

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
НАЦІОНАЛЬНИЙ АВІАЦІЙНИЙ УНІВЕРСИТЕТ
Факультет кібербезпеки, комп'ютерної та програмної інженерії
Кафедра комп'ютерних інформаційних технологій

ДОПУСТИТИ ДО ЗАХИСТУ
Завідувач випускової кафедри
_____ А.С. Савченко
« » _____ 20 р

ДИПЛОМНИЙ ПРОЕКТ

(ПОЯСНЮВАЛЬНА ЗАПИСКА)

ВИПУСКНИКА ОСВІТНЬОГО СПУПЕННЯ «БАКАЛАВР»

**Тема: «Додаток аналізу даних захворюваності коронавірусною хворобою
COVID-19 в країнах світу на IOS/ANDROID»**

Виконавець: студент УС-411 Ковальов Станіслав Святославович
(студент, група, прізвище, ім'я, по батькові)

Керівник: к.т.н., доцент Райчев Ігор Едуардович
(науковий ступень, вчене звання, прізвище, ім'я, по батькові)

Нормоконтролер: ст.викл.Шевченко О.П.
(П.І.Б.)

(підпис)

КИЇВ 2021

НАЦІОНАЛЬНИЙ АВІАЦІЙНИЙ УНІВЕРСИТЕТ

Факультет кібербезпеки, комп'ютерної та програмної інженерії

Кафедра комп'ютерних інформаційних технологій

Освітній ступінь: Бакалавр

Галузь знань, спеціальність, спеціалізація: 12 “Інформаційні технології”,
122 “Комп'ютерні науки”, “Інформаційні управляючі системи та технології”

ЗАТВЕРДЖУЮ

Завідувач кафедри

А.С. Савченко

“ _____ ” _____ 2021 р.

ЗАВДАННЯ

на виконання дипломного проекту студента

Ковальов Станіслав Святославович

1. Тема проекту: «Додаток аналізу даних захворюваності коронавірусною хворобою *COVID-19* в країнах світу на *IOS/ANDROID*» затверджена наказом ректора від «22» квітня 2021р. № 636/ст.
2. Термін виконання роботи: з 11.05.2021 до 14.06.2021р.
3. Вихідні дані до роботи: Документація *React Native*; Документація *JavaScript*; принципи створення додатків на *IOS/ANDROID*; Аналоги додатку аналізу даних захворюваності коронавірусною хворобою *COVID-19*.
4. Зміст пояснювальної записки (перелік питань, що підлягають розробці): аналіз задачі розробки мобільного додатку, інструменти створення мобільного додатку, мобільний додаток аналізу захворюваності коронавірусною хворобою *COVID-19* в країнах світу.
5. Перелік обов'язкового графічного матеріалу: прототипи додатку, аналоги додатку аналізу даних захворюваності коронавірусною хворобою, інтерфейс додатку аналізу даних захворюваності коронавірусною хворобою.

КАЛЕНДАРНИЙ ПЛАН

	Етапи виконання дипломної роботи	Термін виконання етапів	Примітка
1	Ознайомлення з постановкою задачі та вивчення літератури.	10.05.21.-12.05.21.	
2	Розробка та затвердження плану дипломного проекту.	13.05.21.	
3	Проведення консультації з науковим керівником щодо створення першого розділу.	14.05.21.- 18.05.21.	
4	Розгляд основних аналогів додатку захворюваності – створення прототипів на їх основі.	19.05.21.-22.05.21.	
5	Проаналізувати методи створення додатків за допомогою React Native.	23.05.21.-27.05.21.	
6	Загальне редагування, написання висновків та оформлення пояснювальної записки.	28.05.21.-31.05.21.	
7	Проходження нормо-контролю, перепліт пояснювальної записки.	01.06.21.-07.06. 21.	
8	Розробка тексту доповіді. Оформлення графічного матеріалу для презентації.	07.06.21.-09.06.21.	
9	Підготовка до захисту та попередній захист дипломного проекту на випусковій кафедрі дипломного проекту.	10.06.21.-14.06.21.	

Студент

(*Ковальов С.С.*)

Керівник дипломної роботи

(*Райчев І.Е.*)

РЕФЕРАТ

Пояснювальна записка до дипломної роботи на тему «Мобільний додаток аналізу даних захворюваності коронавірусною хворобою *COVID-19* в країнах світу на *IOS/ANDROID*»: 57 с., 47 рис., 17 літературних джерел.

Об'єкт дослідження – процес створення додатку аналізу даних коронавірусною хворобою *COVID-19*.

Предмет дослідження – мобільний додаток на *IOS/ANDROID* аналізу даних захворюваності коронавірусною хворобою *COVID-19*.

Мета дипломного проекту – розробити мобільний додаток на *IOS/ANDROID* аналізу даних захворюваності коронавірусною хворобою *COVID-19*.

Методи проектування – теоретичне ознайомлення з існуючими аналогами, використання здобутих знань для розробки додатку.

Результати дипломного проекту можна розділити на дві групи: теоретичні та практичні. Теоретичні результати можна використовувати при розробці додатку чи ознайомлення з темою розробки додатків, у якому постає проблема аналізу даних, практичні – використовувати в якості готового рішення питання додатку на *IOS/ANDROID* аналізу даних захворюваності коронавірусною хворобою *COVID-19*.

REACT NATIVE, COVID-19, JAVASCRIPT, FIGMA, API, ANDROID/IOS, ДОДАТОК АНАЛІЗУ ДАНИХ, VISUAL STUDIO CODE

ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ ТА СКОРОЧЕНЬ	6
ВСТУП.....	7
РОЗДІЛ 1. АНАЛІЗ ЗАДАЧІ РОЗРОБКИ МОБІЛЬНОГО ДОДАТКУ	9
1.1. Задача розробки мобільного додатку аналізу даних захворюваності коронавірусною хворобою <i>COVID-19</i>	9
1.1.1. <i>COVID-19</i>	10
1.2. Огляд існуючих рішень.....	12
1.3. Порівняння існуючих рішень із запропонованим	20
1.3.1. Створення прототипу додатку в графічному редакторі <i>Figma</i>	21
1.3.2. Створення дизайну додатку	28
1.4. Висновки до розділу.....	29
РОЗДІЛ 2. ІНСТРУМЕНТИ СТВОРЕННЯ ДОДАТКУ	30
2.1. <i>IDE: Visual Studio Code</i>	30
2.2. Мова програмування <i>JavaScript</i>	32
2.3. <i>React Native</i>	33
2.4. <i>API Coronavirus map</i>	38
2.5. Висновки до розділу.....	39
РОЗДІЛ 3. ДОДАТОК АНАЛІЗУ ДАНИХ ЗАХВОРИВАНОСТІ КОРОНАВІРУСНОЮ ХВОРОБОЮ <i>COVID-19</i>	40
3.1. Інтерфейс додатку	40
3.2. Тестування додатку	52
3.3. Висновки до розділу.....	53
ВИСНОВКИ	54
СПИСОК БІБЛІОГРАФІЧНИХ ПОСИЛАНЬ ВИКОРИСТАНИХ ДЖЕРЕЛ	55

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ ТА СКОРОЧЕНЬ

API – програмний інтерфейс

IDE – інтегроване середовище розробки

COVID-19 – хвороба, що набула поширення з кінця 2019 року в Китаї й охопила весь світ

JS – мова програмування *JavaScript*

.apk – файл додатку для *Android*

.ipa – файл додатку для *iPhone*

ВСТУП

Швидким темпом в останні десятиліття розвиваються смартфони. Щодня операційні системи модернізуються, тобто з'являються нові функції та можливості для реалізації більш досконалих додатків з використанням нових технологій розробки. Від розроблення першої операційної системи для мобільного телефону до першого мобільного додатку пройшло не так багато часу – проте, їх можливості та функціональність були досить низькими.

Кожна наступна версія операційної системи збільшує кількість інструментів, що використовуються для розробки. Завдяки цьому відбувається збільшення областей застосування продуктів, що були створені, а також це дає змогу розширити інструментарій додатків, що в кінці призвело до створення додатків, що можуть бути використані в різних сферах нашого життя – на сьогоднішній день існує велика кількість додатків, котрі можна об'єднати в категорії, наприклад: плеєри, навігатори, Інтернет-магазини, додатки для соціальних мереж, фото- та відео-редактори та інші – їх застосовують будь-де.

Сьогодні майже кожен банк має клієнтський додаток, котрий дозволяє спростити взаємодію фізичних та юридичних осіб, що являються клієнтами. Більш того, на даний момент є банки, котрі взаємодіють з клієнтами лише за допомогою мобільного додатку у всесвітній павутині – наприклад, *Monobank*; інша область, котра майже повністю використовує для своєї роботи мобільні додатки – це пасажирські перевезення – більшість служб таксі мають мобільний додаток, за допомогою якого клієнт може, по-перше, замовити авто й розрахуватись, по-друге, спостерігати за рухом автомобіля на карті за допомогою трекінгу, по-третє, дати оцінку й залишити відгук сервісу – це все можливо за допомогою додатку в телефоні.

Крім цього сфера продовольчих товарів також використовує мобільні додатки – за допомогою них можна переглянути асортимент товарів та обрати потрібний, перевірити чи переконатись в наявності програм лояльності, перевірити наявність актуальних знижок чи замовити той чи інший товар в режимі

онлайн. Після замовлення товару клієнт користується послугами доставки замовлень, котрі також використовують власні мобільні додатки – що дають змогу переконатись в котрий термін буде проведена доставка, хто доставить замовлення, а також яка сума й швидкість доставки та інші показники й дані. Як бачимо, області застосування мобільних додатків широкі та збільшуються майже щодня. А отже, однією з найважливіших категорій є охорона здоров'я людини.

Збереження власного життя та життя своїх рідних – це основне завдання кожної людини. 21 століття характеризується не лише епохою технологій, а й новим захворюванням – *COVID-19*, котра вплинула на життя всього світу.

Хто володіє інформацією – той володіє світом, проте, можна сказати трішки інакше: хто володіє інформацією – той знає як захистити своє життя. Питання охорони здоров'я на даний момент стоїть перед всім людством, а отже тема розробки мобільного додатку на *IOS/ANDROID* аналізу даних захворюваності коронавірусною хворобою *COVID-19* в країнах світу є актуальною як ніколи раніше

РОЗДІЛ 1

АНАЛІЗ ЗАДАЧІ РОЗРОБКИ МОБІЛЬНОГО ДОДАТКУ

Нижче буде розглянута та проаналізована інформація щодо *COVID-19*, аналогів додатку аналізу даних захворюваності коронавірусною хворобою *COVID-19* в країнах світу, а також зроблено висновки щодо задачі розробки мобільного додатку на основі аналогів.

1.1. Задача розробки мобільного додатку аналізу даних захворюваності коронавірусною хворобою *COVID-19*

2019 рік став ключовим у сфері охорони здоров'я людини, людство до сих пір не вилікувалось повністю від *COVID-19*, тому потреба в додатку аналізу даних захворюваності коронавірусною хворобою в різних країнах світу є актуальною – останнім часом такі додатки є доволі популярними, адже тенденція захворюваності не зменшується, а навпаки – щодня збільшується, а отже, слід розуміти, яка ситуація на даний момент. Це ми можемо дізнатись за допомогою ведення статистики, що дозволить проаналізувати рівень розповсюдження хвороби в різні періоди часу. Ці додатки дозволяють отримувати нові знання в інтерактивному форматі, використовуючи для цього мобільний телефон. Вони містять певні матеріали у вигляді відео та аудіо записів, текстових документів. Всі ці матеріали зазвичай мають структурований формат. Вони в свою чергу також об'єднуються в певні категорії за практичною цінністю. Додатки такого типу розраховані на категорію користувачів, що користуються ними для отримання повної інформації щодо динаміки захворюваності у певній країні.

Кафедра КІТ				НАУ 21 12 05 000 ПЗ			
Виконав	Ковальов С.С.			Аналіз задачі розробки мобільного додатку	Літера	Аркуш	Аркушів
Керівник	Райчев І.Е.				Д	9	21
Консульт.					УС-411Б 122		
Норм. контр.	Шевченко О.П.						
Зав. Каф.	Савченко А.С.						

Отримуємо висновок, що необхідно дослідити можливість створення додатку для аналізу захворюваності коронавірусною хворобою *COVID-19* в країнах світу з використанням *React Native*.

Отже, основною метою дослідження є створення мобільного додатку, що має виконувати наступні функції:

- відображення інформації рівня захворюваності за заданою структурою;
- забезпечення користувача актуальними даними;
- відображення даних останнього оновлення;
- структуризація статистичних даних;
- зручний та зрозумілий графічний інтерфейс.

1.1.1. COVID-19

Що таке *COVID-19*?

COVID-19 - захворювання, що викликається новим коронавірусів, який називається *SARS-CoV-2*. ВООЗ вперше дізналася про цей новий вірус 31 грудня 2019 року, отримавши повідомлення про групу випадків захворювання «вірусною пневмонією» в місті Ухань, Китайської Народної Республіки [1].

Які симптоми *COVID-19*?

До найбільш поширених симптомів *COVID-19* відносяться:

- лихоманка;
- сухий кашель;
- стомлюваність.

До іншим, менш поширених симптомів, які зустрічаються у ряду пацієнтів, відносяться:

- втрата нюху або смакових відчуттів;
- закладеність носа;
- кон'юнктивіт (або почервоніння очей);
- біль в горлі;
- головний біль;
- біль в м'язах або суглобах;

- різні види висипань на шкірі;
- нудота або блювота;
- діарея;
- озноб або запаморочення.

Важкий перебіг *COVID-19* проявляється наступними симптомами:

- задишка;
- втрата апетиту;
- сплутаність свідомості;
- наполегливі болі або відчуття стискання грудної клітки;
- висока температура тіла (вище 38 ° C).

До інших, менш поширених симптомів відносяться:

- дратівливість;
- сплутаність свідомості;
- зниження рівня свідомості (іноді супроводжується судомами);
- тривожність;
- пригнічений стан;
- порушення сну;
- більш важкі і рідкісні неврологічні ускладнення, такі як інсульт, запальне ураження мозку, делірій і ураження нервів.

Люди будь-якого віку, які поряд з появою лихоманки і / або кашлю помічають погіршене дихання або задишку, біль або відчуття стискання грудної клітини, порушення мови або рухові розлади, повинні негайно звернутися за медичною допомогою. З тим щоб отримати направлення у відповідний заклад, по можливості слід в першу чергу зв'язатися по телефону з постачальником послуг охорони здоров'я, звернутися на «гарячу лінію» або зателефонувати в медичну організацію.

Як можемо бачити, *COVID-19* – це серйозне захворювання, що потребує моніторингу, а отже розробка мобільного додатку для аналізу даних захворюваності в світі – є необхідним рішенням в наш час.

1.2. Огляд існуючих рішень

Розглянемо додатки для аналізу захворюваності коронавірусною хворобою та аналогічні, а також їх особливості, проаналізуємо реалізацію додатків та їх використання.

На даний момент в мережі Інтернет наявна велика кількість додатків, що характеризують рівень захворюваності коронавірусною хворобою, сюди можна віднести наступні додатки: *Covid19 | Info you need to know*, *COVID-19 Symptom Study*, *COVID Shield Mobile App*, *Covid-19-APP*, *Coronavirus (COVID-19) Dashboard*, *COVID-19 tracker*, *Coronavirus Tracker App*, *COVID19-Tracker*, *Vete Corona* – частина з них розроблені за допомогою *React Native*.

Covid-19 | Info you need to know

Доволі простий трекер, що використовується для відображення інформації рівня захворюваності у світі та країнах.

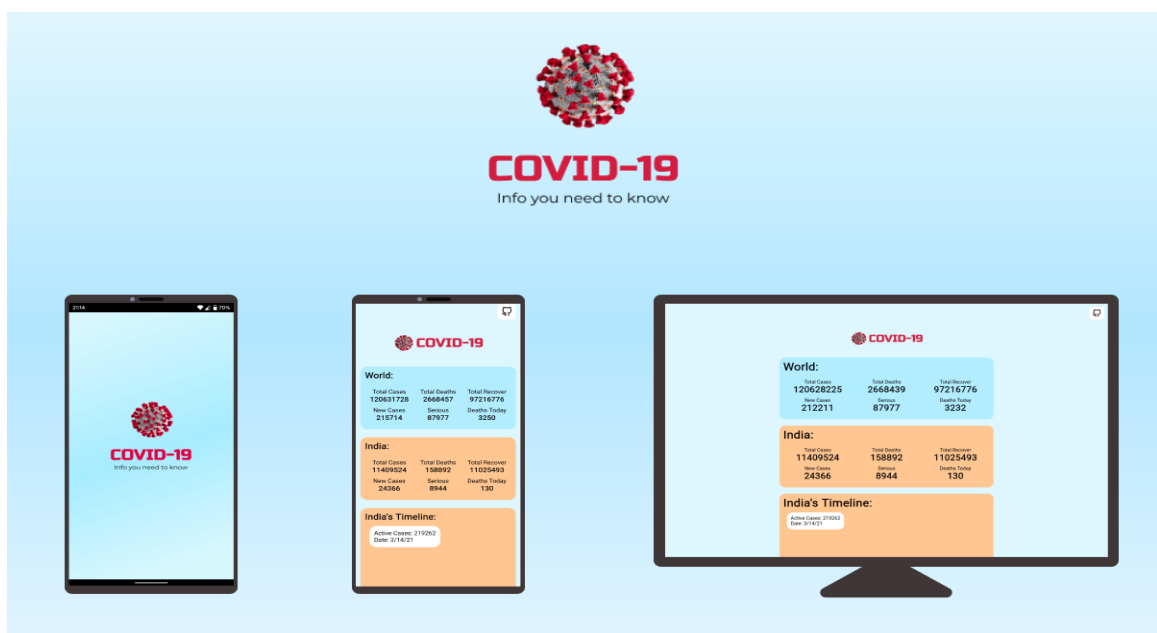


Рис. 1.1. Інтерфейс додатку *Covid-19 | Info you need to know*

Основна функція додатку – виведення статистики захворюваності у світі, що зображено на рисунку 1.1., *Covid19* - це проект *React-Native*, розроблений для підтримки людей, які постраждали від вірусу *Covid-19 (Corona)* у всьому світі. Додаток використовує дуже простий та легкий дизайн, маючи на увазі належну реакцію на різних платформах, таких як *Android* [11], *iOS* [12] та *Web* [13]. Додаток

дозволяє користувачеві відстежувати зростання активних випадків у своїй країні та надає статистику з усього світу.

COVID-19 Symptom Study

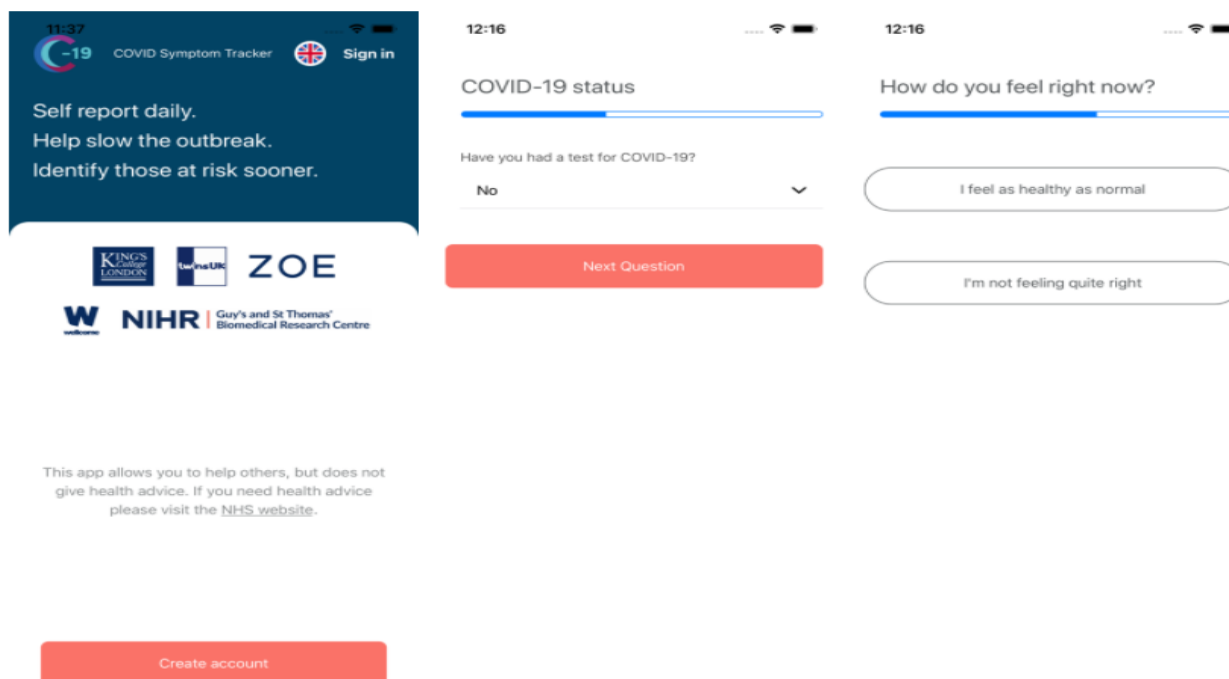


Рис. 1.2. Інтерфейс програми *COVID-19 Symptom Study*

COVID Symptom Study [2] (рисунок 1.2.) – це програма з відкритим кодом для *Android* та *iOS*, метою якої є зупинити поширення хвороби *COVID-19* та допомогти швидше виявити людей, які перебувають у групі ризику.

Дослідження симптомів *COVID* було розроблено лікарями та науковцями Лондонського Королівського коледжу та лікарнями Святого Томаса, що працюють у партнерстві з *ZOE Global Ltd* [14] – медичною науковою компанією.

Функціонал програми направлений на виявлення симптомів коронавірусної хвороби та перевірки взаємодії клієнта додатку з інфікованою людиною.

COVID Shield Mobile App

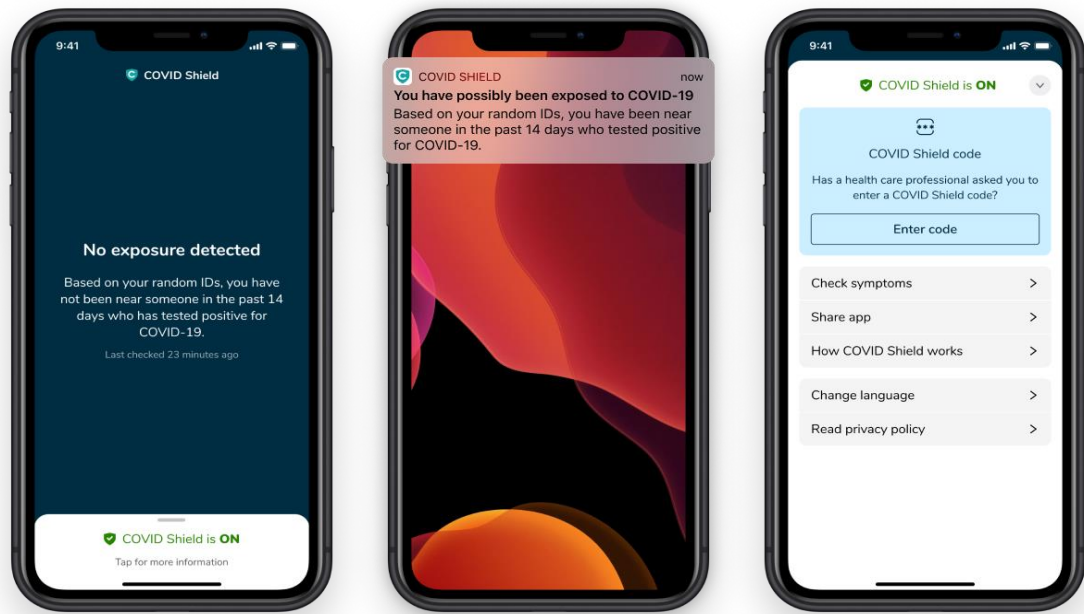


Рис. 1.3. Інтерфейс програми *COVID Shield Mobile App*

Клієнтська програма *React Native* для *Apple / Google Exposure Notification framework*, проінформована вказівками канадських уповноважених з питань конфіденційності. Все працює з обґрунтуванням *COVID Shield* (рисунок 1.3.).

Основний функціонал програми – це перевірка симптомів клієнта додатку, в тому ж числі й можливість контакту з хворим на коронавірус – це відбувається за допомогою перевірки шляхів пересування клієнтів додатку, що мають позитивний чи негативний результат тесту – в результаті чого програма вираховує їх взаємодію між собою.

Крім цього, за допомогою додатку можна перевірити симптоми коронавірусної хвороби.

Covid-19-APP

Експериментальний мобільний додаток для відстеження стану пандемії COVID-19, інтерфейс на рисунках 1.4-1.5.

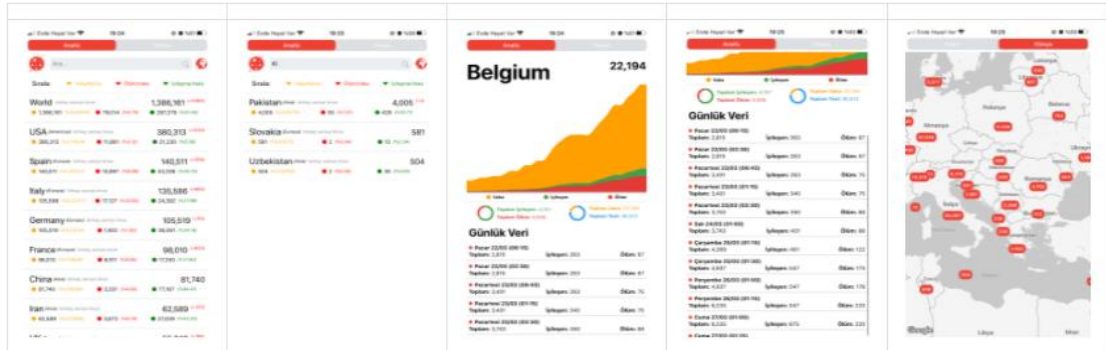


Рис. 1.4. Інтерфейс програми Covid-19-APP

Ця програма використовує *REST Countries v1*, *COVID-19 API* від *Rapid API* та створює статистику даних з простою візуалізацією для користувача.

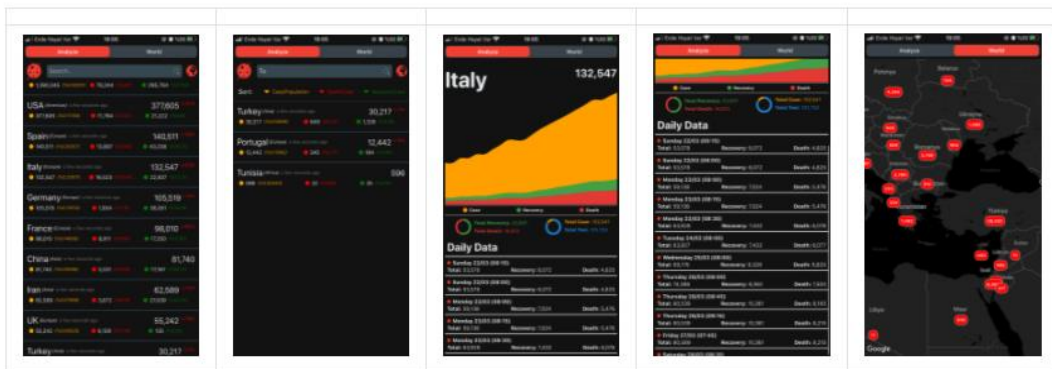


Рис. 1.5. Інтерфейс програми Covid-19-APP

Coronavirus (COVID-19) Dashboard

Інформаційний дашборд коронавірусу (COVID-19) показує динаміку (кривизну графіка) розподілу коронавірусу по країнах (Рис. 1.6.).

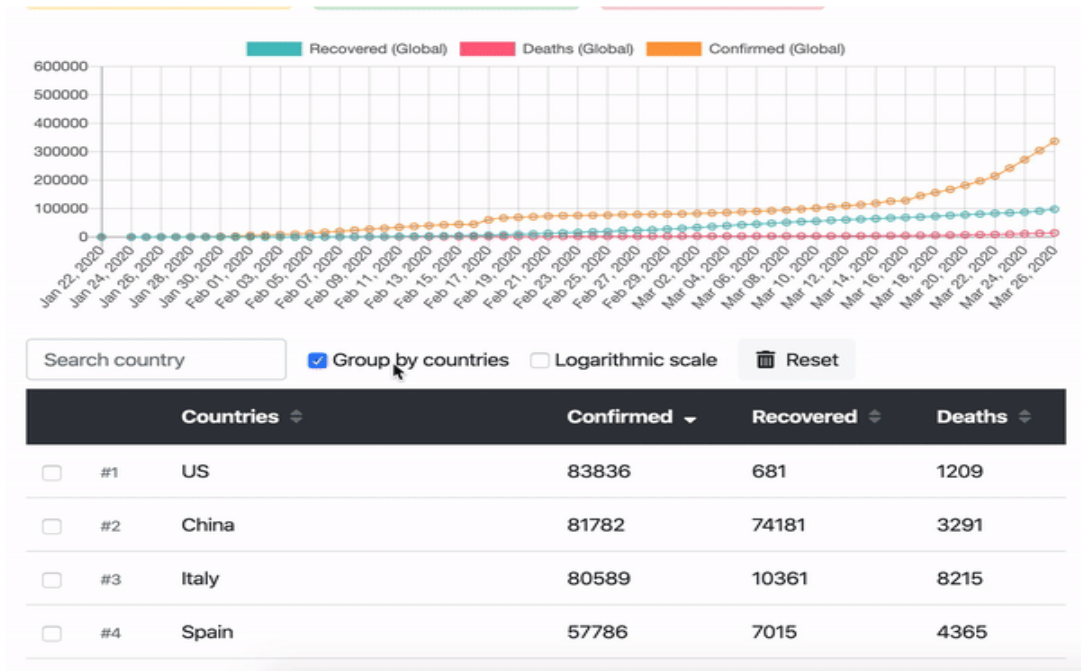


Рис. 1.6. Coronavirus Dashboard

Інформаційна панель все ще є необробленою, але вона забезпечує основну функціональність відображення глобальних даних та таблиць країн.

COVID-19 tracker

На рисунку 1.7 зображені картки, що показують хворих, вилікуваних та кількість смертей у світі. Рисунок 1.8 показує глобальну тенденцію розвитку захворюваності.



Рис. 1.7. COVID-19 tracker

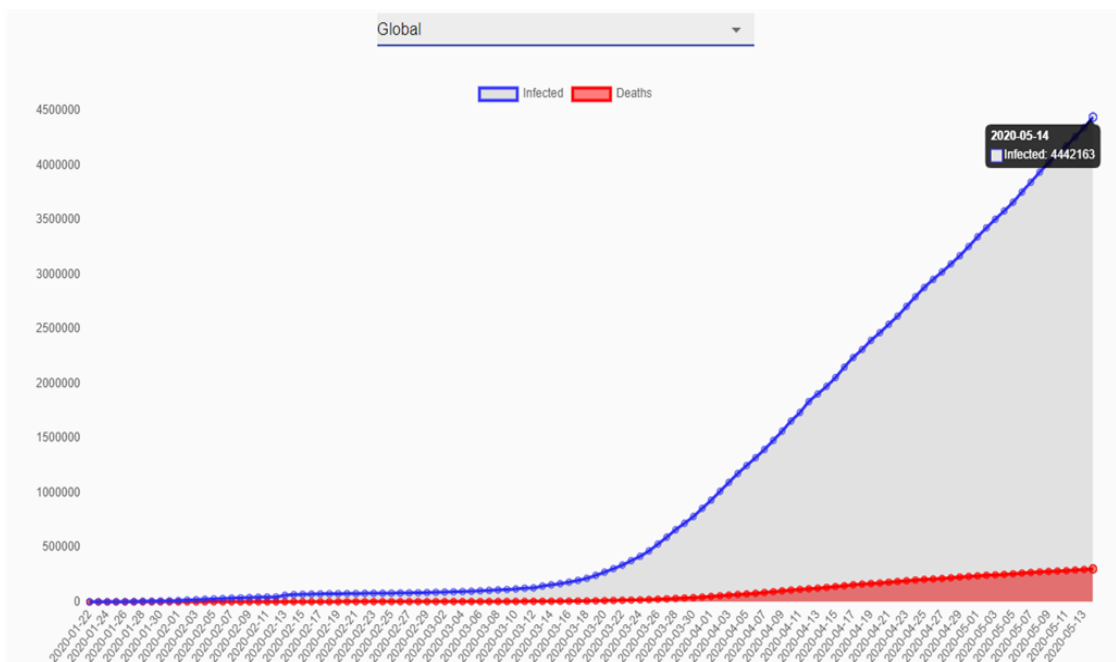


Рис. 1.8. COVID-19 tracker

Coronavirus Tracker App

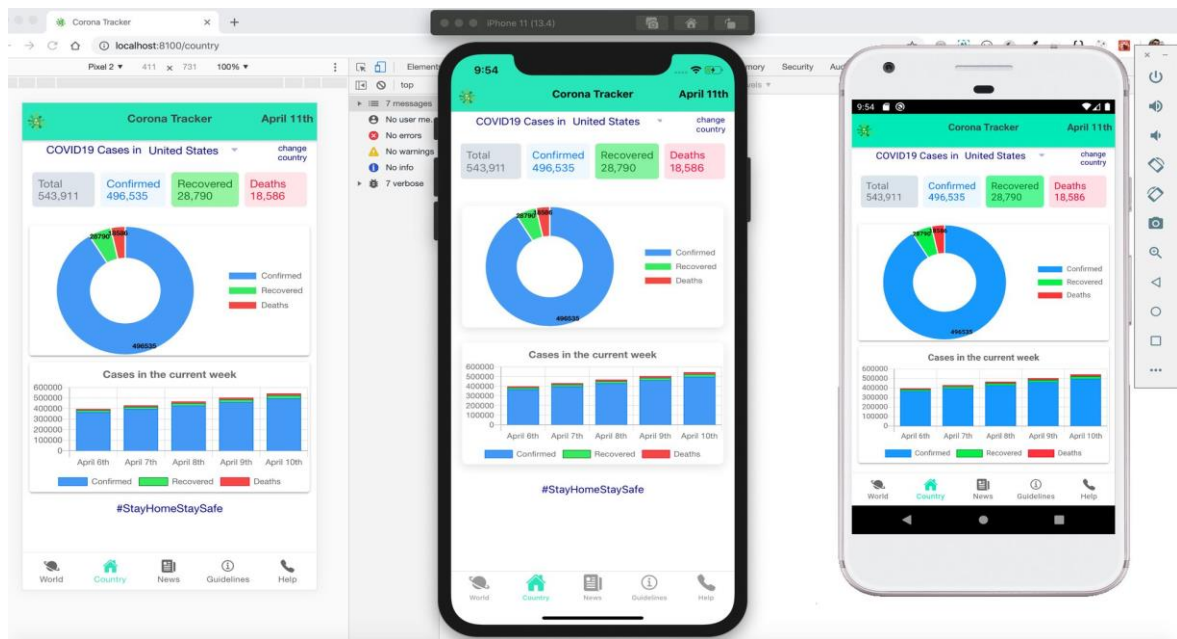


Рис. 1.9. Інтерфейс програми *Coronavirus Tracker App*

Інтерфейс додатку, що зображений на рисунку 1.7, складається з наступних елементів:

- Світ – відображення інформаційної панелі *COVID-19*;
- Країна – дозволяє обрати країну для перевірки стану;
- Новини – отримання останніх новин щодо пандемії коронавірусу;
- Рекомендації – дозволить читати та переглядати поради та рекомендації;
- Довідка – отримання допомоги та залишення відгуку.

COVID19-Tracker

Прогресивний додаток, інтерфейс якого зображено на рисунках 1.10 - 1.11 що виконує функцію трекінгу, перевірки наявних кейсів захворювань, а також має можливість проведення порівняння та збереження обраної країни до закладок.



Рис. 1.10. Головний екран та країни

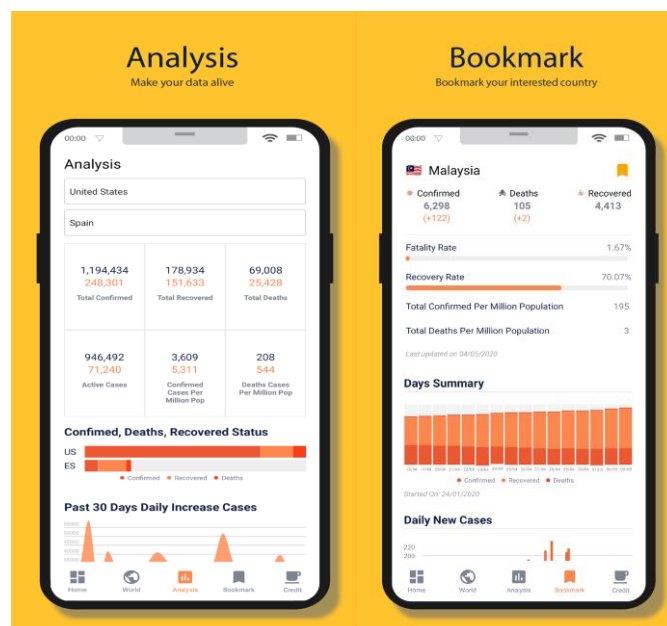


Рис. 1.11. Порівняння та закладки

Vete Corona

Простий, проте ефективний додаток, можливості котрого відображені на рисунку 1.12, а саме: перевірка кількості захворювань, перевірка країни, вибір країни та перевірка безпечної подорожі.

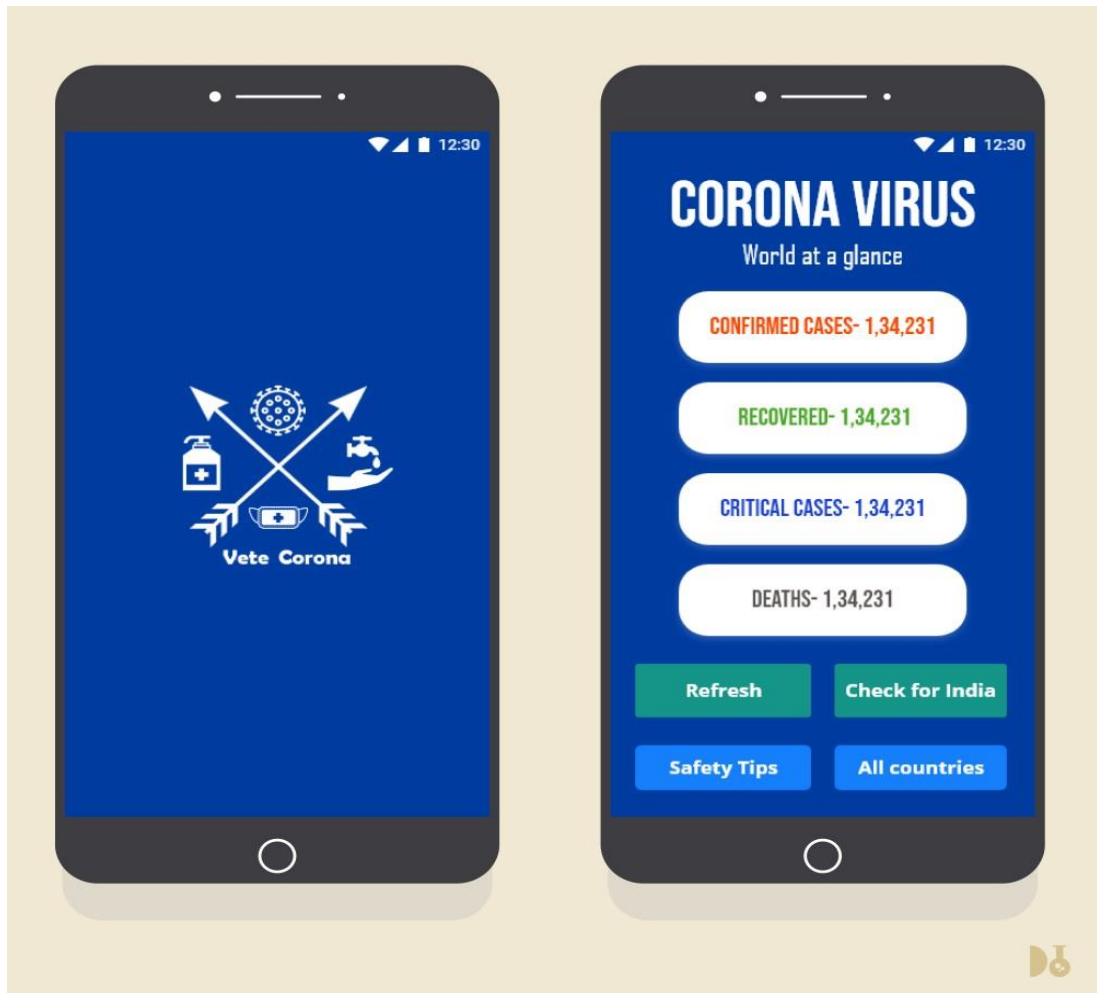


Рис. 1.12. Інтерфейс додатку *Vete Corona*

1.3. Порівняння існуючих рішень із запропонованим

Як можемо бачити, аналогічні додатки є доволі простими у використанні та функціональності, а отже потребують модернізації.

Порівнюючи аналоги додатку аналізу даних захворюваності коронавірусною хворобою *COVID-19* в країнах світу й подібні, було прийнято рішення використати певні частини з інтерфейсу чи дизайну цих додатків, а саме:

1. Відображення статистики світу

2. Відображення статистики країни

3. Інтуїтивно зрозумілий дизайн

Проте, як було сказано вище, цей функціонал не відповідає оптимальним критеріям вимог, тому наступним кроком була модернізація функціональності, а саме:

1. Відображення статистики світу у загальному та на сьогоднішній день;

2. Відображення статистики країни у загальному та на даний момент;

3. Додана можливість збереження країн у закладки;

4. Додана можливість перегляду симптомів, рекомендацій, а також профілактики *COVID-19*;

5. Створено унікальний дизайн з використанням прогресивного принципу: найкращий дизайн – непомітний.

1.3.1. Створення прототипу додатку в графічному редакторі *Figma*

Figma – графічний онлайн-редактор за допомогою котрого можна створити прототип сайту чи мобільного додатку. В нашому випадку за допомогою цього сервісу були створені прототипи екранів майбутнього додатку[3].

Інтерфейс цього редактору є інтуїтивно зрозумілим – використовуються наступні елементи для створення прототипів:

1. Геометричні фігури;

2. Текст;

3. *Frame* (кадр) – в нашому випадку це екран мобільного телефону.

Завдяки можливостям *Figma*, можна переглянути створений прототип на екрані смартфона, не виходячи з редактору й без зайвих зусиль. Нижче зображені екрани, що були створені в ході використання цього редактору.

Всього було створено 12 прототипів екранів, що відповідає поставленій задачі з точки зору функціональності додатку, а саме:

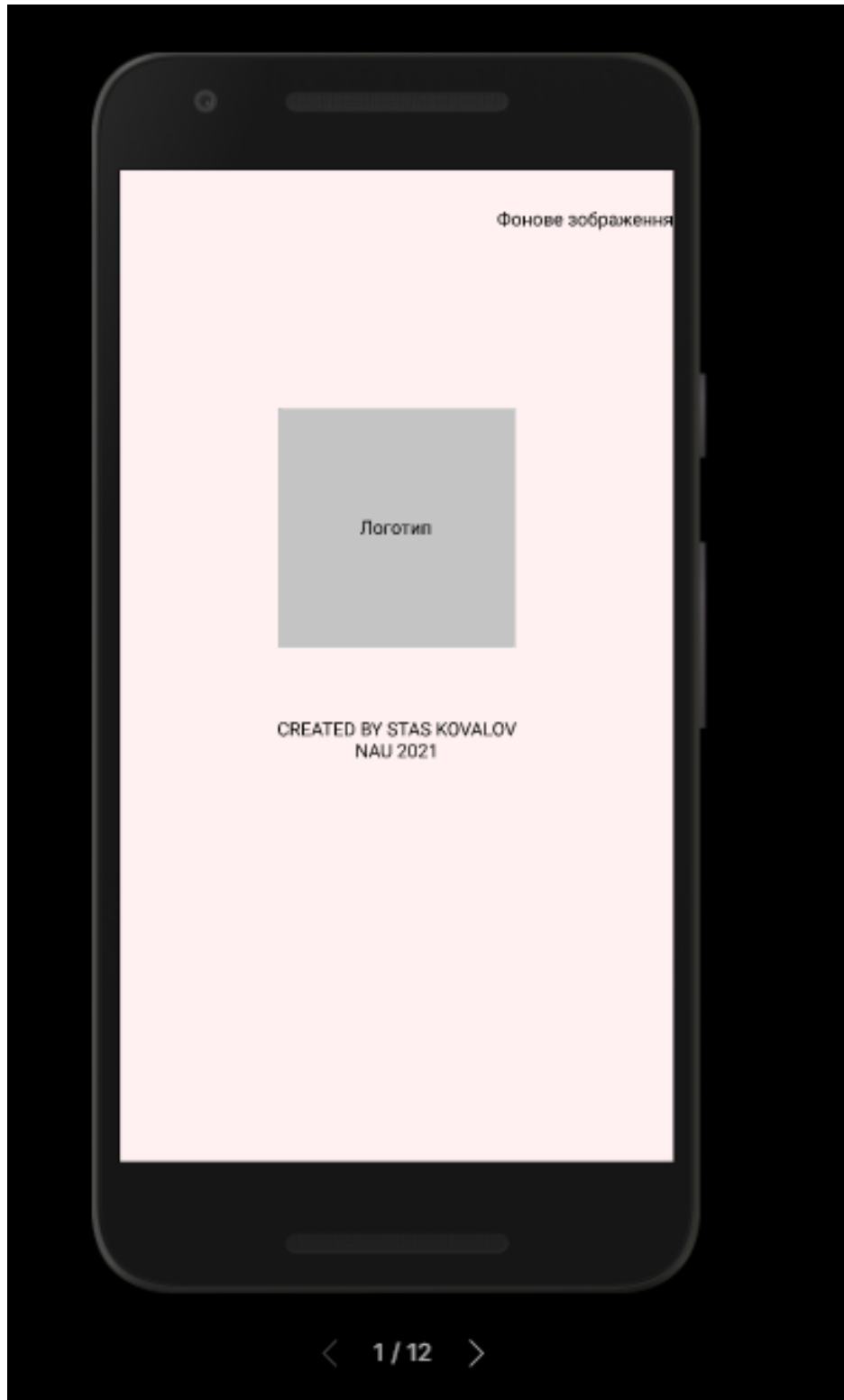


Рис. 1.13. Прототип першого екрану додатка

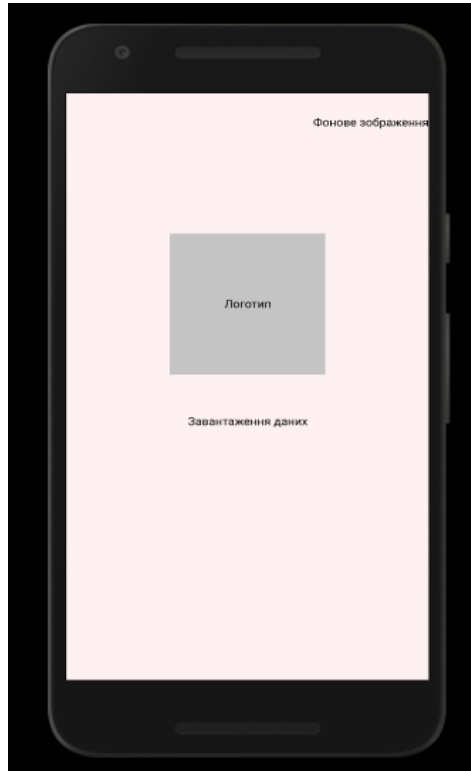


Рис. 1.14. Прототип екрану завантаження додатка



Рис. 1.15. Прототип головного екрану додатка (сумарна статистика)



Рис. 1.16. Прототип головного екрану додатка(статистика сьогодні)

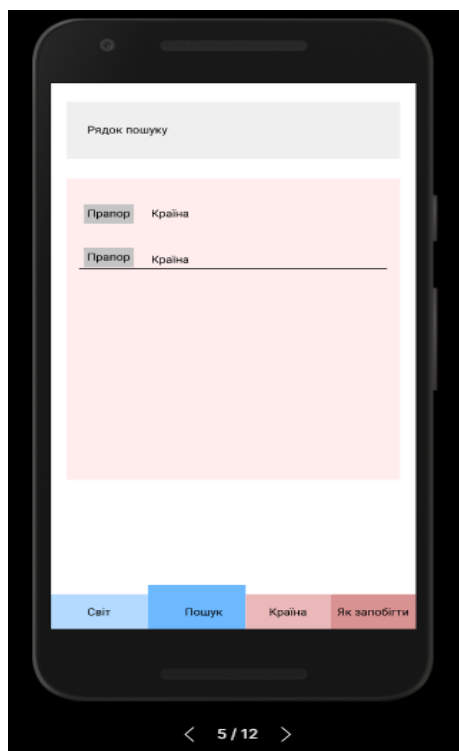


Рис. 1.17. Прототип екрану пошуку країни



Рис. 1.18. Прототип екрану статистики країни



Рис. 1.19. Прототип екрану статистики країни з додаванням в закладки

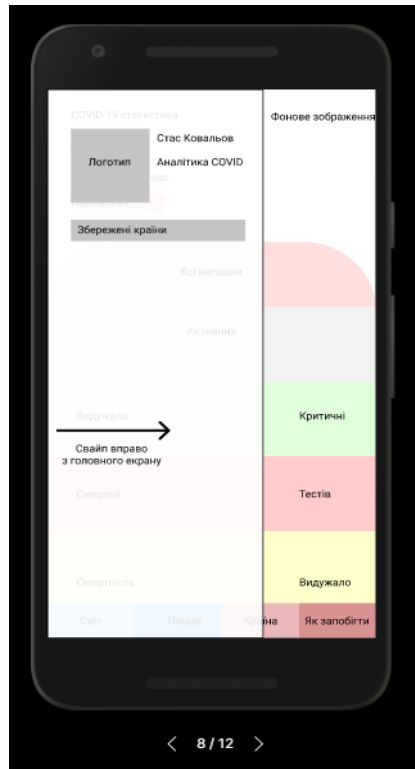


Рис. 1.20. Прототип вкладки збережених країн світу



Рис. 1.21. Прототип екрану збережених країн світу



Рис. 1.22. Прототип екрану симптоми *COVID-19*

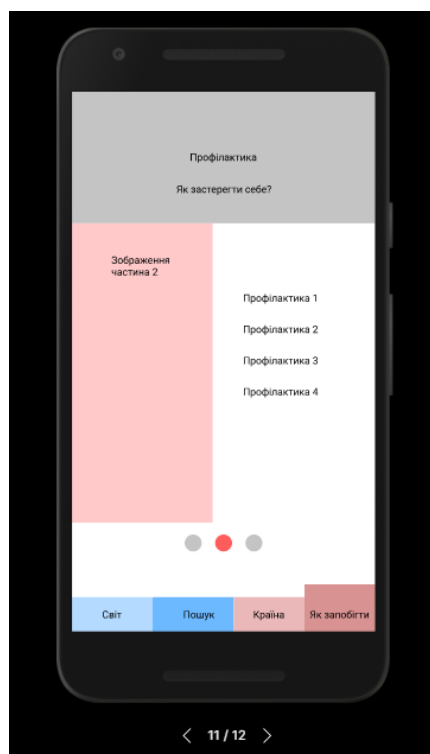


Рис. 1.23. Прототип екрану профілактика *COVID-19*



Рис. 1.24. Прототип екрану як запобігти *COVID-19*

1.3.2. Створення дизайну додатку

Дизайн додатку створювався для будь-якого роду користувачів – починаючи з дітей та закінчуючи людьми в віці. Тобто, є інтуїтивно зрозумілим та не передбачає несподіваних реакцій користувачів – на екрані розташовані кнопки, натискаючи на котрі відбувається взаємодія з логікою додатку.

На головному екрані знаходяться чотири кнопки:

1. Світ – відображення статистики захворюваності у всьому світі.
2. Країна – відображення статистики захворюваності у обраній країні.
3. Пошук – відображення рядку пошуку для пошуку статистики захворюваності у бажаній країні.
4. Як запобігти – відображення симптомів, профілактики та рекомендацій щодо *COVID-19*.

Свайп вправо на головному екрані відображає список збережених країн світу задля швидкого доступу до статистики збереженої країни.

Дизайн рішення буде представлено у розділі три в пункті інтерфейс додатку.

1.4. Висновки до розділу

У результаті проведеного аналізу було визначено задачу створення додатку аналізу даних захворюваності коронавірусною хворобою, проаналізовано й розглянуто аналоги додатку аналізу захворюваності *COVID-19* на основі яких були побудовані прототипи екранів майбутнього додатку та розроблено дизайн.

Для розробки теми дипломної роботи необхідно розглянути інструменти створення мобільного додатку на *IOS/ANDROID*, а саме:

1. *IDE: Visual Studio Code*
2. Мову *JavaScript*;
3. Фреймворк *React Native*;
4. *API coronavirus map* для статистики в світі;

РОЗДІЛ 2

ІНСТРУМЕНТИ СТВОРЕННЯ ДОДАТКУ

В цьому розділі будуть розглянуті інструменти створення мобільного додатку на *IOS/ANDROID* аналізу даних коронавірусною хворобою *COVID-19* країн світу. В першу чергу буде розглянута *IDE – Visual Studio Code*, далі – мова *JavaScript* та фреймворк *React Native*. В кінці розділу – огляд *API coronavirus map* для отримання статистики захворюваності в світі.

2.1. IDE: Visual Studio Code

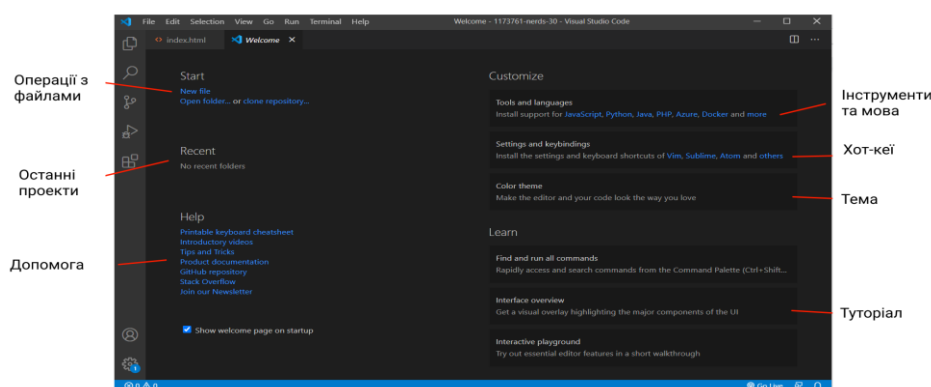


Рис. 2.1. Головний екран *Visual Studio Code*

IDE (Integrated Development Environment) – це інтегроване середовище розробки, що використовується для створення додатків. Існує велика кількість різних середовищ для розробки для різних мов програмування, платформ чи операційних систем. Проте, в цьому розділі буде розглянутий потужний та простий у використанні інструмент – *Visual Studio Code*, завдяки якому проводилась розробка додатку на *IOS/ANDROID* аналізу даних коронавірусною

<i>Кафедра КІТ</i>				<i>НАУ 21 12 05 000 ПЗ</i>			
<i>Виконав</i>	<i>Ковальов С.С.</i>			<i>Інструменти створення додатку</i>	<i>Літера</i>	<i>Аркуш</i>	<i>Аркушів</i>
<i>Керівник</i>	<i>Райчев І.Е.</i>				<i>Д</i>		
<i>Консульт.</i>						30	10
<i>Норм. контр.</i>	<i>Шевченко О.П.</i>				<i>УС-411Б 122</i>		
<i>Зав. Каф.</i>	<i>Савченко А.С.</i>						

хворобою COVID-19 країн світу.

Visual Studio Code – це безкоштовний та популярний редактор коду від компанії *Microsoft* [4]. Цей редактор використовується як для веб-розробки так і для створення мобільних додатків, що й відбулось в ході виконання цієї роботи.

Під час першого запуску редактору відкривається головний екран – рисунок 2.1. – всередині котрого наявні інтерактивні посилання, що відкривають потрібну частину інтерфейсу редактору чи сторінку в браузері. Сам редактор підтримує велику кількість мов програмування та легко налаштовується під потреби користувача.

Розглянемо інтерфейс редактору:

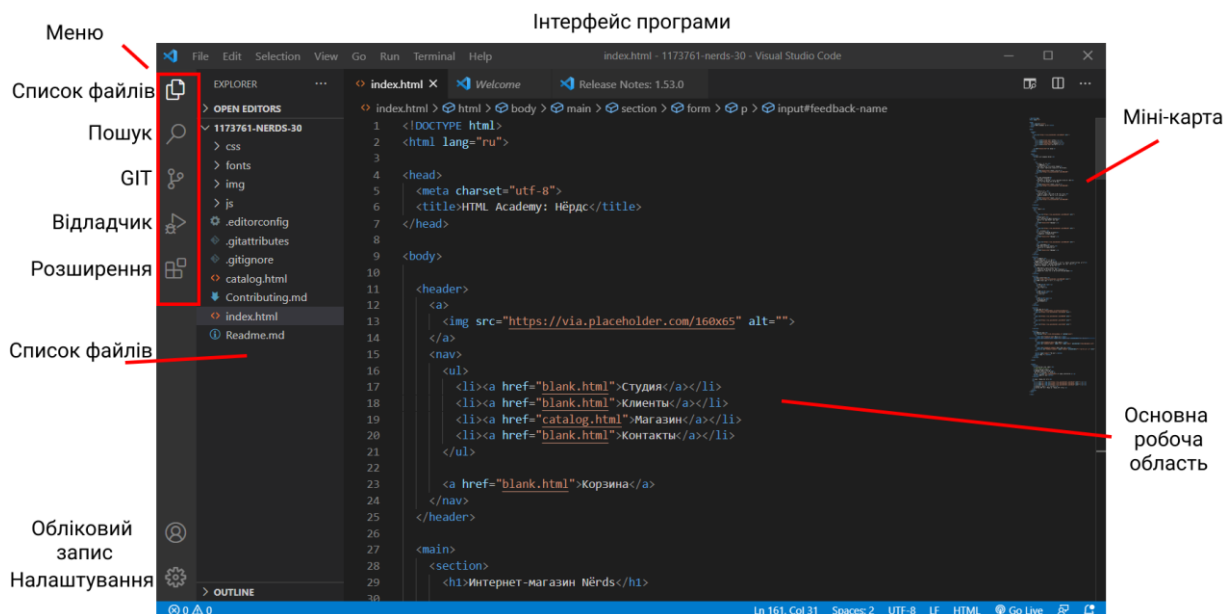


Рис. 2.2. Інтерфейс редактору

Основну частину екрану займає вікно з контентом файлу. У лівій частині екрана розміщено активне меню, в якому знаходяться головні функції редактора. Вкладка провідника відкривається за умовчужанням при запуску програм. У ній виводиться список відкритих файлів та каталог відкритих папок.

Наступний пункт інтерфейсу – пошук (рисунок 2.3.).

У поле можна ввести шукане значення, і редактор покаже, в яких місцях він знаходиться. При бажанні можна змінити значення через поле заміни.

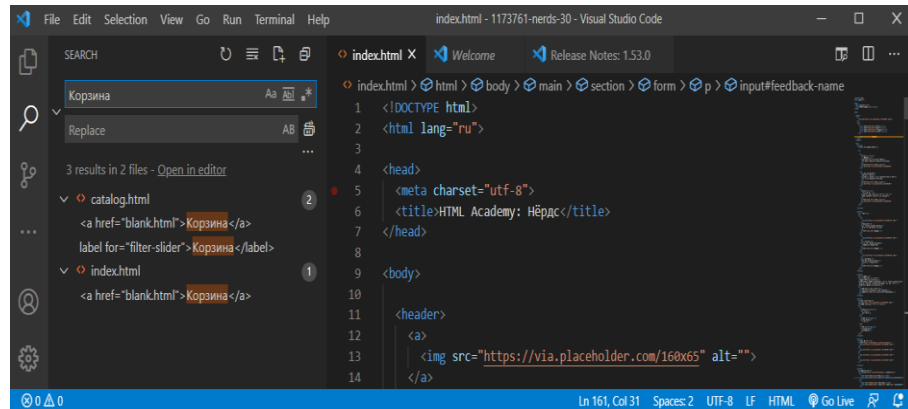


Рис. 2.3. Пошук

2.2. Мова програмування *JavaScript*

JavaScript - мова програмування, що створена *Netscape Communication Corporation* [5]. Дана мова не має ніякого відношення до мови *Java*. Перша назва звучала як *LiveScript*, однак, через величезну популярність мови *Java*, виходячи з комерційних міркувань мову було перейменовано в *JavaScript*. Важлива особливість *JavaScript* – це об'єктна орієнтованість. Розробникам, котрі використовують дану мову доступні численні об'єкти, такі, як документи, гіперпосилання, форми, фрейми і т.д. Об'єкти характеризуються описовою інформацією (властивостями) і можливими діями (методами).

Для розробки мобільних додатків не можна використовувати нативний *JavaScript*. Тому і були розроблені такі *JavaScript* фреймворки, як: *React Native*, *Apache Cordova* і інші. За допомогою даних фреймворків розробник може створювати кросплатформені додатки для різних платформ. Однак, в більшості випадків, використання фреймворків не дозволяє розробити складний додаток, і підходить лише для розробки середніх і малих програм.

Плюси використання *JavaScript*:

1. Кросплатформеність.
2. Безліч фреймворків.
3. Велика кількість документації.

Мінуси використання *JavaScript*:

1. Додаток може бути тільки гібридним.

2. Незважаючи на кроссплатформенність, у кожній платформі свої особливості, і написати єдиний для всіх платформ код не вдасться.

2.3. *React Native*

Розглянемо, що ж таке *React Native*.

React Native – це фреймворк для розробки кроссплатформених додатків для *IOS* і *Android* від компанії *Facebook*[6]. Протягом всього періоду розробки і підтримки проекту, з використанням цього фреймворка, розробник працює з кодовою базою, написаною на *JavaScript*. При необхідності написаний код компілюється в нативні формати: *.apk* для *Android* і *.ipa* для *IOS*, які можна буде додати в магазини: *PlayMarket* для *Android* і *Appstore* для *IOS*, де їх зможуть завантажити потенційні користувачі. Проектування архітектури додатку, опис класів, функцій, методів реалізується ґрунтуючись на функціоналі і можливостях цієї мови. *React Native* ніяк не обмежує розробника у використанні *JavaScript* - фреймворк лише надає елементи, які можуть бути скомпільовані в нативні методи для роботи з функціями мобільного пристрою: отримання геолокації, робота з камерою пристрою, доступ до локального сховища пристрою і так далі. Тому всі переваги роботи з *JavaScript*, також доступні і в розробці мобільних додатків, використовуючи *React Native*:

- *JavaScript* дуже популярна мова: це єдина мова для розробки клієнтський додатків і у нього величезне співтовариство розробників. Тому існує безліч бібліотек і готових модулів, написаних іншими розробниками, які знаходяться в відкритому доступі і які можна використовувати в своєму проекті;

- *JavaScript* має низький поріг входження: цю мову можна вивчити в короткі терміни і відразу ж почати писати код для вирішення реальних завдань;

- як правило, при роботі з цією мовою не використовуються масивні фреймворки з великою кодовою базою, тому можна плавно підвищувати складність реалізованих проектів шляхом вивчення окремих бібліотек при їх інтеграції;

– для створення мобільних додатків розробнику не потрібні будуть знання і досвід в їх розробці. Досить мати досвід в розробці клієнтських додатків з *JavaScript*. Відповідно, набагато легше буде знайти людей для розробки / підтримки проекту;

– одна кодова база для різних типів проектів. Можливе використання одних і тих же функцій в проектах серверних, клієнтський і мобільних додатків. Якщо при роботі над клієнтським додатком, добре себе зарекомендувала одна з підключених бібліотек, то її можна буде використовувати і в розробці мобільного додатку.

Більшість програм в кінцевому підсумку використовуватимуть один із цих основних компонентів, що зображені на зображенні 2.4:

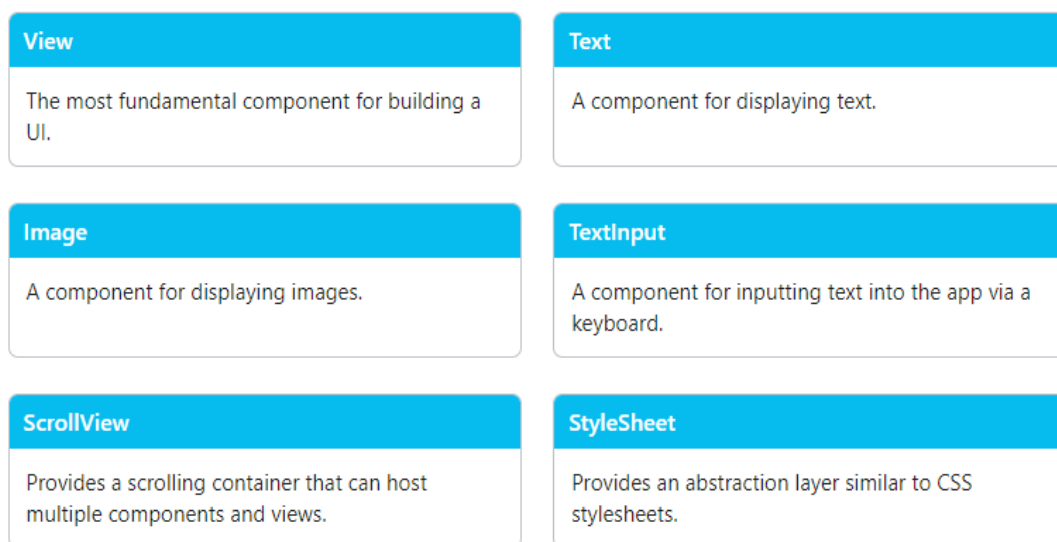
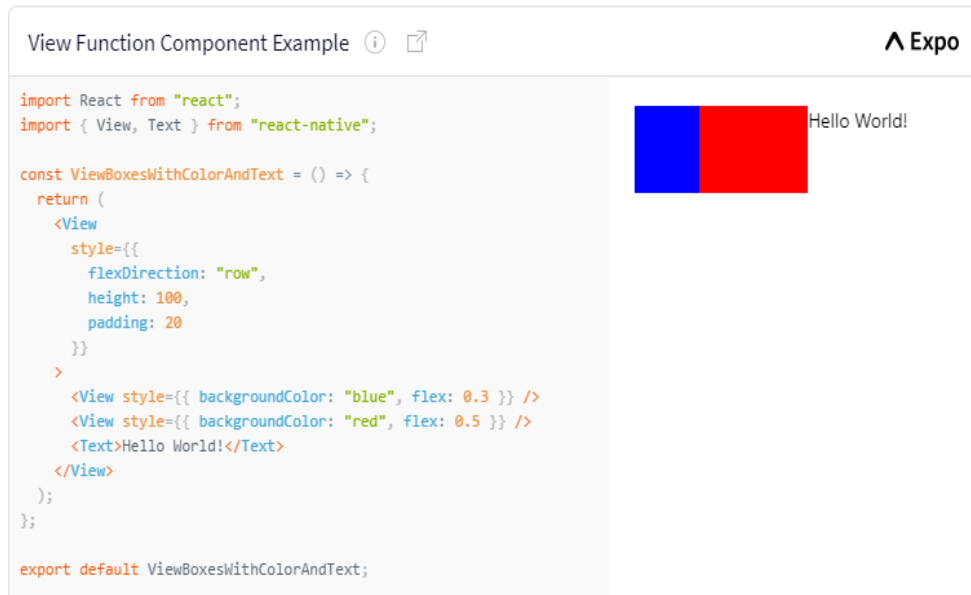


Рис. 2.4. Основні компоненти *React Native*

Найбільш фундаментальний компонент для побудови інтерфейсу, *View* - це контейнер, який підтримує макет із *flexbox*, стилем, деякою обробкою дотиків та елементами управління доступністю. Переглядайте карти безпосередньо у власному еквіваленті подання на будь-якій платформі, на якій працює *React Native*, будь то *UIView*, `<div>`, *android.view* тощо.

View призначений для вкладання в інші подання і може мати від 0 до багатьох дітей будь-якого типу.

Цей приклад (рисунок 2.5) створює подання, яке обертає два поля кольором і текстовий компонент з відступами.



```
View Function Component Example ⓘ ↗ ^ Expo

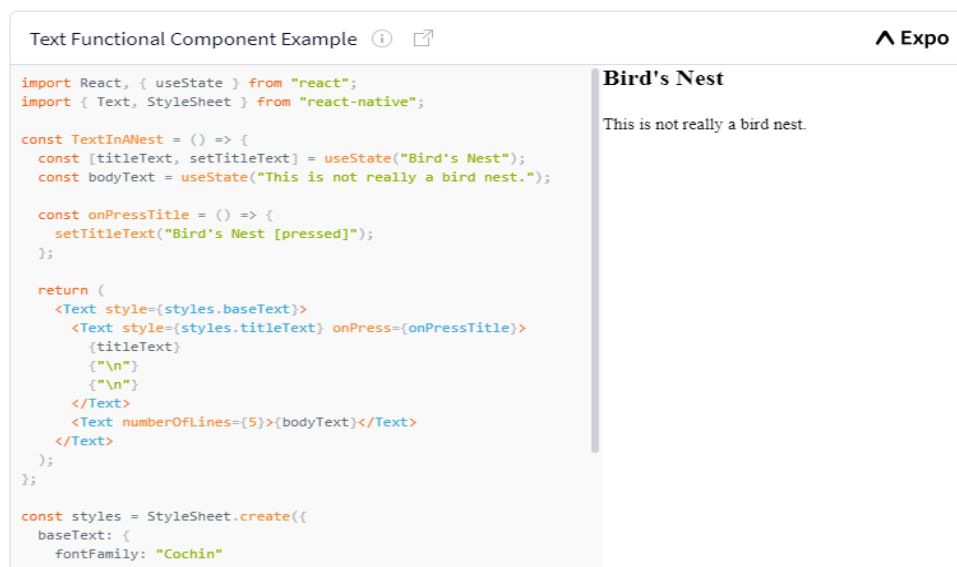
import React from "react";
import { View, Text } from "react-native";

const ViewBoxesWithColorAndText = () => {
  return (
    <View
      style={{
        flexDirection: "row",
        height: 100,
        padding: 20
      }}
    >
      <View style={{ backgroundColor: "blue", flex: 0.3 }} />
      <View style={{ backgroundColor: "red", flex: 0.5 }} />
      <Text>Hello World!</Text>
    </View>
  );
};

export default ViewBoxesWithColorAndText;
```

Рис. 2.5. *View Function Component Example*

Text – компонент *React* для відображення тексту. *Text* підтримує вкладання, укладку та обробку дотиків. У наступному прикладі вкладений заголовок та основний текст успадкують *fontFamily* від *styles.baseText*, але заголовок надає власні додаткові стилі. Заголовок і текст будуть складатися один на одного через буквальні нові рядки (рисунок 2.6):



```
Text Functional Component Example ⓘ ↗ ^ Expo

import React, { useState } from "react";
import { Text, StyleSheet } from "react-native";

const TextInANest = () => {
  const [titleText, setTitleText] = useState("Bird's Nest");
  const bodyText = useState("This is not really a bird nest.");

  const onPressTitle = () => {
    setTitleText("Bird's Nest [pressed]");
  };

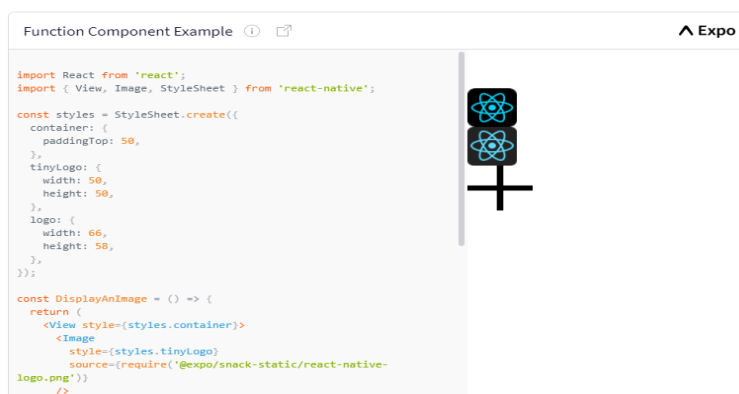
  return (
    <Text style={styles.baseText}>
      <Text style={styles.titleText} onPress={onPressTitle}>
        {titleText}
      </Text>
      <Text style={styles.bodyText}>
        {bodyText}
      </Text>
    </Text>
  );
};

const styles = StyleSheet.create({
  baseText: {
    fontFamily: "Cochin"
  },
  titleText: {
    color: "green"
  },
  bodyText: {
    color: "red"
  }
});
```

Рис. 2.6. *Text Functional Component Example*

Image – компонент *React* для відображення різних типів зображень, включаючи мережеві зображення, статичні ресурси, тимчасові локальні зображення та зображення з локального диска, наприклад, камери.

У цьому прикладі (рисунок 2.7) показано отримання та відображення зображення з локальної пам'яті, а також із мережі та навіть із даних, наданих у схемі *uri „data:”*.



```
Function Component Example ⓘ 📄 ^ Expo

import React from 'react';
import { View, Image, StyleSheet } from 'react-native';

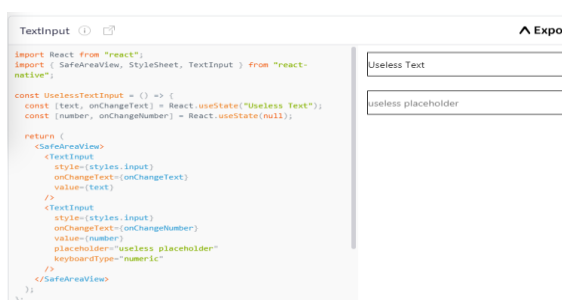
const styles = StyleSheet.create({
  container: {
    padding: 50,
  },
  tinyLogo: {
    width: 50,
    height: 50,
  },
  logo: {
    width: 66,
    height: 58,
  },
});

const DisplayAnImage = () => {
  return (
    <View style={styles.container}>
      <Image
        style={styles.tinyLogo}
        source={require('@expo/snack-static/react-native-
logo.png')}
      />
    </View>
  );
};
```

Рис. 2.7. *Image Component Example*

TextInput – основний компонент для введення тексту в додаток за допомогою клавіатури. Реквізит забезпечує можливість конфігурування кількох функцій, таких як автоматичне виправлення, автоматичне використання великих літер, текст заповнювача та різні типи клавіатури, такі як числова клавіатура.

Основним варіантом використання є складання *TextInput* та підписка на події *onChangeText* для читання вводу користувача. Є також інші події, такі як *onSubmitEditing* та *onFocus*, на які можна підписатися. Мінімальний приклад(рисунок 2.8):



```
TextInput ⓘ 📄 ^ Expo

import React from 'react';
import { SafeAreaView, StyleSheet, TextInput } from 'react-
native';

const UselessTextInput = () => {
  const [text, onChangeText] = React.useState('Useless Text');
  const [number, onChangeNumber] = React.useState(null);

  return (
    <SafeAreaView>
      <TextInput
        style={styles.input}
        onChangeText={onChangeText}
        value={text}
      />
      <TextInput
        style={styles.input}
        onChangeText={onChangeNumber}
        value={number}
        placeholder="useless placeholder"
        keyboardType="numeric"
      />
    </SafeAreaView>
  );
};
```

Рис. 2.8. *TextInput Component Example*

Компонент, який обгортає платформу *ScrollView*, забезпечуючи інтеграцію із системою "відповідача", що блокується дотиком(рисунок 2.9).

Майте на увазі, що *ScrollViews* має мати обмежену висоту, щоб працювати, оскільки вони містять дочірні елементи необмеженої висоти в обмеженому контейнері (за допомогою взаємодії прокрутки). Для того, щоб обмежити висоту *ScrollView* - встановіть висоту *View* безпосередньо (не рекомендується), або переконайтесь, що всі батьківські *View* мають обмежену висоту. Якщо забути перенести `{flex: 1}` вниз у стек view може призвести до помилок, які інспектор елементів швидко налагоджує.

Поки що не підтримує інші автовідповідачі, щоб заблокувати цей вид прокрутки, щоб стати відповідачем.

<ScrollView> проти *<FlatList>* - який із них використовувати?

ScrollView робить відразу всі свої компоненти дочірніми, якщо вони реагують, але це має негативні наслідки.

Уявіть, у вас є дуже довгий список елементів, які ви хочете відобразити, можливо, вміст на кілька екранів. Створення компонентів *JS* та власних *view* для всього одразу, багато з яких може навіть не відобразитися, сприятиме повільній візуалізації та збільшенню використання пам'яті.

Тут у гру вступає *FlatList*. *FlatList* повільно відображає елементи, коли вони мають з'явитися, і видаляє елементи, які прокручуються далеко за межі екрану, щоб заощадити пам'ять та час обробки.

FlatList також зручний, якщо ви хочете відтворити роздільники між вашими елементами, кількома стовпцями, нескінченним завантаженням прокрутки або будь-якою кількістю інших функцій, які він підтримує.

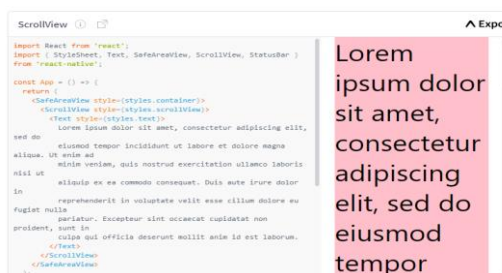
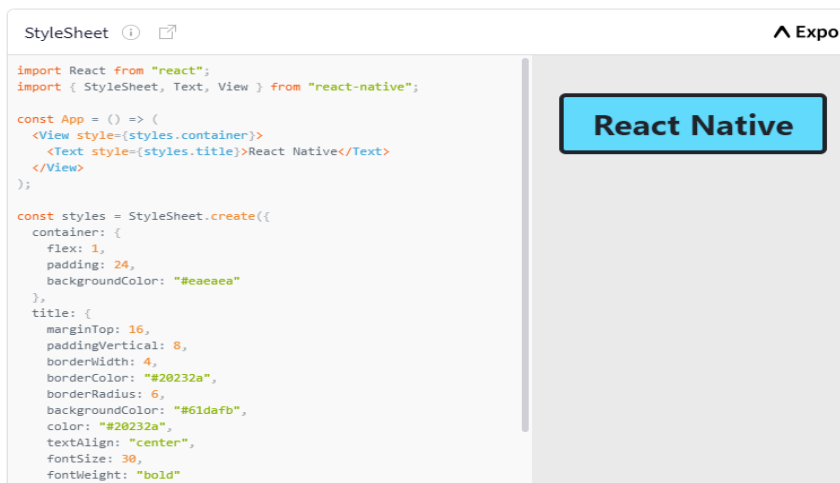


Рис. 2.9. *ScrollView*.

StyleSheet – це абстракція, подібна до таблиці стилів *CSS* [7] – на рисунку 2.4(Ж) зображено приклад.



```
StyleSheet (i) [Exp]
import React from "react";
import { StyleSheet, Text, View } from "react-native";

const App = () => (
  <View style={styles.container}>
    <Text style={styles.title}>React Native</Text>
  </View>
);

const styles = StyleSheet.create({
  container: {
    flex: 1,
    padding: 24,
    backgroundColor: "#eaeaea"
  },
  title: {
    marginTop: 16,
    paddingVertical: 8,
    borderWidth: 4,
    borderColor: "#20232a",
    borderRadius: 6,
    backgroundColor: "#61dafb",
    color: "#20232a",
    textAlign: "center",
    fontSize: 30,
    fontWeight: "bold"
  }
});
```

Рис. 2.10. *StyleSheet*.

2.4. API Coronavirus map

Ключовим моментом є підключення *API* [15] для отримання статистики захворюваності коронавірусною хворобою. Основними критеріями вибору *API* були – актуальність отримуваних даних, безкоштовність та доступність даних різних країн світу.

А отже, оптимальним варіантом став *API coronavirus map* (рисунок 2.11.):

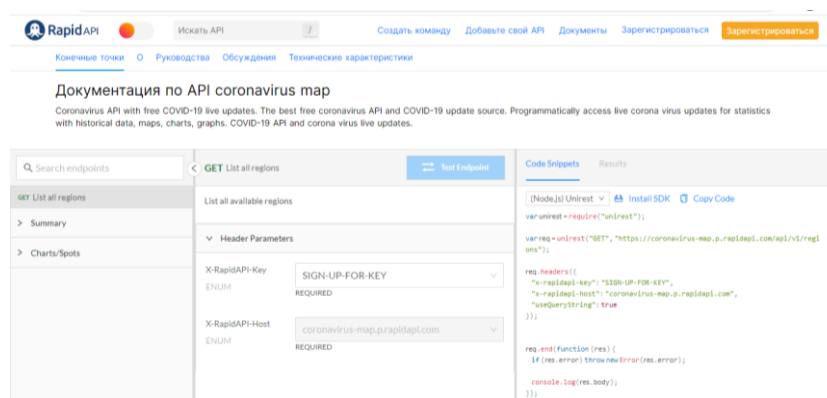


Рис. 2.11. *API coronavirus map*

Що на сайті *rapidapi.com* [8] має відмінну репутацію (рисунок 2.12) та являється безкоштовним, що являється ідеальним варіантом для підключення до додатку аналізу даних захворюваності коронавірусною хворобою *COVID-19* в

країнах світу.

Опис *coronavirus map*: API коронавірусу з безкоштовними оновленнями COVID-19 у реальному часі. Найкраще безкоштовне джерело оновлень API для коронавірусу та COVID-19. Програмно отримувати доступ до оновлень вірусу корона в реальному часі для статистики з історичними даними, картами, діаграмами, графіками [9].



Рис. 2.12. Репутація API *coronavirus map*

2.5. Висновки до розділу

В даному розділі було проаналізовано інструменти та технології для створення мобільного додатку. До цих інструментів увійшли:

1. IDE: *Visual Studio Code*;
2. Мова програмування *JavaScript*;
3. Фреймворк *React Native*;
4. API *coronavirus map* для статистики в світі;

Наступними кроками є:

- огляд створеного додатку аналізу даних захворюваності коронавірусною хворобою COVID-19 в країнах світу
- буде розглянуто інтерфейс додатку,
- дизайн додатку,
- а також тестування функціональності.

РОЗДІЛ 3

ДОДАТОК АНАЛІЗУ ДАНИХ ЗАХВОРЮВАНOSTІ КОРОНАВІРУСНОЮ ХВОРОБОЮ COVID-19

3.1. Інтерфейс додатку

Робота програми починається зі стартового екрану (Рис. 3.1.), в котрому міститься інформація про автора, університет, рік створення, а також обраний логотип додатку, всі ці елементи знаходяться на фоновому зображенні – карті світу.



Рис. 3.1. Перший екран додатку

<i>Кафедра КІТ</i>				<i>НАУ 21 12 05 000 ПЗ</i>			
<i>Виконав</i>	<i>Ковальов С.С.</i>			<i>Додаток аналізу даних захворюваності коронавірусною хворобою COVID-19</i>	<i>Літера</i>	<i>Аркуш</i>	<i>Аркушів</i>
<i>Керівник</i>	<i>Райчев І.Е.</i>				Д	41	14
<i>Консульт.</i>					<i>УС-411Б 122</i>		
<i>Норм. контр.</i>	<i>Шевченко О.П.</i>						
<i>Зав. Каф.</i>	<i>Савченко А.С.</i>						

Після чого користувач отримує повідомлення про те, що відбувається завантаження потрібних для роботи додатку даних – Рис. 3.2. Екран завантаження.

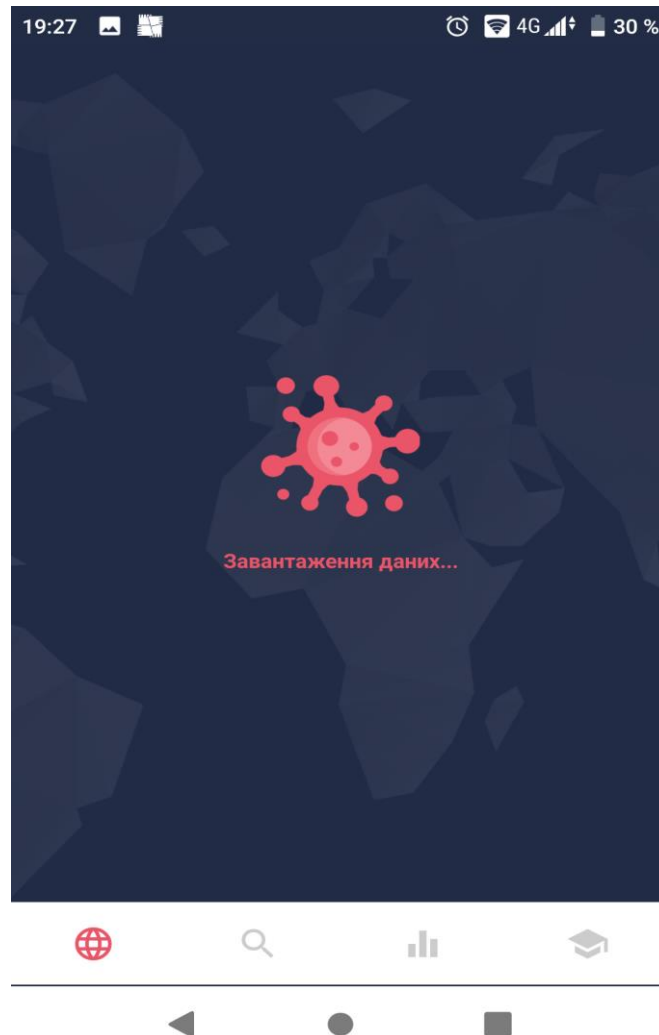


Рис. 3.2. Екран завантаження

Головний екран додатку, на якому розміщені наступні елементи – назва екрану, активна кнопка – Світ, де відображається загальна статистика, перемикач, що регулює відображення за весь час та за сьогоднішній день (Рис.3.4.), кількість активних випадків, кількість людей, що видужали – а також кількість критичних випадків, крім цього – кількість смертей та зроблених тестів та відсоток смертності й людей, що видужало (Рис. 3.3.), виглядає наступним чином:

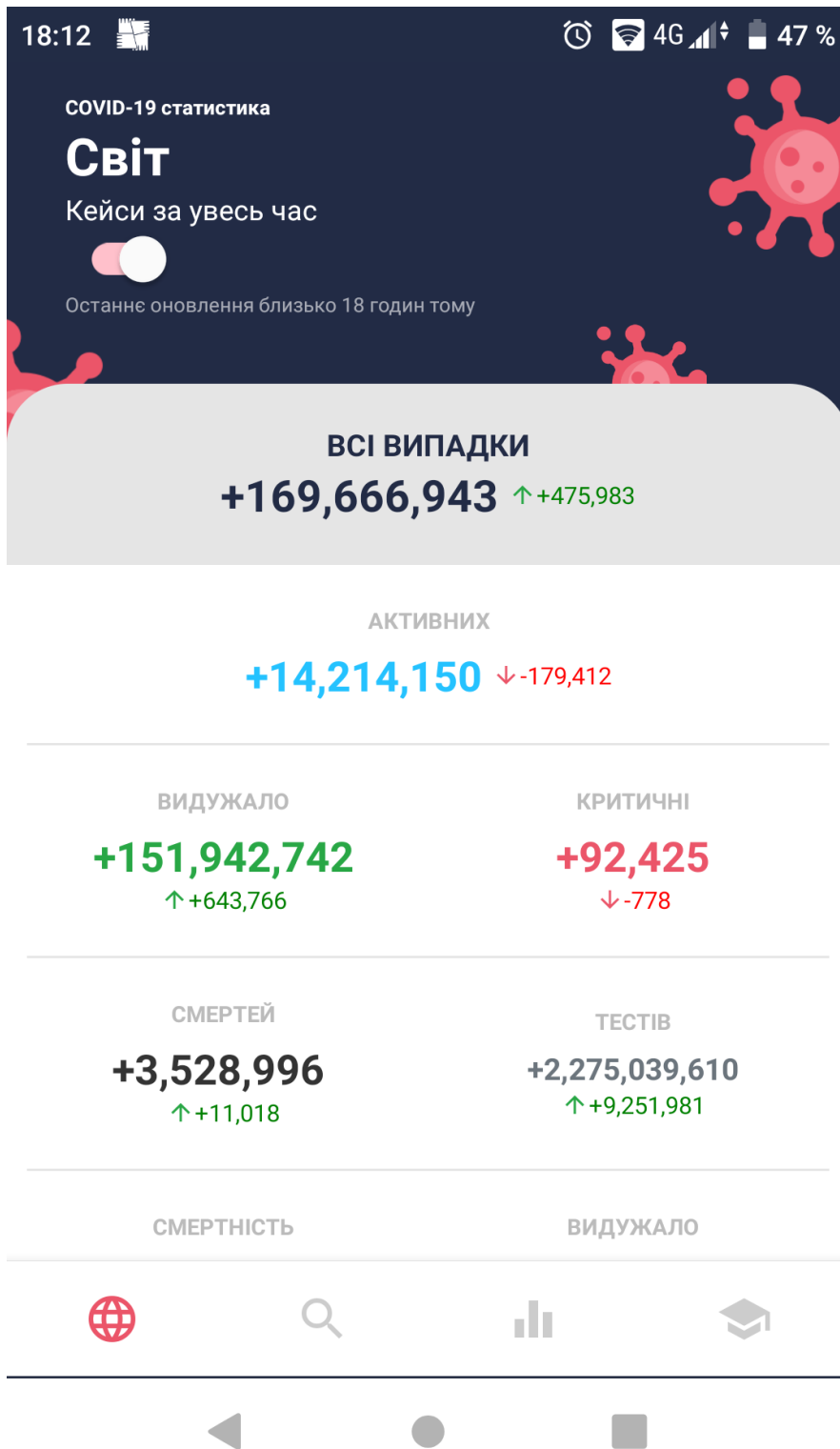


Рис. 3.3. Головний екран додатку із загальною статистикою

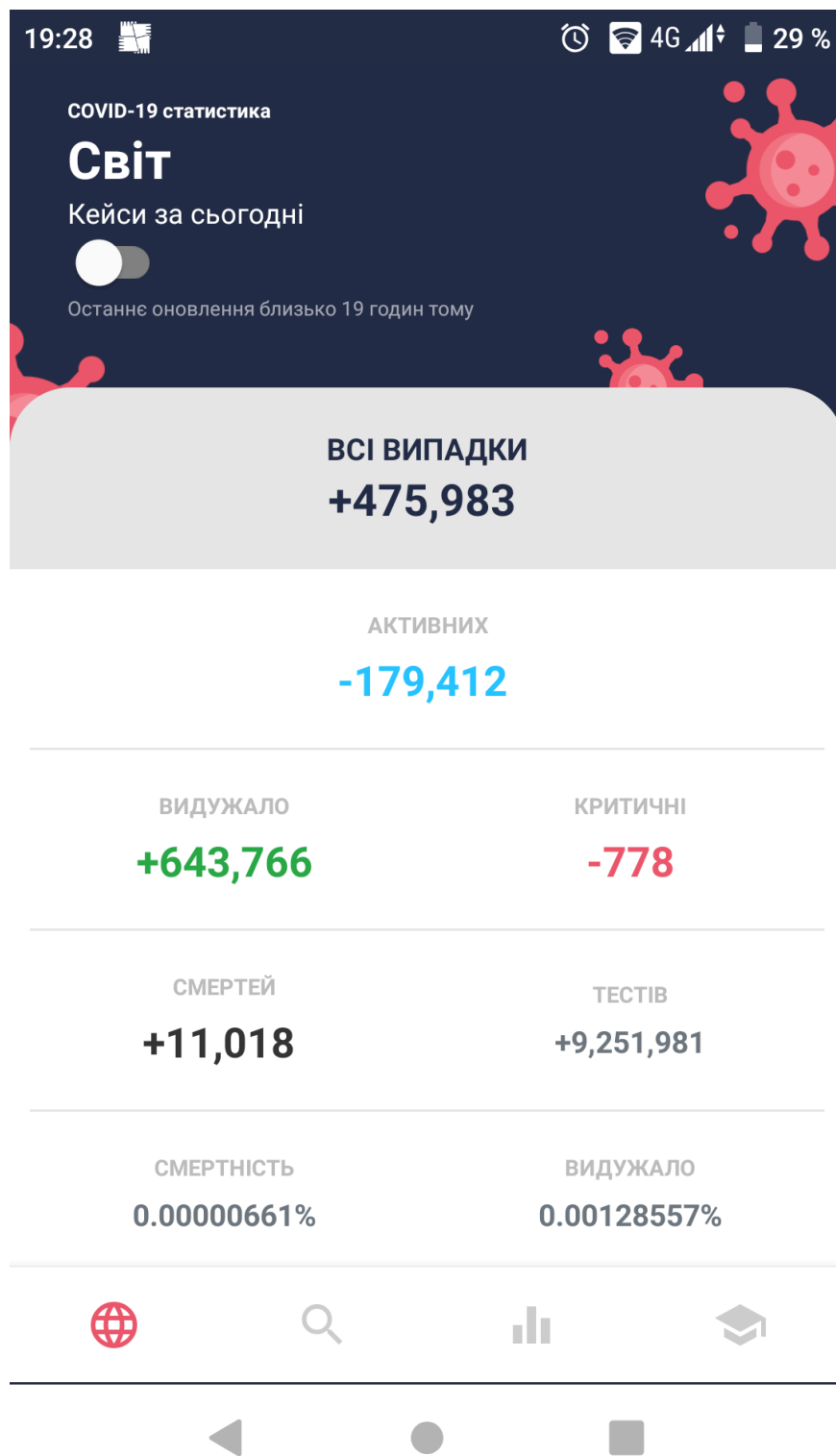


Рис. 3.4. Головний екран додатку зі статистикою на даний момент

Наступною частиною додатку є кнопка пошуку (Рис.3.5.) – де відображається рядок пошуку, в який користувач може ввести потрібну країну та в подальшому переглянути статистику по ній.

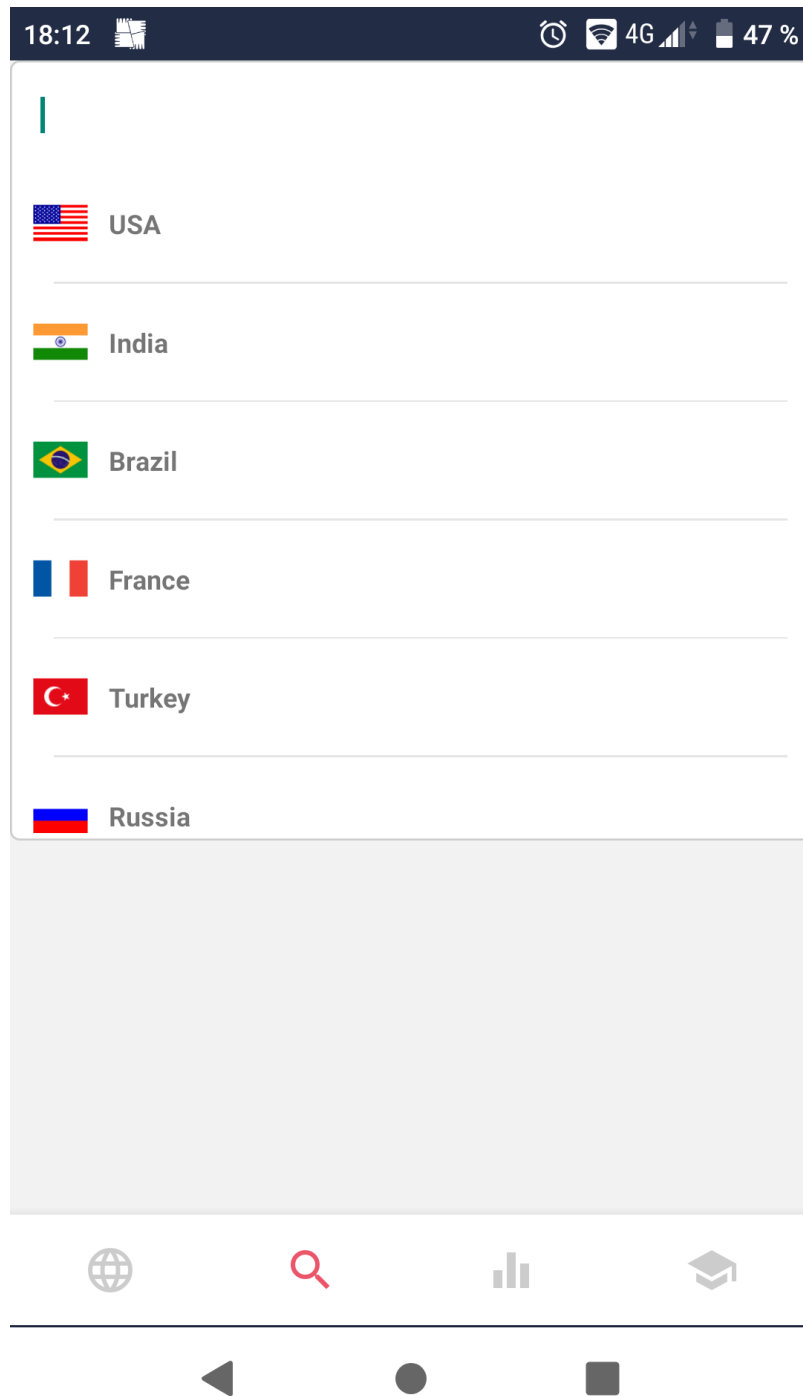


Рис. 3.5. Екран з активною кнопкою пошуку країни для перегляду статистики

Наступна кнопка – Країна (Рис. 3.6), на екрані якої знаходиться інформація щодо назви країни, кількість випадків за весь час, кількість людей, що видужали, кількість померлих та зроблених тестів, а також відсоток смертності та людей, що видужали. Відповідно до поставленої задачі наявна можливість збереження країни до списку збережених, щоб в подальшому можна було швидко отримати доступ до статистики захворюваності цієї країни (Рис. 3.7.).

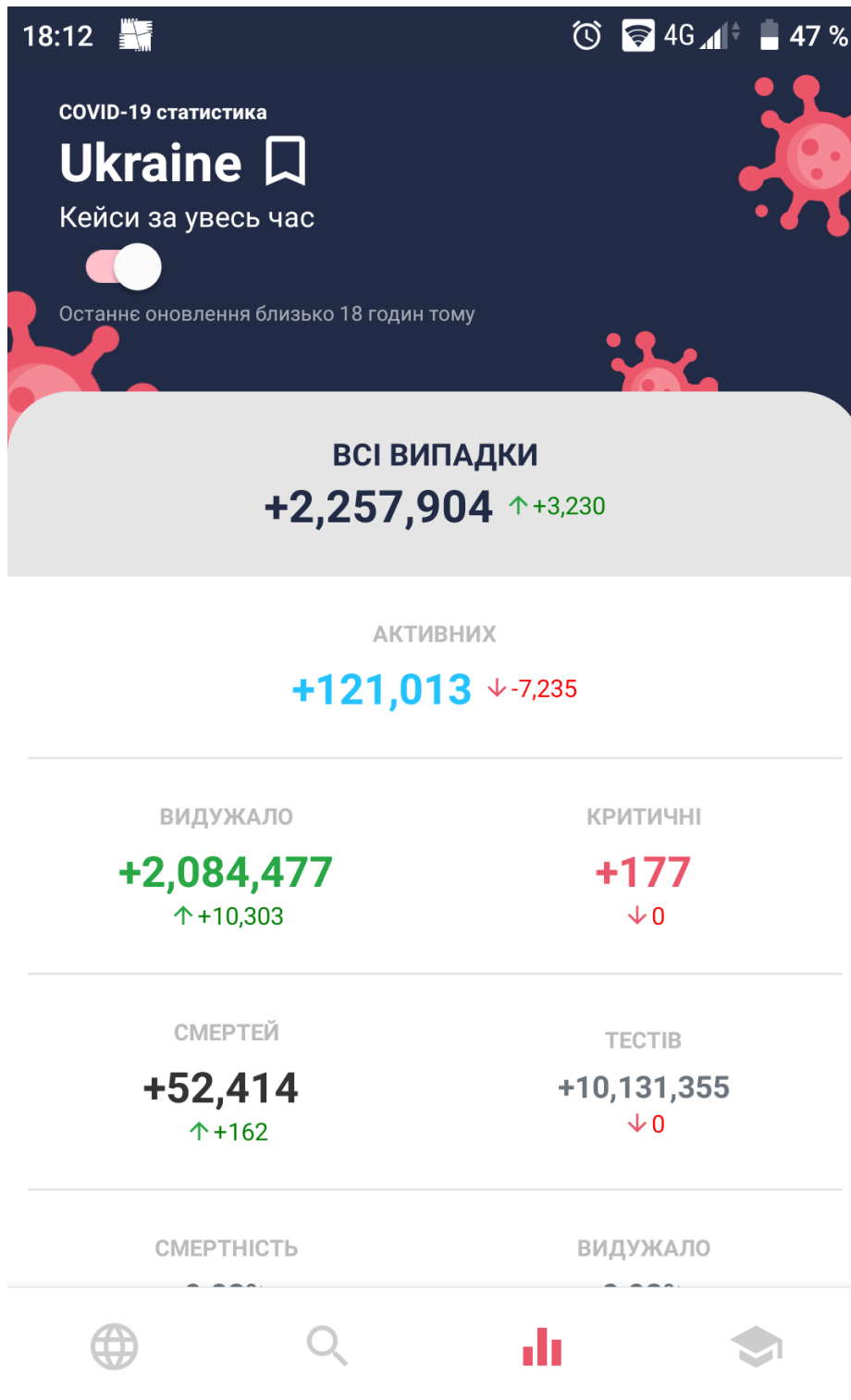


Рис. 3.6. Екран додатку із загальною статистикою захворюваності в країні

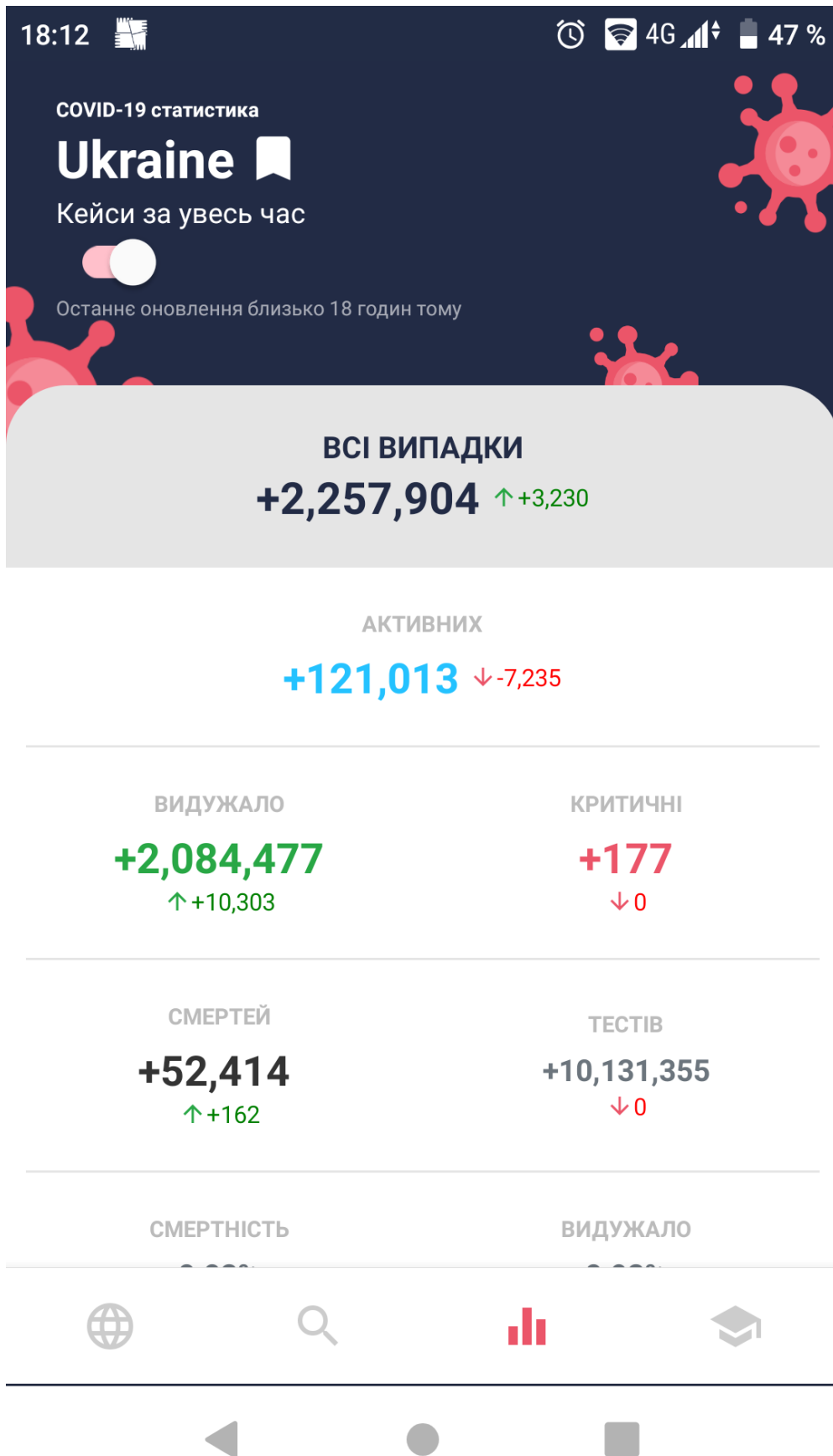


Рис. 3.7. Екран додатку з країною, що збережена до закладок

Після того, як користувач додав країну до списку збережених (натиснувши на закладку поруч з назвою країни) – вона відображається у списку збережених країн, який відображається за допомогою свайпу вправо з головного екрану додатку. Містить наступні елементи – автора, назву додатку, логотип та сам список збережених країн, до котрих користувач в один клік може отримати доступ й переглянути потрібну інформацію без зайвих зусиль й витрат часу.

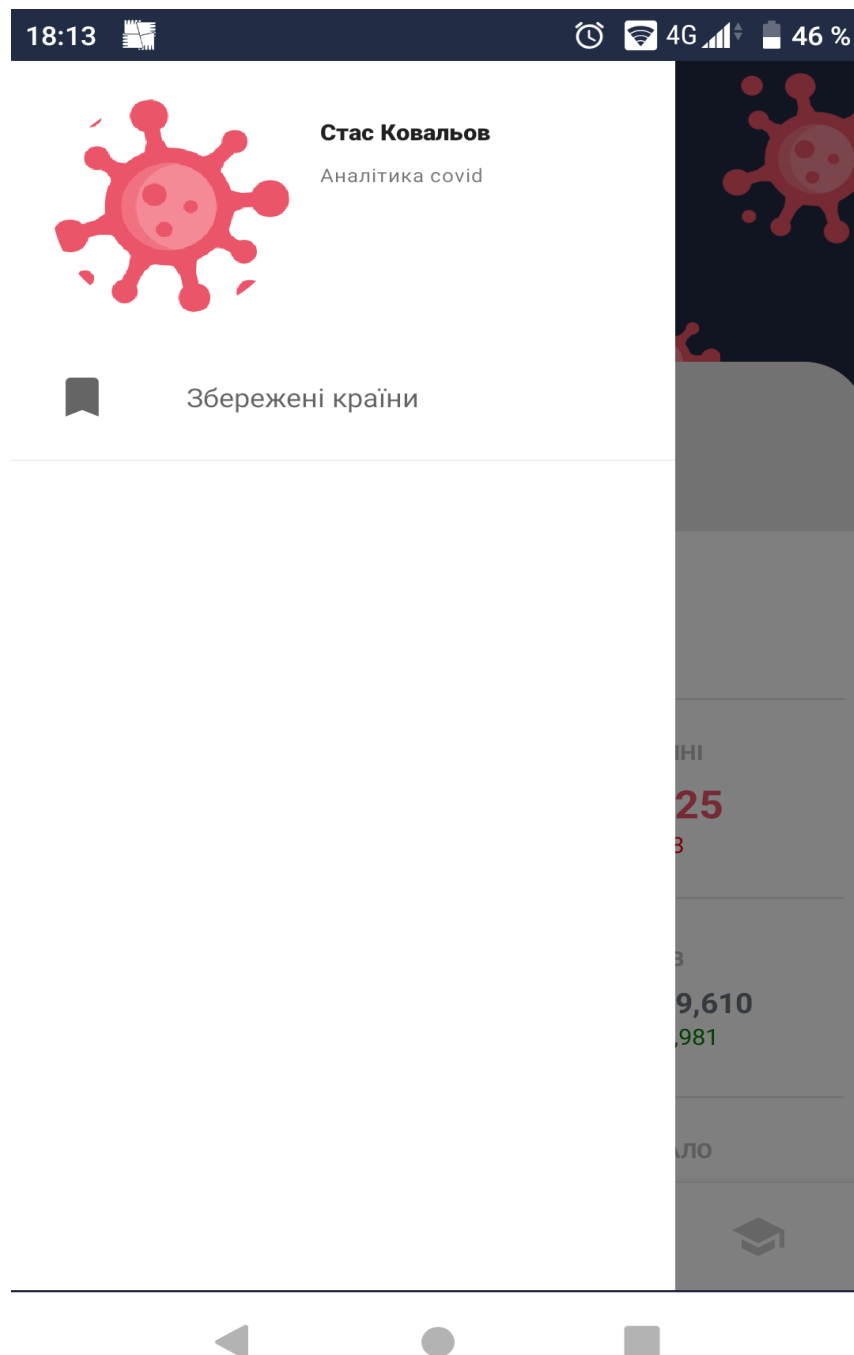


Рис. 3.8. Головний екран додатку зі статистикою на даний момент

Натиснувши на пункт «Збережені країни», користувач опиняється на екрані з країнами, що були додані в закладки – в даному прикладі було додано Україну в список збережених країн, натиснувши на назву країни, користувач опиниться на екрані перегляду статистики країни – в іншому випадку він може повернутись назад – натиснувши кнопку «Повернутись назад», чи видалити країну зі списку збережених – натиснувши на хрестик.

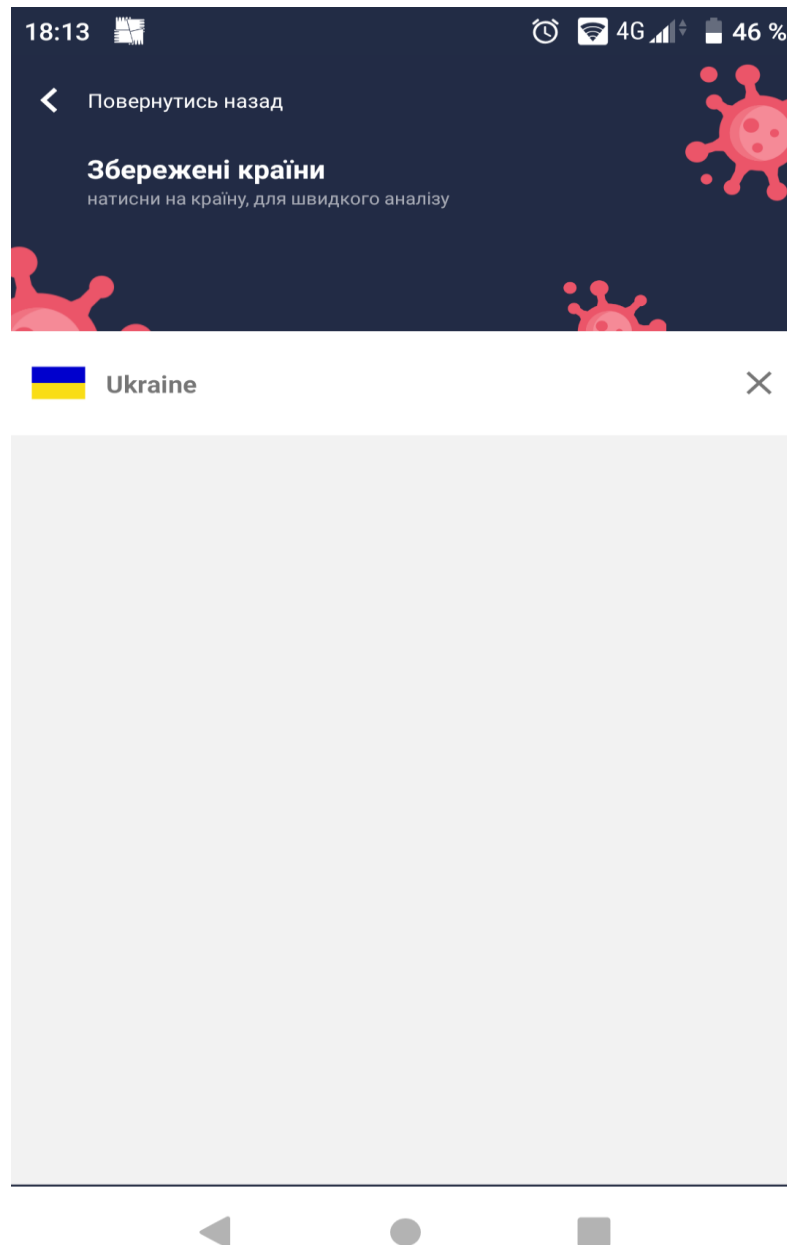


Рис. 3.9. Екран зі збереженими країнами світу

Наступною кнопкою є «Як захиститись від COVID-19»[10], що відображає екран на якому зображені основні симптоми коронавірусної хвороби – Як визначити свої симптоми? (Рис. 3.10.) – профілактика – Як застерегти себе? (Рис. 3.11.) – поширення інфекції – «Як зупинити поширення?» (Рис. 3.12.).

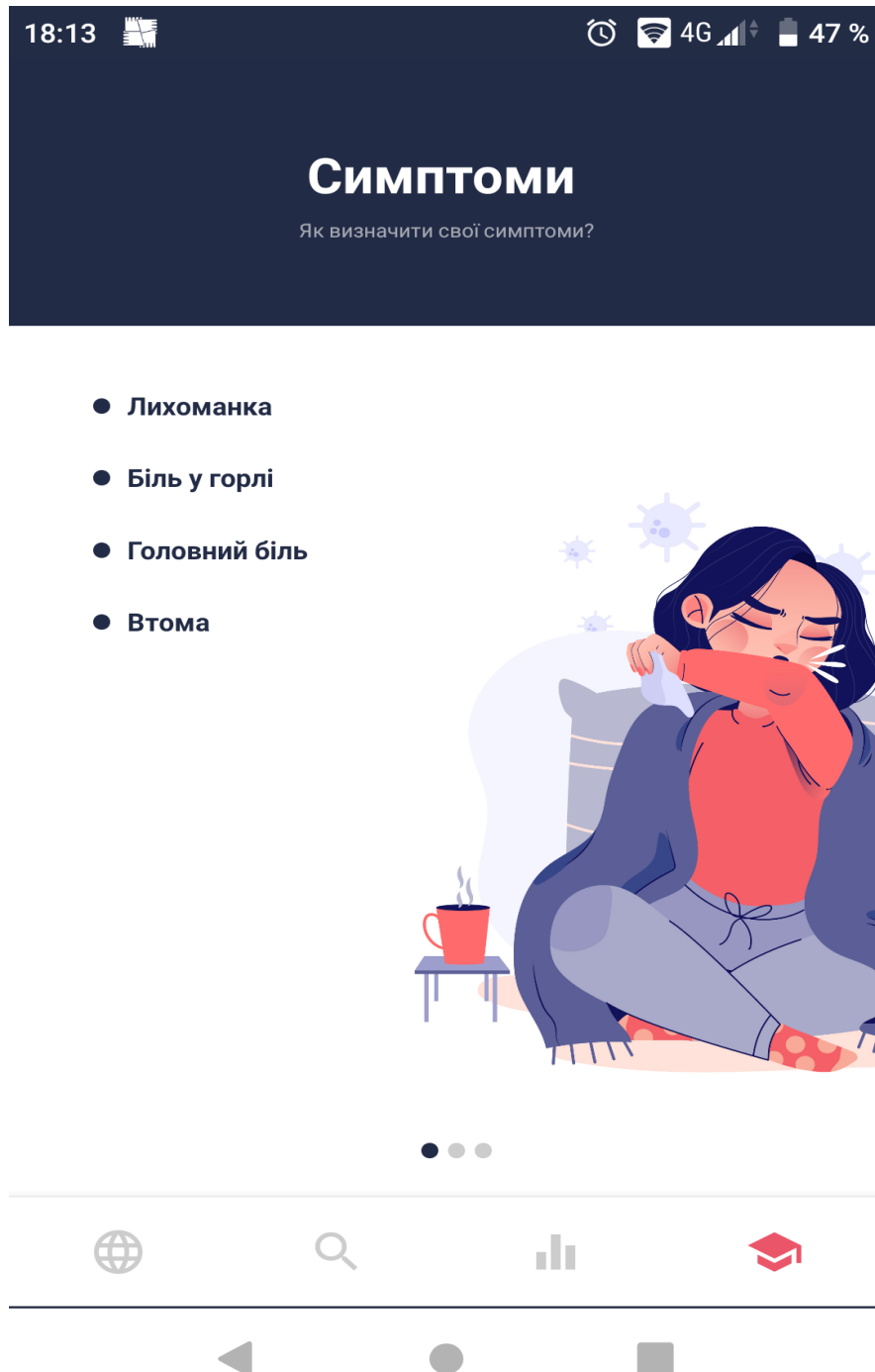


Рис. 3.10. Екран симптомів коронавірусної хвороби

18:13



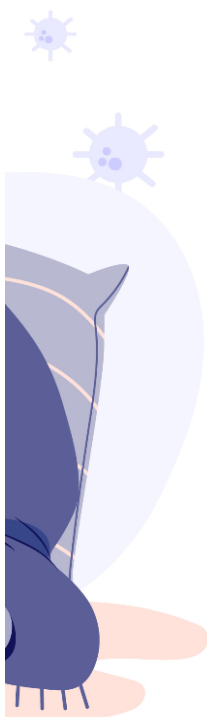
4G



46 %

Профілактика

Як застерегти себе?



- Мийте руки
- Уникайте місць, де багато людей
- Носіть маску
- Тримайте безпечну дистанцію



Рис. 3.11. Екран профілактики коронавірусної хвороби

Поширення інфекції

Як зупинити поширення?

- Виконувати кашльовий етикет
- Очищуйте поверхні та предмети, до яких часто торкаєтеся
- Будьте готові та поінформовані

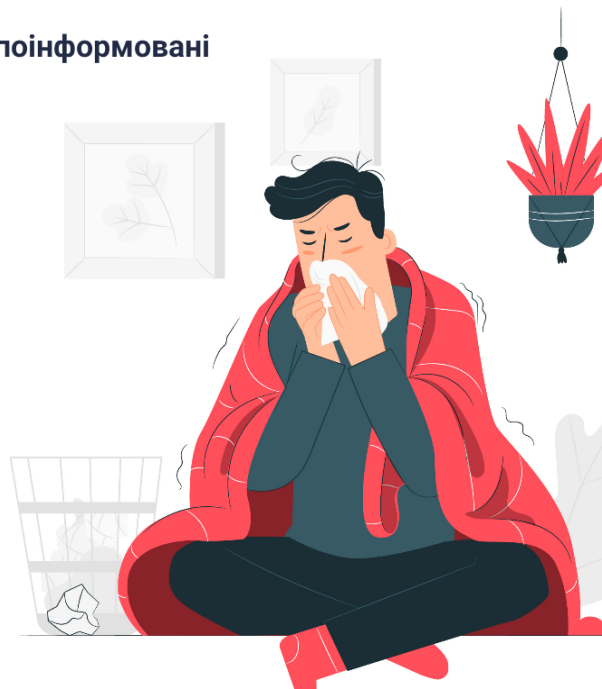


Рис. 3.12. Екран запобігання поширенню інфекції

3.2. Тестування додатку

Перевірку функціональності даного додатку аналізу захворюваності коронавірусною хворобою *COVID-19* варто почати з достовірності обновлюваних даних.

Перш за все, переглянемо статистику в Україні за сьогоднішній день:

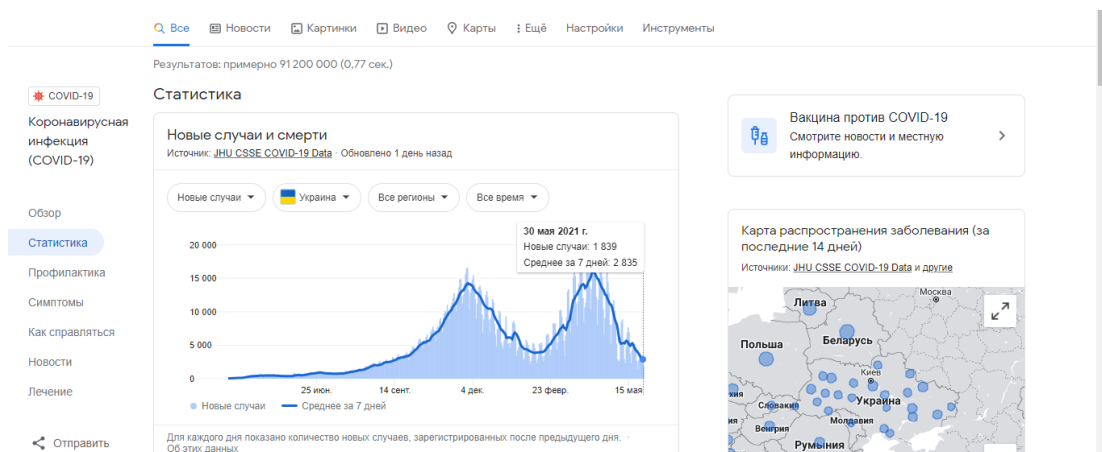


Рис. 3.13. Статистика захворюваності *COVID-19* в Україні

Як бачимо, кількість нових випадків – 1839, датою 30 травня 2021 року.

Тепер перевіримо дані в додатку аналізу даних захворюваності коронавірусною хворобою *COVID-19*, що був створений в ході цієї дипломної роботи:

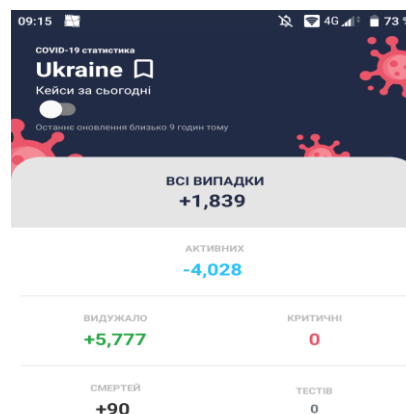


Рис. 3.14. Статистика захворюваності *COVID-19* в Україні в створеному додатку

Як видно з порівняння даних, додаток працює коректно, тому роботу можна вважати успішно завершеною.

3.3. Висновки до розділу

В даному розділі було проаналізовано додаток на *IOS/ANDROID* аналізу даних захворюваності коронавірусною хворобою *COVID-19* в країнах світу. Перш за все було розглянуто створений дизайн додатку – а також інтерфейс. Розглянуто й проаналізовано кожен створений екран додатку, виділено ключові моменти та функціональні особливості створених екранів.

Крім цього було проведено тестування додатку – перевірка на достовірність отримуваних даних. Тестування пройшло успішно, дані отримуються достовірні, а отже дипломну роботу як і створення додатку на *IOS/ANDROID* аналізу даних захворюваності коронавірусною хворобою *COVID-19* в країнах світу можна вважати успішно завершеною.

ВИСНОВКИ

В першому розділі: у результаті проведеного аналізу було визначено задачу створення додатку аналізу даних захворюваності коронавірусною хворобою, проаналізовано й розглянуто аналоги додатку аналізу захворюваності *COVID-19* на основі яких були побудовані прототипи екранів майбутнього додатку та розроблено дизайн.

В другому розділі: було проаналізовано інструменти та технології для створення мобільного додатку. До цих інструментів увійшли:

1. *IDE: Visual Studio Code*;
2. Мова програмування *JavaScript*;
3. Фреймворк *React Native*;
4. *API coronavirus map* для статистики в світі.

В третьому розділі: було проаналізовано додаток на *IOS/ANDROID* аналізу даних захворюваності коронавірусною хворобою *COVID-19* в країнах світу. Перш за все було розглянуто створений дизайн додатку – а також інтерфейс. Розглянуто й проаналізовано кожен створений екран додатку, виділено ключові моменти та функціональні особливості створених екранів.

Крім цього було проведено тестування додатку – перевірка на достовірність отримуваних даних. Тестування пройшло успішно, дані отримуються достовірні, а отже дипломну роботу як і створення додатку на *IOS/ANDROID* аналізу даних захворюваності коронавірусною хворобою *COVID-19* в країнах світу можна вважати успішно завершеною.

СПИСОК БІБЛІОГРАФІЧНИХ ПОСИЛАНЬ ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Коронавірус COVID-19 [Електронний ресурс] – Режим доступу: <https://www.who.int/ru/emergencies/diseases/novel-coronavirus-2019> (дата звернення 15 травня 2021р.). – Назва з екрана.
2. *Trecker COVID-19* [Електронний ресурс] – Режим доступу: <https://github.com/zoe/covid-tracker-react-native> (дата звернення 15 травня 2021р.). – Назва з екрана.
3. *Figma* [Електронний ресурс] – Режим доступу: <https://www.figma.com/> (дата звернення 15 травня 2021р.). – Назва з екрана.
4. *Visual Studio Code* [Електронний ресурс] – Режим доступу: <https://code.visualstudio.com/> (дата звернення 15 травня 2021р.). – Назва з екрана.
5. *JavaScript* [Електронний ресурс] – Режим доступу: https://developer.mozilla.org/ru/docs/Learn/JavaScript/First_steps/What_is_JavaScript (дата звернення 15 травня 2021р.). – Назва з екрана.
6. *React Native* [Електронний ресурс] – Режим доступу: <https://reactnative.dev/> (дата звернення 15 травня 2021р.). – Назва з екрана.
7. *CSS* [Електронний ресурс] – Режим доступу: <https://developer.mozilla.org/ru/docs/Web/CSS> (дата звернення 15 травня 2021р.). – Назва з екрана.
8. *RapidAPI* [Електронний ресурс] – Режим доступу: <https://rapidapi.com/> (дата звернення 15 травня 2021р.). – Назва з екрана.
9. *Coronavirus map* [Електронний ресурс] – Режим доступу: <https://rapidapi.com/Yatko/api/coronavirus-map> (дата звернення 15 травня 2021р.). – Назва з екрана.
10. Вопросы и ответы COVID-19 [Електронний ресурс] – Режим доступу: <https://www.who.int/ru/emergencies/diseases/novel-coronavirus-2019/advice-for-public/q-a-coronaviruses> (дата звернення 15 травня 2021р.). – Назва з екрана.
11. *Android* [Електронний ресурс] – Режим доступу: https://www.android.com/intl/ru_ru/ (дата звернення 15 травня 2021р.). – Назва з екрана.

12. *iOs* [Електронний ресурс] – Режим доступу: <https://www.apple.com/ru/ios/ios-14/> (дата звернення 15 травня 2021р.). – Назва з екрана.
13. *Web Dev* [Електронний ресурс] – Режим доступу: <https://web.dev/> (дата звернення 15 травня 2021р.). – Назва з екрана.
14. *ZOE* [Електронний ресурс] – Режим доступу: <https://covid.joinzoe.com/us/privacy> (дата звернення 15 травня 2021р.). – Назва з екрана.
15. Что такое *API* [Електронний ресурс] – Режим доступу: <https://dev.by/news/что-такое-api-prostym-yazykom> (дата звернення 15 травня 2021р.). – Назва з екрана.
16. ДСТУ 3008-95 Документація. Звіти у сфері науки і техніки. – К.: Держстандарт України, 1995. – 39 с.
17. ГОСТ 2.106-96 [Електронний ресурс] – Режим доступу: <https://files.stroyinf.ru/Data/46/4669.pdf> (дата звернення 15 травня 2021р.). – Назва з екрана.