In order to take screenShot we need a driver instance. There is a interface ITestContext that extends interface IAttributes which has method void setAttribute(String var1, Object var2);

В общем непонятно как но можно сделать вот так:

@BeforeTest

public void setUpDriver(**ITestContext tct**) throws MalformedURLException {

//this value is set in pom.xml (it is controlling the way how ro run the framework wia Docker or locally)

this.driver=Boolean.parseBoolean(ConfigurationProperties.get(Constants.GRID\_ENABLED)) ? getRemoteDriver() : getLocalDriver();

**tct.setAttribute**(Constants.DRIVER, this.driver);

//ITestContext is class that stores information about current test, and one of the methods is to create

//additional attribute which we can use later. Every test will have its own object of ITestContext.

}

tct <- - это обьект который несёт информацию о текущем тесте

По итогу в классе где мы создаём общую сущность driver-а мы как бы записываем драйвер в этот объект ‘tct’ c помощью метода .setAtribute(String var1, Object var2). Каждый тест будет иметь свою сущность этого **ITestContext** .

Также необходимо создать класс который implements ITestListener и override method .onTestFailure(ITestResult result); this method will invoke every time the test fails. Но для его использования нам нужен драйвер, то, что мы и делали в setUpDriver

**public class TestListener implements ITestListener {**

**@Override**

**public void onTestFailure(ITestResult result) {**

**TakesScreenshot driver = (TakesScreenshot) result.getTestContext().getAttribute(Constants.DRIVER);**

**String screenshot = driver.getScreenshotAs(OutputType.BASE64);**

**String htmlImageFormat = "<img width=700px src='data:image/png;base64,%s' />";**

**String htmlImage = String.format(htmlImageFormat, screenshot);**

**Reporter.log(htmlImage);**

**}**

In order to use this class we need to add it to the class which is responsible for creating webdriver instance.

