

**MATEMATICKO-FYZIKÁLNÍ
FAKULTA**
Univerzita Karlova

BAKALÁŘSKÁ PRÁCE

Sabína Ságová

**Webová aplikácia na precvičovanie
databázových dopytov**

Katedra softwaru a výuky informatiky

Vedoucí bakalářské práce: Mgr. Klára Pešková, Ph.D.

Studijní program: Informatika

Praha 2024

Prohlašuji, že jsem tuto bakalářskou práci vypracovala samostatně a výhradně s použitím citovaných pramenů, literatury a dalších odborných zdrojů.

Beru na vědomí, že se na moji práci vztahují práva a povinnosti vyplývající ze zákona č. 121/2000 Sb., autorského zákona v platném znění, zejména skutečnost, že Univerzita Karlova má právo na uzavření licenční smlouvy o užití této práce jako školního díla podle §60 odst. 1 autorského zákona.

V dne

Podpis autora

Moja úprimná vďaka patrí vedúcej bakalárskej práce, Mgr. Kláre Peškovej, Ph.D., za cenné rady, usmerňovanie, odborné konzultácie pri písaní tejto práce a vypracovaní ročníkového projektu.

Název práce: Webová aplikácia na precvičovanie databázových dopytov

Autor: Sabína Ságová

Katedra: Katedra softwaru a výuky informatiky

Vedoucí bakalárske práce: Mgr. Klára Pešková, Ph.D., Katedra softwaru a výuky informatiky

Abstrakt: V rámci tejto práce vznikla vzdelávacia webová aplikácia na precvičovanie databázových dopytov pomocou jazyka SQL. Stránka sa skladá zo štyroch hlavných častí - SQL hřišťa, Úkoly, Diskusní fórum a Test. Študenti majú možnosť voľného experimentovania s databázou v časti SQL hřišťa. V časti Úkoly majú k dispozícii rôzne úlohy na precvičovanie SQL dopytov. Každá úloha má svoje zadanie a študenti môžu napísať riešenie. Po odoslaní riešenia zistia, či ich riešenie bolo správne alebo nie. V časti Diskusní fórum môžu vidieť praktický príklad použitia SQL dopytov. V časti Test si študenti môžu precvičovať úlohy podľa výberu učiteľa. Časť Test umožňuje učiteľom pridávať vlastné úlohy, vytvárať sady úloh a jednoducho ich zdieľať so študentmi. Vďaka použitiu lokálnej databázy nie je potrebné, aby sa študenti niekde registrovali a aplikácia je teda ľahko použiteľná v školách. Prebehlo pilotné testovanie, ktoré ukázalo, že je aplikácia funkčná, ale stále by bolo vhodné aplikáciu otestovať vo výučbe so stredoškolskými študentmi.

Klíčová slova: SQL precvičovanie, vzdelávacia webová aplikácia, stredné školy

Title: Web application for learning SQL

Author: Sabína Ságová

Department: Department of Software and Computer Science Education

Supervisor: Mgr. Klára Pešková, Ph.D., Department of Software and Computer Science Education

Abstract: Within this work, an educational web application was created for practicing database queries using SQL language. The website consists of four main parts - SQL hřiště, Úkoly, Diskusní fórum and Test. Students are free to experiment with the database in the SQL hřiště. In the Úkoly section, they have a variety of tasks to practice SQL queries. Each task has its own assignment and students can write the solution. After submitting the solution, they will find out whether their solution was correct or not. In the Diskusní fórum section, they can see a practical example of using SQL queries. In the Test section, students can practice tasks of the teacher's choice. The Test section allows teachers to add their own tasks, create task sets and easily share them with students. Thanks to the use of a local database there is no need for students to register anywhere, making the application easy to use in schools. Pilot testing has been carried out, which has shown that the application is functional, but it would still be advisable to test the application in the classroom with secondary school students.

Keywords: SQL practice, educational web application, secondary schools

Obsah

Úvod	3
1 Podobné aplikácie	5
1.1 SQLZoo	5
1.2 SQLBolt	6
1.3 SQLCourse	7
1.4 W3Schools	7
1.5 SQL Murder Mystery	8
2 Popis aplikácie	9
2.1 Úvodná stránka	9
2.2 SQL hřiště	10
2.3 Úkoly	10
2.4 Diskusní fórum	11
2.5 Test	12
3 Průručka - študent a učiteľ	13
3.1 Průručka pre študentov	13
3.1.1 Úkoly	13
3.1.2 SQL hřiště	16
3.1.3 Test	21
3.1.4 Diskusní fórum	22
3.2 Průručka pre učiteľov	23
3.2.1 Diskusní fórum	23
3.2.2 Test	25
4 Technické riešenie	31
4.1 Prehľad použitých technológií	31
4.2 Uživatelské rozhranie	31
4.2.1 Bootstrap	32
4.3 SQLite	33

4.4	Flask	33
4.5	API	34
4.5.1	REST API	34
4.5.2	JSON	34
4.6	HTTP	35
4.7	Ajax	36
5	Implementácia	37
5.1	Štruktúra adresára	37
5.2	Zaujímavé funkcie	38
5.2.1	SQL hřiště	38
5.2.2	Úkoly	41
5.2.3	Diskusní fórum	41
5.2.4	Test	42
5.3	API	43
5.4	Štruktúra databázových tabuliek	45
5.5	Požiadavky pre lokálny vývoj	47
6	Pilotné testovanie stránky	49
	Záver	55
	Seznam použité literatury	57
A	Prehľad elektronických príloh	59
A.1	Zoznam skratiek	59

Úvod

Rámcový vzdelávací program (RVP) [1] je základným pravidlom pre vzdelávací systém v ČR, ktorý vydáva Ministerstvo školstva. Určuje, čo by mali žiaci vedieť na rôznych stupňoch vzdelávania. Aktualizácia RVP pre gymnázia nadobudla účinnosť od 1. septembra 2022. Jedným z očakávaných výstupov RVP je znalosť zobrazenia a filtrovania dát podľa databázových dopytov. Jeden z problémov pri výučbe informatiky, s ktorým sa stretávajú učitelia aj študenti, je nedostatok vhodných nástrojov pre výučbu štrukturovaného dopytovacieho jazyka SQL.

Existujúce nástroje, ako je Microsoft Office Access, nie sú vhodné v pedagogickom prostredí kvôli ich komplexnosti a zložitosti. Access bol kedysi populárnym nástrojom na tvorbu databáz, ale v súčasnosti sú moderné alternatívy, ako MySQL, PostgreSQL alebo SQLite, preferovanejšie v praxi pre svoje lepšie funkcie a možnosti. Pre niektorých študentov môže byť Access príliš zložitý na pochopenie a ovládanie, čo môže viesť k frustrácii a zníženiu motivácie pre výučbu databázových zručností. Okrem využitia databázových nástrojov ako Access alebo Excel by ďalšou možnosťou na učenie SQL mohli byť nástroje ako je napríklad PHP Adminer alebo phpMyAdmin. Avšak, správa databázy a účtov pre študentov je náročná pre správcu školskej počítačovej siete. Okrem toho, väčšina existujúcich nástrojov je v anglickom jazyku, čo predstavuje jazykovú bariéru pre študentov. Taktiež tieto nástroje nie sú prispôbosené potrebám stredných škôl a gymnázií, pretože sú určené pre vývojárov v praxi.

Náš projekt rieši tieto problémy prostredníctvom využitia JavaScriptovej databázy, čo eliminuje potrebu vytvárať účty pre študentov. Každý študent nemusí mať vlastné tabuľky ani vlastnú databázu, čo uľahčuje správu. Pri načítaní stránky sa vytvorí lokálna kópia tabuliek. Týmto spôsobom sa vyhýbame situáciám, kedy by žiaci mohli neúmyselne zmazať tabuľky ostatných používateľov.

Takmer každá webová stránka, ktorá interaguje s používateľmi a uchováva dáta, potrebuje nejakú formu databázy. Študenti majú príležitosť vidieť praktický príklad integrácie databázy s webovým rozhraním pomocou vopred pripraveného príkladu Diskusní fórum.

V rámci tejto práce sme vyvinuli aplikáciu, ktorá umožňuje študentom a učiteľom interaktívne precvičovať jazyk SQL. Aplikácia je ľahko prístupná a

užívateľsky prívetivá. Navyše je aplikácia responzívna a dobre sa zobrazuje aj na mobilných telefónoch. Dúfame, že tento nástroj zvýši dostupnosť a kvalitu edukačných materiálov v oblasti informačných systémov, čím podporí efektívnejšiu výučbu a učenie sa štrukturovaného dopytovacieho jazyka SQL na stredných školách a gymnáziách. Okrem samotného precvičovania SQL dopytov ponúka aj širší obraz toho, ako funguje práca s databázami v reálnej situácii.

Kapitola 1

Podobné aplikácie

V tejto kapitole predstavujeme podobné aplikácie.

Pri tvorbe nových aplikácií je nevyhnutné získať prehľad o konkurenčných aplikáciách. Týmto získame informácie o tom, čo funguje efektívne a čo je potrebné zlepšiť. Inšpiráciou z ich silných stránok môžeme vytvoriť kvalitné aplikácie, ktoré oslovia používateľov. Je rovnako dôležité porovnať funkcie iných aplikácií, aby sme lepšie pochopili očakávania používateľov. Napríklad na webovej stránke SQLBolt sa zobrazia dáta z tabuľky podľa nami zadaného dopytu, zatiaľ čo na stránke SQLCourse si môžeme zobrazíť iba správnu odpoveď.

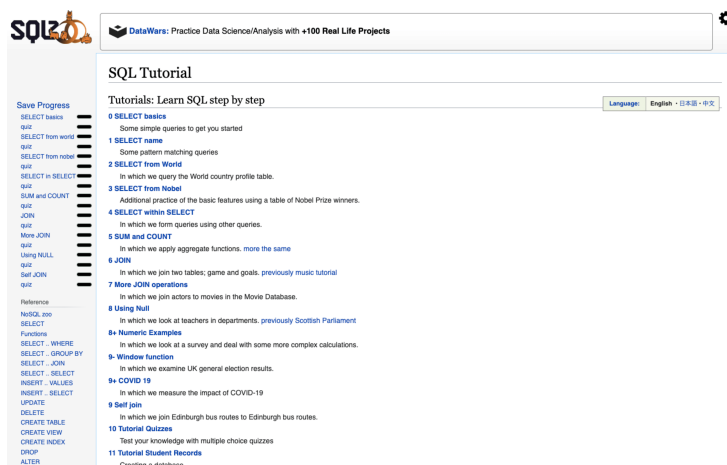
Celkovo povedané, analýza konkurenčných aplikácií predstavuje nevyhnutný prvý krok pri vytváraní nových aplikácií, ktoré uspokoja potreby používateľov a majú potenciál posilniť výučbu SQL na stredných školách a gymnáziách.

A teraz predstavíme aplikácie, ktoré sa zaoberajú podobnou problematikou.

1.1 SQLZoo

SQLZoo¹ je stránka, ktorá slúži na postupné učenie sa SQL. Obsahuje tutoriály s nápovedami a kvízy, pričom tiež ukladá náš pokrok. Na stránke môžeme skúšať písať rôzne SQL dopyty. SQLZoo je webová aplikácia, ktorá sa na prvý pohľad môže zdať trochu zastaraná. Neaktualizovaný dizajn má negatívny vplyv na prezentáciu SQL. SQLZoo by mohla výrazne zlepšiť užívateľský zážitok prostredníctvom aktualizácie na novšiu verziu s modernizovaným užívateľským rozhraním. Takéto zmeny by pravdepodobne prilákali viac používateľov a zvýšili ich spokojnosť s aplikáciou. Po napísaní správneho SQL dopytu sa zobrazí šťastný emotikon. Na stránke nefunguje vytváranie tabuliek a taktiež nefunguje UPDATE TABLE. Celá stránka je v angličtine.

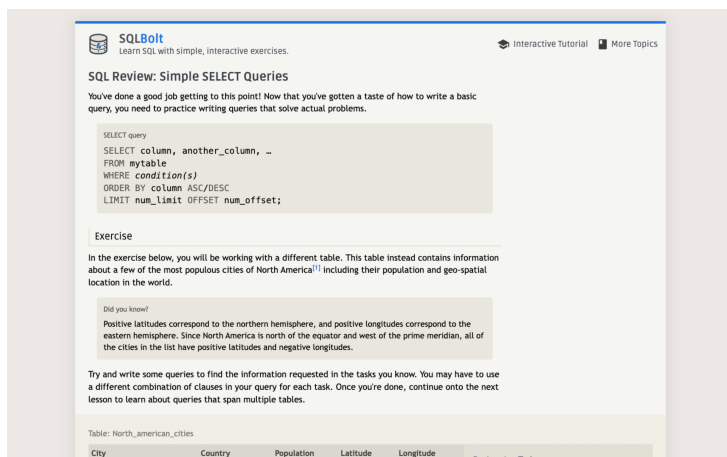
¹Stránka SQLZoo je dostupná na https://www.sqlzoo.net/wiki/SQL_Tutorial.



Obrázek 1.1 Snímka obrazovky aplikácie SQLZoo

1.2 SQLBolt

SQLBolt² je webová aplikácia s jednoduchými a interaktívnymi SQL lekciami a príkladmi. Stránka obsahuje iba jeden typ lekcií, čo môže byť nevýhodou pre tých, ktorí vyhľadávajú viacero variantov. Napriek tomu, je však jednoduchá a prehľadná, čo uľahčuje proces učenia sa. Lekcie sú rozdelené podľa úrovne zložitosti a každá sa skladá z teoretickej a praktickej časti. Teoretická časť poskytuje potrebné vysvetlenia, zatiaľ čo praktická časť umožňuje ich aplikáciu v konkrétnych príkladoch. Stránka umožňuje precvičovanie SQL dopytov. Na stránke nájdeme pár existujúcich tabuliek k dopytom, ale nie je možné vytvárať vlastné tabuľky.

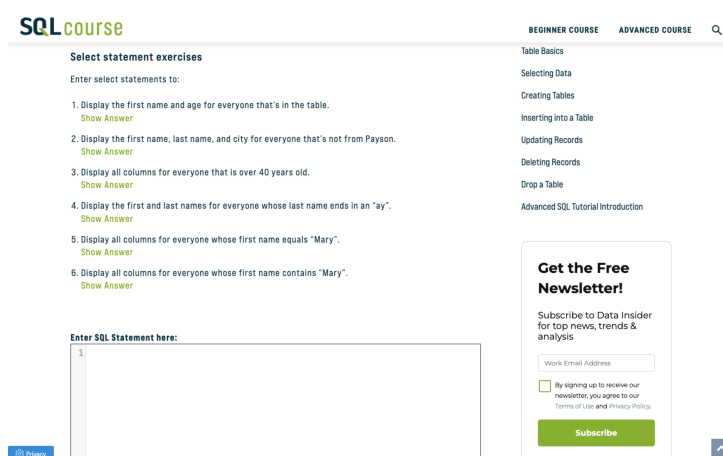


Obrázek 1.2 Snímka obrazovky lekcie z aplikácie SQLBolt

²Stránka SQLBolt je dostupná na <https://sqlbolt.com>.

1.3 SQLCourse

Webová aplikácia **SQLCourse**³ ponúka výber SQL lekcí pre začiatočníkov aj pokročilých užívateľov. Každá lekcia sa zaoberá nielen teóriou, ale aj praktickými príkladmi, kde sa študenti učia pracovať s hodnotami v tabuľkách. Webová aplikácia poskytuje iba zadanie úlohy a možnosť zobrazenia správnej odpovede, čo nám neumožňuje overiť, či je naše vlastné riešenie správne, alebo či existuje iný spôsob, ako úlohu vyriešiť.



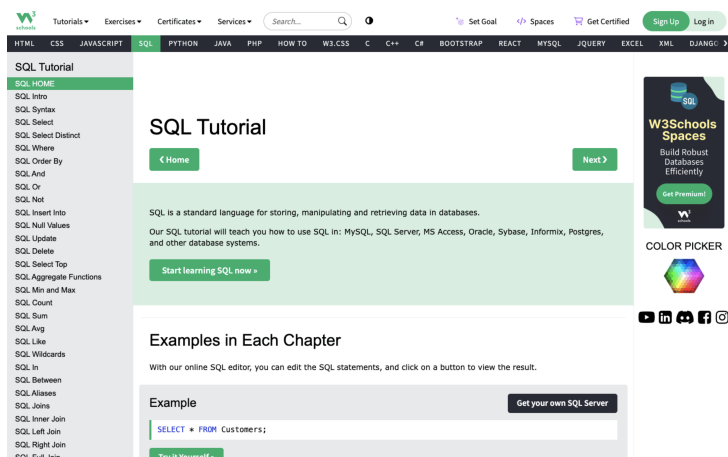
Obrázek 1.3 Snímka obrazovky príkladov z aplikácie SQLCourse

1.4 W3Schools

Webová aplikácia **W3Schools**⁴ nie je len o výučbe jazyka SQL. Ponúka množstvo tutoriálov zameraných na rôzne programovacie jazyky, ako sú napríklad Python, Java alebo C++. Stránka má excelentne navrhnuté užívateľské rozhranie. V hornom paneli sú už spomínané programovacie jazyky a naľavo sú lekcie, ktoré sú dobre spracované. Okrem toho, stránka ponúka aj interaktívne možnosti. Na stránke sa nachádza interaktívne tlačidlo *Try it*, vďaka ktorému je možné si vyskúšať písať dopyty. Stránka obsahuje vopred pripravené tabuľky, ale neponúka možnosť si vytvoriť vlastnú tabuľku. Stránka je vhodná pre začiatočníkov, ale aj pokročilých užívateľov.

³Stránka SQLCourse je dostupná na <https://www.sqlcourse.com>.

⁴Stránka W3Schools je dostupná na <https://www.w3schools.com/sql/>.



Obrázek 1.4 Snímka obrazovky aplikácie W3Schools

1.5 SQL Murder Mystery

SQL Murder Mystery⁵ je webová stránka, ktorá poskytuje interaktívny príklad využitia jazyka SQL na riešenie kriminálnej záhady. Jej cieľom je umožniť používateľom precvičiť si svoje SQL zručnosti prostredníctvom skúmania a analyzovania dát v databázových tabuľkách. Stránka je určená skôr pre pokročilých používateľov než pre začiatočníkov. Na stránke sú taktiež vysvetlené pojmy spojené s SQL. Celkovým cieľom je teda umožniť používateľom prakticky aplikovať svoje znalosti SQL na riešenie zložitého problému, ktorým je identifikácia vraha v rámci zábavnej záhady.



Obrázek 1.5 Snímka obrazovky aplikácie SQL Murder Mystery

⁵Stránka SQL Murder Mystery je dostupná na <https://mystery.knightlab.com>.

Kapitola 2

Popis aplikácie

Stránka sa skladá zo štyroch hlavných častí - SQL hřiště, Úkoly, Diskusní fórum a Test. Na SQL hřišti môžu užívatelia vytvárať vlastné tabuľky a písať databázové dopyty. V Úkoly si môžu precvičiť svoje znalosti na vopred pripravených úlohách. Diskusní fórum je väčší príklad, ktorý získava funkcionality po dopísaní správnych SQL dopytov. Časť Test umožňuje tvoriť vlastné Úkoly.

Aplikácia je dostupná na: <https://slt.ufal.mff.cuni.cz/sabina/>.

2.1 Úvodná stránka

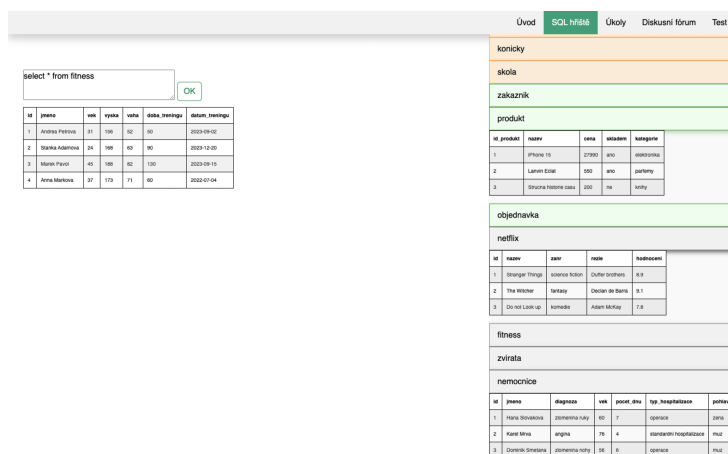
Na úvodnej stránke aplikácie sa nachádza názov projektu spolu s krátkym popisom jeho účelu. Okrem toho tu nájdeme aj rozcestník, ktorý nás naviguje k štyrom hlavným častiam projektu.



Obrázek 2.1 Snímka obrazovky — Úvod

2.2 SQL hřiště

SQL hřiště poskytuje prostredie na voľné precvičovanie SQL dopytov. Užívateľia tu majú k dispozícii dve možnosti. Môžu si vybrať medzi vytváraním vlastných tabuliek alebo prácou s existujúcimi tabuľkami. V pravom paneli je zoznam tabuliek, pri kliknutí na nejakú z nich sa zobrazí jej obsah. SQL hřiště obsahuje desať vopred pripravených tabuliek - *studenti*, *konicky*, *skola*, *zakaznik*, *produkt*, *objednavka*, *netflix*, *fitness*, *zvirata* a *nemocnice*. Ďalej sa na stránke nachádza editačné políčko, kde užívateľ môže písať dopyt. Potom užívateľ klikne na tlačidlo *OK* a zobrazí sa tabuľka podľa dopytu alebo chybová hláška.



The screenshot shows the SQL playground interface. At the top, there are navigation links: Úvod, SQL hřiště (highlighted), Úkoly, Diskusní fórum, and Test. Below the navigation, there is a text input field containing the query "select * from fitness" and an "OK" button. To the right of the input field is a table with the following data:

id	meno	vek	vyška	háva	ohodn. tréningu	datum. tréningu
1	Jana Petrova	31	160	02	50	2023-09-01
2	Marka Adamova	24	180	03	60	2023-12-01
3	Marka Pavol	45	180	02	120	2023-09-15
4	Jana Markova	37	170	11	60	2023-07-04

Below the table, there is a list of tables with their contents:

- konicky**
- skola**
- zakaznik**
- produkt**

id_produkta	nazov	cena	sklad	kategoria
1	iPhone 13	27990	ano	elektronika
2	Lacoste Esprit	590	ano	potrebny
3	Stručna história české	200	ne	knihy
- objednavka**
- netflix**

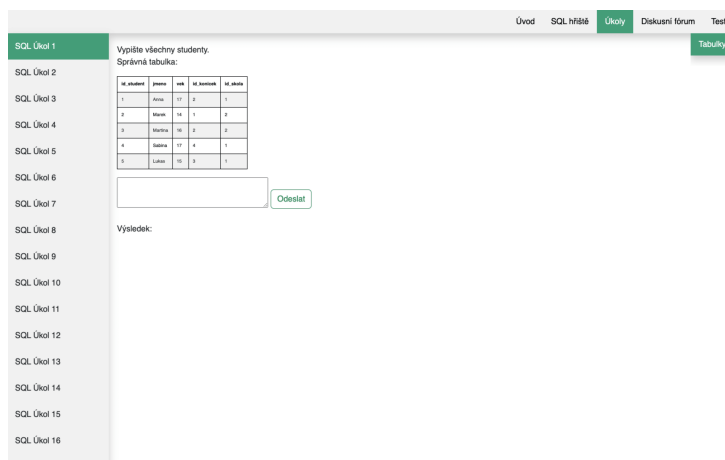
id	nazov	zazr	roka	hodnoceni
1	Stranger Things	sci-fi	2019	8.9
2	The Wheel	netflix	2020	8.1
3	Do not look up	sci-fi	2021	7.8
- fitness**
- zvirata**
- nemocnice**

id	meno	diagnoza	vek	potreb. dnu	typ. hospitalizacie	stav
1	Marka Stankova	občasná kúra	60	7	oproti	stano
2	Karel Mena	angina	75	4	stano	pacienta
3	Olivera Stankova	občasná kúra	55	6	oproti	mlu

Obrázek 2.2 Snímka obrazovky — SQL hřiště

2.3 Úkoly

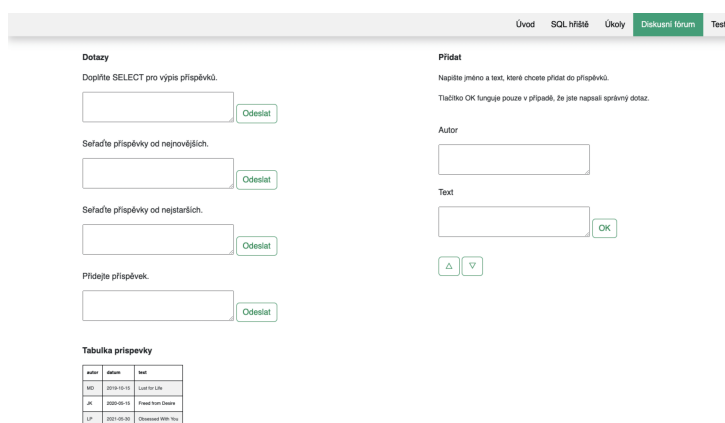
Časť **Úkoly** zahŕňa vopred pripravené úlohy, ktoré sú založené na existujúcich tabuľkách. V ľavom paneli sa nachádza zoznam úloh. Na stránke užívateľ nájde zadanie, správnu tabuľku, okno na zadávanie dopytu a tabuľku podľa zadaného dopytu. V pravom paneli sa nachádza zoznam tabuliek, pri kliknutí na ne sa zobrazí ich obsah. Použili sme rovnaké tabuľky ako v SQL hřišti. V úlohách si žiaci môžu precvičiť operátory ako sú napríklad AND, OR. Taktiež sa v úlohách nachádzajú agregačné funkcie ako sú napríklad avg, sum.



Obrázek 2.3 Snímka obrazovky — Úkoly

2.4 Diskusní fórum

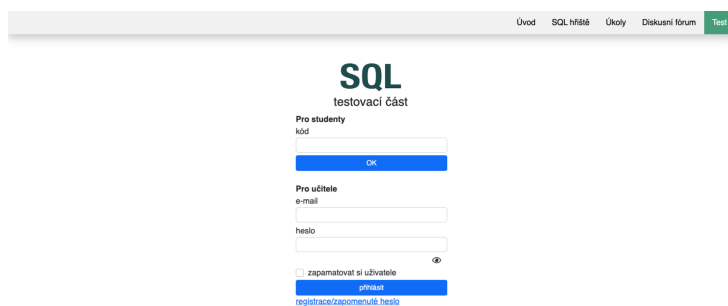
Diskusní fórum ukazuje použitie SQL dopytov v praxi. Chceme študentom ukázať, akú rolu majú databázy v informačných systémoch a na webových stránkach. V Diskusnom fóre nájdeme jeden príklad, ktorý na začiatku nie je funkčný. Avšak, postupným správnym dopĺňaním SQL dopytov sa tento príklad začne postupne rozvíjať a získavať na funkcionalite. Pre zobrazenie tabuľky *prispevky* musíme doplniť správne dopyt na vypísanie celej tabuľky. Pre usporiadanie príspevkov podľa dátumu musíme správne vyplniť dopyty s `ORDER BY`. Ak by sme chceli do tabuľky *prispevky* niečo pridať, tak najprv musíme niečo do nej vložiť pomocou dopytu s `INSERT INTO`.



Obrázek 2.4 Snímka obrazovky — Diskusní fórum

2.5 Test

Časť **Test** umožňuje učiteľom pridávať vlastné úlohy, vytvárať sady úloh a jednoducho ich zdieľať so študentmi. Učitelia sa na stránke musia zaregistrovať. Pri registrácii zadajú svoj e-mail, na ktorý im bude zaslané vygenerované heslo. Po prihlásení budú učitelia schopní vytvárať vlastné úlohy a vyberať aj z našich úloh. Následne si vybrané úlohy uložia pod ich zvoleným kódom. Po zadaní kódu sa zobrazia študentom úlohy podľa výberu učiteľa.



The screenshot shows a web interface for the 'SQL' test. At the top, there is a navigation bar with links for 'Úvod', 'SQL úlohy', 'Úkoly', 'Diskusní fórum', and 'Test'. The main content area is titled 'SQL testovací část'. It is divided into two sections: 'Pro studenty' and 'Pro učitele'. The 'Pro studenty' section has a text input field for 'kód' and a blue 'OK' button. The 'Pro učitele' section has text input fields for 'e-mail' and 'heslo', a checkbox for 'zapamatovat si uživatele', and a blue 'Přihlásit' button. Below the 'Přihlásit' button is a small link: 'registrace/zapomenuté_heslo'.

Obrázek 2.5 Snímka obrazovky — Test

Kapitola 3

Príručka - študent a učiteľ

Táto príručka sa delí na dve časti. V prvej časti sa zameriame na činnosti študentov na našej stránke a v druhej časti na činnosti učiteľov na našej stránke.

3.1 Príručka pre študentov

V tejto časti sa zameriame na činnosti, ktoré sú pre študentov dôležité pri používaní našej stránky. Ak to bude potrebné, poskytneme detailnejšie informácie v podobe snímok obrazovky alebo príkladov správnych dopytov.

Ako vidíme v Kapitole 2, stránka je rozdelená na štyri časti. Teraz nasledujú pokyny, ako tieto časti používať.

3.1.1 Úkoly

V časti Úkoly nájdeme pripravené úlohy, na precvičovanie SQL dopytov. V tejto časti stránky je vyše dvadsať úloh. V aplikácii sú pripravené tabuľky.

V pravom hornom rohu máme tlačidlo s názvom *Tabuľky*. Keď klikneme na toto tlačidlo, tak sa rozbalia všetky tabuľky. Keď klikneme na názov tabuľky, tak sa objaví obsah tabuľky. Obnovením stránky sa tabuľky vrátia do svojho pôvodného stavu, a preto sa nemusíme báť experimentovať a skúšať rôzne dopyty. Všetky tabuľky a ich obsah môžeme vidieť na snímkach obrazovky 3.1.

studenti				
id_student	meno	vek	id_konicek	id_skola
1	Anna	17	2	1
2	Mark	14	1	2
3	Matru	16	2	2
4	Sabrina	17	4	1
5	Lukas	15	3	1

konicky				
id_konicek	nazev	typ	max_cena	max_cas
1	tenisbal	sporty	50	15
2	peci	kulturne umeni	100	10
3	videotry	vorhociasova aktivita	60	15
4	maiba	umeni	50	10

skola		
id_skola	nazev	adresa
1	Gymnazium Zlina	Jurkovych 23, Zlina
2	Gymnazium Praha	Palckova 36, Praha

zakaznik			
id_zakaznik	jmeno	adresa	tel_cislo
1	Sergei Barracuda	Praha 1	222333444
2	Karen Novakova	Praha 5	329329329
3	Filip Malek	Praha 13	123321123

produkt			
id_produk	nazev	cena	skladem
1	iPhone 15	27990	ano
2	Lanvin Eclat	550	ano
3	Struzna historie casu	220	ne

objednavka			
id_objednavka	id_zakaznik	id_produk	datum
1	1	1	2023-01-02
2	2	2	2023-06-09

netflix				
id	nazev	zazr	reze	hodnoceni
1	Stranger Things	science fiction	Duffer brothers	8.9
2	The Witcher	fantasy	Declan de Barra	8.1
3	Do not look up	komedie	Adam McKay	7.8

fitness					
id	jmeno	vek	vyška	haha	datum_treningu
1	Andrea Petrova	31	156	52	2023-09-02
2	Stanka Adamova	24	168	63	2023-12-20
3	Mark Fanci	45	188	82	2023-09-15
4	Anna Markova	37	173	71	2022-07-04
5	Sabrina Plackova	23	160	56	2024-04-05

zvirata			
id	druh	vykalyt	barva
1	kocka	domaci	hradska
2	pes	domaci	corona
3	sun	staj	hradska

nemocnice						
id	jmeno	diagnoza	vek	počet_dnu	typ_hospitalizace	pohlavi
1	Hana Silavskova	zjomenna ruka	60	7	operace	zena
2	Karel Miva	angina	76	4	standardni hospitalizace	muž
3	Dominik Smetana	zjomenna ruka	56	6	operace	muž

Obrázek 3.1 Na týchto snímkach obrazovky vidíme všetky existujúce tabuľky. Skupiny tabuliek sú tematicky farebne rozdelené. Tabuľky s oranžovým a zeleným záhlavím patria k sebe. Tabuľky so šedým záhlavím sú samostatné.

Podme si spolu skúsiť spraviť jednu úlohu a to SQL Úkol 7.

Ako zadanie máme vypísať všetkých ľudí z fitka, ktorí merajú aspoň 160 cm a vážia menej než 72 kg.

SQL Úkol 1

SQL Úkol 2

SQL Úkol 3

SQL Úkol 4

SQL Úkol 5

SQL Úkol 6

SQL Úkol 7

SQL Úkol 8

SQL Úkol 9

SQL Úkol 10

SQL Úkol 11

SQL Úkol 12

SQL Úkol 13

SQL Úkol 14

SQL Úkol 15

SQL Úkol 16

Vypíšte všechny lidi z fitka, kteří měří alespoň 160 cm a váží méně než 72 kg.

Správěná tabuľka:

id	jmeno	vek	vyška	haha	datum_treningu
2	Stanka Adamova	24	168	63	2023-12-20
4	Anna Markova	37	173	71	2022-07-04
5	Sabrina Plackova	23	160	56	2024-04-05

Výsledek:

Úvod SQL nástroj **Úkoly** Diskusní fórum Test

netflix				
id	nazev	zazr	reze	hodnoceni
1	Stranger Things	science fiction	Duffer brothers	8.9
2	The Witcher	fantasy	Declan de Barra	8.1
3	Do not look up	komedie	Adam McKay	7.8

fitness					
id	jmeno	vek	vyška	haha	datum_treningu
1	Andrea Petrova	31	156	52	2023-09-02
2	Stanka Adamova	24	168	63	2023-12-20
3	Mark Fanci	45	188	82	2023-09-15
4	Anna Markova	37	173	71	2022-07-04
5	Sabrina Plackova	23	160	56	2024-04-05

zvirata			
id	druh	vykalyt	barva
1	kocka	domaci	hradska
2	pes	domaci	corona
3	sun	staj	hradska

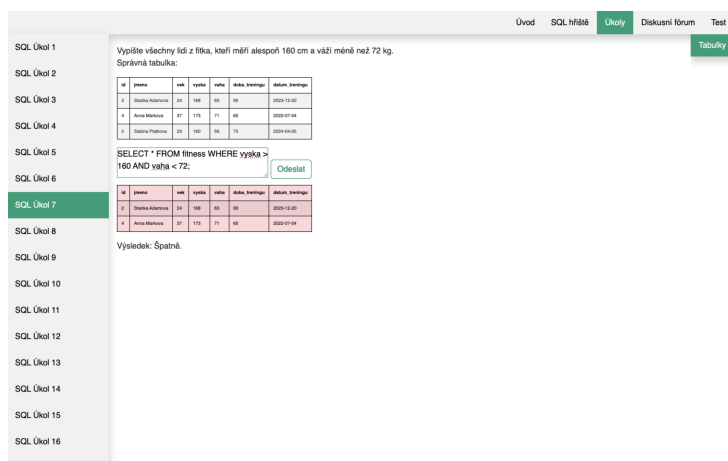
nemocnice						
id	jmeno	diagnoza	vek	počet_dnu	typ_hospitalizace	pohlavi
1	Hana Silavskova	zjomenna ruka	60	7	operace	zena
2	Karel Miva	angina	76	4	standardni hospitalizace	muž
3	Dominik Smetana	zjomenna ruka	56	6	operace	muž

Obrázek 3.2 Snímka obrazovky — SQL Úkol 7. Vľavo vidíme zoznam všetkých úloh. Vpravo vidíme rozliknuté tabuľky. Záhlavie tabuľky, ktorá patrí k danej úlohe je žlté.

Vyskúšame nasledujúci dopyt.

```
SELECT * FROM fitness WHERE vyska > 160 AND vaha < 72;
```

Ako vidíme v snímku obrazovky 3.3, tak sme úlohu správne nevyriešili, pretože výsledok sa zafarbil na červeno.



Úvod SQL úlohy Úlohy Diskusní fórum Test

Tabuľky

SQL Úkol 1 Vypíšte všetkých ľudí z fitka, ktorí majú alespoň 160 cm a vážia menej než 72 kg.

SQL Úkol 2 Správna tabuľka:

id	meno	vek	vyska	vaha	data_zavazky	data_novotny
2	Stanislav Kozmicki	24	168	62	60	2023-12-20
4	Anna Maronova	37	175	71	45	2022-07-04
5	Stanislav Kozmicki	25	160	58	70	2023-08-03

SQL Úkol 3

SQL Úkol 4

SQL Úkol 5

SQL Úkol 6

SQL Úkol 7

```
SELECT * FROM fitness WHERE vyska > 160 AND vaha < 72;
```

Odeslať

id	meno	vek	vyska	vaha	data_zavazky	data_novotny
2	Stanislav Kozmicki	24	168	62	60	2023-12-20
4	Anna Maronova	37	175	71	45	2022-07-04

SQL Úkol 8 Výsledok: Spatné.

SQL Úkol 9

SQL Úkol 10

SQL Úkol 11

SQL Úkol 12

SQL Úkol 13

SQL Úkol 14

SQL Úkol 15

SQL Úkol 16

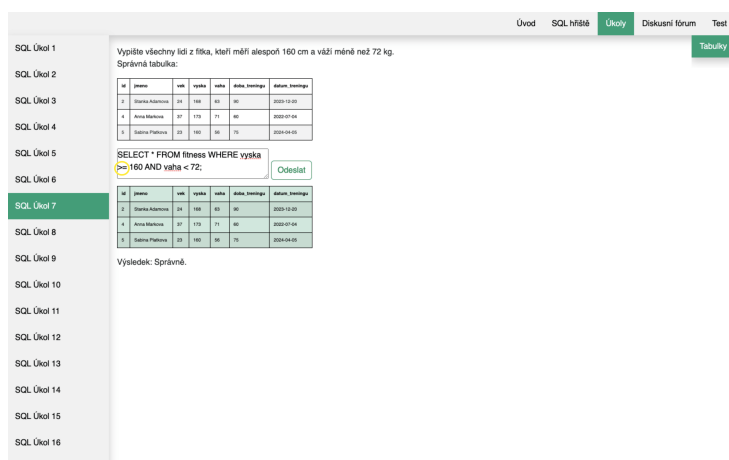
Obrázek 3.3 Snímka obrazovky — SQL Úkol 7 zle. Po zadaní dopytu sa zobrazí tabuľka s výsledkom. V prípade, že je náš SQL dopyt nesprávny, tak sa nám zobrazí červená tabuľka.

Skúsme to teda ešte znova. V zadaní sa píše, že ľudia z fitka majú merať aspoň 160 cm, tak tentokrát pridáme aj rovnosť.

Skúsime zmenený dopyt:

```
SELECT * FROM fitness WHERE vyska >= 160 AND vaha < 72;
```

Ako vidíme v snímku obrazovky 3.4, tak teraz sa nám úlohu podarilo úspešne vyriešiť.



Obrázek 3.4 Snímka obrazovky — SQL Úkol 7 správne. Po zadaní dopytu sa zobrazí tabuľka s výsledkom. V prípade, že je náš SQL dopyt správny, tak sa nám zobrazí správna tabuľka, ktorá je vyfarbená na zeleno. Žltou farbou je zvýraznená zmena v dopyte.

3.1.2 SQL hřiště

SQL hřiště slúži na voľné precvičovanie SQL dopytov, ktoré môže pomôcť pri riešení úloh. Dovoľuje nám aj rôzne dopyty, ktoré modifikujú tabuľku. Buď môžeme pracovať s existujúcimi tabuľkami alebo si vytvoriť vlastnú tabuľku a na nej skúšať rôzne dopyty.

Práca s tabuľkami

id_student	jmeno	vek	id_konicek	id_skola
1	Anna	17	2	1
2	Marek	14	1	2
3	Martina	16	2	2
4	Sabina	17	4	1
5	Lukas	15	3	1

Obrázek 3.5 Snímka obrazovky — Tabuľka studenti

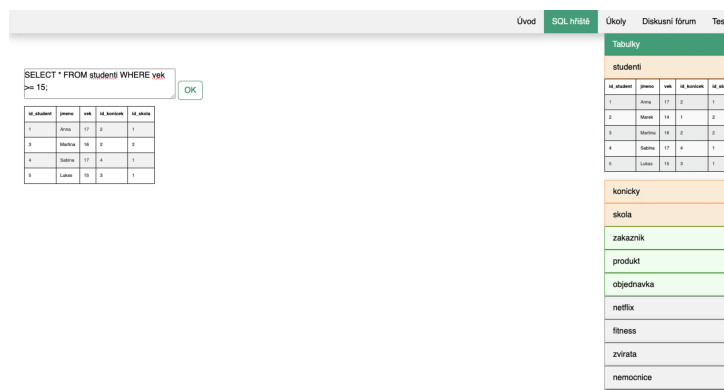
Podme si spolu vypísať tabuľku *studenti*, ktorú vidíme na snímke obrazovky 3.5. Do editačného políčka vložíme nasledujúci dopyt.

```
SELECT * FROM studenti;
```

Teraz si vypíšeme iba študentov, ktorí už môžu mať občiansky preukaz.

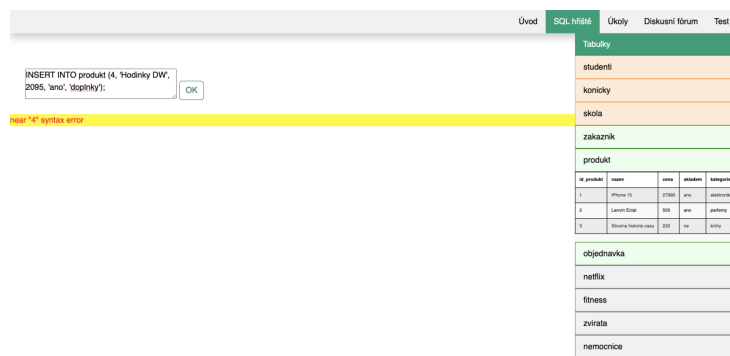
```
SELECT * FROM studenti WHERE vek >= 15;
```

Správny výsledok vidíme v snímke obrazovky 3.6. Správny dopyt môžeme napísať rôznymi spôsobmi.



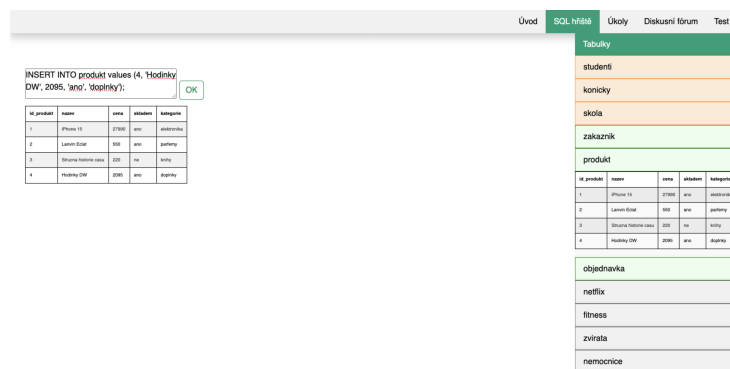
Obrázek 3.6 Snímka obrazovky — Tabuľka *studenti* po zadaní SQL dopytu. Zobrazí sa výsledok zadaného dopytu.

Ak zadáme nesprávny dopyt, tak sa nám zobrazí chybová hláška.



Obrázek 3.7 Snímka obrazovky — SQL hřiště po zadání nesprávného dopytu. V zadanom dopyte je syntaktická chyba.

Keď chybu v dopyte opravíme, klikneme na tlačidlo *OK*, tak chybová hláška zmizne.



Obrázek 3.8 Snímka obrazovky — SQL hřiště po zadání správného dopytu

Nižšie si ukážeme príklady ďalších rôznych dopytov, ktoré si môžete vyskúšať. Čo robia?

```
INSERT INTO studenti VALUES (6, 'Emanuel', 26, 3, 1);
```

```
UPDATE studenti SET jmeno = 'Marina' WHERE id_student = 3;
```

```
DELETE FROM studenti WHERE jmeno LIKE 'L%';
```

```
SELECT studenti.jmeno, studenti.vek, konicky.nazev AS konicek, skola.nazev AS skola FROM studenti JOIN konicky ON studenti.id_konicek = konicky.id_konicek JOIN skola ON studenti.id_skola = skola.id_skola ;
```

Práca s vlastnou tabuľkou

Môžeme si skúsiť vytvoriť vlastnú tabuľku. Naším cieľom je vytvoriť si tabuľku zamestnanci, ktorú vidíme na snímke obrazovky 3.9. Ak sa tam tabuľku podarí vytvoriť, tak sa zobrazí pod existujúcimi tabuľkami.

id_zamestnance	pozice	plat
1	Softwarovy vyvojjar	80000
2	Datovy analytik	90000
3	UI/UX designer	70000

Obrázek 3.9 Snímka obrazovky — Nová tabuľka zamestnanci

```
CREATE TABLE zamestnanci (id_zamestnance, pozice, plat);
```

Do našej novej tabuľky pridáme nejakých zamestancov. Našu tabuľku po vložení týchto dopytov môžeme vidieť v snímke obrazovky 3.9.

```
INSERT INTO zamestnanci VALUES (1, 'Softwarovy vyvojjar', 80000);  
INSERT INTO zamestnanci VALUES (2, 'Datovy analytik', 90000);  
INSERT INTO zamestnanci VALUES (3, 'UI/UX designer', 70000);
```

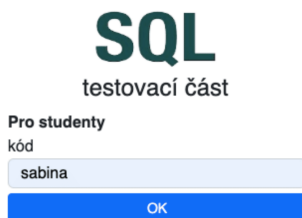
Tabulky		
studenti		
konicky		
skola		
zakaznik		
produkt		
objednavka		
netflix		
fitness		
zvirata		
nemocnice		
zamestnanci		
id_zamestnance	pozice	plat
1	Softwarovy vyvojcar	80000
2	Datovy analytik	90000
3	UI/UX designer	70000

Obrázek 3.10 Snímka obrazovky — Nová tabuľka pod existujúcimi tabuľkami

Na našej novej tabuľke môžeme skúšať rôzne dopyty rovnako ako na existujúcich tabuľkách.

3.1.3 Test

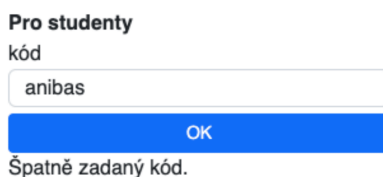
Časť Test slúži k tomu, aby vám učiteľ mohol vytvoriť personalizovanú sadu úloh. K tomu, aby ste si ju mohli zobrazíť potrebujete vložiť kód. Kód študenti dostanú od učiteľa, ak im pripravil vlastnú sadu úloh. Kód **sabina** je príklad kódu. Táto sada úloh obsahuje tri úlohy.



The screenshot shows a web interface for an SQL test. At the top, the word "SQL" is displayed in a large, bold, dark green font, with "testovací část" (test part) underneath in a smaller, grey font. Below this, the text "Pro studenty" (For students) is followed by "kód" (code). A light blue input field contains the text "sabina". Below the input field is a prominent blue button with the text "OK" in white.

Obrázek 3.11 Snímka obrazovky — Zadané kódu pre zobrazenie vlastnej sady úloh

Ak zadáme kód, ktorý neexistuje, tak sa nám zobrazí chybová hláška.

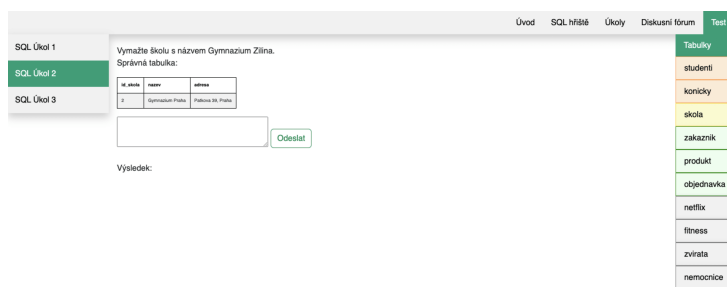


The screenshot shows the same SQL test interface as in the previous image. The input field now contains the text "anibas". Below the input field is a blue button with the text "OK". Below the button, the text "Špatně zadaný kód." (Incorrectly entered code.) is displayed in a grey font.

Obrázek 3.12 Snímka obrazovky — Nesprávny kód

Po zadaní kódu sa zobrazí sada úloh.

Úlohy fungujú rovnako a aj sa zobrazia rovnako ako v časti Úkoly 3.1.1.



The screenshot shows the SQL test interface with a custom set of tasks. The top navigation bar includes "Úvod", "SQL úlohy", "Úkoly", "Diskusní fórum", and "Test". The main content area is divided into three sections: "SQL Úkol 1" with the instruction "Vymažte školu s názvem Gymnazium Zlín.", "SQL Úkol 2" with the instruction "Správna tabuľka:" and a table with columns "id_školy", "názov", and "adresa", and "SQL Úkol 3" with a text input field and an "Odeslať" button. On the right side, there is a vertical list of tables: "Tabuľky", "studenti", "konicky", "skola", "zakaznik", "produkt", "objednavka", "netflix", "fitness", "zvirata", and "nemocnice".

Obrázek 3.13 Snímka obrazovky — Vlastná sada úloh pre kód **sabina**

3.1.4 Diskusní fórum

Diskusní fórum ukazuje použití SQL dopytov v praxi. Chceme ukázat, akú rolu majú databázy v informačných systémoch a na webových stránkach.

V ľavej časti sú štyri okná na doplnenie SQL dopytov. Pod nimi vidíme tabuľku *prispevky*. Po správnom doplnení dopytov v ľavej časti, začne fungovať pravá časť stránky.

Ak doplníme správne SQL dopyt na výpis tabuľky, tak sa nám v pravej časti zobrazí tabuľka *prispevky*. Ak doplníme správne dopyty s `ORDER BY`, tak nám budú fungovať zoraďovacie tlačidlá. Ak doplníme správne dopyt s `INSERT INTO`, tak nám začne fungovať pridávanie príspevkov.

Skúste si samostatne vyriešiť Diskusní fórum.

autor	datum	text
MB	2019-03-01	SQL hříbá
JR	2019-03-01	První test úkoly
LT	2019-03-01	První test úkoly

Obrázek 3.14 Snímka obrazovky — Diskusní fórum

3.2 Príručka pre učiteľov

V tejto časti sa zameriame na činnosti, ktoré sú pre učiteľov dôležité pri používaní našej stránky. Ak to bude potrebné, poskytneme detailnejšie informácie v podobe snímok obrazovky alebo príkladov správnych dopytov.

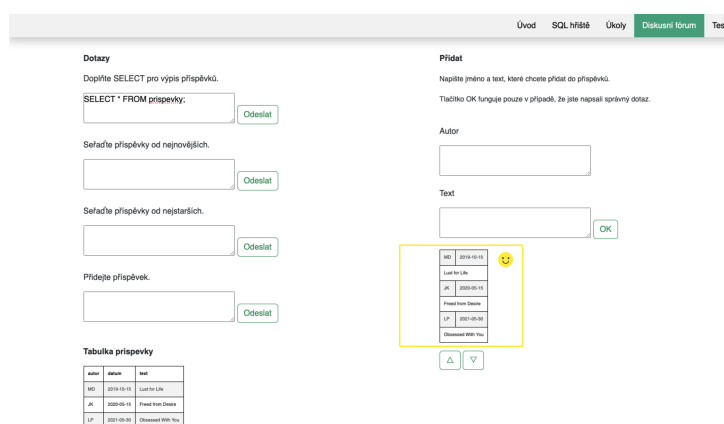
3.2.1 Diskusní fórum

Diskusné fórum slúži na demonštráciu praktického použitia SQL dopytov. Naším cieľom je ukázať študentom dôležitú úlohu, ktorú databázy zohrávajú v informačných systémoch a na webových stránkach.

V diskusnom fóre začneme jednoduchým SQL dopytom na výpis celej tabuľky. Tabuľku *prispevky* môžeme vidieť na stránke.

```
SELECT * FROM prispevky;
```

Na pravej strane stránky sa nám objaví tabuľka *prispevky*.

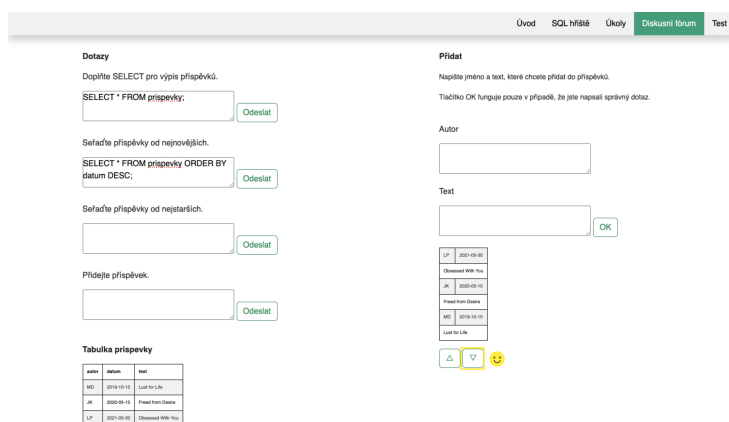


Obrázek 3.15 Snímka obrazovky — Diskusní fórum výpis tabuľky *prispevky*. Emotikony nie sú súčasťou aplikácie.

Doplníme druhý dopyt na zoradenie príspevkov od najnovších.

```
SELECT * FROM prispevky ORDER BY datum DESC;
```

Začalo nám aj fungovať jedno tlačidlo napravo.

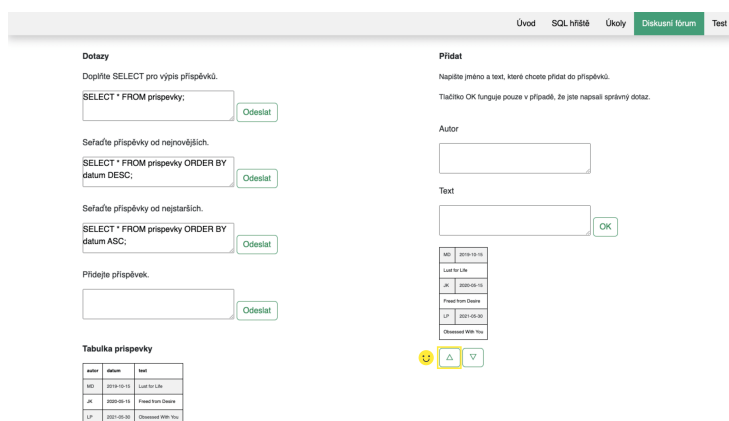


Obrázek 3.16 Snímka obrazovky — Diskusní fórum zoradenie príspevkov od najnovších

Doplníme druhý dopyt na zoradenie príspevkov od najstarších.

```
SELECT * FROM prispevky ORDER BY datum ASC;
```

Po dopísaní správneho tretieho dopytu nám funguje aj druhé tlačidlo a tým pádom nám už fungujú obe tlačidlá na zoraďovanie.

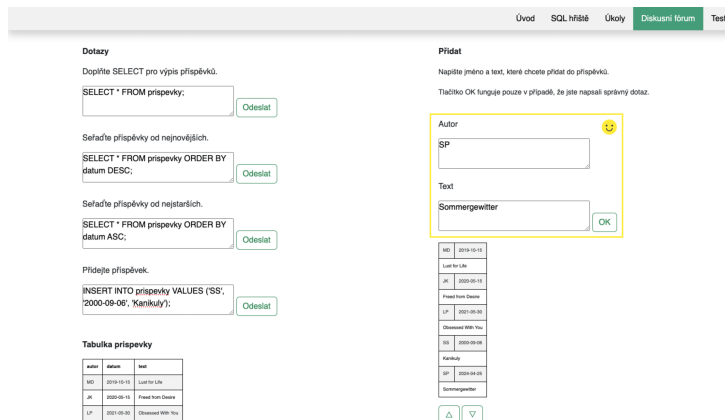


Obrázek 3.17 Snímka obrazovky — Diskusní fórum zoradenie príspevkov od najstarších

Doplníme posledný dopyt. Tentokrát ide o vkladanie nových príspevkov.

```
INSERT INTO prispevky VALUES ('SS', '2000-09-06', 'Kanikuly');
```


Doplnením posledného dopytu sme získali plne funkčné Diskusní fórum. Funguje aj pridávanie príspevkov v pravej časti stránky, čo môžeme vidieť na snímke obrazovky 3.18.



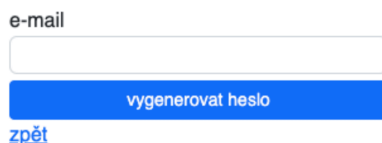
Obrázek 3.18 Snímka obrazovky — Plne funkčné Diskusní fórum

3.2.2 Test

Posledná časť našej stránky je špeciálne určená predovšetkým pre učiteľov. V tejto časti si učiteľ môže vytvoriť vlastnú sadu úloh pre svojich študentov. Úlohy môžu obsahovať ľubovoľné dopyty. Tieto úlohy je možné vytvárať nad existujúcimi tabuľkami a taktiež je možné si vybrať už existujúce úlohy do vlastnej sady úloh. Témy existujúcich tabuliek boli zvolené tak, aby boli blízke študentom.

Registrácia

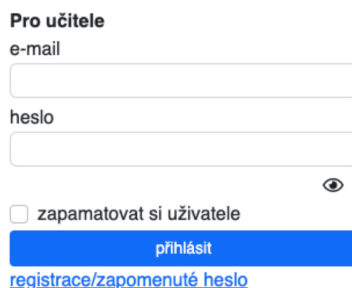
Prvým krokom každého učiteľa bude registrácia. Je potrebné zadať e-mail a kliknúť na tlačidlo *vygenerovat heslo*. Vygenerované heslo príde na zadaný e-mail. Ak heslo zabudneme, tak tu znova zadáme e-mail. Zmena hesla nie je možná. Registrovať sa môže ktokoľvek.



Obrázek 3.19 Snímka obrazovky — Registrácia

Prihlásenie

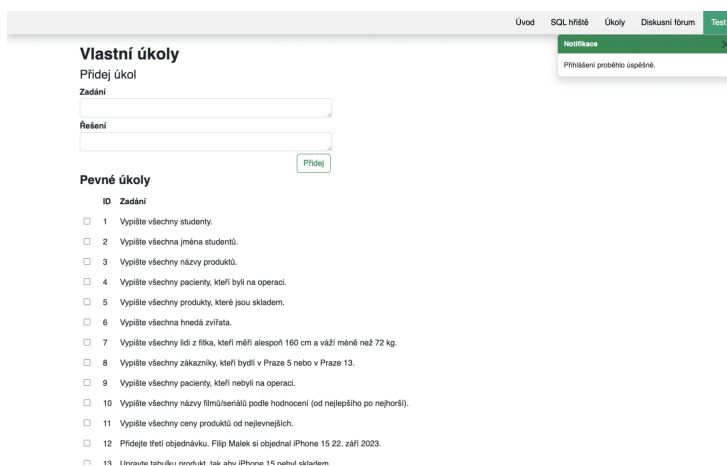
V časti Pro učitele sa prihlásime pomocou e-mailu, ktorý sme zadali a hesla, ktoré nám prišlo.



The screenshot shows a login form titled "Pro učitele". It contains two input fields: "e-mail" and "heslo". Below the "heslo" field is an eye icon for toggling password visibility. There is a checkbox labeled "zapamatovat si uživatele" and a blue button labeled "přihlásit". At the bottom, there is a link "registrace/zapomenuté heslo".

Obrázek 3.20 Snímka obrazovky — Prihlásenie

Po prihlásení sa nám zobrazí stránka na vytváranie a správu vlastných sad úloh.



The screenshot shows the "Vlastní úkoly" page. At the top, there is a navigation bar with links: "Úvod", "SQL úkoly", "Úkoly", "Diskusní fórum", and "Test". A notification box in the top right corner says "Přihlášení proběhlo úspěšně." Below the navigation bar, the page title is "Vlastní úkoly". There is a section "Přidej úkol" with input fields for "Zadání" and "Řešení", and a "Přidej" button. Below that is a section "Pevné úkoly" with a list of 12 tasks, each with a checkbox and a description. The tasks are: 1. Vypište všechny studenty. 2. Vypište všechna jména studentů. 3. Vypište všechny názvy produktů. 4. Vypište všechny pacienty, kteří byli na operaci. 5. Vypište všechny produkty, které jsou skladem. 6. Vypište všechna hnědá zvířata. 7. Vypište všechny lidi z řádku, kteří mají alespoň 160 cm a váží méně než 72 kg. 8. Vypište všechny zákazníky, kteří bydlí v Praze 5 nebo v Praze 13. 9. Vypište všechny pacienty, kteří nebyli na operaci. 10. Vypište všechny názvy filmů/seriálů podle hodnocení (od nejlepšího po nejhorší). 11. Vypište všechny ceny produktů od nejlevnějších. 12. Přidejte třetí objednávku: Filip Malák si objednal iPhone 15 22. září 2023.

Obrázek 3.21 Snímka obrazovky — Prihlásenie

Zoznam existujúcich úloh

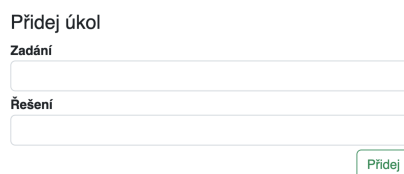
V aplikácii je pripravených 27 úloh:

1. Vypište všechny studenty.
2. Vypište všechna jména studentů.
3. Vypište všechny názvy produktů.

4. Vypište všechny pacienty, kteří byli na operaci.
5. Vypište všechny produkty, které jsou skladem.
6. Vypište všechna hnedá zvířata.
7. Vypište všechny lidi z fitka, kteří měří alespoň 160 cm a váží méně než 72 kg.
8. Vypište všechny zákazníky, kteří bydlí v Praze 5 nebo v Praze 13.
9. Vypište všechny pacienty, kteří nebyli na operaci.
10. Vypište všechny názvy filmů/seriálů podle hodnocení (od nejlepšího po nejhorší).
11. Vypište všechny ceny produktů od nejlevnějších.
12. Přidejte třetí objednávku. Filip Malek si objednal iPhone 15 22. září 2023.
13. Upravte tabulku produkt, tak aby iPhone 15 nebyl skladem.
14. Vymažte školu s názvem Gymnázium Zilina.
15. Vypište cenu nejlevnějšího koníčku.
16. Vypište výšku nejvyššího člověka z fitka.
17. Vypište průměrnou váhu lidí z fitka.
18. Sečtěte dobu tréningu ve fitku.
19. Vypište počet koníčků, které zaberou více než 11 minut.
20. Vypište jména lidí, kteří cvičili mezi 1. červencem a 1. říjnem.
21. Vypište první dva lidi z fitka.
22. Vypište seznam studentů a jejich koníčků.
23. Vypište seznam všech škol a počtu studentů v každé škole, včetně škol, ve kterých není žádný student.
24. Vypište seznam všech zákazníků a jejich objednaných produktů.
25. Vypište celkovou cenu objednávky pro každého zákazníka.
26. Vypište zvířata, která nejsou domácí a mají hnědou barvu.
27. Odstraňte záznamy všech studentů, jejichž jméno začíná písmenem 'L'.

Vytvorenie vlastnej úlohy

V ľavom hornom rohu môžeme vytvárať vlastné úlohy. Viď snímka obrazovky 3.22. Vlastné úlohy môžeme vytvárať nad existujúcimi tabuľkami. Zadanie je text, ktorý sa zobrazí študentom. Riešenie je SQL dopyt, ktorého výsledok sa bude porovnávať s výsledkom SQL dopytu študenta. Overenie správnosti je možné vykonať v SQL hříšti.



Přidej úkol

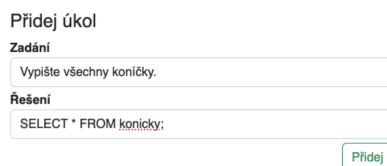
Zadání

Řešení

Přidej

Obrázek 3.22 Snímka obrazovky — Pridaj úlohu

Použitie časti Test si ukážeme na pridaní jednoduchej úlohy. Do políčka zadanie napíšeme *Vypíšte všetky koničky.* a do políčka riešenie napíšeme *SELECT * FROM konicky;*. Klikneme na tlačidlo *Přidej*.



Přidej úkol

Zadání

Řešení

Přidej

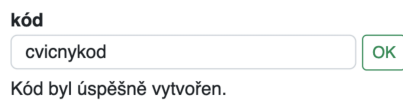
Obrázek 3.23 Snímka obrazovky — Pridaj úlohu

Takto môžeme vytvárať vlastné úlohy.

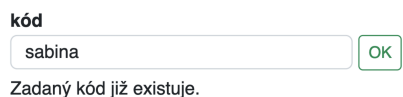
Vytvorenie sady úloh

Sada úloh sú úlohy, ktoré si zvolíme podľa seba. Táto sada môže obsahovať existujúce úlohy, ktoré nájdeme v časti Úkoly, ale aj nové úlohy, ktoré si vytvoríme my. Zaškrtnutím políčok je možné vybrať úlohy, ktoré sa objavia v sade.

Naším posledným krokom bude zvoliť si názov kódu, pod ktorým si tieto úlohy uložíme a stlačenie tlačidla *OK*. Kód, ktorý si zvolíme by mal byť ľahko zapamätateľný, aby sme ho mohli efektívne zdieľať so študentmi. Týchto 5 úloh si uložíme ako kód s názvom **cvicnykod**. Keby nami zvolený kód už existuje, tak si musíme zvoliť iný kód. Ak kód už existuje, tak vidíme chybovú hlášku.



Obrázek 3.24 Snímka obrazovky — Cvičný kód, ktorý bol úspešne vytvorený. Tento kód s úlohami môžeme poslať študentom na precvičovanie.



Obrázek 3.25 Snímka obrazovky — Kód už existuje. V tomto prípade si musíme zvoliť iný kód pre našu sadu úloh.

Pomocou jednoduchého kódu môžeme zdieľať so študentmi sady úloh na prácu v hodine alebo za domácu úlohu. Kód si každý učiteľ zvolí sám a mal by byť ľahko zapamätateľný, aby nám ho stačilo napísať na tabuľu alebo povedať študentom ústne.

Kapitola 4

Technické riešenie

Na začiatok by sme chceli vysvetliť, prečo sme sa rozhodli práve pre webovú aplikáciu. Medzi hlavné výhody webových aplikácií patrí ich prístupnosť. Webové aplikácie sú dostupné z väčšiny moderných zariadení s internetovým pripojením. V dôsledku toho sme sa rozhodli implementovať webovú aplikáciu, ktorá bude prístupná nezávisle od zariadenia alebo operačného systému.

4.1 Prehľad použitých technológií

Na našej stránke sme využili kombináciu technológií na oboch stranách - frontend aj backend.

Na strane frontendu sme pracovali s HTML, CSS a JavaScriptom. Aby sme zabezpečili responzívny dizajn a jednoduché použitie ikon, integrovali sme framework Bootstrap. JavaScript bol využitý na zabezpečenie interaktivity a dynamického správania sa stránky, vrátane načítavania dát zo servera.

Na strane backendu sme sa rozhodli pre jazyk Python a framework Flask. Pomocou Pythonu sme implementovali manipuláciu s dátami v databáze. Pre komunikáciu s databázou sme použili SQL dopyty. Aplikácia je navrhnutá s využitím REST API architektúry, čo umožňuje komunikáciu medzi frontendom a backendom pomocou HTTP požiadaviek a dát vo formáte JSON.

Teraz si tieto časti podrobnejšie rozoberieme.

4.2 Uživatelské rozhranie

Pri navrhovaní stránky sme dbali na jej prehľadné usporiadanie a estetický dojem. Zároveň sme sa snažili vytvoriť prostredie, ktoré bude prívetivé pre študentov, a preto sme zvolili farebnú paletu 4.1 vhodnú pre učenie.

Zelená je vnímaná ako príjemná a harmonická farba, čo môže prispieť k celkovému pozitívnemu dojmu stránky. Rôzne kombinácie šedej farby sú nevýrazné a nerušivé, čím poskytujú stabilné pozadie pre ďalšie prvky dizajnu. Dokonca aj oranžové odtiene môžu pridať energiu do vizuálneho zážitku a zvýrazniť určité prvky, pričom príťahujú pozornosť bez toho, aby pôsobili rušivo.



Obrázek 4.1 Snímka obrazovky farebnej palety

V časti Test je tiež prítomná modrá farba, ktorá je známa svojou schopnosťou podporiť produktivitu. Chybové hlášky sú zafarbené červeno-žlto, čo je kombinácia farieb, ktorá môže pôsobiť rušivo. Táto farebná kombinácia môže podnecovať študentov, aby sa snažili nájsť a opraviť chybu, čo najskôr.

4.2.1 Bootstrap

Takmer každá webová stránka na internete využíva HTML, CSS a JavaScript ako základné stavebné kamene. Dnes existuje množstvo rôznych frameworkov, ktoré rozširujú možnosti alebo vylepšujú vzhľad webových stránok. Napriek tomu sa každá vývojárska cesta začína s HTML, CSS a JavaScriptom. HTML poskytuje základnú štruktúru, CSS pridáva štýl a JavaScript funkčnosť. A práve sem vstupuje Bootstrap. Tvorba plne funkčnej webovej stránky od základu s týmito tromi jazykmi môže byť náročná. Bootstrap zjednoduší celý proces.

Bootstrap¹ je CSS framework, ktorý umožňuje tvorbu používateľsky prívetivých a responzívnych webových stránok bez nadmerných ťažkostí. Je to voľne dostupný a open-source framework, ktorý obsahuje základné HTML a CSS komponenty. Jeho jednoduché komponenty je možné rýchlo osvojiť a implementovať. Bootstrap sa tiež zameriava na vývoj pre mobilné zariadenia, čo je dnes nevyhnutné s ohľadom na veľký počet užívateľov, ktorí prístupujú na internet pomocou mobilných telefónov. Použitie Bootstrapu zabezpečuje, že webové stránky fungujú plynulo na všetkých zariadeniach a prispôbujú sa veľkostiam obrazoviek, čo zvyšuje ich prístupnosť a užívateľskú spokojnosť. [2]

V našej bakalárskej práci sme využili niektoré prvky z Bootstrapu na vylepšenie používateľského rozhrania. Konkrétne sme použili štýly pre tlačidlá a tabuľky. Na

¹Viac informácií na <https://getbootstrap.com>.

tlačidlá sme použili triedy `btn btn-outline-success`, ktoré poskytujú jednoduchý, ale esteticky príjemný vzhľad, a triedy `table table-success` a `table table-danger` sme aplikovali na tabuľky, čo im dáva farebné zvýraznenie podľa ich obsahu.

Tabuľky sme taktiež formátovali pomocou tried `table table-striped` `table-bordered` `table-responsive`. Tieto triedy zabezpečujú, že tabuľky sú prehľadné, čitateľné a rezponzívne, čo znamená, že sa prispôbujú rôznym veľkostiam obrazoviek.

Ďalej sme pridali oznámenia `toast` s triedami `toast toast-header bg-success text-white` a `toast toast-header bg-danger text-white`, ktoré poskytujú informácie o stave na stránke.

4.3 SQLite

SQLite² je relačný databázový systém s otvoreným zdrojovým kódom, ktorý bol vydaný v roku 2000. Cieľom pri vývoji bolo, aby bol ľahko použiteľný, kompaktný, efektívny a spoľahlivý. [3] Jeho jednoduchá integrácia priamo do aplikácie umožňuje efektívne spravovať dáta bez nutnosti samostatného databázového servera. Táto vlastnosť robí SQLite populárnou voľbou pre množstvo aplikácií, od mobilných aplikácií až po väčšie softvérové projekty. S podporou medzinárodného vývojárskeho tímu je SQLite neustále aktualizovaný a testovaný, aby poskytoval spoľahlivé a výkonné riešenia pre všetkých svojich používateľov. [4]

Náš hlavný dôvod pre voľbu SQLite bol ten, že sme chceli lokálnu JavaScriptovú databázu, kde by každý užívateľ mal vlastnú kópiu tabuliek a mohol vykonávať rôzne dopyty, vrátane tých, ktoré modifikujú tabuľku. Týmto spôsobom užívatelia môžu jednoducho používať našu webovú aplikáciu na precvičovanie SQL dopytov bez potreby komplikovaných inštalácií alebo konfigurácií serverového prostredia. Nepotrebujeme, aby mal každý užívateľ svoju vlastnú databázu na serveri.

4.4 Flask

Flask³ je malý framework pre webové aplikácie, ktorý sa väčšinou označuje ako micro-framework, vybudovaný v jazyku Python. Jeho výhodou je silná podpora komunity, dokumentácia a knižnice. [5] Jeho inštalácia je najjednoduchšia pomocou virtuálneho prostredia, ktoré izoluje balíčky a verzie Pythonu. Takto každá aplikácia má prístup len k tým balíčkom, ktoré potrebuje. Používanie virtuálnych prostredí

²Viac informácií na <https://sqlite.org>.

³Viac informácií na <https://flask.palletsprojects.com/en/3.0.x/>.

nevyžaduje administrátorské práva a je zvyčajne sprevádzané nástrojom pip pre inštaláciu balíčkov. [6]

Rozhodli sme sa použiť Flask na vytvorenie API, pretože je jednoduchý na pochopenie a použitie a poskytuje nám všetko, čo potrebujeme pre naše účely bez nutnosti zbytočného učenia sa zložitých vecí. Flask aplikácia určuje všetky koncové body REST API, ktoré umožňujú interakciu medzi frontendom a backendom. V našej aplikácii používame framework Flask spolu s rozšíreniami, ako sú Flask-Mail a Flask-Bcrypt. Taktiež využívame Python knižnicu pymysql na spojenie s databázou. S týmito nástrojmi sme vytvorili systém na registráciu a prihlásenie používateľov, správu úloh a kódov. Učitelia majú možnosť zaregistrovať sa a prihlásiť pomocou svojich e-mailových adries a hesiel. Po úspešnej registrácii sa automaticky zasiela e-mail s vygenerovaným heslom. Heslá sú ukladané v databáze v bezpečnej hashovanej podobe pomocou Flask-Bcrypt. Učitelia majú možnosť pridávať a mazať úlohy, vytvárať kódy a priradovať úlohy ku kódom. Študenti si môžu potom zobrazíť otázky priradené k danému kódu.

4.5 API

API je nástroj pre komunikáciu a interakciu medzi softvérovými aplikáciami, pri tvorbe distribuovaných softvérových systémov. [7] V podstate API funguje ako spojovací prvok medzi rôznymi časťami softvéru, uľahčujúc výmenu údajov a prístup k funkcionalitám. [8]

V našom prípade slúži API ako spojovací prvok medzi backendom a frontendom.

4.5.1 REST API

REST API je rozhranie pre programovanie aplikácií, ktoré dodržiava princípy architektúry REST ako je napríklad bezstavovosť. Architektúra REST sa zameriava na definíciu a manipuláciu zdrojov pomocou URI a štandardných operácií HTTP protokolu, ako sú GET, POST, PUT, DELETE.

Rozhodli sme sa použiť REST API, pretože webové stránky sú predvolene bezstavové. HTTP protokol, ktorým komunikujeme, je bezstavový. REST API nám umožňuje efektívne vytvárať a manipulovať so zdrojmi na webe pomocou štandardných HTTP operácií.

4.5.2 JSON

JSON je textový formát na usporiadanie dát, ktorý je založený na syntaxi objektových literálov jazyka JavaScript. Každá položka je kľúč/hodnota pár, kde kľúč je reťazec a hodnoty môžu byť rôzneho typu. Jeho prehľadná štruktúra umožňuje ľahké

čítanie pre človeka aj počítač. Tento formát je nezávislý na jazyku programovania a často sa používa v rôznych programovacích jazykoch a prostrediach. JSON je často využívaný pri výmene dát medzi klientmi a servermi v moderných webových aplikáciách.

Pre prevod dát do formátu JSON používame rozhranie `jsonify` z Flasku.

Výpis 4.1 Formát zápisu dát v JSON. Príklad zadania jednej našej úlohy vo formáte JSON.

```
1 {
2     "id_ukolu": 1,
3     "id_uzivatele": 1,
4     "reseni": "SELECT * from studenti;",
5     "zadani": "Vypi\u0161te v\u0161echny studenty."
6 }
```

4.6 HTTP

HTTP je protokol, ktorý sa používa na komunikáciu medzi programami cez World Wide Web. Webové prehliadače a servery ho využívajú na dvojcestnú komunikáciu. Každý webový zdroj má unifikovaný identifikátor (URI), ktorý umožňuje prístup k nim. HTTP transakcia sa skladá z požiadavky od klienta a odpovede od servera. Existuje niekoľko rôznych metód HTTP požiadaviek, ktoré hovoria serveru, akú akciu má vykonať. [9]

HTTP metódy, ktoré používame v našom systéme sú popísané v tabuľke 4.1.

HTTP metódy	Popis
GET	Získať údaje zo servera.
POST	Poslať údaje klienta na server.
DELETE	Odstrániť daný zdroj zo servera.

Tabuľka 4.1 HTTP metódy

Stavový kód HTTP je časťou odpovede servera na klientskú požiadavku a charakterizuje, ako bola táto požiadavka spracovaná - či úspešne, neúspešne alebo s chybou.

Stavový kód	Význam
1xx	Informačné
2xx	Úspešné
3xx	Presmerovanie
4xx	Chyba klienta
5xx	Chyba servera

Tabulka 4.2 Stavové kódy

4.7 Ajax

Ajax je skratka pre Asynchrónny JavaScript a XML a popisuje spôsob, ako webové aplikácie môžu komunikovať s serverom bez potreby načítania alebo obnovenia celej stránky. Táto technika je možná vďaka objektu XMLHttpRequest. [10]

Vo výpise 4.2 je jednoduchý príklad, ako použiť Ajax na vykonanie GET požiadavky na server.

Výpis 4.2 Ukážka GET volania a objektu XMLHttpRequest

```
1     function submitFormData(event) {
2     var email = event.target.email.value;
3     var http = new XMLHttpRequest();
4     var url = getApiEndpoint() + '/registrace/email/' + email
5     ;
6     http.open('GET', url, true);
7     http.onreadystatechange = function() {
8         if (http.readyState == 4) {
9             if (http.status == 200) {
10                successToast.show();
11            } else {
12                errorToast.show();
13            }
14        }
15    };
16    http.send();
17 }
```

Kapitola 5

Implementácia

V tejto kapitole sa podrobne pozrieme na implementáciu našej webovej stránky. Zistíte tu, ako je stránka vytvorená, ako sú rôzne časti prepojené a na čo sa používajú jednotlivé technológie. Ďalej sa pozrieme na štruktúru adresára, zaujímavé funkcie, API a štruktúru databázových tabuliek.

V časti 4.1 poskytujeme prehľad použitých technológií.

5.1 Štruktúra adresára

Adresár bakalárskej práce delíme na frontend a backend. Všetky súbory spojené s frontendom sú umiestnené v zložke `rp-sagova/stranka/frontend/`. Táto zložka obsahuje štyri podpriechinky `images/`, `jswasm/`, `scripts/` a `styles/`. V zložke `images` sú umiestnené všetky obrázky. V zložke `jswasm` sú súbory spojené s implementáciou SQLite databázy v jazyku JavaScript. V zložke `scripts` sú JavaScriptové funkcie. V `styles` sa nachádza súbor s CSS štýlom. Všetky súbory spojené s backendom sú umiestnené v zložke `rp-sagova/stranka/api/`.

- `stranka/`
 - `frontend/`
 - * `images/`
 - * `jswasm/`
 - * `scripts/`
 - * `styles/`
 - * `diskusniforum.html`
 - * `index.html`
 - * `otazky.html`

```
* registrace.html
* sqlhriste.html
* test.html
* ukoly.html
* vlastniukoly.html
- api/
```

5.2 Zaujímavé funkcie

V tejto časti sa budeme venovať rôznym zaujímavým funkciám, ktoré vyžadujú detailnejšiu analýzu a implementáciu.

Tieto funkcie budeme rozoberať podľa jednotlivých stránok, aby sme čo najlepšie pochopili technické detaily.

Teraz sa pozrime na jednotlivé funkcie a ich technickú implementáciu.

5.2.1 SQL hřiště

V tejto časti dokumentácie predstavíme dôležité JavaScript funkcie, ktoré boli implementované pre interakciu a manipuláciu s tabuľkami a údajmi na stránke SQL hřiště. Tieto funkcie umožňujú dynamické vytváranie tlačidiel pre tabuľky, ich zobrazenie a manipuláciu s údajmi v nich.

Použitie týchto funkcií predpokladá vytvorenie a inicializáciu SQLite databázy. Vytvorenie kópie tabuliek z MySQL databázy do lokálnej SQLite JavaScriptovej databázy. (Výpis 5.1)¹

Výpis 5.1 Vloženie tabuliek z MySQL databázy do SQLite databázy

```
1 specificTables.forEach(function (tableName) {
2   var xhr = new XMLHttpRequest();
3   const url = getApiEndpoint() + '/tabulka/' +
4     tableName;
5   xhr.open("GET", url, true);
6   xhr.setRequestHeader("Content-Type", "application/
7     json");
8   xhr.onreadystatechange = function () {
9     if (xhr.readyState == 4) {
10      if (xhr.status == 200) {
11        var response = JSON.parse(xhr.responseText)
12        ;
13      }
14    }
15  }
16 }
```

¹Kód pre prácu s SQLite databázou vychádza z dema na stránke <https://sqlite.org/wasm/file/demo-123.js?txt>.

```

10         var table = tableName;
11         var columns = response.column_names.join(",
12             ");
13         var data = response.data;
14         insertData(db, table, columns, data);
15     } else {
16         console.error("Error fetching data for table
17             '" + tableName + "':", xhr.statusText);
18     }
19 }
20 };
    xhr.send();
});

```

Funkcia createNewTable(tableName)

Funkcia createNewTable(tableName) sa používa na vytvorenie nového tlačidla pre tabuľku a pridáva zafarbenie podľa typu tabuľky. Pripraví sa telo HTML tabuľky.

Funkcia removeButton(tableName)

Funkcia removeButton(tableName) sa používa na odstránenie tlačidla pre konkrétnu tabuľku po dopyte DROP TABLE.

Funkcia showDataFromTable(tableName, clicked)

Funkcia showDataFromTable(tableName, clicked) sa používa na zobrazovanie údajov z databázovej tabuľky na stránke. Pri kliknutí sa rozbalí alebo zabalí obsah tabuľky.

Výpis 5.2 Vykonanie PRAGMA dopytu na získanie názvov stĺpcov tabuľky a vytvorenie hlavičky tabuľky.

```

1     db.exec({
2         sql: "PRAGMA table_info(" + tableName + ")",
3         rowMode: "object",
4         callback: function (row) {
5             var cell = document.createElement("th");
6             var cellText = document.createTextNode(row.name);
7             cell.appendChild(cellText);
8             tableRow.appendChild(cell);
9         }.bind({ counter: 0 }),
10    });

```

Výpis 5.3 Vykonanie SELECT dopytu na získanie údajov z tabuľky v SQLite databáze a vytvorenie riadkov HTML tabuľky.

```
1 db.exec({
2   sql: "select * from " + tableName,
3   rowMode: "object",
4   callback: function (row) {
5     var radka = document.createElement("tr");
6     for (var key in row) {
7       var cell = document.createElement("td");
8       var cellText = document.createTextNode(row[
9         key]);
10      cell.appendChild(cellText);
11      radka.appendChild(cell);
12    }
13    tbody.appendChild(radka);
14  }.bind({ counter: 0 }),
15 });
```

Funkcia validate()

Funkcia `validate()` slúži na overenie a vykonanie SQL dopytu zadaného pomocou vstupného poľa. Výsledok sa zobrazí v tabuľke na stránke. Funkcia `handleValidationError(e)` slúži na zachytenie chýb, ktoré sa môžu vyskytnúť pri vykonávaní SQL dopytu.

Používame regulárne výrazy na nahradenie špeciálnych znakov v reťazci pomocou `replace()` funkcie. Funkcia `sanitizeQueryString(queryString)` nahradí všetky výskyty úvodzoviek jednoduchými úvodzovkami. SQL dopyt rozdelíme na slová pomocou funkcie `split()`.

Pri spracovaní SQL dopytu funkcia rozpoznáva nasledujúce typy dopytov:

- CREATE
- INSERT
- DROP
- SELECT
- Iné: Vykoná zadaný SQL dopyt.

Dopyty CREATE a DROP bolo dôležité rozlíšiť, najmä pre zobrazenie zmien v tabuľkách. Pri dopyte CREATE vytvárame nové tlačidlo pre tabuľku a pri dopyte DROP mažeme tlačidlo pre tabuľku.

Používame try-catch block na zachytenie a spracovanie výnimiek, čo pomáha zabezpečiť robustnosť aplikácie v prípade výskytu chyby pri vykonávaní dopytu.

5.2.2 Úkoly

V tejto časti kódu inicializujeme polia a premenné na udržiavanie aktuálneho dopytu, zmenených tabuliek, otázok, správnych dopytov a očakávaných dopytov. Používame funkciu `fetchTasks(url)`, ktorá načíta úlohy z určenej URL adresy.

Funkcia `getTableNameFromQuery(query)` extrahuje názov tabuľky z SQL dopytu.

Vo funkcii `selectQuestion(questionNumber)` sa vykonáva výber otázky na základe zadaného čísla otázky. Následne sa na základe správneho dopytu zisťuje očakávaný výsledok. Ak je typ dopytu `INSERT`, `UPDATE` alebo `DELETE`, tak sa vykoná duplicita tabuľky s premenovaním pomocou funkcie `createSpecialTable(tableName, newTableName, query)` 5.4, kde sa vytvorí nová tabuľka a priradí sa do nej nový názov tabuľky. Očakávaný výsledok potom bude `SELECT` dopyt z tejto novej tabuľky. Tento očakávaný výsledok sa zobrazí pomocou funkcie `showExpectedResult(_taskNumber, query)`.

Výpis 5.4 Vytvorenie duplicity tabuľky s premenovaním

```
1     function createSpecialTable(tableName, newTableName,
2         query) {
3         db.exec({
4             sql: "DROP TABLE IF EXISTS " + newTableName + ";"
5         },
6         db.exec({
7             sql: "CREATE TABLE " + newTableName + " AS SELECT
8                 * FROM " + tableName + ";"
9         });
10        db.exec({
11            sql: query,
12        });
13    }
```

Funkcia `submitAnswer()` odosiela odpoveď používateľa, porovnáva očakávané a skutočné výsledky pre `SELECT` dopyty a zobrazuje výsledok.

5.2.3 Diskusní fórum

Funkcia `showArticles(query)` slúži na asynchrónne zobrazenie príspevkov v tabuľke s oneskorením pomocou funkcie `queryAndFillTable(tableElementName, query, showThead)`. Toto oneskorenie je dosiahnuté pomocou funkcie `delay()`, aby sme umožnili načítanie údajov z MySQL do SQLite a vykonanie dopytu až po načítaní stránky.

Platné dopyty sú preddefinované a aplikácia overuje správnosť zadaného dopytu. Funkcia `handleSelect()` je zodpovedná za spracovanie vstupu od používateľa vo forme SQL dopytu `SELECT` a následné zobrazenie príspevkov v tabuľke. Ak je zadaný dopyt správny, tabuľka sa aktualizuje podľa vstupu, ale v prípade nesprávneho dopytu používateľ dostane informáciu o chybe.

Podobne fungujú aj funkcie `handleOrderByDesc()`, `handleOrderByAsc()` a `handleInsert()`, ktoré spracovávajú vstupy od používateľa a umožňujú interakciu s databázou podľa zadaných SQL dopytov. Tieto funkcie povoľujú tlačidlá, ak sú zadané správne dopyty.

5.2.4 Test

Vytvorili sme tri notifikácie, `successToast`, `errorToast` a `addToast`, pomocou Bootstrapu. `successToast` slúži na oznamovanie úspešných udalostí a `errorToast` slúži na oznamovanie chybových udalostí. Chybové udalosti sú dve: neúspešné prihlásenie a neúspešná registrácia.

Kód

V tejto časti používateľ zadá kód, ktorý reprezentuje sadu úloh. Táto časť kódu funguje podobne ako časť kódu pre úlohy 5.2.2. Vo funkcii `fetchTasks(url)` používame iný koncový bod API. Ak je kód platný, preniesie sa na stránku s otázkami, inak sa zobrazí chybová hláška.

Prihlásenie

Funkcia `login(email, heslo)` je definovaná v rámci Flask aplikácie, ktorá umožňuje používateľom prihlásiť sa pomocou e-mailu a hesla. Funkcia používa MySQL databázu na vyhľadávanie používateľov podľa ich e-mailu a na overenie zhody hesiel. Pre zabezpečenie hesiel sa používa `bcrypt`, čo znamená, že heslá sú zašifrované pred uložením do databázy.

Registrácia

Na odoslanie e-mailov používateľom sa využíva `Flask-Mail`, ktorý umožňuje odosielanie e-mailových správ. Taktiež používame `Gmail SMTP server`. Po úspešnej registrácii je používateľovi odoslané e-mailom vygenerované heslo. V tejto chvíli nie je možné zmeniť heslo, ale nebolo by problém to dorobiť.

5.3 API

API je realizované frameworkom Flask a umožňuje komunikáciu s databázou. Rozhranie API implementuje niekoľko koncových bodov, ktoré umožňujú interakciu s rôznymi aspektmi aplikácie:

- **/prihlaseni/email/<string:email>/heslo/<string:heslo>**
Umožňuje používateľom prihlásiť sa do systému. Overuje identifikačné údaje a v prípade úspešnej autentifikácie vráti ID používateľa. Ak autentifikácia zlyhá, vráti "Wrong password" alebo "User does not exist". Ak autentifikácia prebehne úspešne, tak vráti JSON objekt obsahujúci identifikátor používateľa a HTTP kód 200.
- **/registrace/email/<string:email>**
Slúži na registráciu nových používateľov do systému. Automaticky generuje heslo a odosiela ho na zadanú e-mailovú adresu. Pri volaní koncového bodu sa vráti e-mailová adresa, na ktorú bolo úspešne odoslané vygenerované heslo.
- **/ukoly/id_uzivatele/<int:id_uzivatele>**
Poskytuje zoznam úloh priradených konkrétnemu používateľovi. Vracia zoznam úloh priradených konkrétnemu používateľovi vo formáte JSON.
- **/pridejUkol/id_uzivatele/<int:id_uzivatele>**
Umožňuje prídanie novej úlohy do systému a jej priradenie k danému používateľovi. Po úspešnom prídaní úlohy vráti status "OK".
- **/smazUkol/id_ukolu/<int:id_ukolu>**
Odstraňuje úlohu z databázy. Po úspešnom odstránení vráti status "OK".
- **/vytvorKod/kod/<string:kod>**
Vytvára nové kódy a priradzuje im konkrétne úlohy. Po úspešnom vytvorení kódu a priradení úloh vráti status "OK". Ak kód už existuje, vráti "Code already exists".
- **/pozriKod/kod/<string:kod>**
Poskytuje informácie o konkrétnom kóde vo formáte JSON. Ak kód neexistuje, vráti prázdny zoznam.
- **/testotazky/kod/<string:kod>**
Poskytuje testovacie otázky priradené kódu vo formáte JSON. Ak kód neexistuje, vráti prázdny zoznam.

- **/tabulka/<string:tabulka>**

Služi na získanie dát z konkrétnej tabuľky v databáze a umožňuje prístup k údajom uloženým v danej tabuľke vo formáte JSON. Ak nastane chyba, vráti správu o chybe.

Výpis 5.5 Ukážka kódu napísaná vo Flask frameworku. Získanie testovacích otázok priradených ku kódu.

```
1 @app.route("/testotazky/kod/<string:kod>", methods=['GET '
2     ])
3 def testotazky(kod):
4     cursor = get_mysql_connection().cursor()
5     cursor.execute("""
6         SELECT ukoly.zadani, ukoly.reseni
7         FROM kody
8         JOIN kod_ukol ON kody.id_kodu = kod_ukol.id_kodu
9         JOIN ukoly ON kod_ukol.id_ukolu = ukoly.id_ukolu
10        WHERE kody.kod = %s
11        """, (kod,))
12    data = cursor.fetchall()
13    cursor.close()
14    if data:
15        response = [{"zadani": row[0], "reseni": row[1]}
16                    for row in data]
17        return jsonify(response), 200
18    else:
19        return jsonify([]), 404
```

Výpis 5.6 Testovacie otázky priradené kódu **sabina** vo formáte JSON

```
1 [{"reseni": "SELECT * from nemocnice where
2     typ_hospitalizace = 'operace';",
3     "zadani": "Vypi\u0161te v\u0161echny pacienty, kte\u0159\
4     u00ed byli na operaci."},
5     {"reseni": "DELETE FROM skola WHERE nazev = 'Gymnazium
6     Zilina';",
7     "zadani": "Vyma\u017ete \u0161kolu s n\u00e1zvom Gymnazium
8     Zilina."},
9     {"reseni": "SELECT jmeno from fitness where datum_treningu
10    between '2023-07-01' and '2023-10-01'",
11    "zadani": "Vypi\u0161te jm\u00e9na lid\u00ed, kte\u0159\
12    u00ed cvi\u010dili mezi 1. \u010dervencem a 1. \u0159\
13    u00edjnem."}]
```

5.4 Štruktúra databázových tabuliek

Okrem tabuliek určených na precvičovanie SQL dopytov, ktoré sú na obrázku 3.1, obsahuje databáza tabuľky, ktoré sú vytvorené pomocou nasledujúcich SQL dopytov.

Výpis 5.7 Tabuľka *kody*

```
1 CREATE TABLE `kody` (  
2   `id_kodu` int(255) NOT NULL AUTO_INCREMENT,  
3   `kod` varchar(255) NOT NULL,  
4   `id_uzivatele` int(255) NOT NULL,  
5   PRIMARY KEY (`id_kodu`)  
6 )
```

Výpis 5.8 Tabuľka *kod_ukol*

```
1 CREATE TABLE `kod_ukol` (  
2   `id_kodu` int(255) NOT NULL,  
3   `id_ukolu` int(255) NOT NULL,  
4   KEY `id_kodu` (`id_kodu`),  
5   KEY `id_ukolu` (`id_ukolu`),  
6   CONSTRAINT `kod_ukol_ibfk_1` FOREIGN KEY (`id_kodu`)  
7     REFERENCES `kody` (`id_kodu`),  
8   CONSTRAINT `kod_ukol_ibfk_2` FOREIGN KEY (`id_ukolu`)  
9     REFERENCES `ukoly` (`id_ukolu`)  
10 )
```

Výpis 5.9 Tabuľka *prispevky*

```
1 CREATE TABLE `prispevky` (  
2   `autor` varchar(255) NOT NULL,  
3   `datum` varchar(255) NOT NULL,  
4   `text` varchar(255) NOT NULL  
5 )
```

Výpis 5.10 Tabuška *ukoly*

```
1 CREATE TABLE `ukoly` (  
2   `id_ukolu` int(255) NOT NULL AUTO_INCREMENT,  
3   `zadani` varchar(255) NOT NULL,  
4   `reseni` varchar(255) NOT NULL,  
5   `id_uzivatele` int(255) NOT NULL,  
6   PRIMARY KEY (`id_ukolu`)  
7 )
```

Výpis 5.11 Tabuška *uzivatele*

```
1 CREATE TABLE `uzivatele` (  
2   `id_uzivatele` int(255) NOT NULL AUTO_INCREMENT,  
3   `email` varchar(255) NOT NULL,  
4   `heslo` varchar(255) NOT NULL,  
5   PRIMARY KEY (`id_uzivatele`)  
6 )
```

5.5 Požiadavky pre lokálny vývoj

Predpokladá sa funkčný Python 3.11 a bežiaci MySQL databáza s Adminerom.

Pre úspešné lokálne spustenie projektu je nevyhnutné dodržať nasledujúce požiadavky:

1. Naklonujte si repozitár pomocou `git clone`
`https://gitlab.mff.cuni.cz/teaching/nprg045/peskova/rp-sagova.git`.
2. Vytvorte si súbor `.env` v zložke `rp-sagova/stranka/api/`. Súbor bude obsah nasledujúce premenné:

```
MYSQL_HOST=  
MYSQL_USER=  
MYSQL_PASSWORD=  
MYSQL_DB=  
MYSQL_PORT=  
MAIL_SERVER=  
MAIL_PORT=  
MAIL_USERNAME=  
MAIL_PASSWORD=
```

3. Nainštalujte si virtuálne prostredie pomocou príkazu
`python3 -m venv .venv` na Unix/macOS.

Aktivujte si virtuálne prostredie pomocou príkazu
`source .venv/bin/activate` na Unix/macOS.

Nainštalujte si všetky potrebné Python knižnice uvedené v súbore
`requirements.txt` v zložke `rp-sagova/stranka/api/`.
Stačí spustiť príkaz `pip install -r requirements.txt`
vo vašom virtuálnom prostredí.

4. Importujte zálohu tabuliek MySQL databáze `sagovas.sql`, ktorá sa nachádza v zložke `rp-sagova/`, do nástroja Adminer alebo iného podobného nástroja na správu databáz.
5. Na frontende v súbore `api.js` upravte funkciu `getApiEndpoint()` na `localhost:37197`. Taktiež upravte súbor `.env` podľa vašich údajov a podľa súboru `index.py`. Aby fungovala registrácia je potrebný SMTP server s prihlásením.

6. Pustite Flask v složce `rp-sagova/stranka/api/` pomocou príkazu `python3 index.py` na Unix/macOS.

Kapitola 6

Pilotné testovanie stránky

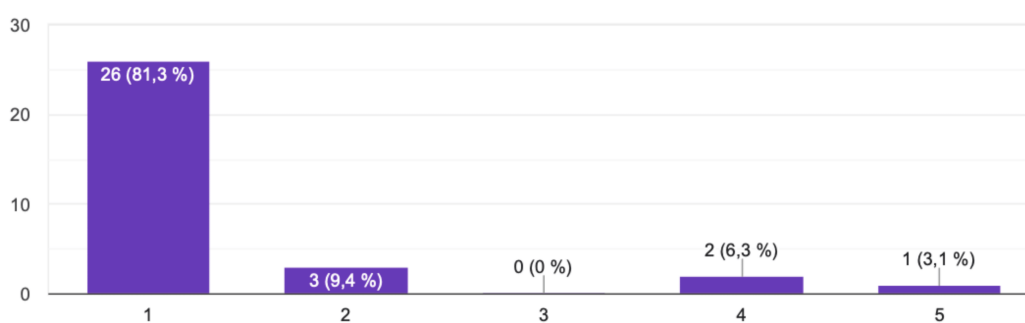
Pilotné testovanie stránky bolo uskutočnené online s cieľom overiť funkčnosť a použiteľnosť aplikácie. Zapojilo sa do neho 32 respondentov. Respondenti boli študenti informatiky na vysokých školách a inžinieri z praxe.

V budúcnosti by bolo vhodné aplikáciu otestovať aj na stredoškolských študentoch v rámci vzdelávacieho procesu.

Dotazník, použitý v rámci testovania, obsahoval 12 otázok a vyplnenie trvalo približne 10 minút. Bol zameraný na získanie názorov a skúseností týkajúcich sa používania interaktívnej webovej aplikácie na precvičovanie SQL dopytov.

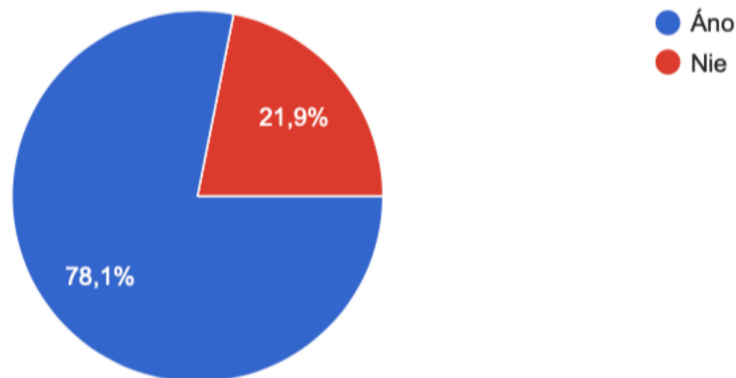
Nasledujúca časť bude zameraná na vyhodnotenie a rozbor výsledkov. Použili sme školskú stupnicu.

1. otázka Úvodná stránka: Ako prehľadne je možné sa na úvodnej stránke zorientovať v ponúkaných možnostiach?



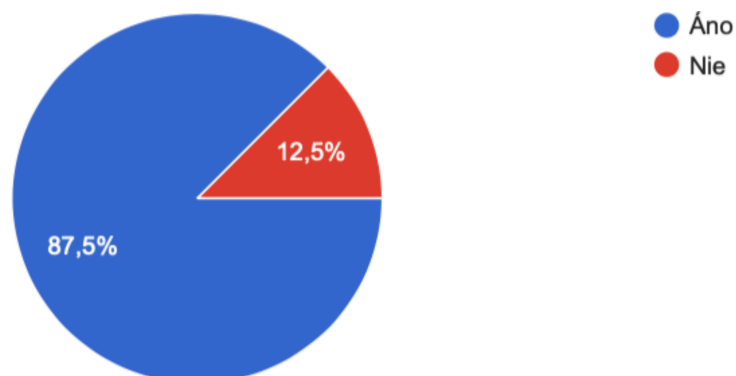
Obrázek 6.1 Odpovede na 1. otázku

2. otázka SQL hřiště: Je možné ľahko identifikovať všetky dostupné tabuľky na stránke?



Obrázek 6.2 Odpovede na 2. otázku

3. otázka SQL hřiště: Je možné jednoducho pridať vlastné tabuľky?



Obrázek 6.3 Odpovede na 3. otázku

4. otázka SQL hřiště: Ako ľahko je možné identifikovať chyby v SQL dopytoch na SQL hřišti?

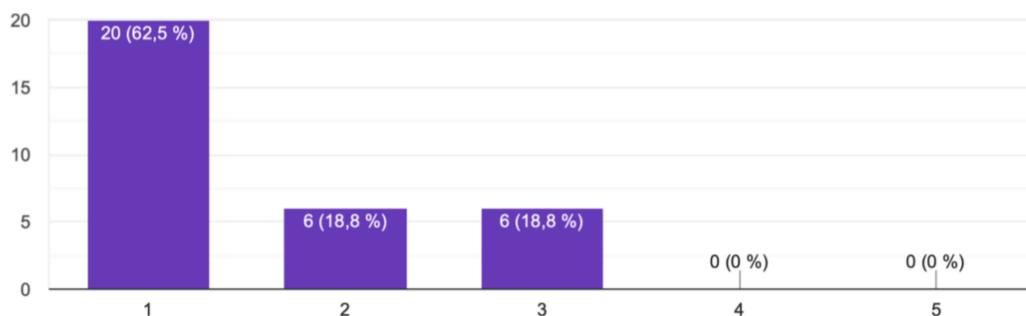
Z odpovedí na túto otázku vyplýva, že väčšina respondentov považuje identifikáciu chýb za celkom zvládnuteľnú, pričom menšina uvádza, že to môže byť občas nejasné.

5. otázka SQL hřiště: Aká je interakcia s tabuľkami v časti SQL hřiště? Je možné jednoducho pracovať s existujúcimi a vytvorenými tabuľkami?

Väčšina respondentov súhlasí, že práca s tabuľkami je jednoduchá a prehľadná. Niekedy majú problém s veľkosťou textu v tabuľkách, hlavne na väčších monitoroch. Navrhujú väčšie textové polia pre zadávanie dopytov či automatické prispôsobenie veľkosti týchto polí podľa potreby. Na mobilných zariadeniach môžu byť tabuľky

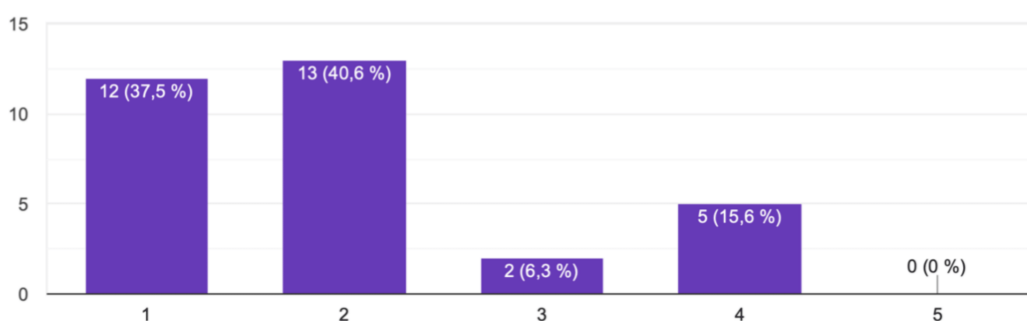
príliš veľké, čo znemožňuje ich pohodlné prezeranie a interakciu s nimi. Respondenti oceňujú prehľadnosť zoznamu tabuliek, avšak niektorí navrhujú jednotnejšie pomenovanie tabuliek.

6. otázka Úkoly: Sú úlohy na precvičovanie SQL dopytov jasne štruktúrované?



Obrázek 6.4 Odpovede na 6. otázku

7. otázka Diskusní fórum: Ako funkčné je použitie diskusného fóra?

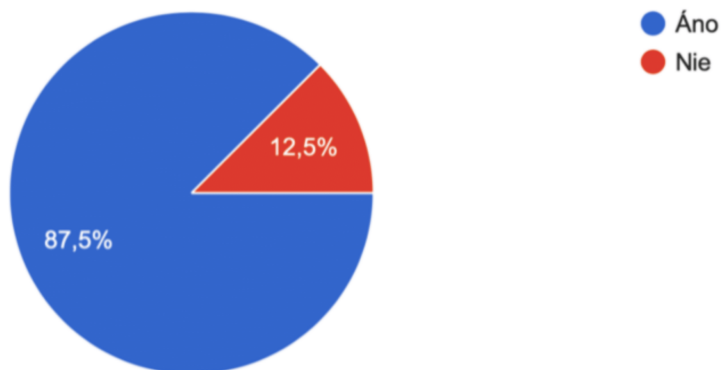


Obrázek 6.5 Odpovede na 7. otázku

8. otázka Test: Aká je užívateľská skúsenosť pri registrácii a prihlásení sa na webovú stránku?

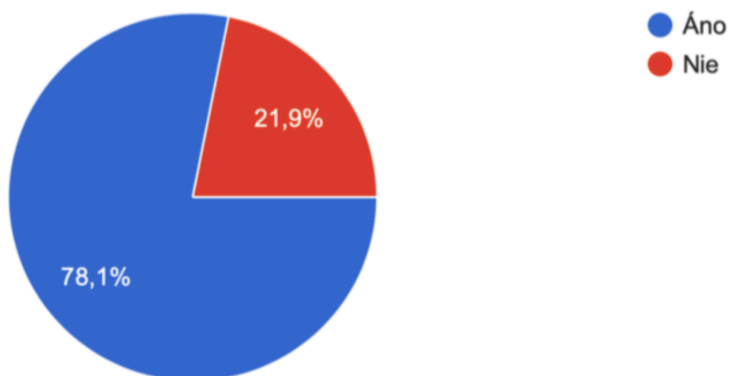
Väčšina respondentov hodnotí skúsenosť s registráciou a prihlásením sa na webovej stránke pozitívne. Označujú ju za dobrú a fungujúcu. Zaznamenané sú aj drobné nedostatky, ako je odhlásenie.

9. otázka Test: Je jednoduché vytvoríť si vlastnú sadu úloh?



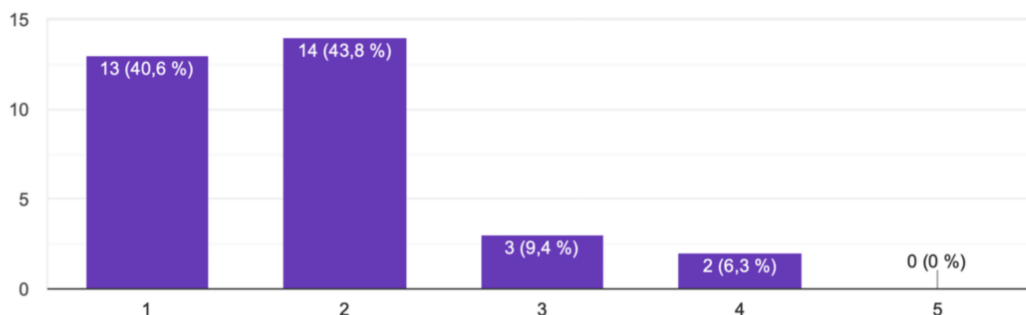
Obrázek 6.6 Odpovede na 9. otázku

10. otázka Test: Je jednoduché otvoriť si vlastnú sadu úloh pomocou kódu?



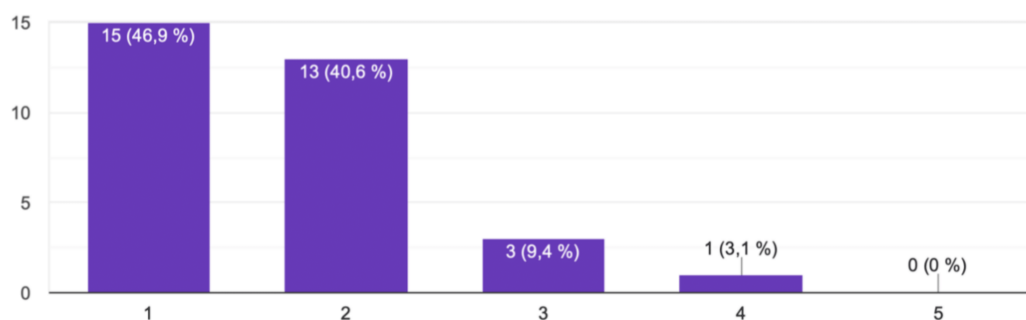
Obrázek 6.7 Odpovede na 10. otázku

11. otázka Ako by ste hodnotili dizajn stránky?



Obrázek 6.8 Odpovede na 11. otázku

12. otázka Ako by ste hodnotili funkčnosť stránky?



Obrázek 6.9 Odpovede na 12. otázku

Záver po výsledkoch hodnotenia ukazuje, že stránka má veľmi dobrý dizajn. Respondenti si vážili prehľadnosti úloh v časti Úkoly, kde je pomocou farieb označené správne a chybné riešenie. Niektorí z nich upozornili na drobné nedostatky, ako je obtiažnejšie nájdenie tlačidla *Tabuľky* alebo nejasnosť významu niektorých prvkov, ako sú uzavreté tabuľky v predvolenom nastavení. Diskusné fórum bolo pre niektorých respondentov obtiažne použiteľné bez prečítania príručky, a niektorí navrhovali jeho vizuálne vylepšenie. Väčšina respondentov vyjadrila spokojnosť s funkčnosťou stránky, hoci niektorí navrhli možné vylepšenia, ako je zobrazenie chýb v časti Úkoly. Celkovo možno povedať, že pilotné testovanie poskytlo užitočnú spätnú väzbu a návrhy na vylepšenie, ktoré môžu prispieť k ďalšiemu rozvoju aplikácie.

Záver

V práci sme predstavili webovú aplikáciu na precvičovanie databázových dopytov. Stránka sa skladá zo štyroch hlavných častí - SQL hřišťe, Úkoly, Diskusní fórum a Test.

Študenti majú možnosť voľného experimentovania s databázou v časti SQL hřišťe. V časti Úkoly majú k dispozícii rôzne úlohy na precvičovanie SQL dopytov. Každá úloha má svoje zadanie a študenti môžu napísať riešenie. Po odoslaní riešenia zistia, či ich riešenie bolo správne alebo nie. V časti Diskusní fórum môžu vidieť praktický príklad použitia SQL dopytov. V časti Test si študenti môžu precvičovať úlohy podľa výberu učiteľa.

Učiteľia majú možnosť vytvárať vlastné úlohy na precvičovanie SQL dopytov a vlastné sady úloh pre študentov.

Vďaka použitiu lokálnej databázy nie je potrebné, aby sa študenti niekde registrovali a aplikácia je teda ľahko použiteľná v školách.

Prebehlo pilotné testovanie, ktoré ukázalo, že je aplikácia funkčná, ale stále by bolo vhodné aplikáciu otestovať vo výučbe so stredoškolskými študentmi.

Na základe spätných hodnotení od respondentov, ktorí testovali aplikáciu, by sa aplikácia mohla rozšíriť o možnosti vizuálneho vylepšenia Diskusného fóra a zobrazenia chýb v časti Úkoly. Taktiež by bolo potrebné spraviť zabezpečenie API.

Seznam použité literatury

- [1] Autorský kolektiv. „Rámcový vzdělávací program pro gymnázia RVP G“. In: (2021).
- [2] Sufyan Bin Uzayr. „Bootstrap (The Ultimate Guide)“. In: (2022). ISBN: 9781032313610.
- [3] Mike Owens. „The Definitive Guide to SQLite“. In: (2006). ISBN: 978159059673.
- [4] *SQLite. About SQLite*. URL: <https://sqlite.org/about.html>. (Cit. 10. 04. 2024).
- [5] Kunal Relan. „Building REST APIs with Flask: Create Python Web Services with MySQL“. In: (2019). ISBN: 9781484250211.
- [6] Miguel Grinberg. „Flask Web Development: Developing Web Applications with Python“. In: (2014). ISBN: 9781449372620.
- [7] Matthias Biehl. „API Architecture: The Big Picture for Building APIs (API University Series)“. In: (2015). ISBN: 9781508676645.
- [8] Arnaud Lauret. „The Design of Web APIs“. In: (říj. 2019). ISBN: 9781617295102.
- [9] David et al. Gourley. „HTTP: The Definitive Guide: The Definitive Guide (Definitive Guides)“. In: (2002). ISBN: 9781508676645.
- [10] John Resig. „Pro JavaScript Techniques“. In: (2006). ISBN: 9781590597279.

Příloha A

Prehľad elektronických príloh

- `rp-sagova` - zložka obsahujúca zdrojové kódy aplikácie
- `exporty` - zložka obsahujúca:
 - `sagovas.sql` zálohu tabuliek
 - `dotaznik.csv` výsledky dotazníka
 - `prirucka.pdf` príručku

A.1 Zoznam skratiek

- **AJAX** - Asynchronous JavaScript and XML
- **API** - Application Programming Interface
- **CSS** - Cascading Style Sheets
- **HTML** - HyperText Markup Language
- **HTTP** - Hypertext Transfer Protocol
- **JSON** - JavaScript Object Notation
- **PHP** - Hypertext Preprocessor
- **REST** - Representational State Transfer
- **RVP** - Rámcový vzdelávací program
- **SQL** - Structured Query Language
- **URI** - Uniform Resource Identifier

- **URL** - Uniform Resource Locator