

ITU - technická správa

FIT schedule maker +

Matúš Moravčík

xmorav48

V aplikácii som mal implementovať bočný panel pre modifikovanie rozvrhu, jednoduchý generátor rozvrhu, vytvoril widget pre rozvrh a backend pre zobrazovanie informácií o určitej hodine v rozvrhu.

Súbory:

- sideBar.dart
- generator.dart
- app.dart
- timetable_container.dart
- homepage.dart

homepage.dart

Obsahuje základný layout aplikácie. Reprezentuje jedinú stránku našej aplikácie. Obsah aplikácie je možné meniť pomocou tabov. Okrem zastrešovania všetkých obsahov jednotlivých tabov jeho úlohou je ihneď pri načítaní aplikácie asynchrónne nahráť niektoré s nevyhnutných dát z <https://www.fit.vut.cz/> na fungovanie aplikácie. Okrem tohto sa stará o rozloženie layoutu aplikácie pri zmene rozlíšenia ako aj pri zmene obsahu.

generator.dart

Definuje widget 'Generator', ktorý je zodpovedný za poskytovanie používateľského rozhrania na dynamické generovanie rozvrhov. Widget obsahuje možnosti na nastavenie obmedzení, ako napr. maximálny počet hodín a cvičení za deň, výber voľných dní... Používateľ môže zadať obmedzenia pomocou widgetov "NumberPicker" a „DaySelector“. Keď filtre sú nastavené, kliknutím na tlačidlo "Vygenerovať" sa spustí proces generovania rozvrhu cez metódu „generateTimetable“ v „AppViewModel“ a následne sa dynamicky prekreslí každý widget, ktorý sledoval zmeny vo vybraných predmetoch daného rozvrhu.

app.dart

V „AppViewModel“ triede som vytvoril funkcie, ktoré slúžia ako notifiers. Ich úlohou je na základe užívateľského vstupu uložiť hodnotu a následne notifikovať všetky widgety, ktoré naslúchajú na zmenu atribútu, kde bola hodnota uložená, aby sa prekreslili. Atribúty, ktoré užívateľ môže meniť zahŕňajú štúdium, semester a ročník štúdia. Ďalej som implementoval funkciu, ktorá vygeneruje

rozvrh z poskytnutých filtrov. Aktuálne funkcia rešpektuje iba výber kedy užívateľ chce mať voľný deň a tiež nerieši konflikty v predmetoch. Po vygenerovaní rozvrhu notifikuje o tom, že nastala zmena v dátovom modeli. Zmena sa prejaví prekreslením každého widgetu, ktorý tento dátový model sledoval.

sideBar.dart

Súbor obsahuje celú implementáciu bočného panelu. Bočný panel slúži na umožnenie užívateľovi meniť dáta v rozvrhu. Zmeny dát, ktoré sú užívateľovi umožnené sú: zmena štúdia, semestru, ročníku štúdia, pridanie alebo odobranie predmetu z rozvrhu. Bočný panel je rozdelený do viacerých komponent. Jednotlivé komponenty naslúchajú zmenám o ktorých ich upozorňuje AppViewModel. Komponenty sú navrhnuté tak aby reagovali iba na pre nich relevantné zmeny hlásené AppViewModelom. Napríklad každé tlačidlo s určitým predmetom sa prekreslí ak sa bude nachádzať medzi zvolenými predmetmi v rozvrhu. Teda ak užívateľ odstráni predmet pomocou iného widgetu ako je ktorýkoľvek v SideBar widget, stačí, že pri tom bol vytvorená notifikácia AppViewModelom a prekreslí sa widget predmetu, ktorého sa zmeny dotkli. Naša aplikácia nenačítava všetky predmety ihneď po jej spustení, keďže by to spustenie aplikácie výrazne spomalilo. Teda pri spustení aplikácie sa načítajú informácie o predmetoch pre zvolené štúdium. Až po zvolení si iného štúdia sa načítajú všetky jeho predmety. Ak sú predmety daného štúdia načítané v aplikácii každý ďalšíkrát sa k nim pristupuje synchrónne načítaním z pamäti. To isté platí aj pre načítanie hodín do predmetov. Až po zvolení určitého predmetu sa overí, či sú jeho hodiny už načítané z <https://www.fit.vut.cz/>. Opäť iba po prvýkrát sa ku nim pristupuje asynchrónne.

timetable_container.dart

Vytvoril som responzívny widget pre rozvrh, ktorý zobrazuje predmety na absolútne pozicionovaných miestach. Rozvrh sa vždy celý prekreslí ak je z rozvrhu odstránený alebo do neho pridaný nejaký predmet, tiež reaguje na zmenu semestru. Jeho úlohou pred zobrazením vyučovacích hodín je overiť ich existenciu v aplikácii. Ak nie sú vyučovacie hodiny načítané rozvrh si ich všetky sám načíta a to asynchrónne pomocou funkcie 'getAllCourseLessonsAsync' z AppViewModelu. Táto funkcia načíta všetky predmety na základe ich identifikačného čísla z <https://www.fit.vut.cz/>. Výhodou tejto funkcie je jej načítavanie všetkých vyučovacích hodín paralelne pre každý predmet. Vždy až po načítaní vyučovacích hodín pre posledný predmet je rozvrh aktualizovaný.

Vytvoril som overlay, ktorý sa zobrazuje po ukázaní myšou na nejaký z predmetov pomocou asynchrónneho časovača, ktorý sa po každom pohybe kurzorom reinitializuje. Ak je kurzor na danej hodine po určitu dobu, zobrazí sa overlay, ktorý zmizne po posune kurzorom. Ak sa kurzor do nejakej doby presunie na overlay, tak overlay nezmizne. Zmizne až keď kurzor sa dostane mimo jeho widget.