



JDI COMPOSITE ELEMENTS

31 JAN 2018

- **UI Objects. JDI**
- **JDI. Composite elements**
- **JDI Settings**
- JDI for any UI

- **All Composite elements**
- **Site Structure**
- **UI Objects**



COMPOSITE ELEMENTS

PAGE STRUCTURE



← → ↻ <https://www.epam.com/careers> ☆



SOLUTIONS

INDUSTRIES

ABOUT

IDEAS

CAREERS

CONTACT



Home > Careers

FIND YOUR DREAM JOB

Keywords (optional):

Saint-Petersburg

SEARCH →

Filter by: All Teams & Roles

THE 5 HOTTEST JOB VACANCIES AROUND EPAM

Senior ATG Developer 🔥	Software Engineering	San Francisco, CA, USA	Apply »
Front-End Engineer 🔥	Software Engineering	Mountain View, CA, USA	Apply »
Senior JavaScript Developer 🔥	Software Engineering	USA	Apply »

ContactPage (**Page**)

└─ ContactForm (**Form**)

└─ Name(TextField)

└─ Location (Dropdown)

└─ Message (TextArea)

└─ SubmitButton(Button)

└─ ~~Save as Draft~~

└─ ~~Submit~~

└─ Description (Text)

└─ Header (**Section**)

EpamSite (Site)

HomePage(Page)

SearchSection (Search)

SearchField(TextField)

SearchButton(Button)

Invitation(Text)

ContactPage(Page)

ContactForm(Form)

...

Description(Text)

```
@BeforeSuite(alwaysRun = true)
public static void setUp() {
    WebSite.init(EpamSite.class);
}
```

public class Header extends **Section**

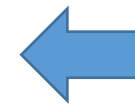
public class LoginForm extends **Form**

public class SearchBar extends **Search**

public class Alert extends **Popup**

public class Navigation extends **Pagination**

@JSite("http://epam.com/")



domain=https://www.epam.com/

public class EpamSite extends **WebSite**

@JPage(url = "/index.html", title = "Good site")

public class HomePage extends **WebPage**


```
@JSite(domain = "http://epam.com/")
public class EpamSite extends Website {
    @JPage(url = "/index.html")
    public static HomePage homepage;
    @JPage(url = "/login", title = "Login page")
    public static LoginPage loginPage;
    @FindBy(css=".nav")
    public static Menu navigation;
}

@BeforeSuite(alwaysRun = true)
public static void setUp() {
    Website.init(EpamSite.class);
}
```

```
public class Header extends Section {  
    @FindBy (css=".submit")  
    public Button submit;  
    @FindBy (css=".followMe")  
    public Link followMe;  
    @FindBy (css=".navigation")  
    public Menu navigation;  
  
    public void openAbout() {  
        followMe.click();  
        navigation.select(ABOUT);  
    }  
}
```

USAGE

```
header.submit.click();  
header.menu.isSelected();  
header.openAbout();
```

```
@JSite(domain = "http://epam.com/")
public class EpamSite extends Website {
    @FindBy(id = "header")
    public static Header header;
    ...
}
```

```
public class Header extends Section {
    @FindBy (css=".submit")
    public Button submit;
    @FindBy (css=".followMe")
    public Link followMe;
    @FindBy (css=".navigation")
    public Menu navigation;
}
```

USAGE

```
header.submit.Click();
```



```
driver
.findElement(By.id("header"))
.findElement(By.css(".submit"))
).click()
```

```
@JPage(url = "/main", title = "Good site", urlTemplate = "/main?\d{10}",  
urlCheckType = MATCH, titleCheckType = CONTAINS)  
public class HomePage extends WebPage
```

USAGE

```
homepage.open();  
homepage.checkOpened();  
homepage.isOpened();
```

```
homepage.refresh();  
homepage.back();  
homepage.forward();  
homepage.addCookie();
```



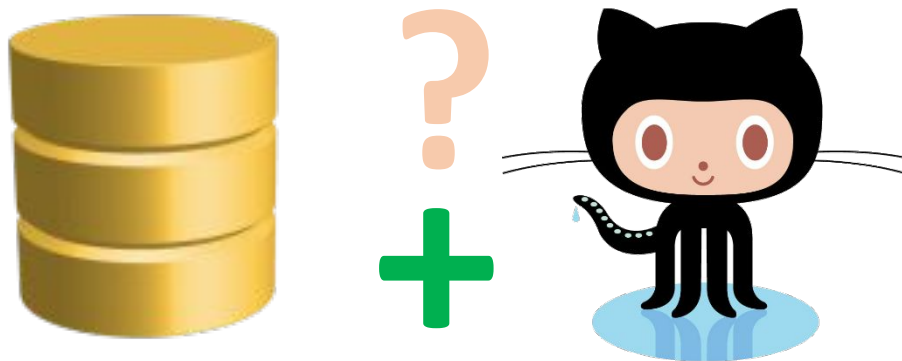


ENTITY DRIVEN TESTING

- Provide List<*User*> for test



- Provide List<*User*> for test
- 0. Have *DefaultUser* in DB



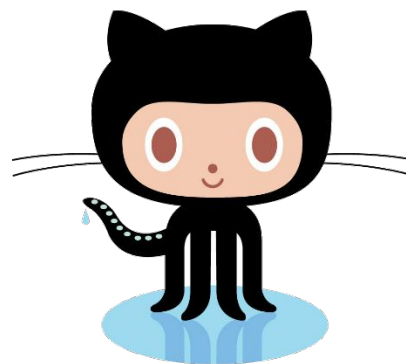
- Provide List<*User*> for test
- 0. Have *DefaultUser* in DB
- 1. Login with *DefaultUser*

LOGIN

Roman

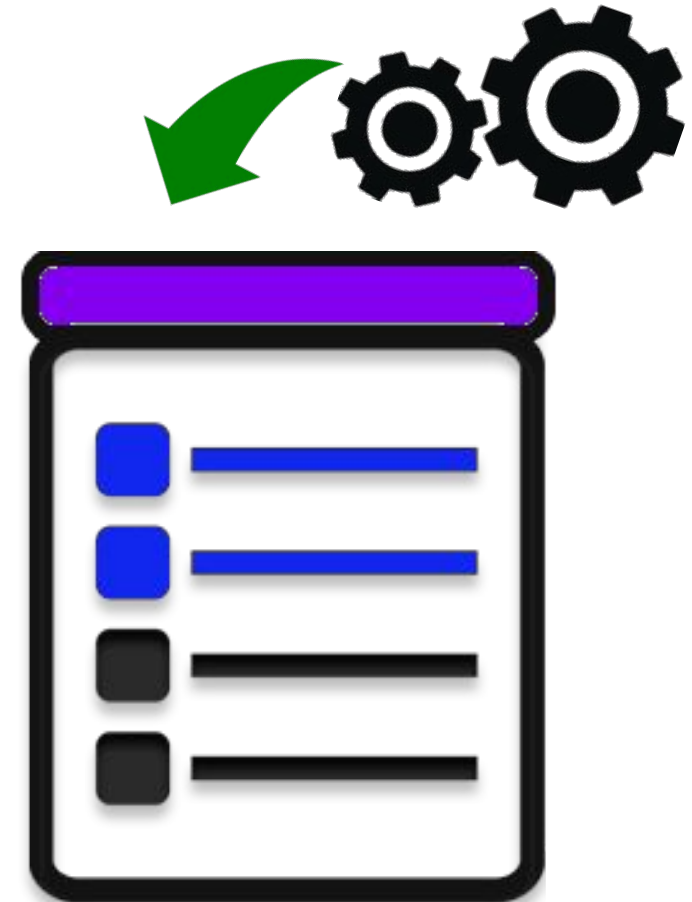
Iovlev

LOGIN



- Provide List<*User*> for test
- 0. Have *DefaultUser* in DB
- 1. Login with *DefaultUser*
- 2. Submit Contact Us Form for *DefaultUser*

CONTACT DETAILS	
<input type="text" value="Roman"/>	<input type="text" value="Iovlev"/>
<input type="text" value="roman_iovlev@epam.com"/>	<input type="text" value="89212345678"/>
<input type="text" value="Epam"/>	<input type="text" value="Academic/Student"/>
<input type="text" value="Russian Federation"/>	



- Provide List<User> for test
- 0. Have *DefaultUser* in DB
- 1. Login with *DefaultUser*
- 2. Submit Contact Us Form for *DefaultUser*
- 3. Get *Act. Opening* from Vacancy table

We found 26 job openings for you

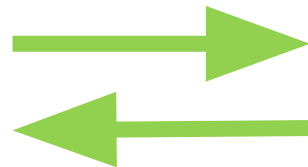
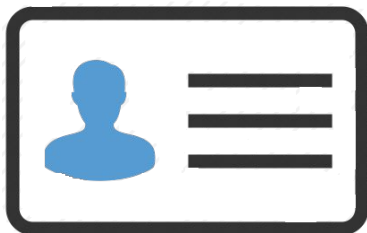
SORT BY:   

3D Artist	User Experience & Design	St-Petersburg, Russia	Apply »
Business Analyst (Mining Industry)	Consulting & Business Analysis	St-Petersburg, Russia	Apply »
Business Analyst/Consultant (OpenLink Endur)	Consulting & Business Analysis	St-Petersburg, Russia	Apply »
Front-end Developer (JavaScript Developer)	Software Engineering	St-Petersburg, Russia	Apply »
Infrastructure Automation Architect (Linux + ...)	Software Architecture	St-Petersburg, Russia	Apply »
Software Engineering		St-Petersburg, Russia	Apply »

www.epam.com/careers/job-listings/job.10229



- Provide List<*User*> for test
- 0. Have *DefaultUser* in DB
- 1. Login with *DefaultUser*
- 2. Submit Contact Us Form for *DefaultUser*
- 3. Get *Act. Opening* from Vacancy table
- 4. Assert *Act. Opening* equals to *Exp. Opening*

Actual**Expected**

```
public class LoginForm extends Form<User> {  
    @FindBy (css=".login")  
    public TextField login;  
    @FindBy (css=".psw")  
    public TextField password;  
  
    @FindBy (css=".submit")  
    public Button submit;  
    @FindBy (css=".cancel")  
    public Button cancel;  
}
```



```
public class User {  
    public String login = "roman";  
  
    public String password = null;  
}
```

```
@Test  
public class simpleTest(User user) {  
    LoginForm.login(user);  
    ...  
}
```

@Test

```
public void formTest(User admin) {  
    loginForm.loginAs(admin);  
    filter.select(admin.name);  
    Assert.each(results).contains(admin.name);  
    admin.update(results.get(1));  
    payForm.submit(admin.creditCard);  
    Assert.areEquals(DB.transactions.get(1),  
        admin.creditCard);  
}
```

USAGE

```
loginForm.fill(user);  
loginForm.submit(user);  
loginForm.verify(user);  
loginForm.check(user);  
  
loginForm.cancel(user);  
loginForm.save(user);  
loginForm.publish(user);  
loginForm.search(user);  
loginForm.update(user);  
...
```




```
public class SearchBar extends Search { }
```

```
public class Navigation extends Pagination { }
```

```
public class Confirmation extends Popup { }
```

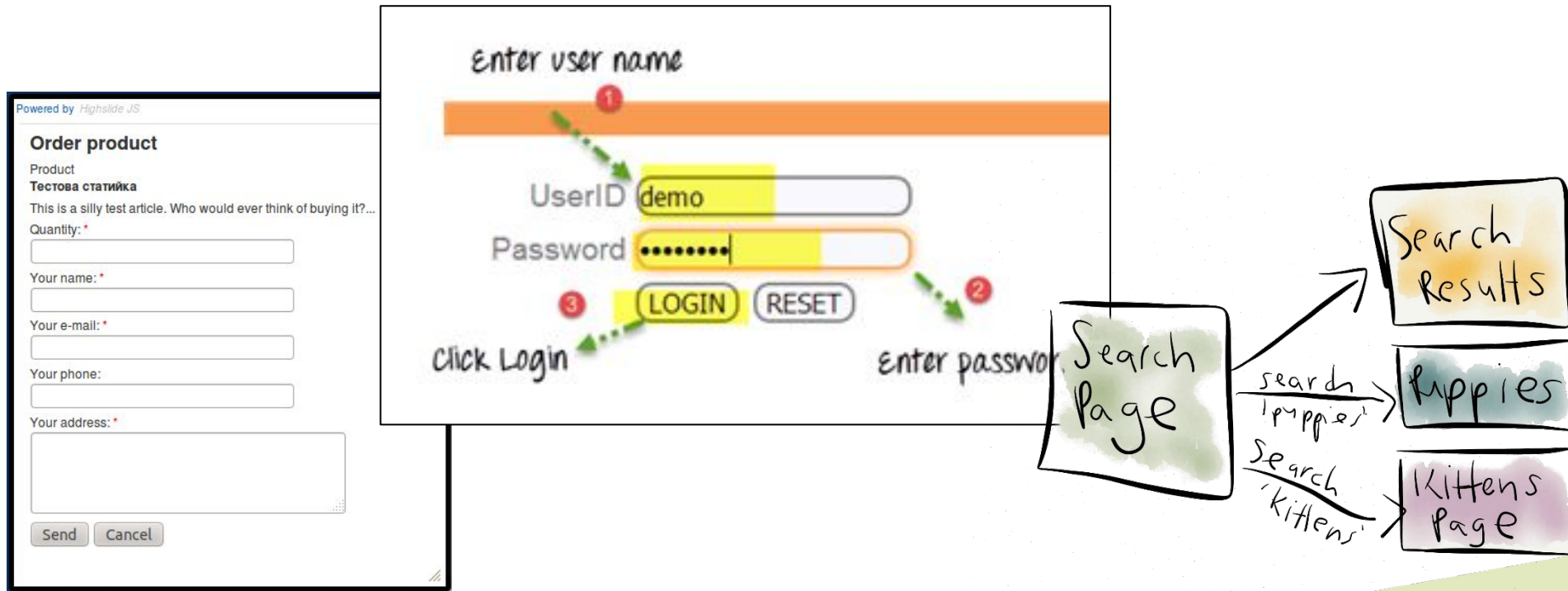
```
...
```

```
public class ContactConfirm extends Confirmation { }
```

```
// implements IComposite
```

UI OBJECTS PATTERN

Page Objects

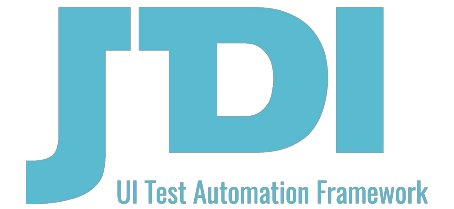


UI Elements



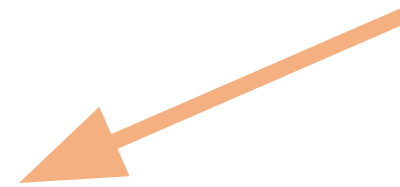
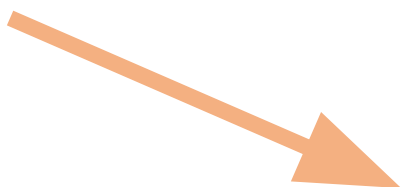
Flexible implementation

Interfaces



Page Objects

UI Elements



UI OBJECTS



Interfaces

- UI Objects. JDI
- JDI. Composite elements
- **JDI Settings**
- **JDI for any UI**



roman.lovlev

roman_iovlev@epam.com



prox318is

danila_morokov@epam.com



troektor1988osu

dmitry_lebedev1@epam.com



atlasova_julia

yulia_atlasova@epam.com

