

# **Semestrálna Práca**

## **VAMZ**

Meno: Richard Závodský  
Rok: 2022/2023

## Opis Aplikácie

Aplikácia slúži na krátkodobé plánovanie práce pre dnešný deň. Z už existujúceho systému na zapisovanie úloh zistí, aké úlohy je treba vykonať, a umožní používateľovi naplánovať prácu na týchto úlohách.

## Zdroj úloh

Všetky úlohy a poznámky si na počítači cez program Emacs zapisujem do textového súboru s formátom Org mode (<https://orgmode.org>). Tento súbor je externe synchronizovaný s mobilom (napr. pomocou aplikácie [Syncthing](#) alebo [Dropbox](#)), kde ho aplikácia prečíta.

V Org formáte má každá úloha aj rôzne dáta, ktoré by aplikácia využívala, napr. deadline úlohy, priorita, očakávaná dĺžka práce, popis, atď.

## Plánovacie bloky

Plánovanie je realizované v blokoch, ktoré obsahujú plánovaný čas začiatku a konca, a úlohu, ktorá sa má počas daného bloku vykonávať.

Existujú 2 typy blokov:

- Plán úlohy – Je spojený s nejakou úlohou uloženou v org súbore
- Jednorázový plán – Nie je spojený so žiadnou dlhodobou úlohou, je vytvorený iba na jeden deň a používa sa pre plánovanie jednorázových úloh/udalostí (napr. nákup)

## História

Aplikácia ukladá históriu plánov, a umožní používateľovi zobraziť plán z hociakého predchádzajúceho dňa.

## Miniaplikácia

Aplikácia obsahuje miniaplikáciu, ktorá zobrazuje aktuálny, prípadne nasledujúci blok.

## Notifikácie

Aplikácia pošle používateľovi notifikáciu o začiatku plánovaného bloku. Používateľ si môže nastaviť, aby aplikácia posielala notifikácie o niekoľko minút skôr.

## Podobné aplikácie

- Google Calendar – Umožňuje plánovať úlohy a udalosti, ale je určená skôr na dlhodobé plánovanie, napr. niekoľko dní dopredu. Vytváranie plánov trvá veľmi dlho, keďže používateľ musí písať názov, čas, atď. pre každý blok samostatne. Úprava plánov (napr. presúvanie blokov) je tiež pomalá. Nedokáže čítať úlohy z org súborov.
- Orgzly – Dokáže čítať a upravovať org súbory, ale nedokáže vytvárať plány. Nepodporuje niektoré potrebné funkcie org súborov.

Táto aplikácia je veľmi špecifická pre moje požiadavky a musí pracovať s mojim vlastným systémom zapisovania úloh, preto aplikácia, ktorá by takéto niečo robila, pravdepodobne neexistuje.

## Návrh aplikácie

Aplikácia obsahuje jednu aktivitu MainActivity, v ktorej sú pomocou knižnice Navigation zobrazené rôzne fragmenty.

### MainFragment

Hlavný fragment aplikácie, zobrazí sa pri spustení aplikácie. Obsahuje vlastný view EditableDayView, ktorý zobrazuje všetky plánovacie bloky na dnešný deň. Podržaním na hornej časti bloku sa môže blok presunúť, podržaním na spodnej časti sa môže zmeniť dĺžka bloku. Kliknutím na blok sa otvorí PlanDetailFragment pre daný blok. Fragment má aj FloatingActionButton, pomocou ktorého sa dá presunúť na PlanAddFragment.

Tento fragment používa ViewBinding.

### PlanAddFragment

Umožňuje používateľovi pridávať plánovacie bloky na dnešný deň. Po otvorení sa používateľovi zobrazia všetky úlohy z org súboru, ktoré sú zoradené podľa priority. Zároveň používateľ môže vytvoriť jednorázové plány. Vybraté úlohy si uchovávajú poradie, v ktorom boli vybrané, a v tom poradí sa aj vložia do dňa.

Tento fragment používa ViewBinding, zoznam úloh je zobrazený pomocou RecyclerView. Jednotlivé úlohy sú zobrazené pomocou layoutu task\_item.xml, ktorý používa DataBinding. Na ukladanie stavu používa ViewModel.

### PlanDetailFragment

Zobrazuje detail daného plánovacieho bloku. Ak je to blok plánovania úlohy, zobrazí informácie o úlohe, ako napr. deadline, prioritu, popis. Ak je to jednorázový blok, zobrazí názov úlohy. Ak je plán z dnešného dňa, dovoľí ho používateľovi aj vymazať.

Tento fragment používa DataBinding.

## **PlanHistoryFragment**

Zobrazuje históriu plánov. Používateľ môže zvoliť dátum a fragment zobrazí pomocou vlastného view DayView plány zo zvoleného dňa. Tento plán už nie je editovateľný.

Tento fragment používa ViewBinding.

## **PreferencesFragment**

Je to fragment generovaný knižnicou Preferences, umožňuje používateľovi meniť nastavenia. Používateľ môže zvoliť cestu k Org súboru, počiatočnú dĺžku plánov a hodinu v ktorej má aplikácia začať pridávať plány.

## **OrgParser**

Dokáže prečítať org súbor a vybrať z neho informácie o úlohách, ich deadlinoch, prioritách, atď. Tieto úlohy sú uložené do inštancií triedy OrgTask.

## **Ukladanie dát**

Jednotlivé plány sú uložené v triede PlanBlock, ktorá je zároveň entitou Room databázy PlanBlockDatabase. Všetky operácie s databázou sú vykonané asynchrónne. Dotazy sú realizované pomocou LiveData, čiže fragmenty aplikácie správne odpovedajú na zmeny v databáze.

Trieda Tasks uchováva úlohy z org súboru, a zároveň pomocou FileObserver pozoruje zmeny v org súbore. Ak zachytí nejakú zmenu, súbor znovu prečíta a aktualizuje dáta. Dáta sú v triede Tasks uložené pomocou LiveData, takže iné triedy môžu reagovať na prípadné zmeny.

## **Notifikácie**

Pri hociakej úprave plánov v dnešnom dni aplikácia nastaví časovače pre dnešné bloky pomocou AlarmManager, ktoré v čase začatia bloku odošlú broadcast, ktorý zachytí NotificationBroadcastReceiver. Ten potom pošle notifikáciu o začatí bloku.

## **Widget**

Aplikácia má jeden widget, ktorý zobrazuje aktuálny alebo nasledujúci blok. Tento widget je vytvorený triedou SmallAppWidgetProvider. Po kliknutí na widget sa otvorí aplikácia. Widget je aktualizovaný vždy pri zatvorení aplikácie, alebo pri notifikáciách triedou NotificationBroadcastReceiver.

## Štýly

Aplikácia má definované vlastné štýly pre tmavý aj svetlý režim, prispôsobuje sa teda nastaveniam telefónu.

## Použité zdroje

V aplikácii je použitá jedna externá knižnica <https://github.com/h6ah4i/android-numberpickerprefcompat>, ktorá pridáva triedu `NumberPickerPreferenceCompat`, ktorá umožní vytvoriť nastavenie pre vybratie čísla.