



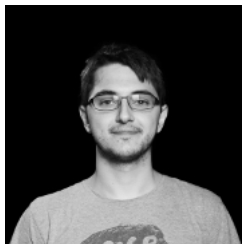
TESTY AUTOMATYCZNE JAKO ODZWIERCIEDLENIE DOMENY BIZNESOWEJ NA PRZYKŁADZIE APLIKACJI MOBILNEJ

■ **MICHAŁ GÓRSKI, MATEUSZ BOŚ, DAGMARA SURMA.**

Witamy!



Dagmara Surma,
Quality Assurance Engineer,



Mateusz Boś
Software Engineer,

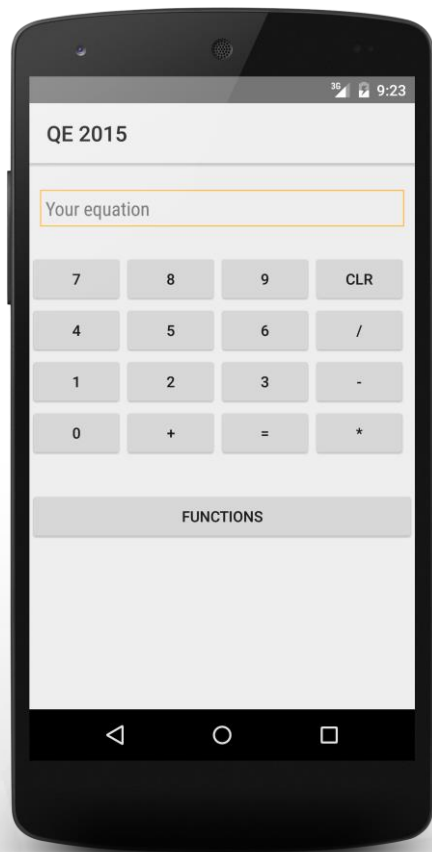


Michał Górski,
Software Engineer,

Cel warsztatu

Utworzenie testów bardziej odpornych na zmiany, łatwiejszych w rozszerzaniu.

Aplikacja



Specyfikacja

Przypadki testowe

Given user sees the main screen

When clicks on three

And clicks on seven

And clicks on add

And clicks on five

And clicks on equal

Then the result is forty two

Robotium, szybkie wprowadzenie

```
new Solo(getInstrumentation(), getActivity());  
solo.getView(id);  
solo.clickOnView(View);  
solo.waitForActivity(Activity.class, timeout);  
solo.waitForCondition(Condition, timeout);  
solo.typeText(viewFirstNumber, "37");  
assertEquals(expected, actual);
```

Piszemy testy

Przydatne wzorce projektowe

Page Object

- Metody publiczne odpowiadają akcjom które są dostępne na ekranie,
- Nie powinno się wystawiać „na zewnątrz” struktury ekranu(kontrolki etc.)
- Nie powinno się wykonywać w nich asercji
- Metody mogą zwracać inne PageObject’y
- Nie muszą reprezentować całej strony

Przydatne wzorce projektowe

Object Geanie / Object Mother / Object Factory

Celem tego wzorca jest dostarczenie złożonych obiektów, które są już odpowiednio zainicjalizowane.

Dzięki temu nasze testy nie muszą zaciemniać swojego celu zbędną inicjalizacją

Przydatne wzorce projektowe

Transporter

Jego celem jest odizolowanie kodu odpowiedzialnego za nawigację do odpowiednich miejsc w aplikacji.

Poprzez odizolowanie kodu nawigacji pozwala zmniejszyć do minimum ilość miejsc wymagających edycji po zmianie struktury UI. Jednocześnie pozwala poprawić czytelność kodu.

Przydatne

Custom assertion

Zwiększa czytelność kodu.



QUALITY
EXCITES

DZIĘKUJĘ ZA UWAGĘ