

**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
СУМСЬКИЙ НАЦІОНАЛЬНИЙ АГРАРНИЙ УНІВЕРСИТЕТ
ФАКУЛЬТЕТ ЕКОНОМІКИ І МЕНЕДЖМЕНТУ**

Кафедра кібернетики та інформатики

КВАЛІФІКАЦІЙНА РОБОТА

освітній ступінь - «Бакалавр»

**на тему: РОЗРОБКА ВЕБЗАСТОСУНКА ДЛЯ
ІНТЕРАКТИВНОГО ВИВЧЕННЯ ДИСЦИПЛІН
ПРИРОДНИЧОГО ЦИКЛУ**

Виконав:

студент спеціальності

126 «Інформаційні системи та технології»

Санжаков Андрій Андрійович

Керівник:

Толбатов Андрій Володимирович

доцент кафедри кібернетики та інформатики

Рецензент:

Яценко Валерій Валерійович

К.т.н., доцент кафедри економічної кібернетики, ННІ бізнесу,
економіки та менеджменту СумДУ

СУМСЬКИЙ НАЦІОНАЛЬНИЙ АГРАРНИЙ УНІВЕРСИТЕТ

Факультет	Економіки і менеджменту
Кафедра	Кібернетики та інформатики
Освітній ступінь	«Бакалавр»
Спеціальність	<u>126 «Інформаційні системи та технології»</u> <u>ОП «Інформаційні системи та технології»</u>

Затверджую:
Завідувач кафедри Світлана АГАДЖАНОВА

« ____ » « ____ » 2024 ____ р.

ЗАВДАННЯ НА КВАЛІФІКАЦІЙНУ РОБОТУ СТУДЕНТУ

Санжакову Андрію Андрійовичу

1. Тема роботи: Розробка вебзастосунка для інтерактивного вивчення дисциплін природничого циклу

2. База практичного дослідження: Сумський НАУ.

керівник роботи Толбатов Андрій Володимирович к.т.н., доцент

затверджена наказом по університету від « ____ » 2024 р. № ____ -н

3. Строк подання студентом закінченого проекту (роботи)

«28» червня 2024 року

4. Вихідні дані до проекту (роботи):

Результати аналітичних досліджень зарубіжних та вітчизняних вчених, інтернет-джерела, нормативні документи за чинним законодавством.

5. Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити):

- Виконати постановку задачі;
- Вибрати середовище для реалізації;
- розробити технічне завдання;
- розробити програмний продукт відповідно до поставлених задач;
- розробити інструкцію для користувача;

6. Дата видачі завдання:

« ____ »

2024 р.

\

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів дипломного проєкту (роботи)	Строк виконання етапів проєкту	Примітка
1	Визначення об'єкта, предмету дослідження, формулювання мети та задач кваліфікаційної роботи, складання плану	січень 2024 р.	
2	Підбір та вивчення літературних джерел, технічної документації	січень 2024 р.	
3	Розробка технічного завдання	січень 2024 р..	
4	Розробка мобільного додатку	січень-квітень 2024 р.	
5	Створення інструкції користування вебдодатком для користувачів	до 10 травня 2024 р.	
6	Оформлення пояснювальної записки	до 30 травня 2024р..	
7	Оформлення презентації	до 2 червня 2024 р.	
8	Опрацювання зауважень керівника, перевірка на автентичність	3 червня 2024 р.	
9	Підготовка доповіді та її попередній захист	5 червня 2024 р.	
10	Перевірка з академічної доброчесності кваліфікаційної роботи	10 червня 2024 р.	
11	Представлення кваліфікаційної роботи на кафедрі, деканат	23 червня 2024 р.	
12	Захист кваліфікаційної роботи	28 червня 2024 р.	

Студент

(підпис)

Андрій САНЖАКОВ

Керівник роботи

(підпис)

Андрій ТОЛБАТОВ

Перевірку на автентичність проведено

(підпис)

Надія БАРАНІК

Дипломна робота допущена до захисту

(підпис)

Маргарита ЛИШЕНКО

АНОТАЦІЯ

Санжаков А.А. Розробка вебзастосунка для інтерактивного вивчення дисциплін природничого циклу

Кваліфікаційна робота зі спеціальності 126 «Інформаційні системи та технології» ОП Інформаційні системи та технології, СНАУ, Суми-2024р. – Рукопис.

Ця робота присвячена розробці мобільного додатку для інтерактивно вивчення природничих дисциплін на базі вебзастосунка «Classroom».

У роботі проведений аналіз сучасних технологій освітнього процесу, виявлено недосконалість освітнього процесу для учнів, а саме інтерактивність та доступ до знань 24/7, підхід до вивчення природничих дисциплін. Запропоновано інтерактивну модель для вивчення природничих дисциплін, а саме інтерактивність та доступність.

Для розробки системи використані сучасні технології програмування та баз даних, що забезпечують швидке та надійне функціонування. Реалізовано інтерактивність, доступність та кросплатформеність, базу даних для мобільного додатку

Проведено тестування застосунків. Результати тестування підтвердили правильність функціонування підсистеми та її відповідність вимогам та очікуванням користувачів.

Отримані результати дозволяють зробити висновок про доцільність впровадження розроблених застосунків інтерактивних застосунків для вивчення природничих дисциплін.

Ключові слова: Kotlin, Android, Room, Android Studio, LiteSQL, база даних, вебдодаток, мобільний додаток

SUMMARY

Sanzhakov A.A. Development of a Web Application for Interactive Study of Natural Science Disciplines

Qualification work in the specialty 126 "Information Systems and Technologies" OP Information Systems and Technologies, SNAU, Sumy-2024. - Manuscript.

This work is dedicated to the development of a mobile application for interactive study of natural science disciplines based on the web application "Classroom".

The work analyzes modern technologies of the educational process, identifies the shortcomings of the educational process for students, namely interactivity and access to knowledge 24/7, and the approach to studying natural science disciplines.

An interactive model for studying natural science disciplines, namely interactivity and accessibility, has been proposed.

Modern programming and database technologies were used to develop the system, ensuring fast and reliable operation. Interactivity, accessibility, and cross-platform compatibility have been implemented, as well as a database for the mobile application.

Application testing was conducted. The test results confirmed the correctness of the subsystem operation and its compliance with the users' requirements and expectations.

The obtained results allow us to conclude the feasibility of implementing the developed interactive applications for studying natural science disciplines.

Keywords: Kotlin, Android, Room, WordPress, PHP, HTML, JS, database, SQL, web application, mobile application.

ЗМІСТ

ВСТУП	7
РОЗДІЛ 1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ	8
1.1 Огляд останніх досліджень і публікацій	8
1.2 Аналіз веб та мобільних застосунків	10
РОЗДІЛ 2 ПОСТАНОВКА ЗАДАЧІ ТА ВИБІР ЗАСОБІВ РЕАЛІЗАЦІЇ	16
2.1 Постановка задачі та визначення мети	16
2.2 Вибір засобів реалізації	18
РОЗДІЛ 3 ПРОЄКТУВАННЯ ТА РЕАЛІЗАЦІЯ МОБІЛЬНОГО ЗАСТОСУНКА ІЗ ТЕМ ПРИРОДНИЧОГО ЦИКЛУ	22
3.1 Розробка функціональних діаграм мобільного застосунку із дисциплін природничого циклу	22
3.2 Проєктування бази даних мобільного застосунку із дисциплін природничого циклу	26
3.3 Реалізація інтерфейсу користувача	31
3.4 Фізична реалізація бази даних	38
ВИСНОВКИ	41
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ	42
ДОДАТКИ	43
Додаток А Технічне завдання	45
Додаток Б Планування робіт	51
Додаток В Копії публікацій	54
Додаток Г Лістинги програм	61

ВСТУП

Сучасні освітні тенденції роблять акцент на інтерактивності та залученості учнів у процес навчання. Використання вебзастосунків та мобільних додатків відкриває нові можливості для вивчення дисциплін природничого циклу, роблячи його цікавішим, доступнішим:

1. Інтерактивні елементи, візуалізації та ігри роблять процес навчання більш захопливим та мотиваційним для учнів.
2. Інтерактивні методи навчання дозволяють учням краще засвоювати матеріал, а також розвивають їхнє критичне мислення та навички розв'язання проблем.
3. Вебзастосунки та мобільні додатки доступні 24/7, що дозволяє учням вчитися в будь-який час і в будь-якому місці.

Це робить тему розробки вебзастосунка для вивчення природничого циклу актуальною та перспективною.

Об'єкт дослідження: система дистанційного навчання.

Предмет дослідження: інтерактивне вивчення дисциплін природничого циклу.

Мета роботи: розробити Android-додаток, на базі «Classroom», який допоможе учням в інтерактивній та цікавій формі вивчати природознавство, роблячи процес навчання більш ефективним та захоплюючим.

Відповідно до поставленої мети сформульовані такі задачі:

1. Виконати огляд останніх досліджень і публікацій з предметної області «Дисципліни природничого циклу».
2. Провести аналіз наявних вебзастосунків та Android-додатків для вивчення природознавства.
3. Сформулювати постановку задачі та вибрати засоби її реалізації
4. Розробити структуру та дизайн Android-додатку.
5. Здійснити фізичну реалізацію бази даних

РОЗДІЛ 1

АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ

1.1 Огляд останніх досліджень і публікацій

Дисципліни природничого циклу є формою збору статистичних даних, яка проводиться у рамках навчання природничих наук з метою обліку майна, дослідження природних явищ, експериментів та інших аспектів цих дисциплін. Природничі записи та журнали є документами адміністративного обліку. Вони містять інформацію про результати досліджень, проведених учнями та викладачами, експериментальні дані, висновки та аналіз природних явищ. Відомості з цих документів можуть використовуватися для підготовки звітів, статей, наукових робіт та інших наукових публікацій.

Огляд останніх досліджень і публікацій з теми дисциплін природничого циклу показує активний розвиток цієї області та постійне поповнення науковими відкриттями та новими ідеями. Дослідники, університети та організації вкладають значні зусилля у розуміння та вдосконалення побутового обліку, що сприяє ефективному веденню навчального процесу та наукової роботи.

Одним з напрямків досліджень є використання інформаційних технологій в дисциплінах природничого циклу. Розробка та впровадження програмного забезпечення, веб-платформ та інших інструментів для автоматизації процесів досліджень, аналізу даних та взаємодії учасників навчального процесу є актуальною задачею. Досліджуються нові технології, такі як віртуальна реальність, штучний інтелект та обробка великих даних, для полегшення введення та обробки даних, покращення якості навчання та сприяння інноваціям у сфері природничих наук.

Природничі науки в сучасному світі знаходяться на передньому краї наукового прогресу та технологічного розвитку. Дослідження та публікації у цій області відображають широкий спектр тем, від фундаментальних досліджень в основних наукових дисциплінах до практичних застосувань новітніх технологій.

Однією з ключових областей досліджень є екологія та охорона навколишнього середовища. Дослідники займаються вивченням екосистем, впливом антропогенної діяльності на біорізноманіття та клімат, пошуком стійких технологій та методів збереження природних ресурсів. Публікації у цій області відображають останні досягнення у сфері відновлювальної енергетики, розвитку екологічно чистих технологій та стратегій адаптації до зміни клімату.

Інший важливий напрям досліджень - біологія та генетика. Вчені вивчають структуру та функції живих організмів, механізми еволюції та генетичні основи хвороб. Публікації у цій області охоплюють такі теми, як клонування, трансгенні технології, генно-інженерні та біоінформатичні дослідження.

Окрему увагу заслуговують інформаційні технології в природничих науках. Великі обсяги даних, отриманих в результаті наукових експериментів та спостережень, потребують ефективних методів аналізу та обробки. Дослідження в галузі комп'ютерної біології, обробки сигналів та зображень, а також розвитку спеціалізованого програмного забезпечення для моделювання та прогнозування відкривають нові можливості для розвитку наукових досліджень у природничих науках.

Загальний тренд в останні роки показує зростання інтердисциплінарних досліджень та колаборації між різними галузями природничих наук та інженерії. Це сприяє створенню нових знань, інноваційних рішень та впровадженню технологічних новацій у різні сфери людської діяльності.

Огляд останніх досліджень і публікацій у природничих науках свідчить про постійний розвиток цієї області та великий потенціал для подальшого

наукового прогресу. Дослідження у цих областях мають важливе значення для збереження навколишнього середовища, розвитку нових технологій та покращення якості життя на Землі.

1.2 Аналіз існуючих вебзастосунків та android-додатків для вивчення природознавства.

Сучасні освітні тенденції роблять акцент на інтерактивності та залученості учнів у процес навчання. Використання вебзастосунків разом із андроїд-додатками відкриває нові можливості для вивчення дисциплін природничого циклу, роблячи його:

1 Більш цікавим: інтерактивні елементи, візуалізації та ігри роблять процес навчання захопливим та мотиваційним для учнів.

2 Більш ефективним: інтерактивні методи навчання дозволяють учням краще засвоювати матеріал, а також розвивають їхнє критичне мислення та навички вирішення проблем.

3 Більш доступним: веб та андроїд застосунки доступні 24/7, що дозволяє учням вчитися в будь-який час і в будь-якому місці.

Це робить тему розробки вебзастосунків для вивчення природничого циклу актуальною та перспективною. Наразі існує багато вебзастосунків, які пропонують різні способи вивчення природознавства.

Деякі з найпопулярніших веб та андроїд застосунків:

1. Khan Academy: Пропонує безкоштовні та платні курси (рисунк 1.2) з різних предметів, включаючи природознавство. Курси містять відеолекції, інтерактивні вправи та тести. Головна сторінка вебзастосунка Khan Academy зображено на рисунку 1.1. Мобільний додаток Khan Academy дозволяє зручно користуватися ресурсом на смартфонах і планшетах з операційною системою Android. Він має ті ж функції, що й вебверсія, а також додаткові можливості для зручного навчання в рухливому форматі.

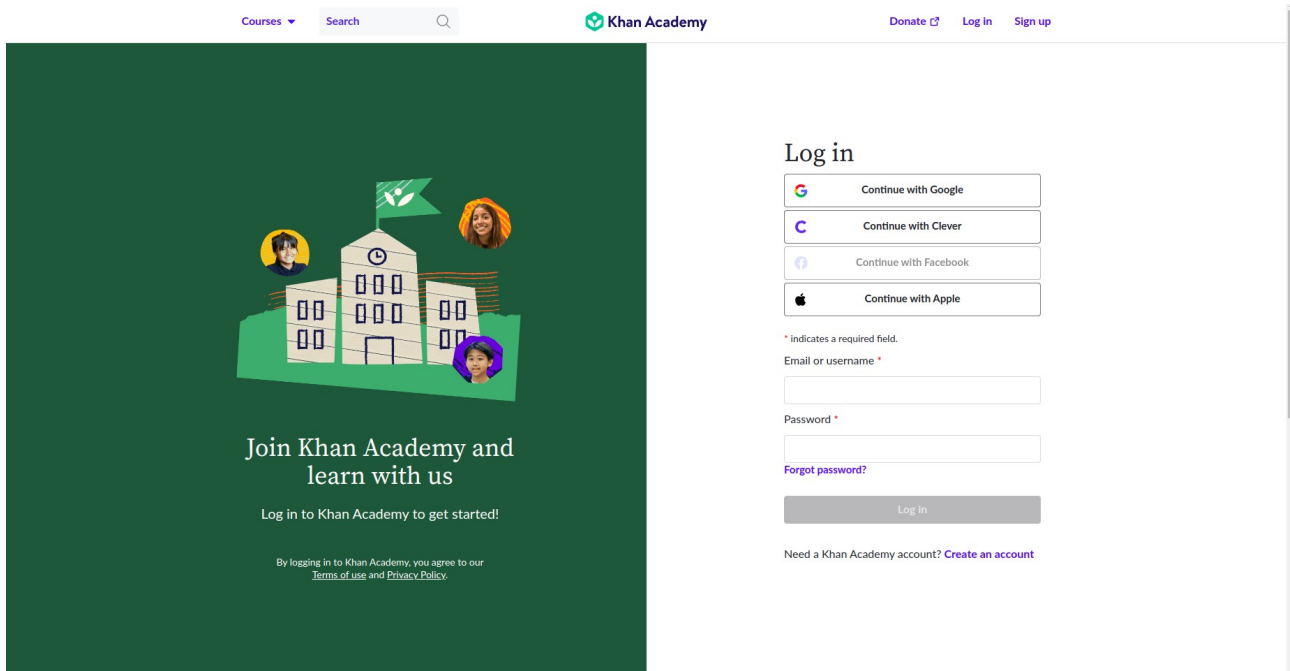


Рисунок 1.1 - Головна сторінка вебзастосунку Khan Academy

Джерело: [1]

Khan Academy[1] включає великий перелік курсів для учнів, сам вебзастосунок інтерактивно орієнтований, мобільна версія нічим не відрізняється від вебверсії. Недоліки Khan Academy:

1. Відсутність української мови.
2. Не всі курсу із традиційної освітньої програми
3. Не всі теми будуть зрозумілі дітям
4. Недостатня інтерактивність із курсу природознавства

Головний недолік цього вебзастосунку це відсутність української мови.

2. National Geographic Kids: Пропонує ігри, вікторини, відео та статті про різні теми з природознавства. Головна сторінка вебзастосунку Khan Academy зображено на рисунку 1.3.

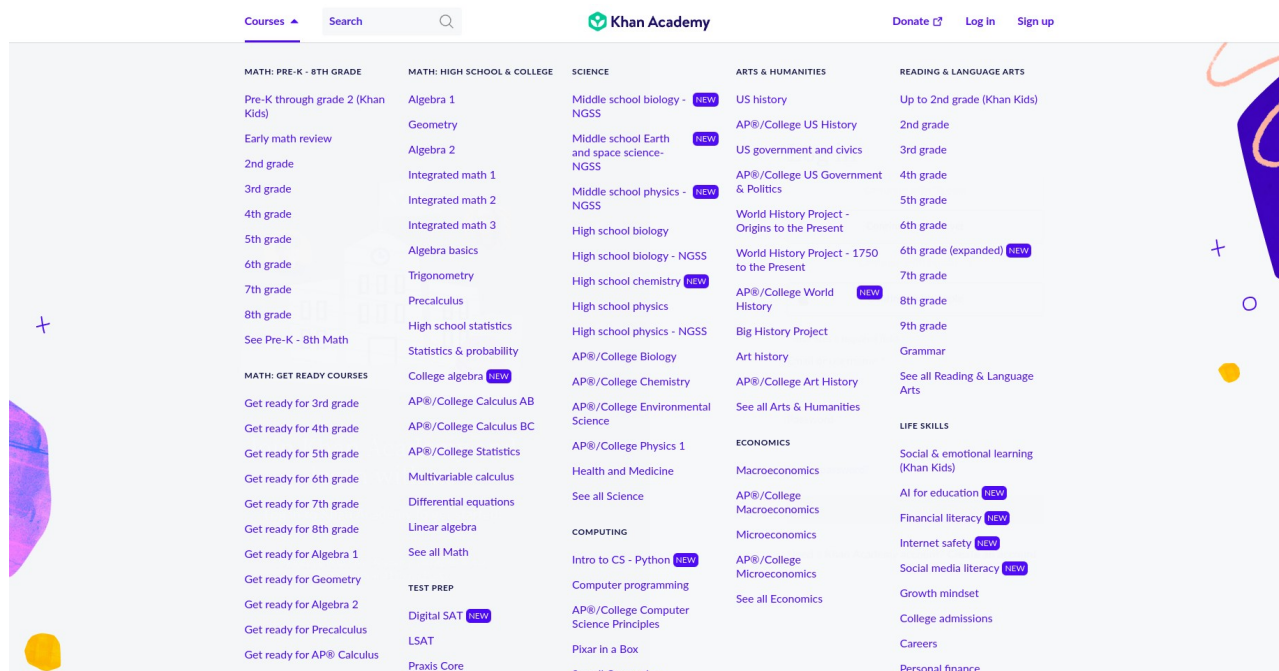


Рисунок 1.2 - Безкоштовні та платні курси Khan Academy

Джерело: [1]

Мобільний додаток National Geographic Kids дозволяє зручно користуватися ресурсом на смартфонах і планшетах з операційною системою Android. Він має ті ж функції, що й вебверсія, а також додаткові можливості для зручного навчання в рухливому форматі.

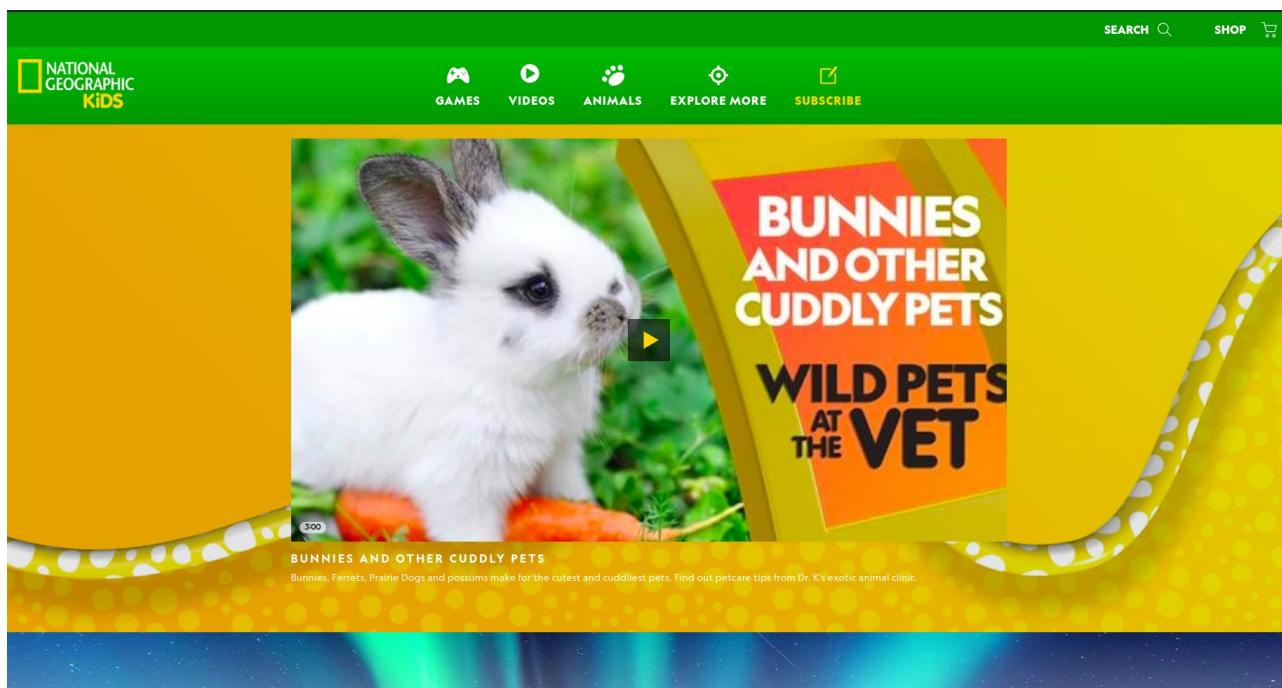


Рисунок 1.3 - Головна сторінка National Geographic Kids

Джерело: [2]

National Geographic Kids[2] складається з ігор, відео та статей про природознавство, є декілько версії веб - сайтів та мобільних додатків. Недоліки National Geographic Kids:

1. Відсутність української мови.
2. Не всі курсу із традиційної освітньої програми
3. Не всі теми будуть зрозумілі дітям
4. Недостатня інтерактивність із курсу природознавства

Головний недолік цього вебзастосунка це відсутність української мови, та недостатня інтерактивність в освітньому процесі.

4 BBC Earth: Цей вебсайт пропонує документальні фільми, фотогалереї та статті про тварин, рослини та природні явища. Головна сторінка вебзастосунка BBC Earth зображено на рисунку 1.4.



Рисунок 1.4 - Головна сторінка веб-застосунку Khan Academy
Джерело: [3]

BBC Earth[3] включає великий перелік документальних фільмів, статей про природознавство для учнів, сам вебзастосунок орієнтований на більш старшу аудиторію хоча і позиціонує себе як дитячий вебзастосунок, відсутня мобільна версія. Недоліки BBC Earth:

1. Відсутність української мови.
2. Не всі курсу із традиційної освітньої програми
3. Не всі теми будуть зрозумілі дітям
4. Недостатня інтерактивність із курсу природознавства

Головний недолік цього вебзастосунку це відсутність української мови.

5. Science Kids[4]: Цей вебсайт пропонує ігри, вікторини, експерименти та статті про різні теми з природознавства. Головна сторінка вебзастосунку Science Kids зображено на рисунку рисунок 1.5.

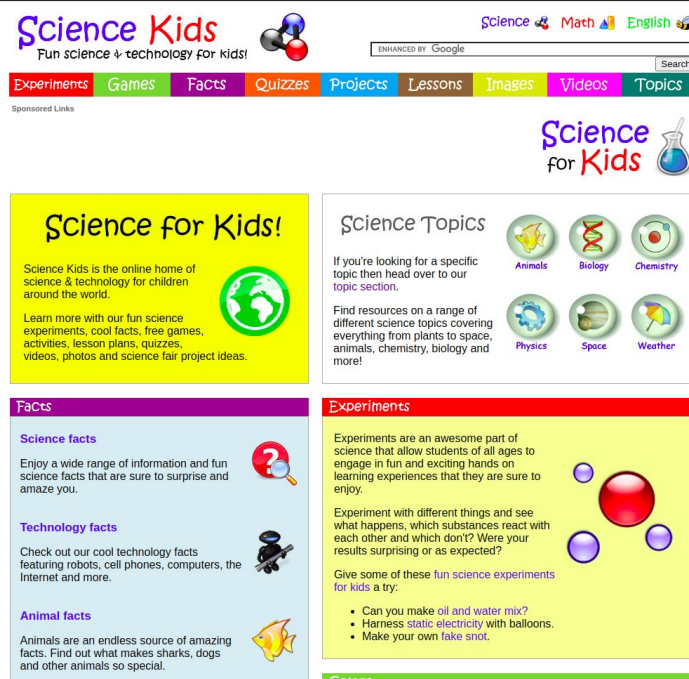


Рисунок 1.5 - Головна сторінка веб-застосунку Science Kids

Джерело: [4]

Science Kids включає великий перелік ігр, статей, відео та експериментів для учнів молодших класів, інтерактивно орієнтований, відсутня мобільна версія.

Недоліки Science Kids:

1. Відсутність української мови.
2. Не всі курсу із традиційної освітньої програми
3. Не всі теми будуть зрозумілі дітям
4. Недостатня інтерактивність із курсу природознавства
5. Стиль оформлення виконан на дуже низькому рівні

Головний недолік цього веб-застосунку це відсутність української мови.

РОЗДІЛ 2

ПОСТАНОВКА ЗАДАЧІ ТА ВИБІР ЗАСОБІВ РЕАЛІЗАЦІЇ

2.1 Постановка задачі та визначення мети

Метою вебзастосунка дисциплін природничого циклу є створення мобільного додатку, який буде присвячений вивченню тем з природознавства.

Цей додаток буде базуватися на вебзастосунку "Classroom", який забезпечить користувачу доступ до андроїд-додатка через спеціальне посилання. Головною метою цього проєкту є забезпечення зручного та ефективного доступу до навчального матеріалу, пов'язаного з природничими науками. Важливими принципами розробки є доступність, інтерактивність, швидкість та зручність використання додатка. Окремо варто виділити доступність, додаток буде мати локальну базу даних, що має на увазі доступність 24/7, без доступу до інтернету в теперішніх умовах це дуже актуально. Слід також додати що, інтерактивність також буде досягатися за допомогою графічної поставної, а саме зображенням різних аспектів природних явищ, включаючи візуалізацію самих цих явищ.

Дані для додатка формуються на основі тем з дисциплін природничого циклу, таких як природознавства, яке досліджує природні явища. Зокрема, додаток буде містити інформацію про природні явища, як вони утворюються, основні елементи, та перелік необхідних знань для безпеки. База даних буде містити перелік природних явищ із теми «Природні явища» із природознавства:

- Смерч
- Землетрус
- Гроза
- Дощ
- Виверження вулкана

- Пилова буря
- Тропічний циклон
- Вогняне торнадо
- Падіння астероїда
- Лісова пожежа
- Сонячне затемнення
- Наднова
- Сонячний спалах
- Лавина каменепад
- Зсув ґрунту
- Цунамі
- Сніг

В базі даних інформація про утворення природних явищ, буде представлено в текстовому форматі, це буде розділ присвячений утворенню цих явищ.

Кожне природне явище буде містити основі складові із яких воно складається. Ці складові будуть складатися із графічного зображення, тексту.

Завдяки графічному зображенню учням простіше буде візуалізувати основні складові природного явища. В базі даних вони також будуть представлені в вигляді тексту та зображення.

Основними функціями додатка будуть:

1. Навчальний матеріал: Представлення інформації з природознавства та у зручній та доступній формі, включаючи тексти та ілюстрації.
2. Взаємодія: Навчальний матеріал буде взаємодіяти із користувачем через мобільний додаток.
3. Зручність використання: Розробка інтерфейсу, який буде зручним для користувачів будь-якого рівня. Підтримка різних функціональностей та можливість швидкого доступу до необхідної інформації.

2.2 Вибір засобів реалізації

Андроїд додаток буде базуватися на вебплатформі «ClassRoom», тому веб частину додатка не треба буде реалізовувати, основа реалізація це андроїд програмування. Опишемо основні характеристики «ClassRoom».

"ClassRoom" - це вебплатформа для управління навчальним процесом, яка надає різноманітні технічні можливості для вчителів та учнів. Основні технічні характеристики цієї платформи включають:

1. Керування курсами та матеріалами: "ClassRoom" дозволяє вчителям створювати курси, завдання та інші навчальні матеріали та керувати ними. Це містить можливість додавати, редагувати та видаляти матеріали, структурувати їх у розділи та модулі, а також керувати доступом учнів до цих матеріалів.

2. Взаємодія через форуми та чати: Платформа надає можливість вчителям та учням спілкуватися через форуми та чати. Це дозволяє обговорювати теми, задавати питання та обмінюватися ідеями, що сприяє зрозумінню матеріалу та підвищує залученість учнів.

3. Оцінювання та звітність: "ClassRoom" дозволяє вчителям створювати та оцінювати завдання, тести та інші види робіт. Також надається можливість генерувати звіти про успішність учнів та аналізувати їхні досягнення.

4 Підтримка мобільних пристроїв: Платформа "ClassRoom" має підтримку мобільних пристроїв, що дозволяє вчителям та учням користуватися не лише на комп'ютерах, а й на смартфонах та планшетах. Це робить доступ до навчального матеріалу більш зручним та доступним.

5. Безпека та конфіденційність даних: "ClassRoom" забезпечує захист особистих даних учасників та конфіденційність навчальної інформації.

Платформа використовує сучасні методи шифрування та захисту даних, щоб забезпечити їх безпеку.

6. Розширені можливості налаштування: Платформа надає вчителям широкі можливості налаштування навчальних курсів та матеріалів, а також

керування доступом учнів до них. Це дозволяє адаптувати платформу до конкретних потреб та вимог навчального закладу.

Тепер перейдемо до мобільної реалізації. Android Studio є основним інтегрованим середовищем розробки (IDE) для створення андроїд-додатків і відоме своєю потужністю та розширюваністю. Ось детальний опис його можливостей:

1. Вбудовані інструменти для розробки та тестування: Android Studio надає широкий набір інструментів для розробки, включаючи візуальні редактори для розмітки та дизайну інтерфейсу користувача (UI), налагоджувач, вбудований емулятор для тестування додатків на різних пристроях, а також підтримку розробки на мові Kotlin.

2. Підтримка Git: Android Studio вбудовує в себе інструменти для роботи з системами контролю версій, зокрема з Git. Це дозволяє розробникам легко керувати версіями свого коду, використовуючи функції, такі як гілки, злиття та інші.

3. Аналізатори коду: Інтегроване середовище містить ряд аналізаторів коду, які допомагають розробникам покращувати якість свого коду. Наприклад, вони можуть виявляти потенційні проблеми в коді, такі як потенційні помилки, неоптимальний код, використання застарілих методів тощо.

4. Підтримка різних мов програмування: Поміж основними мовами програмування, які підтримуються Android Studio, варто відзначити Kotlin, Java та C++. Це дозволяє розробникам вибирати мову програмування, яка найбільше відповідає їх потребам і вподобанням.

5. Розширені можливості налаштування: Android Studio дозволяє розробникам налаштовувати середовище розробки згідно з їхніми потребами. Це включає налаштування різних інструментів, вигляду редактора коду, шрифтів, кольорової схеми та інших параметрів.

6. Підтримка плагінів і розширень: Android Studio має широкий вибір плагінів та розширень, що дозволяють розробникам розширити його

функціональність. Це можуть бути плагіни для автоматизації завдань, розширень для підтримки нових фреймворків чи інструментів, та інше.

Загалом, Android Studio - це потужне інтегроване середовище розробки, яке надає розробникам широкі можливості для створення якісних та ефективних андроїд-додатків.

Основною мовою програмування в інтегрованої середи розробки Android Studio є Kotlin. Kotlin є основною мовою програмування в інтегрованому середовищі розробки Android Studio, і вона надає широкі можливості для створення якісних та ефективних мобільних додатків. Ось деякі ключові переваги та особливості мови Kotlin:

1. Коротка синтаксична конструкція: Kotlin відрізняється чистим та лаконічним синтаксисом, що дозволяє писати менше коду для досягнення тих самих результатів порівняно з Java.
2. Безпека типів: Kotlin пропонує безпечну систему типів, яка допомагає уникнути багатьох типових помилок під час розробки додатків.
3. Нульова безпека: Kotlin вбудовує підтримку для нульової безпеки, що дозволяє уникнути помилок, пов'язаних з нульовими посиланнями.
4. Розширені функціональні можливості: Мова Kotlin підтримує функціональне програмування, включаючи лямбда-вирази, високорівневі функції, онлайн-функції та інші конструкції, що полегшують роботу з колекціями даних.
5. Інтероперабельність з Java: Kotlin повністю сумісний з Java, що дозволяє використовувати чинний Java-код у проєкті Kotlin і навпаки. Це робить перехід на Kotlin більш гладким та безпечним.

Тепер щодо Kotlin в Android Studio. Інтеграція Kotlin в Android Studio дозволяє розробникам швидко та ефективно створювати мобільні додатки для платформи Android. Android Studio надає ряд інструментів та функціональностей для роботи з Kotlin:

1. Автоматичне перетворення коду: Android Studio може автоматично перетворювати Java-код у Kotlin, що полегшує перехід на Kotlin для теперішніх проєктів або бібліотек.

2. Підказки та автодоповнення: Інтеграція Kotlin в Android Studio надає розширені підказки та автодоповнення, що допомагають розробникам швидше та ефективніше писати код.

3. Інструменти для аналізу коду: Android Studio має вбудовані аналізатори коду, які допомагають виявляти та виправляти потенційні проблеми в Kotlin-кодi.

Тепер щодо Compose. Jetpack Compose - це сучасна бібліотека для створення інтерфейсів користувача в Android-додатках з використанням декларативного підходу. Ось деякі особливості та переваги Jetpack Compose:

1. Декларативний підхід: Compose дозволяє описувати інтерфейс користувача за допомогою простих та зрозумілих функцій, що полегшує розробку та розуміння коду.

2. Швидкість розробки: Compose пропонує широкий набір готових компонентів та можливостей для створення різноманітних інтерфейсів, що дозволяє розробникам швидко створювати та модифікувати UI.

3. Інтеграція з Kotlin: Jetpack Compose повністю підтримується мовою Kotlin, що дозволяє використовувати всі переваги Kotlin при розробці інтерфейсів користувача.

4. Інтерактивність та анімація: Compose дозволяє легко додавати інтерактивність та анімацію до інтерфейсу користувача за допомогою вбудованих функцій та ефектів.

Загалом, Jetpack Compose є потужним інструментом для створення сучасних та привабливих інтерфейсів користувача в Android-додатках, а його інтеграція з Kotlin в Android Studio робить процес розробки більш зручним та продуктивним.

РОЗДІЛ 3

ПРОЄКТУВАННЯ ТА РЕАЛІЗАЦІЯ МОБІЛЬНОГО ЗАСТОСУНКА ІЗ ТЕМ ПРИРОДНИЧОГО ЦИКЛУ

3.1 Розробка функціональних діаграм мобільного застосунку із дисциплін природничого циклу

Use Case Diagram - це важлива складова процесу розробки програмного забезпечення, оскільки вона дозволяє розробникам краще розуміти, як саме буде використовуватись їхній продукт. Ця діаграма відображає можливі сценарії використання системи або програмного продукту, що допомагає у розумінні потреб і очікувань користувачів. Перегляд цих сценаріїв допомагає розробникам правильно налаштувати функціонал продукту і забезпечити його відповідність потребам користувачів зображено на рисунку 3.1

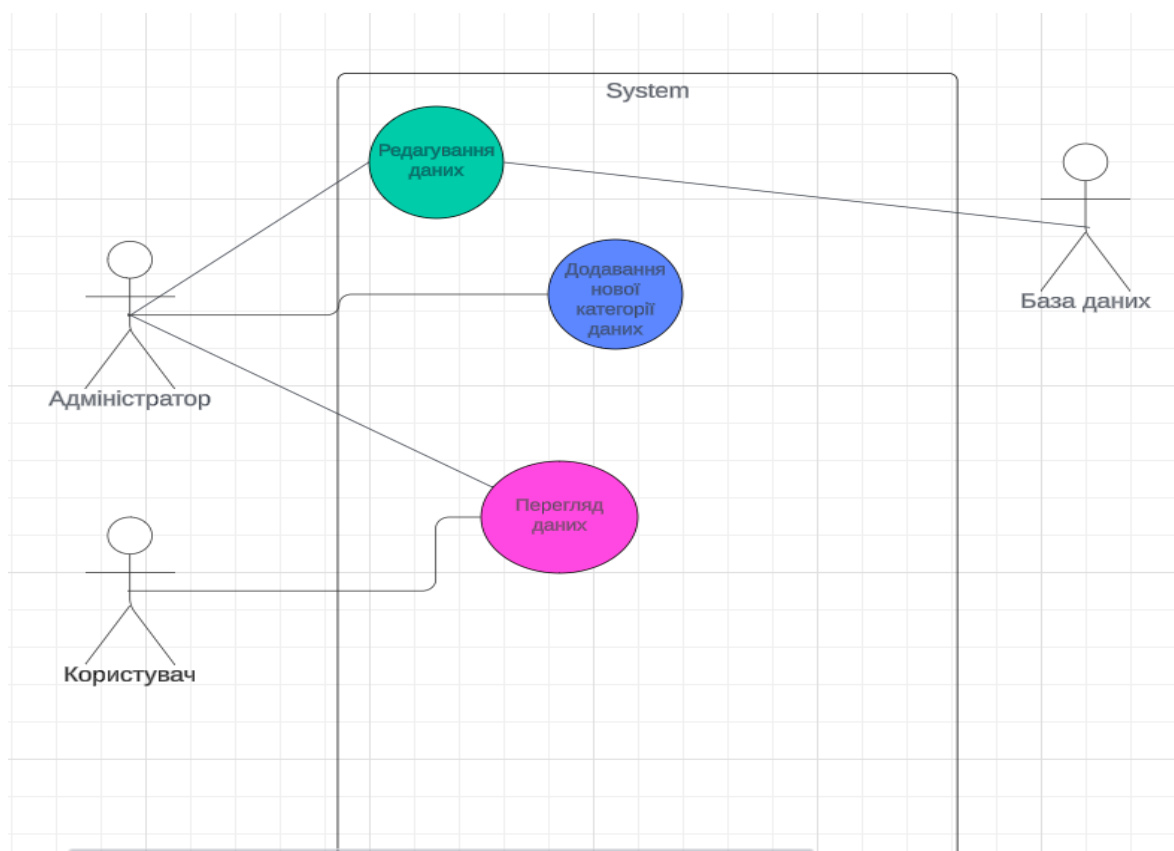


Рисунок 3.1 - Діаграма використання підсистеми

Джерело: запропоновано автором

У цьому контексті актори можуть бути розглянуті таким чином:

- Користувач – учень, студент , який використовує цей додаток.
- Адміністратор - особа, яка має розширений функціонал продукту і відповідає за адміністрування системи.

Функціональні діаграми (також відомих як функціональні блок-схеми або блок-схеми системи) є частиною аналізу та проектування системи. Вони допомагають візуалізувати функціональність системи та взаємодію її компонентів. Контекстна модель мобільного додатку природничі дисципліни зображений на рисунку 3.2

Вхідні дані мобільного додатку природні явища включають:

- Дані із тем природознавство.

Вихідними даними є:

- Змінена база даних;
- Автоматизоване введення курсу із тем природничих дисциплін.

Керівні об'єкти включають:

- Інструкція використання додатку;
- Вхідну базу даних.

Механізми включають:

- Адміністратора;
- Кінцевий користувач;
- Програмне і технічне забезпечення.

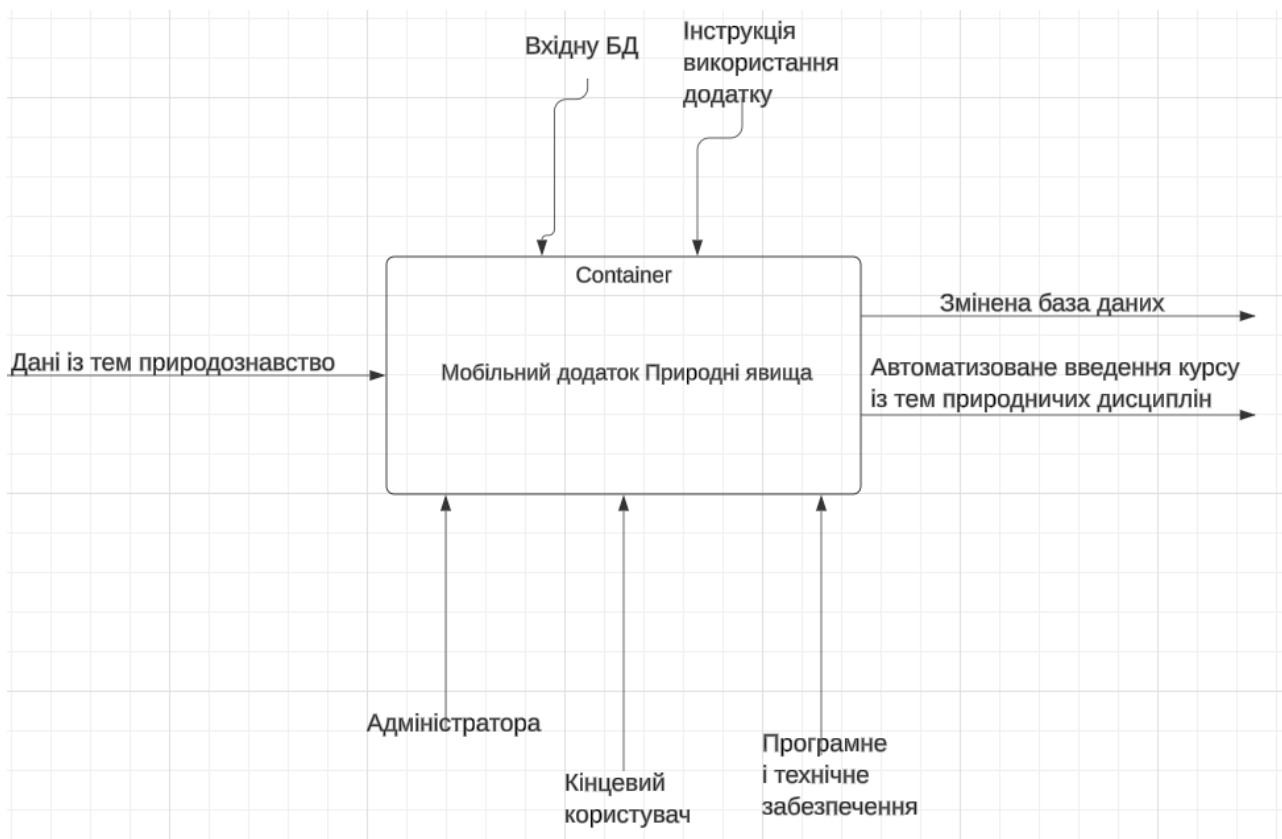


Рисунок 3.2 - Контекстна модель мобільного додатку природничі дисципліни

Джерело: запропоновано автором

Архітектура мобільного додатку визначає структуру, організацію, взаємодію всіх компонентів, для функціонування всієї системи. Зважаючи на те

що доступ до мобільного додатку буде надаватися користувачу через посилання вебдодатка, то серверну частину розробляти не будемо, тому на рисунок 3.3 буде надано зображення архітектури мобільного додатку яке базується на серверах вебдодатка «ClassRoom».

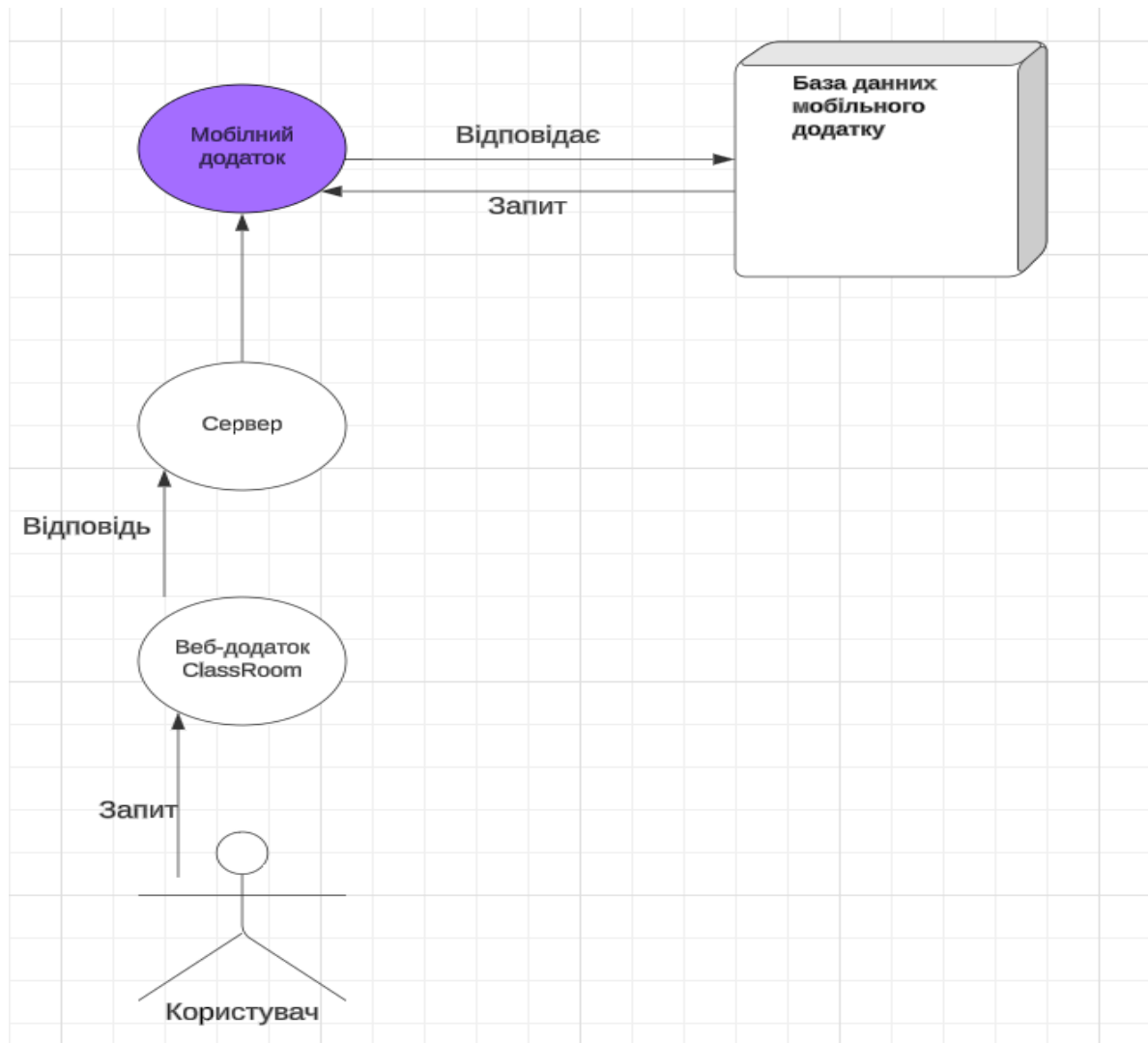


Рисунок 3.3 - Архітектури мобільного додактку природні явища

Джерело: запропоноване автором

3.2 Проектування бази даних мобільного застосунку із дисциплін природничого циклу

База даних мобільного додатку природні явища, формується із тем природничого циклу, ці дані мають тип довідкової інформації. Перелік тем із природознавства наведено в таблиці 3.1

Таблиця 3.1 - Список тем мобільного додатку природні явища

№ п/п	Тема	Тип інформації
1	Смерч	Довідник
2	Землетрус	Довідник
3	Гроза	Довідник
4	Дощ	Довідник
5	Виверження вулкана	Довідник
6	Пилова буря	Довідник
7	Тропічний циклон	Довідник
8	Вогняне торнадо	Довідник
9	Падіння астероїда	Довідник
10	Лісова пожежа	Довідник
11	Сонячне затемнення	Довідник
12	Наднова	Довідник
13	Сонячний спалах	Довідник
14	Лавина	Довідник
15	Каменепад	Довідник
16	Зсув ґрунту	Довідник
17	Цунамі	Довідник
18	Сніг	Довідник

Кожна тема в базі даних складається із категорій про природне явище. Перелік категорій наведено в таблиці 3.2.

Таблиця 3.2 - Список категорій кожного природного явища

№ п/п	Категорія
1	Інформація про природне явище
2	Основні елементи
3	Історична довідка
4	Найпоширеніші місця виникнення

Окремо взята категорія із таблиці 3.2 має власні підкатегорії. Нижче буде наведено в табличному варіанті кожен категорію окремо:

Таблиця 3.3 - Категорія : Інформація про природне явища

Ключ доступу	Назва природного явища	Зображення теми	Опис природного явища	Тема	Головне зображення	Порядковий номер
...	...	...	...	...	...	...

Таблиця 3.4 - Категорія : Основні елементи

Ключ доступу	Назва природного явища	Зображення теми	Перший атрибут	Другий атрибут	Третій атрибут	Назва першого атрибуту	Назва другого атрибуту	Назва третього атрибуту	Головне зображення	Порядковий номер
...	...	...	...	...	...	...	...	...	...	...

Таблиця 3.5 - Категорія : Історична довідка

Ключ доступу	Назва природного явища	Зображення теми	Опис природного явища	Порядковий номер
...	...	...	...	...

Таблиця 3.6 - Категорія : Місця виникнення

Ключ доступу	Назва природного явища	Зображення теми	Опис природного явища	Порядковий номер
...	...	...	...	...

Вся інформація в базі даних формується на основі вхідних даних, які описання вужче, всі дані представлені як довідково-пошукова інформація, вся інформація із бази даних виводиться на екран мобільного пристрою.

Для проєктування бази даних будемо використовуємо семантичну модель "сутність-зв'язок" (метод "ER - діаграм"). ER - діаграма складається із сутності та зв'язків, у сутності є атрибути, атрибути є властивістю сутності.

В таблиці 3.7 наведено перелік сутностей бази даних природного явища.

Таблиця 3.7 - Склад та опис сутностей

№ п/п	Сутності	Позначення	Назва, призначення
1	Природне явище	Природне явище	Інформація про природне явище
2	Основні елементи	Основні елементи	Інформація про основні атрибути явища
3	Історична довідка	Історична довідка	Інформація про історичні записи
4	Місце виникнення	Місце виникнення	Географічне найчастіше виникнення природного явища
5			

Зв'язки між сутностями наводяться в таблиці 3.8

Таблиця 3.8 - Назви зв'язків між інформаційними об'єктами

№ п/п	Перший інформаційний об'єкт	Другий інформаційний об'єкт	Назва зв'язку
1	Тема	Природне явище	Загальна інформація
2	Тема	Основні елементи	Атрибути
3	Тема	Історична довідка	Історія
4	Тема	Місце виникнення	Географічне розташування
5			

Нижче буде представлено структура глобальної ER-діаграми на рисунку 3.4.

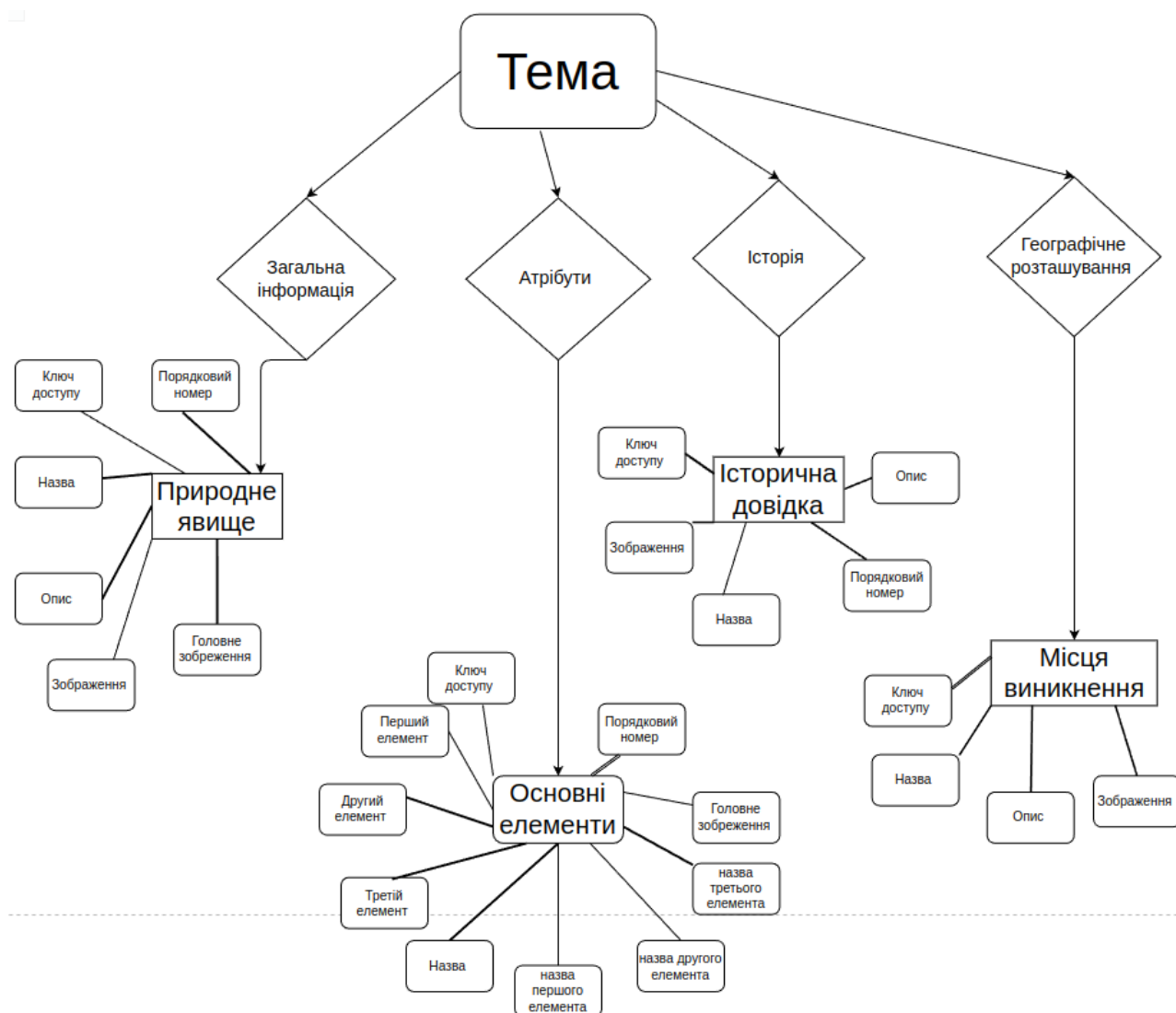


Рисунок 3.4 - Структура глобальної ER-діаграми

Джерело: запропоновано автором

Як показано важче на рисунку 3.1, структурний склад сутностей складається з атрибутів, ці атрибути будуть поповнюватися новими значеннями, звідси отримаємо словник даних бази даних. Словник бази даних наведено в таблиці 3.9.

Таблиця 3.9 - Словник даних

№ п/п	Позначення	Формат
1	Порядковий номер	Числовий
2	Ключ доступу	Числовий
3	Назва	Текстовий
4	Опис	Текстовий
5	Зображення	Числове/ціле
6	Головне зображення	Числове/ціле
7	Перший елемент	Числове/ціле
8	Другий елемент	Числове/ціле
9	Третій елемент	Числове/ціле
10	Назва першого елемента	Текстовий
11	Назва другого елемента	Текстовий
12	Назва третього елемента	Текстовий

В таблиці 3.7 наведено сутності бази даних, використаємо цю таблицю для наведення полів по кожній таблиці даних в реляційній базі даних та помістимо їх в таблицю 3.10

Таблиця 3.10 - Структурний склад інформаційних об'єктів

№ п/п	Позначення		Ключ (*)
	Сутність	Поля	
1	Таблиця_Природне явище	Порядковий номер, Ключ доступу, Назва, Опис, Зображення, Головне зображення	*

Продовження таблиці 3.10

2	Таблиця_Основні елементи	Порядковий номер, Ключ доступу, Назва, Опис, Зображення, Головне зображення, Перший елемент, Другий елемент, Третій елемент, Назва першого елемента, Назва другого елемента, Назва третього елемента	*
3	Таблиця_Історична довідка	Порядковий номер, Ключ доступу, Назва, Опис, Зображення, Головне зображення	*
4	Таблиця_Місця виникнення	Порядковий номер, Ключ доступу, Назва, Опис, Зображення	*

Використовуючи склад інформаційних об'єктів бази даних, наведений у таблиці 3.2., помістимо у таблицю 3.5 структуру кожного інформаційного об'єкту, тобто наведемо список полів (позначення, ознака ключа) по кожній таблиці реляційної бази даних. Ключовим називається те поле таблиці, що однозначно визначає запис таблиці, за яким здійснюється пошук. За допомогою ключів установлюються зв'язки між таблицями.

3.3 Реалізація інтерфейсу користувача

Інтерфейс мобільного додатку, включає багато аспектів, методів та підходів розробки інтерфейсу користувача, не менш важливий аспект це підтримка різних екранів мобільних додатків користувача. Враховуючи це підхід розробки мобільного додатку на базі операційної системи андроїд повинен бути кросплатформеним та екранне незалежним. В розділі 3.1 та 3.2 були описані основні структури мобільного додатку, на основі цих

інформаційних структур розробимо структуру інтерфейсу користувача рисунок 3.5

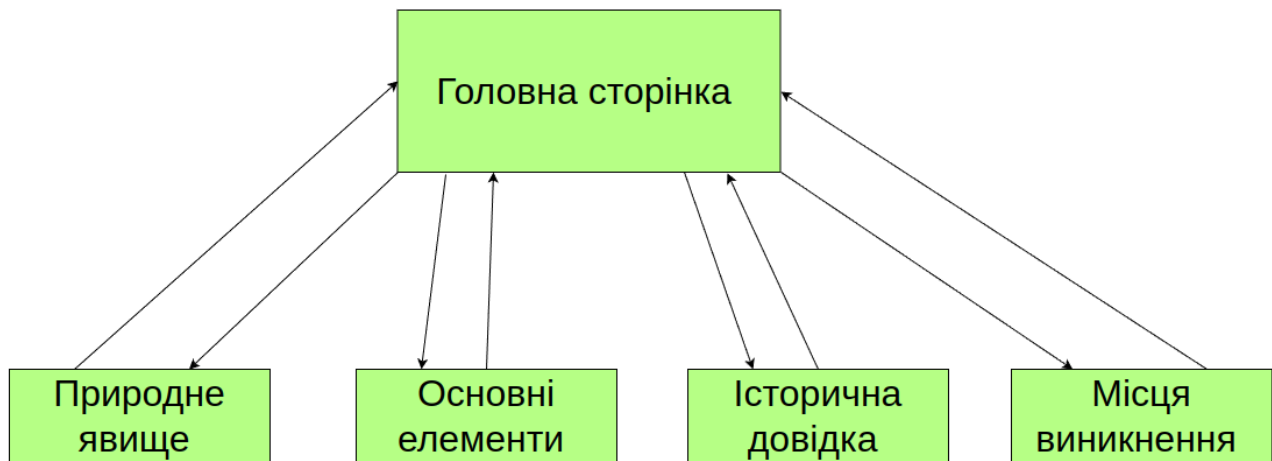


Рисунок. 3.5 - Структура інтерфейсу користувача

Джерело: запропоновано автором

На рисунку 3.5 зображено структуру інтерфейсу мобільного додатку природні явища, виходячи із цієї структури розробимо інтерфейс мобільного додатку по кожній сторінці. Для того, щоб об'єднати всі сторінки мобільного додатку використаємо навігацію по верхній панелі, яка зображена на рис. 3.6. Завдяки цій навігаційній панелі користувач зможе перемикатися з однієї на іншу сторінку мобільного додатку природні явища.

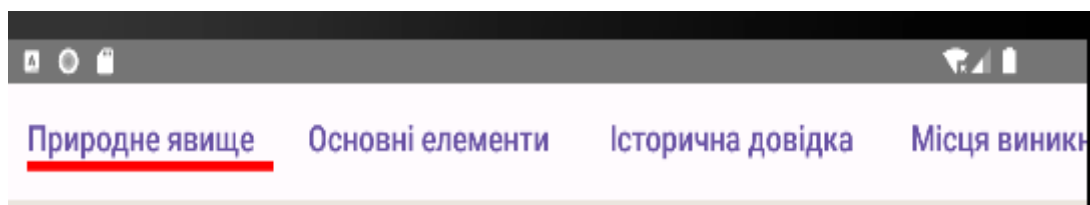


Рисунок 3.6 - Навігаційна панель мобільного додатку

Джерело: запропоновано автором

Особливістю навігаційної панелі із рисунка 3.6 є те що, користувач візуально зможе відрізнити на якій стронцій він перебуває за допомогою червоної лінії.

Головна сторінка буде складатися із кожної теми природного явища, кожний елемент буде складатися із зображення та тексту, головна сторінка зображена на рисунку 3.7.

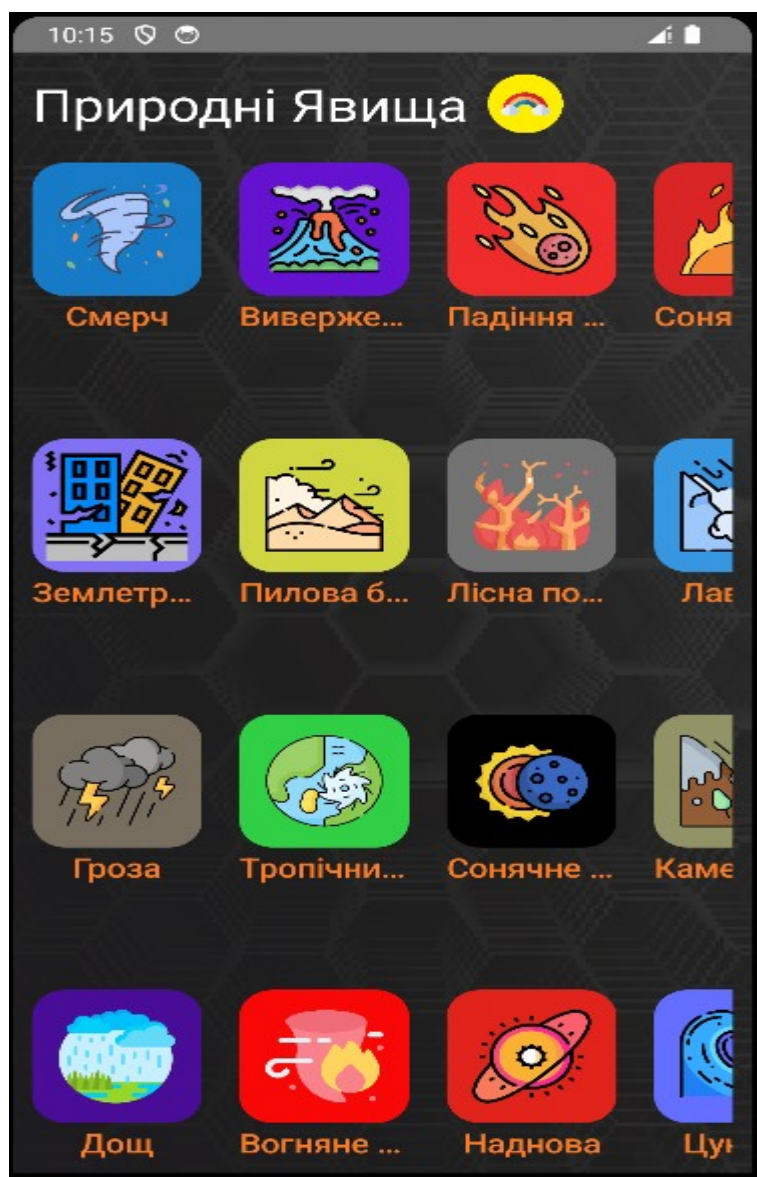


Рисунок 3.7 - Головна сторінка мобільного додатку

Джерело: запропоновано автором

Як видно із рисунка 3.7, макет головної сторінки не розпливається, відображається коректно на мобільному пристрої, враховуючи що стандарт розробки інтерфейсу для мобільних додатків є Матеріал 3, тому при розробці всього макету використовували останню версію Матеріал, тобто третю, в неї входять і шрифти, палітра кольорів, віджети та багато інших елементів макета.

Головна сторінка взаємодіє з іншими сторінками мобільного пристрою через самі UI — елементи, тобто через кожний елемент природного явища. При натисканні на ці - елемент спрацьовує логіка програми та відбувається перехід від однієї сторінки до іншої й навпаки при натисканні на системний значок назад відбувається зворотній перехід, інакше кажучи, до головного меню.

Сторінка природне явище буде складатися із конкретної теми природного явища, інтерфейс цієї сторінки включає текстове та графічні елементи, сторінка зображена на рисунку 3.8.



Рисунок - 3.8 Сторінка природне явище

Джерело: запропоновано автором

Сторінка основні елементи буде складатися із окремих елементів природного явища, як видно на рисунку 3.9 , що дорівнює конкретному явищу.

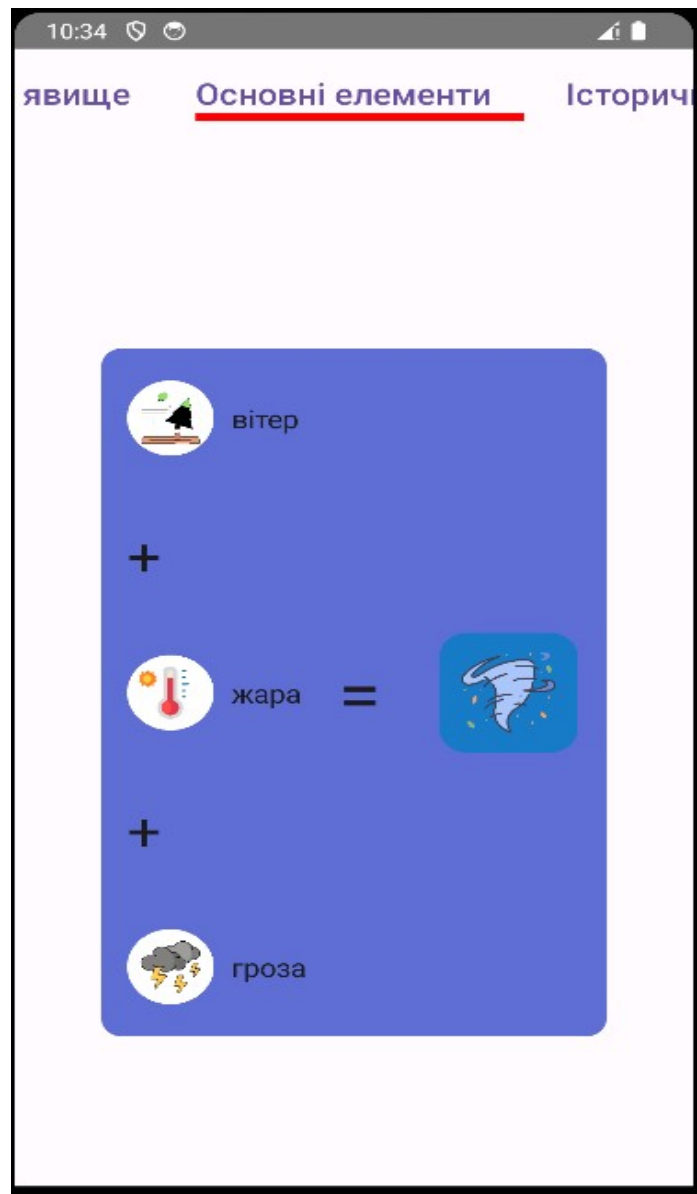


Рисунок 3.9 - Сторінка основні елементи

Джерело: запропоновано автором

Сторінка історична довідка буде складатися із текстового, графічного зображення, сторінка зображена на рисунку 3.10

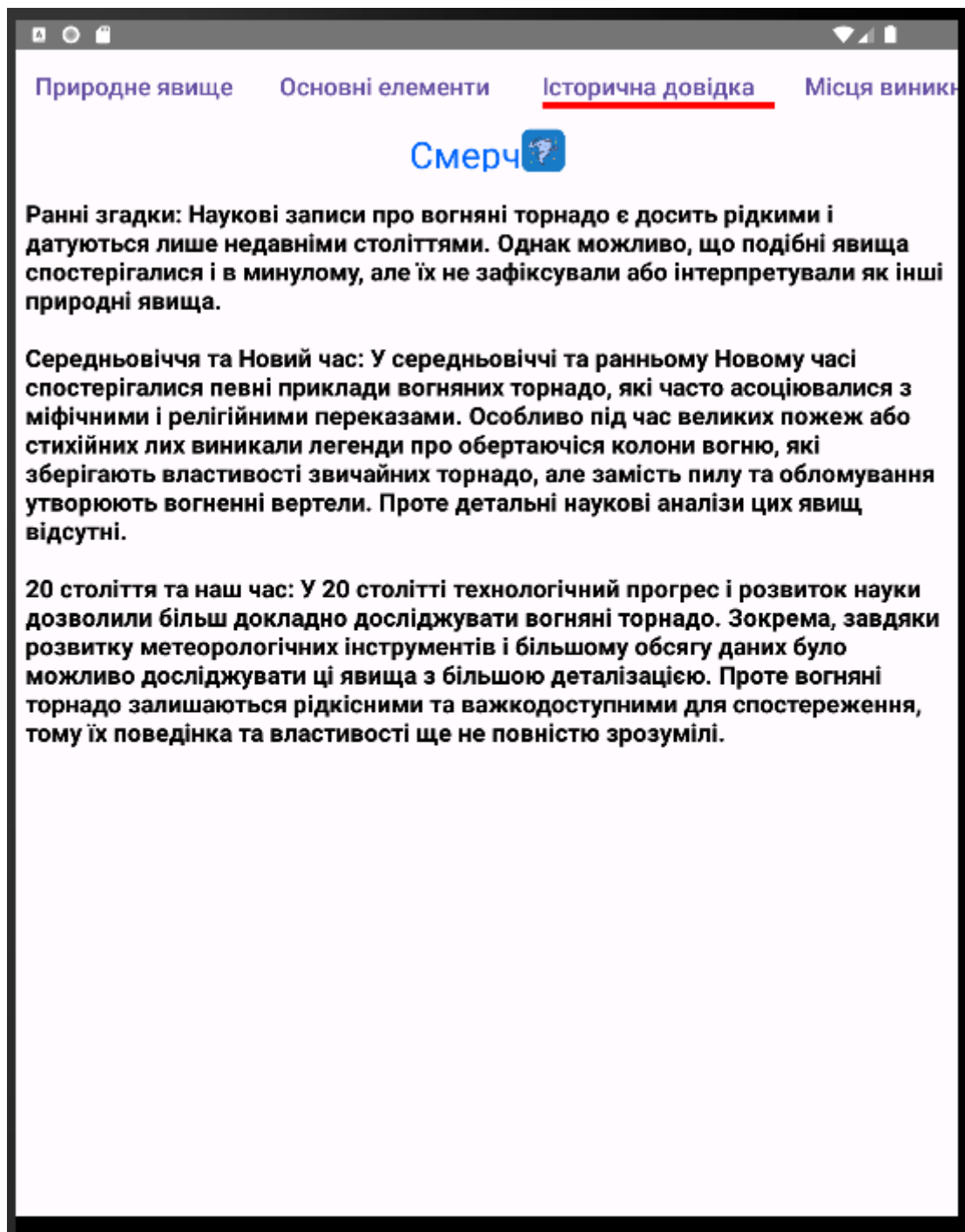


Рисунок 3.10 - Сторінка історична довідка

Джерело : запропоновано автором

Сторінка місця виникнення буде містити текстове та графічне зображення, сторінка зображена на рисунку 3.11

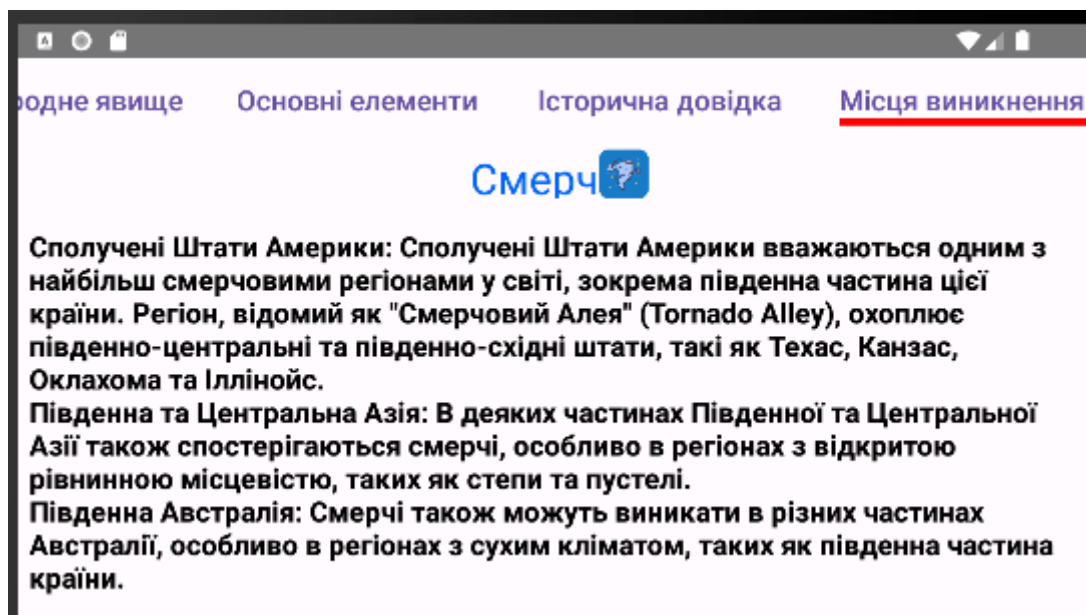


Рисунок 3.11 - Сторінка місця виникнення

Джерело : запропоновано автором

3.4 Фізична реалізація бази даних

Фізична реалізація бази даних мобільного додатку "Природні явища", реалізується в середовищі Room — Android Studio. Room — це настройка над SQLite. Room спроектований на більш високорівневих абстракцій - це зроблено для простоти реалізації бази даних на мобільних пристроях під управлінням операційної системи Android. Вибравши середовище реалізації Android Studio — Room, створимо базу даних мобільного додатку та назвемо її "natural_phenomenon". Відповідно до структурного складу інформаційних об'єктів (таблиця 3.8.) створюються таблиці бази даних та встановлюються зв'язки між ними. У зв'язку з тим що, база даних часом буде доповнюватися новими даними, то на рисунку 3.12 буде представлено лише початкову версію бази даних мобільного додатку природні явища. Реалізація бази даних у середовищі Room дозволяє ефективно керувати даними та забезпечує швидкий доступ до них в мобільному додатку "Природні явища".

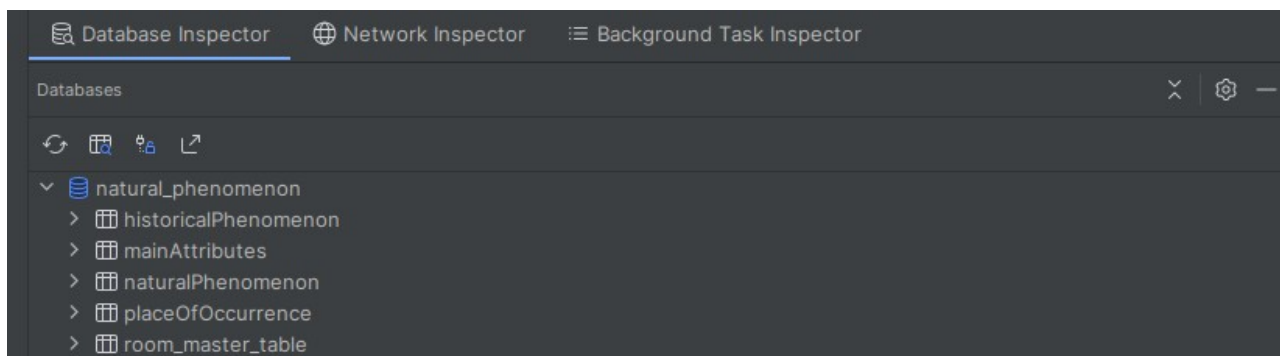


Рисунок 3.12 - Перелік таблиць бази даних natural_phenomenon

Джерело: запропоноване автором

На рисунку 3.13 зображено розгорнуту версію бази даних із рисунка 3.12. Схема бази даних продемонстровано на рисунку 3.14.

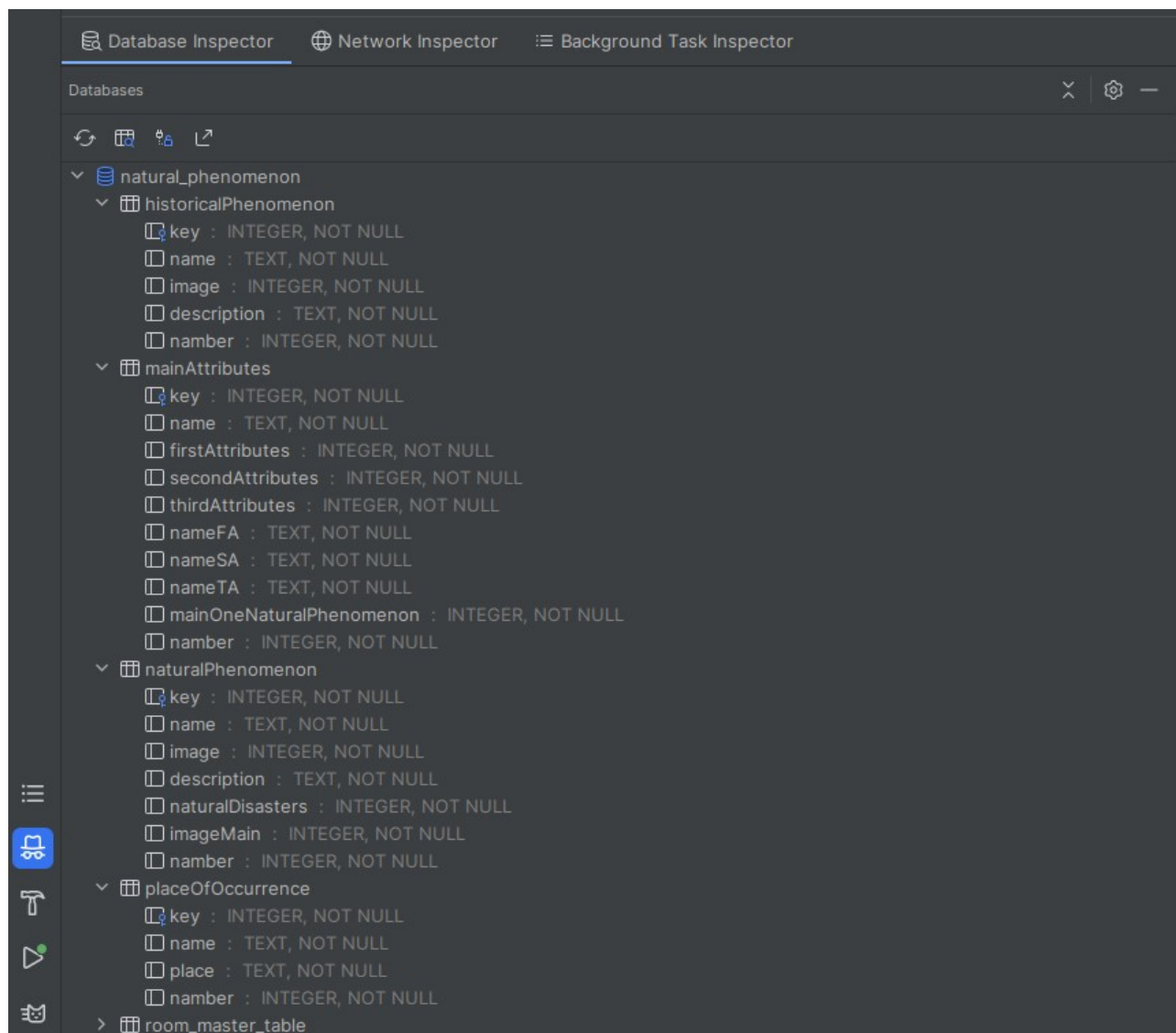


Рисунок 3.13 - Розгорнута версія бази даних natural_phenomenon

Джерело: запропоноване автором

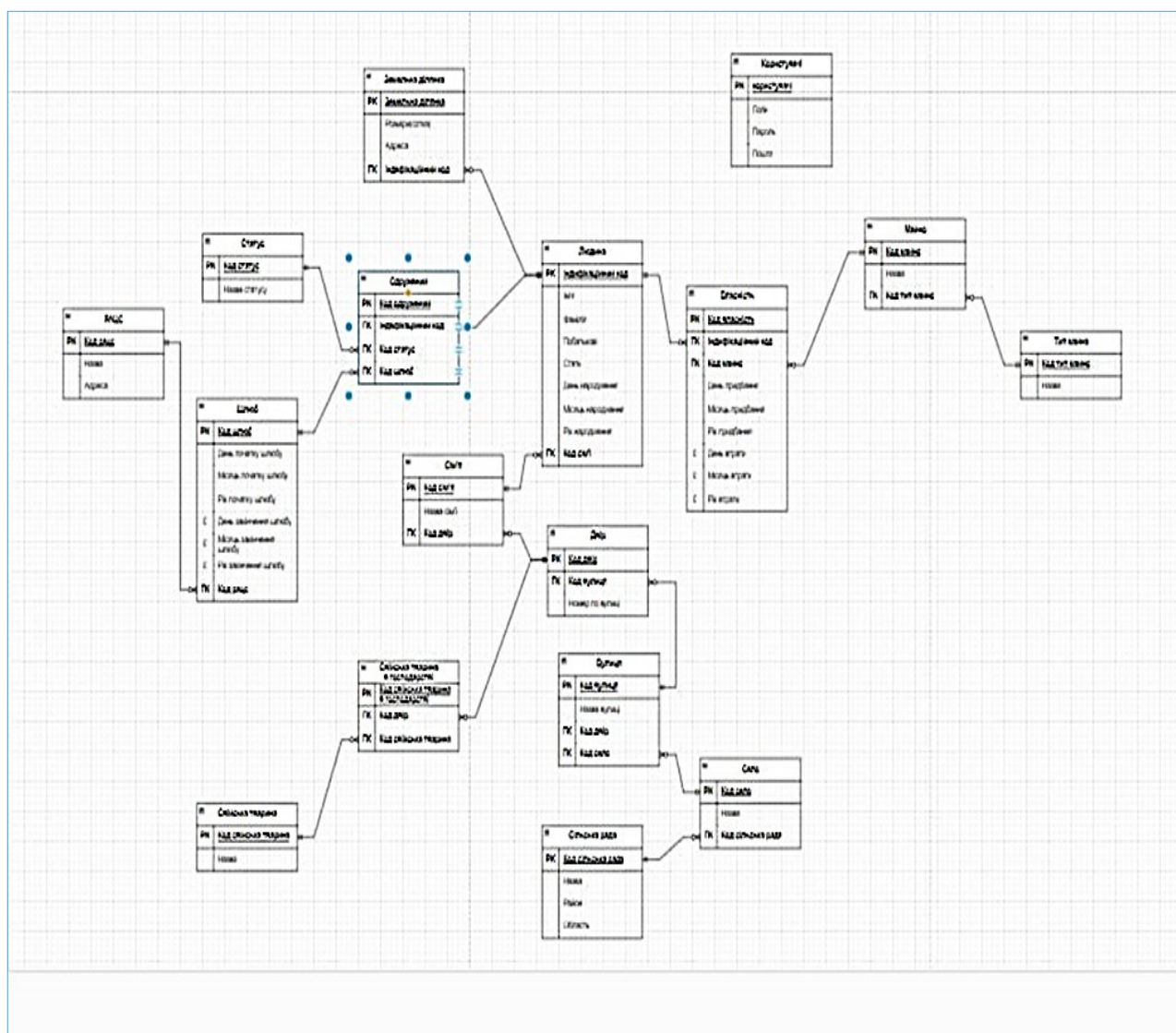


Рисунок 3.14 - Схема бази даних "natural_phenomenon"

Джерело: запропоноване автором

Таким чином, розроблена база даних для мобільного додатку, яка функціонує та відповідає стандартам розробки СУБД під операційну систему Android.

ВИСНОВКИ

У ході кваліфікаційної роботи було розроблено мобільний додаток для дисциплін природничого циклу. Робота містила аналіз вимог до системи та розгляд функціональних можливостей. Ми також провели аналіз схожих програмних рішень, щоб оцінити їхні переваги та недоліки.

Мобільний додаток був створений з використанням сучасних технологій, таких як Kotlin, Java, XML, що забезпечило нам можливість розробити масштабований та ефективний продукт. Ми розробили інтерфейс користувача таким чином, щоб забезпечити його інтуїтивну зрозумілість та зручність.

У розробленому мобільному додатку реалізовано різні модулі, включаючи можливість перегляду та редагування даних.

У процесі розробки були створені прототипи інтерфейсу користувача для різних частин тем природничих дисциплін, що покращує загальний досвід користувача та спрощує його взаємодію з системою.

Також ми реалізували фізичну базу даних у середовищі Android Studio-Room, що дозволяє ефективно збирати, зберігати та обробляти дані мобільного додатку.

В результаті роботи ми досягли поставленої мети та розробили функціональний мобільний додаток для дисциплін природничого циклу, який відповідає вимогам та потребам користувачів. Цей додаток може бути використаний у різних галузях для ефективного представлення та навчання користувачів.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Jetpack Compose UI App Development Toolkit - Android Developers.
URL : <https://developer.android.com/develop/ui/compose/>
(дата звернення:09.02.2024)
2. Бхаргава А. І., Грокаємо алгоритми : підручник Львів , 2024. 256 с.
3. Material 3 -стандарт розробки інтерфейсу користувача.
URL : <https://m3.material.io/> (дата звернення:11.03.2024)
4. Посібник по Jetpack Compose.
URL:<https://developer.android.com/develop/ui/compose/tutorial>
(датазвернення:20.03.2024)
5. Мартін Р. В, Бейтс Б. С.Чиста архітектура: підручник, Фабула, 2020 368 с.
6. Animation in Compose.
URL : <https://developer.android.com/develop/ui/compose/animation/introduction>
(дата звернення:30.03.2024)
7. Сьєрі К. А, Бейтс Б. С., Head First. Java : довідник, Фабула, 2022. 720 с.
8. App bars | Jetpack Compose | Android Developers.
URL : <https://developer.android.com/develop/ui/compose/components/app-bars>
(дата звернення:05.04.2024)
9. База даних Room
URL : <https://developer.android.com/training/data-storage/room>
(дата звернення:10.04.2024)
10. Shaun L., Dunn M. Native Mobile Development: A Cross-Reference for iOS and Android 1st Edition, O'Reilly Media 2019. 392 с.
- 11.Bryan Sills,Brian Gardner, Kristin Marsicano, Chris Stewart. Android Programming: The Big Nerd Ranch Guide (Big Nerd Ranch Guides) 5th Edition. Addison-Wesley Professional, 2022. 688 с.
12. Navigation with Compose | Jetpack Compose | Android Developers.
URL : <https://developer.android.com/develop/ui/compose/navigation>

(дата звернення:15.04.2024)

13. Курс природознавства.

URL : <https://dovidka.biz.ua/category/shkola/prirodnavstvo/>

(дата звернення:14.01.2024)

14. Підручник по правознавству 5 клас.

URL : <https://pidruchnyk.com.ua/1718-piznaiemo-pryrodu-bida-5-klas.html>

(дата звернення:18.01.2024)

ДОДАТКИ

ДОДАТОК А

ТЕХНІЧНЕ ЗАВДАННЯ

**На розробку вебзастосунка для інтерактивного вивчення дисциплін
природничого циклу**

Погоджено:

Кандидат технічних наук, доцент кафедри
кібернетики та інформатики

Студент групи ICT 2001.

Санжаков Андрій Андрійович

1. Призначення й мета створення мобільного додатку

1.1 Призначення мобільного додатку

Мобільний додаток призначений інтерактивного вивчення дисциплін природничого циклу.

1.2 Мета створення мобільного додатку

Розробка андроїд-додатка, на базі «Classroom» який допоможе учням в інтерактивній та цікавій формі вивчати природознавство, роблячи процес навчання більш ефективним та захопливим.

1.3 Цільова аудиторія

У цільовій аудиторії мобільного додатку можна виділити такі групи:

1. Учні молодших класів
2. Студенти

2. Вимоги до мобільного додатку

2.1 Вимоги до мобільного додатку в цілому

2.1.1 Вимоги до структури й функціонування мобільного додатку

Мобільний додаток повинен бути реалізований у вигляді андроїд додатка, доступний через посилання «ClassRoom». Сам андроїд додаток повинен бути доступний офлайн.

2.1.2 Вимоги до персоналу

Персонал має володіти базовими знаннями та навичками роботи із мобільними та десктопними технічними засобами для підтримки й експлуатації мобільного додатку.

2.1.3 Вимоги до збереження мобільного додатку

Уся інформація надана у мобільному додатку буде зберігатися у базі даних реалізованій засобами системи управління базами даних Room(SQLLite).

2.1.4 Вимоги до розмежування доступу

Розроблюваний мобільний додаток не має бути приватним.

Відповідно до прав доступу у мобільний додаток, користувачів можна поділити на користувачів та адміністратора.

Користувачі можуть переглядати усі сторінки мобільного додатку

Адміністратор може редагувати зовнішній вигляд та наповнення мобільного додатку.

2.2 Структура мобільного додатку

2.2.1 Навігація

Відповідно до дизайну мобільного додатку , для навігації буде створено контент меню. Контент меню буде доступне в кожному елементу мобільного додатку, це меню необхідне для швидкого переміщення по мобільному додатку.

Меню буде відображатися в кожному елементі мобільного додатку.

2.2.3 Наповнення мобільного-додатка (контент)

Сторінки всіх розділів мобільного додатку мають бути створені на основі даних з бази даних різних інформаційно-наукових. Редагування вмісту розділів повинно бути можливим через безпосереднє внесення даних в базу даних Room в Android Studio. Наповнення розділів мобільного додатку також повинно здійснюватися за допомогою інструменту Android Studio.

2.3 Вимоги до функціонування системи

2.3.1 Потреби користувача

Адміністративна частина системи керування контентом повинна забезпечувати функції додавання, редагування і видалення сторінок.

2.3.2 Функціональні вимоги

На головній сторінці будуть представлені такі елементи:

Кожний елемент із розділів природничих дисциплін;

Меню навігацію кожного елемента;

2.3.3 Системні вимоги

Системні вимоги відповідно до потреб користувачів:

- інформація по кожному елементу.
- офлайн доступ до інформації мобільного додатку;
- інтерактивний режим мобільного додатку

2.4 Вимоги до видів забезпечення

2.4.1 Вимоги до інформаційного забезпечення

Реалізація мобільного додатку відбувається з використанням:

- Kotlin 2.0.0
- Room 2.6.1
- Java SE 20

2.4.2 Вимоги до лінгвістичного забезпечення

Мобільний додаток має бути виконаний українською мовою.

2.4.3 Вимоги до програмного забезпечення

Програмне забезпечення клієнтської частини повинне задовольняти наступним вимогам:

- Веб-браузер: Internet Explorer 7.0 і вище, або Firefox 3.5 і вище, або Safari 3.2.1 і вище, або Chrome 2 і вище.
- Мобільний додаток : android 6.0 і вище.

3 Склад і зміст робіт зі створення мобільного додатку

Докладний опис етапів роботи зі створення мобільного додатку наведено в таблиці А.1.

Таблиця А.1 – Етапи створення мобільного додатку

№	Склад і зміст робіт	Строк розробки
1	Визначення системних вимог	3 дні

Продовження таблиці А.1

2	Розробка прототипу інтерфейсу користувача	1 день
3	Розробка інтерфейсу користувача	7 днів
5	Створення бази даних	9 днів
6	Розширення функціонала	8 днів
7	Підгонка інтерфейсу під потреби користувача	3 днів
8	Розгортання мобільного додатку	1 день
10	Тестування мобільного додатку	3 днів
11	Перевірка виконаної програми	4 дні
13	Перевірка виконання всього проєкту	1 день
	Загальна тривалість робіт	40 днів

4. Вимоги до складу й змісту робіт із введення мобільного додатку в експлуатацію

Для того, щоб мобільним додатком могли користуватися учні та студенти, необхідно розмістити його у мережі Інтернет, для цього треба розмістити його на освітній платформі «ClassRoom».

ДОДАТОК Б

ПЛАНУВАННЯ РОБІТ

У зв'язку з пандемією та широкомасштабним вторгненням російських військ на територію України, всі аспекти людської діяльності змушені були швидко адаптуватися до нових умов та розвиватися у незвичному форматі. Виходячи з цього, освітній процес повністю перейшов до онлайн-формату.

Мета проекту полягає у створенні програмного продукту, який допоможе учням в інтерактивній та цікавій формі вивчати природознавство, роблячи процес навчання більш ефективним та захопливим. Проект буде завершено вчасно, що підтверджується календарним планом проекту.

Діаграма Ганта. Побудований календарний план виконання проекту. Для того, щоб мати реальне уявлення про тривалість виконання робіт з урахуванням обмеженості у використанні ресурсів, з урахуванням вихідних та святкових днів, побудовано календарний графік. Діаграма Ганта та список робіт діаграми Ганта зображені на рисунку Б.1.

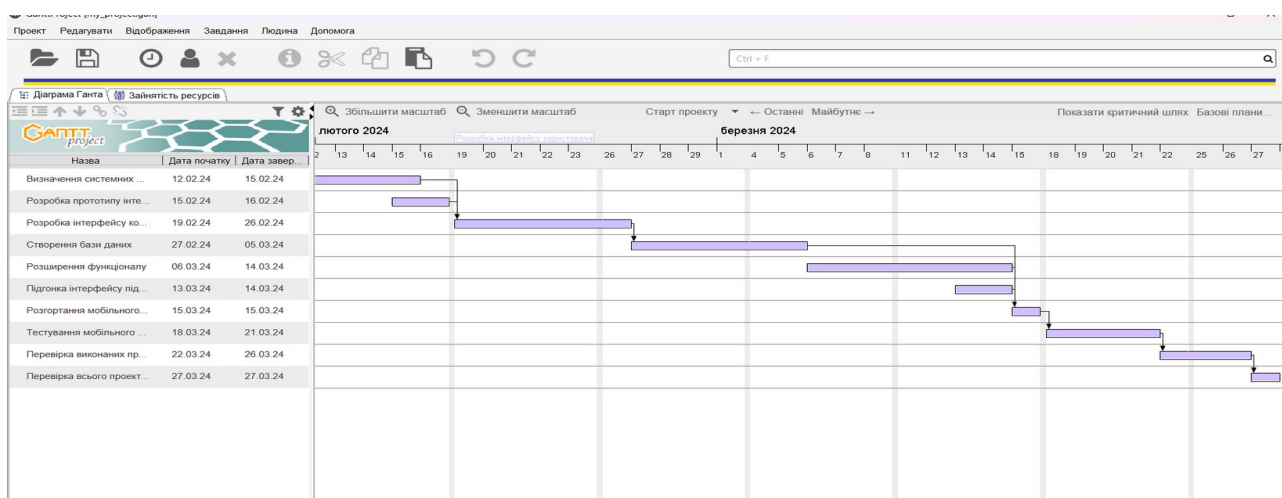


Рисунок Б.1 – Діаграма Ганта

Джерело: запропоновано автором

Управління ризиками проєкту – це систематичний процес виявлення, оцінки та реагування на ризики, які можуть вплинути на успіх проєкт. Метою управління ризиками є мінімізація негативного впливу ризику та максимізація можливостей для успішних проєктів.

Ризик – це ймовірність того, що відбудеться подія, яка може мати негативний або позитивний вплив на цілі організації, проєкт або діяльності. Ризик часто пов'язаний з невизначеністю та ймовірністю відхилення від очікуваного результату.. Процес ідентифікації ризиків включає виявлення та документування характеристик ризиків, які можуть вплинути на проєкт.

Перерахуймо можливі ризики проєкту, визначимо їх відносну важливість і визначимо критичність, а також можливі способи їх уникнення. Введемо умовну шкалу рівнів критичності ризиків(табл. Б.2):

Таблиця Б.2 – Рівні критичності ризиків

1	Незначний
2	Низький
3	Середній
4	Високий
5	Критичний

Ризики проєкту, визначені в результаті ідентифікації, а також їх потенційний вплив та ймовірність виникнення представлені у таблиці Б.3.

Таблиця Б.3 – Ризики проекту

№	Опис	Ймовірність	Вплив
1	Втрата актуальності проєкту	1	4
2	Зміна вимог на пізніх етапах розробки	2	4
3	Незаплановані роботи, що потребують обов'язкового виконання	2	3

Продовження таблиці Б.3

4	Вимушене зміщення термінів виконання завдань або паралельне виконання цих завдань	3	3
5	Нездійснені або важкі задачі	3	5
6	Пошкодження або відсутність обладнання	1	5
7	Непрацездатність розробника	1	5
8	Низьке залучення замовника	2	4

Тепер, складемо матрицю залежності втрат від ймовірності виникнення кожного із зазначених ризиків (табл. Б.4).

Таблиця Б.4 – Матриця «Ймовірність – Втрати»

Ймовірність \ Втрати	1	2	3	4	5
1				1	6,7
2			3	2,8	
3			4		5
4					
5					

ДОДАТОК В
КОПІЇ ПУБЛІКАЦІЙ

**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
СУМСЬКИЙ НАЦІОНАЛЬНИЙ АГРАРНИЙ УНІВЕРСИТЕТ**

**МАТЕРІАЛИ
ВСЕУКРАЇНСЬКОЇ НАУКОВОЇ
КОНФЕРЕНЦІЇ СТУДЕНТІВ
ТА АСПІРАНТІВ, ПРИСВЯЧЕНОЇ
МІЖНАРОДНОМУ ДНЮ СТУДЕНТА**

(13 - 17 листопада 2023 р., м. Суми)

Маслюк Б.О. ЗНАЧЕННЯ ІПОТЕЧНОГО КРЕДИТУВАННЯ У КОНТЕКСТІ РЕАЛІЗАЦІЇ ДЕРЖАВНОЇ СТРАТЕГІЇ ВІДНОВЛЕННЯ АГРАРНОЇ СФЕРИ.....	302
Артеменко Є.А. НАПРЯМИ ПОСТВОЄННОГО ВІДНОВЛЕННЯ ПІДПРИЄМНИЦЬКОЇ АКТИВНОСТІ В УКРАЇНІ.....	303
Василець К. А. ПРОБЛЕМИ ЗАБЕЗПЕЧЕННЯ КОНКУРЕНТОСПРОМОЖНОСТІ АГРАРНОГО СЕКТОРУ УКРАЇНИ В УМОВАХ ВІЙНИ	304
Глух В. П. РОЛЬ БУДІВЕЛЬНИХ ОРГАНІЗАЦІЙ У СТАЛІЙ РЕЛОКАЦІЇ ПРОМИСЛОВОСТІ В УКРАЇНІ ПІД ЧАС ВІЙНИ	305
Гуменник І. М. БІЗНЕС-ПЛАНУВАННЯ ПІДПРИЄМНИЦЬКОЇ ДІЯЛЬНОСТІ В АГРАРНІЙ СФЕРІ	306
Жежеря Ж. В. МІКРОЕКОНОМІЧНИЙ АНАЛІЗ РИНКУ ПРАЦІ В УКРАЇНІ	307
Затара А. М. МЕТОДИ ВИЗНАЧЕННЯ РИЗИК-ФАКТОРІВ	308
Затара А. М. РОЗРОБКА СИСТЕМИ МОНІТОРИНГУ РИЗИКОВИХ ПОДІЙ	309
Кригін В.В. МЕТОДОЛОГІЯ АНАЛІЗУ СОЦІАЛЬНО-ЕКОНОМІЧНОГО РОЗВИТКУ ТЕРИТОРІАЛЬНОЇ ГРОМАДИ	310
Литвиненко О. М. РОЗВИТОК КООПЕРАЦІЇ В АГРАРНОМУ СЕКТОРІ	311
Неледєва А. О. ЕВОЛЮЦІЯ МОТИВАЦІЇ : ПЕРЕЛІК СТРАТЕГІЙ ДЛЯ АДАПТАЦІЇ ДО СУЧАСНИХ УМОВ.....	312
Неледєва А.О. МОТИВАЦІЯ ПЕРСОНАЛУ В УМОВАХ ЗРОСТАННЯ ВПЛИВУ ШТУЧНОГО ІНТЕЛЕКТУ	313
Новіков Д.С. ДОСЛІДЖЕННЯ ЕКОНОМІЧНИХ ПЕРЕДУМОВ ТА ДОЦІЛЬНОСТІ ВИКОРИСТАННЯ АГРОДРОНІВ.....	314
Пігуль О.В. МЕТОДОЛОГІЧНІ ЗАСАДИ ОЦІНКИ СТАЛОСТІ ЖИТЛОВИХ ПРОЄКТІВ	315
Поздирко В.С. НАПРЯМИ ПІДВИЩЕННЯ ЕКОНОМІЧНОЇ ЕФЕКТИВНОСТІ ГАЛУЗІ ЗЕРНОВИРОБНИЦТВА	316
Поздирко В.С. НАПРЯМИ РОЗВИТКУ ГАЛУЗІ РОСЛИННИЦТВА	317
Сьомушін В.О. УПРАВЛІННЯ ЖИТТЄВИМ ЦИКЛОМ БУДІВЕЛЬНОЇ ОРГАНІЗАЦІЇ ЯК ІНСТРУМЕНТ СТАЛОГО РОЗВИТКУ	318
Щенякін Д.О. ЕКОНОМІЧНА ЕФЕКТИВНІСТЬ ВИКОРИСТАННЯ СОНЯЧНИХ ПАНЕЛЕЙ НА НАУКОВО-ДОСЛІДНОМУ ПІДПРИЄМСТВІ	319
Белоконь П.А. СУЧАСНІ АВТОМАТИЗОВАНІ ІНФОРМАЦІЙНІ СИСТЕМИ РЕКРУТИНГУ	320
Стеблянюк В.В. АНАЛІТИЧНИЙ ОГЛЯД ІТ-ТЕХНОЛОГІЙ ЩОДО ОБЛІКУ ВИТРАТ НА ВИРОБНИЦТВО СІЛЬСЬКО-ГОСПОДАРСЬКОЇ ПРОДУКЦІЇ. ПОСТАНОВКА ЗАДАЧІ	321
Агеев О.А. ОГЛЯД ПЛАТФОРМ ЕЛЕКТРОННОГО НАВЧАННЯ.....	322
Блаженко Д.О. МЕТОДИ ТА ПРОБЛЕМИ ПРОЄКТУВАННЯ ІНФОРМАЦІЙНИХ ПІДСИСТЕМ ВЕДЕННЯ ЗАМОВЛЕНЬ ІЗ ВИКОРИСТАННЯМ ТЕХНОЛОГІЇ TRIZ.....	323
Гончар Д.О. АВТОМАТИЗАЦІЯ РОЗРАХУНКУ БАЛАНСУ ГУМУСУ ТА ПОЖИВНИХ РЕЧОВИН В ҐРУНТІ.....	324
Доля А.Р. СУЧАСНІ ТЕХНОЛОГІЇ РОЗРОБКИ АДМІНІСТРАТИВНИХ МОДУЛІВ	325
Дубінчин В.С. РОЗРОБКА ІНФОРМАЦІЙНОЇ ПІДСИСТЕМИ ОЦІНЮВАННЯ ПРАЦЕЗДАТНОСТІ КОМП'ЮТЕРИЗОВАНИХ РОБОЧИХ МІСЦЬ	326
Крамський А.С., Бусенко І.В. ПРОГНОЗУВАННЯ КІЛЬКОСТІ СТУДЕНТІВ В УКРАЇНІ З ВИКОРИСТАННЯМ ЕКОНОМЕТРИЧНИХ МЕТОДІВ	327
Олійник А.В. ПОСТАНОВКА ЗАДАЧІ ТА ПРОЄКТУВАННЯ ПІДСИСТЕМИ ОБЛІКУ ПАЛИВНО-МАСТИЛЬНИХ МАТЕРІАЛІВ В ТЕРИТОРІАЛЬНІЙ ГРОМАДІ.....	328
Руденок О.А. ПОШУК ІНФОРМАЦІЇ В ТЕКСТОВИХ ДОКУМЕНТАХ З УРАХУВАННЯМ КОНТЕКСТУ.....	329
Санжаков А. А. АНАЛІТИЧНИЙ ОГЛЯД ВИКОРИСТАННЯ ІТ- ТЕХНОЛОГІЙ У МОБІЛЬНІЙ РОЗРОБЦІ ДЛЯ ОС АНДРОЇД	330
Сіденко А.С. ПРОБЛЕМИ РЕГІСТРАЦІЇ ТА ВІЗУАЛІЗАЦІЇ ДАНИХ МОНІТОРИНГУ НАВКОЛИШНЬОГО СЕРЕДОВИЩА.....	331
Терещенко С.С. ПОСТАНОВКА ЗАДАЧІ ТА ПРОЄКТУВАННЯ ПІДСИСТЕМИ АВТОМАТИЗАЦІЇ ОБЛІКОВИХ ПРОЦЕСІВ В АТЕЛЬЄ З ВИГОТОВЛЕННЯ ОДЯГУ	332
Чубун М.П. СУЧАСНІ ТЕНДЕНЦІЇ У СТВОРЕННІ ВЕБ-САЙТІВ ДЛЯ ІНТЕРНЕТ-МАГАЗИНІВ КОМП'ЮТЕРИЗОВАНИХ РОБОЧИХ МІСЦЬ	333
Шелковський С.В. СУЧАСНІ ТЕНДЕНЦІЇ ПРОЄКТУВАННЯ ПІДСИСТЕМИ ОБЛІКУ ПІДПРИЄМСТВ ПЕРЕРОБКИ СІЛЬСЬКОГОСПОДАРСЬКОЇ ПРОДУКЦІЇ.....	334
Горбань С.В. ТЕОРЕТИЧНІ АСПЕКТИ ФОРМУВАННЯ ПОЗИТИВНОГО ІМІДЖУ ПІДПРИЄМСТВА.....	335
Miao Yin Hai INFLUENCE OF MANAGEMENT DECISIONS ON THE ENTERPRISE'S INNOVATION STRATEGY.....	336
Zhao Min THE EFFICIENT MANAGEMENT OF TALENTS FOR THE DEVELOPMENT OF ENTERPRISE	337
Заболотна А.І. УПРАВЛІННЯ ПРОЦЕСАМИ ПІДВИЩЕННЯ ЯКОСТІ ЖИТТЯ РЕГІОНУ ЗА МАТЕРІАЛАМИ ДРУЖБІВСЬКОЇ ТГ	338

АНАЛІТИЧНИЙ ОГЛЯД ВИКОРИСТАННЯ ІТ-ТЕХНОЛОГІЙ У МОБІЛЬНІЙ РОЗРОБЦІ ДЛЯ ОС АНДРОЇД

Санжаков А.А., студ. 4 курсу ФЕІМ, спец. «Інформаційні системи та технології»

Науковий керівник: доцент А.В.Толбатов

Сумський НАУ

З впровадженням мобільних технологій у сучасному світі зростає і важливість вибору оптимального способу створення користувацьких інтерфейсів для мобільних додатків. Для цього варто обирати між двома основними підходами: XML-розміткою і Jetpack Compose, враховуючи потреби проекту та власні вподобання розробника. Вибір між ними визначається потребами проекту та власними вподобаннями розробника. Хоча XML-розмітка вже давно використовується в Android-розробці, Compose є новим, але обіцяючим фреймворком, який надає зручний та декларативний спосіб створення інтерфейсів.

XML - розмітка використовує декларативний підхід до створення користувацьких інтерфейсів, де розмітка та логіка розділені. Вона має широкую підтримку але інколи створення складних інтерфейсів може призвести до зайвої вкладеності та більшого об'єму коду. З іншого боку, Jetpack Compose - це новий інструмент в мобільній розробці, який пропонує декларативний спосіб створення інтерфейсів, що робить розробку більш простою та зручною. Він також надає можливості для побудови більш динамічних інтерфейсів.

Аналітичний огляд обох технологій в мобільній розробці

Переваги:

1.XML-розмітка:

2.XML-розмітка використовується в Android SDK дуже довго, і багато проектів використовують цей підхід.

3.В XML зручно розділяти макети і логіку, що сприяє кращій структурі проекту.

4.Є багато інструментів та середовищ розробки, які підтримують роботу з XML-розміткою.

Недоліки:

1.XML-розмітка може бути масштабованою та важко зберігати великі розмітки.

2.Іноді розмітка може бути важко зрозумілою та довгою для написання та підтримки.

Compose-фреймворк:

Переваги:

1. Jetpack Compose - це новий декларативний фреймворк від Google для створення інтерфейсів.

2. Використання Kotlin в Compose спрощує створення інтерфейсів і зменшує кількість коду, необхідного для розробки.

3. Compose має багатий набір вбудованих компонентів та можливостей для створення кастомних інтерфейсів.

Недоліки:

1. Compose є новим фреймворком, і він може вимагати часу для освоєння нових концепцій.

2. На даний момент не всі бібліотеки та компоненти підтримують Compose.

Вибір між XML-розміткою і Compose-фреймворком у мобільній розробці для Android залежить від конкретних потреб проекту та власних вподобань розробника. XML-розмітка є відомою та довго використовується практикою, має широкую підтримку та дозволяє чітко розділяти макети та логіку. Проте вона може бути важкою для підтримки, особливо при створенні великих інтерфейсів.

Compose-фреймворк, навпаки, представляє новий підхід до створення інтерфейсів, що базується на мові програмування Kotlin. Він спрощує розробку та має значний потенціал для зменшення обсягу коду. Однак для його використання може знадобитися час для освоєння нових концепцій, і не всі існуючі бібліотеки і компоненти підтримують його.

Отже, розробники повинні ретельно розглянути потреби свого проекту та свої власні навички для вибору оптимального підходу до створення користувацьких інтерфейсів у мобільних додатках для Android.

Література

1. "Android Developers" - офіційний ресурс від Google, який містить документацію та приклади для XML-розмітки та Jetpack Compose. <https://developer.android.com/>

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

**СУМСЬКИЙ НАЦІОНАЛЬНИЙ
АГРАРНИЙ УНІВЕРСИТЕТ**

МАТЕРІАЛИ

науково-практичної конференції
викладачів, аспірантів та студентів
Сумського НАУ

(14-16 травня 2024 р.)

Luo Hao. A STUDY ON CONSUMER BEHAVIOUR MANAGEMENT THROUGH ADVANCED MARKETING TOOLS	414
Wang Baichao. MANAGEMENT CRISIS IN AGRICULTURAL COMPANIES IN CURRENT SITUATIONS	415
Wang Ruijie. HUMAN RESOURCES MANAGEMENT OPTIMIZATION AND INNOVATION BASED ON ARTIFICIAL INTELLIGENCE	416
Yu Hang. RESEARCH ON THE INTRODUCTION OF MODERN MANAGEMENT TECHNOLOGY INTO ENTERPRISES	417
Zhou Kunming. PECULIARITIES OF MANAGING HIGH-TECH ENTERPRISES IN A TRANSPARENT ECONOMY	418
Бусенко І.В. АНАЛІЗ ОСНОВНИХ РІЗНОВИДІВ SQL-ІН'ЄКЦІЙ, АНАЛІТИКА ЗАСТОСУВАННЯ ТА СПОСОБИ ЗАХИСТУ ВІД НИХ	419
Белоконь П.А. РОЗРОБКА ІНФОРМАЦІЙНОЇ ПІДСИСТЕМИ ПІДТРИМКИ РОБОТИ HR-МЕНЕДЖЕРА	420
Стебляк В. В. РОЗРОБКА ВЕБ-ЗАСТОСУНКУ ОБЛІКУ ВИТРАТ НА ВИРОБНИЦТВО ПРОДУКЦІЇ. РЕАЛІЗАЦІЯ ПРОЄКТУ	421
Фідірко Т.Р. РОЗРОБКА ВЕБЗАСТОСУНКУ ОБЛІКУ ТЕХНІЧНОГО СТАНУ ЛЕГКОВИХ АВТОМОБІЛІВ. ПОСТАНОВКА ЗАДАЧІ	422
Фідірко Т.Р. РОЗРОБКА ВЕБЗАСТОСУНКУ ОБЛІКУ ТЕХНІЧНОГО СТАНУ ЛЕГКОВИХ АВТОМОБІЛІВ. РЕАЛІЗАЦІЯ ПРОЄКТУ	423
Биченко Є.В. РОЗРОБКА ПРОТОТИПУ ТА ПРОГРАМНА РЕАЛІЗАЦІЯ ІНТЕРФЕЙСУ КОРИСТУВАЧА ВЕБОРІЄНТОВАНОЇ ІНФОРМАЦІЙНОЇ СИСТЕМИ ПІДТРИМКИ УПРАВЛІННЯ ДІЯЛЬНІСТЮ ПІДПРИЄМСТВА	424
Корецький В.А. ПРОЄКТУВАННЯ ТА РЕАЛІЗАЦІЯ ВЕБІНФОРМАЦІЙНОЇ ПІДСИСТЕМИ МОНІТОРИНГУ ІНТЕРНЕТ-ТРАФІКУ	425
Латченко І. В. ТА РЕАЛІЗАЦІЯ ВЕБЗАСТОСУНКУ ОРГАНІЗАЦІЇ ДИСТАНЦІЙНИХ КУРСІВ НЕФОРМАЛЬНОЇ ОСВІТИ	426
Хоменко Р.М. ПРОЄКТУВАННЯ ІНТЕРФЕЙСУ КОРИСТУВАЧА ЧАТ-БОТА ДЛЯ ЗРУЧНИХ ТА ЕФЕКТИВНИХ ОНЛАЙН-ЗАМОВЛЕНЬ ЕЛЕКТРОНІКИ	427
Блаженко Д. О. ПРОБЛЕМИ ПРОЄКТУВАННЯ БАЗИ ДАНИХ ПІДСИСТЕМИ ВЕДЕННЯ ЗАМОВЛЕНЬ ДЛЯ ВИГОТОВЛЕННЯ СІЛЬСЬКОГОСПОДАРСЬКОЇ ПРОДУКЦІЇ	428
Ковальов Р. С. ПРАКТИЧНЕ ПРОГРАМУВАННЯ ГЕНЕРАТОРІВ 3D КОНТЕНТУ НА ПРИКЛАДІ ГЕНЕРАТОРУ МЕША РЕЛЬЄФУ МІСЦЕВОСТІ	429
Ковальов Р. С. ПРАКТИЧНЕ ПРОГРАМУВАННЯ ГЕНЕРАТОРІВ 3D КОНТЕНТУ НА ПРИКЛАДІ ГЕНЕРАТОРУ МЕША РЕЛЬЄФУ МІСЦЕВОСТІ	430
Руденко О. А. МІНІМІЗАЦІЯ ЗАТРИМОК У ВЕБДОДАТКАХ ВИКОРИСТОВУЮЧИ ГРАНИЧНІ ТЕХНОЛОГІЇ	431
Сіденко А. С. ПРОБЛЕМИ РЕАЛІЗАЦІЇ ПІДСИСТЕМИ ОБЛІКУ ПРИРОДООХОРОНИ ЗАХОДІВ	432
Терещенко С.С. АВТОМАТИЗАЦІЯ ОБЛІКОВИХ ПРОЦЕСІВ: КЛЮЧ ДО ЕФЕКТИВНОГО УПРАВЛІННЯ В АТЕЛЬЄ З ВИГОТОВЛЕННЯ ОДЯГУ	433
Михайлюк О.С. ОГРУНТУВАННЯ НЕОБХІДНОСТІ РОЗРОБКИ ВЕБ-ОРІЄНТОВАНОЇ ПІДСИСТЕМИ ПІДТРИМКИ ОБЛІКОВИХ ПРОЦЕСІВ РЕГІОНАЛЬНОГО ІНТЕРНЕТ-ПРОВАЙДЕРА	434
Олійник А.В. ПРОЄКТУВАННЯ ТА РЕАЛІЗАЦІЯ ПІДСИСТЕМИ ОБЛІКУ ПАЛИВНО-МАСТИЛЬНИХ МАТЕРІАЛІВ В ТЕРИТОРІАЛЬНІЙ ГРОМАДІ	435
Чубун М.П. АНАЛІЗ ТА ВИЗНАЧЕННЯ ВИМОГ ДО СУЧАСНОГО ВЕБ-САЙТУ	436
Дубінчин В. С. ВИЗНАЧЕННЯ ФУНКЦІОНАЛЬНИХ ВИМОГ ДО ІНФОРМАЦІЙНОЇ ПІДСИСТЕМИ ОЦІНЮВАННЯ ПРАЦЕЗДАТНОСТІ КОМП'ЮТЕРИЗОВАНИХ РОБОЧИХ МІСЦЬ	437
Санжаков А. А. НОВИЙ ПІДХІД РОЗРОБКИ ДИЗАЙНУ ВЕБ-ЗАСТОСУНКІВ – MATERIAL 3	438
Федірко Д.Д. ПРОБЛЕМИ ПРОЄКТУВАННЯ ВЕБ-ЗАСТОСУНКІВ ОЦІНКИ ТАРИФНИХ ПРОПОЗИЦІЙ МОБІЛЬНИХ ОПЕРАТОРІВ УКРАЇНИ	439
Федірко Д.Д. РОЗРОБКА ВЕБ-ЗАСТОСУНКУ ДЛЯ АНАЛІЗУ МОБІЛЬНИХ ОПЕРАТОРІВ УКРАЇНИ	440
Маслак О. М. ДЕРЖАВНЕ РЕГУЛЮВАННЯ ПРОЦЕСІВ АГРОПРОМИСЛОВОГО ВИРОБНИЦТВА НА РЕГІОНАЛЬНОМУ РІВНІ (НА ПРИКЛАДІ СУМСЬКОЇ ОБЛАСТІ)	441
Горбань С.В. ІМІДЖ ПІДПРИЄМСТВА: ВПЛИВ, ФОРМУВАННЯ ТА ОЦІНКА	442
Заболотна А.І. НОРМАТИВНО-ПРАВОВЕ ЗАБЕЗПЕЧЕННЯ ЕКОНОМІЧНОГО РОЗВИТКУ ДРУЖБІВСЬКОЇ ТЕРИТОРІАЛЬНОЇ ГРОМАДИ	443
Гулиніна О.Ю. ШЛЯХИ ОПТИМІЗАЦІЇ ОРГАНІЗАЦІЙНОЇ СТРУКТУРИ УПРАВЛІННЯ ЗАКЛАДУ ОХОРОНИ ЗДОРОВ'Я	444
Сущенко В. О. ПРОЦЕС РОЗВИТКУ ПЕРСОНАЛУ, ЯК ЧИННИК КОНКУРЕНТОЗДАТНОСТІ ПІДПРИЄМСТВА В РИНКОВИХ УМОВАХ	445
Каплун В.М. УПРАВЛІННЯ СІЛЬСЬКОГОСПОДАРСЬКИХ ПІДПРИЄМСТВ ІЗ ЗАСТОСУВАННЯ ПРОГРАМНОГО КОНТРОЛЮ І МОНІТОРИНГУ	446
Бондар А.В. ТЕОРЕТИКО-МЕТОДИЧНІ ПІДХОДИ ДО ВПРОВАДЖЕННЯ ТА ОЦІНКИ СТРАТЕГІЧНОГО УПРАВЛІННЯ РОЗВИТКОМ СІЛЬСЬКОГОСПОДАРСЬКИХ ПІДПРИЄМСТВ	447
Романенко Т.А. СУЧАСНІ НАПРЯМИ ІНТЕНСИФІКАЦІЇ ВИРОБНИЦТВА ПРОДУКЦІЇ РОСЛИННИЦТВА В КОНТЕКСТІ ПРОДОВОЛЬЧОЇ БЕЗПЕКИ	448
Мішенін С. В., Ярова І. Є. ОРГАНІЗАЦІЙНО ЕКОНОМІЧНИЙ МЕХАНІЗМ ЗАБЕЗПЕЧЕННЯ СТАЛОГО ПРОСТОРОВНОГО РОЗВИТКУ ЛІСОВОГО ГОСПОДАРСТВА	449
Hu Guannan CRISIS MANAGEMENT THEORY	450
Zhang Liming HOW DOES BYD COMPANY IMPROVE ITS COMPETITIVENESS?	451
Zhao Jing THE INFLUENCE OF CORPORATE CULTURE ON EMPLOYEE SATISFACTION	452
Cui Xiaoping DEVELOPMENT PATH OF ENTERPRISE MANAGEMENT IN THE ERA OF ARTIFICIAL INTELLIGENCE	453

НОВИЙ ПІДХІД РОЗРОБКИ ДИЗАЙНУ ВЕБ-ЗАСТОСУНКІВ – MATERIAL 3

Санжаков А. А., студ. 4 курсу ФЕІМ, спец. «Інформаційні системи та технології»
Науковий керівник: к.т.н., доц. А.В. Толбатов
Сумський НАУ

Material 3 – це нова дизайнерська система від Google, що пропонує оновлений та витончений спосіб створення інтерфейсів користувача. Це не просто оновлення дизайну, а нова філософія створення мобільних та веб інтерфейсів, що робить їх не лише красивими, але й доступними, адаптивними та динамічними. Material 3 ґрунтується на трьох основних принципах:

Динамічний колір:

1. Динамічні теми генеруються на основі вашого фонового зображення або вибраної кольорової палітри.

2. Теми генерують 3 ключові кольори: основний, додатковий та нейтральний.

3. Ви можете налаштувати динамічну тему, змінюючи фон, акцентні кольори або вибираючи з набору заздалегідь сформованих тем.

Виразність:

1. **Material 3** використовує чітку типографіку з використанням шрифту Google Sans.

2. Шрифт Google Sans доступний у різних стилях та товщинах для кращої читабельності.

3. Тіні та висота шару використовуються для створення візуальної ієрархії та глибини.

Адаптивність:

1. **Material 3** адаптується до будь-якого розміру екрана або пристрою.

2. Компоненти інтерфейсу користувача автоматично змінюють розмір та макет для кращого відображення на будь-якому пристрої.

3. Ви можете використовувати модульну систему розкладки Material 3 для створення адаптивних та змінних макетів.

Доступність: **Material 3** розроблений з урахуванням доступності та відповідає рекомендаціям WCAG 2.1.

Інтернаціоналізація: **Material 3** доступний різними мовами, включаючи українську

Нова кольорова палітра: **Material 3** включає оновлену кольорову палітру, яка ґрунтується на природних кольорах та відтінках. Ця палітра допомагає створити інтерфейси, які є естетично приємними та зручними для очей.

Нова типографіка: **Material 3** використовує оновлену типографіку, яка є чіткою, читабельною та доступною. Ця типографіка допомагає створити інтерфейси, які легко читати та розуміти.

Нові компоненти: **Material 3** включає нові компоненти інтерфейсу користувача, такі як кнопки, картки та меню. Ці компоненти допомагають створити інтерфейси, які є сучасними та інтерактивними.

Переваги використання Матеріалу 3:

1. Створює красиві та унікальні інтерфейси

2. Динамічні теми та виразний дизайн допомагають створити інтерфейси, які приємно використовувати.

3. Підвищує доступність

4. **Material 3** розроблений з урахуванням доступності та відповідає рекомендаціям WCAG 2.1.

5. **Економить час та зусилля** **Material 3** пропонує готові компоненти та інструменти, які допомагають прискорити процес розробки.

6. Створює адаптивні інтерфейси

7. **Material 3** автоматично адаптується до будь-якого розміру екрану або пристрою.

Література

1. "Android Developers" – офіційний ресурс від Google, який містить документацію та приклади для розробки інтерфейсу мобільних застосунків на базі Material 3. <https://developer.android.com/>

ДОДАТОК Г

ЛІСТИНГИ ПРОГРАМ

Лістинг основних файлів керування роботою мобільного додатку:

Main_activity.kt (package : main_av)

```
class MainActivity : ComponentActivity() {
    private val viewModel: MainViewModel by viewModels()
    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        val dataBase = DataBase.getDB(this)
        lifecycleScope.launch {
            dataBase.run {
                for (i in NaturalPhenomenon.naturalPhenomenon) {
                    getDao().insertTableNaturalPhenomenon(
                        EntityNaturalPhenomenon(
                            key = i.number,
                            name = i.name,
                            image = i.image,
                            description = i.description,
                            naturalDisasters = i.naturalDisasters,
                            imageMain = i.imageMain,
                            number = i.number
                        )
                    )
                }
            }
            for (i in MainAttributes.mainAttributes){
                getDao().insertTableMainAttributes(
                    EntityMainAttributes(
```

```

        key = i.number,
        name = i.name,
        firstAttributes= i.firstAttributes,
        secondAttributes = i.secondAttributes,
        thirdAttributes = i.thirdAttributes,
        mainOneNaturalPhenomena = i.mainOneNaturalPhenomena,
        number = i.number,
        nameFirstAttributes = i.nameFirstAttributes,
        nameSecondAttributess = i.nameSecondAttributess,
        nameThirdAttributes = i.nameThirdAttributes
    )
)
}
for (i in HistoricalNaturalPhenomenon.historicalNaturalPhenomenon){
    getDao().insertTableHistoricalNP(
        EntityHistoricalNaturalPhenomenon(
            key = i.number,
            name = i.name,
            image = i.image,
            description = i.historically,
            number = i.number
        )
    )
}
for (i in PlaceOfOccurrence.placeOfOccurrence){
    getDao().insertTablePOO(
        EntityPlaceOfOccurrence(
            key = i.number,
            name = i.name,

```

```

        place = i.place,
        number = i.number
    )
)
}
}
}
setContent {
    val listNaturalPhenomenon by dataBase.getDao()
        .getAllFromDBNaturalPhenomenon().collectAsState(initial = emptyList())
    val listMainAttributes by dataBase.getDao()
        .getAllFromDBMainAttributes().collectAsState(initial = emptyList())

    val listHistoricalNaturalPhenomenon by dataBase.getDao()
        .getAllFromDBHistoricalNP().collectAsState(initial = emptyList())
    val listPlaceOfOccurrence by dataBase.getDao()
        .getAllFromDBPlaceOfOccurrence().collectAsState(initial = emptyList())
    val navHostController = rememberNavController()
    NavigationHost(
        navController = navHostController,
        naturalPhenomenon = listNaturalPhenomenon,
        viewModel = viewModel,
        mainAttributes = listMainAttributes,
        historicalNaturalPhenomenon = listHistoricalNaturalPhenomenon,
        placeOfOccurrence = listPlaceOfOccurrence
    )
}
}
}

```

```
}
```

Dao.kt (pacage data)

```
@androidx.room.Dao
interface Dao {
    @Insert(onConflict = OnConflictStrategy.IGNORE)
    suspend fun insertTableNaturalPhenomenon(naturalPhenomenon:
EntityNaturalPhenomenon )

    @Query("Select * From naturalPhenomenon")
    fun getAllFromDBNaturalPhenomenon():
Flow<List<EntityNaturalPhenomenon>>

    @Insert(onConflict = OnConflictStrategy.IGNORE)
    suspend fun insertTableMainAttributes(mainAttributes: EntityMainAttributes)

    @Query("Select * From mainAttributes")
    fun getAllFromDBMainAttributes(): Flow<List<EntityMainAttributes>>

    @Insert(onConflict = OnConflictStrategy.IGNORE)
    suspend fun insertTableHistoricalNP(historicalNaturalPhenomenon:
EntityHistoricalNaturalPhenomenon)

    @Query("Select * From historicalPhenomenon")
    fun getAllFromDBHistoricalNP():
Flow<List<EntityHistoricalNaturalPhenomenon>>

    @Insert(onConflict = OnConflictStrategy.IGNORE)
    suspend fun insertTablePOO(placeOfOccurrence: EntityPlaceOfOccurrence)

    @Query("Select * From placeOfOccurrence")
    fun getAllFromDPlaceOfOccurrence(): Flow<List<EntityPlaceOfOccurrence>>

}
```

Database.kt(pacage : data)

```
@Database(entities = [
    EntityNaturalPhenomenon::class,
    EntityHistoricalNaturalPhenomenon::class,
    EntityMainAttributes::class,
    EntityPlaceOfOccurrence::class
], version = 1)
```

```

abstract class DataBase: RoomDatabase() {
    abstract fun getDao(): Dao
    companion object{
        @Volatile
        private var INSTANCE :DataBase? = null
        fun getDB(context: Context):DataBase {
            return INSTANCE ?: synchronized(this) {
                val instance = Room.databaseBuilder(
                    context = context.applicationContext,
                    DataBase::class.java,
                    "natural_phenomenon"
                ).build()
                INSTANCE = instance
            }
        }
    }
}

```

Entity.kt(package : data)

```

@Entity(tableName = "naturalPhenomenon")
data class EntityNaturalPhenomenon (
    @PrimaryKey(autoGenerate = true) val key: Int,
    @ColumnInfo("name") val name : String,
    @ColumnInfo(name = "image") val image: Int,
    @ColumnInfo(name = "description") val description: String,
    @ColumnInfo(name = "naturalDisasters") val naturalDisasters:Boolean,
    @ColumnInfo(name = "imageMain") val imageMain:Int,
    @ColumnInfo(name = "number") val number: Int,
)
@Entity(tableName = "mainAttributes")
data class EntityMainAttributes(
    @PrimaryKey(autoGenerate = true) val key: Int,
    @ColumnInfo(name = "name") val name:String,
    @ColumnInfo(name = "firstAttributes") val firstAttributes: Int,
    @ColumnInfo(name = "secondAttributes") val secondAttributes: Int,
    @ColumnInfo(name = "thirdAttributes") val thirdAttributes: Int,
    @ColumnInfo(name = "nameFA") val nameFirstAttributes: String,
    @ColumnInfo(name = "nameSA") val nameSecondAttributess: String,
    @ColumnInfo(name = "nameTA") val nameThirdAttributes: String,
)

```

```

    @ColumnInfo(name = "mainOneNaturalPhenomenon") val
mainOneNaturalPhenomena: Int,
    @ColumnInfo(name = "number") val number: Int
)
@Entity(tableName = "historicalPhenomenon")
data class EntityHistoricalNaturalPhenomenon (
    @PrimaryKey(autoGenerate = true) val key: Int,
    @ColumnInfo("name") val name : String,
    @ColumnInfo(name = "image") val image: Int,
    @ColumnInfo(name = "description") val description: String,
    @ColumnInfo(name = "number") val number: Int
)

```

```

@Entity(tableName = "placeOfOccurrence")
data class EntityPlaceOfOccurrence(
    @PrimaryKey(autoGenerate = true) val key: Int,
    @ColumnInfo("name") val name : String,
    @ColumnInfo(name = "place") val place: String,
    @ColumnInfo(name = "number") val number: Int
)

```

NavHost.kt(package : navigation)

```

@Composable
fun NavigationHost(
    navController: NavHostController,
    viewModel: MainViewModel,
    naturalPhenomenon: List<EntityNaturalPhenomenon>,
    mainAttributes: List<EntityMainAttributes>,
    historicalNaturalPhenomenon: List<EntityHistoricalNaturalPhenomenon>,
    placeOfOccurrence: List<EntityPlaceOfOccurrence>
){
    NavHost(navController = navController, startDestination =Screen.Main.screen )
{
    composable(Screen.Main.screen){
        MainContetn(
            naturalPhenomenon = naturalPhenomenon,
            navHostController = navController,
            viewModel = viewModel
        )
    }
    composable(Screen.OneElement.screen){
        // OnePhenomenal(phenomena = viewModel.naturalElemet)
    }
}
}

```

```

        MainScaffold(
            naturalPhenomenon = naturalPhenomenon[viewModel.holdingForNP],
            mainAttributes = mainAttributes[viewModel.holdingForNP],
            historicalNaturalPhenomenon =
historicalNaturalPhenomenon[viewModel.holdingForNP],
            placeOfOccurrence = placeOfOccurrence[viewModel.holdingForNP]
        )
    }
    /* composable("oneElement"){

    }*/

}
}

```

OneAttributes.kt(pacage : presentashion)

```

@Composable
fun OneAttributes(
    mainAttributes: EntityMainAttributes
) {

    Card(
        Modifier
            .fillMaxSize()
            //height(150.dp)
            .padding(top = 200.dp, bottom = 100.dp, start = 50.dp, end = 50.dp),
        colors = CardDefaults.cardColors(AttributesMain)
        //paint(painter = painterResource(id = R.drawable.storm_back), contentScale =
ContentScale.FillBounds)
    ) {
        Row(horizontalArrangement = Arrangement.SpaceBetween) {
            Column(
                modifier = Modifier
                    .fillMaxHeight()
                    .width(120.dp)
                    .padding(start = 15.dp),
                verticalArrangement = Arrangement.SpaceAround,
                horizontalAlignment = Alignment.Start
            ) {
                Row (verticalAlignment = Alignment.CenterVertically,
                    horizontalArrangement = Arrangement.spacedBy(10.dp)){

```

```

Image(
    modifier = Modifier
        .size(width = 50.dp, height = 50.dp),
    painter = painterResource(id = mainAttributes.firstAttributes),
    contentDescription = ""
)
Text(text = mainAttributes.nameFirstAttributes, fontSize = 17.sp )
}
Text(text = "+", fontWeight = FontWeight.W400, fontSize = 35.sp)
Row (verticalAlignment = Alignment.CenterVertically,
    horizontalArrangement = Arrangement.spacedBy(10.dp)){
    Image(
        modifier = Modifier
            .size(width = 50.dp, height = 50.dp),
        painter = painterResource(id = mainAttributes.secondAttributes),
        contentDescription = ""
    )
    Text(text = mainAttributes.nameSecondAttributess, fontSize = 17.sp )
}
Text(text = "+", fontWeight = FontWeight.W400, fontSize = 35.sp)
Row (verticalAlignment = Alignment.CenterVertically,
    horizontalArrangement = Arrangement.spacedBy(10.dp)){
    Image(
        modifier = Modifier
            .size(width = 50.dp, height = 50.dp),
        painter = painterResource(id = mainAttributes.thirdAttributes),
        contentDescription = ""
    )
    Text(text = mainAttributes.nameThirdAttributes, fontSize = 17.sp )
}
/* Text(text = "+", fontWeight = FontWeight.W400, fontSize = 35.sp)
Image(
    modifier = Modifier
        .size(width = 50.dp, height = 50.dp),
    painter = painterResource(id = mainAttributes.secondAttributes),
    contentDescription = ""
)
Text(text = "+", fontWeight = FontWeight.W400, fontSize = 35.sp)
Image(
    modifier = Modifier
        .size(width = 50.dp, height = 50.dp),
    painter = painterResource(id = mainAttributes.thirdAttributes),
    contentDescription = ""
)

```

```

        )*/
    }
    Row(
        modifier = Modifier
            .fillMaxSize(),
            //.width(100.dp)
            //.padding(start = 5.dp),
        verticalAlignment = Alignment.CenterVertically,
        horizontalArrangement = Arrangement.SpaceAround
    ) {
        Text(text = "=", fontWeight = FontWeight.W400, fontSize = 45.sp)
        //Spacer(modifier = Modifier.padding(start = 10.dp))
        Image(modifier = Modifier
            .size(width = 80.dp, height = 80.dp),
            painter = painterResource(id =
mainAttributes.mainOneNaturalPhenomena), contentDescription = "")
    }
}
}
}

```

OneElement.kt(pacage : presentashion)

@OptIn(ExperimentalMaterial3Api::class)

@Composable

fun OneElement(

navHostController: NavHostController,

naturalPhenomenon: EntityNaturalPhenomenon,

viewModel: MainViewModel,

){

Column(//TODO(" bed(")

modifier = Modifier

.size(90.dp)

/*constrainAs(topmBar) {

start.linkTo(mainContent.start)

end.linkTo(mainContent.end)

*/,

verticalArrangement = Arrangement.spacedBy(1.dp), //spacedBy(1.dp)

horizontalAlignment = Alignment.CenterHorizontally

) {

Card (

// modifier = Modifier.shake(shakeController),

colors = CardDefaults.cardColors(Color.Unspecified),

onClick = {

viewModel.stHoldingForNP(naturalPhenomenon.key-1)

```

        navHostController.navigate(Screen.OneElement.screen)
        //HoldingNP(nP = naturalPhenomenon.number) //TODO(work ?)
        Log.d("AAA","${ viewModel.holdingForNP}")
        // viewModel.setNaturalElemet(naturalPhenomenon)
        /* shakeController.shake(
            ShakeConfig(2, rotateZ = 30f)
        )*/
    }
){
    Image(
        painter = painterResource(id = naturalPhenomenon.image),
        contentDescription = "",
    )
}

```

```

        Text(text = naturalPhenomenon.name, fontSize = 18.sp, fontFamily =
FontFamily.Default,
            fontWeight = FontWeight.W500, color = TestFontColor, maxLines = 1,
overflow = TextOverflow.Ellipsis)
    }
}

```