

**НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ
імені ІГОРЯ СІКОРСЬКОГО»**

ФАКУЛЬТЕТ ПРИКЛАДНОЇ МАТЕМАТИКИ

Кафедра системного програмування і спеціалізованих комп'ютерних систем

«До захисту допущено»

Завідувач кафедри

_____ Романкевич В.О.
(підпис) (ініціали, прізвище)

“ ____ ” червня ____ 2021 р.

**Дипломний проєкт
на здобуття ступеня бакалавра**

зі спеціальності **123 «Комп'ютерна інженерія»**

на тему: **Мобільний додаток для перегляду новин на базі платформи iOS**

Виконав: студент IV курсу, групи КВ-72
(шифр групи)

_____ **Беляєв Ілля Дмитрович** _____
(прізвище, ім'я, по батькові) (підпис)

Керівник доц. каф. СПіСКС, к. т. н., доцент Тарасенко-Клятченко О.В. _____
(посада, науковий ступінь, вчене звання, прізвище та ініціали) (підпис)

Консультант з нормоконтролю к.т.н. Клятченко Я.М. _____
(назва розділу) (посада, вчене звання, науковий ступінь, прізвище, ініціали)
(підпис)

Рецензент к.т.н., ст. вик. ПЗКС ФПМ Хіцко Я.В. _____
(посада, науковий ступінь, вчене звання, науковий ступінь, прізвище та ініціали) (підпис)

Засвідчую, що у цьому дипломному проєкті
немає запозичень з праць інших авторів без
відповідних посилань.

Студент _____
(підпис)

Київ – 2021 року

ВІДОМІСТЬ ДИПЛОМНОГО ПРОЄКТУ

№ з/п	Формат	Позначення	Найменування	Кількість листів	Примітка
1	A4		Завдання на дипломне проєктування	2	
2	A4	ІАЛЦ. 045490.001 ОА	Опис альбому	2	
3	A4	ІАЛЦ. 045490.002 ТЗ	Технічне завдання	4	
4	A4	ІАЛЦ. 045490.003 ТП	Відомість технічного проєкту	2	
5	A4	ІАЛЦ. 045490.004 ПЗ	Пояснювальна записка	65	
6	A4	ІАЛЦ. 045490.005 Д1	Взаємозв'язки основних складових додатку. Схема структурна	1	
7	A4	ІАЛЦ. 045490.006 Д2	Оновлення новин в модулі новин України. Схема алгоритму	1	
8	A4	ІАЛЦ. 045490.007 Д3	Перехід між картками з новинами в модулі тематичних новин. Схема алгоритму	1	
9	A4	ІАЛЦ. 045490.008 Д4	Загальна схема роботи мобільного додатку на базі платформи iOS. Схема алгоритму	1	

				ІАЛЦ.045490.000		
	ПІБ	Підп.	Дата			
Розробн.	Беляєв І.Д.			Відомість дипломного проєкту	Лист	Листів
Керівн.	Тарасенко-Клятенко О.В.				1	1
Консульт.					КПІ ім. Ігоря Сікорського Каф. СПіСКС Гр. КВ-72	
Н/контр.	Клятенко Я.М.					
Зав.каф.	Романкевич В.О.					

**НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ
імені ІГОРЯ СІКОРСЬКОГО»**

ФАКУЛЬТЕТ ПРИКЛАДНОЇ МАТЕМАТИКИ

Кафедра системного програмування і спеціалізованих комп'ютерних систем

Рівень вищої освіти – перший (бакалаврський)

Спеціальність 123 «Комп'ютерна інженерія»

Освітньо-професійна програма «Системне програмування та спеціалізовані комп'ютерні системи»

ЗАТВЕРДЖУЮ

Завідувач кафедри

_____ Віталій РОМАНКЕВИЧ
(підпис) (ініціали, прізвище)

«__» _____ 2021 р.

**ЗАВДАННЯ
на дипломний проєкт студента
Беляєва Іллі Дмитровича**

1. Тема проєкту «Мобільний додаток для перегляду новин на базі платформи iOS»,
керівник проєкту доц. каф. СПіСКС, к.т.н., доцент Тарасенко-Клятченко О.В.,

затверджені наказом по університету від «__» _____ 2021 р. № _____

2. Термін подання студентом проєкту _____

3. Вихідні дані до проєкту: розроблений мобільний iOS-додаток із зручним інтерфейсом та потужним функціоналом для перегляду новин України та закордону.

4. Зміст пояснювальної записки: огляд існуючих рішень та обґрунтування теми дипломного проєкту; огляд вибору засобів реалізації; опис розробленого програмного забезпечення; висновки.

5. Перелік графічного матеріалу: Взаємозв'язки основних складових додатку. Схема структурна; Оновлення новин в модулі новин України. Схема алгоритму;

Перехід між картками з новинами в модулі тематичних новин. Схема алгоритму; Загальна схема роботи мобільного додатку на базі платформи iOS. Схема алгоритму.

6. Консультанти розділів проєкту

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
Норм-контроль	Клятченко Я.М.		

7. Дата видачі завдання _____

Календарний план

№ з/п	Назва етапів виконання дипломного проєкту	Термін виконання етапів проєкту	Примітка
1	Видача завдання на проєктування	04.12.2020	
2	Вивчення літератури за тематикою роботи	03.02.2021	
3	Розроблення та узгодження технічного завдання	04.03.2021	
4	Аналіз існуючих рішень	14.03.2021	
5	Розроблення схеми зберігання даних у віддаленому сховищі та заповнення БД інформацією	22.03.2021	
6	Налаштування пошукової системи	29.03.2021	
7	Побудова інтерфейсу додатку	11.04.2021	
8	Програмування логіки роботи застосунку	30.04.2021	
9	Оформлення документації проєкту	21.05.2021	

Студент

(підпис)

Ілля БЄЛЯЄВ

(Ім'я та ПРІЗВИЩЕ)

Керівник проєкту

(підпис)

Оксана ТАРАСЕНКО-КЛЯТЧЕНКО

(Ім'я та ПРІЗВИЩЕ)

АНОТАЦІЯ

Кваліфікаційна робота включає пояснювальну записку (65 с., 15 рис., 1 табл., 3 додатки).

Об'єкт розробки – мобільний додаток для перегляду новин на базі платформи iOS, який дає можливість оперативно отримувати свіжу інформацію про останні події в Україні та світі.

Функціонал мобільного застосунку полягає у можливості окремо переглядати новини українських ЗМІ щодо України, а також інформацію про події в інших країнах як від вітчизняних засобів масової інформації, так і іноземних іншомовних; можливості перегляду новин України або закордону за обраною темою; функції повнотекстового пошуку новин за останній тиждень у віддаленій базі даних; функції збереження у постійну пам'ять пристрою змісту обраних статей; можливості прослухати текст новини, промовлений природнім людським голосом. Для реалізації даного функціоналу була використана мова програмування Swift, засоби побудову дизайну Storyboard та XIB-файли, мобільна БД Realm, хмарна платформа Firebase та сервіс Algolia в якості віддаленої БД і пошукової системи відповідно, а також хмарний синтезатор мови з тексту – Google Cloud Text-To Speech.

У процесі проектування:

- висунуто вимоги до мобільного додатку перегляду новин,
- проаналізовано переваги та недоліки існуючих аналогів,
- побудований дизайн інтерфейсу застосунку,
- запрограмована логіка роботи iOS додатку,
- розроблена схема зберігання даних у віддаленому сховищі
- налаштована взаємодія сервісу пошуку Algolia з Firebase та мобільним програмним забезпеченням.

Створення такої програми дозволить користувачам смартфонів від корпорації Apple мати на своєму пристрої дійсно зручний, з оригінальним дизайном та потужний за функціоналом засіб для перегляду новин України та закордону.

Ключові слова:

МОБІЛЬНИЙ ДОДАТОК, МОБІЛЬНИЙ ЗАСТОСУНОК, IOS, НОВИНИ,
FIREBASE, ALGOLIA, SWIFT, REALM, GOOGLE CLOUD TEXT-TO-SPEECH

ABSTRACT

Qualifying work includes explanatory note (65 p., 15 pic., 1 table, 3 additions).

Object of developing – mobile app for viewing news on the base of iOS platform that gives an opportunity to get fresh information about the latest events in Ukraine and world.

Functionality of this mobile application consists in: opportunity to view separately news by ukrainian mass media concerning Ukraine and information about events in other countries (by domestic mass media and foreign media on different languages); opportunity of viewing Ukrainian or world news on selected topic; function of full-text search news of the latest week in the remote database; function of storing selected issues content in permanent device memory; opportunity to listen to news text spoken with natural human voice. To realize this functionality were used programming language Swift, design creating tools Storyboard and XIB-files, mobile database Realm, cloud platform Firebase and service Algolia as remote database and search engine accordingly, and also cloud synthesizer language from text - Google Cloud Text-To Speech.

During the process of projecting:

- were put forward requirements on mobile app for news viewing
- was analyzed pros and cons of existing analogs
- was created design of application interface
- was coded work logic of iOS app
- was developed scheme of data keeping in remote storage
- was set up Algolia search service interaction with Firebase and mobile software.

Such program creation allows users of Apple Inc. smartphones to have on own devices really convenient, with original design, functionally powerful remedy for viewing Ukraine and world news.

Keywords:

MOBILE APP, MOBILE APPLICATION, IOS, NEWS, FIREBASE, ALGOLIA, SWIFT, REALM, GOOGLE CLOUD TEXT-TO-SPEECH

Поз.	Формат	ПОЗНАЧЕННЯ	НАЙМЕНУВАННЯ	Кількість аркушів	№ прим.	Примітки
	A4	ІАЛЦ. 045490.002 ТЗ	Мобільний додаток для перегляду новин на базі платформи iOS	4		
			Технічне завдання			
	A4	ІАЛЦ. 045490.003 ТП	Мобільний додаток для перегляду новин на базі платформи iOS	2		
			Відомість технічного проєкту			
	A4	ІАЛЦ. 045490.004 ПЗ	Мобільний додаток для перегляду новин на базі платформи iOS	65		
			Пояснювальна записка			
	A1	ІАЛЦ. 045490.005 Д1	Взаємозв'язки основних складових додатку.	1		
			Схема структурна.			

					ІАЛЦ.045490.001 ОА				
Змін.	Арк.	№ докум.	Підпис	Дата					
Розробив	Белясв І.Д.				Мобільний додаток для перегляду новин на базі платформи iOS		Літ.	Аркуш	Аркушів
Перевірив	Тарасенко-							1	2
	Клятченко О. В.						КПІ ім. Ігоря Сікорського, ФПМ КВ-72		
Н. контроль	Клятченко Я.М.								
Затвердив	Романкевич В.О								
					Опис альбому				

[illegible]

ЗМІСТ

1. НАЙМЕНУВАННЯ ТА ГАЛУЗЬ РОЗРОБКИ.....	2
2. ПІДСТАВА ДЛЯ РОЗРОБКИ.....	2
3. ЦІЛЬ І ПРИЗНАЧЕННЯ РОБОТИ.....	2
4. ДЖЕРЕЛА РОЗРОБКИ.....	2
5. ТЕХНІЧНІ ВИМОГИ.....	2
5.1. Вимоги до програмного продукту, що розробляється.....	2
5.2. Вимоги до апаратного забезпечення.....	3
5.3. Вимоги до програмного та апаратного забезпечення користувача.....	3
6. ЕТАПИ РОЗРОБКИ.....	4

					ІАЛЦ. 045490.002 ТЗ							
Зм.	Арк.	№ докум.	Підп.	Дата								
Розроб.		Беляєв І.Д.			Мобільний додаток для перегляду новин на базі платформи iOS Технічне завдання				Літ.	Аркуш	Аркуші	
Перев.		Тарасенко-									1	4
		Клятченко О.В.										
Н. контр.		Клятченко Я.М.										
Затв.		Романкевич В.О.										
					КПІ ім. Ігоря Сікорського, ФПМ, КВ-72							

1. НАЙМЕНУВАННЯ ТА ГАЛУЗЬ РОЗРОБКИ

Назва розробки: «Мобільний додаток для перегляду новин на базі платформи iOS».

Галузь застосування: організація контакту людей з інформацією, що поширює ЗМІ.

2. ПІДСТАВА ДЛЯ РОЗРОБКИ

Підставою для розробки є завдання на дипломне проектування на здобуття першого (бакалаврського) рівня вищої освіти, затверджене кафедрою системного програмування і спеціалізованих комп'ютерних систем Національного технічного університету України «Київський Політехнічний Інститут імені Ігоря Сікорського».

3. МЕТА І ПРИЗНАЧЕННЯ РОБОТИ

Метою даного проекту є створення застосунку на базі мобільної платформи iOS для зручного оперативного ознайомлення з останніми новинами України та закордону, що інтенсивно генеруються ЗМІ.

4. ДЖЕРЕЛА РОЗРОБКИ

Джерелом інформації є технічна та науково-технічна література, технічна документація, публікації у періодичних виданнях та електронні статті у мережі Інтернет.

5. ТЕХНІЧНІ ВИМОГИ

5.1 Вимоги до програмного продукту, що розробляється

- сумісність з операційною системою iOS
- перегляд змісту окремої новини
- можливість окремого перегляду новин України або закордону

					ІАЛЦ.045490.002 ТЗ	Лист
						2
Зм	Арк.	№ докум.	Підп.	Дата		

- можливість окремого перегляду окремо українських або закордонних новин за обраною темою
- пошук новинних статей за останній тиждень за введеним словом чи словосполученням у віддаленому сховищі
- збереження змісту окремих обраних статей у постійну пам'ять смартфона
- озвучування текстової частини змісту новини природнім людським голосом.

5.2 Вимоги до апаратного забезпечення

- Процесор: Apple A9 та новіше
- Оперативна пам'ять: 2 Гб
- Доступ до мережі Internet (GRPS, 3G, 4G)

5.3 Вимоги до програмного та апаратного забезпечення користувача

- Операційна система iOS версії 14.4 та новіше

					ІАЛЦ.045490.002 ТЗ	Лист
						3
Зм	Арк.	№ докум.	Підп.	Дата		

6. ЕТАПИ РОЗРОБКИ

№ з/п	Назва етапів виконання дипломного проєкту	Термін виконання етапів
1.	Видача завдання на проєктування	04.12.2020
2.	Вивчення літератури за тематикою роботи	03.02.2021
3.	Розроблення та узгодження технічного завдання	04.03.2021
4.	Аналіз існуючих рішень	14.03.2021
5.	Розроблення схеми зберігання даних у віддаленому сховищі та заповнення БД інформацією	22.03.2021
6.	Налаштування пошукової системи	29.03.2021
7.	Побудова інтерфейсу додатка	11.04.2021
8.	Програмування логіки роботи застосунку	30.04.2021
10.	Оформлення документації проєкту	21.05.2021

Поз.	Формат	ПОЗНАЧЕННЯ	НАЙМЕНУВАННЯ	Кількість аркушів	№ прим.	Примітки
	A4	ІАЛЦ. 045490.004 ПЗ	Мобільний додаток для перегляду новин на базі платформи iOS	65		
			Пояснювальна записка			
	A1	ІАЛЦ. 045490.005 Д1	Взаємозв'язки основних складових додатку. Схема структурна.	1		
	A1	ІАЛЦ. 045490.006 Д2	Оновлення новин в модулі новин України. Схема алгоритму.	1		
	A1	ІАЛЦ. 045490.007 Д3	Перехід між картками з новинами в модулі тематичних новин. Схема алгоритму.	1		

					<h2>ІАЛЦ.045490.003 ТП</h2>			
Змін.	Арк.	№ докум.	Підпис	Дата	<p>Мобільний додаток для перегляду новин на базі платформи iOS</p> <p>Відомість технічного проєкту</p>	Літ.	Аркуш	Аркушів
Розробив	Беляєв І.Д.						1	2
Перевірив	Тарасенко-							
	Клятченко О. В.							
Н. контроль	Клятченко Я.М.							
Затвердив	Романкевич В.О					<p>КПІ ім. Ігоря Сікорського, ФПМ KB-72</p>		

[illegible]

ЗМІСТ

СПИСОК ТЕРМІНІВ, СКОРОЧЕНЬ ТА ПОЗНАЧЕНЬ.....	4
ВСТУП.....	6
1. ОГЛЯД ІСНУЮЧИХ РІШЕНЬ ТА ОБГРУНТУВАННЯ ТЕМИ ДИПЛОМНОГО ПРОЄКТУ	8
1.1. Обґрунтування вибору критеріїв оцінювання існуючих рішень.....	8
1.2. Аналіз існуючих рішень.....	9
1.3. Порівняльна таблиця існуючих програмних рішень.....	13
1.4. Висновок.....	14
2. ОГЛЯД ВИБОРУ ЗАСОБІВ РЕАЛІЗАЦІЇ.....	17
2.1.Обґрунтування вибору мови програмування для розробки мобільного застосунку на базі платформи iOS.....	17
2.1.1. Objective-C.....	18
2.1.2. Swift.....	20
2.1.3. Висновки та обґрунтування вибору мови програмування.....	22
2.2.Обґрунтування вибору засобу розробки дизайну користувацького інтерфейсу iOS додатку.....	23
2.2.1. Storyboards.....	23
2.2.2. XIB.....	24
2.2.3. Мова нативного програмування.....	25
2.2.4. SwiftUI.....	26
2.2.5. Висновки.....	28
2.3.Обґрунтування вибору засобу для зберігання даних у постійній пам'яті мобільного пристрою.....	28
2.3.1. UserDefaults.....	29

					ІАЛЦ.045490.004 ПЗ					
Зм.	Арк.	№ докум.	Підп.	Дата						
Розроб.		Беляєв І.Д.			Мобільний додаток для перегляду новин на базі платформи iOS			Літ.	Аркуш	Аркуші
Перев.		Тарасенко-							1	65
		Клятенко О.В.						КПІ ім. Ігоря Сікорського, ФПМ, КВ-72		
Н. контр.		Клятенко Я.М.								
Затв.		Романкевич В.О.								
					Пояснювальна записка					

2.3.2. NSCoder та протокол Codable.....	30
2.3.3. SQLite.....	31
2.3.4. CoreData.....	32
2.3.5. Realm.....	33
2.3.6. Висновки.....	34
2.4.Обґрунтування вибору засобу для перетворення інформації з текстового представлення в аудіопредставлення.....	36
2.4.1. AVSpeechSynthesizer.....	37
2.4.2. Amazon Polly.....	37
2.4.3. Google Cloud Text-To-Speech.....	38
2.4.4. Висновки та обґрунтування вибору засобу.....	39
2.5.Обґрунтування вибору віддаленого сховища та технологій повнотекстового пошуку.....	39
2.6.Висновки.....	40
3. ОПИС РОЗРОБЛЕНОГО ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ.....	41
3.1.Загальний опис структури та роботи додатку.....	41
3.2.Опис структури віддаленої БД та пошукової системи.....	43
3.3.Опис модуля новин України.....	45
3.4.Опис модуля новин світу (закордону).....	49
3.5.Опис модуля повнотекстового пошуку.....	50
3.6.Опис модуля збережених новин.....	52
3.7.Опис модуля тематичних новин.....	53
3.8.Опис модуля перегляду новинної статті.....	56
3.9.Опис модуля фільтрації закордонних новин.....	57
3.10.Рекомендації щодо подальшого вдосконалення.....	59
ВИСНОВКИ.....	61
СПИСОК ВИКОРИСТАНИХ ЛІТЕРАТУРНИХ ДЖЕРЕЛ.....	63
ДОДАТКИ	

Додаток 1. Копії графічних матеріалів

- ІАЛЦ.045490.005 Д1.Взаємозв'язки основних складових додатку. Схема структурна
- ІАЛЦ.045490.006 Д2. Оновлення новин в модулі новин України. Схема алгоритму
- ІАЛЦ.045490.007 Д3. Перехід між картками з новинами в модулі тематичних новин. Схема алгоритму
- ІАЛЦ.045490.008 Д4. Загальна схема роботи мобільного додатку на базі платформи iOS. Схема алгоритму.

Додаток 2. Фрагменти програмного коду.

Додаток 3. Презентація.

					ІАЛЦ.045490.004 ПЗ	Лист
						3
Зм	Арк.	№ докум.	Підп.	Дата		

СПИСОК ТЕРМІНІВ, СКОРОЧЕНЬ ТА ПОЗНАЧЕНЬ

БД – база даних.

Девайс – персональний цифровий пристрій

ЗМІ – засіб масової інформації.

Кросплатформенний – призначений для використання одразу на декількох платформах.

Нативний – призначений для застосування виключно щодо чогось одного конкретного, з яким має максимальну безумовну сумісність.

ОС – операційна система.

Повнотекстовий пошук – пошук документів у базі даних по введених словах, який враховує увесь текстовий зміст документу.

Смартфон – цифровий пристрій, що поєднує в собі функції мобільного телефона та велику частину функціонала персонального комп'ютера.

СУБД – система управління базами даних.

Autolayout – створена фірмою Apple технологія, яка за заданими програмістом закономірностями взаєморозташування елементів на екрані автоматично розраховує розміри кожного елементу для будь-якого розміру екрану (моделі смартфона).

BaaS – Backend as a Service. Відомий також як MBaaS або Mobile Backend as a Service. Модель впровадження технологій серверної частини, яка передбачає використання вже розроблених комерційних програмних рішень віддаленого сховища та серверних API під час розробки веб та ряду мобільних додатків для того, щоб основна увага приділялась створенню саме клієнтського програмного забезпечення.

IDE – Integrated Development Environment (інтегроване середовище розробки). Програма, яка використовується в якості середовища розробки певного програмного забезпечення і яка містить в собі усі необхідні для цього різноманітні утиліти - від текстового редактора до інструментів тестування

					ІАЛЦ.045490.004 ПЗ	Лист
Зм	Арк.	№ докум.	Підп.	Дата		4

та контролю версій.

iOS – мобільна операційна система для смартфонів фірми Apple (iPhone).

RAM – Random Access Memory (пам'ять з довільним доступом). Тип комп'ютерної пам'яті, що використовується для роботи з тимчасовими даними під час виконання операцій.

XIB – XML Interface Builder. Конструктор інтерфейсу, робота якого заснована на оперуванні даними в форматі xml.

XML – Extensible Markup Language. Мова розмітки, яка визначає набір правил для такого кодування документів, при якому вони можуть бути зрозумілі як людині, так і машині.

					ІАЛЦ.045490.004 ПЗ	Лист
						5
Зм	Арк.	№ докум.	Підп.	Дата		

ВСТУП

У сучасному світі володіння цінною інформацією та потрібними знаннями допомагає людині самовдосконалюватись і приймати якомога більш ефективні, вірні рішення для досягнення власних малих та великих цілей.

Людство ніколи не стоїть на місці – кожен день на Землі десь завжди трапляються певні зміни. Люди завжди хочуть бути в курсі останніх подій для того, щоб оперативно реагувати відповідним чином на обставини, що склались. Це і є причиною постійного попиту на такий тип інформації як новини.

Дізнаватись про них можна різними способами. Деякі з цих методів існують ще зі стародавніх часів (наприклад, між людьми в очному спілкуванні). Інші шляхи отримання новин, такі як поліграфічні джерела, радіо, телебачення є більш молодими. З розвитком цифрових технічних засобів та мережі Інтернет передача інформації на великі відстані стала ще простішою та швидшою. Через це діяльність людей та їх взаємодія один з одним стали набагато більш інтенсивними (особливо це відчувається у мегаполісах). В свою чергу динаміка щоденних подій та змін у світі також дуже зростає. Отримувати новини тільки з друкованих щотижневих видань, в розмові і навіть з радіомовлення та ТБ, які зазвичай транслюють їх тільки 2-3 рази на добу, вже недостатньо для того, щоб бути вчасно обізнаним з більшістю найсвіжіших відомостей.

Більш того, при частому надходженні нових звісток постає питання забування людиною деякої частини великої кількості переглянутого тексту і можливості ознайомитись з ним ще раз.

Для того, щоб вирішити ці проблеми, на основі діджитал-пристрою з використанням всесвітньої павутини виникає потреба створення програмного засобу з функціями зручного відображення регулярно надходячих за користувацьким запитом свіжих, а також за попередні декілька днів статей, швидкого пошуку за ознакою звісток за останній тиждень, повторного перегляду змісту обраних відомостей.

					ІАЛЦ.045490.004 ПЗ	Лист
						6
Зм	Арк.	№ докум.	Підп.	Дата		

Таким чином, задача цього дипломного проекту – розробка мобільного додатку для перегляду новин на базі платформи iOS – є актуальною.

					ІАЛЦ.045490.004 ПЗ	Лист
						7
Зм	Арк.	№ докум.	Підп.	Дата		

1. ОГЛЯД ІСНУЮЧИХ РІШЕНЬ ТА ОБРУНТУВАННЯ ТЕМИ ДИПЛОМНОГО ПРОЄКТУ

У першому розділі розглянуто та проаналізовано декілька аналогічних існуючих програмних засобів для перегляду новин. Вони були вибрані з переліку найпопулярніших (на момент написання цього дипломного проекту) додатків у секції новинних програм для платформи iOS в українському сегменті магазину мобільних додатків App Store. Був обраний додаток "Ukr.net"[1], застосунок "Virno"[2], додаток "НВ"[3], застосунок "ГОРДОН: Новости"[4]. Критерієм відбору цих мобільних застосунків було те, щоб вони відображали свіжі новини України, роблячи на них певний акцент, а також останні відомості з-за кордону.

1.1 Обґрунтування вибору критеріїв оцінювання існуючих рішень

Загальним критерієм оцінювання існуючих рішень є зручний зрозумілий інтерфейс, який би дозволив людині отримувати інформацію про останні події без дискомфорту та додаткового витрачання часу на з'ясування принципів користування додатком.

Виходячи з цього, об'єктивними критеріями для оцінки таких програмних засобів є:

- Перегляд змісту окремої статті
- Прослуховування тексту новин
- Зберігання змісту обраної статті в пам'ять мобільного пристрою
- Наявність в програмі окремих розділів для новин України та для новин світу
- Можливість перегляду статей за обраною із запропонованих загальних тем в рамках новин України або світу (окремо)
- Можливість відображення новин світу в тому числі безпосередньо від закордонних інакомовних ЗМІ

					ІАЛЦ.045490.004 ПЗ	Лист
						8
Зм	Арк.	№ докум.	Підп.	Дата		

- Повнотекстовий пошук статей за введеним словом / словосполученням серед усіх новин у віддаленій базі даних

1.2 Аналіз існуючих рішень

Мобільний додаток “Ukr.net” – застосунок однойменного українського інтернет-сервісу новин, який станом на 3 травня 2021 року посідає сьоме місце за відвідуваністю серед усіх сайтів україномовного сегменту Інтернету [5]. Додаток має загальну стрічку з добіркою новин з усіх загальних тем і розділом новин з-за кордону. Крім того, можна дивитись усі звістки за обраною темою, а також доступний перегляд статей за усіма темами в рамках обраного регіону України.

Цей програмний засіб розрахований суто на текстовий формат подачі інформації, має простий, але в деякому плані не дуже зручний і зовсім не багатий на функціонал інтерфейс.

Переваги:

- Перегляд новин за окремими регіонами України
- Перегляд новин за тематичними розділами
- Окрема секція для новин з-за кордону

Недоліки:

- Відсутня можливість прослуховування тексту новин
- Немає можливості зберігання змісту статті на свій пристрій
- Немає окремого розділу для всіх новин України
- В тематичних секціях новини України та світу не відокремлені
- Відсутній повнотекстовий пошук за введеним словом/словосполученням у віддаленій БД
- Відсутня можливість відображення новин світу безпосередньо від інакомовних закордонних ЗМІ

Таким чином, недоліків в цього додатку набагато більше, ніж переваг. Вище вказаним критеріям цей застосунок майже зовсім не відповідає.

					ІАЛЦ.045490.004 ПЗ	Лист
						9
Зм	Арк.	№ докум.	Підп.	Дата		

Мобільний додаток "НВ" – застосунок українського щотижневого новинного інтернет-ресурсу "НВ" ("Новое Время"). У додатку є можливість перегляду останніх звісток у розділі “Головна”, де відображаються всі новини. Крім того, з новинними статтями можна ознайомитись у запропонованих тематичних розділах. Також є можливість у двох окремих секціях переглянути усі новини України та усі новини світу. Присутній повнотекстовий пошук статей, а також додавання їх в розділ “Обрані” при наявності у читача платної підписки на сервіс.

Усі функції цього додатку працюють виключно при наявності інтернет-з’єднання, а деяка невелика частина функціоналу взагалі доступна тільки після внесення коштів. Деякі можливості, які вказані в критеріях оцінювання, в принципі відсутні. Інтерфейс цього програмного засобу в цілому зрозумілий, простий, з не найпримітивнішим дизайнерським рішенням, але є все ж певні недоліки щодо його зручності.

Переваги:

- Наявність в програмі окремих розділів для новин України та для новин світу
- Перегляд новин за тематичними розділами
- Можливість повнотекстового пошуку новинних статей
- Додавання новин в розділ “Обрані”
- Можливість відображення новин світу безпосередньо від інакомовних закордонних ЗМІ

Недоліки

- В тематичних секціях новини України та світу не відокремлені
- Відсутня можливість прослуховування тексту статті
- Немає функції зберігання змісту статті собі на пристрій

Таким чином, переваг у цього мобільного застосунку більше, ніж недоліків, але все ж деяким важливим критеріям оцінювання воно не відповідає.

Мобільний додаток "ГОРДОН: Новости" – застосунок українського

суспільно-політичного інтернет-видання "ГОРДОН", заснованого журналістом Дмитром Гордоном. В цій програмі можна дивитись свіжі відомості в розділі "Усі новини", а також є можливість переглядати разом останні звістки України та світу за окремими темами. Крім того, можна дізнаватись одразу усі новини світу в розділі "Світ".

Інтерфейс цього застосунку має непоганий дизайн, є простим у користуванні, але не дуже зручним. Функціонал мобільного додатку в цілому малий. Зміст статей розрахований не тільки на текстову подачу інформації, а і на відображення зображень.

Переваги:

- Наявність в програмі окремого розділу для всіх новин світу
- Перегляд новинних статей за тематичними розділами
- Можливість відображення новин світу безпосередньо від інакомовних закордонних ЗМІ

Недоліки:

- Відсутня можливість прослуховування тексту новин
- Немає можливості зберігання змісту новин в пам'ять пристрою
- Відсутній повнотекстовий пошук статей у віддаленій БД
- В тематичних розділах новини України та новини світу не відокремлені
- Відсутній окремий розділ для всіх новин України

Таким чином, недоліків в цього додатку набагато більше, ніж переваг. Більшості критерії оцінювання ця програма або відповідає тільки частково, або не відповідає зовсім.

Мобільний додаток "Virno" – новинний застосунок, який розробили у співпраці агентство USAID та організація "Центр демократичних реформ і медіа". Головна ідея цієї програми – відображення правдивих новин про світ та Україну тільки від якісних українських ЗМІ. В додатку можна подивитись усі останні звістки в розділі "Всі" та в секції "Новини". Також є можливість

переглядати разом новини України та з-за кордону за окремими запропонованими загальними темами. Присутній повнотекстовий пошук новинних статей у віддаленій базі даних, а також зберігання окремих статей в розділ "Обрані".

Жодна з функцій цього мобільного додатку не працює без інтернет-з'єднання. Інтерфейс програми в цілому простий для розуміння користувачем і дизайн гарний, але все ж деякий важливий для зручного новинного додатку функціонал в ньому відсутній. Є певні незручності у тих функціях, що реалізовані, а в деяких місцях інтерфейсу в принципі реалізовано не дуже логічні елементи - два дублюючих один одного за призначенням розділи – "Новини" та "Всі" при тому, що інші тематичні розділи також є новинами і достатньо складно зрозуміти в тому числі логіку іменування розділу "Новини".

Переваги:

- Прагнення до відображення тільки правдивих новин
- Повнотекстовий пошук новин у віддаленій БД
- Зберігання новин до розділу "Обрані"
- Перегляд новин за окремими запропонованими темами
- Можливість відображення новин світу безпосередньо від інакомовних закордонних ЗМІ

Недоліки:

- Немає можливості прослухати текст новинної статті
- Відсутня функція зберігання змісту статті в пам'ять свого пристрою
- В тематичних розділах новини України та світу не відокремлюються
- Відсутній окремі розділи для всіх новин України та всіх новин світу
- Складнозрозуміла логіка і доцільність деяких частин інтерфейсу

Таким чином, недоліків і переваг в цьому мобільному додатку в цілому порівну. Застосунок не відповідає всім вимогам зручного новинного програмного засобу.

Зм	Арк.	№ докум.	Підп.	Дата

ІАЛЦ.045490.004 ПЗ

1.3. Порівняльна таблиця існуючих програмних рішень

На основі проведеного аналізу популярних існуючих аналогів новинних додатків за вище приведеними критеріями оцінювання складена підсумкова порівняльна таблиця з перевагами та недоліками розглянутих рішень (таблиця 1.1)

Таблиця 1.1

Порівняльна таблиця існуючих програмних рішень

Критерії оцінювання	Існуючі програмні рішення			
	Мобільний додаток "Ukr.net"	Мобільний додаток "НВ"	Мобільний додаток "ГОРДОН: Новості"	Мобільний додаток "Virno"
Перегляд змісту окремої статті	+	+	+	+
Прослуховування тексту новин	-	-	-	-
Зберігання змісту обраної статті в пам'ять мобільного пристрою	-	-	-	-
Наявність в програмі окремих розділів для новин України та для новин світу	+/-	+	+/-	-

Можливість перегляду статей за обраною із запропонованих загальних тем в рамках новин України або світу (окремо)	+/-	+/-	+/-	+/-
Можливість відображення новин світу в тому числі безпосередньо від закордонних інакомовних ЗМІ	-	+	+	+
Повнотекстовий пошук статей за введеним словом / словосполученням серед усіх новин у віддаленій базі даних	-	+	-	+

1.4. Висновки

У цьому розділі за вищеприведеними критеріями оцінювання було проаналізовано чотири аналогічних існуючих програмних засобів для перегляду новин України та світу. У всіх цих розглянутих популярних додатків відсутня можливість прослуховувати текст новинної статті. Це означає, що при

користуванні такими програмами, наприклад, за кермом автомобіля або в інших ситуаціях, в яких людина не має можливості переглядати текст статті – отримувати інформацію про останні звістки вона ніяк не зможе.

Крім того, у жодного з цих мобільних застосунків немає функції зберігання змісту обраної новини в пам'ять пристрою. Коли користувач за день дивиться немалу кількість різноманітних новин, то, звісно, певну інформацію він може забути. Є звістки, які б людина хотіла переглянути знову у майбутньому, або користувача, наприклад, зацікавив заголовок певної новини і він хоче ознайомитись з цими відомостями пізніше. Може бути ситуація, коли людина деякий час збирається бути в зоні без стабільного інтернет-покриття і при перебуванні в такій місцевості вона захотіла переглянути чи дочитати певні новини. В останньому випадку проблему за допомогою жодного з цих додатків вирішити було б неможливо. З іншого боку, в мобільних застосунках “НВ” та “Virno” є функціонал для додавання новин в розділ “Обрані” для того, щоб потім довго не шукати їх. Але при цьому зміст статті не зберігається в пам'яті пристрою і через деякий час, як стаття буде видалена з віддаленої бази даних (вони зберігаються там тільки обмежений час), вже неможливо буде отримати доступ до її змісту. Тому так важливо реалізовувати не просто функціонал виділення окремих новин серед інших, а саме зберігання звісток в пам'ять.

Також спільним недоліком для всіх аналогічних розглянутих програмних засобів є неможливість переглянути окремо новини України або світу за певною тематикою. У всіх цих додатках доступний функціонал перегляду новинних статей за обраною тематикою лише загалом.

Єдина обов'язкова для зручного новинного додатку вимога, якої дотримуються всі проаналізовані застосунки, це перегляд змісту окремої статті - без цього неможливо уявити жодний мобільний додаток для перегляду новин.

Ситуація з дотриманням решти критеріїв в проаналізованих програмних засобах різна, але в будь-якому разі ніяке з розглянутих існуючих аналогічних рішень не дотримується повною мірою всіх вищенаведених вимог зручного

мобільного додатку для перегляду новин України та світу.

У 21 сторіччі гарно та вчасно проінформована людина це людина, яка може з більшим успіхом вирішувати щоденні проблеми та оперативно реагувати на зміни обставин. Мобільний застосунок перегляду новин, який би відповідав всім вищевказаним необхідним критеріям, дозволив би людині мати у своєму розпорядженні потужний програмний засіб для того, щоб бути у дуже тісному контакті з останніми відомостями на дійсно комфортному рівні, і при цьому, завдяки функціоналу цього додатку, не розгубитися у великій кількості нової інформації. Це є доказом того, що тема та задача цього дипломного проєкту є актуальними.

					ІАЛЦ.045490.004 ПЗ	Лист
						16
Зм	Арк.	№ докум.	Підп.	Дата		

2. ОГЛЯД ВИБОРУ ЗАСОБІВ РЕАЛІЗАЦІЇ

У другому розділі розглядаються можливі засоби реалізації мобільного додатку для перегляду новин України та світу, обираються серед цих засобів найбільш придатні для розробки даного застосунку, а також обґрунтовується такий вибір. Здійснюється вибір в рамках засобів для розробки клієнтської частини, а також для реалізації серверної частини.

Під клієнтською частиною мається на увазі безпосередньо функціонал мобільного додатку під платформу iOS, і для її розробки розглядаються такі типи засобів, як мова програмування, засіб для реалізації дизайну користувацького інтерфейсу мобільного додатку, для зберігання даних в постійній пам'яті мобільного пристрою.

Під серверною частиною маються на увазі програмні технології, які виконують також важливі для роботи додатку функції, але на боці віддаленої комп'ютерної системи. Для реалізації цієї частини здійснюється вибір серед таких засобів як BaaS та вбудована пошукова система.

Основний акцент в дипломному проєкті робиться безпосередньо на розробку мобільного застосунку, тому вибір технологій для клієнтської частини розглядається розгорнуто, а для серверної - загально. Окремо варто зазначити, що, крім того, обґрунтовується обрання такого засобу, який перетворює інформацію з текстового формату в аудіо-представлення. Деякі технології виконують таку функцію на клієнті, інші – віддалено, тому цей вибір також буде проаналізовано розгорнуто.

2.1. Обґрунтування вибору мови програмування для розробки мобільного застосунку під платформу iOS

Кожна з мов програмування, які використовують для розробки мобільних додатків, відноситься до одного з двох існуючих підходів до мобільної розробки,

а саме кросплатформенний та нативний[6].

Кросплатформенний підхід передбачає, що за допомогою певної мови програмування та відповідного їй фреймворку створюється єдиний програмний проект, який компілюється на виході у застосунок одразу для різних мобільних ОС. До мов мобільної кросплатформенної розробки відноситься Javascript (у зв'язці з фреймворком React Native), Dart (у зв'язці з фреймворком Flutter) та C# (у зв'язці з фреймворком Xamarin).

Нативний підхід передбачає, що для кожної мобільної ОС застосунок розробляється окремо за допомогою мов, які використовують для програмування під конкретну платформу. Для такого типу розробки додатків на базі мобільної платформи iOS використовують мови програмування Objective-C[7] та Swift[8].

В рамках даного дипломного проекту розробляється мобільний застосунок під конкретну платформу, і є можливість зробити це як за допомогою мов нативної, так і за допомогою мов кросплатформенної розробки, скомпілювавши вихідний код під обрану ОС. Тим не менш, при використанні першого підходу застосунок немає жодних технічних обмежень в доступі до сервісів та функцій мобільного пристрою, продуктивність додатку достатньо легко контролюється, а також набагато легше отримати її високі показники у випадку програмування певних елементів інтерфейсу, ніж при використанні таких мов, як C#, Javascript та Dart. Тому надалі розглядаються виключно мови програмування нативних додатків під мобільну операційну систему iOS, а саме Objective-C та Swift.

2.1.1. Objective-C

Objective-C – мова програмування, яка була створена в 1983 році американським комп'ютерним вченим Бредом Коксом. До 2014 року просувалась фірмою Apple в якості єдиної мови для нативної розробки застосувань на базі операційної системи iOS.

Мова програмування Objective-C створена на основі мов C та SmallTalk.

Як і C++, Objective-C суміщає в собі основний синтаксис Сі та принципи ООП, але якщо перша сприймає виклик методу об'єкту як виклик функції, то друга розглядає таку дію в рамках ідеї, успадкованої від SmallTalk - відправлення об'єктам повідомлень. Об'єкти можуть передавати отримані повідомлення іншим об'єктам, що в свою чергу відкриває можливості для делегування (в тому числі в тих випадках, коли ці об'єкти знаходяться не на одному пристрої).

Переваги Objective-C:

- **Стабільність та перевіреність часом.** За 38 років існування ця мова вже встигла зазнати ряд змін, більшість серйозних помилок у її роботі було давно вже полагоджено, і останні 7 років корпорація Apple вже не випускає серйозних великих оновлень для цієї мови, тому програмний код, написаний за допомогою Objective-C 7-10 років тому працює до сих пір повністю злагоджено. В свою чергу, молода мова Swift оновлюється дуже часто, і проект, написаний півроку тому, цілком може почати працювати некоректно, якщо його не переписати під нову версію Swift.
- **Підтримка існуючих вже дуже багато років великих кодових баз iOS додатків.** Остання версія iOS, яка була представлена Apple – 14.4.2. До сих пір успішно функціонують чимало додатків, які були випущені ще за iOS версії 7, або навіть 2 чи 3. Це був час, коли Objective-C була єдиною мовою розробки додатків на базі цієї ОС і використовувалась вона в цій області інженерії повсюдно, а на більш молоду мову Swift немало цих проектів досить не перейшли.
- **Повна сумісність з мовами C та C++.** Зустрічаються у iOS розробці проектування таких модулів, які повинні працювати якомога швидше, і Objective-C не в змозі забезпечити потрібну швидкість роботи. Так як синтаксис цієї мови дуже багато унаслідував від мови C, то Objective-C, на відміну від мови Swift, повністю сумісний з кодом написаним за допомогою C/C++, які прекрасно вирішують проблеми швидкості.

2.1.2. Swift

Swift – мова програмування, яку почала розробляти компанія Apple під проводом інженера Кріса Латнера у 2010 році і вперше представила її на всесвітній конференції розробників WWDC у 2014 році. З моменту першого релізу корпорація Apple позиціонує мову Swift, як головну мову програмування додатків для операційної системи iOS, tvOS та watchOS і рекомендує її використовувати всім розробникам застосунків для створення програм на базі останніх моделей техніки Apple. Метою корпорації було розробити таку мову, синтаксис якої був би простий для вивчення та, водночас, яка б суміщала в собі увесь передовий функціонал, характерний для сучасних мов програмування. Тому мова Swift наслідує свій синтаксис від ряду високорівневих мов програмування серед яких C#, Javascript, Ruby, Haskell, Python та інші, поєднуючи в собі як концепції ООП, так і ідеї функціонального програмування.

Переваги мови Swift:

- **Дуже зрозумілий синтаксис.** До появи мови Swift для того, щоб стати розробником застосунків на базі платформи iOS, треба було витратити певний час на вивчення мови Objective-C. Для людей, які перший раз в житті вчилися програмувати, цей час міг бути досить довгим, бо C-подібний синтаксис є достатньо складним за своєю структурою для тих, хто ніколи не мав справу з написанням програмного коду. Swift вирішує цю проблему, маючи граматику з достатньо прозорою логікою.
- **Сучасний функціонал.** Нові для iOS розробки конструкції та системні функції мови Swift підтримують концепції функціонального програмування, що робить проектування iOS застосунків більш гнучким та пропонує більш оптимальні шляхи створення мобільного програмного забезпечення.
- **Компактність програмного коду.** Якщо в Objective-C разом з реалізацією класу обов'язково потрібно оголошення у окремому

Зм.	Арк.	№ докум.	Підп.	Дата

файлі, то в мові Swift необхідними є тільки оголошення класів. Таким чином, код мовою Swift є за розміром менше, ніж цей же код мовою Objective-C.

- **Висока швидкість.** Корпорація Apple прагнула розробити Swift такою мовою, яка за швидкістю виконання могла б стати впевненим конкурентом дуже швидкої мови C. В свою чергу, Objective-C є достатньо повільним через те, що компілює програму під час її виконання.
- **Легше досягти надійності в роботі програм.** Ніхто не буде користуватись мобільним додатком, робота якого буде часто зупинятись через фатальні помилки. Інженери фірми Apple це врахували під час проектування мови Swift і зробили її такою, щоб зменшити ймовірність написання коду, який би спричиняв критичні помилки в роботі програми. Через це мова Swift є типобезпечною, дуже строго типізованою, і в ній підтримуються декілька видів зручної перевірки констант та змінних на наявність в них значення. Таким чином, набагато легше розроблювати додатки, які б працювали надійно. В тому числі через C-подібний синтаксис розроблювати за допомогою Objective-C надійно працюючі застосунки складніше.
- **Підтримка динамічних бібліотек.** У процесі проектування мобільних додатків на базі мобільної платформи iOS є можливість підключати сторонні бібліотеки та користуватись ними під час написання коду. Їх можна підключати статично або динамічно. Під динамічними бібліотеками маються на увазі такі, які в єдиному екземплярі розташовуються в пам'яті, і на які можуть посилатись одразу декілька застосунків. При оновленні динамічної бібліотеки перекомпільовується тільки безпосередньо ця бібліотека. Статичні бібліотеки під'єднуються до файлу виконання під час компіляції і діють тільки в рамках конкретного виконаного файлу. Таким чином, при оновленні

бібліотеки треба перекомпільовувати весь проект, а файли цих бібліотек множаться серед програм. Мова Swift, на відміну від мови Objective-C, підтримує динамічні бібліотеки, і через це мобільні додатки, написані мовою Swift, займають менше місця в пам'яті.

2.1.3. Висновки та обґрунтування вибору мови програмування

Було розглянуто мови нативного програмування мобільних додатків на базі платформи iOS – Objective-C та Swift. У кожній з них є свої сильні та слабкі сторони. Вибір мови Objective-C для створення сучасного мобільного додатку практично виключає виникнення ситуацій, коли треба переписати вже існуючу кодову базу під новий синтаксис. Але разом з тим такий вибір несе із собою необхідність писати більше коду, ніж могло бути у випадку використанні мови Swift, необхідність проектувати сучасне програмне забезпечення не дуже сучасними шляхами, потребу займати більше місця на пристрої, ніж виконуваний файл, скомпільований з вихідного коду мови Swift. Крім того, більше часу витрачається на те, щоб спроектувати дійсно надійно працюючий додаток.

В свою чергу, ще досить молода мова Swift далі стрімко розвивається і оновлюється, тому може час від часу виникати необхідність переписати деяку частину вже написаного проекту під нову версію мови. В інтегроване середовище розробки Xcode, яке було створене фірмою Apple для розробки iOS застосунків, впроваджені засоби автоматичної заміни певних застарілих конструкцій мови на більш актуальні. Таким чином, зменшується в проекті кількість коду, який потрібно буде змінювати вручну. На даний момент в мову Swift ще не перенесли деяку частину важливих функцій з мови Objective-C. Swift повністю сумісний з Objective-C і нема перешкод для того, щоб в коді мовою Swift використовувати певний функціонал Objective-C. Зазвичай частина Objective-C граматики в коді Swift займає не більше 2-3 процентів, якщо взагалі

присутня.

Таким чином, в контексті створення сучасного мобільного застосунку використання мови Swift має набагато менше недоліків і набагато більше переваг, ніж мова Objective-C. Тому розроблювати зручний мобільний додаток перегляду новин України та світу на базі платформи iOS пропонується саме мовою програмування Swift.

2.2. Вибір засобу розробки дизайну користувацького інтерфейсу нативного iOS додатку

Існує декілька засобів розробки дизайну такого інтерфейсу. Інтегрованих середовищ розробки для створення iOS застосувань також є декілька. Деякі такі засоби, як SwiftUI [11] або мова нативного програмування, доступні в усіх існуючих середовищах для створення iOS додатків. До Storyboard [9] та XIB-файлів [10] є доступ виключно в IDE, створеному корпорацією Apple, а саме Xcode. Так як в рамках дипломного проекту використовується Xcode, тому далі будуть розглядатись усі ці засоби.

2.2.1. Storyboard

Storyboard – це засіб створення дизайну мобільного застосунку, який в своїй суті є конструктором інтерфейсу. В ньому створюються контролери, між ними будуються зв'язки, а також розроблюється представлення кожного з контролерів. Storyboards є наймолодшим серед усіх засобів дизайну iOS додатків, які розглядаються в даному дипломному проекті.

Серед **переваг** Storyboard можна виділити наступні:

- **Швидкість створення дизайну.** В Storyboard представлення створюються за допомогою перетягування мишею елементів інтерфейсу (кнопок, повзунків, позначень тощо) у контролери, а також

Зм	Арк.	№ докум.	Підп.	Дата

налаштування співвідношень цих елементів і відстаней між ними за допомогою готового функціоналу конструктора інтерфейсу. Крім того, параметри окремих елементів виставляються у боковій панелі Xcode. В свою чергу, при побудові дизайну за допомогою програмного коду все це треба прописувати вручну, що займає набагато більше часу.

- **Наглядність взаємовідношення контролерів.** Структура мобільного додатку може бути достатньо складною – з великою кількістю контролерів та немалою кількістю зв'язків між ними. У Storyboard можна наглядно бачити те, як в застосунку все влаштовано з точки зору дизайну. В свою чергу, непросто зорієнтуватись в складній структурі дизайнерської частини, якщо вона вся прописана програмним кодом.

2.2.2 XIB

XIB – старіший за Storyboard засіб розробки дизайну додатку на базі платформи iOS. Представлення контролера розбивається на більш дрібні представлення, за кожним з яких закріплюється файл з розширенням .xib, що є по суті конструктором інтерфейсу, а також файл для програмного коду ініціалізації. Перевага використання цього засобу полягає в тому, що знаходити помилки та тестувати роботу кожного контролера стає легшим через відокремлення один від одного його частин.

Головна перевага XIB-файлів це можливість один раз розробити його дизайн та, при необхідності, прописати певний ініціалізуючий код, і потім використовувати це окреме представлення будь-яку кількість разів у різних контролерах. Якщо в Storyboard треба продублювати на інше місце контролер, який має під собою, наприклад, декілька рівнів глибини інших зв'язаних контролерів, необхідно разом з ним продублювати і ці всі контролери також відповідно.

Крім того, як і у випадку Storyboard, перевагою XIB-файлів є швидкість

створення дизайнерської частини застосунку, яка при використанні конструктора значно вища, ніж коли інтерфейс представлення повністю створюється за допомогою коду.

2.2.3 Мова нативного програмування

Використання мови нативного програмування додатку на базі платформи iOS в якості засобу розробки дизайну інтерфейсу передбачає, що все, що стосується дизайнерської частини застосунку, буде прописано кодом за допомогою цієї мови. Сюди входить створення контролерів, їх зв'язків, представлень з усіма їх елементами, в яких необхідно програмно також виставити співвідношення елементів, їх розташування та потрібні характеристики.

Створення дизайну інтерфейсу виключно за допомогою програмного коду це процес, який може зайняти немалий час, але у такого шляху є свої **переваги**, серед яких:

- **Створення складних динамічних представлень.** Представлення, в яких мають бути задіяна комплексна анімація, кастомні часті зміни у відображенні в залежності від певних умов, не можуть ніяк бути створені за допомогою Storyboard чи XIB-файлів. В такому випадку опис дизайну в коді є єдиним рішенням.
- **Висока швидкість компіляції програм.** Storyboard та XIB-файли використовують конструктор інтерфейсу, який в свої суті є XML-файлом. В системі витрачається певний час на те, щоб розібрати XML та внести його параметри в процес компіляції. Це не відбувається у випадку дизайну, прописаного програмно, тому час компіляції у програмному варіанті менше.
- **Зручність багаторазового використання.** Якщо якесь представлення використовується в різних контролерах, чи в одному контролері

повторно, то можна прописати його дизайн в рамках класу і створювати об'єкти цього класу у різних місцях програми.

- **Просте узгодження командної роботи.** Робота Storyboard та XIB-файлів спирається на використання XML-файлів. Дуже часто мобільний додаток розроблює не одна людина, а ціла команда, тобто в рамках проекту над одним Storyboard/XIB можуть працювати декілька людей одразу. І можуть з'являтися дуже великі труднощі, коли зміни треба об'єднати в нову версію Storyboard або XIB, але деякі з них суперечать один одному і перешкоджають об'єднанню. Такі ділянки коду розробникам треба переглядати та обирати потрібні зміни вручну. Справа в тому, що параметри в XML-файлі не є інтуїтивно зрозумілими, і потрібно витратити досить великий час, щоб розібратись з такими проблемами. У випадку програмного створення дизайну усі конфлікти узгодження можна вирішити, виправивши написаний розробниками код, що є набагато швидшим та легшим, ніж редагування XML.

2.2.4 SwiftUI

SwiftUI – це фреймворк, який був представлений корпорацією Apple у 2019 році на конференції WWDC. Він дає можливість використання декларативного принципу розробки дизайну, тобто програмний код набуває описового характеру щодо того, як повинен виглядати інтерфейс.

Серед переваг SwiftUI:

- **Легкість вивчення та створення дизайну.** Декларативний підхід до розробки є набагато більш наочним і є легшим для розуміння, ніж такий шлях побудови інтерфейсу, коли потрібно вказувати, яким чином його зробити.
- **Відсутність необхідності налаштування Autolayout.** В Storyboard, XIB файлах та при побудові дизайну мовою нативного програмування

треба виставляти параметри Autolayout для того, щоб система могла автоматично розрахувати розміри елементів представлення та їх розташування для кожного смартфона і планшета. При описі дизайну за допомогою SwiftUI не потрібно виставляти такі налаштування, бо система це робить повністю автоматично.

- **Live та Debug Preview.** У цьому фреймворку є зручна функція, яка дозволяє переглядати у реальному часі результат виконання програмного коду без витрачання часу на запуск програми в симуляторі. Крім того, є режим пошуку помилок в коді під час такого перегляду.

Серед **недоліків** SwiftUI можна виділити:

- **Малий діапазон підтримуючих версій iOS.** Цей фреймворк підтримує тільки версії iOS 13 та вище. Але ще досить багато мобільних пристроїв Apple, які працюють на iOS версіях 12, 11, 10. Тому SwiftUI працює виключно на досить нових моделях пристроїв.
- **Нестабільність синтаксису.** SwiftUI існує ще лише три роки, і він часто оновлюється. При чому, якщо молода мова Swift має інструменти для автоматичного часткового виправлення існуючого коду під нові версії, то код на SwiftUI треба оновлювати тільки вручну, що іноді може зайняти дійсно багато часу.
- **Низька швидкість роботи частини функціоналу.** Функції Live Preview та Debug Preview дуже часто «підвисають» та можуть не виходити з фіксованого помилкового стану ще довгий час, іноді доходячи до необхідності перезавантажити середовище розробки.
- **Неможливість створення комплексних інтерфейсів.** Граматика Swift User Interface добре справляється з побудовою простого дизайну, але для написання програм з дуже складним інтерфейсом цей інструмент не підходить.
- **Локалізація помилок – процес досить довгий та важкий.** Системні повідомлення про помилки у коді SwiftUI є дуже загальними, та майже

не дають ніяких навіть приблизних відомостей щодо можливої причини помилки. Крім того, через досить малий вік цього інструменту в цілому ще мало існують розробників, які б його використовували та стикались з різноманітними помилками. Як наслідок, на просторах інтернету чи будь-де ще складно знаходити поради та рішення щодо випадків некоректної роботи коду SwiftUI.

2.2.5 Висновки

Був проведений аналіз існуючих засобів розробки дизайну застосунків на базі платформи iOS. Дизайн інтерфейса додатку, який пропонується створити в рамках даного дипломного проєкту, буде не виправдано складно розроблювати за допомогою SwiftUI через пригальмовування роботи деяких функцій фреймворку та ймовірний довгий пошук джерел помилок. В свою чергу, інтерфейс застосунку не передбачається настільки складним, щоб вдаватися до використання мови нативного програмування для побудови дизайну. Недолік Storyboard та XIB щодо узгодження роботи не є в даному випадку критичним, бо дипломний проєкт створюється виключно однією людиною.

Таким чином, серед усіх цих засобів для створення дизайну застосунку пропонується взяти Storyboard та XIB. Їх використання передбачає створення у Storyboard контролерів, зв'язків між ними та їх представлень, а окремі представлення для багаторазового використання будуть побудовані за допомогою XIB-файлів.

2.3. Обґрунтування вибору засобу для зберігання даних у постійній пам'яті мобільного пристрою

Для того, щоб зберігати зміст новинних статей, треба обрати засіб для роботи з постійною пам'яттю девайсу. При чому критерієм такого вибору є те,

що такий інструмент повинен дозволяти зберігати об'єми даних, які теоретично можуть бути дуже великими. Крім того, треба враховувати, що в даному дипломному проекті передбачається, що такі дані не будуть мати складну структуру з багатою кількістю відношень. Існує певний ряд таких інструментів, і одні з них впроваджені фірмою Apple, а інші, такі як Realm[17], є сторонніми.

2.3.1. UserDefaults

UserDefaults[12] – це сховище даних iOS пристроїв, яке використовується для того, щоб зберігати та діставати звідти інформацію за якимось ключем рядкового типу.

Переваги:

- **Швидкий початок роботи з БД.** Не треба підключати жодні бібліотеки, фреймворки чи окремо якимось налаштовувати. Під час програмування в будь-якому кодовому файлі доступний синглтон системного класу UserDefaults, тобто клас має єдиний екземпляр бази даних, котрий доступний у всьому проекті.

Недоліки:

- **Обмежений перелік типів даних.** UserDefaults працює тільки з даними системних типів – рядковий, цілочисельний, булевий, словник тощо.
- **Кількість збереженої інформації.** Нема лімітів щодо об'єму даних, які можна зберегти в UserDefaults. Але справа в тому, що при кожному запуску додатку всі ці дані повинні завантажитись для того, щоб застосунок почав працювати. Якщо кількість інформації розміром до 512 кілобайт ще не сильно впливає на швидкість завантаження, то дані більшого об'єму вже значно гальмують початок роботи застосунку. Швидкість програми є критичним фактором, який формує думку користувача про додаток та може зменшити в нього бажання далі взаємодіяти з цим програмним засобом, якщо програма працює надто

повільно. Тому для того, щоб застосунок був комфортним у користуванні, в UserDefaults не розташовують сумарно інформації більше вище вказаного розміру.

- **Відсутня можливість здійснювати комплексні запити в БД.** UserDefaults дозволяє отримувати дані тільки в одному екземплярі за конкретним ключем. Тобто робити запити на отримання певної частини даних цієї бази за вказаною комплексною умовою неможливо.

2.3.2. NSCoder та протокол Codable

NSCoder[13] - це абстрактний системний клас, який наслідують класи, котрі повинні реалізовувати функціонал архівації та розархівації об'єктів. Для цього здебільшого використовують наслідуючі класи PropertyListEncoder та PropertyListDecoder, а об'єкти повинні відповідати протоколу Codable[14]. Принцип зберігання даних будується на тому, що в кожного додатка на пристрої є своє ізольоване від інших застосувань місце у файловій системі девайсу. Називається воно Sandbox. В цьому місці можуть бути будь-які файли, необхідні для роботи програми. В sandbox ці два класи використовують документ з розширенням .plist, структура якого складається з полів та ключів для доступу до цих полів. Енкодер та декодер перетворюють об'єкти для зберігання у файлі та використання у коді відповідно.

Можна виділити такі **переваги**:

- **Робота з будь-якими типами даних.** Використання цих засобів робить можливим зберігання даних як системних, так і користувацьких типів.

Серед **недоліків**:

- **Зайве навантаження на процесор.** Для кожного типу об'єктів створюється в файловій системі окремий .plist-файл. Якщо треба змінити, видалити чи додати якийсь один об'єкт в файлі, то потрібно замість цього файлу повністю записати новий з потрібними змінами. В

свою чергу, щоб отримати якийсь конкретний об'єкт з файлу, треба декодувати увесь файл в масив і з нього вже обрати потрібний об'єкт.

- **Неможливість робити запити.** Вся робота з .plist-файлом при використанні NSCoder відбувається тільки за допомогою запису всього файлу або читанні цілого файлу. Інших варіантів взаємодії зі збереженими даними немає.
- **Об'єм збережених даних.** Цей параметр для .plist-сховища, в яке записуємо та з якого зчитуємо об'єкти за допомогою NSCoder, обмежується тільки розміром вільної постійної пам'яті девайсу. Тим не менш, чим більше це сховище, тим більше для взаємодії з ним робиться зайвого навантаження на ресурси процесору. Тому фактично дані засоби використовують для збереження тільки відносно невеликої кількості інформації.

2.3.3. SQLite

SQLite [15] – це компактна система управління базами даних. Особливість даної БД є те, що вона працює не окремим процесом в системі, як це часто буває серед інших баз даних, а пропонує бібліотеку з API для взаємодії з нею. Це робить взаємодію з БД більш швидкою.

Переваги:

- **Компактність.** Усі дані, індекси SQLite зберігає у єдиному файлі. Знаходиться він на девайсі, на якому виконується програма, що здійснює запити до цієї СУБД. Це робить використання сховища прозорішим.
- **Можливість створювати бази з великою кількістю сутностей та зв'язків.**
- **Інтеграція в iOS.** На відміну від Realm, програмні засоби для роботи з SQLite вбудовані в операційну систему, на базі якої працюють

смартфони та планшети фірми Apple. Таким чином, застосунки, які прямо безпосередньо використовують цю СУБД для зберігання даних, не потребують підключення додаткових сторонніх бібліотек і можливого зменшення від цього вільної пам'яті на пристрої.

- **Можливість робити комплексні запити.** SQLite дозволяє звертатися до неї з достатньо складними за структурою питаннями. В свою чергу, Core Data [16] все ж націлена на відносно нескладні запити.
- **Спільна робота з базою даних.** Нерідко командам розробників достатньо непросто спільно працювати зі сховищем Core Data, бо навіть коли декілька людей працюють над різними сутностями – чимала кількість проблем трапляється при спробі об'єднання результатів роботи в єдину нову версію. Це вже не говорячи про командні зміни в одній і тій же сутності. В таких випадках на допомогу приходить використання SQLite, який коректно оброблює міграції бази даних та колективні зміни, які вносять програмісти до цієї СУБД.

2.3.4. Core Data

Core Data – це фреймворк, який був створений фірмою Apple у 2005 році. Його головне призначення – зробити взаємодію зі сховищем даних, в якості якого здебільшого використовують SQLite, більш легкою і зрозумілою. Робота зі сховищем побудована на використанні об'єктів певних класів. Кожен клас відповідає за окрему таблицю (його поля відповідають стовпчикам таблиці). Безпосередньо зі сховищем програміст не працює – всі запити здійснюються до шару “Контекст” за допомогою готового програмного функціоналу Core Data. Після будь-яких змін, які робить програміст в контексті, треба зафіксувати їх в постійному сховищі (також готовою операцією контексту). В свою чергу, контекст сам складає і робить потрібні команди усіх запитів до сховища.

Головна мета, з якою був створений фірмою Apple цей фреймворк, це

більша за SQLite зручність та простота, бо в даному випадку не треба витратити час на вивчення мови SQL. Тим паче, Core Data виключає ймовірність певних помилок під час програмування. Наприклад, синтаксичні помилки у написанні запитів SQL.

Переваги:

- **Підтримка сховищ різних типів.** В рамках можливостей роботи з бінарними та xml-файлами фреймворк може здійснювати CRUD операції відносно файлів цього формату.
- **Можливість проектування БД з комплексною структурою.**
- **Вбудований в iOS.**
- **Наочність.** На відміну від SQLite та Realm, в Core Data можна побачити графічне відображення структури сховища.

2.3.5. Realm

Realm – це база даних для мобільних пристроїв. Представлена у 2014 році Александром Стігсенном та Бьярном Крістіансенном. Написана мовою C++. Ця БД націлена на зберігання даних у файлі у формі об’єктів користувацьких класів.

Переваги:

- **Потужна оптимізація запитів.** Ключовою перевагою Realm є принципи “нульового копіювання” та “ледачого завантаження”. Перший принцип полягає у тому, що ця база даних намагається якомога менше здійснювати будь-які копіювання даних з файлу сховища до оперативної пам’яті девайса, коли треба прочитати інформацію з БД. Таким чином, прибирається велика кількість дій, які характерні для Core Data та SQLite, по форматуванню даних при їх читанні. Акцент робиться на можливість роботи з даними без їх попередньої обробки системою. Отримується потрібна інформація за допомогою вираховування її довжини та віртуальної адреси у файлі, що має формат карти пам’яті.

Другий принцип передбачає відкладання завантаження даних у оперативну пам'ять до моменту використання цих даних у програмі, а також можливість точкового прямого доступу фреймворку до даних без завантаження системою окремих великих частин інформації.

- **Зручність використання.** Для того, щоб використовувати SQLite, треба вчити мову SQL, тому в цьому критерії ця БД програє Realm. Порівнюючи з Core Data, Realm також більш зручний фреймворк, бо в рамках нього програміст взаємодіє просто безпосередньо з базою даних, зберігаючи та отримуючи потрібні об'єкти, а написаний за допомогою нього код набагато менш громіздкий.
- **Можливість здійснювати складні запити.**
- **Проектування достатньо складних за структурою баз**

2.3.6. Висновки

Було розглянуто п'ять можливих програмних засобів зберігання даних у мобільних пристроях на базі платформи iOS. Сховище UserDefaults та зберігання даних за допомогою нащадків NSCoder об'єктів, відповідаючих протоколу Codable, не підходить для розробки цього дипломного проекту, бо вони не підтримують зберігання теоретично можливої великої кількості інформації. SQLite, Core Data та Realm це саме бази для такого об'єму даних. Але між цими трьома засобами потрібно обрати такий, який дає можливість якомога швидше почати працювати з БД нескладної структури. Простота та зручність використання є виключною перевагою Realm. Тим паче, варто також зазначити дуже важливий фактор швидкості операцій. Нижче наведений рисунок 2.3, рисунок 2.4, а також рисунок 2.5, які демонструють порівняння в тому числі цих трьох засобів зберігання даних для читання даних, вставки та підрахунку відповідно.

Читання

Читання усіх рядків БД, що відповідають критеріям запити

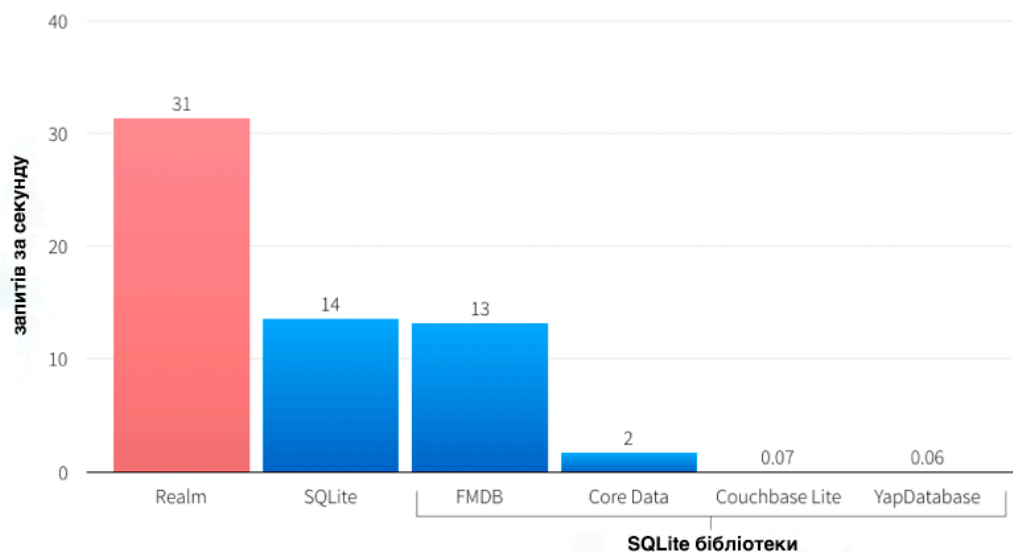


Рисунок 2.1 - Порівняння засобів зберігання даних у кількості запитів читання за одиницю часу

Додавання

Додавання 200 тисяч рядків у БД за одну транзакцію

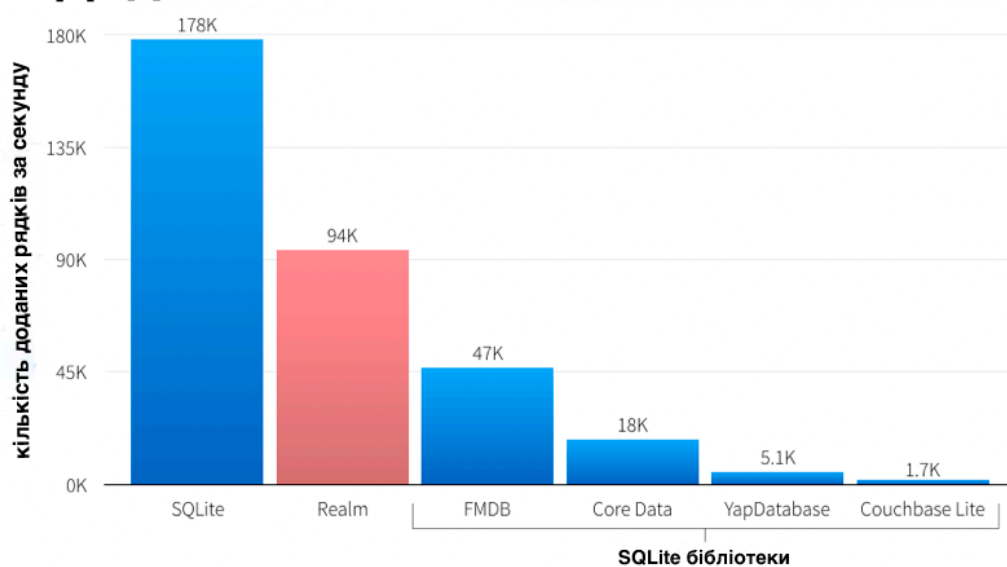


Рисунок 2.2 - Порівняння засобів зберігання даних у кількості доданих рядків за одиницю часу

Підрахунок

Отримання кількості рядків, що відповідають критеріям запиту. Загальна кількість рядків у БД - 200 тисяч.

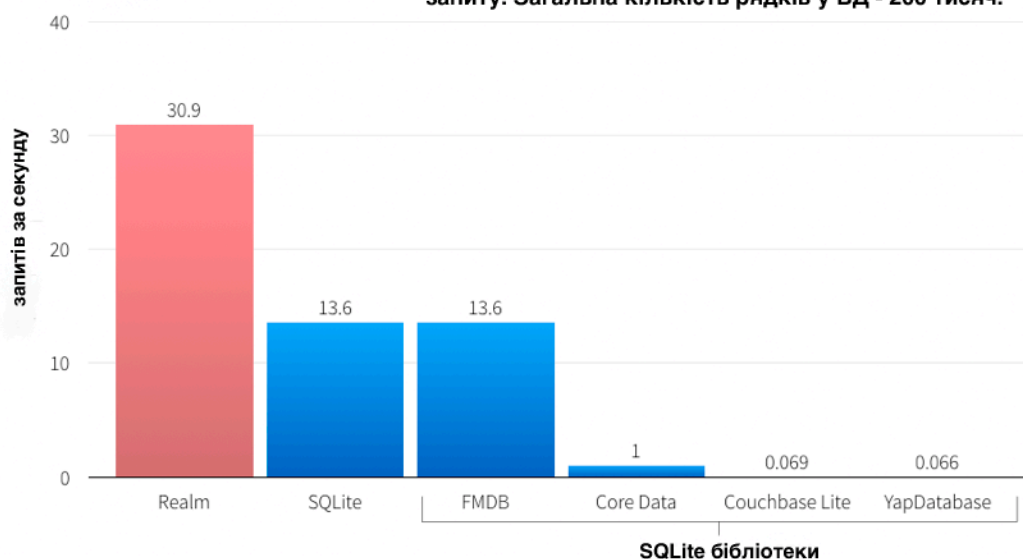


Рисунок 2.3 - Порівняння засобів зберігання даних у кількості запитів підрахунку за одиницю часу

Отже, завдяки серйозній оптимізації операцій Realm лідирує у швидкості виконання більшості з наведених дій. Таким чином, для розробки дипломного проєкту вибір зупиняється на мобільній БД Realm.

2.4 Обґрунтування вибору засобу для перетворення інформації з текстового представлення в аудіопредставлення

В рамках даного дипломного проєкту у мобільному додатку перегляду новин для ще більшої зручності передбачається впровадити можливість прослуховування тексту новин. Для цього треба обрати певний програмний засіб для перетворення інформації з текстового формату в звуковий. Серед таких є нативний для iOS AVSpeechSynthesizer [18], а також сторонні – Amazon Polly [19] та Google Cloud Text-To-Speech [20].

2.4.1 AVSpeechSynthesizer

AVSpeechSynthesizer – клас, який містить методи для аудіо-програвання інформації текстового формату. Є частиною AVFoundation – створеного фірмою Apple фреймворку для обробки звукової інформації.

Переваги:

- **Прослуховування офлайн.** Головною перевагою цього інструменту є те, що він єдиний серед засобів, що аналізуються, може виконувати такий функціонал без наявності будь-якого інтернет-з'єднання – його він зовсім ніколи не потребує.
- **Швидкий початок роботи.** Для того, щоб підключити цей фреймворк, треба усього лиш на початку файлу коду вказати команду `import AVFoundation`, бо клас AVSpeechSynthesizer є нативним інструментом Apple.

Недоліки:

- **Низька якість мовлення.** Вимова текстової інформації, що отримується завдяки AVSpeechSynthesizer, є дуже далекою від людської.
- **Відносно невелика кількість мов.** Цей клас підтримує лише певний ряд мов, серед яких українська відсутня.

2.4.2 Amazon Polly

Amazon Polly – це хмарний сервіс американської фірми Amazon, який є частиною комплексної платформи Amazon Web Services.

Переваги:

- **Вимова, що звучить дуже природньо.**

Серед недоліків є такі:

- **Працює виключно за наявності інтернет-з'єднання**

Зм	Арк.	№ докум.	Підп.	Дата

ІАЛЦ.045490.004 ПЗ

Лист
37

- **Мала кількість мов.** Може бути декілька діалектів одної і тієї ж мови (англійська, наприклад) при тому, що багатьох мов нема взагалі. Українська відсутня.
- **Витрачання часу на підключення до проєкту.** Перед тим, як почати використовувати сервіс Amazon Polly, треба зареєструвати профіль в AWS, під'єднати платіжний засіб і тільки після цього можна почати роботу.

2.4.3 Google Cloud Text-To-Speech

Google Cloud Text-To-Speech – синтезатор мовлення, розроблений компанією Google. Є складовою хмарної платформи Google Cloud Platform.

Переваги:

- **Дуже великий діапазон мов.** Серед цього переліку присутня і державна мова України.
- **Два типу синтеза.** Цей програмний засіб використовує два підходи до перетворення тексту до звуку – це стандартні та wavenet голоси. Стандартна вимова генерується алгоритмами обробки сигналів, і як підсумок отримується в цілому достатньо виразне аудіо-представлення. Wavenet-мовлення отримується завдяки машинному навчанню нейронної мережі, яка вчиться на багатьох різних голосах і вимовах. Таке аудіо виходить особливо виразним, природнім і з правильним розстановками та паузами.

Серед **недоліків** можна виділити:

- **Необхідно інтернет-з'єднання для роботи**
- **Для підключення до застосунку синтезатор потребує спочатку деякого налаштування**

Зм	Арк.	№ докум.	Підп.	Дата

ІАЛЦ.045490.004 ПЗ

2.4.4 Висновки та обґрунтування вибору засобу

Було розглянуто три засоби для отримання аудіо-представлення інформації з текстового формату. Головними критеріями обрання потрібного засобу є якість вимови та робота з достатньо великою кількістю мов, серед яких обов'язково повинна бути українська. Нативна технологія AVSpeechSynthesizer не відповідає жодному з цих критеріїв, хоча і її виключна перевага – виконання в офлайн. Для вибору залишаються дві серверні технології - Amazon Polly та Google Cloud Text-To-Speech. І лише друга відповідає обом критеріям. Тому серед трьох технологій для перетворення текстової інформації в звукову вибір зупиняється на Google Cloud Text-To-Speech, який працює на віддалених серверах.

2.5 Обґрунтування вибору віддаленого сховища та технологій повнотекстового пошуку

Віддалене сховище є основним джерелом новин для користувачів цього мобільного додатку. Туди будуть додаватись нові звістки перед тим, як їх отримує людина до себе на пристрій.

Віддалене сховище може бути спроектоване вручну за допомогою таких програмних засобів як, наприклад, Web API в рамках фреймворку ASP.NET Core, або програмної платформи Node.js. Так як тема цього дипломного проєкта передбачає зосередження саме на побудові безпосередньо мобільного застосунку, то в якості засобу зберігання даних на віддаленій комп'ютерній системі пропонується обрати готове програмне рішення.

Серед таких готових платформ є Firebase [21] від фірми Google, а також CloudKit [22], створена фірмою Apple. Не дивлячись на те, що CloudKit є дійсно сумісним з iOS-девайсами (бо зроблені обидва однією корпорацією) – він є не настільки підходящим для розробки даного мобільного додатку, як BaaS

Firebase. По-перше, CloudKit потребує використання сервісу iCloud, який готові використовувати не всі користувачі iPhone та iPad. По-друге, CloudKit сам по собі блокує можливі подальші кроки розвитку проекту в бік інших мобільних платформ, бо призначений для взаємодії виключно з технікою фірми Apple. В свою чергу, Firebase сумісний з усіма існуючими на даний момент мобільними ОС, а для взаємодії з ним застосункам не потрібно використовувати додаткові сервіси. Тому вибір віддаленого джерела новин зупиняється на Firebase.

У Firebase відсутня функція повнотекстового пошуку, необхідна для того, щоб шукати статті за словами та словосполученнями, що містяться в тексті новини. Тому треба обрати сторонній засіб, який би цей функціонал реалізовував. Два найпопулярніших програмних рішення це Algolia [23] та Elasticsearch [24]. Є певні причини, чому краще обрати перше в якості пошукової системи. Algolia набагато легша у налаштуванні, ніж Elasticsearch, взаємодія якого з користувачем та Firebase вимагає від розробників певних навичків системного адміністрування. Крім того, інтеграція Firebase з Algolia краща, ніж з Elasticsearch. Тому для пошуку статей по введеним слову чи словосполученню у сховищі пропонується обрати Algolia.

2.6 Висновки

В цьому розділі було розглянуто програмні засоби реалізації мобільного додатку перегляду новин на базі платформи iOS. Щодо розробки функцій клієнтської частини було обрано Swift в якості мови програмування, Storyboard та XIB-файли в якості інструментів для побудови інтерфейсу, мобільну базу даних Realm як засіб для збереження даних в постійній пам'яті мобільного пристрою. Для реалізації серверної частини були обрані програмне рішення Firebase як віддалене сховище, Algolia в якості системи для пошуку, а Google Cloud Text-To-Speech як хмарний інструмент для отримання інформації формату аудіо з формату тексту.

3. ОПИС РОЗРОБЛЕНОГО ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

3.1. Загальний опис структури та роботи додатку

Увесь застосунок працює за шаблоном MVC [25], тобто Model-View-Controller. Модель - це дані, що зберігаються, та логіка роботи з ними (реалізація операцій CRUD). Представлення (View) це користувацький інтерфейс. Контролер містить функції, які відносяться до бізнес-логіки. Користувач взаємодіє з контролером для зміни, додавання чи видалення інформації моделі. Відповідно до цього, контролер також змінює представлення. В Storyboard додають контролери, зв'язки між ними і створюють в рамках кожного з контролерів представлення. Окремі View створюють у форматі XIB-файлів. Кожен контролер асоціюють зі своїм реалізованим програмістом класом, який містить увесь функціонал цього контролеру. В якості моделі використовується структура для отримання статті з серверної частини, яка також є структурою для зберігання даних у локальну мобільну базу даних. Для того, щоб користувач міг взаємодіяти з контролером, у View певні елементи зв'язують з функціями в контролері і, таким чином, коли користувач робить відносно них який-небудь жест, то спрацьовує певний функціонал. Або до такого елементу може бути прямо прив'язаний перехід до іншого модуля, і при натисненні на нього на екрані пристрою з'явиться представлення цього контролеру. А для того, щоб контролер міг змінювати представлення, створюють у ньому зв'язуючу зміну (Outlet), через яку можна в коді маніпулювати станом View.

Надалі під модулем буде матись на увазі контролер + власне представлення, яким він керує. Додаток перегляду новин на базі мобільної платформи iOS має чотири основні модуля: для перегляду новин України, для перегляду звісток світу (мається на увазі закордону), модуль пошуку новин за введеним словом чи словосполученням, а також модуль збережених новин. Вони

					ІАЛЦ.045490.004 ПЗ	Лист
						41
Зм	Арк.	№ докум.	Підп.	Дата		

є складовими контролера-контейнера, який називається Tab Bar Controller (рисунок 3.1).

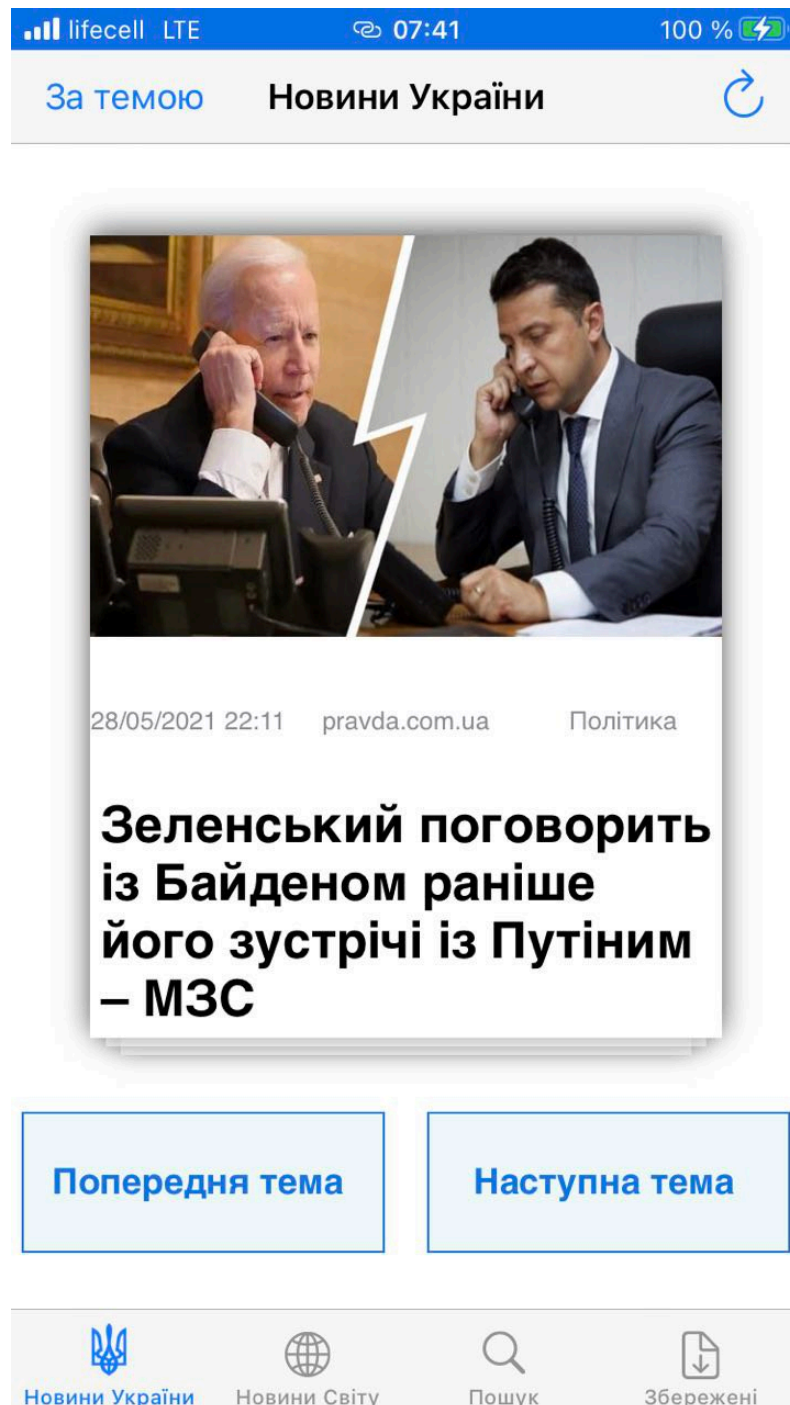


Рисунок 3.1 - Tab Bar Controller, що містить усі головні модулі додатку

Контролер-контейнер не має власного як такого представлення. Він може мати тільки деякі невеликі характерні для нього функціональні елементи

інтерфейсу і слугує для переключення між основними модулями програми та відображення представлення обраного з них. Tab Bar Controller складається з панелі в нижній частині екрану, яка містить ярлики розділів програми, а також безпосередньо місця для показу View. Треба зазначити, що цей контролер містить зв'язки не напряду зі складовими модулями, а з навігаційними контролерами, в які вбудовані ці модулі. Навігаційні контролери потрібні для того, щоб була можливість переходити вглибину до інших модулів, до яких належать перегляд окремої статті, ознайомлення з новинами певної теми, а також фільтр закордонних відомостей.

Для отримання даних в розділі пошуку здійснюються запити до системи Algolia, а в усіх інших контролерах, окрім збережених новин – до віддаленої БД в рамках хмарної платформи Firebase.

Далі більш детально буде розглянуто роботу кожного з модулів, а також структуру віддаленого сховища та пошукової системи.

3.9 Опис структури віддаленої БД та пошукової системи

Віддалене сховище існує в рамках хмарної платформи Firebase. Називається воно Cloud Firestore. Усі дані новин зберігаються у цій БД у форматі документів у колекції “articles”. Кожен документ має поле тексту, країни походження ЗМІ, дати публікації, назви джерела, заголовку, інтернет-посилання на статтю, посилання на зображення, а також теми та мови. Документи не зберігаються у строго визначеному порядку. Для того, щоб працювати з ними відсортованими за певною ознакою, створюється індекс, в якому автоматично документи відмічаються в потрібному порядку, а також в запиті до Firestore в програмному коді застосунку викликається функція order(by:) для роботи з відсортованою інформацією. Автоматизованість Firebase дозволяє не створювати індекс вручну, і під час запуску додатка хмарна платформа покаже помилку про те, що треба створити індекс, та запропонує автоматично створити

його. Вигляд консолі цього сервісу можна побачити на рисунку 3.2.

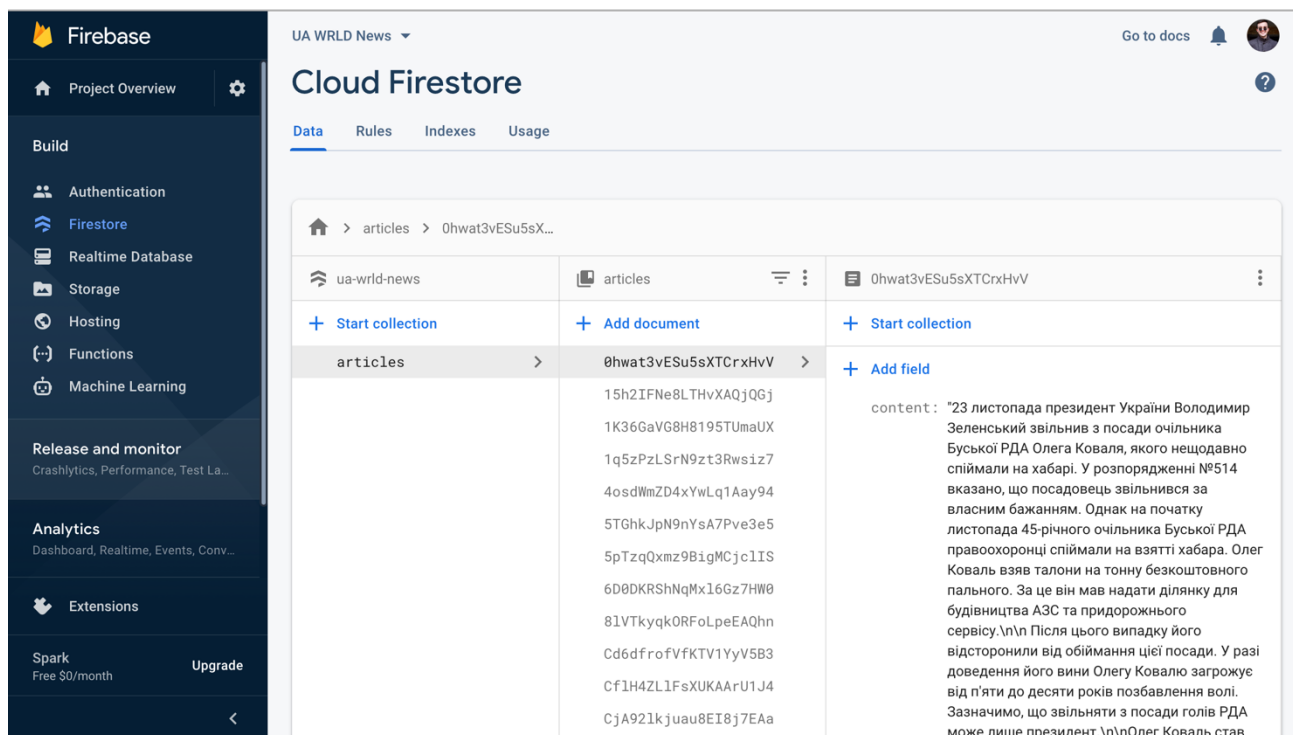


Рисунок 3.2 - Вигляд консольного вікна роботи з Firebase та Firestore.

Система повнотекстового пошуку Algolia дозволяє шукати потрібні документи в базі даних за певним словом чи словосполученням за допомогою індексування слів в документах Firestore. Але для цього необхідно завантажувати ці документи у індекс, що можна робити вручну через консоль в браузері. Щоб автоматизувати цей процес, використовується функціонал Firebase, який зветься Cloud Functions. Це написані програмістом функції мовою Javascript, які спрацьовують тоді, коли трапляються якісь події у Firestore (додавання нових звісток, їх видалення або оновлення). Коли щось з цього здійснюється, то автоматично викликається відповідна функція і індекс Algolia перебудовується системою. Вигляд консольного вікна цієї пошукової системи можна побачити на рисунку 3.3.

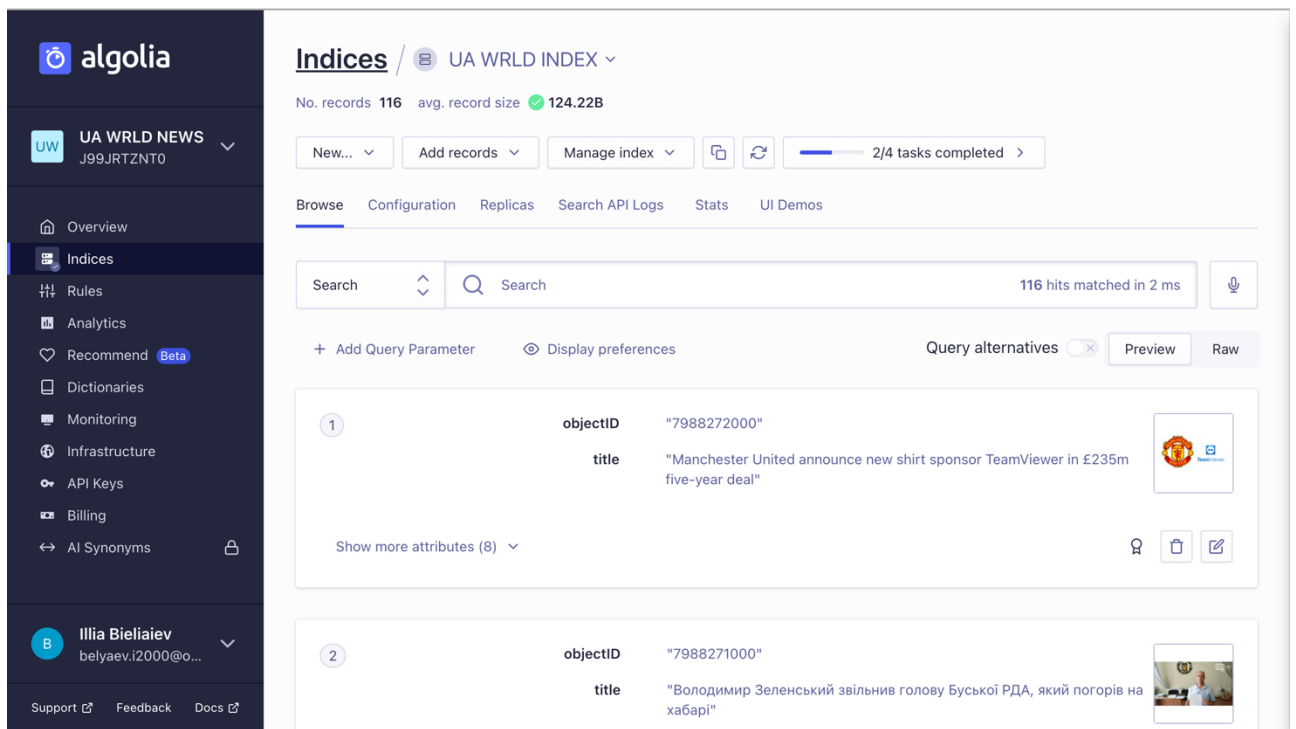


Рисунок 3.3 - Вигляд консолі Algolia

3.2. Опис модуля новин України

Модуль новин України відповідальний за показ новин українських ЗМІ про Україну. Застосований нестандартний підхід по відношенню до дизайну: кожна новина це окрема картка з зображенням, заголовком новини, а також датою, часом публікування, джерелом та темою, до якої відносяться стаття. Новини можна гортати свайпом вправо – до наступної статті, а також вліво до попередньої. Вигляд цього розділу можна подивитись на рисунку 3.4.



12/05/2021 17:00 pravda.com.ua Суспільство

У Києві подорожчає проїзд, але не до 20-25 гривень – Кличко

Попередня тема

Наступна тема

Рисунок 3.4 - Представлення модуля перегляду новин України

Коли новинні статті певної теми пройшли (для кожної виділяється по 4 штуки), з'являється карта з назвою теми наступного блоку (рисунок 3.5).

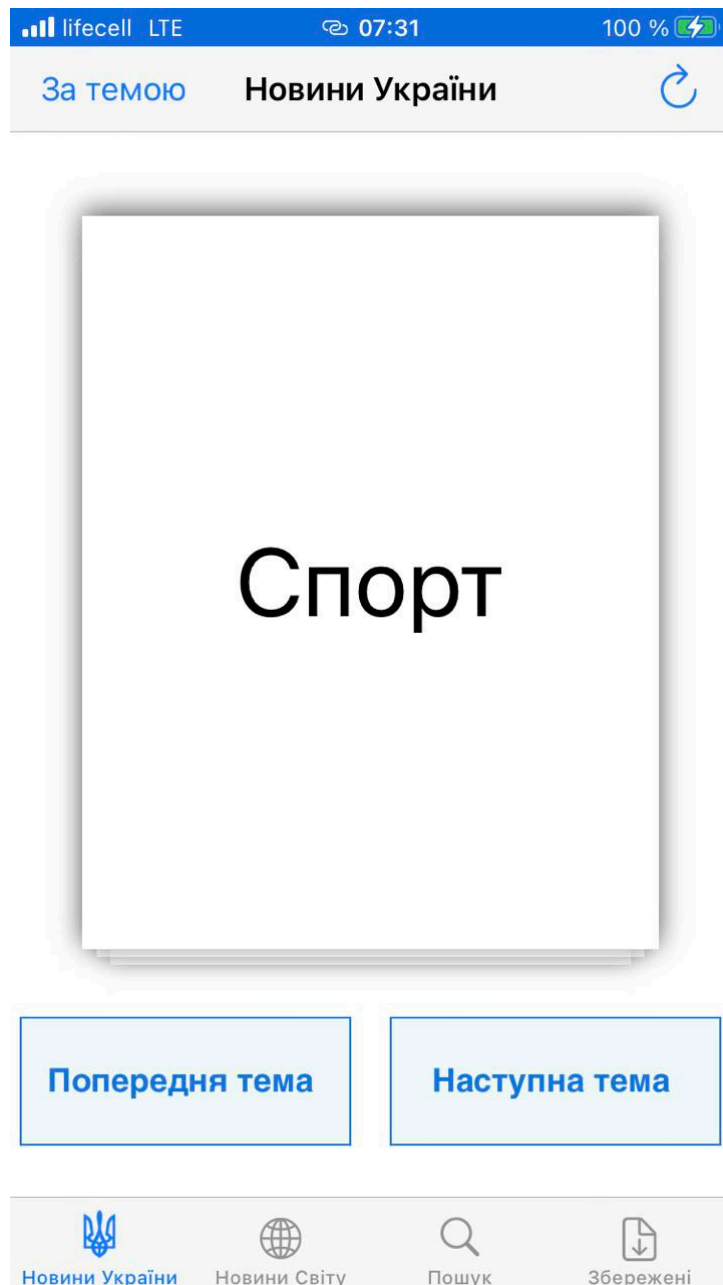


Рисунок 3.5 - Показ картки початку наступного блоку

Таким чином, цей контролер показує по декілька новин з кожної теми, і коли усі тематичні блоки були прогорнуті, то автоматично здійснюється перехід на початок (перший розділ). А якщо свайпати до попередньої статті, то після того, як буде показаний початок першої секції – буде зроблений перехід до останнього розділу. Якщо натиснути на яку-небудь новину, то відбудеться перехід до модуля ознайомлення зі змістом статті. За допомогою двох кнопок

під View карт можна перейти одразу до попередньої або наступної теми. Натиснувши на кнопку “За темою”, відкривається знизу панель, яка дозволяє обрати тему звісток (рисунок 3.6).

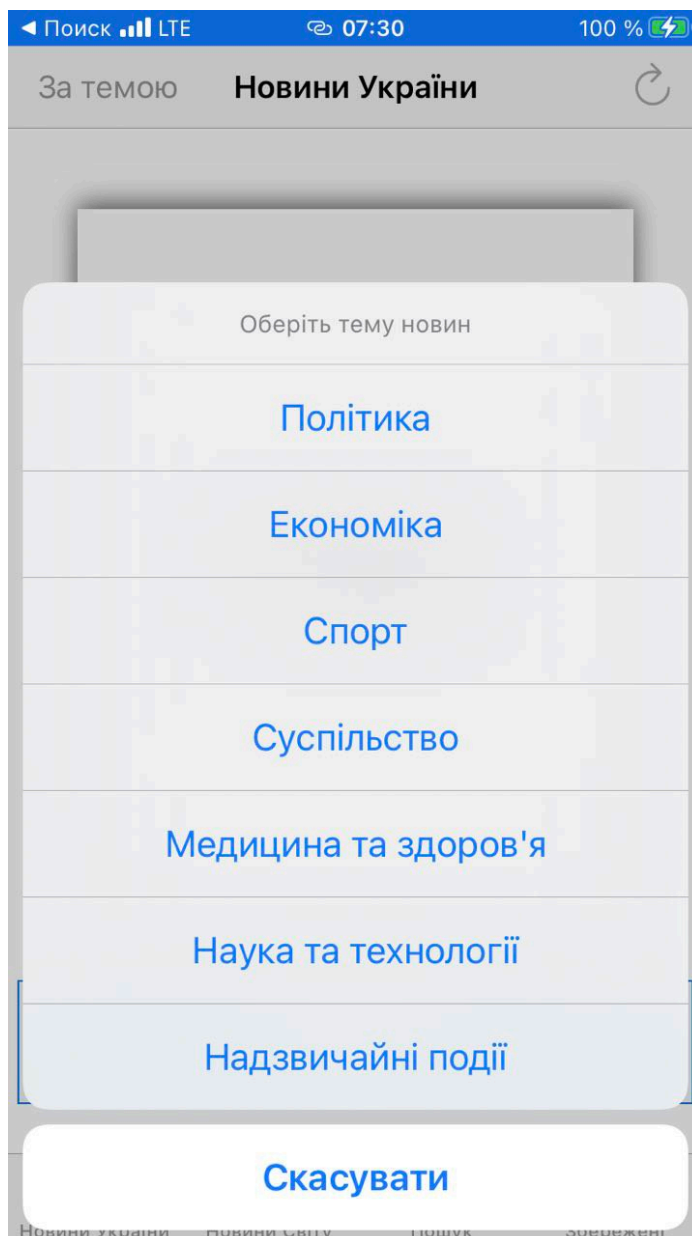


Рисунок 3.6 - Відображена панель для вибору теми

Після того, як була натиснута одна з цих тем, робиться перехід до контролеру перегляду тематичних новин. Якщо натиснути кнопку “Оновити”, то підвантажуються з серверної частини нові статті в кожний блок. Завантаження робиться, знову ж таки, по 4 новини в секцію, якщо нових відомостей за певною

темою більше трьох, або інакше додаються нові статті до тих, що є на пристрої так, щоб відображалось в секції чотири новини.

Коли новини отримуються з віддаленого сховища, то вони зберігаються у структурі даних “впорядкований словник”. Коли на екрані показується кожна карта, то вона потребує для себе окремого представлення, дизайн якого можливо зробити окремо за допомогою XIB-файлу, а дані для View беруться з масиву. Кожна завантажена звістка містить у собі посилання на її картинку. Для того, щоб під час свайпу кожен раз при повторному відображенні певної карти не завантажувати її картинку і, таким чином, не витратити зайвий трафік інтернету, вона зберігається у кеші девайсу - NSCache. Кожен раз, коли застосунок готується до завершення роботи, у відповідній функції-заглушці прописано виконати операції очищення з кешу всіх цих картинок. Крім того, при оновленні коли з секцій в цьому контролері видаляються звістки, то і картинки їх в цей момент також очищуються в кеші.

3.3. Опис модуля новин світу (закордону)

В цьому модулі відображаються новини українських ЗМІ про події закордоном, а також новини закордонних засобів масової інформації про події за кордоном України. Інтерфейс схожий з модулем новин України: такі ж карти для кожної новини, свайп вправо/вліво для переходу до наступної/попередньої статті, дві кнопки під картами для переходу до попередньої або наступної теми, а також кнопка для оновлення і переходу до тематичних новин. Різниця в тому, що тепер є кнопка “Фільтр” для фільтрації отримання звісток. Фільтрувати новини можливо за країною походження ЗМІ, а також за мовою. При здійснюванні запитів до Firebase тепер враховується значення змінних фільтра у класі контролеру. Відображення представлення цього модуля показано на рисунку 3.7.



Рисунок 3.7 - Представлення модуля для відображення новин світу.

3.4. Опис модуля повнотекстового пошуку

Представлення контролера повнотекстового пошуку складається з кнопки пошуку, пошукового рядка, в якому вводиться слово чи словосполучення, яке повинно міститись у знайдених новинах в базі, а також решти місця модуля, в якому показується знову ж у вигляді карт знайдені відомості (рисунок 3.8).

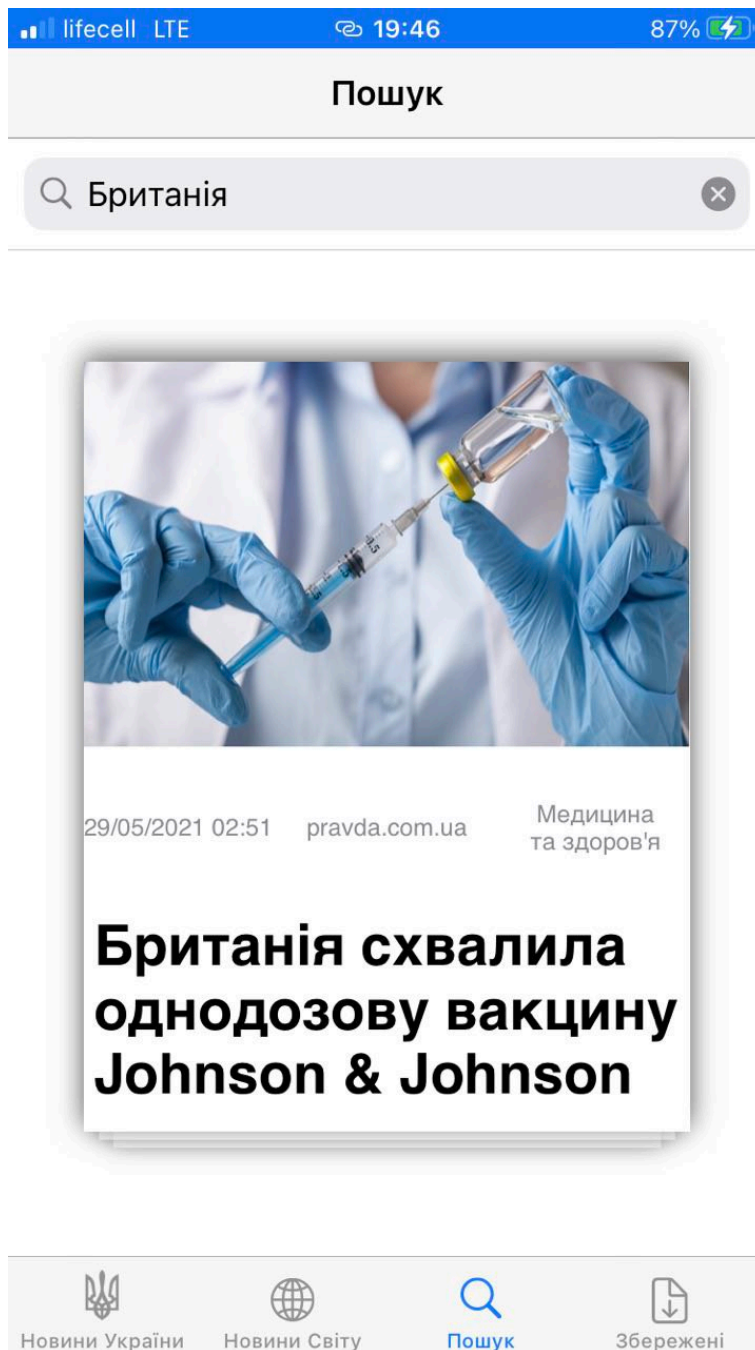


Рисунок 3.8 - Представлення модуля повнотекстового пошуку новин

Після введення значення в рядок, а також натиснення кнопки “Знайти”, робиться запит до індексу Algolia, який повертає колекцію знайдених новин. Під час введення у стрічці слів збоку відображається кнопка скасування. Для того, щоб не використовувати зайвий трафік інтернету, як і під час роботи контролера тематичних новин завантажені зображення тимчасово зберігаються у структурі даних Dictionary.

3.5. Опис модуля збережених новин

Модуль для роботи зі збереженими новинами складається з карт перегляду, між якими робиться перехід вперед та назад за допомогою свайпу вправо та вліво. На рисунку 3.9 можна побачити, як виглядає даний модуль.



Рисунок 3.9 – Представлення модуля збережених новин

Для того, щоб видалити новину, карта якої показується, потрібно натиснути кнопку знизу. Якщо натиснути на карту, то здійсниться перехід до

					ІАЛЦ.045490.004 ПЗ	Лист
						52
Зм	Арк.	№ докум.	Підп.	Дата		

модуля перегляду змісту новини і буде показуватись відповідне представлення. Збережена інформація видалається за допомогою запиту до мобільної БД Realm. Отримання її здійснюється також за допомогою запиту, але та змінна, до якої надходять ці дані, стає слухачем бази даних і автоматично оновлюється при змінах в Realm. Функціонал Realm дозволяє викликати певне замикання при такому оновленні. Якщо користувач при перегляді цієї новини в іншому модулі вирішив її видалити зі збережених, а в цей же час зупинився на ній, коли дивився збережені, то даний контролер в такому замиканні відобразить карту попередньої чи наступної новини.

3.6. Опис модуля тематичних новин

Модуль перегляду новин за обраною темою також має дизайн карток. В лівому верхньому куту View знаходиться кнопка “Назад”, яка дозволяє вийти назад до модуля, з якого здійснився вхід – це або новини України, або новини світу. В правому верхньому куту знаходиться кнопка оновлення. Коли звістки гортаються вперед, то як тільки досягається остання завантажена – підвантажується ще або 10 штук, або менша кількість, якщо це вже решта, яка залишилась в Firestore. Вигляд цього модуля наведено на рисунку 3.10.

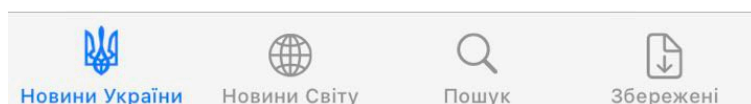
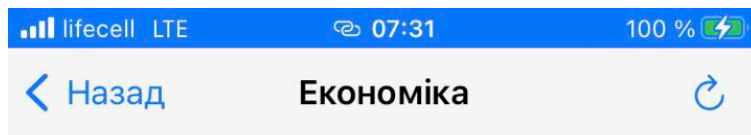


Рисунок 3.10 - Представлення модуля тематичних звісток

Зображення новин також завантажуються окремо у структуру даних “Словник” для того, щоб не витрачати зайвий інтернет-трафік при повторному перегляді. Сумарний набір новин, отриманих з віддаленого сховища при такому перегляді, може бути достатньо великим в плані кількості, і кожна така стаття містить зображення. Клас словника використовує оперативну пам’ять девайсу, і якщо всі ці зображення брати в нього та просто зберігати, то через деякий час

iOS почне робити попередження, що стосуються використання ресурсів і, в кінці кінців, додаток може зупинити свою роботу. Існує також системний клас кешу – NSCache, який також використовує RAM та сам починає зачищатись при критичній кількості зайнятої пам'яті. Крім того, в ньому може бути вручну виставлений параметр максимального розміру зайнятого простору кешу, але не є ясным та зрозумілим механізм вибору об'єктів видалення та момент часу цієї операції – це здійснюється по моделі чорного ящика. Тобто навіть коли задається конкретний ліміт простору кешу, то не факт, що одразу по його досягненні почне очищуватись пам'ять – це може статись і пізніше, а може і взагалі не трапитись. Більш ясно те, що NSCache очищується, коли іншим активним застосункам починає не вистачати оперативної пам'яті, а також, коли зайнята кешем частина RAM близька до усієї вільної на даний момент.

В рамках цього проєкту для зберігання даних більш підходящим є структура даних “Словник” через можливість визначати точно час та об'єкт видалення – так можна зекономити якомога більше трафіку мережі Інтернет. Але тоді потрібно зберігати не більше якоїсь конкретної кількості звісток і відслідковувати це. Ліміт перевіряється вручну в програмному коді. Коли ліміт досягнуто, то виконуються команди очищення перших десяти картинок, або, якщо гортається з кінця назад і треба знов підвантажити крайові перші, то видаляються останні десять. Постає питання вибору цього самого ліміту. Необхідно враховувати, що у смартфонів та планшетів техніки Apple як мінімум 1 гігабайт оперативної пам'яті. Все менше девайсів з такою характеристикою і все більше пристроїв з двома та трьома гігабайтами RAM. Також на прикладі різних сайтів з новинами можна переконатися, що перша картинка в статті займає в середньому десь приблизно 100 кілобайтів. Якщо взяти, наприклад, 700 зображень такого розміру - це вийде приблизно 68,36 мегабайта. Для сучасних (і не тільки) мобільних пристроїв фірми Apple використання додатком такого розміру RAM не є особливо критичним, тому в якості ліміту обирається 700 картинок.

Зм	Арк.	№ докум.	Підп.	Дата

ІАЛЦ.045490.004 ПЗ

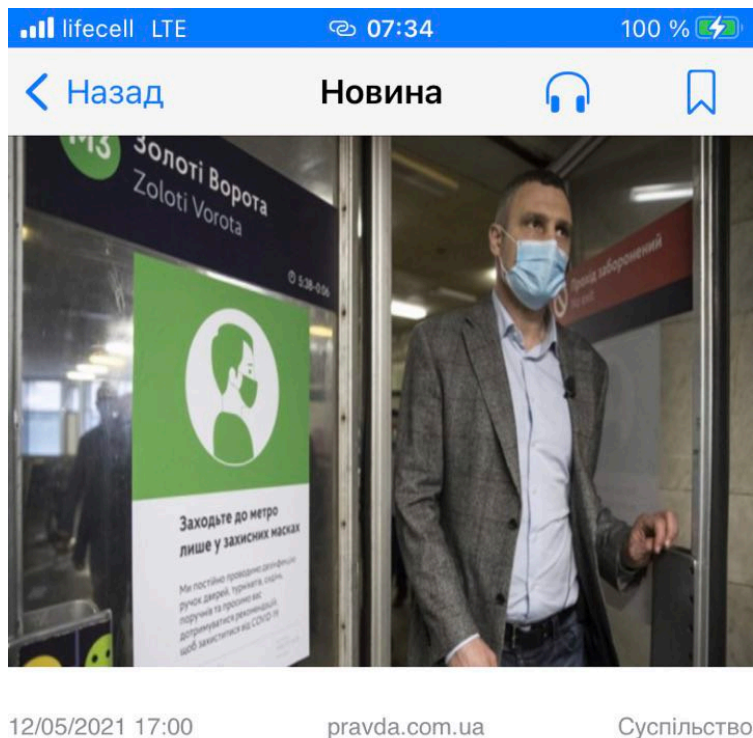
Лист
55

Звичайно, можна також зберігати тимчасові дані у базі даних, але це може призвести до уповільнення роботи застосунку, бо доступ до диску займає більше часу, ніж до RAM. Швидкість роботи мобільного додатку є критичним показником, бо програму, яка працює нешвидко, використовувати ніхто не буде.

Два контролери використовують цей модуль. В одному з них є можливість налаштовувати фільтр. Ці налаштування, при наявності, потрібно застосовувати і в модулі тематичних новин. В класі цього контролера є опціональна змінна для фільтру, від наявності значення в якій залежить формування запиту до Firebase.

3.7 Опис модуля перегляда новинної статті

Даний модуль використовується для ознайомленні зі змістом новини. Спочатку згори розташовується зображення, далі йде заголовок новини, під яким вказується дата, публікації та джерело. Решту контролера знизу займає текст. Згори в правому куті знаходяться кнопки додавання новини до збережених, а також кнопки прослуховування тексту. В лівому верхньому куті розташовується кнопка “Назад” для виходу до попереднього модуля. Збереження або видалення статті з бази даних здійснюється за допомогою натиснення на кнопку з зображенням закладки – таким чином робиться запит до Realm. Для прослуховування статті використовується ярлик із зображенням навушників. Спочатку передається текст до хмарного сервісу Google Cloud Text-To-Speech, в результаті чого отримуються дані в форматі аудіо, а потім за допомогою вбудованого функціоналу iOS програваться ці дані. Якщо під час програвання натиснути кнопку прослуховування ще раз, то воно зупиниться. Якщо натиснути ще раз, то почнеться програвання заново. Представлення цього контролера показано на рисунку 3.11.



У Києві подорожчає проїзд, але не до 20-25 гривень – Кличко

Міський голова Києва Віталій Кличко заявив, що влада міста не планує підвищувати вартість проїзду у громадському транспорті Києва до 20-25 гривень.

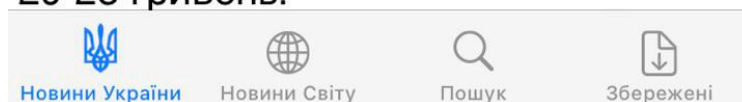


Рисунок 3.11 - Представлення модуля перегляду змісту новини

3.8. Опис модуля фільтрації закордонних новин

Представлення в цьому модулі є табличним (Table View), його вигляд наведено на рисунку 3.12.

Зм	Арк.	№ докум.	Підп.	Дата

ІАЛЦ.045490.004 ПЗ

Лист
57

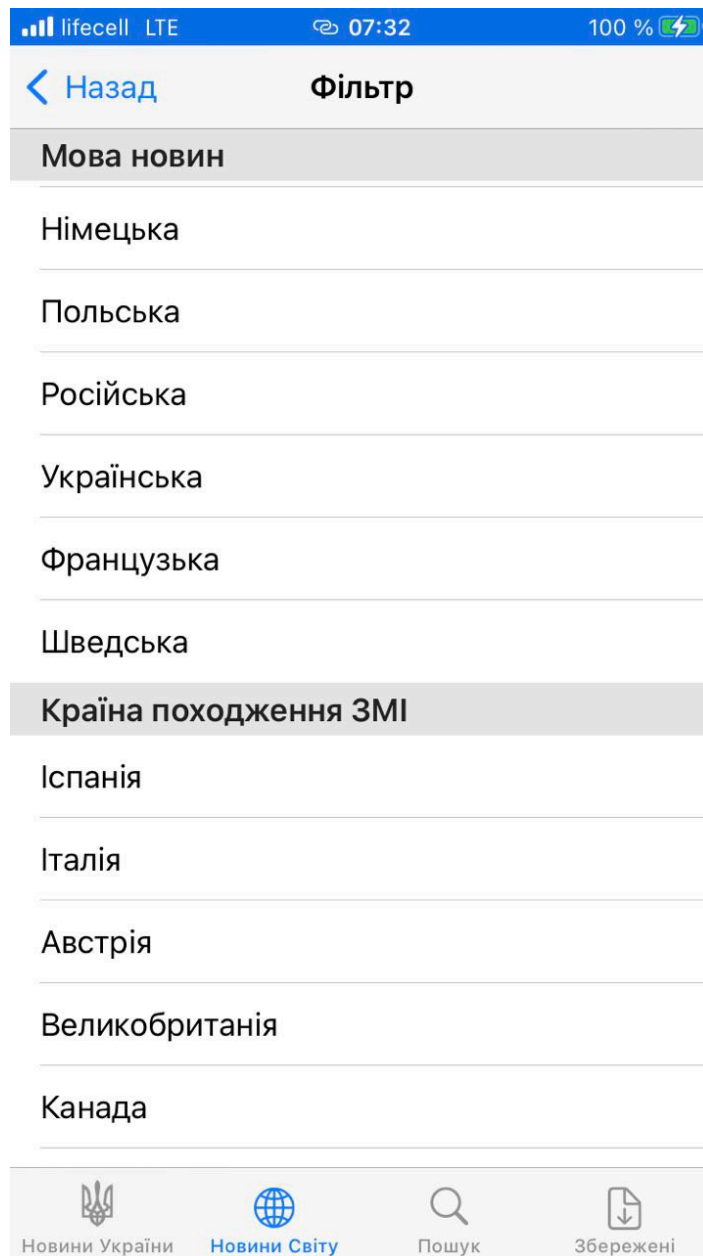


Рисунок 3.12 - Представлення модуля фільтрації новин

Табличне представлення в контексті створення iOS додатків передбачає не таблицю в звичайному розумінні, а формат списку, і є одним з найбільш важливих і основоположних елементів інтерфейсу мобільних застосувань на базі цієї платформи. Даний список розділяється на дві секції – країна походження ЗМІ та мова. В кожній секції є декілька пунктів, кожний з яких можна відмітити, натиснувши на нього, і буде показуватись напроти нього галочка. Після того, як зміни у фільтруванні зроблені, треба натиснути кнопку “Done”, яка розташована у верхньому правому куті, для того, щоб ці зміни зафіксувати і вийти до

попереднього контролеру. Якщо зробити зміни і просто натиснути кнопку “Назад”, то вони не будуть застосовані.

Для того, щоб налаштувати побудову Table View, в функціях зі спеціальними прототипами прописується програмний код. В даному випадку програмуються функції, які вказують представлення для комірок та секцій і заповнюють їх даними з моделі в класі, вказують кількість секцій, кількість пунктів у кожній секції, а також потрібні дії при натисненні пункту цього списку.

3.10 Рекомендації щодо подальшого вдосконалення

Для того, щоб користувачів мобільного додатку ставало дедалі більше, треба постійно поліпшувати його роботу, впроваджувати новий корисний функціонал та періодично випускати оновлення. В рамках даного застосунку серед можливих подальших вдосконалень можна виділити такі:

- Фільтрація в модулі новин України.
- Автоматичне оновлення новин
- Отримання фонових повідомлень про нові звістки від обраних джерел
- У контролері перегляда змісту новини підтримка відображення додаткових зображень, а також відео та постів з соціальних мереж.
- Збільшення функціоналу прослуховування новин, а саме виділення слів, які вимовляються, а також після зупинки програвання можливість продовжити прослуховувати з того ж самого місця. Крім того, покращенням цього пункту може бути і вказання користувачем в тексті слова, з якого потрібно починати програвання.
- Впровадження віджетів для відображення певних новин на робочому столі девайса.
- Підтримка вбудованого голосового асистента Siri для пошуку новин за введеними голосом словами та словосполученнями (в тому числі при ще не запущеному додатку).

- При повторному запуску відновлення за допомогою кешу того стану додатку, в якому він раптово був завершений системою під час знаходження у фоні.

					ІАЛЦ.045490.004 ПЗ	Лист
						60
Зм	Арк.	№ докум.	Підп.	Дата		

ВИСНОВКИ

В рамках даного дипломного проєкту був розроблений мобільний додаток на базі платформи iOS для перегляду новин. Враховуючи переваги та недоліки проаналізованих популярних аналогічних рішень, було створено програмне забезпечення, яке задовольняє усім критеріям зручності та надає користувачу потужний функціонал для оперативного отримання новинних статей України та закордону, що регулярно з'являються кожного дня. Для того, щоб зробити оптимальний з технічної точки зору застосунок, в рамках кожного необхідного для розробки типу інструменту було проаналізовано по декілька програмних засобів і обрані найбільш підходящі.

Таким чином, за допомогою високорівневої сучасної мови програмування Swift, засобів створення дизайну інтерфейсу Storyboard та XIB-файлів, потужно оптимізованої мобільної бази даних Realm, хмарного синтезатора природньої людської мови з тексту Google Cloud Text-To-Speech, а також серверних технологій віддаленого сховища Firestore та пошукової системи Algolia був створений мобільний додаток для смартфонів iPhone, який дає можливість загального перегляду окремо новин України та останніх відомостей закордону, усіх новин України або світу за обраною темою, перегляду новин безпосередньо від іншомовних джерел іноземного походження, повнотекстового пошуку у віддаленому сховищі новин за останній тиждень, а також зберігання змісту статей на власний пристрій для офлайн-перегляду та прослуховування тексту статей.

Надійно працюючі програмні продукти повинні враховувати в тому числі і критичні випадки роботи. Ці випадки не трапляються дуже часто, але без відповідного їх врахування можуть з'являтися серйозні проблеми під час роботи застосунків. Серед таких ситуацій в рамках даного проєкту могло би бути помилкове раптове закінчення роботи програми при критично великому об'ємі зайнятої додатком оперативної пам'яті. За допомогою програмного контролю зберігання даних програми у пам'яті з довільним доступом, ймовірність такого

					ІАЛЦ.045490.004 ПЗ	Лист
Зм	Арк.	№ докум.	Підп.	Дата		61

результату роботи даного продукту було зведено до нуля.

Ознаками високої якості та подальших перспектив програмного забезпечення є періодичні серйозні оновлення та вдосконалення його функціоналу. Серед покращень даного проєкту у майбутньому може мати місце автоматичне оновлення новин, підтримка голосового асистенту Siri для запиту на пошук новин, в тому числі при ще не запусненому додатку, фільтрація новин України, фонові оповіщення про нові звістки, впровадження більш функціонального озвучування новин, відновлення кешованими даними стану додатку після раптового зупинення операційною системою тощо.

					ІАЛЦ.045490.004 ПЗ	Лист
						62
Зм	Арк.	№ докум.	Підп.	Дата		

СПИСОК ВИКОРИСТАНИХ ЛІТЕРАТУРНИХ ДЖЕРЕЛ

1. Мобільний додаток “Ukr.net” [Електронний ресурс]. — Дата візиту 12.02.2021. — Режим доступу до ресурсу: <https://apps.apple.com/ua/app/ukr-net/id1212957488?l=ru>
2. Мобільний додаток “Virno” [Електронний ресурс]. — Дата візиту 14.02.2021. — Режим доступу до ресурсу: <https://virno.com.ua/ru>
3. Мобільний додаток “НВ” [Електронний ресурс]. — Дата візиту 16.02.2021. — Режим доступу до ресурсу: <https://apps.apple.com/ua/app/%D0%BD%D0%B2/id1194455118?l=ru>
4. Мобільний додаток “ГОРДОН: Новості” [Електронний ресурс]. — Дата візиту 18.02.2021. — Режим доступу до ресурсу: <https://apps.apple.com/ua/app/гордон-новости/id1282904011?l=ru>
5. Топ сайти в Україні [Електронний ресурс]. — Дата візиту 22.02.2021. — Режим доступу до ресурсу: <https://www.alexa.com/topsites/countries/UA>
6. Нативний vs кроссплатформений: відмінності підходів у розробці мобільних додатків [Електронний ресурс]. — Дата візиту 24.02.2021. — Режим доступу до ресурсу: <https://www.ironwaterstudio.com/ru/blog/nativnyj-vs-krossplatformennyj-razlichiya-podhodov-v-razrabotke-mobil-nyh-prilozhenij>
7. Документація мови програмування Objective-C [Електронний ресурс]. — Дата візиту 26.02.2021. — Режим доступу до ресурсу: <https://developer.apple.com/library/archive/documentation/Cocoa/Conceptual/ProgrammingWithObjectiveC/Introduction/Introduction.html>
8. Документація мови програмування Swift [Електронний ресурс]. — Дата візиту 27.02.2021. — Режим доступу до ресурсу: <https://docs.swift.org/swift-book/index.html>
9. Створення дизайну за допомогою Storyboard [Електронний ресурс]. — Дата візиту 28.02.2021. — Режим доступу до ресурсу: https://developer.apple.com/library/archive/documentation/ToolsLanguages/Conceptual/Xcode_Overview/DesigningwithStoryboards.html

Зм.	Арк.	№ докум.	Підп.	Дата

ІАЛЦ.045490.004 ПЗ

10. Що таке XIB? [Електронний ресурс]. — Дата візиту 30.02.2021. — Режим доступу до ресурсу: <https://medium.com/@tjcarney89/whats-a-xib-and-why-would-i-ever-use-one-58d608cd5e9b>
11. Документація фреймворку SwiftUI [Електронний ресурс]. — Дата візиту 01.03.2021. — Режим доступу до ресурсу: <https://developer.apple.com/documentation/swiftui>
12. UserDefaults в Swift 5.1 [Електронний ресурс]. — Дата візиту 02.03.2021. — Режим доступу до ресурсу: <https://betterprogramming.pub/userdefaults-in-swift-4-d1a278a0ec79>
13. Клас NSCoder [Електронний ресурс]. — Дата візиту 03.03.2021. — Режим доступу до ресурсу: <https://developer.apple.com/documentation/foundation/nscoder>
14. Протокол Codable [Електронний ресурс]. — Дата візиту 05.03.2021. — Режим доступу до ресурсу: <https://www.swiftbysundell.com/basics/codable/>
15. Документація СУБД SQLite [Електронний ресурс]. — Дата візиту 07.03.2021. — Режим доступу до ресурсу: <https://www.sqlite.org/docs.html>
16. Документація фреймворку Core Data [Електронний ресурс]. — Дата візиту 09.03.2021. — Режим доступу до ресурсу: <https://developer.apple.com/documentation/coredata>
17. Документація мобільної бази даних Realm [Електронний ресурс]. — Дата візиту 11.03.2021. — Режим доступу до ресурсу: <https://docs.mongodb.com/realm/sdk/>
18. Синтезатор мови AVSpeechSynthesizer [Електронний ресурс]. — Дата візиту 14.03.2021. — Режим доступу до ресурсу: <https://nshipster.com/avspeechsynthesizer/>
19. Документація хмарного сервісу Amazon Polly [Електронний ресурс]. — Дата візиту 17.03.2021. — Режим доступу до ресурсу: <https://docs.aws.amazon.com/polly/index.html>

20. Документація синтезатора мови Google Cloud Text-To-Speech [Електронний ресурс]. — Дата візиту 20.03.2021. — Режим доступу до ресурсу: <https://cloud.google.com/text-to-speech/docs>
21. Документація хмарної платформи Firebase [Електронний ресурс]. — Дата візиту 24.03.2021. — Режим доступу до ресурсу: <https://firebase.google.com/docs>
22. Документація фреймворку CloudKit [Електронний ресурс]. — Дата візиту 29.03.2021. — Режим доступу до ресурсу: <https://developer.apple.com/documentation/cloudkit>
23. Документація пошукової системи Algolia [Електронний ресурс]. — Дата візиту 04.04.2021. — Режим доступу до ресурсу: <https://www.algolia.com/doc/>
24. Документація пошукової системи Elasticsearch [Електронний ресурс]. — Дата візиту 08.04.2021. — Режим доступу до ресурсу: <https://www.elastic.co/guide/en/elasticsearch/reference/current/index.html>
25. Шаблон проектування MVC. Пасивна модель. [Електронний ресурс]. — Дата візиту 14.04.2021. — Режим доступу до ресурсу: <https://stefanoborini.com/book-modelviewcontroller/02-mvc-variations/04-variations-on-the-notification-strategy/03-passive-model.html>