

ČESKÉ VYSOKÉ UČENÍ TECHNICKÉ V PRAZE

FAKULTA INFORMAČNÍCH TECHNOLOGIÍ



Webová aplikace pro on-line opravování ručně
psaných dokumentů

DIPLOMOVÁ PRÁCE

2023

Jan PETRŽÍLKA



Zadání diplomové práce

Název:	Webová aplikace pro on-line opravování ručně psaných dokumentů
Student:	RNDr. Jan Petržílka, Ph.D., MBA
Vedoucí:	Ing. Tomáš Kalvoda, Ph.D.
Studijní program:	Informatika
Obor / specializace:	Softwarové inženýrství
Katedra:	Katedra softwarového inženýrství
Platnost zadání:	do konce letního semestru 2022/2023

Pokyny pro vypracování

Cílem práce je navrhnout a realizovat prototyp webové aplikace pro podporu opravování studentských písemných prací (tedy domácích úkolů, zkouškových písemek, atp.).

Aplikace bude poskytovat zejména tyto funkčnosti:

- * Vložení oskenovaných písemných prací a jejich přiřazení studentům, případně nahrání prací samotným studentem.
- * Opravení práce ve webovém prohlížeči způsobem umožňujícím vložit nebo pomocí stylusu ručně připsat poznámky a komentáře.
- * Distribuce opravených prací s opravami zpět studentům.
- * Evidence bodového ohodnocení jednotlivých částí práce a prezentace souhrnných výsledků.

Postupujte v těchto krocích:

- * Proveďte průzkum a analýzu požadavků od učitelů se zkušenostmi s on-line opravováním písemek a úkolů v matematických předmětech na KAM FIT.
- * Navrhněte a implementujte serverovou část aplikace.
- * Navrhněte a implementujte funkční prototyp klientské aplikace pro webové prohlížeče.
- * Navrhněte integraci s aplikací Grades, integraci proveďte a ověřte její funkčnost.
- * Vhodnými postupy otestujte a ověřte správnost implementace.
- * Odstraňte zjištěné nedostatky.



**FAKULTA
INFORMAČNÍCH
TECHNOLOGIÍ
ČVUT V PRAZE**

* Zhodnoťte použitelnost výsledného prototypu aplikace, navrhněte způsob uvedení do budoucího provozu.



Diplomová práce

WEBOVÁ APLIKACE PRO ON-LINE OPRAVOVÁNÍ RUČNĚ PSANÝCH DOKUMENTŮ

Jan Petržílka

Fakulta informačních technologií ČVUT v Praze
Katedra softwarového inženýrství
Vedoucí: Ing. Tomáš Kalvoda, Ph.D.
27. dubna 2023

České vysoké učení technické v Praze
Fakulta informačních technologií

© 2023 Jan Petržílka. Všechna práva vyhrazena.

Tato práce vznikla jako školní dílo na Českém vysokém učení technickém v Praze, Fakultě informačních technologií. Práce je chráněna právními předpisy a mezinárodními úmluvami o právu autorském a právech souvisejících s právem autorským. K jejímu užití, s výjimkou bez uplatněných zákonných licencí nad rámec oprávnění uvedených v Prohlášení, je nezbytný souhlas autora.

Odkaz na tuto práci: Petržílka, Jan. *Webová aplikace pro on-line opravování ručně psaných dokumentů*. Diplomová práce. České vysoké učení technické v Praze, Fakulta informačních technologií, 2023.

Obsah

Poděkování	x
Prohlášení	xi
Abstrakt	xii
Shrnutí	xiii
Seznam zkratk a pojmů	xiv
1 Úvod	1
2 Sběr požadavků	3
2.1 Vytčení funkčního rámce	3
2.2 Ověření proveditelnosti	4
2.3 Příprava požadavků a návrh specifikace	5
2.4 Analýza integračních možností	5
2.5 Vlastní sběr požadavků	5
2.5.1 Struktura hodnocení	5
2.5.2 Třídění prací	6
2.5.3 Opravování se zamykáním	7
2.5.4 Požadavky na editaci	7
2.6 Grafická prezentace výsledků	8
3 Struktura webové aplikace	11
4 Databáze	13
4.1 Definice předmětů	13
4.2 Odevzdání	15
4.3 Lokalizace	17
4.4 Pomocné tabulky	18
5 Aplikační server	21
5.1 Spring Security	22
5.2 Poskytované služby	22
5.2.1 Veřejně poskytované služby	22
5.2.2 Služby pro uživatele třídy USER	23
5.2.3 Služby pro uživatele třídy TEACHER	23
5.2.4 Služby pro uživatele třídy STUDENT	25
5.2.5 Služby pro uživatele třídy ADMIN	26

6	Integrace	27
6.1	Integrace na systém KOS	27
6.1.1	Nastavení integrace pomocí Apps Managera	27
6.1.2	Využívané služby	29
6.2	Integrace na aplikaci Grades a SSO	31
6.2.1	Nastavení integrace pomocí Apps Managera	31
6.2.2	Využívané služby	32
6.3	Integrace na e-mailový server	34
7	Struktura kódu uživatelského rozhraní	35
7.1	Kód v Reactu	35
7.2	Datový model editoru obrázků	36
8	Uživatelské rozhraní	39
8.1	Principy	39
8.2	Ovládání aplikace	41
8.3	Aplikační menu	41
8.4	Veřejné stránky	43
8.4.1	Úvodní stránka	43
8.4.2	SSO přihlášení	43
8.4.3	Lokální přihlášení	44
8.5	Stránky pro učitele	45
8.5.1	Seznam předmětů	46
8.5.2	Seznam paralelek	47
8.5.3	Nastavení paralelky	48
8.5.4	Nové odevzdání	50
8.5.5	Seznam odevzdání	51
8.5.6	Provedení opravy	53
8.5.7	Zápis do Grades	58
8.5.8	Editor	59
8.5.9	Náhled studenta	61
8.5.10	Generování QR kódů	61
8.5.11	Skenování QR kódů	63
8.5.12	Prezentace výsledků	65
8.6	Stránky pro studenty	67
8.6.1	Seznam předmětů	68
8.6.2	Seznam paralelek a milníků	68
8.6.3	Nové odevzdání	69
8.6.4	Seznam odevzdání	70
8.6.5	Náhled odevzdání	71
8.6.6	Náhled obrázku s opravou	71
8.7	Stránky pro správce	72
8.7.1	Lokalizace – seznam stránek	73
8.7.2	Lokalizace – obsah stránky	74
8.7.3	Lokalizace – historie prvku	75
8.7.4	Log	76
8.7.5	Statistika	77
8.7.6	Změna hesla	78
8.7.7	Správa uživatelů	79
8.7.8	Systémové nastavení	80

9	Testování, zhodnocení a uvedení do provozu	81
9.1	Testování	81
9.1.1	Použitý hardware	81
9.1.2	Kapacitní testy	82
9.2	Uživatelské testování	82
9.3	Zhodnocení a uvedení do provozu	83
10	Závěr	85
	Literatura	86
A	Licenční požadavky	93
A.1	Vývojové a pomocné nástroje	93
A.2	Aplikační server	93
A.3	Komponenty uživatelského rozhraní	94
B	Vývojové prostředí	97
B.1	Instalace knihoven pro uživatelské prostředí React	97
B.2	Lokální spuštění aplikace	98
B.3	Kompilace uživatelského rozhraní v Reactu	98
B.4	Kompilace aplikace v Javě	98
C	Provozní prostředí	99
C.1	Instalace Javy	99
C.2	Instalace PostgreSQL	99
C.3	Spuštění aplikace	101
C.4	Struktura adresářů	102
D	Přehled požadavků	103
D.1	Typy uživatelů	103
D.2	Funkční požadavky	103
D.2.1	Funkční požadavky na uživatelské rozhraní	104
D.2.2	Funkční požadavky na serverovou část	104
D.3	Nefunkční požadavky	106
D.3.1	Nefunkční požadavky na uživatelské rozhraní	106
E	Návrh specifikace	107
E.1	Funkční požadavky	107
E.1.1	Případy užití	107
E.2	Technické, servisní a nefunkční požadavky	124
E.2.1	Bezpečnostní požadavky	124
E.2.2	E-mailový SMTP server	124
E.2.3	Migrace dat	124
E.3	Předpokládaný harmonogram	124
F	Obsah datové přílohy	125
G	Kontaktní údaje	127

Seznam obrázků

2.1	Ověření proveditelnosti editace obrázků na webu	4
2.2	Návrh nastavení struktury hodnocení práce pro jednu paralelku podle diskusí s klíčovými uživateli	6
3.1	Struktura webové aplikace	12
4.1	Tabulky definující předměty	14
4.2	Tabulky sloužící pro odevzdání	16
4.3	Tabulky pro uložení lokalizace	17
4.4	Pomocné tabulky	18
5.1	Architektura aplikačního serveru	21
6.1	Založení aplikace typu Service account	27
6.2	Zobrazení přístupových údajů pro připojení ke KOSu	28
6.3	Nastavení přístupu na KOS API	28
6.4	Založení aplikace typu Web application	31
6.5	Zobrazení přístupových údajů pro SSO přihlášení pro připojení ke Grades	31
6.6	Nastavení přístupu na Grades	32
7.1	Struktura kódu v Reactu	35
7.2	Příklad editace pomocí editoru při testu operací s velkým množstvím objektů	36
7.3	Datový model editoru	37
8.1	Potvrzovací dialog u kritických operací	39
8.2	Zpráva o dokončení operace	40
8.3	Indikace čekání na dokončení operace	40
8.4	Koláž použití tlačítek různých barev	41
8.5	Příklad aplikačního menu pro učitele přihlášeného jako lokální uživatel	41
8.6	Výchozí stránka pro správce	42
8.7	Veřejné stránky s přechody	43
8.8	Stránka po otevření aplikace	43
8.9	Přihlášení pomocí SSO brány	44
8.10	Potvrzení oprávnění při přihlášení	44
8.11	Přihlášení lokálního uživatele	45
8.12	Stránky pro učitele s přechody	45
8.13	Seznam vyučovaných předmětů	46
8.14	Zobrazení seznamu paralelek předmětu	47
8.15	Seznam zkoušek a událostí	48
8.16	Nastavení paralelky	49
8.17	Formulář pro odevzdání práce učitelem	50
8.18	Formulář pro odevzdání práce učitelem s vloženými soubory	51
8.19	Zobrazení seznamu odevzdaných prací učiteli	52
8.20	Stránka provedení opravy před provedením změn	53

8.21	Stránka provedení opravy před uložením změn	54
8.22	Stránka provedení opravy před použitím zámků	55
8.23	Stránka provedení opravy po použití zámků	56
8.24	Stránka provedení opravy po uložení změn	57
8.25	Zápis hodnocení do aplikace Grades	58
8.26	Editor při zobrazení na šířku	59
8.27	Editor při zobrazení na výšku	60
8.28	Zvětšení zobrazené plochy při přetečení okraje obrázku	61
8.29	QR kód s identifikací studenta a příkladu	61
8.30	Generování zkušebních archů s QR kódy	62
8.31	Tlačítko pro zobrazení dalších souborů	62
8.32	E-mail se zkušebními archy	63
8.33	Přiřazení QR kódu ke studentovi	64
8.34	Výběr dat pro srovnání výsledků	65
8.35	Srovnání bodového hodnocení milníku studentů dvou paralelek	66
8.36	Srovnání bodového hodnocení části milníku studentů tří paralelek	66
8.37	Stránky pro studenty s přechody	67
8.38	Přehled studovaných předmětů	68
8.39	Přehled paralelek s milníky	68
8.40	Formulář pro odevzdání práce studentem	69
8.41	Formulář pro odevzdání práce studentem s jedním vloženým souborem	70
8.42	Seznam odevzdání zobrazený studentovi	70
8.43	Zobrazení obsahu odevzdání studentovi	71
8.44	Náhled opraveného obrázku zobrazený studentovi na šířku	71
8.45	Náhled opraveného obrázku zobrazený studentovi na výšku	72
8.46	Stránky pro správce s přechody	72
8.47	Seznam stránek (zkráceno)	73
8.48	Obsah jedné stránky – příklad pro stránku přihlášení	74
8.49	Historie nastavení prvku	75
8.50	Aplikační log	76
8.51	Počet požadavků za minutu	77
8.52	Doba odpovědi měřená po minutách	77
8.53	Využití heapu po minutách	78
8.54	Změna hesla lokálního uživatele	78
8.55	Správa lokálních uživatelů	79
8.56	Nastavení systému	80
E.1	Případy užití studenta	107
E.2	Formulář odevzdání práce	108
E.3	Formulář náhledu na práci	109
E.4	Zobrazení opravené práce	110
E.5	Formulář výběru semestru, předmětu a práce studentem	110
E.6	Případy užití učitele	111
E.7	Formulář nastavení parametrů práce	112
E.8	Formulář hromadného odevzdání prací	113
E.9	Formulář jednotlivého odevzdání práce	114
E.10	Přehled oskenovaných prací	115
E.11	Přiřazení stránek studentům	115
E.12	Přehled vložených prací	116
E.13	Rozdělení práce podle hodnocených částí	116
E.14	Přehled částí k opravě	117
E.15	Zobrazení vrstev opravy	118

E.16	Editor pro provedení opravy	119
E.17	Přehled opravených částí	120
E.18	Zobrazení opravené práce	121
E.19	Formulář výběru semestru, předmětu a práce učitelem	121
E.20	Případy užití správce	122
E.21	Případy užití systému	123

Seznam tabulek

4.1	Stavy odevzdání	15
4.2	Stavy obrázků	15
4.3	Záznamy v konfigurační tabulce config	19
5.1	Třídy uživatelů Spring Security	22
5.2	Veřejně poskytované služby	23
5.3	Služby pro uživatele třídy USER	23
5.4	Služby pro uživatele třídy TEACHER – část 1	24
5.5	Služby pro uživatele třídy TEACHER – část 2	25
5.6	Služby pro uživatele třídy STUDENT	25
5.7	Služby pro uživatele třídy ADMIN	26
6.1	Štítky v definici služeb pro přístup ke KOS API	29
6.2	Služba pro získání tokenu pro KOS API	29
6.3	Využívané služby KOS API	30
6.4	Štítky v definici služeb Grades a SSO	33
6.5	Využívané služby SSO	33
6.6	Využívané služby Grades	33
8.1	Rozlišení tlačítek podle barev	41
8.2	Struktura menu podle rolí uživatele	42
B.1	Použité balíčky NPM	98
C.1	Využívané adresáře	102
D.1	Kategorie požadavků	103
D.2	Funkční Požadavky na uživatelské rozhraní	104
D.3	Zpracovávaná data	104
D.4	Operace s daty	105
D.5	Další požadavky	106
D.6	Nefunkční požadavky na uživatelské rozhraní	106
E.1	Parametry práce	112
E.2	Parametry aplikace	122
E.3	Harmonogram práce	124

Děkuji vedoucímu práce Ing. Tomáši Kalvodovi, Ph.D. za cenné připomínky a pečlivé vedení práce po celou dobu jejího trvání, Ing. Michalu Valentovi, Ph.D. za pomoc s integrací a jejím testováním a Ing. Štěpánu Plachému za klíčové informace o integračním rozhraní aplikace Grades.

Děkuji klíčovým uživatelům z Katedry aplikované matematiky za četné podněty, postřehy, rady a za účast na testování aplikace.

Děkuji Fakultě informačních technologií ČVUT v Praze, že mi umožnila tuto práci realizovat.

Prohlášení

Prohlašuji, že jsem předloženou práci vypracoval samostatně a že jsem uvedl veškeré použité informační zdroje v souladu s Metodickým pokynem o dodržování etických principů při přípravě vysokoškolských závěrečných prací.

Beru na vědomí, že se na moji práci vztahují práva a povinnosti vyplývající ze zákona č. 121/2000 Sb., autorského zákona, ve znění pozdějších předpisů. V souladu s ust. § 2373 odst. 2 zákona č. 89/2012 Sb., občanský zákoník, ve znění pozdějších předpisů, tímto uděluji nevýhradní oprávnění (licenci) k užití této mojí práce, a to včetně všech počítačových programů, jež jsou její součástí či přílohou a veškeré jejich dokumentace (dále souhrnně jen „Dílo“), a to všem osobám, které si přejí Dílo užít. Tyto osoby jsou oprávněny Dílo užít jakýmkoli způsobem, který nesnižuje hodnotu Díla a za jakýmkoli účelem (včetně užití k výdělečným účelům). Toto oprávnění je časově, teritoriálně i množstevně neomezené. Každá osoba, která využije výše uvedenou licenci, se však zavazuje udělit ke každému dílu, které vznikne (byť jen zčásti) na základě Díla, úpravou Díla, spojením Díla s jiným dílem, zařazením Díla do díla souborného či zpracováním Díla (včetně překladu) licenci alespoň ve výše uvedeném rozsahu a zároveň zpřístupnit zdrojový kód takového díla alespoň srovnatelným způsobem a ve srovnatelném rozsahu, jako je zpřístupněn zdrojový kód Díla.

V Praze dne 27. dubna 2023

.....

Abstrakt

Obsahem práce je implementace webové aplikace podle požadavků klíčových uživatelů z Katedry aplikované matematiky FIT ČVUT v Praze. Vstupem aplikace jsou oskenované studentské práce nahrané do aplikace studenty nebo vyučujícími. Přiřazení jednotlivých stránek odevzdané práce podle studentů nebo samostatně hodnocených částí úloh je prováděno ručně nebo s využitím QR kódů. QR kódy jsou studentům přiřazeny pomocí mobilní části webové aplikace. Učitelé opravují práce studentů pomocí zabudovaného editoru. Aplikace je integrována na školní informační systém KOS a na fakultní systém pro evidenci bodového hodnocení studentů Grades. Uživatelské rozhraní je provedeno česky a anglicky s plně parametrizovanou lokalizací. Serverová část je implementována v Javě, uživatelské rozhraní ve frameworku React s editorem napsaným v JavaScriptu. Pro ukládání dat je použita databáze PostgreSQL.

Klíčová slova opravování písemných prací, webový editor obrázků, použití QR kódů ke třídění dokumentů, jazyková lokalizace

Abstract

The scope of the thesis is the implementation of a web application according to the requirements of key users from the Department of Applied Mathematics at the FIT CTU in Prague. The input of the application is scanned student work uploaded to the application by students or lecturers. The assignment of individual pages of the submitted work to students or graded parts of assignments is done manually or with the use of QR codes. QR codes are assigned to students using the mobile part of the web application. Teachers correct student work using the built-in editor. The application is integrated to the school's KOS information system and to the faculty system for recording student scores Grades. The user interface is implemented in Czech and English with fully parameterized localization. The server part is implemented in Java, the user interface in the React framework with an editor written in JavaScript. PostgreSQL is used as the database.

Keywords corrections of written work, web-based image editor, using QR codes to sort documents, language localization

Shrnutí

Motivace

Motivací mojí práce bylo využít procesní znalosti hodnocení studentů na FIT ČVUT a zkušenosti s používanými informačními systémy získané během mého studia, využít získané odborné znalosti a dodat webovou aplikaci, která bude odpovídat potřebám uživatelů Katedry aplikované matematiky.

Cíl práce

Cílem práce je implementace webové aplikace s následujícími klíčovými funkcčnostmi:

- Definování způsobu odevzdání práce pro jednotlivé paralelky vyučovaných předmětů.
- Odevzdání studentské práce studentem nebo vyučujícím.
- Roztřídění odevzdaných prací podle studentů a jednotlivě hodnocených částí práce.
- Opravení práce a zaznamenání hodnocení do aplikace Grades.
- Distribuce opravených prací s poznámkami hodnotitele zpět ke studentům.

Těžištěm písemné části práce je popis implementované aplikace tak, aby text mohl sloužit jako vodítko jejím uživatelům a byl oporou při jejím dalším rozvoji a úpravách.

Postup

Při implementaci aplikace jsem nejdříve s pomocí modelu případů užití, wireframů obrazovek a pracovních textů s vedoucím práce vyjasnil

jeho požadavky a očekávání. Následně jsem vytvořil editor sloužící k opravování prací, abych se ujistil o proveditelnosti záměru. Po zjištění možností integrace aplikace se systémy KOS a Grades jsem diskutoval s klíčovými uživateli aplikace jejich požadavky.

Tyto požadavky jsem v maximální možné míře zohlednil při návrhu a implementaci jednotlivých funkcností. Aplikaci jsem připravil k otestování v několikátýdenním předstihu před odevzdáním písemné části práce, podle možností jsem opravil nalezené chyby a zapracoval vznesené připomínky.

Výsledky práce

Výsledná aplikace odpovídá cílům práce a formulovaným požadavkům na její funkcčnost.

Při její implementaci jsem využíval zejména technologie, se kterými jsem se blíže seznámil až v průběhu svého studia, řadu znalostí jsem získal až při vlastní implementaci. Výsledná práce tak představuje maximum toho, co jsem dokázal ve vymezeném čase s využitím svých znalostí naprogramovat. Spíše než o hotovou aplikaci se jedná o prototyp, který obsahuje všechny klíčové funkcčnosti, ale není dosud dostatečně zralý k rutinnímu používání.

Závěr

Webovou aplikaci jsem implementoval dle požadavků zadavatele způsobem popsáním v této práci.

Seznam zkratk a pojmů

Obecné zkratky a pojmy

KOS	Informační systém ČVUT v Praze sloužící k evidenci předmětů, paralelek, učitelů, studentů a dalších entit a vazeb mezi nimi.
Grades	Webová aplikace FIT ČVUT v Praze sloužící k zaznamenávání hodnocení studentů, definuje také milníky (viz níže).
Předmět / Course	Předmět je vyučován jedním nebo více vyučujícími a má zpravidla více paralelek. Studenti si zapisují předmět a mají zapsanou jednu nebo více paralelek. Jednotlivé předměty mohou být vyučovány pouze v některých semestrech.
Paralelka / Parallel	Předmět, který je v daném semestru vyučován, má definované paralelky. Paralelka může být více typů: jedná se o přednášky, cvičení nebo laboratoře. Jednotlivé paralelky mají definované vyučující, studenty, čas a místo výuky.
Milník / Milestone	Při studiu předmětu hodnotí vyučující studentovy studijní výsledky v jednotlivých milnících. Tyto milníky jsou definovány v aplikaci Grades. Hodnocení může být číselné, slovní nebo mít logickou hodnotu. Typickým milníkem je zápočet, semestrální práce nebo její část, domácí úkol nebo zkouška. V aplikaci Grades jsou milníky z pohledu studenta reprezentovány řádky s hodnocením u jednotlivých předmětů, z pohledu učitele pak sloupce (a jednotlivé řádky odpovídají studentům).

Aplikačně specifické pojmy

Písemky / Assignments	Název webové aplikace, jejíž implementace je předmětem této práce.
Část / Part	Milník je složen z více částí, které jsou samostatně hodnocené (například jeden z příkladů v písemce). Hodnocená část milníku se může započítávat do milníku, pro nějž je definovaná, nebo do jiného milníku.
Odevzdání / Submission	Jeden nebo více souborů, které jsou vyučujícím nebo studentem do aplikace pro daný předmět a jeho milník nahrány.

Kapitola 1

Úvod

Na Katedře aplikované matematiky FIT ČVUT v Praze již dříve používali kombinaci různých softwarových nástrojů (Teams Assignments a samostatné grafické editory) k opravování písemných prací studentů, které pokrývaly jednotlivé části cílové funkčnosti, nikdy se ale nejednalo o ucelenou aplikaci zajišťující všechny kroky agendy. Významným akcelerátorem tohoto úsilí byla koronavirová epidemie způsobená patogenem COVID-19, po jejím odeznění se ale vyučující vrátili k opravování písemných prací zejména v listinné podobě.

Při diskusi s vedoucím práce o očekávání od požadovaného řešení jsem kromě předpokládaných přínosů, jako je možnost odevzdávání prací v digitální podobě nebo jejich opravování bez fyzické přítomnosti vyučujícího na pracovišti, zachytil také následující předpokládané přínosy, které mě překvapily:

- Získání možnosti porovnat úspěšnost jednotlivých paralelek a studentů při zdolávání standardizovaných úloh a identifikace těžkých nebo problematických příkladů. Vyučující se snaží dosáhnout stavu, kdy jednotlivé příklady mají v různých letech nebo při různých opakováních zkoušek obdobnou náročnost, ale nemají k dispozici nástroj, jak tuto náročnost z výsledků studentů jednoduše zjistit. Obdobně platí, že jednotlivé paralelky se liší vyučujícím a eventuálně způsobem výuky, a opět není k dispozici nástroj, jak výsledky studentů z jednotlivých paralelek porovnat. Vhodná aplikace by mohla jednak data o úspěšnosti studentů shromáždit, jednak je prezentovat.
- Snížení bariéry studentů pro získání zpětné vazby. Vyučující s překvapením zjistí, že když studentům jejich práce opraví a předají jim bodové hodnocení jejich práce, řada studentů se nepřijde na opravené práce podívat, nemohou se tak ujistit o tom, kde přesně udělali chyby, a poučit se s nich. Vhodná aplikace by mohla opravené písemky studentům doručit až na obrazovku jejich notebooku a umožnit jim tak získat informace, o které nyní přicházejí.
- Snížení časových ztrát vznikajících při neustálém dohledávání příkladů v listinné podobě písemky vyučujícím v průběhu její opravy.

Vedlejším efektem používání takové aplikace vyučujícími a studenty bude rozšíření stávajícího portfolia řešených příkladů z minulých domácích úkolů a zkoušek dostupného studentům, což z mého pohledu na jedné straně ztěžuje vyučujícím přípravu nových příkladů k dalším zkouškám, na druhé straně může studentům přinést cenné studijní materiály.

Trvajícím pocitem potřeby aplikace, která umožní odevzdávat, opravovat a studentům vracet opravené práce, vedl vedoucího diplomové práce k vypsání tématu, k jehož řešení jsem se přihlásil. Výsledkem je implementace aplikace Písemky / Assignments popsaná v této práci.

Sběr požadavků

Na Internetu jsem se pokusil dohledat existující řešení dané úlohy. Zjistil jsem, že na Masarykově univerzitě v Brně používají podobné řešení [1]. Jejich řešení má na první pohled dvě základní odlišnosti od připravované aplikace:

1. Vlastní opravování probíhá automaticky, kdy aplikace vyhodnocuje naskenované zkušební archy, ve kterých studenti začerní připravená políčka podle písmene zvolené odpovědi.
2. Identifikace studenta a zkušebního archu je realizována automatickým rozpoznáním číslic, které student musí na arch namalovat ve tvarech podle vzoru jednotlivých číslic na zkušebním archu.

Omezení aplikace na vyhodnocení pouze definovaných odpovědí by značně omezilo stávající způsob zadávání a hodnocení písemek, kdy studenti zpravidla odevzdávají volný text a vyučující hodnotí nejen výsledek, ale celý proces řešení úlohy.

Moje vlastní zkušenosti s několika předměty, kdy se textové odpovědi nepsaly rukou na papír, ale vyplňovaly pomocí klávesnice do webové aplikace, byly negativní, protože mě tento způsob odevzdávání výsledku omezoval v možnosti text doplnit náčrtkem nebo podstatné věci podtrhnout nebo jinak zvýraznit.

Malování číslic studentem podle vzoru na zkušebním archu má zásadní výhodu spočívající v tom, že vyplňované číslo nemusí být na zkušební arch doplněno aplikací, ale jinak přináší jen nevýhody.

Výsledkem úvah nad popisem existujícího řešení bylo rozhodnutí, že připravovaná aplikace musí mít vhodný editor pro opravování volného textu. Pro identifikaci studenta a zkušebního archu pak musí používat co nejstandardnější nástroje – čili žádné malování číslic a jejich následné aplikační rozpoznávání.

Dále jsem si uvědomil, že podoba aplikace a značná část její logiky bude do značné míry určena systémy, se kterými bude integrována – tedy KOsem a Grades a daty, se kterými tyto aplikace pracují. Dospěl jsem tak k názoru, že téma je vhodné řešit vývojem nové aplikace, která může požadavky zadavatele splnit lépe než řešení převzaté z jiného školního kontextu.

2.1 Vytčení funkčního rámce

Před započítím vývojových prací jsem s vedoucím diplomové práce diskutoval jeho představu o rozsahu funkčnosti aplikace. Shodli jsme se, že aplikace by ideálně měla mít následující vlastnosti:

- Odevzdané soubory ve formátu PDF, JPG nebo PNG by aplikace měla převést na obrázky se standardizovaným bodovým rozlišením a učitel by měl mít možnost pomoci editoru v aplikaci nad obrázky dokreslit samostatně uloženou vrstvu s opravou studentovy práce.
- Aplikace by měla znát vyučované předměty, jejich vyučující a studenty, aby nebylo potřeba údaje, které jsou v KOSu k dispozici, do aplikace znovu zadávat.
- Aplikace by měla umět uložit bodové hodnocení stanovené vyučujícím do aplikace Grades.
- Aplikace by měla nějakým způsobem umožnit roztřídění oskenovaných písemek ze zápočtových testů a zkoušek a podobně na jednotlivé studenty a příklady.

Jediným opravdu nestandardním prvkem, který bylo potřeba vyřešit, byl editor obrázků umožňující opravy studentských prací. V minulých řešeních používaných na Katedře aplikované matematiky FIT ČVUT v Praze používali běžné grafické editory, kdy opravy zaznamenávali přímo do obrázků pořízených skenování studentských prací.

2.2 Ověření proveditelnosti

S vedoucím práce jsme se shodli, že optimální je udržovat původní obrázek vzniklý ze studentské práce a grafickou vrstvu vzniklou jeho opravováním odděleně. Jednak se tím ušetří potřebný datový objem, jednak zůstanou původní obrázky beze změny a je možné je zobrazit bez oprav, jednak je možné vrstvu s opravou postupně měnit bez dopadu na původní obrázek.

21:57 Sun 12. 6. assessments.mujfit.cz

Domácí úkol 2 - Jan Petržílka (petrjz5)
Tuesday, 1 January 2019 14:46

Tento text píše rukou na iPadu v apl. hac. assessments. mujfit.cz

BIK-MLO: úkol č.2
K odevzdání na cvičení 5. ledna. Podobné příklady se objeví v písemce.

Příklad 1. Najděte minimální DNT formule

$f(A, B, C, D) = \sum(0, 1, 3, 4, 8, 9, 11)$ *Ověřme*

Znárodníme v tabulce

	A	B	C	D	f $A=0$	f $A=1$	$\neg A \wedge D$	$\neg A \wedge C \wedge D$	$\neg A \wedge B \wedge D$	DNT
0	8	0	0	0	1	1	00	10	11	11
1	9	0	0	1	1	1	11	00	00	00
2	10	0	1	0	0	0	00	00	00	00
3	11	0	1	1	1	1	11	00	00	11
4	12	0	1	0	1	0	00	10	00	10
5	13	0	1	1	0	0	00	00	00	00
6	14	1	1	0	0	0	00	00	00	00
7	15	1	1	1	0	0	00	00	00	00

destavíme mapu

	C		
	0	1	1
0	0	1	1
1	1	0	0
2	0	0	0
3	1	1	1

minimální DNT = $(\neg B \wedge D) \vee (\neg A \wedge \neg C \wedge \neg D) \vee (\neg B \wedge \neg C \wedge \neg D)$

Smazat vše Stáhnout touchstart 556 141

Obnovit stránku

■ Obrázek 2.1 Ověření proveditelnosti editace obrázků na webu

Nevěděl jsem ale, zda je možné takové řešení realizovat v podobě webové aplikace běžící v běžném webovém prohlížeči, zda a v jakém rozsahu bude možné k opravování písemek použít tablety One by Wacom rozšířené mezi učitelé na katedře a jak bude fungovat rozpoznávání pohybu stylusu. Na Internetu jsem našel existující komerční řešení, aplikaci Sketchpad [2], která podobnou funkčnost obsahovala.

V podobné otázce uživatele na stránkách Stack Overflow [3] byla část nefunkčního kódu, ve kterém byly použito API prohlížeče pro zjišťování polohy myši a její znázorňování do obrázku [4]. Tento kód se stal zárodkem první verze editoru. V něm byla zobrazena část mého domácího úkolu a přes něj bylo možné dělat čáry myší nebo stylusem tak, jak je vidět na obrázku 2.1 pořízeném 12. 6. 2022 na iPadu, kde v prohlížeči běžel zárodek budoucí aplikace.

2.3 Příprava požadavků a návrh specifikace

Po ověření proveditelnosti jsem připravil přehled požadavků na budoucí aplikaci tak, jak jsem je vnímal, a po jejich prodiskutování s vedoucím práce jsem připravil návrh specifikace jako podklad pro další diskusi. Upravené pracovní a dále neaktualizované verze přehledu požadavků a návrhu specifikace jsou přiloženy v přílohách D a E této práce.

2.4 Analýza integračních možností

Před diskusí s klíčovými uživateli jsem se seznámil s daty uloženými v aplikaci KOS a Grades a implementoval jsem přenos předmětů vyučovaných v semestru, jejich paralelek, učitelů a studentů ze systému KOS do aplikace a dále načtení a uložení bodového hodnocení studenta do a z aplikace Grades. To mi umožnilo při diskusí s klíčovými uživateli klást kvalifikované otázky a lépe porozumět jejich požadavkům.

2.5 Vlastní sběr požadavků

Vlastní sběr požadavků proběhl formou diskusí s budoucími klíčovými uživateli z Katedry aplikované matematiky FIT ČVUT v Praze na rozhraní listopadu a prosince 2022. Uživatelům jsem představil základní funkčnosti aplikace pomocí návrhů obrazovek z přílohy E této práce na počítači jsem jim ukázal pracovní verzi editoru studentských prací. To jim umožnilo připomenout si svoje zkušenosti s minulými řešeními, lépe si představit budoucí aplikaci a přesněji formulovat svoje požadavky. Na druhou stranu je možné, že tak vznikl mylný dojem, že aplikace je již téměř hotová a že bude brzo připravena k testování.

Získané požadavky zásadním způsobem ovlivnily strukturu budoucí aplikace a způsob implementace jednotlivých funkcností.

2.5.1 Struktura hodnocení

Jak je vidět na obrázku E.7 v příloze E, podle původního návrhu bylo nastavení předmětu společné pro všechny paralelky. Jak se ukázalo, jednotlivé paralelky stejného předmětu v daném semestru mohou mít nejen různé termíny odevzdání úkolů k jednotlivým milníkům, ale vyučující jednotlivých paralelek mohou mít u stejných milníků úkoly s různým počtem dílčích částí.

Při diskusích s klíčovými uživateli jsem dále zjistil, že situace je ještě komplikovanější, protože hodnocení části jednoho milníku mohou být započítávána v aplikaci Grades k různým milníkům (například si můžeme představit odděleně evidované body za estetickou stránku písemky, barevné podání, míru originality řešení apod.), dokonce bylo požadováno, aby body za jednotlivé části jednoho milníku mohly být započítány k více různým milníkům současně.

Před započítím implementace definice hodnocení jsem proto připravil pracovní návrh nastavení odevzdávání prací pro jednu paralelku zachycený na obrázku 2.2.

Obrázek 2.2 Návrh nastavení struktury hodnocení práce pro jednu paralelku podle diskusí s klíčovými uživateli

Na návrhu je nové tlačítko („Copy their setting to this setup dialog“), které má umožnit uživateli zkopírovat nastavení jiné, již konfigurované paralelky, do nastavení aktuální paralelky. U jednotlivých hodnocených částí je pak určeno, do kterého milníku je jich bodové hodnocení započítáváno. Obě tyto funkčnosti jsou ve výsledné aplikaci implementovány.

Ve střední části obrazovky („Joint evaluation for events“) je pak nastavení společné konfigurace pro více různých milníků. Toho nastavení se ukázalo jako matoucí a implementačně náročné, protože vedlo k velmi komplikovaným vztahům při zapisování bodů v editoru. Proto jsem jej při implementační fázi vypustil.

Požadavek na možnost prodloužení termínu odevzdání pro konkrétního studenta jsem implementoval jako nastavení nového termínu odevzdání na již existujícím odevzdání daného studenta.

Požadavek na možnost přiložení slovního komentáře k odevzdávaným souborům jsem zapracoval a to jak pro odevzdávání souborů studentem, tak učitelem.

2.5.2 Třídění prací

Původní návrh popsany v příloze E předpokládal postupné zpracování každého odevzdání na více různých obrazovkách, na kterých by vyučující nastavoval nebo kontroloval přiřazení jednotlivých obrázků (tedy listů původního odevzdání v grafické podobě) ke studentovi a samostatně hodnocené části. Někteří klíčoví uživatelé shledávali takový postup jako uživatelsky nepřívětivý a požadovali, aby jednotlivé obrázky odevzdání mohli přiřadit ke studentům a hodnoceným částem

rovnou během opravování. Tomuto požadavku jsem vyhověl, a tak místo několika obrazovek se specializovanou funkcí vznikla jedna opravdu složitá obrazovka, na které se odehrává hned několik procesů včetně vlastní editace obrázků.

Požadavek automatického třídění jsem zapracoval s využitím QR kódů a tuto funkci podrobně vysvětluji v kapitole 8.5.10.

Požadavek, aby aplikace ze systému KOS načítala zkoušky a události k předmětu a pracovala s nimi jako s virtuálními paralelkami, jsem implementoval.

Požadavek na evidenci variant zkouškové písemky a přiřazení studenta k variantě písemky jsem neimplementoval. Chybějící funkčnost lze nahradit tak, že jednotlivé varianty písemky vyučující do aplikace vloží jako jiná odevzdání a v komentáři variantu písemky uvede.

2.5.3 Opravování se zamykáním

Při opravování zkouškových písemek nyní vyučující postupují tak, že jednotliví učitelé opravují vždy jen některé příklady. Bylo proto požadováno, aby aplikace opravujícím umožnila zvolit si opravovaný příklad (tedy část milníku) a aplikace mu za předpokladu předchozího přiřazení obrázků ke studentům a příkladům sama vybírá obrázky k opravě. Funkčnost musí správně fungovat i v situaci, kdy jeden příklad opravuje paralelně více vyučujících – aplikace musí sama hlídat, aby stejný příklad jednoho studenta neopravovali dva různí vyučující a nepřepsali si navzájem svoje opravy. Tato funkčnost je v aplikaci implementována.

2.5.4 Požadavky na editaci

Při diskusi nad editorem pro opravování jednotlivých obrázků (výsledná podoba editoru je na obrázcích 8.26 a 8.27) jsem získal největší množství vzájemně si odporujících požadavků.

Následující požadavky jsem zapracoval:

- Zúžil jsem okraje kolem okna editoru, aby plocha pro editaci byla co největší.
- Editor umožňuje kromě čáry kopírující pohyb myši, prstu nebo stylusu vkládat také rovné čáry spojující dva určené body.
- Uživatel může nastavit tloušťku čáry.
- Uživatel může obrázek rotovat opakovaně o 90 stupňů. Na základě experimentů s různými implementacemi funkčnosti jsem se rozhodl zvolit variantu, kdy je rotace součástí opravy vyučujícím (různé opravy téhož obrázku mohou mít různou rotaci) a kdy je rotován pouze podkladový obrázek a nikoliv vrstva s opravami. Funkčnost jsem nejprve implementoval tak, že editor zobrazoval pouze plochu obrázku odpovídající odevzdané práci studenta. U obrázku, který není čtvercový, se mohlo stát, že některé opravy byly uloženy, ale nebyly zobrazeny, protože byly pořízeny v okamžiku, kdy byl obrázek narotován odlišně. Tato vlastnost byla vnímána jako projev chyby aplikace, a tak ve výsledné podobě aplikace při rotaci automaticky rozšiřuje zobrazenou plochu tak, aby byly vidět všechny opravy provedené vyučujícím.
- Přidal jsem tlačítko na nastavení výchozího zvětšení na šířku editovaného obrázku.
- Při opravování prací na monitoru postaveném na výšku dojde k přeskupení ovládacích prvků tak, aby prostor pro provádění opravy byl co největší s co nejvhodnějším poměrem stran.
- Zohlednil jsem možnost odevzdání PDF souboru s extrémně dlouhými stránkami (například v poměru šířky a výšky 1:15). Pokud je výsledný obrázek příliš velký, aplikace jej přeškáluje na menší i za cenu nižšího bodového rozlišení.

- Požadavek na to, aby písemné práce mohl opravovat i vyučující, který není k předmětu přiřazen, jsem implementoval tak, že části aplikace sloužící k opravování mají v URL speciální přístupový kód – předáním URL adresy stránky pro opravování jakéhokoliv odevzdání je možné jinému uživateli aplikace opravování daného odevzdání umožnit.
- Požadavek na možnost umístit na obrázek klikací bod, který by při kliknutí ukázal text, jsem implementoval ve zjednodušené podobě jako možnost vložit komentář k opravenému obrázku. Následující požadavky jsem neimplementoval:
 - Realizace požadavku na dynamické schovávání ovládacích prvků by sice zvětšila plochu pro provádění vlastní opravy, již tak komplikované ovládání editoru by ale ještě více zkomplikovala a jednotlivé prvky, které například obsahují jméno studenta, by nebyly trvale vidět. Z mého pohledu by to mohlo nepříznivě ovlivnit výsledky opravování. Určitě se ale jedná o funkčnost, která by v budoucí verzi aplikace mohla být použita.
 - Požadavek na uživatelskou změnu pořadí stránek a jejich eventuální zkopírování jsem neimplementoval, protože by zkomplikoval již tak značně složitou funkčnost stránky pro opravování. Důvody pro tento požadavek jsou ale významné – pořadí stránek nemusí odpovídat logické návaznosti textu a na některé ze stránek mohou být části různých příkladů. Jako nejjednodušší náhradní řešení jsem do aplikace přidal speciální část milníku s pořadovým číslem 0 (nula), kterou je možné použít v případě, kdy obrázek není možné jednoznačně přiřadit k jedné části milníku.
 - Požadavek na možnost práce opravovat v jiném editoru by znemožnila nebo přinejmenším zkomplikovala stávající řešení, kdy jsou opravy uloženy jako samostatné vrstvy nad původním obrázkem. Pokud k tomu budou v budoucnosti vážné důvody, může budoucí verze aplikace takovou možnost poskytovat. Součástí požadavku byl návrh, aby na exportovaném obrázku byl QR kód, který při importu do aplikace opravovaný obrázek identifikuje a zajistí tak spolehlivost a uživatelskou přívětivost exportu a importu obrázků.
 - Požadavek na opravování textových odevzdání jsem neimplementoval. Do aplikace je možné textové odevzdání vložit jako PDF soubor a aplikace s ním bude pracovat jako s obrázkem.
 - Požadavek na možnost vložit odevzdání za více studentů, přičemž přiřazené body se každému z nich přiřadí ve stejné výši, jsem neimplementoval. Tato funkčnost může být vhodným kandidátem na zapracování do další verze aplikace.
 - Návrh na vhodné využití toho, že učitelé mají k dispozici dva monitory a aplikace může na každém monitoru zobrazovat něco jiného, jsem dále nerozpracoval.

2.6 Grafická prezentace výsledků

Součástí požadavků zadavatele byla také grafická prezentace výsledků studentů. Splnění těchto požadavků mělo oproti ostatním funkcčnostem nižší prioritu, jejich stávající implementace je tak spíše základem možného budoucího rozšíření než funkcčnostmi pokrývající všechny potřeby.

Při návrhu struktury prezentovaných dat jsem se opřel o podklady pro svou jinou studentskou práci. Předpokládám, že jsou alespoň částečně relevantní, přestože popisují situaci v odlišném školním a kulturním kontextu, protože se domnívám, že studenti FIT ČVUT v Praze nevyhledávají studium na této fakultě právě kvůli předmětům vyučovaným Katedrou aplikované matematiky. Role těchto předmětů je při jejich studiu proto podobná jako v případě odkazovaných materiálů.

„National Association of Secondary Principals“ podle [5] používá „Value Added Measures“ (VAM), kdy jsou učitelé hodnoceni (to znamená, že jim je smlouva prodloužena nebo ukončena, případně se od toho odvíjí jejich finanční kompenzace) podle výsledků testů jejich studentů

(zde je požadováno, aby testy kromě výběru z možností obsahovaly také otevřené konstruktivní otázky – zde opět nalézám podporu pro tvrzení, že je potřeba umožnit opravu volného textu produkovaného studentem). Podle „Every Student Succeeds Act“ je požadováno hodnotit také to, jak student dosáhne svých cílů na vysoké škole, v kariéře a v životě (!).

Podle [6] jsou na školách obvykle používána hodnotící kritéria, která nejsou schopna efektivně rozlišit kvalitu výuky různých učitelů, takže většina učitelů je hodnocena jako učitelé s vysokou přidanou hodnotou. Dva národní učitelské odborové svazy, „AFT“ a „NEA“, podporují používání systémů, které identifikují schopnosti učitelů a na jejich takto identifikované schopnosti váží jejich ohodnocení. Tato praxe navazuje na „National Board for Professional Teaching Standards“, který má sloužit k identifikaci výborných učitelů a jejich platovou bonifikaci.

Běžnou praxí je usuzování na schopnosti učitelů podle úspěšnosti jejich studentů ve standardizovaných testech. Učitel, jehož žáci nejsou úspěšní, nedostává příležitost tyto neúspěchy napravit. To má značné dopady na celý průběh vzdělávacího procesu, kdy učitelé svoje studenty připravují zejména na úspěšné vykonání srovnávacích testů, na jejichž výsledku mají vlastní zájem.

Předpokládám, že cílem výuky na katedře je právě vybavení studenta takovými znalostmi a jeho osvojení takového způsobu uvažování, které jej podpoří v dosahování jeho úspěchů při studiu a v dalším profesním životě. Vzhledem k existenci a rutinnímu používání standardizovaných testů – zkouškových písemek, které jsou používány napříč paralelkami vyučovanými různými učiteli, jsem tento požadavek implementoval jako možnost vygenerovat grafy srovnávající úspěšnost studentů z různých paralelek v bodovém hodnocení jednotlivých milníků a jejich částí. To umožní identifikovat rozdíly mezi paralelkami a těžké nebo problematické příklady.

Struktura webové aplikace

Technické řešení webové aplikace vychází z méj bakalářské práce [7] a stojí na stejných technických základech. Jako databáze je použito PostgreSQL [8] a aplikační server je napsán v Javě [9]. Java je jediný programovací jazyk, ve které jsem schopen realizovat požadované řešení tak, aby se jeho struktura blížila obvyklým standardům a aby eventuální pokračovatelé mohli na moji práci dobře navázat. Výběr PostgreSQL je pak dán kombinací požadavku zadavatele provozovat řešení na virtuálním serveru (tedy není možné použití školní instalaci databáze Oracle) a mými zkušenostmi z bakalářské práce s kombinací Javy a právě PostgreSQL.

Aplikační server využívá frameworky Spring Boot [10], [11], [12] a Spring Security [13] a napojení na databázi je řešeno pomocí frameworku Hibernate ORM [14].

Aplikační server využívá knihovnu Apache PDFBox [15] pro práci s PDF dokumenty jako je konverze na obrázky a spojování PDF stránek do jednoho dokumentu, a knihovnu openhtmltopdf [16] pro generování PDF dokumentů s QR kódy. QR kódy jsou pak generovány a v odevzdané práci rozpoznávány pomocí knihovny Google ZXing [17].

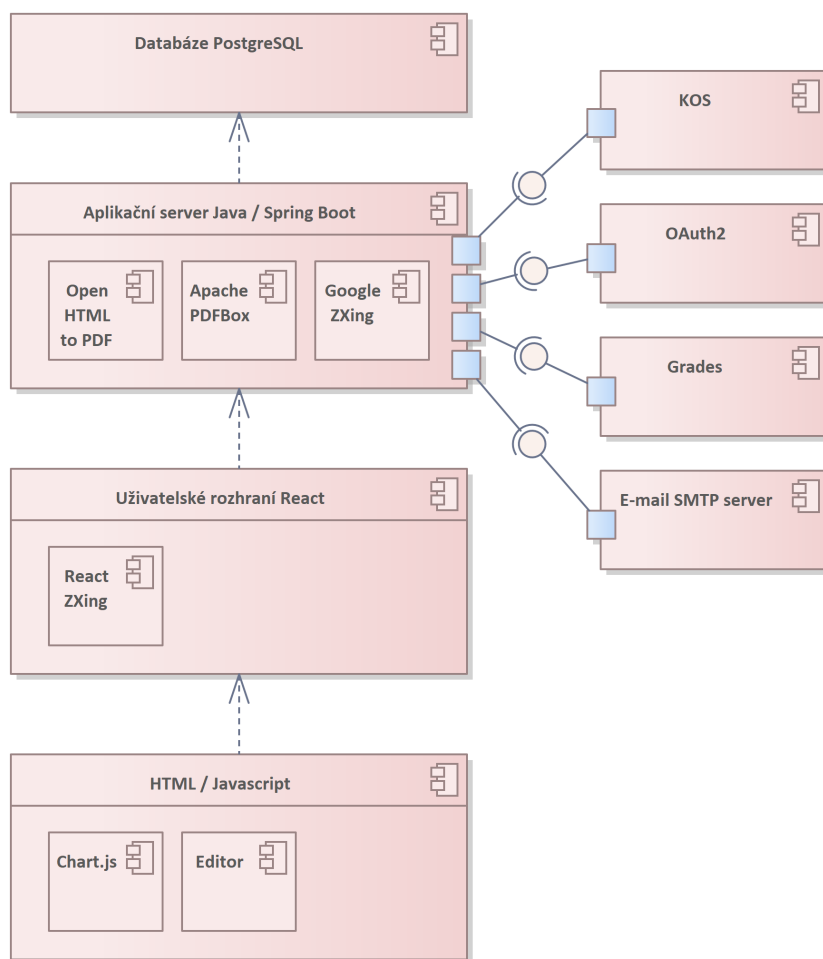
Uživatelské rozhraní bylo v případě méj bakalářské práce přímo generované v HTML s JavaScriptem pomocí kódu v jazyce Java s využitím v aplikaci definovaných šablon stránek a jejich částí, což bylo předmětem kritiky oponenta – uživatelské rozhraní bylo provázáno na kód aplikačního serveru, nebylo oddělené a jednoduše vyměnitelné. V případě aplikace vytvořené v této práci je proto uživatelské rozhraní naprogramováno odděleně s využitím specializovaného frameworku a s aplikačním serverem komunikuje pomocí HTTP rozhraní.

Během studia jsem se seznámil s dvěma vhodnými frameworky pro tvorbu uživatelského rozhraní, a to Angularem [18] a Reactem [19]. V každém z nich jsem naprogramoval jednoduchou webovou aplikaci, abych se mohl lépe rozhodnout, který pro diplomovou práci použiji. Vybral jsem React, protože umožňoval volnější syntaxi a jednodušší integraci s kódem v JavaScriptu. Jak jsem později zjistil, právě volnější syntaxe, která na rozdíl od Angularu využívajícího TypeScript není typově bezpečná, byla zdrojem četných obtížně odhalitelných chyb.

Uživatelské rozhraní v Reactu využívá knihovnu React ZXing [20] pro rozpoznávání QR kódů při přiřazování QR kódu ke studentům.

Nad uživatelským rozhraním v Reactu je použita další vrstva tvořená kódem v HTML s JavaScriptem. V této vrstvě je napsán editor obrázků využívající standardizované API prohlížeče. Grafy se statistikami o běhu serveru a s přehledem hodnocení studentů jsou vytvořeny pomocí knihovny Chart.js [21]. Tato poslední vrstva komunikuje s uživatelským rozhraním v Reactu pomocí sady aplikačních callbacků v JavaScriptu.

Struktura aplikace je znázorněna graficky na obrázku 3.1.



■ **Obrázek 3.1** Struktura webové aplikace

Aplikace je integrována na následující systémy:

- KOS poskytuje informace o semestrech, předmětech, paralelkách, vyučujících, studentech, zkouškách a událostech.
- OAuth2 brána zajišťuje SSO přihlašování uživatelů a poskytuje tokeny pro komunikaci s Grades.
- Grades eviduje milníky a bodová hodnocení studentů.
- SMTP server je používán k odeslání e-mailů.

V dalším textu jednotlivé části aplikace a integrační rozhraní popíši.

V této kapitole popisují aplikační databázi tak, že vysvětlují roli jednotlivých databázových tabulek a v nich uložených dat. Pro zvýšení přehlednosti jsem popisované tabulky rozčlenil do čtyř skupin, mezi jejichž prvky jsou jen malé závislosti (jedinými tabulkami, které mají roli napříč více skupinami, jsou tabulky **person** a **parallel_milestone**).

S výjimkou tabulek **person** a **local_application_user**, kde je primárním klíčem uživatelské jméno uživatele, jsou v tabulkách pro zjednodušení implementace a vzájemných vazeb mezi tabulkami použity umělé číselné primární klíče generované číselnou řadou.

Referenční integrita mezi tabulkami je s níže popsanými výjimkami vynucena pomocí použití cizích klíčů v závislých tabulkách.

Všude, kde je to vhodné, je vykonávání dotazů podpořeno jednoduchými nebo složenými indexy. Tam, kde je to potřeba, jsou tyto indexy definovány jako unikátní a zabraňují tak nevhodnému vytváření duplicitních záznamů.

4.1 Definice předmětů

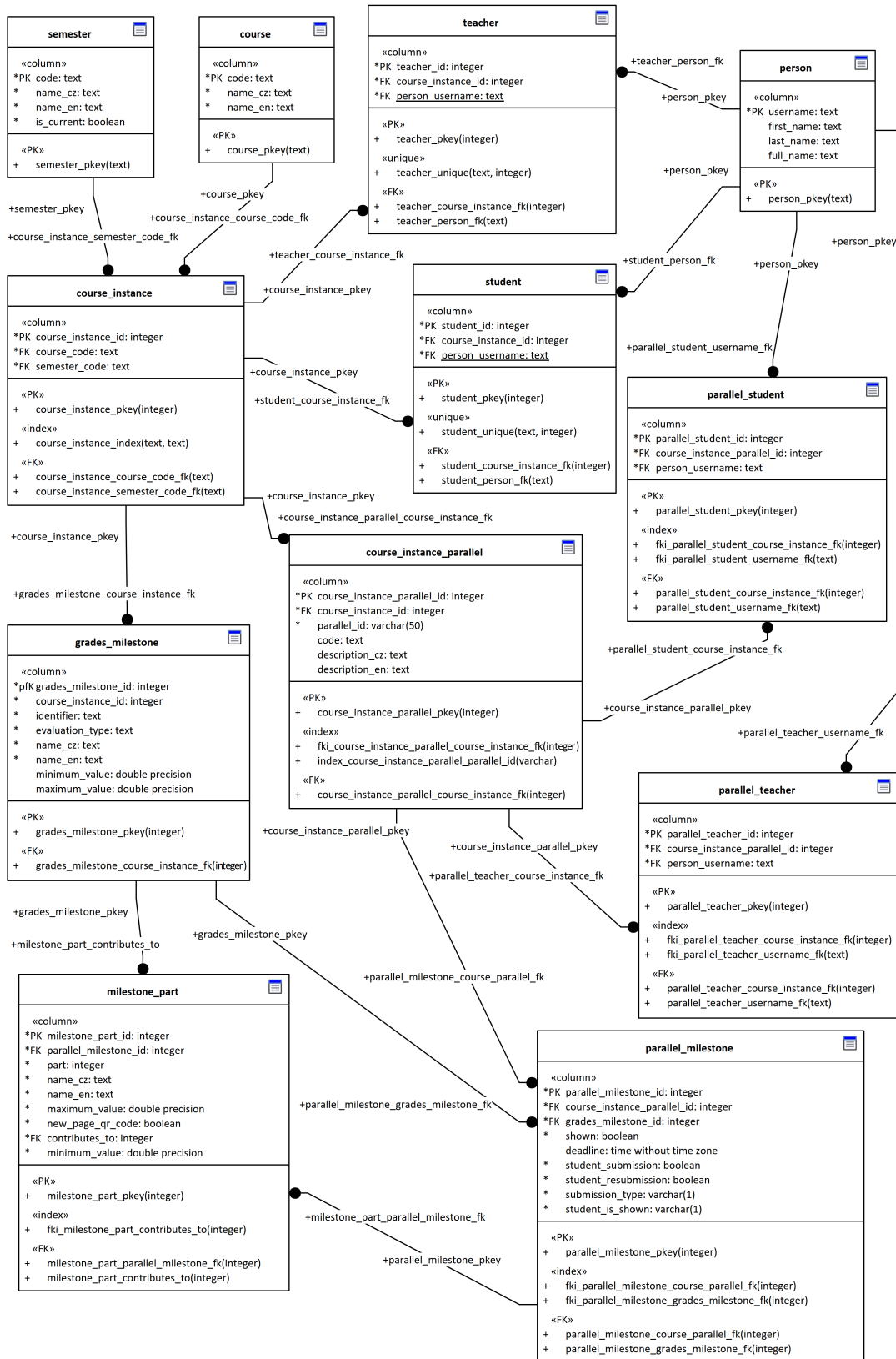
Informace o předmětech jsou importovány ze systému KOS a jsou uloženy spolu s aplikační parametrizací předmětů v tabulkách na obrázku 4.1.

Předmět vyučovaný v daném semestru je definován v tabulce **course_instance**, kdy v tabulkách **semester** je uložen semestr a v tabulce **course** předmět, který může být vyučován ve více semestrech. Údaje v tabulce **course** jsou aktualizovány podle údajů z vyučovaného předmětu v posledním semestru.

V tabulce **course_instance_parallel** jsou uloženy paralelky předmětu, zkoušky a události předmětu v daném semestru. V tabulce **grades_milestone** jsou uloženy milníky importované ze systému Grades a aktualizované podle systému Grades v okamžiku, kdy učitel otevře nastavení předmětu pro daný semestr.

Učitelé a studenti předmětu jsou určeni pomocí tabulek **teacher** a **student**, které definují vazbu mezi tabulkami **course_instance** a **person**. Obdobným způsobem jsou určeni učitelé a studenti paralelek pomocí tabulek **parallel_teacher** a **parallel_student**. Pokud student během semestru změní svoji paralelku u některého z předmětů, aplikace jej zapíše do nové paralelky a ze staré jej odstraní pouze v případě, kdy k němu v jeho původní paralelce nejsou evidována žádná odevzdání nebo hodnocení.

Definice milníku pro jednu paralelku je uložena v tabulce **parallel_milestone**, jeho části pak v tabulce **parallel_part**.



■ Obrázek 4.1 Tabulky definující předměty

4.2 Odevzdání

Odevzdání a jejich opravy nebo hodnocení jsou uložena v tabulkách na obrázku 4.2.

Odevzdání jsou uložena v tabulce **submission** navázané na tabulku **parallel_milestone**. Odevzdání je navázáno na tabulku **person** jak osobou odevzdávajícího, tak osobou, jehož výstupy mohou odevzdání obsahovat.

Každé odevzdání může být tvořeno více soubory popsány v tabulce **submission_file**. Zpracováním souborů vznikají jednotlivé obrázky odpovídající listům odevzdání popsané v tabulce **submission_picture**.

Vrstvy oprav provedené učiteli jsou popsány v tabulce **picture_drawing**, udělené bodové hodnocení a eventuálně platné zámky pro opravování jsou uloženy v tabulce **grading**. Zámek je platný pro kombinaci odevzdání, studenta a části milníku. Referenční integrita vztahu tabulek **grading** a **milestone_part** není zajištěna. Je tak umožněno v případě potřeby počet částí milníku snížit i v případě, kdy k některé části již existuje hodnocení, není to ale doporučený postup.

Jednotlivá odevzdání nabývají postupně stavů uvedených v tabulce 4.1. Stav je popsán pomocí názvu lokalizačního štítku na stránce `submissionState`.

■ **Tabulka 4.1** Stav odevzdání

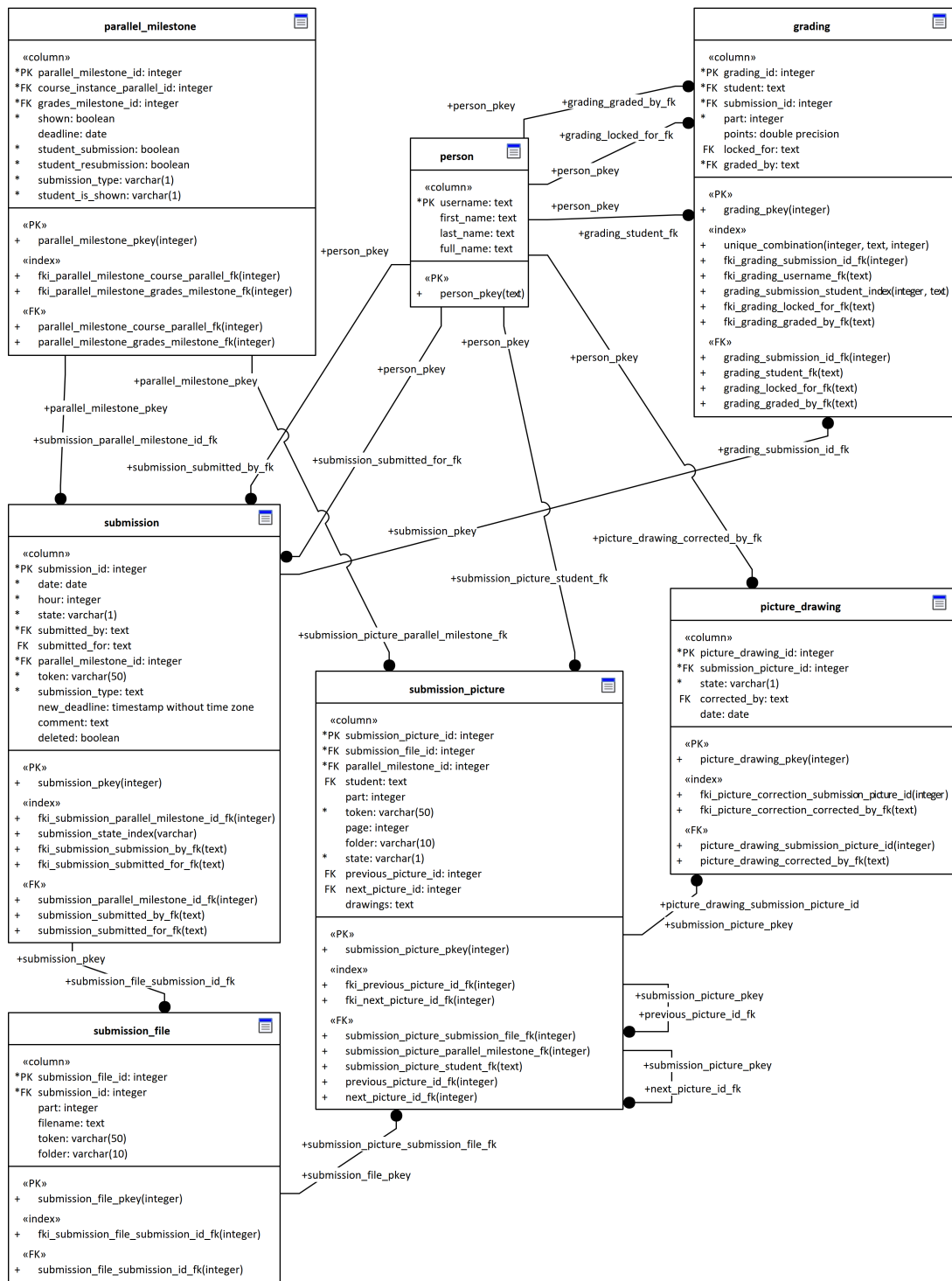
Kód	Stav	Význam
Q	requested	Probíhá odevzdávání a aplikace čeká na všechny soubory a jejich části
S	submitted	Soubory byly odevzdány a uloženy a budou konvertovány na jednotlivé obrázky
N	converted	Konverze souborů na obrázky proběhla a je možné je opravovat
F	conversionFailed	Konverze souborů na obrázky selhala
C	corrected	Konverze souborů na obrázky proběhla a alespoň jeden obrázek je opraven
R	releasedForStudent	Konverze souborů na obrázky proběhla a všechny nesmazané obrázky jsou ve stavu <code>releasedForStudent</code>

Smazané odevzdání má stále původní stav před odevzdáním, pouze je nastaven příznak „deleted“. Tím je umožněno obnovení původního stavu smazaného odevzdání.

Jednotlivé obrázky nabývají stavů uvedených v tabulce 4.2. Stav je popsán pomocí názvu lokalizačního štítku na stránce `submissionCorrection`.

■ **Tabulka 4.2** Stav obrázků

Kód	Stav	Význam
P	notCorrected	Obrázek nebyl opraven
C	corrected	Obrázek byl zobrazen učitelem a následně byl učitelem uložen
D	deleted	Obrázek byl učitelem označen jako smazaný
R	releasedForStudent	Obrázek byl učitelem označen jako uvolněný pro studenty, tyto obrázky z odevzdání ve stavu <code>releasedForStudent</code> jsou studentům zobrazovány

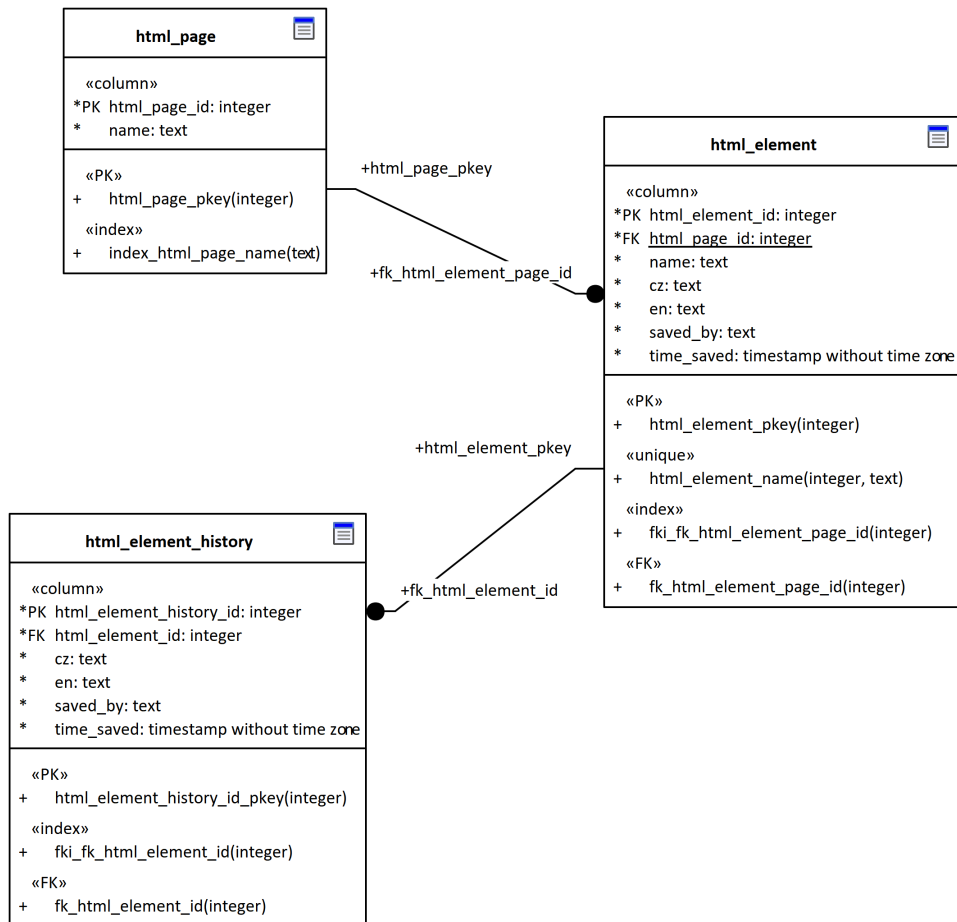


■ Obrázek 4.2 Tabulky sloužící pro odevzdání

4.3 Lokalizace

Naprostá většina textů na uživatelském rozhraní je definována pomocí štítků, jejichž hodnoty v českém a anglickém jazyce jsou uloženy v databázi v tabulkách na obrázku 4.3.

V tabulce **html_page** je uložen název stránky, na které je štítek použit. V některých případech se jedná o stav (např. `submissionState`) nebo část stránky (např. `editor`, `footer`). V tabulce **html_element** je uložen text štítku v obou jazykových mutacích. V tabulce **html_element_history** jsou pak dostupné minulé hodnoty štítků.



■ Obrázek 4.3 Tabulky pro uložení lokalizace

4.4 Pomocné tabulky

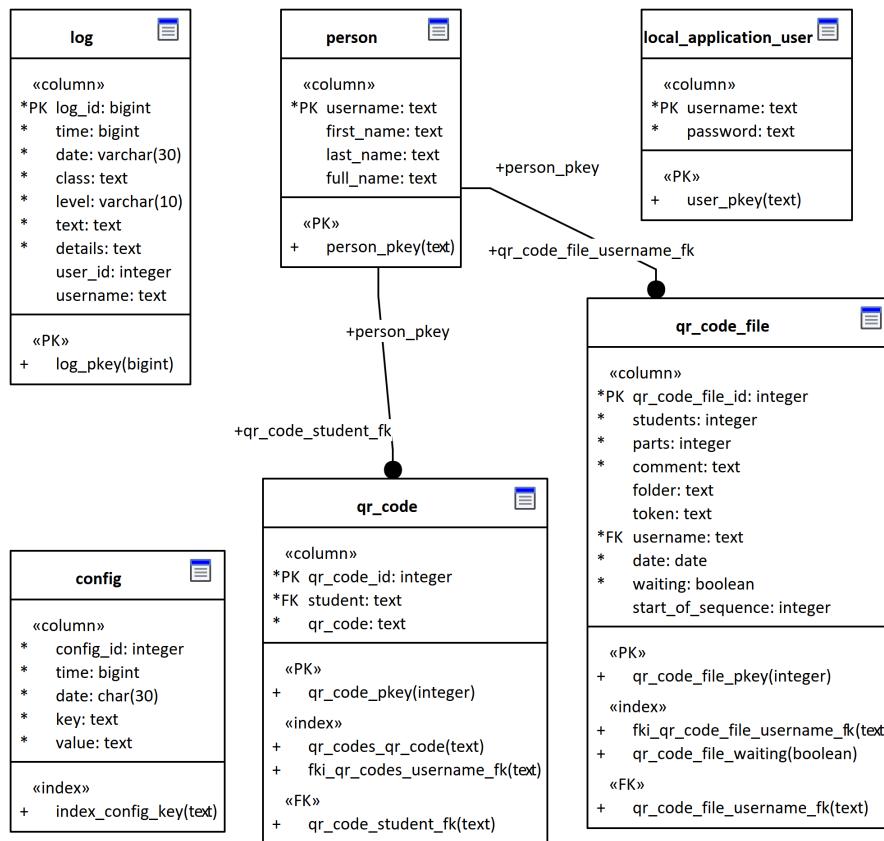
V databázi jsou pomocné tabulky zachycené na obrázku 4.4.

Logování je provedeno do souborů na aplikačním serveru. Přístupy na většinu HTTP služeb jsou navíc zapisovány do tabulky **log**. Nejstarší záznamy přesahující počet 100.000 záznamů jsou pravidelně mazány.

V tabulce **local_application_user** jsou uloženy přihlašovací údaje lokálně definovaných uživatelů. Přihlašovací jméno je uvedeno v čitelné podobě a jeho referenční integrita vůči ostatním tabulkám není zajištěna, heslo je zašifrováno jednosměrně použitelnou šifrou. Samotné prozrazení obsahu tabulky neumožňuje případnému útočníkovi přihlášení do aplikace.

V tabulce **qr_code** je uložena vazba mezi QR kódem a osobou (praktický význam to má pouze pro studenty). Jedna osoba může mít definováno libovolné množství QR kódů, které ji jednoznačně identifikují.

V tabulce **qr_code_file** jsou uvedeny údaje o vygenerovaných souborech s QR kódy. Žádný mechanismus odstraňování starých záznamů není implementován, protože není jasné, jaká bude praxe využívání vygenerovaných souborů s QR kódy.



■ Obrázek 4.4 Pomocné tabulky

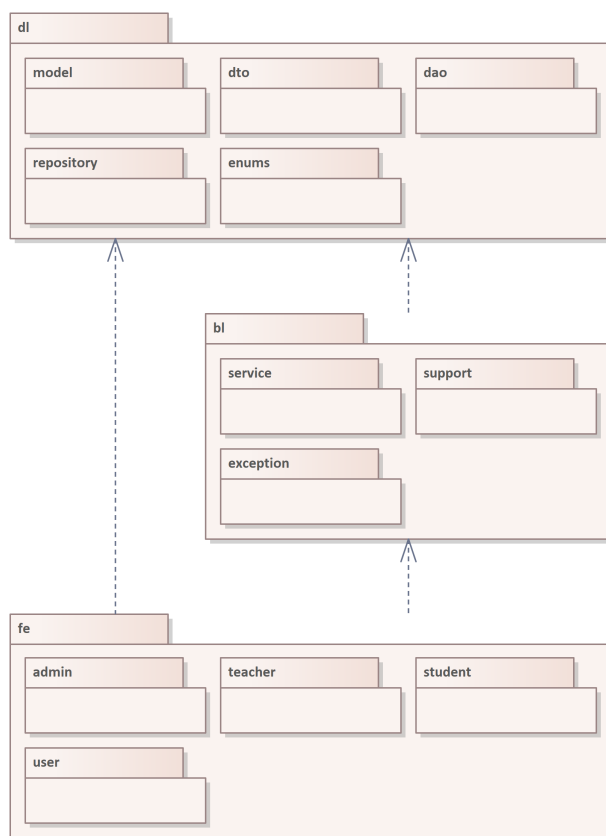
V tabulce **config** jsou uloženy konfigurační údaje popsané v tabulce 4.3.

■ **Tabulka 4.3** Záznamy v konfigurační tabulce config

Klíč	Výchozí stav	Význam
defWidth	2480	Šířka obrázku v bodech, na kterou jsou zvětšeny nebo zmenšeny odevzdané obrázky
dpi	300	Rozlišení konverze PDF souborů na obrázky
emailAuth	false	Použití autentizace e-mailového serveru
emailDomain	@fit.cvut.cz	Doména přihlašovacích jmen uživatelů
emailHost	relay.in.fit.cvut.cz	Adresa e-mailového serveru
emailing	true	Hodnota „false“ vypne odesílání e-mailů
emailPassword		Heslo pro přihlášení k e-mailovému serveru
emailPort	25	Port e-mailového serveru
emailStartTls	true	Nastavení protokolu start TLS
emailUsername	no-reply@fit.cvut.cz	Adresa odesílatele e-mailů
kosClientId		Hodnota Client ID pro připojení ke KOS API
kosClientSecret		Hodnota Client Secret pro připojení ke KOS API
kosSyncRequested	0	Hodnota 1 způsobí mimořádnou synchronizaci s KOsem
lastKosSyncFinish		Datum a čas dokončení poslední synchronizace s KOsem
lostKosSyncStart		Datum a čas začátku poslední synchronizace s KOsem
lastKosSyncStartMillis	0	Časový záznam o poslední synchronizaci s KOsem pro řízení frekvence synchronizace
passwordSalt		Textový řetězec používaný k šifrování hesel místních uživatelů
qr_sequence		Čítač vygenerovaných QR kódů
ssoClientId		Hodnota Client ID pro připojení k OAuth2 bráně
ssoClientSecret		Hodnota Client Secret pro připojení k OAuth2 bráně
ssoRedirectUrl	https://assignments.kam.fit.cvut.cz/api/oauth20	Adresa aplikace, kam je uživatel přesměrován po přihlášení
superAdminInitialPassword		První heslo uživatele superadmin

Aplikační server

Aplikační server je napsán v Javě 11, využívá frameworky Spring Boot a Spring Security a napojení na databázi je řešeno pomocí frameworku Hibernate ORM. Jednotlivé databázové dotazy jsou realizovány jako transakce.



■ **Obrázek 5.1** Architektura aplikačního serveru

Výběr verze Javy je dán použitím knihovny `openhtmltopdf` [16], která je dle autora knihovny testována právě pro Javu 11.

Architektura aplikačního serveru vychází z třívrstvé aplikační architektury a je zachycena na

obrázku 5.1:

- **dl** („data layer“) obsahuje balíčky **model** s definicemi databázových objektů, **dto** s objekty používanými ke komunikaci s uživatelským rozhraním, **dao** s vlastními metodami pro přístup k datům, **repository** pro využití standardních přístupů k datům a nakonec **enums** s číselníky.
- **bl** („business layer“) obsahuje v balíčku **service** služby pro odbavení požadavků, **support** další podpůrné třídy a v balíčku **exception** definice výjimek.
- **fe** („front end“) obsahuje HTTP kontrolery roztríděné do balíčků **admin**, **teacher**, **student** a **user** podle toho, jaká role je pro využití služeb potřeba. Služby dostupné nepřihlášenému uživateli jsou uloženy ve třídě přímo v balíčku **fe** stejně jako definice interceptoru pro logování přístupů.

Konfigurační třídy Spring Boot aplikace, Reactu a Spring Security jsou uloženy v kořenovém adresáři.

5.1 Spring Security

Přístupy k jednotlivým službám jsou chráněny pomocí Spring Security s využitím tříd uživatelů popsaných v tabulce 5.1:

■ **Tabulka 5.1** Třídy uživatelů Spring Security

Třída	Popis
USER	Uživatel je přihlášen lokálním přihlášením nebo prostřednictvím školního SSO přihlášení
TEACHER	Uživatel je přihlášen a existuje alespoň jeden předmět, který vyučuje
STUDENT	Uživatel je přihlášen a existuje alespoň jeden předmět, který studuje
ADMIN	Uživatel je přihlášen lokálním přihlášením

Uživatel může mít více rolí současně.

V případě opravování písemek je přístup ke službám rozvolněn, protože požadavkem zadavatele bylo, aby písemky mohl opravovat libovolný uživatel. Takový uživatel nemusí mít roli TEACHER, ale musí disponovat klíčem pro opravování daného odevzdání.

Při generování dat pro služby poskytované uživatelskému rozhraní aplikace vždy kontroluje, zda uživatel použil klíč nebo zda je vyučujícím nebo studentem daného předmětu.

Hesla lokálních uživatelů jsou uložena do databáze jako hashe vytvořené funkcí Springu BCryptPasswordEncoder() a funkcí Javy sha512() s využitím soli uložené v aplikační konfiguraci. Aplikace při každém svém startu kontroluje, zda je definován lokální uživatel „superadmin“. Pokud tomu tak není, tak jej vytvoří s prvním heslem podle aplikační konfigurace.

5.2 Poskytované služby

Aplikace poskytuje pro uživatelské rozhraní služby vyjmenované níže.

5.2.1 Veřejně poskytované služby

Aplikace veřejně poskytuje služby uvedené v tabulce 5.2.

■ **Tabulka 5.2** Veřejně poskytované služby

Metoda	Služba	Popis
GET	/api/oauth20	Přihlášení uživatele přes SSO
GET	/api/public/csrf	Získání CSRF tokenu Spring Security
GET	/api/public/localization	Získání překladové tabulky pro popisy
POST	/api/public/login	Lokální přihlášení uživatele
DELETE	/api/public/session	Odhlášení uživatele

5.2.2 Služby pro uživatele třídy USER

Aplikace poskytuje pro uživatele třídy USER služby uvedené v tabulce 5.3.

■ **Tabulka 5.3** Služby pro uživatele třídy USER

Metoda	Služba	Popis
GET	/api/user/drawing/{pictureId}/ {pictureToken}	Načtení opravy obrázku
GET	/api/user/editor	Získání HTML kódu editoru obrázků
GET	/api/user/editor.js	Získání kódu editoru obrázků v JavaScriptu
POST	/api/user/fileupload/{submissionId}	Potvrzení kompletního odeslání souboru odevzdání
POST	/api/user/fileupload/{submissionId}/ {submissionFileId}	Odeslání části souboru odevzdání
GET	/api/user/picture/{id}/{token}	Stažení obrázku editorem obrázků nebo prohlížečem obrázků
GET	/api/user/picturedownload/{id}/ {token}	Stažení obrázku uživatelem
GET	/api/user/semester	Získání seznamu semestrů
GET	/api/user/smallpicturedownload/{id}/ {token}	Stažení náhledu obrázku
GET	/api/user/submissionfiledownload/ {id}/{token}	Stažení souboru odevzdání
GET	/api/user/userinfo	Získání CSRF tokenu, uživatelského jména a jeho rolí
GET	/api/user/viewer	Získání HTML kódu prohlížeče obrázků
GET	/api/user/viewer.js	Získání kódu prohlížeče obrázků v JavaScriptu

5.2.3 Služby pro uživatele třídy TEACHER

Aplikace poskytuje pro uživatele třídy TEACHER služby uvedené v tabulkách 5.4 a 5.5.

■ **Tabulka 5.4** Služby pro uživatele třídy TEACHER – část 1

Metoda	Služba	Popis
PUT	/api/teacher/assignstudent/ {semesterCode}/ {courseCode}/{parallelId}	Přiřazení QR kódu studentovi
GET	/api/teacher/course/{semesterCode}	Načtení předmětů semestru
GET	/api/teacher/course/{semesterCode}/ {courseCode}	Načtení detailu předmětu
GET	/api/teacher/drawing/{pictureId}/ {submissionToken}/{drawingId}	Načtení vrstvy s opravou
GET	/api/teacher/exam/{semesterCode}/ {courseCode}	Načtení zkoušek a událostí
PUT	/api/teacher/exam/{semesterCode}/ {courseCode}	Žádost o vložení zkoušky nebo události k předmětu
GET	/api/teacher/grades/{semesterCode}/ {courseCode}/{username}	Načtení Grades
PUT	/api/teacher/grades/{semesterCode}/ {courseCode}	Zápis do Grades
GET	/api/teacher/chart/{semesterCode}/ {courseCode}	Načtení dat pro prezentaci výsledků
GET	/api/teacher/lock/{submissionId}/ {token}	Načtení zámku
PUT	/api/teacher/lock/{submissionId}/ {token}	Vytvoření zámku
DELETE	/api/teacher/lock/{submissionId}/ {token}/{pictureId}	Uvolnění zámku
GET	/api/teacher/milestone/ {semesterCode}/{courseCode}	Načtení milníků
PUT	/api/teacher/parallel/{semesterCode}/ {courseCode}/{parallelId}	Uložení nastavení odevzdávání
GET	/api/teacher/parallelstudents/ {semesterCode}/{courseCode}/ {parallelId}	Načtení studentů
GET	/api/teacher/qrcodefile	Načtení souborů s QR kódy
POST	/api/teacher/qrcodefile	Žádost o generování souboru s QR kódy
GET	/api/teacher/qrcodefiledownload/{id}/ {token}	Stažení souboru s QR kódy
Post	/api/teacher/savecorrections/ {submissionId}/{submissionToken}	Uložení vrstvy s opravou
GET	/api/teacher/submission/ {semesterCode}/{courseCode}/ {parallelId}	Načtení odevzdání bez klíče
GET	/api/teacher/submission/ {submissionId}/{token}	Načtení odevzdání s klíčem
PUT	/api/teacher/submission/delete/ {semesterCode}/{courseCode}/ {parallelId}/{submissionId}	Nastavení odevzdání na smazané

■ **Tabulka 5.5** Služby pro uživatele třídy TEACHER – část 2

Metoda	Služba	Popis
PUT	/api/teacher/submission/release/ {semesterCode}/{courseCode}/ {parallelId}/{submissionId}	Uvolnění odevzdání pro studenty
PUT	/api/teacher/submission/undelede/ {semesterCode}/{courseCode}/ {parallelId}/{submissionId}	Odstranění příznaku smazaného odevzdání
PUT	/api/teacher/submissiondeadline/ {submissionId}/{token}	Nastavení nového termínu odevzdání
POST	/api/teacher/submissionrequest	Žádost o nové odevzdání učitelem

5.2.4 Služby pro uživatele třídy STUDENT

Aplikace poskytuje pro uživatele třídy STUDENT služby uvedené v tabulce 5.6.

■ **Tabulka 5.6** Služby pro uživatele třídy STUDENT

Metoda	Služba	Popis
GET	/api/student/course/{semesterCode}	Načtení předmětů semestru
GET	/api/student/course/{semesterCode}/ courseCode	Načtení detailu předmětu
GET	/api/student/submission/ {semesterCode}/{courseCode}/ {parallelId}/{milestoneIdentifier}	Načtení odevzdání
POST	/api/student/submissionrequest	Žádost o nové odevzdání studentem

5.2.5 Služby pro uživatele třídy ADMIN

Aplikace poskytuje pro uživatele třídy ADMIN služby uvedené v tabulce 5.7.

■ **Tabulka 5.7** Služby pro uživatele třídy ADMIN

Metoda	Služba	Popis
PUT	/api/admin/kosconnection	Nastavení připojení ke KOSu
GET	/api/admin/kossync	Načtení informací o synchronizaci s KOSem
POST	/api/admin/kossync	Žádost o synchronizaci s KOSem
PUT	/api/admin/localization/element	Aktualizace obsahu štítku
GET	/api/admin/localization/element/{elementId}	Načtení historie štítku
GET	/api/admin/localization/new	Načtení lokalizace se smazáním cache
GET	/api/admin/localization/page	Načtení stránek lokalizace
POST	/api/admin/localization/page	Vytvoření nové stránky lokalizace
GET	/api/admin/localization/page/{pageId}	Načtení stránky lokalizace
POST	/api/admin/localization/page/{pageId}/element	Vložení nového štítku
GET	/api/admin/log	Načtení záznamů logu
PATCH	/api/admin/log	Filtrování záznamů logu
PUT	/api/admin/password	Změna hesla lokálního uživatele
PUT	/api/admin/ssconnection	Nastavení připojení ke OAuth2 bráně
GET	/api/admin/statistics	Načtení provozních údajů
GET	/api/admin/user	Načtení lokálních uživatelů
PUT	/api/admin/user	Nastavení hesla lokálního uživatele
POST	/api/admin/user	Vytvoření lokálního uživatele
PATCH	/api/admin/user	Smazání lokálního uživatele

Integrace

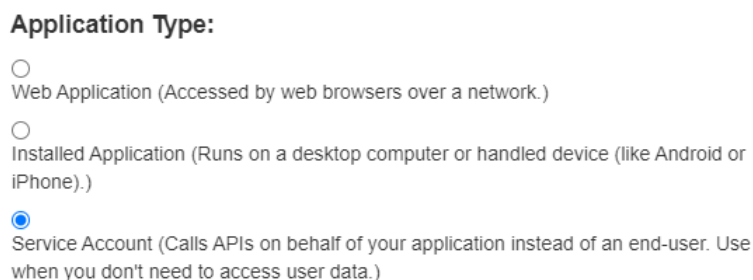
Aplikace je integrována na školní systém KOS, fakultní aplikaci Grades, školní přihlašovací bránu pro SSO přihlášení a školní e-mailový server. V této kapitole popíši podrobnosti této integrace.

Integrační rozhraní se ve školním prostředí nastavují pomocí Apps Managera [22] na adrese <https://auth.fit.cvut.cz/manager/index.xhtml>. Funkční konfiguraci se mi nepodařilo získat z žádné dostupné dokumentace a její dosažení vyžadovalo velké množství pokusů. Aplikace pro opravování písemek potřebuje z hlediska Apps Managera nastavit dva různé projekty. Apps Manager umožňuje vytvořit v rámci jednoho projektu dvě aplikace, ale tímto způsobem jsem funkčnosti nedosáhl.

6.1 Integrace na systém KOS

6.1.1 Nastavení integrace pomocí Apps Managera

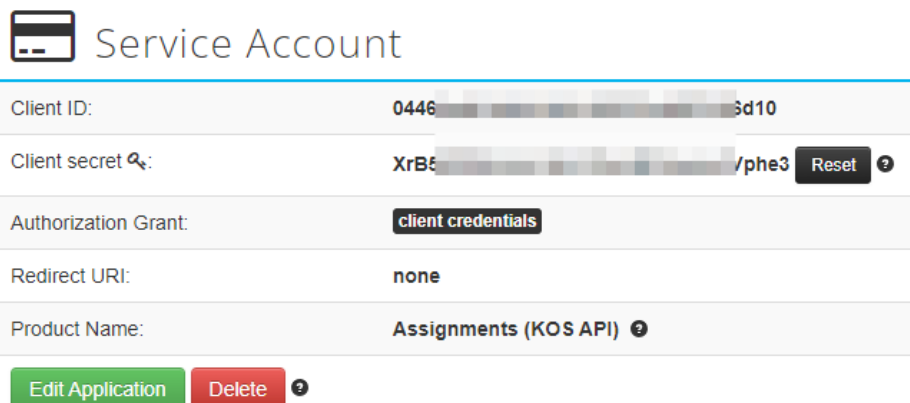
V projektu založeném v Apps Manageru je potřeba vytvořit aplikaci typu „Service account“ tak, jak je vidět na obrázku 6.1.



■ **Obrázek 6.1** Založení aplikace typu Service account

Apps Manager vygeneruje přístupové údaje a zobrazí je v tabulce na obrázku 6.2. Ty je nutné vložit do aplikace pro opravování písemek.

Dále je potřeba aplikaci nastavit přístupy na rozhraní KOS API. Správné nastavení, které funguje, je takové, kdy jsou přístupy zakázány tak, jak je vidět na obrázku 6.3. Aplikace přitom používá scope `urn:ctu:oauth:kosapi:public.readonly`, jehož použití není v nastavení povoleno. Opakovaně a různými kanály jsem se snažil nastavení přístupů vyjasnit, ale přes značné úsilí jsem nedosáhl žádných výsledků.

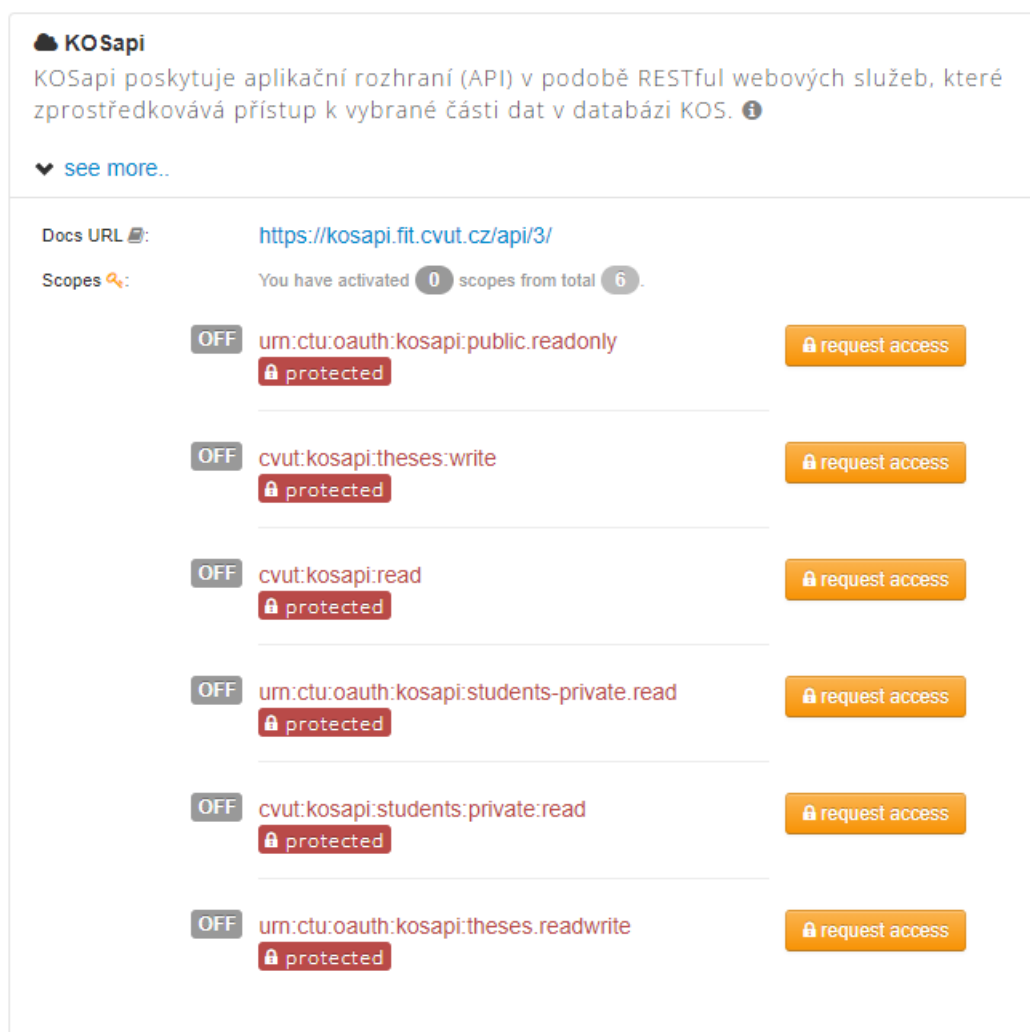


Service Account

Client ID:	0446[redacted]3d10
Client secret 🔑:	XrB5[redacted]/phe3 Reset ⓘ
Authorization Grant:	client_credentials
Redirect URI:	none
Product Name:	Assignments (KOS API) ⓘ

Edit Application Delete ⓘ

■ **Obrázek 6.2** Zobrazení přístupových údajů pro připojení ke KOSu



KOSapi
KOSapi poskytuje aplikační rozhraní (API) v podobě RESTful webových služeb, které zprostředkovává přístup k vybrané části dat v databázi KOS. ⓘ

▼ [see more..](#)

Docs URL 📄: <https://kosapi.fit.cvut.cz/api/3/>

Scopes 🔑: You have activated 0 scopes from total 6.

<input type="checkbox"/>	um:ctu:oauth:kosapi:public.readonly 🔒 protected	request access
<input type="checkbox"/>	cvut:kosapi:theses:write 🔒 protected	request access
<input type="checkbox"/>	cvut:kosapi:read 🔒 protected	request access
<input type="checkbox"/>	um:ctu:oauth:kosapi:students-private.read 🔒 protected	request access
<input type="checkbox"/>	cvut:kosapi:students:private:read 🔒 protected	request access
<input type="checkbox"/>	um:ctu:oauth:kosapi:theses.readwrite 🔒 protected	request access

■ **Obrázek 6.3** Nastavení přístupu na KOS API

6.1.2 Využívané služby

Využívané služby KOS API jsem zprovoznil s využitím on-line dokumentace [23]. V popisu využívaných služeb používám štítky uvedené v tabulce 6.1.

■ **Tabulka 6.1** Štítky v definici služeb pro přístup ke KOS API

Štítek	Popis
{clientId}	Client ID z nastavení přístupu
{clientSecret}	Client secret z nastavení přístupu
{token}	Token získaný z auth.fit.cvut.cz
{offset}	Stránkování při získávání údajů s počtem záznamů přes 1000 položek
{semesterCode}	Identifikátor semestru
{courseCode}	Identifikátor předmětu
{examCode}	Identifikátor zkoušky nebo události

Při přístupu k systému KOS aplikace získá token pomocí služby uvedené v tabulce 6.2 a tento token používá pro služby uvedené v tabulce 6.3. Výběr těchto služeb nebyl jednoduchý – musel jsem vybrat služby, které dávaly požadované údaje a navíc umožňovaly v případě velkého počtu záznamů řadit podle identifikátoru záznamu a nalezené údaje stránkovat. Použitelnou sadu z různých možných kombinací služeb se mi podařilo sestavit se značným úsilím.

■ **Tabulka 6.2** Služba pro získání tokenu pro KOS API

Služba	URL
Získání tokenu	<code>https://auth.fit.cvut.cz/oauth/oauth/token?grant_type=client_credentials&client_id={clientId}&client_secret={clientSecret}&scope=urn:ctu:oauth:kosapi:public.readonly</code>

■ **Tabulka 6.3** Využívané služby KOS API

Služba	URL
Získání seznamu předmětů	https://kosapi.fit.cvut.cz/api/3/divisions/18000/courses?access_token={token}&limit=1000&offset={offset}&sem={semesterCode}
Získání údajů o předmětu	https://kosapi.fit.cvut.cz/api/3/courses/{courseCode}?access_token={token}&sem={semesterCode}&multilang=true
Získání seznamu studentů předmětu	https://kosapi.fit.cvut.cz/api/3/courses/{courseCode}/students?access_token={token}&sem={semesterCode}&limit=1000&offset={offset}&orderBy=id
Získání seznamu paralelek předmětu	https://kosapi.fit.cvut.cz/api/3/courses/{courseCode}/parallels?access_token={token}&sem={semesterCode}&limit=1000
Získání studentů paralelky předmětu	https://kosapi.fit.cvut.cz/api/3/parallels/{parallelCode}/students?access_token={token}&sem={semesterCode}&limit=1000&offset={offset}&orderBy=id
Získání seznamu zkoušek předmětu	https://kosapi.fit.cvut.cz/api/3/courses/{courseCode}/exams?access_token={token}&sem={semesterCode}&multilang=true&limit=1000
Získání seznamu studentů zkoušky	https://kosapi.fit.cvut.cz/api/3/exams/{examCode}/attendees?access_token={token}&multilang=true&limit=1000&offset={offset}
Získání seznamu událostí předmětu	https://kosapi.fit.cvut.cz/api/3/courses/{courseCode}/events?access_token={token}&sem={semesterCode}&multilang=true&limit=1000
Získání seznamu studentů události	https://kosapi.fit.cvut.cz/api/3/courseEvents/{examCode}/attendees?access_token={token}&multilang=true&limit=1000&offset={offset}

6.2 Integrace na aplikaci Grades a SSO

6.2.1 Nastavení integrace pomocí Apps Managera

V projektu založeném v Apps Manageru je potřeba vytvořit aplikaci typu „web application“ tak, jak je vidět na obrázku 6.4.

Application Type:

Web Application (Accessed by web browsers over a network.)

Installed Application (Runs on a desktop computer or handled device (like Android or iPhone).)


Service Account (Calls APIs on behalf of your application instead of an end-user. Use when you don't need to access user data.)




[Learn more about Applications Types?](#)


Redirection URI:

■ **Obrázek 6.4** Založení aplikace typu Web application

Apps Manager vygeneruje přístupové údaje a zobrazí je v tabulce na obrázku 6.5. Ty je nutné vložit do aplikace pro opravování písemek.

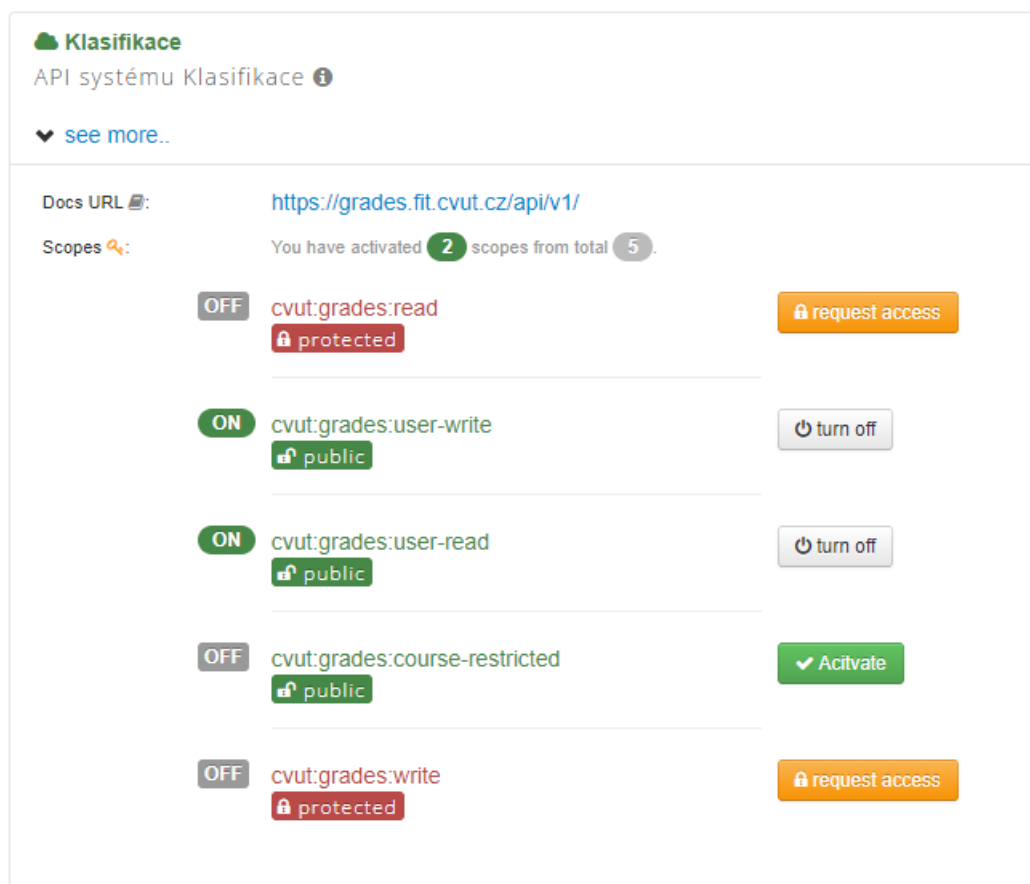
 **Web Application**

Client ID:	890 [redacted] b0d
Client secret 	FM [redacted] YJKM Reset 
Authorization Grant:	authorization code refresh token
Redirect URI:	https://assignments.kam.fit.cvut.cz/api/oauth20
Product Name:	Assignments (Grades Read Write) 

Edit Application Delete 

■ **Obrázek 6.5** Zobrazení přístupových údajů pro SSO přihlášení pro připojení ke Grades

Dále je potřeba aplikaci nastavit přístupy na rozhraní Grades. Správné nastavení je zobrazeno na obrázku 6.6.



■ **Obrázek 6.6** Nastavení přístupu na Grades

Tato integrace má následující využití:

- Umožňuje přihlášení uživatele přes SSO školní bránu. Aplikace zjistí uživatelské jméno přihlášeného uživatele, podle toho v údajích importovaných z KOSu dohledá, zda se jedná o učitele a/nebo o studenta, a přidělí mu aplikační uživatelské role.
- Přihlášeným učitelům umožňuje číst data z aplikace Grades a zapisovat data do aplikace Grades. Aplikace pro opravování písemek pro tento účel po celou dobu trvání přihlášení učitele udržuje a včas obnovuje platný token pro přístup do aplikace Grades.

Pro jednoduchost se studenti do aplikace přihlašují také s požadovanými právy zapisovat a číst z Grades, tento přístup ale není v aplikaci pro opravování písemek využit. Pokud by to bylo potřeba, je možné v aplikaci pro opravování písemek pro studenty připravit jiné přihlašovací tlačítko, které bude umožňovat jejich přihlašování bez požadavku na práva čtení a zápisu vzhledem k aplikaci Grades.

6.2.2 Využívané služby

Využívané služby aplikace Grades a přihlašovací brány jsem zprovoznil s využitím on-line dokumentace [24] a [25].

V popisu využívaných služeb používám štíty uvedené v tabulce 6.4.

■ **Tabulka 6.4** Štítky v definici služeb Grades a SSO

Štítek	Popis
{clientId}	Client ID z nastavení přístupu
{state}	Identifikátor session generovaný s dostatečnou mírou entropie
{code}	Kód vrácený přihlašovací bránou
{redirect}	Redirect URL z nastavení přístupu
{courseCode}	Kód předmětu
{clientSecret}	Client secret z nastavení přístupu
{refreshToken}	Obnovovací token pro získání nového tokenu získaný z auth.fit.cvut.cz
{token}	Token získaný z auth.fit.cvut.cz
{courseCode}	Identifikátor předmětu
{username}	Uživatelské jméno studenta
{semesterCode}	Identifikátor semestru

Při přístupu ke Grades KOS aplikace získá token a username přihlášeného studenta pomocí služeb přihlašovací brány uvedené v tabulce 6.5 a tento token používá pro služby uvedené v tabulce 6.6.

■ **Tabulka 6.5** Využívané služby SSO

Služba	URL
Přesměrování na SSO přihlášení	<code>https://auth.fit.cvut.cz/oauth/oauth/authorize?response_type=code&client_id={clientId}&state={state}&redirect_uri={redirect}</code>
Získání tokenu	<code>https://auth.fit.cvut.cz/oauth/oauth/token?grant_type=authorization_code&code={code}&redirect_uri={redirect}&client_id={clientId}&client_secret={clientSecret}</code>
Obnova tokenu	<code>https://auth.fit.cvut.cz/oauth/oauth/token?grant_type=refresh_token&refresh_token={refreshToken}&client_id={clientId}&client_secret={clientSecret}</code>
Kontrola platnosti tokenu a získání uživatelského jména přihlášeného uživatele	<code>https://auth.fit.cvut.cz/oauth/check_token?token={token}</code>

■ **Tabulka 6.6** Využívané služby Grades

Služba	URL
Čtení záznamů Grades pro jednoho studenta	<code>https://grades.fit.cvut.cz/api/v1/public/courses/{courseCode}/student-classifications/{username}?semester={semesterCode}&showHidden=false</code>
Čtení milníků předmětu	<code>https://grades.fit.cvut.cz/api/v1/public/courses/{courseCode}/definitions?semester={semesterCode}&showHidden=false</code>
Zápis bodů do Grades	<code>https://grades.fit.cvut.cz/api/v1/public/courses/{courseCode}/student-classifications?semester={semesterCode}</code>

6.3 Integrace na e-mailový server

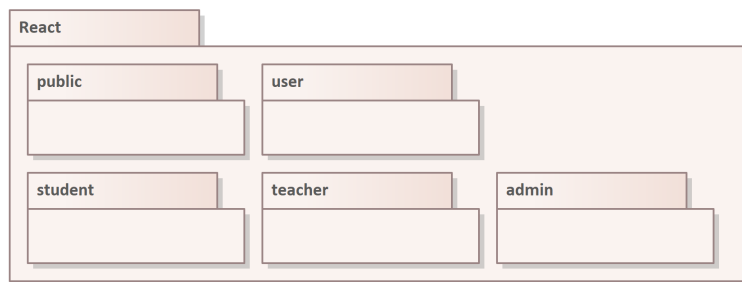
Integrace na školní e-mailový server je přímočará a je implementována pomocí standardních nástrojů Javy. Funkčnost e-mailového serveru je podmíněna registrací ipv4 adresy aplikačního serveru k e-mailovým službám pomocí školního helpdesku.

Aplikace aktuálně odesílá e-maily pouze po vygenerování zkušebních archů s QR kódy. Text e-mailu je definován pomocí štítku v nastavení lokalizace stránky generující QR kódy.

Struktura kódu uživatelského rozhraní

7.1 Kód v Reactu

Struktura kódu uživatelského rozhraní v Reactu je zachycena na obrázku 7.1.



Obrázek 7.1 Struktura kódu v Reactu

V balíčku **public** je uložen kód pro veřejně dostupné obrazovky a veřejně dostupné komponenty, jako je úvodní stránka aplikace, přihlašovací obrazovka, patička, komponenta pro komunikaci s aplikačním serverem, komponenta zobrazující točivé kolečko indikující stav, kdy aplikace pracuje, a komponenta pro zobrazování potvrzujících hlášek v rohu obrazovky.

V balíčku **user** je uložen kód využívaný přihlášenými uživateli: Komponenta dynamicky generovaného aplikačního menu podle rolí přihlášeného uživatele, obrazovka pro sestavení nového odevzdání, tedy vložení souborů k odeslání a zajištění jejich přenosu na aplikační server, a obrazovka pro zobrazení opraveného obrázku.

Kód pro odeslání souboru na server vychází z kódu mojí bakalářské práce [7]. Na rozdíl od bakalářské práce předpokládám, že uživatelé aplikace budou mít k dispozici dostatečně kapacitní datové linky, a proto jsem z kódu pro odesílání souboru odstranil části pro řízení toku dat, které značně zvyšovaly složitost kódu a komplikovaly by jeho budoucí změny.

V balíčku **student** je uložen kód stránek využívaných pouze studenty, tedy zobrazení předmětů a jejich nebo jim přiřazených odevzdání.

V balíčku **teacher** je uložen kód stránek využívaných pouze učiteli, tedy zobrazení předmětů a paralelek, jejich nastavení, zobrazení odevzdání, generování QR kódů a jejich přiřazení studentům, vlastní opravování písemek, zapisování výsledků do Grades a zobrazení výsledků v grafické podobě.

V balíčku **admin** je uložen kód stránek využívaných pouze správci, tedy zobrazení logu, provozních údajů, systémová nastavení, jazyková lokalizace a správa lokálních uživatelů včetně změny hesla.

7.2 Datový model editoru obrázků

Datový model editoru obrázků má pro aplikaci klíčový význam, protože umožňuje jednak provádět editační operace, jednak data o opravách ukládat, načítat a měnit. Datový model jsem navrhl a implementoval tak, aby všechny operace kromě mazání měly konstantní složitost. Při operaci mazání editor počítá průsečík mazané oblasti s existujícími objekty a složitost je lineární k počtu objektů v obrázku.

Pomocí editoru pak lze snadno do obrázku vkládat další objekty a jejich části mazat tak, jak je vidět na obrázku 7.2 (obrázek byl pořízen při testu operací s velkým množstvím objektů; čáry pořízené stylusem jsou kostrbatější / více lomené než ve výsledné verzi aplikace, kdy je trasa pohybu stylusu odečítána s využitím více bodů).



■ **Obrázek 7.2** Příklad editace pomocí editoru při testu operací s velkým množstvím objektů

Datový model (ve skutečnosti upravený třídní model) je zachycen na obrázku 7.3.

Aplikace drží právě jednu instanci třídy **States** (proměnnou *states*), která ukazuje na první stav a aktuální stav editoru reprezentovaný třídou **State**.

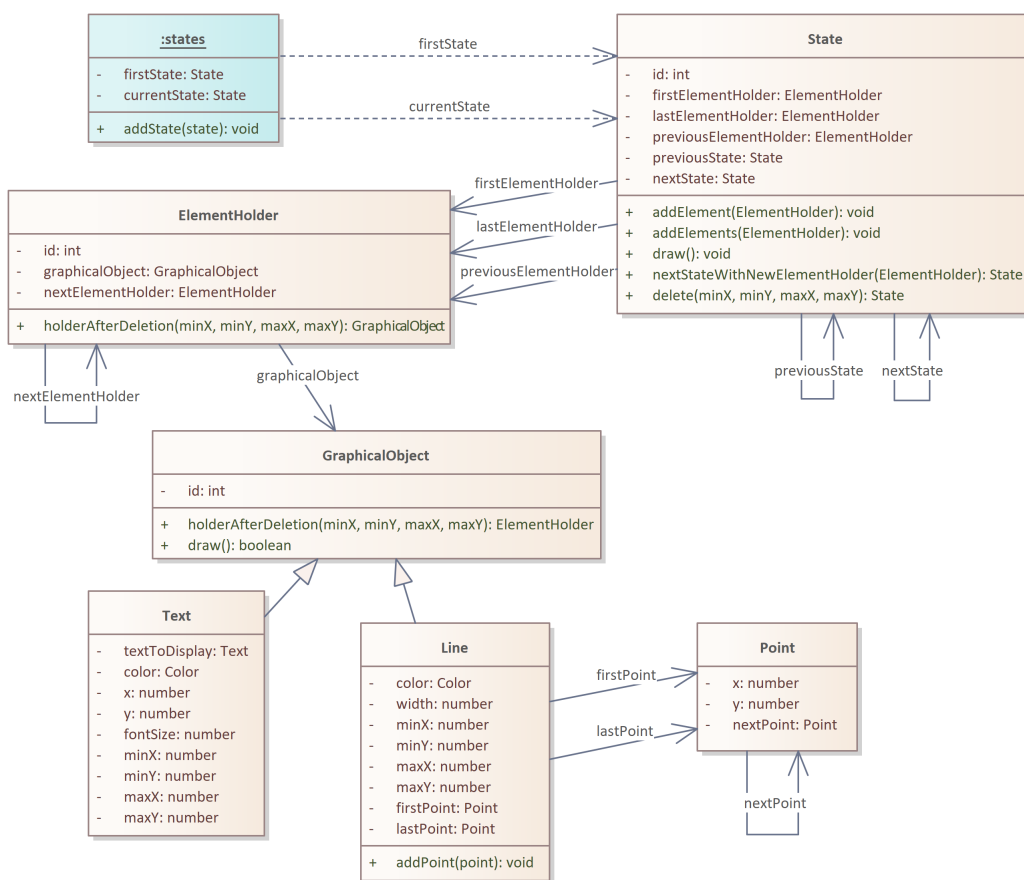
Stav editoru představuje prvky spojového seznamu v možným pohybem v časové posloupnosti stavů vpřed (*nextState*) i vzad (*previousState*). Tak je zajištěna podpora operací „redo“ a „undo“.

Každý stav editoru ukazuje na spojový seznam instancí třídy **ElementHolder**. Používá celkem tři pointery – na první instanci seznamu, na poslední instanci seznamu a na předposlední instanci seznamu. Tím je umožněno to, že více stavů může sdílet části stejného spojového seznamu společného pro více stavů – jeden stav „vlastní“ pouze objekty mezi ukazateli *firstElementHolder* a *lastElementHolder*. Pohyb po seznamu je možný pomocí ukazatele *nextElementHolder*.

Každá instance třídy **ElementHolder** ukazuje na grafický objekt, kterým může být textové pole (**Text**) nebo čára (**Line**). V obou případech objekty drží kromě informací potřebných pro jejich vykreslení také koordináty svého vnějšího pravoúhlého obrysu pro optimalizaci procesu mazání a určení velikosti zobrazované plochy při rotaci obrázku.

Textové pole je definováno polohou, barvou, textem a velikostí fontu.

Čára je definována barvou, tloušťkou čáry a spojovým seznamem bodů.



■ **Obrázek 7.3** Datový model editoru

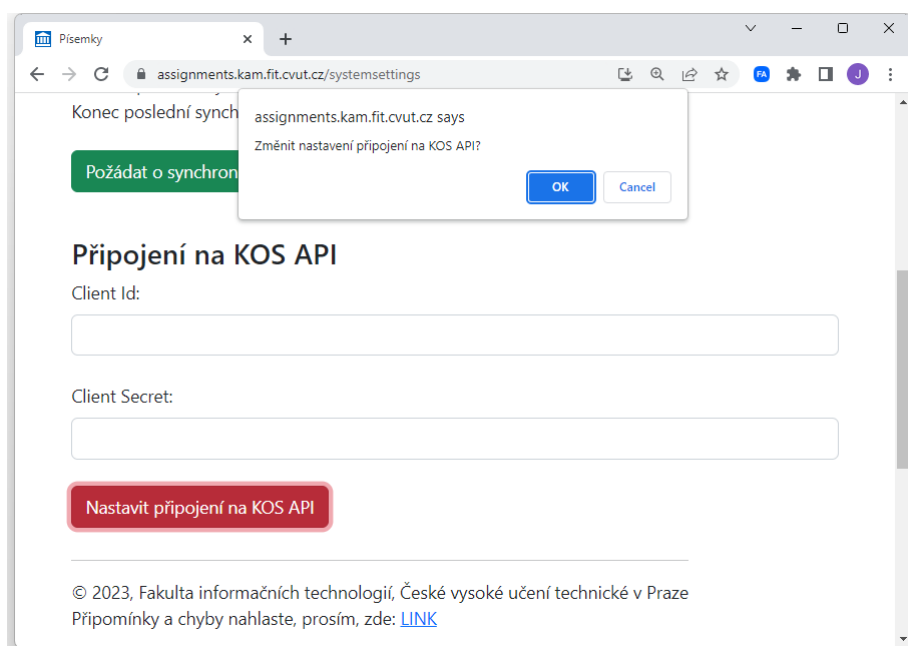
Původně jsem předpokládal, že kód editoru bude v aplikaci pro její správce dostupný a že si jej budou moci upravit podle potřeby. Technicky tomu nic nebrání, kód editoru není součástí kompilovaného aplikačního serveru ani součástí kompilovaného uživatelského rozhraní, je uložen jako resource a může být uložen například v konfigurační databázové tabulce nebo být součástí jazykové lokalizace. Nicméně vzhledem ke složitosti výsledného řešení a s vědomím toho, jak snadné je při změně kódu editoru udělat chybu, která nemusí být rychle odhalena nebo v případě odhalení snadno nalezena a opravena, jsem od tohoto záměru upustil.

Uživatelské rozhraní

Uživatelské rozhraní je provedeno jako webová aplikace pro běžné webové prohlížeče. Je napsána s využitím frameworku React s významným podílem nadstavby naprogramované v HTML a JavaScriptu.

8.1 Principy

Struktura aplikace a její chování vychází z principů Nielsen Norman Group pro uživatelská rozhraní [26].



■ **Obrázek 8.1** Potvrzovací dialog u kritických operací

Uživatel je vždy informován, na které stránce se nachází a jakými daty je stránka naplněna. Aplikace konzistentně využívá slovník, který odpovídá pojmům používaným pro předměty, se kterými operuje. Bodové hodnocení v Grades je odlišně vnímáno studenty (jsou to řádky)

a vyučujícími (jsou to sloupce) a aplikace používá termín milník. Naprostá většina v aplikaci použitých textů je parametrizovaná a uživatelé s rolí správce je mohou změnit.

Uživatel se po aplikaci může volně pohybovat; všude, kde je to vhodné, je použito tlačítko „Návrat“ pro návrat na předchozí stránku. Editor podporuje operace undo a redo. Kritické operace bez možnosti korekce (jako je například změna integračních parametrů) vyžadují potvrzení v dialogovém okně jako na obrázku 8.1.

Ovládací prvky jsou seřazeny v pořadí podle předpokládané četnosti jejich používání a pro snadnější orientaci jsou barevně odlišeny. Ovládací prvky a zobrazené informace jsou popsány a jejich používání vyžaduje použití paměti uživatele jen v omezeném rozsahu.

Až na drobné výjimky v editoru obrázků aplikace využívá pouze standardních ovládacích prvků, na jejichž akce je uživatel zvyklý z jiných obdobných aplikací.

Jednotlivé akce jsou potvrzovány s využitím komponent Toaster [27] a Toastify [28] podobně jako na obrázku 8.2 (po provedení operace se v pravém dolním rohu stránky na několik sekund zobrazí zelená potvrzovací hláška nebo červená chybová hláška), déle trvající akce jsou zpravidla indikovány použitím točivého kolečka indikujícího probíhající činnost na straně aplikace jako na obrázku 8.3.

105	cvičení pondělí lichý týden (16:15:00 - 17:45:00), T9:351	Ing. Michal Valenta Ph.D. (valenta)	Odevzdání	Skenovat QR kódy	Nastavit	Odevzdat
ex-1244874259705	Zkouška (TH:A-942) 05.01.2023 11:00:00 - 05.01.2023 12:00:00	Ing. Michal Valenta Ph.D. (valenta), Ing. Yelena Trofimova Ph.D. (trofiyel)	Odevzdání	Skenovat QR kódy	Nastavit	Odevzdat

Návrat

Seznam zkoušek a událostí:

Zkouška 07.12.2022 18:00:00 - 07.12.2022 19:30:00 (POUZE PISEMKA - varianta C T9:351 (namísto posledního cviceni))

Přidat nebo aktualizovat

✓ Zkoušky jsou nahrávány

✓ Zkoušky byly nahrány

© 2023, Fakulta informačních technologií, České vysoké učení technické v Praze
Připomínky a chyby nahláste, prosím, zde: [LINK](#)

■ **Obrázek 8.2** Zpráva o dokončení operace

Písemky

Předměty

Seznam semestrů

B222 - Letní 2022/2023

Seznam předmětů



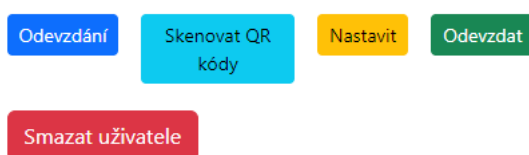
■ **Obrázek 8.3** Indikace čekání na dokončení operace

8.2 Ovládání aplikace

Uživatelské rozhraní je založeno na frameworku Bootstrap [29] a jeho implementaci ve formě komponenty pro framework React [30].

Ovládání aplikace je ve všech jejích částech implementováno shodným způsobem. Žádná změna dat uživatelem není odeslána na server dříve, než uživatel dá k takovému odeslání dat pokyn (například pomocí tlačítka „Uložit“).

Tlačítka jsou barevně odlišena tak, jak je vidět na obrázku 8.4, a jednotlivé barvy mají zpravidla význam popsaný v tabulce 8.1.



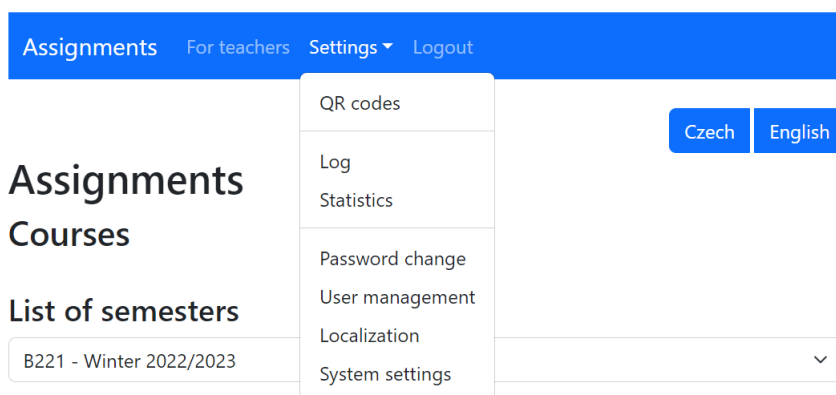
■ **Obrázek 8.4** Koláž použití tlačítek různých barev

■ **Tabulka 8.1** Rozlišení tlačítek podle barev

Barva	Význam
Modrá	Běžná operace, pohyb po uživatelském rozhraní
Zelená	Alternativní operace, často zahrnující zpracování dat
Žlutá	Operace měnící data, obvykle vykonávaná méně často
Červená	Operace s nevratnými nebo obtížně vratnými následky
Světle modrá	Speciální operace vykonávaná jen při zvláštních příležitostech

8.3 Aplikační menu

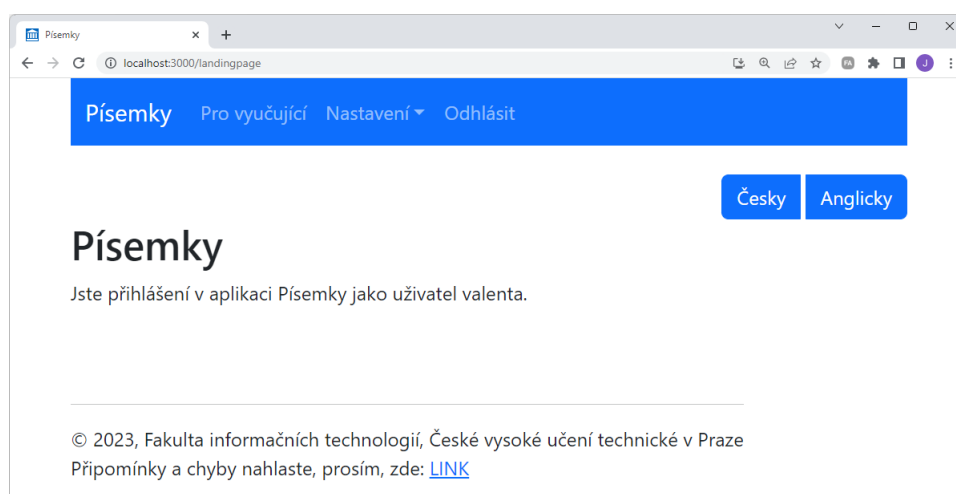
Na jednotlivých aplikačních stránkách je zobrazeno aplikační menu. Aplikační menu je generováno dynamicky podle rolí přihlášeného uživatele. Na obrázku 8.5 je aplikační menu vygenerované pro učitele přihlášeného jako lokální uživatel. Jednotlivé položky menu jsou použity podle tabulky 8.2. Pokud je uživatel přihlášen jako správce, je výchozí stránkou stránka na obrázku 8.6.



■ **Obrázek 8.5** Příklad aplikačního menu pro učitele přihlášeného jako lokální uživatel

■ **Tabulka 8.2** Struktura menu podle rolí uživatele

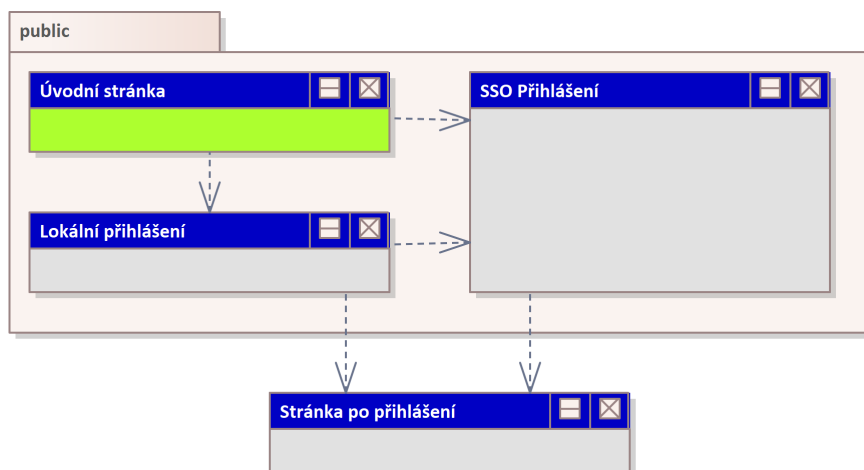
Položka menu	Uživatelské role
Seznam předmětů („Pro vyučující“, „Pro studenty“)	TEACHER, STUDENT (úvodní stránka)
Generování QR kódů	TEACHER
Lokalizace	ADMIN
Log	ADMIN
Statistika	ADMIN
Změna hesla	ADMIN
Správa uživatelů	ADMIN
Systémové nastavení	ADMIN
Odhlásit	USER



■ **Obrázek 8.6** Výchozí stránka pro správce

8.4 Veřejné stránky

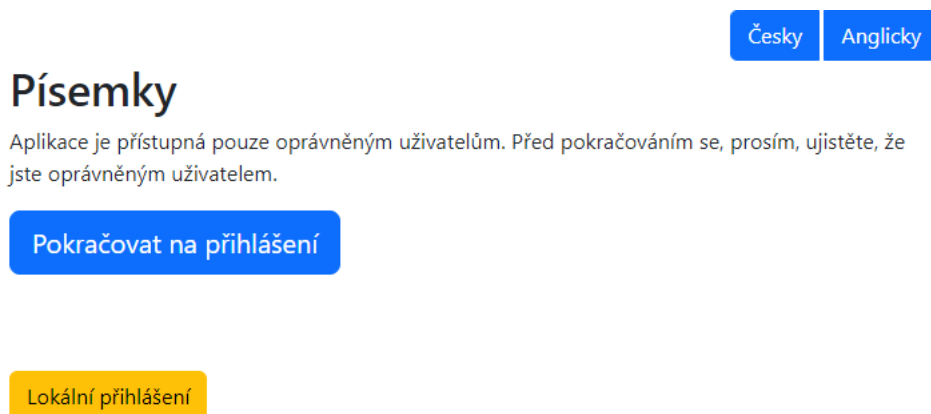
Veřejné stránky jsou spolu s přechody mezi stránkami zobrazeny na obrázku 8.7. Výchozí stránka je označena zeleně.



■ Obrázek 8.7 Veřejné stránky s přechody

8.4.1 Úvodní stránka

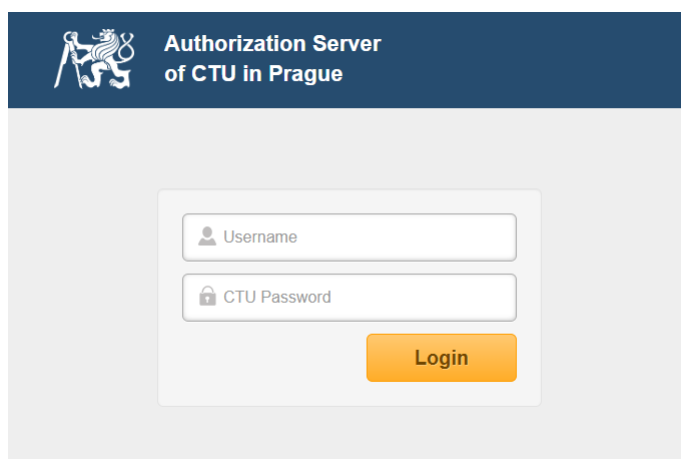
Po otevření aplikace na adrese <https://assignments.kam.fit.cvut.cz> je zobrazena úvodní stránka na obrázku 8.8 a uživatel má na výběr přihlášení pomocí školní SSO brány nebo lokální přihlášení pro správce aplikace.



■ Obrázek 8.8 Stránka po otevření aplikace

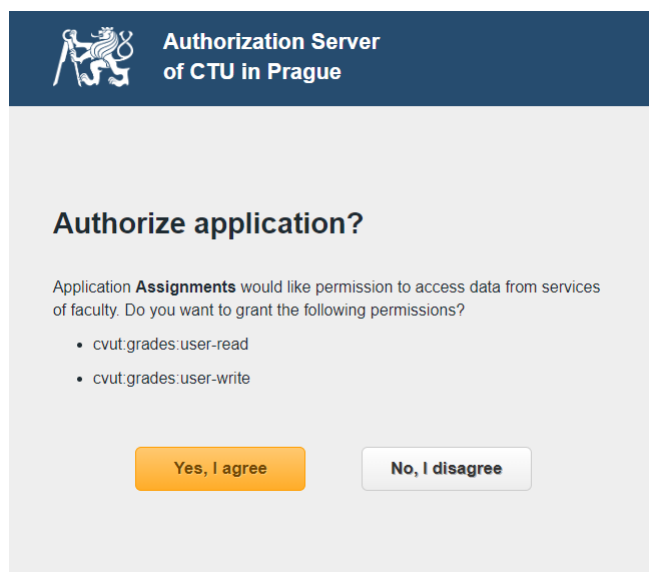
8.4.2 SSO přihlášení

Při přihlášení pomocí školní SSO brány uživatel vloží svoje přihlašovací jméno a heslo do formuláře na obrázku 8.9.



■ **Obrázek 8.9** Přihlášení pomocí SSO brány

Brána zobrazí požadovaný přístupový profil dialogem na obrázku 8.10 a po jeho potvrzení je uživatel do aplikace přihlášen.



■ **Obrázek 8.10** Potvrzení oprávnění při přihlášení

8.4.3 Lokální přihlášení

V případě přihlášení lokálního uživatele aplikace zobrazí přihlašovací formulář na obrázku 8.11. Na formuláři je navíc tlačítko pro přechod na přihlášení pomocí SSO, protože při testování se ukázalo, že některé prohlížeče si přihlašovací stránku ukládají, považující za výchozí stránku webu a zobrazují ji místo úvodní stránky.

Česky

Anglicky

Písemky

Lokální přihlášení

Uživatelské jméno:

Heslo:

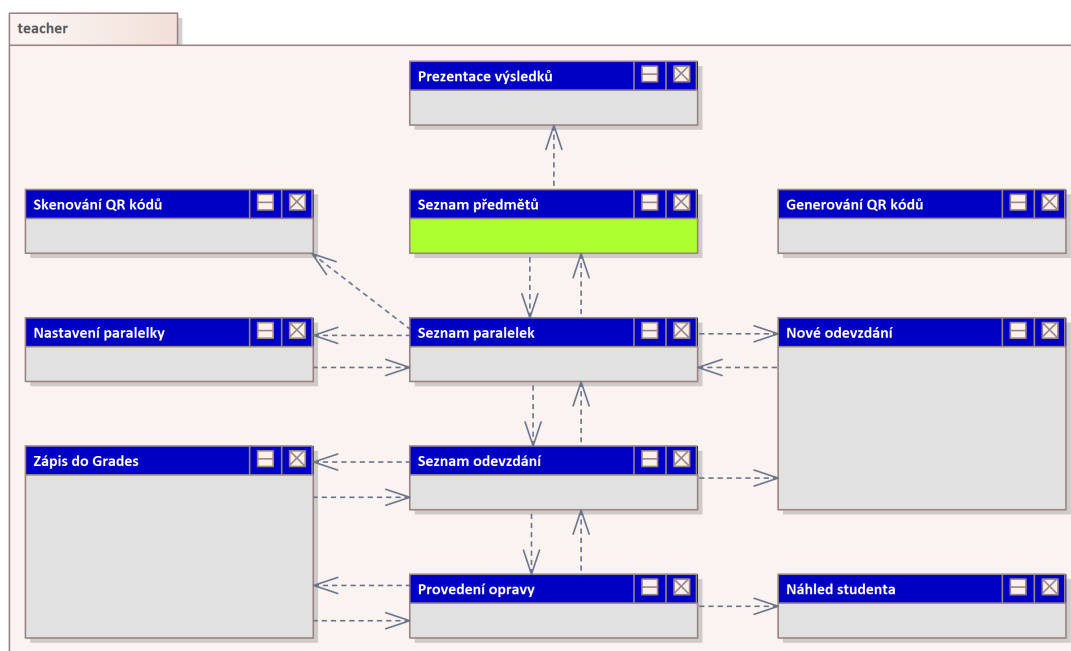
Přihlásit

Přihlášení ČVUT

■ Obrázek 8.11 Přihlášení lokálního uživatele

8.5 Stránky pro učitele

Stránky pro učitele jsou spolu s přechody mezi stránkami zobrazeny na obrázku 8.12. Výchozí stránka je označena zeleně.



■ Obrázek 8.12 Stránky pro učitele s přechody

8.5.1 Seznam předmětů

Úvodní stránkou části pro učitele je stránka na obrázku 8.38. Vyučující vybere semestr a aplikace zobrazí seznam jím vyučovaných předmětů. U každého předmětu může pokračovat na jeho detail nebo si nechat zobrazit výsledky studentů.

Písemky

Předměty

Seznam semestrů

B221 - Zimní 2022/2023

Seznam předmětů

Kód	Název	Operace
BI-BPR	Bakalářský projekt	Předmět Výsledky
BI-BPR.21	Bakalářský projekt	Předmět Výsledky
BI-IDO.21	Úvod do DevOps	Předmět Výsledky
BIE-BPR	Bachelor Project	Předmět Výsledky
BIE-SI2.3	Software Engineering 2	Předmět Výsledky
BIK-BPR	Bakalářský projekt	Předmět Výsledky
BIK-KOM	Konceptuální modelování	Předmět Výsledky
NI-MPR	Magisterský projekt	Předmět Výsledky
NI-PDB	Pokročilé databázové systémy	Předmět Výsledky
NIE-PDB	Advanced Database Systems	Předmět Výsledky

■ **Obrázek 8.13** Seznam vyučovaných předmětů

8.5.2 Seznam paralelek

Pokud uživatel ze seznamu předmětů pokračuje na detail vyučovaného předmětu, aplikace zobrazí seznam paralelek vyučovaného předmětu na obrázku 8.14. Pokud u předmětu existuje přednáška a cvičení se stejným kódem, je tato přednáška a cvičení zobrazeno v oddělených řádkách, ale pod stejným kódem v prvním sloupci tabulky. Aplikace zobrazuje i paralelky předmětu, kteří vyučují pouze jiní vyučující a umožňuje pro ně zobrazit seznam existujících odevzdání nebo skenování QR kódů, neumožňuje ale jejich nastavení nebo odevzdání studentské práce.

V seznamu paralelek na obrázku vidíme přidanou zkoušku dne 5. 1. 2023 v 11:00, se kterou aplikace pracuje jako s virtuální paralelkou – má kód, má přiřazené učitele a studenty, je možné ji nastavit, odevzdat pro ni studentské práce a tyto práce opravit.

Pomocí tlačítek pod tabulkou se vyučující může vrátit na předchozí stránku nebo nahrát z KOSu zkoušky a události. Po kliknutí na tlačítko pro nahrání zkoušek aplikace na stránce nově zobrazí seznam zkoušek a událostí tak, jak je vidět na obrázku 8.15. Pokud zkouška nebo událost není v seznamu paralelek zapsána, aplikace ji tam přidá. Pokud tam již byla přidána v minulosti, aplikace aktualizuje její údaje včetně seznamu přiřazených studentů.

Písemky

Předmět

Seznam paralelek B221 / NI-PDB

Kód	Popis	Vyučující	Operace			
1	přednáška pondělí každý týden (12:45:00 - 14:15:00), T9:107	Ing. Michal Valenta Ph.D. (valenta)	Odevzdání	Skenovat QR kódy	Nastavit	Odevzdat
101	cvičení středa sudý týden (18:00:00 - 19:30:00), T9:351	Ing. Yelena Trofimova Ph.D. (trofiyel)	Odevzdání	Skenovat QR kódy		
102	cvičení čtvrtek sudý týden (16:15:00 - 17:45:00), T9:350	Ing. Michal Valenta Ph.D. (valenta)	Odevzdání	Skenovat QR kódy	Nastavit	Odevzdat
103	cvičení středa lichý týden (18:00:00 - 19:30:00), T9:351	Ing. Yelena Trofimova Ph.D. (trofiyel)	Odevzdání	Skenovat QR kódy		
104	cvičení čtvrtek lichý týden (16:15:00 - 17:45:00), T9:350	Ing. Michal Valenta Ph.D. (valenta)	Odevzdání	Skenovat QR kódy	Nastavit	Odevzdat
105	cvičení pondělí lichý týden (16:15:00 - 17:45:00), T9:351	Ing. Michal Valenta Ph.D. (valenta)	Odevzdání	Skenovat QR kódy	Nastavit	Odevzdat
ex-1244874259705	Zkouška (TH:A-942) 05.01.2023 11:00:00 - 05.01.2023 12:00:00	Ing. Michal Valenta Ph.D. (valenta), Ing. Yelena Trofimova Ph.D. (trofiyel)	Odevzdání	Skenovat QR kódy	Nastavit	Odevzdat

[Návrat](#)
[Nahrát zkoušky](#)

■ **Obrázek 8.14** Zobrazení seznamu paralelek předmětu

105	cvičení pondělí lichý týden (16:15:00 - 17:45:00), T9:351	Ing. Michal Valenta Ph.D. (valenta)	Odevzdání	Skenovat QR kódy	Nastavit	Odevzdat
ex-1244874259705	Zkouška (TH:A-942) 05.01.2023 11:00:00 - 05.01.2023 12:00:00	Ing. Michal Valenta Ph.D. (valenta), Ing. Yelena Trofimova Ph.D. (trofiyel)	Odevzdání	Skenovat QR kódy	Nastavit	Odevzdat

Návrat

Seznam zkoušek a událostí:

Zkouška 07.12.2022 18:00:00 - 07.12.2022 19:30:00 (POUZE PISEMKA - varianta C T9:351 (namisto posledniho cviceni))

Přidat nebo aktualizovat

■ **Obrázek 8.15** Seznam zkoušek a událostí

8.5.3 Nastavení paralelky

Nastavení paralelky provádí vyučující pomocí formuláře na stránce 8.16.

V horní části formuláře lze vybrat jinou paralelku a pomocí tlačítka „Přepsat nastavení“ lze přepsat aktuálně zobrazené nastavení paralelky podle vybrané paralelky. Tato funkčnost je určena pro případy, kdy vyučující nastaví jednu vyučovanou paralelku a toto nastavení chce replikovat na další svoje paralelky, v případě replikace nastavení ostatních paralelek podle jedné vzorově nastavené nebo v případě zkoušek, kdy je potřeba nastavit pouze první zkoušku daného předmětu v semestru a ostatní výskyty zkoušky lze nastavit podle té první.

Jednotlivé milníky odpovídají definovaným událostem v Grades, které mají způsob hodnocení MANUAL a typ hodnocení NUMBER, BOOLEAN nebo STRING. Události z Grades aplikace načte při zobrazení stránky nastavení paralelky s využitím SSO tokenu přihlášeného uživatele.

Termín odevzdání odpovídá nejzazšímu okamžiku, kdy student může práci odevzdat. Lze jej nastavit na libovolnou hodinu a celočíselný násobek 5 minut. Pokud student v termínu již práci odevzдал, může mu učitel nastavit nový termín odevzdání.

Vyučující může nastavit, zda práci může odevzdat také student a zda ji může odevzdat opakovaně.

Vyučující může nastavit, zda aplikace studentovi zobrazuje všechna jeho odevzdání k milníku, nebo jen to poslední.

Dále učitel v nastavení určí, kolik hodnocených částí je při opravování odevzdání k danému milníku rozlišováno. Jedna část je typicky jeden příklad nebo jedna úloha. U každé části vyučující vyplní její název česky a anglicky, minimální a maximální počet udělovaných bodů a milník, do jehož výsledku se hodnocená část započítává.

Pokud odevzdání může mít více částí, aplikace učiteli umožní stanovit, zda student musí jednotlivé části odevzdávat odděleně, najednou, nebo zda se sám může rozhodnout, v jaké struktuře svoji práci odevzdá.

Písemky

Nastavení paralelky

Seznam paralelek:

104 - cvičení čtvrtěk lichý týden (16:15:00 - 17:45:00), T9:350

Kliknutím na tlačítko níže přepíšete nastavení aktuální paralelky podle vybrané paralelky.

Přepsat nastavení

Seznam milníků B221 / NI-PDB / 105 - cvičení pondělí lichý týden (16:15:00 - 17:45:00), T9:351

Nastavení milníku aktivita1 - Aktivní účast 1

Základní nastavení

Zobrazen

Nastavení milníku aktivita2 - Aktivní účast 2

Základní nastavení

Zobrazen

Nastavení milníku aktivita3 - Aktivní účast 3

Základní nastavení

Zobrazen

Termín

21.04.2023

08

:

15

Odevzdání

- Odevzdání studentem
 Opakované odevzdání

Student vidí

- Poslední odevzdání
 Všechna odevzdání

Části

2 (kladné číslo)

Způsob odevzdání

- Po částech
 Všechno najednou
 Podle studenta

Část	Název CZ	Název EN	Minimum	Maximum	Započítá se do
1	Úloha 1	Task 1	0	3	aktivita3 - Aktivní účast 3
2	Úloha 2	Task 2	0	3	aktivita3 - Aktivní účast 3

[Uložit](#)

[Návrat](#)

Obrázek 8.16 Nastavení paralelky

8.5.4 Nové odevzdání

Pokud práci odevzdává vyučující, má k tomu v aplikaci k dispozici stránku na obrázku 8.17. Na stránce vybere milník, pro nějž práci odevzdává, a studenta, pro něhož práci odevzdává. Pokud výběr studenta nevyužije, proběhne přiřazení jednotlivých stránek odevzdaného souboru ke studentům až během opravování.

Písemky

Odevzdání souborů

Vytvořit odevzdání B221 / NI-PDB / 105 - cvičení pondělí lichý týden (16:15:00 - 17:45:00), T9:351

Milníky:

Seznam studentů:

Soubory k odevzdání pro všechny části

Název souboru	Velikost	Operace
---------------	----------	---------

Vybrat soubory

Soubory k odevzdání pro část Úloha 1

Název souboru	Velikost	Operace
---------------	----------	---------

Vybrat soubory

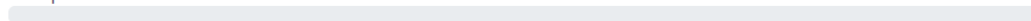
Soubory k odevzdání pro část Úloha 2

Název souboru	Velikost	Operace
---------------	----------	---------

Vybrat soubory

Komentář:

Postup odesílání 0 %



Odevzdat soubory

Návrat

■ **Obrázek 8.17** Formulář pro odevzdání práce učitelem

Učitel se může rozhodnout, zda odevzdá soubory pro všechny části milníku bez rozlišení části milníku nebo pro jednotlivé části milníku odděleně.

Aplikace akceptuje soubory s příponami pdf, jpg, jpeg a png.

Jakmile vyučující do formuláře vloží první soubor, tak aplikace skryje druhou, dosud nevyužitou možnost. V případě zobrazeném na obrázku 8.18 vložil učitel první soubor k možnosti odevzdání souboru k části 1. Do formuláře může dále vkládat další soubory s rozlišením části,

ke které soubory vkládá, ale již nemůže vložit soubor bez určení části, pro kterou jej odevzdává.

Soubor může z formuláře odstranit nebo stáhnout a zkontrolovat. Jakmile soubory odevzdá, aplikace je odešle na server a uloží do fronty ke zpracování.

Písemky

Odevzdání souborů

Vytvořit odevzdání B221 / NI-PDB / 105 - cvičení pondělí lichý týden (16:15:00 - 17:45:00), T9:351

Milníky:

Seznam studentů:

Soubory k odevzdání pro část Úloha 1

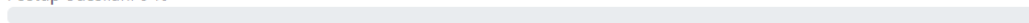
Název souboru	Velikost	Operace
scan.pdf	2201866	<input type="button" value="Stáhnout"/> <input type="button" value="Odstranit"/>

Soubory k odevzdání pro část Úloha 2

Název souboru	Velikost	Operace
31b022d4-4c6a-48c4-8cb9-4f1fad70e1f2.pdf	237121	<input type="button" value="Stáhnout"/> <input type="button" value="Odstranit"/>
55ff196e-a3e2-45b2-a57e-ed83c836e96a.pdf	19766	<input type="button" value="Stáhnout"/> <input type="button" value="Odstranit"/>

Komentář:

Postup odeslání 0 %



Obrázek 8.18 Formulář pro odevzdání práce učitelem s vloženými soubory

8.5.5 Seznam odevzdání

Učitel může ze seznamu paralelek otevřít stránku se seznamem odevzdání pro danou paralelku na obrázku 8.42.

Pokud aktuálně probíhá automatické zpracování odevzdaných prací na obrázky k opravě, aplikace na stránce v její horní části uživatele informuje o aktuálním průběhu tohoto zpracování.

Odevzdání zobrazená v tabulce může učitel filtrovat podle milníku, stavu zpracování, osoby, která práci odevzdala nebo studenta, pro něhož byla práce odevzdaná.

V případě, kdy je odevzdávajícím student, může učitel nastavit nový termín odevzdání, který platí pouze pro dotyčného studenta.

Učitel může stáhnout odevzdané soubory pomocí zobrazených odkazů, provést jejich opravu, výsledky zapsat do Grades nebo odevzdání označit jako uvolněné pro studenty.

Aplikace zobrazuje studentům pouze opravené obrázky z odevzdání, která jsou označena jako uvolněná pro studenty. Odevzdání je takto označené, pokud všechny jeho obrázky jsou ve stavu smazaný nebo uvolněný pro studenty. Pokud vyučující při opravování jednotlivé obrázky jako uvolněné pro studenty neoznačil, může to udělat na této stránce pro všechny nesmazané obrázky daného odevzdání.

Vyučující může zvolená odevzdání označit jako smazaná. Pokud je zobrazí pomocí filtru nastaveného na smazaná odevzdání, může je opět obnovit.

Písemky

Odevzdání

Seznam odevzdání B221 / NI-PDB / 105 - cvičení pondělí lichý týden (16:15:00 - 17:45:00), T9:351

[Návrat](#)

Probíhá zpracování odevzdání / obrázku: 222 / 18502, Sun Apr 09 00:11:28 CEST 2023

Milníky: Stav: Odevzdáno kým: Odevzdáno pro:

ID	Datum	Milník	Stav	Nový termín	Odevzdáno kým / Odevzdáno pro	Soubory	Operace
219	07.04.2023	Aktivní účast 3	Opravené	14.03.2023 05 45	Jan Petržílka (petrzja5) / Jan Petržílka (petrzja5)	• scan.pdf	Opravy Zapsat do Grades Uvolnit pro studenty Smazat
220	08.04.2023	Aktivní účast 3	Uvolněné pro studenty		Jan Petržílka (petrzja5) / Jan Petržílka (petrzja5)	• image1.png	Opravy Zapsat do Grades Smazat
221	09.04.2023	Aktivní účast 3	Uložené		Ing. Michal Valenta Ph.D. (valenta) / Jan Petržílka (petrzja5)	• scan.pdf	Opravy Zapsat do Grades Uvolnit pro studenty Smazat
222	09.04.2023	Aktivní účast 3	Odevzdané		Ing. Michal Valenta Ph.D. (valenta) / ?	• scan.pdf	Opravy Zapsat do Grades Smazat
224	09.04.2023	Aktivní účast 3	Odevzdané		Jan Petržílka (petrzja5) / Jan Petržílka (petrzja5)	• image1.png	Opravy Zapsat do Grades Smazat
224	Komentář: Toto je komentář od studenta k odevzdání						
225	09.04.2023	Aktivní účast 3	Odevzdané		Ing. Michal Valenta Ph.D. (valenta) / ?	• image1.png	Opravy Zapsat do Grades Smazat
225	Komentář: Toto je komentář k odevzdání od učitele						

[Návrat](#)

[Nové odevzdání](#)

Obrázek 8.19 Zobrazení seznamu odevzdaných prací učitele

8.5.6 Provedení opravy

Vlastní provedení opravy vyučující provádí pomocí stránky na obrázku 8.20. Stránka odráží požadavek klíčových uživatelů na přiřazování odevzdaných prací a jejich částí ke studentům nebo příkladům až během opravování, má více funkcí a je opravdu složitá. Z této stránky učitel při opravování otevírá editor, který zobrazuje jednotlivé listy odevzdání a umožňuje je hodnotit a připisovat k nim poznámky.

Stránka má unikátní URL adresu, kterou mohou použít všichni přihlášení uživatelé aplikace bez ohledu na to, zda daný předmět vyučují nebo ne. Jejím sdělením učiteli, který jinak předmět nevyučuje, lze tedy umožnit, aby odevzdané práce opravoval. Stránka pro zápis hodnocení do Grades použitý kód sdílí.

Písemky

Opravování odevzdání

Odevzdání / B221 / NI-PDB / 105 / 223

Stránky

Soubory:

Studenti:

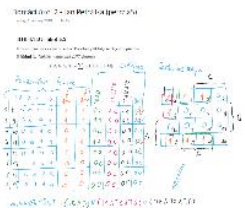
Části:

Stav:

Správa zámek:

Aktualizovat zámky

• submissionCorrection.file Image1.png

Stránka	Náhled	Student	Část	Stav	Opravy	Operace
1 (18511)		petrzja5 → petrzja5	0 → 0	Neopraveno		Opravit

Výsledky

Student	Původní hodnocení	Vytvořené hodnocení	Operace
Jan Petržílka (petrzja5)	Aktivní účast 3: 0 1: Úloha 1: 0 (→ Aktivní účast 3) 2: Úloha 2: 0 (→ Aktivní účast 3)		

[Návrat](#)

[Zapsat do Grades](#)

[◀ Předchozí](#)

[Další >](#)

■ **Obrázek 8.20** Stránka provedení opravy před provedením změn

Jednotlivé zobrazené obrázky odpovídající listům z odevzdaných souborů je možné filtrovat podle souboru, studenta, části milníku nebo stavu opravovaného obrázku. Filtr podle stavu je implementován tak, že při prázdném výběru zobrazuje všechny obrázky kromě obrázků ve stavu smazaný. Ty zobrazuje pouze při filtru nastaveném na tuto hodnotu.

V základním režimu učitel opravuje domácí úkoly nebo písemky svojí paralelky a nepotřebuje využít funkčnosti zámků, které jsou potřeba pro paralelní opravování stejného odevzdání více učiteli.

Písemky

Opravování odevzdání

Odevzdání / B221 / NI-PDB / 105 / 223

Stránky

Soubory:

Studenti:

Části:

Stav:

Správa zámků:

Aktualizovat zámky

• submissionCorrection.file Image1.png

Stránka	Náhled	Student	Část	Stav	Opravy	Operace
1 (18511)		petrja5 → petrja5	0 → 0	Opraveno		Otevřít

Výsledky

Student	Původní hodnocení	Vytvořené hodnocení	Operace
Jan Petržilka (petrja5)	Aktivní účast 3: 0 1: Úloha 1: 0 (→ Aktivní účast 3) 2: Úloha 2: 0 (→ Aktivní účast 3)	Aktivní účast 3: 4 1: Úloha 1: <input type="text" value="1"/> (0 ≤ x ≤ 3) 2: Úloha 2: <input type="text" value="3"/> (0 ≤ x ≤ 3)	Odstranit změny

Uložit změny

■ Obrázek 8.21 Stránka provedení opravy před uložením změn

Pro každý obrázek aplikace zobrazí jeho pořadové číslo v odevzdání a v systému, náhledový obrázek se sníženým rozlišením (lze jej rozkliknout a zobrazit tak původní obrázek ve velkém rozlišení), identifikaci studenta, jemuž je obrázek v databázi přiřazen, a studenta, jemuž je obrázek aktuálně přiřazen na této stránce jako výsledek procesu opravování, část milníku, jemuž je obrázek v databázi přiřazen, a část milníku, jemuž je obrázek aktuálně přiřazen na této stránce

jako výsledek procesu opravování, aktuální stav obrázku a historické vrstvy provedených oprav.

Obrázek, který má přiřazenou část nula („0“), může obsahovat více různých částí. Pomocí tlačítka „Opravit“ lze otevřít editor a provést opravu obrázku.

Po provedení opravy je v tabulce v dolní části stránky zobrazeno přiřazené bodové hodnocení a data je možné uložit do databáze nebo smazat pomocí tlačítek zobrazených na obrázku 8.21. Uživatel může před uložením dat vytvořené bodové hodnocení ještě upravit. Dokud data nejsou uložena, nejsou změny promítnuty do záznamů v databázi.

Písemky

Opravování odevzdání

Odevzdání / B221 / NI-PDB / 105 / 223

Stránky

Soubory:

Studenti:

Části:

0: Více částí

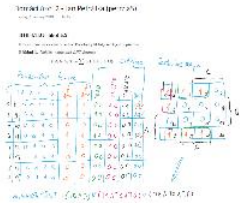
Stav:

Správa zámků:

Zamknout a otevřít

Aktualizovat zámky

• submissionCorrection.file Image1.png

Stránka	Náhled	Student	Část	Stav	Opravy	Operace
1 (18511)		petrzja5 → petrzja5	0 → 0	Opraveno		Otevřít

Výsledky

Student	Původní hodnocení	Vytvořené hodnocení	Operace
Jan Petrzilka (petrzja5)	Aktivní účast 3: 0	Aktivní účast 3: 4	Odstranit změny
	1: Úloha 1: 0 (→ Aktivní účast 3) 2: Úloha 2: 0 (→ Aktivní účast 3)	1: Úloha 1: <input type="text" value="1"/> (0 ≤ x ≤ 3) 2: Úloha 2: <input type="text" value="3"/> (0 ≤ x ≤ 3)	

Uložit změny

■ **Obrázek 8.22** Stránka provedení opravy před použitím zámků

Pokud jedno odevzdání opravuje paralelně více vyučujících, je vhodné využít zámky, které zajistí, že jednu stránku nebude opravovat paralelně více vyučujících a že si navzájem nebudou

při ukládání dat do databáze svoje hodnocení přepisovat.

Před využitím funkčnosti zámků je potřeba vybrat část, která bude daným uživatelem opravována. Zámek je pak uložen v databázi jako vazba mezi odevzdáním, opravujícím, studentem a částí. Při volbě části nula jsou zámky aplikovány na všechny části milníku. Jednotlivé obrázky je potřeba otevřít místo tlačítkem „Otevřít“ tlačítkem „Zamknout a otevřít“ (a aplikace vybere a zamkne dosud nezamčenou část odevzdání pro jednoho studenta) nebo tlačítkem „Otevřít se zámek“ na obrázku 8.22.

Písemky

Opravování odevzdání

Odevzdání / B221 / NI-PDB / 105 / 223

Stránky

Soubory:

Studenti:

Části:

0: Více částí

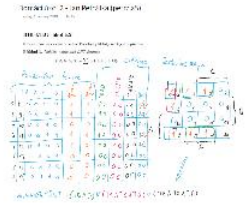
Stav:

Správa zámků:

Zamknout a otevřít

Aktualizovat zámky

• submissionCorrection.file Image1.png

Stránka	Náhled	Student	Část	Stav	Opravy	Operace
1 (18511)		petrzja5 → petrzja5	0 → 0	Opraveno Uzamčeno pro: Ing. Michal Valenta Ph.D. Odemknout		Otevřít Otevřít se zámek

Výsledky

Student	Původní hodnocení	Vytvořené hodnocení	Operace
Jan Petrzilka (petrzja5)	Aktivní účast 3: 0 1: Úloha 1: 0 (→ Aktivní účast 3) 2: Úloha 2: 0 (→ Aktivní účast 3)	Aktivní účast 3: 4 1: Úloha 1: <input type="text" value="1"/> (0 ≤ x ≤ 3) 2: Úloha 2: <input type="text" value="3"/> (0 ≤ x ≤ 3)	Odstranit změny

Uložit změny

■ **Obrázek 8.23** Stránka provedení opravy po použití zámků

Jakmile jsou obrázky zamčeny, jsou tyto zámky na stránce zobrazeny tak, jak je vidět na obrázku 8.23. Jejich stav lze aktualizovat pomocí tlačítka „Aktualizovat zámky“. Uživatel tak

může zjistit, který z kolegů má daný obrázek uzamčen. Pokud uživatel zámek smaže pomocí tlačítka „Odemknout“, může jej zamknout jiný uživatel. Původní uživatel o změně není informován a předpokládá, že obrázek je stále zamčený pro něj. Uložení dat pak skončí chybou.

Písemky

Opravování odevzdání

Odevzdání / B221 / NI-PDB / 105 / 223

Stránky

Soubory:

Studenti:

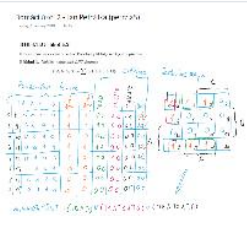
Části:

Stav:

Správa zámků:

Aktualizovat zámky

• submissionCorrection.file Image1.png

Stránka	Náhled	Student	Část	Stav	Opravy	Operace
1 (18511)	 <p>Náhled studenta na uloženou verzi</p>	petrja5 → petrja5	0 → 0	Uvolněno pro studenta	<ul style="list-style-type: none"> 10.04.2023, Ing. Michal Valenta Ph.D. (valenta) 	Opravit

Výsledky

Student	Původní hodnocení	Vytvořené hodnocení	Operace
Jan Petržílka (petrja5)	Aktivní účast 3: 4 1: Úloha 1: 1 (→ Aktivní účast 3) 2: Úloha 2: 3 (→ Aktivní účast 3)		

Návrat

Zapsat do Grades

< Předchozí

Další >

■ Obrázek 8.24 Stránka provedení opravy po uložení změn

Po uložení provedených oprav do databáze je na stránce na obrázku 8.24 zobrazeno nově uložené hodnocení již jako „Původní hodnocení“ a u opraveného obrázku je zobrazena formou linku provedená vrstva opravy. Každá další oprava přidá další vrstvu. Tyto vrstvy si uživatel může jednotlivě zobrazit a pokračovat v opravě od zobrazené vrstvy. Studentovi je zobrazena vždy jen poslední uložená vrstva.

Pokud je obrázek uvolněn pro studenta, může si učitel zobrazit obrázek tak, jak je uvidí student.

Aplikace si uloží pořadí odevzdání podle nastaveného filtru na stránce se seznamem odevzdání, přes kterou uživatel tuto stránku otevřel. Pomocí tlačítek „< Předchozí“ a „Další >“ pak uživatel může otevřít stránku pro opravování odevzdání pro předchozí nebo následující odevzdání podle uloženého pořadí odevzdání a nemusí se vracet na původní stránku se seznamem odevzdání.

Dalším krokem je zápis vytvořeného hodnocení do Grades na obsahově a funkčně navazující stránce.

8.5.7 Zápis do Grades

Stránka pro uložení vytvořeného hodnocení do Grades na obrázku 8.25 má URL adresu sestavenou stejně jako stránka pro provedení oprav – uživatel, který zná URL jedné z těchto stránek, může použít také druhou stránku bez ohledu na to, zda daný předmět vyučuje.

Písemky

Zápis do Grades

Hodnocení / B221 / NI-PDB / 105 - cvičení pondělí lichý týden (16:15:00 - 17:45:00), T9:351 / 223

Seznam hodnocení

Student	Původní hodnocení	Vytvořené hodnocení	Cílové hodnocení
Jan Petržílka (petrzja5)	Aktivní účast 3: 17.2	Aktivní účast 3: 4 1: Úloha 1: 1 (→ Aktivní účast 3) (@ valenta) 2: Úloha 2: 3 (→ Aktivní účast 3) (@ valenta)	Aktivní účast 3: <input type="text" value="17.2"/> (0 <= x <= 25) <input checked="" type="checkbox"/> Toto bude uloženo

Zapsat do Grades

Přepsat novými hodnotami

Přepsat součtem hodnot

Návrat

Otevřít opravu

< Předchozí

Další >

■ **Obrázek 8.25** Zápis hodnocení do aplikace Grades

Na stránce aplikace zobrazí nově vytvořené hodnocení pro dané odevzdání a s využitím SSO tokenů přihlášeného uživatele data pro stejné studenty získaná z Grades. Nové hodnoty je pak možné do Grades uložit, opět s využitím SSO tokenů přihlášeného uživatele.

Stránka umožňuje nově vytvořené hodnoty použít místo původních hodnot nebo je s původními hodnotami sečíst. Až do tohoto okamžiku aplikace používá pro hodnocení výhradně číselné hodnoty, zde dochází k jejich eventuální konverzi na řetězec nebo logickou hodnotu tak, aby ukládaná hodnota odpovídala očekávanému formátu aplikace Grades. V případě číselné hodnoty je ukládána číselná hodnota, v případě řetězce znaková reprezentace číselné hodnoty, v případě logické hodnoty je nula považována na false a ostatní hodnoty za true.

Uživatel také může určit, které hodnoty budou do aplikace Grades zapsány, případně tyto hodnoty změnit.

Aplikace si uloží pořadí odevzdání podle nastaveného filtru na stránce se seznamem odevzdání, přes kterou uživatel tuto stránku otevřel. Pomocí tlačítek „< Předchozí“ a „Další >“ pak uživatel může otevřít stránku pro zápis hodnot do Grades pro předchozí nebo následující odevzdání podle uloženého pořadí odevzdání a nemusí se vracet na původní stránku se seznamem odevzdání.

8.5.8 Editor

Editor lze otevřít ze stránky pro provedení opravy pro kterýkoliv uložený obrázek odpovídající listům odevzdaných souborů. Editor se otevře podle poměru stran okna prohlížeče v zobrazení na šířku na obrázku 8.26 nebo v zobrazení na výšku na obrázku 8.27. Obě tato zobrazení jsou funkčně ekvivalentní. Poměr jednotlivých částí editoru může uživatel ovlivnit pomocí funkce zvětšení a zmenšení zobrazovaného obsahu prohlížeče.

Domácí úkol 2 - Jan Petržílka (petrzja5)
Tuesday, 1 January 2019 14:46

Toto je rukou
psaný text

Toto je vložený text

BIK-MLO: úkol č.2

K odevzdání na cvičení 5. ledna. Podobné příklady se objeví v písemce.

Příklad 1. Najděte minimální DNT formule

$$f(A, B, C, D) = \sum(0, 1, 3, 4, 8, 9, 11).$$

Ověříme

Známe v tabulce

	A	B	C	D	f	f	7A7D	7ABC7D	7A7C7D	DNT
0	8	0	0	0	0	1	1	0	0	1
1	9	0	0	0	1	1	1	0	0	1
2	10	0	0	1	0	0	0	0	0	0
3	11	0	0	1	1	1	1	0	0	1
4	12	0	1	0	0	0	0	1	0	0
5	13	0	1	0	1	0	0	0	0	0
6	14	0	1	1	0	0	0	0	0	0
7	15	0	1	1	1	0	0	0	0	0

Sestavíme mapu

	0	1	2	3	4	5	6	7
0	0	1	1	1	3	2		
1	1	0	5	7	6			
2	0	4	0	13	15	14		
3	1	8	1	9	11	10		

Učitelova ovládací lišta (vlevo):

- Barvné terče: modrý, červený, zelený, černý, bílý, červená čára.
- Ikony: Undo, Redo, Zvětšit, Zmenšit, Nastavit šířku, Přepnout čáry, Rotace, Uložit.
- Políčko: Kreslení prstem (1).
- Nový komentář: Vložit.
- Velikost fontu: 12.
- Políčko: Uvolněno (zaškrtnuto), Smazáno.
- Stránky: 1 / 1 (18490).
- Části: 0: Více částí.
- Student: Jan Petržílka (petrzja5).
- Část: 1: Úloha 1 (Min: 0, Max: 3, Bodů: 1); 2: Úloha 2 (Min: 0, Max: 3, Bodů: 3).

■ **Obrázek 8.26** Editor při zobrazení na šířku

Ovládací prvky v horní polovině části s ovládacími prvky editoru na obrázku 8.26 mají následující význam:

- Barevné terče slouží k výběru barvy kreslení. Prázdný terč k mazání obsahu a symbol pera indikuje zvolenou barvu. Při mazání obsahu uživatel vybere obdélníkovou oblast a editor její obsah odstraní.
- Ve druhé řádce jsou ikony odpovídající návratu na předchozí stav („undo“), přechod na budoucí stav po návratu na předchozí stav („redo“), ikona pro zvětšení obrázku, zmenšení obrázku a nastavení zvětšení obrázku na šířku editačního pole. Ikona s klikatou čarou slouží k přepínání kreslení rovných a klikatých čar. Další ikony na pravém okraji pak slouží k rotaci obrázku a uložení obrázku spolu s vrstvou opravy do souboru ve formátu PNG ke stažení.
- Po zaškrtnutí pole „Kreslení prstem“ lze na dotykovém displayi kreslit prstem i bez použití stylusu. Číslo vedle této možnosti určuje tloušťku kreslené čáry.
- Pomocí tlačítka „Vložit“ je možné do obrázku vložit text s nastavenou velikostí fontu.
- Ke každému obrázku lze vložit komentář, který je studentovi zobrazen.
- Pomocí dvojice zaškrťovacích polí lze obrázek uvolnit pro studenty nebo jej smazat.

Další ovládací prvky slouží k pohybu mezi obrázky:

- Pomocí tlačítka „Zavřít“ se uživatel může vrátit na stránku s přehledem obrázků.
- Pomocí tlačítek \leftarrow a \rightarrow stejně jako pomocí výběru ze seznamu „Stránky“ se uživatel může pohybovat mezi obrázky. Pokud využívá zámek, je rozsah výběru omezen rozsahem aktuálně využívaného zámku.
- Pokud uživatel využívá zámek, může pomocí tlačítka $\}$ požádat o přidělení další skupiny obrázků k opravě – tedy stejné části stejného odevzdání, ale s vazbou na jiného studenta. Pomocí tlačítka \langle a následně pak pomocí tlačítka $\}$ se uživatel může pohybovat pomocí již uzamčených skupin obrázků využívajících vždy stejný zámek.

Poslední ovládací prvky slouží k přiřazení obrázků studentovi nebo části milníků (a není možné je použít v režimu s uzamykáním záznamů) a k vložení hodnocení učitele pro jednotlivé hodnocené části milníku (editor umožní hodnocení jen těch částí, které lze podle aktuálního přiřazení a použitých zámek hodnotit).

Domácí úkol 2 - Jan Petržílka (petrzja5)
 Tuesday, 1 January 2019 14:46

BIK-MLO: úkol č.2 *Toto je rukou psaný text* **Toto je vložený text**

K odevzdání na cvičení 5. ledna. Podobné příklady se objeví v písemce.

Příklad 1. Najděte minimální DNT formule

$$f(A, B, C, D) = \sum(0, 1, 3, 4, 8, 9, 11).$$

Ověříme *sestavíme mapu*

	A	B	C	D	f	f	TAB	TAB	TAB	DNT
	A=0	A=1								
0	0	0	0	0	1	1	00	10	11	1/1
1	0	0	0	1	1	1	11	00	00	1/1

	C			
	1	1	3	0
1	0	5	9	0
6				

Učitelova ovládací lišta:

- Stránky: 1 / 1 (18490)
- Části: 0: Více částí
- Student: Jan Petržílka (petrzja5)
- Část: 1: Úloha 1 (Min: 0, Max: 3, Bodů: 1)
- 2: Úloha 2 (Min: 0, Max: 3, Bodů: 3)

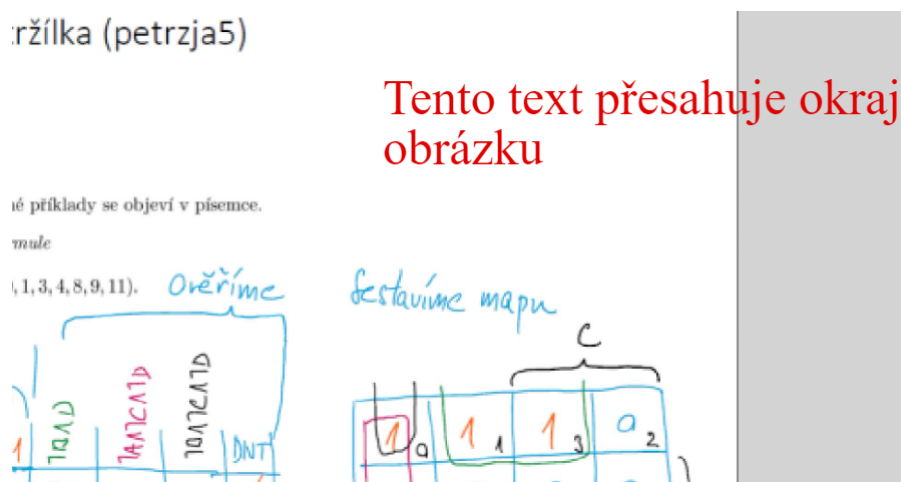
■ **Obrázek 8.27** Editor při zobrazení na výšku

Kromě viditelných ovládacích prvků editor reaguje na následující akce uživatele:

- Myši po stlačení levého tlačítka nebo stylusem je možné kreslit čáry nebo určovat polohu vkládaného textu. Po zapnutí kreslení prstem je možné tyto akce provádět také pohybem prstu na dotykovém displayi.
- Pomocí pohybu myši po stlačení pravého tlačítka je možné obrázkem pohybovat po editačním okně a pomocí kolečka po stlačení pravého tlačítka je možné obrázek zvětšovat a zmenšovat.

Obdobně je možné stylusem místo kreslení pohybovat obrázkem po editačním okně tak, že stlačíme a budeme držet stlačené tlačítko na tělo stylusu.

Pokud vložený text přesahuje plochu obrázku odpovídající odevzdané studentské práce nebo pokud se po rotaci obrázku některé části opravy ocitnou mimo plochu obrázku, aplikace automaticky zvětší zobrazovanou plochu tak, aby byly všechny opravy a vložené texty zobrazeny. Tato situace je zobrazena na obrázku 8.28.



■ **Obrázek 8.28** Zvětšení zobrazené plochy při přetečení okraje obrázku

8.5.9 Náhled studenta

Zobrazení opraveného obrázku studentovi je popsáno v části 8.6.6.

8.5.10 Generování QR kódů

Aplikace umožňuje generovat zkušební archy s QR kódy. Na jednotlivých listech zkušebních archů je zobrazen QR kód na obrázku 8.29.



■ **Obrázek 8.29** QR kód s identifikací studenta a příkladu

Část QR kódu před pomlčkou lze pomocí návazné funkčnosti aplikace přiřadit jednoznačně ke studentovi, část za pomlčkou pak označuje část milníku. Při zpracování odevzdaných souborů aplikace hledá QR kódy, podle části před pomlčkou přiřadí aktuální stránku a všechny další stránky až do dalšího QR kódu studentovi podle vazby QR kódu na studenta. Obdobně podle

části QR kódu za pomlčkou přiřadí aktuální stránku a všechny další stránky až do dalšího kódu za pomlčkou části milníku.

Generování zkušebních archů aplikace umožňuje pomocí funkčnosti na obrázku 8.30.

Písemky

Generování zkušebních archů

Vytvoření souboru

Počet studentů: (s <= 1500, s * p <= 18000)

Počet částí: (p <= 100, s * p <= 18000)

Komentář:

Dodatečný test

Vytvořit požadavek

Vytvořené soubory

ID	Datum	Studentů	Částí	Od	Do	Komentář	Operace
26	09.04.2023	15	7	450	554		Stáhnout
27	09.04.2023	5	7			Dodatečný test	Čeká

■ **Obrázek 8.30** Generování zkušebních archů s QR kódy

Vytvořené soubory

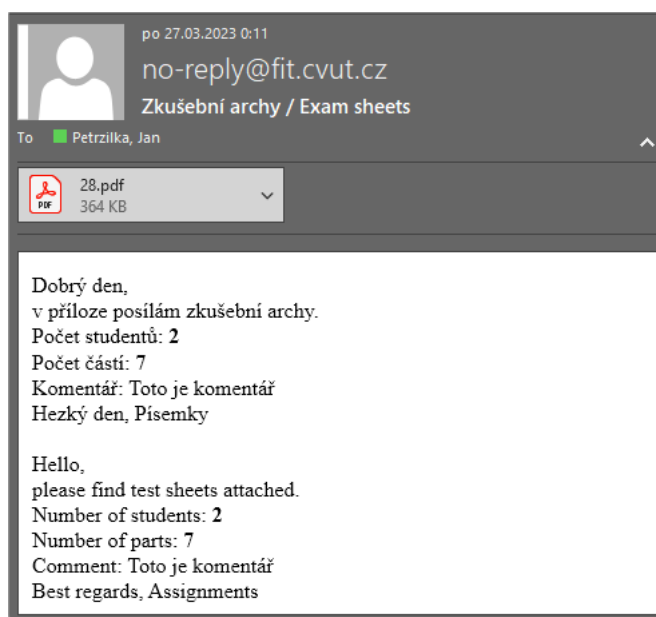
[Ukázat další](#)

ID	Datum	Studentů
27	09.04.2023	5
49	17.04.2023	1

■ **Obrázek 8.31** Tlačítko pro zobrazení dalších souborů

Uživatel vloží počet studentů a počet příkladů a požádá aplikaci o vygenerování zkušebních archů. Aplikace požadavek přijme a zařadí do fronty k vyřízení. Jakmile jsou zkušební archy vytvořeny, odešle je aplikace na e-mailovou adresu uživatele e-mailovou zprávou na obrázku 8.32 a nabídne je ke stažení na původní stránce.

Pokud počet vygenerovaných souborů přesahuje 10, tak aplikace zobrazí jen posledních 10 souborů a tlačítko na obrázku 8.31, pomocí něhož lze zobrazovat další soubory.



■ **Obrázek 8.32** E-mail se zkušebními archy

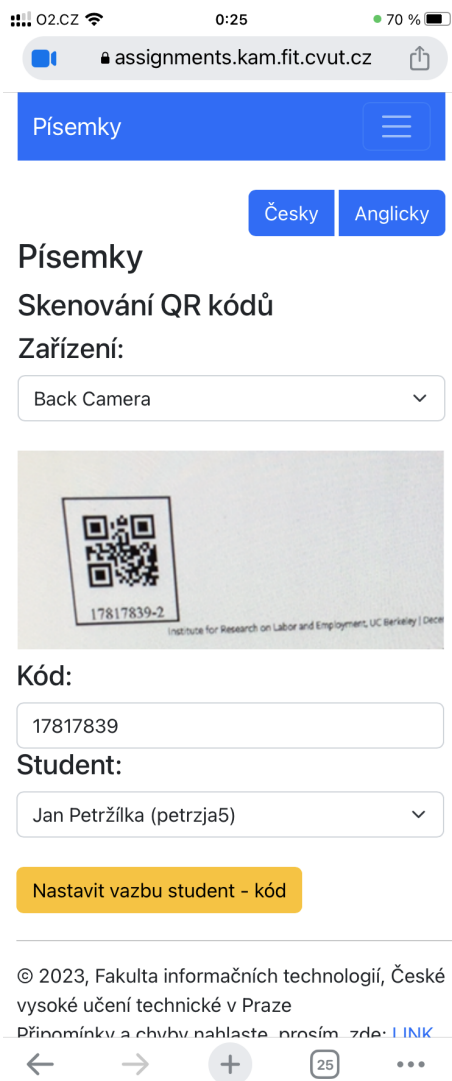
Tento způsob umožňuje vyučujícímu generovat zkušební archy také v jiných aplikacích za předpokladu, že dodrží vymezený rozsah použitých QR kódů ohraničený hodnotami identifikátorů ve sloupcích „Od“ a „Do“ na obrázku 8.30, které odpovídají části kódu před pomlčkou. Jednou vygenerované QR kódy lze použít v dalších letech pro jiné studenty, nebo naopak v budoucnosti lze studentovi při zápisu ke studiu přiřadit QR kód, který bude studenta během studia provázet a který bude student vkládat do svých výstupů (nebo je lepit pomocí samolepek).

Předpokládaný způsob využití je takový, že vyučující připraví před zkouškou nebo písemkou dostatečné množství zkušebních archů a během trvání zkoušky nebo písemky je přiřadí studentům.

8.5.11 Skenování QR kódů

K přiřazení části QR kódu před pomlčkou studentovi slouží mobilní webová stránka, kterou lze otevřít ze seznamu paralelek – tím je zajištěno, že aplikace bude mít správný seznam studentů k přiřazení. Mobilní stránka je zobrazena na obrázku 8.33.

Uživatel může vybrat přední nebo zadní kameru mobilu. Jakmile aplikace rozpozná QR kód, vyplní jeho část před pomlčkou do editovatelného pole – v případě potřeby je možné kód do pole vložit ručně a to i bez použití mobilního telefonu. Uživatel dále ze seznamu vybere studenta a vazbu mezi kódem a studentem do aplikace uloží.



■ **Obrázek 8.33** Přiřazení QR kódu ke studentovi

8.5.12 Prezentace výsledků

Aplikace umožňuje prezentovat výsledky studentů a srovnávat jejich hodnocení u milníků nebo jejich částí napříč paralelkami stejného předmětu v jednom semestru.

Písemky

Výsledky

Nastavení B221 / NI-PDB

Porovnávaný milník:

Aktivní účast 3

Porovnávaná část:

2

Paralelka s porovnávanými studenty:

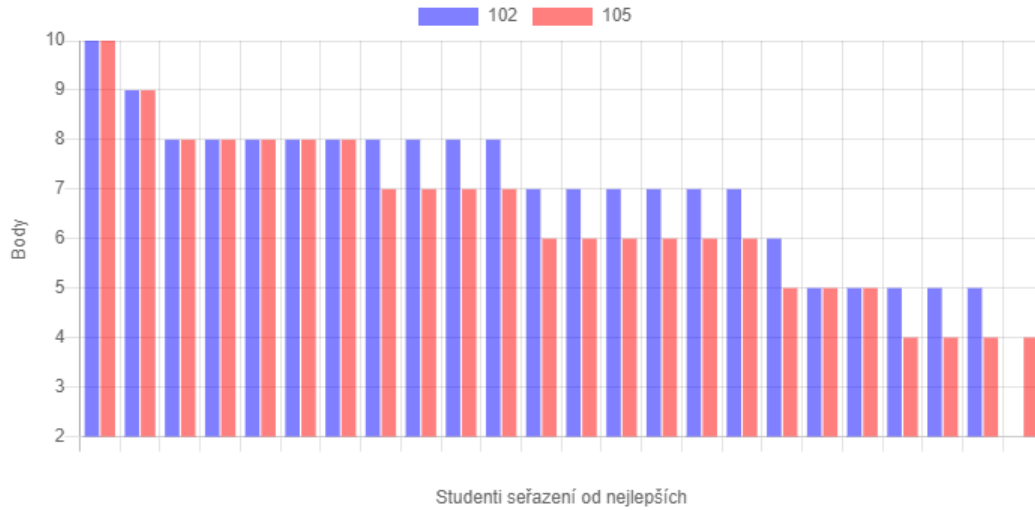
Kód	Popis	Vyučující	Výběr
1	přednáška pondělí každý týden (12:45:00 - 14:15:00), T9:107	Ing. Michal Valenta Ph.D. (valenta)	<input type="checkbox"/> Zobrazit
101	cvičení středa sudý týden (18:00:00 - 19:30:00), T9:351	Ing. Yelena Trofimova Ph.D. (trofiyel)	<input type="checkbox"/> Zobrazit
102	cvičení čtvrtek sudý týden (16:15:00 - 17:45:00), T9:350	Ing. Michal Valenta Ph.D. (valenta)	<input type="checkbox"/> Zobrazit
103	cvičení středa lichý týden (18:00:00 - 19:30:00), T9:351	Ing. Yelena Trofimova Ph.D. (trofiyel)	<input type="checkbox"/> Zobrazit
104	cvičení čtvrtek lichý týden (16:15:00 - 17:45:00), T9:350	Ing. Michal Valenta Ph.D. (valenta)	<input type="checkbox"/> Zobrazit
105	cvičení pondělí lichý týden (16:15:00 - 17:45:00), T9:351	Ing. Michal Valenta Ph.D. (valenta)	<input type="checkbox"/> Zobrazit
ex-1244874259705	Zkouška (TH:A-942) 05.01.2023 11:00:00 - 05.01.2023 12:00:00	Ing. Michal Valenta Ph.D. (valenta), Ing. Yelena Trofimova Ph.D. (trofiyel)	<input type="checkbox"/> Zobrazit

■ **Obrázek 8.34** Výběr dat pro srovnání výsledků

Výběr srovnávaných údajů uživatel provede pomocí formuláře na obrázku 8.34. Uživatel vybere pro daný předmět milník a paralelky ke srovnání. Aplikace s pomocí jeho SSO přihlašovacího tokenu získá data z Grades pro vybrané paralelky a zobrazí je do grafu na obrázku 8.35 tak, že u každé paralelky seřadí výsledky jednotlivých studentů od nejlepších po nejhorší. Hodnota -1 odpovídá chybějícím datům například v situaci, kdy různé paralelky mají různý počet studentů.

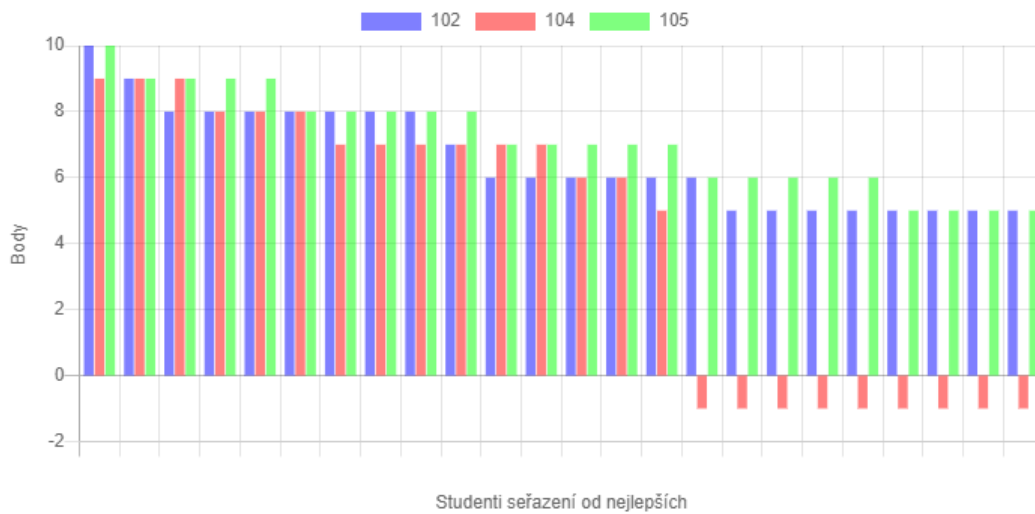
Obdobným způsobem může uživatel vybrat jen část milníku a aplikace využije uložených aplikačních dat k zobrazení výsledku.

Výsledky pro B221 / NI-PDB / Zkouška - část 1 (pojmy, koncepty)



■ **Obrázek 8.35** Srovnání bodového hodnocení milníku studentů dvou paralelek, zobrazená data jsou náhodná

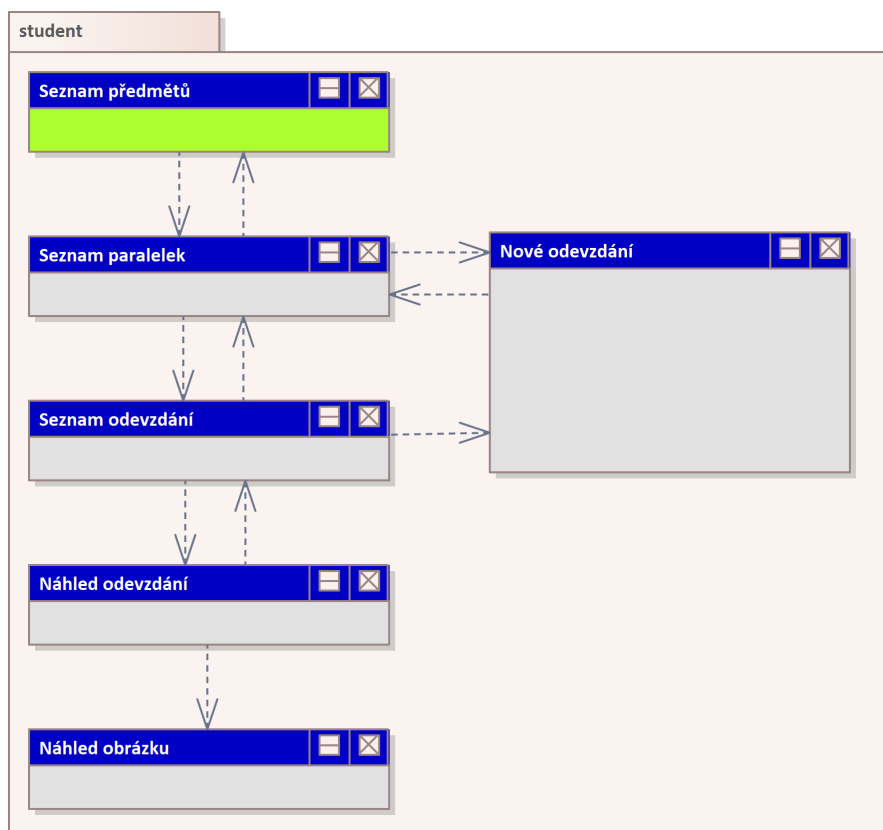
Výsledky pro B221 / NI-PDB / Aktivní účast 3 / 2



■ **Obrázek 8.36** Srovnání bodového hodnocení části milníku studentů tří paralelek, zobrazená data jsou náhodná

8.6 Stránky pro studenty

Stránky pro studenty jsou spolu s přechody mezi stránkami zobrazeny na obrázku 8.37. Výchozí stránka je označena zeleně.



■ Obrázek 8.37 Stránky pro studenty s přechody

8.6.1 Seznam předmětů

Úvodní stránkou pro studenty je přehled jimi studovaných předmětů na obrázku 8.38. Student vybere semestr a aplikace k němu zobrazí studované předměty. Pokud je předmět nastaven pro použití v této aplikaci, je možné jej rozkliknout a pokračovat na zobrazení paralelek předmětu.

Písemky

Předměty

Seznam semestrů

B221 - Zimní 2022/2023

Seznam předmětů

Kód	Název	Operace
NI-DSS	Systémy podpory rozhodování	
NI-MPR	Magisterský projekt	
NI-NUR	Návrh uživatelského rozhraní	
NI-PDB	Pokročilé databázové systémy	Zobrazit
NI-SEP	Světová ekonomika a podnikání II.	

■ **Obrázek 8.38** Přehled studovaných předmětů

8.6.2 Seznam paralelek a milníků

Po rozkliknutí předmětu na přehledu studovaných předmětů aplikace zobrazí na obrázku 8.39 paralelky předmětu s milníky, které má student zapsané nebo u kterých aplikace eviduje jeho práce. U každé zobrazené paralelky je možné zobrazit existující odevzdání a v případě, kdy je u milníku nastaveno odevzdávání prací studentem, přejít na stránku pro nové odevzdání práce.

Písemky

Předmět

Seznam paralelek B221 / NI-PDB

Kód	Popis	Milník	Operace
105	cvičení pondělí lichý týden (16:15:00 - 17:45:00), T9:351	Aktivní účast 3	Zobrazit Odevzdat

Návrat

■ **Obrázek 8.39** Přehled paralelek s milníky

8.6.3 Nové odevzdání

Po přechodu na stránku pro nové odevzdání aplikace zobrazí formulář pro odeslání souborů na obrázku 8.40. V tomto případě má student na výběr, jestli odevzdá soubory pro všechny části milníku bez rozlišení části milníku nebo pro jednotlivé části milníku odděleně. Nejzazší termín pro odevzdání není učitelem určen a ve formuláři není vyplněn.

Aplikace akceptuje soubory s příponami pdf, jpg, jpeg a png.

Jakmile do formuláře vloží první soubor, tak aplikace skryje druhou, studentem nevyužitou možnost. V případě zobrazeném na obrázku 8.41 vložil student první soubor k možnosti odevzdání souboru bez rozlišení části. Do formuláře může vkládat další soubory, ale již nemůže vložit soubor s určením části, pro kterou jej odevzdává. V případě na tomto obrázku je nejzazší termín odevzdání určen a na formuláři je zobrazen.

Soubor může z formuláře odstranit nebo stáhnout a zkontrolovat. Jakmile soubory odevzdá, aplikace je odešle na server a uloží do fronty ke zpracování.

Písemky

Odevzdání souborů

Vytvořit odevzdání B221 / NI-PDB / 105 - cvičení pondělí lichý týden (16:15:00 - 17:45:00), T9:351 / Aktivní účast 3

Termín:

Soubory k odevzdání pro všechny části

Název souboru	Velikost	Operace
---------------	----------	---------

Vybrat soubory

Soubory k odevzdání pro část Úloha 1

Název souboru	Velikost	Operace
---------------	----------	---------

Vybrat soubory

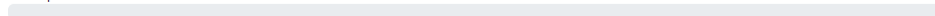
Soubory k odevzdání pro část Úloha 2

Název souboru	Velikost	Operace
---------------	----------	---------

Vybrat soubory

Komentář:

Postup odesílání 0 %



Odevzdat soubory

Návrat

■ **Obrázek 8.40** Formulář pro odevzdání práce studentem

Písemky

Odevzdání souborů

Vytvořit odevzdání B221 / NI-PDB / 105 - cvičení pondělí lichý týden (16:15:00 - 17:45:00), T9:351 / Aktivní účast 3

Termín: 26.04.2023 08:45

Soubory k odevzdání pro všechny části

Název souboru	Velikost	Operace
scan.pdf	2201866	Stáhnout Odstranit

[Vybrat soubory](#)

Komentář:

Postup odeslání 0 %

[Odevzdat soubory](#) [Návrat](#)

■ **Obrázek 8.41** Formulář pro odevzdání práce studentem s jedním vloženým souborem

8.6.4 Seznam odevzdání

Při zobrazení seznamu odevzdání aplikace zobrazí jejich přehled na obrázku 4.2. V tomto případě jsou zobrazena dvě odevzdání a obě jsou již opravená a ve stavu, který umožňuje zobrazení výsledku opravy studentům.

Soubory, které odeslal sám student, může stáhnout pomocí linku v seznamu. Kliknutím na tlačítko zobrazí náhled vybraného odevzdání.

Písemky

Seznam odevzdání

Seznam odevzdání B221 / NI-PDB / 105 - cvičení pondělí lichý týden (16:15:00 - 17:45:00), T9:351 / Aktivní účast 3

ID	Datum	Stav	Odevzdáno kým	Soubory	Operace
219	07.04.2023	Uvolněné pro studenty	Jan Petřílka (petrja5)	<ul style="list-style-type: none"> scan.pdf 	Náhled
220	08.04.2023	Uvolněné pro studenty	Jan Petřílka (petrja5)	<ul style="list-style-type: none"> image1.png 	Náhled

[Návrat](#) [Nové odevzdání](#)

■ **Obrázek 8.42** Seznam odevzdání zobrazený studentovi

8.6.5 Náhled odevzdání

Při zobrazení náhledu odevzdání aplikace zobrazí všechny soubory vybraného odevzdání a jejich stránky. U každé stránky zobrazí náhledový obrázek v malém rozlišení bez vložených oprav a tlačítko pro zobrazení obrázku stránky s velkým rozlišením včetně opravy vložené vyučujícím.

Při rozkliknutí náhledového obrázku aplikace zobrazí obrázek ve velkém rozlišení bez vrstvy opravy učitelem. Student tak vidí, co učitel opravoval.

Písemky

Odevzdání

Odevzdání B221 / NI-PDB / 105 - cvičení pondělí lichý týden (16:15:00 - 17:45:00), T9:351 / Aktivní účast 3 / 220

- Soubor Image1.png

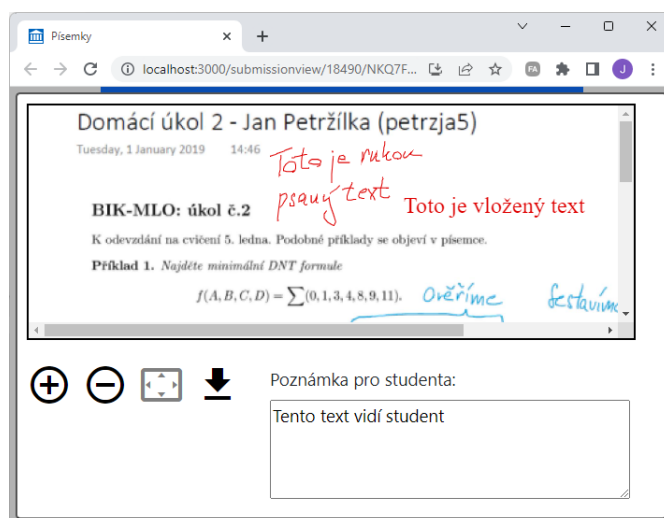
Stránka	Náhled	Operace
1 (18490)		Oprava

[Návrat](#)

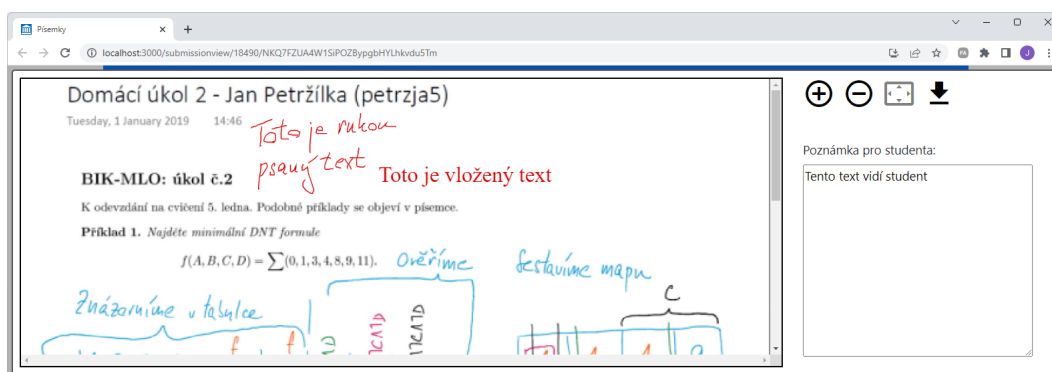
■ **Obrázek 8.43** Zobrazení obsahu odevzdání studentovi

8.6.6 Náhled obrázku s opravou

Při zobrazení náhledu obrázku s opravou aplikace zobrazí podle poměru stran okna prohlížeče opravený obrázek s vrstvou opravy vložené učitelem a eventuálně s poznámkou vloženou učitelem při opravě tak, jak je vidět na obrázcích 8.44 a 8.45.



■ **Obrázek 8.44** Náhled opraveného obrázku zobrazený studentovi na šířku



■ **Obrázek 8.45** Náhled opraveného obrázku zobrazený studentovi na výšku

Jednotlivé ovládací prvky umožňují zvětšení zobrazeného obrázku, zmenšení zobrazeného obrázku, nastavení zvětšení obrázku tak, že jeho šířka odpovídá šířce okna, a stažení souboru se spojenými vrstvami původního obrázku a jeho opravy.

8.7 Stránky pro správce

Stránky pro správce jsou spolu s přechody mezi stránkami zobrazeny na obrázku 8.46.



■ **Obrázek 8.46** Stránky pro správce s přechody

8.7.1 Lokalizace – seznam stránek

Po otevření položky menu Lokalizace otevře aplikace stránku na obrázku 8.47 se seznamem všech stránek s lokalizačními štítky. Uživatel může přidat novou stránku nebo otevřít editaci některé ze stránek.

Písemky

Lokalizace

Seznam stránek:

#	Název stránky	Operace
110	changePassword	Editace
111	checkPassword	Editace
86	editor	Editace
113	footer	Editace
107	uploadData	Editace
109	userManagement	Editace

Nová stránka

Název nové stránky

Vložit novou stránku

■ **Obrázek 8.47** Seznam stránek (zkráceno)

8.7.2 Lokalizace – obsah stránky

Po otevření editace jedné stránky aplikace zobrazí seznam štítků zvolené stránky na obrázku 8.48. Uživatel se může podívat na aktuální nastavení jednotlivých štítků, změnit jejich hodnotu, přidat nový štítek nebo si nechat zobrazit historii nastavení štítku.

Písemky

Lokalizace

Obsah stránky: login

CVUT (petrzja5, 04.04.2023 06:43:52) ∨

password (petrzja5, 22.08.2022 23:09:32) ∨

submit (petrzja5, 04.04.2023 06:45:58) ∧

Česky:

Přihlásit

Anglicky:

Log in

[Uložit prvek](#)

[Historie prvku](#)

title (petrzja5, 04.04.2023 06:43:20) ∨

username (petrzja5, 22.08.2022 23:09:20) ∨

wrongCredentials (petrzja5, 23.08.2022 10:05:14) ∨

Nový prvek

Název nového prvku

Česky

Anglicky

[Vložit nový prvek](#)

[Zpět](#)

■ **Obrázek 8.48** Obsah jedné stránky – příklad pro stránku přihlášení

8.7.3 Lokalizace – historie prvku

Na přehledu historických hodnot štítků na obrázku 8.49 jsou zobrazeny historické hodnoty zvoleného štítku.

Písemky
Lokalizace

Historie prvku ze stránky login / submit

submit (petrja5, 04.04.2023 06:45:58) ^

Česky:
Přihlásit

Anglicky:
Log in

submit (petrja5, 22.08.2022 23:09:49) ^

Česky:
Log in

Anglicky:
Přihlásit

Zpět

■ **Obrázek 8.49** Historie nastavení prvku

8.7.4 Log

Na stránce se záznamy o práci aplikace na obrázku 8.50 jsou zobrazeny události zaznamenané v aplikačním logu. Záznamy je možné filtrovat. Na konkrétním zobrazeném příkladě vidíme sekvenci práce uživatele s přihlašovacím jménem legerjan s aplikací. Uživatel byl přihlášen pomocí SSO přihlášení, před vypršením platnosti jeho SSO tokenu aplikace jeho platnost obnovila a hodinu po jeho poslední operaci na aplikačním serveru token z paměti aplikačního serveru odstranila.

Písemky

Log

Záznamy

Od Do Řádků

#	Datum	Třída	Úroveň	Text	Podrobnosti	Uživatelské id	Uživatelské jméno
69843	2023-04-05T11:01:29.374238Z	class ...bl.service.SessionToken	info	SSO token removed	legerjan		
69842	2023-04-05T10:51:44.303969Z	class ...bl.service.SessionToken	info	SSO token refreshed	legerjan		
69841	2023-04-05T10:51:44.218467Z	class ...bl.service.SessionToken	info	Refreshing SSO token	legerjan		
69840	2023-04-05T10:01:29.099200Z	class ...fe.teacher.TeacherCourses	info	rest call	getTeacherSubmissions		legerjan
69839	2023-04-05T10:01:13.723159Z	class ...fe.teacher.TeacherCourses	info	rest call	getTeacherCourse		legerjan
69838	2023-04-05T10:00:56.403491Z	class ...fe.teacher.TeacherCourses	info	rest call	putParallelSetting		legerjan
69837	2023-04-05T10:00:37.840745Z	class ...fe.teacher.TeacherCourses	info	rest call	getTeacherParallel		legerjan
69836	2023-04-05T10:00:37.839989Z	class ...fe.teacher.TeacherCourses	info	rest call	getTeacherCourse		legerjan
69835	2023-04-05T10:00:33.076612Z	class ...fe.teacher.TeacherCourses	info	rest call	getTeacherCourse		legerjan
69834	2023-04-05T10:00:29.482344Z	class ...fe.teacher.TeacherCourses	info	rest call	putParallelSetting		legerjan

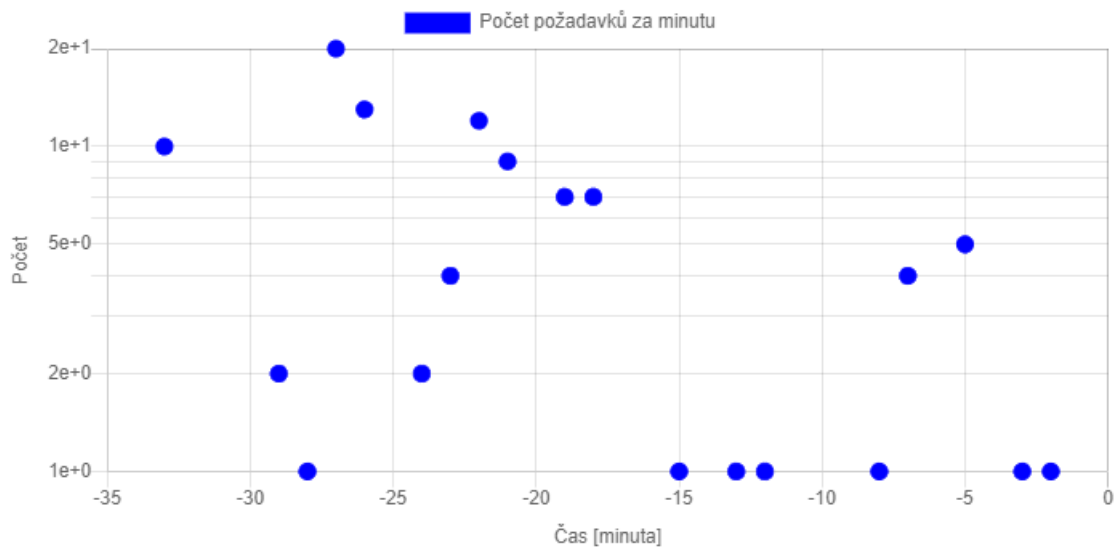
« < 1 2 ... 38 39 40 41 42 7020 7021 > »

■ **Obrázek 8.50** Aplikační log

8.7.5 Statistika

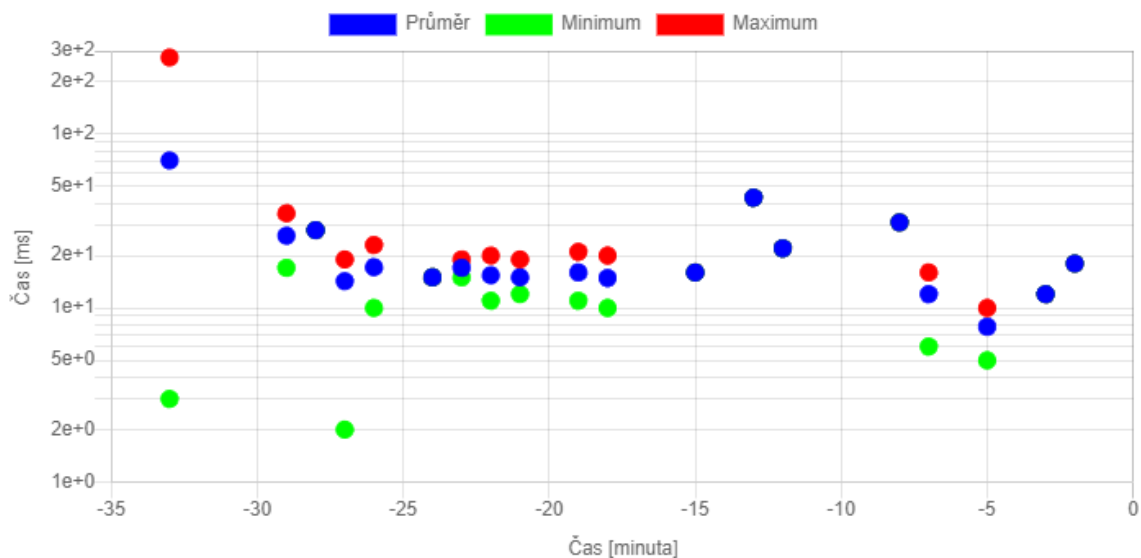
Při zobrazení statistických informací aplikace zobrazuje počet vyřízených požadavků za minutu na obrázku 8.51, dobu odpovědi měřenou za jednu minutu na obrázku 8.52 a využití heapu během jedné minuty na obrázku 8.53. Časové údaje na vodorovné ose jsou vztaženy k aktuálnímu času a jsou uchovávány 7 dní zpětně. Při restartu aplikačního serveru je historie měření ztracena.

Počet požadavků za minutu



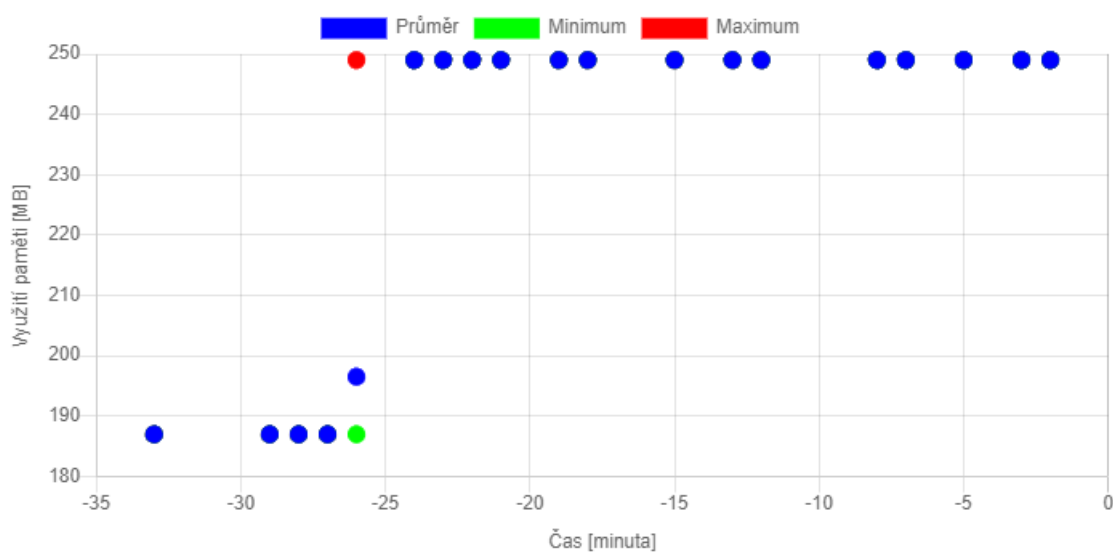
■ Obrázek 8.51 Počet požadavků za minutu

Doba odpovědi měřená po minutách



■ Obrázek 8.52 Doba odpovědi měřená po minutách

Využití heapu měřené po minutách



■ **Obrázek 8.53** Využití heapu po minutách

8.7.6 Změna hesla

Na stránce pro změnu hesla na obrázku 8.54 může lokálně přihlášený uživatel změnit svoje heslo. Aplikace vyžaduje minimální délku hesla 8 znaků, použití velkého písmene, malého písmene, číslice a nealfanumerického znaku. Heslo je kontrolováno při jeho vkládání do pole a aplikace indikuje zjištěné nedostatky.

Písemky

Změna hesla

Stávající heslo:

Nové heslo:

Chybí číslice

Nové heslo znovu:

Hesla se neshodují

Změnit heslo

■ **Obrázek 8.54** Změna hesla lokálního uživatele

8.7.7 Správa uživatelů

Na stránce pro správu lokálních uživatelů na obrázku 8.55 může správce nastavit nové heslo existujícímu lokálnímu uživateli, existujícího lokálního uživatele smazat nebo nového lokálního uživatele založit.

Písemky
Správa uživatelů
Nastavení uživatelů

Seznam uživatelů:

Jan Petržílka (petrzja5) ▾

Nové heslo:

Nové heslo znovu:

Nastavit nové heslo **Smazat uživatele**

Nový uživatel

Nové uživatelské jméno:

Nové heslo:

 ⓘ
Chybí velké písmeno

Nové heslo znovu:

 ⓘ
Hesla se neshodují

Vytvořit nového uživatele

■ **Obrázek 8.55** Správa lokálních uživatelů

8.7.8 Systémové nastavení

Na stránce systémového nastavení na obrázku 8.56 může uživatel vypnout nebo zapnout jazykovou lokalizaci pro **používaný prohlížeč**. Při vypnuté lokalizaci jsou místo českých a anglických popisů zobrazeny kódy lokalizačních štítků. Po eventuální změně nastavení lokalizace se nové texty zobrazí až po smazání cache **v používaném prohlížeči a na serveru**.

Na stránce je dále uveden čas poslední zahájení a ukončení synchronizace se systémem KOS, případně je možné si novou synchronizaci vyžádat. Aplikace hlídá, aby synchronizace neběžela příliš často a negenerovala tak zátěž pro systém KOS.

Uživatel může na této stránce dále nastavit parametry pro integraci se systémem KOS a pro přihlášení přes SSO a tedy pro integraci s aplikací Grades.

Písemky

Systémová nastavení

Lokalizace

Synchronizace s KOSem

Začátek poslední synchronizace: Fri Apr 07 04:00:29 UTC 2023

Konec poslední synchronizace: Fri Apr 07 04:02:09 UTC 2023

Připojení na KOS API

Client Id:

Client Secret:

Připojení na SSO

Client Id:

Client Secret:

Redirection URL:

■ **Obrázek 8.56** Nastavení systému

Testování, zhodnocení a uvedení do provozu

9.1 Testování

Funkčnost aplikace jsem ověřil jejím testováním při vývoji. Značným problémem byla nedostupnost testových instalací školních systémů, takže jsem měl zejména při komunikaci s Grades k dispozici jen data o mnou absolvovaných předmětech a nedisponoval jsem přístupem k datům s rolí vyučujícího. Testování přístupu k aplikaci Grades v učitelské roli tak otestoval vedoucí diplomové práce; při vývoji aplikace mi poskytl značnou součinnost Ing. Michal Valenta, Ph.D. a umožnil mi použít data jím vyučovaného předmětu NI-PDB (proto jsou ukázky aplikačních stránek v této práci zejména z předmětu NI-PDB).

Až do vytvoření školního serveru `assignments.kam.in.fit.cvut.cz` přístupného na veřejné adrese `https://assignments.kam.fit.cvut.cz` na konci března roku 2023 jsem pro testování používal virtuální privátní server hostovaný u společnosti WEDOS Internet, a. s. a více různých subdomén domény `mujfit.cz`.

V době odevzdání této práce probíhá zkusmé použití aplikace při výuce, kdy je její funkčnost ověřována, jsou identifikovány chyby implementace a tyto chyby odstraňuji.

Žádné formální testování funkčnosti aplikace ani použitelnosti jejího uživatelského rozhraní neproběhlo. Nebyla vytvořena testová data a sestaveny testové případy a testové scénáře. Nejsou proto k dispozici protokoly o provádění testů a není známo, jaké procento testované funkčnosti v testech obstálo, jaké procento vykazovalo chyby a jak byly tyto chyby kategorizovány.

Kód aplikace není pokryt automatizovanými testy.

Statické testování s využitím aplikace SonarQube probíhalo pouze v omezeném rozsahu a to zejména v začátcích aplikačního vývoje.

9.1.1 Použitý hardware

Aplikaci jsem vyvinul a základním způsobem otestoval na notebooku Lenovo ThinkPad T480 s 32 GB RAM s operačním systémem Microsoft Windows 10 Professional a později Microsoft Windows 11 Professional. Funkčnost editoru obrázků v Linuxu jsem ověřil na notebooku Lenovo ThinkPad T410 s 8 GB RAM s operačním systémem Ubuntu 22.04.2 LTS.

Schopnost editovat obrázky pomocí tabletu připojeného do USB portu jsem implementoval pomocí tabletu One by Wacom M.

Použití dotykového monitoru ve Windows jsem implementoval s využitím dotykového monitoru Asus MB169B+. Použití dotykového displeje se stylusem ve Windows jsem implementoval

s využitím notebooku Lenovo ThinkBook 14s Yoga G2 IAP s Windows 11 Professional.

Mobilní část se skenováním QR kódů jsem implementoval a funkčnost jsem ověřil s využitím následujících mobilních zařízení s operačním systémem iOS a Android: Apple iPhone SE 2022, Apple iPad 2018, Samsung Galaxy A3, Lenovo Yoga 3.

9.1.2 Kapacitní testy

S nastavenou velikostí aplikačního heapu 2500 MB aplikace úspěšně zpracovala PDF soubor s oskenovanými obrázky na 3200 stranách. Soubor byl velký 1,63 GB.

9.2 Uživatelské testování

Aplikaci bylo možné použít až v průběhu letního semestru školního roku 2022/2023 (cca od půlky dubna 2023), což výrazně omezilo možnosti jejího uživatelského testování a získání zpětné vazby od uživatelů. Vzhledem k povaze aplikace a nedostupnosti testových instancí systému KOS a aplikace Grades není možné aplikaci volně testovat – testování musí probíhat při výuce na skutečných datech. Relevantní zkušenosti tak vzniknou až při jejím předpokládaném použití během zimního semestru školního roku 2023/2024.

Někteří vyučující z Katedry aplikované matematiky aplikaci v omezeném rozsahu při výuce použili a předali mi svoje připomínky. Všechny získané podněty jsem zapracoval.

Kromě opravy drobných chyb (jako například zobrazování anglické verze kalendáře na české verzi stránky) jsem v aplikaci udělal následující funkční změny:

- Podle původního návrhu začínal týden na českém kalendáři pondělním a na anglickém nedělí. Uživatelé mohou používat obě jazykové mutace i případě, kdy nejsou roditelým anglickým mluvčím. Proto jsem implementaci změnil tak, že obě jazykové mutace používají kalendář s týdny začínajícími v pondělí.
- Umožnil jsem smazání obrázku z odevzdání nebo celého odevzdání. V případě obrázku se jedná o nový stav, v případě celého odevzdání o příznak tohoto odevzdání – operace je tedy vratná. V případě obrázků je funkčnost potřeba pro odstranění pomocných listů jako jsou oddělovače odevzdání, v případě odevzdání pro odstranění odevzdání nahraných k jinému předmětu, paralelce nebo studentovi.
- Termín odevzdání nejpozději k půlnoci určeného dne jsem změnil tak, že vyučující může určit okamžik během dne jako libovolný celočíselný násobek 5 minut. Tato změna jednak umožní poskytnout studentům více času na dokončení práce, jednak rozloží zátěž serveru.
- Zobrazení seznamu vygenerovaných zkušebních archů jsem omezil na 10 položek s možností zvýšit zobrazovaný počet souborů dvakrát (a to i opakovaně). Aktuálně není jisté, jak budou zkušební archy s QR kódy používány, a tak jsem neimplementoval mazání starých záznamů.
- Nejvýznamnější změnou byla nová možnost přechodu mezi jednotlivými odevzdáními. V původní verzi aplikace se musel vyučující po dokončení opravy jednoho odevzdání vrátit na stránku se seznamem odevzdání a na ní dohledat další odevzdání k opravě. Nově jsem na stránku pro opravování jednoho odevzdání a uložení výsledku do Grades přidal tlačítka, která uživateli umožňují pohyb mezi odevzdáními podle pořadí odevzdání a nastavených filtrů při posledním použití stránky se seznamem odevzdání.

Předpokládám, že další podobné úpravy stejně jako opravy chyb bude potřeba pro hladké používání aplikace provést i v budoucnosti.

9.3 Zhodnocení a uvedení do provozu

Aplikace, kterou jsem v rámci této diplomové práce navrhl a implementoval, je založena na principech a technologiích, které dávají dobré předpoklady pro její budoucí rutinní používání při opravování studentských písemných prací.

V průběhu uplynulých 12 měsíců jsem pracoval na analýze požadavků a implementaci klíčových funkcí. Výsledná aplikace byla zatím použita pouze okrajově několika uživateli v režimu, který lze považovat spíše za rozšířené testování než za skutečné nasazení do provozu.

Podle diskusí s vedoucím diplomové práce a dalšími klíčovými uživateli aplikace se domnívám, že na Katedře aplikované matematiky FIT ČVUT v Praze působí hned několik vyučujících, kteří aplikaci plánují při výuce použít.

V aplikaci jsou zatím příležitostně nalézány chyby bránící jejímu řádnému užívání, které jsem zatím schopen rychle odstraňovat. Žádný významný problém, jehož řešení by vyžadovalo strukturální změny celé aplikace nebo jejích významných částí, zatím nebyl identifikován. Až budou tyto běžné chyby opraveny a aplikace bude stabilizovaná, přijde čas na identifikaci možných optimalizací stávající implementace a eventuální změny nebo rozšíření její funkčnosti na základě zkušeností s jejím používáním.

Předpokládám, že po naplnění databáze daty bude potřeba některé databázové dotazy nebo služby aplikačního serveru optimalizovat. Poměrně snadná a přínosná může být paralelizace generování dokumentů s QR kódy nebo paralelizace generování obrázků z odevzdaných PDF dokumentů.

Dobře oddělená vrstva uživatelského rozhraní umožňuje vytvoření mobilního uživatelského rozhraní, a to zejména pro studentskou část aplikace.

Vzhledem k minulým zkušenostem s podobnými, ale méně komplexními řešeními na Katedře aplikované matematiky FIT ČVUT v Praze, které nikdy k trvalému rutinnímu provozu nedospěly, se domnívám, že pro udržení této aplikace v provozu bude potřeba trvalé manažerské pozornosti určených vyučujících z katedry, kteří se budou věnovat produktovému managementu vzniklé aplikace, a vytvoření nějakého technického řešitelského týmu, který se bude orientovat v technickém řešení této aplikace a bude zajišťovat kontinuitu technické podpory pro eventuální další studenty, kteří aplikaci budou v rámci svých studentských prací měnit nebo rozšiřovat.

Na fakultě je používáno několik podobných aplikací a zkušenosti s jejich provozem budou jistě relevantní i pro tuto aplikaci.

Kapitola 10

Závěr

Zadání diplomové práce jsem splnil v celém požadovaném rozsahu.

Navrhl jsem a realizoval prototyp webové aplikace pro podporu opravování studentských písemných prací. Aplikace poskytuje následující funkčnosti:

- Vložení oskenovaných písemných prací a jejich přiřazení studentům, případně nahrání prací samotným studentem.
- Opravení práce ve webovém prohlížeči způsobem umožňujícím vložit nebo pomocí stylusu ručně připsat poznámky a komentáře.
- Distribuce opravených prací s opravami zpět studentům.
- Evidence bodového ohodnocení jednotlivých částí práce a prezentace souhrnných výsledků.

Postupoval jsem v těchto krocích:

- Analyzoval jsem předmětnou doménu a určil jsem možnosti technického řešení jak na straně editoru obrázků, tak na straně integrace se školními systémy.
- Provedl jsem průzkum a analýzu požadavků od učitelů se zkušenostmi s on-line opravováním písemek a úkolů v matematických předmětech na KAM FIT ČVUT v Praze.
- Navrhl jsem a implementoval serverovou část aplikace.
- Navrhl jsem a implementoval funkční prototyp klientské aplikace pro webové prohlížeče.
- Navrhl jsem integraci s aplikací Grades a systémem KOS, integraci jsem realizoval a ověřil její funkčnost.
- Aplikaci jsem ve spolupráci s vedoucím práce a s klíčovými uživateli otestoval a zjištěné nedostatky jsem odstranil.
- Zhodnotil jsem použitelnost výsledného prototypu aplikace a navrhl jsem způsob jejího uvedení do budoucího provozu.

Aplikace je k dispozici uživatelům z řad vyučujících a studentů FIT ČVUT v Praze na adrese <https://assignments.kam.fit.cvut.cz>.

Literatura

- [1] BRANDEJSOVÁ, Jitka, Michal BRANDEJS, Gustav NOVOTNÝ. *Na Masarykově univerzitě opravují písemky počítače* [online]. AMV, 2008. [přístup 4. 4. 2023]. Dostupné z https://is.muni.cz/clanky/2008_AMV
- [2] SKETCH.IO, Inc. *Sketchpad* [software]. 2023. [přístup 4. 4. 2023]. Dostupné z <https://sketch.io/>
- [3] DELFINO, *Drawing on a HTML5 Canvas: how to draw at tip of mouse cursor* [online]. [přístup 12. 6. 2023]. Dostupné z <https://stackoverflow.com/questions/52211465/drawing-on-a-html5-canvas-how-to-draw-at-tip-of-mouse-cursor>
- [4] MOZILLA AND INDIVIDUAL CONTRIBUTORS. Web API. In: *MDN Web Docs* [online]. Mozilla, 2023. [přístup 16. 4. 2023]. Dostupné z <https://developer.mozilla.org/en-US/docs/Web/API>
- [5] NATIONAL ASSOCIATION OF SECONDARY SCHOOL PRINCIPALS. *Value-added measured in teacher evaluation* [online]. [přístup 13. 11. 2022] Dostupné z <https://www.nassp.org/top-issues-in-education/position-statements/value-added-measures-in-teacher-evaluation/>
- [6] GLAZERMAN, Steven, Dan GOLDHABER, Susanna LOEB, Stephen RAUDENBUSH, Douglas STAIGER, and Grover J. Russ Whitehurst. *Evaluating Teachers: The Important Role of Value-Added*. [online]. The Brookings Institution, [přístup 13. 11. 2022], Dostupné z <https://www.brookings.edu/research/evaluating-teachers-the-important-role-of-value-added/>
- [7] PETRŽÍLKA, Jan. *Návrh a implementace webového rozhraní pro telerehabilitaci*. Diplomová práce. České vysoké učení technické v Praze, Fakulta informačních technologií, 2021.
- [8] THE POSTGRESQL GLOBAL DEVELOPMENT GROUP. *PostgreSQL* [software]. 2021. [přístup 13. 4. 2021]. Dostupné z <https://www.postgresql.org/>
- [9] Oracle AND/OR ITS AFFILIATES. *Java* [software]. 2023. [přístup 16. 4. 2023] Dostupné z <https://www.oracle.com/java/>
- [10] VMWARE, INC. OR ITS AFFILIATES. *Spring* [software]. 2021. [přístup 13. 4. 2021]. Dostupné z <https://spring.io>
- [11] VMWARE, INC. OR ITS AFFILIATES. *Guides* [online]. VMware, Inc. or its affiliates, 2021. [cit. 13. 4. 2021]. Dostupné z <https://spring.io/guides>

- [12] VMWARE, INC. OR ITS AFFILIATES. *Spring Boot* [software]. 2021. [přístup 13. 4. 2021]. Dostupné z <https://spring.io/projects/spring-boot>
- [13] VMWARE, INC. OR ITS AFFILIATES. *Spring Security* [software]. 2021. [přístup 13. 4. 2021]. Dostupné z <https://spring.io/projects/spring-security>
- [14] HIBERNATE. *Hibernate ORM* [software]. 2023. [přístup 16. 4. 2023]. Dostupné z <https://hibernate.org/orm/>.
- [15] THE APACHE SOFTWARE FOUNDATION. *Maven* [software]. 2021. [přístup 12. 4. 2021]. Dostupné z <https://maven.apache.org/>
- [16] DANFICLE. *openhtmltopdf* [software]. 2023. [přístup 15. 4. 2023]. Dostupné z <https://github.com/danfickle/openhtmltopdf>
- [17] ZXING. *ZXing (Zebra Crossing) barcode scanning library for Java, Android* [software]. 2023. [přístup 15. 4. 2023]. Dostupné z <https://github.com/zxing/zxing>
- [18] ANGULAR TEAM AT GOOGLE. *Angular* [software]. 2021. Dostupné z <https://angular.io/>
- [19] META. *React* [software]. 2023. Dostupné z <https://react.dev/>
- [20] ALFREDSSON, Adam. *react-zxing*. [software]. 2023. [přístup 15. 4. 2023] Dostupné z <https://github.com/adamalfredsson/react-zxing>
- [21] *Chart.js* [software]. 2023. [přístup 16. 4. 2023] Dostupné z <https://www.chartjs.org/>
- [22] *APPS MANAGER*. [software] ČVUT v Praze, [přístup 7. 4. 2023], Dostupné z <https://auth.fit.cvut.cz/manager/app-types.xhtml>
- [23] JIRŮTKA, Jakub. *KOSapi*. [online]. ČVUT v Praze, [přístup 6. 4. 2023], Dostupné z <https://kosapi.fit.cvut.cz/projects/kosapi>
- [24] *FIT Classification REST API*. [online]. ČVUT v Praze, [přístup 6. 4. 2023], Dostupné z <https://grades.fit.cvut.cz/api/v1/swagger-ui.html>
- [25] JIRŮTKA, Jakub. *OAuth 2.0*, [online]. ČVUT v Praze, [přístup 6. 4. 2023], Dostupné z <https://rozvoj.fit.cvut.cz/Main/oauth2>
- [26] NIELSEN NORMAN GROUP. *10 Usability Heuristics for User Interface Design*. [online]. Nielsen Norman Group, [přístup 7. 4. 2023], Dostupné z <https://www.nngroup.com/articles/ten-usability-heuristics/>
- [27] FJÄLLEMARK, Hans, John PAPA & Tim FERRELL. *Toastr* [software]. 2014. [přístup 13. 4. 2021]. Dostupné z <https://codeseven.github.io/toastr/>
- [28] KHADRA, Fadi. *React-Toastify*. [software]. 2023. [přístup 15. 4. 2023] Dostupné z <https://github.com/fkhadra/react-toastify>
- [29] BOOTSTRAP TEAM. *Bootstrap* [software]. 2021. [přístup 13. 4. 2021]. Dostupné z <https://getbootstrap.com/>
- [30] COLLINGS, Stephen J., Matthew HONNIBAL, Pieter VANDERWERFF. *React-Bootstrap*. [software]. 2023. [přístup 15. 4. 2023] Dostupné z <https://github.com/react-bootstrap/react-bootstrap>
- [31] BALSAMIQ STUDIOS, LLC. *Balsamiq Mockups 3* [software]. 2017. [přístup 13. 4. 2021]. Nástupnický software se stejnými funkcemi, ale jinou podkladovou technologií dostupný z <https://balsamiq.com/wireframes/mockups3fordesktop/>

- [32] SPARX SYSTEMS PTY LTD. *Enterprise Architect 15.2 Corporate Edition* [software]. 2021. [přístup 13. 4. 2021]. Dostupné z <https://www.sparxsystems.com/>
- [33] TRACKER SOFTWARE PRODUCTS (Canada) Ltd. *PDF-XChange Editor 9.0* [software]. 2021. [přístup 12. 4. 2021]. Dostupné z <https://www.tracker-software.com/product/pdf-xchange-editor>
- [34] JASC SOFTWARE, INC. *Jasc Paint Shop Pro* [software]. 2000.
- [35] JETBRAINS S. R. O. *IntelliJ IDEA 2022.3.2 (Ultimate Edition)* [software]. [přístup 14. 4. 2023]. Dostupné z <https://www.jetbrains.com/idea/>
- [36] SONARSOURCE S.A, SWITZERLAND. *SonarQube* [software]. 2021. [přístup 12. 4. 2021]. Dostupné z <https://www.sonarqube.org/>
- [37] DOCKER, INC. *Docker Desktop* [software]. 2021. [přístup 13. 4. 2021]. Dostupné z <https://www.docker.com/products/docker-desktop>
- [38] ORACLE CORPORATION AND/OR ITS AFFILIATES. *OpenJDK* [software]. 2021. [přístup 12. 4. 2021]. Dostupné z <https://openjdk.java.net/>
- [39] ORACLE. *Java SE Development Kit 11*. [software]. 2021. [přístup 12. 4. 2021]. Dostupné z <https://www.oracle.com/java/technologies/javase-jdk11-downloads.html>
- [40] THE APACHE SOFTWARE FOUNDATION. *Apache Tomcat* [software]. 2021. [přístup 12. 4. 2021]. Dostupné z <https://tomcat.apache.org/>
- [41] THE APACHE SOFTWARE FOUNDATION. *PDFBox* [software]. 2023. [přístup 15. 4. 2023]. Dostupné z <https://pdfbox.apache.org/>
- [42] THE PGADMIN DEVELOPMENT TEAM. *pgAdmin* [software]. 2021. [přístup 13. 4. 2021]. Dostupné z <https://www.pgadmin.org/>
- [43] SZCZEPAN FABER AND FRIENDS. *Mockito* [software]. 2021. [přístup 16. 4. 2023]. Dostupné z <https://site.mockito.org/>
- [44] THE JUNIT TEAM. *JUnit* [software]. 2021. [přístup 16. 4. 2023]. Dostupné z <https://junit.org/junit5/>
- [45] MOUNTAINMINDS GMBH & CO. KG AND CONTRIBUTORS. *JaCoCo Java Code Coverage Library* [software]. 2023. [přístup 16. 4. 2023]. Dostupné z <https://jacoco.org/jacoco>
- [46] QUALITY OPEN SOFTWARE (QOS.CH). *LogBack* [software]. 2023. [přístup 16. 4. 2023]. Dostupné z <https://logback.qos.ch/>
- [47] NPM, INC. *NPM*. [software]. 2023. [přístup 15. 4. 2023] Dostupné z <https://www.npmjs.com>
- [48] OPENJS FOUNDATION AND NODE.JS CONTRIBUTORS. *Node.js*. [software]. 2023. [přístup 15. 4. 2023] Dostupné z <https://nodejs.org/en>
- [49] HACKERONE INC AND INDIVIDUAL CONTRIBUTORS. *React Date Picker*. [software]. 2023. [přístup 15. 4. 2023] Dostupné z <https://github.com/Hacker0x01/react-datepicker>
- [50] GOOGLE, INC. *Google Fonts*. [software]. 2023. [přístup 15. 4. 2023] Dostupné z <https://developers.google.com/fonts>

- [51] ALFREDSSON, Adam. *react-media-devices*. [software]. 2023. [přístup 15. 4. 2023] Dostupné z <https://github.com/adamalfredsson/react-media-devices>
- [52] GALLAGHER, Nicolas. *React Native for Web*. [software]. 2023. [přístup 15. 4. 2023] Dostupné z <https://github.com/necolas/react-native-web>
- [53] REMIX SOFTWARE. *React Router*. [software]. 2022. [přístup 15. 4. 2023] Dostupné z <https://github.com/remix-run/react-router>
- [54] OPENJS FOUNDATION AND JQUERY CONTRIBUTORS. *jQuery* [software]. 2021. [přístup 13. 4. 2021]. Dostupné z <https://jquery.com/>
- [55] GITLAB. *GitLab* [software]. 2021. [přístup 13. 4. 2021]. Dostupné z <https://about.gitlab.com/>

Přílohy

Licenční požadavky

Aplikace není zatížena žádnými licenčními požadavky, které by znemožňovaly její další úpravy nebo komerční použití. V této části práce uvádím použité nástroje a komponenty. Pokud je použití některého z artefaktů omezeno licencí, jsou v této části uvedeny podrobnosti.

A.1 Vývojové a pomocné nástroje

Vývoj probíhal na počítači s komerční licencí operačního systému Microsoft Windows 10 Professional a později Microsoft Windows 11 Professional a s použitím komerční licence Microsoft Office Professional 2016.

Při vývoji aplikace a při přípravě dokumentace byly použity komerční licence následujících nástrojů:

- Balsamiq Mockups 3 [31]
- Enterprise Architect 15.2 Corporate Edition [32]
- PDF-XChange Editor [33]
- Jasc Paint Shop Pro 7.01 [34]
- IntelliJ IDEA Ultimate 2020.3 a 2022.3.2 [35]

Licenční ujednání SonarQube Community Edition [36] použité v lokálně provozovaném kontejneru Docker [37] je k dispozici na adrese <http://www.gnu.org/licenses/lgpl-3.0.txt> a je uloženo na médiu ve složce `/docs/licence/SonarQube`. Licenční ujednání Dockeru je k dispozici na adrese <https://www.docker.com/legal/docker-terms-service> a je uloženo na médiu ve složce `/docs/licence/Docker`.

A.2 Aplikační server

Aplikace je napsána v jazyce Java 11. K dispozici jsou dvě verze – Open JDK [38], kterou je možné použít bezplatně, a Oracle JDK [39], kdy je potřeba mít ke komerčnímu provozu zakoupenou placenou licenci. Licenční ujednání varianty Open JDK je dostupné na adrese <https://openjdk.java.net/legal/gplv2+ce.html> a je uloženo na médiu ve složce `/docs/licence/OpenJDK`.

Licenční ujednání použitého serveru Tomcat [40] je k dispozici na adrese <https://www.apache.org/licenses/LICENSE-2.0.html> a je uloženo na médiu ve složce `/docs/licence`

/Tomcat. Na toto ujednání na stejné adrese se odkazuje licenční ujednání Frameworku Vaadin, jehož drobnou část používá Spring Framework pro práci se soubory typu Json (patrně se jedná o chybu sestavení knihoven Spring Framework, protože stejný kód je jejich součástí). Stejně licenční ujednání používá nástroj Maven [41] a nástroj PDFBox [15] použitý pro práci s PDF soubory. Licenční ujednání je k dispozici na adrese <https://maven.apache.org/ref/3.0/license.html> a je uloženo na médiu ve složce /docs/licence/Maven. Shodné licenční ujednání používá Spring Framework [10]. Licenční ujednání je k dispozici na adrese <https://github.com/spring-projects/spring-framework/blob/main/LICENSE.txt> a je uloženo na médiu ve složce /docs/licence/Spring. Spring Boot [12] a Spring Security [13] používají rovněž stejná licenční ujednání. Jsou k dispozici na adresách <https://github.com/spring-projects/spring-boot/blob/master/LICENSE.txt> a <https://github.com/spring-projects/spring-security/blob/master/LICENSE.txt> a jsou uložena na médiu ve složkách /docs/licence/SpringBoot a /docs/licence/SpringSecurity.

Licenční ujednání pro použití databáze PostgreSQL [8] a nástroje pgAdmin [42] je k dispozici na adrese <https://www.pgadmin.org/licence/#postgresql> a je uloženo na médiu ve složce /docs/licence/postgre.

Licenční ujednání LGPL-2.1 pro použití frameworku Hibernate ORM [14] je k dispozici na adrese <https://opensource.org/license/lgpl-2-1/> a je uloženo na médiu ve složce /docs/licence/hibernate.

Ke generování PDF souborů je použita knihovna openhtmltopdf [16]. Její licenční ujednání LGPL-3.0 je k dispozici na adrese <https://www.gnu.org/licenses/lgpl-3.0.html> a je uloženo na médiu ve složce /docs/licence/openhtmltopdf.

QR kódy jsou generovány a na aplikačním serveru rozpoznávány pomocí knihovny ZXing [17]. Její licenční ujednání založené na licenci Apache 2.0 je k dispozici na adrese <https://github.com/zxing/zxing/blob/master/LICENSE> a je uloženo na médiu ve složce /docs/licence/zxing.

Licenční ujednání nástroje pro testování Mockito [43] je k dispozici na adrese <https://github.com/mockito/mockito/wiki/License> a je uloženo na médiu ve složce /docs/licence/Mockito. Licenční ujednání nástroje pro testování JUnit [44] je k dispozici na adrese <https://junit.org/junit4/license.html> a je uloženo na médiu ve složce /docs/licence/JUnit. Licenční ujednání nástroje JaCoCo pro testování pokrytí aplikace je k dispozici na adrese <https://www.jacoco.org/jacoco/trunk/doc/license.html> je uloženo na médiu ve složce /docs/licence/JaCoCo.

Licenční ujednání logovací komponenty LogBack [46] je k dispozici na adrese <http://logback.qos.ch/license.html> a je uloženo na médiu ve složce /docs/licence/LogBack.

A.3 Komponenty uživatelského rozhraní

Licenční ujednání JavaScriptového package manageru NPM [47] („Node Package Manager“) je k dispozici na adrese <https://docs.npmjs.com/policies/npm-license> a je uloženo na médiu ve složce /docs/licence/npm.

Licenční ujednání Node.js [48] je k dispozici na adrese <https://github.com/nodejs/node/blob/main/LICENSE> a je uloženo na médiu ve složce /docs/licence/nodejs. Toto licenční ujednání je založeno na MIT licenci stejně jako licenční ujednání frameworku React [19]. To je k dispozici na adrese <https://github.com/facebook/react/blob/main/LICENSE> a je uloženo na médiu ve složce /docs/licence/react.

Licenční ujednání komponenty react-datepicker [49] je také založeno na licenci MIT a je k dispozici na adrese <https://github.com/Hacker0x01/react-datepicker/blob/HEAD/LICENSE> a je uloženo na médiu ve složce /docs/licence/datepicker.

Licenční ujednání komponenty material-design-icons [50] založené na licenci Apache 2.0 a je k dispozici na adrese <https://github.com/google/material-design-icons/blob/master/LICENSE> a je uloženo na médiu ve složce /docs/licence/material-design-icons.

Zatímco komponenta ZXing [17] využívaná aplikačním serverem má licenční ujednání založené na licenci Apache 2.0, webová komponenta react-zxing [20] se odvolává na licenci MIT, její znění ale neuvádí. Odkaz na MIT licenci je uložen na médiu ve složce `/docs/licence/zxing`.

Ani komponenta react-media-devices [51] licenční ujednání nezveřejňuje, pouze odkazuje na licenci MIT. Odkaz na MIT licenci je uložen na médiu ve složce `/docs/licence/react-media-devices`.

Licenční ujednání komponenty react-native-web [52] založené na MIT licenci je k dispozici na adrese <https://github.com/necolas/react-native-web/blob/master/LICENSE> a je uložené na médiu ve složce `/docs/licence/react-native-web`.

Licenční ujednání React Routeru [53] založené na MIT licenci je k dispozici na adrese <https://github.com/remix-run/react-router/blob/main/LICENSE.md> a je uložené na médiu ve složce `/docs/licence/reactrouter`.

Licenční ujednání frameworku Bootstrap [29] založené na MIT licenci je k dispozici na adrese <https://github.com/twbs/bootstrap/blob/v4.0.0/LICENSE> a je uložené na médiu ve složce `/docs/licence/Bootstrap`. Licenční ujednání jeho nadstavby react-bootstrap [30] je založené také na MIT licenci a je k dispozici na adrese <https://github.com/react-bootstrap/react-bootstrap/blob/master/LICENSE> a je uložené na médiu ve složce `/docs/licence/react-bootstrap`.

Licenční ujednání jQuery [54] se řídí standardní MIT licencí. Odkaz na licenci MIT je uveden na adrese <https://jquery.org/license/>, samotná licence je k dispozici na adrese <https://tldrlegal.com/license/mit-license>. Dokumenty jsou uloženy na médiu ve složce `/docs/licence/jquery`. Licenční ujednání notifikační komponenty toastr [27] je k dispozici na adrese <https://github.com/CodeSeven/toastr/blob/master/LICENSE> a je uložena na médiu ve složce `/docs/licence/toastr`. Nadstavba pro react ve formě komponenty react-toastify [28] má licenci typu MIT k dispozici na adrese <https://github.com/fkhadra/react-toastify/blob/main/LICENSE> a je uložena na médiu ve složce `/docs/licence/toastify`.

Grafy jsou generovány pomocí komponenty Chart.js [21] s licenci založenou na MIT licenci. Licenční ujednání je k dispozici na adrese <https://github.com/chartjs/Chart.js/blob/master/LICENSE.md> je uloženo na médiu ve složce `/docs/licence/chartjs`.

Vývojové prostředí

Zdrojové soubory aplikace jsou uloženy v níže uvedené adresářové struktuře:

frontend.....	soubory uživatelského rozhraní v Reactu
├─ build.....	kompilované soubory uživatelského rozhraní
├─ node_modules.....	systémové knihovny
├─ public.....	konfigurační soubory
├─ src.....	zdrojové soubory uživatelského rozhraní v Reactu
└─ src.....	
├─ main.....	
├─ java.....	zdrojové soubory aplikace v Javě
├─ resources.....	
├─ pages.....	zdrojové soubory editoru a grafů, pomocné soubory
├─ static.....	kompilované soubory uživatelského rozhraní
├─ application.properties.....	parametry aplikace, port, spojení na databázi
└─ logback-spring.xml.....	konfigurace logování
├─ .gitlab-ci.yml.....	konfigurační soubor continuous integration pro GitLab [55]
└─ pom.xml.....	konfigurační soubor nástroje Maven

Uživatelské prostředí v Reactu je naprogramováno pomocí souborů v adresáři `frontend/src`. Při kompilaci jsou výsledné soubory uloženy do adresáře `frontend/build`. Odtud je potřeba je zkopírovat do adresáře `src/resources/static`, kde jsou použity nástrojem Maven při sestavení spustitelného souboru ve formátu `.jar`. Před kopírováním je potřeba původní adresář `src/main/resources/static/static` smazat. V předchozí větě není opakování slova „static“ chyba, jedná se o dva různé adresáře se stejným názvem.

B.1 Instalace knihoven pro uživatelské prostředí React

Na počítač je potřeba nainstalovat Node.js a JavaScript package manager NPM. Při implementaci aplikace jsem používal verzi Node.js 16.16.0 ze 7. července 2022, která zahrnovala instalaci NPM ve verzi 8.11.

React a používané knihovny je potřeba nainstalovat spuštěním níže uvedeného příkazu v adresáři `frontend`:

```
PS T:\frontend> npm install react@18.2.0
```

JavaScript package manager NPM dohledá deklarované závislosti a nainstaluje balíčky uvedené v tabulce B.1.

■ **Tabulka B.1** Použité balíčky NPM

Balíček	verze
react	18.2.0
react-native-web	0.18.7
react-dom	18.2.0
react-router-dom	6.3.0
react-scripts	5.0.1
reactstrap	9.1.2
react-bootstrap	2.4.0
react-toastify	9.0.7
bootstrap	5.2.0
react-datepicker	4.8.0
material-design-icons	3.0.1
react-zxing	1.1.0
react-media-devices	1.1.0

Provedení skriptu může trvat i desítky minut. Při instalaci je vytvořen adresář `frontend/node_modules` a do něj je uloženo přes 130.000 souborů o celkové velikosti přesahující 300 MB. Počet souborů a jejich celková velikost bude při používání prostředí dále růst.

B.2 Lokální spuštění aplikace

Uživatelské rozhraní je možné spustit následujícím příkazem spuštěným v adresáři `frontend`:

```
PS T:\frontend> npm start
```

Aplikační rozhraní běží lokálně na portu 3000 (`http://localhost:3000`) a očekává webovou aplikaci v Javě dostupnou na portu 8086.

B.3 Kompilace uživatelského rozhraní v Reactu

Uživatelské rozhraní je možné zkompileovat následujícím příkazem spuštěným v adresáři `frontend`:

```
PS T:\frontend> npm run build
```

Zkompileované soubory je potřeba zkopírovat postupem popsáním v úvodu této kapitoly.

B.4 Kompilace aplikace v Javě

Na počítači musí být nainstalován Java JDK ve verzi podporující Java 11 nebo vyšší. Aplikaci je možné sestavit pomocí nástroje Maven následujícím příkazem:

```
PS T:\> mvn clean package -Dmaven.test.skip=true
```

Pro provozní školní prostředí je potřeba aplikaci kompilovat tak, aby byl http server dostupný na portu 80, zatímco pro lokální spuštění aplikace je potřeba nastavit port 8086. Nastavení portu se provádí v souboru `src/main/resources/application.properties`.

Provozní prostředí

Aplikace je nainstalována na školní server na adrese `assignments.kam.in.fit.cvut.cz` dostupný pouze z vnitřní sítě fakulty. Webová aplikace je na serveru dostupná na portu 80 s využitím protokolu HTTP, který je pomocí školní reverzní proxy mapován veřejně dostupnou webovou adresu `https://assignments.kam.fit.cvut.cz`, na port 443 a protokol HTTPS. Na reverzní proxy jsou povolené příchozí požadavky velké až 100 MB pro potřeby uploadu souboru. Tento limit není využit, uživatelské rozhraní rozděluje uploadované soubory na části o velikosti nejvýše 8 MB.

Operačním systémem serveru je Linux Ubuntu 22.04.2 LTS. Na serveru je nainstalována databáze PostgreSQL 14.7 a běhové prostředí Java 11, které využívá aplikace distribuovaná ve formátu souboru `.jar`. Uživatelské rozhraní napsané v Reactu je uloženo v souboru `.jar` a na serveru tedy není potřeba instalovat Node.js a uživatelské rozhraní tedy nevyžaduje běh dalšího samostatného procesu.

Server má alokované 4 GB RAM a tato velikost je z mého pohledu minimální vhodná velikost paměti serveru. Server má přiřazené 4 virtuální CPU, což je z mého pohledu více než dostačující, a 64 GB diskového prostoru (a po nainstalování součástí uvedených níže a načtení a uložení studentů, učitelů a předmětů za zimní a letní semestr školního roku 2022/23 do databáze je cca 56 GB prostoru k dispozici).

Aplikace pro logování používá framework LogBack, který je součástí SpringBoot frameworku.

C.1 Instalace Javy

Před instalací je potřeba provést aktualizaci souborů s informacemi o dostupných balíčcích a jejich verzí pomocí následujícího příkazu:

```
root@assignments:~# sudo apt-get update
```

Po aktualizaci informací je možné na server nainstalovat Javu 11:

```
root@assignments:~# sudo apt install openjdk-11-jdk
```

Tím je instalace Javy dokončena. Port 80 byl povolen administrátorem serveru při jeho instalaci.

C.2 Instalace PostgreSQL

Nejprve je potřeba nainstalovat a spustit databázi následující sekvencí příkazů:

```
root@assignments:~# sudo apt-get install postgresql postgresql-contrib
root@assignments:~# sudo update-rc.d postgresql enable
root@assignments:~# sudo service postgresql start
```

Následně je potřeba změnit heslo databázového uživatele postgres:

```
root@assignments:~# su - postgres
postgres@assignments:~$ psql
postgres=# ALTER USER postgres PASSWORD 'ZDE-JE-HESLO';
```

Heslo není v tomto veřejném dokumentu uloženo, je k dispozici na serveru v souboru `/root/postgreSQL.password.txt`.

Pomocí příkazu níže lze vypsat umístění konfiguračního souboru:

```
postgres=# show config_file;
postgres=# exit
postgres@assignments:~$ exit
root@assignments:~#
```

Umístění konfiguračního souboru je `/etc/postgresql/14/main/postgresql.conf`.

Nyní povolíme přístup k databázi před port 5432 (začátek řádek zkrácen):

```
root@a...nts:~# sudo firewall-cmd --zone=public --add-port=5432/tcp --permanent
root@a...nts:~# sudo ufw allow 5432
```

Do konfiguračního souboru na adrese uvedené výše přidáme novou řádku, abychom se k databázi mohli připojit odkudkoliv:

```
listen_addresses = '*'
```

Do souboru `/etc/postgresql/14/main/pg_hba.conf` přidáme následující dvě řádky pod již existující nadpis uvozený znakem `#`:

```
# TYPE DATABASE USER ADDRESS METHOD
host all all 0.0.0.0/0 md5
host all all ::/0 md5
```

Nyní databázi restartujeme:

```
root@assignments:~# sudo systemctl restart postgresql
```

Nainstalujeme `net-tools` a ověříme, že je databáze na portu dostupná:

```
root@assignments:~# sudo apt-install net-tools
root@assignments:~# netstat -nlp | grep 5432
tcp        0      0 0.0.0.0:5432      0.0.0.0:*        LISTEN    924/postgres
tcp6       0      0 :::5432          :::*             LISTEN    924/postgres
root@assignments:~#
```

Číslo 924 odpovídá číslu procesu běžící databáze a může být odlišné.

Nyní se k databázi můžeme přihlásit pomocí nástroje `pgAdmin` a vytvořit uživatele „assignments“ (v terminologii databáze PostgreSQL se jedná o vytvoření role „assignments“ s právem přihlášení) a databázi „assignments“ vlastněnou uživatelem „assignments“. Údaje o databázi „assignments“ a uživateli „assignments“ jsou uvedeny v konfiguraci aplikace. Heslo uživatele „assignments“ není v tomto veřejném dokumentu uloženo, je k dispozici na serveru v souboru `/root/postgreSQL.password.txt`.

Skript pro vytvoření databázových tabulek a jejich naplnění iniciálními daty je k dispozici na médiu, které tvoří přílohu tohoto dokumentu.

C.3 Spuštění aplikace

Spustitelný soubor je uložen v adresáři `/pisemky`, ve kterém aplikace také běží.

Aplikaci lze spustit následujícím příkazem:

```
java -jar -Xmx2500m pisemky.jar > log.txt &
```

Parametr `-Xmx2500m` nastavuje maximální velikost aplikačního heapu na 2500 MB. Bez tohoto nastavení by velikost aplikačního heapu byla cca 1 GB stanovená jako minimum z 1 GB a jedné čtvrtiny velikosti operační paměti serveru.

Otestoval jsem, že v této konfiguraci aplikace zpracovala PDF soubor s 3200 oskenovanými stránkami o velikosti 1,63 GB.

Při spuštění aplikace lze také vložit do příkazového řádku parametry pro přístup k databázi:

```
java -Dspring.datasource.url=<URL>
-Dspring.datasource.driverClassName=org.postgresql.Driver
-Dspring.datasource.username=<USER> -Dspring.datasource.password=<PASS>
-jar pisemky.jar
```

kde

- `<URL>` je adresa databáze,
- `<USER>` je identifikace uživatele databáze a
- `<PASS>` je přístupové heslo uživatele databáze.

Server může být z provozních důvodů potřeba restartovat. Je proto vhodné nastavit server tak, aby se aplikace po jeho restartu automaticky spustila.

V adresáři `/etc/systemd/system` je potřeba vytvořit soubor `assignments.service` s následujícím obsahem:

```
[Unit]
Description=Assignments
After=postgresql.service
Requires=postgresql.service

[Service]
Type=simple
KillMode=process
User=root
Group=root
ExecStart=/pisemky/run.sh
StandardOutput=journal+console

[Install]
WantedBy=multi-user.target
root@assignments:/etc/systemd/system#
```

Tento soubor je nutné zaregistrovat následujícím příkazem:

```
root@assignments:~# systemctl enable assignments.service
```

Obsah souboru `/pisemky/run.sh` je následující:

```
#!/bin/sh
cd /pisemky
java -jar -Xmx2500m pisemky.jar > log.txt &
```

C.4 Struktura adresářů

V adresáři `/piskemky` jsou další podadresáře využívané aplikací uvedené v tabulce C.1.

■ **Tabulka C.1** Využívané adresáře

Adresář	Obsah
drawings	Soubory s vrstvami oprav vytvořené pomocí editoru. Soubory mají název ve tvaru <code>id.txt</code> , kde <code>id</code> je primární klíč záznamu v tabulce <code>picture_drawing</code> .
logs	Soubory vytvořené logovacím frameworkem LogBack.
qrfiles	Zkušební archy s QR kódy. Soubory mají název ve tvaru <code>id.pdf</code> , kde <code>id</code> je primární klíč záznamu v tabulce <code>qr_code_file</code> .
pictures	Obrázky vygenerované z vložených PDF souborů a obrázků. Soubory mají název ve tvaru <code>id.jpg</code> a <code>id_small.jpg</code> , kde <code>id</code> je primární klíč záznamu v tabulce <code>submission_picture</code> . Obrázek s rozlišením „small“ obsahuje miniaturu pro potřeby náhledu.
submissions	Odevzdané soubory. Soubory mají název ve tvaru <code>id.data</code> , kde <code>id</code> je primární klíč záznamu v tabulce <code>submission_file</code> .

Soubory jsou v jednotlivých adresářích `drawings`, `pictures` a `submissions` uloženy v adresářové struktuře podle roku, měsíce a dne vytvoření záznamu:

```

adresář ..... drawings, pictures, qrfiles, submissions
├── rok ..... rok vytvoření záznamu
│   ├── měsíc ..... měsíc vytvoření záznamu
│   │   ├── den ..... den vytvoření záznamu
│   │   │   └── datové soubory..vrstvy oprav, zkušební archy, obrázky, miniatury, odevzdané
│   │   │       soubory

```

Přehled požadavků

V této kapitole jsou popsány požadavky zadavatele na výsledné softwarové řešení. Požadavky jsou rozděleny na funkční a nefunkční a dále kategorizovány podle tabulky D.1. Text sloužil jako podklad pro diskusi se zadavatelem a nebyl aktualizován podle skutečně implementované aplikace.

■ **Tabulka D.1** Kategorie požadavků

Kategorie požadavku	Typ požadavku
A	Požadavek musí být splněn, jeho splnění je kritériem úspěchu
B	Požadavek je vhodné splnit, pokud jeho splnění nebude možné, je potřeba mít náhradní postup
C	Ostatní požadavky

D.1 Typy uživatelů

Aplikaci budou používat následující typy uživatelů:

- **Studenti**
- **Učitelé**
- **Správci**

Aplikace bude pro přihlášení učitelů, studentů a správců využívat mechanismus přihlášení školním uživatelským identifikátorem a heslem (SSO). Rozlišení typu přístupu bude provedeno pomocí rolí (<https://ist.cvut.cz/nase-sluzby/kategorie-sluzeb/systemy-osoby-pristupy-prava/role/>). Jedna osoba (přihlášený uživatel) tedy může mít všechny uvedené role současně.

Během vývoje a testování a pro potřeby naléhavého přístupu do aplikace bude aplikace dále umožňovat přihlášení uživatele jeho lokálně definovaným uživatelským přístupem.

D.2 Funkční požadavky

Na aplikaci jsou kladeny níže uvedené funkční požadavky.

D.2.1 Funkční požadavky na uživatelské rozhraní

Funkční požadavky kladené na uživatelské rozhraní jsou uvedeny v tabulce D.2.

■ **Tabulka D.2** Funkční požadavky na uživatelské rozhraní

Číslo požadavku	Kategorie požadavku	Popis požadavku
F1.1	A	Uživatelské rozhraní pro studenty bude umožňovat odevzdat písemnou práci ve formátu PDF, PNG nebo JPG
F1.2	A	Uživatelské rozhraní pro učitele bude umožňovat opravu práce ve webovém prohlížeči způsobem umožňujícím vložit nebo pomocí stylusu připsat poznámky k jednotlivým místům práce
F1.3	A	Uživatelské rozhraní pro studenty bude umožňovat náhled na opravenou práci s poznámkami a bodovým hodnocením učitele
F1.4	A	Uživatelské rozhraní bude provedeno česky a anglicky
F1.4	B	Texty uživatelského rozhraní budou definované v aplikaci a uživatel s rolí správce je bude moci změnit
F1.5	C	Zdrojový kód editoru pro opravování prací bude uložen v aplikaci a bude možné jej změnit; původní i nový kód bude možné vyzkoušet na vzorové práci, aniž by byla opravována skutečná studentská práce

D.2.2 Funkční požadavky na serverovou část

D.2.2.1 Zpracovávaná data

Aplikace bude zpracovávat data uvedená v tabulce D.3

■ **Tabulka D.3** Zpracovávaná data

Číslo požadavku	Kategorie požadavku	Popis zpracovávaných dat
F2.1	A	Data o uživatelích
F2.2	A	Data o semestrech
F2.3	A	Data o předmětech vyučovaných v jednom semestru
F2.4	A	Data o písemných pracích a jejich částech odevzdávaných v rámci předmětu vyučovaném v daném semestru
F2.5	A	Přiřazení studentů k jednotlivým předmětům vyučovaným v daném semestru
F2.6	A	Přiřazení učitelů k jednotlivým předmětům vyučovaným v daném semestru
F2.7	A	Záznamy o provedených opravách písemných prací
F2.8	A	Výsledky bodového ohodnocení jednotlivých částí písemných prací

D.2.2.2 Podporované operace s daty

Aplikace bude podporovat operace prováděné s daty uvedené v tabulce D.4:

■ **Tabulka D.4** Operace s daty

Číslo požadavku	Kategorie požadavku	Popis operace s daty
F3.1	A	Získání dat o semestrech (integrace)
F3.2	A	Získání dat o předmětech (integrace)
F3.3	A	Získání dat o písemných pracích (integrace)
F3.4	A	Doplnění dat o písemných pracích a jejich částech pro potřeby provádění jejich oprav
F3.5	A	Získání dat o studentech (integrace)
F3.6	A	Získání dat o učitelích (integrace)
F3.7	A	Získání dat o vazbě studentů na předměty v semestru (integrace)
F3.8	A	Získání dat o vazbě učitelů na předměty v semestru (integrace)
F3.9	A	Rozdělení oskenovaného dokumentu na jednotlivé písemné práce, rozsah podpory procesu učitelem bude upřesněn v průběhu realizace
F3.10	A	Přiřazení oskenovaného dokumentu konkrétnímu studentovi a písemné práci jako evidované události, rozsah podpory procesu učitelem bude upřesněn v průběhu realizace
F3.11	A	Rozdělení studentské písemné práce na části, které mají samostatné bodové ohodnocení; rozsah podpory procesu učitelem bude upřesněn v průběhu realizace
F3.12	A	Zpracování opravy písemné práce a výsledků bodového ohodnocení jednotlivých částí písemné práce
F3.13	A	Odeslání výsledků bodového ohodnocení jednotlivých částí práce do aplikace Grades (integrace)

Operace označené textem „(integrace)“ budou zajištěny prostřednictvím integrace s externími systémy.

D.2.2.3 Další operace

Aplikace bude podporovat další operace uvedené v tabulce D.5:

■ **Tabulka D.5** Další požadavky

Číslo požadavku	Kategorie požadavku	Popis operace s daty
F4.1	B	Odeslání e-mailového upozornění studentovi o opravené práci
F4.2	A	Stanovení, zda písemnou práci může odevzdat student
F4.3	C	Stanovení, zda písemnou práci lze odevzdat najednou, po bodově hodnocených částech, nebo oběma způsoby
F4.4	B	Stanovení nejzazšího termínu pro odevzdání písemné práce studentem, nastavení možnosti dalšího odevzdání písemné práce studentem
F4.4	C	Stanovení, zda v případě existence více opravených verzí práce má student k dispozici jen poslední opravenou verzi nebo všechny opravené verze

D.3 Nefunkční požadavky

Na aplikaci jsou kladeny níže uvedené nefunkční požadavky.

D.3.1 Nefunkční požadavky na uživatelské rozhraní

Na uživatelské rozhraní jsou kladeny nefunkční požadavky uvedené v tabulce D.6.

■ **Tabulka D.6** Nefunkční požadavky na uživatelské rozhraní

Číslo požadavku	Kategorie požadavku	Popis požadavku
N1.1	A	Uživatelským rozhraním bude internetový prohlížeč
N1.2	A	Aplikace bude podporovat práci uživatele v Microsoft Windows, Linuxu (Ubuntu desktop) a na iPadu
N1.3	A	Uživatelské rozhraní pro studenty bude podporovat v Microsoft Windows prohlížeče Edge, Chrome a Firefox; na Linuxu Chrome a Firefox a na iPadu Chrome a Safari
N1.4	A	Uživatelské rozhraní pro učitele a správce bude podporovat v Microsoft Windows prohlížeče Chrome nebo Firefox; na Linuxu Chrome nebo Firefox a na iPadu Chrome nebo Safari

Návrh specifikace

V průběhu analýzy požadavků vznikl návrh specifikace aplikace jako výstup diskuse se zadavatelem. Hlavním cílem dokumentu bylo požadavky zadavatele shrnout a vzájemně se ujistit o jejich správném porozumění, definovat hlavní rysy aplikace a určit očekávané výstupy práce.

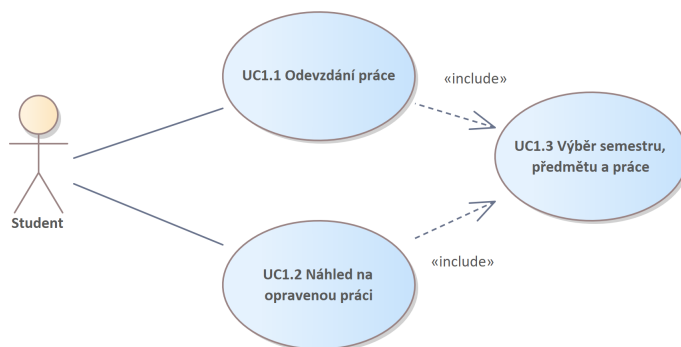
Text návrhu specifikace byl přípravou aplikace překonán a s výjimkou jazykových změn a formátování do nové šablony nebyl aktualizován. Přestože výsledná aplikace z návrhu specifikace vychází a v hlavních rysech jej respektuje, některé dílčí funkčnosti jsou implementovány odlišně.

E.1 Funkční požadavky

E.1.1 Případy užití

E.1.1.1 Případy užití studenta

Klíčové případy užití studenta jsou uvedeny na obrázku E.1.



■ Obrázek E.1 Případy užití studenta

Jednotlivé případy užití jsou popsány dále.

E.1.1.1.1 UC1.1 Odevzdání práce

A Web Page

Submissions Change Language Logout

Assignments

Make a Submission

Select assignment

see UC13

Assignment name: Homework 2

Select and submit files for the below listed 3 parts

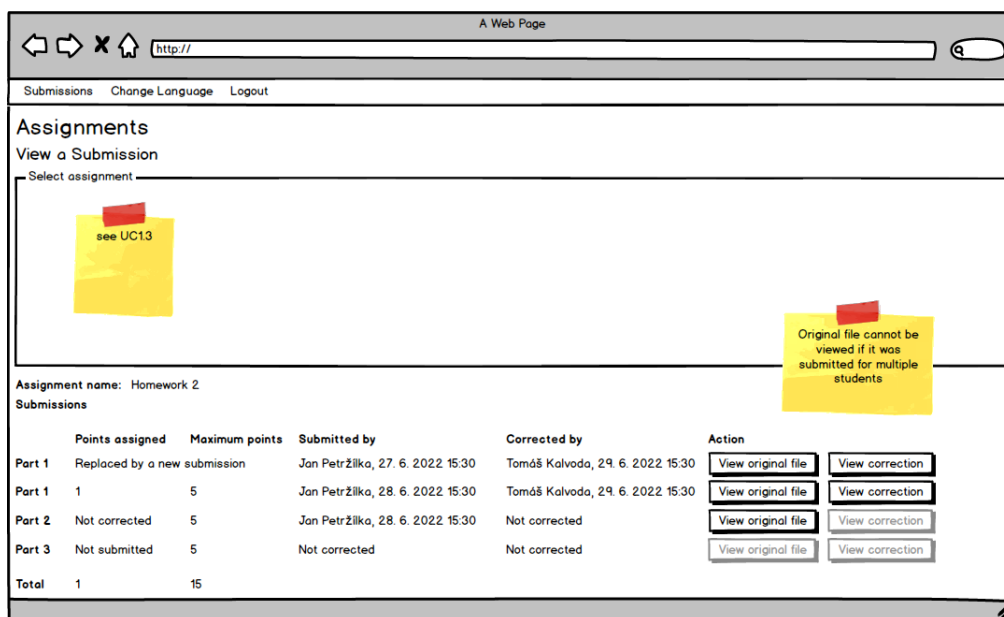
	Uploaded files	Actions
Part 1	<ul style="list-style-type: none">MyFile1.pngMyFile2.jpgMyFile3.pdf	<input type="button" value="Select and upload files"/> <input type="button" value="View"/> <input type="button" value="Delete"/>
Part 2		<input type="button" value="Select and upload files"/> <input type="button" value="View"/> <input type="button" value="Delete"/>
Part 3		<input type="button" value="Select and upload files"/> <input type="button" value="View"/> <input type="button" value="Delete"/>

■ **Obrázek E.2** Formulář odevzdání práce

Student vybere písemnou práci podle případu užití UC1.3.

Student vybere a odešle soubor nebo soubory s písemnou prací pomocí formuláře na obrázku E.2 podle toho, zda je pro danou písemnou práci určeno, že student má odevzdat práci s rozlišením na úlohy nebo bez tohoto rozlišení.

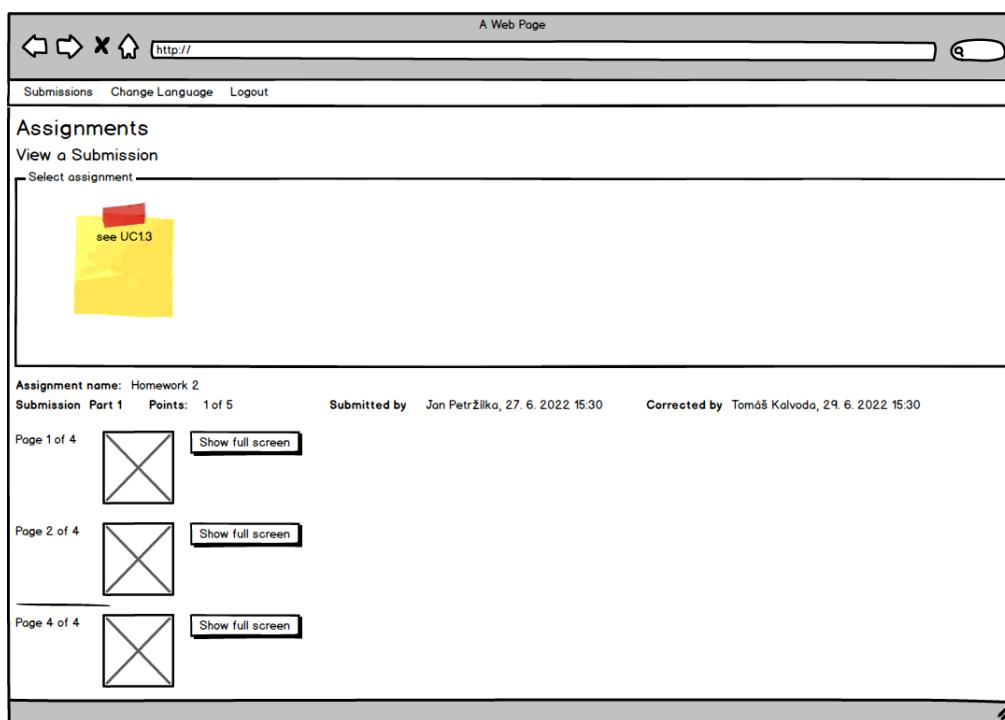
E.1.1.1.2 UC1.2 Náhled na opravenou práci



■ Obrázek E.3 Formulář náhledu na práci

Student vybere písemnou práci podle případu užití UC1.3.

Aplikace na formuláři E.3 zobrazí jednotlivé části práce spolu s uděleným bodovým ohodnocením a informacemi o tom, kdo a kdy danou část práce opravil. Student může zobrazit opravenou část práce s poznámkami učitele, který práci opravil, tak, jak je zachyceno na obrázku E.4.



■ Obrázek E.4 Zobrazení opravené práce

E.1.1.1.3 UC1.3 Výběr semestru, předmětu a práce studentem

Select assignment

Select semester: B212 - B212 Letní 2021/2022

Select subject: B212 - NI-VSM Selected statistical Methods

Assignment	Deadline	State	Action
Homework 1	March 10, 2022	Graded	Submit View
Homework 2		Graded - open for resubmission	Submit View
Homework 3		Submitted	Submit View
Homework 4		Not submitted	Submit View

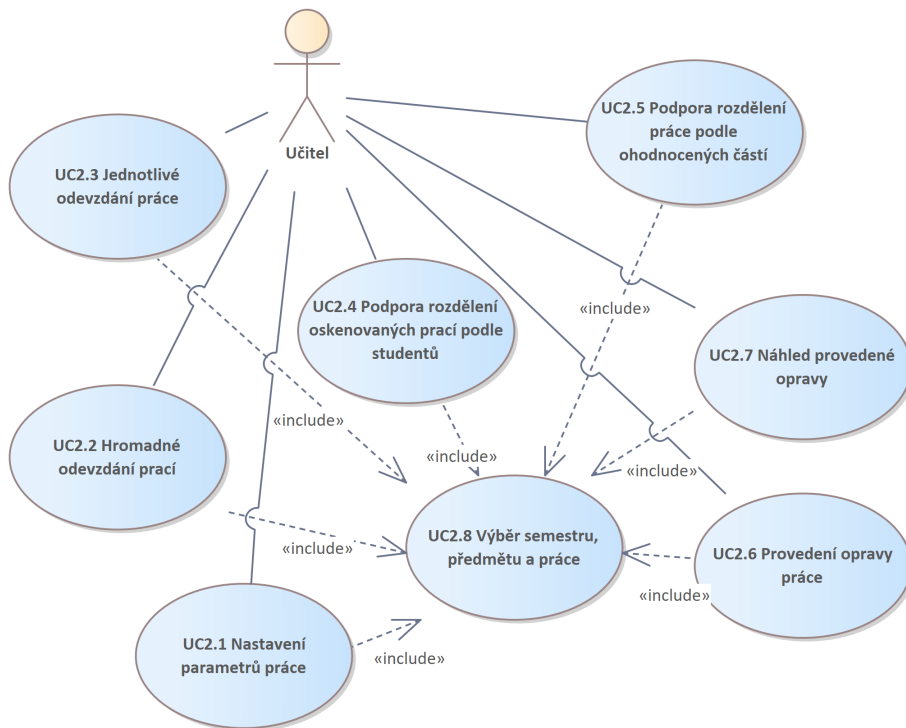
■ Obrázek E.5 Formulář výběru semestru, předmětu a práce studentem

Student vybere semestr a předmět. Aplikace zobrazí seznam písemných prací pro vybraný semestr a předmět tak, jak je znázorněno na obrázku E.5.

U jednotlivých písemných prací je zobrazen jejich stav. Student má možnost danou písemnou práci vybrat a pokračovat v případě užití UC1.1 nebo UC1.2.

E.1.1.2 Případy užití učitele

Klíčové případy užití učitele jsou uvedeny na obrázku E.6.



■ **Obrázek E.6** Případy užití učitele

E.1.1.2.1 UC2.1 Nastavení parametrů práce

■ Obrázek E.7 Formulář nastavení parametrů práce

Učitel vybere písemnou práci podle případu užití UC2.8.
Učitel nastaví parametry práce v rozsahu podle tabulky E.1.

■ Tabulka E.1 Parametry práce

Parametr	Popis
Deadline	Poslední den, kdy student může práci odevzdat
Parts	Počet částí práce
Submission by student	Zda může práci odevzdat sám student
Re-submission by student	Zda student může práci odevzdat opakovaně
Submission type	Zda student musí práci odevzdat v souborech po úlohách, nebo zda ji může práci odevzdat bez rozdělení na úlohy nebo zda si může vybrat
Student is shown	Zda má student přístup ke všem opraveným odevzdaným souborům nebo jen k těm posledním
Name	Název části práce
Maximum points	Maximální počet bodů části práce

E.1.1.2.2 UC2.2 Hromadné odevzdání prací

A Web Page

Submissions Change Language Logout

Assignments

Multiple Submission

Select assignment

see UC2.8

Assignment name: Homework 2

Select and submit files for 3 parts

Uploaded files

Part 1

- MyFile1.png
- MyFile2.jpg
- MyFile3.pdf

Actions

Select and upload files View Delete

Part 2

Select and upload files View Delete

Part 3

Select and upload files View Delete

Submit the files uploaded above

Select and submit files for the assignment

Uploaded files

- MyFile1.png
- MyFile2.jpg
- MyFile3.pdf

Select and upload files View Delete

Actions

Submit the files uploaded above

■ **Obrázek E.8** Formulář hromadného odevzdání prací

Učitel vybere písemnou práci podle případu užití UC2.8.

Učitel odevzdá práce za více studentů pomocí formuláře na obrázku E.8 bez ohledu na nastavení způsobu odevzdávání prací pro studenty.

E.1.1.2.3 UC2.3 Jednotlivé odevzdání práce

The screenshot shows a web browser window titled "A Web Page" with a URL bar containing "http://". The page content includes a navigation menu with "Submissions", "Change Language", and "Logout". The main heading is "Assignments" followed by "Single Submission". A "Select assignment" dropdown menu is present, with a yellow sticky note icon and the text "see UC2.8" overlaid on it. Below this, the "Assignment name" is "Homework 2" and the "Student" is "Jan Petržilka (petrzja5)".

There are two main sections for file uploads:

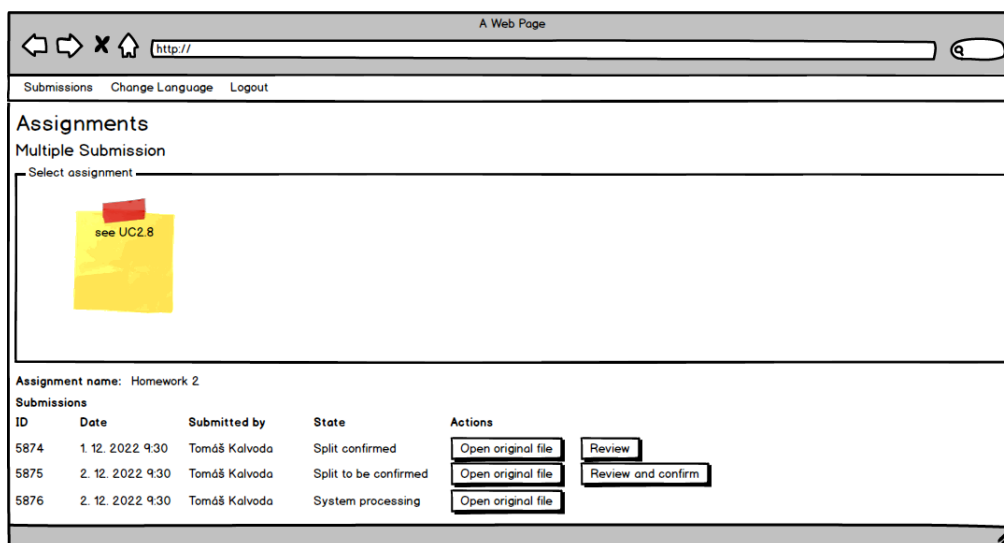
- Left Section: "Select and submit files for 3 parts"**
 - Part 1:** "Uploaded files" list contains "MyFile1.png", "MyFile2.jpg", and "MyFile3.pdf". "Actions" include "Select and upload files", "View", and "Delete".
 - Part 2:** "Uploaded files" list is empty. "Actions" include "Select and upload files", "View", and "Delete".
 - Part 3:** "Uploaded files" list is empty. "Actions" include "Select and upload files", "View", and "Delete".
 - A "Submit the files uploaded above" button is located at the bottom of this section.
- Right Section: "Select and submit files for the assignment"**
 - Uploaded files:** list contains "MyFile1.png", "MyFile2.jpg", and "MyFile3.pdf". "Actions" include "Select and upload files", "View", and "Delete".
 - Actions:** includes a "Submit the files uploaded above" button.

■ **Obrázek E.9** Formulář jednotlivého odevzdání práce

Učitel vybere písemnou práci podle případu užití UC2.8.

Učitel odevzdá práci za vybraného studenta pomocí formuláře na obrázku E.9; bez ohledu na nastavení způsobu odevzdávání prací pro studenty si může vybrat, zda práce odevzdá po částech nebo najednou.

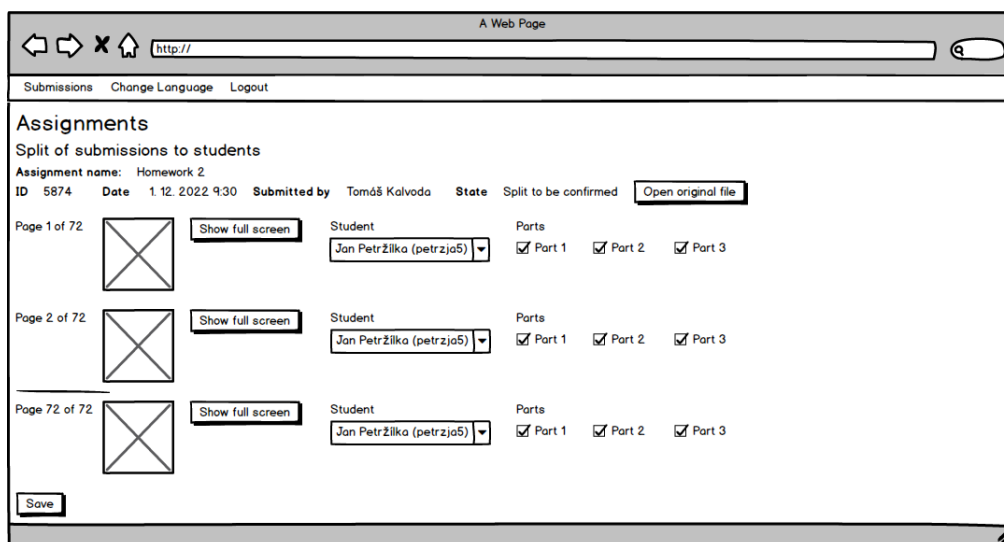
E.1.1.2.4 UC2.4 Podpora rozdělení oskenovaných prací podle studentů



■ Obrázek E.10 Přehled oskenovaných prací

Učitel vybere písemnou práci podle případu užití UC2.8.

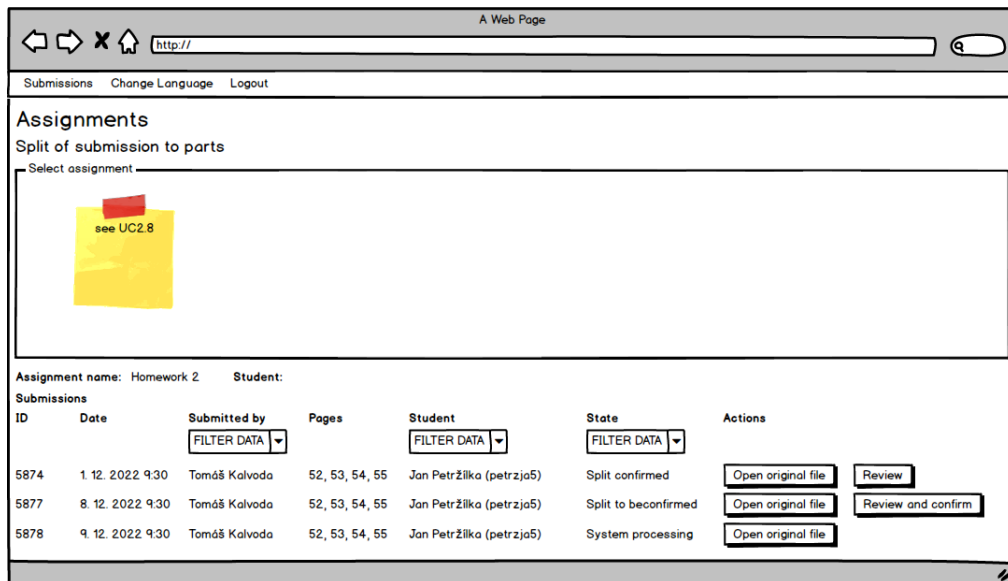
Aplikace zobrazí seznam vložených souborů spolu se stavem zpracování tak, jak je zachyceno na obrázku E.10. Dokud aplikace vložený soubor zpracovává, není možné s ním pracovat. Jakmile aplikace soubor zpracuje, je možné výsledek zpracování otevřít a potvrdit. Tím se obsah zpracovávaného souboru stává dostupným pro rozdělení jednotlivých částí práce k samostatně hodnoceným částem. Výsledek zpracování, který byl již potvrzen, lze otevřít a revidovat.



■ Obrázek E.11 Přiřazení stránek studentům

Při otevření souboru aplikace zobrazí jeho jednotlivé stránky tak, jak je znázorněno na obrázku E.11. Učitel může nastavit nebo změnit přiřazení stránky ke studentovi a eventuálně také vyznačit příslušnost stránky k jednotlivé samostatně hodnocené části písemné práce.

E.1.1.2.5 UC2.5 Podpora rozdělení práce podle ohodnocených částí

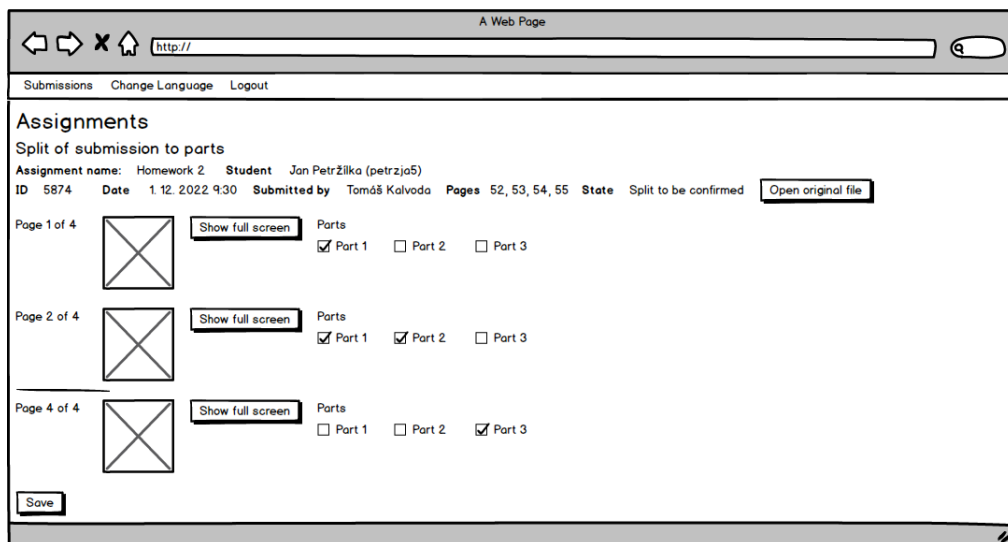


■ Obrázek E.12 Přehled vložených prací

Učitel vybere písemnou práci podle případu užití UC2.8.

Učitel použije pole pro filtrování zobrazených položek.

Aplikace zobrazí seznam vložených prací spolu se stavem zpracování tak, jak je zachyceno na obrázku E.12. Dokud aplikace vložený soubor zpracovává, není možné s ním pracovat. Jakmile aplikace soubor zpracuje, je možné výsledek zpracování otevřít a potvrdit. Tím se obsah zpracovávaného souboru stává dostupným pro opravování. Výsledek zpracování, který byl již potvrzen, lze otevřít a revidovat.

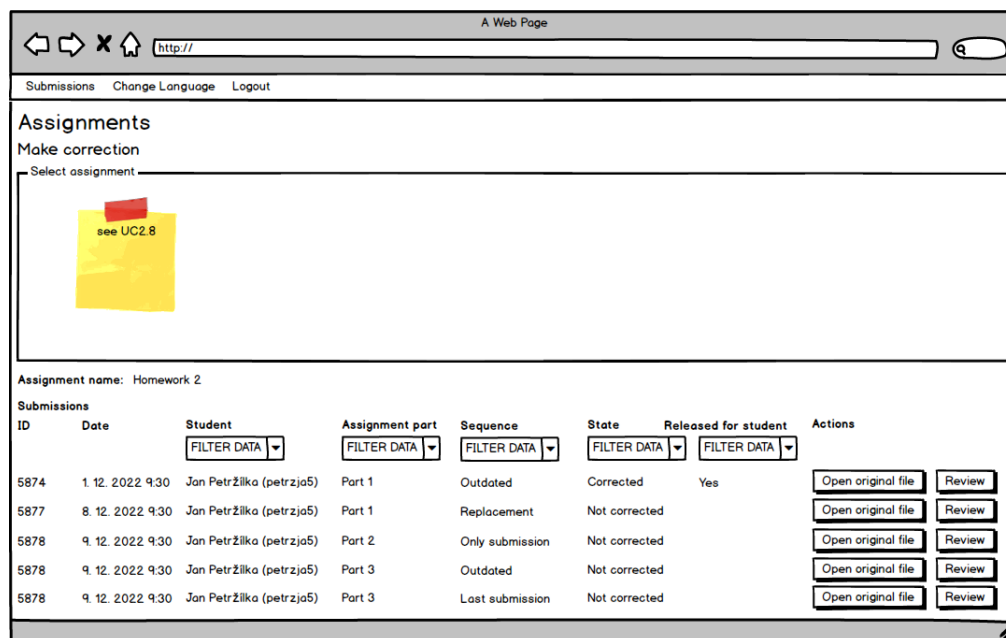


■ Obrázek E.13 Rozdělení práce podle hodnocených částí

Při otevření práce aplikace zobrazí jeho jednotlivé stránky tak, jak je znázorněno na obrázku

E.13. Učitel může nastavit nebo změnit rozdělení stránek práce podle jednotlivých hodnocených částí práce.

E.1.1.2.6 UC2.6 Provedení opravy práce



■ **Obrázek E.14** Přehled částí k opravě

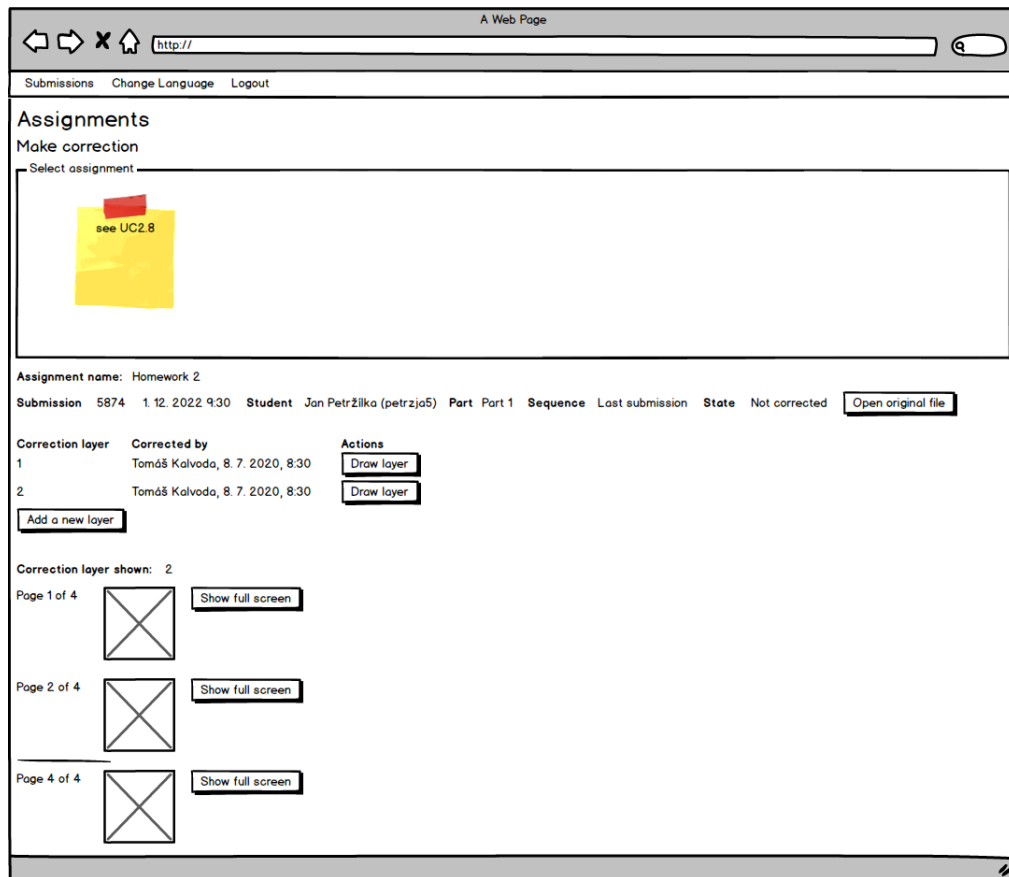
Učitel vybere písemnou práci podle případu užití UC2.8.

Učitel použije pole pro filtrování zobrazených položek.

Aplikace zobrazí seznam částí písemných prací, které jsou již opravené nebo připravené k opravě tak, jak je zachyceno na obrázku E.14. Pokud pro daného studenta existuje více odevzdaných souborů, jsou seřazeny podle data vzestupně a tato skutečnost je u nich vyznačena spolu se stavem zpracování.

Jednotlivé provedené a uložené opravy tvoří samostatné vrstvy, se kterými lze pracovat odděleně.

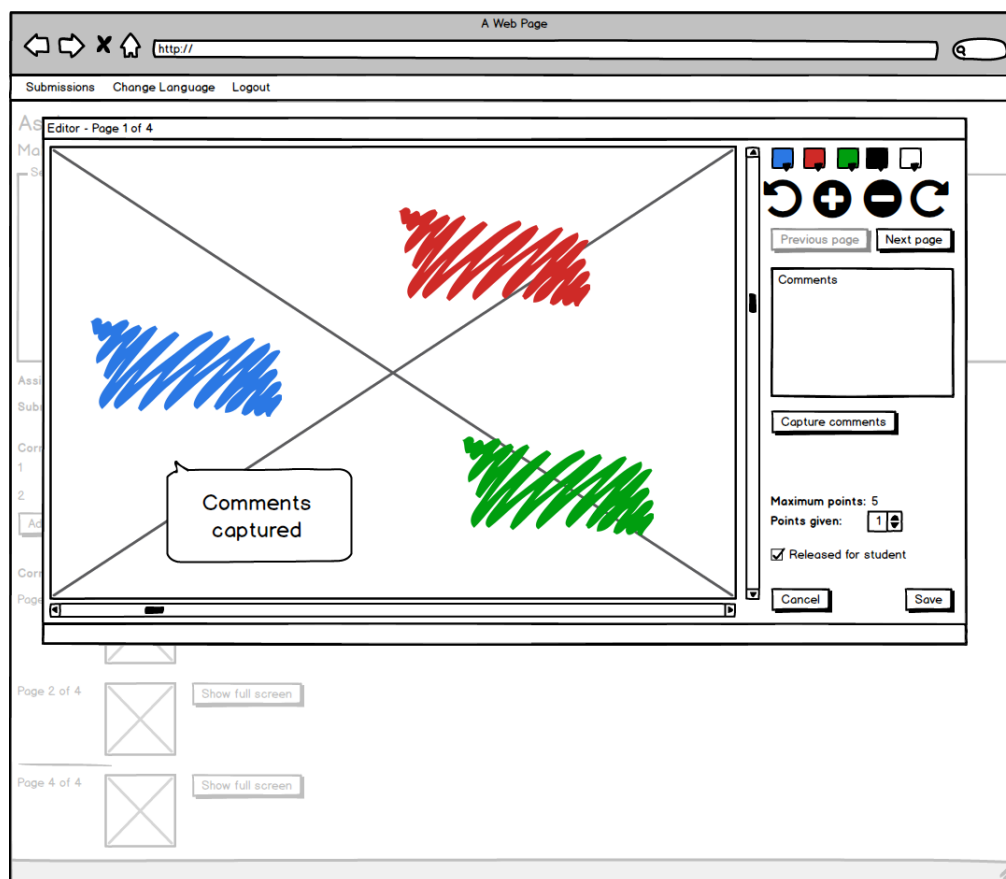
Učitel vybere konkrétní část k opravě. Aplikace zobrazí seznam existujících vrstev s opravami, učitel může zobrazit obsah předchozích vrstev nebo založit novou vrstvu a část písemné práce opravit. Formulář, který k tomu použije, je zachycen na obrázku E.15.



■ **Obrázek E.15** Zobrazení vrstev opravy

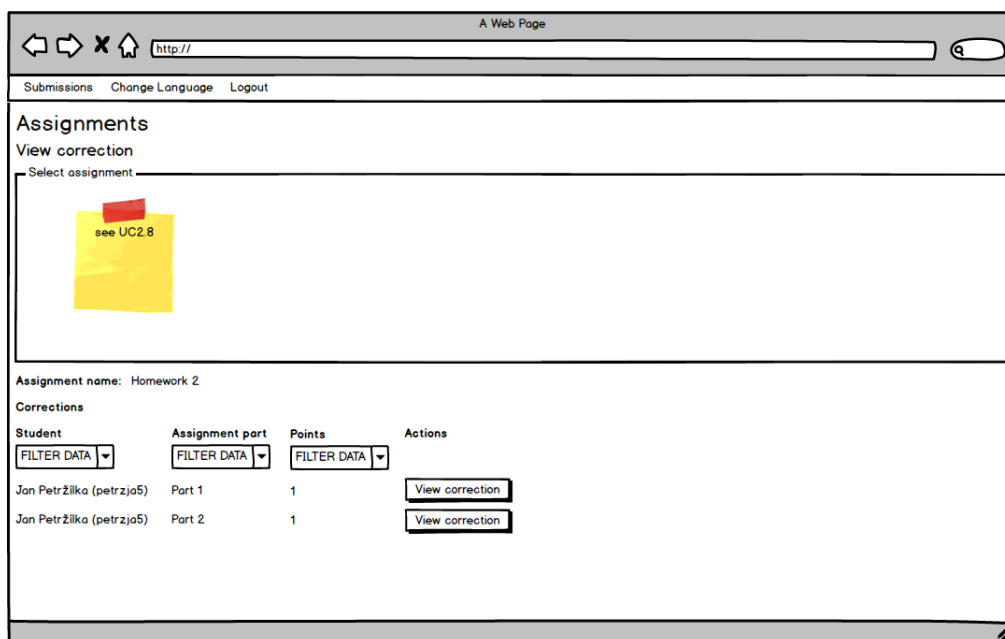
Po výběru konkrétní stránky aplikace otevře editor na obrázku E.16, kde budou zachyceny předchozí vrstvy opravy až do vybrané vrstvy. V případě nové vrstvy bude učitel moci opravu provést nebo změnit a výsledek opravy uložit.

Editor umožňuje do opravované stránky vložit poznámky nebo pomocí myši nebo stylusu na tabletu nebo dotykové obrazovce poznámky napsat. Při použití bílé barvy dojde ke smazání poznámek v okolí bodu, kde byla bílá barva použita. Je důležité, aby tyto poznámky byly smazány a nikoliv pouze překryty bílou barvou, protože jinak by v dalších vrstvách stále byly dostupné.



■ Obrázek E.16 Editor pro provedení opravy

E.1.1.2.7 UC2.7 Náhled provedené opravy



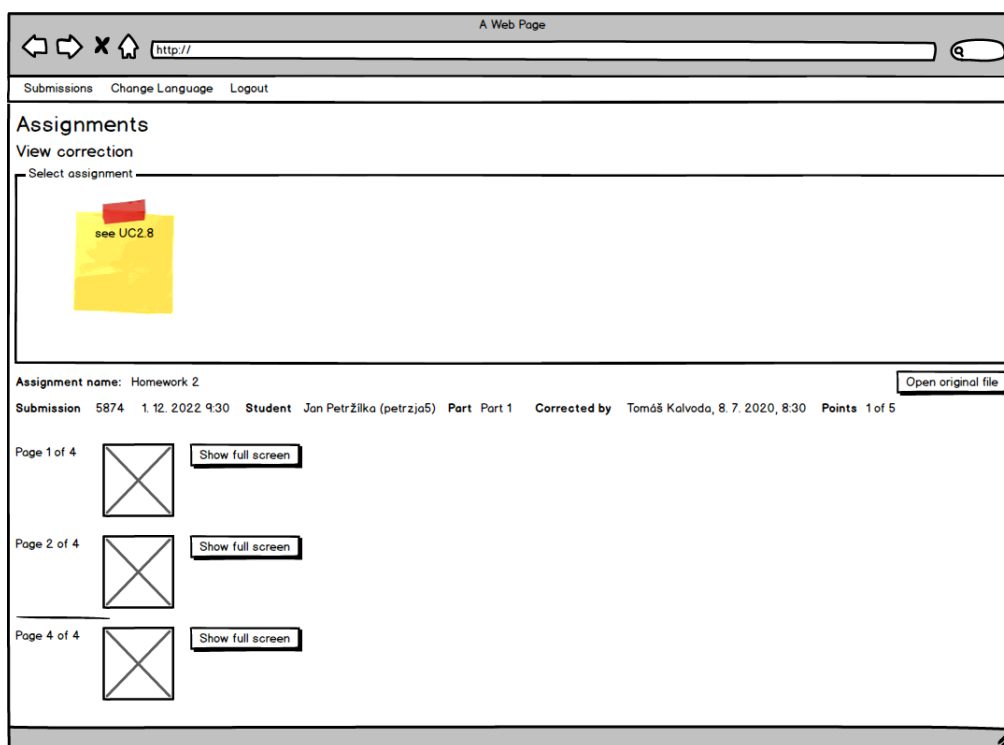
■ **Obrázek E.17** Přehled opravených částí

Učitel vybere písemnou práci podle případu užití UC2.8.

Učitel použije pole pro filtrování zobrazených položek.

Aplikace zobrazí seznam částí písemných prací, které jsou již opravené tak, jak je zachyceno na obrázku E.17. Pokud pro daného studenta existuje více odevzdaných nebo opravených souborů, je zobrazen ten soubor a ta oprava, která je viditelná studentem.

Učitel vybere konkrétní část k zobrazení. Aplikace zobrazí stránky vybrané části s provedenou opravou tak, jak je zachyceno na obrázku E.18.



■ **Obrázek E.18** Zobrazení opravené práce

E.1.1.2.8 UC2.8 Výběr semestru, předmětu a práce učitelem

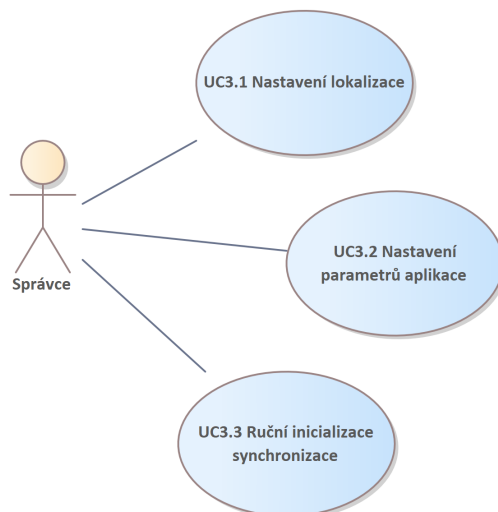
■ **Obrázek E.19** Formulář výběru semestru, předmětu a práce učitelem

Učitel vybere semestr a předmět. Aplikace zobrazí seznam písemných prací pro vybraný semestr a předmět tak, jak je znázorněno na obrázku E.19.

Učitel vybere písemnou práci a pokračuje v některém z případů užití UC2.1, UC2.2, UC2.3, UC2.4, UC2.5, UC2.6, nebo UC2.7.

E.1.1.3 Případy užití správce

Klíčové případy užití správce jsou uvedeny na obrázku E.20.



■ **Obrázek E.20** Případy užití správce

E.1.1.3.1 UC3.1 Nastavení lokalizace

Správce aplikace bude mít k dispozici formuláře, pomocí nichž bude definovat aplikační texty v českém a anglickém jazyce.

E.1.1.3.2 UC3.2 Nastavení parametrů aplikace

Správce nastaví parametry aplikace v rozsahu podle tabulky E.2.

■ **Tabulka E.2** Parametry aplikace

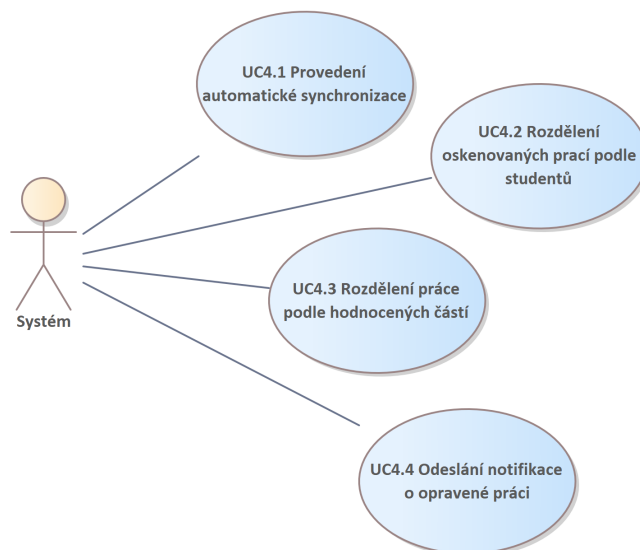
Parametr	Popis
e-mailly	Nastavení SMTP serveru pro odesílání e-mailů
SSO	Nastavení pro umožnění SSO přihlášení
Integrace	Nastavení pro integraci s externími systémy včetně frekvence synchronizace dat

E.1.1.3.3 UC3.3 Ruční inicializace synchronizace

Správce aplikace bude moci ručně vyvolat synchronizaci dat s externími aplikacemi.

E.1.1.4 Případy užití systému

Klíčové případy užití systému jsou uvedeny na obrázku E.21.



■ **Obrázek E.21** Případy užití systému

E.1.1.4.1 UC4.1 Provedení automatické synchronizace

Aplikace bude synchronizovat svoje data podle s externími aplikacemi s frekvencí podle nastavení aplikace.

E.1.1.4.2 UC4.2 Rozdělení oskenovaných prací podle studentů

Po vložení souhrnného souboru s písemnými pracemi za více studentů ve formátu PDF aplikace obsah souboru převede na obrázky a navrhne rozdělení těchto obrázků podle jednotlivých studentů.

E.1.1.4.3 UC4.3 Rozdělení práce podle ohodnocených částí

Po vložení souhrnného souboru s písemnou prací jednoho studenta k více úlohám ve formátu PDF aplikace obsah souboru převede na obrázky a navrhne rozdělení těchto obrázků podle jednotlivých úloh.

Aplikace obdobně navrhne rozdělení výstupu z případu užití UC4.2 pro jednoho studenta podle jednotlivých úloh.

E.1.1.4.4 UC4.4 Odeslání notifikace o opravené práci

Jakmile jsou všechny části odevzdané práce přiřazené jednomu studentovi opravené, aplikace zašle studentovi notifikaci o této skutečnosti.

E.2 Technické, servisní a nefunkční požadavky

E.2.1 Bezpečnostní požadavky

Uživatelským identifikátorem lokálního uživatele je e-mailová adresa. Uživatelé se budou do aplikace přihlašovat pomocí svého uživatelského identifikátoru a hesla.

Obnova hesla uživatele bude provedena odesláním linku na e-mailovou adresu uživatele, o který uživatel požádá vyplněním své e-mailové adresy. Bezpečnost uživatelského účtu je tedy limitována bezpečností e-mailové služby.

Aplikace odešle uživateli link na vytvoření hesla. Uživatel klikne na obdržený link a definuje a uloží svoje nové heslo.

Při vkládání nového hesla bude aplikace indikovat sílu zvoleného hesla a nepovolí použití slabého hesla.

Aplikace bude detekovat počet neúspěšných požadavků na přihlášení v čase. V případě překročení určené meze bude proces přihlášení trvat podstatně déle a počet souběžně probíhajících přihlašovacích procesů bude omezen.

E.2.2 E-mailový SMTP server

V produkční fázi bude používán externí (existující) SMTP server ČVUT. V testovací fázi je možné použít jiný SMTP server zajištěný studentem / autorem této práce.

E.2.3 Migrace dat

Žádná data z předchozího řešení nebudou migrována.

E.3 Předpokládaný harmonogram

Předpokládaný časový harmonogram je uveden v tabulce E.3.

■ **Tabulka E.3** Harmonogram práce

Milník	Datum	Popis
1	15. 7. 2022	Shoda o funkčnosti a struktuře aplikace
2	15. 8. 2022	Editor lze použít k opravování písemek
3	15. 10. 2022	Aplikace funguje v rozsahu hlavních případů užití bez integrace se systémy FITu
4	15. 11. 2022	Aplikace je nasazena v prostředí FITu.
5	15. 12. 2022	Aplikace je integrována na další systémy FITu, funguje přihlašování uživatelů
6	15. 2. 2023	Aplikaci lze použít v ověřovacím provozu k opravování písemek
7	15. 4. 2023	Aplikace je předána do provozu
8	4. 5. 2023	Nejzazší termín odevzdání práce, přihlášení ke SZZ

Obsah datové přílohy

Datová příloha je k dispozici elektronicky v odevzdávacím systému pro závěrečné práce a ke stažení na adrese: <https://jan.petrzilka.cz/473-218/medium.zip>

Datová příloha má následující obsah:

	readme.txt.....	stručný popis obsahu média
	demo.txt.....	pozvánka na předvedení aplikace pro zájemce
	thesis.pdf	tato diplomová práce ve formátu PDF
	docs	dokumentace
	licence.....	licenční ujednání použitých nástrojů a komponent
	thesis.....	zdrojová forma této bakalářská práce ve formátu L ^A T _E X
	sql	create script a insert script pro vytvoření databáze a její úvodní parametrizaci
	frontend.....	zdrojové soubory uživatelského rozhraní v Reactu
	src	zdrojové soubory aplikace v Javě, zdrojové soubory editoru a grafů
	pom.xml	konfigurační soubor nástroje Maven

Kontaktní údaje

Pokud se na aplikaci chcete podívat, domluvte si předvedení s využitím kontaktních údajů uvedených níže.

Pokud máte jakékoliv připomínky nebo dotazy, kontaktuje mě, prosím, podle potřeby.

Kontaktní údaje:

Jan Petrzílka
jan@petrzilka.cz
+420 777818097

