

Autoškola

Projekt ITU, 2016Z

Číslo projektu: 5

Číslo a názov tímu: 139. xtomas32

Autor: Adrián Tomašov (xtomas32)

Další členové tímu: xbarna02, xurban66

Termín řešení: 19. 9. - 19. 12. 2016

Abstrakt

Cieľom projektu je vytvoriť efektívnejšiu aplikáciu pre výučbu nových motoristov, ktorí sa nie len naučia správne testové odpovede, ktoré potrebujú k úspešnému absolvovaniu skúšky, ale aj precvičia svoje zručnosti pri riešení križovatiek. Pre nových vodičov je často náročné riešiť komplexné križovatky, preto sa chceme zamerať na to, aby riešenie rôznych križovatiek nebol žiadny problém. Preto sa v našej aplikácii bude dať zvoliť možnosť precvičovania menších oblastí testových otázok, napríklad spomínané križovatky, dopravné značenie a podobne. Aplikácia bude mať jedinečný vzhľad, aby sa užívateľ príjemne cítil pri učení. Ak zabezpečíme komfort užívateľského rozhrania, veríme, že dosiahneme pozitívne výsledky.

Cílové požiadavky na aplikaci a její rozhraní

Cieľom našej aplikácie je vytvoriť pohodlné rozhranie medzi užívateľom a látkou, ktorú sa má naučiť. Ak dosiahneme dostatočne prehľadné a efektívne rozhranie, užívateľ bude lepšie chápať preberanú látku, čím by sme mohli dosiahnuť menšiu nehodovosť na cestách. Avšak naším hlavným cieľom je zefektívnenie prípravy na testy predovšetkým pre študentov autoškoly, ako aj pre kohokoľvek ľubovoľného veku, kto si bude chcieť precvičiť znalosti autoškoly a testových otázok. Zefektívnenie prípravy by malo viesť k zaisteniu väčšieho percenta úspešných žiakov v autoškole, ktorí budú lepšie a ľahšie integrovaní do sveta motorizmu.

Studium cílove skupiny a případy použití

Aplikáciu tvoríme hlavne pre študentov autoškoly. Najväčšiu časť budu tvoriť mladí ľudia od 16 do 25 rokov. Táto skupina ľudí používa moderné technológie v každodennom živote, preto by nemali mať žiadny problém s ovladaním aplikácie. Aplikácia bude navrhnutá hlavne pre využívanie na PC alebo notebookoch, ale pozmeniť UI, aby bol vhodný aj pre telefón je jednou z možností. Mala by mať aj interakciu pre vstup z dotykového displeja, keďže je ním dnes väčšina zariadení vybavená. Jediným obmedzením je počet aktívnych tlačidiel, pretože chceme jednoduchú intuitívnu aplikáciu a nie komplikovaný software pre raketových inžinierov.

Existujúci řešení

Na internete je dnes mnoho online webových riešení. Sám som niektoré z nich používal. Výhodou bolo, že testy boli aktuálne a boli vyhodnotené ihneď po ukončení testu. Na druhej strane väčšine aplikácií chýbali rôzne funkcie ako napríklad test iba z niektorých tématických okruhov (dopravné značenie, predpisy, križovatky). Chýbala im napríklad aj databáza starých výsledkov, pre porovnávanie zlepšenia skúseností užívateľa, funkcia tlačenia testov a podobne. Preto by sme radi tieto funkcie implementovali do našej aplikácie.

Návrh GUI

V našej implementácii riešenia cheme vytvoriť mnoho funkcií, ktoré budú prehľadné, aby sa užívateľ necítil zahltený a vedel plynulo používať aplikáciu. Jednou z nich je aj náhodné generovanie testov podľa kategórie, nezávisle od testov, ktoré sú používané priamo na skúškach na získanie oprávnenia na vodičský preukaz. Tak isto si tým môže precvičiť znalosti o cestnej premávke. Ďalšou funkciou je generovanie názorných obrázkových situácií križovatiek s následným vysvetlením pre hlbšie pochopenie. Podobná funkcia bude generovať aj obrázky značiek, s popisom ich významu. Nebude chýbať ani funkcia na vyhodnocovanie výsledkov a bude generovať štatistiky úspešností. Každá chybná odpoveď bude zachytená vo funkcii, ktorá si bude chybné odpovede ukladať a následne bude obsahovať ich opätovné vysvetlenia a kategórie z ktorej bola chybná odpoveď zachytená. Bude môcť generovať testy s otázkami, v ktorých mal užívateľ najviac chýb. Väčšina z týchto funkcií sa bude zobrazovať užívateľovi pod click-able tlačítkom, ktoré vygeneruje test, križovatku, prípadne značku. Užívateľ si v štatistikách môže zobrazíť koľko testov spravil úspešne, teda získal aspoň 50 z 55 bodov. Nakoniec si užívateľ

budemôcť pustiť znovu a uvidí či sa v nich zlepšuje alebo nie.

Návrh a implementace back-endu

Použijeme SQL databázu, ktorá bude napojená na rozhranie pomocou mnohých funkcií. Databázu používame pretože bude pracovať s väčším počtom dát keď sa aplikáciu uvedie do funkčnosti. Aj keď pri testoch nebude plne využívaná. Back-ende bude pozostávať z Javascriptu – Node.js, prípadne PHP, ktoré budú komunikovať s MySQL. Ako API na komunikáciu s front-endu a back-endu využijeme JSON, nakoľko je to jednoduchšie ako pracovať s XML. Funkcia na generovanie testov pošle požiadavku do databázy na potrebné dáta a tie sa následne predajú front endu v polospracovanej forme pomocou JSON.

Návrh užívateľských testů

Najvhodnejším spôsobom bude iteratívne testovanie. Budeme porovnávať rýchlosť odozvy na užívateľské rozhranie u ľudí v rôznych vekových kategóriách. Ak budú výsledky porovnávaní kladné, prípadne lepsie ako ppredachádzajúce, vieme, že náš projekt napreduje správnym smerom. Ak nie, položíme testovacím subjektom množinu otázok, ktoré nám pomôžu naviesť projekt do cieľa. Užívateľ je pokojný práve vtedy, keď dokáže pohodlne a hlavne rýchlo ovládať aplikáciu, takže by nemal dlho rozmýšľať nad tým ako čo spraviť. Samozrejme aj rozhranie musí byť pekné, mal motiváciu sa k aplikácií ešte niekedy vrátiť alebo ju aspoň doporučiť svojim známym. Preto medzi kľúčové prvky patrí napríklad mnoho animovaných obrázkov a podobne. Vzorku užívateľov budeme hľadať hlavne v autoškole. V neskorších fázach testovania by bolo vhodné požiadať autoškolu o spoluprácu na testovaní. Aj inštruktory v autoškole by nám s testovaním mohli pomôcť, pretože majú veľa rokov skúseností s vyučovaním. Užívatelia sa budú pokúšať o základné ovládanie aplikácie. To pozostáva zo spustenia si testu, prezrieť si výdledky predchádzajúcich testov, orientácia v rozhraní a pod. K testom by malo byť ešte zahrnuté avygenerovanie a pustenie si testu na základe zamerania(križovatky, značky), prípadne test z najčastejšie nesprávne zodpovedaných otázok.

Nástroje pro tvorbu GUI

Pre náš projekt je najvhodnejšie použiť webové technológie, pretože sú dostupnejšie a univerzálnejšie ak sa zameriame na cross-platform aplikácie. Aplikácie sa nemusí inštalovať, stačí si otvoriť webový prehliadač a môžete používať aplikáciu. Nevýhodou sú staršie prehliadače, pretože nemusia podporovať najnovšie technológie, takže sa výsledná stránka nemusí zobrazit správne alebo podporovať všetky funkcie

stránky. Vývoj webových stránok je jednoduchší s v porovnaní s vývojom kernelu. Na CSS3 použijeme bootstrap a na javascript použijeme podpornú knižnicu jQuery a iné. Tieto rozšírenia používame najmä pre rýchlejší a menej náročný agilný vývoj aplikácie. HTML5 prezentuje dáta samotné, ktoré následne CSS3 upraví do krajšej formy a javascript s pomocou jQuery vtvoria animované rozhranie pre príjemnejší pocit užívateľa.

Testovací protokol - Autoškola

Verzia aplikácie : _____

Dátum : _____

ID testera : _____

1. Ako dlho trvalo subjektu spustiť test typu B a ako naňho pôsobilo rozhranie ?

2. Ako dlho mu trvalo vypracovať test a nerušili ho elementy pri jeho vyplňaní ?

3. Ako dlho mu trvalo nájsť rozhranie pre návrat k nesprávnej odpovedi ?

4. Ako zložitý mu prišiel tento test a nestratil sa v orientácii v GUI ?

5. Ako sa mu páčil vzhľad štatistík a našiel všetky potrebné informácie, ktoré hľadal ?

6. Ako sa mu páčil test s križovatkami a pomohli mu animácie pri rýchlosti vyplňania ?

7. Za ako dlho dokončil test so značkami a ako sa mu páčilo toto rozhranie ?

8. Ako dlho hľadal v menu možnosť štatistík a našiel všetky požadované informácie ?

9. Ako sa zmenila jeho úspešnosť na základe vygenerovaného testu podľa jeho znalostí ?

10. Ako sa mu páčilo rozhranie na výber otázok z kategórii a vedel čo má robiť ?