ŽILINSKÁ UNIVERZITA V ŽILINE

FAKULTA RIADENIA A INFORMATIKY

**ESP32 Senzory**

SEMESTRÁLNA PRÁCA

A picture containing black, screenshot, design

Description automatically generated

Vypracoval: Karol Šugár

Predmet: Vývoj mobilných aplikácii

Akademický rok: 2022/2023

# Špecifikácia zadania, definovanie problému

Zadanie: Zaznamenávanie údajov z meteostanice(ESP32 + senzory)

Popis aplikácie: Aplikácia bude môcť spracovať aktuálne informácie o teplote, vlhkosti– buď z diaľkového zariadenia ESP32 cez API alebo zmeteostanice, ktorámáAPI zadarmo. Ešte som sa nerozhodol, ktor ý zdroj informácii použijem. Dáta zo zariadenia ESP32 sa budú prenášať bezdrôtovo cez WI-FI protokol. Tieto dáta si aplikácia bude môcť uložiť do svojho lokálného priestoru a spracovať ich do grafu. Dokážeme aj ovládať LED na diaľku. Aplikácia má aj čiernu tému na večerné hodiny.

Návrh procesov : Generátor grafu: z uložených dát aplikácia vytvorí graf

Uloženie grafu: aplikácia dokáže vyexportovať graf do PNG súboru

Ovládanie LED: Aplikácia dokáže ovládať hardvér na diaľku.

Pripojenie sa na diaľkové zariadenie: Aplikácia vie sa pripojiť na zariadenie

Získavanie údajov : Aplikácia dokáže zo vzdialeného zariadenia cez JSON dáta získať dáta a uložiť si ich do vlastnej databázy

Analýza použitia aplikácie

Aplikácia slúži na zobrazenie aktuálnej teploty, vlhkosti a iných informácii z rôznych senzorov,ktoré sú dostupné na diaľkovom zariadení/meteostanice.

Tieto informácie sú veľmi užitočné a viem si predstaviť, že by sa aplikácia využívala na dennej báze.

Vďaka ukladaniu aktuálnej teploty dokážeme vygenerovať graf a spracovať tieto údaje ďalej, príp. porovnať s predchádzajúcimi grafmi, ktoré sme si v minulosti uložili.

Arduino&ESP32

<https://play.google.com/store/apps/details?id=com.Zdravevski.TalktoESP32>

TalktoESP32Bluetooth <https://play.google.com/store/apps/details?id=io.dabbleapp&hl=sk&gl=US>

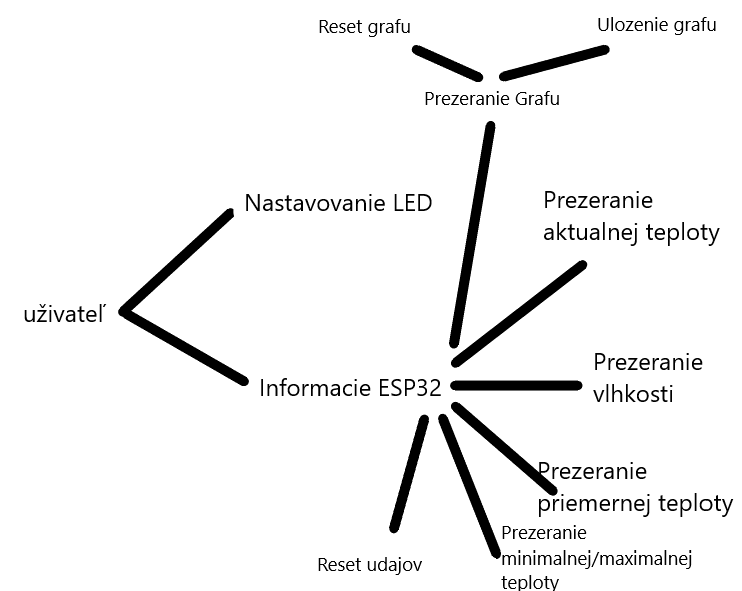
SHMUapp

<https://play.google.com/store/apps/details?id=sk.earendil.shmuapp&hl=sk&gl=US>

ESP32 sa vždy musí naprogramovať podľa svojej predstavy a teda vždy inak komunikuje so zariadením. Moje riešenie využíva JSON parsovanie na webstránke ESP32 a ukladanie do aplikácie

# Návrh riešenia problému

Krátka analýza



Návrh Aplikácie

UML DIAGRAM

A picture containing text, diagram, sketch, drawing

Description automatically generated

# Popis implementácie

Room Databáza- používam Room na vytvorenie lokálnej databázy pre ukladanie dát. Trieda AppDatabase je abstraktná trieda, ktorá rozširuje RoomDatabase a definuje konfiguráciu databázy vrátane entít a verzie. V tomto prípade je entita roomMeteo.MeteoStanica a verzia je 2.

Databáza sa vytvára v triede MainActivity pomocou metódy Room.databaseBuilder.Cez databázu pristupujem cez DAO triedu, ktorá je v podstate interface pre tabuľku meteoStanica.

V mojom projekte používam LiveData na sledovanie zmien v databáze a aktualizáciu rozhrania. V triede MeteoViewModel sa vytvorili atribúty typu MutableLiveData pre ukladanie teplotných a vlhkostných dát. Tieto atribúty sú inicializované v metóde init pomocou observera. Potreba init bola kvôli padaniu aplikácie pri štarte a odklonenie príkazu mimo mainThread. Týmto sa vytvorilo nenáročné rozhranie, ktoré sa aktualizuje len vtedy, ak sa databáza zmení.

Použil som ViewModel v mojom projekte na oddelenie logiky pre získavanie a spracovanie dát od používateľského rozhrania. Trieda MeteoViewModel obsahuje atribúty pre ukladanie dát získaných z databázy a metódy pre ich spracovanie. Samozrejme to aj bolo vyžadované, lebo som používal v MainAcitivity dva fragmenty a teda najlepšie mi prišlo použiť view na zdieľanie databázy. Týmto spôsobom som oddelil logiku pre prácu s dátami od rozhrania.

ViewModel tiež pomáha napríklad pri otočení obrazovky. Niekedy sa mi stávalo, že pri otočení sa mi vyresetovali všetky údaje v textových poliach .

**MainActivity+2fragmenty(FirstFragment-Status,SecondFragment-Graf)**

Najprv som začal s implementáciou načítavania údajov zo zariadenia, lebo sa mi to zdalo najdôležitejšie. Počas zisťovania ako chcem spracovávať dáta, som narazil na informáciu, že fragmenty sú vhodná cesta na zdieľanie informácii vrámci jednej aktivity. Vytvoril som xml content\_main, kde menim fragmenty. Navrch toho som dal activity\_main, kde je aj toolbar. Takto sa nebudú prekrývať prvky medzi toolbarom fragmentami.

A screenshot of a phone

Description automatically generated with medium confidence A screenshot of a phone

Description automatically generated with medium confidence

Navigácia:  
Fragmenty reagujú na tlačítko naspäť, idú na predošlý fragment.

Widgety:

Používam z widgetu(?) Toast

A screenshot of a computer

Description automatically generated with low confidence

Notifikácie:

Pri nepripojení na požadovanú WIFI, vyskočí notifikácia s tlačítkom ohľadom chyby na spode obrazovky.

A close up of a keyboard

Description automatically generated with low confidence

Externé knižnice:

com.github.PhilJay:MPAndroidChart

**LifeCycles:**  
Pri tejto aktivite som chcel používať onResume(), keď si použivateľ vyberie WIFI, na ktorú sa chce pripojiť , bohužiaľ kvôli chybe v androide som to nemohol spojazdniť. Používam ale aspoň onStop() cyklus na zastavenie threadu a nezaťažoval aplikáciu, keď sa vrátim z MainActivity do MenuActivity . Používam aj lifecycleScope na sledovanie

**MenuActivity**

A screenshot of a computer chip

Description automatically generated with medium confidenceTáto aktivita slúži len ako premostenie dvoch zvlášť aktivít, ActivityLED a MainActivity.

**ActivityLED**

Táto aktivita využíva hardvérových prvkov na zariadení -LED a mení jeho farby podľa výberu použivateľa. A picture containing screenshot, text, colorfulness, graphics

Description automatically generated A picture containing screenshot, text, colorfulness, graphics

Description automatically generated

Externé knižnice :

github.QuadFlask:colorpicker:0.0.15

# Zoznam použitých zdrojov

<https://github.com/PhilJay/MPAndroidChart/blob/master/MPChartExample/src/main/java/com/xxmassdeveloper/mpchartexample/LineChartTime.java>

<https://medium.com/mindorks/using-room-database-with-livedata-android-jetpack-cbf89b677b47>

<https://google-developer-training.github.io/android-developer-fundamentals-course-concepts-v2/unit-4-saving-user-data/lesson-10-storing-data-with-room/10-1-c-room-livedata-viewmodel/10-1-c-room-livedata-viewmodel.html>

<https://www.geeksforgeeks.org/how-to-add-image-to-drawable-folder-in-android-studio/>

https://github.com/QuadFlask/colorpicker#usage