

Katedra informatiky
Přírodovědecká fakulta
Univerzita Palackého v Olomouci

BAKALÁŘSKÁ PRÁCE

System pro evaluaci výuky na střední škole



2024

Vedoucí práce:
Mgr. Tomáš Kühn, Ph.D.

Lukáš Rybenský

Studijní program: Informatika,
Specializace: Programování a vývoj
software

Bibliografické údaje

Autor: Lukáš Rybenský
Název práce: Systém pro evaluaci výuky na střední škole
Typ práce: bakalářská práce
Pracoviště: Katedra informatiky, Přírodovědecká fakulta, Univerzita Palackého v Olomouci
Rok obhajoby: 2024
Studijní program: Informatika, Specializace: Programování a vývoj software
Vedoucí práce: Mgr. Tomáš Kühn, Ph.D.
Počet stran: 55
Přílohy: elektronická data v úložišti katedry informatiky
Jazyk práce: český

Bibliographic info

Author: Lukáš Rybenský
Title: System for Evaluation of Education
Thesis type: bachelor thesis
Department: Department of Computer Science, Faculty of Science, Palacký University Olomouc
Year of defense: 2024
Study program: Computer Science, Specialization: Programming and Software Development
Supervisor: Mgr. Tomáš Kühn, Ph.D.
Page count: 55
Supplements: electronic data in the storage of department of computer science
Thesis language: Czech

Anotace

Dosud neexistuje systém, který by se primárně zabýval evaluací výuky ze strany žáků na středních školách v České republice. Tato bakalářská práce se zaměřuje na návrh a implementaci takového systému. Při implementaci se dbalo na jednoduchost a přehlednost systému. Součástí práce bylo důkladné zjišťování uživatelských požadavků a zkoumání již existujících řešení.

Synopsis

There is still no system that would primarily focus on evaluation of education from the students' perspective in high schools in the Czech Republic. This bachelor thesis focuses on the design and implementation of such a system. During the implementation, attention was paid to the simplicity and clarity of the system. Part of the work was a thorough investigation of user requirements and examination of already existing solutions.

Klíčová slova: C#; webová aplikace; evaluace výuky; střední škola

Keywords: C#; web application; evaluation of education; high school

Děkuji vedoucímu práce Mgr. Tomáši Kührovi, Ph.D. za rady ohledně návrhu systému a za motivaci k práci. Také bych chtěl poděkovat Mgr. Radku Janoščíkovi, Ph.D. za odborné rady spojené s použitými technologiemi. Dále děkuji všem ze školy Gymnázium Hladnov a Jazyková škola s právem státní jazykové zkoušky, kteří se podíleli na vývoji a testování mého systému a především pak Mgr. Martinu Vonáškoví za vynikající spolupráci. Také děkuji své rodině, přítelkyni a svým kamarádům za neustálou podporu.

Odevzdáním tohoto textu jeho autor/ka místopřísežně prohlašuje, že celou práci včetně příloh vypracoval/a samostatně a za použití pouze zdrojů citovaných v textu práce a uvedených v seznamu literatury.

Obsah

1	Úvod	9
1.1	Motivace	9
1.2	Cíl práce	10
2	Spolupráce	11
3	Existující řešení	12
3.1	Papírové dotazníky	12
3.2	Formuláře Google a Microsoft Forms	12
3.3	Moodle	13
4	Technický popis systému	15
4.1	Použité technologie	15
4.1.1	C#	15
4.1.2	Entity Framework Core	15
4.1.3	HTML	16
4.1.4	CSS	16
4.1.5	Knihovna Bootstrap (verze 5.1.3)	16
4.1.6	Razor Pages	17
4.1.7	Javascript	17
4.1.8	SQL server	17
4.2	NuGet balíčky	18
4.2.1	Microsoft.EntityFrameworkCore	18
4.2.2	Microsoft.EntityFrameworkCore.Proxies	18
4.2.3	Microsoft.EntityFrameworkCore.SqlServer	18
4.2.4	Microsoft.EntityFrameworkCore.Design	18
4.2.5	Microsoft.EntityFrameworkCore.Tools	18
4.2.6	Microsoft.AspNetCore.Identity.EntityFrameworkCore	18
4.2.7	Microsoft.AspNetCore.Identity.UI	19
4.2.8	Rotativa.AspNetCore	19
4.2.9	MailKit	19
4.2.10	CsvHelper	19
4.2.11	Hangfire	19
4.3	Architektura systému	20
4.3.1	MVC	20
4.3.2	Výhody architektury MVC	20
4.3.3	Struktura projektu	21
4.4	Zajímavé problémy	22
4.4.1	Náhrávání informací	22
4.4.2	Registrace učitelů a žáků	24
4.4.3	Třídní skupiny a povinně volitelné předměty	25

5	Uživatelská dokumentace	29
5.1	Registrace	29
5.2	Přihlašování a odhlašování	30
5.3	Systém z pohledu administrátora	32
5.3.1	Stránky s předměty, učiteli, třídami a rozvrhovými akcemi	34
5.3.2	Tvorba dotazníků	37
5.3.3	Evaluaace	40
5.4	Systém z pohledu učitele	42
5.5	Systém z pohledu žáka	45
5.6	Obecné dodatky k systému	46
6	Nedostatky a možná rozšíření systému	47
6.1	Grafy výsledků	47
6.2	Další typy dotazníků	47
6.3	Další typy otázek	47
6.4	Pořadí otázek	47
6.5	Vizuální stránka systému	47
6.6	Nasazení aplikace	48
	Závěr	49
	Conclusions	50
	A Zpětná vazba z Gymnázia Hladnov	51
	B Obsah elektronických dat	52
	Literatura	53

Seznam obrázků

1	Příklady CSV souborů pro přidání předmětů a tříd	23
2	Příklad CSV souboru pro přidání všech učitelů	23
3	Příklad CSV souboru pro provázání učitelů s předměty a třídami .	23
4	Příklad CSV souboru pro přidání všech žáků	23
5	Příklad CSV souboru pro přidání žáků do konkrétní třídy	24
6	Příklad stránky se seznamem předmětů připravených k hodnocení, kterou žák uvidí po přihlášení do systému.	25
7	Příklad stránky se seznamem rozvrhových akcí z pohledu admi- nistrátora	26
8	Seznam předmětů připravených k ohodnocení před stisknutím še- dého tlačítka	27
9	Seznam předmětů připravených k ohodnocení po stisknutí šedého tlačítka	27
10	Potom, co žák ohodnotil první z povinně volitelných rozvrhových akcí, se zobrazí následující seznam.	28
11	Domovská stránka	29
12	Registrační formulář	30
13	Stránka s přihlašovacím formulářem	31
14	Domovská stránka po odhlášení uživatele	31
15	Stránka s přehledem pro administrátora	32
16	Stránka pro zahájení evaluace na dané škole	33
17	Stránka pro úpravu osobních údajů administrátora	34
18	Stránka se seznamem učitelů	35
19	Formulář pro úpravu osobních údajů konkrétního učitele	35
20	Stránka s třídami (bez hlavičky a navigace)	36
21	Stránka zobrazující detail konkrétní třídy	37
22	Stránka pro úpravu dotazníků administrátorem (bez hlavičky a navigace)	38
23	Část stránky pro úpravu otázek ke konkrétnímu předmětu z po- hledu administrátora	39
24	Stránka pro hledání a prohížení si výsledků z evaluací	40
25	Stránka porovnávacího módu pro administrátory	41
26	Stránka pro odstraňování evaluací	41
27	Příklad stránky s přehledem pro konkrétního učitele	42
28	Stránka pro správu účtu konkrétního učitele (bez hlavičky a na- vigace)	43
29	Stránka pro vyhledávání výsledků z evaluací pro učitele	44
30	Stránka porovnávacího módu pro učitele	44
31	Úvodní stránka pro žáka (když neprobíhá evaluace)	45
32	Stránka s dotazníkem pro konkrétní předmět a konkrétního učitele z pohledu studenta	46

Seznam zdrojových kódů

1	Příklad práce s datovým modelem v Razor Pages	17
---	---	----

1 Úvod

Evaluace výuky se zabývá hodnocením výuky a slouží k jejímu zkvalitnění. V mé práci se zabývám evaluací výuky ze strany žáků. Žák zpravidla hodnotí předmět a také vyučujícího, který tento předmět vyučoval. To však není jediná zpětná vazba, kterou mohou žáci poskytnout. Škola by se mohla také například zajímat o zkušenosti, které žáci nasbírali v průběhu celého jejich studia.

Neexistuje pouze jeden způsob, jak takovou evaluaci provádět. Evaluaci výuky lze uskutečnit s pomocí dotazníků, pohovorů se studenty, myšlenkových map ale i dalších způsobů. V mé práci se zaměřuji pouze na dotazníky, toto zaměření přináší několik výhod:

- Dotazníky jsou velmi vhodné pro tvorbu informačního systému.
- Dotazníky jsou časově výhodné jak pro studenty, tak i pro ty, kteří evaluační dotazníky vytvářejí.
- Dotazníky poskytují kvantitativní zpětnou vazbu a do jisté míry mohou poskytovat i kvalitativní zpětnou vazbu.

Evaluace výuky může přinést i problémy. Jedním z nich je zpětná vazba od žáků, která nemusí být vždy dostatečně kvalitní. Žáci totiž umí dobře popsat věci jako například dostupnost materiálů nebo klima ve třídě, ale už nemusí dobře popsat odbornou stránku výuky, jakou je například metodika. Nutno podotknout, že žáci by neměli sloužit jako jediný zdroj zpětné vazby pro hodnocení učitelů [1].

Další nevýhodou je to, že realizace jakékoliv evaluace zabere značnou porci času. Nejdříve se musí nastavit systém, jakým proběhne sběr dat od žáků, potom následuje samotný sběr dat, na závěr je třeba vyhodnotit výsledky a případně vykonat další činnosti na základě nasbíraných výsledků. Vedení školy může například vést rozhovory s jednotlivými učiteli o jejich výsledcích.

1.1 Motivace

Dosud neexistuje žádný jednoduchý a uživatelsky přívětivý systém, který by se primárně zaměřoval na realizaci evaluace na středních školách. Sice existují nějaké nástroje s kterými to lze vyřešit, ale všechny mají své nedostatky. V současné době (březen 2024) v České republice sice není povinností, aby střední školy prováděly evaluaci výuky, ale přesto některé střední školy chtějí získávat zpětnou vazbu od svých studentů.

Na Přírodovědecké fakultě Univerzity Palackého v Olomouci již existuje systém pro evaluaci výuky. Díky tomu můžou sami žáci přispívat ke zkvalitnění výuky na celé Přírodovědecké fakultě. To byl pro mě jeden z hlavních impulsů, proč vytvořit obdobný systém i pro všechny střední školy.

1.2 Cíl práce

Na základě výše uvedených důvodů jsem se rozhodl vytvořit systém, který umožní realizovat evaluaci výuky na jakékoliv střední škole v České republice. Nejprve jsem musel zjistit základní požadavky na systém. Nebylo těžké vymyslet základní funkcionalitu, ta je pospána v následujících bodech:

- Systém musí počítat se správou uživatelů včetně jejich oprávnění.
- Uživatelé musí být přiřazeni ke své škole a žádné jiné.
- Systém musí umožnit tvorbu dotazníků.
- Systém musí umožnit studentům vyplnit odpovídající dotazníky.
- Systém musí uchovávat nasbíraná data (dotazníky) od studentů a prezentovat výsledky těchto dat.
- Používání systému by mělo být pro všechny uživatele snadné a intuitivní.
- Za korektnost dat pro danou školu bude zodpovídat administrátor na dané škole. Administrátorů může být na jedné škole více než jeden.
- Systém by měl být co nejvíce obecný, aby se dal použít na všech středních školách v České republice.

Pro lepší pochopení uživatelských požadavků jsem vytvořil online dotazníky pro tři skupiny lidí: žáci, učitelé a vedení školy. Dotazníky byly anonymní a byly rozeslány především mezi střední školy v Olomouci a v Ostravě. Celkem jsem obdržel 170 odpovědí od žáků, 37 odpovědí od učitelů a 3 odpovědi od vedení škol.

Z dotazníků vyplynuly další požadavky na systém:

- Otázky k dotazníkům by měly být rozděleny do tří skupin. První skupina by měla pokrývat společné otázky pro všechny předměty. Druhá skupina by měla pokrývat společné otázky pro daný předmět (například pro matematiku). Třetí skupina by měla pokrývat jedinečné otázky, které budou moct vytvářet samotní učitelé, a které se budou vztahovat ke konkrétnímu předmětu.
- Musí být umožněno vytvářet uzavřené (ve kterých mají žáci na výběr z několika možností) i otevřené otázky (slovní komentář).
- Při prezentaci dat by měl být u vyplněných uzavřených otázek vidět také aritmetický průměr odpovědí.
- Mělo by být možné srovnávat výsledky jednoho učitele s druhým a také výsledky jednoho učitele z dvou různých období, kdy se evaluace konala. Tuto funkcionalitu by měl využívat výhradně administrátor na dané škole. Administrátor bude moct povolit nebo zakázat používání této funkcionality samotnými učiteli na dané škole.

2 Spolupráce

Při vývoji systému pro evaluaci výuky jsem spolupracoval s lidmi z vedení, učiteli a žáky ze školy Gymnázium Hladnov a Jazyková škola s právem státní jazykové zkoušky (dále jen Gymnázium Hladnov). Tito lidé se podíleli na vývoji a testování mého systému.

O této škole jsem věděl, že už dříve evaluaci výuky provozovala a mohla by mi pomoci s tvorbou mého nového systému. Lidé z vedení školy mi vyšli velmi vstřícně, když jsem je o spolupráci požádal. Ukázalo se totiž, že na škole se už evaluace výuky neprovádí, přestože je o to stále zájem. Je to z toho důvodu, že ke sbírání dat od žáků využívali Formuláře Google [2]. Důvodu, proč tento nástroj není vhodný pro evaluaci výuky, se věnuji v kapitole 3.2.

Spolupráce vypadala tak, že já jsem byl v roli vývojáře a lidé ze školy v roli zákazníka. Díky dobré komunikaci jsem byl schopný rychle zjistit, které požadavky na systém mi ještě chybí, a které jsou ty nejdůležitější.

Tato spolupráce měla i svou stinnou stránku. Ve vývoji systému jsem musel být dostatečně přizpůsobivý, jelikož se požadavky ze strany školy během vývoje měnily. To se odráží i v technologiích, které jsem použil (viz kapitola 4.1). Musel jsem také dávat dobrý pozor na to, které požadavky ze školy jsou skutečně důležité a nebudou porušovat obecnost systému. Diskutování o možných řešeních a prezentování systému také zabraly nějaký čas navíc. Dále bylo nutné zajistit pohodlné testování pro lidi z Gymnázia Hladnov.

Testování na straně Gymnázia Hladnov vypadalo následovně. Díky platformě Azure jsem byl schopný můj systém jednoduše dostat do online prostoru. Lidé ze školy pak byli schopni používat můj systém jako jakoukoliv jinou webovou aplikaci. Testovací data si vytvořili sami. Pokud někdo ze školy během testování narazil na nějaký problém, tak mě o něm hned informoval. Na mě potom bylo, abych daný problém opravil a nahrál novou verzi aplikace, kterou mohli na škole dále testovat. K debugování jsem používal i možnost vzdáleného krokování (remote debugging). To mi umožňovalo krokovat aplikaci, i když běžela online.

3 Existující řešení

Přirozeně se nabízí otázka: Proč nepoužít již existující řešení? Jak jsem již nastínil v úvodu, tak všechna stávající řešení mají nějaké nedostatky a nejsou primárně určená k evaluaci výuky na střední škole.

3.1 Papírové dotazníky

Papírové dotazníky jsou asi nejjednodušší formou realizace evaluace výuky, ale přináší spoustu problémů. Žáci jsou nuceni vyplnit fyzické papíry, což znamená, že je musí od někoho obdržet a někomu zase odevzdat. Pokud by žáci měli možnost vzít si tyto papíry domů, mohli by je ztratit nebo poškodit, proto by bylo pravděpodobné, že by žáci byli nuceni vyplnit tyto papírové dotazníky ve škole v době výuky. To má za následek dvě nepříjemné věci. Vyplňování dotazníků zabere čas, který by mohl být věnován výuce. Žáci jsou navíc pod časovým stresem, jelikož jsou nuceni vyplnit dotazníky do striktního časového limitu a už se k nim nebudou moct vrátit. Následkem toho mohou být odpovědi od žáků zkreslené a méně přínosné.

Další velkou nevýhodou je to, že výsledky z papírových dotazníků nelze žádným dobrým způsobem zpracovat a filtrovat. Pokud bychom chtěli například srovnat výsledky jednoho učitele za předchozí tři roky s jeho výsledky za aktuální rok, bylo by to náročné a nepohodlné.

Významným problémem je také samotné tištění papírů a jejich následné uchovávání. Vytisknout velké množství papírů pro všechny studenty na škole není vůbec ekonomické. Informace v papírové podobě může být těžké uchovat tak, aby se nepoškodily. Navíc bychom po nějaké době nejspíše čelili neúměrnému množství papírů, které bychom museli uchovat.

3.2 Formuláře Google a Microsoft Forms

Formuláře Google a Microsoft Forms [3] jsou nástroje, které poskytují skoro stejnou funkcionalitu. Oba slouží k tvorbě online dotazníků, sběru dat z dotazníků a umožňují export nasbíraných dat do souboru. Exportovaný soubor je potom možné otevřít buď v programu Tabulky Google [4] nebo v programu Microsoft Excel [5], záleží na zvoleném nástroji pro tvorbu dotazníků. V obou případech jde exportovaná data dobře filtrovat. V případě Formulářů Google je export možný také do *CSV souboru*. CSV je zkratka pro Comma-Separated Values. Jedná se o soubor, ve kterém jsou data oddělená *oddělovačem*. Tímto oddělovačem může být jakýkoliv znak, typicky je to znak čárky (anglicky comma).

Nespornou výhodou těchto nástrojů je jednoduchost vytváření dotazníků a následný sběr odpovědí. Při tvorbě dotazníků máme na výběr hned z několika možných typů otázek. Díky tomu lze jakýkoliv dotazník přispůsobit na míru konkrétním potřebám uživatele. Export nasbíraných dat do souboru se na první pohled také může jevit jako výhoda.

Nyní se podívejme na problémy těchto nástrojů v kontextu evaluace výuky. Hlavní problém se týká exportu dat. Data z vyplněných dotazníků můžeme exportovat do jednoho jediného souboru, pak nám po krátké době vznikne příliš velký soubor. V takovém souboru bude těžké se orientovat a navíc mohou nastat problémy spojené s výkonem, které nejsou uživatelsky přívětivé. Další problém tohoto přístupu je ten, že otázky pro všechny předměty na škole by byly úplně stejné. To přináší problém i pro studenty, kteří by museli vyplňovat tyto dotazníky. Abychom všechny dotazníky dostali do jediného souboru, musely by být koncipované tak, že žák bude muset pro každý dotazník vždy vybrat trojici (třída, učitel, předmět), kterou bude hodnotit. To je z pohledu žáka zbytečné zdržení při vyplňování.

Pokud bychom se rozhodli rozdělit výsledky z evaluačních dotazníků do více souborů (logicky podle vyučovaných předmětů na škole), vyřešili bychom problém stejných otázek pro všechny předměty. Pořád by ale hrozilo, že i tyto soubory dosáhnou neúměrné velikosti a bude obtížné s nimi pracovat. Další problém vzniká ve chvíli, kdy bychom chtěli mít několik společných otázek pro všechny předměty. Všechny takové otázky by se při tvorbě dotazníků museli opakovat, takže bychom si přidělávali práci s vytvářením dotazníků. Žáci by navíc pořád u každého dotazníku museli zadávat jejich třídu a učitele, který je daný předmět vyučoval.

3.3 Moodle

Moodle [6] je open-source software, který slouží k vytváření elektronických kurzů, online testování, distribuci studijních materiálů, komunikaci mezi žáky a učiteli, ale i k věcem jako je evidence žáků, tříd, předmětů a učitelů. Umožňuje také vytvářet dotazníky, které se dají použít pro evaluaci výuky. Tyto dotazníky nemusí vytvářet jen jeden člověk, ale může se na nich podílet jakýkoliv učitel, který k nim má přístup. Moodle umožňuje rozdělit otázky k dotazníkům na tři požadované kategorie:

- společné otázky pro všechny předměty
- společné otázky pro daný předmět
- otázky od učitelů pro daný předmět

Moodle tedy vlastně splňuje základní požadovanou funkcionalitu. Proč jej tedy nepoužít pro evaluaci výuky? Problém nastává s použitím Moodlu jako takového. Pro jeho využití je třeba instalace [7] a ta pro většinu středních škol nemusí být vůbec jednoduchá. Instalace se skládá z několika netriviálních kroků, mezi které patří i nastavení vlastního serveru a vytvoření databáze. To navíc není jediný problém Moodlu.

Další problém se týká jeho nepřehlednosti. Na Gymnázium Hladnov mají nainstalovaný systém Moodle, ale v konečném důsledku jej skoro nikdo nepoužívá. Pro většinu žáků i učitelů je Moodle nepřehledný i pro jednoduché úkoly, jako je

sdílení výukových materiálů nebo psaní online testů. To je důvod, proč se Moodle na Gymnáziu Hladnov pro evaluaci výuky nepoužívá. Většina učitelů, by potřebovala jeden den nebo i více času na zaučení, aby se naučili se systémem pořádně pracovat. Nepřehlednost Moodle pramení i z toho, že poskytuje velké množství funkcionality.

4 Technický popis systému

Mnou navržený systém jsem se rozhodl implementovat jako responzivní webovou aplikaci, a to hned z několika důvodů:

- Aplikace bude volně dostupná na internetu a uživatelé se do ní budou moct jednoduše přihlásit.
- K použití této aplikace nebude třeba žádná komplikovaná instalace.
- Lidé, kteří budou vytvářet dotazníky a zpracovávat nasbíraná data z vyplněných dotazníků mohou pracovat na notebooku, stolním počítači nebo jiném zařízení s velkou obrazovkou, zatímco studenti budou schopi vyplňovat dotazníky i z chytrého mobilního telefonu.

4.1 Použité technologie

Jako vývojové prostředí jsem si zvolil Visual Studio (2022) [8], protože obsahuje spoustu nástrojů pro vývoj webových aplikací na platformě .NET [9] (v mém případě .NET 6.0). Nejužitečnější pro mě byly funkce pro debugování: krokování a sledování hodnot proměnných. Tyto dvě funkce lze použít ve chvíli, kdy je aplikace spuštěná lokálně a díky tomu jsem mohl jednoduše a velmi efektivně sledovat případy toho, co se stane, když uživatel například klikne na nějaké tlačítko nebo odkaz.

4.1.1 C#

Veškerou logiku aplikace jsem napsal v jazyce C# [10]. C# je vysokoúrovňový objektově orientovaný programovací jazyk. Byl vyvinut společností Microsoft v roce 2000 a je založený na jazycích C++ a Java. Společnost jazyk neustále vyvíjí a udržuje. Syntaxi má podobnou jazyku C.

Součástí jazyka je *LINQ* [11] (Language Integrated Query), což je sada technologií přímo integrovaná do jazyka C#, která slouží k dotazování nad různými datovými zdroji. To například znamená, že nezáleží na tom, jestli pracujeme s kolekcí objektů nebo s relační databází. Dotaz vytvořený pomocí LINQ bude v obou případech vypadat stejně. Toho ve své práci hojně využívám. LINQ mimo jiné podporuje i lambda výrazy (anonymní funkce) a anonymní třídy, které slouží k rychlému vytvoření objektů na základě vyžádaných informací z databáze. Pokud ovšem chceme využít anonymních tříd v kombinaci s LINQ, musíme použít klíčové slovo „var“.

4.1.2 Entity Framework Core

Entity Framework Core [12] je moderní open-source ORM (object relation mapper). To znamená, že umožňuje mapovat námi definované třídy (neboli *entity*)

do tabulek v relační databázi. Tomuto postupu se říká Code-First a právě tento přístup ve své práci využívám.

Mezi hlavní výhody tohoto frameworku patří využití LINQ pro dotazy na databázi a možnost použít *migrace*. Migrace slouží k aktualizaci schématu databáze. Pomocí nich lze například měnit názvy sloupců tabulek, přidávat a odebírat sloupce z tabulek nebo přidávat a odebírat samotné tabulky. Migrace jsou velmi užitečným nástrojem pro práci s databází, obzvláště v kombinaci s Entity Framework Core.

Entity Framework Core podporuje několik databázových systémů, včetně SQL Serveru [13] (viz kapitola 4.1.8).

4.1.3 HTML

HTML [14] (HyperText Markup Language) je značkový jazyk, který slouží pro tvorbu webových stránek a popisuje statickou strukturu webové stránky. Webová stránka získává svůj význam a strukturu pomocí HTML značek (neboli tagů). Pomocí těchto značek můžeme odkazovat na konkrétní části dané stránky a nebo se lze odkazovat i na úplně jiné stránky (pomocí tagu „<a>“).

Organizace W3C [15] se stará o standardizaci jazyka. Standardizace je dobrá z toho důvodu, že webové stránky vytvořené v jazyce HTML se zobrazují v různých webových prohlížečích stejně. Navíc je také zajištěna kompatibilita mezi jednotlivými verzemi jazyka HTML.

Jazyk HTML by měl sloužit pouze pro popis struktury stránky a ne pro její vizualizaci. Odpovědnost za vizualizaci webové stránky je přesunuta především na CSS [16].

4.1.4 CSS

CSS (Cascading Style Sheets neboli kaskádové styly) je jazyk, který se používá pro úpravu vzhledu webových stránek napsaných v jazyce HTML. Díky CSS je oddělena struktura a vzhled webové stránky, takže lze upravovat vzhled stránky, aniž by se změnil HTML kód.

Jazyk CSS je založený na *selektorech* a *vlastnostech*. Selektory slouží k výběru HTML elementu a díky nim můžeme HTML elementům přiřazovat konkrétní vlastnosti ovlivňující jejich vzhled (například velikost písma, barva textu, ohraničení, pozice).

Pro vývoj webových stránek se často používají CSS knihovny. Jejich výhodou je efektivnější a rychlejší práce, než kdybychom museli psát pouze vlastní CSS.

4.1.5 Knihovna Bootstrap (verze 5.1.3)

Bootstrap [17] je knihovna CSS a Javascriptu, která slouží k efektivní tvorbě responsivních webových stránek. Obsahuje již předdefinované styly a komponenty, které mohou výrazně urychlit vývoj. Ve své práci nepoužívám část knihovny

Bootstrap, která se věnuje Javascriptu, ale pouze část knihovny zaměřující se na CSS.

4.1.6 Razor Pages

Model Razor Pages [18] je součástí ASP.NET Core a může významně ulehčit a zefektivnit práci při tvorbě webových stránek. Zásadní výhodou Razor Pages je to, že lze pracovat s jazyky C Sharp a HTML najednou v jednom souboru s příponou „.cshtml“.

V principu fungují tak, že pracují s nějakým datovým *modelem*. Abychom mohli nějaký model použít, musíme mít na začátku souboru direktivu, která říká, s jakým modelem chceme pracovat. Příklad takové direktivy může vypadat následovně: `@model Models.LoginModel`.

Tento příklad říká, že chceme pracovat s modelem, který má název „Login-Model“ a jeho definice je obsažena ve složce s názvem „Models“. S tímto modelem pak už můžeme jakkoliv pracovat, například:

Zdrojový kód 1: Příklad práce s datovým modelem v Razor Pages

```
@if (Model.Name != null)
{
    <p>Model.Name</p>
}
```

Výše uvedený zdrojový kód, znamená, že pokud vlastnost (property) Name našeho modelu není null, pak se na výsledné webové stránce zobrazí odstavec s obsahem vlastnosti Name.

Toto není jediný způsob, jak pracovat s daty a Razor Pages. Můžeme ještě použít *ViewBag*, což je dynamický objekt, který slouží pro předání dat z *kontroleru* (controller) do *pohledu* (view). Více o kontrolerech a pohledech píší v kapitole 4.3.1. Pomocí ViewBagu můžeme jednoduše předávat a získávat anonymní objekty, které se často používají při dotazování databáze pomocí LINQ.

Problémem ViewBagu je, že ztrácíme typovou kontrolu a při jeho nadužívání může dojít ke ztrátě srozumitelnosti kódu.

4.1.7 Javascript

Javascript [19] je skriptovací jazyk, který se dá využít pro psaní backendu i front-endu aplikace. Já jej ve své práci využívám pouze pro frontend. Pomocí něj můžu manipulovat s obsahem a strukturou webové stránky. Pokud uživatel například stiskne nějaké tlačítko, můžu na tuto akci reagovat s pomocí Javascriptu třeba tím, že uživateli zobrazím vyskakovací okno, které bude chtít od uživatele danou akci potvrdit. Nepoužívám při tom žádné další knihovny.

4.1.8 SQL server

SQL Server je relační databázový systém, který vyvinula firma Microsoft. SQL Server je přizpůsoben pro širokou škálu aplikací a je pevně propojen s dalšími

technologiami od Microsoftu, jako jsou .NET Framework, Visual Studio, Azure a další.

4.2 NuGet balíčky

NuGet [20] balíček představuje jeden soubor, který obsahuje ucelený software (například knihovnu nebo framework), který mohou vývojáři na platformě .NET použít ve svých projektech. Nástroje pro získání NuGet balíčků jsou již integrované ve vývojovém prostředí Visual Studio.

4.2.1 Microsoft.EntityFrameworkCore

Balíček Microsoft.EntityFrameworkCore [21] je nutný pro zprovoznění Entity Framework Core (viz kapitola 4.1.2) v aplikaci.

4.2.2 Microsoft.EntityFrameworkCore.Proxies

Microsoft.EntityFrameworkCore.Proxies [22] podporuje opožděné načítání dat z databáze. To znamená, že pokud máme entity s podřízenými entitami, při počátečním zatížení se načtou pouze nadřazené entity. Proxy servery zjistí, kdy se provede pokus o přístup k podřízeným entitám a automaticky je načte na vyžádání. Tímto způsobem balíček Microsoft.EntityFrameworkCore.Proxies optimalizuje načítání dat z databáze.

4.2.3 Microsoft.EntityFrameworkCore.SqlServer

Tento balíček propojuje Entity Framework Core (viz kapitola 4.1.2) se SQL Serverem (viz kapitola 4.1.8). Microsoft.EntityFrameworkCore.SqlServer [23] umožňuje vytvářet a spravovat databázové migrace.

4.2.4 Microsoft.EntityFrameworkCore.Design

Microsoft.EntityFrameworkCore.Design [24] se zaměřuje na návrh a modelování databázových schémat. Poskytuje nástroje k vytváření a aplikování databázových migrací. Je důležitý pro vývojový proces.

4.2.5 Microsoft.EntityFrameworkCore.Tools

Hlavním nástrojem balíčku Microsoft.EntityFrameworkCore.Tools [25] je „dotnet ef“, s pomocí něj lze spouštět příkazy pro Entity Framework Core z příkazové řádky.

4.2.6 Microsoft.AspNetCore.Identity.EntityFrameworkCore

Balíček Microsoft.AspNetCore.Identity.EntityFrameworkCore [26] zajišťuje správu uživatelských účtů pro webové aplikace využívající Entity Framework Core. Díky

němu můžeme lehce vytvářet a ukládat uživatelské účty. K ukládání uživatelských hesel používá hashovací funkci.

Balíček obsahuje třídy a metody k autorizaci uživatelů podle rolí. Řeší přihlašování a odhlašování uživatelů. Navíc podporuje vícefázové ověření (pomocí emailu nebo SMS zpráv). Pro přístup k citlivým datům využívá takzvané tokeny. Ty můžeme rozdělit do několika kategorií (například: potvrzovací, obnovovací nebo autentizační). S pomocí tokenů lze autentizovat a autorizovat uživatele a také lze zařídit to, že uživatel nemusí opakovaně zadávat své přihlašovací údaje.

4.2.7 Microsoft.AspNetCore.Identity.UI

Microsoft.AspNetCore.Identity.UI [27] obsahuje předpřipravené stránky a šablony týkající se uživatelské autentizace a autorizace. Díky němu jsem byl schopný rychle začlenit veškerou funkcionalitu ohledně uživatelů do mé aplikace.

4.2.8 Rotativa.AspNetCore

Díky knihovně Rotativa.AspNetCore [28] lze jednoduše převést HTML kód do formátu pdf. Tato knihovna je postavená na knihovně WkHtmlToPdf.

4.2.9 MailKit

MailKit [29] je open-source knihovna, která poskytuje rozhraní pro odesílání emailů, příjem emailů, uprovavání hlaviček emailů, šifrování a odesílání příloh. Pomocí něj jsem implementoval rozhraní IEmailSender. Toto rozhraní využívá balíček Microsoft.AspNetCore.Identity.UI.

4.2.10 CsvHelper

Tato open-source knihovna je určena pro čtení a zápis CSV souborů. CsvHelper [30] podporuje různé formáty CSV souborů, včetně rozdílných oddělovačů. Umožňuje také mapovat data z CSV souborů do objektů definovaných třídami v jazyce C#.

4.2.11 Hangfire

Open-source knihovna Hangfire [31] se stará o plánování úloh. Pomocí ní můžeme zavolat konkrétní úlohu v určený čas nebo zařídit, že se konkrétní úloha bude spouštět v učitém intervalu. Toho se dá využít například k aktualizaci dat v databázi.

4.3 Architektura systému

Architektura hraje zásadní roli ve vývoji, údržbě a také v budoucím rozvoji jakéhokoliv informačního systému. Pro webové aplikace existuje hned několik osvědčených vzorů architektur, které lze rovnou použít. Jednou z nich je i architektura Model-View-Controller [32] (dále jen MVC).

4.3.1 MVC

Architektura MVC rozděluje systém na tři vrstvy: *modely* (models), *pohledy* (views) a *kontrolery*¹ (controllers). Každá z těchto vrstev je odpovědná za jinou část systému. To napomáhá k udržitelnosti a organizaci systému a díky tomu lze systém v budoucnu snadněji rozšířit o další funkcionalitu. Jelikož jsou jednotlivé vrstvy oddělené, je testování systému snadnější.

Datová vrstva (vrstva modelů) je zodpovědná za veškerou logiku spojenou se zpracováním dat. To typicky obnáší přenos dat mezi pohledy a kontrolery. Pomocí této vrstvy lze rovněž načítat data z databáze nebo data do databáze přidávat. Modely by měly být nezávislé na uživatelském rozhraní a logice systému, která je obsažená v kontrolerech. Změny v modelech by tedy neměly mít vliv na ostatní části systému.

Pohledy jsou komponenty používané pro zobrazení uživatelského rozhraní. Informace v pohledech jsou často generovány na základě informací předaných z kontrolerů.

Kontrolery zajišťují logiku celého systému a především propojují modely s pohledy. Kontroler může manipulovat s daty modelů a může rozhodovat o tom, jaké informace (a jakým způsobem) budou zobrazeny v konkrétním pohledu.

4.3.2 Výhody architektury MVC

Architektura MVC přináší do mého systému i další výhody, které jsem nezmínil v předchozí kapitole. Ty se odvíjí od použitých technologií. Entity Framework Core (viz kapitola 4.1.2) mapuje objekty do relační databáze. Tímto způsobem je datová vrstva oddělena od databáze a můžu tak pracovat rovnou se samotnými objekty. Nemusím pracovat pomocí přímých dotazů na databázi.

Díky MVC architektuře jsem byl schopný velice rychle zakomponovat logiku autorizace a autentizace uživatelů do mého systému (s pomocí balíčků popsaných v kapitolách 4.2.6 a 4.2.7). Jeden celý kontroler může být označen atributem, který udává, jaký uživatel k němu má přístup. Nejenže můžu takový atribut použít pro celý kontroler, ale můžu jej použít i pro konkrétní metodu v kontroleru. Pro dvě různé metody v jednom kontroleru pak můžu mít navíc dva různé atributy přístupu. Tyto atributy podporují rozlišování uživatelů na základě jejich rolí. To jako celek vytváří silný nástroj pro řízení přístupu uživatelů.

¹Někdy se můžeme setkat i s pojmem řadič.

4.3.3 Struktura projektu

Všechn kód je rozdělen a uspořádán do složek a souborů. V kořenové složce se nachází soubory `appsettings.json` a `Program.cs`, které slouží hlavně ke konfiguraci. Složky `Models`, `Views` a `Controllers` slouží k účelům vysvětlených v kapitole 4.3.1.

Další důležitou složkou je `wwwroot`, ta obsahuje kódy CSS, Javascriptu a také obsahuje složku spojenou s knihovnou `Bootstrap` a složku spojenou s knihovnou `Rotativa.AspNetCore`.

Složka s názvem `Areas` obsahuje důležitou funkcionalitu spojenou se správou uživatelů. Byla vytvořena pomocí balíčku `Microsoft.AspNetCore.Identity.UI`. Tato složka obsahuje pohledy i funkcionalitu, která je s nimi spojená. Důvodů, proč není obsah složky `Areas` rozdělen do složek `Models`, `Views`, `Controllers`, je několik. Takové rozdělení by zabralo významný čas. Pro programátory, kteří mají zkušenosti s balíčkem `Microsoft.AspNetCore.Identity.UI` je hned patrné, co složka `Areas` obsahuje a navíc to zvyšuje čitelnost a přispívá k lepší organizaci celého projektu.

Složka `Data` obsahuje soubor `ApplicationDbContext.cs` a složku `Migrations`, ve které jsou uloženy soubory jednotlivých migrací. Soubor `ApplicationDbContext.cs` představuje třídu, která slouží k mapování entit do tabulek v databázi. Může obsahovat také další různá nastavení týkající se tabulek, které se mají vytvořit z příslušných entit. (Například může udávat, jak by se daná tabulka pro danou entitu měla jmenovat.)

Poslední důležitou složkou je složka `Services`. Ta obsahuje především implementaci rozhraní `IMailSender`, které je důležité pro odesílání emailů. Složka také obsahuje soubor `HangfireService.cs`, který slouží k definování plánovaných úloh.

4.4 Zajímavé problémy

V následující kapitole popíšu různé problémy, které při vývoji systému nastaly, a jejich řešení.

4.4.1 Náhrávání informací

Pro správný chod mého systému je nutné si pro každou střední školu udržovat následující informace:

- Jaké předměty se na dané škole vyučují?
- Jací učitelé na dané škole vyučují?
- Jaké třídy má daná škola?
- Který učitel vyučuje konkrétní předmět v konkrétní třídě?
- Jací studenti patří do konkrétní třídy?

Je jasné, že všechny tyto informace musí být poskytnuty ze strany školy. Bylo tedy nutné vymyslet, jak všechny tyto informace efektivně dostat do mého systému. Vymyslel jsem řešení v podobě CSV souborů (viz 3.2). Je jednoduché pochopit jejich princip, i když o nich člověk nic neví a jejich vytvoření je také jednoduché.

Pro každou položku z výše uvedeného seznamu bude škola moct vytvořit CSV soubor a ten nahrát do systému. Strukturu těchto souborů jsem vymýšlel ve spolupráci s Gymnáziem Hladnov tak, aby jejich vytvoření bylo co nejjednodušší za použití informačního systému Bakaláři [33]. Informační systém Bakaláři totiž používá většina škol v České republice. Vytváření CSV souborů podle vymyšlené struktury by však nemělo být těžké ani za pomoci jiných informačních systémů určených pro správu vzdělávacích procesů ve škole.

Struktura jednotlivých CSV souborů je popsána na následujících příkladech zobrazených v programu Microsoft Excel.

	A	B
1	předmět	
2	Matematika	
3	Český jazyk	
4	Dějepis	
5	Informatika	
6	Mediální výchova	

(a) Příklad CSV souboru pro přidání všech vyučovaných předmětů

	A	B
1	třída	
2	1.A	
3	3.C	

(b) Příklad CSV souboru pro přidání všech tříd

Obrázek 1: Příklady CSV souborů pro přidání předmětů a tříd.

	A	B	C
1	jméno	příjmení	email
2	Martin	Vonášek	martin.vony1@gmail.com
3	Eva	Gurná	gurna1@hladnov.cz
4	Jiří	Košťál	kostal1@hladnov.cz

Obrázek 2: Příklad CSV souboru pro přidání všech učitelů

	A	B	C	D	E
1	třída	předmět	učitel jméno	učitel příjmení	skupina
2	1.A	Matematika	Jiří	Košťál	
3	1.A	Dějepis	Martin	Vonášek	
4	1.A	Český jazyk	Eva	Gurná	
5	3.C	Matematika	Martin	Vonášek	

Obrázek 3: Příklad CSV souboru pro provázání učitelů s předměty a třídami

	A	B	C	D
1	jméno	příjmení	email	třída
2	Josef	Král	josef.kral@seznam.cz	1.A
3	Pavel	Novák	pavel.novak@hladnov.info	1.A
4	Petr	Jirák	petrjirak@gmail.com	3.C

Obrázek 4: Příklad CSV souboru pro přidání všech žáků

	A	B	C
1	jméno	příjmení	email
2	Josef	Král	josef.kral@seznam.cz
3	Pavel	Novák	pavel.novak@hladnov.info
4	John	Doe	john@gmail.com
5	Jane	Doe	jane@gmail.com

Obrázek 5: Příklad CSV souboru pro přidání žáků do konkrétní třídy

4.4.2 Registrace učitelů a žáků

Dalším problémem byl způsob, jak registrovat všechny učitele a žáky spojené s konkrétní školou. Nabízí se jednoduché řešení. Každý by se do systému registroval sám za sebe. Tím by také odpadla nutnost přidávat učitele a studenty pomocí CSV souborů. To by ale mohlo přinést nepříjemné komplikace. Učitelé by si museli zvolit třídy, ve kterých vyučují a předměty, které vyučují. Student by se potom nějakým způsobem musel přiřadit do své třídy. S tímto řešením by mohly nastat problémy v tom smyslu, že by se některý z těchto uživatelů registrovat nechtěl nebo by se přiřadil do špatné třídy a podobně.

Z výše zmíněných důvodů vypadá registrace učitelů a žáků následovně. Jakmile administrátor přidá ostatní uživatele do systému (s pomocí CSV souborů nebo klasických formulářů), vygeneruje se takto přidaným uživatelům heslo o několika náhodných znacích. Seznamy uživatelů s těmito hesly jsou pro administrátory dostupné v systému. Administrátoři by měli zajistit distribuci těchto hesel mezi uživatele a měli by uživatele informovat o tom, že jejich vygenerované heslo slouží pouze k prvnímu přihlášení do systému.

Je to tak z toho důvodu, že vygenerovaná hesla se ukládají do databáze dvakrát. Jednou jsou hesla uložena klasickým bezpečným způsobem s pomocí hashovací funkce. Podruhé jsou vygenerovaná hesla uložena v jejich čisté podobě (není použita hashovací funkce), aby k nim administrátoři měli přístup. Hesla uložena v čisté podobě se dál nijak neaktualizují. Uchovávat vygenerovaná hesla tímto způsobem je velmi nebezpečné, pokud by totiž potenciální útočník získal přístup k datům z databáze, mohl by tyto hesla vidět a zneužít.

Administrátoři jsou však upozorněni na to, aby informovali ostatní uživatele o tom, že by si hned po prvním přihlášení měli heslo změnit. Navíc je pravděpodobné, že by si uživatelé heslo změnili sami, jelikož se bude skládat z několika náhodných znaků. Všichni uživatelé si mohou změnit své heslo s pomocí své emailové adresy, aniž by byli do systému přihlášení.

Celá situace by se ale dala zvládnout mnohem lépe. Administrátoři by nemuseli mít neustále k dispozici seznam uživatelů s jejich vygenerovanými hesly. Při vygenerování hesel by stačilo poslat seznam uživatelů na emailovou adresu administrátora. Díky tomu bych nemusel ukládat vygenerovaná hesla do databáze v jejich čisté podobě a administrátor by pořád mohl poskytnout vygenerovaná

hesla ostatním uživatelům. Situace se trochu komplikuje tím, že žáky i učitele lze do systému přidávat i ručně pomocí klasického formuláře, ale to by se dalo vyřešit. Na úpravu stávajícího řešení jsem však už neměl čas.

4.4.3 Třídní skupiny a povinně volitelné předměty

Uživatel mého systému, který je zaregistrovaný jako žák, má možnost ohodnotit předměty, které jsou vyučovány v jeho třídě. Po přihlášení do systému, uvidí seznam všech předmětů, které může ohodnotit, pokud zrovna na jeho škole probíhá evaluace. (Administrátor na dané škole musí sám evaluaci zahájit, pokud chce sbírat odpovědi od žáků.)

Home icon | [Správa účtu](#) | [Odhlásit se](#)

Student Pavel Novák
Třída: 1.A

Váš seznam předmětů k evaluaci:

Název předmětu	Vyučující	Stav
Matematika	Jiří Košťál	Evaluovat
Dějepis	Martin Vonášek	Evaluovat
Český jazyk	Eva Gurná	Evaluovat

Obrázek 6: Příklad stránky se seznamem předmětů připravených k hodnocení, kterou žák uvidí po přihlášení do systému.

Problém nastává ve chvíli, kdy v jedné třídě mohou jeden předmět vyučovat dva různí učitelé (nebo i více než dva). Například žáci v jedné třídě jsou rozděleni do dvou různých skupin pro předmět Anglický jazyk a každá skupina má jiného učitele. Jedna skupina by měla ohodnotit Anglický jazyk pro prvního učitele a druhá skupina pro druhého učitele. Zároveň by nemělo být možné, aby žák z jedné skupiny ohodnotil Anglický jazyk pro oba učitele.

Podobný problém se týká i povinně volitelných předmětů. Pokud v rámci třídy existují předměty, které jsou pro žáky povinně volitelné, pak by žáci měli ohodnotit jen ty předměty, kterých se skutečně účastnili.

Řešení, které jsem v práci použil, je následující. Studenti hodnotí vždy dvojici: učitel, předmět. Které tyto dvojice hodnotí je dáno *rozvrhovými akcemi*. Rozvrhové akce říkají, že konkrétní učitel vyučuje konkrétní předmět v konkrétní třídě. Obrázek 7 ukazuje příklad, jak mohou rozvrhové akce na škole vypadat. Obrázek 3 vlastně ukazuje přidání rozvrhových akcí do mého systému.

Učitel	Předmět	Třída	Skupina [?]		
Eva Gurná	Český jazyk	1.A		Upravit	Smazat
Eva Gurná	Anglický jazyk	1.A	aj1a	Upravit	Smazat
Jiří Košťál	Anglický jazyk	1.A	aj1a	Upravit	Smazat
Jiří Košťál	Matematika	1.A		Upravit	Smazat
Martin Vonášek	Dějepis	1.A		Upravit	Smazat
Martin Vonášek	Matematika	3.C		Upravit	Smazat

[+ Přidat rozvrhovou akci](#)

Obrázek 7: Příklad stránky se seznamem rozvrhových akcí z pohledu administrátora

Ve skutečnosti tedy můžeme říct, že žák hodnotí rozvrhové akce. Ke každé rozvrhové akci lze přiřadit *skupinu*. Pomocí skupiny lze provázat povinně volitelné rozvrhové akce z jedné třídy. Žáci z dané třídy pak budou moci ohodnotit pouze jednu z povinně volitelných rozvrhových akcí. Mimo to budou moci samozřejmě ohodnotit i společné rozvrhové akce pro celou třídu. To řeší oba výše zmíněné problémy. Příklad z obrázku 7 znamená, že žáci ze třídy 1.A budou moci ohodnotit Anglický jazyk jen jednou a to buď pro Evu Gurnou, nebo pro Jiřího Košťála.

Následující obrázky dokumentují, jak bude vypadat seznam předmětů určených k ohodnocení pro všechny studenty ze třídy 1.A. Pro tento účel byly použity rozvrhové akce zachycené na obrázku 7.

📄 Váš seznam předmětů k evaluaci:

Název předmětu	Vyučující	Stav
Matematika	Jiří Košťál	Evaluovat
Dějepis	Martin Vonášek	Evaluovat
Český jazyk	Eva Gurná	Evaluovat

Vyberte **pouze jednu** z následujících možností ▾

Obrázek 8: Seznam předmětů připravených k ohodnocení před stisknutím šedého tlačítka

📄 Váš seznam předmětů k evaluaci:

Název předmětu	Vyučující	Stav
Matematika	Jiří Košťál	Evaluovat
Dějepis	Martin Vonášek	Evaluovat
Český jazyk	Eva Gurná	Evaluovat

Vyberte **pouze jednu** z následujících možností ▲

Název předmětu	Vyučující	Stav
Anglický jazyk	Eva Gurná	Evaluovat
Anglický jazyk	Jiří Košťál	Evaluovat

Obrázek 9: Seznam předmětů připravených k ohodnocení po stisknutí šedého tlačítka

☐ Váš seznam předmětů k evaluaci:

Název předmětu	Vyučující	Stav
Matematika	Jiří Košťál	Evaluovat
Dějepis	Martin Vonásek	Evaluovat
Český jazyk	Eva Gurná	Evaluovat

Vyberte **pouze jednu** z následujících možností ▾

Název předmětu	Vyučující	Stav
Anglický jazyk	Eva Gurná	Evaluace odeslána ✓
Anglický jazyk	Jiří Košťál	Nelze evaluovat

Obrázek 10: Potom, co žák ohodnotil první z povinně volitelných rozvrhových akcí, se zobrazí následující seznam.

Toto řešení není dokonalé, jelikož žák může úmyslně či neúmyslně ohodnotit nesprávnou rozvrhovou akci. V nejlepším případě by žák měl vidět pouze takové rozvrhové akce k hodnocení, kterých se skutečně účastnil a žádné jiné. (Vizuálně by tomu odpovídal obrázek 6.) Tohoto výsledku by šlo dosáhnout i s pomocí existujícího řešení skupin, bylo by ale nutné rozšířit systém i o další rozsáhlou funkcionalitu, na to ale už nezbyl čas. (Například by bylo nutné umožnit přidávat žáky do systému pomocí CSV souboru včetně skupin, do kterých patří. To by samozřejmě nebyla jediná nutná změna.)

5 Uživatelská dokumentace

V této kapitole představím stěžejní funkcionalitu mého systému z pohledu uživatelů. Systém by měl být dostatečně intuitivní na to, aby ho uživatel byl schopen používat i bez dokumentace. Zobrazení některých částí systému se může pro různé typy zařízení lišit, jelikož jsem svůj systém vytvořil jako responzivní webovou aplikaci. Pro všechny obrázky jsem se vždy snažil vybrat takové zobrazení, které nejlépe popisuje konkrétní situaci.

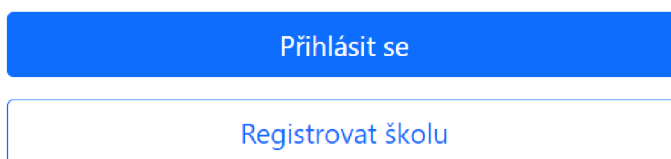
5.1 Registrace

Každá škola, která chce používat můj systém, se musí nejdříve zaregistrovat. O registraci školy by se měl zpravidla postarat někdo z vedení školy, ale nemusí tomu tak nutně být. Registraci lze udělat pomocí tlačítka „Registrovat školu“ na domovské stránce (obr. 11). To uživatele přesune na formulář (obr. 12), který je třeba vyplnit. Po validním vyplnění formuláře musí uživatel potvrdit registraci skrze svůj email. Jakmile to udělá, je informován o tom, že registrace proběhla v pořádku a může se do systému přihlásit z domovské stránky (obr. 11).

K úspěšné registraci je třeba potvrdit zadanou emailovou adresu uživatele. To lze udělat do tří hodin od odeslání registrovacího formuláře. Na emailovou adresu uživatele se odešle potvrzovací email (z emailové adresy evalupace01@gmail.com) s krze ten uživatel dokončí registraci. Pokud to uživatel nestihne nebo mu potvrzovací email z nějakého důvodu nepřijde, může kdykoliv zažádat o nový potvrzovací email skrze přihlašovací obrazovku (obr. 13).

Vítejte v aplikaci EvalUPace

Před použitím aplikace je nutné se přihlásit



Obrázek 11: Domovská stránka

Zaregistrujte Vaší školu

Vytvořte ke své škole administrátorský uživatelský účet

Obrázek 12: Registrační formulář

5.2 Přihlašování a odhlašování

Přihlašování je pro všechny uživatele stejné. Na domovské stránce (obr. 11) uživatelé kliknou na tlačítko „Přihlásit se“ a to je přesune na stránku s přihlašovacím formulářem (obr. 13). Zde si uživatelé také mohou změnit své heslo (pomocí své emailové adresy) v případě, že jej zapomněli. Po přihlášení se uživatel přesune na jemu odpovídající stránku. To je zajištěno pomocí uživatelských rolí.

Přihlaste se ke svému účtu

Zapamatovat si mě

Přihlásit se

[Zapomněli jste své heslo?](#)

[Znovu odeslat potvrzovací email](#)

Obrázek 13: Stránka s přihlašovacím formulářem

Všichni uživatelé mají možnost se ze systému odhlásit. Pokud to udělají, je jim zobrazena následující stránka:

Vítejte v aplikaci EvalUPace

Před použitím aplikace je nutné se přihlásit

Byli jste úspěšně odhlášeni

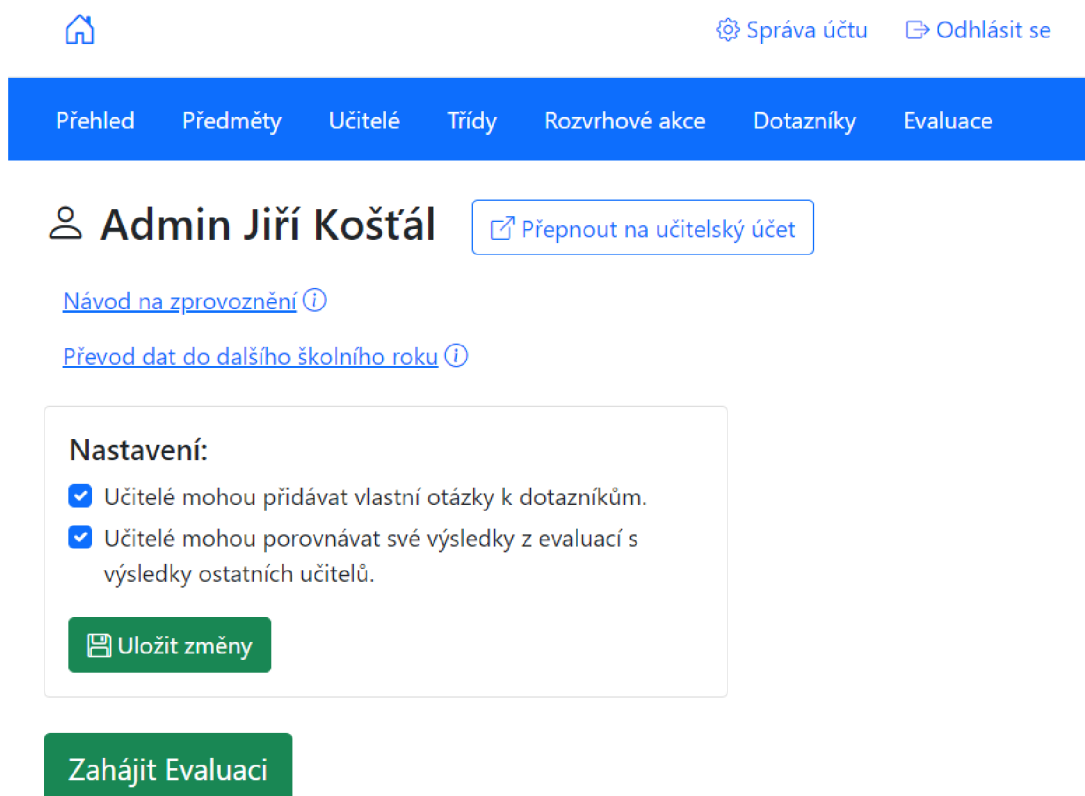
Přihlásit se

[Registrovat školu](#)

Obrázek 14: Domovská stránka po odhlášení uživatele

5.3 Systém z pohledu administrátora

Když se do systému přihlásí administrátor, uvidí následující stránku s přehledem:



Obrázek 15: Stránka s přehledem pro administrátora

Pokud je administrátor také učitelem na dané škole, uvidí vedle svého jména i odkaz s popisem „Přepnout na učitelský účet“. Tento odkaz přesměruje administrátora na stránku s přehledem pro učitele (obr. 27). Pokud administrátor není zároveň učitelem, pak odkaz „Přepnout na učitelský účet“ neuvidí. Dále se na stránce nacházejí dva informační odkazy, pomocí kterých se administrátoři mohou dozvědět důležité informace o systému. Následuje formulář týkající se nastavení systému pro danou školu a tlačítko „Zahájit Evaluaci“. Toto tlačítko přesune administrátora na následující stránku:

Evaluaci můžete pojmenovat předtím, než ji zahájíte.

Podle jejího jména a data zahájení ji budete schopni vyhledat.

Zahájit Evaluaci

⚠ Upozornění:

Před zahájením byste měli zkontrolovat, že všechny Vaše údaje v aplikaci (žáci, rozvrhové akce, dotazníky, atd.) korespondují s aktuálními údaji na Vaší škole. Pokud zahájíte evaluaci, nebude možné do jejího ukončení provádět následující akce:

upravovat otázky k dotazníkům

odstraňovat rozvrhové akce

odstraňovat/upravovat žáky, kteří již provedli evaluaci

přidávat objekty pomocí csv souborů

← Zpět na přehled

Obrázek 16: Stránka pro zahájení evaluace na dané škole

Na této stránce je administrátor informován o tom, jaké akce nebude moci po zahájení evaluace vykonávat. Stisknutím tlačítka „Zahájit Evaluaci“ na stránce z obrázku 16 dojde k tomu, že žáci budou moci vyplňovat předpřipravené dotazníky k jednotlivým předmětům (Pokud evaluace není zahájena, tak nemohou). Poté je administrátor přesměrován na stránku přehledu (obr. 15) a tlačítko „Zahájit Evaluaci“ na této stránce se změní na tlačítko „Ukončit Evaluaci“. Po stisknutí tlačítka „Ukončit Evaluaci“ se ukončí sběr odpovědí od žáků a ti už nebudou moci poslat žádné další odpovědi, dokud se evaluace opět nezahájí.

Hlavička (bílé pole s modrými tlačítky) a navigace (modré pole s bílými texty) jsou součástí téměř všech stránek, ke kterým se administrátor může dostat. V hlavičce stránky jsou tři tlačítka. Tlačítko vlevo nahoře (ikona domu) přesune administrátora na stránku přehledu (obr. 15), To samé dělá i odkaz v navigaci s nápisem „Přehled“. Tlačítko „Odhlásit se“ slouží k odhlášení uživatele. Tlačítko „Správa účtu“ přesune uživatele na stránku, kde může změnit své osobní údaje včetně emailu a hesla (obr. 17).

Tady můžete změnit své dosavadní údaje:

Jméno Jiří
Příjmení: Košťál
Email kostal@hladnov.cz

Aktualizovat údaje

[✎ Změnit heslo](#)

[← Zpět na přehled](#)

Obrázek 17: Stránka pro úpravu osobních údajů administrátora

5.3.1 Stránky s předměty, učiteli, třídami a rozvrhovými akcemi

Všechny stránky týkající se předmětů, učitelů, tříd a rozvrhových akcí mají podobnou strukturu. Všechny obsahují hlavičku a navigaci z obrázku 15, seznam týkající se daných objektů a formulář pro nahrání daných objektů pomocí CSV souboru. Tyto objekty lze zpravidla vytvářet, upravovat a odstraňovat. (Případně lze s nimi provádět i další akce.) Pro upravování, mazání a případné další akce má každý objekt odpovídající tlačítko. Výše zmíněnou strukturu ukazuje příklad stránky s učiteli (obr. 18) nebo také příklad stránky rozvrhových akcí (obr. 7) z kapitoly 4.4.3. Stránky lze přepínat pomocí výše zmíněné navigace.



Tady jsou učitelé na Vaší škole:

Jméno	Email	Vygenerované heslo	Práva
Eva Gurná	gurna@hladnov.cz	heslo1234	Přidat admina Upravit Odstranit
Jiří Košťál	kostal@hladnov.cz	heslo1234	Odebrat admina Upravit Odstranit
Lukáš Rybenský	rybensky@upol.cz	heslo1234	Přidat admina Upravit Odstranit
Martin Vonášek	martin.vony@gmail.com	heslo1234	Odebrat admina Upravit Odstranit

[+ Přidat učitele](#)[Vytisknout seznam](#)

Přidejte učitele pomocí csv souboru:

[Vybrat soubor](#) Soubor nevybrán

[Nahrát csv soubor](#)

Obrázek 18: Stránka se seznamem učitelů

Upravte následující údaje o učiteli:

Jméno
Eva

Příjmení:
Gurná

Email
gurna@hladnov.cz

[Aktualizovat údaje](#)

[← Zpět na přehled učitelů](#)

Obrázek 19: Formulář pro úpravu osobních údajů konkrétního učitele

Tlačítko „Vytisknout seznam“ na stránce učitelů (obr. 18) umožňuje vytisknout seznam učitelů s jejich vygenerovanými hesly. Funkce tlačítka závisí na nastavení výchozí tiskárny na zařízení uživatele. Pokud výchozí tiskárnou uživatele není fyzické zařízení, uživateli se stáhne soubor s výše zmíněným seznamem ve formátu PDF. Jinak se rovnou zobrazí náhled k tisku.

Formulář (obr. 19) se zobrazí po kliknutí na tlačítko „Upravit“ u konkrétního učitele. Obdobný formulář se zobrazí i po kliknutí na tlačítko „Přidat učitele“. Formuláře pro přidávání a úpravu objektů daného typu (předmět, učitel, třída, rozvrhová akce) se samozřejmě odlišují.

Stránka obsahující seznam tříd (obr. 20) se lehce odlišuje tím, že zahrnuje i formulář pro přidání všech žáků na škole pomocí CSV souboru. Pokud na škole zrovna probíhá evaluace, pak seznam tříd obsahuje i informaci o tom, kolik žáků z dané třídy evaluaci ještě neudělalo. Navíc pro každou třídu je možné zobrazit její detail tlačítkem „Seznam žáků“. Na obrázku 21 je pak vyobrazen příklad detailu konkrétní třídy.

Tady jsou třídy na Vaší škole:

The screenshot shows a web interface for managing classes. At the top, there is a light blue header with the text "Název třídy". Below this, there is a table with two rows of class information:

Název třídy		
1.A	Seznam žáků	🗑️
3.C	Seznam žáků	🗑️

Below the table, there is a green button labeled "+ Přidat třídu".

Underneath, there is a section titled "Přidejte třídy pomocí csv souboru:". It contains a text input field with the placeholder "Vybrat soubor" and "Soubor nevybrán". Below the input field is a blue button labeled "Nahrát csv soubor".

Below this section, there is another section titled "Přidejte všechny studenty pomocí csv souboru:". It also contains a text input field with the placeholder "Vybrat soubor" and "Soubor nevybrán". Below the input field is a blue button labeled "Nahrát csv soubor".

Obrázek 20: Stránka s třídami (bez hlavičky a navigace)

← Zpět na přehled tříd

Seznam žáků ve třídě 1.A:

Jméno	Email	Vygenerované heslo		
Josef Král	josef.kral@seznam.cz	heslo1234	Upravit	
Pavel Novák	pavel.novak@hladnov.info	heslo1234	Upravit	

[+ Přidat žáka](#) [Vytisknout seznam](#)

Přidejte studenty do třídy 1.A pomocí csv souboru:

Soubor nevybrán

[Nahrát csv soubor](#)

Obrázek 21: Stránka zobrazující detail konkrétní třídy

5.3.2 Tvorba dotazníků

Na tvorbě a upravování dotazníků se podílí především administrátoři, ale i učitelé mohou do jisté míry dotazníky upravovat. O tom, jak učitelé mohou upravovat dotazníky, píší v kapitole 5.4.

Na stránku s dotazníky (obr. 22) se administrátor dostane kliknutím na odkaz v navigaci (obr. 15) s názvem „Dotazníky“. Na této stránce se nachází odkazy v podobě tlačítek (s nápisy „Vytvořit otázku“, „Upravit otázku“ a podobně). Tyto tlačítka přemístí administrátora na stránku, kde může vytvářet a upravovat otázky. Lze upravovat jak společné otázky ke všem předmětům, tak otázky specifické pro daný předmět. Společné otázky pro všechny předměty se upravují zvlášť a jejich změny se projeví v dotaznících pro všechny předměty. Mimo to je k dispozici ke každému předmětu náhled toho, jak výsledný dotazník pro daný předmět bude vypadat.













Tady je přehled Vašich dotazníků.

Společné otázky pro všechny dotazníky.

Tyto otázky žáci uvidí při hodnocení každého předmětu.

 [Upravit společné otázky](#)

Vytvořte otázky pro konkrétní předmět:

Název předmětu	Možnosti
Anglický jazyk	 Vytvořit otázky  Náhled
Český jazyk	 Upravit otázky  Náhled
Dějepis	 Vytvořit otázky  Náhled
Informatika	 Vytvořit otázky  Náhled
Matematika	 Upravit otázky  Náhled
Mediální výchova	 Vytvořit otázky  Náhled

Obrázek 22: Stránka pro úpravu dotazníků administrátorem (bez hlavičky a navigace)

Následující obrázek nastiňuje, jak může vypadat samotná stránka pro úpravu otázek konkrétního předmětu. Na této stránce se zobrazují i informace o tom, jaké otázky jsou již součástí dotazníku pro daný předmět. To mohou být společné otázky pro všechny předměty nebo také otázky vytvořené učiteli.

Upravte otázky k předmětu Český jazyk.

Nezapomeňte, že součástí dotazníku pro předmět Český jazyk jsou i společné otázky (2):

Chodíte radi do skoly?


Jak se dnes mate?

Přidejte nebo upravte otázky:


Máte rádi Český jazyk?

Vyberte typ otázky:

Povinná

 Smazat otázku

+ Přidat další otázku

 Uložit úpravy

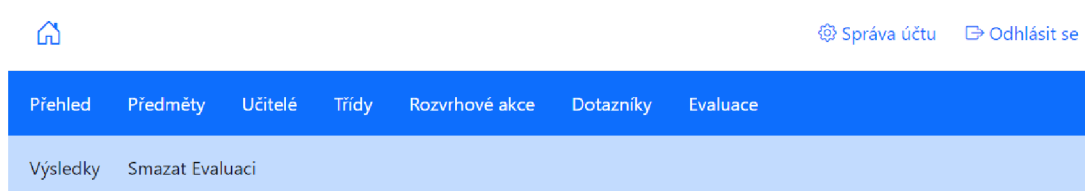
Obrázek 23: Část stránky pro úpravu otázek ke konkrétnímu předmětu z pohledu administrátora

Stránka pro úpravu společných otázek pro všechny předměty vypadá velmi podobně jako stránka pro úpravu otázek ke konkrétnímu předmětu (obr. 23). Aby uživatel vytvořil otázku, tak musí stisknout tlačítko „Přidat další otázku“. Toto tlačítko vygeneruje formulář do kterého musí administrátor zadat text otázky, typ otázky a zaškrtnout, jestli bude vyplnění této otázky pro žáky povinné nebo ne. Existují tři základní typy otázek: otázka typu ano nebo ne, uzavřená otázka s možnostmi jedna až pět, otevřená otázka (studenti mohou napsat komentář). Jakékoliv vytvořené otázky je možné smazat. Pro uložení otázek je nutné kliknout na tlačítko „Uložit úpravy“, jinak se provedené změny neuloží.

5.3.3 Evaluace

Pokud chce administrátor manipulovat s proběhlými evaluacemi, musí v navigaci (obr. 15) kliknout na odkaz „Evaluace“. Poté se mu objeví stránka, na které je možné vyhledávat a prohlížet si výsledky (odpovědi od studentů) z evaluací (obr. 24). Navíc se po kliknutí na odkaz „Evaluace“ objeví podnavigace s odkazy „Výsledky“ a „Smazat Evaluaci“. Odkaz „Výsledky“ přemístí administrátora na tu samou stránku, která se zobrazí po kliknutí na odkaz „Evaluace“ (obr. 24).

Pomocí formuláře na následujícím obrázku lze vyhledávat výsledky z evaluací. Pokud uživatel odeslal tento formulář, pak se mu na té samé stránce zobrazí i výsledky pro zadané hodnoty (pod formulářem). Tyto výsledky lze i vytisknout pomocí tlačítka „Tisk výsledků“. To může být užitečné například pro pohovory s učiteli ohledně jejich výsledků.



Tady si můžete prohlížet výsledky z evaluací.

Vyberte evaluaci:

Vyberte učitele:

Vyberte předmět:

Všechny třídy 1.A 3.C

Obrázek 24: Stránka pro hledání a prohlížení si výsledků z evaluací

Nad formulářem je také tlačítko „Porovnávací mód“. To přenese uživatele na stránku porovnávacího módu (obr. 25). Porovnávací mód slouží k porovnávání výsledků ze dvou formulářů, které mají stejnou podobu jako na obrázku 24. Načtené výsledky z těchto dvou formulářů se zobrazí vedle sebe. Tento mód je však přístupný pouze pro dostatečně velké obrazovky (právě proto, že výsledky pro porovnání se zobrazují vedle sebe).

← Klasický mód

Vyberte evaluaci:
Všechny evaluace

Vyberte učitele:
Všichni učitelé

Vyberte předmět:
Anglický jazyk

Všechny třídy 1.A 3.C

Najít Výsledky

Obrázek 25: Stránka porovnávacího módu pro administrátory

Administrátor může ještě kliknout na odkaz „Smazat Evaluaci“ v podnavigaci. Ten přemístí administrátora na stránku, kde může smazat již proběhlé evaluace. Smázání evaluace může být užitečné například z důvodu testování.

Smazáním evaluace přijdete o všechna její nasbíraná data.

Vyberte evaluaci, kterou chcete smazat:

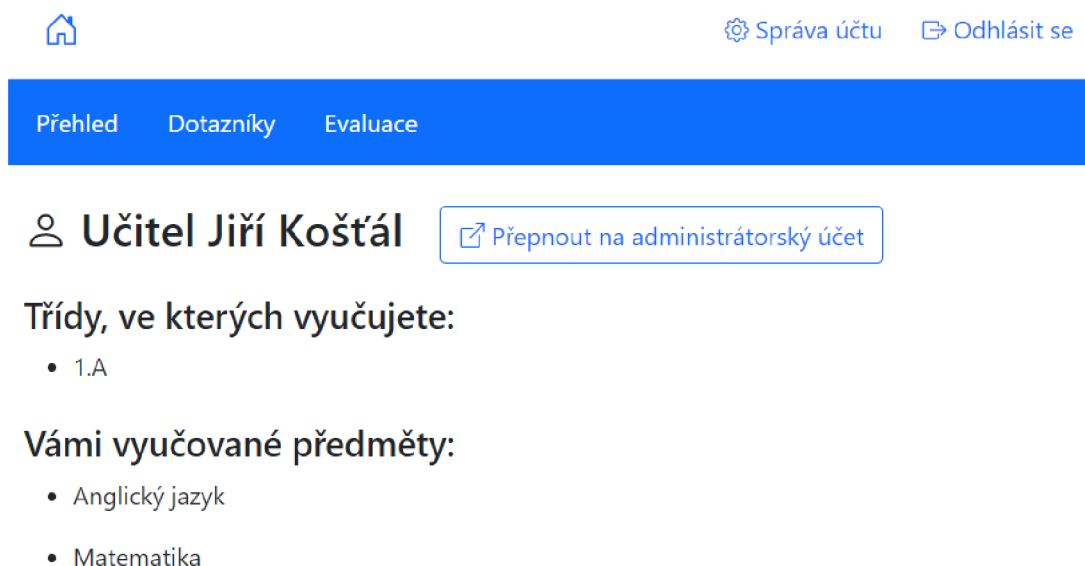
[Beze jména] - 10.4.2024, 13:47

Smazat evaluaci

Obrázek 26: Stránka pro odstraňování evaluací

5.4 Systém z pohledu učitele

Pokud je uživatel učitelem a zároveň administrátorem na dané škole, zobrazí se mu stránka přehledu pro administrátora (obr. 15), ze které se může dostat na stránku přehledu pro učitele 27. Pokud učitel není zároveň administrátorem, automaticky se mu zobrazí stránka přehledu pro učitele.



Obrázek 27: Příklad stránky s přehledem pro konkrétního učitele

Učitel uvidí tlačítko „Přepnout na administrátorský účet“ vedle svého jména jen v případě, pokud je zároveň i administrátorem na dané škole. Dále učitel vidí seznam tříd, ve kterých vyučuje a seznam předmětů, které vyučuje.

Hlavička a navigace jsou podobné hlavičce a navigaci administrátora (obr. 15). Tlačítko vlevo nahoře (ikona domu) přesune učitele na stránku přehledu (obr. 27). To samé dělá i odkaz „Přehled“ v navigaci. Tlačítko „Odhlásit se“ slouží k odhlášení učitele. Tlačítko „Správa účtu“ přesune učitele na stránku, kde může změnit svůj email a svoje heslo (obr. 28).

Osobní údaje


Celé jméno: Jiří Košťál

Třídy, ve kterých vyučujete:

- 1.A

Vámi vyučované předměty:

- Anglický jazyk
- Matematika

 Pokud některé Vaše údaje neodpovídají, kontaktujte administrátora na Vaší škole.

Email
kostal@hladnov.cz

Aktualizovat Email

[Změnit heslo](#)

Obrázek 28: Stránka pro správu účtu konkrétního učitele (bez hlavičky a navigace)

Funkcionalita pro úpravu dotazníků a vyhledávání výsledků z evaluací funguje velmi podobně jako pro administrátora (viz kapitoly 5.3.2, 5.3.3), ale pro učitele existují určitá omezení.

Učitel nemůže jakkoliv upravovat společné otázky pro všechny předměty. Může upravovat otázky pouze pro ty předměty, které on sám vyučuje.

Když učitel klikne na odkaz „Evaluace“ v navigaci (obr. 27), pak se mu nezobrazí žádná podnavigace, pouze je přemístěn na stránku, kde může vyhledávat a prohlížet si své výsledky z evaluací. Ve vyhledávacím formuláři (obr. 29) navíc učitel chybí možnost „Vyberte učitele“ (obr. 24). Učitel si tak může zobrazit pouze ty výsledky, které souvisí přímo s ním.

Učitelům může být administrátorem zakázáno používat porovnávací mód, ale pokud je to povoleno, pak má porovnávací mód v případě učitelů také jistá omezení. Formulář na levé straně neobsahuje možnost „Vyberte učitele“ (stejně jako v klasickém vyhledávacím módu pro učitele). Formulář vpravo sice vypadá stejně jako na obrázku 24, ale možnosti, ze kterých lze vybírat učitele, předmět a třídy, jsou omezené. Učitel může vybírat pouze takové třídy, ve kterých vyučuje a předmět, který vyučuje. Ze všech učitelů na škole může vybírat pouze ty, kteří vyučují stejné předměty. Opět platí, že výsledky z odeslaných formulářů se zobrazí vedle sebe, a že porovnávací mód je dostupný jen pro dostatečně velké obrazovky.

[🏠](#) [Správa účtu](#) [Odhlásit se](#)

[Přehled](#) [Dotazníky](#) [Evaluace](#)

Tady si můžete prohlížet výsledky z evaluací.

[📊 Porovnávací mód](#)

Vyberte evaluaci:
Všechny evaluace ▾

Vyberte předmět:
Anglický jazyk ▾

Všechny třídy 1.A

[Najít Výsledky](#)

Obrázek 29: Stránka pro vyhledávání výsledků z evaluací pro učitele

[🏠](#) [Správa účtu](#) [Odhlásit se](#)

[Přehled](#) [Dotazníky](#) [Evaluace](#)

Tady si můžete prohlížet výsledky z evaluací.

[← Klasický mód](#)

Vyberte evaluaci:
Všechny evaluace ▾

Vyberte předmět:
Anglický jazyk ▾

Všechny třídy 1.A

[Najít Výsledky](#)

Vyberte evaluaci:
Všechny evaluace ▾

Vyberte učitele:
Všichni učitelé ▾

Vyberte předmět:
Anglický jazyk

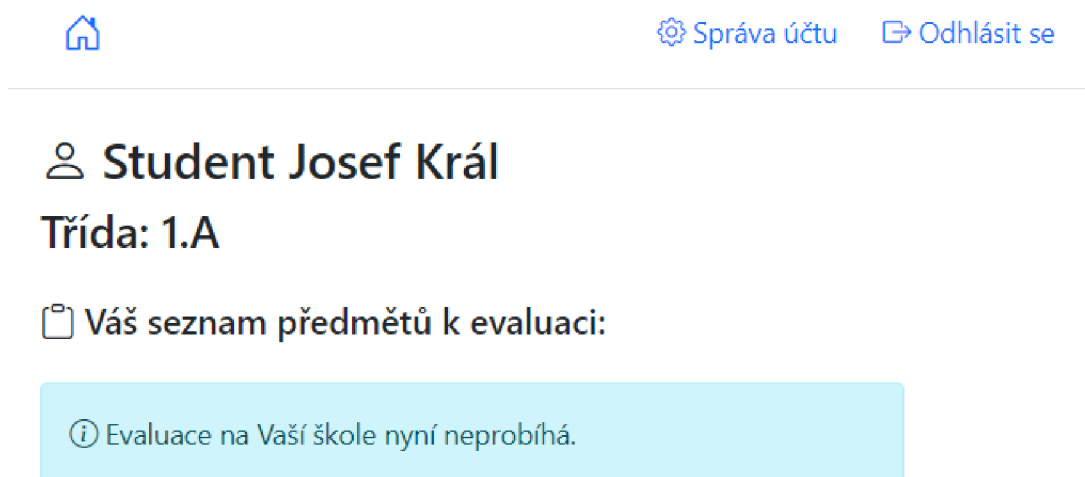
Všechny třídy 1.A

[Najít Výsledky](#)

Obrázek 30: Stránka porovnávacího módu pro učitele

5.5 Systém z pohledu žáka

Pokud se žák přihlásí do systému ve chvíli, kdy na jeho škole evaluace neprobíhá, je mu zobrazena následující stránka:



Obrázek 31: Úvodní stránka pro žáka (když neprobíhá evaluace)

Tlačítko s ikonou domu v hlavičce přesune žáka na úvodní stránku, která je žákovi zobrazena, když se přihlásí do systému (obr. 31 nebo 6). Tlačítko „Odhlásit se“ slouží k odhlášení žáka ze systému. Tlačítko „Správa účtu“ přesune žáka na stránku, kde může změnit svůj email a svoje heslo. Tato stránka vypadá velmi podobně jako stránka pro správu účtu učitele (obr. 28).

Pokud se žák do systému přihlásí, když evaluace probíhá, pak uvidí seznam předmětů, které může ohodnotit. Tuto situaci dokumentují obrázky 6, 8, 9, 10 z kapitoly 4.4.3. Jakmile žák klikne na tlačítko „Evaluovat“ u konkrétního předmětu, zobrazí se mu stránka s dotazníkem k vyplnění. Následující obrázek ukazuje příklad stránky s dotazníkem pro předmět Český jazyk a pro vyučujícího Evu Gurnou.

[← Zpět na přehled](#)

Předmět: **Český jazyk** Vyučující: **Eva Gurná**

Vyplňte prosím následující otázky:

Chodíte rádi do školy?

* Povinná otázka

Vaše odpověď:

- Ano
 Ne

Jak moc máte rádi Český jazyk? (1 - nejvíce, 5 - nejméně)

Nepovinná otázka

Vaše odpověď:

- 1 2 3 4 5

[Odeslat dotazník](#)

Obrázek 32: Stránka s dotazníkem pro konkrétní předmět a konkrétního učitele z pohledu studenta

Formulář z obrázku 32 mohou žáci odeslat pouze, pokud odpověděli na všechny povinné otázky. Pokud žák neodpověděl na všechny povinné otázky, formulář se neodešle a žák je na to upozorněn. Zároveň všechny odpovědi, které žák měl vyplněné, zůstávají vyplněné i při špatném pokusu o odeslání formuláře.

5.6 Obecné dodatky k systému

Za správnost údajů na dané škole je zodpovědný administrátor pro danou školu. Pokud nějaký učitel nebo žák, zjistí, že jeho údaje neodpovídají, měl by o tom informovat nějakého administrátora z dané školy.

Všechny formuláře v systému jsou opatřeny bezpečnostními prvky, aby uživatelé nevedli systém do nekonzistentního stavu. Kontrolují se také přístupová práva uživatelů ke všem stránkám v systému. Není tedy možné, aby se například žák dostal ke stránce administrátora, na které lze vyhledávat výsledky z evaluací.

6 Nedostatky a možná rozšíření systému

V následující kapitole popíši, jak lze můj systém dále rozšířit. Tato rozšíření vychází především z požadavků od Gymnázia Hladnov. Zmiňuji pouze ty nejpodstatnější rozšíření. Určitě by se dala vymyslet i další rozšiřující funkcionalita. Všechny problémy, které v této kapitole popisuji, by se daly v mém systému nějakým způsobem vyřešit. Z časových důvodů jsem je však již vyřešit nedokázal.

6.1 Grafy výsledků

Momentálně se výsledky z evaluací zobrazují ve formě tabulek. Mnohem lepším řešením by však bylo, kdyby se výsledky zobrazovaly ve formě grafů. Například sloupcové grafy by byly velmi vhodné.

6.2 Další typy dotazníků

Můj systém se zaměřuje pouze na případ, kdy žák hodnotí předmět a daného učitele. Škola by ale mohla od žáka požadovat i jinou zpětnou vazbu (viz kapitola 1). Bylo by tedy vhodné přidat do systému možnost vytvořit dotazníky nezávislé na předmětu a učiteli. Je dobré poznamenat, že můj systém už teď obsahuje spoustu částí, které by se pro tuto funkcionalitu daly využít.

6.3 Další typy otázek

Systém by se rovněž mohl rozšířit o další typy otázek kromě tří základních (viz kapitola 5.3.2). Tomu, jak by nový typ otázky vypadal, nejsou kladeny téměř žádné meze. Mohli bychom se inspirovat například aplikací Formuláře Google.

6.4 Pořadí otázek

Pořadí otázek v současné verzi systému nejde po jejich vytvoření změnit. To je z uživatelského pohledu velmi nepříjemné. Pro změnu pořadí otázek můj systém obsahuje všechny potřebné komponenty, kromě uživatelského rozhraní.

6.5 Vizuální stránka systému

Vizuální stránka systému není příliš zajímavá, to je zapříčiněno i tím, že jsem použil CSS knihovnu Bootstrap (viz 4.1.5). Navíc bychom mohli chtít vylepšit jednotlivé části systému. Například formulář pro tvorbu otázek bychom mohli změnit do podoby, jakou mají Formuláře Google nebo Microsoft Forms. Možnost k vylepšení vizuální stránky systému je opravdu hodně. Patří mezi ně i výše zmiňované grafy výsledků.

6.6 Nasazení aplikace

Můj systém nyní ještě není dostupný online. Na vhodném řešení, jak zpřístupnit můj systém všem středním školám, pracuji ve spolupráci s Gymnáziem Hladnov.

Závěr

V rámci této bakalářské práce jsem zjišťoval hlavní požadavky na systém pro evaluaci výuky, navrhl jsem řešení v podobě webové aplikace a toto řešení jsem implementoval. Aplikaci jsem vyvinul tak, aby ji bylo možné používat na všech typech zařízení. Na vývoji a především na testování se podílela škola Gymnázium Hladnov. Díky zpětné vazby z této školy jsem byl schopný lépe porozumět požadavkům na evaluační systém. Zpětná vazba ze školy ohledně naší spolupráce je obsažena v příloze [A](#).

Zjišťovat požadavky na systém, navrhnout systém, implementovat jej, vylepšovat systém na základě zpětné vazby, testování a opravování systému bylo časově velmi náročné. Z toho pramení, že systém má své nedostatky a není zprovozněn online, nicméně obsahuje tu nejdůležitější funkcionalitu pro realizaci evaluace výuky na střední škole. V budoucnu plánuji na systému dále pracovat a vylepšovat jej na základě nedostatků zmíněných v kapitole [6](#).

Conclusions

As a part of this bachelor's thesis, I identified the main requirements for an education evaluation system, designed a solution in the form of a web application, and implemented this solution. I have developed the application to be usable on all types of devices. The school Gymnázium Hladnov participated in the development and, particularly, in testing. Thanks to the feedback from this school, I was able to better understand the requirements for the evaluation system. The feedback from the school regarding our collaboration is included in Appendix [A](#).

The process of gathering system requirements, designing the system, implementing it, improving the system based on feedback, testing and fixing the system was time consuming. It follows from this that the system has its shortcomings and is not operational online, however, it contains the most crucial functionality for the implementation of education evaluation in high school. In the future, I plan to continue working on the system and improving it based on the deficiencies mentioned in Chapter [6](#).

A Zpětná vazba z Gymnázia Hladnov

Tato příloha obsahuje pohled ze školy Gymnázium Hladnov na můj vytvořený systém. Autorem textu je učitel Mgr. Martin Vonášek, který zajišťuje organizaci evaluace na škole.

„Plánování vzniku evaluační aplikace, která by umožnila naší škole získat alespoň rámcovou představu o tom, jak studenti vnímají průběh a kvalitu výuky jednotlivých předmětů, nebylo věcí novou. Již delší dobu jsme ve škole používali dostupné nástroje Google Forms, ale získaná data bylo jednak obtížné vyhodnotit (pomocí kontingenčních tabulek) a zároveň celý evaluační dotazník působil robustně a nepřehledně (Studenti museli vždy sami ‘naklikat’, koho chtějí hodnotit, jaký předmět chtějí hodnotit a následně toto provádět celé znovu při hodnocení dalšího předmětu.). Od pravidelného celoškolního evaluování jsme tedy postupně upustili.

V rámci prvního testování nového systému během výuky na naší škole se ukázalo, že aplikace je pro studenty snadno dostupná (snadné přihlašování), přehledná a srozumitelná. Z pohledu administrátora vítám snadnou tvorbu studentských účtů (CSV soubory) a přehledné prostředí. Zvláště sekce ‘Rozvrhové akce’ umožňuje snadno nastavit, co a koho mají žáci hodnotit.

Během další spolupráce bychom rádi uvítali možnost vytvářet evaluační dotazníky nejen pro celou třídu, ale i pro skupiny (semináře, půlené hodiny), což zatím není úplně ideální. Dále bychom chtěli mít možnost vytvářet nezávislé dotazníky, které nesouvisí pouze s evaluací předmětů a učitelů. Stejně tak plánujeme rozvíjet výstupní podobu získaných dat (grafy, přehledné tabulky) a také další možnosti zobrazení jednotlivých otázek pro studenty (posuvníky na škále, výběr z vícero možností). Vzhledem k tomu, jak mohou učitelé vytvářet otázky a opakovaně je využívat, bychom ocenili i případný systém verzování jednotlivých otázek. Také by mohlo být vhodné otázky seskupovat do vlastních složek.

Věříme, že ve spolupráci s Lukášem budeme pokračovat a budeme moci aplikaci a její vývoj dále připomínkovat. V takové situaci jsme ochotni za budoucí verzi aplikace i platit a myslíme, že kvalitně postavená aplikace pro evaluaci výuky, která umí vzít v potaz požadavky učitelů, by byla většinou škol vítána. “

B Obsah elektronických dat

Součástí práce jsou také elektronická data v systému katedry informatiky. Jejich struktura je následující.

text/

Adresář s textem práce ve formátu PDF, vytvořený s použitím závazného stylu KI PřF UP v Olomouci pro závěrečné práce, včetně všech příloh, a všechny soubory potřebné pro bezproblémové vytvoření PDF dokumentu textu (případně v ZIP archivu), tj. zdrojový text textu a příloh, vložené obrázky, apod.

src/

Složka, která obsahuje všechny soubory a adresáře nutné ke zprovoznění aplikace. Adresářová struktura projektu je popsána v kapitole 4.3.3. Součástí jsou všechny zdrojové kódy, které webová aplikace používá.

README.txt

Je textový soubor, který detailně popisuje, jak spustit aplikaci lokálně.

data/

Tato složka obsahuje jednoduché příklady CSV souborů, které mohou být podle potřeby využity při posouzení práce.

U veškerých cizích obsažených materiálů jejich zahrnutí dovoluují podmínky pro jejich veřejné šíření nebo přiložený souhlas držitele práv k užití. Pro všechny použité (a citované) materiály, u kterých toto není splněno a nejsou tak obsaženy, je uveden jejich zdroj, např. webová adresa, v bibliografii nebo textu práce nebo souboru `README.txt`.

Literatura

- [1] Brinko, Kathleen T. The Practice of Giving Feedback to Improve Teaching: What Is Effective? *The Journal of Higher Education* [online]. 1993, vol. 64, no. 5 [cit. 2024-4-4]. Dostupný z: <https://doi.org/10.2307/2959994>.
- [2] Google. *Formuláře Google* [online]. [cit. 2024-4-4]. Dostupný z: <https://www.google.cz/intl/cs/forms/about/>.
- [3] Microsoft. *Microsoft Forms* [online]. [cit. 2024-4-4]. Dostupný z: <https://www.microsoft.com/cs-cz/microsoft-365/online-surveys-polls-quizzes>.
- [4] Google. *Tabulky Google* [online]. [cit. 2024-4-4]. Dostupný z: <https://www.google.cz/intl/cs/sheets/about/>.
- [5] Microsoft. *Microsoft Excel* [online]. [cit. 2024-4-4]. Dostupný z: <https://www.microsoft.com/en-us/microsoft-365/excel>.
- [6] Dougiamas, Martin. *Moodle* [online]. [cit. 2024-4-4]. Dostupný z: <https://moodle.org/>.
- [7] Dougiamas, Martin. *Instalace systému Moodle* [online]. [cit. 2024-4-4]. Dostupný z: https://docs.moodle.org/403/en/Installing_Moodle.
- [8] Microsoft. *Visual Studio* [online]. [cit. 2024-4-4]. Dostupný z: <https://visualstudio.microsoft.com/cs/>.
- [9] Microsoft. *Platforma .NET* [online]. [cit. 2024-4-4]. Dostupný z: <https://learn.microsoft.com/cs-cz/dotnet/core/introduction>.
- [10] Microsoft. *Dokumentace jazyka C#* [online]. [cit. 2024-4-4]. Dostupný z: <https://learn.microsoft.com/cs-cz/dotnet/csharp/>.
- [11] Microsoft. *LINQ* [online]. [cit. 2024-4-4]. Dostupný z: <https://learn.microsoft.com/cs-cz/dotnet/csharp/linq/>.
- [12] Microsoft. *Entity Framework Core* [online]. [cit. 2024-4-4]. Dostupný z: <https://learn.microsoft.com/cs-cz/ef/core/>.
- [13] Microsoft. *SQL Server* [online]. [cit. 2024-4-4]. Dostupný z: <https://www.microsoft.com/en-us/evalcenter/evaluate-sql-server-2019>.
- [14] Consortium, World Wide Web. *HTML* [online]. [cit. 2024-4-4]. Dostupný z: <https://developer.mozilla.org/en-US/docs/Web/HTML>.
- [15] Consortium, World Wide Web. *World Wide Web Consortium (W3C)* [online]. [cit. 2024-4-4]. Dostupný z: <https://www.w3.org/>.
- [16] Consortium, World Wide Web. *CSS* [online]. [cit. 2024-4-4]. Dostupný z: <https://developer.mozilla.org/en-US/docs/Web/CSS>.
- [17] Mark Otto, Jacob Thornton. *Bootstrap* [online]. [cit. 2024-4-8]. Dostupný z: <https://getbootstrap.com/docs/5.1/getting-started/introduction/>.

- [18] Microsoft. *Razor Pages* [online]. [cit. 2024-4-4]. Dostupný z: <https://learn.microsoft.com/cs-cz/aspnet/core/razor-pages/?view=aspnetcore-8.0&tabs=visual-studio>).
- [19] Mozilla. *JavaScript* [online]. [cit. 2024-4-4]. Dostupný z: <https://developer.mozilla.org/en-US/docs/Web/JavaScript>).
- [20] Microsoft. *NuGet* [online]. [cit. 2024-4-4]. Dostupný z: <https://learn.microsoft.com/cs-cz/nuget/what-is-nuget>).
- [21] Microsoft. *Microsoft.EntityFrameworkCore* [online]. [cit. 2024-4-4]. Dostupný z: <https://www.nuget.org/packages/Microsoft.EntityFrameworkCore>).
- [22] Microsoft. *Microsoft.EntityFrameworkCore.Proxies* [online]. [cit. 2024-4-4]. Dostupný z: <https://learn.microsoft.com/en-us/ef/core/querying/related-data/lazy>).
- [23] Microsoft. *Microsoft.EntityFrameworkCore.SqlServer* [online]. [cit. 2024-4-4]. Dostupný z: <https://www.nuget.org/packages/Microsoft.EntityFrameworkCore.SqlServer/>).
- [24] Microsoft. *Microsoft.EntityFrameworkCore.Design* [online]. [cit. 2024-4-4]. Dostupný z: <https://www.nuget.org/packages/Microsoft.EntityFrameworkCore.Design>).
- [25] Microsoft. *Microsoft.EntityFrameworkCore.Tools* [online]. [cit. 2024-4-4]. Dostupný z: <https://www.nuget.org/packages/Microsoft.EntityFrameworkCore.Tools>).
- [26] Microsoft. *Microsoft.AspNetCore.Identity.EntityFrameworkCore* [online]. [cit. 2024-4-4]. Dostupný z: <https://www.nuget.org/packages/Microsoft.AspNetCore.Identity.EntityFrameworkCore>).
- [27] Microsoft. *Microsoft.AspNetCore.Identity.UI* [online]. [cit. 2024-4-4]. Dostupný z: <https://www.nuget.org/packages/Microsoft.AspNetCore.Identity.UI>).
- [28] Bozio, Giorgio. *Rotativa.AspNetCore* [online]. [cit. 2024-4-4]. Dostupný z: <https://www.nuget.org/packages/Rotativa.AspNetCore>).
- [29] Stedfast, Jeffrey. *MailKit* [online]. [cit. 2024-4-4]. Dostupný z: <https://www.nuget.org/packages/MailKit/>).
- [30] Close, Josh. *CsvHelper* [online]. [cit. 2024-4-4]. Dostupný z: <https://joshclose.github.io/CsvHelper/>).
- [31] Odinokov, Sergey. *Hangfire* [online]. [cit. 2024-4-4]. Dostupný z: <https://www.hangfire.io/>).
- [32] Microsoft. *Architektura MVC* [online]. [cit. 2024-4-9]. Dostupný z: <https://learn.microsoft.com/cs-cz/aspnet/core/mvc/overview?view=aspnetcore-6.0>).

[33] s.r.o., BAKALÁŘI software. *Bakaláři* [online]. [cit. 2024-4-9]. Dostupný z: <https://www.bakalari.cz/>.