

A still life composition featuring a vintage camera, a clock, a globe, and a map. The background is a large, aged world map. In the foreground, a vintage camera with a black leather strap and a large lens is positioned on the left. To its right is a round, antique-style clock with a wooden frame. Further right is a small glass jar containing pens and pencils. On the far right is a globe on a wooden stand. The scene is lit with warm, golden light, creating a nostalgic and travel-themed atmosphere.

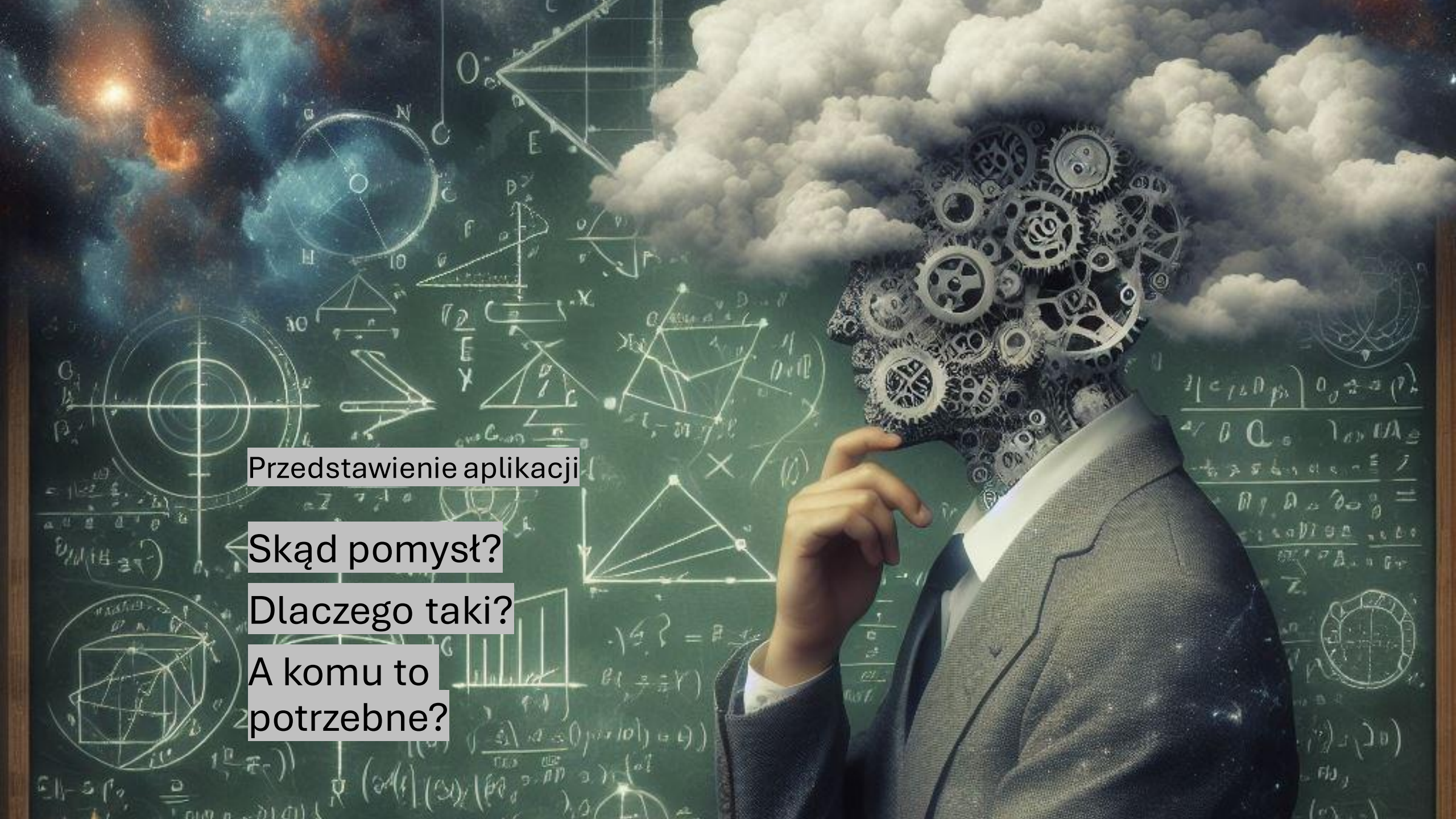
One Smile Travel (projekt JPWP- Aplikacja Mobilna w Android Studio)

Autorzy: Szymon Domagała, Piotr
Stasiak

Rozkład jazdy

- Przedstawienie aplikacji
- Skąd pomysł i dlaczego?
- Na czym opiera się projekt
- Techniczne przedstawienie aplikacji
- Kotlin vs Java
- Wstęp teoretyczny do zadań





Przedstawienie aplikacji

Skąd pomysł?

Dlaczego taki?

A komu to
potrzebne?

Użycie apki na własnym telefonie z Androidem

Wejść w ustawienia telefonu

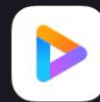
- Przewiń w dół i znajdź sekcję "O telefonie" lub "O urządzeniu". Dotknij tego.
- Znajdź opcję "Numer kompilacji" i dotknij ją siedem razy. Będziesz musiał potwierdzić swoje hasło lub kod PIN, jeśli masz je ustawione.
- Po siedmiokrotnym dotknięciu powinieneś zobaczyć komunikat informujący, że "Opcje programisty" zostały włączone.
- Teraz wróć do ustawień głównych. Powinieneś teraz zobaczyć nową opcję na dole listy o nazwie "Opcje programisty". Wejdź w nie
- Włącz debugowanie bezprzewodowe i zezwól na działanie w sieci ważne jest to aby wasz komputer z aplikacją był w tej samej sieci co telefon.
- Wybierz jedną z opcji połączenia się z android studio

15:36

69%



Ustawienia



Mi Wideo



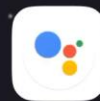
Pogoda



Narzędzia



Notatki



Asystent



Mi
Przeglądarka



LinkedIn



Netflix



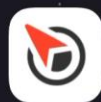
Facebook



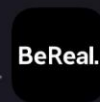
Instagram



Spotify



Yanosik



BeReal.



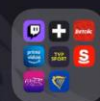
Flashscore



IKO



Folder



Folder



TikTok



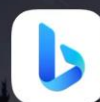
mobileMPK



Adobe Scan



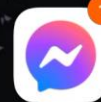
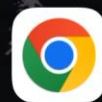
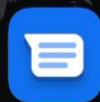
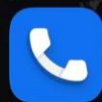
Jakdojade



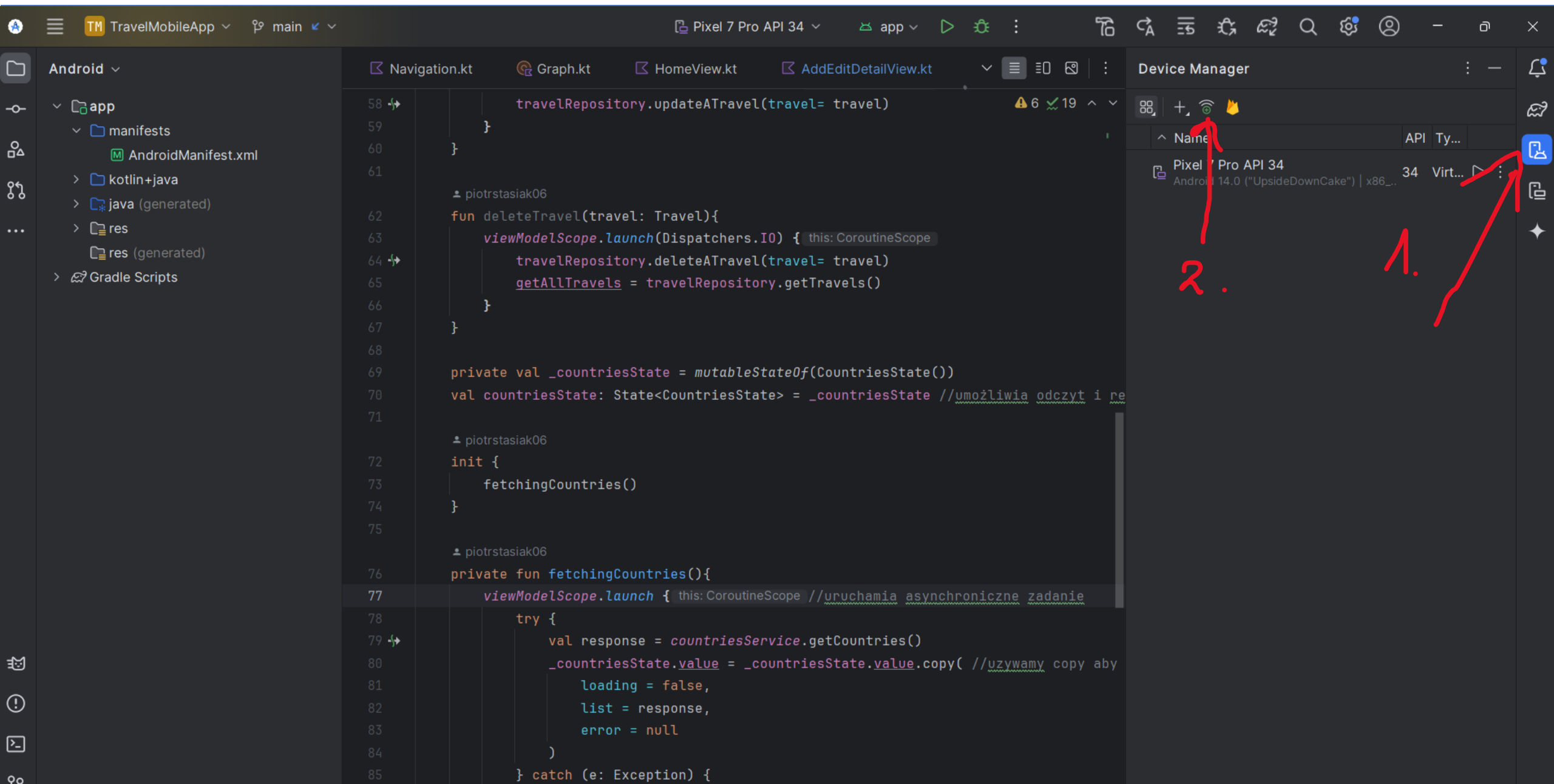
Bing



Authenticator



Połączenie z android studio





Pair new devices over Wi-Fi

Pair devices to enable wireless debugging. Pair camera-enabled devices using a QR code. Other devices can be paired using a pairing code. [Learn more](#)

Pair using QR code

Pair using pairing code

To pair an Android 11+ device
scan the QR code from your device



QR scanner available at:

Developer options > Wireless debugging > Pair using QR code

Close

Przedstawienie aplikacji



11:14



Email

Password

Login

Don't have an account? Sign up.


Map of your
travels


Your travels


Chat rooms



11:14



Email

Password

First Name

Last Name

Sign Up

Already have an account? Sign in.


Map of your
travels


Your travels


Chat rooms





Moldova



Bulgaria



Denmark



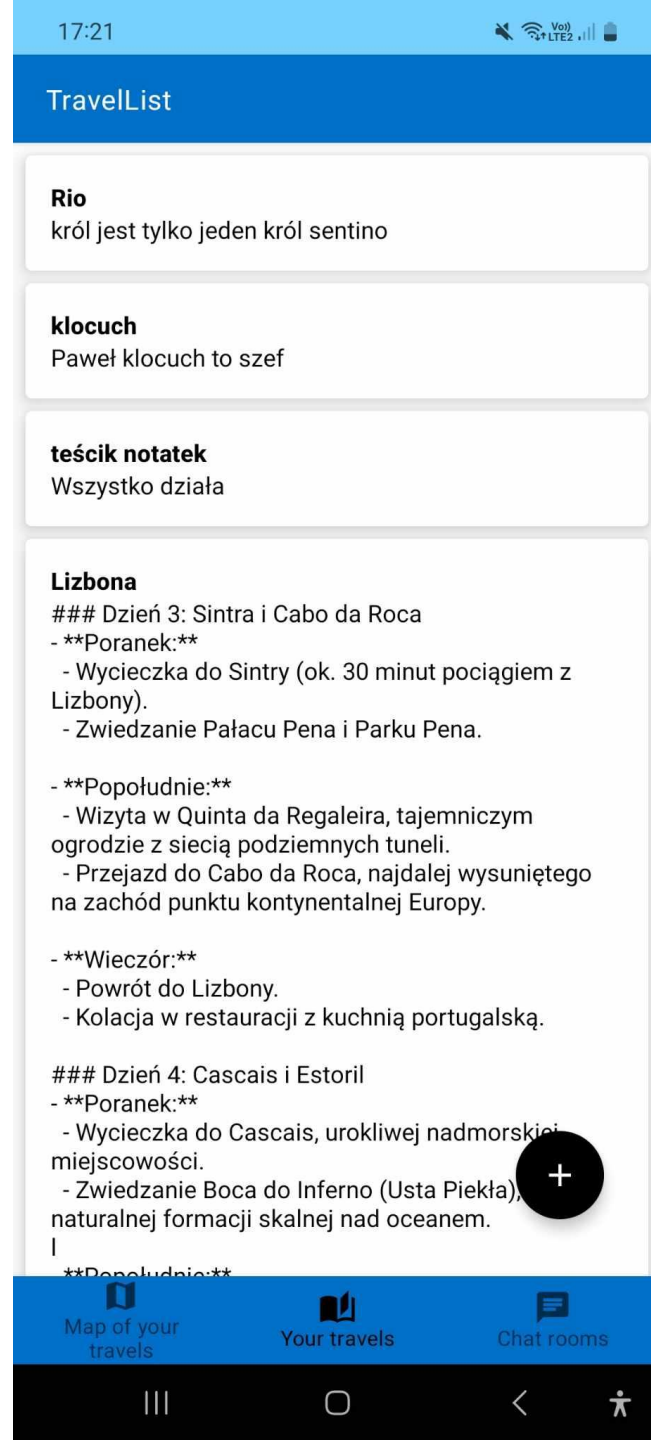
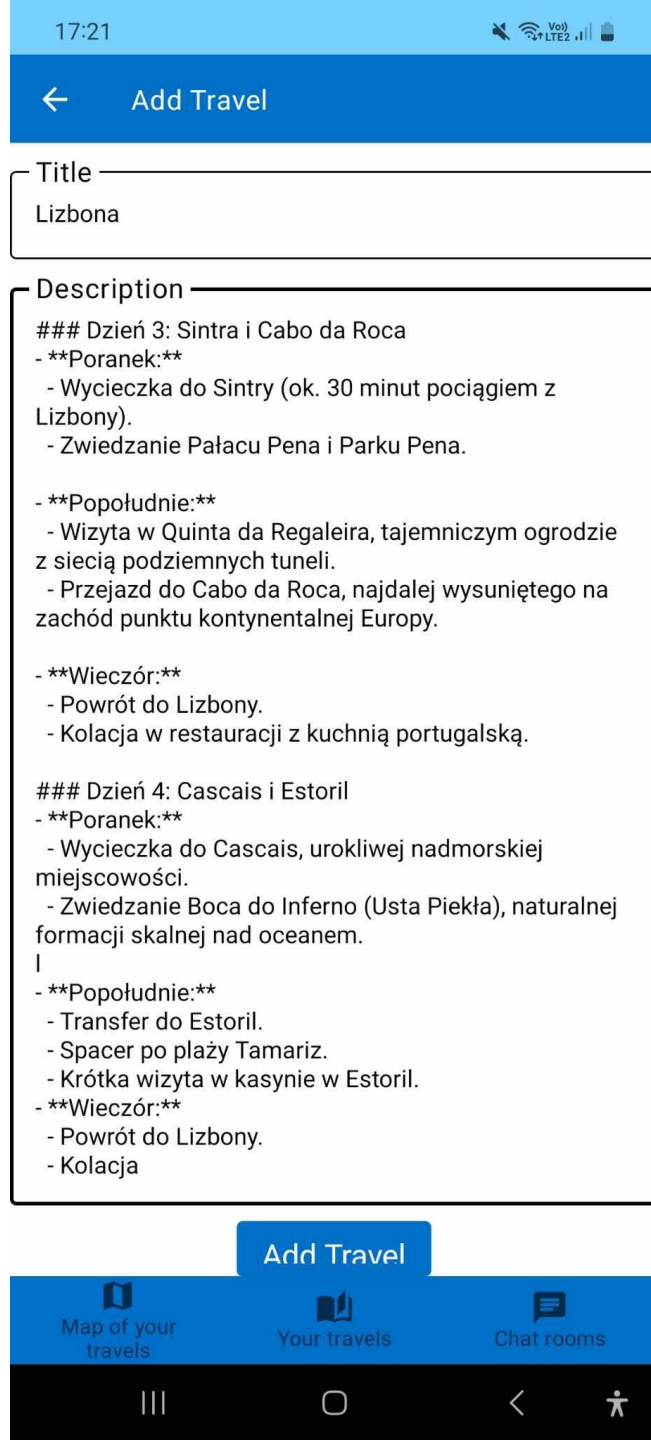
Luxembourg

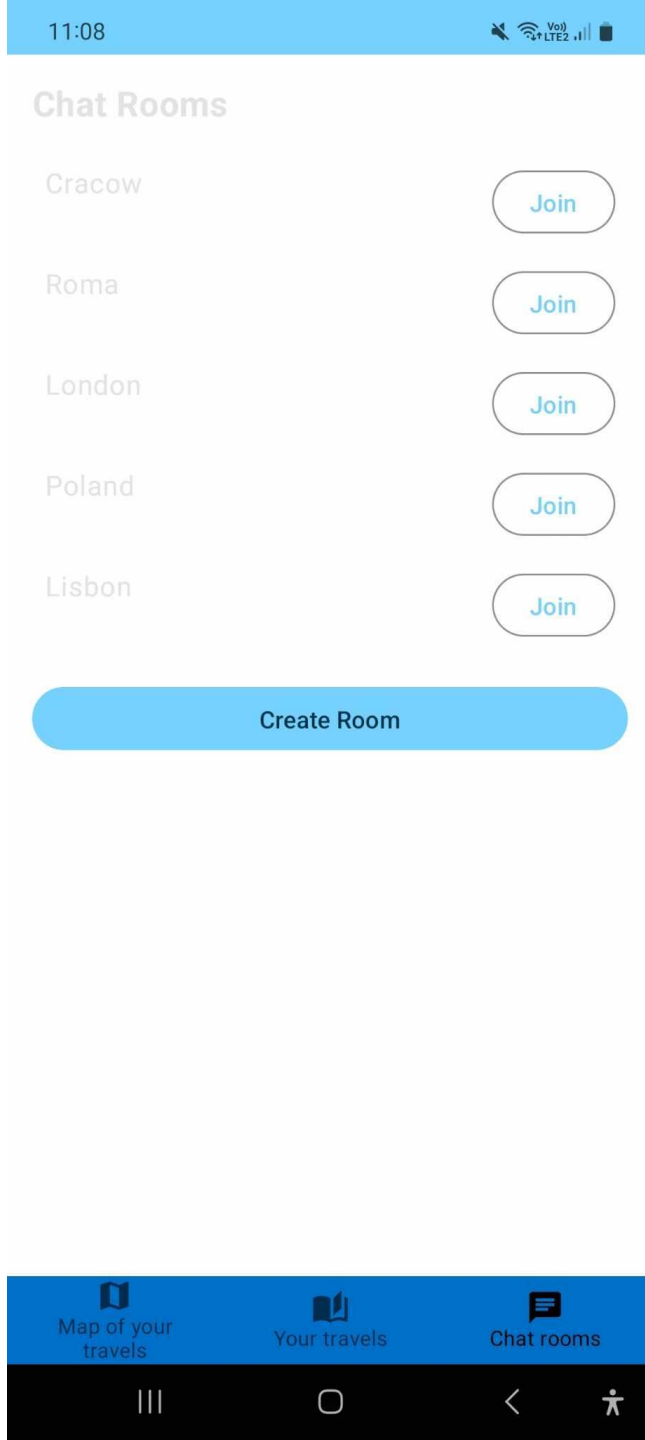


Faroe Islands



Slovenia





12:03



Search



Ryanair



Airbnb



Booking.com



Lime



TIER



Adobe Scan



Millennium



Useless



Teams



Medito



Joy -
Meditation a...



Headphones



TravelMobile-
App



Revolut



Fabryka Sily



QR & Barcode
Scanner



Docs



LinkedIn





Parę słów o technicznej stronie projektu

Co zostało wykorzystane przy tworzeniu aplikacji

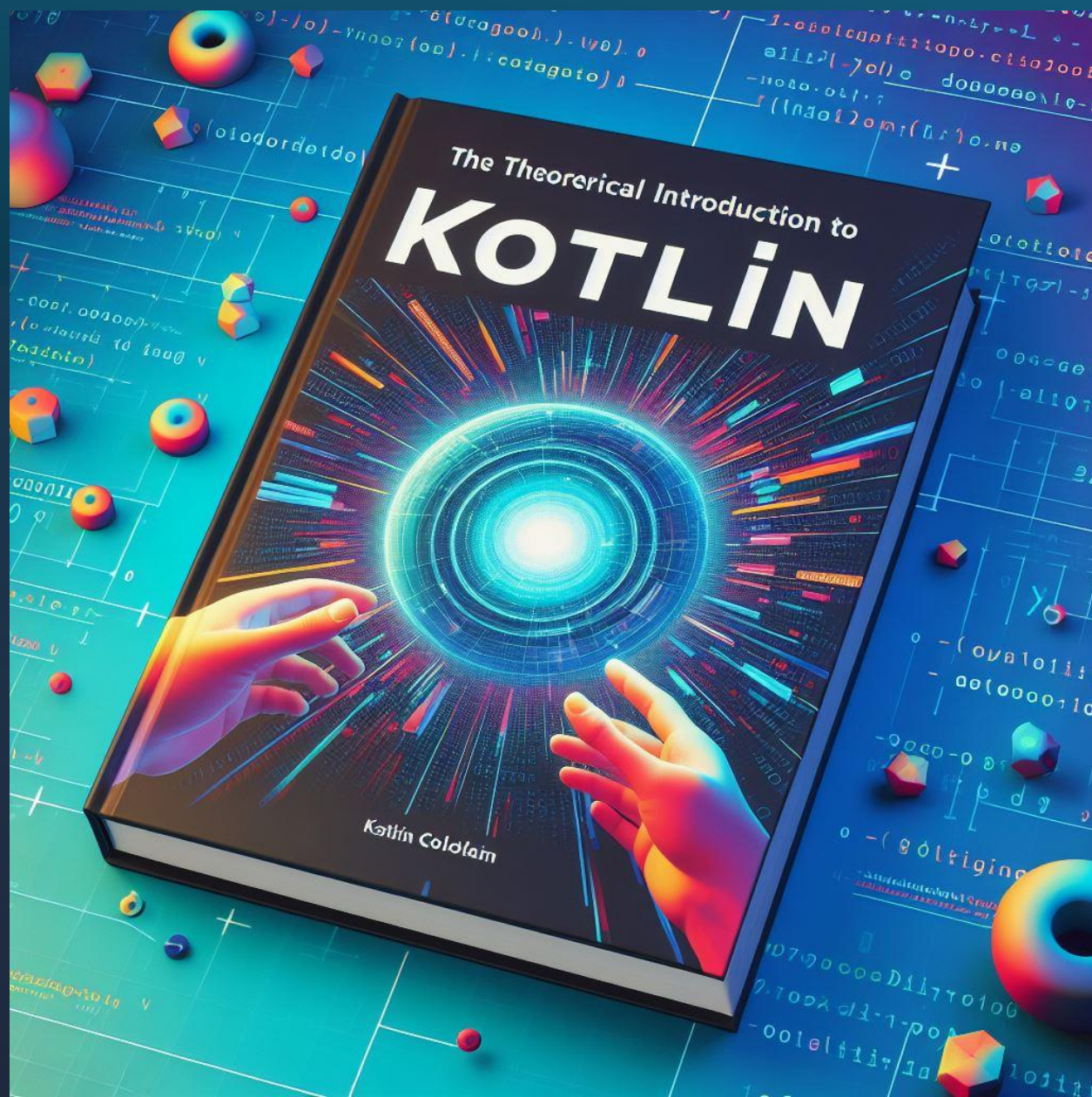
- API (application programming interface)
- Baza danych Firebase

Kotlin vs Java



vs





Podstawowe typy danych i tablice

```
fun main() {  
    val intVar: Int = 10  
    val stringVar: String = "Hello, Kotlin!"  
    val floatVar: Float = 3.14f  
}
```

```
fun main() {  
    val numbers: List<Int> = listOf(1, 2, 3, 4, 5)  
    val strings: List<String> = listOf("apple", "banana", "orange")  
}
```

Funkcje

```
fun add(a: Int, b: Int): Int {  
    return a + b  
}  
  
fun main() {  
    val result = add(5, 3) // result = 8  
}
```


Klasy

```
fun main() {  
    class Person(val name: String, val age: Int)  
  
    val person = Person("John", 30)  
    println("Name: ${person.name}, Age: ${person.age}")  
}
```

```
fun main() {  
    val greeting = "Hello"  
    val name = "Alice"  
    val message = "$greeting, $name!" // "Hello, Alice!"  
}
```

Instrukcje warunkowe

```
fun main() {  
    val x = 10  
    if (x > 5) {  
        println("x jest większe od 5")  
    } else {  
        println("x nie jest większe od 5")  
    }  
}
```

Funkcja random

```
fun main() {  
    //Losowa liczba z podanego zakresu  
    val losowaLiczba = (1..10).random()  
    println("Wylosowana liczba: $losowaLiczba")  
  
    //Losowa litera z zakresu od 'a' do 'z'  
    val losowyZnak = ('a'..'z').random()  
    println("Wylosowany znak: $losowyZnak")  
}
```


Źródła

- <https://www.imaginarycloud.com/blog/kotlin-vs-java/>
- <https://www.udemy.com/course/android-kotlin-developer/?couponCode=KEEPLEARNING>
- <https://designer.microsoft.com/image-creator>