

**Technická univerzita v Košiciach
Fakulta elektrotechniky a informatiky**

**Metóda ikonicko-textovej formy výuky
hendikepovaných detí na báze webových
technológií**

Bakalárska práca

2022

František Harničár

**Technická univerzita v Košiciach
Fakulta elektrotechniky a informatiky**

**Metóda ikonicko-textovej formy výuky
hendikepovaných detí na báze webových
technológií**

Bakalárska práca

Študijný program: Informatika
Študijný odbor: 9.2.1. Informatika
Školiace pracovisko: Katedra počítačov a informatiky (KPI)
Školiteľ: Ing. Miriama Mattová
Konzultant: Ing. Miriama Mattová

Košice 2022

František Harničár

Abstrakt v SJ

Bakárská práca je zameraná na vývoj webovej aplikácie pre metódu ikonicko-textovej formy výuky hendikepovaných detí. Práca obsahuje analýzu možných variantov riešení systému spoločne s analýzou technológií, s ktorými je možné systém implementovať. Ďalej práca obsahuje návrh obrazoviek a databázy webovej aplikácie. V obsahu návrhu je zahrnutý aj opis rolí učiteľa a študenta. Po návrhu nasleduje samotná implementácia systému vytvorená podľa návrhu. V overení sa testoval systém po používateľskej ako aj po technickej stránke.

Kľúčové slová v SJ

Webová aplikácia, Ikonicko-textová metóda, Systém rolí, Vzdelávanie, Vzdelávací portál

Abstrakt v AJ

The bachelor thesis is focused on the development of a web application for the method of icon-text form of teaching handicapped children. The work contains an analysis of possible variants of system solutions together with an analysis of technologies with which the system is possible to implement. Thesis also contains the design of screens and web application databases. In the design is included description of the roles teacher and student. Then it is followed by the actual implementation of the system created according to the design. It has been tested in verification system on the user as well as on the technical side.

Kľúčové slová v AJ

Web application, Icon-text form, Role system, Education, Educational portal

Bibliografická citácia

HARNIČÁR, František. *Metóda ikonicko-textovej formy výuky hendikepovaných detí na báze webových technológií*. Košice: Technická univerzita v Košiciach, Fakulta elektrotechniky a informatiky, 2022. 41s. Vedúci práce: Ing. Miriama Mattová

TECHNICKÁ UNIVERZITA V KOŠICIACH
FAKULTA ELEKTROTECHNIKY A INFORMATIKY
Katedra počítačov a informatiky

ZADANIE
BAKALÁRSKEJ PRÁCE

Študijný odbor: **Informatika**

Študijný program: **Informatika**

Názov práce:

**Metóda ikonicko-textovej formy výuky hendikepovaných detí na
báze webových technológií**

**Ikonic-text method form of teaching for handicapped childrens via web
technologies**

Študent: **František Harničár**

Školiteľ: **Ing. Miriama Mattová**

Školiace pracovisko: **Katedra počítačov a informatiky**

Konzultant práce: **Ing. Miriama Mattová**

Pracovisko konzultanta: **Katedra počítačov a informatiky**

Pokyny na vypracovanie bakalárskej práce:

1. Analyzovať webové možnosti pre metódu ikonicko-textovej formy výuky.
2. Na základe analýzy navrhnúť webový systém pre danú metódu.
3. Navrhnuť riadenie systému na báze roli študenta a učiteľa s možnosťou vytvárania úloh v roli učiteľa.
4. Implementovať navrhnutý systém v bode 2. a 3.
5. Overiť a vyhodnotiť funkčnosť implementovaného systému.
6. Vypracovať dokumentáciu podľa pokynov školiteľa.

Jazyk, v ktorom sa práca vypracuje: slovenský

Termín pre odovzdanie práce: 27.05.2022

Dátum zadania bakalárskej práce: 29.10.2021



A handwritten signature in blue ink, appearing to be "Liberios Vokorokos".

prof. Ing. Liberios Vokorokos, PhD.
dekan fakulty

Čestné vyhlásenie

Vyhlasujem, že som záverečnú prácu vypracoval(a) samostatne s použitím uvedenej odbornej literatúry.

Košice, 27.5.2022

.....

Vlastnoručný podpis

Podakovanie

Na tomto mieste by som rád poďakoval svojmu vedúcemu práce *Ing. Miriame Mattovej* za jej čas a odborné vedenie počas riešenia mojej záverečnej práce.

Rovnako by som sa rád poďakoval svojim rodičom a priateľom za ich podporu a povzbudzovanie počas celého môjho štúdia.

Obsah

Úvod	1
1 Analýza probelmatiky	3
1.1 Súvisiace riešenia	3
1.2 Spôsoby riešenia	4
1.3 Možnosti jazykovej implementácie	7
1.4 Analytický záver	9
2 Návrh riešenia webovej aplikácie	10
2.1 Všeobecná architektúra systému	10
2.2 Webová časť	12
2.3 Databázová časť	16
3 Implementácia návrhu riešenia webovej aplikácie	19
3.1 Implementácia všeobecnej architektúry systému	19
3.2 Rola učiteľa	21
3.3 Rola študenta	27
3.4 Databázová časť	31
4 Overenie webovej aplikácie	34
4.1 Overenie používateľom	34
4.2 Overenie zaťažovým testom	37
5 Záver	39
Literatúra	40
Zoznam skratiek	42
Zoznam príloh	43

A	Používateľská príručka	44
A.1	Funkcie programov	44
A.1.1	Technické požiadavky	44
A.1.2	Spustenie aplikácie	44
A.2	Použitie programu	45
A.2.1	Prihásenie do roly učiteľa	45
A.2.2	Prihásenie do roly študenta	46
A.2.3	Domovská obrazovka učiteľa	47
A.2.4	Obrazovka vytvorenia testu	48
A.2.5	Obrazovka hodnotenia testu	49
A.2.6	Domovská obrazovka študenta	51
A.2.7	Obrazovka testu študenta	52
A.2.8	Obrazovka po odovzdaní testu študenta	53
B	Systémová príručka	55
B.1	Funkcia programu	55
B.2	Funkcie programov	55
B.2.1	Technické požiadavky na spustenie webovej aplikácie	55
B.3	Preklad programu	55
B.3.1	Spustenie aplikácie	56
B.3.2	Zoznam zdrojových súborov klientskej časti aplikácie	57
B.3.3	Popis jednotlivých komponentov klientskej časti	58
B.3.4	Zoznam zdrojových súborov serverovej časti aplikácie	59
B.3.5	Popis jednotlivých komponentov serverovej časti	59
B.3.6	Vytvorenie databazy	60
C	Formulár z používateľského testovania	61

Zoznam obrázkov

2.1	Všeobecná architektúra webovej aplikacie	11
2.2	Návrh úvodnej obrazovky	12
2.3	Návrh obrazovky študenta	13
2.4	Návrh obrazovky testu študenta	14
2.5	Návrh obrazovky učiteľa	15
2.6	Návrh obrazovky uloženia testu učiteľom	16
2.7	UML diagram návrhu databázy	17
3.1	Všeobecná architektúra webovej aplikacie po implementácií	20
3.2	Úvodná obrazovka	21
3.3	Prihlasovací formulár pre učiteľa	22
3.4	Domovská obrazovka učiteľa	23
3.5	Obrazovka úpravy a vytvárania testu	24
3.6	Formulár uloženia testu	24
3.7	Obrazovka hodnotenia testu	26
3.8	Test konkrétneho riešiteľa	26
3.9	Prihlasovací formulár pre študenta	27
3.10	Domovská obrazovka študenta	28
3.11	Obrazovka testu študenta	29
3.12	Obrazovka hodnotenia riešiteľov po vypracovaní testu	30
3.13	Typ kartičky riešenia úlohy doplnením textu	30
3.14	Typ riešenia úlohy pridaním piktogramu	31
3.15	UML diagram databázy	32
4.1	Graf odpovedí na otázku: Bolo pre vás jednoduché orientovať sa v aplikácii?	35
4.2	Graf odpovedí na otázku: Prostredie aplikácie na vás pôsobilo príjemne?	36
4.3	Znázornenie načítania testu	38

A.1	Úvodná obrazovka po sputení aplikácie	45
A.2	Prihlasovací formulár	46
A.3	Prihlasovací formulár	46
A.4	Domovská obrazovka učiteľa	47
A.5	Obrazovka vytvorenia testu	48
A.6	Formulár uloženia testu	49
A.7	Obrazovka hodnotenia testu	50
A.8	Obrazovka hodnotenia konkrétneho testu	50
A.9	Domovská obrazovka študenta	51
A.10	Obrazovka testu študenta	52
A.11	Typ otázky: Doplniť obrázok	53
A.12	Obrazovka po odovzdaní testu študenta	53
C.1	Scenár formulára	61
C.2	Otázka: Bolo pre vás jednoduché orientovať sa v aplikácii?	61
C.3	Otázka: Prostredie aplikácie na vás pôsobilo príjemne?	62
C.4	Otázka: Mali ste problém s niektorým z bodov scenára? Ak áno, uvedte číslo bodu a vzniknutý problém.	62
C.5	Otázka: Existuje funkcionálna, ktorú by ste do aplikácie pridali? Ak áno, uveďte akú?	62
C.6	Otázka: Je niečo, čo by ste na aplikácii zmenili?	62

Úvod

Je pozoruhodné sledovať akým rýchlym tempom sa technológie okolo nás vyvíjajú, a kde sme sa pri pohľade o 100 rokov späť posunuli. V porovnaní s minulosťou je možné sledovať v oblasti zdravotníctva zlepšenie kvality života, v oblasti komunikácie je možné sa spojiť s človekom v reálnom čase, aj keď sa nachádza na opačnej strane zemegule, v oblasti vzdelávania je možné pozorovať značné zefektívnenie a zjednodušenie vyučovaného učiva využitím rôznych moderných technológií. V každej z týchto, ale aj iných oblastí je stále čo zlepšovať a vyvíjať efektívnejšie riešenia.

Táto bakalárska práca sa z tohto dôvodu zameriava na zefektívnenie ikonicko - textovej metódy výuky hendikepovaných detí využitím webových technológií. Vytvorí sa webová aplikácia zložená z dvoch rolí. Jedná z rolí bude rola učiteľa. Tá bude primárne slúžiť na vytváranie testov. Testy budú vytvárané na základe ikonicko-textovej metódy. Druhou z rolí bude rola študenta. Princípom tejto role bude riešenie testov vytvorených rolou učiteľa.

Implementáciou tejto metódy tak, aby bola dostupná z webového prehliadača sa ušetrí nutnosť inštalácie aplikácie do zariadenia používateľa. Teda toto riešenie by malo byť prístupné všetkým používateľom, ktorí majú počítač a pripojenie k internetu.

Formulácia úlohy

Prvou úlohou v rámci bakalárskej práce bude analyzovať webové možnosti pre metódu ikonicko-textovej formy výuky. V rámci druhej úlohy na základe analýzy navrhnuť webový systém pre danú metódu a riadenie systému na báze roli študenta a učiteľa s možnosťou vytvárania úloh v roli učiteľa.

Na základe vypracovaného návrhu je následne potrebné implementovať daný systém. Následne je potrebné vytvorenú implementáciu overiť a vyhodnotiť funkčnosť implementovaného systému.

Poslednou úlohou tejto bakalárskej práce bude vypracovať dokumentáciu podľa pokynov školiteľa.

1 Analýza problematiky

Žijeme v dobe kedy je osobný kontakt veľkou vzácnosťou. Vzdelávanie detí sa presunulo zo školských lavíc do detských izieb. Touto nečakanou udalosťou sa prišlo na to, že existuje mnoho foriem výuky, ktoré nie sú zdigitalizované a je možné ich vykonávať iba osobným kontaktom učiteľa a študenta, alebo sú zdigitalizované, ale software je zastaralý a ťažko širiteľný. Aj táto forma výuky je toho jasným príkladom. Táto kapitola sa preto zameria na analýzu súvisiacich riešení a následne na analýzu možných riešení.

1.1 Súvisiace riešenia

Pri hľadaní existujúcich riešení sa použil vyhľadávač Google a internetový portál IEEE Xplore. Riešení tejto problematiky sa nenašlo toľko, ako sa predpokladalo. Nasledujúce riešenia opisujú podobnú problematiku.

Poloautomatické testovanie študentského softvéru pod Unix(R). Je opísaná metóda na vytvorenie bezpečného prostredia hierarchickej skupiny v Unix(R), kde má fakulta voľný prístup k súborom študentov, ale v ktorom študenti nemajú prístup ani k súborom fakulty, ani k súborom iných študentov. Prostredie sa používa na implementáciu poloautomatického testovacieho systému pre študentský softvér. Testovanie sa používa na hodnotenie aj na praktickú pomoc študentom. Testy generujú trvalý záznam výsledkov s časovou pečiatkou. Poskytujú sa vzorky typického programového priradenia, testovací skript v Perle a ukážky záznamu pre vyhovujúce a nevyhovujúce programy. [1]

Návrh systému kolaboratívneho vzdelávacieho portálu na podporu komunit učenia sa učiteľov. Tento článok popisuje návrh systému kolaboratívneho vzdelávacieho portálu, ktorý bol vytvorený ako alternatívne riešenie problémov s distribúciou výučby, ktoré sa vyskytli medzi učiteľskou komunitou. Systém je navrhnutý pomocou metódy výučby vyučovacej hodiny, ktorá sa v súčasnosti bežne používa pri rozvoji vyučovacích aktivít. Účelom dizajnu systému kolaboratívneho vzdelávacieho portálu je vytvoriť médium, ktoré môže učiteľom uľahčiť

vytváranie plánov hodín nepretržite s ostatnými kolegami učiteľmi v rovnakej komunite. [2]

Aplikácia pre podporu kolaboratívnej práce so študentmi. Tato práca popisuje návrh aplikácie pre podporu kolaboratívnej práce so študentami. Podrobne opisuje vlastnosti technológií Web 2.0 a Web 3.0 a následne ich porovnáva. Ďalej sa práca zameriava na vyhodnotenie vhodných technológií potrebných na vývoj aplikácie. Práca pokračuje opisom praktického návrhu a riešenej aplikácie vhodnej pre kolaboratívnu prácu medzi študentami a pedagógmi. Následne aplikácia je navrhnutá tak, že umožňuje študentom kolaboratívne spolupracovať s využitím prvkov moderného internetového fóra a wikipédie.[3]

Vzdelávací portál pre technológiu kolaboratívnej práce vo virtuálnom priestore. Vzdelávací portál je koncept navrhnutý tak, aby pomohol samoštúdiu, v ktorom sa kladie dôraz na problémové učenie. Portál zhromažďuje informácie z mnohých zdrojov a usporiada obsah tak, aby podporoval samoštúdium. V rámci tohto výskumu sa vyvinul webový vzdelávací portál, kde majú používatelia prístup k obsahu odkiaľkoľvek a kedykoľvek. Ako obsah sa zvolil využitie a vývoj videokonferenčného balíka. Portál je navrhnutý tak, aby poskytoval interaktívny obsah aj programovacie cvičenia. Študenti môžu ľahko získať základné vedomosti o tom, ako sa vyvíja balík videokonferencií. Inými slovami, pre študentov inžinierstva poskytujeme vzdelávacie centrum s vlastným prístupom. [4]

1.2 Spôsobý riešenia

Táto podkapitola sa bude zaoberať akými rôznymi spôsobmi by bolo možné vzniknutú problematiku vyriešiť. Navrhne sa viacero variantov a na základe výhod a nevýhod sa vyberie najoptimálnejšie riešenie, ktoré sa následne navrhne a implementuje v nasledujúcich kapitolách.

Variant 1

Jednou z možností pri riešení tejto problematiky je vytvoriť webovú aplikáciu, ktorá by bola verejne dostupná a nevyžadovala by si žiadne prihlásenie, avšak iba z role študenta. Po navštívení webovej aplikácie by sa používateľ dostal na úvodnú obrazovku. Na obrazovke by mal zobrazené dve tlačidlá. Jedno tlačidlo by bolo pre vstup do role učiteľa a druhé do role študenta. Po stlačení na tlačidlo role študenta by používateľ musel zadať meno pre vstup do aplikácie. S týmto menom by figuroval pri vykonávaní všetkých úkonov v aplikácii. Po zadaní mena, by sa používateľ dostal priamo k všetkým dostupným testom a mohol by ich vypraco-

vať. Po vypracovaní by sa používateľovi zobrazila tabuľka, kde by mal zobrazené všetky hodnotenia testov spoločne s jeho vlastným.

Ak používateľ chce v aplikácií využiť rolu učiteľa musí sa prihlásiť. Učiteľ by navštívil webovú aplikáciu rovnako ako študent. Následne by sa kliknutím na tlačidlo učiteľa a prihlásením dostal do svojho prostredia. V tomto prostredí by učiteľ mohol testy vytvárať, editovať alebo aj odstraňovať. Pri vytváraní testov by mal na výber zo širokej škály úloh obsiahnuté v projekte s názvom **Virtuálne-realitné technológie súčasnosti a vzdelávanie postihnutých ľudí**[5]. Učiteľ by mal ešte možnosť prechádzať si výsledkami testov. Každý test by mal svoj zoznam mien aj s výsledným hodnotením.

Výhody:

- Prihlásenie z pohľadu študenta je jednoduché,
- študent si nepotrebuje zapamätávať prihlasovacie údaje,
- webovú aplikáciu je možné používať v škole, nie je obmedzená na použitie z domu,
- učiteľ ma prehľad o správnosti vypracovaných testov.

Nevýhody:

- Učiteľ nemá presné údaje o tom, kto konkrétny test vypracoval,
- meno, pod ktorým bol test vypracovaný môže byť neuvedené alebo vymyslené.

Variant 2

Ďalšou možnosťou riešenia tejto problematiky je vytvoriť podobne riešenie ako Variant 1 s tým rozdielom, že síce webová aplikácia by bola verejne dostupná, ale jej používanie by bolo podmienené prihlásením. Na to, aby sa študent vedel dostať k vytvoreným testom, musel by sa prihlásiť pod svojím prihlasovacím menom a heslom. Po prihlásení by aplikácia vyzerala veľmi podobne.

Z hľadiska role učiteľa, by sa aplikácia veľmi nezmenila. Prihlasoval by sa rovnako, ako pri variante 1, s tým rozdielom, že prihlasovanie by prebiehalo hneď pri navštívení stránky. Prihlasovanie do oboch rolí by bolo realizované pomocou jedného prihlasovacieho formulára. Prihlasovanie by sa navrhlo tak, aby bolo schopné rozlíšiť, kto sa do systému prihlasuje. Pokiaľ by prihlasovacie údaje boli

zadané správne a patrili by študentovi, používateľ by sa prihlásil do role študenta. Ale naopak, ak by prihlasovacie údaje boli správne a patrili by učiteľovi, používateľ by sa prihlásil do role učiteľa.

Výhody:

- Každý študent má svoj vlastný účet,
- nie je možné vydávať sa za iného študenta,
- učiteľ má prehľad o tom, kto test vypracoval a akú mal úspešnosť,
- webovú aplikáciu je možné používať v škole, nie je obmedzená na použitie z domu.

Nevýhody:

- Prihlásenie z pohľadu študenta je komplikovanejšie,
- študent je nútený si zapamätať svoje prihlasovacie údaje.

Variant 3

Tento spôsob riešenia problematiky by prihlasovanie študenta do webovej aplikácie riešil pomocou používateľovej IP adresy. Študent by pri prvom navštívení tejto webovej aplikácie zadal svoje meno a priezvisko. Tieto údaje by sa následne uložili do databázy spoločne aj s jeho IP adresou, z ktorej sa pripája na server webovej aplikácie. Pri opätovnom navštívení webovej aplikácie by si aplikácia pamätala jeho IP adresu a prihlásila by ho pod jeho menom do systému.

Prihlasovanie učiteľa by bolo realizované klasickým spôsobom mena a hesla. Prihlásenie do roly študenta a učiteľa by bolo oddelené na úvodnej obrazovke aplikácie. Na tejto obrazovke by boli zobrazené dve tlačidlá. Prvé tlačidlo by nieslo nápis Študent a druhé tlačidlo nápis Učiteľ. Po kliknutí na konkrétne tlačidlo by používateľa presmerovalo k jeho spôsobu prihlásenia. Po prihlásení do systému sa študentovi zobrazia testy a bude s nimi môcť interagovať ako vo Variante 2. Po prihlásení do systému bude učiteľ môcť vykonávať úkony ako vo Variante 1.

Výhody:

- Prihlásenie z pohľadu študenta je jednoduché,
- študent si nepotrebuje zapamätávať prihlasovacie údaje,

- učiteľ ma prehľad o tom, kto test vypracoval a akú mal úspešnosť.

Nevýhody:

- Prihlásenie študenta je obmedzené iba na jednu IP adresu,
- pokiaľ by sa aplikácia využívala v škole bola by nepoužiteľná.

1.3 Možnosti jazykovej implementácie

Na realizáciu navrhnutého spôsobu riešenia sa ponúka pestrý zoznam programovacích jazykov. Preto sa táto podkapitola bude zaoberať analýzou a správnym výberom programovacích jazykov a technológií na realizáciu navrhnutého spôsobu riešenia.

Analyza frontend-oveho riesenia

Frontend je v jednoduchosti všetko s čím používateľ interaguje vo webových aplikáciách, preto je veľmi dôležité ho správne navrhnuť. Cieľom navrhnutia správneho frontendu je zabezpečiť, aby používatelia pri otvorení stránky videli informácie vo formáte, ktorý je ľahko čitateľný a relevantný.

Na logické umiestnenie komponentov do používateľovho rozhrania je možné zvážiť jazyk HTML. Naopak, pri vytváraní dizajnu a používateľsky prívetivom prostredí je možné zvážiť jazyky CSS a Javascript.

Je veľkou pravdepodobnosťou, že webová aplikácia bude zobrazovaná na rôznych veľkých obrazovkách. Preto je potrebné webovú aplikáciu vyvinúť responzívne, aby ju bolo možné zobraziť čitateľne a používateľsky príjemne pri použití akejkoľvek veľkosti výstupnej obrazovky. Tento vzniknutý problém je možné jednoduchšie riešiť frameworkom s názvom Bootstrap. Bootstrap je obrovská zbierka praktických, opakovane použiteľných kúskov kódu napísaných v jazykoch HTML, CSS a JavaScript. [6]

Pri ďalšom postupe pri vytváraní frontendu je možné uvažovať aj nad použitím frameworkov React alebo Angular. Veľkou výhodou frameworku angular je možnosť dopisovať do HTML kódu napríklad podmienky alebo aj cykly. Ale na druhej strane veľkou nevýhodou použitia angularu pri riešení tejto problematiky je to, že tento framework sa používa zväčša pre väčšie webové aplikácie. Problém spočíva v tom, že pre spojazdnenie a správne fungovanie celého systému je potrebné implementovať rôzne podporné moduly a knižnice, ktoré môžu byť kapacitne náročnejšie ako, keď by implementácia bola riešená napríklad aj frameworkom React.

React disponuje aj ďalšími výhodami, ktoré je možné využiť pri zostavovaní vyššie spomenutého riešenia problematiky. React používa virtuálny DOM.[7] Ten vie webovú aplikáciu výrazne zrýchliť a v konečnom dôsledku aj vytvorí lepšie používateľské prostredie. Ďalším plusom pri použití Reactu je veľké množstvo existujúcej riešenej podobnej problematiky. Tieto riešenia sú verejne dostupné a je možné niektoré ich časti s menšími úpravami aplikovať do tejto problematiky.

Analýza backend-ového riešenia

Backend v jednoduchosti organizuje a ukladá dáta, ako aj zabezpečuje, aby všetko na používateľskej strane fungovalo správne. Aj, keď backend neinteraguje s používateľmi priamo, hrá nenahraditeľnú úlohu v zákulisí pridávaním kľúčových funkcií na stránku. Bez dobre navrhnutého backendu nebude frontend správne fungovať. Pri implementácii tejto problematiky by mohol backend slúžiť napríklad na spracovanie prihlasovania používateľov alebo aj na vyhodnocovanie správnosti testu prípadne ukladanie výsledkov do databázy. Jednoducho na všetko, čo je používateľovi neviditeľné.

Na vytvorenie backendu je možné zvážiť runtime prostredie Node.js. Toto prostredie umožňuje spúšťať program napísaný v jazyku Javascript. [8] Veľkou výhodou je, preto použitie rovnakého programovacieho jazyka aj na frontend, ako aj na backend. To umožňuje jednoduchšie spravovanie webovej aplikácie, ako aj opravu vzniknutých chýb. Použitím týchto technológií by bolo možné implementovať všetky požiadavky, ktoré boli spomenuté v kapitole 1.2.

Analýza databázového riešenia

Pri používaní navrhnutého riešenia webovej aplikácie bude nutné vzniknuté dáta niekde uschovávať. Bude potrebné ukladať napríklad prihlasovacie dáta alebo aj mená študentov spoločne s ich výsledkami testov, ktoré vypracovali. Na ukladanie týchto dát je možné zvážiť databázový systém PostgreSQL.

Výhodou použitia tohto databázového systému je jeho podpora súborového typu JSON. [9] Tento typ súboru by mohol byť využívaný pri ukladaní vytváraných testov učiteľom. Ďalšou výhodou použitia databázového systému PostgreSQL je jeho podpora vyššie spomenutého backendového riešenia.

1.4 Analytický záver

Na základe analýzy možných riešení, možnosti jazykovej implementácie a sumarizácie výhod a nevýhod sa táto práca zameria na návrh riešenia podľa Variantu 1 pri použití zvažovaných implementačných technológií. Riešenie pomocou tohto variantu je používateľský najviac praktické a je možné ho nasadiť prakticky všade, nie je obmedzený iba na použitie doma, ako bol napríklad Variant 3.

2 Návrh riešenia webovej aplikácie

Pri návrhu webovej aplikácie je potrebné klásť dôraz na dizajn. Keďže webová aplikácia bude určená pre výučbu hendikepovaných detí je potrebný dizajn aplikácie navrhnuť, čo najjednoduchšie, s čo najmenším počtom rušivých podnetov. Je potrebné zabezpečiť, aby študent pri riešení testu ostal sústredený na test, ktorý dostal od svojho učiteľa. Ako bolo spomenuté v predchádzajúcej kapitole, návrh riešenia bude zameraný na Variant 1.

2.1 Všeobecná architektúra systému

Webovú aplikáciu je možné rozdeliť na dve základné časti. Prvá časť je časť webová. V jednoduchosti, je to časť webovej aplikácie, ktorú ma používateľ zobrazenú na svojej obrazovke a vie s ňou interagovať. Táto časť sa ďalej delí na dve podčasti-role:

Učiteľ - používateľ vie použitím tejto role vytvárať úlohy pre rolu študenta.

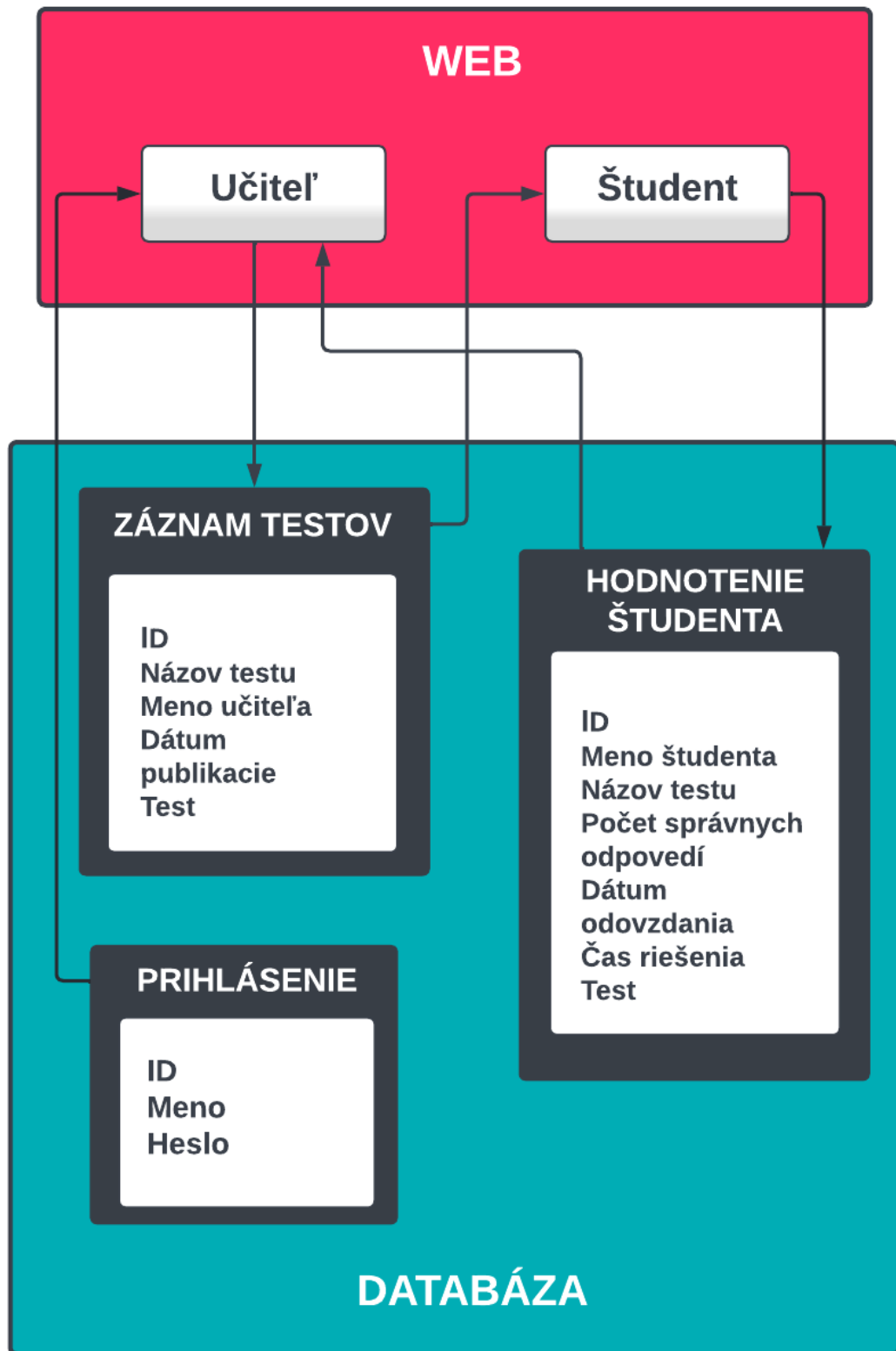
Študent - použitím tejto role vie používateľ vypracovávať úlohy, ktoré mu používateľ zadal z role učiteľa.

Druhá základná časť sa nazýva databáza alebo je ju možné nazvať aj serverová časť. V tejto časti sa nachádzajú všetky uložené údaje, ktoré sú potrebné pre správne fungovanie webovej aplikácie. Používateľ vie na základe roly, v ktorej sa nachádza ukladať alebo čítať údaje z jednotlivých tabuliek databázy. Databáza obsahuje 3 tabuľky:

Záznam testov - v tejto tabuľke su uložené údaje o všetkých úlohách, ako aj samotné úlohy.

Hodnotenie študenta - táto tabuľka obsahuje informácie o vypracovaných úlohách študentov, spoločne aj so samotnými úlohami a odpoveďami študentov.

Prihlásenie - táto tabuľka obsahuje prihlasovacie údaje pre rolu učiteľa.



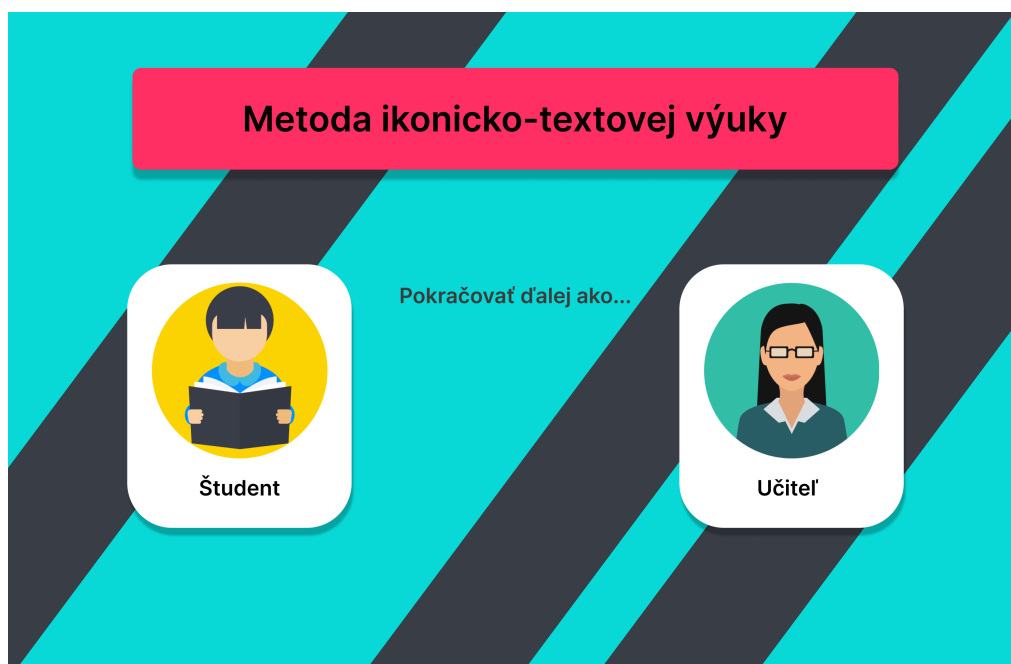
Obr. 2.1: Všeobecná architektúra webovej aplikácie

2.2 Webová časť

Táto časť webovej aplikácie pozostáva hlavne z dizajnu a používateľského zážitku. Táto podkapitola sa bude zaoberať návrhom jednotlivých obrazoviek webovej aplikácie. Je potrebné správne navrhnuť rozloženie rôznych komponentov na obrazovke tak, aby práca s nimi bola intuitívna a jednoduchá.

Úvodná obrazovka

Táto obrazovka by bola prvá, ktorú by používateľ videl pri navštívení webovej aplikácie. Na obrazovke by sa nachádzal názov vzdelávacej metódy a dve tlačidlá na základe, ktorých by si používateľ vedel vybrať rolu, v ktorej by pokračoval ďalej v aplikácii. Po zvolení si svojej role bude používateľ požiadany v prípade role učiteľa o prihlásenie zadaním mena a hesla. V prípade študenta bude používateľ požiadany o zadanie mena, pod ktorým chce figurovať počas používania webovej aplikácie.



Obr. 2.2: Návrh úvodnej obrazovky

Obrazovka študenta

Touto obrazovkou by používateľ vstúpil do prostredia študenta. Na obrazovke by mal študent zobrazené všetky potrebné informácie. Meno, pod ktorým figuruje v aplikácii a zoznam všetkých dostupných testov. Každý test by bol rozlíšený svojim názvom a podobne by bol zobrazený aj dátum, kedy bol test pridaný do systému. Ďalej by test niesol informáciu o tom, kto test zostavil. Tieto testy by študent po kliknutí na zelené tlačidlo "Vypracovať", mohol vyriešiť. Po stlačení tlačidla by študentovi vybehlo okno s upozornením, že ak chce pokračovať na riešenie testu, tak po stlačení tlačidla v danom okne sa spustí test a s ním aj časomiera.



Obr. 2.3: Návrh obrazovky študenta

Obrazovka testu študenta

Po kliknutí študenta na tlačidlo "Vypracovať", sa študentovi zobrazí test, ktorý si vybral. Pozadie testu sa zmení z priehľadného na plnú bielu, aby sa zvýšil kontrast a otázka bola jasne viditeľná. V modrom rámečku je zobrazené umiestnenie zadania úlohy. V tejto časti sa bude nachádzať všetko potrebné pre študenta, aby vedel danú úlohu vyriešiť. Dole pod týmto blokom bude umiestnené miesto na vyplnenie zadanej úlohy v teste. Pokiaľ bude úloh v teste viac, bude mať študent zobrazené, na akej otázke sa nachádza a následne aj navigáciu v zadaných úlohách. Ďalším údajom pre študenta na tejto obrazovke bude čas, za ktorý tieto úlohy vyrieši. Tento čas bude následne učiteľ môcť vidieť pri prezeraní hodno-

tených študentov. Po dokončení všetkých úloh v teste bude môcť študent poslať svoje vyplnené úlohy na hodnotenie.

Hodnotenie bude vykonávať systém, ktorý následne študentovi zobrazí, koľko z koľkých úloh mal správne vyriešených. Študent si následne bude môcť pozrieť aj výsledky ostatných študentov, ktorí riešili tento test. Zobrazí sa mu tabuľka so všetkými výsledkami daného testu spoločne aj s menami riešiteľov a ich dobou riešenia. Údaje by boli zobrazené od tých najlepších riešiteľov, až po tých najhorších na základe správnych odpovedí. Ak by sa počet správnych odpovedí u dvoch študentov zhodoval, vyššie v tabuľke by bol študent s kratším časom riešenia.



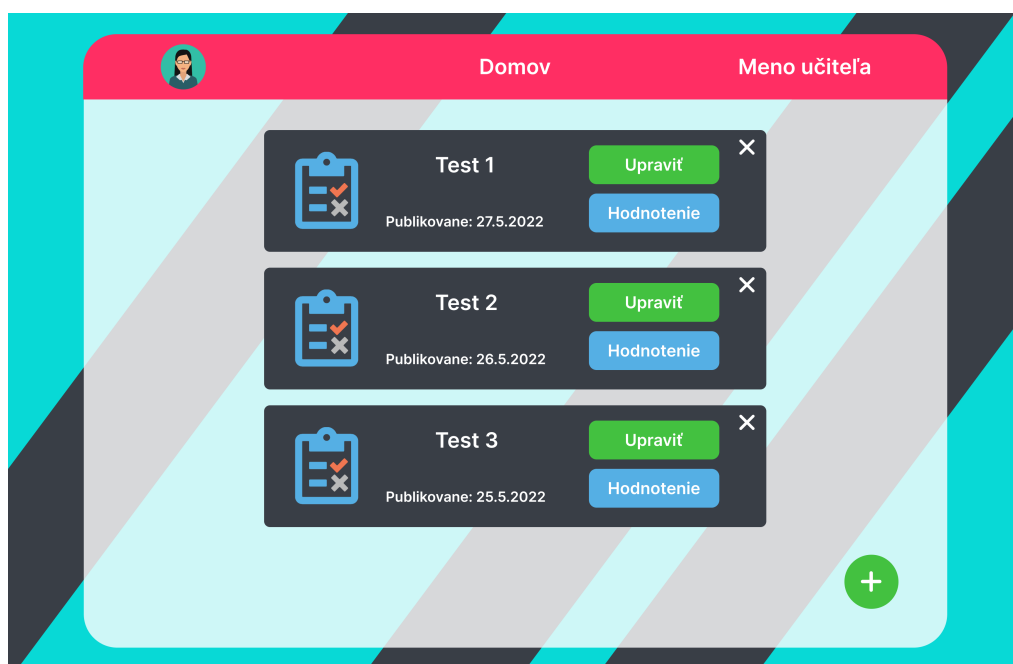
Obr. 2.4: Návrh obrazovky testu študenta

Obrazovka učiteľa

Po úspešnom prihlásení po výbere role učiteľa sa na úvodnej obrazovke používateľ dostane do používateľského prostredia učiteľa. V tomto prostredí bude mať používateľ zobrazené všetky testy, ktoré vytvoril. S týmito testami bude používateľ môcť vykonávať rôzne operácie ako:

- vytvárať nové testy,
- upravovať už vytvorené testy,
- odstraňovať už vytvorené testy,
- prezerať si hodnotenia študentov, ktorí tento test vypracovali.

Pri prezeraní hodnotených testov sa učiteľovi zobrazí tabuľka so všetkými hodnoteniami študentov, ktorí vypracovali zvolený test. Na tabuľke budú zobrazené základné údaje, ako meno, počet správnych odpovedí a čas, za ktorý test študent vypracoval. Ak bude učiteľ chcieť vidieť, ako študent konkrétne odpovedal v každej úlohe, bude si môcť otvoriť vypracovaný test konkrétneho žiaka a vidieť presne, ako študent odpovedal na každú úlohu v teste. Zobrazenie testu bude identické so študentovým zobrazením, s tým rozdielom, že učiteľ už nebude môcť upravovať odpoveď žiaka.



Obr. 2.5: Návrh obrazovky učiteľa

Obrazovka vytvárania testu učiteľom

Do tohto prostredia sa používateľ dostane kliknutím na tlačidlo plus v pravom dolnom rohu zobrazeného na obrázku 2.5. Toto prostredie slúži na vytvorenie testu pre študenta. Rozloženie objektov je podobné ako u študenta na obrázku 2.4. Učiteľ si vie v hornej časti obrazovky zvoliť typy úloh:

- prvý typ úlohy bude prevod piktogramov do písanej vety,
- druhý typ úlohy bude opačný a to prevod písanej vety do piktogramov.

Následne po výbere typu úlohy vie učiteľ úlohu zostaviť. Do spodnej časti obrazovky učiteľ zadá správne riešenie úlohy a môže pokračovať buď v zostavovaní ďalších úloh alebo môže vytváranie testu ukončiť. Ak chce učiteľ test uložiť, musí

mu ešte zadať názov a dátum od kedy chce, aby bol test viditeľný pre študentov. Po stlačení tlačidla uložiť sa test zobrazí na jeho domovskej stránke. Formulár na uloženie testu je navrhnutý na obrázku 2.6.

Pri dodatočnom upravovaní testu učiteľom bude používateľské prostredie rovnaké, ako pri vytváraní testu. Používateľ bude mať rovnaké možnosti. Test bude môcť prerobiť, ako len bude chcieť.

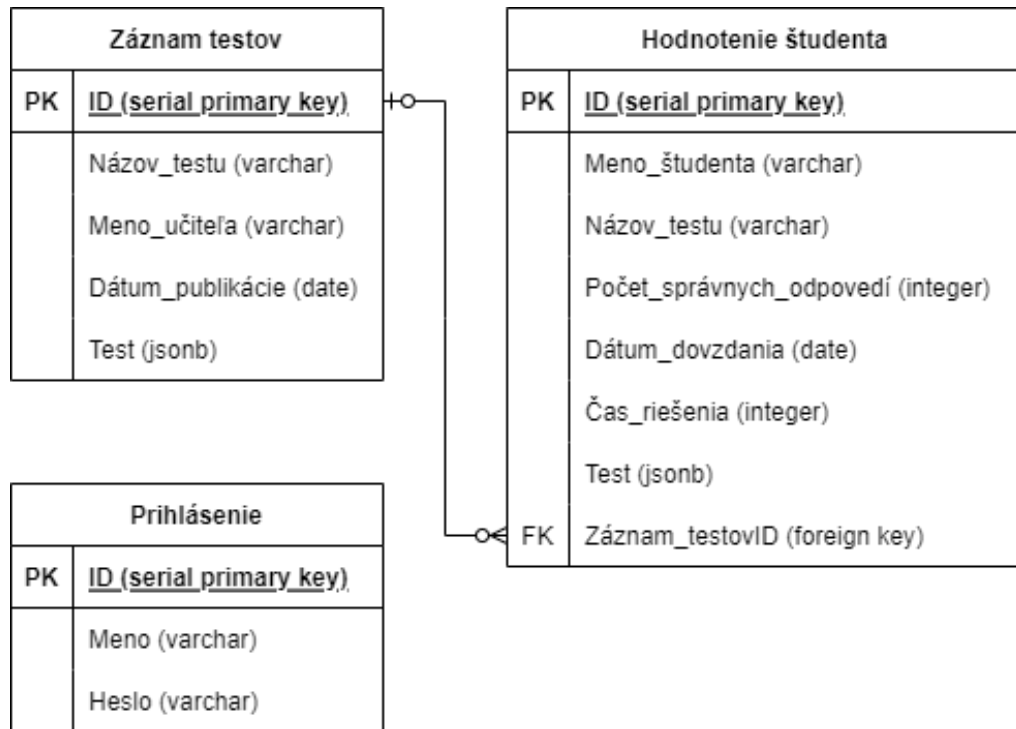


The image shows a user interface for creating a test. At the top, there is a pink header bar containing a user profile icon on the left, the text "Vytváranie testu" in the center, and "Meno učiteľa" on the right. Below the header is a light blue background with a dark grey modal form in the center. The modal form has a title "Názov testu" and a text input field containing "Test 1". Below that is a label "Dátum publikacie" and a date selection field showing "27.5.2022" with a dropdown arrow. At the bottom of the modal is a green button labeled "Uložiť".

Obr. 2.6: Návrh obrazovky uloženia testu učiteľom

2.3 Databázová časť

Ďalšou neodeliteľnou časťou tejto aplikácie je databáza. Všetky navrhnuté úkony v dizajne aplikácie spolupracujú svojím spôsobom s databázou a jej tabuľkami. Pri niektorých úkonoch v aplikácii sa dáta do databázy ukladajú, pri niektorých čítajú alebo aj odstraňujú. Táto podkapitola sa bude venovať návrhu tabuliek databázy.



Obr. 2.7: UML diagram návrhu databázy

Prihlásenie

Tabuľka prihlásenie, ako už názov napovedá bude slúžiť na prihlasovanie učiteľov do systému. Bude obsahovať tri stĺpce. A to jedinečný identifikátor, meno používateľa a heslo. Heslo bude uložené v tvare hashu.

Prihlasovanie by prebiehalo následovne. Používateľ by zadal svoje meno a heslo. Následne, ak by sa zadané meno našlo v tabuľke, zadané heslo by sa previedlo na hash a porovnávalo by sa s uloženým hashom v tabuľke. Ak by sa meno aj hash hesla zhodoval, používateľ by sa prihlásil.

Meno a heslo bude používateľovi priradené, nebude si vedieť vytvoriť svoje prihlasovacie údaje samostatne. Ak sa používaním aplikácie príde na fakt, že tato možnosť je potrebná, táto funkcia sa vytvorí dodatočne.

Záznam testov

Tabuľka záznam testov bude slúžiť na ukladanie testov vytvorených učiteľom. Budú v nej uložené všetky údaje o teste, ako aj samotný test spoločne so správnymi riešeniami úloh. Táto tabuľka by interagovala s rolou študenta aj učiteľa. Učiteľ by do tabuľky údaje vkladal, odstraňoval alebo aj upravoval. Študent by naopak, vedel údaje z tejto tabuľky iba čítať.

Test v tabuľke by bol uložený v typovom tvare súborového systému JSON. Spoločne s testom by bolo uložené aj samotné riešenie každej úlohy zadanej učiteľom. S riešením a znením úlohy by úloha niesla ešte svoj typ.

Hodnotenie študenta

Táto tabuľka obsahuje zo všetkých tabuliek systému najviac informácií. Sú v nej uložené všetky riešenia testov všetkých študentov. Údaje do tejto tabuľky bude vedieť zapísať iba rola študenta pri odovzdaní testu. Akonáhle študent odovzdá svoje riešenie, údaje sa uložia do tabuľky. Následne sa porovná riešenie študenta s riešením, ktoré zdefinoval, ako správne učiteľ a študent dostane výsledok o tom, koľko úloh mal správnych. Tento výsledok sa zápíše podobne ako ostatné údaje do tabuľky.

Učiteľ, ako bolo spomínané vo webovej časti bude mať možnosť pozrieť si hodnotenie študentov spoločne s ich presným riešením zadaných úloh. Táto funkcia bude realizovaná selektom v tabuľke na základe názvu testu. Tým sa vyselektuje iba test, ktorého výsledky chce učiteľ vidieť. Ak učiteľ odstráni test, odstránia sa aj všetky záznamy o riešeníach daného testu.

3 Implementácia návrhu riešenia webovej aplikácie

Táto časť práce je implementovaná na základe návrhu riešenia v kapitole 2. Pri implementácií sa zistilo zopár nedostatkov, na ktoré sa pri návrhu zabudlo alebo ich bolo potrebné obohatiť o zmenu logiky. V tejto kapitole budú opísané všetky zmeny, ktoré nastali v porovnaní s návrhom, ako aj opis samotnej implementácie.

3.1 Implementácia všeobecnej architektúry systému

Táto časť implementácie sa líši iba minimálne od návrhu. Do systému sa implementovalo viacero vzťahov, ktoré sú znázornené šípkami obrázkom 3.1 na nasledujúcej strane. Zásadnou zmenou oproti návrhu je pridanie novej tabuľky do databázy. Táto tabuľka nesie názov Zoznam piktogramov.

Pre jednoduchšie prepojenie frontendu a backendu aplikácie sa využil backendový rámec express.js. [10] Pomocou tohoto rámca boli posielané všetky informácie na frontend a späť.

Učiteľ - ako bolo spomenuté v návrhu, táto rola vie komunikovať so všetkými tabuľkami v databáze. Podobne vie komunikovať aj s rolou študenta ale iba skrz tabuľky databázy.

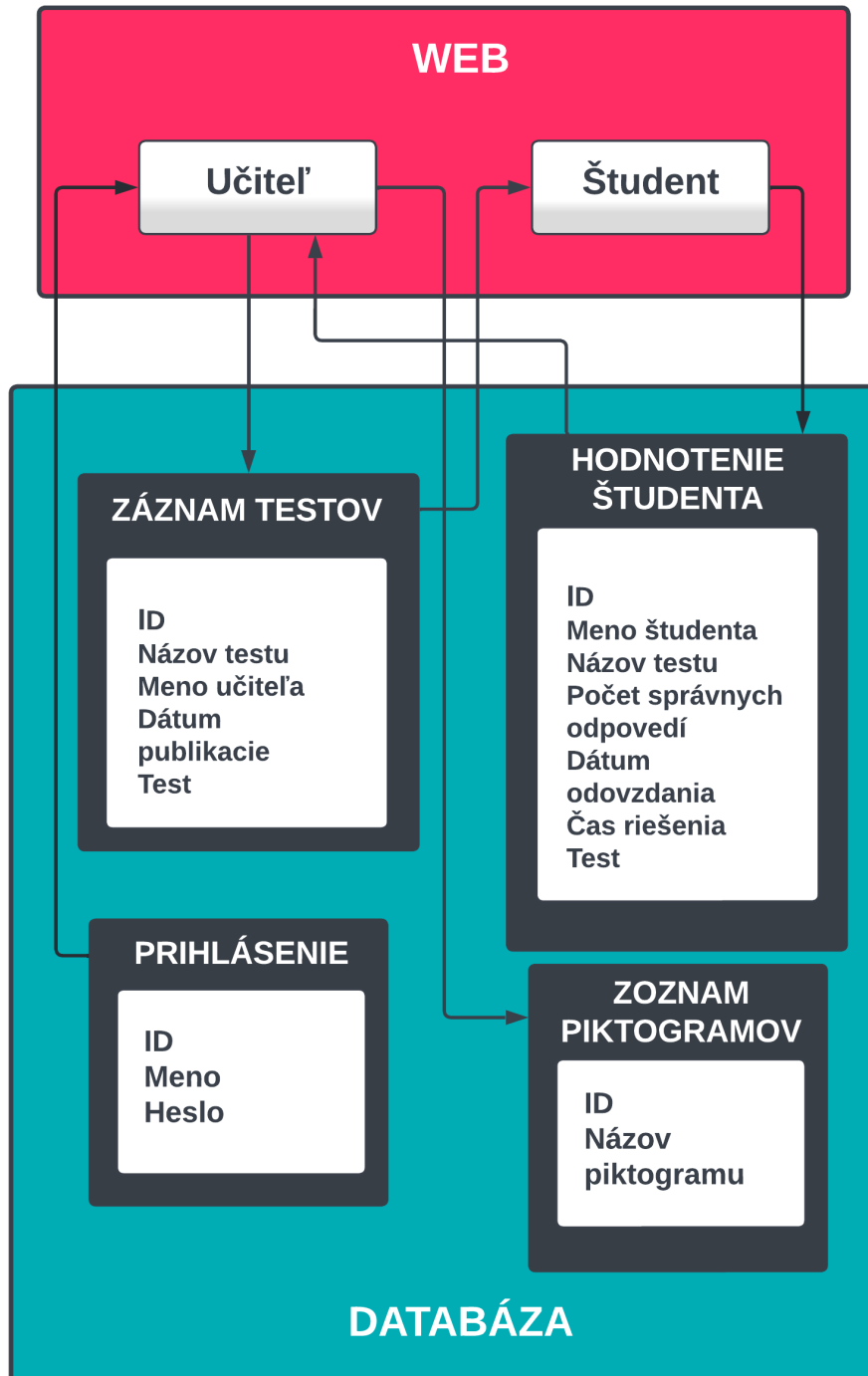
Študent - naopak, rola študenta vie komunikovať iba s dvoma tabuľkami. S rolou učiteľa vie študent komunikovať skrz tabuľku hodnotenia študenta.

Záznam testov - v tejto tabuľke su uschovávané všetky vytvorené testy učiteľmi spoločne s ich základnými údajmi.

Hodnotenie študenta - táto tabuľka uschováva všetky vypracované riešenia testov študentov aj s údajmi opisujúcimi kto vypracoval daný test, za aký čas a s akým hodnotením.

Prihlásenie - obsahom tejto tabuľky sú prihlasovacie údaje role učiteľa.

Zoznam piktogramov - táto tabuľka bola do databázy pridaná z dôvodu uscho-
vania zoznamu názvov piktogramov, ktoré sú využívané pri ikonicko-textovej
metóde výuky. K tomuto zoznamu piktogramov má prístup iba rola učiteľa.



Obr. 3.1: Všeobecná architektúra webovej aplikácie po implementácii

3.2 Rola učiteľa

Prvá vec, na ktorú používateľ narazí pri použití webovej aplikácie je úvodná obrazovka. Táto obrazovka bola implementovaná na základe návrhu v kapitole 2.2. Na tejto obrazovke má používateľ na výber z dvoch tlačidiel. Prvým tlačidlom s nápisom a piktogramom študenta sa používateľ dostane do role študenta. Naopak druhým tlačidlom, obdobne s nápisom a piktogramom, ale tentokrát učiteľa sa vie používateľ dostať do tejto role.



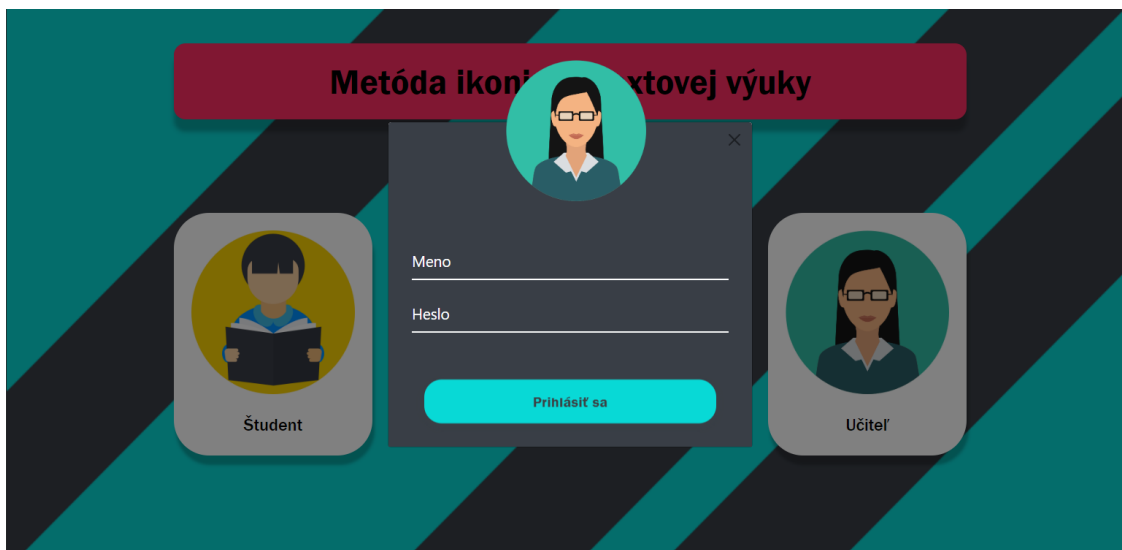
Obr. 3.2: Úvodná obrazovka

Prihlásenie

Po kliknutí na tlačidlo učiteľa sa používateľovi zobrazí vyskakovací formulár. Tento formulár od používateľa žiada jeho prihlasovacie meno a heslo. Ak používateľ zadá zle prihlasovacie údaje a pokúsi sa prihlásiť do systému, formulár sa vyfarbí na červeno.

Ak používateľ zadá správne prihlasovacie údaje, do prehliadača používateľa sa uloží cookies s jedinečným id vďaka, ktorému systém vie, že používateľ je prihlásený a má oprávnenie používať aplikáciu. V prípade, ak by sa používateľ neprihlásil, čoho dôsledkom by bolo aj nevytvorenie cookies, a pokúsil by sa pristupovať pomocou URL adresy k častiam aplikácie, ktoré si prihlásenie vyžadujú, bol by automaticky presmerovaný na úvodnú obrazovku so zobrazeným prihlasovacím formulárom.

Pri odhlasovaní používateľa zo systému, sa z prehliadača používateľa odstráni cookies, ktoré bolo vytvorené pri prihlásení.



Obr. 3.3: Prihlasovací formulár pre učiteľa

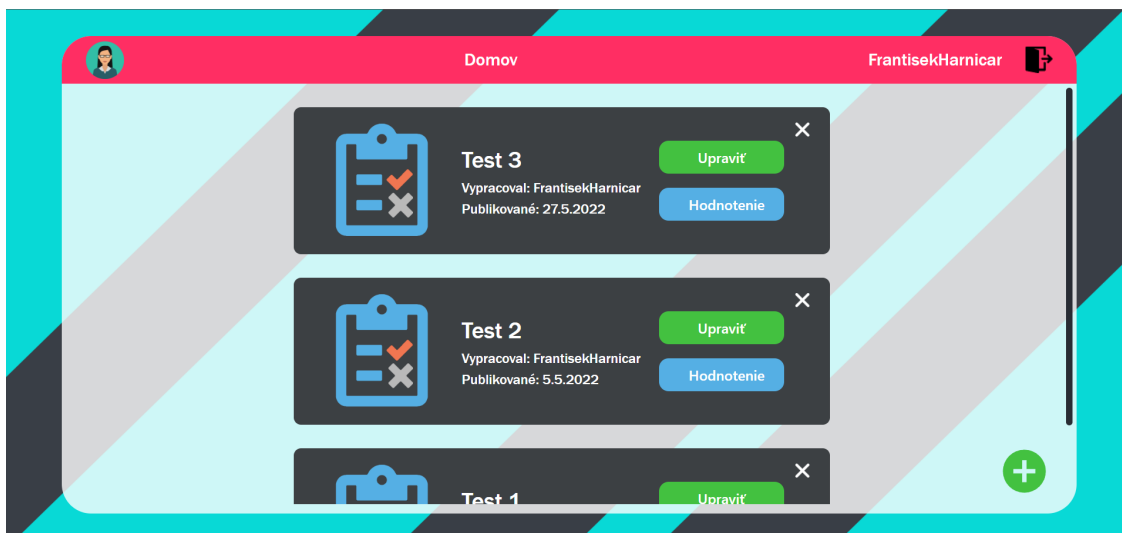
Domovská obrazovka

Po úspešnom prihlásení do systému sa používateľovi zobrazí jeho domovská obrazovka. Z tejto obrazovky sa môže dostať ku všetkým jemu prideleným funkcionalitám, ktoré boli navrhnuté v kapitole 2.2 časti obrazovka učiteľa.

V strede obrazovky ma zobrazené všetky ním vytvorené testy. Tieto testy sú zobrazené podľa poradia v akom boli vytvarané. Test, ktorý bol vytvorený ako posledný je zobrazený ako prvý. Každý test je zobrazený vo vlastnom okne, kde je možné pristupovať k úkonom ako úprava, hodnotenie alebo odstránenie konkrétneho testu.

Odstránenie je riešene pomocou bieleho krížika v pravom hornom rohu každého testu. Po kliknutí na daný krížik sa používateľovi zobrazí vyskakovacie okno pre potvrdenie odstránenia. Táto funkcionality nebola zahrnutá v návrhu ale pri impementácii sa prišlo na fakt, že je potrebné tento úkon potvrdiť viackrát z dôvodu nenávratnej zmeny.

Ďalšou pridanou funkcionalitou oproti návrhu je tlačidlo odhlásenia v pravom hornom rohu aplikácie. Odhlásenia sa z aplikácie je používateľovi umožnené z ktorejkoľvek časti aplikácie. Na vytváranie nových testov slúži zelené tlačidlo s ikonou plus v pravom dolnom rohu aplikácie.



Obr. 3.4: Domovská obrazovka učiteľa

Obrazovka úpravy a vytvárania testu

Po kliknutí na tlačidlo pridávania testov sa používateľovi zobrazí obrazovka znázornená na obrázku 3.5. Táto obrazovka prešla najväčšími zmenami v porovnaní s návrhom. V návrhu bola obrazovka vytvárania testov riešená rozdelením na dve časti. Jedna časť by bola na zadanie úlohy a druhá na riešenie. Ale pri implementácii sa dospelo k názoru, že také riešenie nie je realizovateľné.

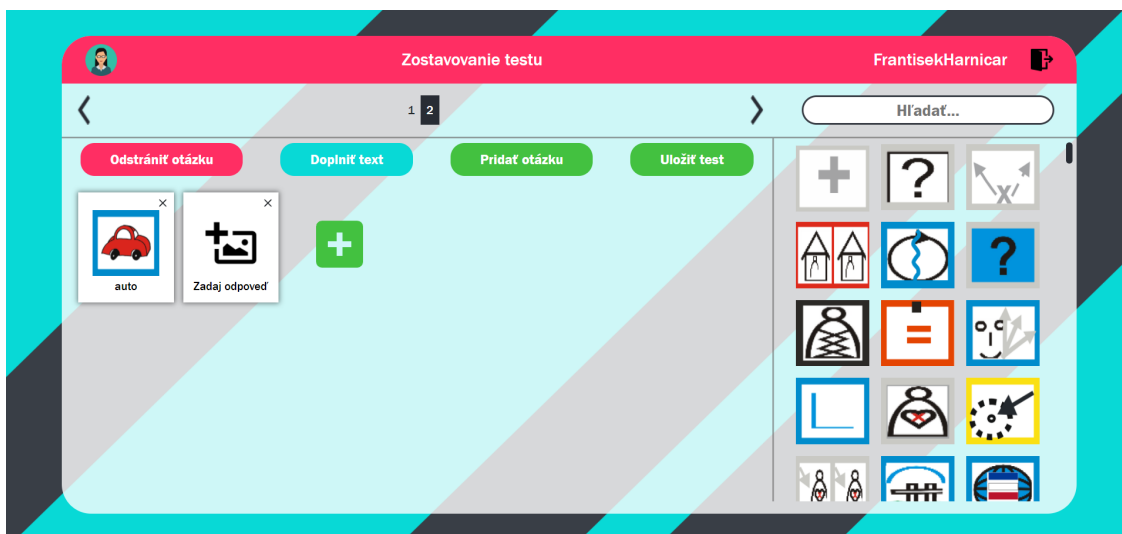
Obrazovka sa preto implementovala úplne iným spôsobom. Horná časť obrazovky slúži na navigáciu medzi vytvorenými úlohami. Hneď pod navigáciou sú štyri tlačidlá, ktoré slúžia na:

- červené tlačidlo na odstránenie aktuálne zobrazenej úlohy,
- modré tlačidlo na zmenu typu riešenia aktuálne zobrazenej úlohy,
- prvé zelené tlačidlo na pridávanie ďalšej úlohy,
- druhé zelené tlačidlo na uloženie zostaveného testu.

Pod spomínanými tlačidlami sa nachádza plocha, na ktorej je možné pridávať pomocou zeleného tlačidla plus nové slová, alebo popríklad kartičky. Tieto kartičky je možné odstraňovať krížikom v pravom hornom rohu. Do kartičky je možné pridávať piktogramy a text. Po pridaní piktogramu do kartičky sa spoločne s ním pridá aj jeho názov. Tento názov môže používateľ ľubovoľne meniť. Akú odpoveď zadá, tá bude správna.

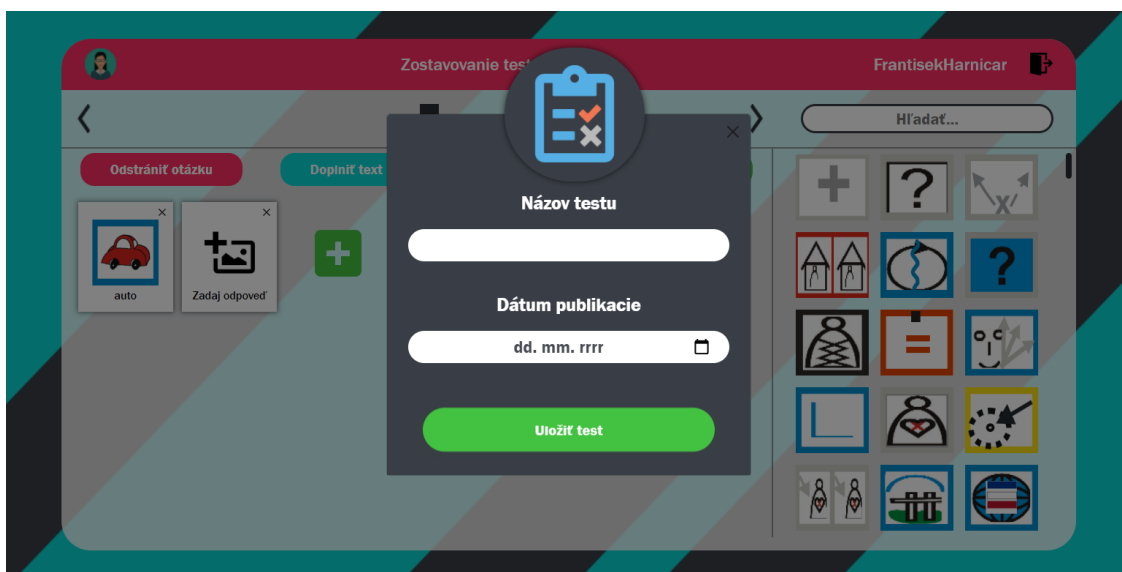
Po pridaní prvej kartičky sa používateľovi na pravej strane zobrazia všetky dostupné piktogramy, ktoré môže použiť. Tieto piktogramy je možné vyhľadávať

pomocou vyhľadávacieho poľa zobrazeného nad nimi. Pri priradení piktogramu kartičke je nutné kliknúť na obrázok priradenia obrázka na kartičke a následne kliknúť na zvolený piktogram vpravo.



Obr. 3.5: Obrazovka úpravy a vytvárania testu

Ak používateľ test úspešne vytvoril a chce ho uložiť, kliknutím na zelené tlačidlo uložiť test sa mu zobrazí vyskakovacie okno zobrazené na obrázku 3.6. V tomto okne je používateľ povinný zadať, pod akým názvom chce test uložiť a zadať deň, od kedy bude test viditeľný pre študentov. Ak sa používateľ pokúsi uložiť test bez zadania jedného alebo oboch parametrov, systém používateľa nepustí a vyfarbí formulár na červeno.



Obr. 3.6: Formulár uloženia testu

Po stlačení tlačidla uložiť test vo formulári sa vytvorený test zapisaný v tvare json súboru odošle spoločne s názvom testu, dátumom publikácie a menom učiteľa na backend a následne sa uloží do databázy. Používateľ je po úspešnom uložení do databázy následne presmerovaný na domovskú obrazovku.

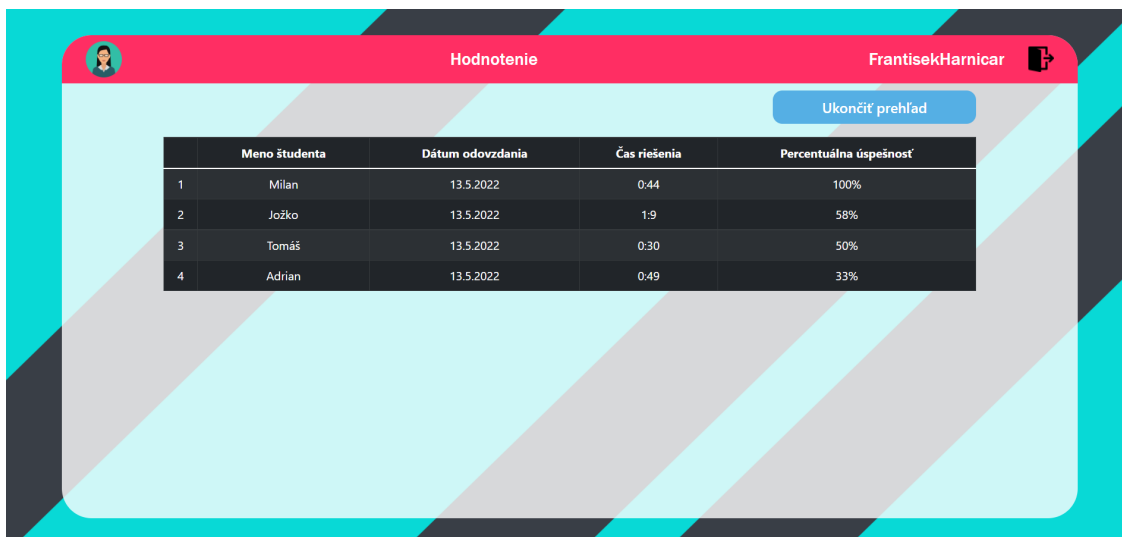
Ak používateľ chce už vytvorený test upraviť kliknutím na tlačidlo upraviť, nad konkrétnym testom sa zobrazí rovnaká obrazovka, ako pri vytváraní nového testu. Test, ktorý chce používateľ upraviť sa vo vývojovom prostredí zobrazí tak, ako ho používateľ uložil. Môže pokračovať v práci tam, kde prestal. Po upravení testu je test potrebné znova uložiť. Je možné zmeniť aj názov testu alebo dátum publikácie.

Obrazovka hodnotenia

Ďalšou možnosťou učiteľa je pozrieť si hodnotenie každého zverejneného testu. Každý test zobrazený na domovskej obrazovke obsahuje tlačidlo hodnotenia. Po stlačení tohto tlačidla sa používateľovi zobrazí obrazovka, ktorá je na obrázku 3.7. Táto obrazovka obsahuje tabuľku všetkých riešiteľov daného testu. Každý riadok tabuľky nesie informácie o riešiteľovi ako:

- meno študenta, ktorý test vypracoval,
- dátum, kedy bol test vypracovaný,
- čas, za ktorý bol test vypracovaný,
- úspešnosť riešenia znázornená v percentách.

Dáta v tabuľke su zoradené podľa úspešnosti riešiteľa v teste. Riešiteľ s najväčším percentuálnym ziskom je zobrazený ako prvý v tabuľke. Každý riadok v tabuľke je klikateľný a je možné zobrazíť test, ktorý odovzdal riešiteľ.



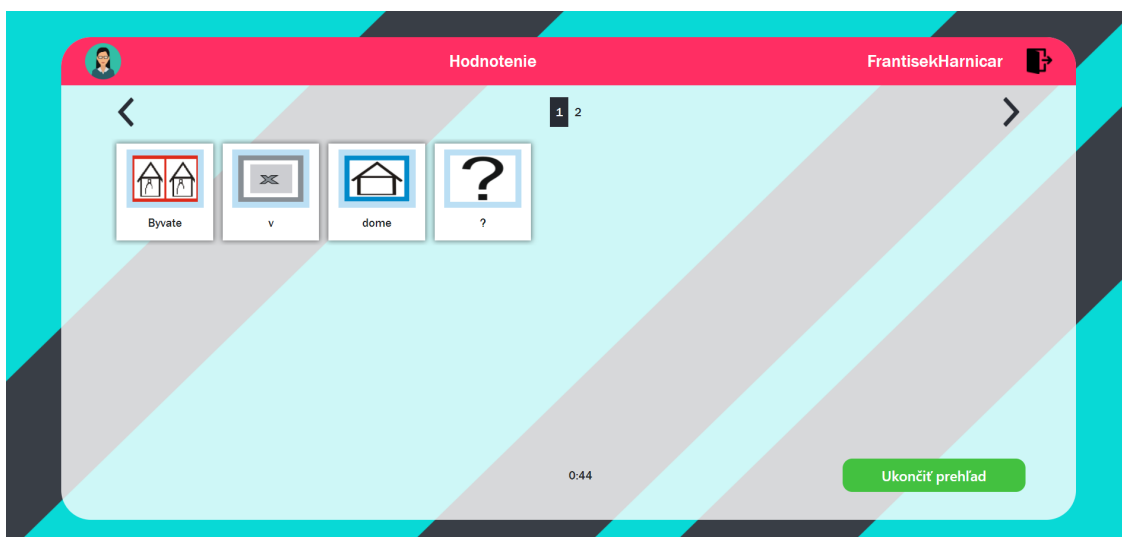
The screenshot shows a web application interface for test evaluation. At the top, there is a pink header with a user profile icon, the title 'Hodnotenie', and the name 'FrantisekHarnicar'. A blue button labeled 'Ukončiť prehľad' is located in the top right. Below the header is a table with the following data:

	Meno študenta	Dátum odovzdania	Čas riešenia	Percentuálna úspešnosť
1	Milan	13.5.2022	0:44	100%
2	Jožko	13.5.2022	1:9	58%
3	Tomáš	13.5.2022	0:30	50%
4	Adrian	13.5.2022	0:49	33%

Obr. 3.7: Obrazovka hodnotenia testu

Po rozkliknutí konkrétneho riadku sa používateľovi zobrazí obrazovka, ktorá je na obrázku 3.8. Na tejto obrazovke používateľ nájde test v takom zobrazení ako ho vidí študent. V hornej časti testu je zobrazená navigácia pomocou, ktorej sa používateľ môže presúvať a prezerať si otázky.

Používateľ nemôže test nijako upravovať. Na zobrazených kartičkách je odpoveď riešiteľa zvýraznená svetlomodrým oramovaním. Dole pod zobrazenými kartičkami je zobrazený čas, za ktorý bol tento test vypracovaný. Na pravo od spomínaného času sa nachádza tlačidlo, ktorým sa vie používateľ vrátiť späť k tabuľke so všetkými riešeniami.



Obr. 3.8: Test konkrétneho riešiteľa

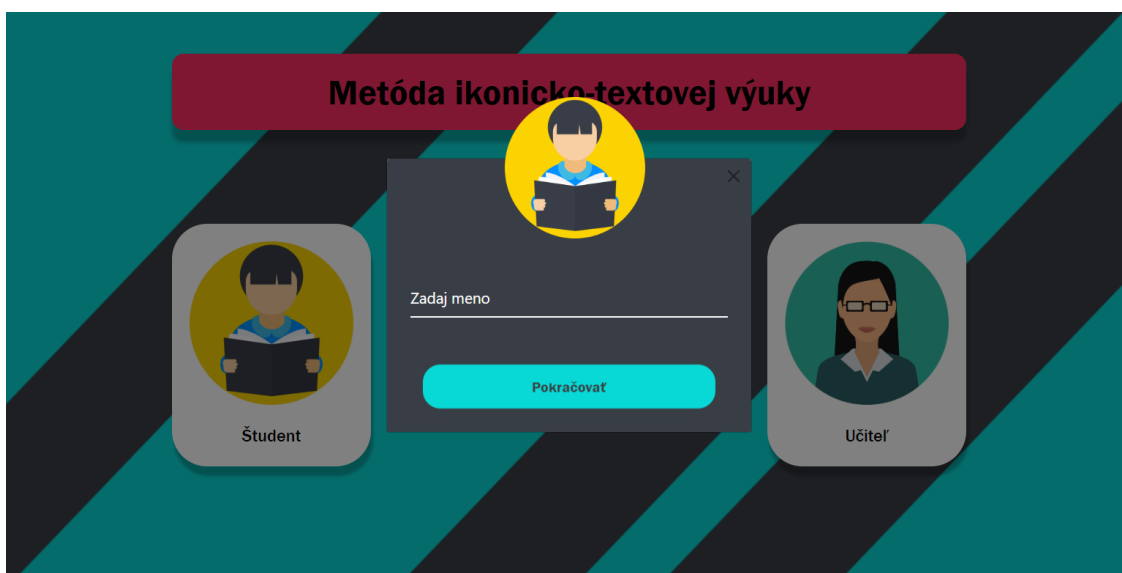
3.3 Rola študenta

Druhou rolou implementovaného systému je rola študenta. K tejto role sa vie používateľ dostať podobne ako k role učiteľa použitím úvodnej obrazovky. Úvodná obrazovka implementovaného systému je zobrazená na obrázku 3.2. Kliknutím na tlačidlo s piktogramom a nápisom študent sa používateľ vie dostať k počítačovej časti role študenta, a tou je prihlásenie.

Prihlásenie

Po kliknutí na tlačidlo sa používateľovi zobrazí vyskakovací formulár s požiadavkou zadania mena. Používateľ pod týmto zadaným menom následne figuruje po celú dobu používania role študenta, až do chvíle odhlásenia. Zadané meno môže byť ľubovoľné, nie je tu riešená žiadna autorizácia.

Zadané meno používateľom sa po stlačení tlačidla pokračovať nahrá do úložiska prehliadača a následne je používateľ presmerovaný na domovskú obrazovku. Ak používateľ opustí webovú aplikáciu bez odhlásenia, po opätovnom navštívení aplikácie a kliknutí na tlačidlo študenta na úvodnej obrazovke od používateľa nebude vyžadované meno, namiesto toho sa bude v aplikácii používať predošlé. Ak používateľ nezadá žiadne meno, štandardne je meno nastavené na Používateľ.

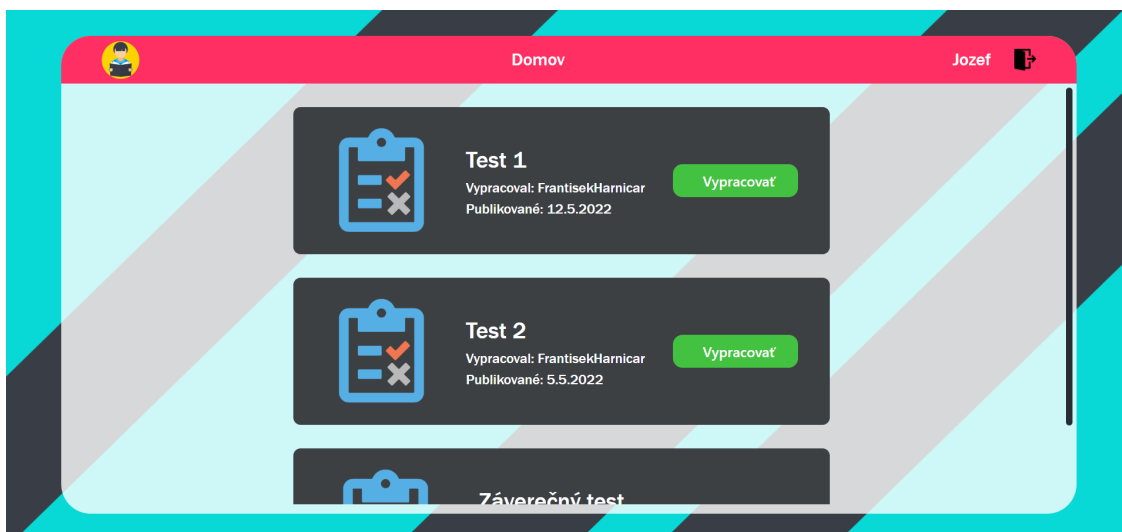


Obr. 3.9: Prihlasovací formulár pre študenta

Domovská obrazovka

Po zadaní mena a stlačení tlačidla pokračovať sa používateľ dostane na domovskú obrazovku role študenta zobrazenú na obrázku 3.10. Na tejto obrazovke má používateľ zobrazené všetky dostupné testy od všetkých učiteľov. Testy sú zoradené podľa dátumu publikácie od toho najnovšieho.

Používateľ v tejto role nemá toľko možností ako v role učiteľa. Ďalším krokom pre študenta je výber testu, ktorý chce riešiť a následne jeho vypracovanie. Každý test zobrazený na obrazovke obsahuje zelené tlačidlo s nápisom vypracovať, po ktorého stlačení sa test nespustí. Po kliknutí na tlačidlo sa používateľovi zobrazí vyskakovacie okno s upozornením a dvoma tlačidlami. Vyskakovacie okno upozorňuje používateľa, že po kliknutí na tlačidlo spustiť test sa otvorí daný test a spustí časomera. Ďalšie tlačidlo vo vyskakovacom okne slúži na zavretie daného okna.



Obr. 3.10: Domovská obrazovka študenta

Obrazovka testu študenta

Táto obrazovka pri implementácii prešla oproti návrhu radikálnymi zmenami z dôvodu zlého navrhnutia rozpoloženia komponentov.

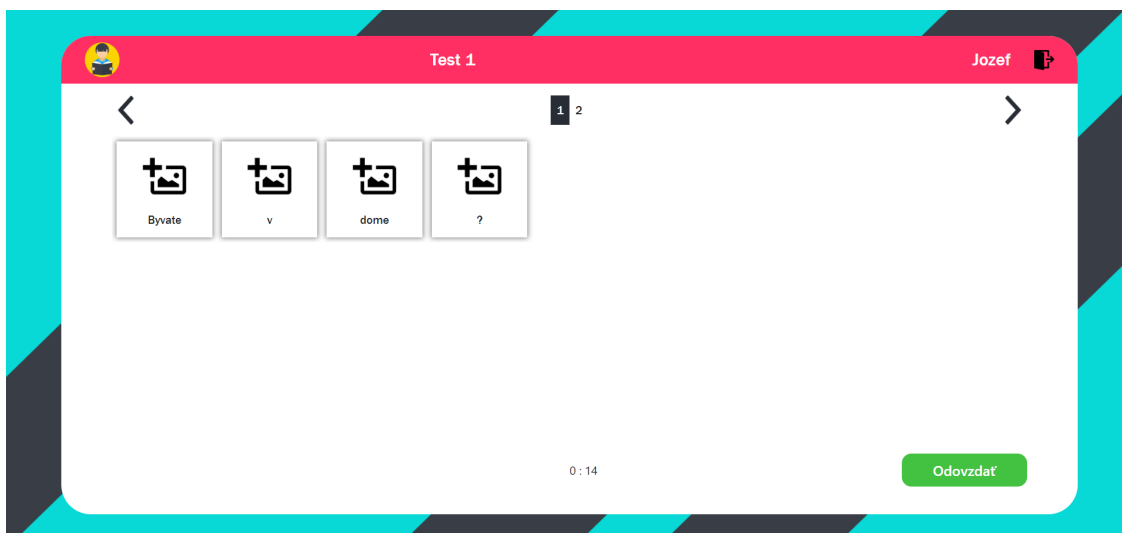
Po spustení testu sa používateľovi zobrazí obrazovka na obrázku 3.11. Táto obrazovka ako jediná v systéme obsahuje plné biele pozadie kvôli zamädzaniu rušivých podnetov z transparentného pozadia.

Vo vrchnej časti testu je zobrazená navigácia v teste. Obsahuje dve šípky, ktorými je možný pohyb medzi otázkami. Ukazovateľ aktuálnej zobrazenej otázky

sa nachádza v strede. Číslo aktuálnej otázky je zobrazené v tmavom obdĺžniku. Pod spomínanou navigáciou sú zobrazené kartičky konkrétnej úlohy.

Na spodnej časti obrazovky plynie čas, ktorý sa spustí pri vstupe do testu a zastaví sa až pri odovzdaní. Vedľa bežiaceho času je zobrazené tlačidlo na odovzдание a následnú kontrolu zadaných odpovedí. Toto tlačidlo je prístupné z každej časti testu. Po jeho stlačení sa používateľovi zobrazí vyskakovacie okno, ktoré ho požiada o opätovné potvrdenie odovzdania. Týmto potvrdením sa predchádza pred náhodným klikom na toto tlačidlo počas riešenia testu.

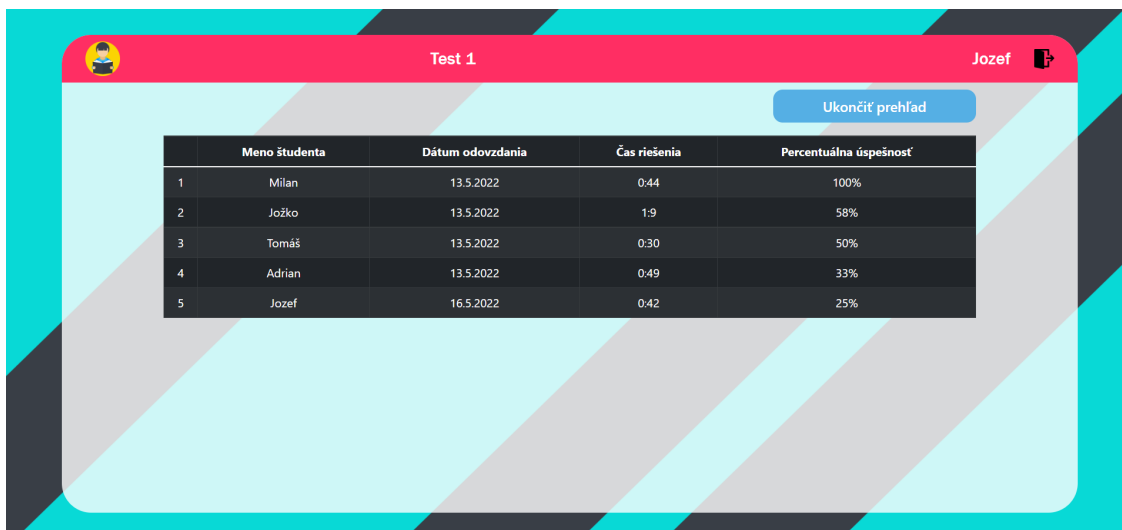
Pri odovzdavaní sa porovnáva vypracovaný test študenta s testom, ktorý vytvoril učiteľ. Na základe počtu správne zadaných slov sa vypočíta percentuálna úspešnosť daného riešenia. Tento vypracovaný test spoločne s menom riešiteľa, časom riešenia, percentuálnou úspešnosťou, názvom testu a dátumom odovzdania je zapísaný do tabuľky hodnotenia študenta v databáze.



Obr. 3.11: Obrazovka testu študenta

Po úspešnom uložení výsledkov sa používateľovi zobrazí tabuľka so všetkými riešeniami daného testu. Tabuľka je vo svojej podstate identická s tabuľkou, ktorú si vie zobrazíť učiteľ ku každému testu. Na rozdiel od tabuľky učiteľa, táto tabuľka nemá klikateľné riadky a preto si používateľ nevie zobrazíť vypracované testy iných riešiteľov.

Táto obrazovka obsahuje ešte tlačidlo, ktoré sa nachádza nad tabuľkou. Po jeho stlačení je používateľ presmerovaný na jeho domovskú obrazovku a môže pokračovať v riešení ďalších dostupných testov.



	Meno študenta	Dátum odovzdania	Čas riešenia	Percentuálna úspešnosť
1	Milan	13.5.2022	0:44	100%
2	Jožko	13.5.2022	1:9	58%
3	Tomáš	13.5.2022	0:30	50%
4	Adrián	13.5.2022	0:49	33%
5	Jozef	16.5.2022	0:42	25%

Obr. 3.12: Obrazovka hodnotenia riešiteľov po vypracovaní testu

Riešenie testov študentom

Test môže byť tvorený viacerými úlohami alebo aj jednou. Každá úloha je tvorená karičkami reprezentujúce jedno slovo. Tieto karičky môžu tvoriť vetu alebo aj súvetie. Záleží na učiteľovi ako test vytvorí.

Každá úloha má svoj typ riešenia. Buď musí študent doplniť význam piktoqramu v karičke alebo doplniť piktoqram na základe významu slova zadaného v karičke.

Riešenie úlohy doplnením textu na základe piktoqramu je implementované zobrazením piktoqramu v hornej časti karičky a následne pod piktoqramom je zobrazený text Zadaj odpoveď. Kliknutím na tento text je možné zadať odpoveď. Odpoveď v karičke sa ukladá automaticky, nie je potrebné ju nijako potvrdzovať. Príklad opisovanej karičky je zobrazený nižšie na obrázku 3.13.



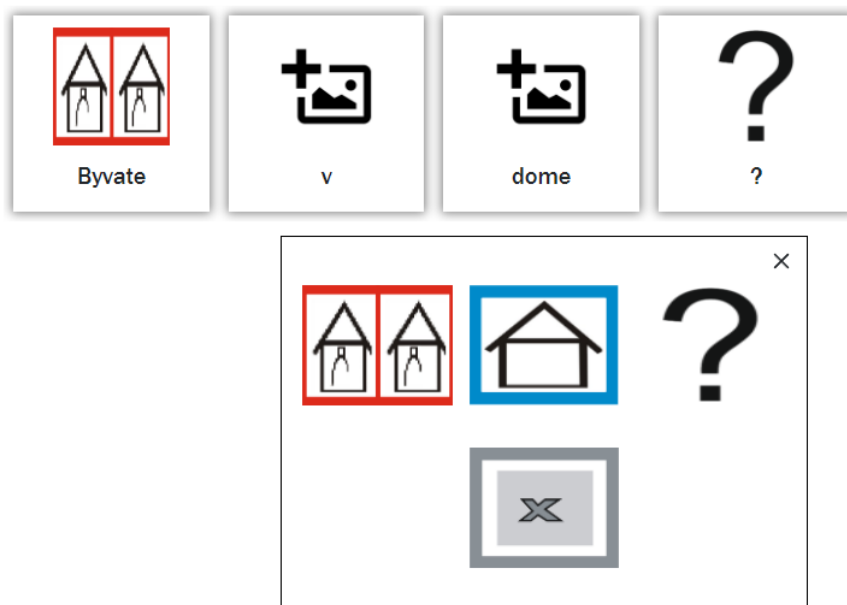
Obr. 3.13: Typ karičky riešenia úlohy doplnením textu

Riešenie úlohy doplnenia piktogramu na základe zadaného slova je implementované použitím vyskakovacieho okna po kliknutí na obrázok pridania piktogramu zobrazeného na kartičke. Príklad použitia vyskovacacieho okna je zobrazený na obrázku 3.14.

Prvým krokom pre priradenie piktogramu kartičke je kliknutie na obrázok pridania obrázka. Po kliknutí na daný obrázok sa používateľovi zobrazí vyskakovacie okno, ktorého obsahom sú všetky piktogramy použité v danej úlohe. Dané okno sa zobrazí stále v blízkosti kartičky, do ktorej chce používateľ priradiť piktogram.

V prípade, ak by bol vo vete použitý jeden piktogram viackrát vo vyskakovacom okne sa tento piktogram zobrazí iba raz. Podobne je pozícia piktogramov vo vyskakovacom okne generovaná náhodne. Následne priradenie piktogramu z vyskovacacieho okna je vykonané kliknutím na konkrétny piktogram. Tento piktogram sa priradi do kartičky a uloží sa.

Ak chce používateľ zmeniť svoj výber piktogramu v kartičke, postup na zmenu piktogramu je rovnaký, ako na jeho priradenie. [11]

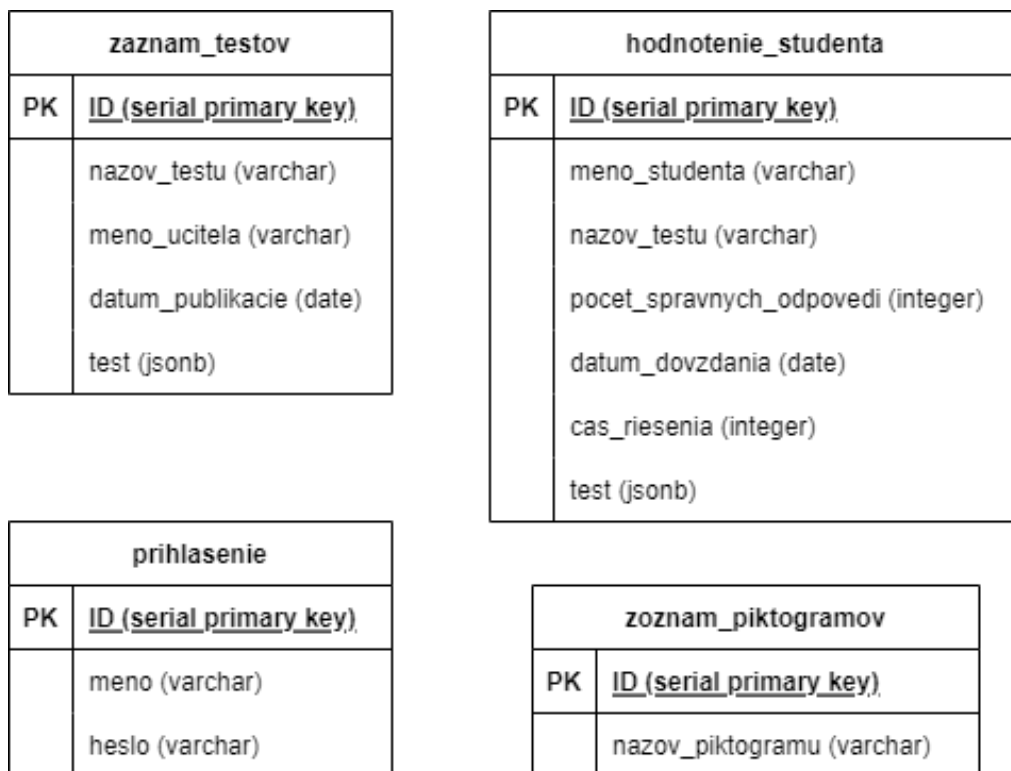


Obr. 3.14: Typ riešenia úlohy pridaním piktogramu

3.4 Databázová časť

Ďalšou implementovanou časťou webovej aplikácie je databáza. Do databázy bola pridaná ďalšia tabuľka v porovnaní s návrhom. Táto tabuľka slúži na uschovanie názvov všetkých piktogramov, ku ktorým má rola učiteľa prístup.

Ďalšou zmenou v porovnaní s návrhom je odstránenie vzťahov medzi tabuľkami, keďže aplikácia sa implementovala tak, že táto vlastnosť nebola využitá. Nakoľko cudzí kľúč v tabuľke hodnotenia študenta nebol potrebný, tento stĺpec sa odstránil.



Obr. 3.15: UML diagram databázy

Prihlásenie

Tabuľka prihlásenia bola implementovaná na základe návrhu implementácie. Boli vytvorené 3 stĺpce id, meno a heslo. Do týchto stĺpcov sa nahrali údaje a to automaticky generované id, čitateľne prihlasovacie meno a heslo, ktoré je uložené v nečitateľnom tvare hash.

Pri prihlásení sa zadané údaje používateľom pošlú na backend aplikácie a vyhľadá sa v databáze zadané prihlasovacie meno. Ak sa meno v databáze nájde vyberie sa z databázy k nemu prisluchajúce heslo v hash tvare. Následne sa heslo zadané používateľom prevedie do hash tvaru a porovná sa s heslom z databázy. Ak sú hashe oboch hesiel rovnaké, v tom prípade autorizácia prebehla úspešne a používateľ je následne prihlásený.

Prihlasovacie údaje sú používateľom pridelené. Registrácia v aplikácii nie je možná.

Záznam testov

Táto tabuľka v implementácii slúži na uchovávanie testov vytvorených rolou učiteľa. Spoločne s testom sú v tabuľke uschované aj základné informácie o teste, ako názov testu, meno učiteľa ktorý test vytvoril a dátum od kedy má byť test viditeľný pre študentov.

Test je v tabuľke uložený vo formáte suborového typu JSON. Uložený test obsahuje všetky potrebné informácie na zostavenie testu pre študenta a aj jeho následnú kontrolu a vyhodnotenie.

Prístup k tabuľke majú oba role. Rola študenta vie dáta iba čítať, ale len tie, ktoré sú mu dátumovo prístupne. Rola učiteľa vie dáta rôzne upravovať, vytvárať nové alebo aj odstraňovať už vytvorené.

Hodnotenie študenta

V tejto tabuľke sú uschované vypracované testy študentov spoločne s údajmi ako čas, za ktorý bol test vypracovaný, meno študenta, dátum kedy bol test vypracovaný, názov testu a percentuálne vyhodnotenie testu. Vyhodnotenie testu sa vykonáva porovnaním odpovedí študenta s odpoveďami, ktoré zadal ako správne učiteľ. Po spočítaní všetkých správnych odpovedí sa vypočíta percentuálna úspešnosť študenta a zapiše sa do tabuľky.

Do tejto tabuľky vie údaje zapisovať iba rola študenta. Zasa naopak, odstraňovanie údajov z tejto tabuľky má na starosti rola učiteľa. Po odstránení testu, na ktorý boli riešenia vytvorené sa všetky odstránia z tejto tabuľky.

Zoznam piktogramov

Táto tabuľka nebola zahrnutá v návrhu implementácie. Pri implementácii sa ale prišlo na fakt, že je potrebné niekde uschovať všetky obrázky, ktoré môže učiteľ pri tvorbe testu využívať. Preto táto tabuľka slúži na uschovanie názvov všetkých piktogramov, ku ktorým má prístup učiteľ pri tvorbe testov.

V tabuľke sú uchované iba názvy piktogramov. Samotné piktogramy vďaka tomu, že ich veľkosť je veľmi malá, nebolo ich potrebné ukladať na externé úložisko, uložili sa do projektu a prístupuje sa k nim na základe ich názvu.

4 Overenie webovej aplikácie

Overenie implementovanej webovej aplikácie prebehlo dvoma spôsobmi. Prvý spôsob bol zameraný na používateľa. Vytvoril sa scenár, podľa ktorého mali používatelia používať aplikáciu. Po prejdení všetkých krokov mali používatelia odpovedať na otázky vo formulári.

Druhým spôsobom overenia aplikácie boli záťažové testy. Do aplikácie sa pridalo väčšie množstvo údajov a testovalo sa, ako na to aplikácia reaguje.

4.1 Overenie používateľom

Aplikáciu overovalo osem používateľov, prevažne v mladšom veku. Žiadny z používateľov nemal doterajšie skúsenosti s ikonicko-textovou metódou výuky. Na základe tejto skutočnosti bola používateľom pred začatím overovania jemne približená táto metóda.

Overenie prebiehalo následovne. Používateľovi bol podaný formulár. Na začiatku formulára bol zadaný scénar, ktorého sa mal používateľ držať a na základe neho postupovať v aplikácii. Obsah scenára bol následovný:

1. Prihláste sa do aplikácie ako učiteľ zadaním mena a hesla.
2. Otvorte prostredie na vytvorenie nového testu.
3. Test bude obsahovať dve vety. Každá veta bude v samostatnej úlohe. Prvá veta bude typu "Doplniť text", druhá zasa "Doplniť obrázok".
4. Obsah viet bude následovný: 1 - auto je cervene, 2 - kral a kralovna byvaju v hrade.
5. Test uložte s názvom "Test formulára", dátum publikácie nech je dnes.
6. Odhláste sa z aplikácie.
7. Prihláste sa do aplikácie ako študent.

8. Vyplňte a odovzdajte test, ktorý ste vytvorili.

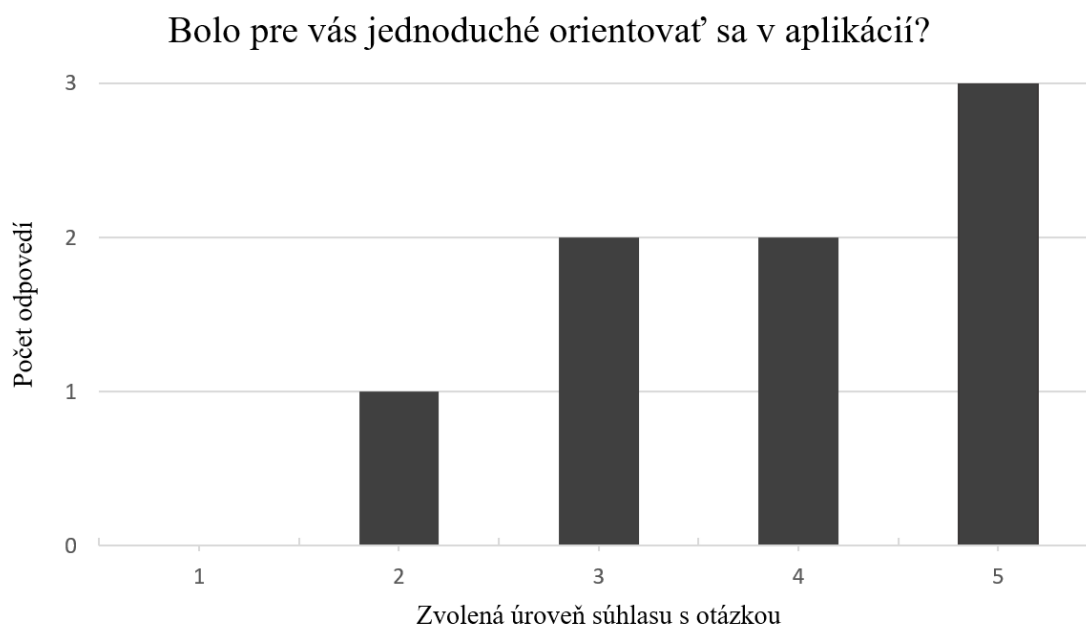
9. Odhláste sa z aplikácie.

Scenár nezahrňoval všetky časti aplikácie. Jeho obsahom bola základná funkcionálna aplikácia. Vytvorenie testu rolou učiteľa a vypracovanie vytvoreného testu študentom.

Po vypracovaní celého scenára používateľ vyplnil formulár s otázkami. Otázky boli smerované na zistenie nedostatkov aplikácie.

Bolo pre vás jednoduché orientovať sa v aplikácií?

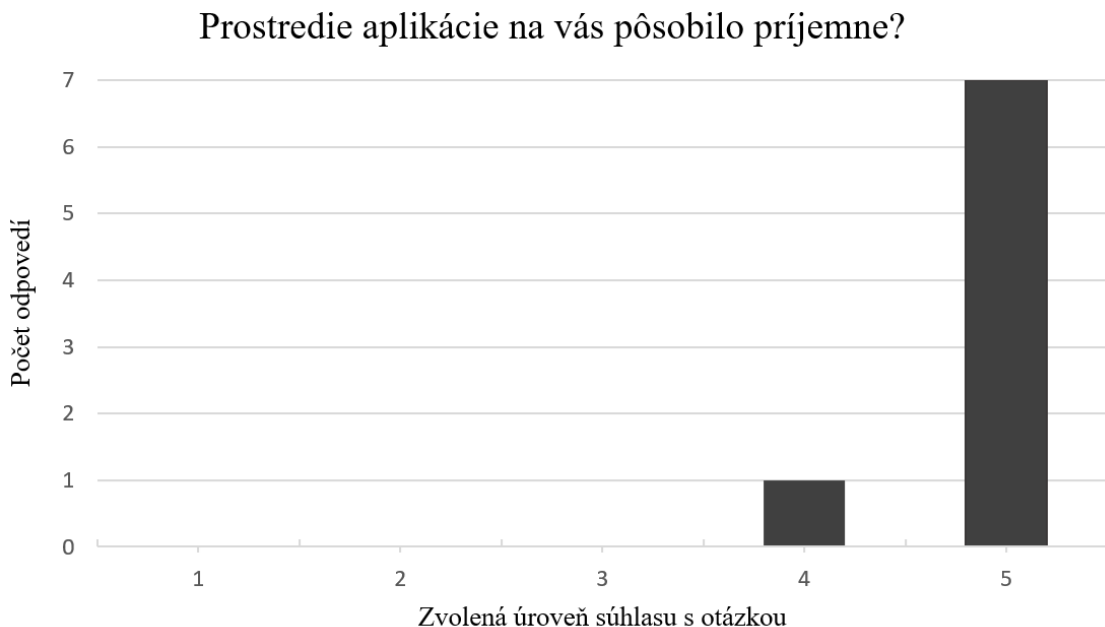
Cieľom tejto otázky bolo zistiť od používateľa, ako dobre sa vedel orientovať v aplikácií počas priebehu scenára. Na grafe môžeme vidieť, že odpoveď na túto otázku bola rôznorodá. Čím väčšie číslo používateľ zadal, tým bol viac spokojný z orientáciou v aplikácií. Prevažujú kladné odpovede, ale je tu možné nájsť aj zapornú odpoveď. Číslo odpovede je zobrazené na horizontálnej osi a počet používateľov, ktorí ju zvolili je zobrazený na osi vertikálnej.



Obr. 4.1: Graf odpovedí na otázku: Bolo pre vás jednoduché orientovať sa v aplikácii?

Prostredie aplikácie na vás pôsobilo príjemne?

Táto otázka zisťuje ako na používateľa pôsobilo prostredie aplikácie. Graf odpovedí je v tomto prípade jednoznačnejší. Takmer všetci používatelia boli veľmi spokojní s dizajnom aplikácie.



Obr. 4.2: Graf odpovedí na otázku: Prostredie aplikácie na vás pôsobilo príjemne?

Mali ste problém s niektorým z bodov scenára? Ak áno, uveďte číslo bodu a vzniknutý problém.

Cieľom tejto otázky bolo zistiť, ktorý bod scenára robil používateľom najväčší problém a o aký problém išlo. Dvaja používatelia nemali žiadne problémy. Zvyšní používatelia mali prevažne problémy v bodoch 3 a 4. Išlo o problémy spojené s vytváraním testov. Jeden z problémov bol, že nebolo používateľom jasné, aký typ otázky majú momentálne zvolený, keďže je typ otázky zobrazený v tlačidle.

Ďalší problém, na ktorý používatelia narazili spočíval v tom, že si neuvedomovali možnosť upravovať význam kartičky pri vytváraní vety.

Existuje funkcionálnosť, ktorú by ste do aplikácie pridali? Ak áno, uveďte akú.

Hlavným bodom tejto otázky bolo zistiť, akú novú funkcionálnosť by do aplikácie pridali používatelia. Traja používatelia boli s aplikáciou spokojní a nenavrholi

žiadnu funkcionálnu navyše. Jedným z návrhov bolo na obrazovke vytvárania testu pri piktogramoch načítaných z databázy pridať to, čo daný obrázok znamená. Obrázky by boli takto ľahšie čitateľné.

Ďalším z návrhov bolo vytvorenie sprievodcu v aplikácii, ktorý by používateľovi ukázal základné funkcionality, ktoré môže vykonávať. V odpovediach na otázku zaznel ešte návrh, pridať spätné vrátenie sa k výsledkom testu po jeho vypracovaní v role študenta. Pomocou tejto funkcie, by si mohol študent pozrieť čo zadal v teste správne alebo nesprávne pri jeho vypracovaní.

Je niečo, čo by ste na aplikácii zmenili?

Cieľom tejto otázky bolo zistiť, akú implementovanú funkcionálnu by používatelia zmenili. Traja používatelia nemali záujem niečo na aplikácii meniť. Jeden z opýtaných navrhol zvýrazniť pole vyhľadávania piktogramov pri vytváraní testov tak, aby bolo jasnejšie viditeľné.

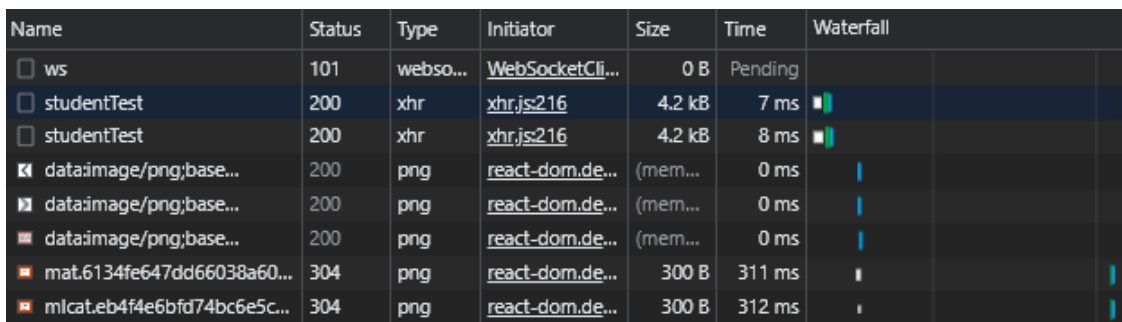
Ďalším návrhom používateľa bola zmena v navigácii pri prechode medzi otázkami v teste, ako pri role študenta, podobne aj pri role učiteľa. Používateľ navrhol pridať do navigácie možnosť kliknutia na číslo otázky a následného presmerovania na vybranú otázku.

4.2 Overenie zaťažovým testom

Pri upravovaní testov, ako aj pri ich vypracovaní môže vzniknúť otázka, ako dlho bude trvať načítanie testu u študenta, ako aj u učiteľa, ak učiteľ vytvorí nadštandardne veľký test. Preto sa v tejto kapitole nasimuluje vytvorenie nadštandardne veľkého testu a následne sa zistí, ako rýchlo to implementovaný program spracoval.

Samotná simulácia

Pre potreby simulácie sa vytvoril test pomocou role učiteľa. Test obsahoval 35 otázok a 367 kartičiek. Do prvej otázky sa pridalo v pomere najviac slov, keďže tá sa zobrazuje pri zobrazení testu ako prvá. Táto otázka obsahovala 152 kartičiek s unikátnymi piktogramami.



Name	Status	Type	Initiator	Size	Time	Waterfall
ws	101	webso...	WebSocketCli...	0 B	Pending	
studentTest	200	xhr	xhr.js:216	4.2 kB	7 ms	
studentTest	200	xhr	xhr.js:216	4.2 kB	8 ms	
data:image/png;base...	200	png	react-dom.de...	(mem...	0 ms	
data:image/png;base...	200	png	react-dom.de...	(mem...	0 ms	
data:image/png;base...	200	png	react-dom.de...	(mem...	0 ms	
mat.6134fe647dd66038a60...	304	png	react-dom.de...	300 B	311 ms	
mlcat.eb4f4e6bfd74bc6e5c...	304	png	react-dom.de...	300 B	312 ms	

Obr. 4.3: Znáznorenie načítania testu

Ako je možné vidieť na obrázku 4.3 otvorenie vytvoreného testu trvalo v konečnom dôsledku krátky čas na to, aby bolo nutné vytvárať nejaké rozhranie, ktoré by slúžilo na informovanie používateľa o priebehu načítavania testu. Pri otvorení testu je vidieť menšiu odozvu, ale tá je pri bežnom používaní aplikácie nepovšimnuteľná.

5 Záver

V práci bolo potrebné vypracovať analýzu webových možností pre metódu ikonicko - textovej formy výuky hendikepovaných detí. Z tohto dôvodu je vypracovaná analytická časť, v ktorej sú analyzované možné varianty webového riešenia, ako aj analýza možných nástrojov pri samotnej implementácii.

Ďalším z bodov práce je vypracovaný návrh systému, ktorý bol zostavený na základe vypracovanej analýzy. Jeho obsahom je návrh riadenia systému rolí učiteľa a študenta. Taktiež obsahuje návrh všetkých obrazoviek webovej aplikácie spoločne s databázovým riešením.

Na základe návrhu bolo ďalej potrebné implementovať systém. Preto práca obsahuje kapitolu 3, v ktorej je opísaný existujúci implementovaný systém. Tento systém obsahuje zopár zmien v porovnaní s návrhom. Tieto zmeny su zahrnuté v spomínanej kapitole.

Po implementovaní systému bolo potrebné vytvorený systém overiť. Overenie je vypracované v kapitole 4 dvoma spôsobmi. Prvý spôsob overenia bol realizovaný predložením systému používateľom, ktorý mali v aplikácii postupovať na základe scenára a následného vyplnenia formulára. Druhý spôsob overenia bol realizovaný záťažovým testom.

Do implementovanej aplikácie by bolo možné pridať zopár vylepšení navrhovaných používateľmi v kapitole 4.1. Alebo je možné uvažovať aj nad vylepšeniami, ako implementovanie kooperácie medzi učiteľom a študentom napríklad tak, aby učiteľ vedel sledovať študenta pri vypracovávaní testu. Ďalšou možnosťou vylepšenia by bolo pridanie viacerých typov úloh v teste.

Literatúra

1. FELDMAN, J.M.; JONES, J. Semiautomatic testing of student software under Unix(R). *IEEE Transactions on Education*. 1997, roč. 40, č. 2, s. 158–161. Dostupné z DOI: 10.1109/13.572331.
2. MAULANA, Muhammad Yusuf; BANDUNG, Yoanes. System design of collaborative learning portal to support teacher learning communities. 2013, s. 1–5. Dostupné z DOI: 10.1109/ICTSS.2013.6588119.
3. KOZEL, Tomas. Aplikácia pre podporu kolaboratívnej práce so študentmi. 2012, s. 31. Dostupné tiež z: <https://opac.crzp.sk/?fn=detailBiblioForm&sid=FB227A2458B7F9F4106245480396>.
4. SOPHONPATTANAKIT, Supaporn; ACHALAKUL, Tiranee. Learning Portal for the Technology of Collaborative Work on Virtual Space. 2008, s. 178–181. Dostupné z DOI: 10.1109/ASEA.2008.35.
5. PETRÍKOVÁ DARINA, Sobota Branislav a. *Virtuálno-reality technológie súčasnosti a vzdelávanie postihnutých ľudí* [elektronický zdroj]. 2019. ISBN 978-80-553-3389-2.
6. BOOTSRAP, getbootstrap.com. *Bootstrap Documentation*. Dostupné tiež z: <https://getbootstrap.com/docs/5.1/getting-started/introduction/>.
7. REACT, reactjs.org. *React Documentation*. Dostupné tiež z: <https://reactjs.org/docs/>.
8. NODE.JS, nodejs.org. *Node.js Documentation*. Dostupné tiež z: <https://nodejs.org/en/docs/>.
9. POSTGRESQL, postgresql.org. *PostgreSQL Documentation*. Dostupné tiež z: <https://www.postgresql.org/docs/>.
10. EXPRESS, expressjs.com. *Express Documentation*. Dostupné tiež z: <https://expressjs.com/en/api.html>.

11. HAMBALEK, Samuel. *Metóda ikonicko-textovej formy výuky hendikepovaných detí na báze webových technológií*. 2022. Bakalárska práca. "3.2 - Implementácia webového rozhrania".

Zoznam skratiek

CSS Cascading Style Sheets.

DOM Document Object Model.

HTML HyperText Markup Language.

IP Internet Protocol.

JSON JavaScript Object Notation.

UML Unified Modeling Language.

URL Universal Resources Locator.

Zoznam príloh

Príloha A Používateľská príručka

Príloha B Systémová príručka

Príloha C Formulár z používateľského testovania

Príloha D CD médium

A Používateľská príručka

A.1 Funkcie programov

A.1.1 Technické požiadavky

Na spustenie webovej aplikácie je potrebné, aby zariadenie splňalo minimálne tieto požiadavky:

- Operačný systém Windows 7 - optimalizované na windows 10,
- webový prehliadač - optimalizované na prehliadač Google chrome
- 1Gb voľného priestoru na internom úložisku
- inštalované všetky potrebné knižnice

A.1.2 Spustenie aplikácie

Aplikácia je v štádiu kedy nie je dostupná z externých zdrojov. Je uschovaná na médiu priloženom v záverečnej práci. Je potrebné ju z média stiahnuť a vykonať nasledujúce kroky:

1. Skopírovať aplikáciu z média a uložiť ju na interné úložisko zariadenia.
2. Je potrebné súbor aplikácie rozbaľiť, keďže je v súborovom formáte zip.
3. Je potrebné otvoriť priečinok **my-app** a v ňom vykonať pomocou príkazu **npm** inštaláciu všetkých potrebných knižníc.
4. Rovnakú operáciu je potrebné vykonať v súbore Server a inštalovať všetky knižnice pomocou príkazu **npm**.
5. Následne v priečinku **Server** je potrebné zadať príkaz **node index** pre spustenie servera.
6. V priečinku **my-app** je potrebné do konzoly zadať **npm start** ,ktorým sa spustí aplikácia.

A.2 Použitie programu

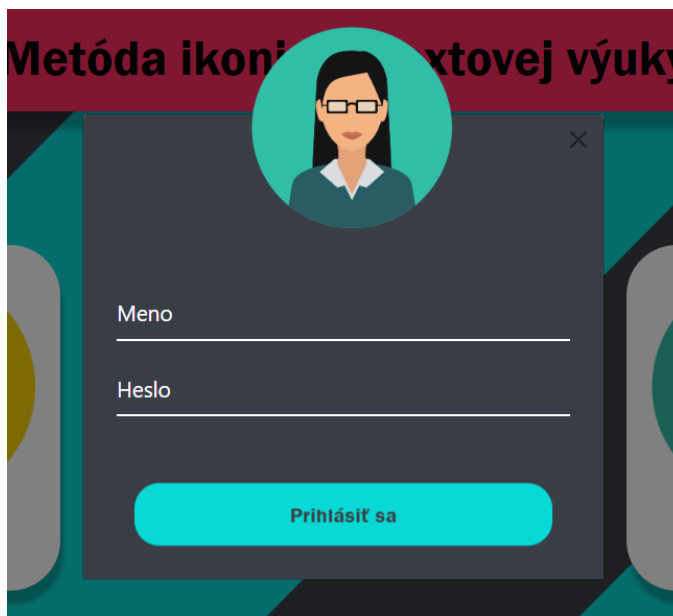
Vytvorená aplikácia sa skladá z dvoch častí(rolí). Každá časť aplikácie má svoju hlavnú obrazovku, ale obe majú spoločnú úvodnú obrazovku. Na obrázku A.1 je možné vidieť obrazovku po spustení aplikácie.



Obr. A.1: Úvodná obrazovka po spustení aplikácie

A.2.1 Prihásenie do roly učiteľa

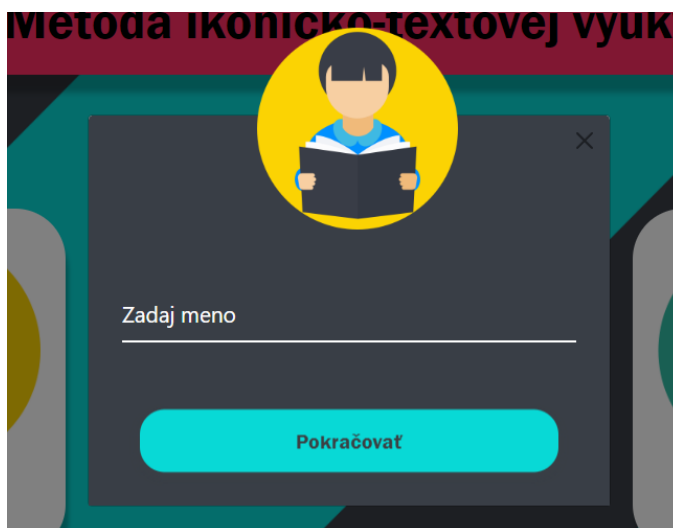
Prihlásenie do roly učiteľa sa vykonáva stlačením tlačidla učiteľ na úvodnej obrazovke. Po jeho stlačení sa otvorí vyskakovacie okno, kde je potrebné zadať prihlasovacie údaje, ktoré su pridelené poskytovateľom. Pri zadaní nesprávnych údajov sa prihlasovací formulár zafarbí na červeno. Po zadaní správnych údajov je potrebné stlačiť tlačidlo **Prihlásiť sa** zobrazené pod formulárom. Po prihlásení bude používateľ presmerovaný na úvodnú obrazovku učiteľa. Formulár je zobrazený na obrázku A.2.

The image shows a login form titled "Metóda ikonicko-textovej výuky" (Iconic-textual learning method). At the top, there is a circular profile picture of a woman with glasses. Below it, there are two input fields: "Meno" (Name) and "Heslo" (Password). At the bottom of the form is a large cyan button labeled "Prihlásiť sa" (Log in).

Obr. A.2: Prihlasovací formulár

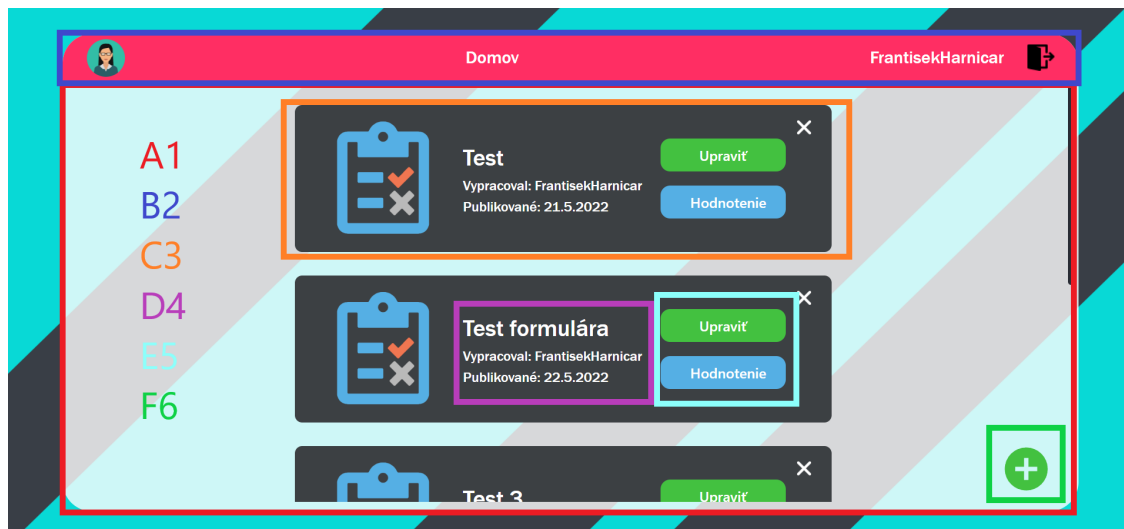
A.2.2 Prihásenie do roly študenta

Pri prihlásení do roly študenta je potrebné kliknúť na úvodnej obrazovke na tlačidlo študenta. Po kliknutí sa otvorí formulár, do ktorého je potrebné zadať meno. Meno môže byť ľubovoľné. S týmto menom používateľ vystupuje následne počas celej doby používania aplikácie, pokiaľ sa neodhlási. Po stlačení tlačidla **Pokračovať** bude používateľ presmerovaný na domovskú obrazovku študenta.

The image shows a login form titled "Metóda ikonicko-textovej výuky" (Iconic-textual learning method). At the top, there is a circular profile picture of a student reading a book. Below it, there is one input field labeled "Zadaj meno" (Enter name). At the bottom of the form is a large cyan button labeled "Pokračovať" (Continue).

Obr. A.3: Prihlasovací formulár

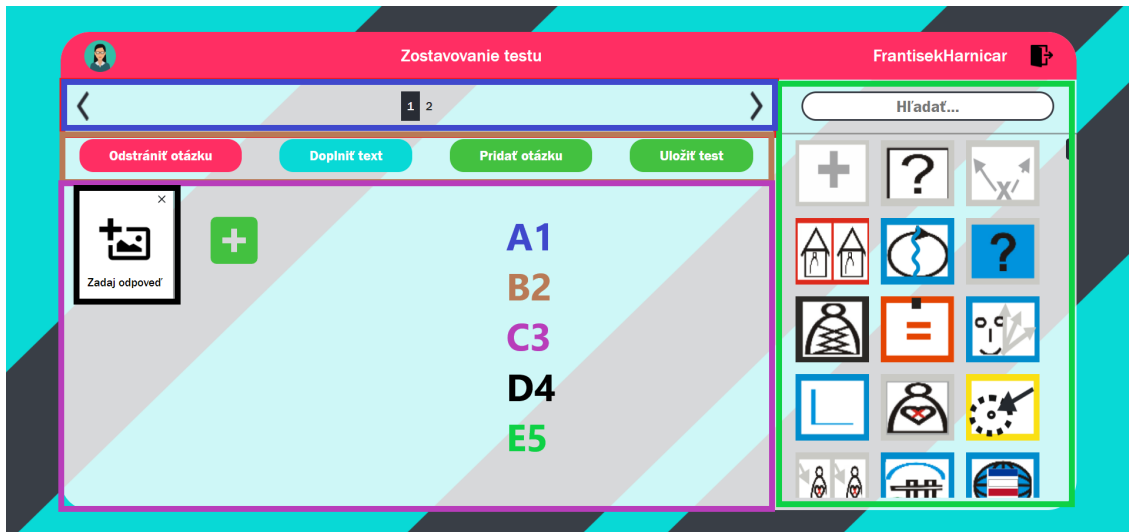
A.2.3 Domovská obrazovka učiteľa



Obr. A.4: Domovská obrazovka učiteľa

- A1** - V oblasti označenej touto farbou sa zobrazujú všetky testy vytvorené daným učiteľom.
- B2** - V tejto oblasti je zobrazený piktogram učiteľa, po ktorého kliknutí je používateľ presmerovaný na domovskú obrazovku, názov obrazovky, na ktorej sa používateľ nachádza spoločne s menom prihláseného učiteľa a tlačidlom odhlásenia. Táto časť aplikácie je zobrazená v každej časti aplikácie a má rovnaké vlastnosti.
- C3** - Znázornená oblasť označuje konkrétny test. V pravom hornom rohu sa nachádza krížik, ktorým je možné test odstrániť.
- D4** - V tejto oblasti sa nachádzajú základné informácie o teste, ako jeho názov, dátum publikácie a meno autora.
- E5** - Oblasť znázorňuje tlačidlá upraviť a hodnotenie. Tlačidlom upraviť je možné konkrétny test upraviť. Tlačidlom hodnotenie je možné zobrazíť si hodnotenie testu.
- F6** - Oblasť znázorňuje tlačidlo plus, pomocou ktorého je možné vytvoriť nový test.

A.2.4 Obrazovka vytvorenia testu



Obr. A.5: Obrazovka vytvorenia testu

A1 - V oblasti označenej touto farbou je zobrazená navigácia medzi otázkami.

B2 - Táto oblasť obsahuje 4 tlačidlá:

1. Červené tlačidlo slúži na odstránenie otázky, na ktorej sa používateľ nachádza.
2. Modré tlačidlo slúži na zmenu typu otázky. Buď doplniť text alebo doplniť obrázok. Aktuálny typ otázky je zobrazený na tlačidle.
3. Prvé zelené tlačidlo slúži na pridanie novej otázky.
4. Posledné tlačidlo slúži na otvorenie formulára na uloženie testu.

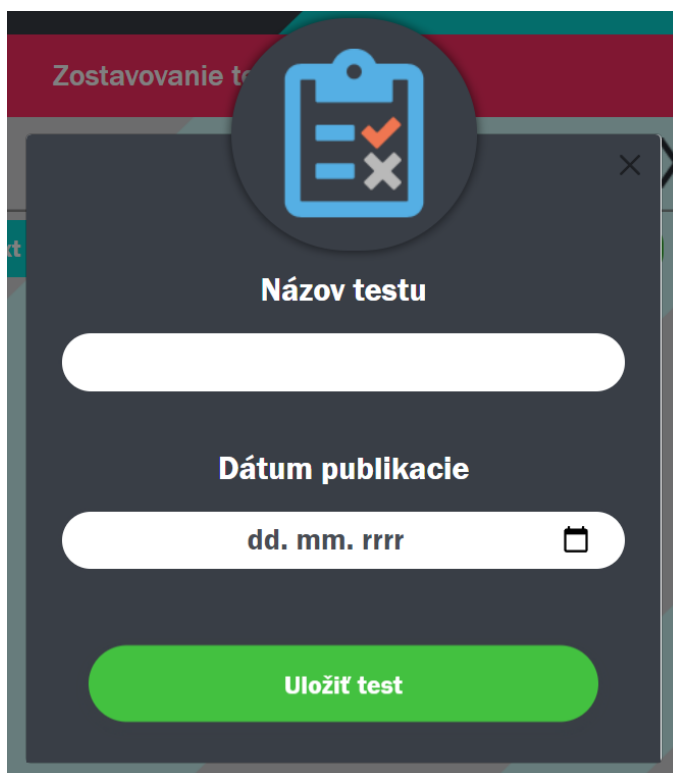
C3 - Oblasť znázorňuje kartičky, ktoré su pridané v konkrétnej otázke. Kartičku je možné pridať zeleným pluskom pred zobrazeným pred poslednou kartičkou.

D4 - Oblasť znázorňuje kartičku. V kartičke je v hornej časti zobrazený krížik, ktorým je možné konkrétnu kartičku odstrániť. Ďalej je v kartičke zobrazený obrázok, po ktorého kliknutí je možné pridať akýkoľvek piktogram z časti E5. Pod obrázkom je znázornený text významu obrázka. Ten je možné po kliknutí naňho upravovať.

E5 - V tejto oblasti su zobrazené všetky piktogramy, ktoré sú uložené v aplikácii. Tieto piktogramy je možné po kliknutí na ne pridávať do oblasti D4. V

hornej časti sa nachádza pole pomocou, ktorého je možné vyhľadávať piktogramy podľa ich názvu.

Po stlačení tlačidla uložiť test sa používateľovi zobrazí formulár na uloženie testu do databázy zobrazený na obrázku A.6. Na uloženie testu je do formulára potrebné pridať názov testu a dátum publikácie. Od daného dátumu bude mať rola študenta vidieť test. Stlačením tlačidla uložiť test sa test uloží do databázy a používateľ sa presmeruje na domovskú obrazovku.

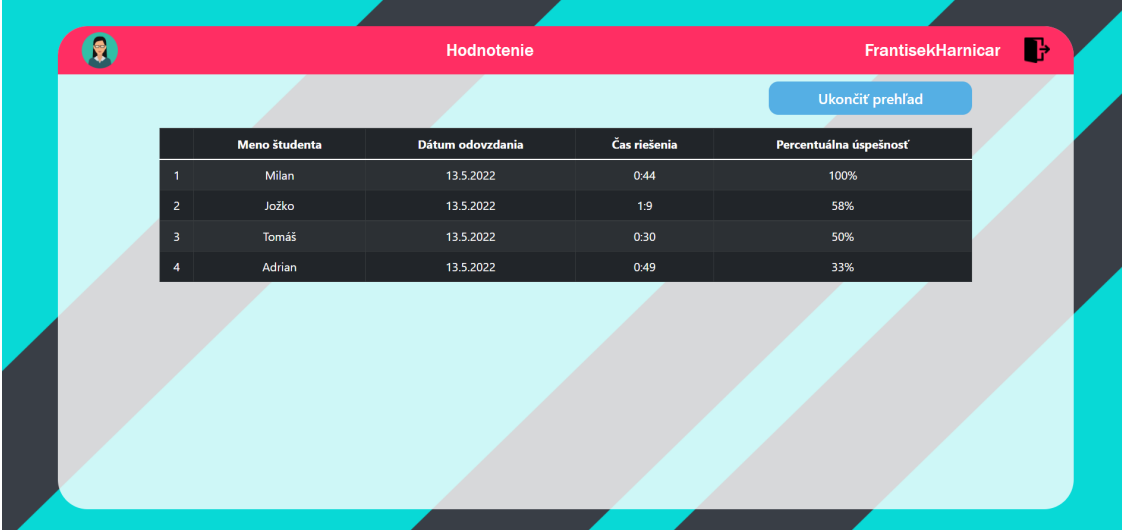


Obr. A.6: Formulár uloženia testu

Pri upravovaní už vytvoreného testu sa obrazovka správa identicky. Jediným rozdielom je, že upravovaný test sa nahrá do systému a zobrazí sa na obrazovke presne ako pred uložením.

A.2.5 Obrazovka hodnotenia testu

Táto obrazovka zobrazuje tabuľku všetkých riešení zvoleného testu. Výsledky v tabuľke su zoradené od najlepšieho percentuálneho hodnotenia smerom dole. Každé riešenie je možné zobraziť kliknutím na riadok v tabuľke. Nad tabuľkou sa nachádza tlačidlo, ktorým je možné prehľad výsledkov ukončiť.

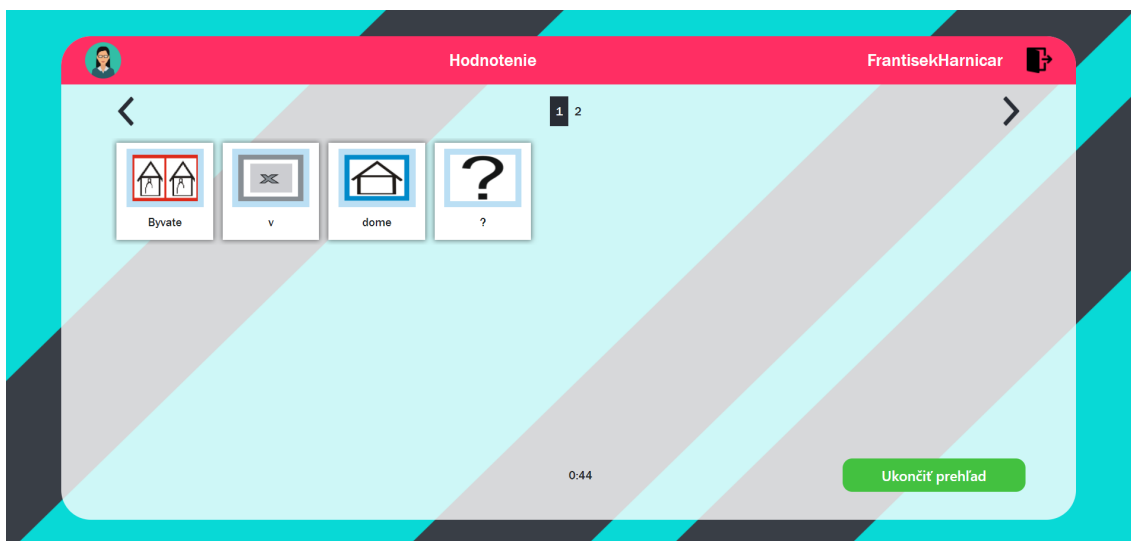


The screenshot shows a mobile application interface for test evaluation. At the top, there is a pink header with a user profile icon, the title 'Hodnotenie', and the name 'FrantisekHarnicar'. A blue button labeled 'Ukončiť prehľad' is located in the top right. Below the header is a table with the following data:

	Meno študenta	Dátum odovzdania	Čas riešenia	Percentuálna úspešnosť
1	Milan	13.5.2022	0:44	100%
2	Jožko	13.5.2022	1:9	58%
3	Tomáš	13.5.2022	0:30	50%
4	Adrian	13.5.2022	0:49	33%

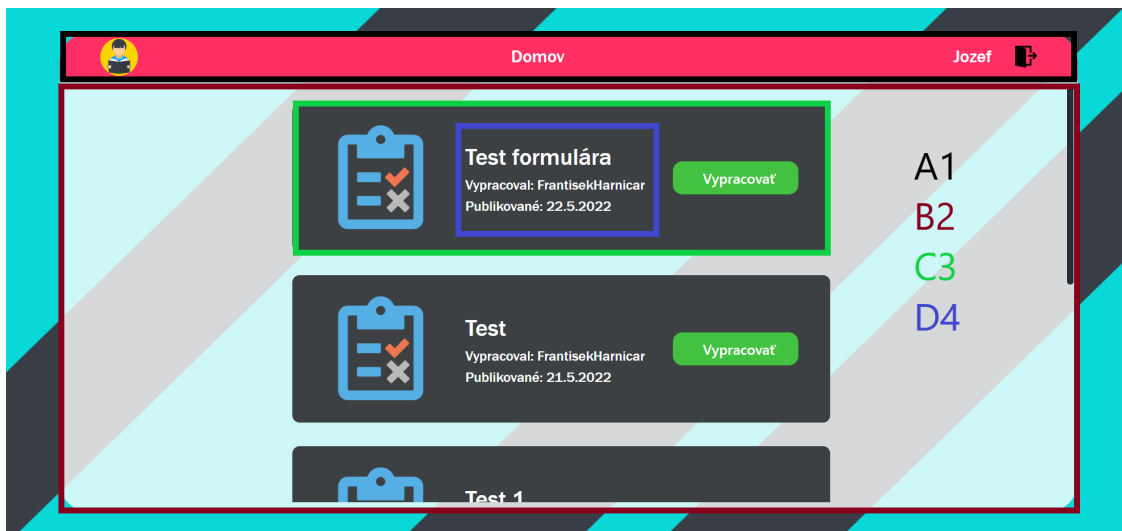
Obr. A.7: Obrázok hodnotenia testu

Po kliknutí na riadok v tabuľke sa zobrazí obrazovka, na ktorej je konkrétne riešenie študenta. V kartičkách modré oramovanie značí, čo v úlohe dosadzoval študent pri riešení. Tlačidlom ukončiť prehľad sa používateľ vráti naspäť k zobrazenej tabuľke.



Obr. A.8: Obrázok hodnotenia konkrétneho testu

A.2.6 Domovská obrazovka študenta



Obr. A.9: Domovská obrazovka študenta

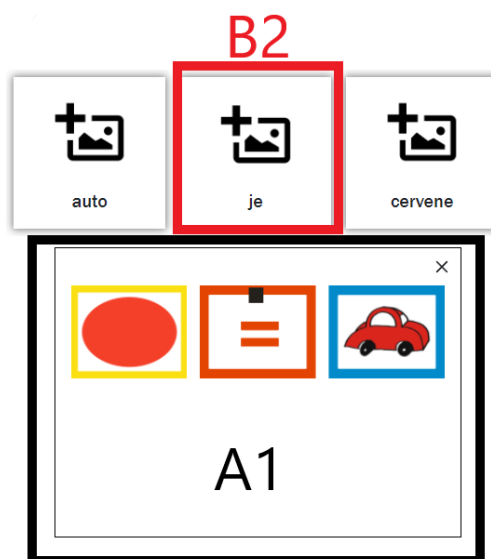
- A1** - V tejto oblasti sa nachádza piktogram pomocou, ktorého je možné vrátiť sa na domovskú obrazovku s ktorejkoľvek časti aplikácie. Hneď vedľa sa nachádza názov obrazovky, ktorá je momentálne zobrazená. Pravá strana oblasti je určená pre meno používateľa, ktoré zadal pri vstupe a tlačidlo odhlásenia. Táto oblasť je zobrazená vo všetkých častiach aplikácie.
- B2** - V tejto oblasti su zobrazené všetky dostupné testy pre študenta. Sú zoradené od toho najnovšieho.
- C3** - Táto oblasť zobrazuje samotný test. V strede sa nachádza popis testu a v pravej časti sa nachádza tlačidlo, po ktorého stlačení sa spustí vybraný test.
- D4** - V tejto oblasti je zobrazený popis testu. Nachádza sa tu názov testu, meno autora a dátum odkedy je test viditeľný.

A.2.7 Obrazovka testu študenta



Obr. A.10: Obrazovka testu študenta

- A1** - Táto oblasť obrazovky znázorňuje navigáciu medzi otázkami. Číslo s tmavým podkladom značí aktuálnu otázku v teste.
- B2** - Oblasť obsah otázky v teste. Je tvorený z kartičiek. Test je dvoj úlohový. Táto úloha je typu doplniť text. Kliknutím na text pod piktogramom v kartičke je možné text pridať.
- C3** - Táto oblasť zobrazuje v ľavej časti čas trvania testu od spstenia. V pravej časti je tlačidlo na odovzdanie testu.



Obr. A.11: Typ otázky: Doplniť obrázok

A1 - Po kliknutí na piktogram pridania obrázka v kartičke B2 sa zobrazí toto vyskakovacie okno. Kliknutím na vybraný obrázok sa pridá do kartičky. Obrázky vo vyskakovacom okne su zobrazené náhodne.

B2 - V spodnej časti kartičky je zobrazený text, podľa ktorého má študent pridať piktogram so správnym významom z okna A1.

A.2.8 Obrazovka po odovzdaní testu študenta

	Meno študenta	Dátum odovzdania	Čas riešenia	Percentuálna úspešnosť
1	steve	22.5.2022	0:53	100%
2	Jozef	25.5.2022	44:19	0%

Obr. A.12: Obrazovka po odovzdaní testu študenta

Táto obrazovka zobrazuje všetky hodnotenia študentov daného testu. V tabuľke sú zoradené hodnotenia od najvyššieho. Hodnotenie sa vypočítava tak, že sa spočítajú všetky slová a všetky správne odpovede študenta. Na základe týchto dvoch údajov sa vypočíta príslušné percentuálne hodnotenie. Tlačidlom ukončiť prehľad sa študent dostane na domovskú obrazovku.

B Systémová príručka

B.1 Funkcia programu

Aplikácia vyvinutá v rámci tejto bakalárskej práce je určená na pomoc pri výuke hendikepovaných detí pomocou ikonicko-textovej metódy. Systém je vytvorený na báze rolí učiteľa a študenta, kde rola učiteľa vie vytvárať úlohy role študenta. Systém je rozdelený na dve časti. Časť frontendu slúži na komunikáciu systému s používateľom a opačne. Frontend komunikuje ešte s backendom, ktorý komunikuje s databázou. Na základe tejto komunikácie je postavené celé fungovanie aplikácie.

B.2 Funkcie programov

B.2.1 Technické požiadavky na spustenie webovej aplikácie

Na spustenie webovej aplikácie je potrebné, aby zariadenie splňalo minimálne tieto požiadavky:

- Operačný systém Windows 7 - optimalizované na windows 10,
- webový prehliadač - optimalizované na prehliadač Google chrome
- 1Gb voľného priestoru na internom úložisku
- inštalované všetky potrebné knižnice

B.3 Preklad programu

Klientská časť aplikácie bola vyvíjaná vo frameworku React. Serverová časť bola vyvíjaná pomocou Node.js a frameworku express.js. Databázy boli riešené pomocou PostgreSQL.

B.3.1 Spustenie aplikácie

Aplikácia je v štádiu kedy nie je dostupná z externých zdrojov. Je uschovaná na médiu priloženom v záverečnej práci. Je potrebné ju z média stiahnuť a vykonať nasledujúce kroky:

1. Skopírovať aplikáciu z média a uložiť ju na interne úložisko zariadenia.
2. Je potrebné súbor aplikácie rozbaľiť, keďže je v súborovom formáte zip.
3. Je potrebné otvoriť priečinok **my-app** a v ňom vykonať pomocou príkazu **npm** inštaláciu všetkých potrebných knižníc:
 - react-bootstrap,
 - axios,
 - react-dom,
 - react-icons,
 - react-router-dom,
 - react-timer-hook,
 - react-scripts.
4. Rovnakú operáciu je potrebné vykonať v súbore Server a inštalovať všetky knižnice pomocou príkazu **npm**.
 - axios,
 - bcrypt,
 - body-parse,
 - concurrently,
 - cookie-parser,
 - cors,
 - express,
 - express-session,
 - nodemon,
 - pg.
5. Následne v priečinku **Server** je potrebné zadať príkaz **node index** pre spustenie servera.
6. V priečinku **my-app** je potrebné do konzoly zadať **npm start**, ktorým sa spustí aplikácia.

B.3.2 Zoznam zdrojových súborov klientskej časti aplikácie

Klientská časť je rozdelená do viacerých priečinkov. Sú v nich uložené zdrojové kódy aplikácie. Priečinok screens obsahuje zdrojové kódy ku všetkým obrazovkám, ako aj ku všetkým komponentom. Priečinok styles obsahuje súbory zdrojových kódov štýlov komponentov a obrazoviek.

Obsah priečinka screens:

- Addtest.js
- DBImages.js
- InputModal.js
- MainScreen.js
- PageForStudentTest.js
- PicText.js
- ProtectedRoutes.jsx
- Student.js
- StudentLogin.js
- StudentTest.js
- Table.js
- Teacher.js
- TeacherLogin.js
- TeacherRating.js
- TeacherTest.js
- TestInfo.js

Obsah priečinka styles:

- addTest.css
- App.css
- mainScreen.css
- modalStyle.scss
- PicText.css
- StudentTeacher.css
- Table.css
- teacher.scss
- tests.css
- Priečnok so všetkými obrázkami v aplikácii

B.3.3 Popis jednotlivých komponentov klientskej časti

Činnosť komponentov klientskej časti:

StudentLogin - prihlásenie študenta do aplikácie.

TeacherLogin - prihlásenie učiteľa do aplikácie.

AddTest - vytvorenie testu, úprava existujúceho testu.

DBImages - komponent na zobrazenie obrázkov z databázy pri vytváraní a úprave testu.

InputModal - komponent vyskakovacieho okna pri riešení testu študenta.

MainScreen - úvodná obrazovka.

PageForStudentTest - test študenta.

PicText - kartička tvorená obrázkom a textom v teste.

ProtectedRoutes - zabezpečuje ochranu pred vstupom do aplikácie bez prihlásenia.

Student - domovská obrazovka študenta.

StudentTest - komponent testu zobrazovaný na domovskej obrazovke študenta.

Table - tabuľka hodnotení.

Teacher - domovská obrazovka učiteľa.

TeacherRating - obrazovka hodnotenia učiteľa.

TeacherTest - komponent testu zobrazovaný na domovskej obrazovke učiteľa.

TestInfo - navigácia medzi úlohami.

B.3.4 Zoznam zdrojových súborov serverovej časti aplikácie

Hlavný súbor, ktorým sa spúšťa server sa nazýva `index.js`, ten sa nachádza v priečinku `server`. Zvyšok súborov zdrojových kódov servera sa nachádzajú v priečinku `routes`.

Názvy zdrojových kódov servera:

- `index.js`
- `handler.js`
- `student.js`
- `teacher.js`

B.3.5 Popis jednotlivých komponentov serverovej časti

Činnosť komponentov serverovej časti:

index - hlavý súbor serverovej časti, jeho spustením sa spustia všetky ostatné zdrojové súbory.

handler - vykonáva všetky úkony ohľadom prihlásenia oboch rolí aplikácie.

student - vykonáva všetky komunikácie študenta so serverom a databázou.

teacher - vykonáva všetky komunikácie učiteľa so serverom a databázou.

B.3.6 Vytvorenie databázy

Na vytvorenie tabuliek databázy je potrebné použiť tento script:

```
CREATE TABLE zaznam_testov (  
    id serial primary key,  
    nazov_testu varchar,  
    meno_ucitela varchar,  
    datum_publicacie timestamp,  
    test jsonb  
);
```

```
CREATE TABLE hodnotenie_studenta (  
    id serial primary key,  
    nazov_testu varchar,  
    meno_studenta varchar,  
    pocet_spravnych_odpovedi integer,  
    datum_odovzdania timestamp,  
    cas_riesenia varchar,  
    test jsonb  
);
```

```
CREATE TABLE zoznam_piktogramov (  
    id serial primary key,  
    nazov_piktogramu varchar  
);
```

```
CREATE TABLE prihlasenie (  
    id serial primary key,  
    meno varchar,  
    heslo varchar  
);
```

C Formulár z používateľského testovania

Používateľské testovanie

Vašou úlohou v tomto formulári bude postupovať vo webovej aplikácii podľa scenára spomenutého nižšie. Po vypracovaní scenára nasleduje vyplnenie otázok tohto formulára.

Scenár:

1. Prihláste sa do aplikácie ako učiteľ zadaním mena a hesla.
2. Otvorte prostredie na vytvorenie nového testu.
3. Test bude obsahovať dve vety. Každá veta bude v samostatnej úlohe. Prvá veta bude typu "Doplniť text", druhá zasa "Doplniť obrázok".
4. Obsah viet bude nasledovný: 1 - "auto je cervene",
2 - "kral a kralovna byvaju v hrade"
5. Test uložte s názvom "Test formulára", dátum publikácie nech je dnes.
6. Odhláste sa z aplikácie.
7. Prihláste sa do aplikácie ako študent.
8. Vyplňte a odovzdajte test, ktorý ste vytvorili.
9. Odhláste sa z aplikácie.

Obr. C.1: Scenár formulára

Bolo pre vás jednoduché orientovať sa v aplikácii? *

	1	2	3	4	5	
Nesúhlasím	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Súhlasím

Obr. C.2: Otázka: Bolo pre vás jednoduché orientovať sa v aplikácii?

Prostredie aplikácie na vás pôsobilo príjemne? *

	1	2	3	4	5	
Nesúhlasím	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Súhlasím

Obr. C.3: Otázka: Prostredie aplikácie na vás pôsobilo príjemne?

Mali ste problém s niektorým z bodov scenára? Ak áno, uveďte číslo bodu a vzniknutý problém. *

Vaša odpoveď

Obr. C.4: Otázka: Mali ste problém s niektorým z bodov scenára? Ak áno, uveďte číslo bodu a vzniknutý problém.

Existuje funkcionálnosť, ktorú by ste do aplikácie pridali? Ak áno, uveďte akú? *

Vaša odpoveď

Obr. C.5: Otázka: Existuje funkcionálnosť, ktorú by ste do aplikácie pridali? Ak áno, uveďte akú?

Je niečo, čo by ste na aplikácii zmenili? *

Vaša odpoveď

Obr. C.6: Otázka: Je niečo, čo by ste na aplikácii zmenili?