



А. В. Лебединський¹, Ю. В. Щербініна², Б. С. Карпішен³

¹ХНАДУ, м. Харків, Україна, lebedynskyi@khadi.kharkov.ua,
ORCID iD: 0000-0002-5086-8209

²ХНАДУ, м. Харків, Україна, sherbinina953@gmail.com, ORCID iD: 0009-0006-9262-2633

³ХНАДУ, м. Харків, Україна, karpishen.bogdan@gmail.com, ORCID iD: 0009-0001-1790-9048

ІНТЕГРАЦІЯ ANDROID-ТЕХНОЛОГІЙ У ПРОЦЕС НАВЧАННЯ ОСНОВАМ АВТОМОБІЛЬНОГО ДИЗАЙНУ

Інтеграція Android-технологій у процес навчання основам автомобільного дизайну спрямована на вирішення ключової проблеми — опанування цікавої та перспективної професії. Розробка мобільного додатку створює можливості для формування спільноти однодумців, забезпечуючи користувачів приватними чатами для спілкування та тематичними дошками для обміну корисною інформацією через Інтернет-посилання. Освітня Android-платформа також може використовуватися як допоміжний інструмент у різних навчальних закладах, наприклад, у Харківському національному автомобільно-дорожньому університеті, з урахуванням його айденітики. Мобільний додаток містить лекції, тести та практичні завдання, що поєднують текстові й графічні матеріали, щоб залучати користувачів. Інтуїтивно зрозумілий функціонал, сучасний дизайн та доступ до нових знань забезпечують повне задоволення потреб користувачів і сприяють ефективному навчанню.

АВТОМОБІЛЬ, МОБІЛЬНИЙ ДОДАТОК, ДИЗАЙН, ОНЛАЙН-НАВЧАННЯ, ANDROID, JAVA

A. V. Lebedynskyi, Y. V. Shcherbinina, B. S. Karpishen Integration of Android Technologies into the Process of Teaching the Basics of Automotive Design. The integration of Android technologies into the process of teaching the basics of automotive design is aimed at solving a key problem - mastering an interesting and promising profession. The development of the mobile application creates opportunities for the formation of a community of like-minded people, providing users with private chats for communication and thematic boards for sharing useful information via Internet links. The educational Android platform can also be used as an auxiliary tool in different educational institutions, such as Kharkiv National Automobile and Highway University, taking into account its identity. The mobile app contains lectures, tests, and practical exercises that combine text and graphics to engage users. Intuitive functionality, modern design and access to new knowledge ensure full satisfaction of users' needs and promote effective learning.

AUTOMOBILE, MOBILE APPLICATION, DESIGN, ONLINE EDUCATION, ANDROID, JAVA

Вступ

Мобільні додатки стають незамінними інструментами в процесі навчання. Вони роблять освіту доступною, дозволяючи працювати з матеріалами у власному ритмі. Завдяки інтерактивності та зручному формату, мобільні додатки не лише спрощують засвоєння інформації, а й роблять процес навчання цікавим і ефективним. Попри різноманіття освітніх програм, на платформі Google Play Market відсутні додатки, присвячені вивченню дизайну автомобілів, що створює потребу в розробці такого продукту.

Цей проект спрямований на створення освітнього мобільного додатку, який допоможе користувачам освоїти професію дизайнера автомобілів. Додаток включатиме теоретичний матеріал, практичні завдання, тести для закріплення знань, а також інструменти для інтерактивної взаємодії. Користувачі зможуть відстежувати свій прогрес через календар успішності та отримувати підтримку через приватні чати. Додаток також сприятиме формуванню спільноти, де зацікавлені користувачі зможуть обмінюватися ідеями та інформацією.

Результатом проекту стане функціональний мобільний додаток, який буде цікавим як студентам, так і провідним дизайнерам та інженерам автомобільної промисловості. Він може бути інтегрований

в освітній процес, наприклад, у Харківському національному автомобільно-дорожньому університеті, доповнюючи традиційні методи навчання. Цей додаток полегшить доступ до якісної освіти у сфері автомобільного дизайну, поєднуючи сучасні технології та освітні потреби.

1. Аналіз предметної області

1.1. Роль мобільних додатків у сучасному онлайн-навчанні

Мобільні пристрої набувають все більшого поширення, а освітні додатки для них стають дедалі популярнішими, що робить мобільне навчання важливою складовою сучасних освітніх технологій.

Використання новітніх методів навчання, зокрема цифрових технологій, сприяє постійному вдосконаленню системи вищої освіти. Інтернет як глобальний феномен значно вплинув на розвиток дистанційного навчання, яке охоплює широку аудиторію та є ключовим компонентом освітньої модернізації.

З 2019 року, під час пандемії COVID-19, більшість навчальних закладів перейшли на дистанційний формат через необхідність дотримання безпеки. Цей вимушений перехід започаткував масове впровадження онлайн-освіти.

У дослідженні мобільного навчання [1] зазначено, що ініціатива Elphick iPilot надала студентам

електронні пристрої для освітніх цілей. Результати опитувань показали, що такі проекти позитивно впливають на цифрову грамотність студентів, стимулюють їхню креативність, підвищують впевненість та сприяють ефективності навчання. Також покращується комунікація в групах, особливо під час колективної роботи. Це демонструє, що технічні можливості мобільних пристроїв можуть стати корисним інструментом у навчальному процесі.

Мобільні додатки для освіти мають численні переваги, зокрема доступність навчання в будь-який час і будь-де. Користувачі можуть обирати курси різної спрямованості: від вивчення мов до освоєння технічних навичок. Багато додатків пропонують інтерактивні завдання, тести, лекції у форматі тексту та відео, що дає змогу адаптувати навчання до індивідуальних потреб кожного користувача.

Ще однією значною перевагою є економія коштів, оскільки дистанційне навчання дозволяє студентам уникнути витрат на проживання та транспорт. Мобільне навчання сприяє розвитку самостійності: воно вчить знаходити та обробляти інформацію, користуватися онлайн-ресурсами, розвивати навички й отримувати знання у власному темпі [2].

Освітній центр GoStudy рекомендує кілька корисних додатків для студентів [3]:

– Coursera. Цей додаток відкриває доступ до численних курсів, створених університетами та організаціями з усього світу. Він пропонує відеолекції, практичні завдання, тестування та можливість отримання сертифікатів після завершення навчання.

– Udemy. Платформа з широким вибором курсів у різних галузях — від програмування та дизайну до бізнесу та мистецтва. Усі курси включають відеоуроки, інтерактивні вправи та тести.

– edX. Додаток, який пропонує курси від топових університетів і організацій, таких як MIT, Harvard і Microsoft. Тут можна знайти навчальні програми з різних дисциплін.

– Duolingo. Якщо ви прагнете вивчити нову мову, цей додаток стане ідеальним помічником. У ньому доступно понад 30 мов, а уроки подані в ігровій формі для легшого засвоєння матеріалу.

– Khan Academy. Пропонує тисячі безкоштовних відеоуроків із математики, природничих наук, економіки та інших тем, що робить навчання доступним для кожного.

– Any.do. Зручний додаток для управління завданнями. Він дозволяє створювати персональні списки справ, встановлювати нагадування та організувати свій день максимально ефективно.

На ринку існує чимало додатків для організації завдань, вивчення іноземних мов або проходження різноманітних освітніх курсів. Однак серед них відсутній спеціалізований додаток, який би надавав

структуровану та корисну інформацію для опанування професії автомобільного дизайнера — від знайомства з історією автомобілебудування до опанування навичок 3D-моделювання.

1.2. Дослідження додатків-аналогів для навчання

Рішення для створення додатків, орієнтованих на вивчення автомобільного дизайну, поки що відсутні. Проте існує велика кількість освітніх додатків, які пропонують різноманітний функціонал і дизайн. Незважаючи на відмінності у візуальному оформленні та досвіді користувачів, їх об'єднує спільна мета — забезпечити зручність і функціональність, які задовольняють потреби аудиторії.

Дизайнери часто спираються на власний досвід, але слід пам'ятати, що технологічне середовище постійно змінюється, і підходи, які працювали раніше, можуть втратити актуальність. Саме тому важливо регулярно переглядати та адаптувати рішення відповідно до сучасних вимог [4].

Аналіз і вивчення аналогічних додатків є важливим етапом у процесі створення власного продукту. Дослідження існуючих рішень дозволяє розуміти, як вони вирішують потреби користувачів і які недоліки мають. Це допомагає знайти можливості для вдосконалення, визначити конкурентні переваги, а також виявити ніші, які ще не заповнені. Такий підхід дозволяє створити додаток, який краще відповідає очікуванням цільової аудиторії та пропонує унікальні функції. Аналіз успішних кейсів може надихнути на впровадження інноваційних функцій, що зроблять продукт більш привабливим, і допоможе уникнути помилок, яких припустилися інші розробники. Крім того, для систематичного підходу до аналізу ринку та потреб користувачів корисно застосовувати інструменти, такі як Business Model Canvas.

Business Model Canvas — це стратегічний інструмент, який допомагає проектувати, аналізувати та оцінювати бізнес-моделі. Він широко використовується як стартапами, так і великими компаніями по всьому світу. Ця методика, описана в книзі Business Model Generation [5], дозволяє глибше зрозуміти потреби ринку, оцінити конкурентів та сформулювати ефективний план розвитку продукту.

Для визначення потреб майбутніх користувачів додатку та підкреслення цінності свого продукту, було використано інструмент Value Proposition Canvas. Цей інструмент дозволяє компаніям створювати та вдосконалювати свої продукти чи послуги, орієнтуючись на вимоги та потреби своїх клієнтів, доповнюючи методику Business Model Canvas [6].

Заповнення Value Proposition Canvas передбачає детальний аналіз потреб, бажань та проблем цільової аудиторії. На рис. 1 показано, як саме цей додаток відповідає на існуючі запити та проблеми користувачів.

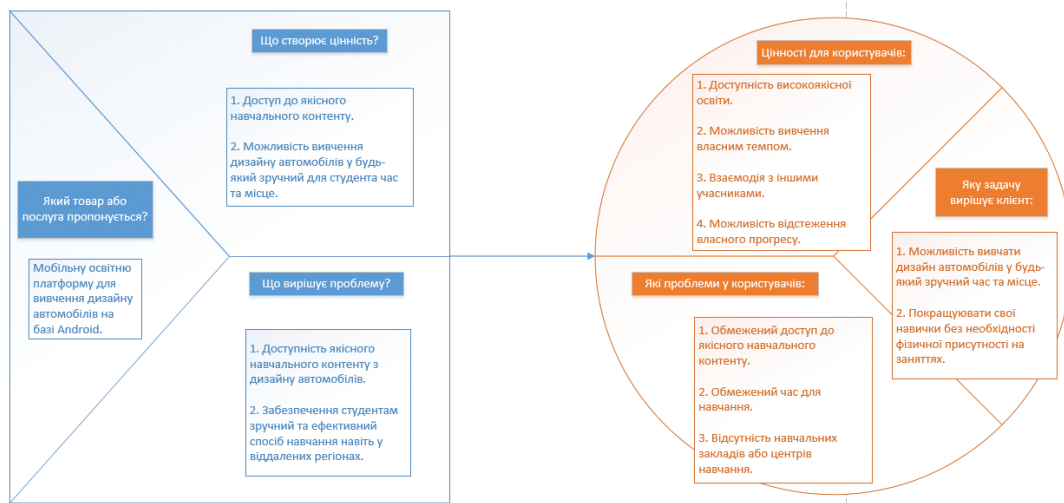


Рис. 1. Value Proposition Canvas для мобільного додатку з вивчення автомобільного дизайну

Перелічимо три додатки-аналоги (табл. 1) для навчання, що містять різний функціонал:

1. Google Classroom — це безкоштовний сервіс, розроблений Google який призначений для навчання та співпраці в навчальних закладах. Цей інструмент дозволяє вчителям створювати онлайн-класи, де вони можуть розміщувати завдання, надавати матеріали для читання, проводити тести, відстежувати прогрес учнів і спілкуватися з ними. Основні функції: створення класів; розміщення завдань; організація матеріалів; відстеження прогресу; спілкування.

2. Udeу — це онлайн-платформа для навчання, де користувачі можуть здобувати нові знання та навички в різних сферах життя через доступ до широкого

спектру курсів. Ця платформа надає можливість навчання на всіх рівнях, від початкового до професійного, і включає курси з таких галузей, як бізнес, програмування, маркетинг, дизайн, освіта тощо. Основні функції: різноманітність курсів; відкритий доступ; різні формати навчання; сертифікати; підтримка від авторів курсів.

3. SoloLearn — це мобільний додаток, який надає можливість навчання програмування та інших технічних навичок безкоштовно. Додаток створений для початківців та тих, хто бажає розвинути свої навички програмування на будь-якому рівні. Основні особливості: інтерактивні уроки; кодингові виклики; бейджі та сертифікати; практичні проекти.

Таблиця 1

Аналіз розглянутих додатків для навчання онлайн

Критерій	Додатки-аналоги		
	Google Classroom	Udeу	Sololearn
Зручний та сучасний інтерфейс	+	–	–
Опрацювання курсів	+	+	+
Проходження тестів	+	+	+
Можливість отримати оцінку за завдання	+	–	–
Приватна переписка з будь-яким учасником курсу	+	–	–
Відслідковування свого прогресу навчання за календарем	–	–	–
Дошки з корисними посиланнями	–	–	+

Згідно з таблицею 1, додатки Udeу та Sololearn не надають можливості отримати оцінку за виконані завдання, що може значно знизити мотивацію для проходження уроків та виконання завдань. Користувач не зможе дізнатися, чи правильно він виконав завдання та чи засвоїв матеріал. Відсутність можливості приватного спілкування та обговорень також може вплинути на процес навчання, оскільки у разі виникнення запитань користувач не зможе отримати відповіді.

Дивно, але жоден з аналізованих додатків не дозволяє відслідковувати свій прогрес у зручному форматі, де можна бачити всі оцінки за завдання та тести в одному місці. Важливо бачити, коли було виконано кожне завдання і в який день, оскільки саме постійність і систематичність є запорукою успіху. Пройти курс за один день — це малоефективно. Найкраще виділяти кілька годин на день і стабільно виконувати завдання, крок за кроком. Використання календаря для відслідковування прогресу може допомогти в цьому процесі.

У додатках Google Classroom та Udemu відсутня функція створення інтерактивних дошок, що не є критичним недоліком, але позбавляє користувачів додаткової мотивації для збереження матеріалів після виконання завдань чи завершення курсів. У Google Classroom можна створювати форми та окремі рубрики для завантаження матеріалів, але ці функції схожі на функціонал для завдань і тестів, що обмежує їх можливості.

2. Вимоги до мобільного додатку

Після того, як обрані додатки проаналізовані, можна виділити основні вимоги користувачів, тобто потенційно той мінімальний функціонал, котрий необхідний для навчання та взаємодії користувача з додатком. Важливо пам'ятати, що вимоги можуть змінюватися протягом проекту, тому важливо мати систему моніторингу та управління змінами [7].

Вимоги користувачів до додатку можуть включати в себе вимоги до мінімального функціоналу додатку, а саме:

- інтуїтивно зрозумілий та сучасний інтерфейс. Додаток повинен мати простий та естетично приємний інтерфейс, який буде легким для освоєння та використання;

- доступ до курсів. Користувач отримає можливість вивчати курс з автомобільного дизайну, що охоплює всі ключові аспекти цієї професії, а також виконувати практичні завдання та переглядати лекційні матеріали з подальшою перевіркою;

- тестування знань. Функція тестування для оцінки розуміння матеріалу та визначення рівня знань користувача;

- оцінка за виконані завдання. Вбудована система оцінювання, яка дає об'єктивні та зрозумілі відгуки щодо виконаних завдань;

- приватні повідомлення. Можливість вести приватне спілкування з іншими учасниками курсу, задавати питання та обговорювати навчальний матеріал;

- моніторинг навчального прогресу. Функція, яка дозволяє відслідковувати виконані завдання, тестування, а також оцінки і загальний прогрес у навчанні;

- дошки з корисними ресурсами. Користувачі можуть отримати доступ до додаткових матеріалів, корисних посилань та ресурсів, що допомагають розширити їхні знання та навички в автомобільному дизайні. Кожен учасник може створювати необмежену кількість таких дошок.

Цей набір вимог є основним для зручного онлайн-навчання. Користувач прагне працювати з додатком, який має сучасний дизайн і простий у використанні. Найефективніший підхід до навчання — це створення курсу, який поетапно надає матеріал у форматі лекцій, доповнених ілюстраціями для кращого розуміння та запам'ятовування.

3. Розробка макета інтерфейсу користувача

Макет, створений у сервісі Figma [8], значно полегшить програмну реалізацію інтерфейсу та функціоналу додатку. Одразу можна промалювати зовнішній вигляд додатку, включаючи іконки, форми, екрани, дрібні деталі, які є великою складовою загального дизайну. Побудова макета у Figma відіграє ключове значення у розробці, оскільки це перший крок у визначенні вигляду та функціональності майбутнього цифрового продукту. Макет дозволяє візуалізувати концепції та ідеї продукту. Перед початком проектуванням програмного продукту необхідно створити макет, який відображає загальну структуру і компоненти інтерфейсу користувача, адже без цього додаток може вийти неструктурований.

Розробник повинен створити такий інтерфейс, який сприятиме легкому сприйняттю інформації, а не ускладнюватиме її. Людина завжди відчуває певний стрес, коли стикається з чимось новим, і додаток не є винятком. Тому важливо, щоб дизайнер або розробник допомогли користувачу подолати ці труднощі. Структура інтерфейсу має бути організована таким чином, щоб поступово направляти увагу користувача і змінювати фокус на важливі моменти.

Щодо зовнішнього вигляду інтерфейсу (User Interface), необхідно звернути увагу на ключові елементи, які формують загальне враження — це кольори, шрифти та іконки. Було вибрано кольори синій та помаранчевий. Вони гармонійно поєднуються та допомагають в майбутньому підвищити впізнаваність додатку.

Можна виділити основні правила, який слід притримуватись, щоб додатком було комфортно користуватись. Ось деякі з них [9,10,11]:

1. Продумувати навігацію всередині додатку та архітектуру. Важливо одразу продумати всі сценарії використання додатку та використання функцій всередині нього. Після чого потрібно відмалювати всі можливі екрани, якщо користувач натиснув на якусь кнопку або, якщо сталась помилка введення даних у поля і т.д. Це допоможе передбачити можливі помилки усередині додатку при проектуванні.

2. Функціональний мінімалізм. Функціональний мінімалізм у UX дизайні — це підхід, який ставить перед собою завдання спростити і спрощувати користувацький досвід, залишаючи лише необхідні функції та елементи. Основна ідея полягає в тому, щоб уникнути зайвої складності та забезпечити максимальну зрозумілість та легкість взаємодії з продуктом для користувачів.

Функціональний мінімалізм може включати в себе такі аспекти: відсів надмірних або малопотрібних функцій; використання простих форм, кольорів і шрифтів; мінімальна кількість кроків; інтуїтивна навігація (рис. 2).



Рис 2. Зовнішній вигляд панелі навігації мобільного додатку і вигляд сторінки з лекцією

4. Програмна реалізація мобільного додатку

4.1. Вибір мови програмування

Вибір мови програмування для Android-розробки залежить від кількох факторів, таких як особисті вподобання розробника, досвід, вимоги проекту тощо. Для розробки додатків на платформі Android основними мовами програмування є Java та Kotlin. Протягом багатьох років Java була основною мовою для створення Android-додатків, і вона досі залишається популярною серед розробників. Однак Kotlin, який Google офіційно підтримує як альтернативу Java, стає все більш популярним завдяки своїм сучасним можливостям, безпеці та іншим перевагам. Обидві мови дозволяють створювати ефективні та високоякісні додатки для Android.

Оскільки Java має довгу історію та є більш популярною мовою, вона володіє великим співтовариством розробників. Це забезпечує наявність численних ресурсів, документації та підтримки, що може значно спростити процес навчання та вирішення технічних проблем. Крім того, Java отримує офіційну підтримку від Google.

4.2. Вибір середі розробки

Для розробки Android-додатків основним інструментом є Android Studio [12] — інтегроване середовище розробки (IDE), створене спеціально для цієї платформи. Android Studio містить безліч корисних інструментів для розробки, тестування, налагодження та оптимізації додатків. Серед можливостей — емулятори пристроїв, візуальні редактори для створення інтерфейсів та інструменти для аналізу продуктивності.

4.3. Вибір бази даних

У якості бази даних було обрано Firebase [13] яка надає набір інструментів та сервісів для розробки мобільних та вебдодатків, а також сервіси для зберігання структурованих даних в хмарі: користувацьких даних, фотографій, відео, документів та іншої інформації. Також Firebase надає сервіси для аутентифікації користувачів через різні методи, такі як електронна пошта, соціальні мережі, номер телефону і т.д.

4.4. Автентифікація

Автентифікація — це процес перевірки ідентичності користувача для надання доступу до функцій, ресурсів або даних додатку. Автентифікація забезпечує безпеку і захист конфіденційності даних, а також дозволяє додатку відслідковувати активності користувачів та надавати персоналізований досвід.

Клас AuthActivity (рис. 3) успадковує (extends) функціональність класу AppCompatActivity. AppCompatActivity — це базовий клас, який надає підтримку для роботи зі старими версіями платформи Android [14]

У методі onCreate() описується сценарій при натисканні або кнопки для реєстрації («RegistrationActivity.class»), або кнопки для входу («LoginActivity.class»). Для того, щоб відслідкувати, на яку кнопку натиснув користувач, на кожну з них вішається прослуховувач setOnClickListener() і в методі startActivity() описується подія «Intent» натиснутої кнопки.

```

AuthActivity.java activity_auth.xml
1 package com.yulia.shcherbinina.drive.auth;
2
3 import ...
4
5
6
7
8
9
10 public class AuthActivity extends AppCompatActivity {
11
12     @Override
13     protected void onCreate(Bundle savedInstanceState) {
14         super.onCreate(savedInstanceState);
15         setContentView(R.layout.activity_auth);
16
17         findViewById(R.id.registration).setOnClickListener((v) -> {
18             startActivity(RegistrationActivity.class);
19         });
20
21         findViewById(R.id.login).setOnClickListener((v) -> {
22             startActivity(LoginActivity.class);
23         });
24     }
25
26     2 usages
27     private void startActivity(Class<?> cls) {
28         startActivity(new Intent(packageContext, this, cls));
29     }

```

Рис. 3. Кодова реалізація автентифікації користувача

4.5. Основна навігація у додатку

У класі «BaseActivity» додана загальна логіка для всіх «активіті» головної сторінки, а тобто: для користувачів, календаря, вкладки мої курси, чатів, ресурсів, щоб перемикались між ними.

У методі attachNavigation() описується процес перемикання між вкладками головного меню додатку.

У залежності від натискання на певну кнопку, записується клас, який треба створити і стартує метод `startActivity()`, у якому відбувається перевірка, чи відкрита «активіті» на даний момент. Дії перемикання між «активіті» відбуваються за допомогою анімації, щоб перехід між вкладками був плавний.

4.6. Процес створення курсу у додатку

Клас «`ModulesActivity`» відображає загальне вікно додатку під назвою «Мої курси» (рис. 4). Цей клас успадковується від класу «`BaseActivity`» і також клас «`ModulesActivity`» виконує функції, які визначені у класі «`OnCourseClickListener`».

```

1 package com.yulia.shcherbinina.drive.modules;
2
3 import ...
4
5 public class ModulesActivity extends BaseActivity
6     implements OnCourseClickListener{
7
8     @Override
9     protected void onCreate(Bundle savedInstanceState) {
10         super.onCreate(savedInstanceState);
11         setContentView(R.layout.activity_modules);
12         attachNavigation(R.id.modules);
13
14         ((App)getApplication()).setCurrentUserName();
15
16         RecyclerView recyclerView = findViewById(R.id.coursesRecyclerView);
17         List<Course> courses = new ArrayList<>();
18         recyclerView.setHasFixedSize(true);
19         courses.add(new Course("Дизайн автомобілів"));
20         CourseListAdapter courseListAdapter = new CourseListAdapter(
21             courses, onCourseClickListener: this);
22         recyclerView.setAdapter(courseListAdapter);
23     }
24 }
    
```

Рис. 4. Кодова реалізація вікна додатку «Мої курси»

За допомогою класу «`App`» викликається метод `setCurrentUserName()`, для того, щоб у курсі, коли це необхідно, фіксувалось ім'я користувача. Це відбувається у методі `setCurrentUserName()`. Якщо користувач раніше не заходив у даний курс, то з репозиторію завантажується користувач за «айді». У методі `getCurrentUserId()` відбувається схожа логіка: якщо користувача не існує, з бази даних за допомогою об'єкта «`auth`» витягується «айді». Коли «айді» завантажено, то у методі `setCurrentUserName()` береться його ім'я та присвоюється полю «`currentUserName`». Таким чином достається з бази даних користувач.

Курс визначений за допомогою прокручуваного списку «`RecyclerView`» [15] на випадок, якщо курс буде займати більше, ніж частину екрану. За допомогою об'єкта «`courses`» викликана функція «`add`», яка додає назву курсу, тобто курс «Дизайн автомобілів», структура якого визначена у класі «`Course`». Всередині нього визначається ім'я курсу, «гетери» та «сетери».

Усі модулі, у тому числі і лекції містяться у класі «`LocalCourse`». У цьому класі є статичний повний курс «`CourseFull`» і статичний модульний об'єкт «`ModuleObject`».

4.7. Вікно користувачів додатку

Клас «`UsersActivity`» містить логіку відображення всіх користувачів, які зареєстровані в додатку і також мають доступ до базового курсу з вивчення дизайну автомобілів (рис. 5). Цей клас містить змінні, які відповідають за кількість користувачів, пошук, прокручуваний список користувачів, адаптер для цього списку та лист користувачів.

```

1 package com.yulia.shcherbinina.drive.users;
2
3 import ...
4
5 public class UsersActivity extends BaseActivity implements UsersAdapter.OnUserClickListener
6
7 {
8     2 usages
9     private TextView usersCount;
10    2 usages
11    private AppCompatActivity inputSearch;
12    3 usages
13    private RecyclerView usersRecyclerView;
14
15    5 usages
16    private UsersAdapter usersAdapter;
17    4 usages
18    private List<User> users = new ArrayList<>();
19
20    @Override
21    protected void onCreate(Bundle savedInstanceState) {
22        super.onCreate(savedInstanceState);
23        setContentView(R.layout.activity_users);
24        attachNavigation(R.id.users);
25
26        initView();
27
28        usersAdapter = new UsersAdapter(users, onUserClickListener: this);
29        usersRecyclerView.setAdapter(usersAdapter);
30        usersRecyclerView.setHasFixedSize(true);
31    }
32 }
    
```

Рис. 5. Кодова реалізація вікна користувачів

У методі `onChanged()` проходить перевірка, чи введено щось у поле пошуку (довжина строки 0), встановлюється повний список, а якщо є зміни, створюється новий список, який фільтрується методом `filteredUsers()`. У цьому методі система проходить по всім користувачам, якщо користувач ввів ім'я, воно додається в список і повертається оновлений метод `filteredUsers()`.

4.8. Приватні чати з користувачами

Клас «`ChatsActivity`» (рис. 6) описує реалізацію електронної взаємодії двох користувачів всередині додатку за допомогою приватних чатів. Цей клас успадковує клас «`BaseActivity`» та виконує функції, які визначені в класі «`ChatsAdapter`».

```

1 package com.yulia.shcherbinina.drive.chats;
2
3 import ...
4
5 public class ChatsActivity extends BaseActivity implements ChatsAdapter.OnChatClickListener
6
7 {
8     1 usage
9     private final FirebaseDatabase databaseReference = FirebaseDatabase.getInstance();
10    2 usages
11    private final DatabaseReference lastChatsReference = databaseReference.getReference("paths");
12    3 usages
13    private String currentUserId;
14    3 usages
15    private RecyclerView lastChatsRecyclerView;
16
17    5 usages
18    private ChatsAdapter chatsAdapter;
19
20    2 usages
21    private AppCompatActivity inputSearch;
22
23    8 usages
24    private List<ChatShot> chats = new ArrayList<>();
25
26    2 usages
27    private TextView countChats;
28 }
    
```

Рис. 6. Кодова реалізація вікна з приватними переписками користувачів

4.9. Налаштування додатку

Клас «SettingsActivity» відповідає за редагування даних користувача та виходу з облікового запису (рис. 7). Цей клас містить текстові поля з ім'ям користувача, пароля, який шифрується зірочками задля безпеки та тексту про вихід зі сторінки користувача.

```

1 package com.yulia.shcherbinina.drive.settings;
2
3 import ...
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35 public class SettingsActivity extends AppCompatActivity {
36
37     2 usages
38     public static final String RE_AUTH_KEY = "reAuth";
39
40     5 usages
41     private TextView userName, passwordStars, exitText;
42
43     6 usages
44     private AppCompatActivity inputName, inputPass;
45
46
47
48
49     4 usages
50     private ImageView editNameImage, editPassImage, exitImage;
51
52
53
54
55     5 usages
56     private boolean isEditingName = false;
57
58     7 usages
59     private boolean isEditingPass = false;
60
61
62
63
64     2 usages
65     private boolean isReauthStart = false;
66
67     1 usage
68     private ProgressBar progressBarEditName, progressBarEditPass, progressBar;
69
70
71
72
73     3 usages
74     private Repository repository;
75
76     3 usages
77     private FirebaseAuth auth;
78
79
80

```

Рис. 7. Кодова реалізація вікна з налаштуваннями мобільного додатку

Висновки

Було створено освітню Android-платформу для вивчення автомобільного дизайну, що включає інтуїтивно зрозумілий та сучасний інтерфейс, можливість проходити курси, тестувати свої знання, отримувати оцінки за виконані завдання, вести приватні переписки з іншими учасниками курсу, а також відслідковувати свій прогрес через календар та створювати дошки з корисними матеріалами. Ці функції надають користувачам можливість ефективно освоювати нові знання та розширювати свої навички.

У роботі також розглянуто перспективи кар'єри автомобільного дизайнера та необхідні для цього знання і вміння. Приведені дослідження підтверджують, що мобільні додатки та дистанційне навчання мають позитивний ефект на освітній процес. Дизайнер автомобілів — це креативна та складна професія, що вимагає знань у галузі інженерії та володіння програмними засобами для візуалізації. Було виявлено, що в Україні наразі відсутні спеціалізовані освітні платформи для навчання автомобільного дизайну.

Основна цінність цього додатку полягає в тому, що він дає змогу користувачам здобути базові знання в галузі автомобільного дизайну або розширити свої уявлення про цю сферу. У порівнянні з іншими додатками, цей продукт має важливі функціональні можливості, які дозволяють задовольнити потреби користувачів.

На основі макету, розробленого в Figma, додаток був програмно реалізований з використанням Java та

Android Studio. Такий додаток може бути корисним у навчальних закладах, оскільки він заповнює вакуум на ринку освітніх додатків, надаючи можливість вивчати автомобільний дизайн, чого наразі неможливо досягти за допомогою інших мобільних платформ в Україні.

Список літератури

- [1] Mobile Learning—Trends and Practices. MDPI. URL: <https://www.mdpi.com/2227-7102/9/1/33#B1-education-09-00033> (дата звернення: 27.04.2024).
- [2] The Effectiveness of mobile learning media using Smart Apps Creator on Economic subjects – DOAJ. Directory of Open Access Journals – DOAJ. URL: <https://doaj.org/article/02cc851e419b436ca1e1ecf51b74a160> (дата звернення: 27.11.2024).
- [3] Найкращі мобільні додатки для студента | GOSTUDY URL: <https://www.gostudy.cz/uk/blog/soveti-eksperta/mobilnye-prilozheniya-dlya-studentov> (дата звернення: 27.11.2024).
- [4] Методи досліджень у дизайні | DOU: URL: <https://dou.ua/lenta/articles/design-research-methods/> (дата звернення: 27.11.2024).
- [5] Business Model Canvas. Download the Official Template. Corporate Innovation Strategy, Tools & Training | Strategyzer. URL: <https://www.strategyzer.com/library/the-business-model-canvas> (дата звернення: 27.11.2024).
- [6] Value Proposition: як створити ціннісну пропозицію сервісним та продуктовим ІТ-компаніям (гайд із прикладами) | Companera_Agency. Companera Agency: inbound & outbound marketing. URL: <https://companera.com.ua/blog/value-proposition-yak-stvoriti-cinnisnu-propoziciyu-servisnim-ta-produktovim-it-kompaniyam-gajd-iz-prikladami> (дата звернення: 27.11.2024).
- [7] Визначення вимог: що це таке і як це застосовувати? | Повний посібник - Рішення Visure. Visure Solutions. URL: <https://visuresolutions.com/uk/блог/визначення-вимог/> (дата звернення: 27.11.2024).
- [8] Figma: що це таке, для чого і кому потрібна – GoIT Global. GoIT Global. URL: <https://goit.global.ua/articles/figma-shcho-tse-take-dlia-choho-i-komu-potribna/> (дата звернення: 27.11.2024).
- [9] 7 основних помилок в UX-дизайні мобільних додатків | Продизайн. Продизайн. URL: <https://prodesign.in.ua/2016/05/7-osnovnyh-pomylok-v-ux-dyzajni-mobilnyh-dodatkov/> (дата звернення: 27.11.2024).
- [10] Покращуємо UX/UI мобільного додатка | Продизайн. Продизайн. URL: <https://prodesign.in.ua/2021/03/pokrashhuemo-ux-ui-mobilnogo-dodatka/> (дата звернення: 27.11.2024).
- [11] Styles – Material Design 3. Material Design. URL: <https://m3.material.io/styles> (дата звернення: 27.11.2024).
- [12] Developer guides | Android Developers. Android Developers. URL: <https://developer.android.com/guide> (date of access: 21.11.2024).
- [13] Firebase як бекенд для будь-яких застосунків, та як використовувати Firebase-сервіси | DOU: URL: <https://dou.ua/forums/topic/44058/> (дата звернення: 27.11.2024).
- [14] AppCompatActivity | Android Developers. Android Developers. URL: <https://developer.android.com/reference/androidx/appcompat/app/AppCompatActivity> (дата звернення: 27.11.2024).
- [15] Using the RecyclerView | CodePath Android Cliffnotes. CodePath Cliffnotes. URL: <https://guides.codepath.com/android/using-the-recyclerview> (дата звернення: 21.11.2024).

Надійшло до редакції 22.10.2024