Požiadavky na semestrálnu prácu

# Všeobecné informácie

## Základné požiadavky na semestrálnu prácu

1. Práca musí byť vytvorená ako aplikácia v jazyku **Kotlin s Jetpack Compose UI toolkit**.
2. Semestrálnu prácu je možné vypracovať v jazyku Flutter, alebo Swift (iOS). V prípade, ak chcete pracovať v týchto technológiách, **je nutné sa vopred dohodnúť** s vyučujúcimi. Termín je do konca 4. týždňa semestra, potom už nie je možné semestrálnu prácu vypracovať v inej technológii ako v bode 1.
3. Zdrojové kódy a dokumentácia sa odovzdávajú cez Github, GitLab, BitBucket, alebo iný GIT server. Vyučujúci **musí mať prístup** k zdrojovým kódom aplikácie. Android aplikácia sa musí dať skompilovať na poslednej dostupnej verzii Android Studia a musí byť spustiteľná na emulátore, ako aj na fyzickom zariadení. Android aplikácia, ktorá nespĺňa tieto podmienky, **nebude hodnotená**.
4. Ako tému semestrálne práce nie je možné odovzdať aplikácie, ktoré boli vytvárané na cvičeniach a aplikácie typu: *Poznámky, diár, pripomienky* a pod.

## Hodnotenie priebežného stavu sem. práce

1. Do konca **7. týždňa** semestra je potrebné odovzdať dokument s rozpracovaním semestrálnej práce.
2. Dokument musí byť vytvorený v **šablóne** (je zverejnená v MS Teams) s obsahom v nej uvedenej. Šablóna je vo formáte MS Word, nezabudnite ju pre odovzdaním uložiť ako PDF!
3. Študent odovzdá vytvorený dokument **výhradne vo formáte PDF** v publikovanom zadaní MS Teams (záložka *Assignments/Priradené úlohy*). Nie je potrebná osobná obhajoba.
4. Za kvalitne spracovaný dokument je možné získať max. 10 bodov.
5. Iný formát ako PDF dokument -3 body.
6. Z hodnotenia je potrebné získať min. 6 bodov, inak **nebude možné** odovzdať semestrálnu prácu.

## Obhajoba semestrálnej práce

1. Termíny obhajoby hotovej práce sú plánované v týchto termínoch:
   * **13. týždeň** semestra
   * v priebehu skúškového obdobia (bude upresnené neskôr)
2. Študent do zadania v *MS Teams* **vloží odkaz** na prácu umiestnenú na niektorom z GIT serverov.
3. Obhajoba semestrálnej práce v prihlásenom termíne bude pozostávať z prezentácie aplikácie, popisu tvorby aplikácie, orientácie v zdrojovom kóde, odpovedí na otázky, diskusie, atď.

## Dokumentácia

1. Súčasťou repozitára so semestrálnou prácou musí byť aj dokumentácia **výhradne vo formáte PDF**. Dokumentáciu uložte do adresára **dokumentacia**.
2. Na vypracovanie dokumentácie použite rovnakú šablónu ako pri priebežnom hodnotení semestrálnej práce.
3. Dokumentácia musí obsahovať nasledovné časti (môže rozširovať dokument k priebežnému stavu):
   * Skutočný návrh riešenia problému
     + Krátka analýza (napr. diagramy prípadov použitia realizovaných v aplikácii)
     + Návrh riešenia (napr. diagram tried, stavové a sekvenčné diagramy, diagram aktivít)
   * Popis implementácie – **klaďte dôraz na bodovo hodnotené časti** aplikácie, tak, aby bolo z dokumentácie zrejmé, ktoré hodnotené časti ste využili a akým spôsobom.
   * Zoznam použitých zdrojov
4. Bodové zrážky pri nedodržaní pravidiel:
   * Chýbajúca dokumentácia -20 bodov,
   * Zlá úroveň dokumentácie (neúplnosť, neprehľadnosť, veľké množstvo chýb, atď.) až do -10 bodov,
   * Aplikácia je iná ako v dokumente k priebežnému stavu -10 bodov
   * Nesprávny formát dokumentácie (iné ako PDF) -3 body.
   * Umiestnenie súboru v inom adresári ako **dokumentacia** -3 body.

# Hodnotenie aplikácie

1. Hodnotenie aplikácie sa skladá z nasledovných častí:
   * Zložitosť aplikácie – max. 40 bodov
   * Použitá programátorská technika – max. 30 bodov
   * Dizajn, UX aplikácie a výsledný dojem z aplikácie – max. 20 bodov
2. Časť aplikácie, ktorá je implementovaná ako *webview* a iba do neho načítaná webová stránka, sa pri hodnotení nebude brať do úvahy.
3. Identické časti aplikácie (napr. obrazovky) s minimálnymi odlišnosťami sa hodnotia len raz.

## Zložitosť aplikácie (40 bodov)

1. Všeobecné požiadavky**:**
   * Aplikácia musí správne reagovať na otočenie displeja z polohy na výšku na polohu na šírku (*portrait/landscape*) a naopak (pozor na uchovávanie stavu aktivity pred otočením, napr. pokiaľ je niekde použitý formulár, po otočení musia zostať polia vyplnené). Ak má aplikácia zablokované otáčanie, je potrebné to zdôvodniť. V prípade nekorektného správania sa aplikácie, prípadne pádu, bodová zrážka do -10 bodov.
   * Všetky zdroje ako texty, obrázky, multimédiá a pod. v projekte by musia uvedené ako *resources* (ak to nie je možné, je potrebné iné riešenie obhájiť), bodová zrážka do -10 bodov.
2. Obrazovky – max. 6 bodov za každú s prihliadnutím na jej spracovanie a komplexnosť.
   * Obrazovka je komponent zobrazený na celom displeji zariadenia a rieši unikátnu funkciu v aplikácii.
   * Maximálne je možné získať 24 bodov.
3. Využitie *AndroidX* komponentov:
   * Komponenty *LifeCycles, Navigation, Paging, Room, ViewModel, WorkManager* – max. 4 body za každý použitý komponent (nie za viacnásobné použitie toho istého komponentu).
4. *Widget miniaplikácia*max. 5 bodovza každú.
5. Notifikácia2 body za každú (max. 6 bodov).
6. Použitie externého frameworku / knižnice – max. 4 body za každý / každú.
7. Za využitie komponentov *service, broadcast receiver, content provider* – max. 4 body za každý podľa ich spracovania a komplexnosti.
   * Maximálny počet je 16 bodov.
8. Použitie senzora – max. 4 body za každý.
9. Použitie sieťovej komunikácie – max. 5 bodov.

## Použitá programátorská technika (30 bodov)

1. Práca s Git serverom (max. 10 bodov)
   * Počet a štruktúra git commitov
   * Správne pomenovanie commitov
   * Časové rozloženie práce na projekte
2. Návrh aplikácie, architektúra (max. 15 bodov)
   * Oddelenie logiky a používateľského rozhrania,
   * Použitie *clean* architektúry,
   * Použitie návrhových vzorov.
3. Dodržiavanie zásad kódovania (*coding standards*). (max. 5 bodov)
   * Vhodná voľba identifikátorov a názvov súborov,
   * Správne ošetrenie výnimiek,
   * Rozčlenenie aplikácie do balíčkov,
   * Dokumentačné komentáre nad každým verejným typom a štandardné komentáre všade tam, kde je to potrebné na pochopenie zdrojového kódu,
   * Žiadny duplicitný kód a pod.
4. Správne uplatnenie princípov OOP.

## Dizajn a UX aplikácie, výsledný dojem a teoretické znalosti (20 bodov)

Kritériá hodnotenia:

1. Kreativita nápadu a unikátnosť aplikácie
2. Intuitívnosť a jednoduchosť použitia, komfort používateľského rozhrania.
3. Výsledný dojem z aplikácie – rozsah, vzhľad, prepracovanosť, kompletnosť.
4. Orientácia študenta vo vlastnom kóde, schopnosť reagovať na položené otázky.
5. Znalosť základných konceptov vývoja Android aplikácií.

# Záverečné upozornenia

Neodovzdávajte kód aplikácie, ktoré ste vytvorili pomocou tutoriálov, alebo ChatGPT, pokiaľ im dôkladne nerozumiete. Súčasťou obhajoby je aj diskusia k práci. Ak študent nevie odpovedať na otázky k svojmu zdrojovému kódu, je to dôvod na ukončenie obhajoby a **udelenie známky Fx**, bez možnosti opravy. Podľa závažnosti prípadu, môže byť prípad postúpený na disciplinárnu komisiu FRI.

1. Študent odovzdaním práce čestne prehlasuje, že celý projekt (prácu) vypracoval samostatne s využitím uvedených zdrojov.
2. Ak pri tvorbe aplikácie použijete nejaký tutoriál, prípadne ChatGPT, je to potrebné uviesť priamo v zdrojovom kóde. Takto **neoznačená časť kódu bude považovaná za plagiát.**
3. Za plagiát sa bude považovať i časť zdrojových kódov, ktoré budú obsahovať rovnaký alebo refaktorovaný kód inej práce, resp. tutoriálu (zmenené názvy identifikátorov, poradie deklarácie a definície premenných, metód, atď.) – výnimku tvoria časti známych kódov a návrhové vzory (v tomto prípade uveďte zdroj, napr. URL, z ktorého ste čerpali).