nainstalovaný dotnet core a nuget.exe):

1. dotnet pack <cesta\_k\_BooksService.Client.csproj> --output <nuget\_slozka>
2. dotnet pack <cesta\_k\_BooksService.Common.csproj> --output <nuget\_slozka>
3. nuget sources add -Source <nuget\_slozka> -Name <nejake\_jmeno>
4. dotnet run --project <cesta\_k\_BooksService.csproj>
5. dotnet run --project <cesta\_k\_BooksWeb.csproj>
6. dotnet run --project <cesta\_k\_BooksWpf.csproj>

Databázi není třeba řešit. Při spuštění služby tato automaticky upraví databázi na nejnovější verzi dle migrací nebo ji vytvoří, pokud neexistuje. Toto zařizuje třída *MigrationStartupFilter*. Je však potřeba mít MS SQL Server. Ve Windows tento existuje nativně.

Ve výsledku běží služba na localhost:12345 a web na localhost:5001. Databáze je vytvořena v localdb pod názvem BooksDb.

Cesta ke sdílené technické zprávě na google docs:

[https://docs.google.com/document/d/1caeVDowmzpTdYR4\\_syoA0088W\\_oc0Qov6mA5ptFd71s/edit#](https://docs.google.com/document/d/1caeVDowmzpTdYR4_syoA0088W_oc0Qov6mA5ptFd71s/edit#)

## 11. Závěr

Z výsledků testů vyplývá, že uživatelé neměli problémy s používáním aplikace, ba naopak. Aplikace je zaujala a měli zájem ji využívat dále. Uživatelé především ocenili přehlednost aplikací a její jednoduchost. Většinou se jim líbilo barevné provedení a zejména design seznamu knížek na webu, v němž je i fotka a stručný popis. Dále ocenili široké možnosti filtrace.

### **xloren15**

Dle mého názoru tým pracoval výborně. Pravidelně jsme měli schůzky a společně jsme řešili nejen design, nýbrž i implementaci jednotlivých částí. Nebylo to tak, že by si každý hrál na svém písečku. Každý si tak de facto vyzkoušel vše, ať už ve větší či menší míře. Hlavními přínosy mi byly prohloubení znalostí práce na webovém api a platformě WPF. I na webu jsem ovšem pochytil pár nových detailů. Neméně přínosná byla i zkušenost s testováním aplikací uživateli. Příště bych se však více zaměřil na lepší správu času. Projekt jsme měli naplánovaný dobře a předčasné termíny měly sloužit k vytvoření rezervy pro případ, že by se nestíhalo. Tato rezerva byla využita plně, proto plán sice splnil účel, nicméně časový polštářek je dobré mít vždy.

### **xstane45**

I když jsem před projektem již měl zkušenost s tvorbou webu a webových aplikací, a to jak čistě klientských (pomocí HTML, CSS a JS), tak serverových (PHP), a zároveň zkušenost s .NET frameworkem a jazykem C#, na projektu jsem se naučil nejen jak tyto dovednosti propojit, ale i jak se více soustředit na dobrý návrh uživatelského rozhraní, a to včetně správného postupu jeho tvorby (aplikace mnou v minulosti navrhované měla GUI, která byla tvořena v podstatě během tvorby aplikace samotné, bez hlubší myšlenky a často bylo nejednotné). Díky podnětům a návrhům kamaráda (Senior Interaction Designer) z firmy Red Hat jsem se dozvěděl opravdu mnoho o testování uživatelského rozhraní a o testování jeho uživatelů.

Chtěl bych ocenit práci jednotlivců v týmu, ale i práci týmu jako celku. Každá část, na které závisel další vývoj, byla hotová včas - služba prostředkující přístupy k databázi byla hotová týden po zadání projektu, tj. pak byl dostatek času na vývoj, úpravy a testování, což vedlo k dalšímu zlepšení GUI a UX. Pro příště by se dala maximálně zlepšit vzájemná informovanost o stavu vývoje, aby byla časová rezerva využitelná i ostatními členy týmu, kteří by byli schopni v případě zpoždění člena pomoci, aniž by museli zásadně zasahovat do svých časových plánů, a celkově by týmové "nestíhání" probíhalo v menším stresu a více v klidu.

### **xkalab11**

Chtěl bych poděkovat každému členu týmu za úžasnou osobní zodpovědnost, členové týmu byli ochotni si navzájem pomáhat. Schůzky byly pravidelné a vždy dorazili všichni členové týmu. Rozdělení práce bylo spravedlivé a okruhy byly dobře rozdělené, takže málokdy musel někdo na někoho čekat. Trošku pokulhávala vzájemná informovanost ohledně stavu vývoje, takže bych příště ocenil ještě častější krátké schůzky po vzoru agilních metodik.

# Reference

<https://wpf-tutorial.com>

<https://www.dotvvm.com/docs/tutorials/introduction/latest>

<https://getbootstrap.com/docs/5.0/getting-started/introduction/>

<https://knockoutjs.com>

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

<https://www.crazyegg.com/blog/user-experience-testing/>

<https://userpeek.com/blog/user-experience-testing/>

[https://uxplanet.org/ultimate-guide-to-user-research-bed4a57d260,](https://uxplanet.org/ultimate-guide-to-user-research-bed4a57d260)

<https://www.nngroup.com/articles/first-rule-of-usability-dont-listen-to-users/>