

Slovenská technická univerzita
Fakulta informatiky a informačných technológií
Ilkovičova 2, 842 16 Bratislava 4

Virtuálna FIIT

Breakpoint
Dokumentácia k dielu

Vedúca tímu: Mgr. Alena Kovárová, PhD.

Členovia tímu: Bc. Filip Mazán, Bc. Veronika Olešová, Bc. Filip Šoltés, Bc. Michal Kučera,
Bc. Jozef Karas, Bc. Daniel Pribul

Školský rok: 2014/2015

Obsah

1	Úvod.....	1-1
2	Slovník pojmov.....	2-1
3	Globálne ciele pre zimný semester	3-1
4	Celkový pohľad.....	4-1
4.1	Pôvodná verzia.....	4-1
4.2	Naše zmeny	4-3
4.2.1	Stará aplikácia.....	4-3
4.2.2	Nová aplikácia	4-4
5	Moduly Virtual FIIT	5-1
5.1	AIS	5-1
5.1.1	Pozrieť sa na šablóny AIS, prečo parser zlyháva	5-1
5.1.2	Kompletne prerobiť interakciu s AIS	5-2
5.2	Jedálne.....	5-3
5.2.1	Oprava chyby - Zobrazovanie obedov	5-3
5.2.2	Obedy - oprava pamäte cache	5-4
5.2.3	Spraviť stránku s obedmi, zatiaľ po dizajnovej stránke netreba.....	5-4
5.3	Rozvrh	5-5
5.3.1	Fix prihlásenia do AIS - treba mazať pamäť cache	5-5
5.3.2	Prerobenie rozvrhu.....	5-5
5.4	MHD.....	5-6
5.5	Vyhľadávanie	5-6
5.6	Mapy.....	5-7
5.6.1	Spraviť stránku s mapu budovy, zatiaľ SVG formát + tlačidlá poschodí.....	5-7

Obsah

5.7	QR Kódy	5-8
5.8	BLE lokalizácia	5-8
5.8.1	Vytvorenie zásuvného modulu na detekciu a zobrazenie BLE zariadení.....	5-8
5.8.2	Vybudovať aplikáciu na zber, zozbierať dáta.....	5-9
5.9	Knižnica	5-10
5.9.1	Úradné hodiny knižnice	5-11
5.10	Harmonogram.....	5-11
5.10.1	Zmeniť neprehľadný harmonogram.....	5-11
5.11	O aplikácii.....	5-12
5.11.1	Spraviť stránku O aplikácii s funkčnými linkami.....	5-12
5.12	Nahlasovanie chýb.....	5-13
5.12.1	Spraviť nahlasovanie chýb s funkčným formulárom a hláškami používateľovi .	5-13
5.13	Domovská obrazovka	5-13
5.13.1	Spraviť úvodnú stránku (homescreen) s ikonkami	5-14
5.14	Ostatné	5-14
5.14.1	Vytvoriť localStorage factory	5-15
Príloha A	Diagramy obrazoviek.....	A-1

1 Úvod

Určite sa už každý z vás aspoň raz ocitol v situácii, kedy sa dostal do nového prostredia a cítil sa úplne stratený. Nepomohli vám ani navigačné tabule či ľudia naokolo. Sme presvedčení, že práve my vám dokážeme ponúknuť riešenie na tento problém – vyvíjame mobilnú aplikáciu s mapami, navigáciou a mnohými ďalšími informáciami o budove a dianí v nej. Je celkom logické, že ako prvá budova, pre ktorú to robíme je budova našej školy, preto sa aj náš projekt volá Virtuálna FIIT. Takže momentálne je aplikácia určená predovšetkým študentom prvého ročníka, ktorí sa takisto cítia často na začiatku semestra stratení. Samozrejme, vývoj riadime tak, aby sa celé riešenie dalo jednoducho preniesť na akúkoľvek inú budovu, či už školu, úrady, nemocnice, nákupné centrá, biznis centrá alebo veľké firemné budovy. Do tohto projektu bolo investovaného už veľa úsilia našimi predchodcami. Takže už dnes aplikácia poskytuje študentom mnoho funkcionalít, medzi ktoré patrí napríklad prezeranie si máp fakulty, vlastného rozvrhu, obedov v okolí či odchod najbližších autobusov. My máme v pláne ich udržiavať, vylepšovať a rozšíriť o nové.

V tomto dokumente je okrem globálnych cieľov stanovených na zimný semester aj podrobný popis vyvíjanej aplikácie, jej celková architektúra a všetky jej moduly. V architektúre je najprv popísaný stav, v akom sme aplikáciu dostali, a potom aj stav v akom sa aplikácia aktuálne nachádza. Každý modul potom obsahuje zoznam používateľských príbehov a podrobný popis úloh, ktoré sa v rámci nich riešili.

2 Slovník pojmov

Výraz	Vysvetlenie
parsovanie	proces, pri ktorom sa vstupný text transformuje na určité dátové štruktúry
parser	nástroj, ktorý vykonáva parsovanie
cache	vyrovnávacía pamäť
CORS	skratka pre <i>Cross-origin resource sharing</i> ;
BLE	skratka pre <i>Bluetooth low energy</i>
API	skratka pre <i>Application programming interface</i>
PhoneGap	voľne dostupný rámec na tvorbu mobilných aplikácii pomocou webových technológií
Cordova	súbor API, umožňujúci pristupovať vývojárom k natívnym funkciám zariadenia
front-end	časť aplikácie viditeľná bežným návštevníkom
AIS	skratka pre <i>Akademický informačný systém</i>

3 Globálne ciele pre zimný semester

Najväčší potenciál vidíme v lokalizácii a následnej navigácii používateľa v budove. Toto sme si na začiatku semestra stanovili ako svoj hlavný cieľ v tomto projekte, pretože bez tejto funkcionality sa žiadny informačný systém viazaný na komplexné budovy neobíde. Takúto navigáciu rozhodne ocenia používatelia, ktorí neradi strácajú čas študovaním mapy. Na otvorených priestranstvách sa na tento účel používa GPS signál, ten však vo vnútorných priestoroch zlyháva, a teda je potrebné nájsť iné riešenie lokalizácie. Spomedzi existujúcich technológií nám vyšlo najpraktickejšie použitie nízkoenergetických vysieláčov – Bluetooth LE Beacon-ov. Existujúce implementácie takéhoto riešenia nie sú dostatočne presné, čo sme prijali ako výzvu pre implementáciu vlastnej navigácie. Spôsobov, akým je možné problém uchopiť je viacero. Vzniká možnosť poňať projekt výskumne a zistiť, ktorá metóda by bola v danom prostredí najvhodnejšia. Do konca zimného semestra plánujeme vytvoriť prototyp, v ktorom bude fungovať lokalizácia v rámci miestnosti a do konca školského roka plne fungujúci modul, ktorý bude spolu s celou doterajšou aplikáciou opäť nasadený na Google Play obchod.

Menej náročným, ale snád' rovnako dôležitým cieľom je aj prepojenie našej aplikácie so systémom Askalot (portál pre študentov na kladenie otázok a vyhľadávanie/získanie odpovedí). Tiež plánujeme vyriešiť problém so zabezpečením spojenia (to nebolo možné zabezpečiť kvôli obmedzeniam produkčného servera) tým, že si nainštalujeme svoj vlastný produkčný server a zoptimalizujeme tak aj celú komunikáciu medzi klientom a serverom. V prípade záujmu iných fakúlt STU radi rozšírime našu aplikáciu aj pre nich a to s minimálnou námahou.

V neposlednom rade medzi naše ciele patrí získanie schopnosti práce v tíme. Na to, aby sme boli v tomto projekte úspešní, je potrebná dobrá tímová komunikácia, spolupráca a pochopenie jeden druhého.

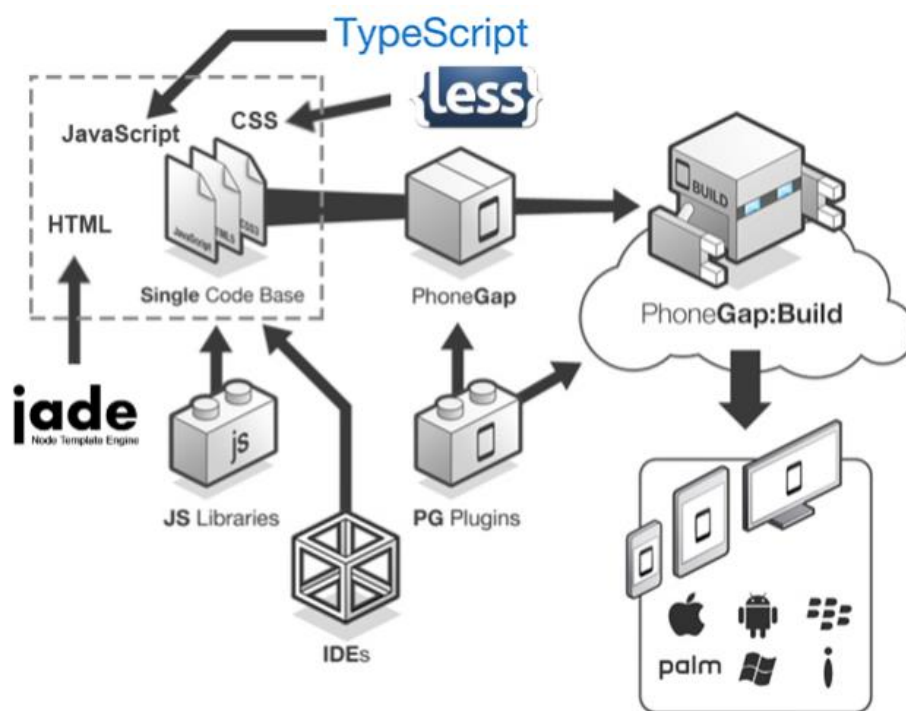
To čo sa nám v priebehu zimného semestra podarilo spraviť, je podrobnejšie rozpísané v kapitole *4.2 Naše zmeny*.

4 Celkový pohľad

4.1 Pôvodná verzia

Zdedili sme aplikáciu, na ktorej pred nami pracovalo už niekoľko tímov. Verzia, ktorú sme dostali bola postavená na technológii PhoneGap, ktorá umožňuje vytvárať aplikácie pre rôzne mobilné platformy pomocou bežných webových technológií ako sú HTML, JavaScript alebo CSS. Aplikácia VirtualFiit, žiaľ podporuje iba operačný systém Android od verzie 2.3.3. Táto aplikácia je taktiež dostupná vo forme webovej stránky.

Na zjednodušenie práce s kaskádovými štýlmi bola využitá knižnica Less, na doplnenie JavaScript-u o OOP prvky knižnica TypeScript a šablóny boli vytvárané pomocou šablónového agregátu xjade. Na zjednodušenie vývoja aplikácie bol použitý Grunt.js.



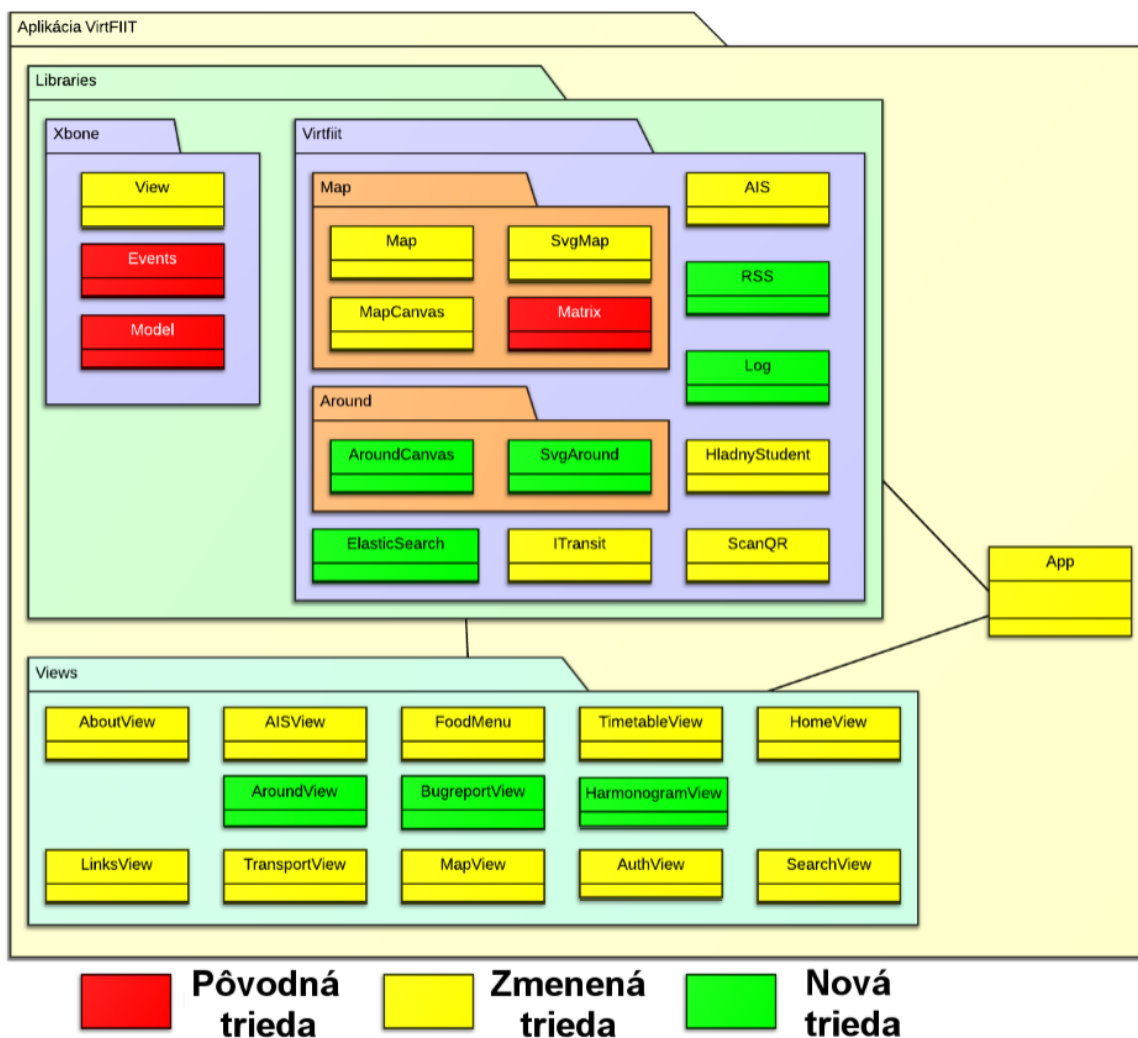
Obr. 1 Diagram použitých technológií a ich prepojenia (zdroj: minuloročný tím)

V aplikácii sa využívajú princípy MVC, kde dátový model, riadiaca logika a používateľské rozhranie boli od seba oddelené. Organizácia dostupných tried je zobrazená na Obr. 2. Hlavnou triedou aplikácie je trieda *App*.

4 Celkový pohľad

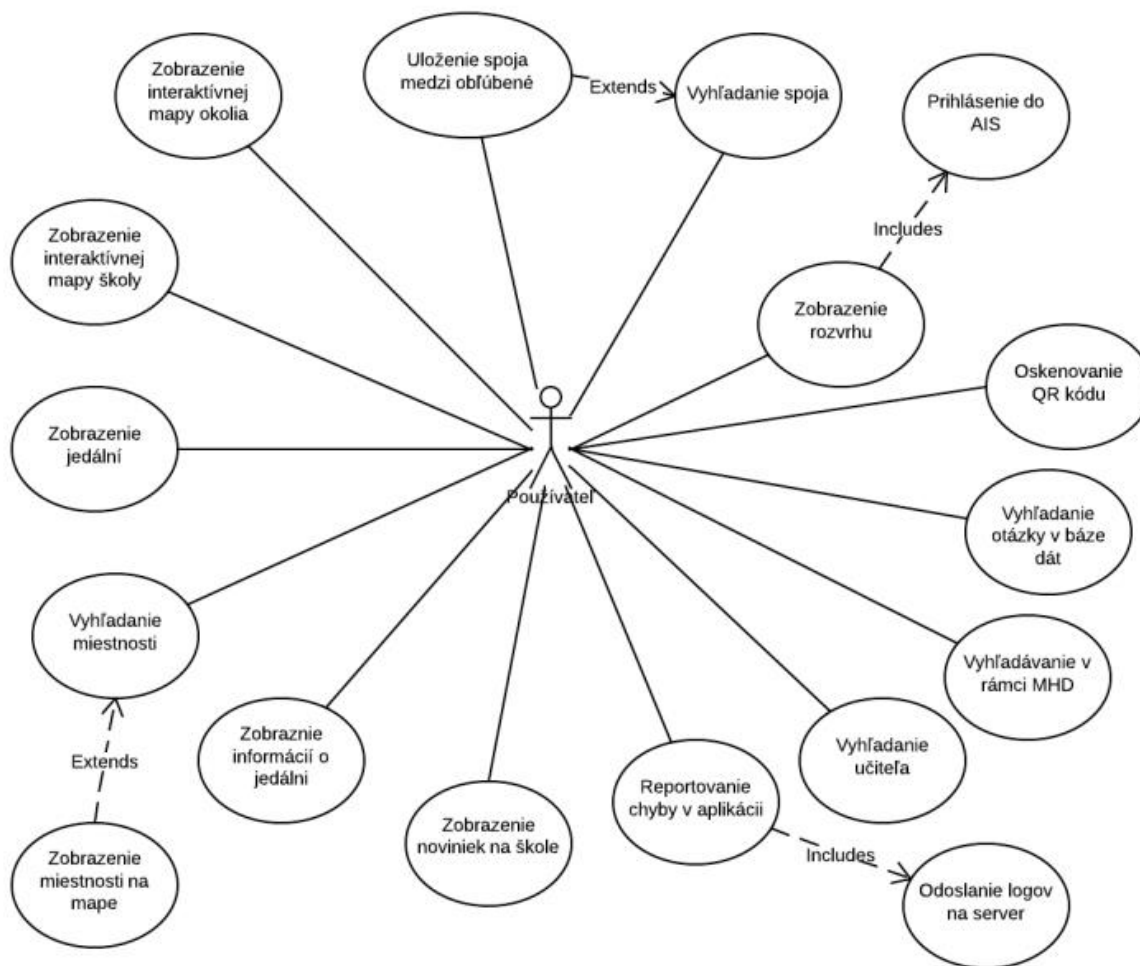
Aplikácia pozostávala z týchto modulov: *AIS*, *Jedálne*, *Rozvrh*, *MHD*, *Vyhľadavanie*, *Mapy*, *QR kódy*, *BLE lokalizácia* a ďalších informačných obrazoviek. Bližší popis pôvodného stavu a našich zmien v týchto moduloch sa nachádza v kapitole *Moduly Virtual FIIT*.

Aplikácia pracuje aj s informáciami tretích strán. Od *is.stuba.sk* a *fiit.stuba.sk* získava parsovaním údaje o škole. Z *itransit.sk* a *hladnystudent.sk* získava prostredníctvom API cestovné poriadky a jedálne lístky.



Obr. 2 Diagram balíkov s vyznačenými zmenami predošlého tímu (zdroj: minuloročný tím)

4 Celkový pohľad



Obr. 3 Diagram prípadov použitia identifikovaný predošlým tímom (zdroj: minuloročný tím)

4.2 Naše zmeny

V priebehu semestra sme sa rozhodli ukončiť vývoj starej verzie aplikácie a začať vyvíjať odznovu. Naša zmeny, ktoré sme v rámci tohto projektu vykonali, sa preto dajú rozdeliť na dve časti. Na práce vykonané v starej verzii aplikácie a na vývoj novej verzie.

4.2.1 Stará aplikácia

Pomerne veľké množstvo času sme strávili opravou chýb, ktoré sme objavili v pôvodnej verzii projektu. Išlo o nefungujúce prihlasovanie pre študentov s vlastnými šablónami pre AIS, chybné zobrazovanie a načítavanie jedálnych lístkov a problém pri mazaní rozvrhu z *cache*.

Prihlasovanie bolo presmerované cez náš server, kde sa využíva menej striktný parser. Prihlasovacie údaje sa budú po novom posielat' cez protokol https. Zmeny sú zobrazené na *Obr. 8*.

Niektoré obrazovky v aplikácii sme považovali za neprehľadné a rozhodli sme sa ich zmeniť. Zatiaľ sme upravili harmonogram a rozvrh, pričom pri rozvrhu sa okrem vzhľadu pridávali aj ďalšie obrazovky pre detail predmetu/cvičenia a pre pohľad na celý týždeň. Tieto zmeny sú zachytené na *Obr. 7*.

Keďže *PhoneGap* neobsahuje žiadne vhodné zásuvné moduly na prácu s BLE technológiou, vytvorili sme nový *Cordova* zásuvný modul s názvom *BeaconPlugin*. Ten zatiaľ dokáže skenovať BLE zariadenia v okolí.

4.2.2 Nová aplikácia

Po 3. šprinte sme sa rozhodli vytvoriť aplikáciu Virtuálna FIIT nanovo. Dôvodov bolo viacero. V prvom rade sa nám nepáčil technický stav aplikácie, spôsob akým bola naprogramovaná a použité technológie, ktoré spôsobovali množstvo problémov a zbrzdžovali vývoj. V druhom rade to bol aj zastaralý vzhľad aplikácie a jej slabá použiteľnosť. Rozhodli sme sa preto začať odznovu, použiť iné technológie, zmeniť dizajn a prehodnotiť niektoré funkcie. Pri výbere vhodnej technológie sme kládli dôraz na multiplatformovosť, pohodlnejší vývoj a plynulejší chod aplikácie. Rozhodli sme sa pre rámec *Ionic*¹, ktorý tieto naše podmienky spĺňal. Ten má v sebe už zabudovaný JavaScript rámec *AngularJS*² a CSS rozšírenie *Sass*³. *Ionic* navyše obsahuje množstvo predpripravených CSS tried, ktoré výrazne urýchľujú prototypovanie obrazoviek. Pri práci so šablónami sme sa rozhodli vynechať nástroj *XJade* a využívať budeme iba HTML doplnené o direktívy z *AngularJS*. *Ionic* využíva takisto technológiu *PhoneGap*, čiže tá zostala nezmenená. Použité technológie a ich prepojenie je znázornené na *Obr. 5*.

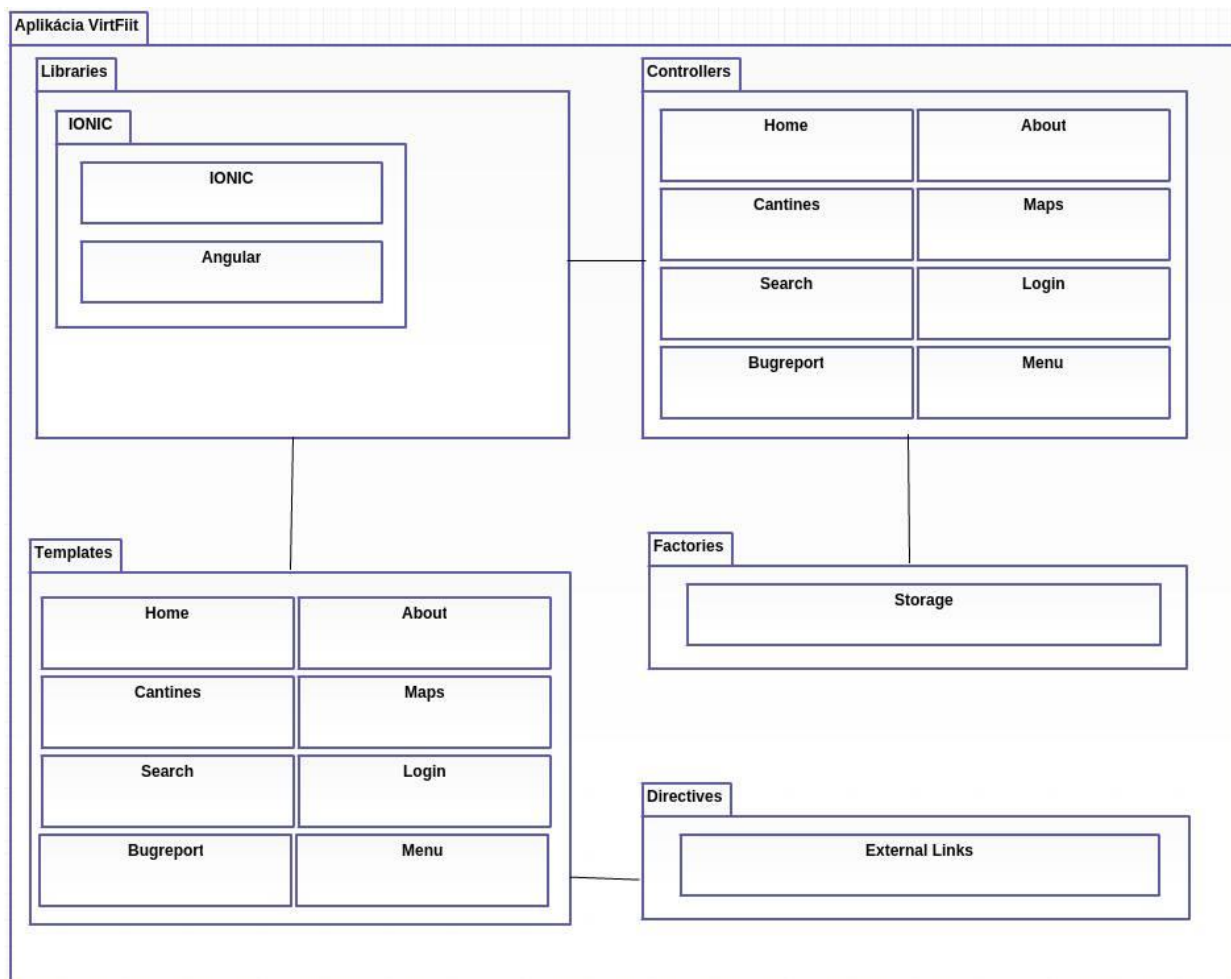
Architektúra nového systému je vyobrazená na *Obr. 4*. V novej aplikácii nebudeme využívať žiadne triedy ani inú funkcionálnosť zo starej.

¹ <http://ionicframework.com/>

² <https://angularjs.org/>

³ <http://sass-lang.com/>

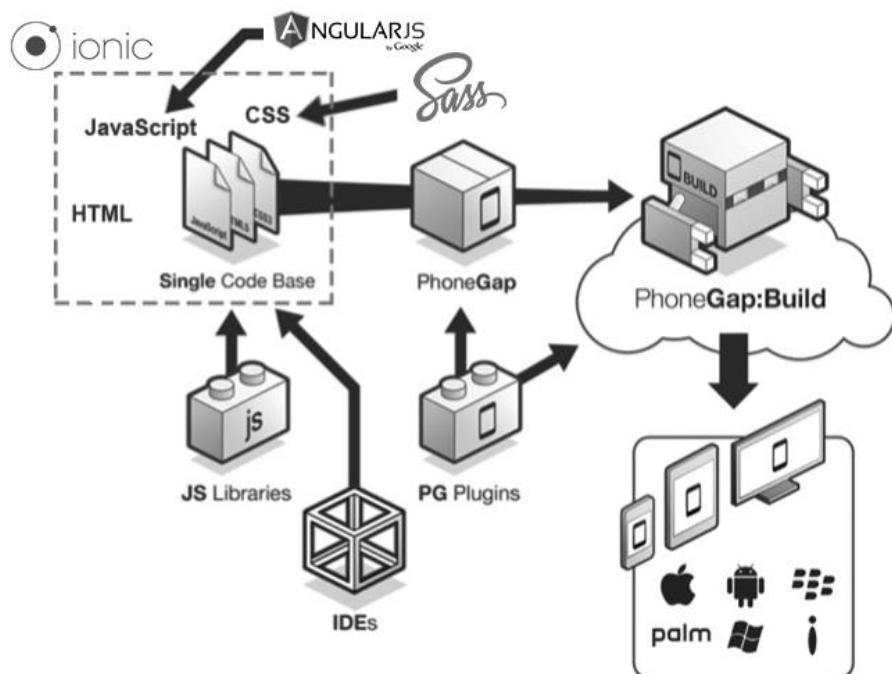
4 Celkový pohľad



Obr. 4 Diagram balíkov nového systému

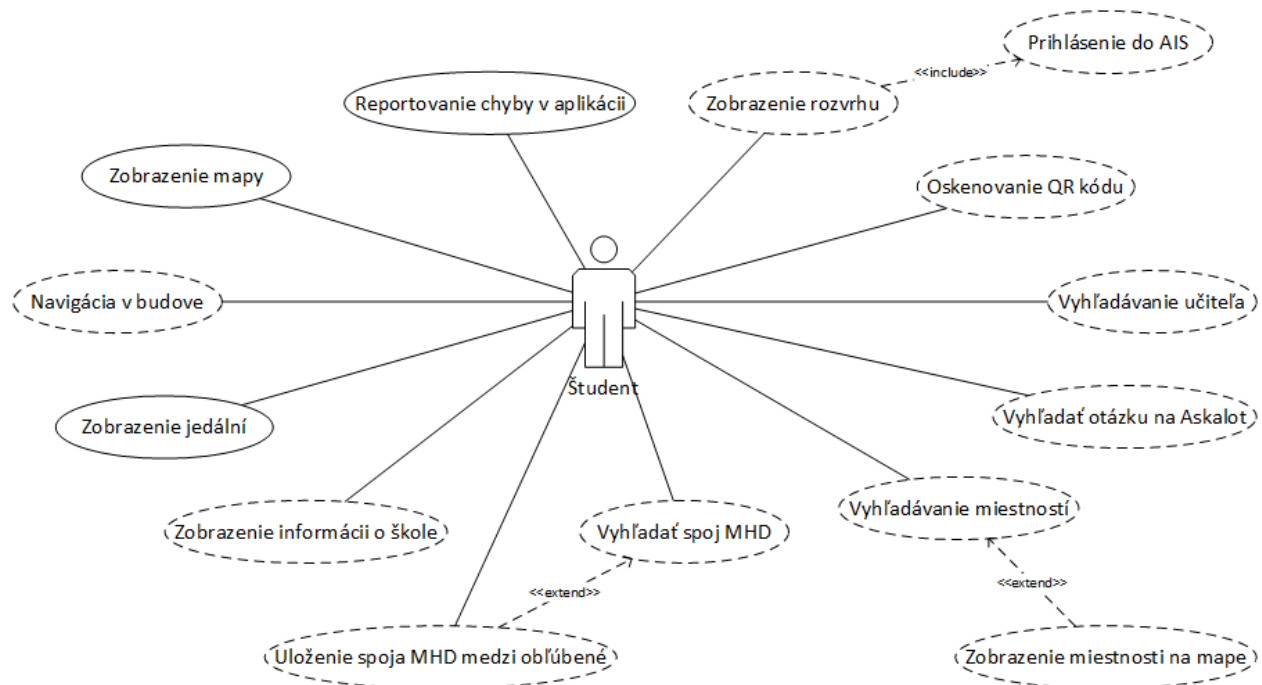
Po vizuálnej stránke sme sa rozhodli dať prednosť modernejšiemu dizajnu. Radi by sme využili v súčasnosti populárny tzv. plochý dizajn (*flat design*), svetlejšie farby a jednotné ikony. Bol vytvorený grafický návrh úvodnej obrazovky (Obr. 7 - C2) a v jeho duchu sa bude niest' aj zvyšok aplikácie. Finálnu podobu dizajnu ešte iterujeme.

4 Celkový pohľad



Obr. 5 Upravený diagram použitých technológií a ich prepojenie

Diagram prípadov použitia, ktoré sme identifikovali aj v novej aplikácii, sa nachádza na Obr. 6. Prípady použitia vyznačené prerušovanou čiarou zatiaľ neboli implementované.

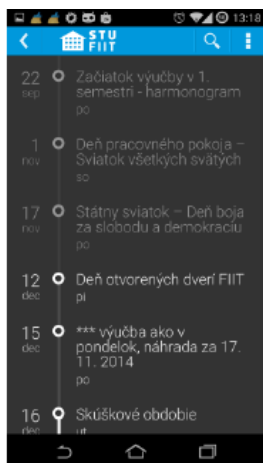


Obr. 6 Diagram prípadov použitia nového systému

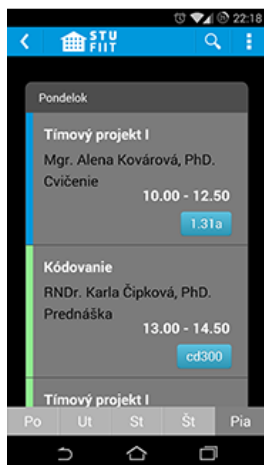
4 Celkový pohľad



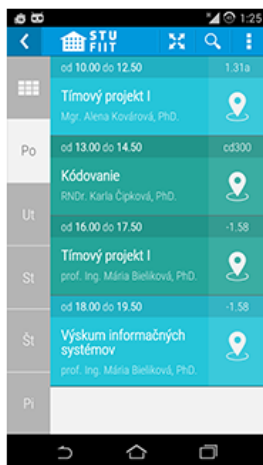
A1



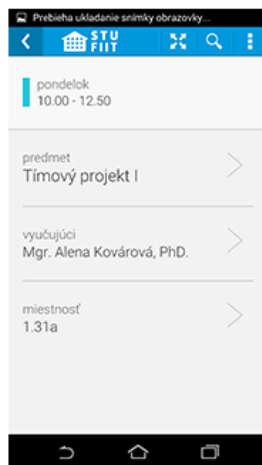
A2



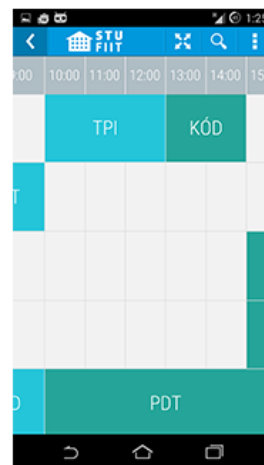
B1



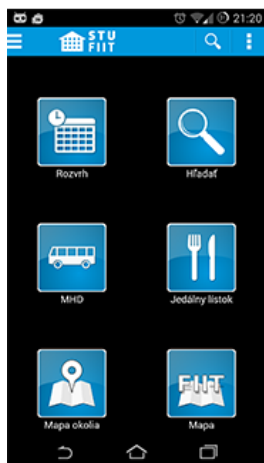
B2



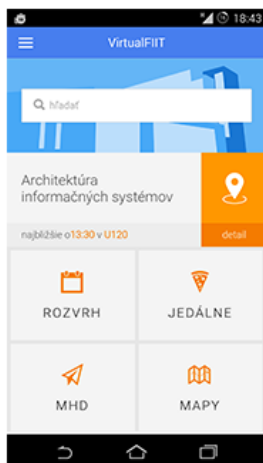
B3



B4



C1



C2

Obr. 7 Upravené obrazovky, A – harmonogram, B – rozvrh, C – domov (ľavá strana staré, pravá strana nové)

5 Moduly Virtual FIIT

5.1 AIS

Pôvodná verzia: Tento modul sprístupňuje používateľom informácie o miestnostiach, vyučujúcich, predmetoch, a rozvrhoch a zabezpečuje aj prihlasovanie do aplikácie. Nakoľko AIS neposkytuje žiadne API, tieto informácie sa získavajú parsovaním webovej verzie AIS. To sa vykonáva v pravidelných intervaloch alebo pri prihlasovaní.

Naše zmeny: Zistilo sa, že prihlasovanie do aplikácie zlyhá, ak má používateľ nastavenú v AIS inú ako základnú šablónu. Prihlasovanie sme teda presmerovali cez náš server, kde sa využíva menej striktný parser. Prihlasovacie údaje sa budú po novom posielat' cez protokol https.

Používateľský príbeh: *Študent sa chce v aplikácii prihlásiť, aby si mohol pozrieť svoj rozvrh.*

5.1.1 Pozrieť sa na šablóny AIS, prečo parser zlyháva

VFIIT-14, Veronika Olešová, Filip Šoltés

Analýza - Používatelia s istými šablónami majú problém prihlásiť sa do AIS cez mobilnú aplikáciu. Po zadaní prihlasovacích údajov a následnej snahe o prihlásenie sa, je obsah formulára vymazaný a používateľ sa nie je schopný dostať ku svojmu rozvrhu. Používateľ nedokáže zistiť z akého dôvodu sa toto deje. Dôvodom zlyhávania šablón je nesprávny formát ich rozšíreného HTML. Príkladom je nesprávne ukončenie HTML elementu alebo chýbajúce úvodzovky pri odkazovaní sa na obrázok. Domnievame sa, že *DOMParser*, ktorý je v programe použitý, potom nedokáže spracovať takéto HTML a používateľ sa nemôže prihlásiť.

Riešenie - Dočasne sme tento problém vyriešili tak, že pri každom prihlásení sa so zlou šablónou vypisujeme chybovú hlášku. V ďalšom šprinte si dávame za cieľ nahradiť *DOMParser* menej striktným parserom.

Testovanie - Otestované boli 3 šablóny s nesprávnym formátom, pri ktorých bola úspešne vypísaná chybová hláška. So správnou šablónou sa používateľ prihlási stále bez problémov. Test prebiehal na mobile HTC Desire 500 s Androidom verzie 4.1.2.

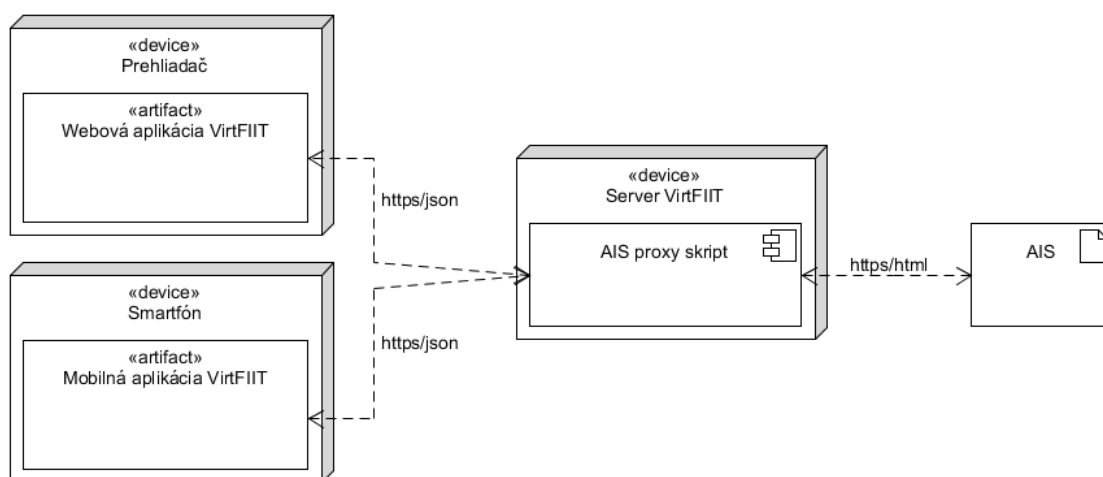
5.1.2 Kompletne prerobiť interakciu s AIS

VFIIT – 36, Veronika Olešová, Filip Mazán, Filip Šoltés

Analýza - Táto úloha nadväzuje na úlohu “VFIIT-14 - Pozrieť sa na šablóny AIS, prečo parser zlyháva?”, kde sme sa rozhodli pre nahradenie *DOMParsera* menej striktným parserom a zároveň na chybu “VFIIT-25 Oprava webovej verzie aplikácie kvôli prihlasovaniu do AIS”. Príčinou tejto chyby bola reštriktívna politika CORS (*Cross-origin resource sharing*), ktorá nám nedovoľovala z moderných prehliadačov pristupovať priamo k AIS. Bolo potrebné nájsť spôsob, ako obísť CORS ochranu a zároveň sa vysporiadať so šablónami, ktoré tvoria nevalidné HTML prvky - tieto prvky odmietal doterajší *DOMParser* spracovať.

Návrh - Navrhnuté bolo riešenie s použitím nášho servera ako proxy medzi aplikáciou a AIS.

Riešenie - Vyvinuli sme teda serverový skript, ktorému pošleme šifrovane cez protokol https prihlasovacie údaje. Tento skript cez knižnicu cURL komunikuje s AIS a sťahuje potrebné stránky. Tieto sú potom spracovávané štandardným parserom zabudovaným v jazyku PHP, ktorý dovoľuje aj nestriktné dokumenty. Na výstupe zo skriptu sú už spracované údaje z AIS, konkrétne ID študenta a jeho rozvrh. Klient stiahne tieto dáta vo formáte JSON a používa ich ako doteraz.



Obr. 8 Diagram znázorňujúci nový spôsob prihlasovania

Testovanie - Aplikáciu sme počas implementácie ladili pomocou webového rozhrania, kde sme ju potom aj testovali. Bolo testované správne zobrazovanie rozvrhov či už počas prebiehajúceho alebo skúškového obdobia. Naše riešenie bolo testované aj na mobilných telefónoch v ladiacom móde, kedy fungovalo všetko podľa očakávaní. Riešenie však nemožno nasadiť skôr, ako budeme mať na našom serveri validný a overený certifikát.

5.2 Jedálne

Pôvodná verzia: Tento modul sprístupňuje študentom aktuálnu ponuku jedálni v okolí školy. Jedálne lístky sa získavajú pomocou API zo serveru *hladnystudent.sk*. Získané dáta sa ukladajú do *localStorage*, aby boli dostupné aj bez internetu.

Naše zmeny: Opravili sme viacero chýb, ktoré znemožňovali používateľom korektne zobrazovať jedálne lístky alebo zobrazovali neaktuálne údaje. Jedálny lístok sa v niektoré dni na zariadeniach s menším displejom nedal posúvať. Počas stretnutia sa prišlo aj na chybu, kedy sa na rôznych zariadeniach zobrazoval rozdielny jedálny lístok.

Zobrazenie jedálnych lístkov je jedna z najobľúbenejších funkcií aplikácie a denne sa na ňu spolieha množstvo študentov a preto by mala fungovať korektne.

Používateľský príbeh: *Študent je hladný a chce si pozrieť jedálny lístok.*

5.2.1 Oprava chyby - Zobrazovanie obedov

VFIIIT-20, Filip Mazán, Jozef Karas

Analýza - Na niektorých mobiloch (napríklad Huawei Y300) sa vyskytla chyba v zobrazovaní obedov. Chyba bola v tom, že miesto toho, aby si používateľ mohol zrolovať (zhora nadol) vygenerovanú stránku a pozrieť všetky obedy dostupné na daný deň v danej jedálni, mohol len rolovať sprava doľava a tým videl len pár obedov.

Riešenie - Našli sme chybu, ktorá bola vo vykresľovaní obedov. Bolo ubratých cca 20px vo vykresľovaní. Kvôli chybnému rozkladaniu vznikali dlhé slová. Preto za každú čiarku a bodku treba doplniť medzeru.

Testovanie - Zmeny boli nasadené bez problémov pričom testovanie úspešne prebehlo na zariadení Huawei Y300 s verziou Androidu 4.1.1.

5.2.2 Obedy - oprava pamäte cache

VFIIT – 49, Jozef Karas

Analýza - Zistili sme, že niekedy obedy v aplikácii Virtuálna FIIT nesedia s obedmi na www.hladnystudent.sk, odkiaľ ťaháme informácie. Chyba bude najskôr v zlom mazaní *cache*.

Riešenie - Upravili sme zdrojový kód tak, že v súbore `hladnystudent.js` bol vo funkcií vymazania starých dát zmenený jeden riadok tak, aby pred načítaním dát - obedov z internetu vymazal všetky predchádzajúce informácie.

Testovanie - Oprava bola úspešne odskúšaná na všetkých našich Android zariadeniach v tíme.

Používateľský príbeh: Študent chce prehľadnejšiu a použiteľnejšiu aplikáciu, aby sa mu s ňou lepšie pracovalo.

5.2.3 Spraviť stránku s obedmi, zatiaľ po dizajnovej stránke netreba

VFIIT-73, Filip Šoltés

Analýza – V rámci prerábania celej aplikácie je potrebné prerobiť aj obrazovku, na ktorej je možné zobrazit' jedálny lístok bufetov a jedální v blízkosti školy. Je tu možné vybrať si konkrétny deň a konkrétnu jedáleň, pričom na obrazovke sa zobrazí zoznam s názvami jedál, ich cenami, alergénmi a pod.

Návrh – Keďže naším cieľom je čo najviac odľahčiť klientskú stranu aplikácie, rozhodli sme sa API volania presunúť na serverovú časť. V aplikácii bude vytvorený ovládač *CantineController*, ktorý bude vykonávať POST volania na serverovú časť a tak získavať dáta. Dáta budú ukladané do lokálneho úložiska, aby boli prístupné aj bez internetového pripojenia.

Riešenie – Obrazovka Jedálne bola pridaná podľa návrhu

Testovanie – Obrazovka bola otestovaná na všetkých zariadeniach, ktoré vlastníme. Bolo testované postupné prepínanie dní a jednotlivých jedální.

5.3 Rozvrh

Pôvodná verzia: Zobrazuje osobný rozvrh študenta. Ten sa získava parsovaním z AIS po prihlásení a ukladá sa do *cache*, aby sa dal neskôr prezerať aj bez internetu. Celý rozvrh bol reprezentovaný vo forme dlhého zoznamu.

Naše zmeny: Opravenie chyby, ktorá spôsobovala, že bolo možné vidieť cudzí rozvrh. Spravila sa nová verzia rozvrhu, kde pribudol pohľad na celý týždeň a upravil sa vzhľad aktuálnej verzie.

Používateľský príbeh: *Študent si požičia kamarátov mobil, aby si mohol pozrieť svoj rozvrh.*

5.3.1 Fix prihlásenia do AIS - treba mazať pamäť cache

VFIIIT-19, Michal Kučera

Analýza - Používateľ po prihlásení mohol vidieť rozvrh predchádzajúceho používateľa. Rozvrh sa po prihlásení ukladá do pamäte *localStorage*, odkiaľ sa pravdepodobne po odhlásení/prihlásení nevymazával.

Návrh - Vymazať rozvrh z pamäte *localStorage* vždy pri odhlásení používateľa.

Riešenie - Do metódy *logout()* v triede *ais*, bolo pridané mazanie rozvrhu z pamäte *localStorage* a nastavenie aktuálneho rozvrhu na *null*.

Testovanie – Mazanie starého rozvrhu z *localStorage* bolo testované na zariadení Xperia Z s Androidom 4.4.4 pomocou aplikácie *WeinRe*. Nasimulovanie reálnej situácie si vyžadovalo použitie dvoch AIS účtov a preto testovanie prebehlo na viacerých zariadeniach ako aj na viacerých účtoch a všetko fungovalo podľa očakávania.

Používateľský príbeh: *Študentovi chýba v rozvrhu prehľad celého týždňa v rámci jednej obrazovky, aby sa vedel lepšie zorientovať.*

5.3.2 Prerobenie rozvrhu

VFIIIT-24, VFIIIT-51, Daniel Pribul, Michal Kučera

Analýza – Pôvodný rozvrh sa študentom javil ako neprehľadný. Všetky položky rozvrhu boli zobrazené pod sebou, pričom na obrazovku sa nezmestili viac ako 2 naraz. Chýbal pohľad na celý týždeň.

Návrh – Navrhli sme pridať ďalší pohľad na rozvrh, ktorý by zobrazoval celý týždeň a bolo by možné si ho ľubovoľne približovať (Obr. 7 – B3). Pri takomto pohľade mali byť v rozvrhu použité iba skratky predmetov. Po kliknutí na skratku by sa mal zobrazíť detail danej položky rozvrhu (Obr. 7 – B4), ktorý by už obsahoval všetky potrebné údaje a odkiaľ by sa dalo prekliknúť na miestnosť, vyučujúceho alebo predmet, tak ako v pôvodnej verzii. Pôvodná obrazovka mala zostať zachovaná, pričom by na nej boli zobrazené iba položky rozvrhu aktuálne zvoleného dňa (Obr. 7 – B2). Menu s dňami zostalo zachované, ale zmenilo sa na vertikálne. Navrchu pribudlo tlačidlo na zmenu pohľadu.

Riešenie – Bol vypracovaný grafický návrh, ktorý bol prezentovaný zvyšným členom tímu. Implementácia potom pokračovala podľa návrhu, ale problém nastal s ľubovoľným približovaním, ktoré sa nám ani po dlhej dobe nepodarilo vyriešiť. Nakoniec sme to vyriešili iba 1-stupňovým približovaním cez tlačidlo v hornej lište. Bola upravovaná trieda *timetableView.ts* a pridané 2 nové šablóny *timetable-extended.xjade* a *detail.xjade*. Zmeny boli robené tak, aby sa prejavili aj v rozvrhoch profilov vyučujúcich, miestností a predmetov. Kvôli zlému časovému odhadu a neočakávaným nástrahám zo strany použitých technológií sa nám nepodarilo dokončiť úlohu za jeden šprint a musela byť dokončená v tom nasledujúcom.

Testovanie – Testovanie úspešne prebehlo na rôznych zariadeniach s operačným systémom Android od verzie 2.3.3 po 4.4.4.

5.4 MHD

Pôvodná verzia: Tento modul sprístupňuje študentom odchody spojov MHD zo zastávok z okolia školy. Cestovné poriadky sa získavajú pomocou API zo serveru *itransit.sk*. Získané dáta sa ukladajú do *localStorage*, aby boli dostupné aj bez internetu.

Naše zmeny: Na tomto module sme žiadne zmeny nevykonali.

5.5 Vyhľadávanie

Pôvodná verzia: Vyhľadávanie miestností, učiteľov a predmetov.

Naše zmeny: Na tomto module sme žiadne zmeny nevykonali.

5.6 Mapy

Pôvodná verzia: Aplikácia obsahuje mapu školy a mapu okolia. Mapy sú vo formáte *svg*, čo umožňuje interakciu používateľa s objektami na mape. Na viaceré objekty (miestnosti, jedálne, zastávky) je možné kliknúť alebo ich podľa potreby farebne vyznačiť (miestnosti). Mapy je možné približovať.

Naše zmeny: V pôvodnej verzii sme žiadne zmeny nevykonali. V novej verzii bol modul vytvorený nanovo.

Používateľský príbeh: *Študent chce prehľadnejšiu a použiteľnejšiu aplikáciu, aby sa mu s ňou lepšie pracovalo.*

5.6.1 Spraviť stránku s mapu budovy, zatiaľ SVG formát + tlačidlá poschodí

VFIIIT-74, Veronika Olešová

Analýza - Pôvodnú funkcionálnosť máp chceme postupne sprístupniť aj v novej aplikácii. Zo začiatku bude stačiť, ak sa po kliknutí na tlačidlo poschodia zobrazí prislúchajúca mapa v SVG formáte. Nie je potrebné implementovať približovanie mapy ani kliknutie na miestnosť.

Riešenie - Tlačidlá sú zatiaľ prevzaté z CSS komponentov rámca *Ionic*:

```
<div class="button-bar-inline">
```

Ich dizajn bude treba navrhnuť tak, aby boli všetky prehľadne rozmiestnené.

Na načítanie SVG obrázka sme nakoniec použili *ng-include*, ktorý vloží všetky informácie z tohto súboru do html:

```
<ng-include src="ctrl.getFloorSource()"></ng-include>
```

Skúšali sme aj iné načítavanie obrázkov akým je napríklad použitie elementu *<object>*, pri ktorom sa vyskytli problémy na telefónoch (na webovej stránke to nevykazovalo

žiadne problémy). Problém bol úspešne odstránený avšak použitie *ng-include* sa nám zdá ako najspoľahlivejšie riešenie.

Testovanie - Zobrazovanie máp sa správa na všetkých našich mobiloch podľa očakávaní.

5.7 QR Kódy

Pôvodná verzia: Aplikácia obsahuje čítačku QR kódov, ktorá po nasnímaní kódu zobrazí profil vyučujúceho. QR kódy obsahujú vizitku vo formáte *vCard* a mali by byť rozmiestnené po celej škole.

Naše zmeny: Na tomto module sme žiadne zmeny nevykonali.

5.8 BLE lokalizácia

Nový modul: Tento modul sa venuje navigácii používateľa v rámci budovy školy. Pozícia sa bude určovať na základe sily *bluetooth* signálu vysielaného z BLE zariadení rozmiestnených po škole. Aktuálny stav umožňuje určiť polohu v rámci miestností.

Používateľský príbeh: *Študent chce v aplikácii na mape budovy vidieť, kde sa práve nachádza, aby sa vedel rýchlejšie zorientovať.*

Tento používateľský príbeh sme sa rozhodli riešiť kvôli tomu, že patrí medzi naše hlavné ciele a chceli by sme našim spolužiakom pomôcť lepšie sa orientovať v budove našej školy.

5.8.1 Vytvorenie zásuvného modulu na detekciu a zobrazenie BLE zariadení

VFIIIT-4, Filip Mazán, Veronika Olešová

Analýza – Keďže navigácia vnútri budovy pomocou GPS nefunguje, zvolili sme technológiu BLE Beacon. T.j. potrebujeme zistiť, ako cez *PhoneGap* dokážeme pracovať s BLE vysielачmi. *PhoneGap* neobsahuje žiadne použiteľné zásuvné moduly, ktoré by túto funkciu podporovali, treba spraviť vlastný.

Návrh - Vytvoriť natívny *Cordova* zásuvný modul na platformu Android, ktorý zabezpečuje skenovanie BLE zariadení. Zásuvný modul treba prepojiť s front-endom aplikácie Virtuálna FIIT, a teda zobrazovať údaje získané o zásuvného modulu.

Riešenie - Pridaný bol *BeaconPlugin*, ktorý vykonáva navrhovanú funkcionality. Do hlavného menu bola pridaná položka „Beacony“, ktorá na obrazovke zobrazí skenovaciu obrazovku, na ktorej sa zobrazujú nájdené BLE zariadenia.

Testovanie - Zmeny boli nasadené a otestované na dvoch zariadeniach, ktoré podporujú technológiu BLE. Testovanie nevykázalo žiadne chyby.

5.8.2 Vybudovať aplikáciu na zber, zozbierať dáta

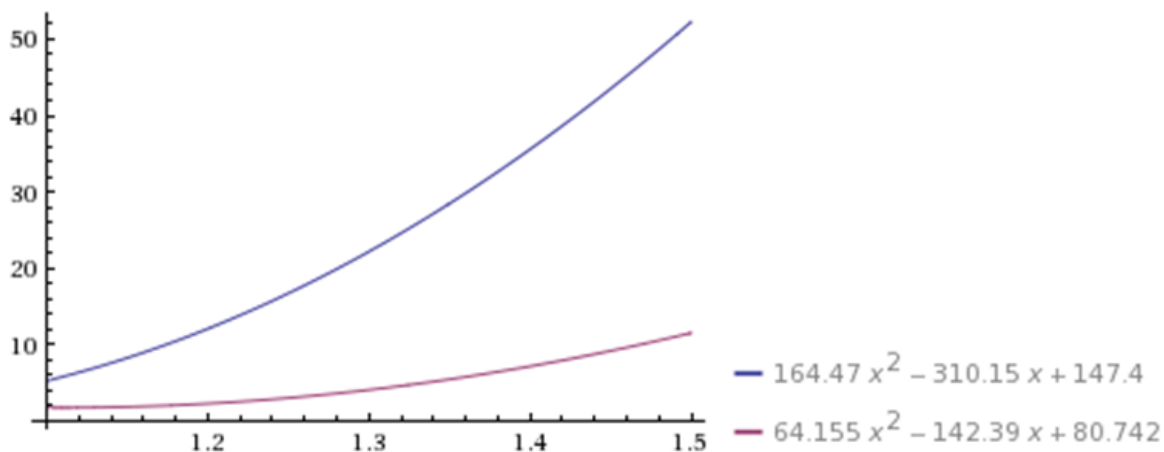
VFIT-72, Filip Mazán, Veronika Olešová

Analýza - Aby sme zistili, ako sa signál Bluetooth šíri v priestore, je nutné vykonať testovanie. Zo zistených dát budeme vedieť, ako signál so vzdialenosťou od vysielačov klesá a budeme schopní usúdiť, či má zmysel pokračovať v ceste trilaterácie alebo sa začať zaoberať metódami postavenými na *fingerprintingu* - digitálnych odtlačkoch priestoru, ako napr. neurónové siete, Bayesove filtre či časticové filtre.

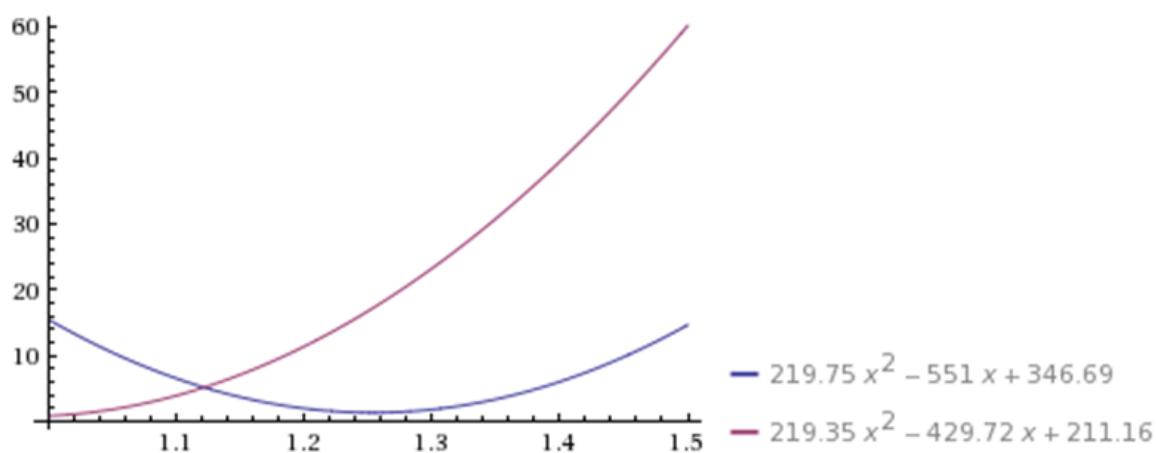
Návrh - Ku zberu dát potrebujeme navrhnuť aplikáciu na OS Android, ktorá je schopná zachytávať silu signálu Bluetooth vysielačného z viacerých vysielačov. Aplikácia musí vedieť zozbierané dáta poskytnúť v podobe určenej na ďalšie spracovanie (export do csv súboru). Aplikácia bude potom využitá na zbery v dvoch priestoroch - chodba a otvorený priestor s použitím dvoch typov vysielačov - na batérie a USB vysielač. Dáta budú vyhodnotené a bude nájdená čo najlepšia funkcia opisujúca pokles sily signálu so vzdialenosťou.

Riešenie - Vytvorili sme aplikáciu na zber dát, dáta sme zozbierali podľa návrhu a vyhodnotili.

5 Moduly Virtual FIIT



Obr. 9 Meranie na chodbe, modrá - USB, červená - batéria. Os X - podiel sily signálu a sily vysieláča, os Y - vzdialenosť



Obr. 10 Meranie v otvorenom priestore, modrá - USB, červená - batéria. Os X - podiel sily signálu a sily vysieláča, os Y - vzdialenosť

Z grafov vidno, že spôsoby lokalizácie založené na odhade vzdialenosti od vysieláča na základe sily signálu je veľmi nepresné a závisí nielen od typu vysieláča, ale aj od priestoru, v ktorom je umiestnený. Preto má význam sa zaoberať metódami postavenými na inom princípe.

5.9 Knižnica

Táto obrazovka sa v aplikácii pred tým nenachádzala. Obsahuje úradné hodiny knižnice a odkaz na mapu.

Používateľský príbeh: Študent chce ísť do knižnice, ale nechce tam ísť naprázdno. Hodila by sa mu informácia o úradných hodinách knižnice.

5.9.1 Úradné hodiny knižnice

VFIIIT-10, Michal Kučera, Jozef Karas

Analýza - V aplikácii sa v pôvodnom stave nedali nájsť úradné hodiny knižnice. Úradné hodiny študijného oddelenia sa už ale v aplikácii nachádzali. Odkaz v menu smeroval na miestnosť, pričom úradné hodiny boli vypísane v jej popise.

Návrh - Pridať informácie o knižnici po vzore študijného oddelenia.

Riešenie - Do bočného menu bola pridaná položka knižnica, ktorá po vzore študijného oddelenia odkazovala na stránku miestnosti (v tomto prípade miestnosti -1.43). Do súboru *rooms.json* sme k príslušnej miestnosti do atribútu *desc* pridali úradné hodiny knižnice. Aby sa pri stlačení tlačidla "zobrazíť miestnosť" knižnica korektne vyznačila na mape, bolo potrebné do súboru *map/-1.svg* pridať atribút *path* pre túto miestnosť. Nakoniec sa vytvorila nová ikona pre študijné oddelenie, nakoľko stará bola vhodnejšia práve pre knižnicu.

Testovanie - Zmeny boli nasadené bez problémov pričom testovanie úspešne prebehlo na zariadeniach Xperia Z s verziou Androidu 4.4.4 a Huawei Y300 s verziou Androidu 4.1.1.

5.10 Harmonogram

Pôvodná verzia: Táto obrazovka obsahuje zoznam dôležitých dátumov na fakulte. Odkaz na túto obrazovku sa nachádza v bočnom menu.

Naše zmeny: Pridanie časovej osi a zmena vzhľadu.

Používateľský príbeh: Študent chce prehľadnejší harmonogram, ktorý sa mu nebude zlievať, aby v ňom vedel rýchlejšie nájsť to, čo potrebuje.

5.10.1 Zmeniť neprehľadný harmonogram

VFIIIT-31, Jozef Karas, Michal Kučera

Analýza - Po spýtaní sa okoloidúcich spolužiakov sme zistili, že aktuálne zobrazenie harmonogramu je neprehľadné, pretože dátumy sa opticky zlievajú s udalosťami, čo výrazne zhoršuje čitateľnosť. Inšpirovali sme sa už existujúcimi riešeniami na internete a rozhodli sa dátumy oddeliť od udalostí časovou osou.

Návrh – Vytvoriť prvotný grafický návrh, v rámci tímu ho prekonzultovať, aplikovať zmeny a týmto štýlom dopracovať ku konečnej podobe. Tá sa potom implementuje do aplikácie.

Riešenie - Do nového zobrazenia harmonogramu sme pridali časovú os modrej farby s bielymi bodkami s dátumami a akciami (*Obr. 7 – A2*). Pokiaľ bol dátum časové rozhranie (od-do), časová čiara medzi dvoma bodmi bola tiež biela.

Testovanie - Nová verzia harmonogramu bola úspešne otestovaná na všetkých našich Android zariadeniach v tíme.

5.11 O aplikácii

Pôvodná verzia: Táto obrazovka obsahuje informácie o aplikácii, tímoch ktoré na nej pracovali a aktuálnej verzii.

Naše zmeny: V pôvodnej verzii sme na tejto obrazovke okrem aktualizácie údajov nič nezmenili. V novej verzii aplikácie bola táto obrazovka vytvorená nanovo.

Používateľský príbeh: *Študent chce prehľadnejšiu a použiteľnejšiu aplikáciu, aby sa mu s ňou lepšie pracovalo.*

5.11.1 Spraviť stránku O aplikácii s funkčnými linkami

VFIIIT-70, Filip Mazán

Analýza - V rámci prerábania celej aplikácie je nutné vytvoriť obrazovky, ktoré sa nachádzajú v pôvodnej aplikácii. Jednou z nich je obrazovka O aplikácii. Táto obsahuje textové informácie, odkazy na predošlé tímy a linku na formulár pre odoslanie chyby.

Návrh - Vytvorený bude ovládač *AboutController*, v ktorom budú prehľadne zapísané predošlé tímy spolu s linkami na ich domovskú webovú stránku. Súbor zobrazenia vypíše

statické textové informácie, predošlé tímy a linku na formulár. Linka na obrazovku O aplikácii bude pridaná do hlavného menu aplikácie.

Riešenie - Obrazovka *O aplikácii* bola pridaná podľa návrhu.

Testovanie - Obrazovka bola otestovaná na všetkých zariadeniach, ktoré vlastníme.

5.12 Nahlasovanie chýb

Pôvodná verzia: Táto obrazovka slúži na odosielanie informácií o chybe zo strany používateľov.

Naše zmeny: V starej verzii aplikácie sme na tejto obrazovke nič nemenili. V novej verzii bola spravená nanovo.

Používateľský príbeh: *Študent chce prehľadnejšiu a použiteľnejšiu aplikáciu, aby sa mu s ňou lepšie pracovalo.*

5.12.1 Spraviť nahlasovanie chýb s funkčným formulárom a hláškami používateľovi *VFII-69, Daniel Pribul*

Analýza - Potreba získavať dáta od používateľov o chybách v aplikácii zostáva aktuálna aj v novej verzii. Stránka obsahuje pole na vyplnenie textu a tlačidlo na odoslanie správy o chybe so spätnou väzbou používateľovi.

Návrh - Správa o chybe sa bude odosielať pomocou metódy POST http protokolu. Po odoslaní sa zobrazí informácia o úspešnom, resp. neúspešnom odoslaní správy o chybe. V prípade úspechu sa správa z formulára vymaže.

Riešenie – Nahlasovanie chýb bolo implementované podľa návrhu.

Testovanie – Nahlasovanie chýb bolo otestované na všetkých zariadeniach, ktoré vlastníme.

5.13 Domovská obrazovka

Pôvodná verzia: Obrazovka obsahuje 6 tlačidiel na *Rozvrh, Hľadanie, Jedálne, Mapu FIIT, Mapu okolia a MHD.*

Naše zmeny: V pôvodnej verzii aplikácie sme túto obrazovku nemenili. V novej verzii prešla niekoľkými zmenami. Počet tlačidiel sa zúžil na 4 a pribudla informácia o najbližšej vyučovacej hodine. Odkaz na vyhľadávanie bol nahradený samotným vyhľadávacím poľom a bola pridaná.

Používateľský príbeh: *Študent chce prehľadnejšiu a použiteľnejšiu aplikáciu, aby sa mu s ňou lepšie pracovalo.*

5.13.1 Spraviť úvodnú stránku (homescreen) s ikonkami

VFIIT-71, Michal Kučera, Daniel Pribul

Analýza – Úvodná obrazovka v pôvodnej verzii aplikácie podľa nás nevyužívala naplno svoj potenciál. Pôvodných 6 tlačidiel sme sa rozhodli zúžiť na 4. Rozhodli sme sa vynechať tlačidlo vyhľadávania a nahradiť ho samotným vyhľadávaním. Zvyšok vzniknutého miesta sme sa rozhodli zaplniť informáciou o najbližšej hodine a odkazom na miestnosť na mape, kde táto hodina bude prebiehať. Dostupný bude aj odkaz na podrobnejší detail tejto hodiny.

Návrh – Vypracovanie grafického návrhu, zapracovanie pripomienok od členov tímu a implementovanie schváleného návrhu do aplikácie. Zatiaľ iba staticky.

Riešenie - Bol vypracovaný prvotný grafický návrh, ktorý bol predvedený zvyšku tímu. Následne boli zapracované pripomienky a postupne sa týmto spôsobom dopracovalo k výslednej podobe úvodnej stránky (*Obr. 7 – C2*). Tá sa potom implementovala. Pri testovaní sa vyskytol problém pri vysunutí klávesnice, kedy sa celá obrazovka zúžila na cca polovicu svojej pôvodnej veľkosti. Tento stav sa podarilo vyriešiť nastavením výšky obrazovky v súbore *home.js* vždy pri načítaní obrazovky.

Testovanie – Testovanie úspešne prebehlo na všetkých zariadeniach v našom tíme s verziou operačného systému Android v rozsahu 2.3.5 až 4.4.4.

5.14 Ostatné

Úlohy, ktoré nie je možné jednoznačne priradiť k žiadnemu modulu.

Používateľský príbeh: *Študent chce prehľadnejšiu a použiteľnejšiu aplikáciu, aby sa mu s ňou lepšie pracovalo.*

5.14.1 Vytvoriť localStorage factory

VFIIIT-75, Filip Mazán

Analýza - Tak ako v pôvodnej aplikácii, je potrebné perzistentne uchovávať dáta v zariadení. Na toto slúži tzv. *localStorage*, ktorú poskytujú ako prehliadače, tak aj aplikácie v mobilných zariadeniach. Preto je nutné túto úložnú kapacitu sprístupniť do našej aplikácie.

Návrh - Úložisko sprístupníme pomocou štruktúry *Factory* v AngularJS umiestnenú v súbore *storage.js* a bude obsahovať aspoň metódy na uloženie, načítanie a vymazanie hodnoty.

Riešenie - Úložisko bolo implementované podľa návrhu. Sprístupňuje metódy:

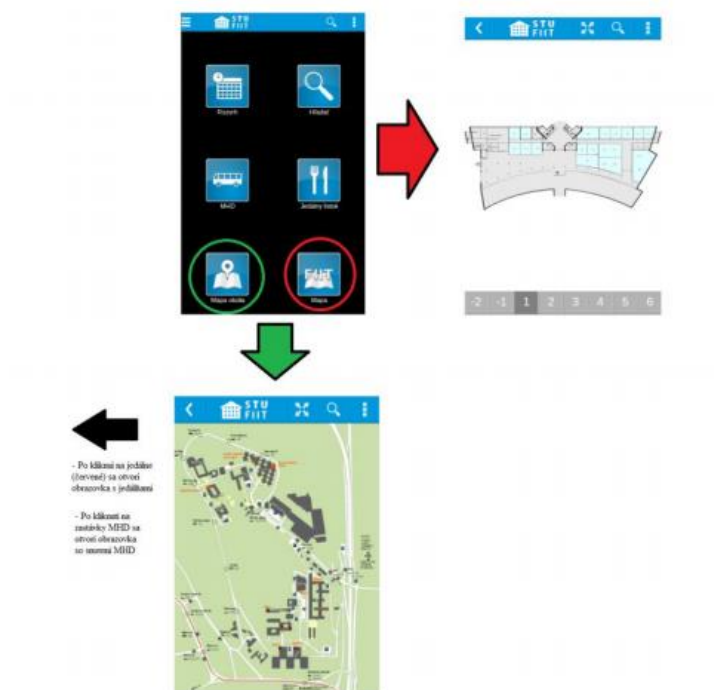
Tab. 1 Implementované metódy

metóda	popis
<code>\$storage.get(\$key)</code>	načíta a vráti hodnotu s kľúčom <code>\$key</code>
<code>\$storage.set(\$key, \$value)</code>	uloží hodnotu <code>\$value</code> s kľúčom <code>\$key</code>
<code>\$storage.contains(\$key)</code>	vráti hodnotu pravda/nepravda podľa toho, či sa v úložisku nachádza hodnota s kľúčom <code>\$key</code>
<code>\$storage.delete(\$key)</code>	vymaže hodnotu s kľúčom <code>\$key</code>
<code>\$storage.isAvailable()</code>	vráti hodnotu pravda/nepravda podľa toho, či je úložisko na zariadení podporované
<code>\$storage.getLength()</code>	vráti počet hodnôt uložených v úložisku

Testovanie - Úložisko bolo otestované na všetkých zariadeniach, ktoré vlastníme.

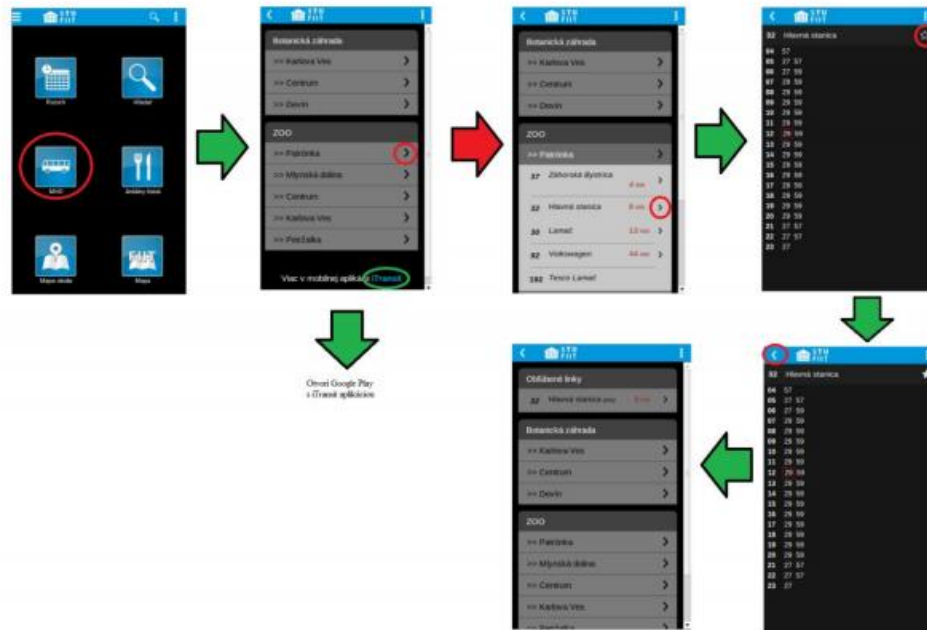
Príloha A Diagramy obrazoviek

Táto príloha obsahuje diagramy obrazoviek pôvodnej verzie aplikácie, ktoré sa kvôli svojmu rozsahu neuviedli v kapitole *Celkový pohľad*. Vyhotovil ich predchádzajúci tím a obrazovky na nasledujúcich diagramoch zostali aj naďalej nezmenené.



Obr. 11 Diagram obrazoviek máp (zdroj: minuloročný tím)

Príloha A Diagramy obrazoviek



Otvori Google Play s IT aplikáciou

Obr. 12 Diagram obrazoviek pre cestovné poriadky (zdroj: minuloročný tím)



Otvori Google Play a v danej aplikácii Hľady študent

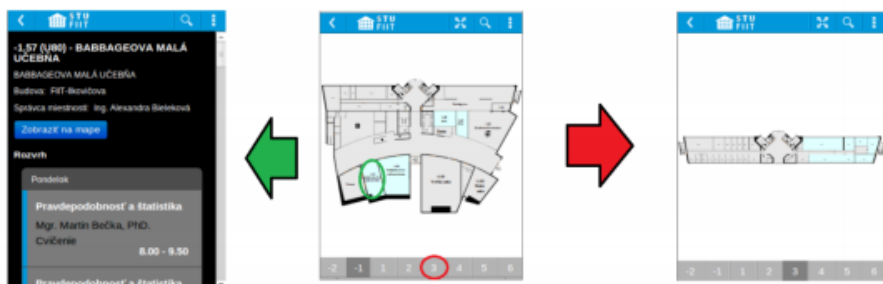
Prehliadni menu dňa

Otvori Google maps v prehliadači

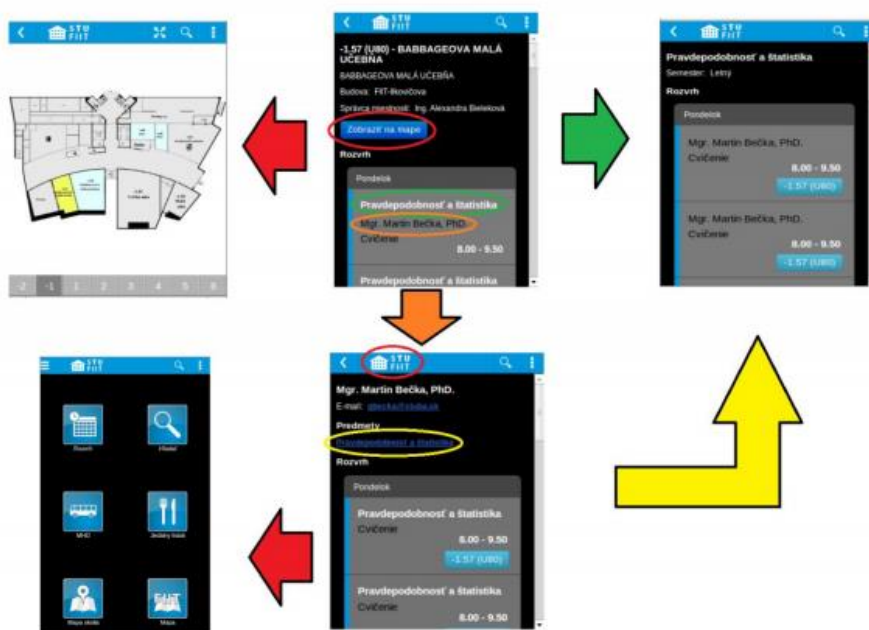
Otvori domovskú stránku jedálne

Obr. 13 Diagram obrazoviek jedálnych lístkov (zdroj: minuloročný tím)

Príloha A Diagramy obrazoviek

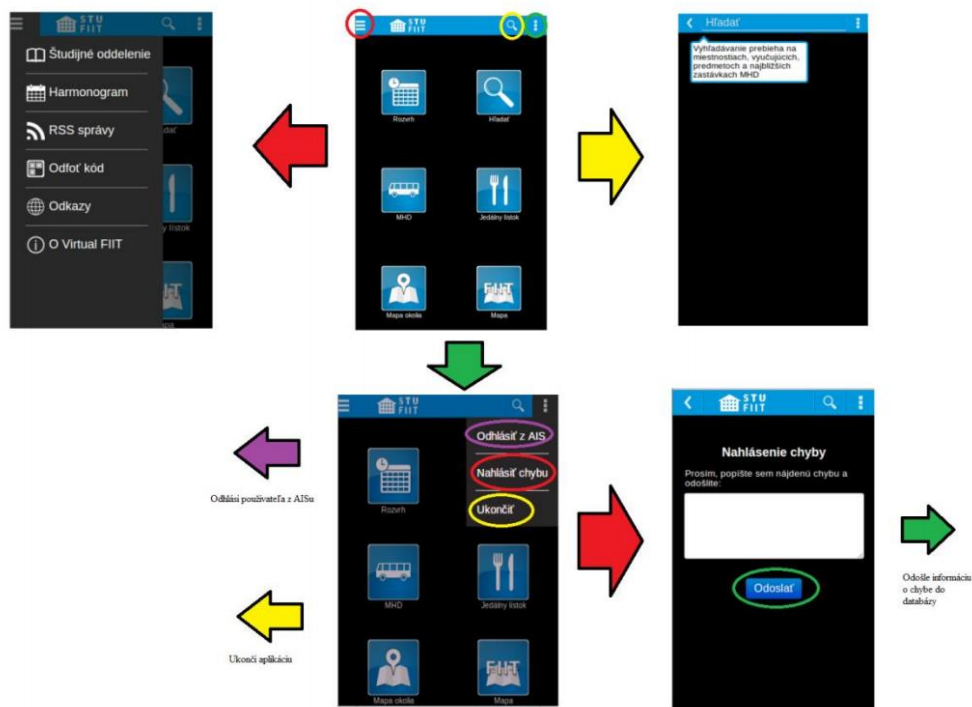


Obr. 14 Diagram obrazoviek pre miestnosti (zdroj: minuloročný tím)

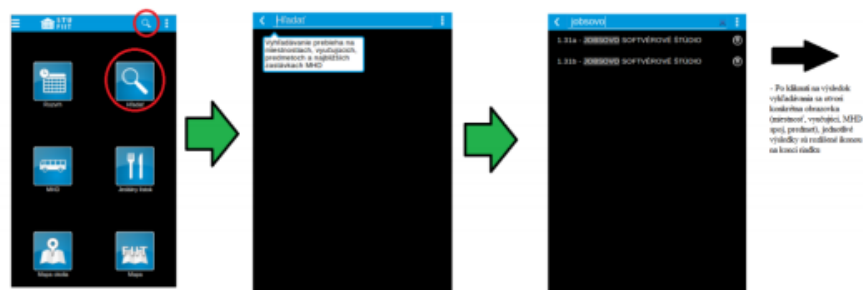


Obr. 15 Diagram obrazoviek vyučujúcich (zdroj: minuloročný tím)

Príloha A Diagramy obrazoviek



Obr. 16 Diagram obrazoviek hlavného menu (zdroj: minuloročný tím)



Obr. 17 Diagram obrazoviek pre vyhľadávanie (zdroj: minuloročný tím)