



ŽILINSKÁ UNIVERZITA V ŽILINE

Fakulta riadenia
a informatiky

Semestrálna práca z predmetu
vývoj aplikácií pre mobilné zariadenia

APLIKÁCIE NA PRE ÚĽOVÉ VÁHY

Vypracoval: Michal Petrán

Študijná skupina: 5ZYI24

Akademický rok: 2024/2025

V Žiline dňa 8.6.2025



Skutočný návrh riešenia problému

Krátka analýza

Cieľom aplikácie je umožniť používateľovi (včelárovi) sledovať stav svojich úľov, vizualizovať dáta zo senzorov a zaznamenávať udalosti a poznámky k úľom.

Aplikácia slúži ako mobilný klient k serveru, ktorý ukladá dáta z jednotlivých senzorov (hmotnosť, teplota, vlhkosť).

Používateľské prípady:

- Zobrazíť aktuálny stav úľov (hmotnosť, teplota, vlhkosť)
- Prezeráť historické dáta v grafoch
- Pridať / odstrániť / editovať úľ v aplikácii
- Zaznamenávať udalosti do denníka
- Vyhľadávať v denníku
- Nastaviť režim zobrazenia (dark/light mode), notifikácie a ich parametre
- Získať upozornenie pri neštandardnom stave (pokles hmotnosti, vysoká teplota)

Návrh riešenia

Architektúra aplikácie je založená na modernom prístupe:

- **Frontend:** Android aplikácia v jazyku Kotlin, postavená na Compose UI frameworku
- **Backend:** REST API vytvorené v PHP, ktoré pristupuje k MySQL databáze a poskytuje potrebné údaje aplikácii
- **Databáza:** MySQL serverová databáza

Hlavné časti aplikácie:

1. **Zobrazenie dát (Prehľad):**
 - Hlavná obrazovka s poslednými hodnotami hmotnosti a zmien oproti predchádzajúcemu záznamu.
 - Možnosť manuálne obnoviť dáta (Refresh).
2. **Grafy:**
 - Vizualizácia historických dát (hmotnosť, teplota, vlhkosť).
 - Grafy umožňujú lepšie porozumieť trendom a stavu úľov.
3. **Denník:**
 - Záznamy o udalostiach, ktoré si včelár zapisuje manuálne (napríklad kontrola úľa, pridanie nadstavka, liečenie a pod.).
 - Vyhľadávanie v existujúcich záznamoch.



4. Správa úľov:

- Možnosť pridávať nové úle do databázy a upravovať existujúce.

5. Nastavenia:

- Prepínanie medzi dark/light režimom.
- Zapnutie/vypnutie notifikácií.
- Nastavenie parametrov notifikácií (napríklad prahové hodnoty pre upozornenie).

Sieťová komunikácia:

- Prebieha cez REST API na serveri.
- Na strane Android aplikácie sa využíva knižnica OkHttp na HTTP komunikáciu a knižnica Gson na parsovanie JSON odpovedí.

Práca s dátami:

- Aplikácia využíva Room databázu pre:
 - Lokálne ukladanie nastavení
 - Ukladanie zoznamu úľov
 - Ukladanie záznamov denníka
- ViewModely a LiveData zabezpečujú oddelenie logiky od UI a správne riadenie životného cyklu.

Pozadie a notifikácie:

- WorkManager je využitý na pravidelné získavanie nových dát v pozadí.
- V prípade, že sa zistia anomálie (napríklad pokles hmotnosti pod určitú hranicu), aplikácia zobrazí systémovú notifikáciu.

Zmeny oproti checkpointu

Zmena architektúry backendu a práce s dátami

Pôvodne: aplikácia komunikovala priamo s databázou PostgreSQL

Nové riešenie:

- Používam databázu **MySQL** na serveri
- Na komunikáciu s databázou je vytvorené vlastné **PHP REST API**
- Aplikácia teraz pristupuje k dátam výhradne cez REST API
- Všetky CRUD operácie (čítanie, zápis, úprava, mazanie) prebiehajú cez API

Zmeny vo vzhľade a UX/UI

Celkový vizuálny štýl aplikácie bol prepracovaný:

- Nové farby, štýly...

Zmeny v obrazovkách

A) Obrazovka *Grafy* — kompletne nanovo vytvorená

- Vyber úľa, vyber typu dát a obdobia

B) Obrazovka *Prehľad*

- Funkcionalita ostala rovnaká (zobrazenie posledných hodnôt a zmien)
- **Nové:**
 - **Tlačidlo *Refresh*** - umožňuje manuálne načítať aktuálne dáta zo servera
 - Vizuálne vylepšenia (čistejší dizajn, lepšie rozloženie údajov)

C) Obrazovka *Denník*

- Základná funkcionality (pridávať, upravovať, mazať poznámky) ostáva
- **Nové:**
 - Pridané **vyhľadávanie poznámok** (fulltextové vyhľadávanie podľa textu poznámky, typu úkonu alebo dátumu)

D) Nová obrazovka *Pridať/Editovať úle*

- Úplne nová obrazovka
- Možnosti:
 - **Pridať nový úľ** — do aplikácie pridá nový úľ
 - **Editovať existujúci úľ** — umožňuje upraviť názov úľa, hodnoty...
 - **Odstrániť úľ** — odstráni úľ z aplikácie

E) Nová obrazovka *Nastavenia*

- Úplne nová obrazovka v aplikácii
- Možnosti:
 - **Prepínač Dark/Light mode** — zmena vzhľadu aplikácie
 - **Zapnutie/vypnutie notifikácií** — globálne nastavenie pre prijímanie notifikácií
 - **Nastavenie notifikácií** — konfigurácia, pri ktorých udalostiach chce používateľ dostať notifikáciu (pokles hmotnosti, interval notifikácií...)



Popis Implementácie

Využitie AndroidX komponentov

- **Navigation** - Navigácia medzi hlavnými obrazovkami aplikácie (Prehľad, Grafy, Denník, Nastavenia, Správa úľov)
- **Room** – Lokálne ukladanie nastavení, úľov a záznamov z denníka
- **ViewModel** - Použitý v každej obrazovke
- **WorkManager** - Pre získavanie dát zo servera aj keď je aplikácia na pozadí alebo zatvorená + posielanie notifikácií pri splnení podmienok (napr. pokles hmotnosti, vysoká teplota)
- **Lifecycle** – onCreate()...

Použitie externého frameworku / knižnice

- **OkHttp** (com.squareup.okhttp3:okhttp) - Sieťová komunikácia s REST API na serveri
- **Gson** (com.google.code.gson) - Parsovanie JSON odpovedí z API
- **MPAndroidChart** (com.github.PhilJay:MPAndroidChart) - Vykresľovanie grafov na obrazovke *Grafy*
- **dotenv-kotlin** (io.github.cdimascio:dotenv-kotlin) - Na čítanie premenných prostredia (.env)

Použitie sieťovej komunikácie

Komunikácia zo serverom cez API na ktorom je uložená MySQL databáza z príslušnými tabuľkami

Zoznam zdrojov

<https://developer.android.com/kotlin/campaign/learn>

<https://developer.android.com/training/data-storage/room> - Room DB