International Journal of Applied Engineering Research ISSN 0973-4562 Volume 13, Number 12 (2018) pp. 10316-10326
© Research India Publications. http://www.ripublication.com

An Algorithm for Semantic based Automated Functional Testing on Web

Deepali Sharma*, Sanjay Kumar Malik**

**M.Tech. IT, USIC&T, Guru Gobind Singh Indraprastha University, Sector-16C, Dwarka, New Delhi-110078, India.*
***Associate Professor, USIC&T, Guru Gobind Singh Indraprastha University, Sector-16C, Dwarka, New Delhi-110078, India.*

Abstract

Functional testing plays a significant role in improving quality of any software as well as its maintenance. Since web plays a vital role in modern times, therefore, there is a need of exploring as well as proposing various approaches or algorithms of functional testing on web, incorporating machine intelligence which is the emerging need of the information technology (IT) industry. The present web is approaching towards an intelligent web known as semantic web where ontology is the key concern for embedding semantic based concepts towards machine-understandable web. One of the most significant web services are semantic web services which are the web services that are semantically commented, keeping in mind the end goal to influence the services to become machine understandable, consequently permitting service discovery, and invocation to be done automatically or with least human efforts. Testing of these semantic web services may be used as a prime measure for quality assurance.

The objective of this paper is to explore, analyze, discuss and summarize the state of ontological functional testing of semantic web services towards a machine-understandable approach on web. In this paper, first, semantic web services have been explored along with its testing stages and classification of testing in semantic web services. Software testing and functional testing in semantic web services have also been presented in this section. Second, literature survey has been provided in the direction of semantic web services and ontology concerns. Third, automatic ontology aided functional testing has been primarily focused in context to semantic web services, where a case study has been taken for E-commerce website and an algorithm has been proposed along with a flowchart. Here, ‘GoodRelations ontology’ for annotation of different aspects of E-commerce on the web is outlined and an attempt has been made to explore the concerned research aspects. Fourth, pseudo code has been presented along with the automated code for ontology based automated functional testing for an E-commerce website using the “PyCharm” tool. Also, the results and the inferences of the proposed methodology have been focused.

Keywords: Functional testing, E-commerce, Semantic Web Services, Software testing, GoodRelations Ontology, Knowledge representation

INTRODUCTION

Machine to machine communications are gaining popularity these days. A web service is characterized as a system framework intended to support machine-to-machine communication over a network for the exchange of data over web. It also has an interface explained in a machine-processable format by World Wide Web Consortium (W3C[1], 2004). The current web includes various types of information such as structured, unstructured and semi-structured information in HTML (Hypertext Markup Language) based web pages which are searched through keyword queries on the web. However, these technological standards have limited search capabilities as the searches can’t determine the type of information on a selected page.

The next generation web, the semantic web, whose main objective is the exchange of data, i.e. the data which is defined, linked and represented with the standards of semantic web may be used by both machines and humans for further exploration along with establishing the implicit relations. Towards the fulfillment of this goal, web services and semantic web technologies are integrated in Semantic Web Services (SWS). Semantic web services are self-describing and flexible to automatic discovery, composition and invocation rather than the conventional web services. It operates at the server side of a client– server framework for machine-to-machine interoperability on web. These services are the trading of semantic information from various sources and services without losing their meaning [1]. The SWS add the semantics to the present web services which transforms the data to be machine-readable and reusable. The semantic layer is the most varying between the semantic web services and the conventional web services. Testing has to be accomplished over the semantic layer while testing the traditional web services.

As testing of any software plays a vital role in the improvement of quality of the software, it is an integral part of software development process. IEEE[2] [2] characterizes testing as the way towards working on a system under specific conditions, recording the outcomes, and making an assessment of some part of the system or framework.

The paper is organized as follows- Section 1, explores the semantic web services, various stages in testing of SWS and its procedure for classification of testing. Software testing and functional testing in semantic web service have also been presented. Section 2, discusses literature survey which includes the various significant aspects like semantic web services, ontology and test case specification. In Section 3, a case study has been taken where a methodology has been proposed with the help of an automated ontological functional testing algorithm and flowchart. GoodRelations ontology for ‘Pen drive’ as a product has also been presented in this

[1] https://www.w3.org/

[2] https://www.ieee.org/

section. In Section 4, pseudo code has been presented along with the automated code for ontology based automated functional testing for an E-commerce website using the “PyCharm” tool. E-commerce is the process of buying and selling the products and services through the internet in the form of a business. The results and inferences of the implemented code have also been presented in this section. Finally, the conclusion and future scope has been discussed in Section 5.

Semantic Web Services (SWS)

SWS are an enhancement of the internet services description through the development of explanations on the web, made in order to encourage the mechanization of the service communications [3]. Web services turned out to be all the more attractive as a fresh out of the new worldview for building programming framework. However, semantic web services assume a critical part inside the setting of a significant and smart web. In the SWS paradigm, information moves towards becoming machine-reasonable and understandable. SWS can work together in preparing the information without losing their importance.

GoodRelations is one of the most powerful ontology for product, price, store, and company data for publishing all of the details in an exceedingly friendly way to search engines, mobile applications, and browser extensions. By adding a little of additional code to the website used in this work, it was made sure that potential customers understand all significant features and services and therefore the advantages of doing business (*http://www.heppnetz.de/ontologies/goodrelations/v1.html*). The semantic web services give a standard system that allows the data to be shared and reused across application and gather limits. One may consider SWS as a strategy to address information on the web, or as a database that is all inclusive joined, in an exceedingly way understandable by machines.

SWS Testing Stages

Semantic web service testing consists of a minimum of three stages, as presented in Figure 1, which are development testing, repository testing and end-client testing [4].

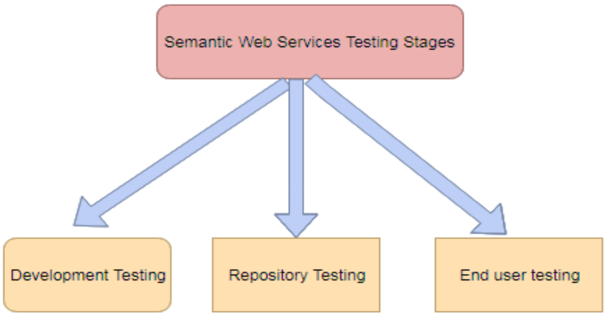


Figure 1: Stages in testing of Semantic Web services [4]

The three stages in testing of SWS are as follows [4]:

1) Development testing - performed by designer of the web services.
2) Repository testing - performed by an operator in the interest of the catalog where the administrations will be open from.
3) End client testing - conducted by the client of the web services to affirm that the services still works legitimately.

It should be recorded that in the process of testing, the test case creation and test case execution are two completely variant steps. Although, it's a well proven fact that no matter which test case selection methodology is employed, few code bugs might still escape detection [5].

Procedure for Classification of Semantic Web Services Testing

SWS testing procedure may be classified as follows [6]:

1. Test case Generation

Test cases are a collection of inputs and also the expected results created to utilize a specific program path or to verify the consistency to a specification. Manually making test cases will be tedious task in addition to time overwhelming. Furthermore, creating test cases manually doesn't support the SWS objective of facilitating the automation of internet services usage [6].

2. Mutation Testing

Mutation testing is a process where two or more mutant programs may be executed at the same time with a similar test case according to the flexibility of the tests to observe the mutants of the test [6, 20].

3. Test Selection and Execution

Although, functional testing is critical to confirm the quality of a software system, executing all possible test cases will be high-ticket as it takes up valuable time, machine and tester resources. Software risk assessment identifies crucial elements of the system which have a high failure probability rate or which causes serious consequences due to its failure [2].

Software Testing

Software testing is defined as the process of judgment of a software item to notice the variations between the provided input and the expected output. Testing assesses the standard of the product. Software testing may be a method that has to be done throughout the development process. Testing is very important to satisfy the right performance of service-oriented systems that have the ability to choose the services dynamically and use them. In order to verify the right functioning of a service-oriented system, ability among all its elements and consolidation of those elements should be adequately tested [2].

Functional Testing in SWS

Functional Testing is a type of software testing where the functions (features) are tested against the requirements provided. It is performed by providing the input and expecting the output. It is of many types such as unit testing, integration testing, sanity testing, regression testing, system testing and user acceptance testing [7]. "It" is a method to verify that all the functions are meeting their goals as specified in functional requirements.

Functional testing is a quality assurance technique which is often used, since functionality is the primary concern of any software system and hence, SWS are no exception for web. Functional testing in SWS is concerned with testing the functionalities that are uncovered by the SWS based upon the functional requirements. Semantic web services are self-describing and amenable to automatic discovery, composition and invocation. In most cases, the testers and the customers are not developers [8]. The testers have the access to the web services and test the functionality based on the specified interface without having knowledge of system's internal working. Hence, a web service is tested as black box testing.

LITERATURE SURVEY

Tahir et al. [2] discuss reasonableness which focuses on programming framework and semantic web administrations. To boot, functional testing is a most part utilized quality confirmation procedure. Functional testing is bothered with testing the functionalities which are uncovered by the semantic web service based on its useful descriptions. The shoppers and analyzers of the web services aren't their developers in most cases. In this manner, the analyzers in some cases approach the web service exclusively through its negligible and announced interface without knowing the inward working of the service [2].

They also discussed the two important variations between semantic web services and traditional web services that could affect the move to be followed once testing the Semantic web services:

- The existence of semantic layer: Semantics of conventional web services aren't insignificant. The analyzer of conventional web services needs to make her/his own particular explanation of the accompanying:
    - o The preconditions have to be satisfied prior to entering the services.
    - o The state changes coming from the execution of the services. On account of semantic web services the upper semantic data is pre-indicated by ontology languages. The desired semantic data is machine-understandable and perceivable.
- Heterogeneity of standards [2]: They are totally dissimilar activities and diverge non-compatible ontology languages for semantic web services that display totally extraordinary levels of customs.

Heckel and Mariani [9] outlined the test case specification. A test case specification comprises of three sections: the precondition, the test sequence, and the expected outcome [9]. The precondition indicates requirements that are anticipated that would hold for the condition of the server when the experiment is executed. Conditions on parameters are contained in the test sequence which indicates conditions on input parameters together with the request of service invocations. The expected outcome is acquired by executing the rule for the produced input esteems.

In GoodRelations ontology by Hepp [10] for an E-commerce case study, which has primarily focused on Product or Service: Instance, Model, and Class: within the merchandise and services domain, which discover multiple sorts of abstract entities once it involves describing what's being offered.

Villagra and Berrocal [11] discusses about the advantages of ontology which are as follows:

- Reuse: The ontology can be reused in the domain.
- Explicit: Since the ontology defines the concepts, functions, relationships, properties, constraints that compose it, hence, it is explicit.
- Interoperable: It can support communication between the systems that are developed at different sites.
- Knowledge level: The knowledge level verification and validation [11].

PROPOSED METHODOLOGY: A CASE STUDY INCORPORATING E-COMMERCE DOMAIN

In this work, the target is on automated functional based SWS testing for an E-commerce website. Specifically, functional testing to derive the test cases for SWS from whole completely different specifications has been applied. In particular, the accommodating testing to determine test cases for SWS from entire unique details has been applied. Since the amount of experiments does not basically add worth to testing, hence, it is essential to experience their quality. In addition, to construct the approach achievable and smart, a tool to change all the investigation steps is utilized. At last, to deal with the well-amazing E-Commerce Service, a web service that uncovers item information and E-Commerce utility has been connected. E-commerce is the process of purchasing and selling different products or services through internet in the form of a business [12]. The present work includes usage for Automation test generation that allows the information to be reusable and machine-readable.

In this work, a service-oriented test data creation approach has been introduced that addresses the matter of productiveness in machine-controlled generation of realistic test information. During this approach, the requirement for previous information and manual tester inputs are decreased by investment the information which will be acquired from compositions of the many existing net services. An ontological functional testing algorithm has been proposed in this work for an E-commerce website. Also, the

GoodRelations vocabulary that is W3C schema for the considered E-commerce case study example has been taken where the searched item array is the 'Product' class and is further linked with another class 'Brand'.

The benefits of the planned approach [13] are stated in Table 1:

Table 1: Benefits of the planned approach

Benefits	Description
Automated	Upgraded effectiveness in machine-driven test information generation for the information varieties that can't be effectively created.
Tailored	Test generation supported the tester criteria and optimized test data generation supported the data supply.
Applicability	Ability to come up with test information to check any system with linguistics information.
Minimized	Stripped dependence on the prevailing information sources like information bases and session data.

Proposed Ontological Functional Testing Algorithm

The proposed algorithm is followed by GoodRelations Ontology which is an E-commerce ontology that supports exchange of e-commerce information such as product name, price, offers, terms and conditions, specifications on the web. It handles the individual relationships between a customer, a merchant and the products and services offered (*http://semanticweb.org/wiki/GoodRelations.html*).

The following algorithm has been proposed in this work:

Step 1: INITIALIZE item array 'items[i]' with individuals of Product class present in GoodRelations Ontology. Remove prefix in URI of each individual.

{"Bhagvad Geeta", "Dell Vostro 2660", "Pendrive 128 GB", "Hard Disk"} ∈ Product

items[i] = ["Pendrive 128 GB", "Hard Disk", "Bhagvad Geeta", "Dell Vostro 2660"]

Step 2: REPEAT steps 3 to 5 for each item in items array.

Step 3: Send values of array elements in the Search box using "send_keys" method of webdriver.

Step 4: Invoke check procedure which will validate results of the searched query with queried item.

> **Step4.1:** For each instance in the result set, repeat step 4.2.
>
> **Step4.2:** If instance matches queried item specifications then the test case is Successful/Passed else it will fail.

Step 5: Invoke the clear procedure which will clear the search for that item.

(Step 4.1 and Step 4.2 are not implemented in this work and will be a part of future work.)

Proposed flowchart for Automated Ontological functional testing of an E-commerce in E-commerce domain

The flow chart for the automation functional testing of an E-commerce website has been presented in figure 2.

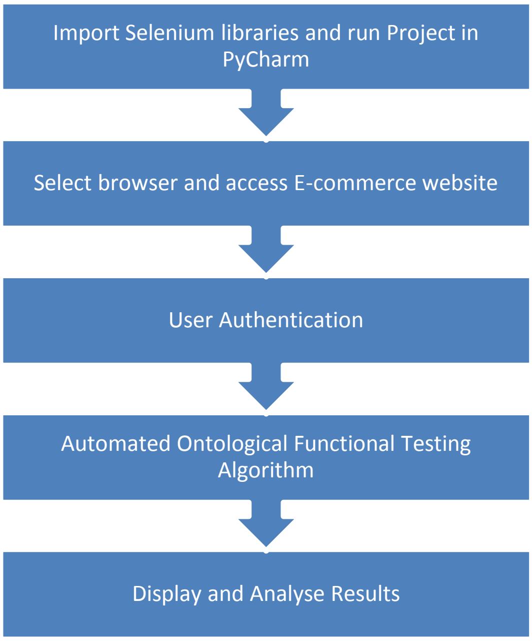

Import Selenium libraries and run Project in PyCharm
Select browser and access E-commerce website
User Authentication
Automated Ontological Functional Testing Algorithm
Display and Analyse Results

Figure 2: Flow chart for automated ontological functional testing of an E-commerce website

Functions are tested by providing them input and expecting the output. In this work, the functional testing has been implemented on an E-commerce website where user searches for any particular item and the expected output is the result appeared for the searched item. Also, the GoodRelations ontology which is W3C schema has been imported and applied.

The flowchart for automated ontological functional testing has been elaborated below:

1. In Module 1, import selenium libraries which are in-built libraries such as "webdriver" and "time" have been imported which provides all the webdriver implementations and time specifications respectively. It also states run project in PyCharm, PyCharm is a tool that is used specifically for Python language. It is an Integrated Development Environment (IDE), and it is used across platforms like Windows, Linux, and Mac etc.
2. In Module 2, the browser selection is made where the instance of Chrome Webdriver is created with the help of selenium webdriver library. This is followed by accessing website with browser.get navigates to the page given by the URL (Flipkart[3]).

[3] https://www.flipkart.com/

3. Then, Module 3 describes user authentication where the keys are being sent for username and password and it is similar to entering the keys using keyboard.
4. Module 4 performs ontological functional testing where the queried item has been searched in the result set such as the item (Pendrive 128 GB) is queried in the search box of the accessed website. It then returns the result, if the queried item is present in the result set than the test is passed/successful else the test is failed/unsuccessful. Then this module applies GoodRelations ontology which is ontology for e-commerce for the queried item. The queried item is the product or service class, the brand name is associated with the brand class, the product make or model is sub-class of the brand. Lastly, after the results are matched in the result set, the search box is cleared.
5. Finally, the testing results are displayed on the console which can be used to check the status of the functional testing for the site considered.

In Figure 3, the 'GoodRelations Ontology' has been focused as a case study. It focuses on "Pendrive" as a product or services class which is linked to another class 'Brand' which is further linked to sub-class 'make or model' that specifies the model of that particular brand.

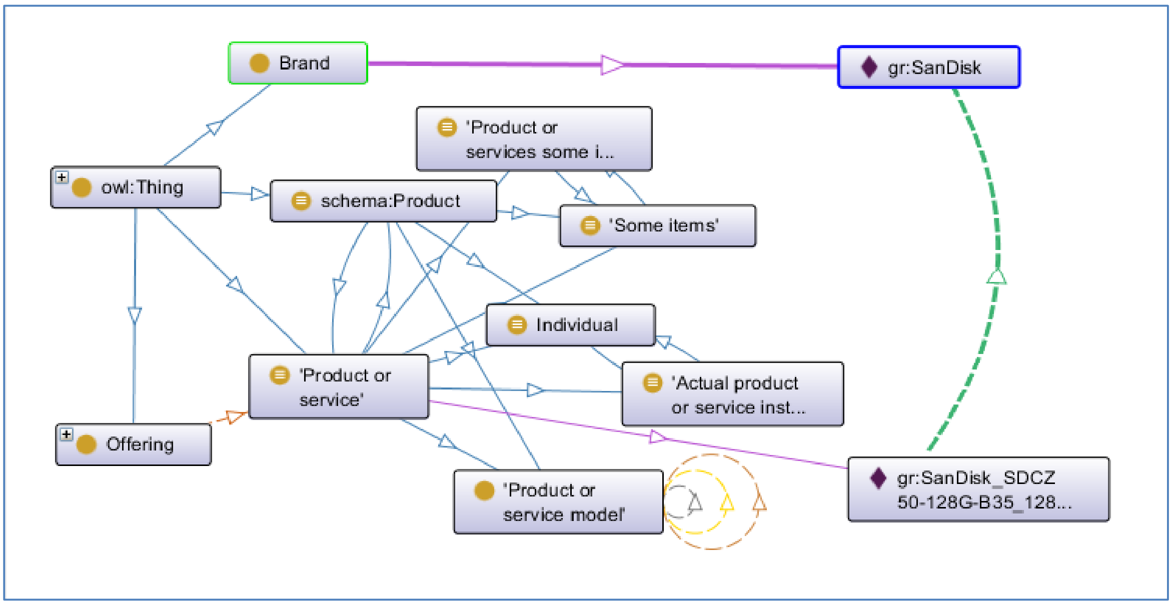

Brand
gr:SanDisk
'Product or services some i...
owl:Thing
schema:Product
'Some items'
Individual
'Product or service'
'Actual product or service inst...
Offering
'Product or service model'
gr:SanDisk_SDCZ 50-128G-B35_128...

Figure 3: GoodRelations Ontology for "Pen drive as a Product"

IMPLEMENTATION AND RESULTS

The implementation and results of the methodology in the form of a proposed algorithm and the case study has been presented which includes the automated code of the ontology based functional testing using the 'PyCharm' tool in selenium and python language. Also, the results of the executable automation code and 'GoodRelations ontology' along with its inferences have been presented.

Extending E-commerce Ontology (GoodRelations Ontology):

First, select actual products, like for instance, Pendrive or hard disk. Second, certain product makes and models, for example, the Pendrive make and model SanDisk brand or the hard disk create Seagate brand. Sometimes there exist actual products that are of several make and model, however, all of them have an identity of their own. Particularly, they take issue in many properties. Third, classes of actual products that are similar in perform, like for instance the category "Pendrive" for product and services that subsumes all actual pen drives.

Pseudo Code

```
def search_item():
    items[i] = ["Pendrive 128 GB", "Hard Disk", "Bhagvad Geeta", "Dell Vostro 2660"]
    Input = send_keys.items[i];
    Output_list = ['Pendrive 6GB', 'Pendrive SanDisk 128GB', 'Dell Vostro 15 3000 Series', 'Bhagvad Geeta', 'Dell Vostro Laptop 8GB RAM', 'Laptop 16GB RAM', 'Hard Disk']
    for(i=0; i<=3; i++)
    {
    if items[i] in output_list:
        return items;
    else:
        send_keys.clear()
        return i++
    }
```

Ontological Automation Functional Testing Coding

Selenium is a popular and portable open-source software testing framework for automating the web application. It is used across different platforms and browsers. Selenium provides a domain-specific language to write the code or the tests in famous programming languages, such as Java, Python, C++, PHP etc.

PyCharm is a tool that is used specifically for Python language *(https://www.jetbrains.com/pycharm/)*. It is an Integrated Development Environment (IDE), and it is used across platforms like Windows, Linux, and Mac etc.

Using the tool PyCharm and implementing the code in Selenium and Python language, the below results have been obtained.

In Figure 4, the snapshot of the PyCharm tool has been displayed where a python file 'E-commerce' has been created. The file extension for a python file is .py.

Figure 5 shows the automated test script for E-commerce website 'Flipkart' which is written in Python and Selenium. It also shows the functional test case creation where search for 'Pendrive' or 'hard disk' item has been made.

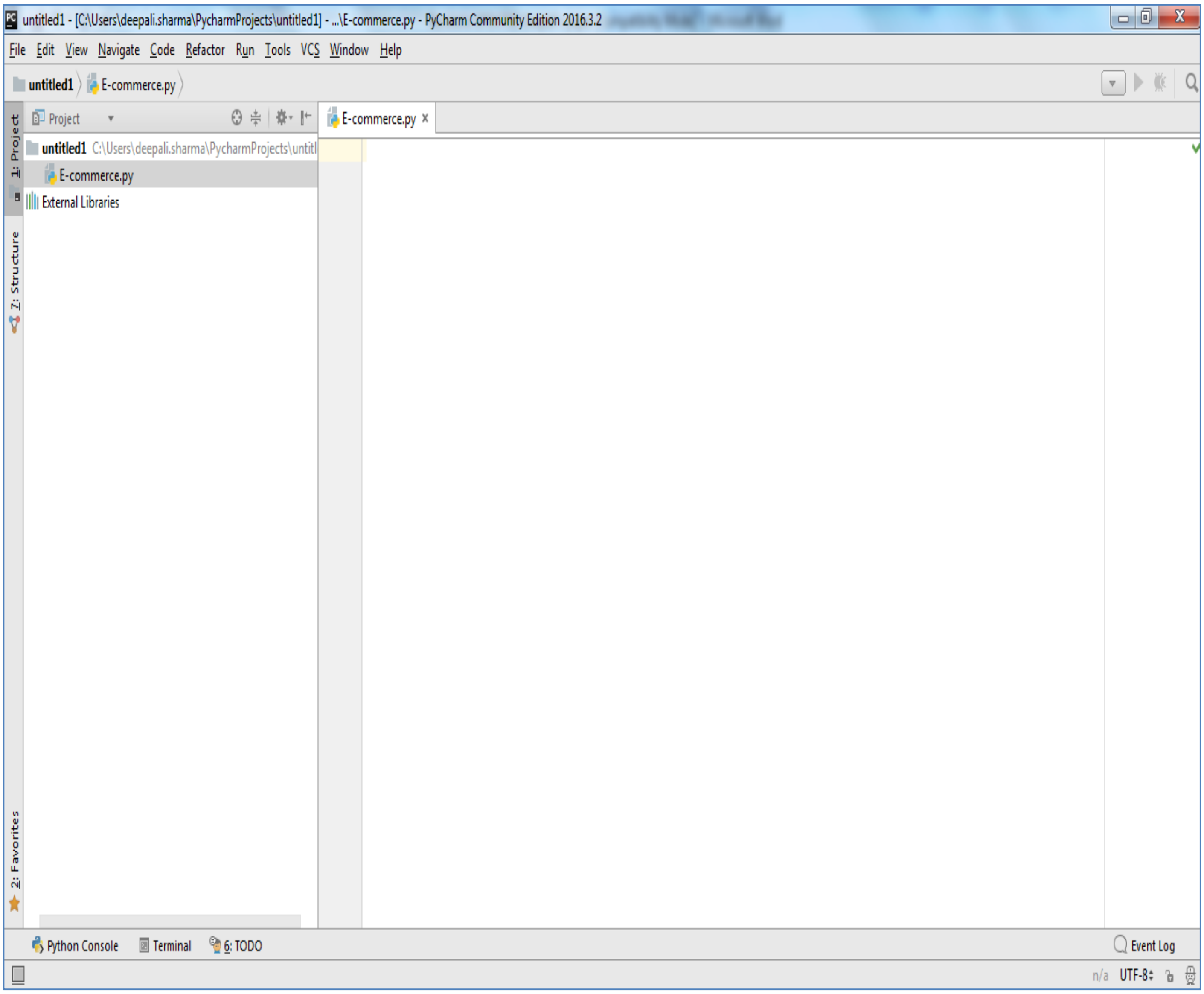
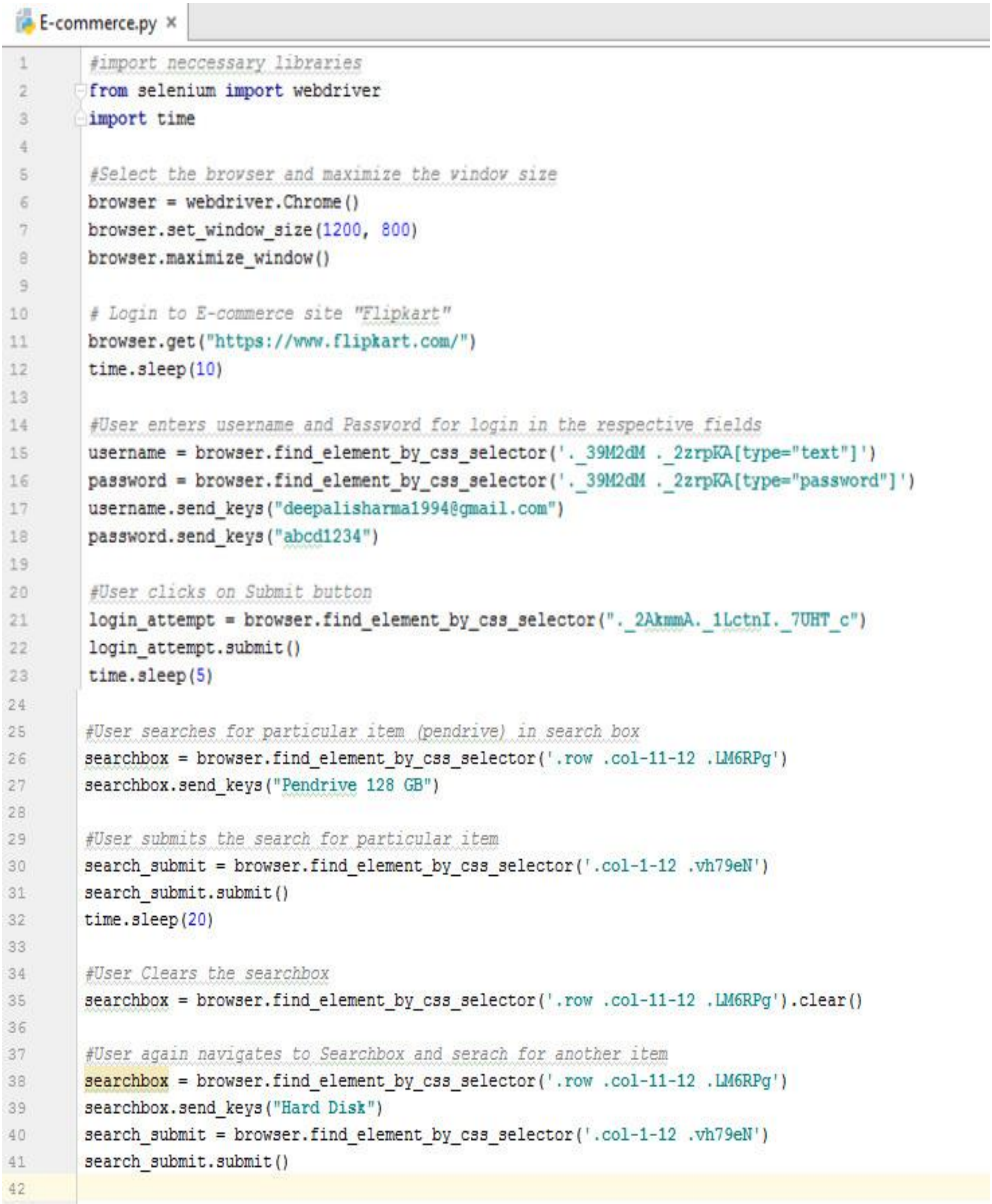

untitled1 - [C:\Users\deepali.sharma\PycharmProjects\untitled1] - ...\E-commerce.py - PyCharm Community Edition 2016.3.2
File Edit View Navigate Code Refactor Run Tools VCS Window Help
untitled1 E-commerce.py
Project
untitled1 C:\Users\deepali.sharma\PycharmProjects\untitl
E-commerce.py
External Libraries
E-commerce.py
1: Project
7: Structure
2: Favorites
Python Console
Terminal
6: TODO
Event Log
n/a UTF-8

Figure 4: Snapshot of "PyCharm" Tool

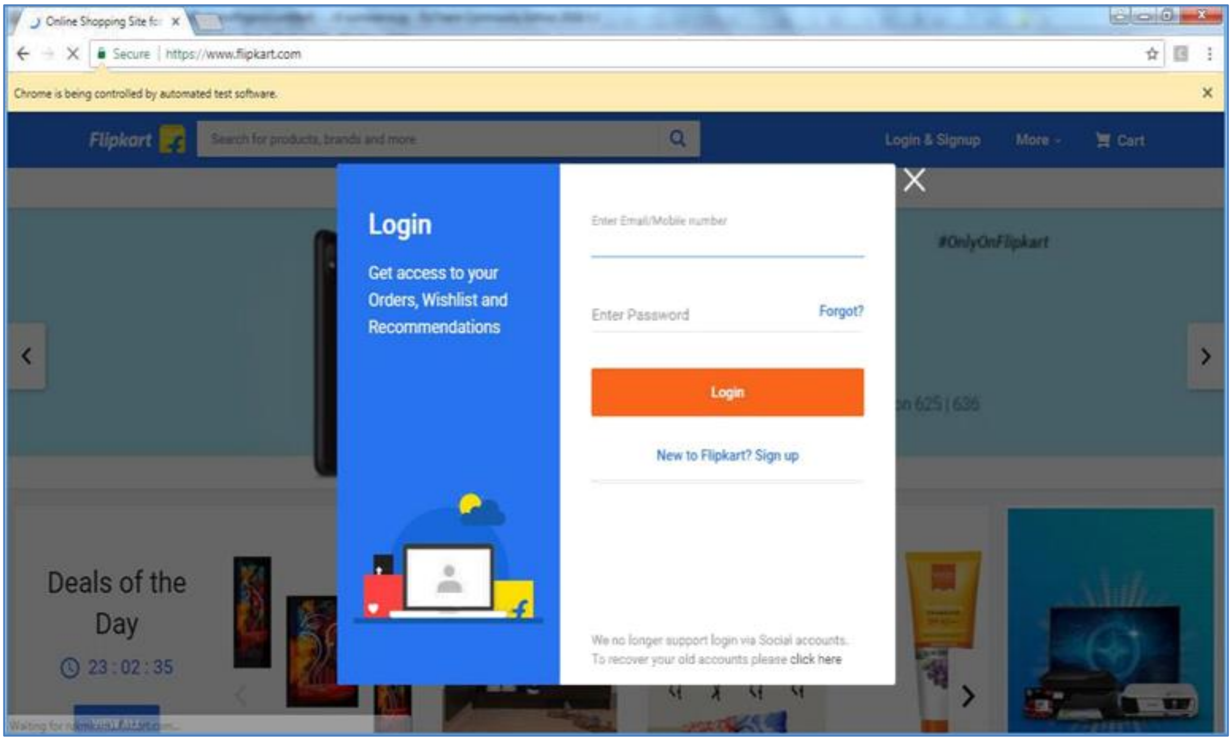


```python
#import neccessary libraries
from selenium import webdriver
import time

#Select the browser and maximize the window size
browser = webdriver.Chrome()
browser.set_window_size(1200, 800)
browser.maximize_window()

# Login to E-commerce site "Flipkart"
browser.get("https://www.flipkart.com/")
time.sleep(10)

#User enters username and Password for login in the respective fields
username = browser.find_element_by_css_selector('._39M2dM ._2zrpKA[type="text"]')
password = browser.find_element_by_css_selector('._39M2dM ._2zrpKA[type="password"]')
username.send_keys("deepalisharma1994@gmail.com")
password.send_keys("abcd1234")

#User clicks on Submit button
login_attempt = browser.find_element_by_css_selector("._2AkmmA._1LctnI._7UHT_c")
login_attempt.submit()
time.sleep(5)

#User searches for particular item (pendrive) in search box
searchbox = browser.find_element_by_css_selector('.row .col-11-12 .LM6RPg')
searchbox.send_keys("Pendrive 128 GB")

#User submits the search for particular item
search_submit = browser.find_element_by_css_selector('.col-1-12 .vh79eN')
search_submit.submit()
time.sleep(20)

#User Clears the searchbox
searchbox = browser.find_element_by_css_selector('.row .col-11-12 .LM6RPg').clear()

#User again navigates to Searchbox and serach for another item
searchbox = browser.find_element_by_css_selector('.row .col-11-12 .LM6RPg')
searchbox.send_keys("Hard Disk")
search_submit = browser.find_element_by_css_selector('.col-1-12 .vh79eN')
search_submit.submit()
```

Figure 5: Snippet of automated test case for E-commerce Website

After the script has been automated and run, the Flipkart website has been navigated by the code with a login pop-up as indicated in Figure 6.

In Figure 7, the username and password are getting entered itself in the Login Popup that was shown in Figure 5 with the help of the automated code.

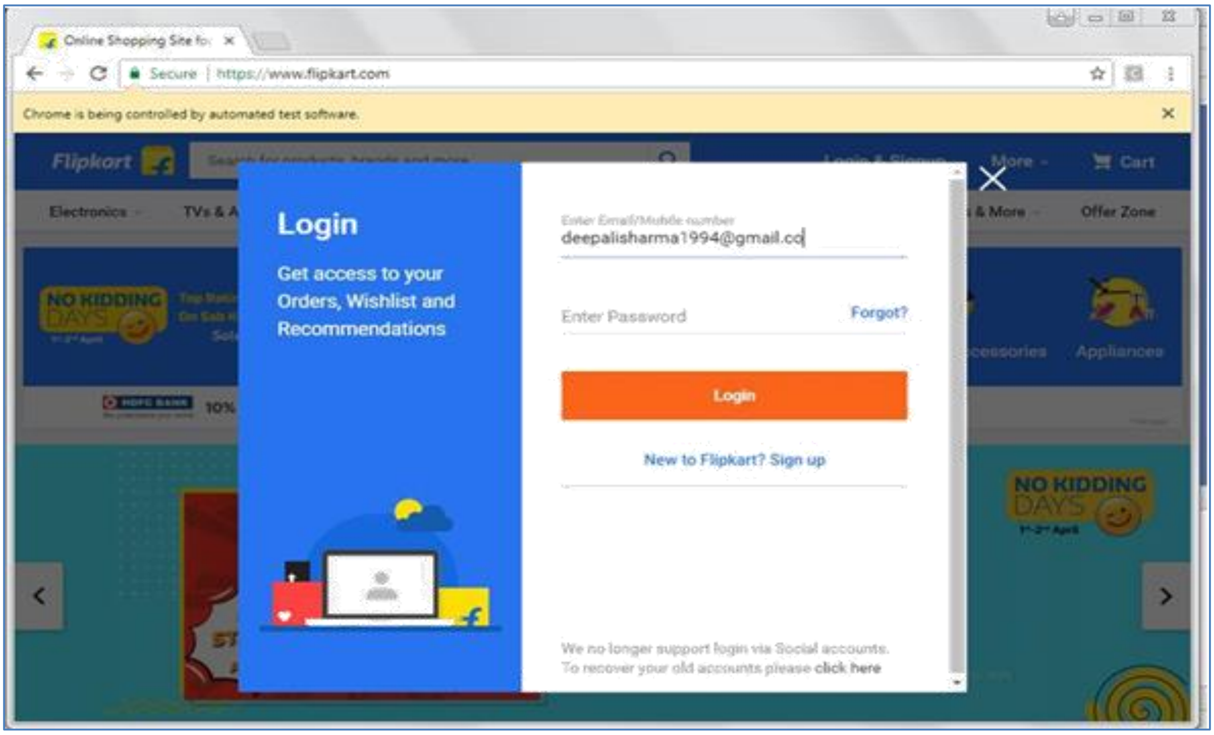


https://www.flipkart.com

Chrome is being controlled by automated test software.

We no longer support login via Social accounts. To recover your old accounts please click here

Figure 6: The URL gets hit by the automated script with Login Page



https://www.flipkart.com

Chrome is being controlled by automated test software.

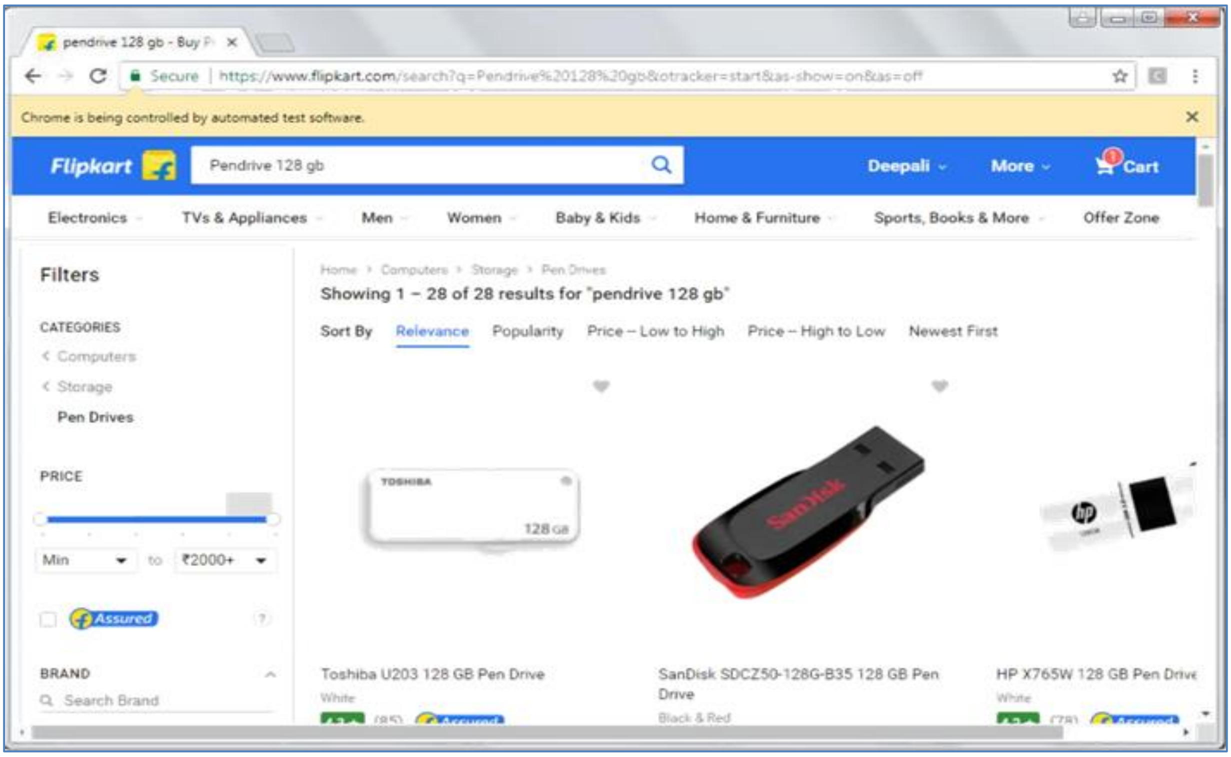
Enter Email/Mobile number: deepalisharma1994@gmail.co

We no longer support login via Social accounts. To recover your old accounts please click here

Figure 7: The user's credentials are getting entered

In Figure 8, the functional approach has been made which searches for ‘Pendrive 128 GB’ by inputting the text or sending the keys in the Search box and the output has been displayed with all the Pendrives with the storage of 128 GB.

In Figure 9, another similar functional approach has been made with search for another item “Hard Disk”. It first clears the search box text “Pen drive 128 GB” and then sends the keys for “Hard Disk” for which results are shown.



pendrive 128 gb - Buy P

https://www.flipkart.com/search?q=Pendrive%20128%20gb&otracker=start&as-show=on&as=off

Chrome is being controlled by automated test software.

Pendrive 128 gb

Home › Computers › Storage › Pen Drives

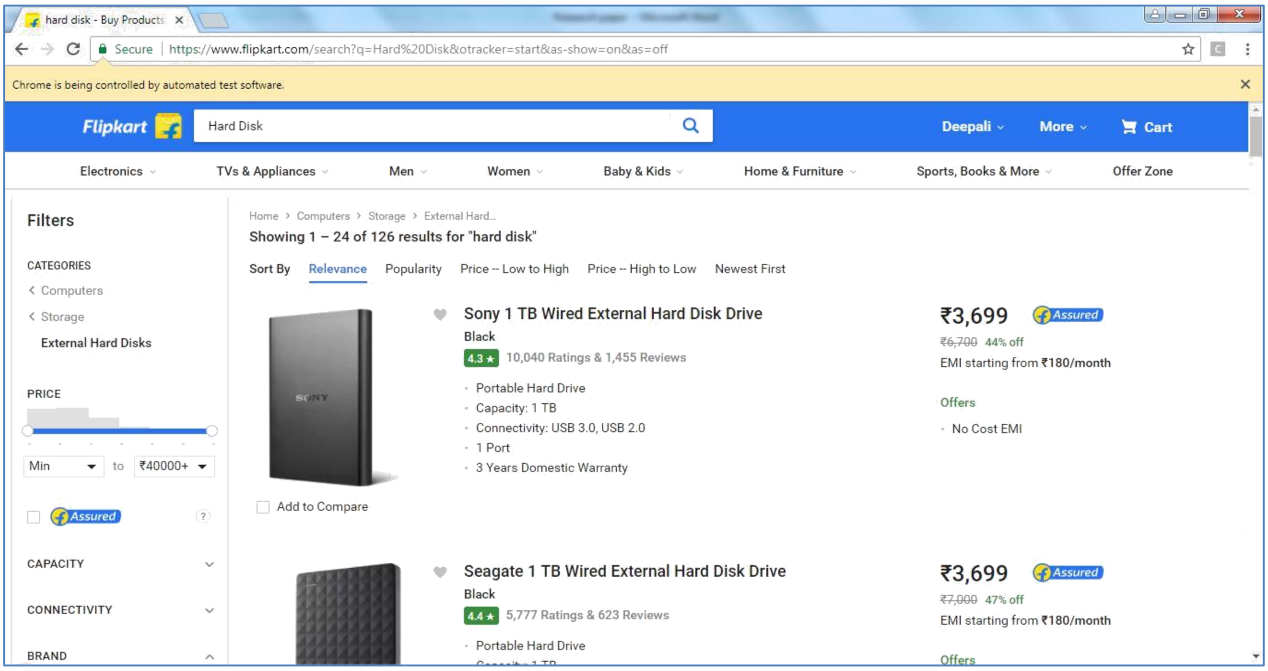
Showing 1 – 28 of 28 results for "pendrive 128 gb"

Toshiba U203 128 GB Pen Drive
White

SanDisk SDCZ50-128G-B35 128 GB Pen Drive
Black & Red

HP X765W 128 GB Pen Drive
White

Figure 8: ‘Pen drive 128 GB’ as Search Result



hard disk - Buy Products

https://www.flipkart.com/search?q=Hard%20Disk&otracker=start&as-show=on&as=off

Chrome is being controlled by automated test software.

Hard Disk

Home › Computers › Storage › External Hard...

Showing 1 – 24 of 126 results for "hard disk"

Sony 1 TB Wired External Hard Disk Drive
Black
4.3 ★ 10,040 Ratings & 1,455 Reviews
- Portable Hard Drive
- Capacity: 1 TB
- Connectivity: USB 3.0, USB 2.0
- 1 Port
- 3 Years Domestic Warranty

₹3,699 Assured
~~₹6,700~~ 44% off
EMI starting from ₹180/month
Offers
- No Cost EMI

Seagate 1 TB Wired External Hard Disk Drive
Black
4.4 ★ 5,777 Ratings & 623 Reviews
- Portable Hard Drive

₹3,699 Assured
~~₹7,000~~ 47% off
EMI starting from ₹180/month
Offers

Figure 9: ‘Hard Disk’ as a search Result

Inferences

1. The knowledge representation is done based on ontology in e-commerce domain and can also be extended to other domains having products and services representations.
2. The proposed methodology can be used to check consistency of the product inventory of e-commerce sites (Flipkart, Amazon, Snapdeal etc.) and can also be extended to other domains having products and services representations.
3. The proposed algorithm supports automation, which is currently in demand in IT industry.

CONCLUSION AND FUTURE SCOPE

In this paper, a review on semantic based automation functional testing on web, with the aim of understanding how testing of SWS differs from testing normal web services has been provided [6]. The most prominent approaches of testing SWS are presented and classified into various categories such as mutation testing, test case creation and test selection, yet it is not possible to claim that the list is exhaustive. The software testing and the functional testing in semantic web services have been discussed. The flowchart and an algorithm has been proposed for the case study – ontological automation functional testing of an E-commerce website. A programmed approach to get execution tests for semantic web services that sketch out the Inputs, Outputs, Preconditions, Effects (IOPEs) worldview has been proposed [4]. This approach acknowledges these testing objectives to get a connection of the online services as a test suite. Other element of the approach is production of confirmation links. Likewise, this method grants production of attainable and executable experiments which might be connected to various interfaces through which the web service is additionally gotten to. The testing tasks need expertise and knowledge of human tester so as to perform the task. There is a belief that utilization of semantics will minimize human efforts in performing testing tasks [1], however, most of the approaches are still in their early stages with several of them solely coverage results of early findings of the test service.

The ‘GoodRelations ontology’ for an E-commerce has been primarily focused for the case study which is performed for the queried item ‘Pendrive’ as a product. A lot of work has to be done to reinforce analysis during this promising area of SWS testing.

Flipkart like E-commerce websites may incorporate semantics in their services using the standard ontologies such as W3C recommended GoodRelations Ontology. The shortcomings of this work are that it is in reference to the tendency to only crawl the merchandise connected pages from the pre-defined sites, and that the annotation theme is simply for specific sites, i.e., it extremely depends on the built of the agreement or report [14]. Thus, future work includes the creation of standard ontologies for various product areas such as GoodRelations ontology and to add the mechanics to the information thus reducing the irrelevant information. Also, the instances for the automated code result set are passed or failed will be a part of the future work.

REFERENCES

[1] C. Pedrinaci, J. Domingue, A.P. Sheth, 2011, “Semantic Web Services.” In: Domingue J., Fensel D., Hendler J.A. (eds) Handbook of Semantic Web Technologies. Springer, Berlin, Heidelberg

[2] A. Tahir, D. Tosi, S. Morase, 2013, “A systematic review on the functional testing of semantic web services” Proceedings of the Journal of Systematic and Software. 2013 Nov; 86(11):2877–89.

[3] M. Vargas-Vera, M. Nagy, 2009, “Challenges on Sematic Web Services”, Proc. IGI Global

[4] A. M. Paradkar, A. Sinha, C. Williams, R.D. Johnson, S. Outterson, C. Shriver, and C. Liang, 2007, "Automated Functional Conformance Test Generation for Semantic Web Services," Proc. IEEE International Conference on Web Services (ICWS), IEEE Computer Society, 9-13 July2007, pp. 110-117, doi:10.1109/ICWS.2007.48

[5] M. S. Jokhio, 2009, "Goal-Based Testing of Semantic Web Services," Proc. 24th IEEE/ACM International Conference on Automated Software Engineering (ASE), IEEE Computer Society, 16-20 November 2009, pp. 707-711

[6] H.M. Rusli, S. Ibrahim, M. Puteh, N. Mahrin, 2012, “The Role of Semantics in Testing Semantic Web Services” in International Conference on Semantic Web and Web Services

[7] I. Hooda, R. Chhillar, 2015, "Software Test Process Testing Types and Techniques", International Journal of Computer Applications, vol. 111, no. 13, pp. 10-14

[8] N. Anari, G. Ahakian, H. Saboohi, 2016, "A Comparative Study of Testing Semantic Web Services" in preceedings of Indian Journal of Science and Technology, 9. 10.17485/ijst/2016/v9i48/109316

[9] R. Heckel, L. Mariani, 2005, "Automatic conformance testing of web services", Proc. Fundamental Approaches to Software Engineering (FASE 05), pp. 34-48.

[10] M. Hepp, 2008, “ GoodRelations: An ontology for describing products and services offers on the web” In: Gangemi, A., Euzenat, J. (eds.) EKAW 2008. LNCS (LNAI), vol. 5268, pp. 329–346. Springer, Heidelberg (2008)

[11] J. E. L. Vergara de, V. A. Villagrá and J. Berrocal, 2002, “Semantic management: advantages of using an ontology-based management information meta-model” Proceedings of the 9th Workshop of the HP Open View University Association (HP-OVUA)

[12] T. Ordysiski, 2011, “Ontology of E-Commerce Solution”, Studies & Proceedings of Polish Association for Knowledge Management.

[13] M. Bozkurt, M. Harman, 2011, “Automatically Generating Realistic Test Input From Web Services” in Service Oriented System Engineering (SOSE), 2011 IEEE 6th International Symposium, pp. 13-24.

[14] L. Zhang, M. Zhu, W. Huang, 2009, “A Framework for an Ontology-based E-commerce Product Information Retrieval System”, JOURNAL OF COMPUTERS, VOL. 4, NO. 6, JUNE 2009

[15] X. Bai, and R. S. Kenett, 2009, "Risk-Based Adaptive Group Testing of Semantic Web Services," Proc. 33rd Annual IEEE International Computer Software and Applications Conference (COMPSAC'09), IEEE Computer Society, 20-24July 2009, pp. 485-490, doi:10.1109/COMPSAC.2009.180

[16] X. Bai, S. Lee, W.-T. Tsai, Y. Chen, 2008, “Ontology-based test modeling and partition testing of web services” in: Proceedings of the IEEE International Conference on Web Services., IEEE Computer Society, Washington, DC, USA, pp.465–472.

[17] T.-D. Cao, P. Felix, and R. Castanet, 2010, "WSOTF: An Automatic Testing Tool for Web Services Composition," Proc. Fifth International Conference on Internet and Web Applications and Services (ICIW), IEEE Computer Society, 9-15 May 2010, pp. 7-12, doi:10.1109/ICIW.2010.9

[18] G. Dai, X. Bai, Y. Wang, and F. Dai, 2007, “Contract-Based Testing for Web Services,” in Proceedings of the 31st Annual International Computer Software and Applications Conference. Washington, DC, USA: IEEE Computer Society, 2007, pp. 517–526.

[19] M. S. Jokhio, G. Dobbie, and J. Sun, 2009, "Towards Specification Based Testing for Semantic Web Services, "Proc. Australian Software Engineering Conference (ASWEC), IEEE Computer Society, 14-17 April 2009, pp.54-63, doi:10.1109/ASWEC.2009.38

[20] M. S. Jokhio, G. Dobbie, T. Hu and J. Sun, 2014, "Automated Mutation-Based Test Case Evaluation for Semantic Web Services," 2014 23rd Australian Software Engineering Conference, Milsons Point, NSW, pp. 210-219.

[21] M. Shaban , G. Dobbie, J. Sun, 2009, “A framework for testing semantic web services using model checking” Proceedings of the Fourth South East European Workshop on Formal Methods, SEEFM’09, IEEE Computer Society; Washington DC, USA. 2009. p. 17–24.