AN AUTOMATED WEB SCRAPING TOOL

FOR MALAYSIA TOURISM

By

CHOONG WEI JEN

A REPORT

SUBMITTED TO

Universiti Tunku Abdul Rahman

in partial fulfilment of the requirements

for the degree of

BACHELOR OF COMPUTER SCIENCE (HONS)

Faculty of Information and Communication Technology
(Perak Campus)

JANUARY 2019

UNIVERSITI TUNKU ABDUL RAHMAN

REPORT STATUS DECLARATION FORM

Title: ____

____

____

Academic Session: ____________

I ____

(CAPITAL LETTER)

declare that I allow this Final Year Project Report to be kept in

Universiti Tunku Abdul Rahman Library subject to the regulations as follows:

1. The dissertation is a property of the Library.
2. The Library is allowed to make copies of this dissertation for academic purposes.

Verified by,

________________________ ________________________

(Author's signature) (Supervisor's signature)

Address:

____

________________________ ________________________

________________________ Supervisor's name

Date: ____________________ **Date**: ____________________

AN AUTOMATED WEB SCRAPING TOOL

FOR MALAYSIA TOURISM

By

CHOONG WEI JEN

A REPORT

SUBMITTED TO

Universiti Tunku Abdul Rahman

in partial fulfilment of the requirements

for the degree of

BACHELOR OF COMPUTER SCIENCE (HONS)

Faculty of Information and Communication Technology
(Perak Campus)

JANUARY 2019

DECLARATION OF ORIGINALITY

I declare that this report entitled "**AN AUTOMATED WEB SCRAPING TOOL FOR MALAYSIAN TOURISM ANALYSIS**" is my own work except as cited in the references. The report has not been accepted for any degree and is not being submitted concurrently in candidature for any degree or other award.

Signature : ______________________

Name : ______________________

Date : ______________________

ACKNOWLEDGEMENTS

I would like to express my sincere thanks and appreciation to my supervisor, Dr Liew Soung Yue who has given me this great opportunity to participate in this data analytics project. It is my first step to establish a career in data analytics field. I also want to give my thanks to Mr.Pradeep for giving insightful feedback for this project. I really appreciate it and wanted to say thank you. Thank you, Dr.Liew and Mr.Pradeep.

Finally, I want to say thank you to my parents, my family and my friends for providing their support to me throughout the course.

ABSTRACT

This project is a web scraper design project for Malaysia tourism data. Data are the essential element of the data analytics process, but most public tourism data on the Internet have been overlooked for its value due to the process to collect data is very time-consuming and difficult. Therefore, this project is motivated to provide a low-cost and simple solution for collecting public tourism data on the Internet. Insights will be offered to those who want to build their own web scraper on the methodology, concept, and design through the realization of this project.

As for the technical part, agile System Development Life Cycle (SDLC) methodology is being adopted throughout this project. Emphasize of this project has been placed on capturing the public tourism data from the travel website by targeting the HTML code structure of that particular website. Thus, this project will be demonstrating how to interpret the HTML code structure of a website and how to locate targeted element for data extraction through HTML locator. Besides, this project will discuss on the selection of the most suitable programming language, libraries, tools and frameworks. As this project will be developed in Python, therefore the understanding on building a simple user interface using Python and the technique to save the extracted data into a csv file will be delivered as well. Furthermore, this project also covered some degree of data pre-processing because the extracted data attributes may have excessive text. A very important aspect in this project is to test the performance of the proposed system, therefore the most appropriate testing approach will also be surveyed and implemented on the system. Last but not least, a contingency plan regarding backup and recovery will also be discussed in case of event that system encountered errors.

A web scraping system which is specifically designed for Malaysia tourism will be developed to ease the process of collecting tourism data and it could potentially bring the focus of tourism industries and government sector on the public tourism data for the improvement of Malaysia tourism.

Table of Contents

LIST OF FIGURES

LIST OF TABLES

Chapter 1: Introduction

1.1 Problem Statement and Motivation

There are many public tourism data available on the Internet which could be potentially valuable assets for data analytics, but most of them have been wasted without being analyzed. These wasted data could have been collected and used to improve Malaysia tourism. However, many tourism industries such as hotels and travel agencies are not convinced enough to make use of these data because they might think that the cost and technical difficulties to collect these data is far too great while it could only bring little to none value to their business.

The motivation behind this project is to provide a low-cost and simple solution for collecting public tourism data on the Internet which could potentially bringing the focus of tourism industries and government sector to the value of collecting these data. Through the completion of this project, it could provide a convenient way for those who wish to perform data analytics on Malaysia tourism field to collect the travel-related data. In this project, much emphasizes have been put on developing a data collecting tool to extract and save the tourism data from the Internet. These collected data could then be further used for data analytics purpose and this would prove that tourism data on the Internet should not be overlooked, but instead businesses and government should be starting to realize its value. As such, the overall picture involves two major parts which is “online public tourism data collection” and “data analytics”, and this project will solely focused on the former.

1.2 Project Scope

The main concern of this project is to propose an efficient system that is able to collect travel-related data on Malaysia tourism. Therefore, reviews on the various method in collecting data will be performed. After revising the strengths and weaknesses of the data collection approaches, a web scraping tool targeting Malaysia tourism would be proposed. The implementation of the web scraping tool would be studied and the final product would be delivered upon the completion of this project.

The web scraping tool is an online data extracting system that allow users to get the tourism data without any coding themselves. To achieve the user-friendliness and usability of the system, a simple interface would be included so that any user can use this system easily. The proposed system will also cover the basic functionality which is to scrape the tourism data within travel website and then save the result. Besides, several testing methods will also be researched and carried out to ensure that the system is able to perform its functionality without errors. Furthermore, a backup and recovery mechanism will be implemented in case of the possibility that there are errors which caused the system to be interrupted in the middle of scrapping. Last but not least, data pre-processing will also be carried out in the case of unnecessary text captured.

The targeted data attributes are specifically predefined for the purpose of further analysis on Malaysia Tourism which will be discussed in Chapter 4.5. This system will target 3 main travel categories which are hotels, restaurants, and attractions as tourists will always concern about the questions such as where to stay, where to eat, and where to play. As this project is currently solely intended for Malaysian tourism aspect, therefore the region that is to be covered for scraping will be the states and federal territories of Malaysia such as Penang, Perak, and Kuala Lumpur.

1.3 Project Objectives

This project's main objective is to build a firm foundation for online public tourism data collection for Malaysia tourism field. The amount of data is very crucial to data analytics process, therefore a web scraping tool which can automatically collect the public tourism data from the Internet will be developed.

This project will focus on scraping the public tourism data from travel website. Through observation, it is noticed that many travel websites have provided recommending services for several categories of tourism information such as accommodation, place of attraction and restaurant. Among these services provided, there is actually a lot of valuable data that this project can prioritize to capture including travel website's user rating, comment, and their review date.

Although this project is part of data analytics, but other phases of the data analytics process which has included data cleaning, data analyzing, data interpreting and data visualizing is not covered in this project. Besides, this project will not cover the procedure to obtain the data through the other way such as requesting tourism data from the government or creating own apps to collect private data (eg. username, age, gender) from tourist.

1.4 Impact, Significance and Contribution

Malaysia is a unique country with many races, cultures and also the beautiful natural environment, it has attracted many tourists all over the world to visit it every years. Therefore, tourism is a very important aspect to the Malaysia economy and has contributed much to the government's income. This project has an insight on the potential of applying data analytics on Malaysia tourism field, therefore this project is aimed to build a foundation for data collection phase of data analytics which is to collect public tourism data of tourists from the Internet. This project is expected to bring benefits to those especially data analysts who are intended to perform data analytics on Malaysia tourism field but doesn't have enough data to do so.

Data is categorized into public data, private data and government data. Government data is the data preserved by government which is usually highly confidential while public data is the data that is available on the Internet such as the comments and reviews made in the online forum or social media. Private data is the data kept by the individual or organization that provide services to people, it usually contain travel website's user confidential data such as location, email, telephone number, age, gender and etc. in order to customize services for each individual user of the travel website.

The first step in conducting data analytics is to have enough data, therefore a data analyst will have 3 choices which is public data, private data and government data as explained above. However, requesting government data is usually a long process as government has to verify requestor's identity and credits, evaluate the project they are working on, and be convinced the requestor will not use the data for other purposes. Besides, the requesting private data from organization will normally involve trust issues, legal issues, and also benefits of both side. On the other hand, one can just capture the public data from the Internet without requesting it.

With the realization of this project, data analysts can easily obtain the data they needed for the data analytics process. Due to the ease of obtaining data, more time and resources which has been saved can increase the efficiency of data analytics process , thus increased the productivity. In long run, this project can indirectly benefit to the Malaysia society by improving tourism field through the mean of data analytics.

For example, there are few enormous labeled dataset such as MNIST, CIFAR, and ImageNet available on the Internet. MNIST has more than 60 000 of training images and 10 000 of testing images of handwritten digits which is widely used by people to test their machine learning algorithm. This has contributed much to the Machine Learning/Deep Learning field by providing a large yet standard dataset to the community so that one does not need to spend much time into collecting the dataset for implementing their system.

1.5 Background Information

Malaysia has always been a travel attraction and has attracted people all around the world to visit Malaysia. According to the Tourism Malaysia (n.d.), Malaysia have attracted 26.8 million of tourist arrivals and has earned RM82.1 billion in 2016. The statistic concluded by Ministry of Tourism and Culture Malaysia has showed that tourism has generated a great amount of income for Malaysia and contributed much to Malaysia economy. The strength of tourism in Malaysia is that Malaysia has a beautiful natural environment and there is a diversity in cultures and foods. Besides, most Malaysians are able to communicate in English, and therefore tourists who are visiting Malaysia will not have communication barrier with local people. However, by referring to the statistic done by Ministry of Tourism and Culture Malaysia, the tourist arrivals and tourist receipt vary each year, and it is important to find out why the number has changed in order to further develop tourism sector. Therefore, data analytics could be applied here to find out the trends and identify problems quickly, so that decision making process can be eased and response can be made.

Data analytic is a method of processing and analyzing raw data so that conclusion can be drawn. It involves a 5-steps process of collecting data, cleaning data, analyzing data, interpreting data and visualizing data. Normally, people will link data analytics to big data when mentioned. However, the term data analytics is only a general term for processing of data over time, but it will be eventually evolve into big data when the demand of data is high.

Due to the fact that much data especially the public tourism data has been wasted throughout years when they could be of use for many improvements on Malaysia tourism, and the reason why this project is so important is because there would be so much more useful information can be obtained from these wasted data through data analysis. Tourist behavior is defined as the changes in how the tourists behave according to their attitudes throughout their travel (Vuuren & Slabbert, 2011). When there are more data collected for data analytic, the current trend of tourist's travelling pattern and purchasing behavior could be better understood and comprehended. For example, by analyzing tourist's gender, age and nationality, the characteristic of high-spending tourist group and their purchasing motive could be observed. With this information, tourism businesses can adapt their business to the trend and enforce strategy for more profit. Besides, data analytics allow the prediction of an outcome based on the data collected after analyzation. Moreover, an abnormal phenomenon could also be identified quickly and measures could be carry out to handle it effectively. For example, the predicted outcome stated that particular tourist area should be attracting more tourist at a particular year, but the actual outcome state that the tourists visiting that area has decreased, and that's how changes is spotted so that response can be made quickly, such as conduct research to find out the problem and solve it.

In conclusion, anyone who is ambitious to perform data analytics, they will need a lot of data, and this is the reason why this project is playing an important role here as it is aimed to build a foundation for data collection system model which is to collect data for data analytics. It is expected to be beneficial to all tourism businesses and government agency. In the end, this project will be focusing on continuously capturing as much public tourism data as possible for data analytics.

1.6 Highlight of Achievement

The proposed system has successfully achieved its main functionality which is to identify the intended tourism data on the travel website in order to extract it down and save it into a csv file.

Besides, the complete execution time for the proposed system has been massively reduced. For example, the first working prototype version of the system has been recorded that 3 hours will be needed to scrape about 6,600 entries of data while the final deliverance of this project has the record time of scraping about 82,000 data entries within 22 hours. The proposed system has improved from scraping about 36 data entries per second to 62 data entries every second, which its execution time has been roughly improved by 72%.

In addition, the contingency plan regarding backup and recovery has also been implemented for the proposed system. The first working prototype will required a complete restart when encountering errors during runtime and all the previously scrapped data will be lost while the final deliverable is able to continue scraping the data from where it failed and therefore preserved the previously scrapped data. Besides, the contingency plan regarding the network instability has also been implemented to reduce the occurrence of the errors caused by the network problem. However, there are still much improvements can be done to handle the network problem which will be discussed in the later chapter.

1.7 Report Organization

In Chapter 1, the general aspects of project such as problem statement, motivation, project scope and objectives is defined in detail. Besides, project's impact, significance, contributions, and background information of the project is also discussed,

In Chapter 2, some research papers and works on the existing method and practices to collect data has been discussed. These practices has been reviewed to highlight their strength and criticize its weakness. The possible improvements and refinement has been discussed in this chapter in the effort of overcoming its weaknesses.

In Chapter 3, the complete development flow of system is discussed in details. It includes the how the system is development such as what has to be done in each of the development state and its reason. Besides, several system flowchart has been attached to show the overview of how the system will be processed during run time, and each block has been discussed in detail.

In Chapter 4, several aspects regarding the development of the system has been discussed. This includes the methodology adopted in this project and its general work process, selected tools for development such as programming language, user requirement for using the system, evaluation on the adopted system testing method, explanation on the output result of the system, and finally the legal and ethical issues regarding the system.

In Chapter 5, the challenges encountered during the system development process and the solutions has been discussed.

The last chapter will be providing the summary of the project and the future works to be done in order to further enhance the usability of the system will also be mentioned.

Chapter 2: Literature Review

Before data analytics can be conducted, a large amount of data must be first be prepared to get a convincing result. As there are various way to collect data, therefore one has to determine the most suitable method for the task nature. The first step in doing so is to define the data requirements clearly. After that, one may roughly know which methodology is best fit for the task. For example, if the researcher wants to collect the private data such as ages and gender, web scraping method might not work as most website will protect these information. Then, the next step is to study on the techniques of the selected method in order to obtain the targeted data from desired source. In this section, the existing methods in collecting data which has been practiced by other researcher will be reviewed. The strength of the existing methods, their weakness, and the possible ways to resolve the weakness will be discussed as well.

2.1 Field Study

The most commonly practiced data collection technique is to conduct a field study. Traditionally, people will conduct their own research to obtain data, which has included observation, interview and questionnaire, focus group and etc. According to the report "*A COMMUNITY-BASED TOURISM PLANNING PROCESS MODEL: KYUQUOT SOUND AREA. B.C.*" written by Pinel, D. P. (1998, pp.53-64), he went on a field study in Kyuquot to collect first hand data for his research. He have adopted methodology including participant observation, both formal and informal interviewing, and focus group. To ease his field study, Pinel lived with 2 different hosts during his visit as he stated that this could expose himself to a more variety of encounters and it would convince people that his research is more reliable and unbiased. Although Pinel is focus on collecting community based tourism data, but the technique he has adopted can be apply to this project by conducting a field study on various hotel. For example, one could conduct their own observation by living in the hotel for several night. During the stay in the hotel, the characteristics of tourist and their choice of hotel could be observed, such as tourists choosing this hotel is mostly in their senior age. After obtaining observation, a follow-up research could be conducted via an interview or focus group with the tourist to find out the reason behind it and therefore collecting their comment on the hotel. Besides, one could also write their own review on the hotel in term of its quality of service, cleanliness and so on.

The strength of adopting field study in collecting data is that one can get to know more detailed information. The data collected through field study is more likely to reflect real life situation. For example, in this project, the criticisms on the particular hotel website might be written by its competitor while the good comments might be written by its employee, but this is not likely to happen in field study as the focused subject is mainly the hotel customer who is neither on hotel side or its competitor side.

The weakness of field study is that the data collected might be biased and inaccurate, and it is not a sustainable data collection technique. Speaking of data inaccuracy, the data will be collected during the short stay of field study at particular hotel can never speak the same for the hotel for rest of the year, because what have been obtained can be different when under the same context but at a different date and time. For example, it might be a coincidence that more senior age tourists happen to check-in to this particular hotel, not because the hotel is favourite by senior age tourists. In fact, the situation might be entirely different such that there is a crowd of younger tourist check-in few days after the field study has been conducted. Besides, it would be completely biased if the researcher rely on their own experience at the hotel as data. There are a lot of options to consider when choosing a hotel, some people will prefer cheaper hotel while other might prioritize quality of service over price, and therefore the result can't be trustworthy. Furthermore, field study is not a sustainable data collection technique as it can only collect a limited amount of data over time while this project required to collect a large scale of data continuously. Therefore, field study is not a practical way of collecting data in this project.

To resolve the problem of field study in collecting data, data sharing could be a good method in obtaining data for this project.

2.2 Data Sharing

As mentioned above, field study has addressed its weakness such that the technique itself is unsustainable while the data it collected can be biased and inaccurate, data sharing is a better data collection method which could solve these problems. Data sharing is a process of exchanging data where the data is open and freely available while its process patterns and formats are known and standardized (Anon., n.d.). For example, CIFAR-10 is one of the most popular dataset used in Machine Learning field for computer vision task, the process of obtaining it can be considered as data sharing as it is an open source dataset which can be easily obtain from the Internet and its format has been standardized. Another type of data sharing is to request data from the organization or individual who own data, which if the permission is granted, then these data and its metadata can be used legally. By referring to the article *Injury Prevention* by Quigg et. al. (2012, pp.315-320), the process of data sharing has been demonstrated. As their objective is to examine how data sharing via local injury surveillance system can contribute to the prevention of violence, therefore they have established a data sharing session through a series of meetings, the focus of discussion was set on the issues regarding data availability, legislation and confidentiality. Although the focus of the article is different with this project, but the technique used can be apply to this project as well. For example, one can also schedule a meeting with data holders such as hotel management to convince them why they should share their data and what they will get in return.

The strength of the technique they have used is that data collected is complete and accurate, data is large in scale, and most importantly, data collection can be sustainable. First of all, hotels have to collect their customer's personal information and data all the time and store them for a long period of time until it is no longer important and relevant. Therefore, the data that will be obtained from the hotel must be large in scale. As some hotel might want to conduct their own data analysis on their customer to figure out which customer group they should be focusing on, therefore the data they have collected from their customer must be complete for this purpose. Opposing to the previous data collection method which is field study, this technique can actually allow researcher to obtain the data of the entire particular year, which is considered even more accurate in describing the hotel. Besides, it is possible that a partnership relationship can be formed with the hotel management so that they can provide their data continuously to achieve sustainability of data collection in this project.

However, the problem of data sharing is that most organizations are not willing to provide their data. The main reason is due to legal and privacy issue. As the legislation of Malaysia has stated, a data user is prohibited from processing personal data of a data subject without consent (Personal Data Protection Act 2010, 2010). Therefore, the hotel management could be sued in a legal due process if they are found to be leaking their data illegally. Moreover, why would they want to provide the data in the first place? Even if they are convinced that the project could benefit them, how can they make sure that their data will not be misused for other malicious purpose? It is completely reasonable if the hotel management decide not to share their data as there is too much risk in doing so. Even if they agree to share their data, a lot of efforts and works such as filtering the sensitive and confidential information must be done before it can be shared, which is the reason why most hotel management wouldn’t provide their data in the first place as it is resource consuming as well. Therefore, data sharing is also not recommended in this project.

To resolve the limitation encountered in this data collecting method, one could manually obtain data from the travel website through copy and paste.

2.3 Manually Copy and Paste from the Travel Website

As data sharing will have a difficulty in collecting data due to hotel management not willing to provide their data, one could collect data from the website through Internet. Due to the rising tide and high demand of E-commerce, businesses especially hotels have started to make use of the Internet to conduct their business. For example, most hotels will have their own website to provide information for their potential customer and allow their customer to book rooms prior to their travel. Among these hotel website, some may have prepared a user review board or forum for website user to rate the hotel and provide feedback, and these the data is exactly what the project needed. By specifying the keyword “manually” while capturing website data, it means that there must be a human operator to go through the website of each hotel one by one to look for the data that this project might need. They then have to copy down the related data and paste or manually key-in these data into the database.

The strength of this data collecting method is that it is cheaper in term of money and resources, and the data collected can be informative. When conducting a field study at particular hotel, the researcher will have to pay for the accommodation while data sharing will required whoever requested for data to have some leverage to negotiate with hotel management. However, capturing data from the website is obviously has a huge advantage over these 2 data collection methods as it demands only a little resource such that it will only require an operator to have a computer and a stable Internet connection for it to work. Besides, a lot of informative data that could not obtain from hotel management through data sharing can be captured down from the website. For example, the data provided by the hotel management mostly is about their customer which could specify the characteristic of tourists that would most likely to choose the particular hotel, but the data obtained from website can answer why tourists choose the hotel and their comments on it. For example, if there are many comments mentioned that the quality of service of the hotel is great, one could potentially add a label “Good Quality of Service” to the hotel and it could be a potential option of preference for tourist. Therefore, capture data from website is a practical way to greatly reduce the budget for this project while obtaining informative data.

The weakness of manually website data capturing is addressed as the process is slow and time consuming while it is also has a high demand on human resources. The reason this data collection method is slow and consume a lot of time is because human operator will have to identify the data that they need from the website. Before the operator can start to capture data, they will need to have a brief understanding on the website's structure to know that where the data is located in the website. For example, where is the user review board in the hotel website? After they have figured out where the data is, they will have to filter for the target data. For example, keywords in the website user's comment such as "Good service" and "Clean and tidy" are also a valuable data to this project, the operator will need to identify them by reading the user comment carefully so that important keywords are not missed, not to mention that there might be hundreds or thousands of user reviews in the website. Besides, there are thousands of hotel websites available on the Internet, this project will need to obtain data from as much different source as possible in the long run so that data analysis can be even more accurate. While capturing data manually from a single website already used up much time and effort, not to mention that it is impossible for an assigned personnel to work all day long just to collect the data, and therefore there will be a great demand on human resource to do the job such that a group of people in charge of capturing a new hotel website while another group of people focus on updating the latest user review from the previously captured website. Thus, manually capture data from website is not a practical method either.

To resolve the problem as mentioned above, one could make the whole process automated by using the existing data scraping software for website on the market.

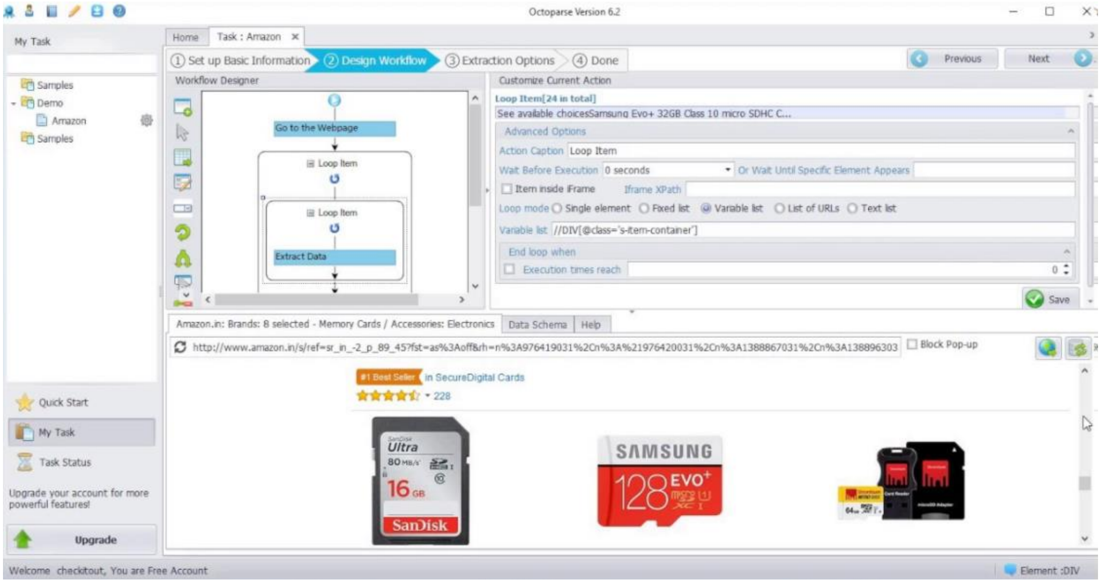
2.4 Existing Automated Web Scraping Software- Octoparse

Due to manually capture data from website is way too slow and human-resource demanding, another way is to do it in an automated way which is to use existing data scraping software. In fact, website data scraping software has the same nature as the previous data collection method, except that it is fully automated by computer and make the whole process of collecting data a lot easier. While there are many existing data scraping software available on the market, Octoparse is considered one of the best among them. Octoparse is a powerful and well-written web scraping program which is provided and maintained by its vendor, the working theory behind it is that it used spiders to browse through the entire contents of the website and scrape out the desired data such as website user's rating into spreadsheet or directly into database (Octoparse, 2014). To begin scraping data from website with Octoparse, user only have to setup the data extraction schedule for the first time and then it can automatically update the data extraction without needing human operator to function it for the rest of system's lifetime.

The strength of Octoparse is that it is faster in collecting data, easy to learn and use, and is sustainable. As Octoparse make the process of collecting website data automated, it is without a question has a signification effect in speeding up data collection phase. The reason behind this is because human operator can't be working all day long and eventually will need to rest due to physical constraint, meanwhile computer do not have such constraint and it can work continuously with faster computing speed. Besides, Octoparse is easy to learn and use due to it is a complete software with a good user interface. Octoparse has a user friendly interface, even user with no specific technical knowledge will be able to master how to use it within a short amount of time. Moreover, Octoparse is actually a sustainable solution to this project because of its scheduling data extraction feature. As mentioned above, this project is aimed to collect data in a continuous way, and Octoparse is happen to have this feature that will allow its user to schedule how often the data extraction from the website should be so that system that will be delivered by this project can always stay updated with these website. In this way, there would be no need for human operator to manually do the update once schedule has been set.

The weakness of Octoparse is that it is quite expensive, has no freedom of customization to the program and user has to completely rely on the vendor. Although Octoparse vendor allows user to use its software for free by signing up to Octoparse website, but the function provided is very limited. For example, free version has no data extraction scheduling feature and free user can only have a maximum of 2 concurrent run of the program on a local machine. For more functionality, user will have to subscribe to it on a monthly basis. Due to the needs of this project, subscription to the professional plan which will cost $209 (USD) per month is a must to complete the project. Besides, the ownership of the software belongs to its software vendor, user only pay for the license to use Octoparse when they subscribe to it, and therefore there is no source code available for user. Without source code, user can only carry out task that are supported by its pre-built functions without the freedom to customize the program itself. It is also worth mentioning that by using Octoparse, it means that user are completely relying on the vendor. Try to imagine what if something happen to the software vendor? For example, natural disaster has occurred and Octoparse vendor has to stop their service for a period of time, or malfunction is spotted and they are unable to solve it in the short time, it could be a huge loss to the user in term of business consideration.

To resolve this, one could write their own web scraping tool from scratch, so that they will have to freedom to customize their program and will not have to worry about the budget.



Loop Item[24 in total]

See available choicesSamsung Evo+ 32GB Class 10 micro SDHC C...

Action Caption: Loop Item

Wait Before Execution: 0 seconds   Or Wait Until Specific Element Appears:

□ Item inside iFrame   Iframe XPath:

Loop mode: ○ Single element ○ Fixed list ● Variable list ○ List of URLs ○ Text list

Variable list: //DIV[@class='s-item-container']

□ Execution times reach: 0

Amazon.in: Brands: 8 selected - Memory Cards / Accessories: Electronics

http://www.amazon.in/s/ref=sr_in_-2_p_89_45?fst=as%3Aoff&rh=n%3A976419031%2Cn%3A%21976420031%2Cn%3A1388867031%2Cn%3A138896303 □ Block Pop-up

Welcome checkitout, You are Free Account

Figure 2.4 User interface of Octoparse.

2.5 Web Scraping Tool- BeautifulSoup

As Octoparse is too expensive, has no freedom of customization to the program and user has to completely rely on the vendor, another way is to write a web scraping tool from scratch. Therefore, one of the available Python libraries which is known as BeautifulSoup can be utilized for this purpose. According to the website Crummy.com (2019), BeautifulSoup is specifically developed for the projects which are in quick turnaround nature such as screen-scraping. Its main functionality is to parse and extract data from the webpage.

The strength of Beautiful soup is that it is beginner-friendly as it is having a simple and easy to understand syntax which also does not require developer to write much code for the application. It also has a complete documentation support which also provide a lot of examples. Therefore, developer who does not have any experiences in web scraping can be easily get started with it and learn how to use it. Besides, BeautifulSoup is able to automatically handle the encoding of the HTML or XML documents so that developer does not need to spend time in spec

However, this library alone is not powerful enough to handle the scraping and therefore it has to work with other libraries such as “request” and “urlib2” to download the webpage in order to parse the HTML documents. Besides, BeautifulSoup does not support well to a complex logic and therefore not much customization can be done to the project. As the customization is limited, thus the extensibility of the project will be constrained and that is why there is not much related project in the web scraping field. This library is normally used for learning purpose only.

To resolve this, a more powerful and complete Python framework which is called Scrapy can be adopted into this project.

2.6 Web Scraping Tool- Scrapy

As BeautifulSoup is not powerful enough to handle the scraping alone and is usually used for learning purpose only. Therefore Scrapy is being introduced to this project. According to the website Scrapy.org (2008), it is a complete and collaborative data extracting framework specifically designed for Python. Scrapy allows developer creates an automated bot which is known as "spider" to crawl through the webpage. It will take an URL as input and then access it to download the webpage data then parse the HTML documents.

The strength of Scrapy is that the created spider will support customizations. For example, Scrapy is compatible with other libraries such as BeautifulSoup in order to extract the data from the downloaded DOM or modify the data. Besides, it also support the parsing selectors such as Xpath or CSS Selector which served to extract the data from the HTML documents. In addition, Scrapy is extremely fast in terms of the performance time and therefore it is suitable to work on a large dataset. There is also a large community using Scrapy for web scraping project and therefore there are a great community support in helping fixing the potential issues.

However, Scrapy is not able to handle well with the dynamic webpages which rely on the javascript or AJAX to build the contents of the webpage. The only possible way to render the complete DOM contents is access the webpage using a browser because javascript or AJAX is executed on top of the browser engine. Scrapy on the other hand just retrieve the webpage source code and does not support the functionality to interpret the javascript or AJAX code.

Therefore, another Python framework known as Selenium which can access the webpage through browser session can be utilized in this project to resolve Scrapy's inabilities.

Chapter 3: System Design

3.1 System Development Overview

To develop this project, much works and efforts has been done in order to achieve the objective and also obtain the best result which is able to match the expectation of this project. For further details, system development for this project has been separated into 3 main parts, including preparation, coding, and testing.

Preparation

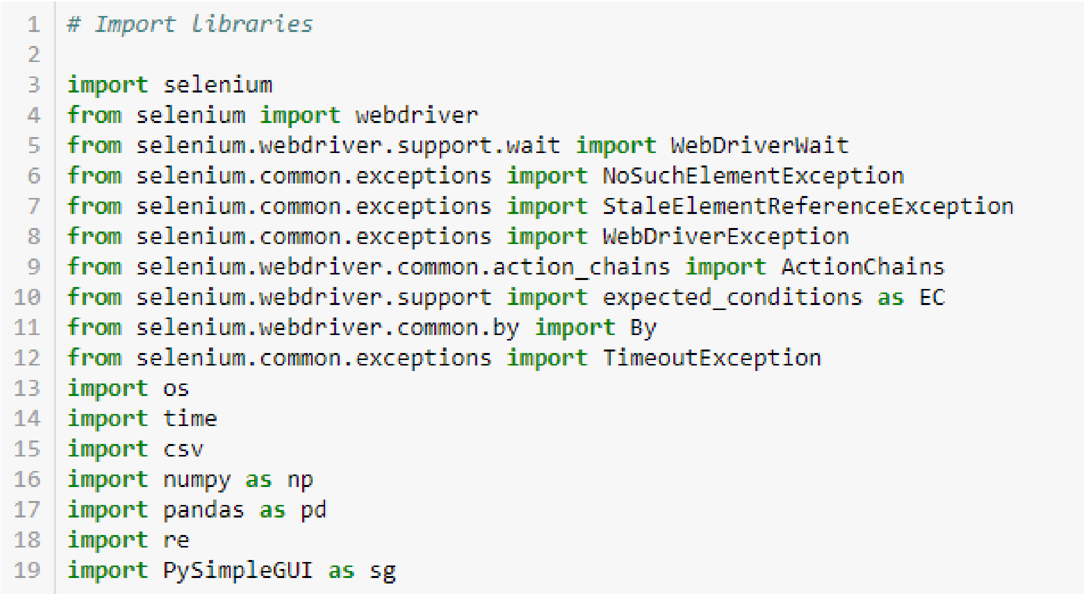
This project is developed by using several tools and required knowledge on various fields. Before getting started on the development process, one should acquire knowledge on HTML5 and CSS, and the skillset to write code in Python as this system is solely developed using this programming language. Python is preferred in this project due to there are many available libraries that are readily to support data extraction functionality. Before deciding on which tools and libraries to be used, system requirements must reviewed and identified to ensure the main functionality can be successfully implemented while any other libraries can be added after depend on the additional functionality. In this project, the Integrated Development Environment (IDE) “Jupyter Notebook” is encouraged to be used in development process as it has a Graphical User Interface (GUI) for better coding environment and easier debugging while there are 4 main libraries which are “NumPy”, “pandas”, “PySimpleGUI” and “selenium” must be utilized in this project. NumPy is one of the Python library that provide a large collection of high-level mathematical functions to operate on large and multi-dimensional arrays. Pandas is used in this project as it provides data structures and operations to manipulate tables. It is notable that “pandas” has a 2-dimensional labeled data structure called DataFrame which is able to store NumPy array of different type into each column. PySimpleGUI is an easy to use yet a powerful tool building a simple graphical user interface. Selenium is used to realize the data scraping functionality and it is a powerful framework that automates browser which is widely used by developer to test their web application. However, it is also powerful in scraping data from website due to its ability to simulate user’s action in a more efficient way.

For example, how long does it takes for user to copy and paste the whole webpage information into a csv file? On the other hand, selenium can scrape them as soon as the webpage is fully loaded. As for selenium to perform its function as intended, a browser driver is needed for it to control the browser behaviors. ChromeDriver is chosen as the tool for selenium to do its work as for personal preference.

After deciding the tools and libraries that needed to develop the main functionality of the system, next step is to download and install them into the computer. ChromeDriver can be easily downloaded on the Internet while Jupyter Notebook while any other identified Python's libraries can be downloaded and installed using the Anaconda prompt provided that there is an Internet connection. For example, one can download NumPy by entering the command `pip install numpy` or `conda install numpy`. However, installing the library package using Conda is sometime preferred as it is able to manage the package dependencies which may install, upgrade or sometimes downgrade other required package for the intended package to work. Furthermore, one should examine whether path is set for Python and its libraries when unable to use the Python tools or import Python libraries.

Coding

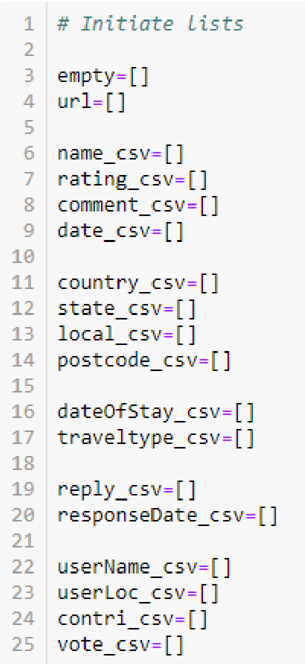
After coding environment has been successfully set up, coding phase is ready to be conducted. First, the libraries that have been installed earlier must be imported into the IDE first as shown in the figure below.



```python
# Import libraries

import selenium
from selenium import webdriver
from selenium.webdriver.support.wait import WebDriverWait
from selenium.common.exceptions import NoSuchElementException
from selenium.common.exceptions import StaleElementReferenceException
from selenium.common.exceptions import WebDriverException
from selenium.webdriver.common.action_chains import ActionChains
from selenium.webdriver.support import expected_conditions as EC
from selenium.webdriver.common.by import By
from selenium.common.exceptions import TimeoutException
import os
import time
import csv
import numpy as np
import pandas as pd
import re
import PySimpleGUI as sg
```

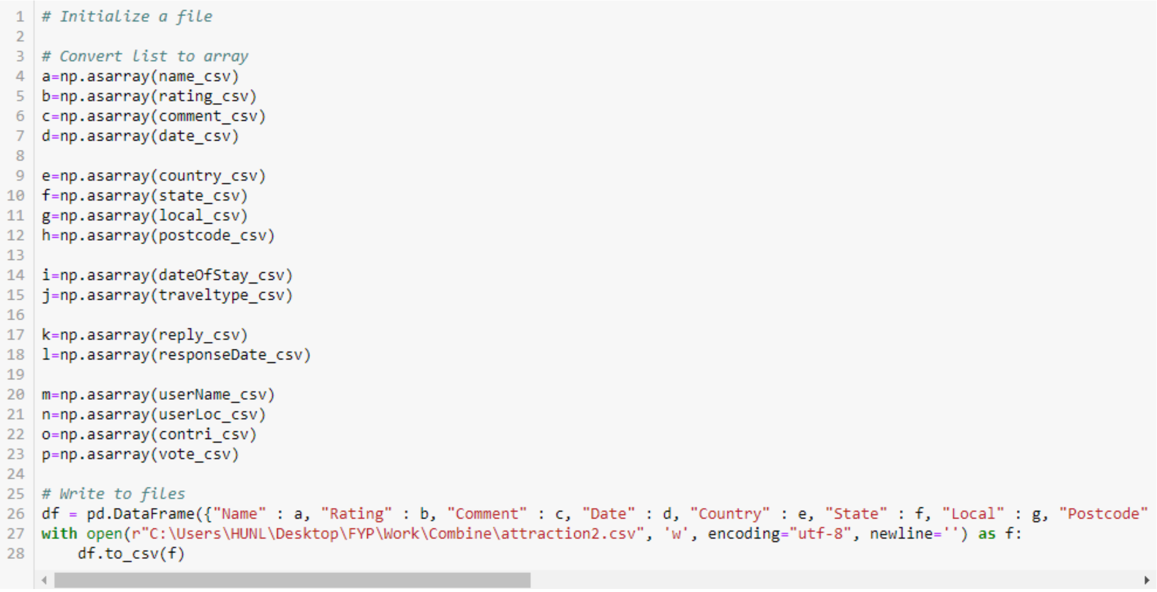
Figure 3.1.1 Imported libraries.

The next step is to initiate several empty lists for storing the data to be extracted and then convert them to NumPy array. A named csv file must be first initialized with named column so that the extracted data can be appended into this file. Implementation-wise, a data frame is created with each of the previously initialized empty array being assigned to its named column before the data frame is written into a csv file. The code snippets of the implementation is shown as below.



```python
# Initiate lists

empty=[]
url=[]

name_csv=[]
rating_csv=[]
comment_csv=[]
date_csv=[]

country_csv=[]
state_csv=[]
local_csv=[]
postcode_csv=[]

dateOfStay_csv=[]
traveltype_csv=[]

reply_csv=[]
responseDate_csv=[]

userName_csv=[]
userLoc_csv=[]
contri_csv=[]
vote_csv=[]
```

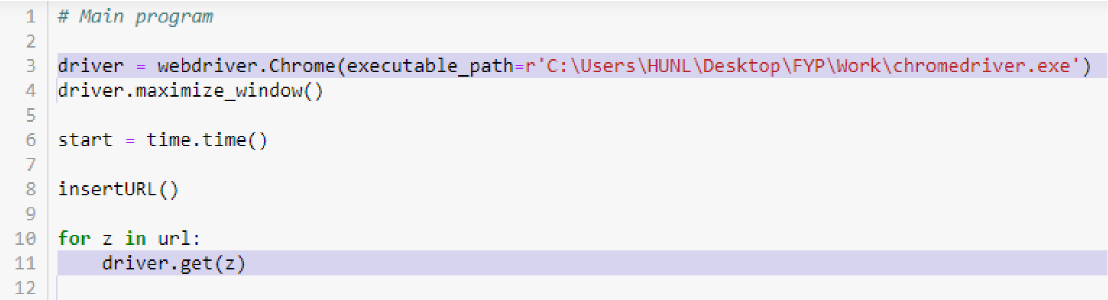
Figure 3.1.2 Initiate lists.



```
# Initialize a file

# Convert list to array
a=np.asarray(name_csv)
b=np.asarray(rating_csv)
c=np.asarray(comment_csv)
d=np.asarray(date_csv)

e=np.asarray(country_csv)
f=np.asarray(state_csv)
g=np.asarray(local_csv)
h=np.asarray(postcode_csv)

i=np.asarray(dateOfStay_csv)
j=np.asarray(traveltype_csv)

k=np.asarray(reply_csv)
l=np.asarray(responseDate_csv)

m=np.asarray(userName_csv)
n=np.asarray(userLoc_csv)
o=np.asarray(contri_csv)
p=np.asarray(vote_csv)

# Write to files
df = pd.DataFrame({"Name" : a, "Rating" : b, "Comment" : c, "Date" : d, "Country" : e, "State" : f, "Local" : g, "Postcode"
with open(r"C:\Users\HUNL\Desktop\FYP\Work\Combine\attraction2.csv", 'w', encoding="utf-8", newline='') as f:
    df.to_csv(f)
```

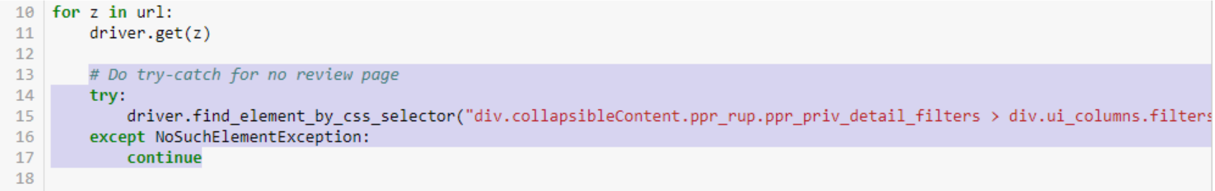
Figure 3.1.3 Create a csv file.

A main function is then defined based on the webpage structure and also the logic to interact with it. For example, the logic to be implemented for the system consists of verifying whether the travel webpage has reviews, specifying the interaction with the webpage so that it target intended data to scrape, and verifying whether the link has next page of reviews. A very important step to take note is that execute path of the ChromeDriver must be correctly specified in order to open the Chrome browser. After finish scraping, the results must be appended to the previously initialized csv file. The lists and arrays must then be cleared after finish scraping one link to avoid duplication of data as this program behaves in a way that append the result of each link into the csv file at a time. It is also notable to mention that clearing the unneeded memories is able to improve program efficiency as it is observed that larger variable memory can affect the performance such that the program may hang or lag.



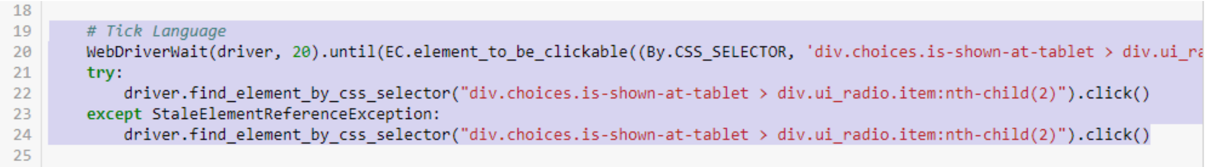
```
# Main program

driver = webdriver.Chrome(executable_path=r'C:\Users\HUNL\Desktop\FYP\Work\chromedriver.exe')
driver.maximize_window()

start = time.time()

insertURL()

for z in url:
    driver.get(z)

```

Figure 3.1.4 Open Chrome browser.



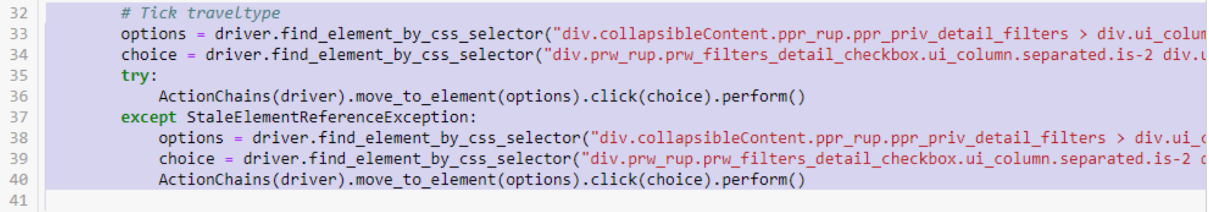
```
for z in url:
    driver.get(z)

    # Do try-catch for no review page
    try:
        driver.find_element_by_css_selector("div.collapsibleContent.ppr_rup.ppr_priv_detail_filters > div.ui_columns.filters
    except NoSuchElementException:
        continue

```

Figure 3.1.5 Check if webpage has targeted data.



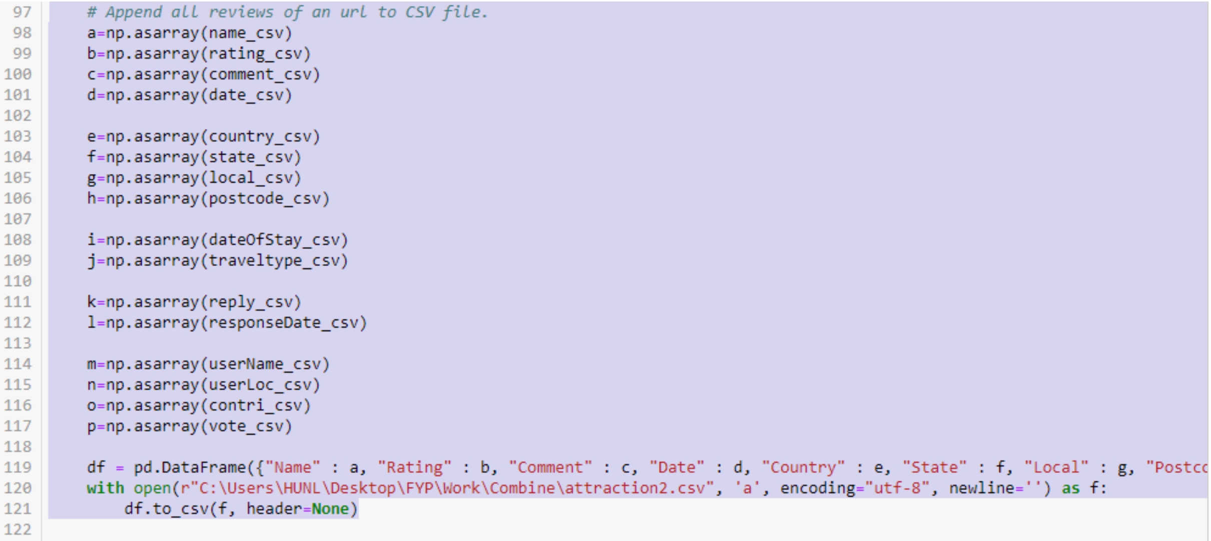
```

    # Tick Language
    WebDriverWait(driver, 20).until(EC.element_to_be_clickable((By.CSS_SELECTOR, 'div.choices.is-shown-at-tablet > div.ui_ra
    try:
        driver.find_element_by_css_selector("div.choices.is-shown-at-tablet > div.ui_radio.item:nth-child(2)").click()
    except StaleElementReferenceException:
        driver.find_element_by_css_selector("div.choices.is-shown-at-tablet > div.ui_radio.item:nth-child(2)").click()

```

Figure 3.1.6 Interact with webpage to show only data of target language.



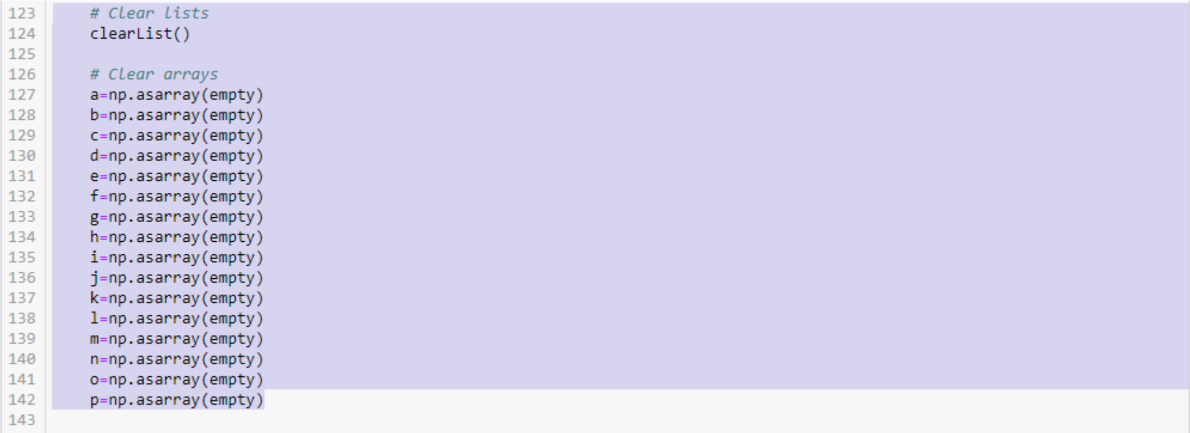
```
        # Tick traveltype
        options = driver.find_element_by_css_selector("div.collapsibleContent.ppr_rup.ppr_priv_detail_filters > div.ui_colum
        choice = driver.find_element_by_css_selector("div.prw_rup.prw_filters_detail_checkbox.ui_column.separated.is-2 div.u
        try:
            ActionChains(driver).move_to_element(options).click(choice).perform()
        except StaleElementReferenceException:
            options = driver.find_element_by_css_selector("div.collapsibleContent.ppr_rup.ppr_priv_detail_filters > div.ui_c
            choice = driver.find_element_by_css_selector("div.prw_rup.prw_filters_detail_checkbox.ui_column.separated.is-2 c
            ActionChains(driver).move_to_element(options).click(choice).perform()

```

Figure 3.1.7 Interact with webpage to show only intended data. (Same code is used to untick)



```
    # Append all reviews of an url to CSV file.
    a=np.asarray(name_csv)
    b=np.asarray(rating_csv)
    c=np.asarray(comment_csv)
    d=np.asarray(date_csv)

    e=np.asarray(country_csv)
    f=np.asarray(state_csv)
    g=np.asarray(local_csv)
    h=np.asarray(postcode_csv)

    i=np.asarray(dateOfStay_csv)
    j=np.asarray(traveltype_csv)

    k=np.asarray(reply_csv)
    l=np.asarray(responseDate_csv)

    m=np.asarray(userName_csv)
    n=np.asarray(userLoc_csv)
    o=np.asarray(contri_csv)
    p=np.asarray(vote_csv)

    df = pd.DataFrame({"Name" : a, "Rating" : b, "Comment" : c, "Date" : d, "Country" : e, "State" : f, "Local" : g, "Postcc
    with open(r"C:\Users\HUNL\Desktop\FYP\Work\Combine\attraction2.csv", 'a', encoding="utf-8", newline='') as f:
        df.to_csv(f, header=None)

```

Figure 3.1.8 Append the results into csv file.



```
    # Clear lists
    clearList()

    # Clear arrays
    a=np.asarray(empty)
    b=np.asarray(empty)
    c=np.asarray(empty)
    d=np.asarray(empty)
    e=np.asarray(empty)
    f=np.asarray(empty)
    g=np.asarray(empty)
    h=np.asarray(empty)
    i=np.asarray(empty)
    j=np.asarray(empty)
    k=np.asarray(empty)
    l=np.asarray(empty)
    m=np.asarray(empty)
    n=np.asarray(empty)
    o=np.asarray(empty)
    p=np.asarray(empty)
```

Figure 3.1.9 Release the memory.

Next, the functionality of scraping webpage content shall be implemented by defining a callable function named scraper(). Within this function, there are some logic to be programmed before scraping data such as expanding review content, verifying webpage interface, check if there exist next review container, and clicking into another element. As for verifying webpage interface, it has been noticed that under some random condition, the webpage may have different interface which is made up of different HTML structure. Furthermore, click into another webpage is an action specified for hotel scraping as the software must click into the user profile to scrape certain data. The data scraping is done through HTML locator provided by selenium. There are many HTML locators can be used in this system, CSS Selector is chosen among them and the reasons will be specified on later chapter. Developer should utilize Google Chrome’s “Inspect Elements” feature to view the webpage’s HTML code structure and have a deep understanding on how to use the CSS selector to locate the targeted data in order to scrape accurately and successfully. Some basic knowledge in utilizing CSS selector are shown below.

driver.find_element_by_css_selector(“#HEADING”)

➢ By specifying “#” for the element, it locate it by ID.

driver.find_element_by_css_selector(“.taLnk.ulBlueLinks”)

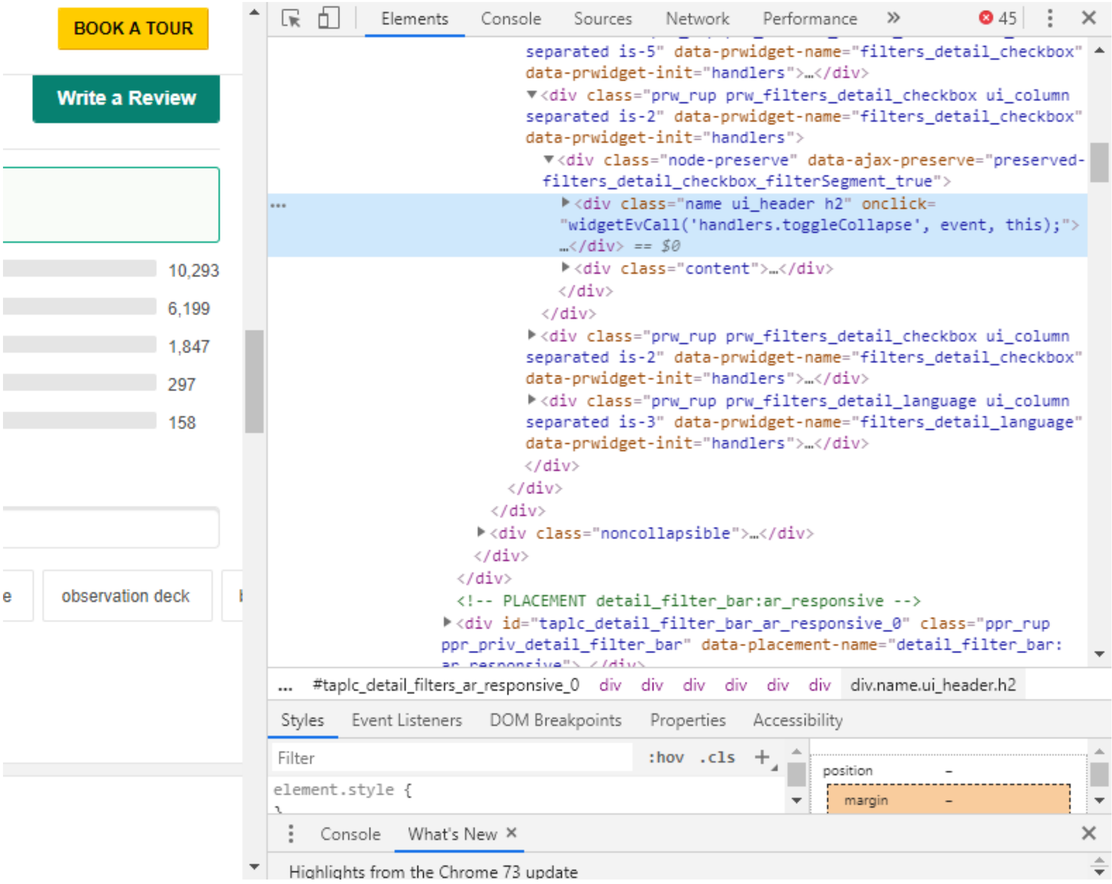
➢ By specifying “.” for the element, it locate it by class.

```
driver.find_element_by_css_selector("div.is-hidden-mobile > span.detail > span.locality")
```

- By specifying ">" between elements, it locate the element in a consecutive matter. For example, span.detail must be the direct child of div.is-hidden-mobile.

```
driver.find_element_by_css_selector("div.navLinks li.attractions.twoLines > a")
```

- By specifying space between elements, the second element does not necessary be a direct child of the first element.



```
            separated is-5" data-prwidget-name="filters_detail_checkbox"
            data-prwidget-init="handlers">…</div>
           ▼<div class="prw_rup prw_filters_detail_checkbox ui_column
            separated is-2" data-prwidget-name="filters_detail_checkbox"
            data-prwidget-init="handlers">
             ▼<div class="node-preserve" data-ajax-preserve="preserved-
              filters_detail_checkbox_filterSegment_true">
...            ▶<div class="name ui_header h2" onclick=
                "widgetEvCall('handlers.toggleCollapse', event, this);">
                …</div> == $0
               ▶<div class="content">…</div>
               </div>
             </div>
           ▶<div class="prw_rup prw_filters_detail_checkbox ui_column
            separated is-2" data-prwidget-name="filters_detail_checkbox"
            data-prwidget-init="handlers">…</div>
           ▶<div class="prw_rup prw_filters_detail_language ui_column
            separated is-3" data-prwidget-name="filters_detail_language"
            data-prwidget-init="handlers">…</div>
            </div>
          </div>
        </div>
       ▶<div class="noncollapsible">…</div>
       </div>
     </div>
     <!-- PLACEMENT detail_filter_bar:ar_responsive -->
    ▶<div id="taplc_detail_filter_bar_ar_responsive_0" class="ppr_rup
    ppr_priv_detail_filter_bar" data-placement-name="detail_filter_bar:
```

```
... #taplc_detail_filters_ar_responsive_0  div  div  div  div  div  div  div.name.ui_header.h2
```

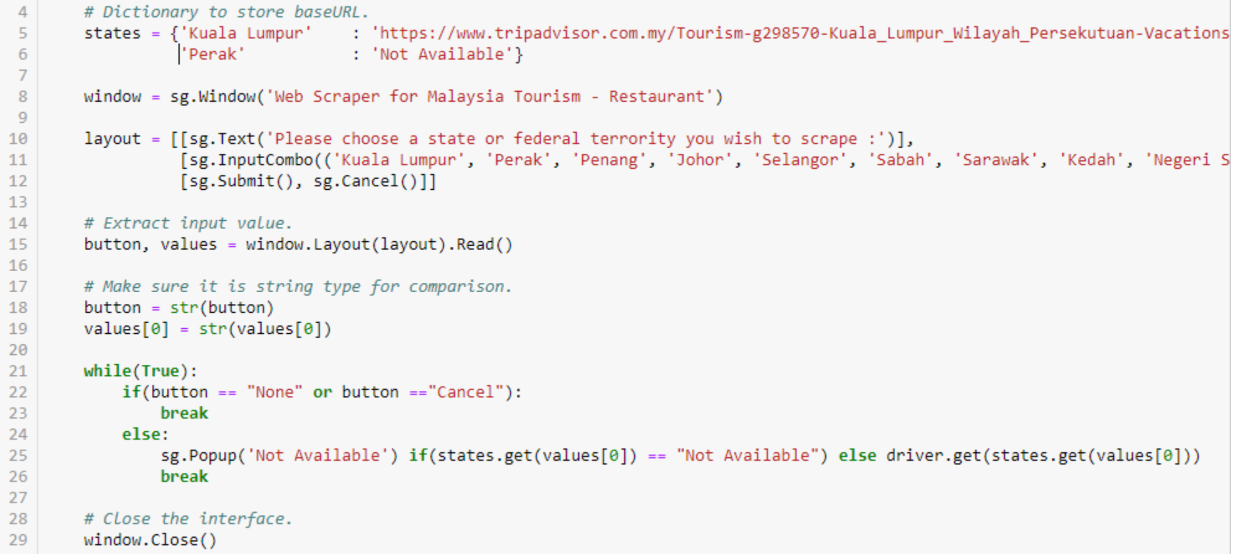
```
element.style {
```

Figure 3.1.10 "Inspect Elements" feature.

Now that the system already have a mature scraping functionality, what it needed next is a list of links of the webpage to be scrapped. To deal with this issue, the function insertURL() is defined. When this function is being called, a graphical user interface (GUI) will be triggered which will then provide user with a drop down list to select the regional travel data they wish to obtain. The implementation of the GUI has 3 steps, which is to initiate a window and specify its layout, extract the user input, and close the window as shown in the figure 3.1.11. The code below is defining a dropdown list with 5 inputs to be shown at a time from the list and having the length size of 20.

sg.InputCombo((‘input1’,’input2’), size=(20,5))

Each of the input is associated with a base URL that include all the links to every travel sites within the region. For example, the base URL will be the link to the Kuala Lumpur webpage as shown in figure 3.1.13 while the branch URLs would be the webpages of every travel sites in Kuala Lumpur as shown in the figure 3.1.14. There are also certain logic must be implemented in this function such as clicking into some elements, checking the interface of webpage that contain links, and check if webpage has next page of links. The need to check the interface type is because Kuala Lumpur has a different interface for the first webpage that contains links.



```
# Dictionary to store baseURL.
states = {'Kuala Lumpur'    : 'https://www.tripadvisor.com.my/Tourism-g298570-Kuala_Lumpur_Wilayah_Persekutuan-Vacations
          'Perak'           : 'Not Available'}

window = sg.Window('Web Scraper for Malaysia Tourism - Restaurant')

layout = [[sg.Text('Please choose a state or federal terrority you wish to scrape :')],
          [sg.InputCombo(('Kuala Lumpur', 'Perak', 'Penang', 'Johor', 'Selangor', 'Sabah', 'Sarawak', 'Kedah', 'Negeri S
          [sg.Submit(), sg.Cancel()]]

# Extract input value.
button, values = window.Layout(layout).Read()

# Make sure it is string type for comparison.
button = str(button)
values[0] = str(values[0])

while(True):
    if(button == "None" or button =="Cancel"):
        break
    else:
        sg.Popup('Not Available') if(states.get(values[0]) == "Not Available") else driver.get(states.get(values[0]))
        break

# Close the interface.
window.Close()
```

Figure 3.1.11 Implementation of the system GUI.

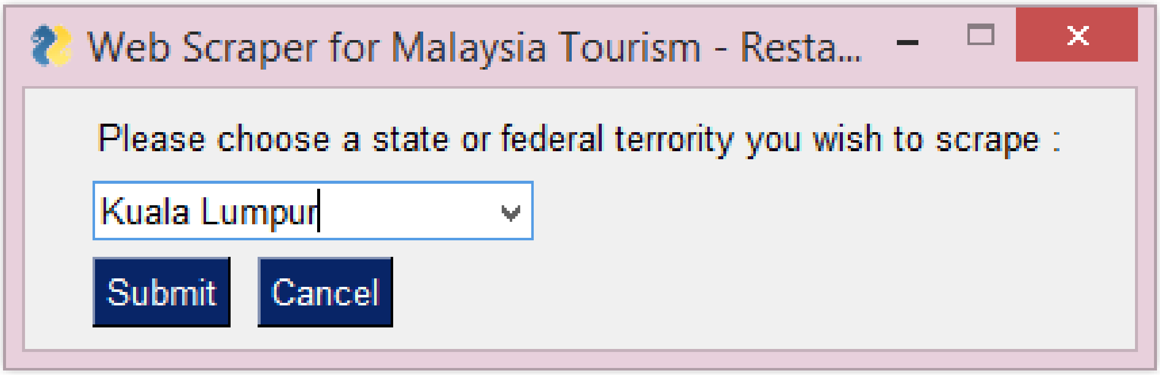
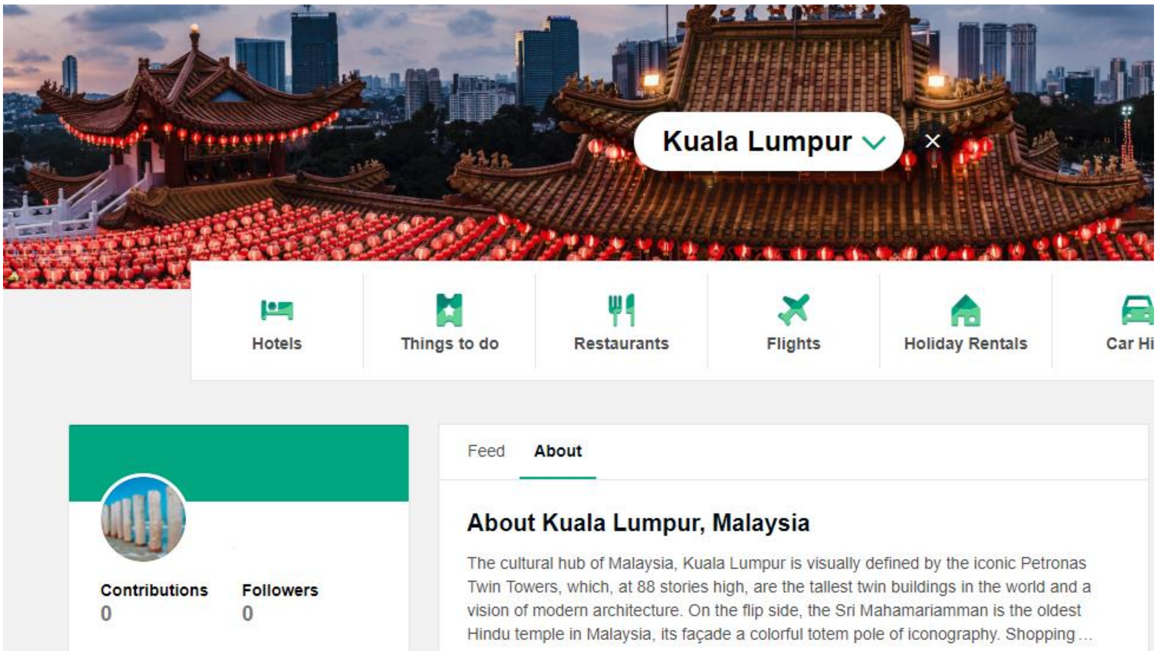

Web Scraper for Malaysia Tourism - Resta...
Please choose a state or federal terrority you wish to scrape :
Kuala Lumpur
Submit
Cancel

Figure 3.1.12 Graphical interface of the system.



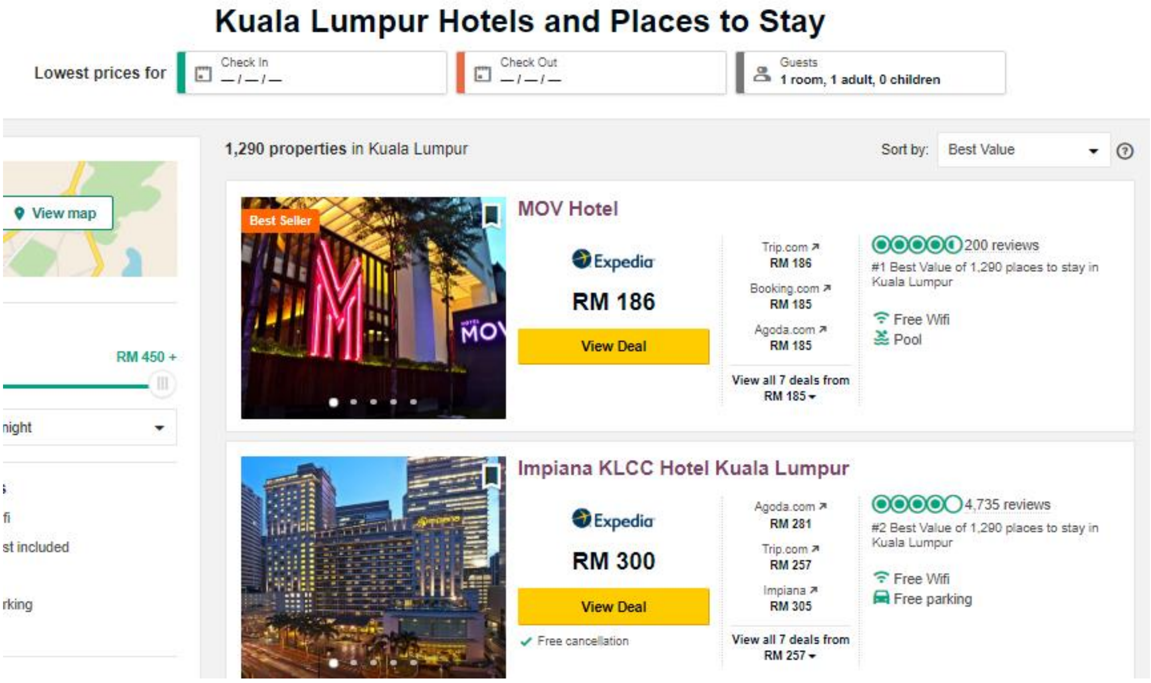
Contributions 0

Followers 0

About Kuala Lumpur, Malaysia

The cultural hub of Malaysia, Kuala Lumpur is visually defined by the iconic Petronas Twin Towers, which, at 88 stories high, are the tallest twin buildings in the world and a vision of modern architecture. On the flip side, the Sri Mahamariamman is the oldest Hindu temple in Malaysia, its façade a colorful totem pole of iconography. Shopping ...

Figure 3.1.13 Base URL of the region.



Kuala Lumpur Hotels and Places to Stay

Lowest prices for Check In —/—/— Check Out —/—/— Guests 1 room, 1 adult, 0 children

1,290 properties in Kuala Lumpur

Sort by: Best Value

MOV Hotel

Expedia RM 186

Trip.com RM 186
Booking.com RM 185
Agoda.com RM 185
View all 7 deals from RM 185

200 reviews
#1 Best Value of 1,290 places to stay in Kuala Lumpur
Free Wifi
Pool

Impiana KLCC Hotel Kuala Lumpur

Expedia RM 300

✓ Free cancellation

Agoda.com RM 281
Trip.com RM 257
Impiana RM 305
View all 7 deals from RM 257

4,735 reviews
#2 Best Value of 1,290 places to stay in Kuala Lumpur
Free Wifi
Free parking

Figure 3.1.14 Branch URLs of all travel sites within the region.

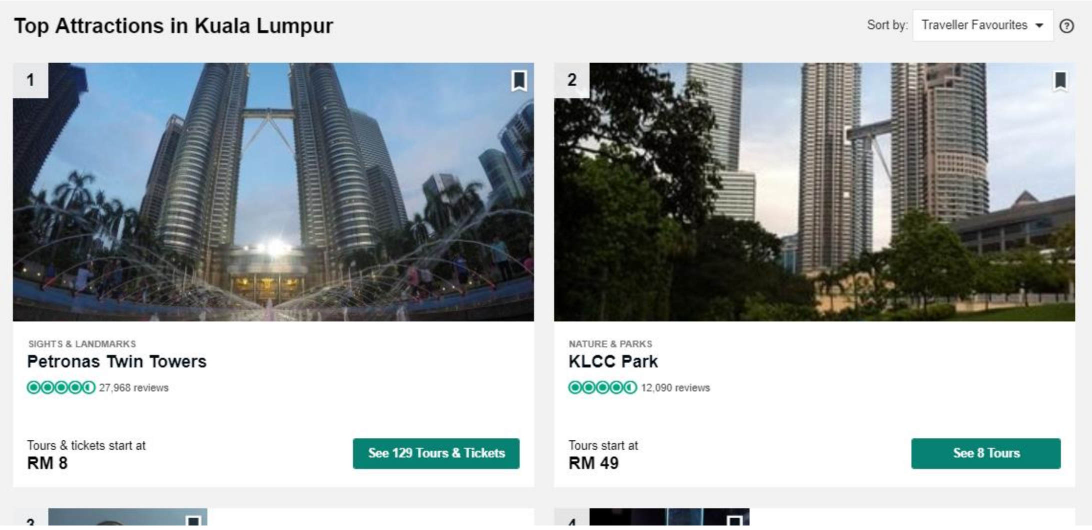
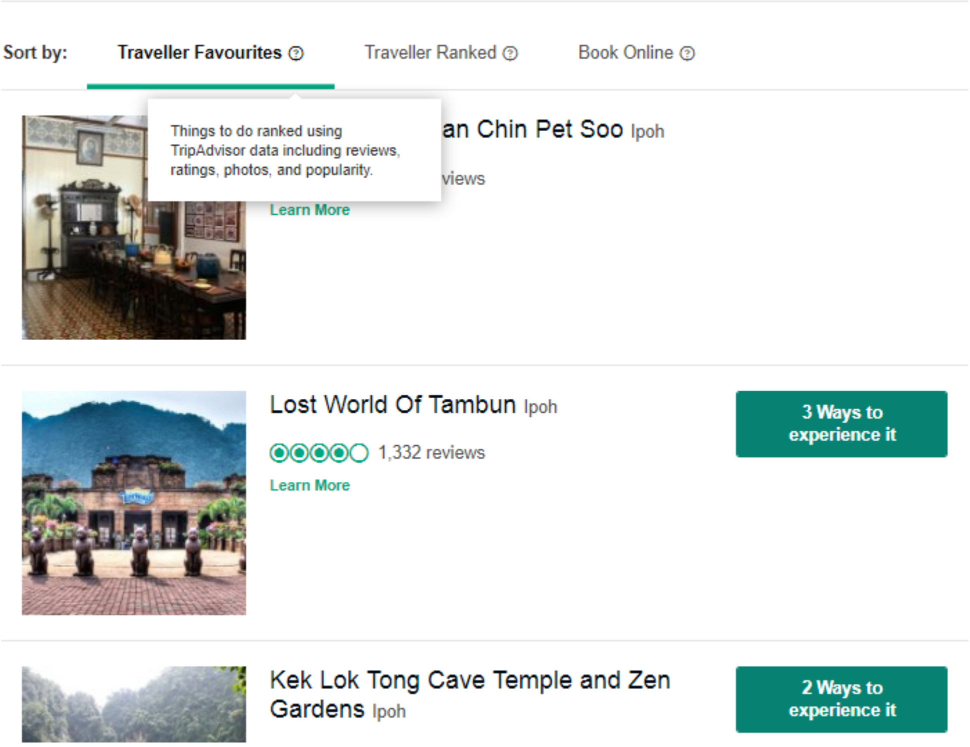
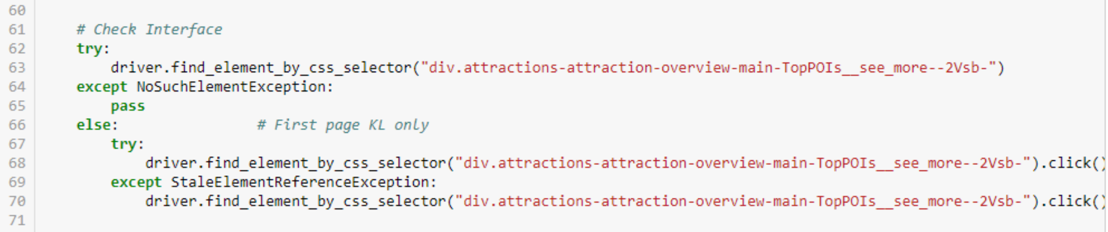


Figure 3.1.15 Interface of Kuala Lumpur attraction.



Things to do ranked using TripAdvisor data including reviews, ratings, photos, and popularity.

Figure 3.1.16 Interface of other attraction.

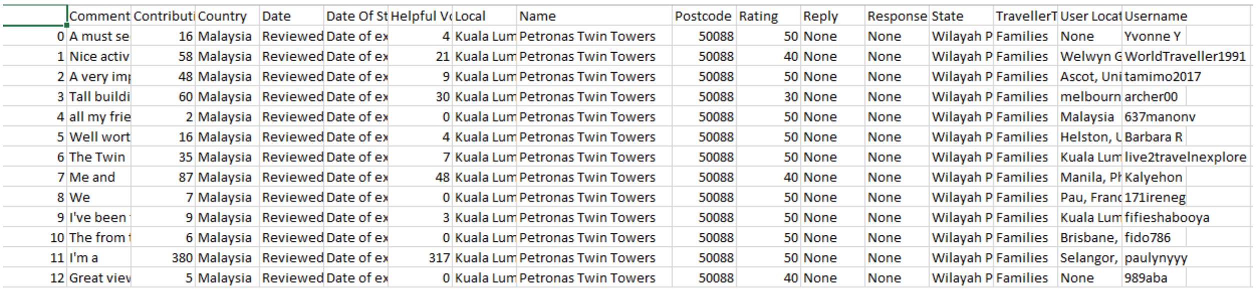


```

    # Check Interface
    try:
        driver.find_element_by_css_selector("div.attractions-attraction-overview-main-TopPOIs__see_more--2Vsb-")
    except NoSuchElementException:
        pass
    else:                   # First page KL only
        try:
            driver.find_element_by_css_selector("div.attractions-attraction-overview-main-TopPOIs__see_more--2Vsb-").click(
        except StaleElementReferenceException:
            driver.find_element_by_css_selector("div.attractions-attraction-overview-main-TopPOIs__see_more--2Vsb-").click(

```

Figure 3.1.17 Checking interface for attraction.

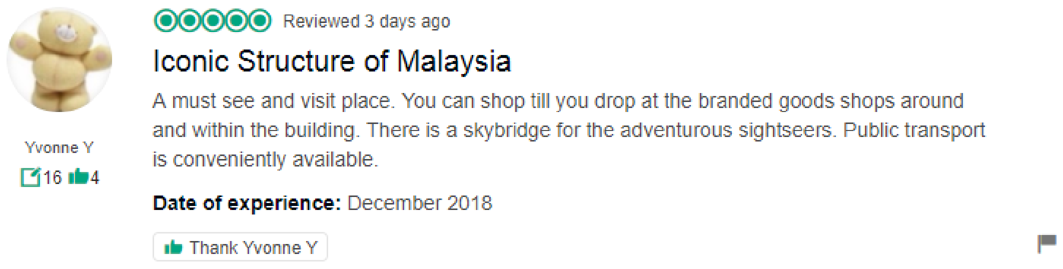
Testing

After the coding phase has been completed, testing is conducted before the delivery to ensure there are no unexpected errors and the proposed system is able to do what it was designed for. The main criteria of testing is the correctness of the scrapped data such as no duplicate data entry, no unmatched feature within a data entry and no missing data. Manual testing is adopted in the project such that a personnel has to be present during the execution of the program and check whether there is any abnormal behavior. The personnel should also verify the correctness of the scrapped data with the target data from the scrapped webpage. However, the scrapped reviews can be more than hundred thousand and it is impossible for developer to verify the entry one by one with the webpage, therefore sampling technique is used to scrape only a few webpages with less number of reviews and then verify the scrapped data with the data on sampled webpages. Besides, NumPy and pandas can also be used to verify for the duplication of data in the case that there is a large number of scrapped data. The code shown in Figure showed that data entry (rows) of data frame has been converted into tuple and np.unique() is called to drop duplicated data entry. When the length of the data frame before and after the preprocessing are not of the same length, then there must be duplication of data. It is notable that this testing method is implemented in a way that combine all the columns as a single value to compare with other data entry. This is because if only a single column is chosen, there might be false positive result for the testing. For example, the column "Name" is intended to have a lot of duplicate name although the column "Username" may look perfect, but there might be a situation where same might user reviewed several places.



	Comment	Contributi	Country	Date	Date Of St	Helpful V	Local	Name	Postcode	Rating	Reply	Response	State	TravellerT	User Locat	Username
0	A must se	16	Malaysia	Reviewed	Date of ex	4	Kuala Lum	Petronas Twin Towers	50088	50	None	None	Wilayah P	Families	None	Yvonne Y
1	Nice activ	58	Malaysia	Reviewed	Date of ex	21	Kuala Lum	Petronas Twin Towers	50088	40	None	None	Wilayah P	Families	Welwyn G	WorldTraveller1991
2	A very imp	48	Malaysia	Reviewed	Date of ex	9	Kuala Lum	Petronas Twin Towers	50088	50	None	None	Wilayah P	Families	Ascot, Uni	tamimo2017
3	Tall buildi	60	Malaysia	Reviewed	Date of ex	30	Kuala Lum	Petronas Twin Towers	50088	30	None	None	Wilayah P	Families	melbourn	archer00
4	all my frie	2	Malaysia	Reviewed	Date of ex	0	Kuala Lum	Petronas Twin Towers	50088	50	None	None	Wilayah P	Families	Malaysia	637manonv
5	Well wort	16	Malaysia	Reviewed	Date of ex	4	Kuala Lum	Petronas Twin Towers	50088	50	None	None	Wilayah P	Families	Helston, U	Barbara R
6	The Twin	35	Malaysia	Reviewed	Date of ex	7	Kuala Lum	Petronas Twin Towers	50088	50	None	None	Wilayah P	Families	Kuala Lum	live2travelnexplore
7	Me and	87	Malaysia	Reviewed	Date of ex	48	Kuala Lum	Petronas Twin Towers	50088	40	None	None	Wilayah P	Families	Manila, Ph	Kalyehon
8	We	7	Malaysia	Reviewed	Date of ex	0	Kuala Lum	Petronas Twin Towers	50088	50	None	None	Wilayah P	Families	Pau, Franc	171ireneg
9	I've been	9	Malaysia	Reviewed	Date of ex	3	Kuala Lum	Petronas Twin Towers	50088	50	None	None	Wilayah P	Families	Kuala Lum	fifieshabooya
10	The from	6	Malaysia	Reviewed	Date of ex	0	Kuala Lum	Petronas Twin Towers	50088	50	None	None	Wilayah P	Families	Brisbane,	fido786
11	I'm a	380	Malaysia	Reviewed	Date of ex	317	Kuala Lum	Petronas Twin Towers	50088	50	None	None	Wilayah P	Families	Selangor,	paulynyyy
12	Great viev	5	Malaysia	Reviewed	Date of ex	0	Kuala Lum	Petronas Twin Towers	50088	40	None	None	Wilayah P	Families	None	989aba

Figure 3.1.18 Example of scrapped attraction data.



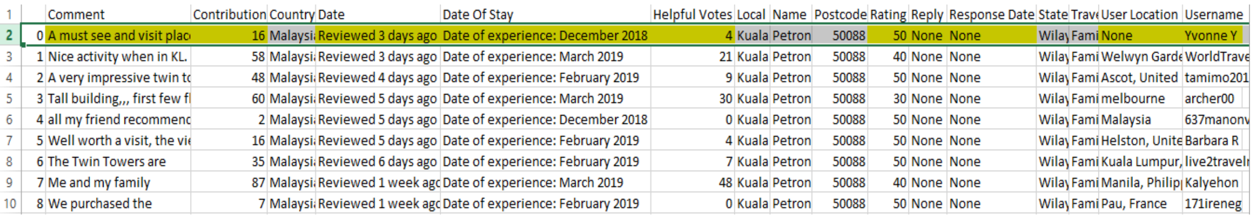
Reviewed 3 days ago

Iconic Structure of Malaysia

A must see and visit place. You can shop till you drop at the branded goods shops around and within the building. There is a skybridge for the adventurous sightseers. Public transport is conveniently available.

Date of experience: December 2018

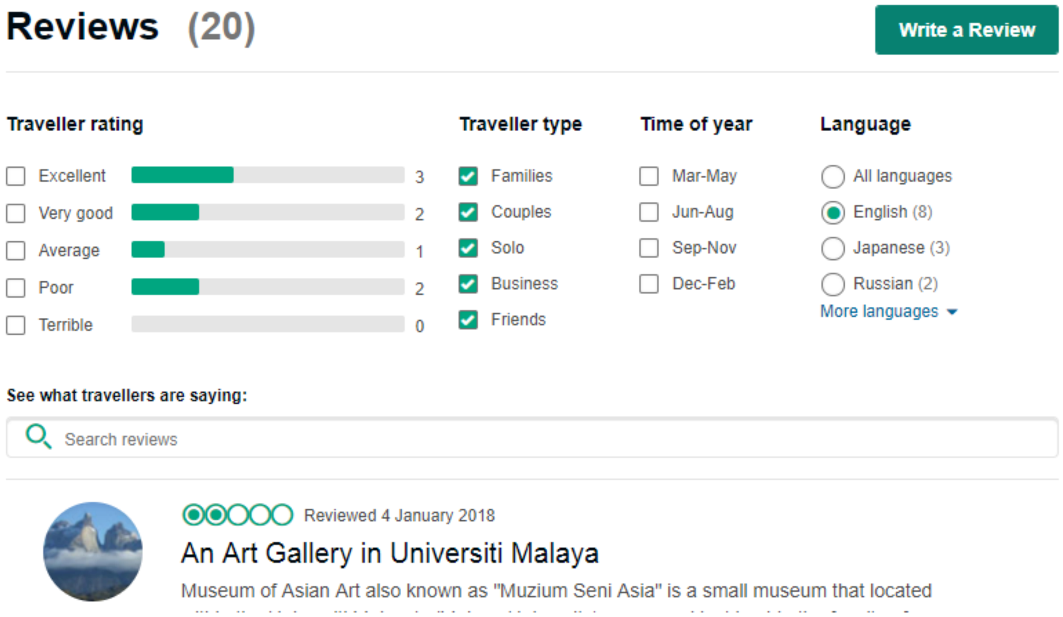
Figure 3.1.19 Sample data from travel webpage.



1		Comment	Contribution	Country	Date	Date Of Stay	Helpful Votes	Local	Name	Postcode	Rating	Reply	Response Date	State	Trave	User Location	Username
2	0	A must see and visit plac	16	Malaysi	Reviewed 3 days ago	Date of experience: December 2018	4	Kuala	Petron	50088	50	None	None	Wilay	Fami	None	Yvonne Y
3	1	Nice activity when in KL.	58	Malaysi	Reviewed 3 days ago	Date of experience: March 2019	21	Kuala	Petron	50088	40	None	None	Wilay	Fami	Welwyn Garde	WorldTrave
4	2	A very impressive twin tc	48	Malaysi	Reviewed 4 days ago	Date of experience: February 2019	9	Kuala	Petron	50088	50	None	None	Wilay	Fami	Ascot, United	tamimo201
5	3	Tall building,,, first few fl	60	Malaysi	Reviewed 5 days ago	Date of experience: March 2019	30	Kuala	Petron	50088	30	None	None	Wilay	Fami	melbourne	archer00
6	4	all my friend recommenc	2	Malaysi	Reviewed 5 days ago	Date of experience: December 2018	0	Kuala	Petron	50088	50	None	None	Wilay	Fami	Malaysia	637manonv
7	5	Well worth a visit, the vi	16	Malaysi	Reviewed 5 days ago	Date of experience: February 2019	4	Kuala	Petron	50088	50	None	None	Wilay	Fami	Helston, Unite	Barbara R
8	6	The Twin Towers are	35	Malaysi	Reviewed 6 days ago	Date of experience: February 2019	7	Kuala	Petron	50088	50	None	None	Wilay	Fami	Kuala Lumpur,	live2travel
9	7	Me and my family	87	Malaysi	Reviewed 1 week agc	Date of experience: March 2019	48	Kuala	Petron	50088	40	None	None	Wilay	Fami	Manila, Philip	Kalyehon
10	8	We purchased the	7	Malaysi	Reviewed 1 week agc	Date of experience: February 2019	0	Kuala	Petron	50088	50	None	None	Wilay	Fami	Pau, France	171ireneg

Figure 3.1.20 Example showing scrapped data matched travel webpage data in Figure.

As Figure showed the available data of the webpage, the highlighted part in Figure showed the data scrapped from that webpage. The un-highlighted part is scrapped from another part of the same webpage which is not shown here.



Reviews (20)

Write a Review

Traveller rating

□ Excellent 3
□ Very good 2
□ Average 1
□ Poor 2
□ Terrible 0

Traveller type

☑ Families
☑ Couples
☑ Solo
☑ Business
☑ Friends

Time of year

□ Mar-May
□ Jun-Aug
□ Sep-Nov
□ Dec-Feb

Language

○ All languages
● English (8)
○ Japanese (3)
○ Russian (2)
More languages

See what travellers are saying:

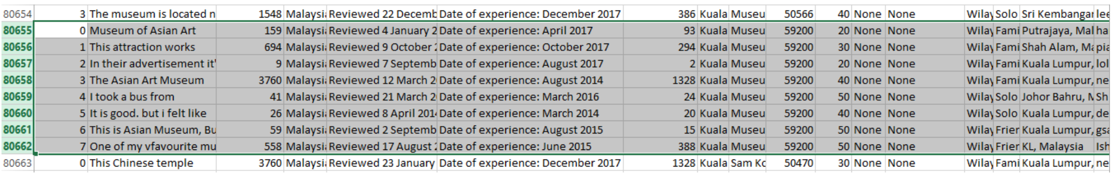
Search reviews

Reviewed 4 January 2018

An Art Gallery in Universiti Malaya

Museum of Asian Art also known as "Muzium Seni Asia" is a small museum that located

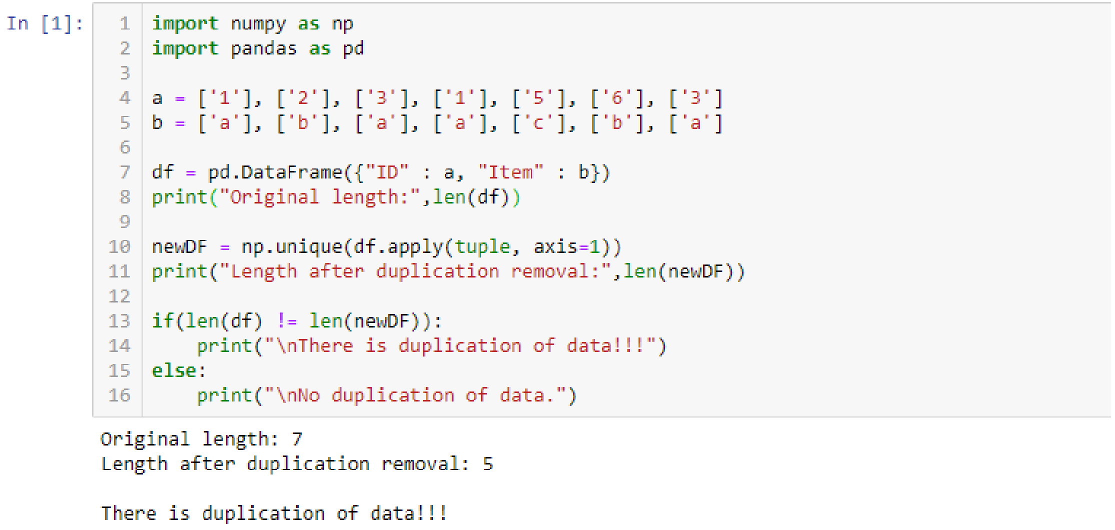
Figure 3.1.21 Overview of an attraction from travel website.



80654	3	The museum is located n	1548	Malaysi	Reviewed 22 Decemb	Date of experience: December 2017	386	Kuala	Museu	50566	40	None	None	Wilay	Solo	Sri Kembangar	lee
80655	0	Museum of Asian Art	159	Malaysi	Reviewed 4 January 2	Date of experience: April 2017	93	Kuala	Museu	59200	20	None	None	Wilay	Fami	Putrajaya, Mal	ha
80656	1	This attraction works	694	Malaysi	Reviewed 9 October :	Date of experience: October 2017	294	Kuala	Museu	59200	30	None	None	Wilay	Fami	Shah Alam, Ma	pia
80657	2	In their advertisement it'	9	Malaysi	Reviewed 7 Septemb	Date of experience: August 2017	2	Kuala	Museu	59200	20	None	None	Wilay	Fami	Kuala Lumpur,	lol
80658	3	The Asian Art Museum	3760	Malaysi	Reviewed 12 March 2	Date of experience: August 2014	1328	Kuala	Museu	59200	40	None	None	Wilay	Fami	Kuala Lumpur,	ne
80659	4	I took a bus from	41	Malaysi	Reviewed 21 March 2	Date of experience: March 2016	24	Kuala	Museu	59200	50	None	None	Wilay	Solo	Johor Bahru, N	Sh
80660	5	It is good. but i felt like	26	Malaysi	Reviewed 8 April 201	Date of experience: March 2014	20	Kuala	Museu	59200	40	None	None	Wilay	Solo	Kuala Lumpur,	de
80661	6	This is Asian Museum, Bu	59	Malaysi	Reviewed 2 Septemb	Date of experience: August 2015	15	Kuala	Museu	59200	50	None	None	Wilay	Frier	Kuala Lumpur,	gsa
80662	7	One of my vfavourite mu	558	Malaysi	Reviewed 17 August :	Date of experience: June 2015	388	Kuala	Museu	59200	50	None	None	Wilay	Frier	KL, Malaysia	Ish
80663	0	This Chinese temple	3760	Malaysi	Reviewed 23 January	Date of experience: December 2017	1328	Kuala	Sam Kc	50470	30	None	None	Wilay	Fami	Kuala Lumpur,	ne

Figure 3.1.22 Example showing scrapped data do not have missing data entry.

Figure showed there are total 20 reviews in the webpage, but since this program only target English comment, so there would be a total of 8 reviews scrapped in ideal. However, some reviews do not have “Traveller Type”, therefore the number of scrapped data might be sometime lesser. Figure showed the number of data scrapped from the webpage.



```python
import numpy as np
import pandas as pd

a = ['1'], ['2'], ['3'], ['1'], ['5'], ['6'], ['3']
b = ['a'], ['b'], ['a'], ['a'], ['c'], ['b'], ['a']

df = pd.DataFrame({"ID" : a, "Item" : b})
print("Original length:",len(df))

newDF = np.unique(df.apply(tuple, axis=1))
print("Length after duplication removal:",len(newDF))

if(len(df) != len(newDF)):
    print("\nThere is duplication of data!!!")
else:
    print("\nNo duplication of data.")
```

```
Original length: 7
Length after duplication removal: 5

There is duplication of data!!!
```

Figure 3.1.23 Example code to test duplication of data.

3.2 System Flowchart

3.2.1 Main Function

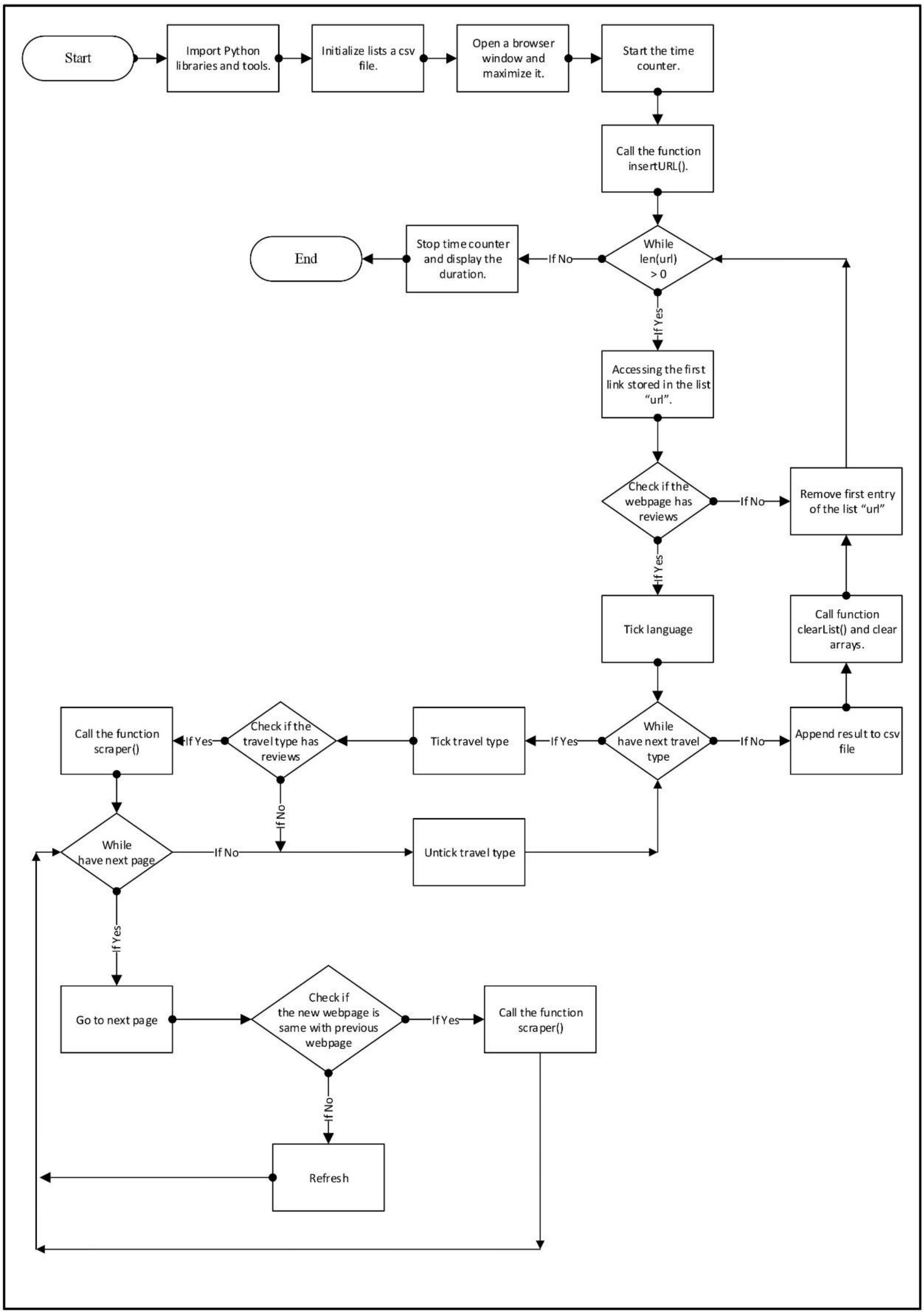

Start
Import Python libraries and tools.
Initialize lists a csv file.
Open a browser window and maximize it.
Start the time counter.
Call the function insertURL().
While len(url) > 0
If No
Stop time counter and display the duration.
End
If Yes
Accessing the first link stored in the list "url".
Check if the webpage has reviews
If No
Remove first entry of the list "url"
If Yes
Tick language
Call function clearList() and clear arrays.
While have next travel type
If No
Append result to csv file
If Yes
Tick travel type
Check if the travel type has reviews
If Yes
Call the function scraper()
If No
While have next page
If No
Untick travel type
If Yes
Go to next page
Check if the new webpage is same with previous webpage
If Yes
Call the function scraper()
If No
Refresh

Figure 3.2.1 System flowchart of Attraction Scraper & Restaurant Scraper.

Attraction Scraper and Restaurant Scraper have the same coding logic, therefore they have the same system flow. The flowchart above showed how the system is executed.

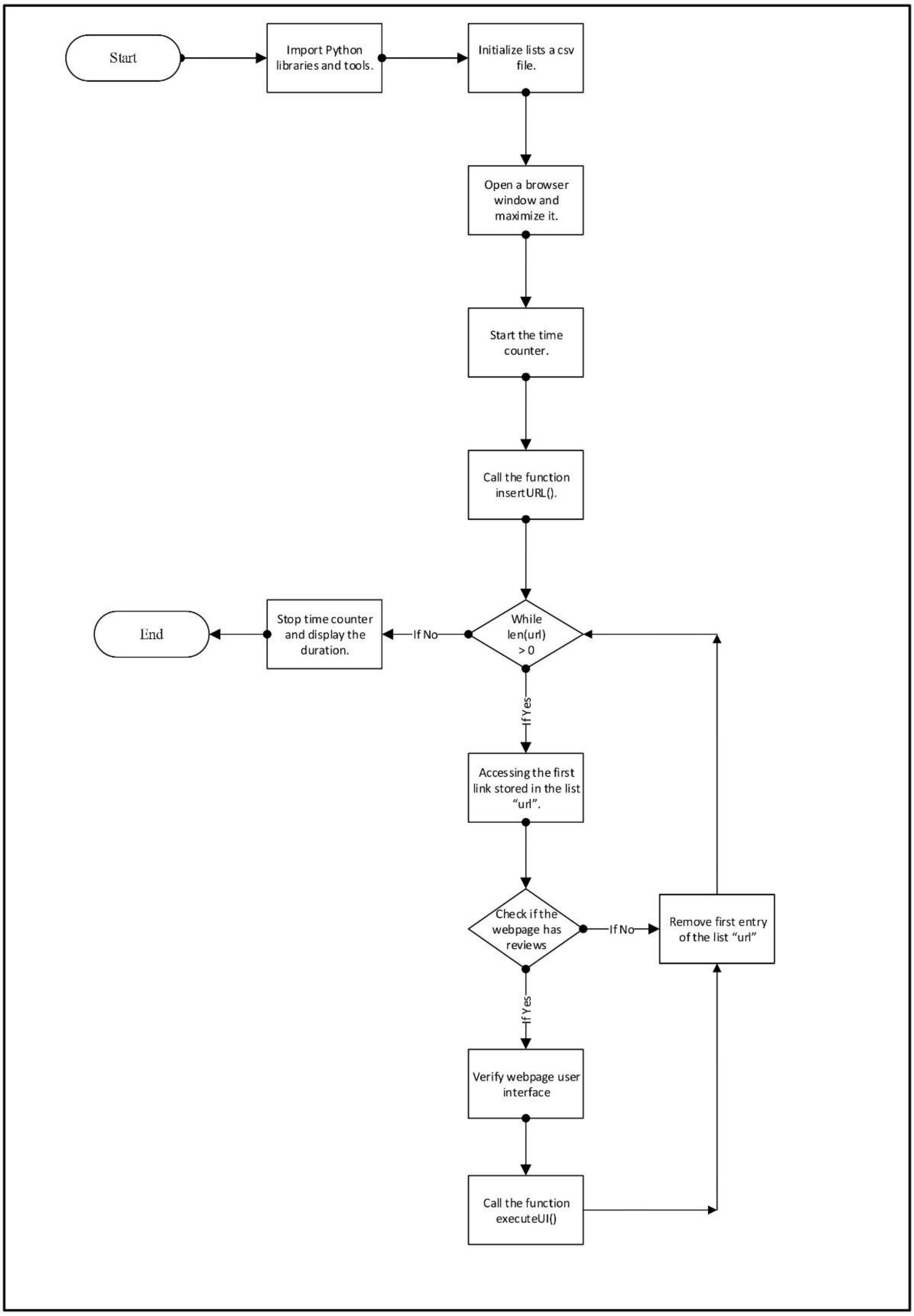

Start
Import Python libraries and tools.
Initialize lists a csv file.
Open a browser window and maximize it.
Start the time counter.
Call the function insertURL().
While len(url) > 0
If No
Stop time counter and display the duration.
End
If Yes
Accessing the first link stored in the list "url".
Check if the webpage has reviews
If No
Remove first entry of the list "url"
If Yes
Verify webpage user interface
Call the function executeUI()

Figure 3.2.2 System flowchart of hotel scraper.

Since Hotel Scraper have 2 interfaces that are triggered randomly when the hotel webpage is triggered, therefore it has a different coding logic from Attraction Scraper and Restaurant Scraper. The flowchart as shown above is more simplified because some process has been defined in the function "executeUI()".

3.2.2 Other Functions

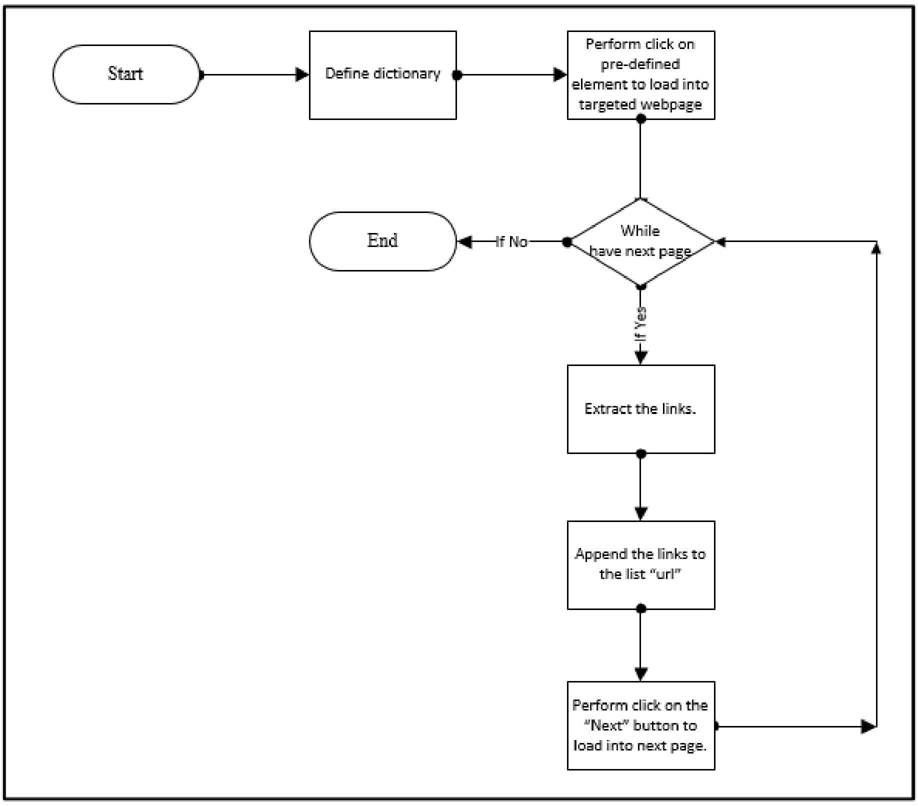

Start
Define dictionary
Perform click on pre-defined element to load into targeted webpage
While have next page
If No
End
If Yes
Extract the links.
Append the links to the list "url"
Perform click on the "Next" button to load into next page.

Figure 3.2.3 Flowchart of the function insertURL().

This function is shared among all 3 scrapers. A dictionary is defined within this function

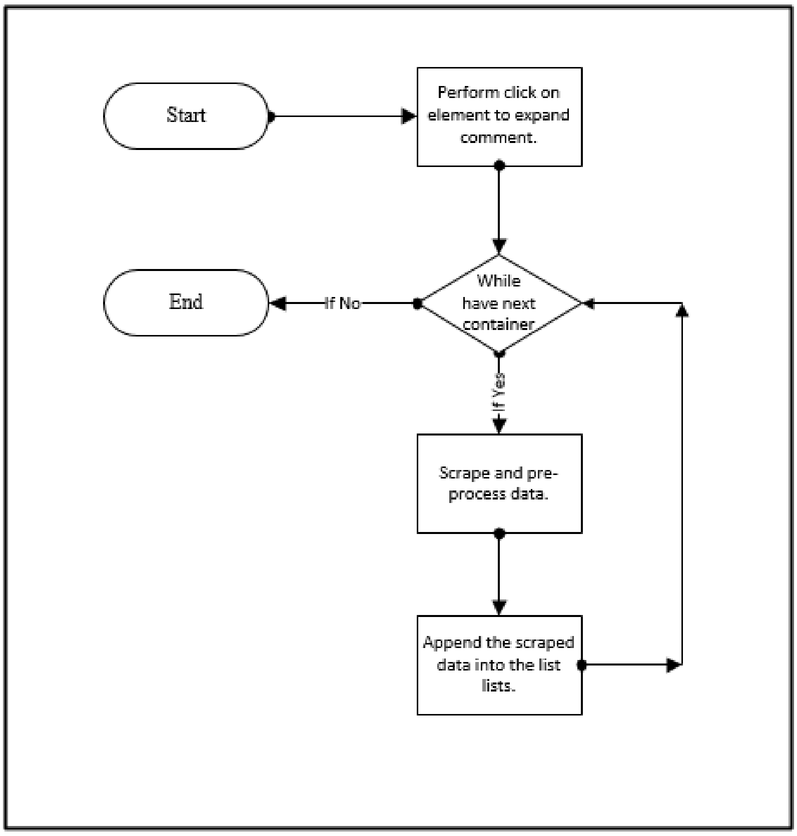

Start
Perform click on element to expand comment.
End
If No
While have next container
If Yes
Scrape and pre-process data.
Append the scraped data into the list lists.

Figure 3.2.4 Flowchart of the functions scraper(), oldUI_scraper(), and newUI_scraper().

This function is shared among all scrapers as well. Attraction Scraper and Restaurant Scraper both have each of this function which is named as “scraper()” while Hotel Scraper has 2 of this function which are “oldUI_scraper()” and “newUI_scraper()”. These 2 function of Hotel Scraper are having the same coding logic but they have different HTML locator due to different HTML structure caused by different webpage interface.

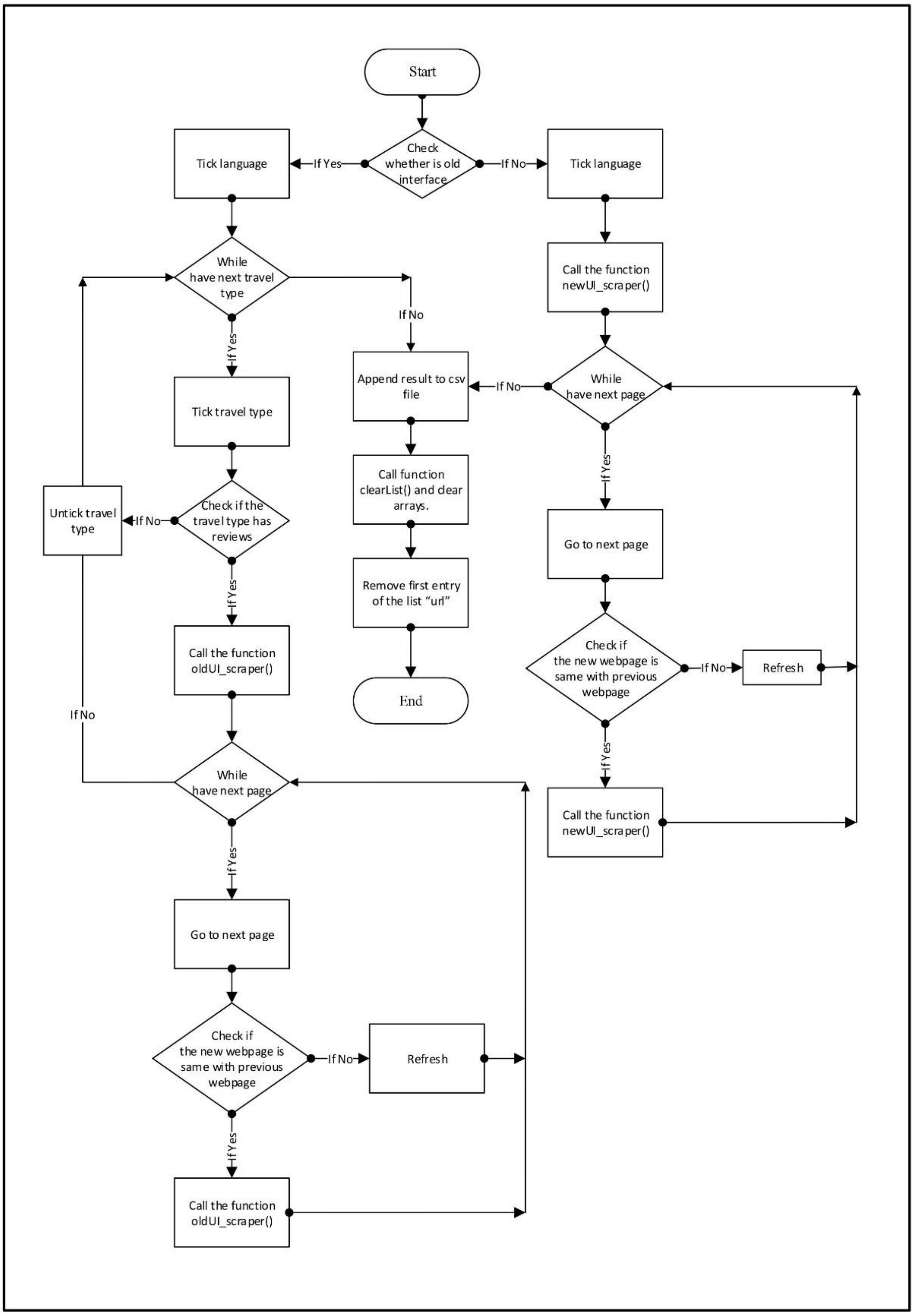

Start
Check whether is old interface
If Yes
If No
Tick language
Tick language
While have next travel type
If No
If Yes
Call the function newUI_scraper()
Append result to csv file
If No
While have next page
Tick travel type
Call function clearList() and clear arrays.
If Yes
Untick travel type
If No
Check if the travel type has reviews
Go to next page
Remove first entry of the list "url"
If Yes
Call the function oldUI_scraper()
End
Check if the new webpage is same with previous webpage
If No
Refresh
If No
If Yes
While have next page
Call the function newUI_scraper()
If Yes
Go to next page
Check if the new webpage is same with previous webpage
If No
Refresh
If Yes
Call the function oldUI_scraper()

Figure 3.2.5 Flowchart of the function executeUI() – Only available for Hotel Scraper.

This function is only available for Hotel Scraper.

Chapter 4: Discussion on System

4.1 Methodology and General Work Procedures

This project has adopted agile development methodology. The reason this methodology has been chosen is because this project is not considered as a large mission-critical project, therefore testing and efficient coding practices is more focused rather than detailed design documentation. The benefits of agile development is that it is able to build the system quickly and has the ability to change the system requirements at any point during the life of project development process. This methodology can develop a system faster is because it saved the time in defining the complete requirements and writing the design documentation. On the other hand, it is very difficult to attempt defining every requirements at the beginning of the project and also regulate the changes to the previously defined requirements, therefore the nature of agile development which can change the requirements at any point during project development is a more realistic and better approach than other methodologies.

Before starting the coding, planning phase has to be carried out so that one can have more understanding on the system that is to be developed. For example, feasibility analysis on whether the proposed system is a realistic goal in term of money and time has been done. Several articles and research papers related to this project has also been reviewed to have more understanding on the existing system such as their strengths, their weaknesses and how can this project improve from the existing system. Furthermore, requirement gathering techniques including observation on the existing system and report inspection has been conducted in order to define the requirements for the project. Besides, the software and hardware needed for the realization of the project is also been studied. After the information and resources needed to realize the project has been determined, coding phase is then being conducted. Finally, the written code has been tested to check if there is any errors and to verify if it is able to do what it is designed to before it is ready to be delivered.

4.2 Tools

For software aspects, Python programming language is used to develop the data scraper. Jupyter Notebook is chosen as the environment to write the code in Python as it is browser-based, support interactivity and has a good interface to demonstrate the code. Besides, a few Python libraries such as selenium, NumPy and pandas has been imported in order to collect data from website. NumPy and pandas are used to organize the collected data while PySimpleGUI is a Python library that has been utilized in this project to build a simple user interface for user-friendliness to those without coding background. Selenium is the chosen framework that is able to use the browser driver such as chromedriver.exe to open a browser window for accessing the target website and therefore start capturing data. In another word, selenium is the browser session itself which a set of actions has been written to do task. It is good for this project as this project will target dynamic websites which reply on javascript or AJAX to build its website content, which only browser can get the actual rendered DOM contents. Besides, it also have a good documentation support and the coding is relatively easy and understandable due to much nicer syntax.

For hardware aspects, this project required a high-end computer with a decent processor and high amount of RAM to run at a faster rate. However, since a laptop with normal specifications has been used during the development of this project, which is an i5 processor and 4GB RAM, therefore the program is running slowly in this case.

For connectivity aspects, this project required a good Internet connection to carry out its function. This is because the data scraper need to connect to the Internet for accessing the website to start collecting data. With a bad connection, the program will run very slow and in some case, the program will be forced to terminate due to inaccessible network.

4.3 User Requirements

This system is designed to be easy to use which does not required user to have any specific skills as there is a user interface which will guide the user and user will just have to run the program and it will handle the rest. However, the nature of this proposed system is actually specifically designed for the technical personnel involved in the tourism aspects because the data scraper is specifically designed to capture data from the travel website. The data collected by this system will not be useful to normal people in their daily life while it will be very useful to those who are interested to conduct a research on tourism aspect. Therefore, our main target user will be focused on data analysts, businesses, students and researchers who are involved in the tourism field.

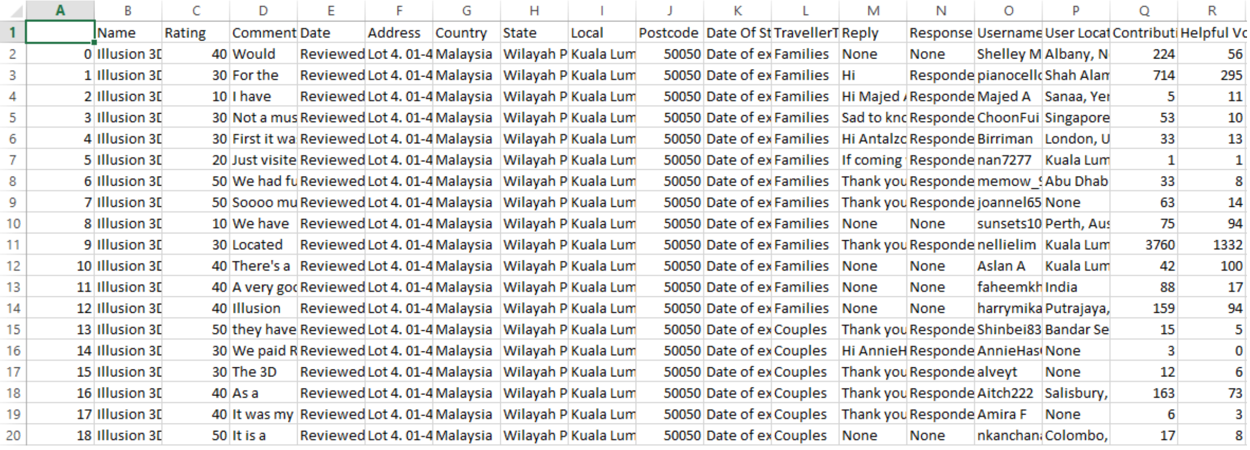
4.4 Testing Evaluation

Throughout the early development stages, it has been found that the most common reason that leads to the system scrapped incorrect data is due to logical error within the code. Since logical error will not throw out exception or error, therefore it is difficult to identify if there is any, and that is why a lot of time has been spent on reviewing the code and test it multiple times for ensuring the correctness of data.

Besides, there is one exception that often encountered during the execution of the program is "StaleElementReferenceException" which is caused by the network problem. The occurrence of this exception during testing stage of the program is mainly due to the program located the element before the webpage could be fully loaded. Therefore, the element is in stale state and further action on it such as clicking or retrieving its text will then throw out this exception. Since this exception may be thrown at random depending on the stability of the network and it is nearly impossible to predict whether the program will encounter it during each execution, thus the program is not guaranteed to execute successfully and therefore a lot of time will be wasted. Hence, this problem has inspired the requirement to have a backup and recovery mechanism which could re-run the program starting from the links that failed.

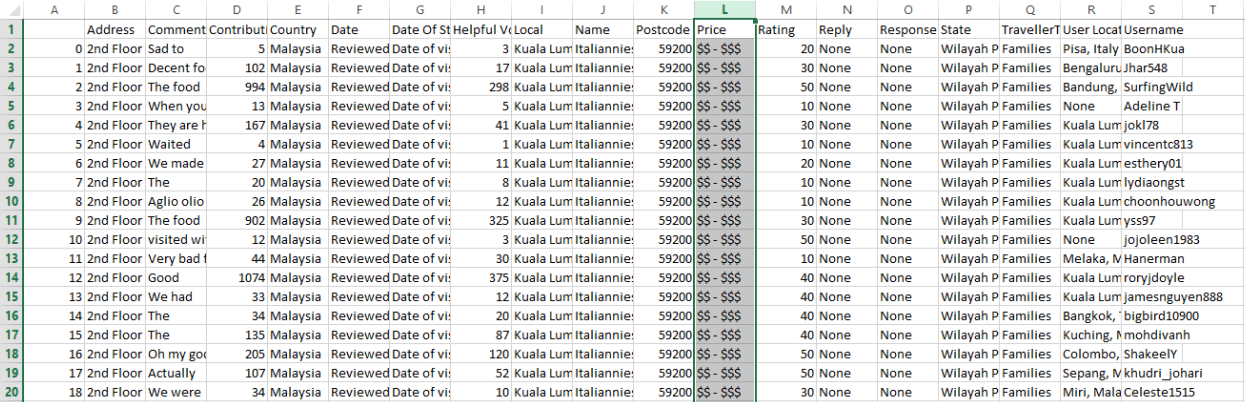
4.5 System Results

The final output of this program will be csv file(s) that consists of the scrapped data entries which each of the csv are named as “restaurant.csv”, “hotel.csv”, and “attraction.csv” to represent the categories of travel information respectively. These 3 categories of travel data are having the same numbers and types of attributes except for hotel having an additional attribute named “Ranking” which indicate the rated class of the hotel while restaurant has an additional attribute named “Price” that specify the overall price range of the restaurant’s foods. Both the “Price” and “Ranking” attribute are useful for analyzing the whether it has the impact on the positivity of the travel site such as a particular hotel’s rating and number of visitors. The figures show the sample of scrapped tourism data of each category.



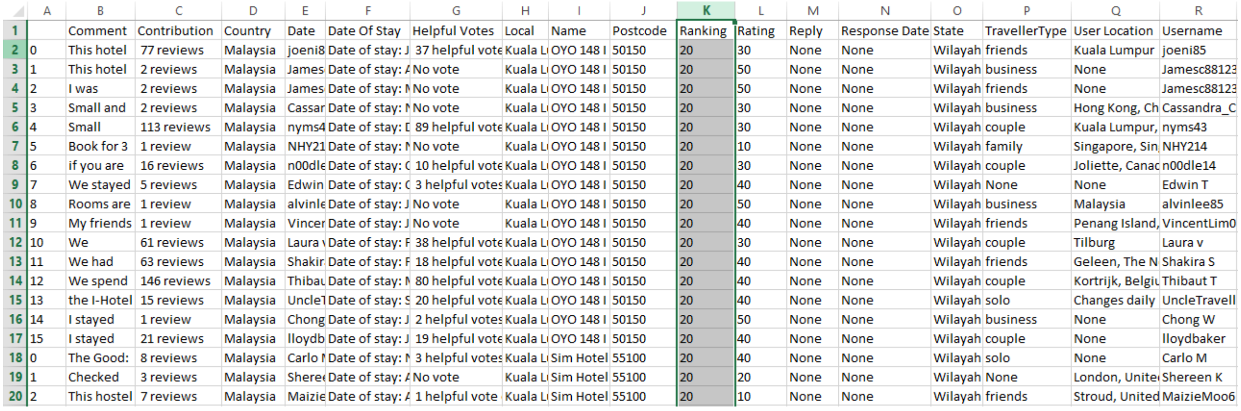
	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R
1		Name	Rating	Comment	Date	Address	Country	State	Local	Postcode	Date Of St	TravellerT	Reply	Response	Username	User Locat	Contributi	Helpful Vc
2	0	Illusion 3D	40	Would	Reviewed	Lot 4. 01-4	Malaysia	Wilayah P	Kuala Lum	50050	Date of ex	Families	None	None	Shelley M	Albany, N	224	56
3	1	Illusion 3D	30	For the	Reviewed	Lot 4. 01-4	Malaysia	Wilayah P	Kuala Lum	50050	Date of ex	Families	Hi	Responde	pianocello	Shah Alam	714	295
4	2	Illusion 3D	10	I have	Reviewed	Lot 4. 01-4	Malaysia	Wilayah P	Kuala Lum	50050	Date of ex	Families	Hi Majed A	Responde	Majed A	Sanaa, Yer	5	11
5	3	Illusion 3D	30	Not a mus	Reviewed	Lot 4. 01-4	Malaysia	Wilayah P	Kuala Lum	50050	Date of ex	Families	Sad to kno	Responde	ChoonFui	Singapore	53	10
6	4	Illusion 3D	30	First it wa	Reviewed	Lot 4. 01-4	Malaysia	Wilayah P	Kuala Lum	50050	Date of ex	Families	Hi Antalzc	Responde	Birriman	London, U	33	13
7	5	Illusion 3D	20	Just visite	Reviewed	Lot 4. 01-4	Malaysia	Wilayah P	Kuala Lum	50050	Date of ex	Families	If coming	Responde	nan7277	Kuala Lum	1	1
8	6	Illusion 3D	50	We had fu	Reviewed	Lot 4. 01-4	Malaysia	Wilayah P	Kuala Lum	50050	Date of ex	Families	Thank you	Responde	memow_	Abu Dhab	33	8
9	7	Illusion 3D	50	Soooo mu	Reviewed	Lot 4. 01-4	Malaysia	Wilayah P	Kuala Lum	50050	Date of ex	Families	Thank you	Responde	joannel65	None	63	14
10	8	Illusion 3D	10	We have	Reviewed	Lot 4. 01-4	Malaysia	Wilayah P	Kuala Lum	50050	Date of ex	Families	None	None	sunsets10	Perth, Aus	75	94
11	9	Illusion 3D	30	Located	Reviewed	Lot 4. 01-4	Malaysia	Wilayah P	Kuala Lum	50050	Date of ex	Families	Thank you	Responde	nellielim	Kuala Lum	3760	1332
12	10	Illusion 3D	40	There's a	Reviewed	Lot 4. 01-4	Malaysia	Wilayah P	Kuala Lum	50050	Date of ex	Families	None	None	Aslan A	Kuala Lum	42	100
13	11	Illusion 3D	40	A very goo	Reviewed	Lot 4. 01-4	Malaysia	Wilayah P	Kuala Lum	50050	Date of ex	Families	None	None	faheemkh	India	88	17
14	12	Illusion 3D	40	Illusion	Reviewed	Lot 4. 01-4	Malaysia	Wilayah P	Kuala Lum	50050	Date of ex	Families	None	None	harrymika	Putrajaya,	159	94
15	13	Illusion 3D	50	they have	Reviewed	Lot 4. 01-4	Malaysia	Wilayah P	Kuala Lum	50050	Date of ex	Couples	Thank you	Responde	Shinbei83	Bandar Se	15	5
16	14	Illusion 3D	30	We paid R	Reviewed	Lot 4. 01-4	Malaysia	Wilayah P	Kuala Lum	50050	Date of ex	Couples	Hi AnnieH	Responde	AnnieHas	None	3	0
17	15	Illusion 3D	30	The 3D	Reviewed	Lot 4. 01-4	Malaysia	Wilayah P	Kuala Lum	50050	Date of ex	Couples	Thank you	Responde	alveyt	None	12	6
18	16	Illusion 3D	40	As a	Reviewed	Lot 4. 01-4	Malaysia	Wilayah P	Kuala Lum	50050	Date of ex	Couples	Thank you	Responde	Aitch222	Salisbury,	163	73
19	17	Illusion 3D	40	It was my	Reviewed	Lot 4. 01-4	Malaysia	Wilayah P	Kuala Lum	50050	Date of ex	Couples	Thank you	Responde	Amira F	None	6	3
20	18	Illusion 3D	50	It is a	Reviewed	Lot 4. 01-4	Malaysia	Wilayah P	Kuala Lum	50050	Date of ex	Couples	None	None	nkanchan	Colombo,	17	8

Figure 4.5.1 Full travel data under attraction.



	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T
1		Address	Comment	Contributi	Country	Date	Date Of St	Helpful Vc	Local	Name	Postcode	Price	Rating	Reply	Response	State	TravellerT	User Locat	Username	
2	0	2nd Floor	Sad to	5	Malaysia	Reviewed	Date of vi	3	Kuala Lum	Italiannie	59200	$$ - $$$	20	None	None	Wilayah P	Families	Pisa, Italy	BoonHKua	
3	1	2nd Floor	Decent fo	102	Malaysia	Reviewed	Date of vi	17	Kuala Lum	Italiannie	59200	$$ - $$$	30	None	None	Wilayah P	Families	Bengaluru	Jhar548	
4	2	2nd Floor	The food	994	Malaysia	Reviewed	Date of vi	298	Kuala Lum	Italiannie	59200	$$ - $$$	50	None	None	Wilayah P	Families	Bandung,	SurfingWild	
5	3	2nd Floor	When you	13	Malaysia	Reviewed	Date of vi	5	Kuala Lum	Italiannie	59200	$$ - $$$	10	None	None	Wilayah P	Families	None	Adeline T	
6	4	2nd Floor	They are h	167	Malaysia	Reviewed	Date of vi	41	Kuala Lum	Italiannie	59200	$$ - $$$	30	None	None	Wilayah P	Families	Kuala Lum	jokl78	
7	5	2nd Floor	Waited	4	Malaysia	Reviewed	Date of vi	1	Kuala Lum	Italiannie	59200	$$ - $$$	10	None	None	Wilayah P	Families	Kuala Lum	vincentc813	
8	6	2nd Floor	We made	27	Malaysia	Reviewed	Date of vi	11	Kuala Lum	Italiannie	59200	$$ - $$$	20	None	None	Wilayah P	Families	Kuala Lum	esthery01	
9	7	2nd Floor	The	20	Malaysia	Reviewed	Date of vi	8	Kuala Lum	Italiannie	59200	$$ - $$$	10	None	None	Wilayah P	Families	Kuala Lum	lydiaongst	
10	8	2nd Floor	Aglio olio	26	Malaysia	Reviewed	Date of vi	12	Kuala Lum	Italiannie	59200	$$ - $$$	10	None	None	Wilayah P	Families	Kuala Lum	choonhouwong	
11	9	2nd Floor	The food	902	Malaysia	Reviewed	Date of vi	325	Kuala Lum	Italiannie	59200	$$ - $$$	30	None	None	Wilayah P	Families	Kuala Lum	yss97	
12	10	2nd Floor	visited wi	12	Malaysia	Reviewed	Date of vi	3	Kuala Lum	Italiannie	59200	$$ - $$$	50	None	None	Wilayah P	Families	None	jojoleen1983	
13	11	2nd Floor	Very bad f	44	Malaysia	Reviewed	Date of vi	30	Kuala Lum	Italiannie	59200	$$ - $$$	10	None	None	Wilayah P	Families	Melaka, M	Hanerman	
14	12	2nd Floor	Good	1074	Malaysia	Reviewed	Date of vi	375	Kuala Lum	Italiannie	59200	$$ - $$$	40	None	None	Wilayah P	Families	Kuala Lum	roryjdoyle	
15	13	2nd Floor	We had	33	Malaysia	Reviewed	Date of vi	12	Kuala Lum	Italiannie	59200	$$ - $$$	40	None	None	Wilayah P	Families	Kuala Lum	jamesnguyen888	
16	14	2nd Floor	The	34	Malaysia	Reviewed	Date of vi	20	Kuala Lum	Italiannie	59200	$$ - $$$	40	None	None	Wilayah P	Families	Bangkok,	bigbird10900	
17	15	2nd Floor	The	135	Malaysia	Reviewed	Date of vi	87	Kuala Lum	Italiannie	59200	$$ - $$$	40	None	None	Wilayah P	Families	Kuching, M	mohdivanh	
18	16	2nd Floor	Oh my goo	205	Malaysia	Reviewed	Date of vi	120	Kuala Lum	Italiannie	59200	$$ - $$$	50	None	None	Wilayah P	Families	Colombo,	ShakeelY	
19	17	2nd Floor	Actually	107	Malaysia	Reviewed	Date of vi	52	Kuala Lum	Italiannie	59200	$$ - $$$	50	None	None	Wilayah P	Families	Sepang, M	khudri_johari	
20	18	2nd Floor	We were	34	Malaysia	Reviewed	Date of vi	10	Kuala Lum	Italiannie	59200	$$ - $$$	30	None	None	Wilayah P	Families	Miri, Mala	Celeste1515	

Figure 4.5.2 Full travel data under restaurant. Highlighted column is its additional attribute.



	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R
1		Comment	Contribution	Country	Date	Date Of Stay	Helpful Votes	Local	Name	Postcode	Ranking	Rating	Reply	Response Date	State	TravellerType	User Location	Username
2	0	This hotel	77 reviews	Malaysia	joeni8	Date of stay: J	37 helpful vote	Kuala L	OYO 148 I	50150	20	30	None	None	Wilayah	friends	Kuala Lumpur	joeni85
3	1	This hotel	2 reviews	Malaysia	James	Date of stay: A	No vote	Kuala L	OYO 148 I	50150	20	50	None	None	Wilayah	business	None	Jamesc88123
4	2	I was	2 reviews	Malaysia	James	Date of stay: M	No vote	Kuala L	OYO 148 I	50150	20	50	None	None	Wilayah	friends	None	Jamesc88123
5	3	Small and	2 reviews	Malaysia	Cassar	Date of stay: M	No vote	Kuala L	OYO 148 I	50150	20	30	None	None	Wilayah	business	Hong Kong, Ch	Cassandra_C
6	4	Small	113 reviews	Malaysia	nyms4	Date of stay: D	89 helpful vote	Kuala L	OYO 148 I	50150	20	30	None	None	Wilayah	couple	Kuala Lumpur,	nyms43
7	5	Book for 3	1 review	Malaysia	NHY21	Date of stay: M	No vote	Kuala L	OYO 148 I	50150	20	10	None	None	Wilayah	family	Singapore, Sin	NHY214
8	6	if you are	16 reviews	Malaysia	n00dle	Date of stay: O	10 helpful vote	Kuala L	OYO 148 I	50150	20	30	None	None	Wilayah	couple	Joliette, Canac	n00dle14
9	7	We stayed	5 reviews	Malaysia	Edwin	Date of stay: O	3 helpful votes	Kuala L	OYO 148 I	50150	20	40	None	None	Wilayah	None	None	Edwin T
10	8	Rooms are	1 review	Malaysia	alvinle	Date of stay: J	No vote	Kuala L	OYO 148 I	50150	20	50	None	None	Wilayah	business	Malaysia	alvinlee85
11	9	My friends	1 review	Malaysia	Vincer	Date of stay: J	No vote	Kuala L	OYO 148 I	50150	20	40	None	None	Wilayah	friends	Penang Island,	VincentLim0
12	10	We	61 reviews	Malaysia	Laura v	Date of stay: F	38 helpful vote	Kuala L	OYO 148 I	50150	20	30	None	None	Wilayah	couple	Tilburg	Laura v
13	11	We had	63 reviews	Malaysia	Shakir	Date of stay: F	18 helpful vote	Kuala L	OYO 148 I	50150	20	40	None	None	Wilayah	friends	Geleen, The N	Shakira S
14	12	We spend	146 reviews	Malaysia	Thibau	Date of stay: M	80 helpful vote	Kuala L	OYO 148 I	50150	20	40	None	None	Wilayah	couple	Kortrijk, Belgiu	Thibaut T
15	13	the I-Hotel	15 reviews	Malaysia	UncleT	Date of stay: S	20 helpful vote	Kuala L	OYO 148 I	50150	20	40	None	None	Wilayah	solo	Changes daily	UncleTravell
16	14	I stayed	1 review	Malaysia	Chong	Date of stay: J	2 helpful votes	Kuala L	OYO 148 I	50150	20	50	None	None	Wilayah	business	None	Chong W
17	15	I stayed	21 reviews	Malaysia	lloydb	Date of stay: J	19 helpful vote	Kuala L	OYO 148 I	50150	20	40	None	None	Wilayah	couple	None	lloydbaker
18	0	The Good:	8 reviews	Malaysia	Carlo M	Date of stay: M	3 helpful votes	Kuala L	Sim Hotel	55100	20	40	None	None	Wilayah	solo	None	Carlo M
19	1	Checked	3 reviews	Malaysia	Sheree	Date of stay: A	No vote	Kuala L	Sim Hotel	55100	20	20	None	None	Wilayah	None	London, Unite	Shereen K
20	2	This hostel	7 reviews	Malaysia	Maizie	Date of stay: A	1 helpful vote	Kuala L	Sim Hotel	55100	20	10	None	None	Wilayah	friends	Stroud, United	MaizieMoo6

Figure 4.5.3 Full travel data under hotel. Highlighted column is its additional attribute.

As mentioned above, there are a total of 17 shared attributes among the 3 categories of travel data. The scrapped attributes can be further categorized into travel site information, travel site management's reply contents, website user's basic information, and website user's review contents. The attributes, the explanation on the attributes, and reasons for collecting the attributes will discussed in the tables below.

Attribute Name	**Explanation**
1. Reply	• Specify if there is official reply from management and its content. This attribute can be further analyzed with text sentiment analysis and to find out whether it has impact on the travel site reputation. For example, a well-organized travel site that often reply to the reviews may have a better overall rating. • Output "None" if not available.
2. Response Date	• Specify the response date from the travel site management. This attribute can be further analyzed to see whether the quickness of response time will contribute to the good overall rating of the travel site. For example, the quicker the reply is made, the reviewer may feel respected and is more likely to give a good rating or travel there again. • Eg: Responded 1 February 2019 • Output "None" if not available.

Table 4.5.1 Travel site management's reply contents.

Attribute Name	**Explanation**
1. Name	• Specify the name of the travel site. • Eg: Petronas Twin Towers
2. Address	• Specify the address of the travel site. This information can be converted to latitude and longitude so that the location can be plotted on the map for visualization purpose. • Eg: 1 Jalan Imbi | Level 5 & 7 • Output null value if not available.
3. Country	• Specify which country the travel site is located. This attribute is useful for data analytics purpose in analyzing the tourism statistics by countries. • Eg: Malaysia
4. State	• Specify which state the travel site is located. This attribute is useful for data analytics purpose in analyzing the tourism statistics by states. • Eg: Perak
5. Local	• Specify which city or area the travel site is located. This attribute is useful for data analytics purpose in analyzing the tourism statistics by city or area. • Eg: Kampar
6. Postcode	• Specify postcode of the travel site. This attribute is useful when the address of the travel site is not available, then it can be combined with other attribute such as “Name” and “State” to locate the travel site on the map. • Eg: 31900 • Output “None” if not available.

Table 4.5.2 Travel site information.

Attribute Name	**Explanation**
1. Username	• Specify the name of the reviewer. • Eg: pianocello • Output "A travel website Member" if hidden profile.
2. User Location	• Specify the location of the reviewer. Analysis can be done on this attribute to find out what is the region that most visitors came from to visit the travel site. • Eg: Albany, New Zealand • Output "None" if not available.
3. Contributions	• Specify the total number of reviews made by the website user. It has a huge impact on the credibility of the website user's reviews. • Eg: 77 reviews
4. Helpful Votes	• Specify the total number of up-votes the website user has received. As the vote is given by the other user if they found the review to be accurate and helpful, therefore this attribute is also expected to contribute much to the trustworthiness and credibility of the website user's reviews. • Eg: 37 helpful votes and likes • Output "0" or "No vote" if not available.

Table 4.5.3 Website user's basic information.

Attribute Name	**Explanation**
1. Rating	• Specify the reviewer's rating score on the travel site. This attribute describe the overall satisfaction of the reviewer on the particular travel site. • Eg: 40
2. Comment	• Specify the text comment of the review. This attribute can be analyzed with text sentiment analysis to verify if the sentiment score of the comment fit the rating score. For example, a review with good comment but bad rating can be deemed as untrusted data.
3. Date	• Specify the date the review was posted. This attribute can be further break down to analyze the data by months or by years. • Eg: Reviewed 31 December 2016
4. Date Of Stay	• Specify the date that the reviewer visited the travel site. The relationship between this attribute and the date the review was made can be analyzed to find out its impact on the travel site reputation. For example, will the visitor more likely to rate the travel site immediately when they are satisfied with the services provided by the travel site? • Eg: Date of experience: March 2019 • Output null value if not available.
5. TravellerType	• Specify the travel type of the reviewer during the visitation. For example, analysis can be done on this attribute to find out which type of traveler the travel site attracted the most. • Eg: Families • Output "None" if not available.

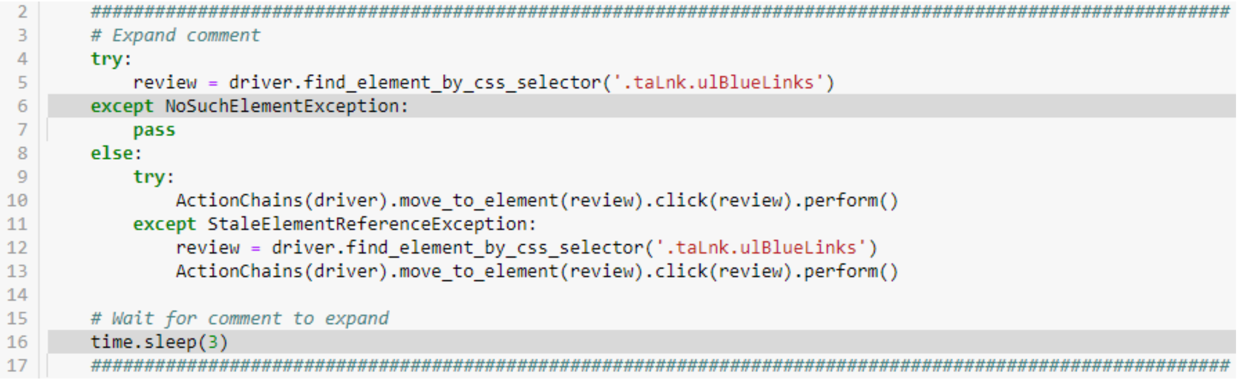
Table 4.5.4 Website user's review contents.

4.6 Legal and Ethical Issues

As the nature of the system is a bot, therefore one of the most frequently asked questions on the web scraper are the legibility and ethical issues. This matter has been much emphasize in this project and a thorough study has been performed to find out the answer to the question. As a result of the research, the development of the web scraper itself is completely legal and doesn't have any ethical issues. Besides, the deployment of the web scraper to retrieve data from the Internet website is also considered legal as the data available on the website is public data which does not contain private data such as users' sensitive information. However, this action might be breaking the rules of certain website due to the nature of the program. Some website has stated clearly in their terms and conditions that bots are not allowed to be deployed for interacting with the website, this is mainly because of the website owner are aware of the potential problem of server overloading due to large traffic created by the bots. In conclusion, deploying a web scraper to retrieve data from a website is considered legal unless stated in the website's terms and conditions while it is still a debatable topic whether it is ethical or not.

Chapter 5: Implementation Challenges

During the project development, several challenges and problems has occurred, which one of these problems is the network connection issue. A critical problem has been found which is proven to be able to cause the inaccuracy of data. For example, this program has a feature which is to click the “More” button if there is any in order to expand the full text, so that accuracy of data collected can be achieved. However, with the low bandwidth network connection, sometime the website will just capture the unexpanded data before the “More” button is clicked. Therefore, there is a need to modify the code to be compatible with the low bandwidth network connection by adding a command which is served to pause the execution of the program for 3 seconds before the next instructions is being executed. Another example is that the webpage is not fully loaded yet and the program is trying to locate the html element within the page, therefore an error “NoSuchElementException” will be thrown as the program is unable to locate the html element. Therefore a try-catch exception handling is implemented to cope with the problem.



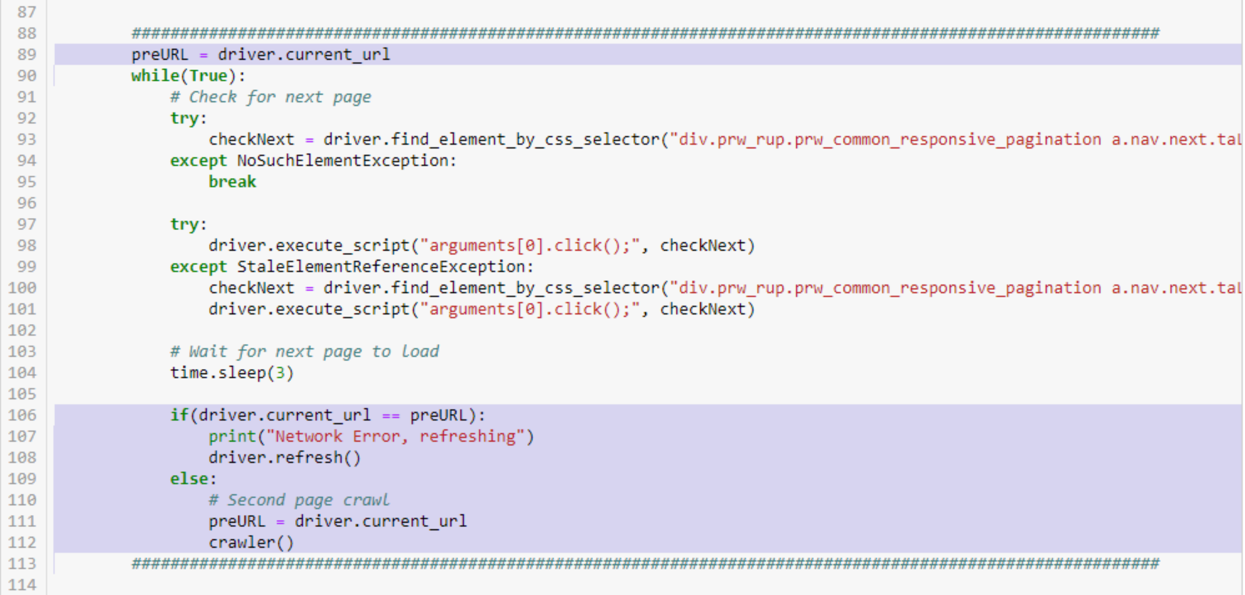
```
##################################################################################################################
# Expand comment
try:
    review = driver.find_element_by_css_selector('.taLnk.ulBlueLinks')
except NoSuchElementException:
    pass
else:
    try:
        ActionChains(driver).move_to_element(review).click(review).perform()
    except StaleElementReferenceException:
        review = driver.find_element_by_css_selector('.taLnk.ulBlueLinks')
        ActionChains(driver).move_to_element(review).click(review).perform()

# Wait for comment to expand
time.sleep(3)
##################################################################################################################
```

Figure 5.1 Solution associated with connection issue.

However, the program execution pausing method has also brought a problem which it has caused the execution time of the program to be way too long. For example, one webpage needs 1.2 second to be loaded fully, but the program has to be paused for 3 seconds, therefore 1.8 seconds is already wasted. This might seems like a small number, but if there are 100,000 webpages to be scrapped and there will be a resultant of 180,000 seconds being unexploited. To solve this problem, the program can simply continue executing once the page is fully loaded. There is a selenium function which is known as “WebDriverWait” is therefore being exploited into the program in order to pause the program until the intended element is loaded. However, some part must still use the pause method as some element will always be present in the DOM whether actions are applied or not, therefore the wait method will not work. The syntax of WebDriverWait is as shown below.

```
WebDriverWait(driver,5).until(EC.presence_of_element_located((By.CSS_SELECTOR,’ ’)))
```

Besides, an error regarding network connectivity has also been spotted which is the program will keep on scraping the same webpage over and over when the network is disconnected. This is usually happen when the network is disconnected when the program is in the middle of scraping a webpage, and after the webpage is fully scrapped, the program will click into the next webpage and continue scrapping. However, due to network disconnection, the webpage did not load into the next page and the program does not know that the webpage did not update. Thus, a verification on the current page and previous scrapped page is implemented so that when both page are the same, the program triggers the refresh button of the browser to re-scrape the webpage again.



```

        ############################################################################################################
        preURL = driver.current_url
        while(True):
            # Check for next page
            try:
                checkNext = driver.find_element_by_css_selector("div.prw_rup.prw_common_responsive_pagination a.nav.next.tal
            except NoSuchElementException:
                break

            try:
                driver.execute_script("arguments[0].click();", checkNext)
            except StaleElementReferenceException:
                checkNext = driver.find_element_by_css_selector("div.prw_rup.prw_common_responsive_pagination a.nav.next.tal
                driver.execute_script("arguments[0].click();", checkNext)

            # Wait for next page to load
            time.sleep(3)

            if(driver.current_url == preURL):
                print("Network Error, refreshing")
                driver.refresh()
            else:
                # Second page crawl
                preURL = driver.current_url
                crawler()
        ############################################################################################################

```

Figure 5.2 Verification on the new page and previous page.

Another problem caused by network issue is “StaleElementReferenceException”. To understand the problem, one must first have the concept of Document Object Model (DOM). The diagram below shows the general structure of DOM.

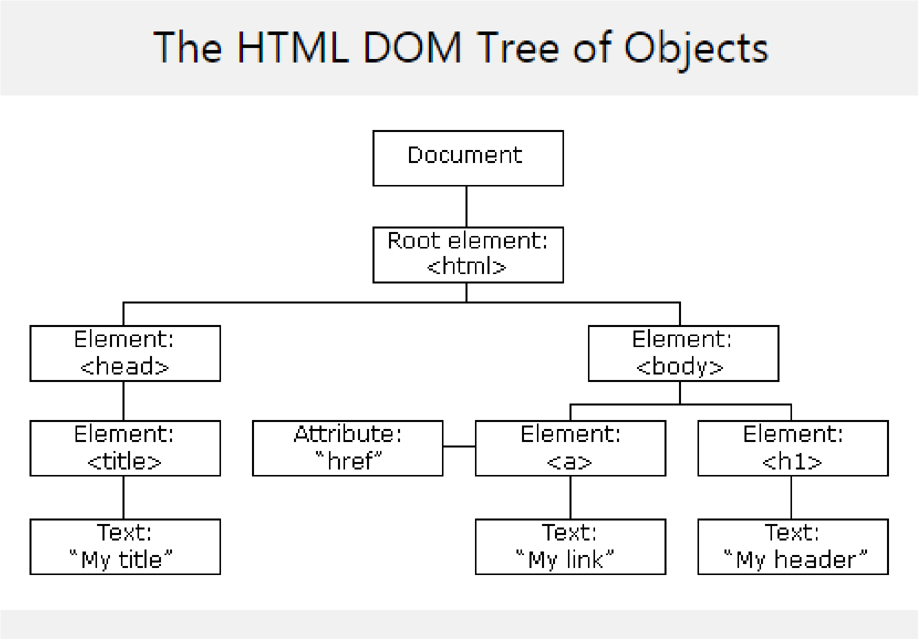
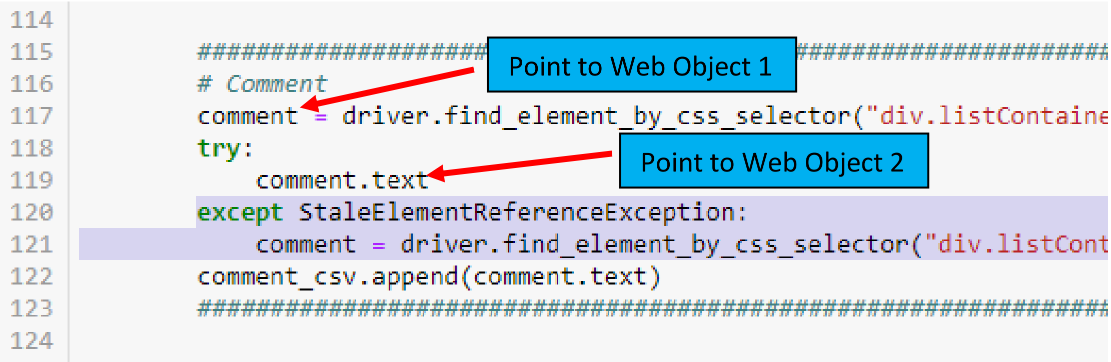


Figure 5.3 HTML DOM Tree of Objects (Anon., n.d.).

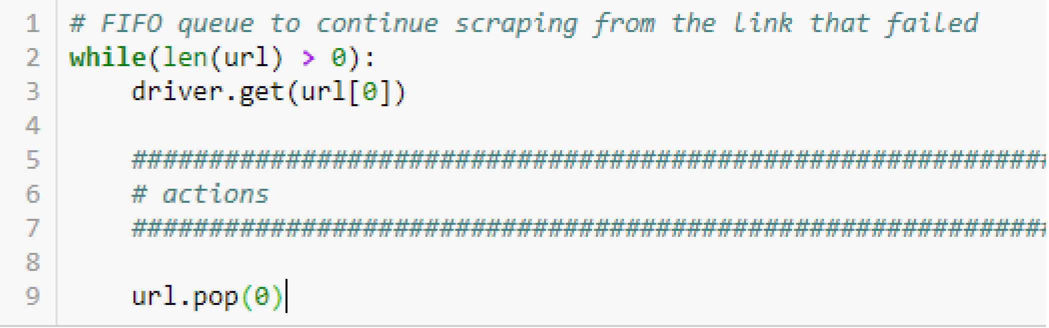
DOM is a logical tree document that allows programs and scripts to access and update its content, structure, and style dynamically. When a webpage is loaded and a target web element is being initiated, there could be a chance that changes can still be applied to the DOM and the changes made to the DOM will trigger a new web object. Therefore, when user located a web element while the DOM is still being updated, the located web element which is still pointing to the old web object will be stale, then this exception will be thrown once the DOM finished update and action such as extracting text or clicking is being performed to the old web object. For example, a user located a web element and clicks on it on the same page will get stale element reference exception if did not relocated the web element from the latest web object after page refresh. The figure below show the general understanding of the exception and the temporal solution (exception handling) to deal with it.



```

        ####################                          ######################
        # Comment
        comment = driver.find_element_by_css_selector("div.listContaine
        try:
            comment.text
        except StaleElementReferenceException:
            comment = driver.find_element_by_css_selector("div.listCont
        comment_csv.append(comment.text)
        ###############################################################

```

Figure 5.4 Stale Element Explanation

However, there is still no guarantee that stale element reference exception will not occur again unless we pause the program long enough for the webpage to be fully loaded, but we do not want that because too much time will be wasted as stated above. It is a trade-off between system reliability and system performance, neither choice can be perfect for this project. Since we will be expecting errors during execution time, then a recovery procedure must be present. The code has been modified in a way such that the program is able to continue scraping from the link that encountered errors, therefore the queue data structure which has a First-In-First-Out (FIFO) manner would be perfect. The figure below shows the implementation of the recovery method.



```
# FIFO queue to continue scraping from the link that failed
while(len(url) > 0):
    driver.get(url[0])

    ###########################################################
    # actions
    ###########################################################

    url.pop(0)
```

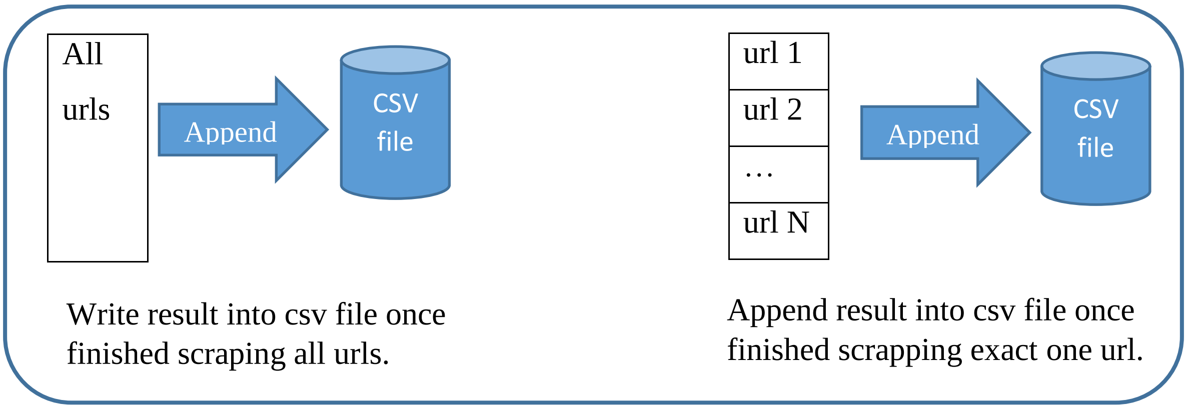
Figure 5.5 Queue data structure for recovery.

Furthermore, the web driver need to get the links in order to start scrapping the data of the webpages. Therefore, the function insertURL() is responsible for obtaining a list of links from the base URL provided by the user. However, the behaviour of the base URL webpage’s content being updated is unlike the links, it just directly make changes to the DOM to update the contents while the links is to load into a new DOM. Therefore, the wait method cannot be used as the web element is always present and instead pause method is used. The problem with pause method is that the program will obtain the same links if the DOM is not updated with the new contents when the specified pause time is exceeded, this is because the program will continue its execution after its specified time regardless of the update status of the DOM. Thus, duplication of the links are suspected and a simple solution to remove the duplication is introduced with just one line of code as shown below.

```
url = list(set(url))
```

- Transform the list into tuple and then back to list as the nature of tuple will automatically remove the duplicated entries.

Other than that, bottleneck on the hardware has been found as well. The computer with a weaker processor and a smaller RAM size will take more times to finish compiling the program. This has caused much trouble and bad system performance during the development of this project as the time needed to complete running the program will be quite long, which can take up to hours depending on the amount of data to be scrapped from the website. This has reduced the productivity of the project development as the time that has been wasted could be used to add another features or solve the coding problem instead of just waiting for the result of the compiling program. For example, a computer with a decent hardware component would take 30 minutes to finish running the program, but a computer with outdated hardware component might take more than 1 hour to run the program. Due to there is a shortage of budget for decent hardware installation, therefore program performance optimization has been implemented in the coding. The optimization being done here is by appending the result into csv file part by part and immediately release the appended memories to free up the occupied resources that is no longer needed. The figures below show the concept between write whole results into csv file and append part result into csv file.

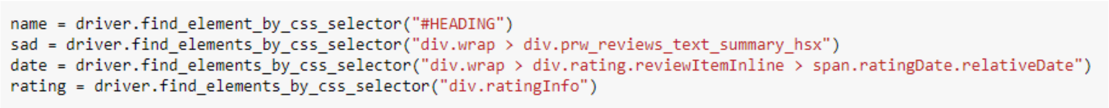


Write result into csv file once finished scraping all urls.

Append result into csv file once finished scrapping exact one url.

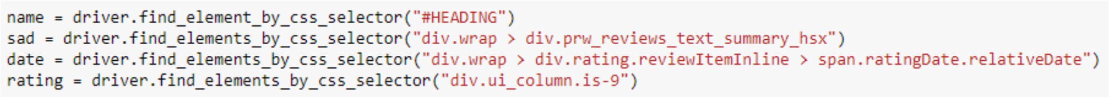
Figure 5.6 Optimization on program performance.

Lastly, this project has also encountered some technical problems. One of these problems is the difference code structure of websites. The nature of this project is to inspect the HTML code of the target website and capture that particular element. However, each website has its own HTML coding structures. Moreover, even a single website will have different HTML code structures for each category of tourism information. For example, “Restaurants”, “Hotels” and “Things to do” each has its own code structure. Therefore, a temporal solution has been adopted which is to create 3 separate programs for each of them.



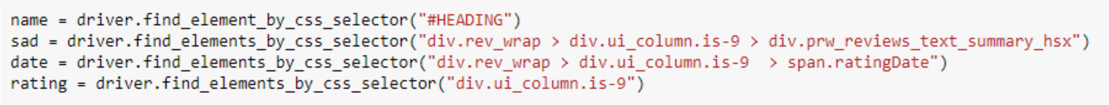
```
name = driver.find_element_by_css_selector("#HEADING")
sad = driver.find_elements_by_css_selector("div.wrap > div.prw_reviews_text_summary_hsx")
date = driver.find_elements_by_css_selector("div.wrap > div.rating.reviewItemInline > span.ratingDate.relativeDate")
rating = driver.find_elements_by_css_selector("div.ratingInfo")
```

Figure 5.7 Sample of HTML code structure of Attraction.



```
name = driver.find_element_by_css_selector("#HEADING")
sad = driver.find_elements_by_css_selector("div.wrap > div.prw_reviews_text_summary_hsx")
date = driver.find_elements_by_css_selector("div.wrap > div.rating.reviewItemInline > span.ratingDate.relativeDate")
rating = driver.find_elements_by_css_selector("div.ui_column.is-9")
```

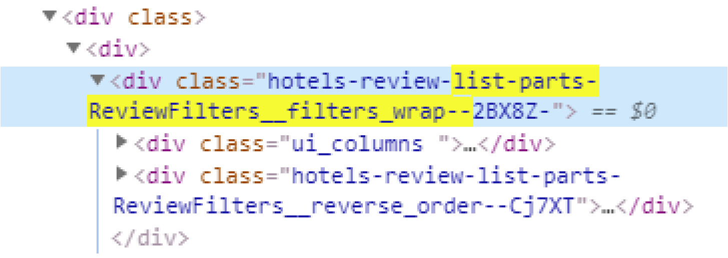
Figure 5.8 Sample of HTML code structure of Restaurants.



```
name = driver.find_element_by_css_selector("#HEADING")
sad = driver.find_elements_by_css_selector("div.rev_wrap > div.ui_column.is-9 > div.prw_reviews_text_summary_hsx")
date = driver.find_elements_by_css_selector("div.rev_wrap > div.ui_column.is-9  > span.ratingDate")
rating = driver.find_elements_by_css_selector("div.ui_column.is-9")
```

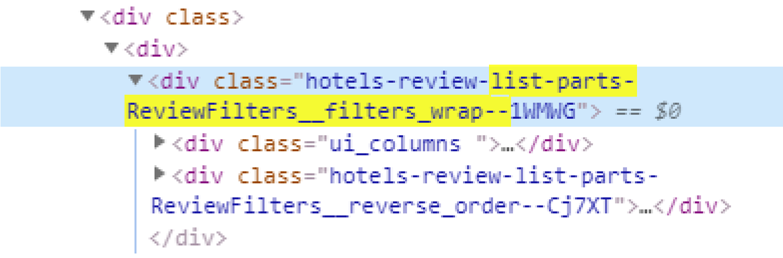
Figure 5.9 Sample of HTML code structure of Accommodation.

In addition, the html structure and its web elements are always changing and updating, therefore a maintenance routine is needed in order for this program to function as intend all the time. The figures below show the comparison before and after website update.



```
▼<div class>
  ▼<div>
    ▼<div class="hotels-review-list-parts-
    ReviewFilters__filters_wrap--2BX8Z-"> == $0
      ▶<div class="ui_columns ">…</div>
      ▶<div class="hotels-review-list-parts-
      ReviewFilters__reverse_order--Cj7XT">…</div>
      </div>
```

Figure 5.10 Old web element name.



```
▼<div class>
  ▼<div>
    ▼<div class="hotels-review-list-parts-
    ReviewFilters__filters_wrap--1WMWG"> == $0
      ▶<div class="ui_columns ">…</div>
      ▶<div class="hotels-review-list-parts-
      ReviewFilters__reverse_order--Cj7XT">…</div>
      </div>
```

Figure 5.11 Updated web element name.

For example, it has been noticed that there is an often changes or update to the class name of the web elements under the webpages of the hotel, therefore a sub-string matching logic has been adopted in response to the often updated webpage. The code below show the However, this may not work in the webpage of travel categories such as restaurant and attraction due to there is not an identifiable pattern in the update. The

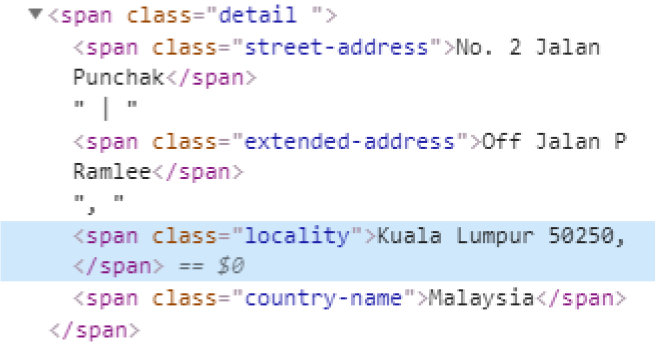
```
div.hotels-review-list-parts-ReviewFilters__filters_wrap--1WMWG
```

- ➢ Previous method to locate web elements.

```
div[class^='hotels-review-list-parts-ReviewFilters__filters_wrap--']
```

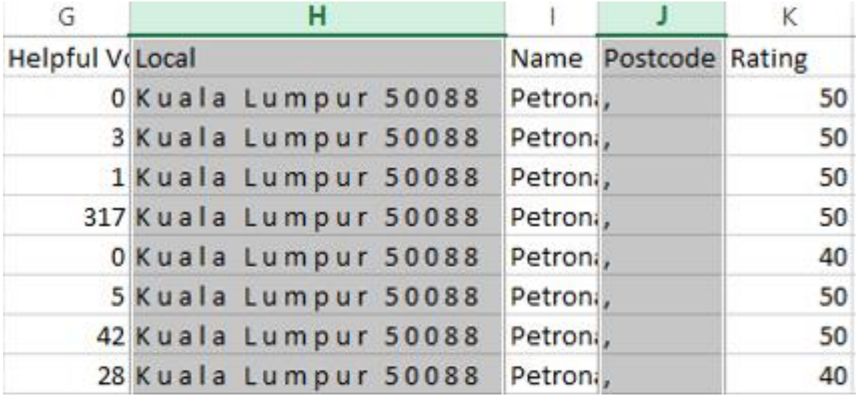
- ➢ Sub-string matching method to locate web elements.
- ➢ "^" matches class name starts with sub-string '___'
- ➢ Another sub-string matching variances are "$" that matches string ends with sub-string '___' and "*" matches string containing sub-string '___'.

Besides, another technical problem faced is the scrapped data is not as expected as shown in the figure 5.13. The figure shows the data within the column "Local" is expected to be Kuala Lumpur while the column "Postcode" is expected to be 50088. This is because the data from these 2 columns are split from the exactly same web element (as shown in figure) through data pre-processing. However, it is difficult to write a code that can fit well to various pattern of scrapped data. For example, the previous pre-processing code works well for Kampar with the web element text “Kampar 31900,” but not for Kuala Lumpur with the web element text “Kuala Lumpur 50250,” or “Kuala Lumpur,”. Therefore, research has been conducted to find out the code that is able to fit well to all possible pattern of the scrapped data content, and the result of the research suggest that regular expression is a good option in dealing with patterns. The implementation of the scrapped data pre-processing is shown in the figure 5.14.



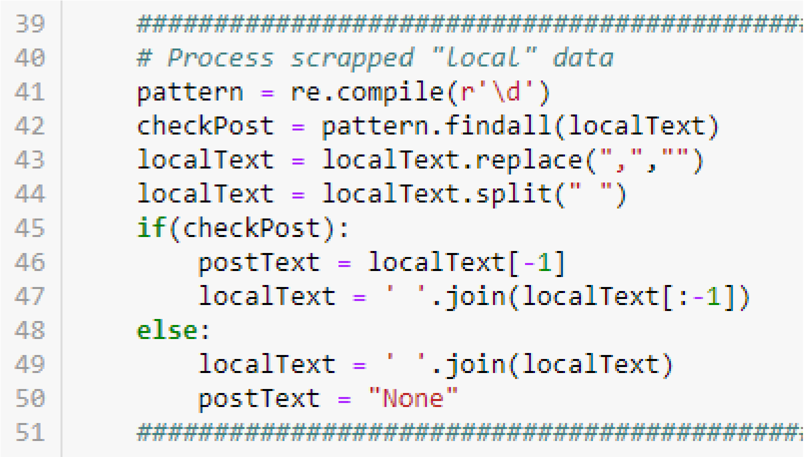
```
▼<span class="detail ">
    <span class="street-address">No. 2 Jalan
    Punchak</span>
    " | "
    <span class="extended-address">Off Jalan P
    Ramlee</span>
    ", "
    <span class="locality">Kuala Lumpur 50250,
    </span> == $0
    <span class="country-name">Malaysia</span>
  </span>
```

Figure 5.12 Local and Postcode is scrapped from this web element.



G	H	I	J	K
Helpful V	Local	Name	Postcode	Rating
0	Kuala Lumpur 50088	Petron	,	50
3	Kuala Lumpur 50088	Petron	,	50
1	Kuala Lumpur 50088	Petron	,	50
317	Kuala Lumpur 50088	Petron	,	50
0	Kuala Lumpur 50088	Petron	,	40
5	Kuala Lumpur 50088	Petron	,	50
42	Kuala Lumpur 50088	Petron	,	50
28	Kuala Lumpur 50088	Petron	,	40

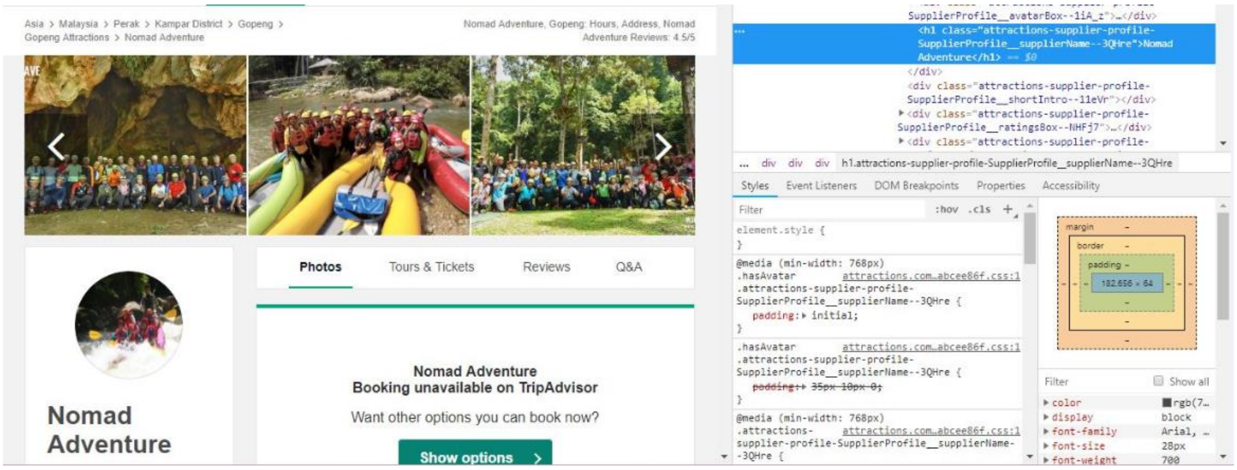
Figure 5.13 Scrapped data not as expected.



```python
    ###########################################
    # Process scrapped "local" data
    pattern = re.compile(r'\d')
    checkPost = pattern.findall(localText)
    localText = localText.replace(",","")
    localText = localText.split(" ")
    if(checkPost):
        postText = localText[-1]
        localText = ' '.join(localText[:-1])
    else:
        localText = ' '.join(localText)
        postText = "None"
    ###########################################
```

Figure 5.14 Pre-process data.

Furthermore, some webpage may contain alternate interface which is triggered in a random manner. For example, hotel webpage has two interface which is the main one as shown in figure and the alternate one has the interface similar to attraction, while attraction has 2 interface as well which the alternate interface is shown in the figure 5.15. Therefore, there is a requirement to implement another set of logic for the alternate interface and also re-locate the web element as the html structure for the alternate interface is different. Sometimes, there will be some data cannot be scrapped as they are not available in the alternate interface as shown in figure. Due to the fact that the interface is triggered in a random manner, therefore it is very hard to inspect its html structure for writing a set of logic for the alternate interface and also testing it. The only explainable theory behind this abnormal behaviour is that there exists several server, which the one server will be chosen as the main server which provide the provide the main interface while another server may be act as a backup server or secondary server that can only be accessed under some condition. However, this is solely an inference as we may never know whether the theory is correct unless we have the information on the system architecture of the website.



Asia > Malaysia > Perak > Kampar District > Gopeng > Gopeng Attractions > Nomad Adventure

Nomad Adventure, Gopeng: Hours, Address, Nomad Adventure Reviews: 4.5/5

Photos | Tours & Tickets | Reviews | Q&A

Nomad Adventure

Nomad Adventure
Booking unavailable on TripAdvisor

Want other options you can book now?

Show options >

```
SupplierProfile__avatarBox--1iA_z">…</div>
  <h1 class="attractions-supplier-profile-
  SupplierProfile__supplierName--3QHre">Nomad
  Adventure</h1> == $0
</div>
<div class="attractions-supplier-profile-
SupplierProfile__shortIntro--1leVr"></div>
▸<div class="attractions-supplier-profile-
SupplierProfile__ratingsBox--NHFj7">…</div>
▸<div class="attractions-supplier-profile-
```

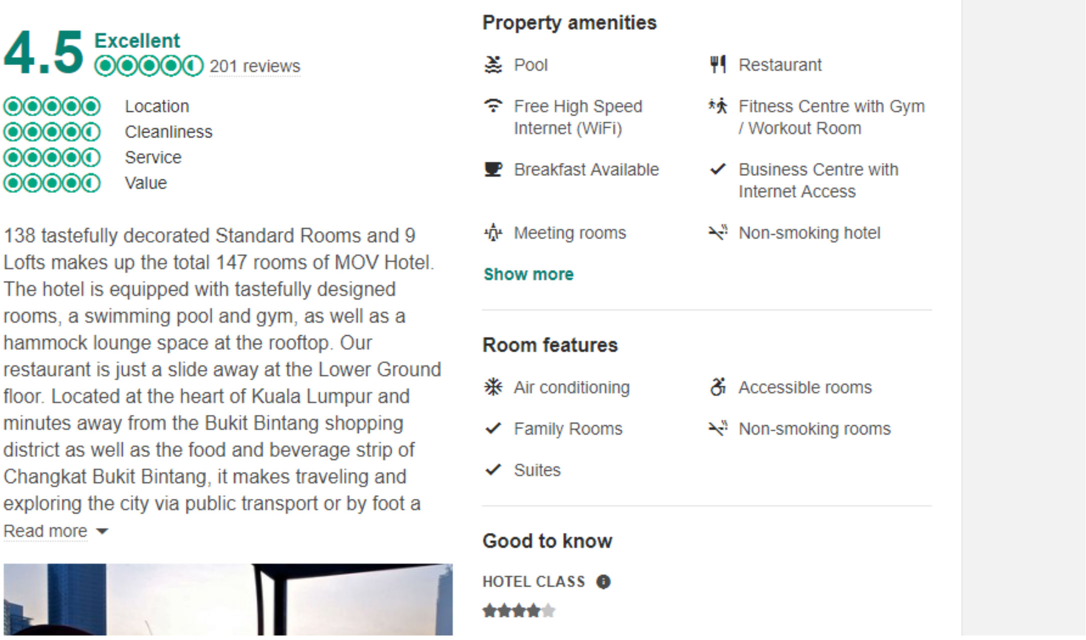
… div div div h1.attractions-supplier-profile-SupplierProfile__supplierName--3QHre

Styles | Event Listeners | DOM Breakpoints | Properties | Accessibility

```
element.style {
}
@media (min-width: 768px)
.hasAvatar          attractions.com…abcee86f.css:1
.attractions-supplier-profile-
SupplierProfile__supplierName--3QHre {
  padding:▸ initial;
}
.hasAvatar          attractions.com…abcee86f.css:1
.attractions-supplier-profile-
SupplierProfile__supplierName--3QHre {
  ~~padding:▸ 35px 10px 0;~~
}
@media (min-width: 768px)
.attractions-       attractions.com…abcee86f.css:1
supplier-profile-SupplierProfile__supplierName-
-3QHre {
```

▸ color	■rgb(7…
▸ display	block
▸ font-family	Arial, …
▸ font-size	28px
▸ font-weight	700

Figure 5.15 Alternate interface of attraction's webpage.



4.5 Excellent
●●●●◐ 201 reviews

●●●●●	Location
●●●●◐	Cleanliness
●●●●◐	Service
●●●●◐	Value

138 tastefully decorated Standard Rooms and 9 Lofts makes up the total 147 rooms of MOV Hotel. The hotel is equipped with tastefully designed rooms, a swimming pool and gym, as well as a hammock lounge space at the rooftop. Our restaurant is just a slide away at the Lower Ground floor. Located at the heart of Kuala Lumpur and minutes away from the Bukit Bintang shopping district as well as the food and beverage strip of Changkat Bukit Bintang, it makes traveling and exploring the city via public transport or by foot a

Read more ▾

Property amenities

- Pool
- Restaurant
- Free High Speed Internet (WiFi)
- Fitness Centre with Gym / Workout Room
- Breakfast Available
- ✓ Business Centre with Internet Access
- Meeting rooms
- Non-smoking hotel

Show more

Room features

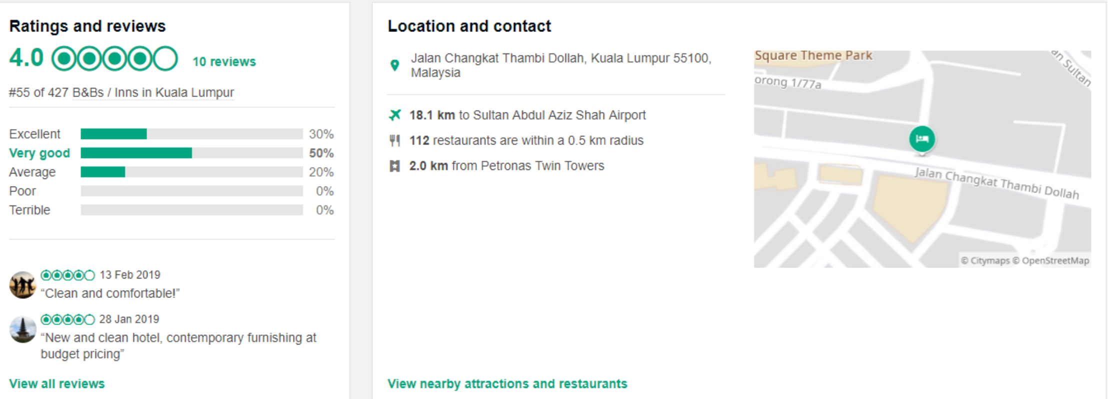
- Air conditioning
- Accessible rooms
- ✓ Family Rooms
- Non-smoking rooms
- ✓ Suites

Good to know

HOTEL CLASS

★★★★☆

Figure 5.16 Main interface of hotel's webpage.



Ratings and reviews

4.0 ●●●●○ 10 reviews

#55 of 427 B&Bs / Inns in Kuala Lumpur

Excellent	30%
Very good	**50%**
Average	20%
Poor	0%
Terrible	0%

●●●●○ 13 Feb 2019
"Clean and comfortable!"

●●●●○ 28 Jan 2019
"New and clean hotel, contemporary furnishing at budget pricing"

View all reviews

Location and contact

Jalan Changkat Thambi Dollah, Kuala Lumpur 55100, Malaysia

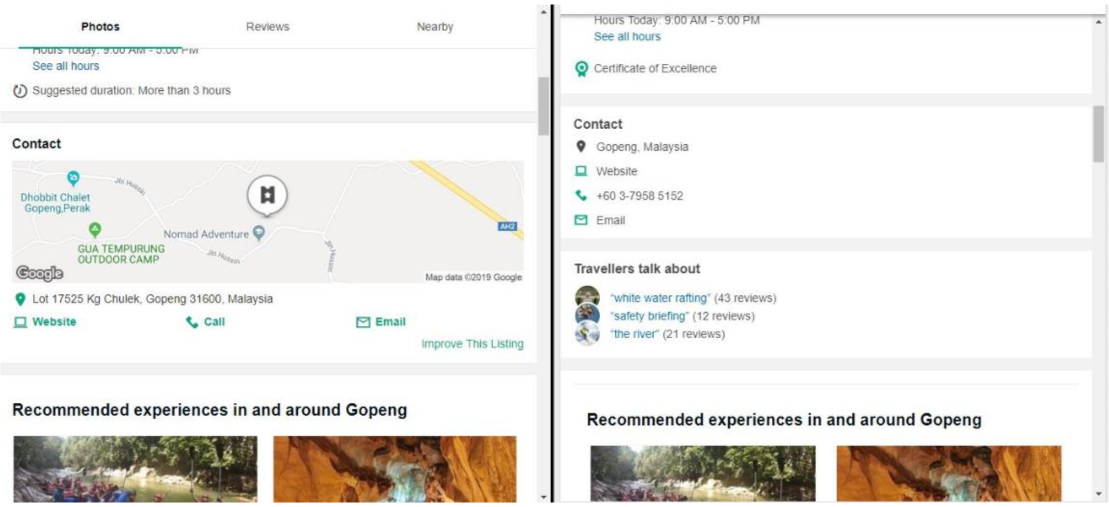
18.1 km to Sultan Abdul Aziz Shah Airport

112 restaurants are within a 0.5 km radius

2.0 km from Petronas Twin Towers

View nearby attractions and restaurants

Figure 5.17 Alternate interface of hotel's webpage.



Photos Reviews Nearby

Hours Today: 9:00 AM - 5:00 PM
See all hours
Suggested duration: More than 3 hours

Contact

Lot 17525 Kg Chulek, Gopeng 31600, Malaysia
Website Call Email
Improve This Listing

Recommended experiences in and around Gopeng

Hours Today: 9:00 AM - 5:00 PM
See all hours
Certificate of Excellence

Contact
Gopeng, Malaysia
Website
+60 3-7958 5152
Email

Travellers talk about
"white water rafting" (43 reviews)
"safety briefing" (12 reviews)
"the river" (21 reviews)

Recommended experiences in and around Gopeng

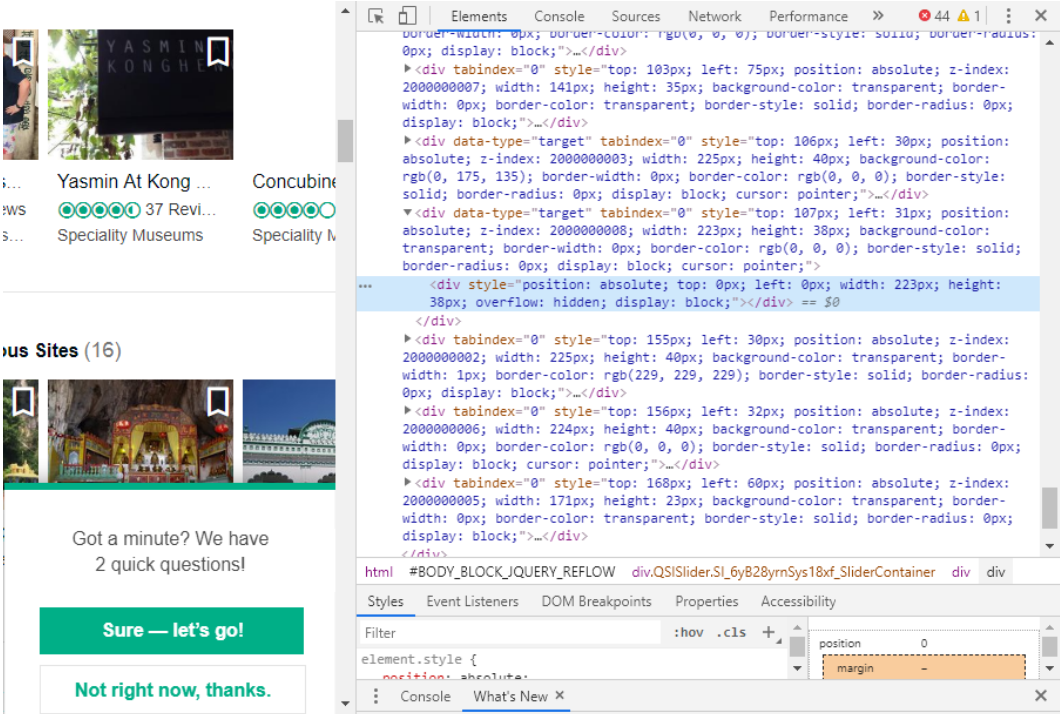
Figure 5.18 Can't scrape certain data due to different interface.

Another problem is website caching problem. Browser will store the website caches, therefore when running the program, the old html locator can still be used to locate the web element as it is already in the cache. However, some of the new or updated web element can't be located and must re-locate the update web element. This can be solved by simply clearing the browser cache. Besides, there is also a problem that some reviews are lost due to traveller type are not selected. There is currently no way to solve the problem and we can only choose to leave out the reviews with no traveler type as traveler type is one of the most important attributes to be analyzed for tourism sector.

Also, another problem is "WebDriverException" error. This error usually happens when the program is clicking the unintended web element as the intended element is not within the display window for the program to click. Therefore, the solution is simply scroll to the intended element so that it is within the display window. The implementation is shown in the code below.

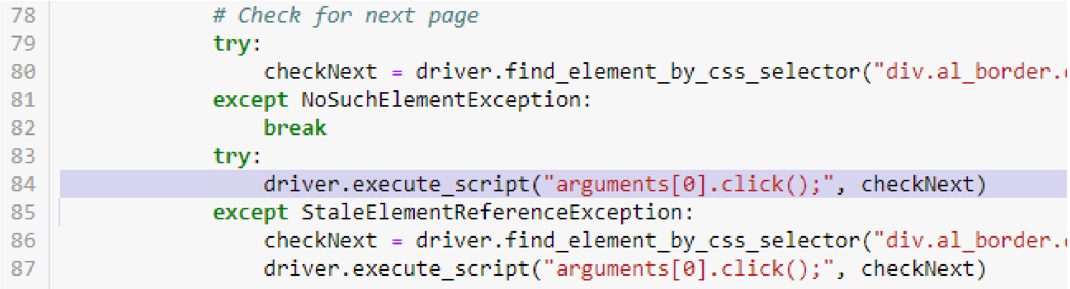
```
ActionChains(driver).move_to_element(review).click(review).perform()
```

One more problem is that some webpages have a pop-up or ads which are block the web element that the program is targeting to click. In order to click the element beneath the pop-up, one can choose to close the pop-up first and then proceed to clicking the element. But the process is not straight-forward and required a long line of code, not to mention that it way too difficult to locate the element of the pop-up due to no id or class name for locating element as shown in the figure 5.19. Therefore Javascript executor is used as it can bypass the pop-up and straight away interact with the element within the DOM.



```
border-width: 0px; border-color: rgb(0, 0, 0); border-style: solid; border-radius:
0px; display: block;">…</div>
▶<div tabindex="0" style="top: 103px; left: 75px; position: absolute; z-index:
2000000007; width: 141px; height: 35px; background-color: transparent; border-
width: 0px; border-color: transparent; border-style: solid; border-radius: 0px;
display: block;">…</div>
▶<div data-type="target" tabindex="0" style="top: 106px; left: 30px; position:
absolute; z-index: 2000000003; width: 225px; height: 40px; background-color:
rgb(0, 175, 135); border-width: 0px; border-color: rgb(0, 0, 0); border-style:
solid; border-radius: 0px; display: block; cursor: pointer;">…</div>
▼<div data-type="target" tabindex="0" style="top: 107px; left: 31px; position:
absolute; z-index: 2000000008; width: 223px; height: 38px; background-color:
transparent; border-width: 0px; border-color: rgb(0, 0, 0); border-style: solid;
border-radius: 0px; display: block; cursor: pointer;">
...    <div style="position: absolute; top: 0px; left: 0px; width: 223px; height:
       38px; overflow: hidden; display: block;"></div> == $0
  </div>
▶<div tabindex="0" style="top: 155px; left: 30px; position: absolute; z-index:
2000000002; width: 225px; height: 40px; background-color: transparent; border-
width: 1px; border-color: rgb(229, 229, 229); border-style: solid; border-radius:
0px; display: block;">…</div>
▶<div tabindex="0" style="top: 156px; left: 32px; position: absolute; z-index:
2000000006; width: 224px; height: 40px; background-color: transparent; border-
width: 0px; border-color: rgb(0, 0, 0); border-style: solid; border-radius: 0px;
display: block; cursor: pointer;">…</div>
▶<div tabindex="0" style="top: 168px; left: 60px; position: absolute; z-index:
2000000005; width: 171px; height: 23px; background-color: transparent; border-
width: 0px; border-color: transparent; border-style: solid; border-radius: 0px;
display: block;">…</div>
</div>
html  #BODY_BLOCK_JQUERY_REFLOW  div.QSISlider.SI_6yB28yrnSys18xf_SliderContainer  div  div
element.style {
  position: absolute;
```

Figure 5.19 Inspect element on the pop-up.



```
            # Check for next page
            try:
                checkNext = driver.find_element_by_css_selector("div.al_border.
            except NoSuchElementException:
                break
            try:
                driver.execute_script("arguments[0].click();", checkNext)
            except StaleElementReferenceException:
                checkNext = driver.find_element_by_css_selector("div.al_border.
                driver.execute_script("arguments[0].click();", checkNext)
```

Figure 5.20 Solution to click blocked element.

Chapter 6: Conclusion

In conclusion, the problem statement of this project is that there is too much public data of tourist available on the Internet has been wasted while they could be potentially valuable assets for data analytics on tourism field. Therefore, the motive to initiate this project is to provide convenience way for those who are involved in the tourim field especially data analyst to collect the tourism data for data analytics purpose. The proposed solution is to build a ready data collecting system model which will be able to capture user's interested data from the travel and tourism related website.

This proposed system model works by utilizing selected browser driver to open a browser windows in order to access to the targeted website, and then it will extract the data by locating the HTML element in the DOM. After the data has been captured, it will be put into several lists to categorize the attributes of the data collected. The lists will then be converted into NumPy array and is then combined into data frame by using panda library before inserted into the CSV files.

As most time in data analytics is used for collecting data and pre-processing data, the system provide a convenient way to collect data and also has some degree of data-preprocessing upon the collected data. With the realization of this project, data analysts can easily obtain the data they needed for the data analytics process and therefore improving Malaysia tourism.

The proposed system has successfully achieved its main functionality which is to identify the intended tourism data on the travel website in order to scrape it down and input into the csv file. Besides, the complete execution time for the proposed system has been massively reduced by roughly 72%. Contingency plan regarding backup and recovery has also been implemented for the proposed system. Besides, the contingency plan regarding the network instability has also been implemented to reduce the occurrence of the errors caused by the network problem.

There are some future works can be done to further enhance it such as giving user more option to store data beside csv file. Such as directly store into database. Can scrape oversea region. Can further enhance performance and efficiency. Develop the recovery plan more user-friendly, now is only developer friendly.

To be brief, this project is served to build a foundation for data collection phase of data analytics. With the realization of this project, data needed for data analytics process can be easily obtained and therefore can increase the efficiency and productivity of data analytics process as the time and resources spent on getting data can be saved. In the end, data analytics will help improving the tourism field and the society will be benefited from it, therefore this project has also indirectly contributed to that end.

Reference

Tourism Malaysia (2017), Malaysia Tourism Statistic in Brief [Online]. Available at https://www.tourism.gov.my/statistics. [Accessed: 12 April 2018].

Anon., n.d. What is Data Sharing. [Online]

Available at: http://www.igi-global.com/dictionary/data-sharing/6815

Vuuren, C. V. & Slabbert, E., 2011. TRAVEL MOTIVATIONS AND BEHAVIOUR OF TOURISTS TO A SOUTH AFRICAN RESORT. INTERNATIONAL CONFERENCE ON TOURISM & MANAGEMENT STUDIES, Volume 1.

Pinel, D. (1998), "A COMMUNITY-BASED TOURISM PLANNING PROCESS MODEL: KYUQUOT SOUND AREA. B.C." [Online] pp.53-64. Available at: http://www.collectionscanada.gc.ca/obj/s4/f2/dsk2/tape15/PQDD_0006/MQ31857.pdf [Accessed: 12 April 2018].

Quigg Z., Hughes K., Bellis M. A. (2012), "Data sharing for prevention: a case study in the development of a comprehensive emergency department injury surveillance system and its use in preventing violence and alcohol-related harms," Injury Prevention, 18, pp.315-320.

Personal Data Protection Act 2010 (Act 709) (PDPA)

Octoparse, computer software 2014. Available from: https://www.octoparse.com/. [Accessed: 12 April 2018].

Anon., 2019. Beautiful Soup: We call him Tortoise because it taught us.. [Online] Available at: https://www.crummy.com/software/BeautifulSoup/

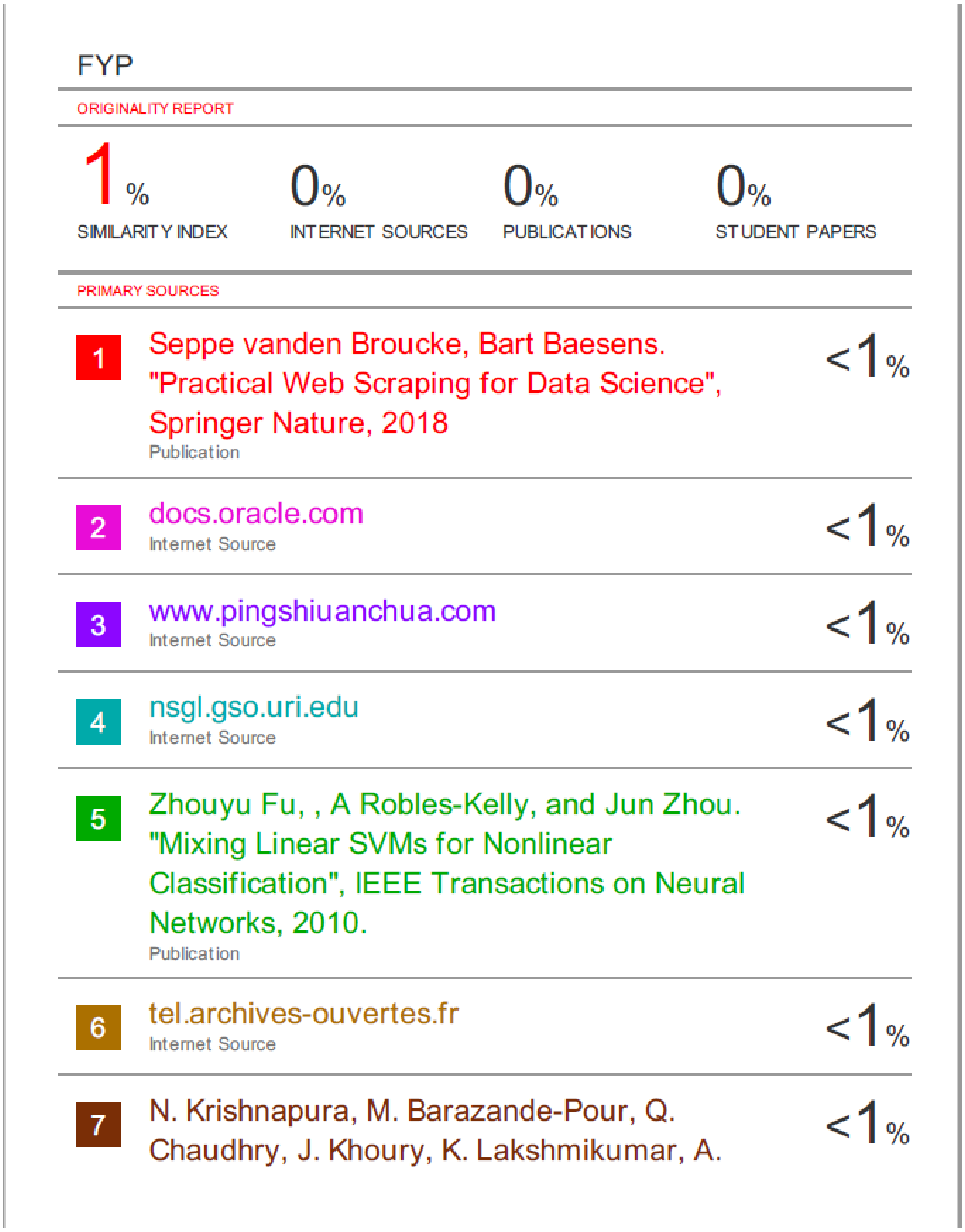
Scrapy, web crawling framework 2008. Available from: https://scrapy.org/. [Accessed: 12 April 2018].

Anon., n.d. What is the HTML DOM?. [Online]

Available at: https://www.w3schools.com/whatis/whatis_htmldom.asp

Plagiarism Check

Plagiarism Check Result



FYP

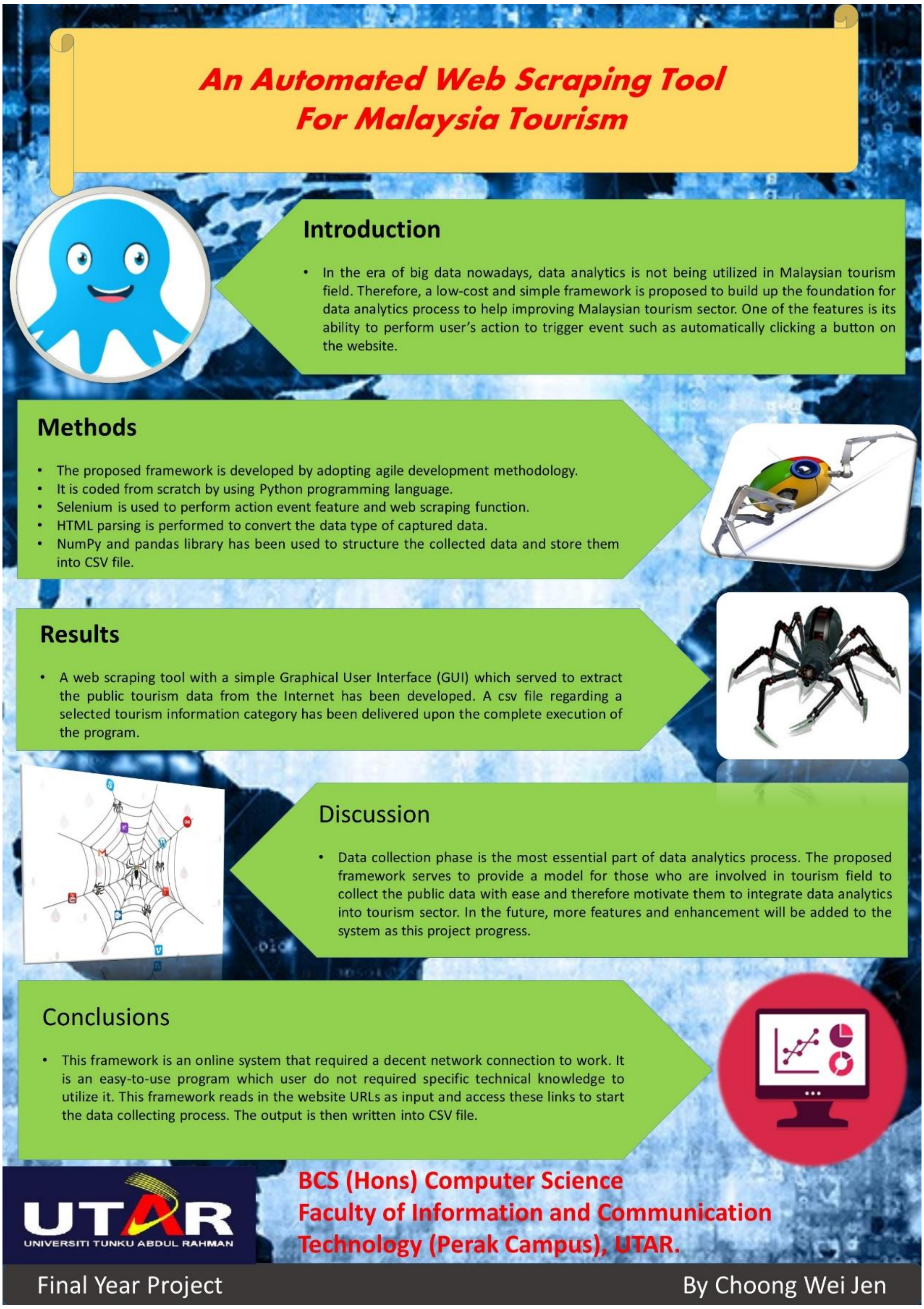
ORIGINALITY REPORT

1%	0%	0%	0%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	Seppe vanden Broucke, Bart Baesens. "Practical Web Scraping for Data Science", Springer Nature, 2018 Publication	<1%
2	docs.oracle.com Internet Source	<1%
3	www.pingshiuanchua.com Internet Source	<1%
4	nsgl.gso.uri.edu Internet Source	<1%
5	Zhouyu Fu, , A Robles-Kelly, and Jun Zhou. "Mixing Linear SVMs for Nonlinear Classification", IEEE Transactions on Neural Networks, 2010. Publication	<1%
6	tel.archives-ouvertes.fr Internet Source	<1%
7	N. Krishnapura, M. Barazande-Pour, Q. Chaudhry, J. Khoury, K. Lakshmikumar, A.	<1%

Poster



An Automated Web Scraping Tool For Malaysia Tourism

Introduction

- In the era of big data nowadays, data analytics is not being utilized in Malaysian tourism field. Therefore, a low-cost and simple framework is proposed to build up the foundation for data analytics process to help improving Malaysian tourism sector. One of the features is its ability to perform user's action to trigger event such as automatically clicking a button on the website.

Methods

- The proposed framework is developed by adopting agile development methodology.
- It is coded from scratch by using Python programming language.
- Selenium is used to perform action event feature and web scraping function.
- HTML parsing is performed to convert the data type of captured data.
- NumPy and pandas library has been used to structure the collected data and store them into CSV file.

Results

- A web scraping tool with a simple Graphical User Interface (GUI) which served to extract the public tourism data from the Internet has been developed. A csv file regarding a selected tourism information category has been delivered upon the complete execution of the program.

Discussion

- Data collection phase is the most essential part of data analytics process. The proposed framework serves to provide a model for those who are involved in tourism field to collect the public data with ease and therefore motivate them to integrate data analytics into tourism sector. In the future, more features and enhancement will be added to the system as this project progress.

Conclusions

- This framework is an online system that required a decent network connection to work. It is an easy-to-use program which user do not required specific technical knowledge to utilize it. This framework reads in the website URLs as input and access these links to start the data collecting process. The output is then written into CSV file.

BCS (Hons) Computer Science
Faculty of Information and Communication Technology (Perak Campus), UTAR.

Final Year Project

By Choong Wei Jen

Universiti Tunku Abdul Rahman			
Form Title : Supervisor's Comments on Originality Report Generated by Turnitin for Submission of Final Year Project Report (for Undergraduate Programmes)			
Form Number: FM-IAD-005	Rev No.: 0	Effective Date: 01/10/2013	Page No.: 1of 1

FACULTY OF INFORMATION AND COMMUNICATION TECHNOLOGY

Full Name(s) of Candidate(s)	
ID Number(s)	
Programme / Course	
Title of Final Year Project	

Similarity	**Supervisor's Comments (Compulsory if parameters of originality exceeds the limits approved by UTAR)**
Overall similarity index:________ % **Similarity by source** Internet Sources: ________________ % Publications: ________________ % Student Papers: ________________ %	
Number of individual sources listed of more than 3% similarity: _________	
Parameters of originality required and limits approved by UTAR are as Follows: **(i) Overall similarity index is 20% and below, and** **(ii) Matching of individual sources listed must be less than 3% each, and** **(iii) Matching texts in continuous block must not exceed 8 words** ***Note: Parameters (i) – (ii) shall exclude quotes, bibliography and text matches which are less than 8 words.***	

Note Supervisor/Candidate(s) is/are required to provide softcopy of full set of the originality report to Faculty/Institute

Based on the above results, I hereby declare that I am satisfied with the originality of the Final Year Project Report submitted by my student(s) as named above.

____
Signature of Supervisor

Name: __________________________

Date: ___________________________

____
Signature of Co-Supervisor

Name: __________________________

Date: ___________________________

UNIVERSITI TUNKU ABDUL RAHMAN

FACULTY OF INFORMATION & COMMUNICATION TECHNOLOGY (KAMPAR CAMPUS)

CHECKLIST FOR FYP2 THESIS SUBMISSION

Student Id	
Student Name	
Supervisor Name	

TICK (√)	**DOCUMENT ITEMS** Your report must include all the items below. Put a tick on the left column after you have checked your report with respect to the corresponding item.
	Front Cover
	Signed Report Status Declaration Form
	Title Page
	Signed form of the Declaration of Originality
	Acknowledgement
	Abstract
	Table of Contents
	List of Figures (if applicable)
	List of Tables (if applicable)
	List of Symbols (if applicable)
	List of Abbreviations (if applicable)
	Chapters / Content
	Bibliography (or References)
	All references in bibliography are cited in the thesis, especially in the chapter of literature review
	Appendices (if applicable)
	Poster
	Signed Turnitin Report (Plagiarism Check Result - Form Number: FM-IAD-005)

*Include this form (checklist) in the thesis (Bind together as the last page)

I, the author, have checked and confirmed all the items listed in the table are included in my report. ________________________ (Signature of Student) Date:	Supervisor verification. Report with incorrect format can get 5 mark (1 grade) reduction. ________________________ (Signature of Supervisor) Date: