

Розробка Android застосунка системи управління університетом.



Мета кваліфікаційної роботи

Задача - розгляд архітектурних підходів, шаблонів проектування, бібліотек та інструментів в розробці під ОС Android а також імплементація функціоналу в Android застосунку системи управління університетом, для взаємодії з університетським веб-сервісом по курсовим роботам, і авторизація за університетським акаунтом, необхідна для аутентифікації при запитах до вищезгаданого веб сервісу.

Авторизація повинна здійснюватися за допомогою Microsoft акаунту, що входить до організації університету. Для авторизації потрібно використати сервіс Azure AD, який надає можливість авторизуватись за допомогою акаунту організації, та отримати токен, необхідний для подальшої взаємодії з сервісами по курсовій роботі, як студент чи керівник роботи.

Схематично створити use cases на основі REST API веб серверу роботи з курсовими.

Функціонал для роботи з курсовими імплементувати відповідно до побудованих use cases.

Результатом роботи повинен бути імплементація бізнес логіки взаємодії з веб сервісом по курсовим роботам в Android застосунок системи управління університетом. Імплементація функціоналу повинна відповідати принципам і підходам, використовуваним в застосунку, а також сам застосунок повинен залишатись простим, зрозумілим, легко масштабованим.

Мета кваліфікаційної роботи

Також, для мене було важливо, щоб описова частина роботи могла бути максимально корисною для її читачів, які починають розробку під Android, і можливо потребують зібраної в одному місці інформації, по розгляду, аналізу і використанню загальних архітектурних підходів, аспектів розробки Android MVVM застосунку, авторизації та подальшої роботи з токеном, варіантів комунікації з сервером, etc.

Тому додаткова задача - побудова певних підрозділів описової роботи у вигляді інструкції, де будуть розглянуті певні підходи та принципи в розробці клієнтських застосунків для комунікації з сервером, а також, і в розробці під ОС Android загалом.

Описова робота має містити розділи, семантично еквівалентні до розділів:

- огляд та порівняння найпоширеніших архітектурних підходів при розробці під ОС Android;
- детальніший огляд архітектурного патерну MVVM в скоупі Android розробки;
- огляд популярних бібліотек та інструментів для розробки під ОС Android;
- підключення авторизації за допомогою акаунту Microsoft;
- впровадження функціоналу комунікації з сервером по REST API при MVVM підході.

Розглянуті підходи до вирішення завдання

Огляд та порівняння архітектурних підходів при розробці під ОС Android

Огляд бібліотек та інструментів, проблем які вони вирішують, та прикладів використання

Огляд реалізації комунікації застосунку з сервером

Схематичне створення use cases та побудова на їх основі бізнес логіки

Детальніший огляд використання архітектурного патерну MVVM з компонентами, які надає ОС Android в контексті розробки клієнтського Android застосунку для авторизації та подальшої комунікації з веб сервісами з використанням автентифікації на основі отриманого токєну



Проблеми пов'язані з розробкою та їх вирішення

Проблема розуміння предметної області

Проблема створення різних інтерфейсів доступу до функціоналу залежно від ролі користувача

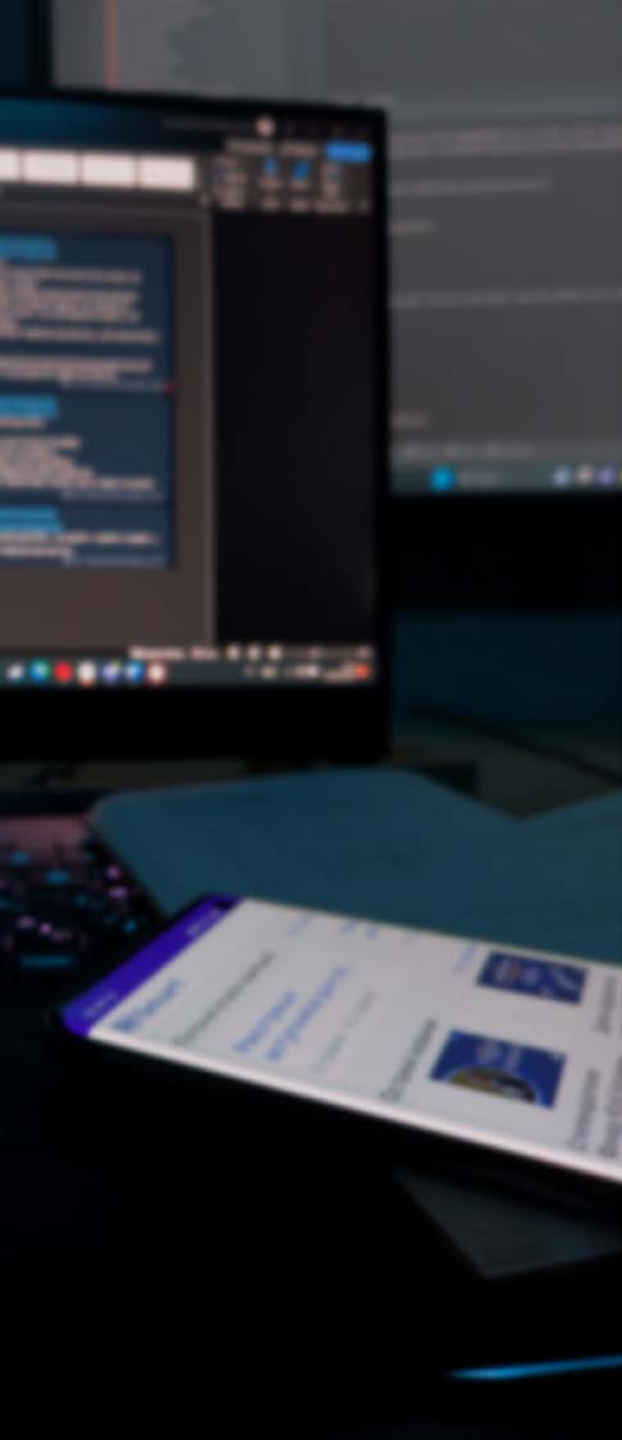
Проблема реалізації аутентифікації Microsoft OAuth за допомогою Azure AD.



Результати роботи: Огляд та порівняння архітектурних підходів при розробці Android застосунків

В розрізі даної роботи було оглянуто та порівняно різні архітектурні підходи в розробці під ОС Android, також оглянуто детальніше MVVM підхід та бібліотеки й інструменти, які зручно використовувати при даному підході. В ході виконання роботи, було лаконічно визначено переваги всіх розглянутих підходів та інструментів, такі як перевикористання компонентів, спрощення процесу розробки, спрощення тестування, зменшення залежності між компонентами. Також для бібліотек наведено приклади використання та зазначено головні проблеми, які вони вирішують.

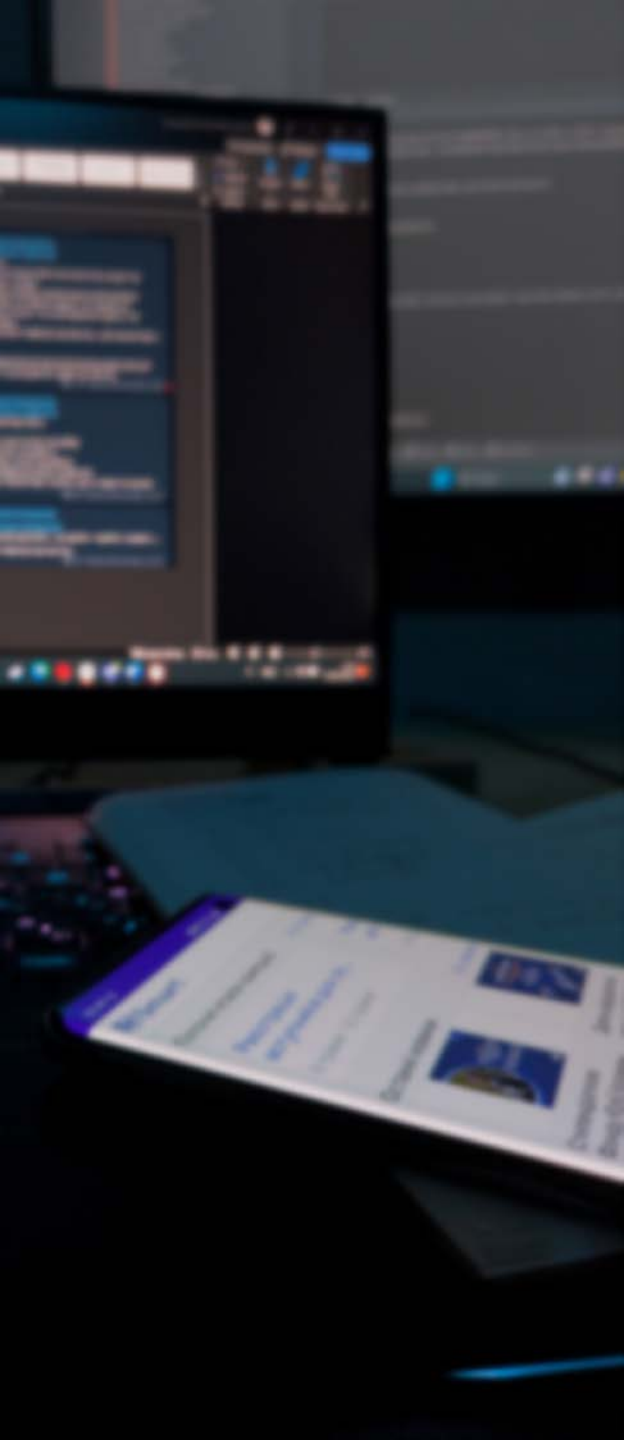
Дана робота може слугувати вступом в розробку клієнтських застосунків під ОС Android, з вимогами збереження зрозумілості коду та збереження легкої масштабованості при ймовірних подальших сценаріях розробки застосунку.



Результати роботи: Інші аспекти розробки Android додатку

Зроблено акцент на важливості певних моментів в розробці, безпосередньо не пов'язаних з написанням коду, наприклад на розумінні предметної області, схематичного опису бізнес логіки, а також на плануванні процесу розробки, на проблемних моментах, та пошуку елегантних рішень.

В цілому, розробка Android-додатків вимагає уваги до деталей і розуміння ширшого контексту, а не просто написання коду. Використання відповідних архітектурних підходів, бібліотек та інструментів може значно полегшити розробку та забезпечити якість та розширюваність додатку.



Результати роботи: розглянуто можливості подальшого розвитку

Крім того, в роботі було розглянуто можливості подальшого розвитку застосунку. Одним з таких напрямків розвитку може бути імплементація сервісів, пов'язаних з вибором дисциплін, груп, переглядом індивідуальних планів та розкладом. Реалізація цих сервісів потребуватиме глибшого аналізу предметної області та розуміння логіки всіх необхідних процесів у системі.

При розробці застосунку в цьому напрямку можуть виникнути різноманітні проблеми, наприклад, необхідність зменшення навантаження на сервер. Важливо проаналізувати та врахувати всі можливі складнощі, що можуть виникнути, і знайти ефективні рішення для їх вирішення.

Результати роботи: подальша розробка

Імплементація таких сервісів може мати значний вплив на спільноту університету. Наприклад, студенти зможуть легко обирати дисципліни, формувати свої індивідуальні навчальні плани та отримувати доступ до розкладу занять безпосередньо з мобільного пристрою Android. Це сприятиме покращенню процесу навчання та забезпеченню зручності для студентів.

Ураховуючи потенційні переваги і користь для спільноти університету, імплементація цих сервісів може бути значним кроком у поліпшенні функціональності та користувацького досвіду застосунку. Важливо продовжувати глибокий аналіз предметної області, шукати оптимальні рішення та працювати над подальшим розвитком застосунку, щоб забезпечити його успішне використання в університетській спільноті.