

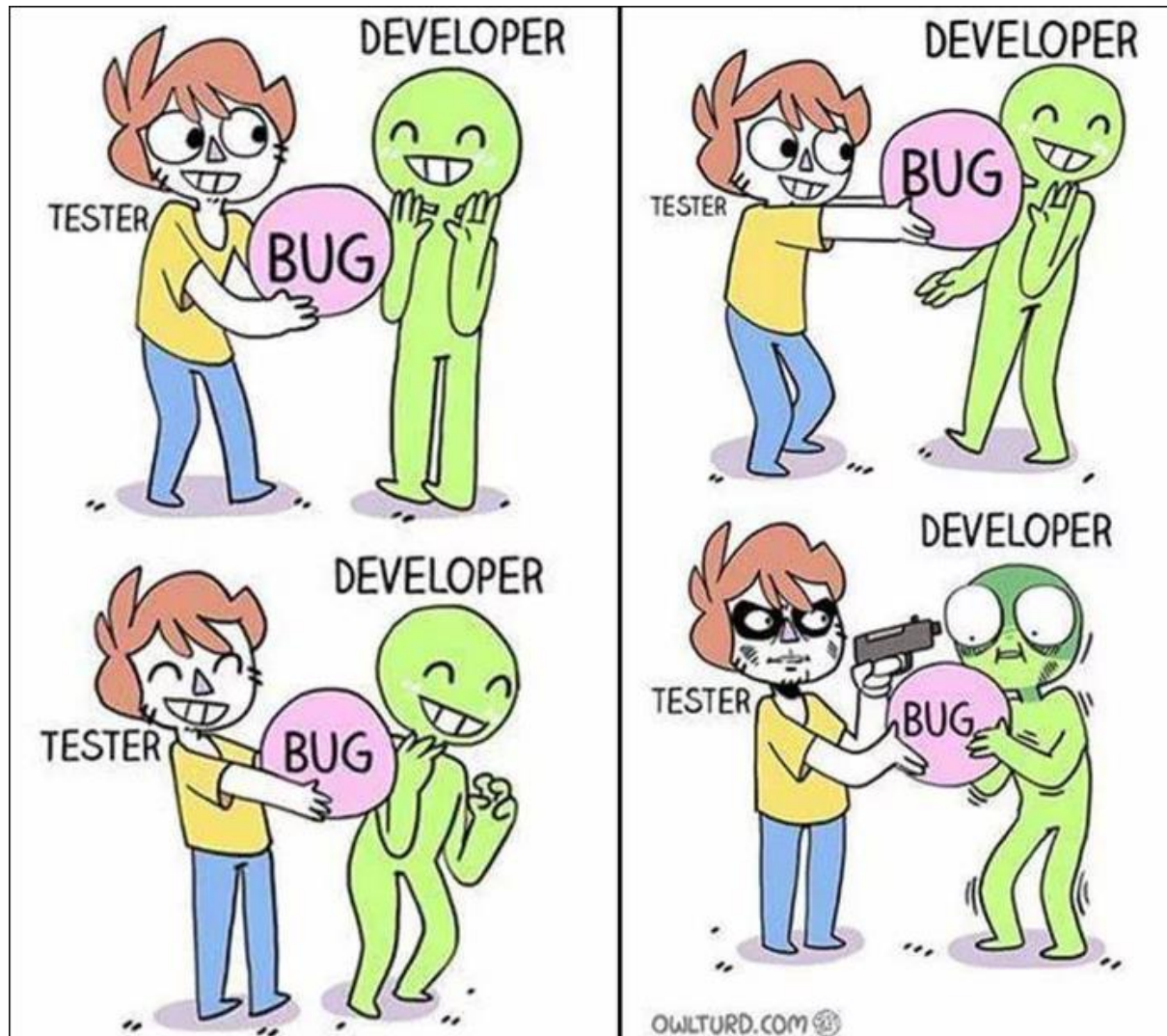
# Testy API

Czy da się przetestować 1600 API w 5 minut...

Gerard Miśta, Projektant IT  
Katowice • 6 Luty 2019



# Po co w 5 minut...



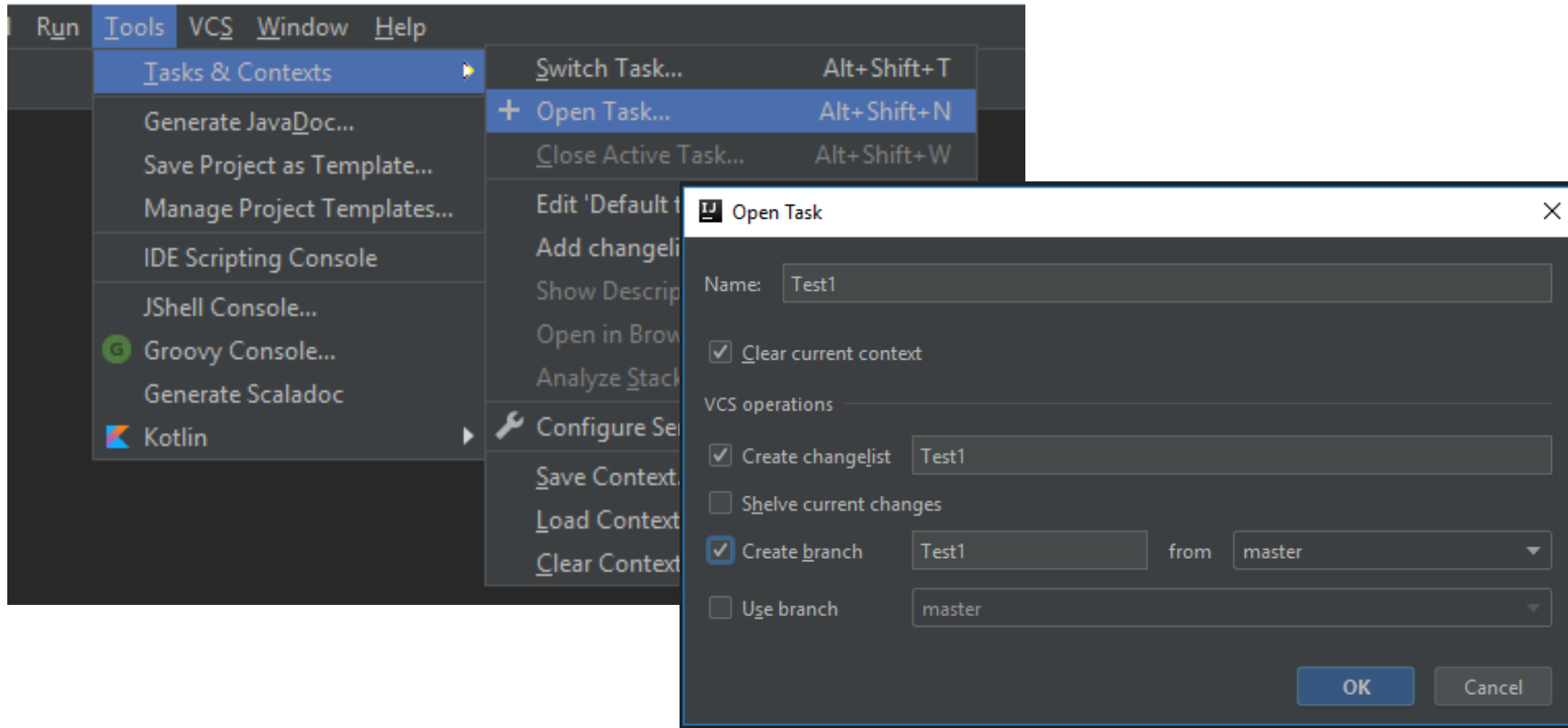
## Po co w 5 minut...



# Proces tworzenia testów – przygotowanie

- Autorski przewodnik do pisania testów + Proces oparty na Tasks w IntelliJ

IntelliJ IDEA

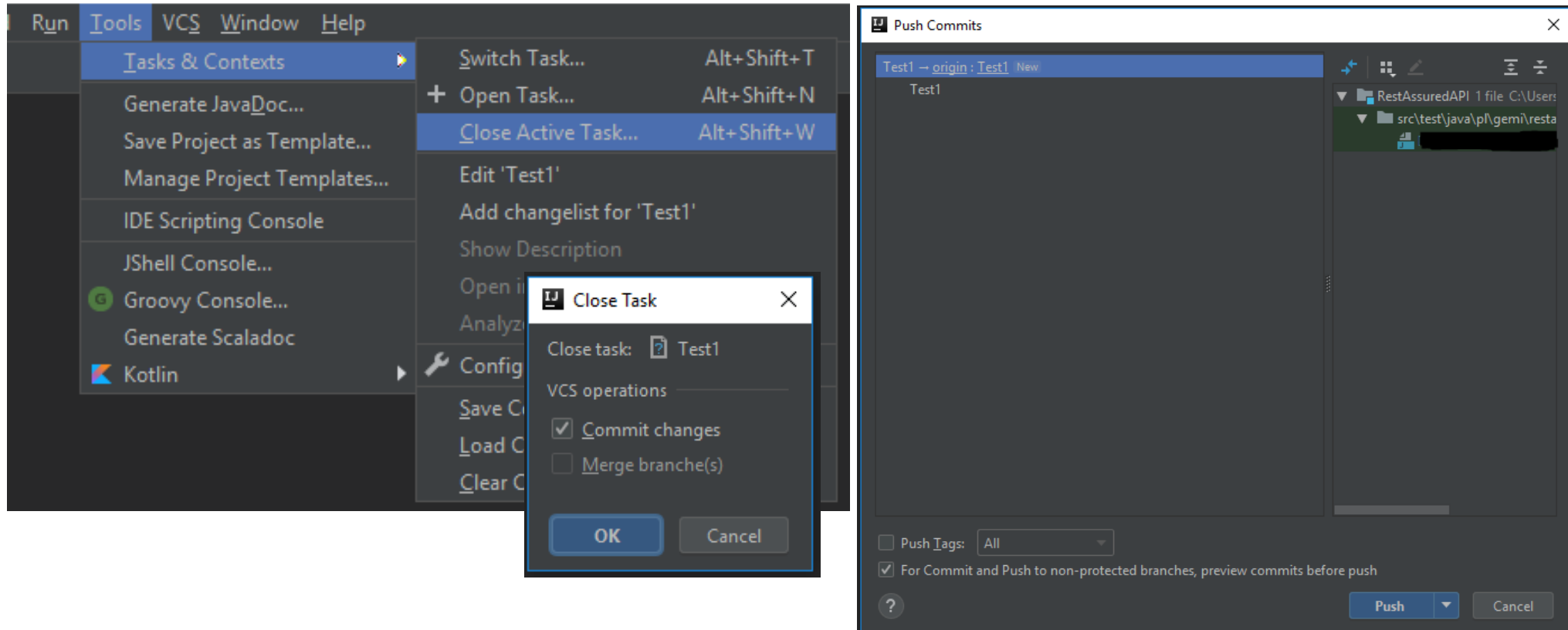




# Proces tworzenia testów – zakończenie

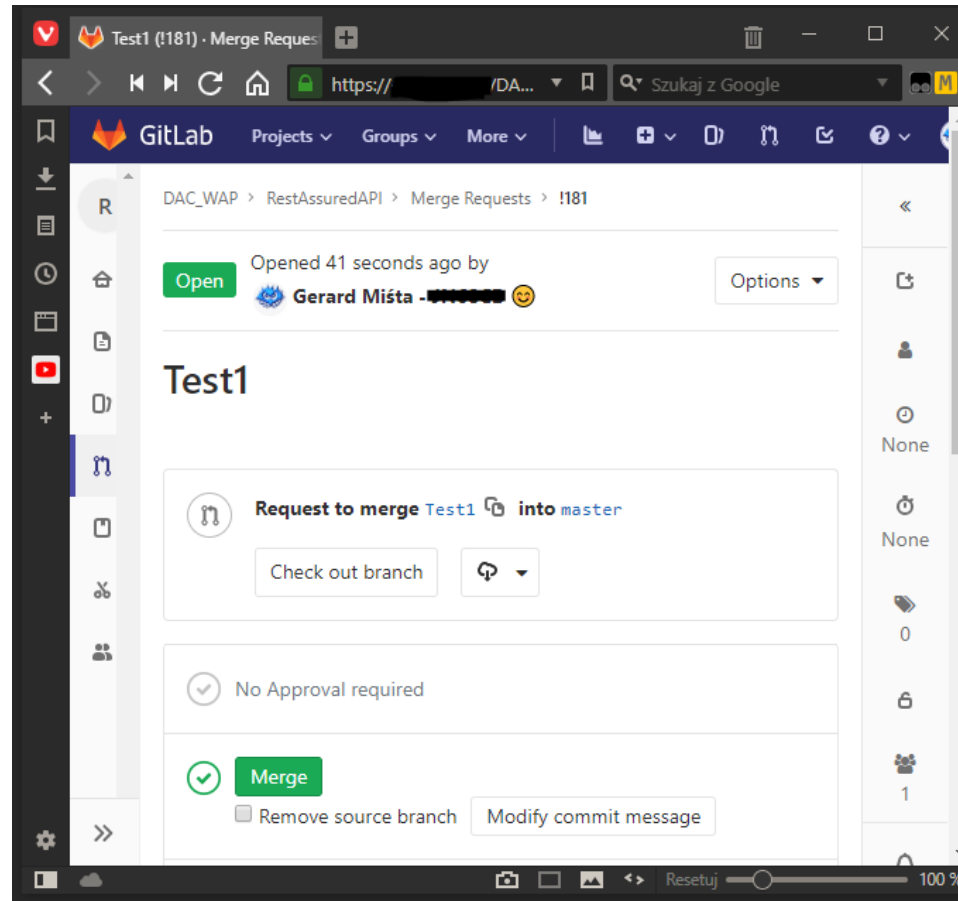
- Zakończenie procesu w IntelliJ – Close ActiveTask...

IntelliJ IDEA



# Proces tworzenia testów - zakończenie

- Każdy test przechodzi weryfikację przed „merge to master” - Merge Request



# Technologia – główna metoda

- Autorski framework oparty na Rest-Assured + TestNG + AsserJ + inne ☺

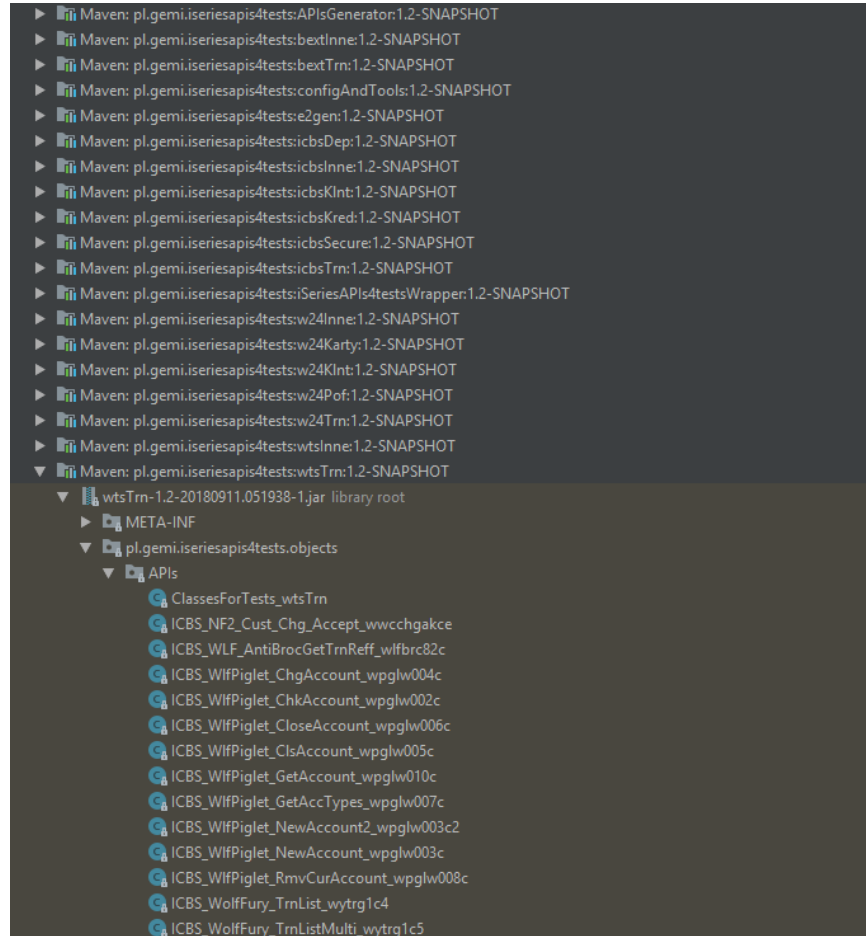


```
protected void getResponse(AbstractTest test) {
    // @formatter:off
    Object response =
        given().
            spec(specReqLog).
            auth().
                oauth2(tokenBeforeClass).
            log().all().
        when().
            header( headerName: "Connection", headerValue: "keep-alive").
            contentType("application/json").
            accept("application/json").
            body(test.getRequestBody()).
            post(test.getUrl()).
        then().
            log().all().
            assertThat().
                statusCode(200).
            extract().
                .response().as(test.getResponseClass());
    // @formatter:on
    test.setResponse(response);
}
```

AsserJ

# Technologia – API'ki

- Biblioteki zawierające ponad 1600 API - generowane automatycznie z Swaggerów.



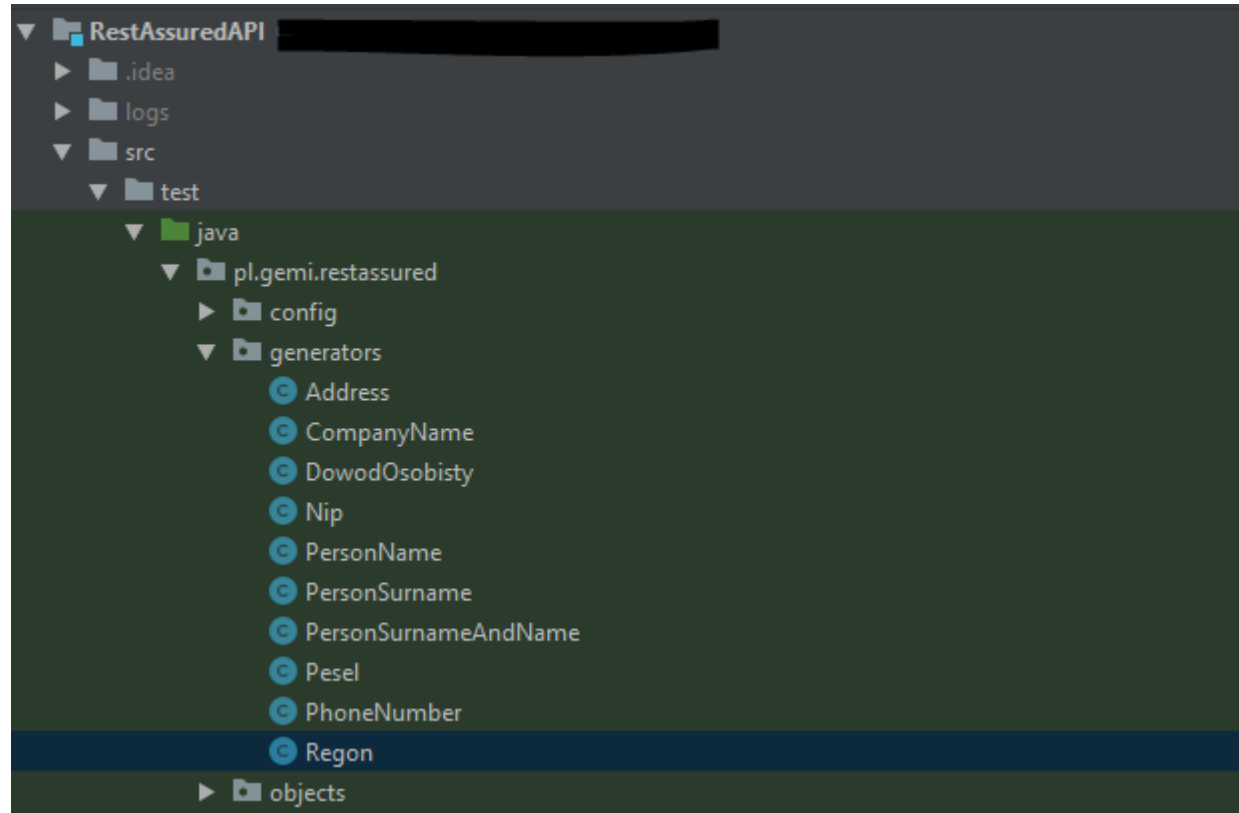


# Technologia – API'ki

- Proces generowania biblioteki API:
  - Budowanie configAndTools – dane na temat środowisk i portów
  - Budowanie APIsGenerator – najważniejszy moduł generujący „owrapowane” API
  - Budowanie poszczególnych bibliotek API:
    - Deleting ...\\objects\\APIs (includes = [\*\*/\*.java], excludes = [])
    - Deleting ...\\objects\\model (includes = [\*\*/\*.java], excludes = [])
    - Deleting ...\\src\\main\\resources (includes = [\*\*/\*.json, \*\*/\*.csv], excludes = [])
    - **Ściągnięcie Swaggera**
    - **swagger-codegen-maven-plugin**
    - **Ściągnięcie pliku pomocniczego zawierającego szczegóły o API**
    - **exec-maven-plugin – wywołanie APIsGenerator**
    - Ponowne budowanie biblioteki API
  - Zbudowanie Artefaktu który zawiera dependency do wszystkich bibliotek
  - Deploy wszystkich artefaktów ze źródłami do repozytorium Artefaktów (SNAPSHOTs)

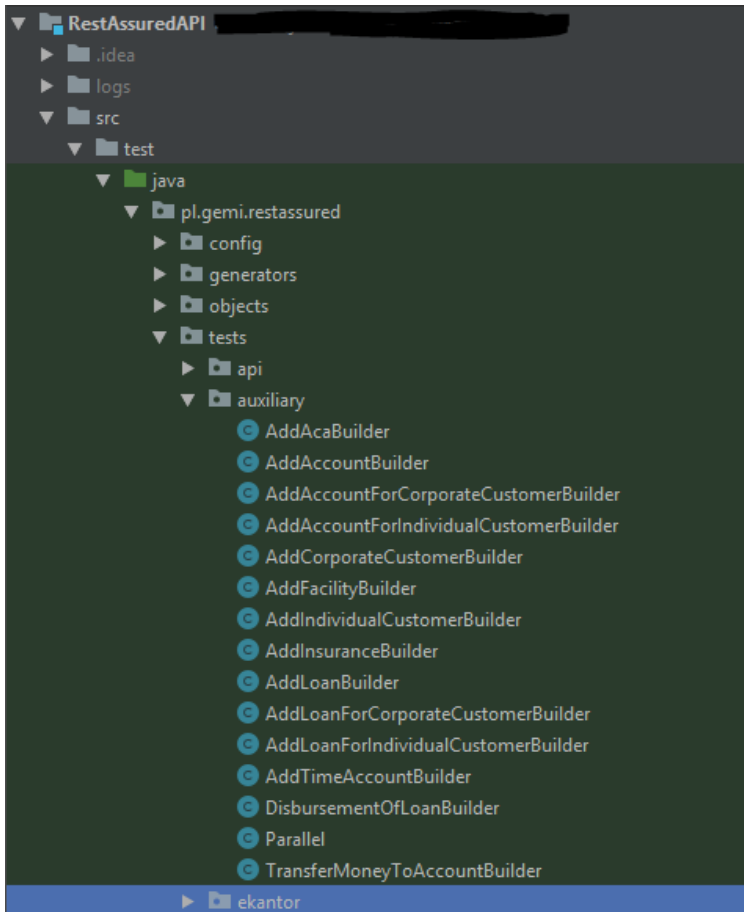
# Technologia – narzędzia pomocnicze

- Generatory – 10 szt.



# Technologia – narzędzia pomocnicze

- Buildery – 14 szt.



```
AddIndividualCustomerBuilder addIndividualCustomerBuilder =
    new AddIndividualCustomerBuilder().with
    idualCustomerBuilder
    tring idNumber)
    (m) withID (String idNumber)
    (m) withAddress (Address address)
    (m) withName (String name)
    (m) withSegment (String segment)
    (m) withSurname (String surname)
    (m) withPesel (String pesel)
    (m) withAge (BigDecimal age)
    (m) withNIP (BigDecimal NIP)
    (m) withMobilePhoneCountryCode (BigDecimal mobilePhoneCountryCode)
    (m) withMobilePhoneNumber (BigDecimal mobilePhoneNumber)
    Ctrl+Down and Ctrl+Up will move caret down and up in the editor >>
```

```
AddIndividualCustomerBuilder addIndividualCustomerBuilder =
    new AddIndividualCustomerBuilder()
        .withAge (new BigDecimal ( val: 32))
        .withSegment ("7A")
        .withMobilePhoneCountryCode (new BigDecimal ( val: 48))
        .withMobilePhoneNumber (new BigDecimal ( val: 888777666));

String customerNumber = addIndividualCustomerBuilder.build();
```

# Testy funkcjonalne - assertThat

```
ICBS_WTS_Sys_GetSts_wwttsxstsTests.java x
1 package pl.gemi.restassured.tests.api.icbs;
2
3 import ...
4
12
13 public class ICBS_WTS_Sys_GetSts_wwttsxstsTests extends TestBaseClass{
14
15
16 /**
17  * <b>Author:</b> GeMi <br>
18  * <b>Date:</b> 2018-06-20<br>
19  * <b>Description:</b> Check if status of WTS is "U" and if ICBS Date is current date.
20  *
21  */
22 @Test
23 public void checkIfWTSstatusIsUandICBSdateIsCurrentDate() {
24     ICBS_WTS_Sys_GetSts_wwttsxsts icbs_WTSSysGetSts_wwttsxstsObj = new ICBS_WTS_Sys_GetSts_wwttsxsts();
25     setDefaultValues(icbs_WTSSysGetSts_wwttsxstsObj.getRequest(), icbs_WTSSysGetSts_wwttsxstsObj.getCountin());
26     icbs_WTSSysGetSts_wwttsxstsObj.getRequest().setField("w");
27     getResponse(icbs_WTSSysGetSts_wwttsxstsObj);
28     assertThat(icbs_WTSSysGetSts_wwttsxstsObj.getResponseBody().getWwtra0o4().getWwicksdat()).isEqualTo(LocalDate.now().format(DateTimeFormatter.ofPattern("YYYY-MM-dd")));
29     assertThat(icbs_WTSSysGetSts_wwttsxstsObj.getResponseBody().getWwtra0o4().getWwstatus()).isEqualTo("U");
30 }
31 }
32
33
```

# Testy funkcjonalne – SoftAssertions fajne jest 😊

```
ICBS_WTS_Sys_GetSts_wwttsxstsTestsSoftAsserctions.java x
1  package pl.gemi.restassured.tests.api.icbs;
2
3  import ...
12
13 public class ICBS_WTS_Sys_GetSts_wwttsxstsTestsSoftAsserctions extends TestBaseClass{
14     /**
15      * <b>Author:</b> GeMi <br>
16      * <b>Date:</b> 2018-06-22<br>
17      * <b>Description:</b> Check if status of WTS is "U" and if ICBS Date is current date.
18      *
19      */
20     @Test
21     public void checkIfWTSstatusIsUandICBSdateIsCurrentDate() {
22         ICBS_WTS_Sys_GetSts_wwttsxsts icbs_WTSSysGetSts_wwttsxstsObj = new ICBS_WTS_Sys_GetSts_wwttsxsts();
23
24         setDefaultValues(icbs_WTSSysGetSts_wwttsxstsObj.getRequest(), icbs_WTSSysGetSts_wwttsxstsObj.getCountin());
25
26         icbs_WTSSysGetSts_wwttsxstsObj.getRequest().setField("w");
27
28         getResponse(icbs_WTSSysGetSts_wwttsxstsObj);
29
30         // use SoftAssertions instead of direct assertThat methods
31         SoftAssertions softly = new SoftAssertions();
32         softly.assertThat(icbs_WTSSysGetSts_wwttsxstsObj.getResponseBody().getWwtra0o4().getWwicsdat()).isEqualTo(LocalDate.now().format(DateTimeFormatter.ofPattern("YYYY-MM-dd")));
33         softly.assertThat(icbs_WTSSysGetSts_wwttsxstsObj.getResponseBody().getWwtra0o4().getWwstatus()).isEqualTo("U");
34         //Regexp w AssertJ
35         softly.assertThat(icbs_WTSSysGetSts_wwttsxstsObj.getResponseBody().getWwtra0o4().getWwstatus()).matches("[Uu]");
36         softly.assertThat(icbs_WTSSysGetSts_wwttsxstsObj.getResponseBody().getWwtra0o4().getWwstatus()).doesNotMatch("[^Uu]");
37         // Don't forget to call SoftAssertions global verification !
38         softly.assertAll();
39     }
40 }
41
42
```

# Testy funkcjonalne – wywołanie

The screenshot shows the Jenkins web interface for a Maven project named "WAP-RestAssuredAPI-TA-UAT". The browser address bar shows the URL "https://.../Testy/job/WAP-RestAssuredAPI-TA-UAT/". The Jenkins logo and user information "Miśta, G. (Gerard) | Wyloguj" are visible at the top. The breadcrumb navigation indicates the path: "Jenkins > Miśta, G. (Gerard) > Moje widoki > Testy > WAP-RestAssuredAPI-TA-UAT".

The main content area is titled "Maven project WAP-RestAssuredAPI-TA-UAT". On the left, there is a sidebar with navigation options: "Powrót do tablicy", "Status", "Rejestr zmian", "Przestrzeń robocza", "Uruchom", "Konfiguruj", "Moduły", and "Ask for renaming".

In the center, there are two links: "Przestrzeń robocza" (with a folder icon) and "Ostatnie zmiany" (with a document icon). On the right, there is a button "Wyłącz projekt" and a link "dodaj opis".

Below the "Przestrzeń robocza" and "Ostatnie zmiany" links, there is a bar chart titled "Zmiany wyników testów". The y-axis is labeled "count" and ranges from 0 to 3500. The x-axis represents time. The chart shows a single bar with a count of approximately 1500. There is a "Resetuj" button and a "100%" zoom level indicator at the bottom of the chart.



# Smoke testy – prawie automatycznie

```
APISmokeTestsFull.java x
1 package pl.gemi.restassured.tests.api;
2
3 import ...
4
5
6
7
8
9 public class APISmokeTestsFull extends TestBaseClass {
10
11     @Test
12     public void icbs_WTS_Sys_GetSts_wwttsxsts_SmokeTest() {
13         ICBS_WTS_Sys_GetSts_wwttsxsts icbs_WTS_Sys_GetSts_wwttsxstsObj = new ICBS_WTS_Sys_GetSts_wwttsxsts();
14         setDefaultValues(icbs_WTS_Sys_GetSts_wwttsxstsObj.getRequest(), icbs_WTS_Sys_GetSts_wwttsxstsObj.getCountin());
15         getResponse(icbs_WTS_Sys_GetSts_wwttsxstsObj);
16     }
17
18     @Test
19     public void bskapi_SO_get_branch_user_iacgetbru_SmokeTest() {
20         BSKAPI_SO_get_branch_user_iacgetbru bskapi_SO_get_branch_user_iacgetbruObj = new BSKAPI_SO_get_branch_user_iacgetbru();
21         setDefaultValues(bskapi_SO_get_branch_user_iacgetbruObj.getRequest(), bskapi_SO_get_branch_user_iacgetbruObj.getCountin());
22         getResponse(bskapi_SO_get_branch_user_iacgetbruObj);
23     }
24
25     @Test
26     public void icbs_Change_Officer_Branch_cfcchgofbr_SmokeTest() {
27         ICBS_Change_Officer_Branch_cfcchgofbr icbs_Change_Officer_Branch_cfcchgofbrObj = new ICBS_Change_Officer_Branch_cfcchgofbr();
28         setDefaultValues(icbs_Change_Officer_Branch_cfcchgofbrObj.getRequest(), icbs_Change_Officer_Branch_cfcchgofbrObj.getCountin());
29         getResponse(icbs_Change_Officer_Branch_cfcchgofbrObj);
30     }
31 }
```

# Smoke testy - automatyzacja

```
IcbsInneSmokeTests.java x
1  package pl.gemi.restassured.tests.api.smoke;
2
3  import ...
13
14  public class IcbsInneSmokeTests extends AbstractSmoke {
15
16      /**
17       * Excluded API's for smoke tests
18       */
19      protected Class[] getExclusions() { return super.getExclusions(); }
22
23      @DataProvider(name = "icbsInneDataProvider")
24      public Object[] getDp() {
25          List<Class> tests = new LinkedList<>(Arrays.asList(ClassesForTests_icbsInne.list));
26          List<Class> exclusions = Arrays.asList(getExclusions());
27          tests.removeAll(exclusions);
28          return tests.toArray();
29      }
30
31      @Test(dataProvider = "icbsInneDataProvider")
32      /unchecked/
33      public void apiTest(Class clazz) throws NoSuchMethodException, IllegalAccessException, InvocationTargetException, InstantiationException {
34          AbstractTest test = (AbstractTest) clazz.getConstructor().newInstance();
35          SettingAndGetting.setDefaultValues(test.getRequest(), test.getCountin());
36          getResponse(test);
37      }
38  }
39
```

# Smoke testy - wywołanie

The screenshot shows the Jenkins web interface for a Maven project named "Maven project WAP-RestAssuredAPI-TA-SmokeTests-UAT". The browser address bar shows the URL "https://.../Testy/job/WAP-RestAssuredAPI-TA-SmokeTests-UAT/". The Jenkins logo and user name "Mišta, G. (Gerard)" are visible in the top navigation bar. The left sidebar contains various navigation options: "Powrót do tablicy", "Status", "Rejestr zmian", "Przestrzeń robocza", "Uruchom", "Konfiguruj", "Moduły", "Ask for renaming", and "Ask for deletion". The main content area displays the project name and a bar chart titled "Zmiany wyników testów" (Test Results Changes). The chart shows a count of test results over time, with a y-axis labeled "count" ranging from 0 to 8000. The chart shows a low count of test results, with a "Wyłącz projekt" (Disable project) button visible. Below the chart, there are links for "Przestrzeń robocza", "Ostatnie zmiany", and "Ostatni wynik testów ('6 błędów / -7')".

# Co zyskaliśmy... Code in JAVA!

- Uruchamianie równoległe – prawie brak ograniczeń ;)
- Proste dodawanie nowych API, również spoza biblioteki – wystarczy Swagger i chwila ;)
- Kontrola nad jakością testów – MergeRequest, każdy nowy test to nowy branch
- Prosta ścieżka: RestAssuredAPI -> Serwer
- Bezpieczeństwo rozwiązania – Token generowany na podstawie certyfikatu
- Możliwość uruchomienia testów na środowiskach DEV, UAT, PREPROD, inne...
- API jako obiekty, ale również można wysłać dowolne Body (JSON, XML, Octet-Stream)
- Raport wykonania testów przez SPLUNK Alert + Monitoring dostępności

Czy da się przetestować 1600 API w 5 minut...

**DEMO**

