

# Testowanie aplikacji w Android Studio

Wykorzystywane narzędzia:

- JUnit (Jupiter)
- Espresso

Podział na dwa katalogi/odrębne rodzaje testów:

- Test- testy uruchamiana bez potrzeby emulatora ani urządzenia z androidem, bezpośrednio na komputerze. Testowanie funkcji w kodzie, integralności baz danych.
- AndroidTest- testy wykonywane za pośrednictwem urządzenia z androidem lub emulatora. Testowanie API, Interfejsu Graficznego. Sprawdzanie łącza bluetooth pomiędzy fizycznymi urządzeniami.

Przebieg testów:

1. Testy jednostkowe oparte na podstawowych funkcjonalnościach aplikacji.
2. Ewentualne poszerzenie/edycja scenariuszy testowych.
3. Testowanie nowych funkcjonalności wymyślonych przez zespół w trakcie tworzenia aplikacji.
4. W przyszłości testowanie pilotów w praktyce, na fizycznych urządzeniach.

Przykładowy test jednostkowy, sprawdzający czy urządzenie zostało poprawnie dodane do listy urządzeń:

```
55      @Test
56      public void shouldCreateDevice(){
57          ExampleUnitTest exampleUnitTest = new ExampleUnitTest();
58          exampleUnitTest.addDevice( name: "Android", id: 1);
59          assertEquals( expected: false, getAllDevices().isEmpty());
60          assertEquals( expected: 1,getAllDevices().size());
61      }
62  }
```

Funkcje pomocnicze do testowania:

```
41
42     public void addDevice(String name, int id){
43         Device device = new Device(name, id);
44         validateDevice(device);
45     }
46
47     @ private boolean checkIfContactAlreadyExists(Device device){
48         return(deviceList.containsKey(device.id));
49     }
```

Testowanie czy Id pierwszego urządzenia jest poprawne

```
72     @Test
73     public void shouldCheckID() {
74         Device device = new Device( name: "Android1", createId());
75         assertEquals( expected: 1, device.id);
76     }
```

Test przy walidacji danych. Sprawdzenie czy wyjątek przy podaniu nazwy jest rzucany poprawnie.

```
78     @Test
79     public void shouldntAcceptEmptyName() {
80         Device device = new Device( name: "", id: 1);
81         assertThrows(RuntimeException.class, () -> device.validateName());
82     }
```

Drugi test walidacji danych. Tym razem Id.

```
84     @Test
85     public void shouldntAcceptEmptyId() {
86         Device device = new Device( name: "Android2", id: 0);
87         assertThrows(RuntimeException.class, () -> device.validateId());
88     }
```