

Zadání 2. projektu ITS 2023/24

- Autor, revize: Aleš Smrčka, 2024-03-13
- Název projektu: GUI testy e-commerce systému OpenCart
- Přílohy: Projekt2-skripty.zip obsahující soubory `docker-compose.yml`, `get_driver.py` a `run-test.sh`

Cíl

Vytvořte automatické testy (Selenium) pro vámi navrženou testovací sadu na základě BDD scénářů.

Úkoly

1. Studium a instalace nástrojů

Nastudujte integrované prostředí pro tvorbu testů Selenium IDE:

- [ITS10] Přednáška ITS v 9. týdnu LS 2023/24
- [SeleniumIDE] <https://www.selenium.dev/selenium-ide/>
- [SeleniumPythonBindings] <http://selenium-python.readthedocs.io/>

1.1 Instalace Selenium IDE Nainstalujte si Selenium IDE pro Firefox nebo Chrome. Potřebujete prohlížeč s nainstalovaným rozšířením Selenium IDE: <https://www.selenium.dev/selenium-ide/>

1.2 Zkušební test Vytvořte si zkušební test principem nahrání (record+replay) v Selenium IDE. Někdy pro identifikaci objektů/cílů (tj. tlačítek, odkazů, rámečků) pro příkazy Selenia nestačí identifikace poskytnutá skrz Selenium IDE (name, id, link text), ale je potřeba identifikaci upřesnit. K tomu může posloužit CSS selektor nebo výraz XPath.

- [CSSSelectors] http://www.w3schools.com/cssref/css_selectors.asp
- [ChromeConsoleXPath] <https://stackoverflow.com/a/22573161>
- [XPath] https://www.w3schools.com/xml/xpath_intro.asp

1.3 Instalace behave Nainstalujte si projekt behave [Behave] pro automatizaci BDD testů. Vyzkoušejte si základní příklad [BDDHello].

- [Behave] <https://behave.readthedocs.io/en/latest/install.html>
- [BDDHello] <https://behave.readthedocs.io/en/latest/tutorial.html>

1.4 Instalace Python-selenium Nainstalujte si rozhraní Selenium pro jazyk Python3 (modul selenium). V tomto projektu se očekává Python verze 3.

- [PyPi] <https://pypi.org/project/selenium/>
- [RTDInstall] <https://selenium-python.readthedocs.io/installation.html>

V případě, že nemáte počítač s administrátorským přístupem:

- [PipUser] https://pip.pypa.io/en/latest/user_guide/#user-installs

1.5 Vazba na Selenium Server Upravte test v jazyce Python pro spuštění na Selenium Server.

Používejte `selenium.webdriver.Remote`. Inspirujte se na [RTDSeServer] a příkladem v souboru `get_driver.py`. Selenium server/grid pro účely projektu ITS musíte rozjet na vlastním počítači. Ideální je použít platformu docker a oficiální kontejnery pro Selenium Hub (spravuje připojení na jednotlivé automatizované webové prohlížeče) a uzly (webové prohlížeče). Pro automatizaci spuštění a ukončení Selenium Serveru a testovacího prohlížeče Firefox máte upraven soubor `docker-compose.yml` z repozitáře [SUT].

Časový limit pro čekání na DOM elementy je doporučen max 15 sekund.

- [SUT] <https://pajda.fit.vutbr.cz/smrcka/its-2024>
- [RTDSeServer] <https://selenium-python.readthedocs.org/getting-started.html#using-selenium-with-remote-webdriver>

2. Implementujte testy

2.1 Vámi navržené testy Implementujte automatizované testy vaší sekce testované aplikace. Testy budou zahrnovat vámi navržené testy v testovacím plánu odevzdaným v prvním projektu. Implementované testy nemusí být úplně stejné, jako testy popsané v návrhu testů v testovacím plánu.

Testy vyžadující nahrávání souborů nebo operace drag&drop implementovat nemusíte.

- [SeleniumPractices] <https://mestachs.wordpress.com/2012/08/13/selenium-best-practices/>
- [PageObject] <http://blog.activelylazy.co.uk/2011/07/09/page-objects-in-selenium-2-0/>
- [PageObjectEx] https://uberconf.com/blog/john_smart/2010/08/selenium_2_web_driver_the_land_where_page_objects_

2.2 Ověřte testy Testy ověřte na (a přizpůsobte ke) spuštění na vzdáleném Selenium Serveru běžící v kontejneru dostupném na localhost [ITSSH]. Monitorovat připojené webové prohlížeče můžete na webovém rozhraní Selenium Hubu [ITSSM]. Sledovat přímo prohlížeč můžete buď prostřednictvím Selenium Hubu a sledováním jednotlivého sezení (Sessions a klik na ikonu kamery), nebo přímo pomocí webové implementace VNC klienta pro Selenium Firefox [ITSVNC].

- [ITSSH] <http://localhost:4444/wd/hub>
- [ITSSM] <http://localhost:4444/ui>
- [ITSVNC] <http://localhost:7900>

3. Sepište report (volitelné).

Report není povinný. Report sepište, pokud:

- jste našli nějakou chybu (report musí obsahovat popis chyby), nebo
- pokud se implementované testy liší od navržených testů v 1. projektu.

V jiném případě report nepište.

3.1 Požadavky na report: Formát reportu je PDF (žádná titulní strana), rozsah 1-3 strany. Report obsahuje následující části:

1. hlavičku (kdo, kdy, kontext),
2. testy v původním plánu neimplementované,
3. testy implementované a nepopsané v původním plánu,
4. popis nových testů.
5. nalezené chyby v podobě stručného hlášení o chybách (bug report).

Pokud by nějaká část měla být prázdná, do reportu ji nezahrnujte.

4. Odevzdání

Odevzdejte své řešení prostřednictvím archivu xLOGIN99.zip. Archiv bude obsahovat:

```
-/ kořenový adresář archivu
+- report.pdf (nepovinný) report z testování
+- requirements.txt (nepovinný) závislosti Python balíčků
+- environment.py (nepovinný) prostředí pro Behave
+- features/
  +- *.feature soubory se scénáři
  +- steps/
    +- *.py soubory s kroky pro Python-Selenium
```

Pro spuštění testů ! na Selenium Serveru! musí být úspěšně provedeny tyto kroky:

```
$ unzip xLOGIN99.zip
$ [ -f requirements.txt ] && pip3 install -r requirements.txt
$ docker-compose up -d
$ behave
```