

Moderné trendy vo výučbe programovania

Modern trends in teaching programming

Miroslav Virág¹

Školiteľ: Martin Cápaj²

^{1,2}Katedra informatiky, FPV UKF, Tr. A. Hlinku 1, 949 74 Nitra,

¹miroslav.virag@student.ukf.sk, ²mcapaj@ukf.sk

Abstrakt

Článok popisuje systém, ktorý sme navrhli ako pomôcku pri výučbe programovania. Systém pozostáva z hry pre mobilné zariadenie a webovej databázovej aplikácie.

Prostredníctvom hry si žiaci nevedome precvičujú logické a analytické myslenie, priestorovú orientáciu a učia sa používať základné algoritmické konštrukcie. Údaje z hry sú odosielané do databázy a následne sprostredkované webovou aplikáciou učiteľovi, čo umožňuje individuálny prístup k žiakom.

Pri práci sme použili množstvo nástrojov, z nich najhlavnejšími sú: herný engine Unity určený na tvorbu hier, framework pre tvorbu webových aplikácií Laravel a na základe neho založený micro-framework Lumen.

Hlavným cieľom práce bolo vytvoriť pútavú hru, ktorá bude odosielať údaje na server za účelom ich analýzy.

Kľúčové slová: gamifikácia, výučba programovania, Unity, Laravel

Abstract

The article describes system developed as a tool for support of programming learning. System consists of game for mobile device and web database application.

Pupils are unconsciously developing their logical and analytical thinking, spatial orientation and are learning to use basic algorithmic constructions. The data collected from game are sent to database and presented to teacher by web application which provides individual approach to pupils.

We have used many tools during the development, the most important are: game engine Unity, web application framework Laravel and micro-framework based on Laravel called Lumen.

The main objective of our work was to develop eye-catching game which will be collecting and sending data to server in order to further analysis.

Key Words: gamification, teaching programming, Unity, Laravel.

Úvod

Štúdie uvádzajú, že videohry môžu pozitívne vplývať na schopnosti riešenia problémov, širšie získavanie znalostí a motiváciu. Výučba formou hry je veľmi vhodná, z dôvodu, že nepôsobí ako povinná aktivita [1]. Na základe tohto faktu bolo vytvorených množstvo hier na vzdelávanie z rôznych odvetví, z ktorých najpopulárnejšie sú hry na výučbu programovania.

Najväčšou inšpiráciou pre našu prácu bol projekt Robot Emil pod vedením prof. Ivana Kalaša, ktorý nám ho bol predstaviť osobne na workshope, ktorý sa konal na našej univerzite (fotogaléria workshopu dostupná na: <https://www.eteacher.sk/fotogaleria/workshop-programovanie-na-prvom-stupni-zs-je-fantasticka-prilezitost-podme-ho-robit-poriadne#>). Robot Emil je hra vytvorená v hernom engine Unity doplnená pracovnými zošitmi. Zameriava sa na výučbu programovania a rozvoj logického myslenia u žiakov 3. a 4. ročníka ZŠ a vyvíjajú sa aj prostredia pre predškolský ročník materskej školy.

Žiaci na základe zadania v pracovnom zošite riešia v tímoch a nikdy nie osamote rôzne úlohy v hre a následne prebieha diskusia o riešeníach s ostatnými tímami a učiteľom. Týmto sa rozvíjajú aj komunikačné schopnosti, tímová práca, argumentácia a schopnosť prezentovať vlastné názory.

Projekt ďalej vyvíjajú v spolupráci s učiteľmi práve na základe spätnej väzby žiakov, pričom kladú dôraz na schopnosť riešiť dané problémy tak, aby ďalšie úrovne neboli príliš jednoduché, ale aby stále z hľadiska náročnosti zostali atraktívne [2]. Viac informácií o projekte Robot Emil môžete získať na oficiálnej stránke projektu: <https://www.robotemil.com/sk/>.

My sme v našej práci zautomatizovali získavanie spätnej väzby tým, že aplikácia odosiela výsledky z hry – žiak musí mať stály prístup na internet, ktoré môžeme analyzovať a urýchliť tak vývoj úrovni potrebnej náročnosti, prípadne získať podnet na upravenie príliš náročných úrovní. Toto riešenie taktiež umožňuje učiteľovi, aby jednoducho na základe jednotlivých výsledkov identifikoval žiaka, ktorý má problémy pochopiť princíp konkrétnych prvkov

a individuálne sa mu venoval, alebo naopak objavil nadaného žiaka.

1 Unity

Je najpoužívanejší herný engine určený na tvorbu 2D a 3D hier na svete. V súčasnosti je 50% všetkých mobilných hier a 60% všetkých aplikácií pre rozšírenú a virtuálnu realitu vytvorených v Unity[3].

Najväčšou výhodou tohto prostredia je portabilita aplikácií, ktoré sú v ňom vytvorené na množstvo rôznych platforiem. Medzi hlavné platformy, na ktoré možno exportovať vytvorenú aplikáciu patria: Windows, Mac, Linux, Android a iOS. V prípade, že v kóde aplikácie nie sú využívané špecifické hardvérové prvky konkrétnej platformy (senzory na smartfónoch, vstup z klávesnice na počítači), je možné zmeniť cieľovú platformu pre vytvorenú aplikáciu bez akýchkoľvek zmien v kóde iba jednoduchými nastaveniami v rámci Unity[4].

Unity sa stále vyvíja a pribúdajú stále ďalšie prvky na zrýchlenie vývoja. K systémom zabezpečujúcim fungovanie fyziky, zvuku, networkingu, grafiky, ale aj mnohým iným, pribudol počas obdobia našej práce s týmto prostredím manažér nainštalovaných verzií Unity s názvom Unity Hub. Unity Hub umožňuje prehľadné spravovanie projektov a verzií vývojového prostredia. Pri inštalácii Unity je možné zvoliť všetky vývojové nástroje na rôzne platformy takže nie je nutná ich samostatná inštalácia.

Pre nás bolo významným prínosom doplnenie editora na vytváranie grafiky na základe „tilesetov“ keďže sme nemuseli využívať externé nástroje. **Tileset** je súbor 2D grafických prvkov šírený väčšinou ako jeden obrázok, ktorý sa následne po stiahnutí rozdelí pomocou editora na jednotlivé kusy a vytvorí sa z nich paleta. Ukladaním častí grafiky z palety do mriežky sa tvorí mapa úrovne.

Vývoj hier v Unity je postavený na troch základných stavebných blokoch: trieda `GameObject`, trieda `Component`, a Premenná [5].

1.1 Trieda `GameObject`

Každý objekt, ktorý sa v hre vyskytuje ako napríklad postavy, svetlá, špeciálne efekty, patrí pod základnú triedu `GameObject` [5].

1.2 Trieda `Component`

Prvky odvodené z triedy `Component` definujú a kontrolujú správanie tried `GameObject`, na ktorých sú pripnuté. Najbežnejšie sú: `Transform` – definuje pozíciu, rotáciu a veľkosť objektu, ktorý patrí pod

triedu `GameObject`, `Rigidbody` – zabezpečuje fyzikálne vlastnosti, `Collider` – umožňuje detekciu kolízií.

1.3 Skript

Patrí pod triedu `Component` a teda sa rovnako pripína na `GameObject`. Používa sa na implementáciu hernej logiky a v kombinácii s ostatnými prvkami, ktoré sú odvodené od triedy `Component` zabezpečuje celkové fungovanie hry.

Unity sa zameralo hlavne na podporu programovacieho jazyka C# a je inštalované spolu s vývojovým prostredím Visual Studio. Ešte pred dvoma rokmi Unity defaultne využívalo vývojové prostredie `MonoDevelop` a bolo možné si jednoducho zvoliť spomedzi troch programovacích jazykov: C#, `UnityScript` – veľmi podobný JavaScriptu a najmenej používaný bol jazyk `Boo`.

2 Laravel

Laravel je php framework na tvorbu webových aplikácií, ktorý zľahčuje vývojový proces zjednodušením repetitívnych úkonov vyskytujúcich sa vo väčšine dnešných webových aplikáciách čo zahŕňa napríklad smerovanie, autentifikáciu, kešovanie a sedenia [7]. Na zobrazovanie údajov využíva Laravel takzvaný `Blade`, ktorý na rozdiel od iných podobných nástrojov neobmedzuje vo `view-och` používanie programovacieho jazyka php.

2.1 Model-view-controller

Laravel je založený na návrhovom vzore Model-view-controller v skratke MVC, ktorý predstavili s objektovo-orientovaným programovacím jazykom `Smalltalk'80` Krasner a Pope a ktorý sa následne po publikácii koncom osemdesiatich rokov začal hromadne používať a boli na základe neho tvorené ďalšie návrhové vzory[8].

Podstatou MVC je rozdelenie aplikácie na tri hlavné vrstvy: Model, View a Controller. Model má za úlohu získavať a ukladať údaje do databázy. View slúži na zobrazovanie informácií používateľovi. Controller vybavuje požiadavky používateľa, využíva údaje, ktoré mu poskytuje Model a prostredníctvom View-u ich používateľovi zobrazuje [9].

2.2 Eloquent

Je objektovo-relačný mapovací nástroj. Definuje štandard, akým sa v rámci Laravel projektu pristupuje k údajom uloženým v databáze. Umožňuje z databázy získavať údaje prostredníctvom príkazov

programovacieho jazyka php namiesto SQL, ale umožňuje aj ich kombináciu [10].

2.2 Artisan

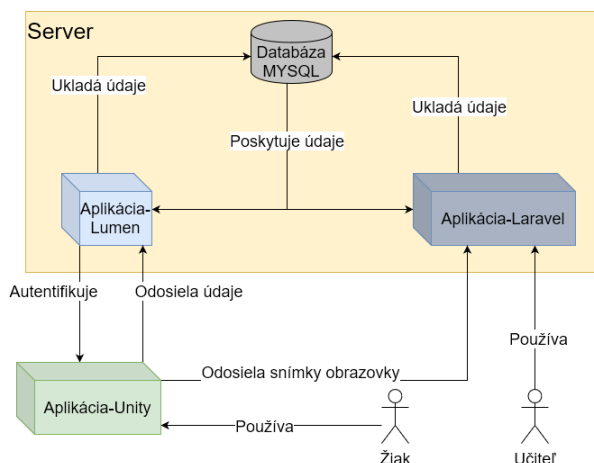
Je názov rozhrania príkazového riadku, ktoré Laravel obsahuje [11]. V ňom je možné jedným príkazom vytvoriť v projekte funkcionality, ktorá zabezpečuje autentifikáciu a sedenia. Ďalšími príkazmi je možné vytvoriť napríklad migrácie alebo seedy. Migrácie sú súbory, ktoré reprezentujú tabuľky databázy a po ich následnom nakonfigurovaní je možné jedným príkazom na základe migrácií vytvoriť na serveri celú databázu, čím zabezpečujú jednoduché verzionovanie štruktúry databázy v kombinácii s nástrojom GitHub. Prostredníctvom seedov je možné obdobným spôsobom naplniť databázu údajmi.

3 Lumen

Dokáže spracovávať až 1900 požiadaviek za sekundu, čo ho robí jedným z najrýchlejších dostupných micro-frameworkov. Keďže je založený na frameworku Laravel obsahuje niektoré rovnaké prvky ako napríklad Eloquent, validáciu či smerovanie [12]. Lumen je určený na micro-services a rýchle vybavovanie API volaní a preto neobsahuje Laravel Blade a teda nie je vhodný na zobrazovanie informácií. Avšak projekt vytvorený vo frameworku Lumen je v prípade potreby možné preniesť do frameworku Laravel prekopírovaním kódu bez akýchkoľvek zmien [12].

4 Vytvorený systém

Pozostáva z troch aplikácií a jednej MYSQL databázy. Na serveri spolu s databázou bežia dve aplikácie, ktoré s ňou komunikujú. Prvá je vytvorená vo frameworku Laravel a druhá v Lumen. Tretia aplikácia je výučbová hra, komunikujúca prostredníctvom internetu s aplikáciami vytvorenými v Laravel a Lumen. Diagram logiky nášho systému vytvorený v online programe draw.io môžete vidieť na obrázku 1.



Obr. 1 Diagram systému
Zdroj: Autor

Aplikácia vytvorená vo frameworku Laravel slúži ako webové rozhranie pre učiteľov na zobrazovanie výsledkov a štatistík žiakov, základné nastavovanie hry a tiež na správu účtov žiakov a tvorenie tried. Žiaci do nej nemajú prístup.

Aplikácia vytvorená vo frameworku Lumen komunikuje s hrou vytvorenou v Unity, slúži na autorizáciu žiakov, poskytovanie nastavení a ukladanie výsledkov a registrácií, ktoré sú jej posielané z hry do databázy.

Na tvorbu hry je použitý herný engine Unity, hra je určená pre mobilné zariadenia s operačným systémom Android.

4.1 Databáza

Pozostáva zo šiestich tabuliek jej návrh vytvorený v programe MySQL Workbench môžete vidieť na obrázku 2.



Obr. 2 Databáza systému

Zdroj: Autor

Tabuľka user slúži na ukladanie prihlasovacích a identifikačných údajov žiakov a učiteľov, riadok role rozlišuje či ide o učiteľa (hodnota 1), alebo žiaka (hodnota 0). Obsahuje cudzí kľúč tabuľky classroom, ktorý určuje, do ktorej triedy žiak patrí. Riadok api_token slúži ako unikátny identifikátor na autorizáciu žiakov kedy po každom prihlásení žiaka do vytvorenej hry je vygenerovaný nový api_token.

Tabuľka level nesie informácie o všetkých úrovniach, vytvorených v hre.

User_has_level obsahuje údaje o konkrétnych pokusoch žiaka prejsť danú úroveň. Pričom každý záznam obsahuje aj cudzí kľúč tabuľky session, ktorá nesie údaje o sedení hry. Pri každom spustení hry sa vygeneruje a uloží identifikátor nového sedenia, ktoré sa následne priradí ku každému pokusu, aby bolo možné identifikovať koľko úrovní prešiel žiak naraz.

5 Hra

Cieľom hry je vytvoriť postupnosť krokov z príkazov vo forme ikon, po ktorej vykonaní sa hlavný hrdina dostane do portálu. Hrdina spolu s hernými panelmi na ukladanie príkazov predstavuje jednoduché vývojové prostredie kedy pri zadaní príkazu, ktorý nie je možné vykonať, sa príslušné pole vysvieti na červeno, hrdina zastane a zamávaním rúk dá žiakovi najavo, že niekde nastala chyba.

Hra bola vytvorená na základe voľne dostupných grafických (tilesety a ostatné grafické prvky [13], spritesheet na animáciu hrdinu [14], ikony použité

v príkazoch a tlačidlách [15], [16], [17], font písma [18]) a zvukových prvkov [19].

Prvé úrovne sú veľmi jednoduché určené na zoznamovanie sa s prostredím a funkcionalitou. Hra obsahuje jeden hlavný panel, ktorý sa nachádza v hornej časti obrazovky a základné príkazy na ovládanie pohybu hrdinu (obrázok 3), ktoré do panelu žiak ukladá a následne sa vykonajú.



Obr.3 Úroveň 6

Zdroj: Autor

V prvej úrovni hra poskytuje iba príkaz chod', aby si žiak osvojil ukladanie príkazov do hlavného panelu, mazanie príkazov a spúšťanie sekvencie. Počet príkazov a funkcionalita hry sa postupne rozširujú v graficky rozmanitých dvadsiatich úrovniach, pričom na začiatku každej úrovne s novými prvkami, je žiak oboznámený s ich fungovaním prostredníctvom video-návodu, ktorý je v prípade potreby možné prehrať znovu, kliknutím na tlačidlo informácie. Príkazy je možné ukladáť do hlavného panelu dvoma spôsobmi:

- formou drag and drop, teda ťahania a púšťania na určité miesto,
- ťukaním na príkazy pričom tie sa defaultne zobrazia v aktívnom paneli na najbližšom voľnom mieste zľava, v prípade, že má hráč k dispozícii viac panelov, môže zvoliť panel, do ktorého chce ukladáť príkazy ťukaním, ťuknutím do panelu, pričom takýto panel má mierne jasnejšiu farbu.

Do hry postupne pribúdajú možnosti zbierať mince, povinnosť otvárať portál aktivovaním orbú alebo viacerých orbú príslušnej farby, ďalší farebne odlišný portál s príslušnými orbami, špeciálny orb, ktorý otvára oba portály a prídavné panely, kde je možné tvoriť modifikácie metód a slučiek. Orby, mince, portály a všetky pohyby hlavného hrdinu sú animované. Ukladanie príkazov, zbieranie objektov a otváranie a zatváranie portálov je sprevádzané zvukmi.

Na obrázku 4 môžeme vidieť poslednú úroveň hry kde sú obsiahnuté všetky prvky, s ktorými sa počas hry žiak stretne. Špeciálne príkazy slučka a metóda, ktoré sú v paneli pre výber príkazov bielej farby, sa využívajú uložením do poľa červenej, zelenej alebo modrej farby. Každá farba má príslušné polia vo vysúvacom paneli, ktorý je viditeľný na obrázku 4. Po uložení špeciálneho príkazu z jedného z 3 farebných polí do hlavného panelu sa v prípade, že príde na rad, vykoná sekvencia v poliach príslušnej farby, nachádzajúcich sa v prídavnom vysúvacom paneli. V teórií programovania reprezentuje tento panel podprogramy.



Obr. 4 Úroveň 20
Zdroj: Autor

Špeciálne príkazy je možné kombinovať, pričom sme kládli dôraz, aby bola dodržaná sekvenčná logika reálnych programovacích jazykov. Je možné rekurzívne volať metódy a vnárať do seba slučky a metódy.

Po úspešnom prejdení úrovne dostane žiak spätnú väzbu vo forme počtu hviezdíčiek s maximálnym počtom 3. Na získanie maximálneho počtu hviezdíčiek musí žiak obsadiť príkazmi čo najmenej polí v paneloch a získať pri tom čo najviac mincí. Hranice maximálneho počtu zaplnených polí a potrebných získaných mincí na získanie určitého hodnotenia sú preddefinované pre každú úroveň.

5.1 Zber údajov

Aby bolo možné zbierať údaje o hre, je nutný stály prístup na internet. Po spustení hry sa vyžaduje prihlásenie žiaka. Ak žiak nemá vytvorený účet, je možné sa prostredníctvom hry registrovať, v prípade že to bolo povolené vo webovom rozhraní. Následne si hra pamätá prihlásenie, takže prihlasovanie stačí vykonať iba raz. Hra samozrejme poskytuje možnosť odhlásenia, aby bolo možné prihlásiť iného hráča.

K odosielaniu údajov dochádza v dvoch prípadoch: po každom spustení sekvencie a po úspešnom prejdení úrovne. Po spustení sekvencie sa odosiela čas strávený

v hre, počet príkazov v paneloch a dokonca aj počet minimalizovaných hry. Po úspešnom prejdení hry sa k týmto údajom ešte pridá hodnotenie, počet nazbieraných mincí, počet nazbieraných orbov a snímka obrazovky, kde je možné vidieť celé riešenie úrovne.

6 Webové rozhranie

Vizuálna časť rozhrania bola vytvorená na základe voľne dostupnej Bootstrap šablóny s názvom AdminLTE [20], ktorá poskytuje množstvo nástrojov pripravených na priame využitie. Z nich sme využili tabuľky, validácie, grafy a navigáciu.

Po prihlásení rozhranie poskytuje učiteľom prehľad o výsledkoch žiakov v hre. V úvodnej obrazovke (obrázok 5) zobrazuje žiaka s najlepším a najhorším priemerom hodnotení a žiaka, ktorý aplikáciu minimalizoval najviac krát zo všetkých, čím sa môže odhaliť žiak, ktorý podvádza alebo nemá záujem o takúto formu výučby. Tieto informácie umožňujú odhalenie problémových alebo výnimočných žiakov, ktorým sa učiteľ následne môže individuálne venovať. Na tejto obrazovke je možné vidieť aj základné štatistiky ako sú počty sedení počas roka, priemerný počet pokusov v jednotlivých úrovniach, priemerný čas prejdenia jednotlivých úrovní a priemerné hodnotenie jednotlivých úrovní. Na základe týchto štatistík vieme určiť či vytvorená úroveň bola náročnosťou vhodná pre zvolenú skupinu žiakov, prípadne konkrétne úrovne upraviť alebo vytvoriť nové úrovne.

Prechody medzi obrazovkami zabezpečuje okrajová lišta. V hornej časti obrazovky sa nachádza ďalšia lišta, ktorá umožňuje prístup k nastaveniam hry a odhlásenie.

Na obrazovke nastavenia učiteľ určuje, do ktorej triedy budú patriť všetci novo registrovaní žiaci, povoľuje registrácie, prípadne môže zrušiť vyžadovanie internetového pripojenia a hru bude možné hrať aj bez prihlásenia a internetového pripojenia.

V ďalších obrazovkách má učiteľ možnosť vytvárať triedy, do ktorých sú žiaci priradovaní a ďalšie účty žiakov a učiteľov.

6.1 Tabuľky

Ako môžeme vidieť na obrázku 5, do hlavných tabuliek Žiaci, Úrovne, Triedy, ktoré rozhranie obsahuje, je umožnený prístup z okrajovej lišty. Nad všetkými tabuľkami je možné vykonávať rovnaké operácie aké vidíme na obrázku 6 a to sú: filtrovanie riadkov tabuľky podľa abecedy na základe údajov v každom stĺpci, určovanie počtu zobrazených riadkov, vyhľadávanie

riadkov tabuľky na základe zadaného slova a prechody medzi stranami tabuľky.

Tabuľka **Žiaci** (obrázok 6) obsahuje údaje o všetkých žiakoch a to sú: meno, priezvisko, prihlasovacie meno, priemerné hodnotenie a trieda, do ktorej patrí. V tabuľke je možné upraviť údaje a odstrániť žiaka alebo prejsť na ďalšie tabuľky: všetky prejdené úrovne a všetky sedenia zvoleného žiaka. Po kliknutí na tlačidlo zobrazí prejdené úrovne sa zobrazí tabuľka s úrovňami, ktoré žiak prešiel a hodnoteniami za konkrétne úrovne. Z tejto tabuľky je možné zobrazíť všetky pokusy žiaka o prejdenie úrovne alebo výlučne úspešné pokusy, tieto tabuľky obsahujú údaje zbierané z hry a v prípade, že bol pokus úspešný, ponúkajú možnosť zobrazíť snímku obrazovky konkrétneho pokusu.

Tabuľka **Úrovne** zobrazuje charakteristiky všetkých úrovní a je možné cez ňu zobrazíť všetkých žiakov, ktorí prešli zvolenú úroveň. Zobrazená tabuľka obsahuje základné údaje o žiakoch, hodnotenie za zvolenú úroveň a opäť poskytuje možnosť zobrazíť úspešné a všetky pokusy o prejdenie úrovne.

V tabuľke **Triedy** vidí učiteľ názov triedy, priemer hodnotení žiakov patriacich do tejto triedy, s možnosťou ich zobrazenia v ďalšej tabuľke identickej s tabuľkou Žiaci.

7 Výsledky a diskusia

Prostredníctvom popísaných technológií sme vytvorili systém na podporu výučby programovania, ktorý pozostáva z 20 úrovňovej hry určenej pre mobilné zariadenia s operačným systémom Android, ktorá odosiela údaje zbierané počas hry na server, aplikácie, ktorá tieto údaje ukladá do databázy a webovej aplikácie, ktorá poskytuje učiteľom prehľad o výsledkoch žiakov.

Automatická spätná väzba v podobe odosielania údajov z hry umožňuje získavať informácie aj o žiakoch, ktorí sa nezúčastnia prezenčnej výučby a poskytuje vhodný odrazový mostík pre tvorenie ďalších a optimalizáciu už vytvorených úrovní.

Záver

Vytvorený systém môže v praxi slúžiť ako doplnok výučby programovania a na rozvoj logického a analytického myslenia v piatom až siedmom ročníku základných škôl.

Na základe štúdií, ktoré sme pri našej práci objavili sme zistili, že výučba formou hry môže byť veľmi atraktívna, motivujúca a úspešná z dôvodu jej nepovinného pôsobenia.

Vytvorený systém je pripravený na pridanie nových funkcionalít ako napríklad: možnosť vytvárať nové úrovne priamo v hre alebo vo webovom rozhraní, odosielanie reťazcov všetkých použitých príkazov za účelom komplexnej analýzy dát či implementácia databázy do hry za účelom offline zbierania dát a následnej synchronizácie s online databázou.

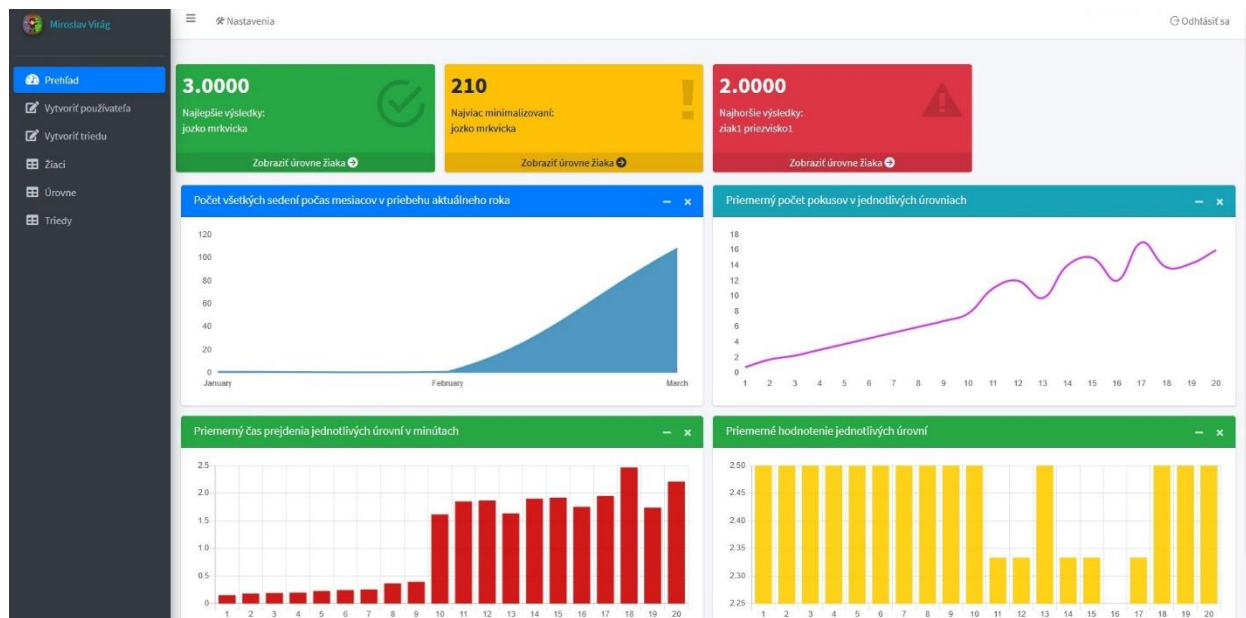
PodĎakovanie

Ďakujem môjmu školiteľovi Mgr. Martinovi Cápayovi, PhD. a konzultantovi doc. Mgr. Martinovi Drlíkovi, PhD. za cenné rady pri písaní tohto článku.

Použitá literatúra

- [1] Perrotta, C., Featherstone, G., Aston, H. and Houghton, E. (2013). Game-based Learning: Latest Evidence and Future Directions (NFER Research Programme: Innovation in Education). Slough: NFER. [Citované: 9.marec 2020]
- [2] Prof. Ivan Kalaš: Emil prináša zmenu. Nie len na Slovensku [Citované: 9.marec 2020] <https://www.ucn.sk/vzdelavanie/prof-ivan-kalas-emil-prinasa-zmenu-nie-len-na-slovensku>
- [3] Public Relations [Citované: 9.marec 2020] <https://unity3d.com/public-relations>
- [4] Porting a Project Between Platforms [Citované: 9.marec 2020] <https://docs.unity3d.com/520/Documentation/Manual/HOWTO-PortingBetweenPlatforms.html>
- [5] Scripting in Unity for experienced programmers [Citované: 9.marec 2019] <https://unity.com/how-to/programming-unity>
- [6] Web application development with Laravel PHP Framework version 4 [Citované: 10.marec 2020] <https://www.theseus.fi/bitstream/handle/10024/74052/Author.pdf>
- [7] Blade Templates [Citované: 10.marec 2020] <https://laravel.com/docs/5.8/blade>
- [8] A Journey Through the Land of Model-View-* Design Patterns [Citované: 10.marec 2020] <http://homepage.lnu.se/staff/daweaa/papers/2014WICSA.pdf>
- [9] Web development using C# MVC and ExtJS [Citované: 10.marec 2020] https://repository.stcloudstate.edu/cgi/viewcontent.cgi?article=1006&context=csit_etds
- [10] Eloquent [Citované: 10.marec 2020] <https://laravel.com/docs/5.8/eloquent#eloquent-model-conventions>

- [11] Artisan Console [Citované: 10.marec 2020]
<https://laravel.com/docs/5.1/artisan>
- [12] Lumen [Citované: 10.marec 2020]
<https://lumen.laravel.com/>
- [13] Open game art [Citované: 26.marec 2020]
<https://opengameart.org/>
- [14] Character generator [Citované: 26.marec 2020]
<http://gaurav.munjial.us/Universal-LPC-Spritesheet-Character-Generator/#>
- [15] Kenney [Citované: 26.marec 2020]
<http://kenney.nl/assets/game-icons>
- [16] Flaticon [Citované: 26.marec 2020]
<https://www.flaticon.com/>
- [17] Icons 8 [Citované: 26.marec 2020]
<https://icons8.com/>
- [18] Font [Citované: 26.marec 2020]
<https://www.fontspace.com/dieter-steffmann/devinne-swash>
- [19] Freesound [Citované: 26.marec 2020]
<https://freesound.org/>
- [20] AdminLTE [Citované: 26.marec 2020]
<https://adminlte.io/>



Obr. 5 Úvodná obrazovka webového rozhrania

Zdroj: Autor

Počet riadkov v tabuľke 10

Vyhľadať v tabuľke

ID	Meno	Priezvisko	Login	Priemer bodov	Trieda	Akcia
2	jozko	mrkvicka	jmrkva@mail.com	3.0000	5.A	Zobrazit prejdené úrovně Zobrazit sedenia Upravit Odstranit
3	ziak1	priezvisko1	login1	2.0000	5.B	Zobrazit prejdené úrovně Zobrazit sedenia Upravit Odstranit
4	ziak2	priezvisko2	login2	2.0000	5.B	Zobrazit prejdené úrovně Zobrazit sedenia Upravit Odstranit
5	ziak3	priezvisko3	login3		5.B	Zobrazit prejdené úrovně Zobrazit sedenia Upravit Odstranit
6	ziak4	priezvisko4	login4		5.B	Zobrazit prejdené úrovně Zobrazit sedenia Upravit Odstranit
7	ziak5	priezvisko5	login5		5.B	Zobrazit prejdené úrovně Zobrazit sedenia Upravit Odstranit
8	ziak6	priezvisko6	login6		5.B	Zobrazit prejdené úrovně Zobrazit sedenia Upravit Odstranit
9	ziak7	priezvisko7	login7		5.B	Zobrazit prejdené úrovně Zobrazit sedenia Upravit Odstranit
10	ziak8	priezvisko8	login8		5.B	Zobrazit prejdené úrovně Zobrazit sedenia Upravit Odstranit
11	ziak9	priezvisko9	login9		5.B	Zobrazit prejdené úrovně Zobrazit sedenia Upravit Odstranit

Stránka 1 z 2

Predchádzajúca strana 1 2 Nasledujúca strana

Obr. 6 Tabuľka Žiaci
Zdroj: Autor