**Прізвище:** Нестор

**Ім'я:** Христина

**Група:** КН-316

**Кафедра:** ІСМ

**Дисципліна:** Програмування мобільних пристроїв

**Перевірив:** Григорович В. Г.

**ЗВІТ**

до лабораторної роботи № 2

на тему "Розроблення інтерфейсу мобільного додатку"

**Мета роботи**

Навчитися розробляти простий інтерфейс програми, вивчити засоби розмітки FrameLayout, LinearLayout, вивчити елементи інтерфейсу EditText, ImageView, ScrollView, Button, начитися динамічно створювати елементи інтерфейсу з файлів розмітки, обробляти події, виконувати переходи між елементами програми.

**Виконання роботи**

1. Для створення користувацького інтерфейсу в подальшому буде використовуватись. Interface Builder – інструмент, створений компанією Apple саме для цієї цілі. Альтернативно – інтерфейс можливо також створювати, пишучи відповідний код, але для прототипування та на ранніх стадіях проекту, Interface Builder має свою перевагу – наочність.
2. Був створений новий проект – WonderChat- додаток для обміну тестовими повідомленнями та голосового зв’язку в реальному часі.
3. В даний момент було створено 2 можливі екрани – для логування та реєстрації відповідно: “Welcome to WonderChat”. “Profile”. Було застосовано наступні засоби UIKit:

* UILabel – для відображення короткого заголовку.
* UITextField – для надання можливості користувачі вводити свої персональні дані.
* UIButton – кнопки, для підтвердження дій та навігації в програмі.
* UIImageView – для відображення графічного елемента – в нашому випадку – фотографія(аватар) користувача
* UITabBarButton – кнопки особливого застосування для переходу між екранами нашого застосунку.

1. Для створення таблиць у додатках iOS застососують TableView – спеціальний обєкт у бібліотеці UIKit. Для імплементації таблиці у проект, контроллер нашого екрану з таблицею - ViewController повинен задовільняти мінімальні вимоги для TableView, а точніше повинен підлягати під протокол UITableViewDataSource та UITableViewDelegate:  
     
   override func tableView(\_ tableView: UITableView, cellForRowAt indexPath: IndexPath) -> UITableViewCell

override func tableView(\_ tableView: UITableView, numberOfRowsInSection section: Int) -> Int

Перший метод застосовують для конфігурації таблиці, а саме наповнення її даними з нашої моделі.

Другий метод повертає кількість рядків, які необхідно згенерувати для відображення.

Незалежно від кількості рядків у нашій таблиці, логіка влаштована таким чином, що генерується лиш невелика їх кількість, та якщо не всі рядки поміщаються на екрані мобільного пристрою, то при гортанні нашої таблиці, верхні рядки, які вийшли за межі екрану використовуються повторно для відображення нижніх рядків таблиці. Така реалізація забезпечує плавне гортання таблиці без надмірного навантаження на пам’ять та процесор.

1. За відображення кожного екрану у мобільному додатку відповідає клас - ViewController або його наслідники. ViewController не може відповідати більше, ніж за 1 екран.  
   ViewController відповідальний за звязок користувацького інтерфейсу з даними. Така архітектура називається – Model View Controller (MVC) та є найбільш поширеною при створення мобільних додатків.  
   Ідея цього підходу полягає в тому, щоб якомога чіткіше розділити обовязки компонентів програми.

**Висновок**

Навчилась розробляти простий інтерфейс програми.