

DOI: 10.36910/6775-2524-0560-2020-40-18

УДК: 004.58 (34)

Гринюк Сергій Васильович, асистент<https://orcid.org/0000-0002-0080-3167>**Бортник Катерина Яківна**, к.т.н., доцент<http://orcid.org/0000-0001-5282-099X>**Поліщук Микола Миколайович**, к.т.н., в.о. доцента<https://orcid.org/0000-0002-1218-5925>

Луцький національний технічний університет

МОБІЛЬНИЙ ДОДАТОК ДЛЯ РОБОТИ КУРАТОРА ГРУПИ ЗАСОБАМИ ANDROID STUDIO

Гринюк С.В., Бортник К.Я., Поліщук М.М. Мобільний додаток для роботи куратора групи засобами **Android Studio**. Для створення програмної системи у роботі було пройдено усі етапи розробки клієнт-серверного додатку, а саме: розглянуто декілька додатків-аналогів, що реалізують схожий функціонал, на етапі проектування програмної системи розроблено архітектуру клієнта, сервера та додатку в цілому, складено загальну структуру діаграму таблиць в базі даних, на етапі програмної реалізації, з використанням об'єктноорієнтованого підходу в інтегрованому середовищі розробки програмного забезпечення **Android Studio**, реалізовано функції клієнтської системи, які написані на мові програмування **Kotlin**. В середовищі веб-програмування **PHPStorm** було розроблено набір серверного функціоналу, який спростив та покращив роботу додатку з базами даних. Розроблено інтерфейс програми.

Ключові слова: додаток, PHP, сервер, архітектура, програма.

Гринюк С.В., Бортник Е.Я., Полищук Н.Н. Мобильное приложение для работы куратора группы средствами **Android Studio**. Для создания программной системы в работе было пройдено все этапы разработки клиент-серверного приложения, а именно: рассмотрено несколько приложений-аналогов, реализующих похожий функционал, на этапе проектирования программной системы разработаны архитектуру клиента, сервера и приложения в целом, составлен общую структуры диаграмму таблиц в базе данных, на этапе программной реализации, с использованием объектноориентированого подхода в интегрированной среде разработки программного обеспечения **Android Studio**, реализованы функции клиентской системы, написанные на языке программирования **Kotlin**. В среде веб-программирования **PHPStorm** был разработан набор серверного функционала, который упростил и улучшил работу приложения с базами данных. Разработан интерфейс программы.

Ключевые слова: приложение, PHP, сервер, архитектура, программа.

Hryniuk S., Bortnyk K., Polishchuk M. Mobile application for working as a group curator by means of **Android Studio**. To create a software system, all stages of client-server application development were passed, namely: considered several analog applications that implement a similar function, at the design stage of the software system developed the architecture of the client, server and application as a whole, compiled a general structure chart tables in the database, at the stage of software implementation, using object-oriented approach in an integrated development environment **Android Studio** software, implemented the functions of the client system, which are written in the programming language **Kotlin**. In the **PHPStorm** web programming environment, a set of server functionality has been developed that simplifies and improves the application's application with databases. The program interface has been developed.

Keywords: application, PHP, server, architecture, program.

Вступ. Викладач який перебуває на посаді куратора академічної групи повинен стимулювати студентів до саморозвитку та самонавчання. Доступ до системи де знаходяться всі оцінки, кожного з його студентів дозволить краще йому орієнтуватися хто як навчається, а підтримка зв'язку з батьками студентів допоможе краще реалізовувати свої функції.

Перед як почати розробляти свій додаток потрібно проаналізувати декілька найкращих існуючих аналогів, та можливо перейняти собі деякі хороші практики його реалізації, та навпаки - уникнути поганих практик.

При розробці таких застосунків можна спочатку зробити версію з базовим функціоналом, а потім доповнювати її новими функціями і оновляти. Особливо дієвим такий підхід буде у тому випадку, якщо реалізувати систему відгуків та пропозицій від реальних користувачів.

У межах цієї проаналізовано два додатки з функціоналом подібним до цього.

Одним з таких додатків є сервіс «Нові знання» (режим доступу: <https://nz.ua/page/support>) . Загальний вигляд головної сторінки сервісу «Нові знання» подано на рис. 1.



Рисунок 1 – Скріншот головної сторінки сайту «Нові знання»

Цей сайт є однією зі складових Всеукраїнського проекту «Курс: Освіта».

Його користувачі – це керівники загальноосвітніх навчальних закладів України (директори шкіл, їх заступники), завучі, вчителі, учні, батьки або опікуни школярів. Якщо переглянути функціонал сайту, то можна виділити основні функції, які можливо реалізувати в рамках цього проекту.

Батьки та школярі, після того як отримують у своїх вчителів дані для авторизації, можуть переглядати оцінки за весь період навчання у своєму щоденнику, та переглядати домашні завдання. Також батьки можуть проконтролювати відвідуваність занять своєю дитиною, та слідкувати за шкільними новинами.

А вчителі в свою чергу можуть керувати навчальними журналами, взаємодіяти з розкладом та журналом свого класу, створювати внутрішню звітність. При цьому адміністрація школи може керувати всіма журналами та надавати доступ до них класним керівникам.

Також на сайті можна переглянути всіх шкіл, які користуються даним сервісом та при потребі подати заявку на реєстрацію в будь-яку з них.

Серед плюсів сайту «Нові знання» можна виділити:

- Велика база навчальних закладів.
- Можливість реєстрації в навчальні заклади дошкільної та середньої освіти.
- Простота у використанні.
- Наявність технічної підтримки та детальних інструкцій використання.

Ще одним сайтом зі схожим функціоналом є «Щоденник» (рис.2) (режим доступу: <http://shodennik.ua>). Це Всеукраїнська безкоштовна освітня мережа, що формує електронне середовище для вчителів, учнів та їх батьків. Проект працює за підтримки Міністерства освіти і науки, молоді та спорту України, Інституту Інноваційних технологій і змісту освіти, регіональних адміністрацій, управлінь освіти. На сьогоднішній день це найбільша освітня мережа в Україні, яка надає свій функціонал на безкоштовній основі. За даними на травень 2020 року в мережі налічується більше 10 620 шкіл і понад 1 028 430 учнів, 148 514 вчителів та 253 060 батьків.

Серед функціоналу цього сервісу можна виділити:

- Особиста сторінка для кожного користувача.
- Медіатека.
- Можливість створювати групи та події.
- Словники та перекладач.



Рисунок 2 – Скріншот головної сторінки сайту «Щоденник»

Детальний огляд всіх цих сервісів дозволив прийняти рішення про те яку функціональність матиме розроблювана у межах цієї роботи система «Android-додаток для підтримки роботи куратора групи»

Постановка проблеми. Мета роботи – проаналізувати існуючі веб-сайти навчальних закладів, виділити їх основні переваги та недоліки, а також створити, на основі проведеного аналізу, Android додаток, який би включав всі плюси розглянутих сайтів, та виключав їх мінуси.

Для реалізації додатку була обрана клієнт-серверна архітектура, яка надасть змогу одночасного доступу до системи багатьох користувачів. Основною мовою програмування для клієнта буде Kotlin, а для сервера PHP з підключенням до бази даних MySQL. Середовищем розробки для клієнта було обрано Android Studio, а для сервера ліцензійна версія PHPStorm [1].

Виклад основного матеріалу. Куратором академічної групи обирається провідний фахівець, досвідчений педагог з урахуванням напрямку підготовки та профілю діяльності факультету, стажу викладацької роботи у вищому навчальному закладі. Обов'язковою умовою його перебування на цій посаді є викладання лекцій або проведення семінарсько-практичних занять в даній академічній групі.

Від куратора значною мірою залежить дисциплінованість та успішність студентів. Якщо куратор зуміє з перших днів зорганізувати студентів, зосередити їхню увагу на важливих речах, допоможе й

підтримає там, де вчорашній школяр цього потребує, то студент швидко адаптується до нових умов і вимог. У роботі куратора передбачається виховна діяльність академічної групи, створення умов для самовираження кожного студента і розвитку кожної особистості. Доступ куратора до додатку який відображає успішність кожного студента його групи, дозволить йому стимулювати їх до навчання, та контролювати їхню відвідуваність, а також при необхідності взаємодіяти з батьками студента.

Тому для підвищення ефективності роботи куратора, потрібно розробити таке програмне забезпечення, яке дозволить куратору за успішністю та відвідуванням студентів його академічної групи, та при потребі корегувати її за допомогою спілкування з самим студентом, або його батьками.

На сьогодні практично всі вик користуються смартфонами, і значна половина з них це користувачі операційної системи Android, тому було обрано саме цю платформу для розробки програмної системи. Також додаток має ряд переваг над веб-сайтами чи вебдодатками на ПК, а саме [1,2]:

1. Завжди доступний (смартфон, на відміну від ПК, ви завжди носите з собою);
2. Швидкий доступ до системи (не потрібно переходити на сайт системи в браузері, достатньо просто вибрати його на робочому столі вашого смартфона, який завжди з вами);
3. Зручний інтерфейс;
4. Оптимізоване використання інтернет трафіку, оскільки додаток завантажує тільки данні які потрібні йому для обробки (нічого лишнього, ніяких скриптів, плагінів і тд.)

Оскільки кожен куратор має одну окрему групу, то у системі будуть відбуватися такі бізнес-процеси:

1. Процес реєстрації:
 - 1.1. Заповнення всіх необхідних даних (П.І.Б., Email, номер телефону, пароль, група і тип запису).
 - 1.2. Обробка даних на клієнті.
 - 1.3. Відправка даних на сервер.
 - 1.4. Перевірка даних на сервері.
 - 1.5. Створення запису в базі даних (при успішній перевірці).
 - 1.6. Повернення в клієнт результату виконання.
2. Процес авторизації
 - 2.1. Заповнення всіх необхідних даних (Email, пароль).
 - 2.2. Обробка даних на клієнті.
 - 2.3. Відправка даних на сервер.
 - 2.4. Перевірка існування користувача з вказаною поштовою скринькою та відповідності введеного паролю.
 - 2.5. Повернення в клієнт результату виконання:
 - 2.5.1. Дані користувача та його групи при успішній авторизації.
 - 2.5.2. Повідомлення про помилку при невдалій авторизації.
3. Можливості роботи в системі:
 - 3.1. Перегляд списку студентів групи та їх успішності.
 - 3.2. Відправка повідомлень на поштову скриньку студента, або його батьків.
 - 3.3. Додавання нових користувачів.
4. Процес відправки повідомлення:
 - 4.1. Вибір адресатів.
 - 4.2. Введення тексту повідомлення.
 - 4.3. Відправка повідомлення на поштову скриньку.

Для роботи з додатком, користувач (Адміністратор/Куратор/Студент) повинен авторизуватись. Після авторизації програма виведе на екран інформацію та функції, до яких користувачу надано доступ.

В студента буде доступ тільки до власного рейтингу та можливості зв'язатись з його куратором чи кимось з групи.

Куратор зможе переглядати список групи та їх успішності, матиме всю основну інформацію про батьків кожного студента і з може написати будь-кому з ким повідомлення на емейл, або подзвонити. Також зможе реєструвати нових студентів у своїй групі.

В свою чергу користувач з правами доступу «Адміністратор» матиме повний доступ до всіх функцій, таких як: реєстрація нових користувачів в будь-якій групі та з будь-якими правами доступу, перегляд списку всіх груп, успішності студентів та інформації про них.

Додаток матиме «клієнт-серверну» архітектуру. Тобто він буде складатись з двох основних частин: клієнт та сервер. На даний момент це дві практично невід'ємні частини будь-якого додатку,

пов'язаного з роботою через мережу «Інтернет». Вони взаємодіють між собою за допомогою різноманітних мережевих протоколів, наприклад, IP-протокол, HTTP-протокол, FTP-протокол та інші. При цьому варто зазначити що в основі цієї взаємодії лежить принцип того, що клієнт завжди перший звертається до сервера. Такі звернення називаються HTTP запити, вони також містять в собі інформацію, яка дає серверу зрозуміти як їх обробляти. В свою чергу клієнт може тільки відповідати на запити клієнта - це називають HTTP-відповіддю. Але крім основної інформації вони містять в собі ще й спеціальні коди стану, які дозволяють клієнту дізнатись те, як сервер прийняв його запит [1,3].

Цей архітектурний шаблон був розроблений в першу чергу для того, щоб розділити навантаження між учасниками обміну інформацією, а також щоб розділити код «постачальника» і «замовника» даних. Тому зазвичай клієнтське програмне забезпечення і серверне програмне забезпечення працюють на різних пристроях паралельно, хоча можуть і на одному. При цьому один сервер може обробляти одночасно багато запитів від різних клієнтів, їх максимальна кількість залежить від потужності серверної машини і від того що клієнт хоче отримати у відповідь від сервера.

Домінантна кількість мережевих протоколів побудовані на принципі «клієнт-сервер», тому в їх основі зазвичай лежать однакові, або схожі принципи взаємодії, а різниця проявляється лише в деталях, які обумовлені особливостями і специфікацією області, для якої вони розроблялись.

Конкретно у цьому проекті буде використано HTTP-протокол, так як він доволі простий у реалізації.

На рисунку 3 наведено діаграму, яка візуалізує клієнт-серверного додатку

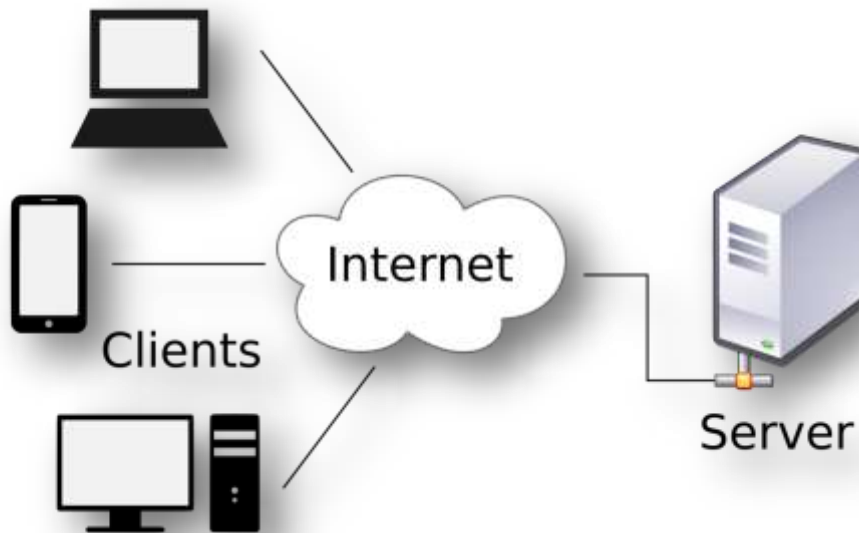


Рисунок 3 – Структура клієнт-серверного додатку

Послідовність роботи системи авторизації:

1. Клієнт вводить дані.
2. LoginViewModel відслідковує всі зміни даних на LoginActivity та виводить сповіщення при неправильності введених даних.
3. Клієнт нажимає кнопку «Авторизуватись».
4. LoginViewModel перехоплює всі дані та надсилає їх до AsyncLogin.
5. AsyncLogin в свою чергу створює новий фоновий процес в якому дані надсилаються до LoginDataSource на обробку.
6. LoginDataSource надсилає всі дані на сервер, отримує відповідь та обробляє її. Після обробки звертається назад до класу AsyncLogin та передає результат авторизації (Повідомлення про помилку, або інформацію про авторизованого користувача).
7. AsyncLogin передає результат в екземпляр класу LoginViewModel
8. А LoginViewModel сповіщає LoginActivity про те, які функції йому потрібно виконати в залежності від результату.
9. LoginActivity виводить повідомлення про помилку, чи авторизує користувача та перенаправляє на головний екран додатку MainActivity [1,4].

На основі всіх приведених вище класів і працює практично вся система авторизації в додатку. Аналогічно до неї працюють й інші частини додатку. В тому числі система реєстрації, єдина відмінність полягає в тому, що при реєстрації доводиться обробляти більшу кількість даних.

Нижче буде наведено декілька скріншотів інтерфейсу додатку.

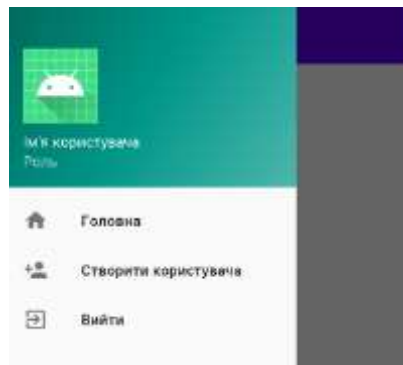


Рисунок 4 – Скріншот меню навігації додатку

Прізвище _____

Ім'я _____

По батькові _____

Електронна пошта _____

Номер телефону _____

Пароль _____

Повторіть пароль _____

Група ▼

Студент ▼

Рисунок 5 - Авторизація

Група ▼

Студент ▼

П.І.Б. батька _____

Емейл батька _____

Номер телефону батька _____

П.І.Б. матері _____

Емейл матері _____

Номер телефону матері _____

СТВОРИТИ КОРИСТУВАЧА

Рисунок 6 - Авторизація

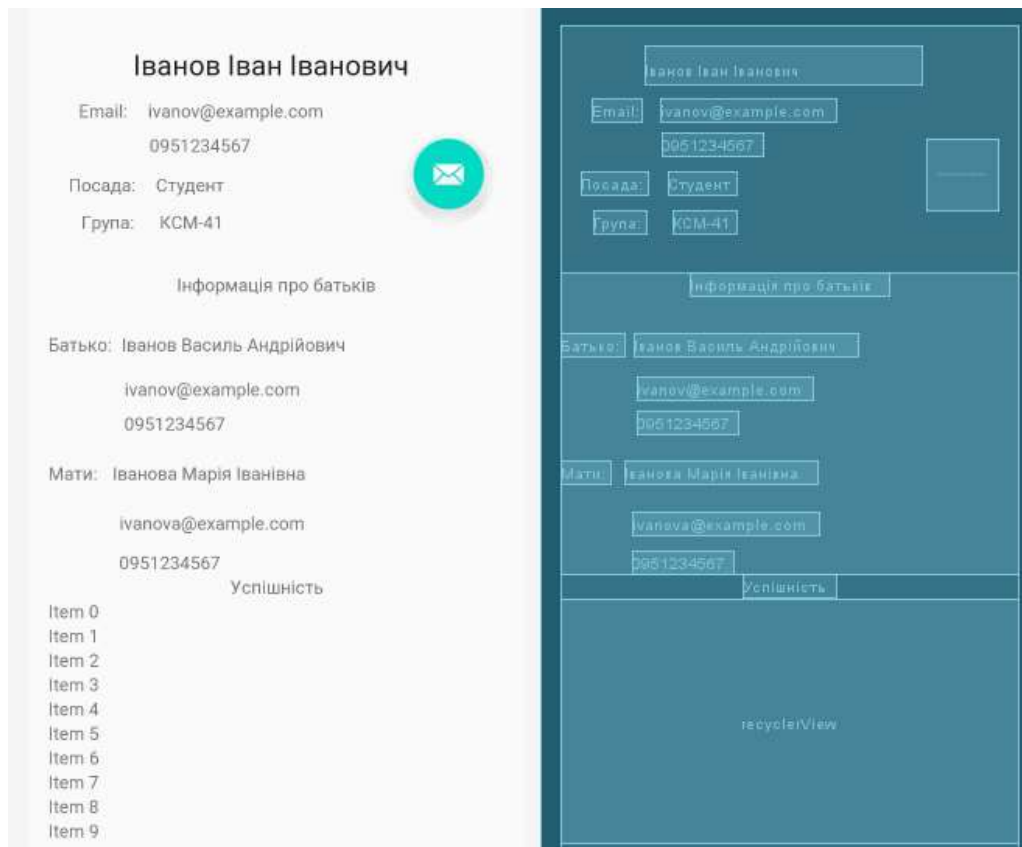


Рисунок 7 – Скріншот інформації про користувача

Висновки. Під час створення програмної системи у роботі було пройдено усі етапи розробки клієнт-серверного додатку, а саме:

1. Було розглянуто декілька додатків-аналогів, що реалізують схожий функціонал.
2. На етапі проектування програмної системи розроблено архітектуру клієнта, сервера та додатку в цілому. Також складено загальну структури діаграму таблиць в базі даних.
3. На етапі програмної реалізації, з використанням об'єктноорієнтованого підходу в інтегрованому середовищі розробки програмного забезпечення Android Studio, реалізовано функції клієнтської системи, які написані на мові програмування Kotlin. Також в середовищі веб-програмування PHPStorm було розроблено набір серверного функціоналу, який спростив та покращив роботу додатку з базами даних. Розроблено інтерфейс програми.

Було змодельовано невелику рейтингову систему успішності студентів, щоб протестувати додаток, та наочно показати можливість роботи з рейтинговою системою. В майбутньому можливо буде удосконалити додаток та прив'язати до університетської системи оцінювання студентів. Програма дає куратору групи можливість відслідковувати в реальному часі успішність всіх його студентів, та при потребі зв'язатись з окремим студентом чи його батьками для подальшого підвищення продуктивності навчання.

Список бібліографічного опису.

1. Коновалюк Д.Ю. Мобільний додаток для роботи куратора групи засобами Android Studio: випускна кваліфікаційна робота: 123 «Комп'ютерна інженерія»/ Коновалюк Дмитро Юрійович . – Л., 2020. – 79 с.
2. Вендров А.М. Проектирование программного обеспечения экономических информационных систем: Учебник. – 2-е изд., перераб. и доп.– М.: Финансы и статистика, 2006. – 544 с: ил.
3. Documentation for app developers [Електронний ресурс]. - Режим доступу : <https://developer.android.com/docs?hl=ru>
4. Kotlin [Електронний ресурс]. - Режим доступу: <https://uk.wikipedia.org/wiki/Kotlin>
5. Немного о Kotlin [Електронний ресурс]. - Режим доступу: <https://habr.com/ru/post/277479/>

References.

1. Konovalyuk D.Yu. Mobile application for the work of the curator of the group by means of Android Studio: final qualifying work: 123 "Computer Engineering" / Konovalyuk Dmitry Yurievich. - L., 2020. - 79 p.
2. Vendrov AM Software design of economic information systems: Textbook. - 2nd ed., Reworked. i dop.– M .: Finansy i statistika, 2006. - 544 s: il.
3. Documentation for app developers [Electronic resource]. - Access mode: <https://developer.android.com/docs?hl=en>
4. Kotlin [Electronic resource]. - Access mode: <https://uk.wikipedia.org/wiki/Kotlin>
5. A little about Kotlin [Electronic resource]. - Access mode: <https://habr.com/ru/post/277479>