Analyzing IT job salaries in Vietnam by an analysis web application

Case study: Deploying via the database on the ITviec.com website

LAB University of Applied Sciences
Bachelor of Business Administration, Business Information Technology
2021
Trang Le

Abstract

Author(s) Trang Le	Publication type Thesis, UAS	Completion year 2021
	Number of pages 67	

Title of the thesis

Analyzing IT job salaries in Vietnam by an analysis web application

Case study: Deploying via the database on the Itviec.com website

Degree

Business Information Technology

Abstract

A minimal implementation for a web application comparing IT job salaries offered in the Vietnamese labor market efficiently and easily is presented. To resolve the requirements and technical feasibility, two research questions and a case study were conducted. The case study is a web application via the database on the Itviec.com website.

The thesis has two main viewpoints which are the technical side and the business side based on two research questions. The technical part discusses the applied technologies and their implementation. The business part introduces the requirements including identifying the customer need and the web application wireframe. The thesis follows the inductive approach. Both quantitative methods and qualitative methods are used to help answer the main research questions.

On the completion of the project, the suggestion for further research is provided to enhance more functions that serve the business in the future.

Keywords

Web scraping, data visualization, data analytics, streamlit, beautiful soup

Contents

Appendices

Abbreviations and Terms

API	Application Programming Interface
CSV	Comma-Separated Values
DOM	Document Object Model
GUI	Graphical User Interface
HTML	Hyper Text Markup Language
HTTP	Hyper Text Transfer Protocol
I/O	Input/Output
SME	Small and Medium-sized Enterprises
URL	Uniform Resource Locator
XML	Extensible Markup Language

1 Introduction

1.1 Background

In the Vietnamese IT labor market, salaries differ between foreign and Vietnamese companies (Vietnam Briefing 2019). The salary level is determined by three main factors such as the company's budget, job responsibilities, and the employee's qualifications (Indeed 2021). Therefore, when candidates seek jobs, candidates refer to specialized knowledge and professional skills. In addition, they also invest time to research about the company, employee benefits, career journey, and especially the level of salary before applying.

Vietnamese IT employers often use the expression salary negotiable in recruitment notices. Recruiters are unwilling to open the salary for recruitment positions (Glassdoor 2019). Therefore, it is hard for job seekers especially newbies to find out the salary range. There is no standard to know that the negotiated salary is higher or lower than the market value. Negotiation skills play a crucial role in the interview in this case.

1.2 Objectives

Vietnamese job seekers may collect salary data manually to calculate an average salary. However, this is ineffective and not necessarily accurate. As the result, this thesis aims to produce a web application to help Vietnamese IT job seekers. The web application helps them reflect on and make well-informed decisions when applying for a job. The developing process will identify the customer's demand to define the wireframe. The web application will be built by following the wireframe.

The thesis is a technical component for the business model that other companies can develop ideas in the future. Thus, the case study will only look at the technical side of this subject, involving creation and operation. The case study is to deploy an analysis web application via a database on an actual website called Itviec.com.

1.3 Research questions

The study has two main research questions. The first research question is the following:

What indicators are job seekers interested in?

The following subordinate research questions should help answer the main research question:

- What are the key performance metrics and salary information for jobseekers?

- What kind of time value report is used to visualize salary information?

The second research question is as follows:

What is the most effective way to deploy a salary analysis web application to maximize its benefits?

The following sub-ordinate research questions help answer the second main research question:

- What are the techniques to deploy the web application?
- What are the advantages of techniques compared to other methods in development?

The sub-questions help in achieving the thesis's purpose.

1.4 Research and data collecting methods

This chapter discusses how the thesis's initial concept was addressed. The methods used to do research and gather data during the thesis process are also introduced.

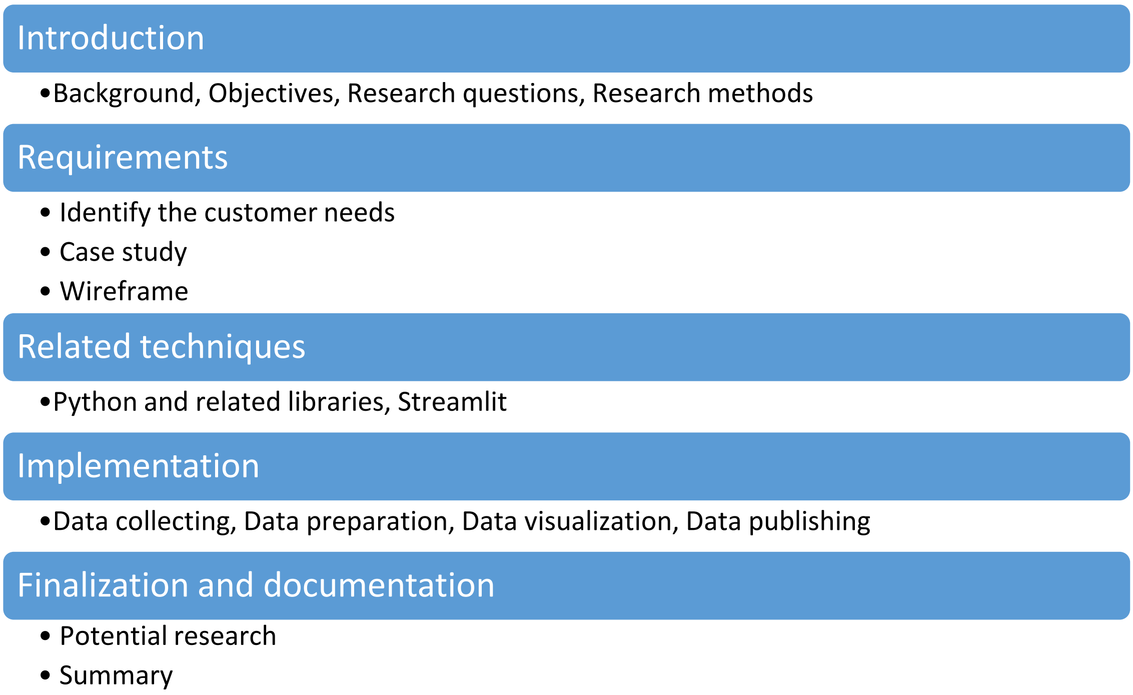
There are two research approaches: an inductive and a deductive approach. The inductive approach is appropriate for research that begins with gathering data that does not have a specific hypothesis or model. This technique is significantly more adaptable and allows for a more open research process during the theory project development. (Locke 2007, 880-885.) A deductive technique presents a comprehensive beginning point and entails the application of several hypotheses and tests to completely update this model. This method is best utilized when there is a hypothesis to test, and it may be used to modify a previously established assumption. (Nola & Sankey 2007, 108-109.) In this thesis, the salary analysis web application is created based on the desires of users. Therefore, this thesis follows the inductive approach.

The thesis applies both quantitative and qualitative research methods. Quantitative methods are used to help answer the first main research question, whereas qualitative methods are used to help answer the second main research question. According to Bhandari (2020), quantitative research collects and analyzes numerical data to discover models and averages. The quantitative is also used to forecast and illustrate the general features of a larger population. Therefore, individual interviews and a survey will be conducted to extract more information to find out jobseekers' desires. Qualitative research aims to clarify specific aspects to comprehend concepts, ideas, or experiences. It enables readers to gain in-depth insights into a topic or to generate fresh research ideas (Bhandari 2020). The following qualitative methods will be used in the related technical part:

- Literature review: the definition, the history, the usability.
- Information collection from publications and books, previous research, the Internet, and other sources.
- Criteria for comparing alternatives: the popularity, the utilization, the expenses, the maintenance, the potential.

1.5 Thesis structure

The thesis will be structured as follows:



Introduction

- Background, Objectives, Research questions, Research methods

Requirements

- Identify the customer needs
- Case study
- Wireframe

Related techniques

- Python and related libraries, Streamlit

Implementation

- Data collecting, Data preparation, Data visualization, Data publishing

Finalization and documentation

- Potential research
- Summary

Figure 1. Thesis structure

This thesis has five main chapters. The first chapter is presented as the introduction of the study. It introduces the background, the objectives, research questions, and research methods. It also provides the thesis structure.

Chapter 2 discusses the thesis's business aspect, which includes the web application requirements. It begins with defining the interview and survey design and then moves on to collecting and analyzing the results. The next sub-chapter introduces the wireframe. Finally, the final sub-chapter goes into detail regarding the case study.

Chapter 3 is the theoretical section which introduces all associated techniques. In the procedure of web application, the four main techniques will be used. Thus, the four sub-chapters will present each technique in two separate second sub-chapters. Python, Beautiful Soup, Pandas, and Streamlit are the correlative sub-chapters in this chapter.

Chapter 4 describes the web application developing process by following the case study. This procedure consists of four distinct steps. Thus, the chapter will contain four sub-chapters such as data collection, data preparation, data visualization, and data publishing.

Finally, chapter 5 answers the research questions. It also provides the validity and limitations of this study. In addition, the suggestion for further research is also presented.

2 Requirements

2.1 Identify the customer needs

This section answers the first main research question. Firstly, the chapter explains why developers need to concentrate on identifying the customer needs. In addition, it also provides the analysis of customer needs via survey and interview methods.

2.1.1 Overview

There are several reasons why developers must first determine customer needs before delivering a web application. Firstly, customer needs are non-technical, reflecting the customers' opinion of the product rather than the actual design standards, though they are usually intertwined. It is the reason why developers need to recognize customer needs to begin addressing the web application. In addition, the web application is the product to serve exactly what the buyer desires. (Simpson-Wolf 2021.) The first step should be to collect data from clients. It would be hard to identify their needs without their input because customer input is a guideline for product development. The purpose of gathering requirements is considerably different from that of a sales call. The goal is to elicit an honest expression of needs, not to persuade a customer of what he or she needs. (Ulrich & Eppinger 2012, 79.)

There are several methods for gathering data. The popular method is survey and interview. These will be the primary data collection and compilation techniques for this study.

Survey

The following information describes the target respondents of the survey:

- The number of respondents: 25-30 people
- Objects: freshers, fresh graduates, juniors who have less than 2 years of experience
- Location:
  - Oversea students who would like to find a job in Vietnam
  - Students and employees in Vietnam

Interview

The information below describes the interview's intended audience**:**

- The number of interviewees: 2-3 people

- Objects: seniors who have more than 3 years of experience
- Location:
    - Employees in Vietnam

The survey and questions were created in early October 2021. The interviews were then held between October 15 and October 19, 2021. The data was analyzed over the last week of October. The below second sub-chapters will discuss the survey and interview design, and the outcomes of the process.

2.1.2 Interview design

According to Wilson (2014), there are three categories of interviews such as structured, unstructured, and semi-structured. A structured interview has a clearly defined series of questions that generate shorter responses from participants. However, it may not allow respondents to elaborate on their answers, thus resulting in the loss of potentially significant data. An unstructured interview, on the other hand, deals with broader questions. It can lead to answers that are not very informative, or, worse, to the conversation veering completely off-topic. (Wilson 2014, 176.) A semi-structured interview combines features of both structured and unstructured interviews. It includes the planned questions, but with a much wider approach to allow the respondent to suggest additional appropriate questions and ideas. (Wilson 2014, 177.) This thesis applies the semi-structured interview method.

The five interview questions focus on the level of interest when experienced people use the salary analysis web application versus non-experienced persons. Question 1 aims to find out which recruitment websites the interviewees use the most to obtain reputable data sources. Question 2 seeks to elicit salary information regarding interviewees' behavior to reframe any potential extra functions. Question 3 is about their awareness of the salary function on the existing recruitment websites as well as their curiosity. Question 4 helps find out whether or not interviewees are interested in the salary information. Question 5 helps find out how to improve the web application based on the interviewees' suggestions.

Index	Questions
1	What are some of your favorite recruitment websites?
2	What are your purposes for using a recruitment website?

3	Do you know about the salary analyzing function on the website? If yes, what situation do you often use this in? If not, are you interested in it and why?
4	Do you often compare between the current year's salary and the previous year's wage level in the labor market?
5	Do you have any suggestions for a centralized salary analysis web application?

Table 1. Interview questions

2.1.3 Survey design

While an interview needs a significant time investment from individuals, UX surveys are a simple and quick technique to collect data from users. It can be conducted in a variety of methods, including online, in person, or via mail. A clear questionnaire and a well-designed structure are indicators of high-quality survey responses. (Rosenzweig 2015.)

This questionnaire was carried out by using Google Forms. The questionnaire has three questions which are questions 1 to 3 in Table 1. However, the respondents are inexperienced persons. The questionnaire includes multiple-choice questions, checkbox questions, and open-ended questions. Furthermore, four new questions were established for special purposes, as shown in Table 2 below. Question 1 concerns the difficulties that younger people confront when looking for salary information. Question 2 assists creators in defining the respondents' interests with salary analysis web application. Question 3 focuses on finding out what indicators non-experienced people's interest when they decide to look for a new job. When visualizing salary information, question 4 determines the time value report of salary information. The full questionnaire is available in Appendix 1 at the end of this thesis.

Index	Questions
1	What is the biggest challenge you face when searching for salary information?
2	Are you interested in having a web application to centralize this information from several recruitment websites?

3	If you are searching for jobs, what are the indicators that you are excited about?
4	What kind of time value report is used to visualize salary information?

Table 2. Survey questions

2.1.4 Collecting results

Survey

There are 25 people that took part in the survey. They are the end-users of this web application. Sixty percent of the respondents are freshers or juniors with less than two years of work experience. Twelve percent are recent graduates and 28 percent are junior or senior students.

Twenty out of 25 respondents (80%) are interested in the salary analysis application to obtain salary information from several recruitment websites (Figure 2).

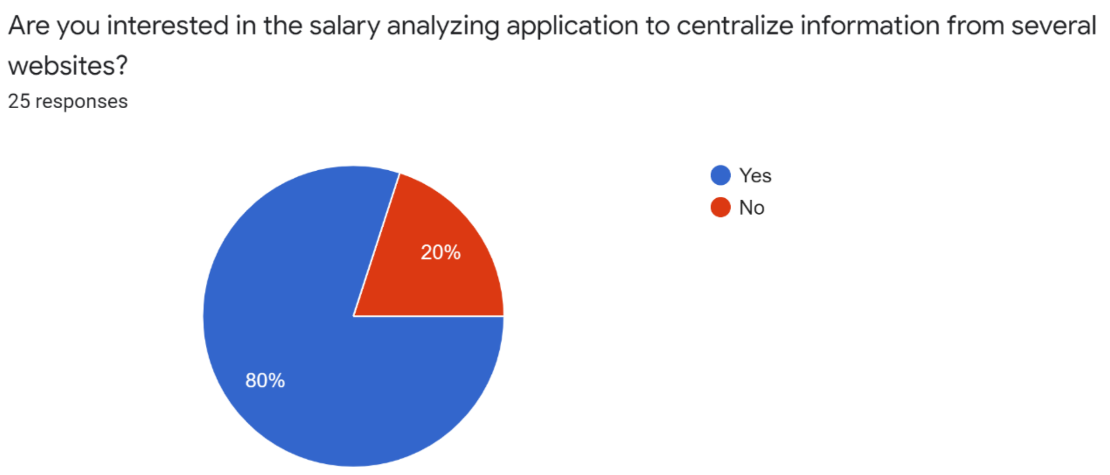


Figure 2. The measure of end-user interest

Only ten of the 25 respondents (40%) know about the salary analysis function on recruitment websites (Figure 3).

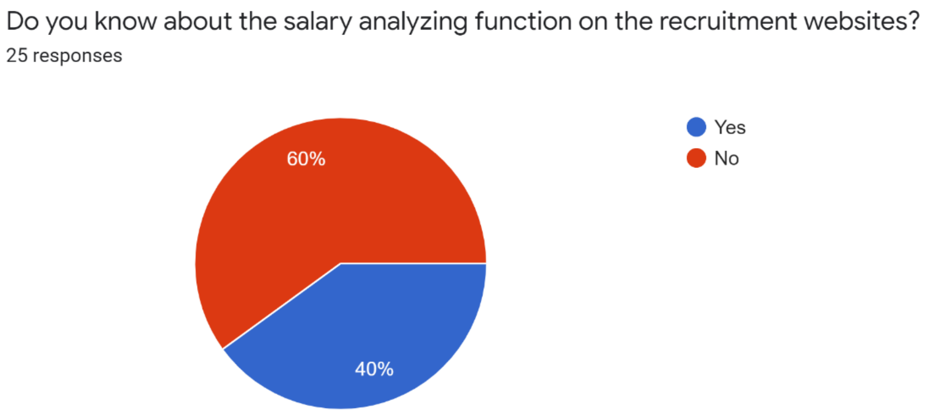
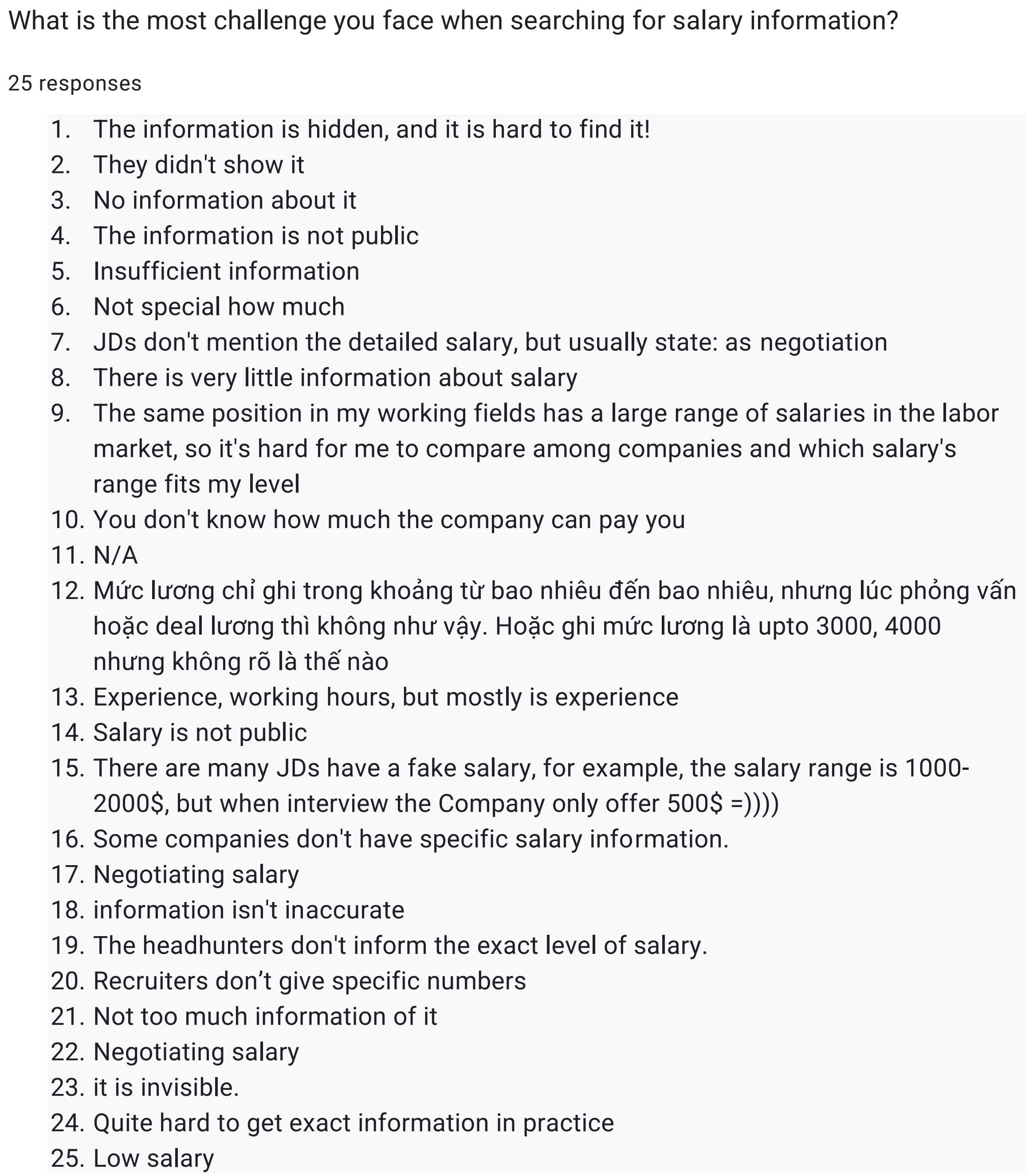


Figure 3. The popularity of salary analyzing function.

When respondents were asked about the biggest challenge they face when searching for salary information, many respondents stated that salary information is not shown or is hidden (Image 1).



What is the most challenge you face when searching for salary information?

25 responses

1. The information is hidden, and it is hard to find it!
2. They didn't show it
3. No information about it
4. The information is not public
5. Insufficient information
6. Not special how much
7. JDs don't mention the detailed salary, but usually state: as negotiation
8. There is very little information about salary
9. The same position in my working fields has a large range of salaries in the labor market, so it's hard for me to compare among companies and which salary's range fits my level
10. You don't know how much the company can pay you
11. N/A
12. Mức lương chỉ ghi trong khoảng từ bao nhiêu đến bao nhiêu, nhưng lúc phỏng vấn hoặc deal lương thì không như vậy. Hoặc ghi mức lương là upto 3000, 4000 nhưng không rõ là thế nào
13. Experience, working hours, but mostly is experience
14. Salary is not public
15. There are many JDs have a fake salary, for example, the salary range is 1000-2000$, but when interview the Company only offer 500$ =))))
16. Some companies don't have specific salary information.
17. Negotiating salary
18. information isn't inaccurate
19. The headhunters don't inform the exact level of salary.
20. Recruiters don’t give specific numbers
21. Not too much information of it
22. Negotiating salary
23. it is invisible.
24. Quite hard to get exact information in practice
25. Low salary

Image 1. The biggest challenges respondents’ have faced when searching for salary information

Question 6 focused on finding out the indicators the respondents are interested in. Salary, job description, company, and location are the four most important indicators (Figure 5).

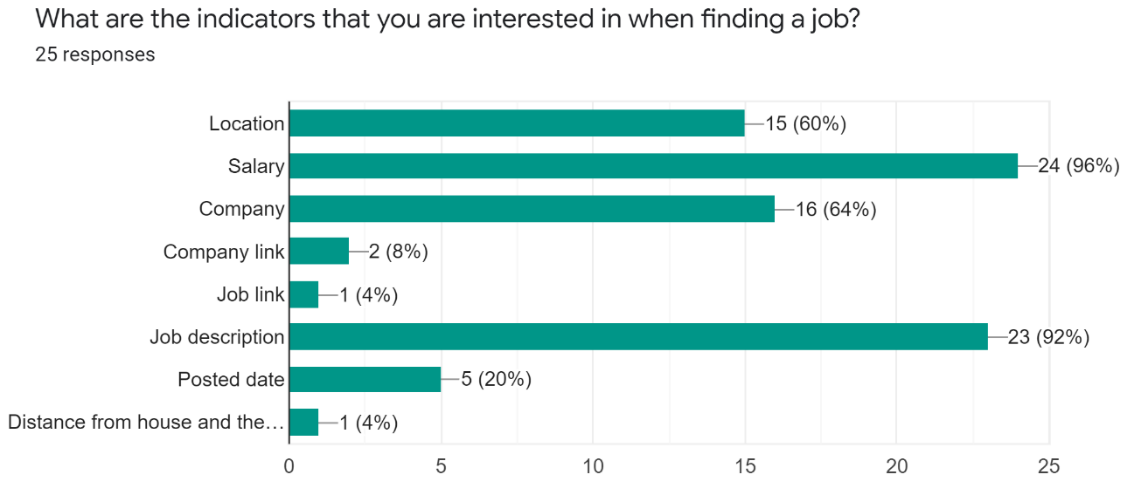


Figure 4. The indicators of web application

Question 7 is a multiple-choice question. It explores the time value report that respondents are interested in for visualizing salary information. Figure 6 below reveals that the respondents prefer monthly (64%) and yearly (36%) time value reports.

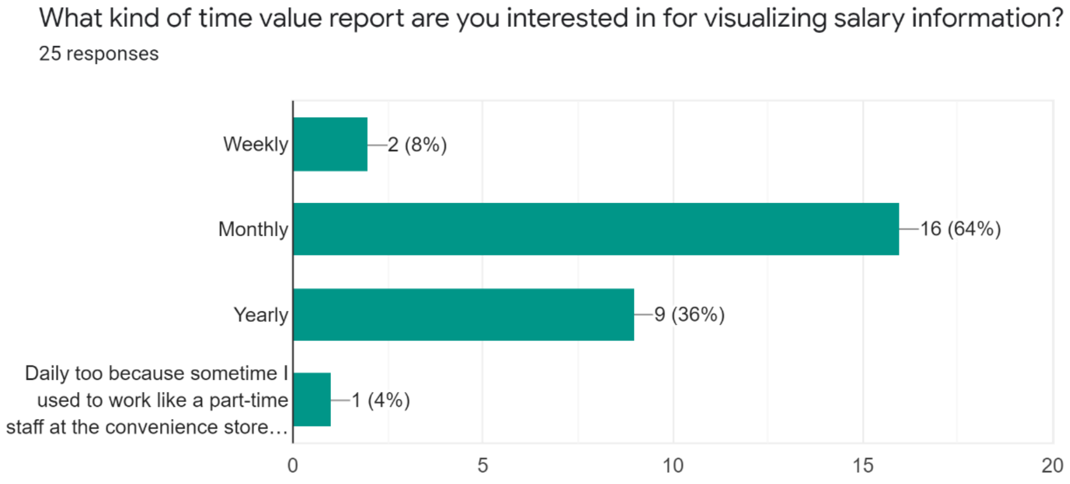


Figure 5. The time value for visualizing salary information

Finally, question 8 lists several well-known recruitment websites. The aim is to find out the respondents' favorites. The top popular websites are Linkedln, Vietnamworks, Itviec, Glassdoor, and TopCV (Figure 7).

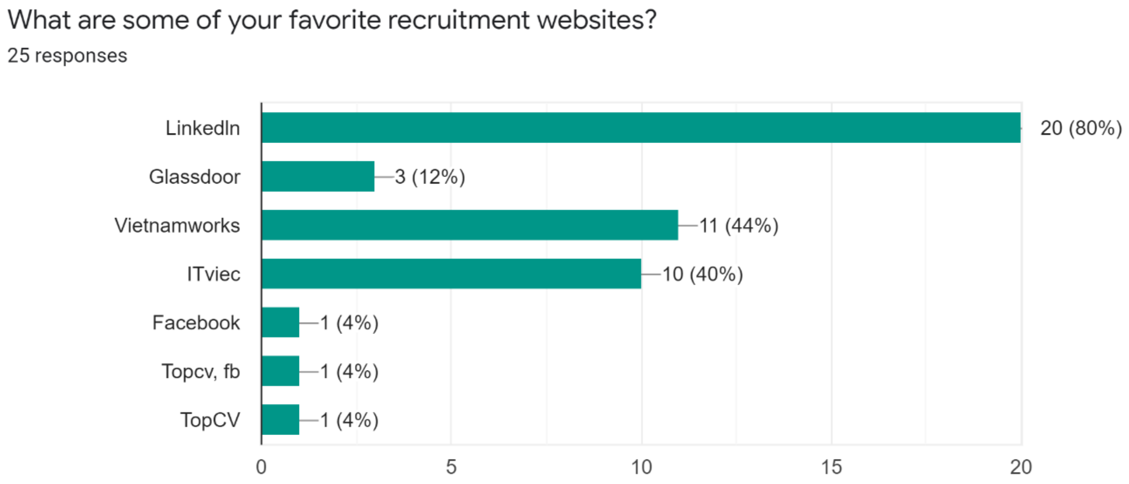


Figure 6. The popular recruitment websites

Interview

The interview for this study included three interviewees. The first interviewee is a senior business intelligence developer. He has worked at Robert Bosch Vietnam for 3 years. In opposition to people who have less than 2 years of experience, he is not overly concerned with the salary assessing function. He also frequently uses popular job-search websites like Linkedin, Itviec, and Vietnamworks. The goal of using these websites is to connect with others in the same business and to stay up to speed on new trends and information by following the sharing status of connected people and companies. Salary does not play an important role in his interest. He cares more about the career path, the ability to get promotions, benefits, the organization of the team. For salary information, he prefers to read the yearly salary report from big companies such as Adeco, or Nielsen. When asked about the application, he believes that it would be more useful to persons who are younger or less experienced than him. He also suggested that because IT occupations are so diverse, they should be separated into distinct divisions such as web development, AI, software development, mobile development, and so on.

The second interviewee is also a senior business intelligence developer. He has almost five years of experience. He is currently employed for Shopee Vietnam. In general, he shares the same viewpoint as the previous interviewee. He believes the salary application will appeal to recent graduates because it is difficult for them to assess the value they may contribute to the organization and demand a commensurate wage at that time. As a result, a tool that gives them an average wage will help them a lot in the interview dealing round. He is aware of the salary functions on the existing recruitment websites, but he is not overly concerned. He uses LinkedIn to build his working network relationships, share expertise, and keep up with the latest updates. He recommended that when users hover over the information of job or company, it will navigate directly into the correlative website into the new blank tab of the web browser. It serves a user-oriented objective, allowing all activities and information to be centralized on a single webpage.

The final interviewee is a senior data analyst at Lifesup Co. Ltd, a Vietnamese SME technology company. He has been the technical head here for five years. In contrast to the previous two interviewers, he is interested in pay information since he believes that benefits in small businesses are not as appealing as those in large corporations. Because salary is his primary monthly outcome, he is concerned about the annual rate of increase in salary level. He frequently uses recruiting to look for freelance employment, compares his compensation to that of the same level in a large corporation to propose an appropriate salary adjustment, or connects with friends to look for more appealing options. He is aware of the salary function on Glassdoor and LinkedIn, but they are ineffective for him. He claims that it does not reflect the true level of income depending on the position in the Vietnamese labor market. When asked about the salary web application, he enthusiastically accepted because it saves him time by allowing him to gather information from a single website rather than multiple recruitment websites. He suggested that the job title should be separated into distinct position levels such as fresher, junior, senior, and manager.

In conclusion, the paragraph analyzes the result of the interviews. Question 1 was answered by the first interviewee. The rest of the interviewees did not respond in detail about this question. LinkedIn is the same response they mentioned in this question. Thus, LinkedIn is a recruitment website that can be used as a reputable source. When they were asked about their behavior in question 2, they are usually used LinkedIn as social media for individual purposes. All interviewees have known about the salary function on LinkedIn in question 3. However, there are only one of three interviewees that is interested in the salary information. Thus, the end users' web application should focus on non-experienced people who have less than 2 years of experience. In addition, all interviewees give useful recommendations to help improve the web application.

2.2 Wireframe

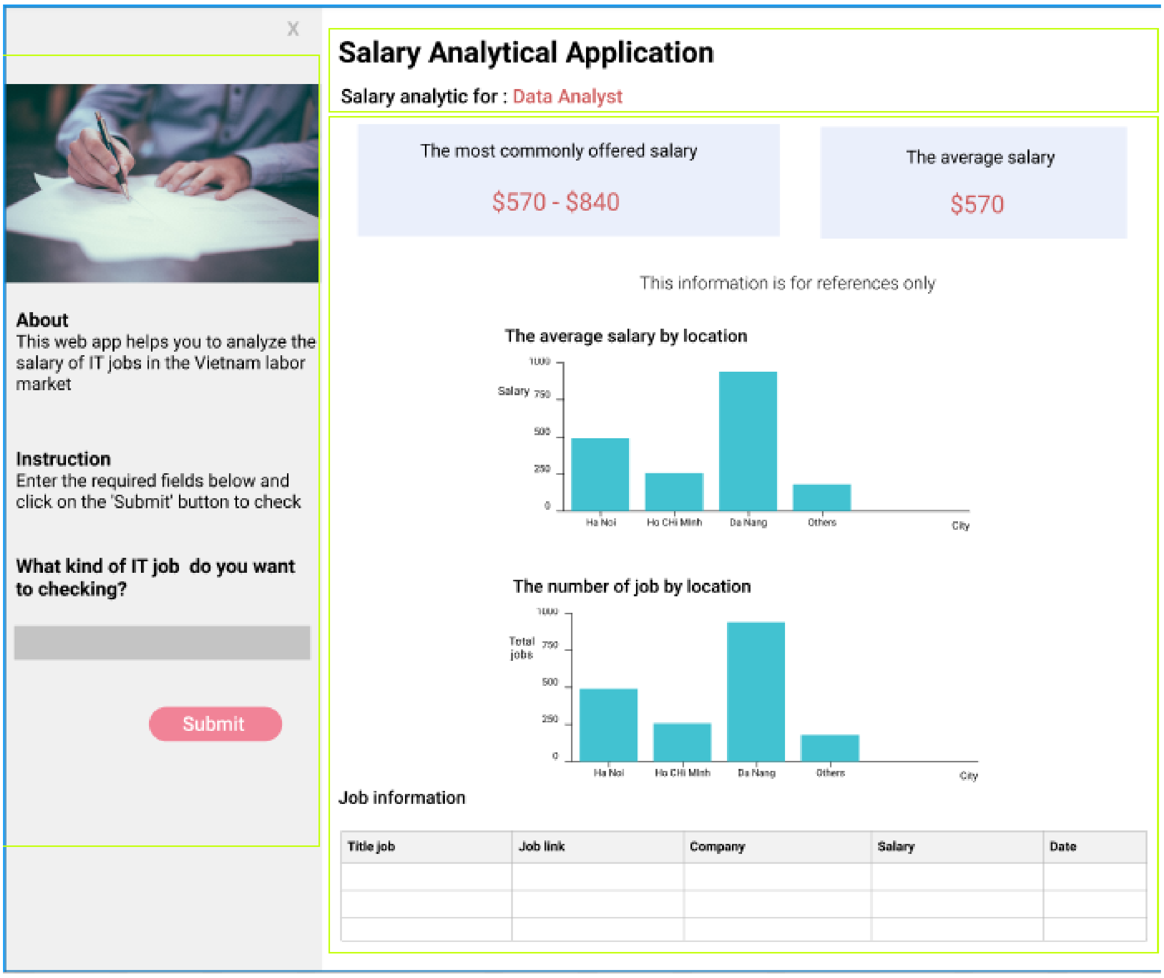
Wireframing is a process in which designers generate overviews of interactive products to identify the structure and flow of potential design solutions. These outlines depend on user and business requirements. Wireframes, whether on paper or in software, aid teams and stakeholders in developing optimal, user-oriented prototypes and products. (Interaction Design 2021.)

Figma is a cloud-based design tool that is functional and feature-wise comparable to Sketch (Kopf 2021). This thesis uses Figma as a tool to create the wireframe of the web application.

According to the identifying customer needs in chapter 2.1, the key performance metrics should be designed in the wireframe including salary, location, company, and job formation detail. There are three main components of the web application including:

- Navigation part: It is on the left of the website to present the overview, the brief instruction of application. In addition, it establishes the searching box to help users centralize the output.
- Header part: It presents the name of the application and the name of the job search.
- Chart visualization part: It visualizes the key defined metrics in different charts to help end-users to catch up with the insights.

The wireframe is designed on a website layout that is only suitable for the desktop interface. To understand each step to build the web application as the same as wireframe, the solution will be explained in the chapter: Implementation.



Salary Analytical Application

Salary analytic for : Data Analyst

The most commonly offered salary

$570 - $840

The average salary

$570

This information is for references only

The average salary by location

The number of job by location

Job information

Title job	Job link	Company	Salary	Date

About
This web app helps you to analyze the salary of IT jobs in the Vietnam labor market

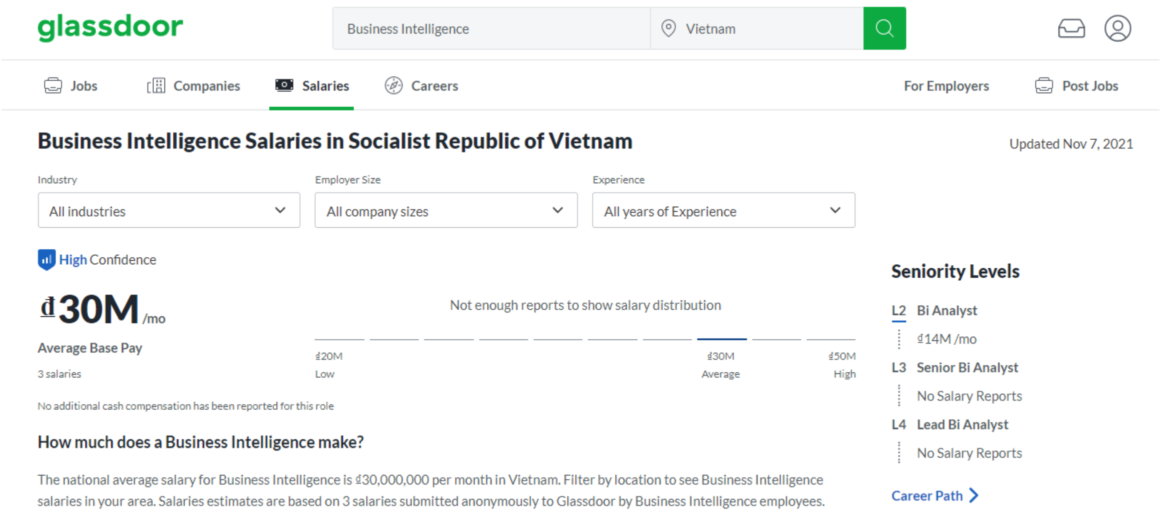
Instruction
Enter the required fields below and click on the 'Submit' button to check

What kind of IT job do you want to checking?

Figure 7. Screenshot from Figma: The wireframe of the salary analysis web application

3 Case study

Numerous third-party recruitment websites merged the salary analysis function in their website. People can find this function via the most popular websites such as Glassdoor or LinkedIn as shown in image 2 below. However, Glassdoor also does not provide enough data to make the accuracy of results because the IT Vietnam companies prefer using Vietnamese websites to post the jobs rather than global websites.



Business Intelligence Salaries in Socialist Republic of Vietnam

Updated Nov 7, 2021

Industry: All industries
Employer Size: All company sizes
Experience: All years of Experience

High Confidence

₫30M /mo

Average Base Pay

3 salaries

Not enough reports to show salary distribution

₫20M	₫30M	₫50M
Low	Average	High

No additional cash compensation has been reported for this role

How much does a Business Intelligence make?

The national average salary for Business Intelligence is ₫30,000,000 per month in Vietnam. Filter by location to see Business Intelligence salaries in your area. Salaries estimates are based on 3 salaries submitted anonymously to Glassdoor by Business Intelligence employees.

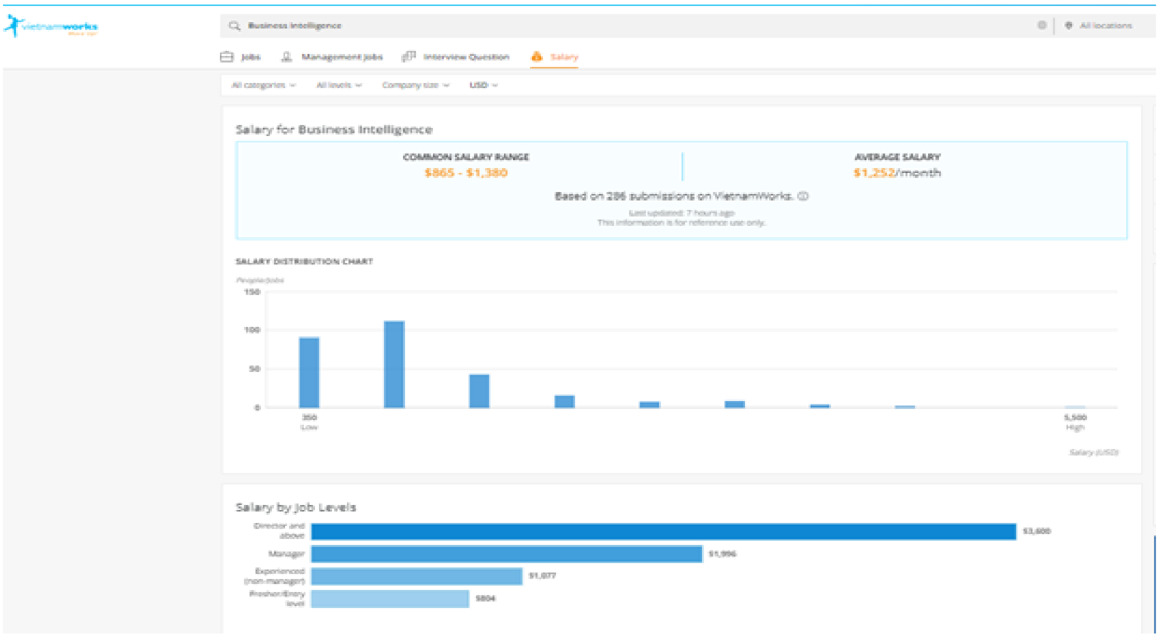
Seniority Levels

L2 Bi Analyst
₫14M /mo
L3 Senior Bi Analyst
No Salary Reports
L4 Lead Bi Analyst
No Salary Reports

Career Path >

Image 2. Screenshot from Glassdoor’s website: the user interface of the salary analysis function

One of the most popular Vietnamese websites that implement the salary analysis function is Vietnamworks. But the database is collected from the posted job on their website as shown in image 3 below. It leads to data source bias because the database is analyzed by using only one data source.



Based on 286 submissions on VietnamWorks.

Last updated: 7 hours ago
This information is for reference use only.

Image 3. Screenshot from Vietnamwork's website: the user interface of the salary analysis function

When comparing the wireframe with the interfaces of Glassdoor and Vietnamworks, creators have more divergent points of view. Both websites offer two levels of slicer for users to choose from. Without searching by job title or location, they offer a variety of options such as company size, amount of expertise, industry, and so on. However, because of the limitation of the database in the case study, there are not enough resources to gather further information. In the wireframe, there are still two main drop-down lists to choose from job title and location. Glassdoor and Vietnamworks both have a bar chart that visualizes salary based on a level position. However, the wireframe does not refer to this function on the web application. It can be explored and used as a potential function in the future phase of development. Vietnamworks employs a card chart to display the salary range and the average salary. The wireframe specifies one table with distinct columns to convey this information. Furthermore, both websites display statistics using a bar chart. As a result, table charts and bar charts are the most common style of chart.

According to the identifying customer need part above, Vietnamese job seekers are interested in a web application that will collect data from internal and external recruitment websites. The web application presents all information needed in one centralized place.

This thesis only provides the technical solutions and the practice implementation. Therefore, the case study will establish a web application. The database is scraped directly from an example of the most popular recruitment website in Vietnam to ensure the quality of information. The implementation will apply the same as the other recruitment websites when collecting data into the database. The following information illustrates the scope of the data collecting:

- Website: https://itviec.com/
- Location: Vietnam
- City: Hanoi, Ho Chi Minh, Da Nang, Others
- Topic: IT jobs
- Size of dataset: around 3000-4000 rows

In addition, this project will use spreadsheets (such as Microsoft Excel) as a place of storing data instead of using a database system. Thus, the data cleaning step prefers using both Excel formulas and Python libraries.

4 Related techniques

4.1 Python

Python is a high-level multi-purpose programing language that can handle a wide range of programming models. In the following chapter, the thesis introduces the advantages of Python and the comparison between Python and R in data analytics.

4.1.1 The advantages of Python

Python seems like a powerful language to cover the whole spectrum of ability in the following areas as based on Python documentation (2021):

- Web Development
- Scientific and Numeric
- Education
- Desktop GUIs
- Software Development
- Business Applications
- Game Development

According to Cass (2021), Python, along with Java, C, C++, and C# is one of the most popular programming languages. Many large firms have made it a higher priority programming language such as Google, IBM, Reddit. Many more companies are beginning to follow this trend and use Python for their development. The first structural reasoning of Python emphasizes code coherence through its extensive use of crucial whitespaces. Its language creates an article-organized approach designed to aid software engineers in writing clear, logical code for small and large-scale projects. It plays the key to the foundation of other programming ideal paradigms such as procedural, object-oriented, and practical programming. (Kuhlman 2012.) As stated by GeeksforGeeks (2021), the following are the most convincing reasons why many developers accept Python that with to fully benefits from all of its features:

- **Open source**: Python is entirely free to download and use. It can already help to reduce project costs. Aside from that, it takes a simple setup to function.

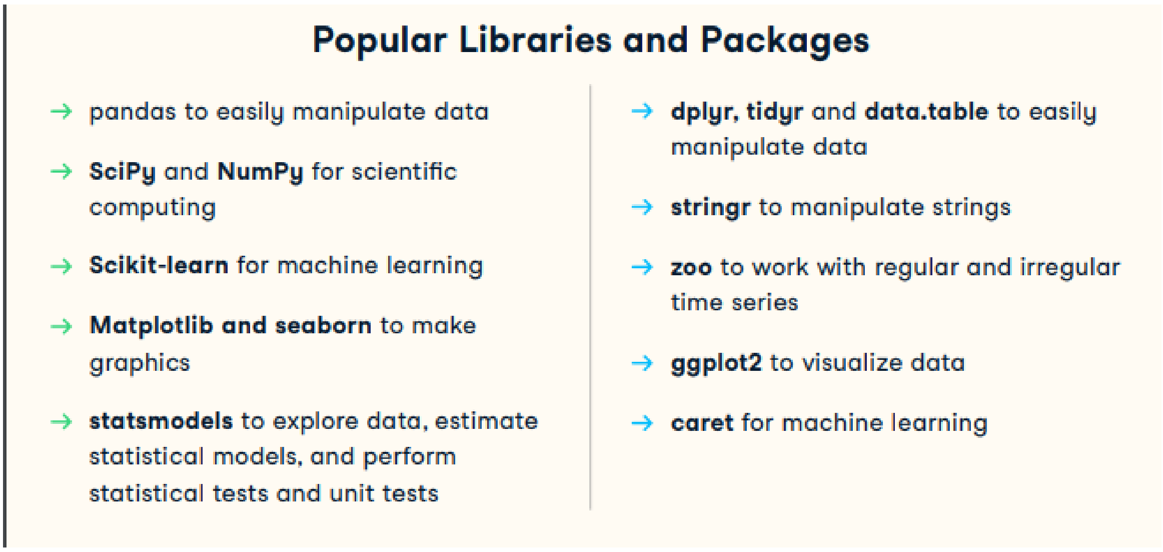
- **Simple syntax**: The syntax is relatively basic, which lowers' syntactical overhead and steepens the learning curve. It is often regarded as an excellent choice for a first programming language.
- **Versatile**: By building, using, and reusing data structures, one may reduce the amount of coding necessary to complete a job.
- **Modules and libraries**: Python has multiple standard libraries and extensions for many types of programming activities.
- **Large community**: Not only is the Python user community huge, but it also provides a variety of assistance, distributes resources, and generally encourages the learning process.

4.1.2 Python and R in data analytics

There are various data analytics programming languages, such as Hadoop, R programming, and SAS. But Python and R are the commonly utilized and simple tools for working in data analytics. The following paragraphs explores the differences that distinguish them and how to select the best one for the special situation.

The primary difference between the two languages is their approach to data analytics. Python is used by programmers to perform data analysis and machine learning in scalable production contexts. For example, coders could use Python to incorporate face recognition into your mobile API or to create a machine learning application. R, on the other hand, is used by data scientists for sophisticated statistical analysis, which is accompanied by a few lines of code and stunning data visualizations. R could be used for customer behavior analysis or genomics research, for example. (IBM team 2021.)

When it comes to data sciences, both languages offer popular libraries and packages to help with analyzing jobs rapidly, but depending on the perspective of the working process, each library serves a specific purpose, as shown in image 3 below. (Datacamp 2021.) Like for instance, Python provides a wide range of file formats, from CSV files to web-sourced JSON. SQL tables can also be imported directly into Python scripts. The Python requests package for web development allows you to easily get data from the web for generating datasets. Whereas R is intended for data analysts to import data from Excel, CSV, and text files. Files created in Minitab or SPSS can also be converted into R data frames. While Python is more versatile for web data extraction, new R utilities such as Rvest are developed for basic web scraping. (IBM team 2021.)



Popular Libraries and Packages

- → pandas to easily manipulate data
- → **SciPy** and **NumPy** for scientific computing
- → **Scikit-learn** for machine learning
- → **Matplotlib and seaborn** to make graphics
- → **statsmodels** to explore data, estimate statistical models, and perform statistical tests and unit tests

- → **dplyr, tidyr** and **data.table** to easily manipulate data
- → **stringr** to manipulate strings
- → **zoo** to work with regular and irregular time series
- → **ggplot2** to visualize data
- → **caret** for machine learning

Image 3. The Python and R popular libraries and Packages (Datacamp 2021)

Both languages are well-acknowledged to be capable of dealing with the bulk of data science challenges, with the rest relying on methodology, team competencies, and available resources, which are mostly independent of the language (Karakan 2020). Therefore, choosing the appropriate language is dependent on their scenario. According to IBM Team (2021), the following questions are some things to think about:

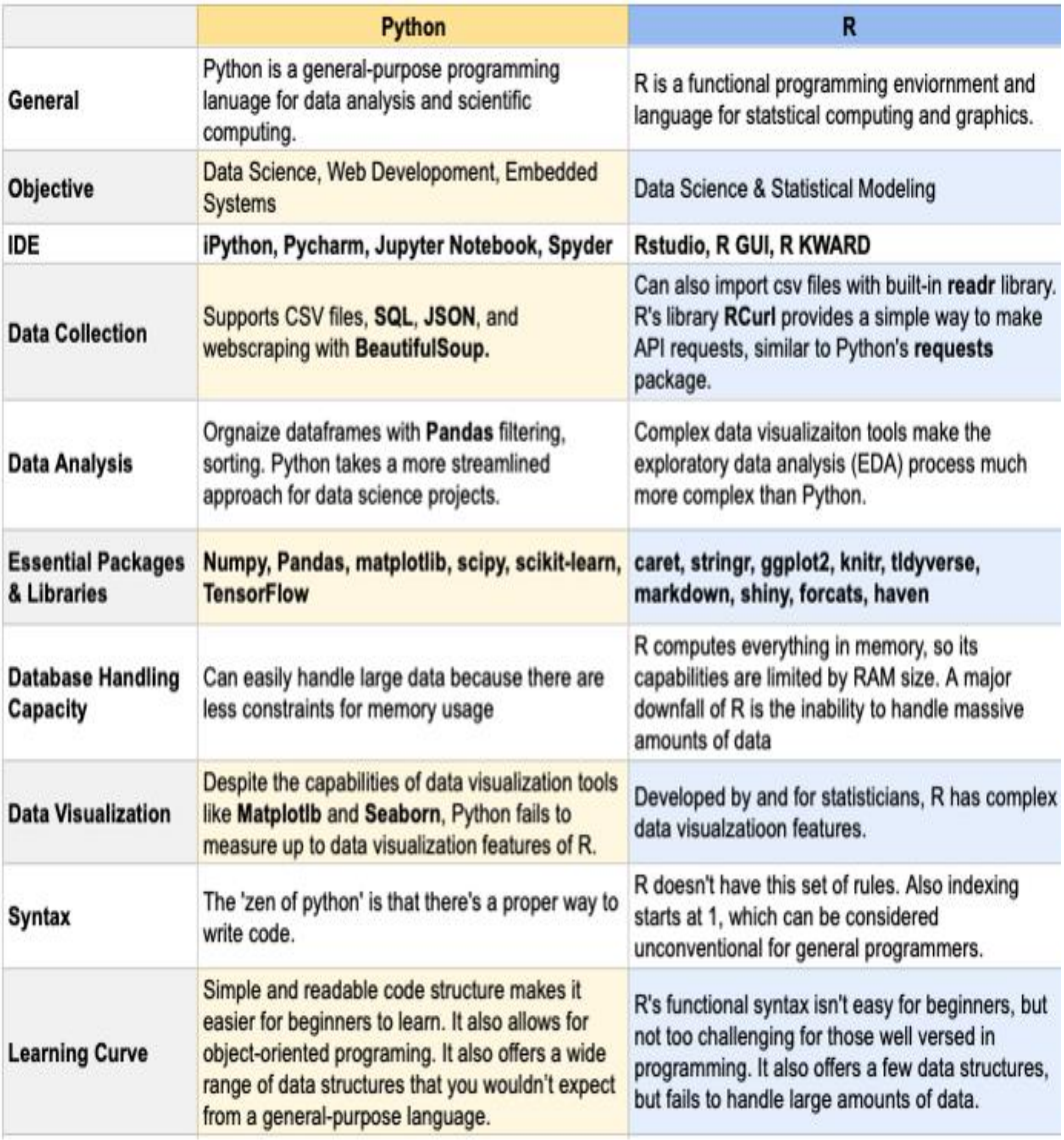
- **What do their colleagues use?**

  R is a statistical tool used by academics, engineers, and scientists who do not have programming expertise. Python is a production-ready language that is used in a variety of industry, research, and engineering workflows.

- **What problems are developers trying to solve?**

  R programming is better suited for statistical learning, with unrivaled data exploration and experimental packages. Python is a better choice for machine learning and large-scale applications, particularly data processing within web applications.

Thus, Python is an appropriate programming language in this project based on the operation of generating data from a website. This research digs into the specific analytical Python libraries used, which include Beautiful Soup and Pandas, as will be stated in the following chapter.



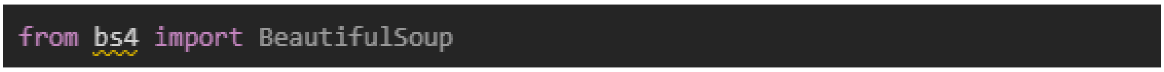
	Python	R
General	Python is a general-purpose programming lanuage for data analysis and scientific computing.	R is a functional programming enviornment and language for statstical computing and graphics.
Objective	Data Science, Web Developoment, Embedded Systems	Data Science & Statistical Modeling
IDE	**iPython, Pycharm, Jupyter Notebook, Spyder**	**Rstudio, R GUI, R KWARD**
Data Collection	Supports CSV files, **SQL**, **JSON**, and webscraping with **BeautifulSoup.**	Can also import csv files with built-in **readr** library. R's library **RCurl** provides a simple way to make API requests, similar to Python's **requests** package.
Data Analysis	Orgnaize dataframes with **Pandas** filtering, sorting. Python takes a more streamlined approach for data science projects.	Complex data visualizaiton tools make the exploratory data analysis (EDA) process much more complex than Python.
Essential Packages & Libraries	**Numpy, Pandas, matplotlib, scipy, scikit-learn, TensorFlow**	**caret, stringr, ggplot2, knitr, tldyverse, markdown, shiny, forcats, haven**
Database Handling Capacity	Can easily handle large data because there are less constraints for memory usage	R computes everything in memory, so its capabilities are limited by RAM size. A major downfall of R is the inability to handle massive amounts of data
Data Visualization	Despite the capabilities of data visualization tools like **Matplotlb** and **Seaborn**, Python fails to measure up to data visualization features of R.	Developed by and for statisticians, R has complex data visualzatioon features.
Syntax	The 'zen of python' is that there's a proper way to write code.	R doesn't have this set of rules. Also indexing starts at 1, which can be considered unconventional for general programmers.
Learning Curve	Simple and readable code structure makes it easier for beginners to learn. It also allows for object-oriented programing. It also offers a wide range of data structures that you wouldn't expect from a general-purpose language.	R's functional syntax isn't easy for beginners, but not too challenging for those well versed in programming. It also offers a few data structures, but fails to handle large amounts of data.

Table 3. The difference between Python and R (Kung 2020)

4.2 Beautiful Soup

Beautiful Soup is a Python library based on the foundation of an HTML/XML analytics engine. It is used for extracting, analyzing, and editing information in the DOM tree of web pages. It supplies a series of concise DOM visitor interfaces, helping developers quickly build a system prototype and obtain experiment data. Additionally, it has high cross-platform flexibility. Beautiful Soup helps users extract specific material from the webpage, remove the HTML syntax and save the relevant data. It is a web scraping library to assist developers with the range of tasks involving isolating titles and links to pulling all contents by following the Beautiful Soup documentation. (Wieringa 2012.)

It was built firstly by Leonard Richardson in 2004. The popular current version is BeautifulSoup 4.8.1 and works with Python 2.7, Python 3.2, or the other higher versions. (Beautiful Soup documentation 2021.) This implementation will use Beautiful Soup 4.8.1 and Python version 3.2.



```
from bs4 import BeautifulSoup
```

Image 4. Syntax of importing package in Beautiful Soup 4 (Beautiful Soup documentation 2021)

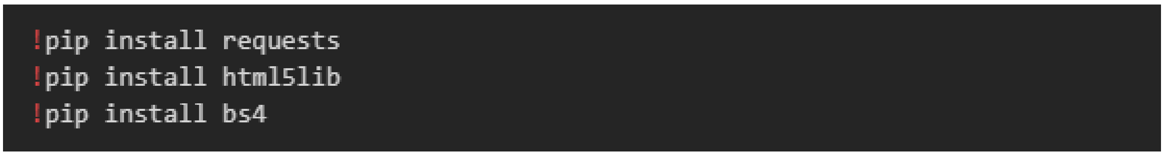
4.2.1 Web scraping with Beautiful Soup

There are two methods for scraping data from a website. The first method is extracting data by using the website's API if the website provides. The second method is accessing the webpage's HTML structure and extracting valuable data. The second method is called web scraping. Beautiful Soup is a Python library for serving web scraping purposes.

According to Kumar (2021), the implementation of web scraping by using the Beautiful Soup framework includes 3 main steps.

- **Step 1: Installing the necessary third-party libraries**

Using pip syntax is the simplest way to install external libraries in Python. It is a package management system for installing and managing Python-based software packages. Developers prefer to use the command prompt in the window to directly install libraries rather than writing it in JupyterNotebook. The following figure refers to the necessary third-party HTTP libraries for Python requests.

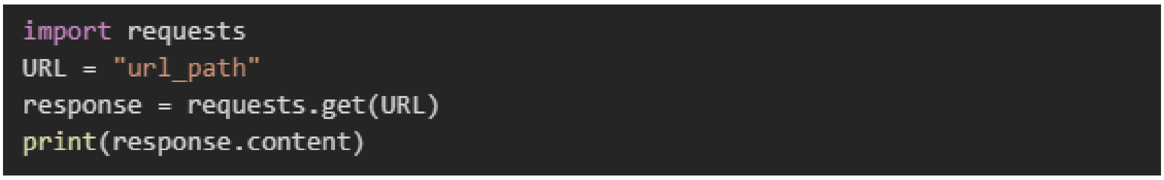


```
!pip install requests
!pip install html5lib
!pip install bs4
```

Image 5. Screenshot from code script: the syntax of installing the Python libraries in the JupyterNotebook environment

- **Step 2: Pulling HTML text from a website**

After installing the libraries, it needs to import into the Python code environment for use. The given URL's website will be entered to send an HTTP request and save the server answer in a response object. The webpage's raw HTML is retrieved by printing the content of the server answer.



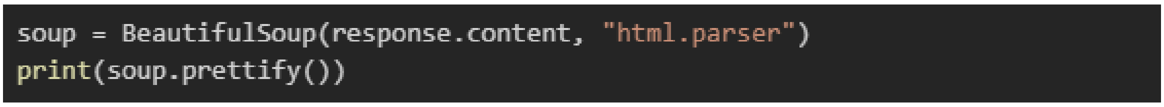
```
import requests
URL = "url_path"
response = requests.get(URL)
print(response.content)
```

Image 6. Screenshot from example code script: the syntax of extracting HTML text

- **Step 3: Parsing the HTML content**

The Beautiful Soup library has the advantage of being built on top of HTML parsing libraries such as html5lib, XML, HTML. parser and so on. The response object and the parser library can be coded at the same time as two different arguments:

- Response.content: it illustrates the raw HTML content
- HTML.parser: the particular parser library would like to be used



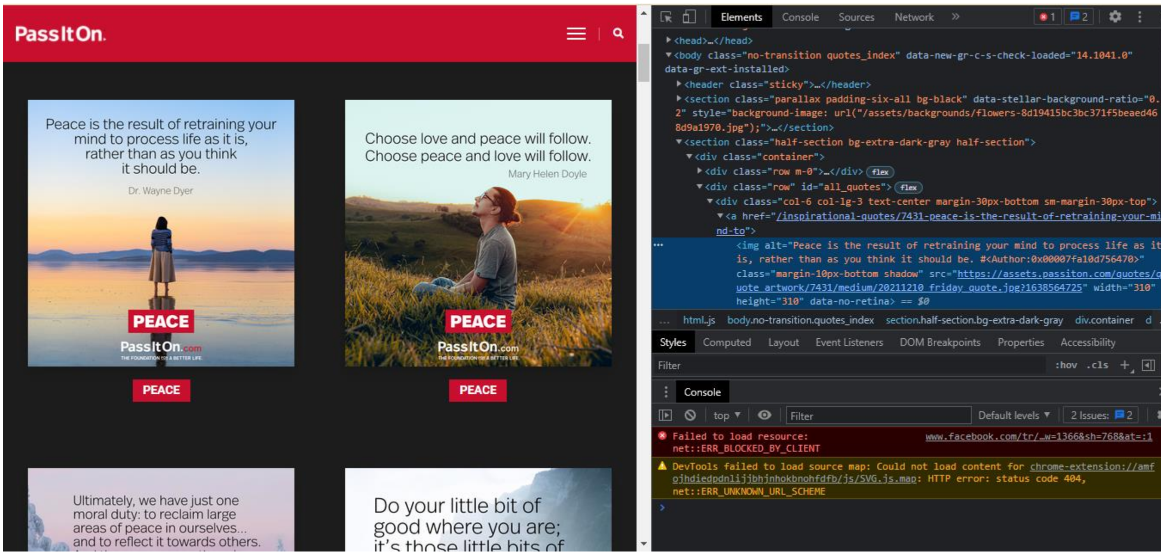
```
soup = BeautifulSoup(response.content, "html.parser")
print(soup.prettify())
```

Image 7. Screenshot from example code script: the syntax of parsing the HTML content

In addition, soup. prettify() is printed for displaying a visual representation of the parse tree which is generated from the HTML text

- **Step 4: Navigating and searching the parse tree**

It is the final step to extract the information from HTML text. The subject called soup above included all of the data in the hierarchical structure that can be retrieved programmatically. By understanding the webpage structure, developers can extract any information inside or outside tags to serve their purpose. Coders can acquire access to the HTML structure by diving into the source code of web browsers utilizing the inspecting functions. The appropriate things will light up in the element part when users move their mouse over the HTML language on the webpage. This approach assists in determining which HTML tags provide the needed information for scraping. Finally, the find() and findAll() function is built to find and scrape all of the information contained within the tag.


PassItOn.
Peace is the result of retraining your mind to process life as it is, rather than as you think it should be.
Dr. Wayne Dyer
PEACE
PassItOn.com
PEACE
Choose love and peace will follow. Choose peace and love will follow.
Mary Helen Doyle
PEACE
PassItOn.com
PEACE
Ultimately, we have just one moral duty: to reclaim large areas of peace in ourselves... and to reflect it towards others.
Do your little bit of good where you are; it's those little bits of
Elements
Console
Sources
Network
1
2
html.js
body.no-transition.quotes_index
section.half-section.bg-extra-dark-gray
div.container
Styles
Computed
Layout
Event Listeners
DOM Breakpoints
Properties
Accessibility
Filter
:hov
.cls
Console
top
Filter
Default levels
2 Issues: 2

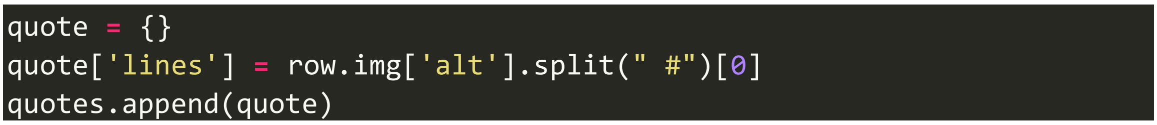
```
▸<head>…</head>
▾<body class="no-transition quotes_index" data-new-gr-c-s-check-loaded="14.1041.0"
data-gr-ext-installed>
  ▸<header class="sticky">…</header>
  ▸<section class="parallax padding-six-all bg-black" data-stellar-background-ratio="0.
  2" style="background-image: url("/assets/backgrounds/flowers-8d19415bc3bc371f5beaed46
  8d9a1970.jpg");">…</section>
  ▾<section class="half-section bg-extra-dark-gray half-section">
    ▾<div class="container">
      ▸<div class="row m-0">…</div> flex
      ▾<div class="row" id="all_quotes"> flex
        ▾<div class="col-6 col-lg-3 text-center margin-30px-bottom sm-margin-30px-top">
          ▾<a href="/inspirational-quotes/7431-peace-is-the-result-of-retraining-your-mi
          nd-to">
...            <img alt="Peace is the result of retraining your mind to process life as it
             is, rather than as you think it should be. #<Author:0x00007fa10d756470>"
             class="margin-10px-bottom shadow" src="https://assets.passiton.com/quotes/q
             uote_artwork/7431/medium/20211210_friday_quote.jpg?1638564725" width="310"
             height="310" data-no-retina> == $0
```

```
Failed to load resource:                    www.facebook.com/tr/…w=1366&sh=768&at=:1
net::ERR_BLOCKED_BY_CLIENT
DevTools failed to load source map: Could not load content for chrome-extension://amf
ojhdiedpdnlijjbhjnhokbnohfdfb/js/SVG.js.map: HTTP error: status code 404,
net::ERR_UNKNOWN_URL_SCHEME
>
```

Image 8. Screenshot from PassItOn’s website: the user interface of the web inspecting function

Image 8 above depicts the web inspecting function and how it works. Developers, for example, would like to scrape the titles of articles. When coders click on the title on the right side, the highlighted code line appears. The tag that includes the title of the article is div, and the id name is 'all_quotes'. It is now time to locate the tag using the find() and findAll() technique as shown in image 9 below.

```
table = soup.find('div', attrs = {'id':'all_quotes'})

for row in table.findAll('div',attrs = {'class':'col-6 col-lg-3 text-center margin-30px-bottom sm-margin-30px-top'}):
```



```
quote = {}
quote['lines'] = row.img['alt'].split(" #")[0]
quotes.append(quote)
```

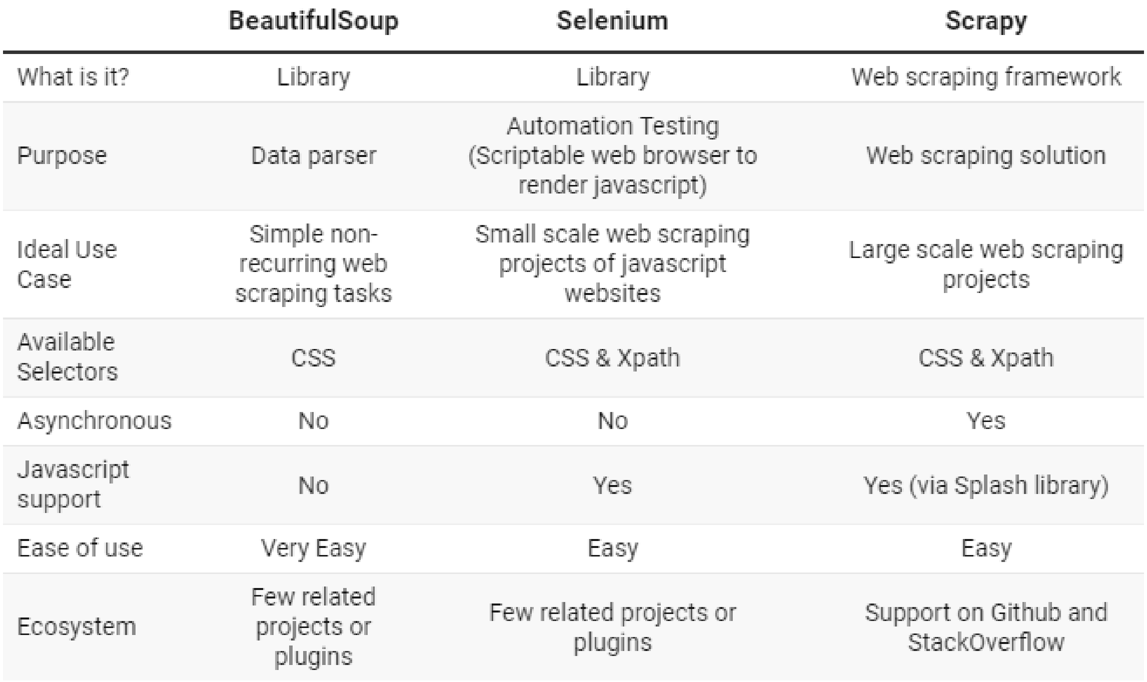
Image 9. The code script example of scraping the PassItOn website (Kumar 2021)

4.2.2 Beautiful Soup and other frameworks

While web scraping provides the various task sizes and websites, numerous libraries can be utilized to make the task simpler. Besides Beautiful Soup, Scrapy and Selenium are also popular Python libraries. When it comes to selecting a specific library to perform web scraping operations, people must consider several key factors because of its own set of pros and cons. Therefore, this second sub-chapter illustrates the comparison criteria of each library. There are three essential points which they must emphasize. (Andrade 2020.)

- **Ideal use case**: Scrapy is the ideal solution for a large-scale project due to its architecture and capabilities. It also makes project migration easier which is advantageous for large projects. Beautiful Soup is better suited for tiny and basic projects, but Selenium falls somewhere in the middle. Selenium can extract data from a website that utilizes JavaScript.
- **Performance**: Scrapy is the fastest since it is asynchronous, developed specifically for web scrapping, and written in Python. Beautiful Soup and Selenium, on the other hand, are ineffective when it comes to scraping huge volumes of data.
- **Ease of use**: Beautiful Soup is the most user-friendly web scraping tool. The reason is its simplicity and clear approach to assist novices in learning quickly. Selenium and Scrapy are harder to learn and require more time and effort to achieve.

According to Andrade (2020), table 4 presented in detail the differences difference between Beautiful Soup, Selenium, and Scrapy.



	BeautifulSoup	Selenium	Scrapy
What is it?	Library	Library	Web scraping framework
Purpose	Data parser	Automation Testing (Scriptable web browser to render javascript)	Web scraping solution
Ideal Use Case	Simple non-recurring web scraping tasks	Small scale web scraping projects of javascript websites	Large scale web scraping projects
Available Selectors	CSS	CSS & Xpath	CSS & Xpath
Asynchronous	No	No	Yes
Javascript support	No	Yes	Yes (via Splash library)
Ease of use	Very Easy	Easy	Easy
Ecosystem	Few related projects or plugins	Few related projects or plugins	Support on Github and StackOverflow

Table 4. The difference between Beautiful Soup, Selenium, and Scrapy (Andrade 2020)

4.3 Pandas

Pandas is a Python package that provides numerous fast, expressible, and flexible data structures to allow Python to interact with relational or labeled data simply. Pandas can be used for multiple data analysis purposes. This sub-chapter introduces the Pandas library and demonstrates how to work with Pandas DataFrames such as reading, exporting, or aggregating data frames. In addition, it also illustrates data cleaning with Pandas.

4.3.1 Data collection with Pandas

There are several available file formats or data sources that data analysts need to import for analyzing and then interpret in another format when finishing the process. Pandas enable integration with a wide range of popular file types such as CSV, Excel, SQL, JSON, or Parquet. Each of these data sources can be imported using a function with the prefix read_* and be read into a pandas dataframe. The to_* methods are used to store data in the same way as shown in image 10 below. When presenting a DataFrame, Pandas has many methods and attributes that can be used to always double-check the data after reading it in. For a first check, the head/tail/info method and dtypes attribute are useful. (Pandas documentation 2021a.)

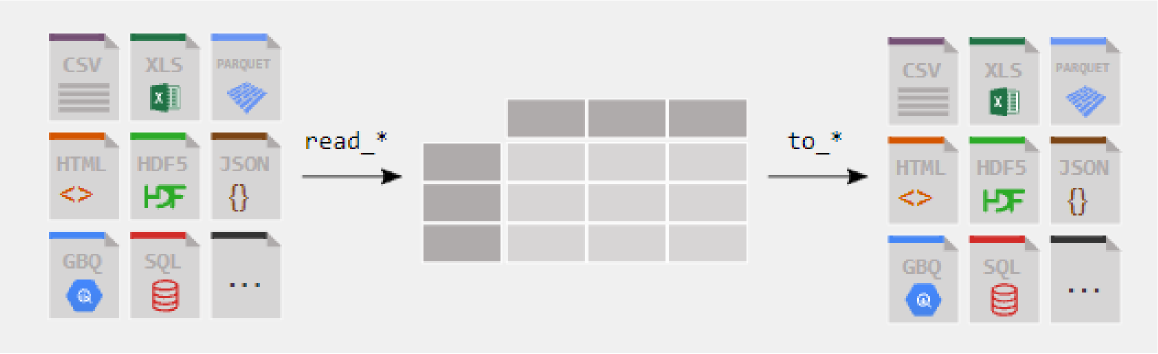
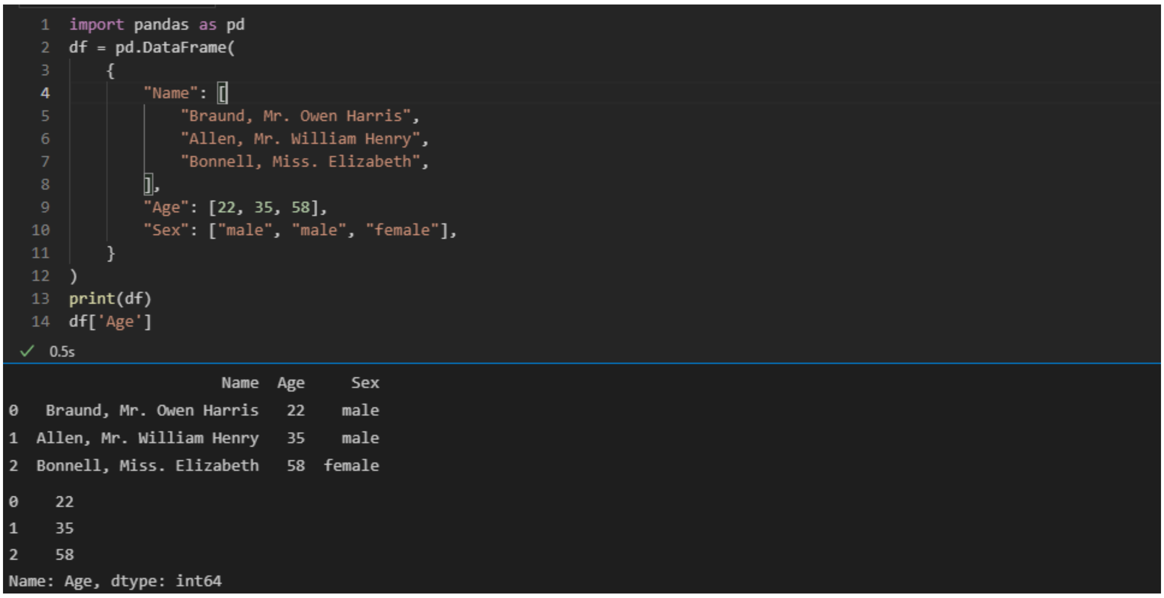


Image 10. How to read and write tabular data in Pandas (Pandas documentation 2021a)

Pandas is the tool for working with tabular data such as data contained in spreadsheets or databases. Data is referred to create a DataFrame to manually store in a table. When implementing a Python dictionary of the list, the dictionary keys serve as the column headings, and the values in each list serve as DataFrame columns. A DataFrame is a two-dimensional data structure that may contain various sorts of data in columns. It is comparable to a spreadsheet, a SQL table, or the data.frame function in R program language. Every single column of Pandas DataFrame is a Pandas Series. It can be picked by using the column label in between square brackets. (Pandas documentation 2021b.)



```
import pandas as pd
df = pd.DataFrame(
    {
        "Name": [
            "Braund, Mr. Owen Harris",
            "Allen, Mr. William Henry",
            "Bonnell, Miss. Elizabeth",
        ],
        "Age": [22, 35, 58],
        "Sex": ["male", "male", "female"],
    }
)
print(df)
df['Age']
```

```
                       Name  Age     Sex
0   Braund, Mr. Owen Harris   22    male
1  Allen, Mr. William Henry   35    male
2  Bonnell, Miss. Elizabeth   58  female

0    22
1    35
2    58
Name: Age, dtype: int64
```

Image 11. The code script example of dataframe and the output (Pandas documentation 2021b)

4.3.2 Data cleaning with Pandas

According to Molin (2019, 49-50), data cleaning is the process of preparing data and converting it to a format that can be analyzed. The terrible fact of data is frequently unclean which means that it must be cleaned before it can be used. There are some potential problems with the data as below:

- **Human error:** It occurs when data is entered inaccurately such as entering 100 instead of 1000 or making typos. Furthermore, the same words may be recorded in many versions such as New York City, NYC, or nyc.
- **Computer error:** It leads to missing data because people record entries in a long format.
- **Unexpected values**: All entries in the column are handled as string types rather than numeric values because of a missing value in a numeric column
- **Incomplete information:** Data is collected without missing information because participants do not fill in. For example, some people do not answer the optional questions in the survey.
- **Resolution:** The data may have been collected per second when the research requires hourly data.
- **Relevance of the field:** Data is frequently collected or generated as a consequence of the process rather than purposefully for analysis. Therefore, it needs to tidy up to make it usable
- **Format of the data:** Data is captured in an unsuitable format, necessitating its reshaping.

Most of these data quality concerns are resolvable. The responsibility of data analysts is to carefully evaluate the data and deal with any issues that are analyzed to get biased. This procedure can be handled in detail by Data Cleaning with Pandas. (Molin 2019, 49-50.)

Furthermore, duplicate rows in CSV files can be deleted by combining Pandas and Python algorithms. To begin, the open() function will open two separate CSV files, one as an input file and one as an output file. The input file is an existing file that contains raw data that must be cleaned. The output file is the new file that contains the data once the duplicate data has been removed. The CSV data will be converted into a dictionary so that the Python algorithms may be applied. Creating a loop examines each row in the column for duplicate rows. If one row matches the data in another row, the first row will be eliminated.

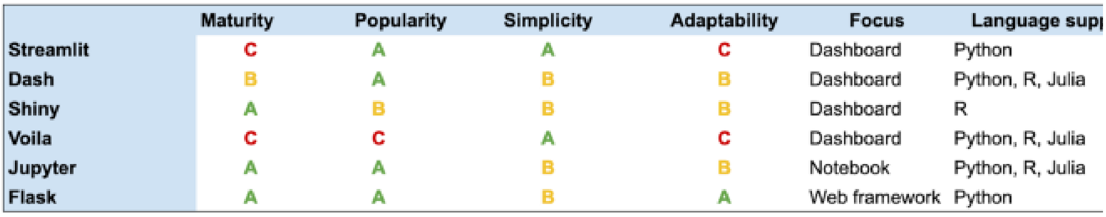
4.4 Streamlit

Streamlit is an open-source Python framework for creating and sharing attractive, personalized web applications for machine learning and data science (Streamlit 2021a). The subchapter presents the comparison between Streamlit and other frameworks. It also provides instructions on how to create a web application by Streamlit.

4.4.1 Streamlit and other frameworks

There are several components of a data dashboard that include analyzing, visualizing, interacting, and serving. Developers spent a lot of time and effort generating all of the glue code to connect these components in the past. But until now, in recent libraries like Streamlit and Dash, these components are combined into a single package. Therefore, developers need to decide which library to use can be difficult. The following list is the criteria that how people often use to compare and choose which one is suitable for their project . (Schmitt 2020.)

- **Maturity:** It is determined by the project's age and stability.
- **Popularity**: It is measured by adoption and GitHub stars.
  - The two most popular libraries are Streamlit and Dash which are both complete dashboarding solutions written in Python.
- **Simplicity:** The ease with which the library can be used.
  - Streamlit is more structured and emphasizes simplicity. It solely supports Python-based data analysis and has a restricted number of widgets (such as sliders) from which to pick.
- **Adaptability**: It is assessed by how adaptable and opinionated the library is.
  - Dash is the more adaptable and mature library. Although it is written in Python and encourages users to its plotting library, it is also compatible with other charting libraries and languages such as R or Julia.
- **Focus:** It is defined by the problem it solves.
- **Language support:** The main language supported by the library.



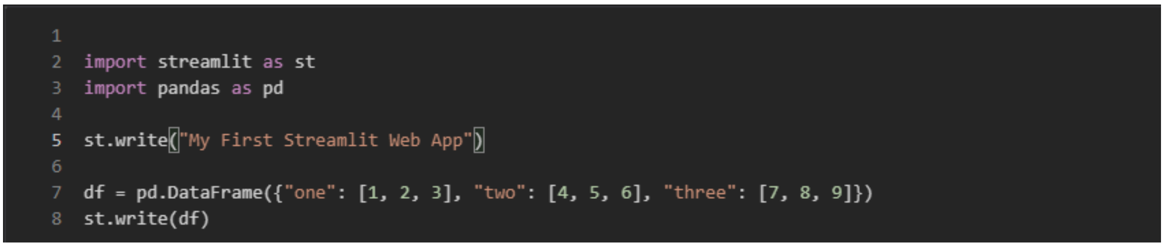
	Maturity	Popularity	Simplicity	Adaptability	Focus	Language sup
Streamlit	C	A	A	C	Dashboard	Python
Dash	B	A	B	B	Dashboard	Python, R, Julia
Shiny	A	B	B	B	Dashboard	R
Voila	C	C	A	C	Dashboard	Python, R, Julia
Jupyter	A	A	B	B	Notebook	Python, R, Julia
Flask	A	A	B	A	Web framework	Python

Table 4. Comparing data dashboarding tools and frameworks (Schmitt 2020)

In conclusion, according to table 4 above, developers can acquire a quick situation need to choose Streamlit as a solution. Streamlit is suitable for developers who already use Python for analytics and want to get a prototype of the dashboard up and running as soon as feasible.

4.4.2 Creating a web application by Streamlit

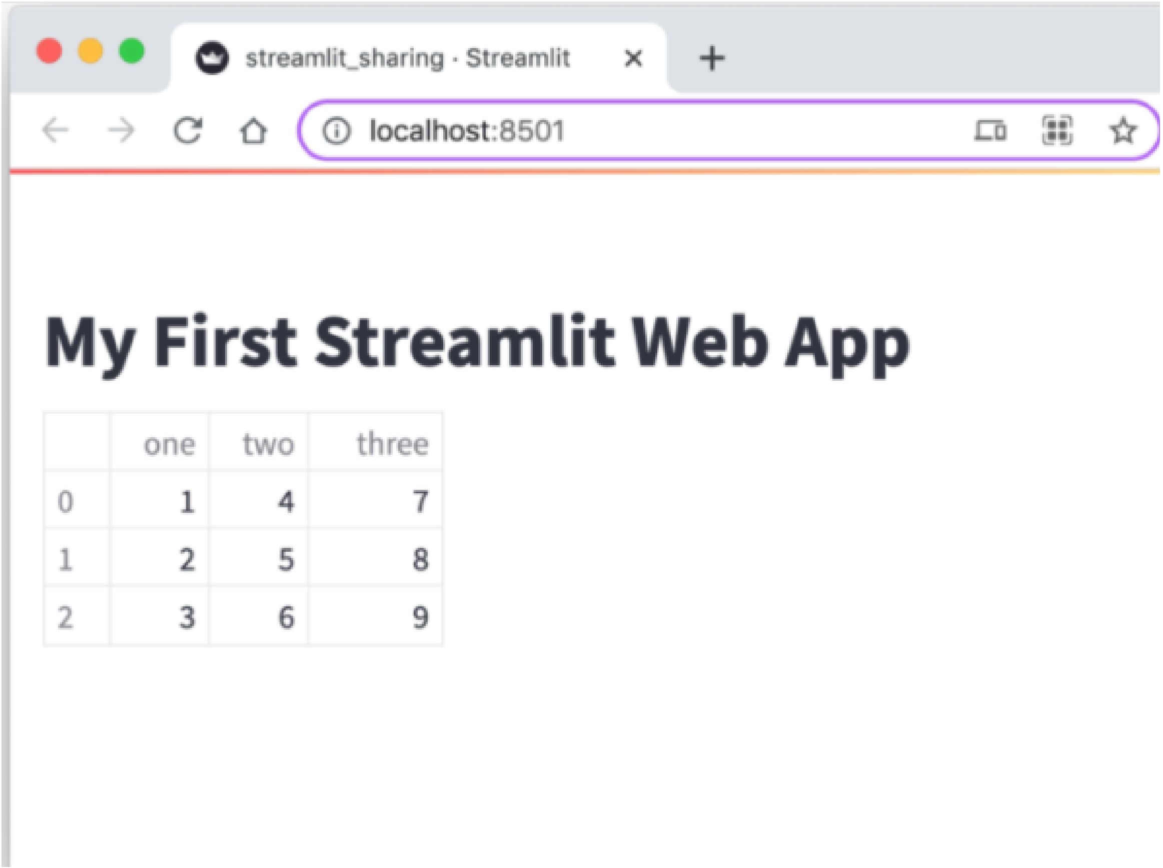
It is easy to start building a dashboard application via Streamlit within three main steps. Firstly, developers need to install Streamlit package into the coding environment. Streamlit can be installed by using the pip tool: pip install streamlit. After installing Streamlit environment, people can start to create the web application’s script. The following image 12 describes the writing of the scrips in Python language that developers can establish an application via Streamlit.



```

import streamlit as st
import pandas as pd

st.write("My First Streamlit Web App")

df = pd.DataFrame({"one": [1, 2, 3], "two": [4, 5, 6], "three": [7, 8, 9]})
st.write(df)
```

Image 12. The code script example of creating a web application via Streamlit (Cui 2021)

Finally, coders would like to check out the interface of the application. To execute the program locally, people need to enter the code line in the terminal: Streamlit run ‘path’.py. Developers must ensure the path is the directory of the Python script saved. If they are not in the same directory, the complete path of the file will be used instead. After running the command, a web browser will run automatically to present the web application as shown in image 13 below.



My First Streamlit Web App

	one	two	three
0	1	4	7
1	2	5	8
2	3	6	9

Image 13. Screenshot from the output of Streamlit web application in local URL (Cui 2021)

Instead of publishing on localhost, developers can put the application on a public web server that anybody can access. Streamlit and Github are used in tandem to complete the application deployment process. Before deploying the web application, the web application folders, including code scripts and other related documents, must commit to a new public repository on GitHub. Because Streamlit Cloud launches application directly from their GitHub repository. According to Streamlit (2021b), the following list illustrates step by step how to publish an web application to the public webserver.

- **Step 1:** After running the code script on localhost, click "Deploy this app" in the upper right corner of the web browser to open the setting window.

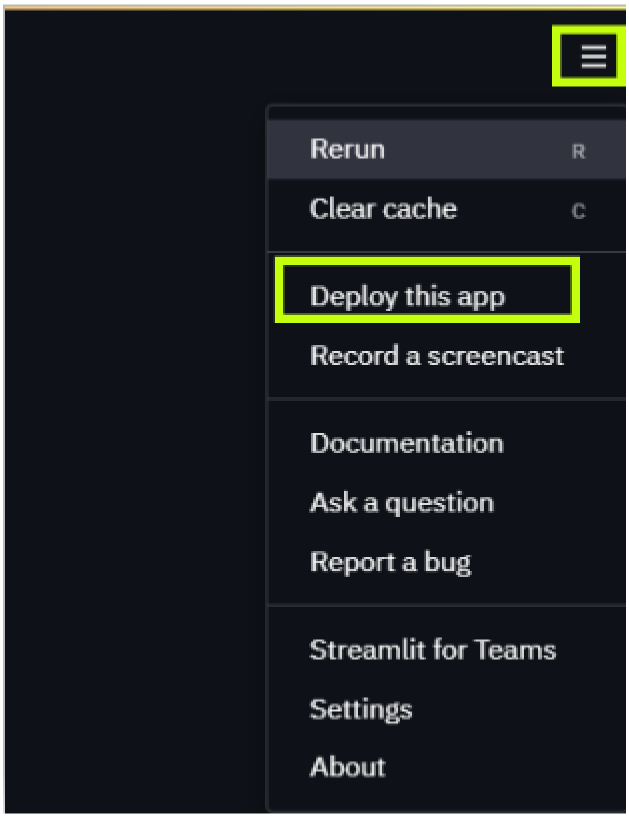
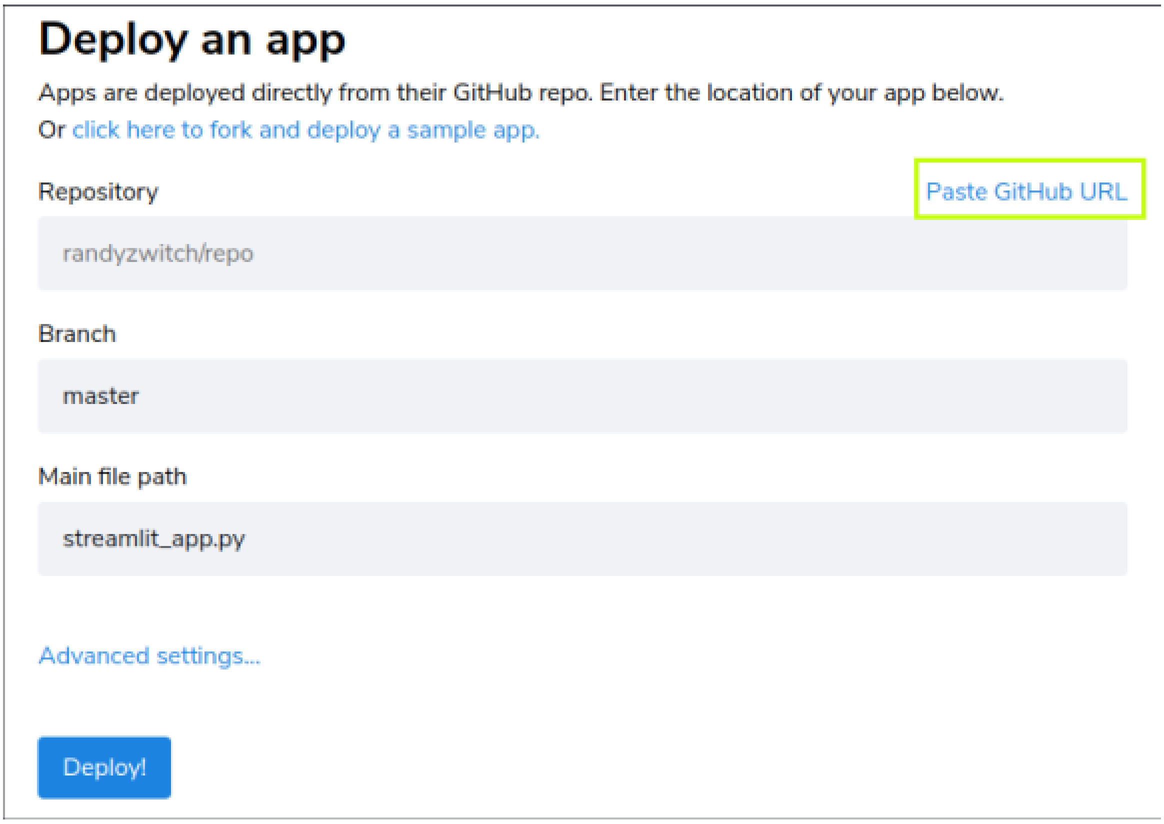


Image 14. Streamlit deploying the web application button

- **Step 2:** Then developers need to enter the repo, branch, and file path before clicking "Deploy." Coders may also click "Paste GitHub URL" as a shortcut.



Deploy an app

Apps are deployed directly from their GitHub repo. Enter the location of your app below.
Or click here to fork and deploy a sample app.

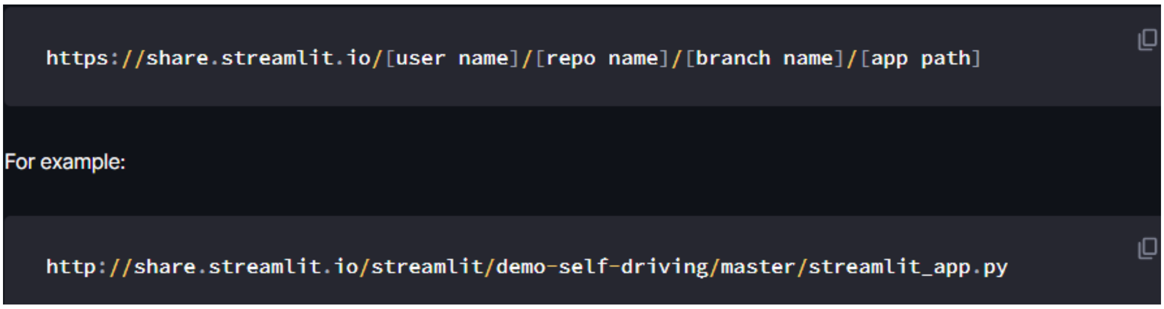
Repository: randyzwitch/repo

Branch: master

Main file path: streamlit_app.py

Image 15.The example of deploying the web application setting window (Streamlit 2021b)

The web application is deployed in two steps. Most web applications are deployed in a matter of minutes. However, if the web application contains many parts, the first deployment may take some time. Any adjustments do not have an impact on their dependencies. They should be seen immediately following the initial deployment. (Streamlit 2021b.) The public URL is generated following the execution of the web application on the cloud. The URL format will be generated like image 16.



```
https://share.streamlit.io/[user name]/[repo name]/[branch name]/[app path]
```

For example:

```
http://share.streamlit.io/streamlit/demo-self-driving/master/streamlit_app.py
```

Image 16. The public URL format of web application (Streamlit 2021b)

5 Implementation

5.1 Phase 1: Data collecting

Building a multi-functional application takes a lot of time and effort. As a result, dividing the process into small phases ensures that every stage of the application is meticulously micro-managed. In this project, the customer needs and wireframe is utilized as a reference to keep the project on track. Based on the requirements mentioned in chapter 2, the planning phase divides the development process into four steps. Collecting data is the first step of the development process.

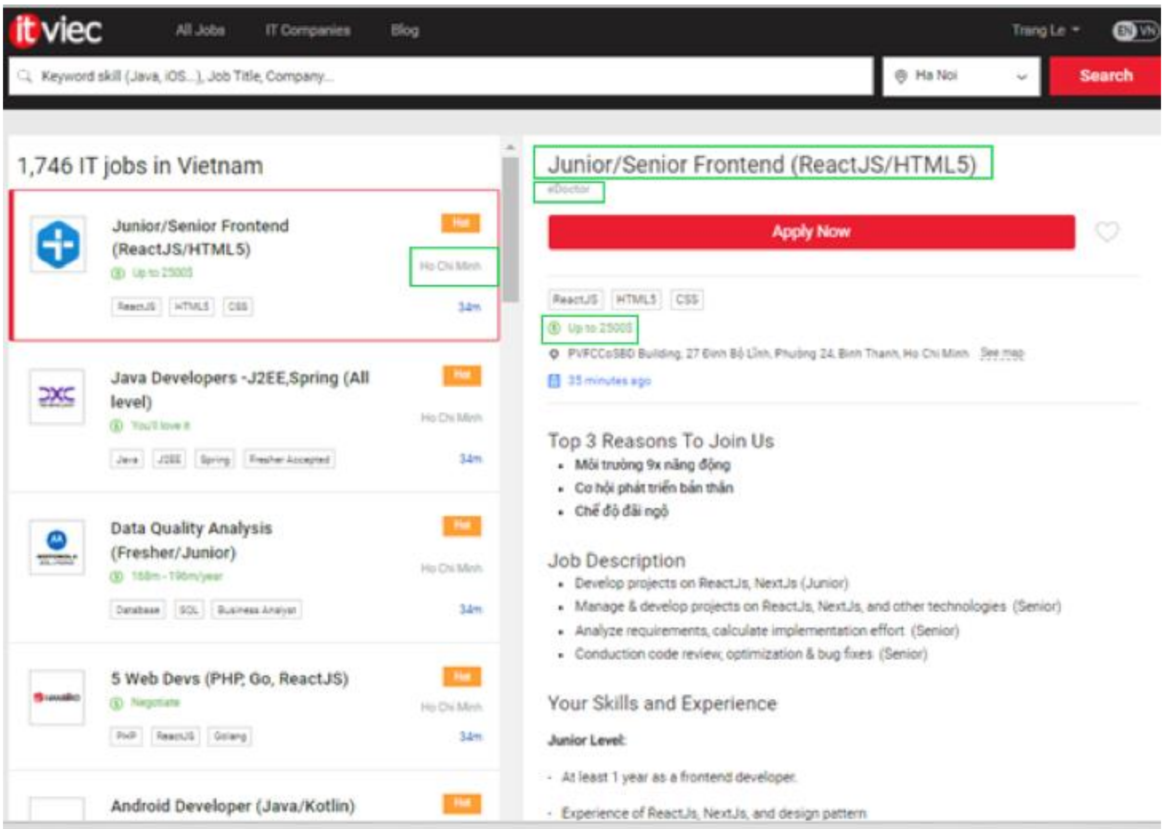
5.1.1 Table structure

The table is a type of data structure that arranges data into rows and columns. It is capable of storing and displaying structured data. (Techterms 2021.) According to the defining demand of users, there are four fields needed such as job name, salary, company, and location. In addition, it will be better if the application can have more detailed information including a job link to access the job description, company link to catch up the review of this company.

Spreadsheets were originally developed for a single user, and their qualities reflect this. They are ideal for a single or small group of users who do not need to perform a lot of highly complex data manipulation. (Oracle 2021.) In addition, using a spreadsheet is a suitable way to store data in this project with a small size and simple table structure.

5.1.2 Data scraping

Beautiful Soup is the Python library to use for scraping data from the ITviec website. Image 17 is the interface of this website. The green squares have marked all information needed to scrap on the website.



1,746 IT jobs in Vietnam

Junior/Senior Frontend (ReactJS/HTML5)
Up to 2500$
ReactJS HTML5 CSS
Ho Chi Minh
34m

Java Developers -J2EE,Spring (All level)
You'll love it
Java J2EE Spring Fresher Accepted
Ho Chi Minh
34m

Data Quality Analysis (Fresher/Junior)
168m - 196m/year
Database SQL Business Analyst
Ho Chi Minh
34m

5 Web Devs (PHP, Go, ReactJS)
Negotiate
PHP ReactJS Golang
Ho Chi Minh
34m

Android Developer (Java/Kotlin)

Junior/Senior Frontend (ReactJS/HTML5)
eDoctor
ReactJS HTML5 CSS
Up to 2500$
PVFCCoSBD Building, 27 Đinh Bộ Lĩnh, Phường 24, Bình Thạnh, Ho Chi Minh See map
35 minutes ago

Top 3 Reasons To Join Us
- Môi trường 9x năng động
- Cơ hội phát triển bản thân
- Chế độ đãi ngộ

Job Description
- Develop projects on ReactJs, NextJs (Junior)
- Manage & develop projects on ReactJs, NextJs, and other technologies (Senior)
- Analyze requirements, calculate implementation effort (Senior)
- Conduction code review, optimization & bug fixes (Senior)

Your Skills and Experience

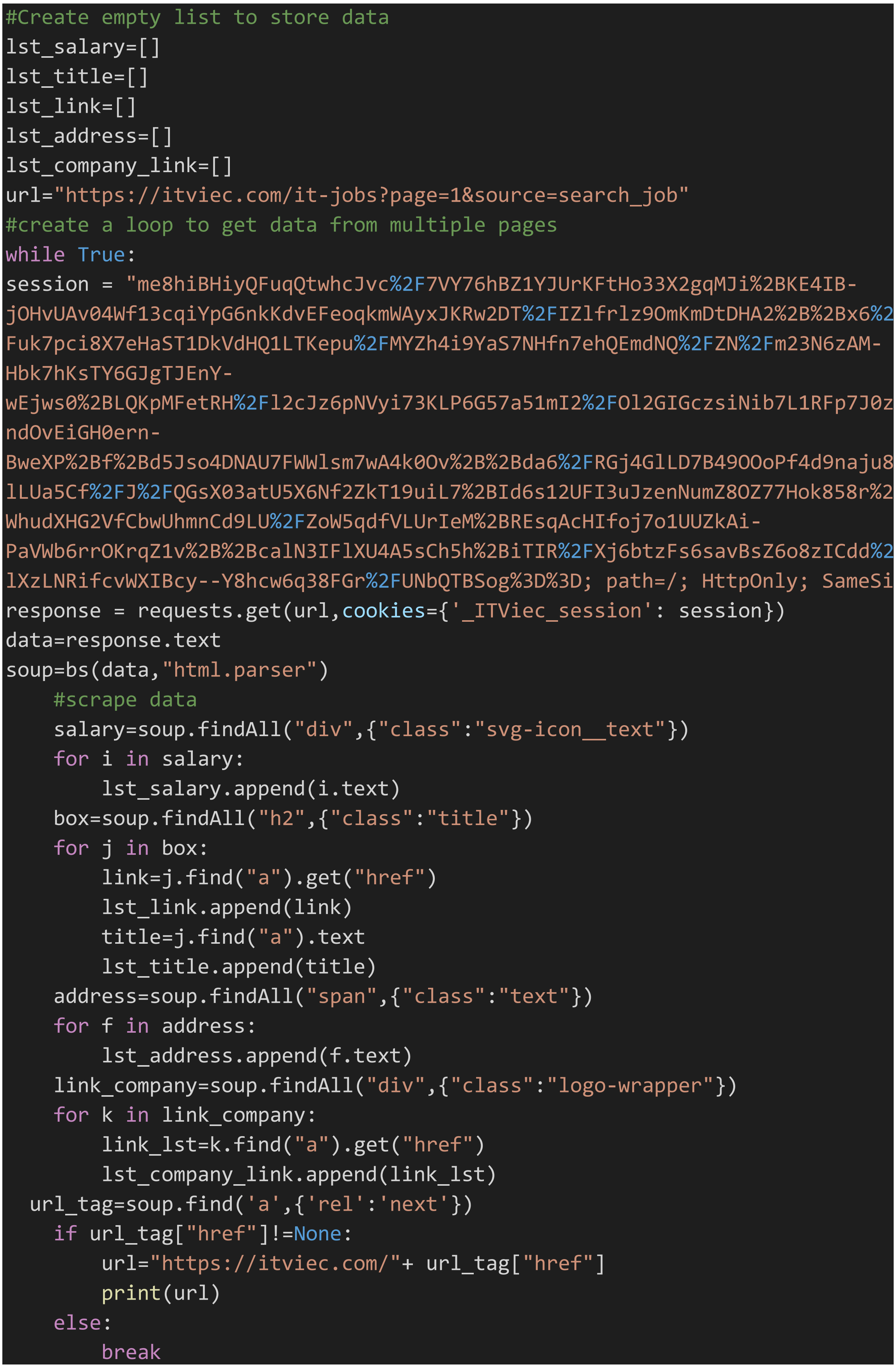
Junior Level:

- At least 1 year as a frontend developer.
- Experience of ReactJs, NextJs, and design pattern

Image 17. Screenshot from ITViec.com ‘s website: the user interface of the website

According to the theoretical instruction, all step by step of the scraping process will be applied. Beautiful Soup is the main library. However, there are some other libraries needed to support the development. The Request library allows sending HTTP/1.1 requests simply without adding manually the query string in the URL or form-encode POST data. Datetime library supports saving the information related to date-time type. Another library is Pandas which handle the job to save database into CSV files.

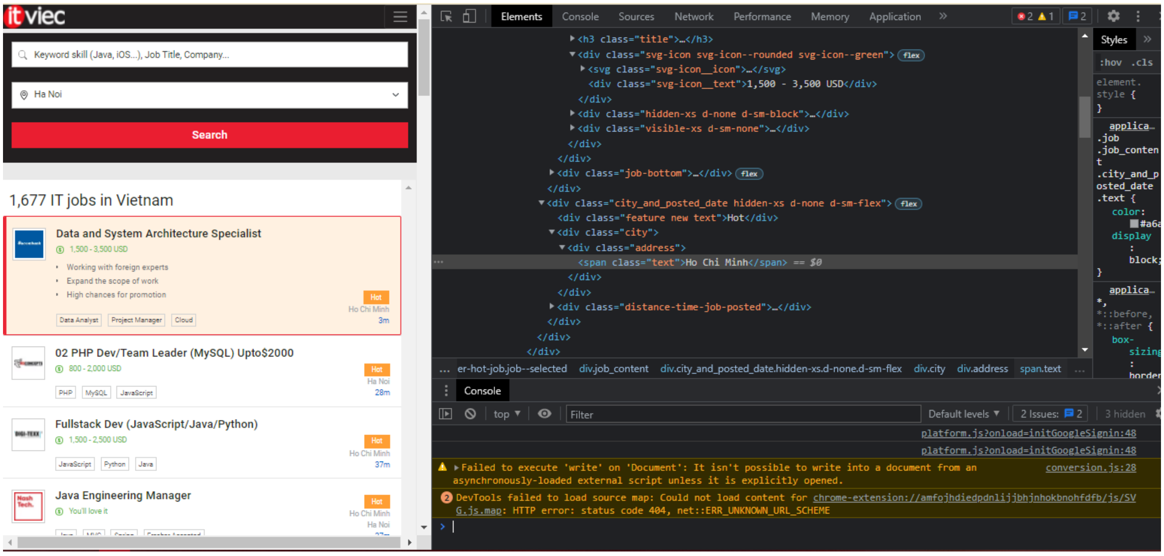
After that, the code script of the scraping process will be executed as shown in image 18 below. The website requires logging in to access the salary information. Thus, developers must include an ID in each request's cookie to fool the server into thinking the request is coming from a logged-in user. The information is stored in multiple pages with the same HTML structure, so implementing a loop is the best way to obtain data on each page. The find() and findAll() functions are designed to locate and scrape all of the needed data included within the tag.



```
#Create empty list to store data
lst_salary=[]
lst_title=[]
lst_link=[]
lst_address=[]
lst_company_link=[]
url="https://itviec.com/it-jobs?page=1&source=search_job"
#create a loop to get data from multiple pages
while True:
session = "me8hiBHiyQFuqQtwhcJvc%2F7VY76hBZ1YJUrKFtHo33X2gqMJi%2BKE4IB-
jOHvUAv04Wf13cqiYpG6nkKdvEFeoqkmWAyxJKRw2DT%2FIZlfrlz9OmKmDtDHA2%2B%2Bx6%2
Fuk7pci8X7eHaST1DkVdHQ1LTKepu%2FMYZh4i9YaS7NHfn7ehQEmdNQ%2FZN%2Fm23N6zAM-
Hbk7hKsTY6GJgTJEnY-
wEjws0%2BLQKpMFetRH%2Fl2cJz6pNVyi73KLP6G57a51mI2%2FOl2GIGczsiNib7L1RFp7J0z
ndOvEiGH0ern-
BweXP%2Bf%2Bd5Jso4DNAU7FWWlsm7wA4k0Ov%2B%2Bda6%2FRGj4GlLD7B49OOoPf4d9naju8
lLUa5Cf%2FJ%2FQGsX03atU5X6Nf2ZkT19uiL7%2BId6s12UFI3uJzenNumZ8OZ77Hok858r%2
WhudXHG2VfCbwUhmnCd9LU%2FZoW5qdfVLUrIeM%2BREsqAcHIfoj7o1UUZkAi-
PaVWb6rrOKrqZ1v%2B%2BcalN3IFlXU4A5sCh5h%2BiTIR%2FXj6btzFs6savBsZ6o8zICdd%2
lXzLNRifcvWXIBcy--Y8hcw6q38FGr%2FUNbQTBSog%3D%3D; path=/; HttpOnly; SameSi
response = requests.get(url,cookies={'_ITViec_session': session})
data=response.text
soup=bs(data,"html.parser")
    #scrape data
    salary=soup.findAll("div",{"class":"svg-icon__text"})
    for i in salary:
        lst_salary.append(i.text)
    box=soup.findAll("h2",{"class":"title"})
    for j in box:
        link=j.find("a").get("href")
        lst_link.append(link)
        title=j.find("a").text
        lst_title.append(title)
    address=soup.findAll("span",{"class":"text"})
    for f in address:
        lst_address.append(f.text)
    link_company=soup.findAll("div",{"class":"logo-wrapper"})
    for k in link_company:
        link_lst=k.find("a").get("href")
        lst_company_link.append(link_lst)
  url_tag=soup.find('a',{'rel':'next'})
    if url_tag["href"]!=None:
        url="https://itviec.com/"+ url_tag["href"]
        print(url)
    else:
        break
```

Image 18. The code script of web application in the web scraping stage

Image 19 shows the HTML structure of the Itviec website by using the inspecting function. Developers hover over each indicator to locate the tag of information. For example, the location information is in the <span> tag as shown in image 19 below.



1,677 IT jobs in Vietnam

Data and System Architecture Specialist
1,500 - 3,500 USD
- Working with foreign experts
- Expand the scope of work
- High chances for promotion

Data Analyst | Project Manager | Cloud
Hot
Ho Chi Minh
3m

02 PHP Dev/Team Leader (MySQL) Upto$2000
800 - 2,000 USD
PHP | MySQL | JavaScript
Hot
Ha Noi
28m

Fullstack Dev (JavaScript/Java/Python)
1,500 - 2,500 USD
JavaScript | Python | Java
Hot
Ho Chi Minh
37m

Java Engineering Manager
You'll love it
Hot
Ho Chi Minh
Ha Noi

```
        <h3 class="title">…</h3>
        <div class="svg-icon svg-icon--rounded svg-icon--green"> flex
          <svg class="svg-icon__icon">…</svg>
          <div class="svg-icon__text">1,500 - 3,500 USD</div>
        </div>
        <div class="hidden-xs d-none d-sm-block">…</div>
        <div class="visible-xs d-sm-none">…</div>
      </div>
    </div>
    <div class="job-bottom">…</div> flex
  </div>
  <div class="city_and_posted_date hidden-xs d-none d-sm-flex"> flex
    <div class="feature new text">Hot</div>
    <div class="city">
      <div class="address">
        <span class="text">Ho Chi Minh</span> == $0
      </div>
    </div>
    <div class="distance-time-job-posted">…</div>
  </div>
</div>
</div>
```

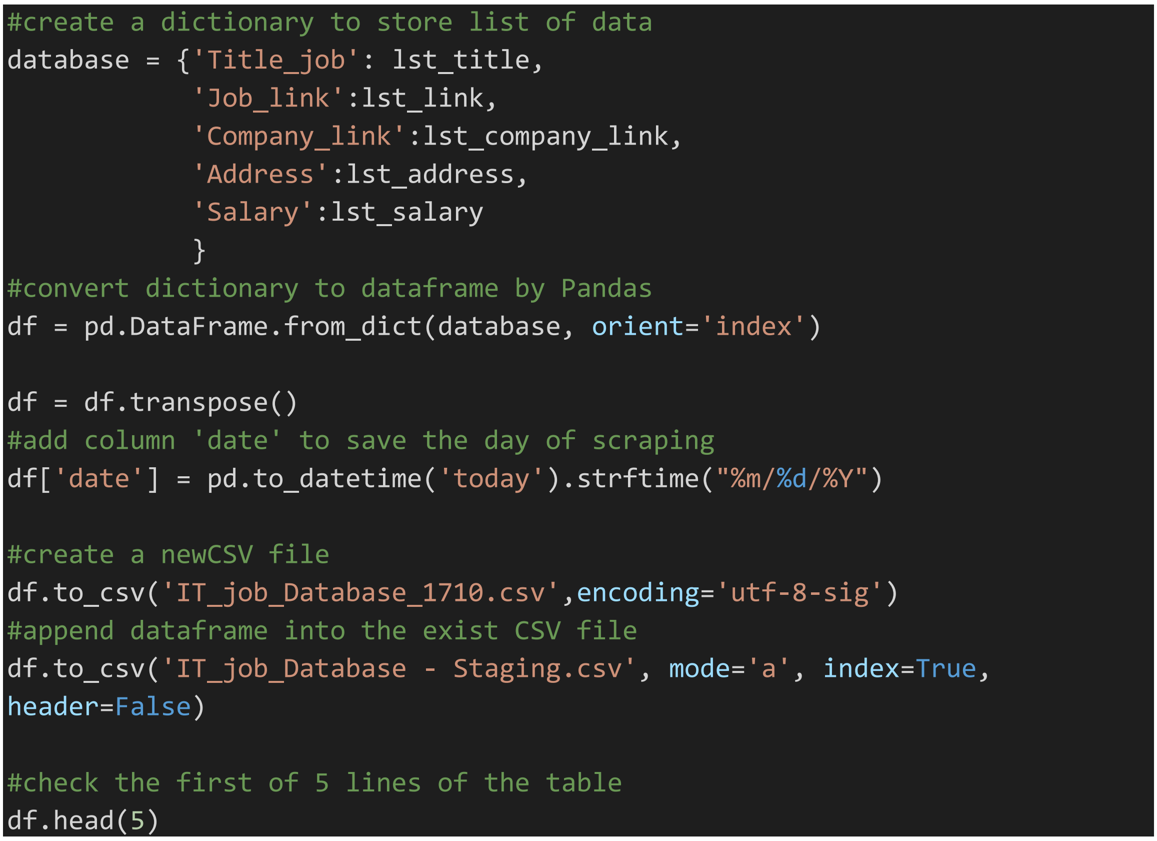
```
element.
style {
}
applica…
.job
.job_conten
t
.city_and_p
osted_date
.text {
  color:
    #a6a
  display
    :
    block;
}
applica…
*,
*::before,
*::after {
  box-
    sizing
    :
    border
```

… er-hot-job.job--selected div.job_content div.city_and_posted_date.hidden-xs.d-none.d-sm-flex div.city div.address span.text …

```
                                                   platform.js?onload=initGoogleSignin:48
                                                   platform.js?onload=initGoogleSignin:48
⚠ ▸Failed to execute 'write' on 'Document': It isn't possible to write into a document from an   conversion.js:28
  asynchronously-loaded external script unless it is explicitly opened.
2 DevTools failed to load source map: Could not load content for chrome-extension://amfojhdiedpdnlijjbhjnhokbnohfdfb/js/SV
  G.js.map: HTTP error: status code 404, net::ERR_UNKNOWN_URL_SCHEME
> |
```

Image 19. Screenshot from Itviec.com's website: the HTML website's structure by using the inspecting function.

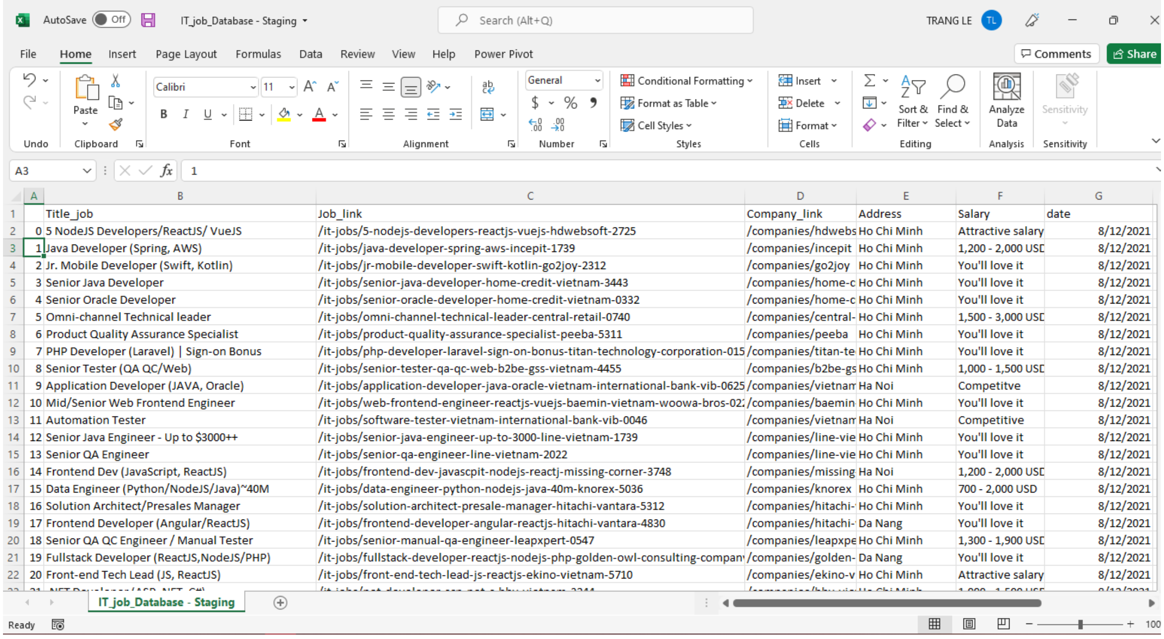
The Pandas library was used to save the obtained data to CSV files. Data was saved in the dictionary type after the scraping procedure. Pandas converted the dictionary to a data-frame, which was then stored in a tabular table. The extraction date information was stored in a new column. Finally, to_csv() was used to generate a new CSV file for each scraping process. By using mode = 'a', the new dataframe will append to an existing CSV file rather than creating a new one. The code script for storing data is shown in image 20 below.



```
#create a dictionary to store list of data
database = {'Title_job': lst_title,
            'Job_link':lst_link,
            'Company_link':lst_company_link,
            'Address':lst_address,
            'Salary':lst_salary
            }
#convert dictionary to dataframe by Pandas
df = pd.DataFrame.from_dict(database, orient='index')

df = df.transpose()
#add column 'date' to save the day of scraping
df['date'] = pd.to_datetime('today').strftime("%m/%d/%Y")

#create a newCSV file
df.to_csv('IT_job_Database_1710.csv',encoding='utf-8-sig')
#append dataframe into the exist CSV file
df.to_csv('IT_job_Database - Staging.csv', mode='a', index=True,
header=False)

#check the first of 5 lines of the table
df.head(5)
```

Image 20. The code script of web application in storing data stage

Scraping data is executed daily, resulting in a new CSV file each time. All files must be integrated and turned into a single staging file as shown in image 21 below.

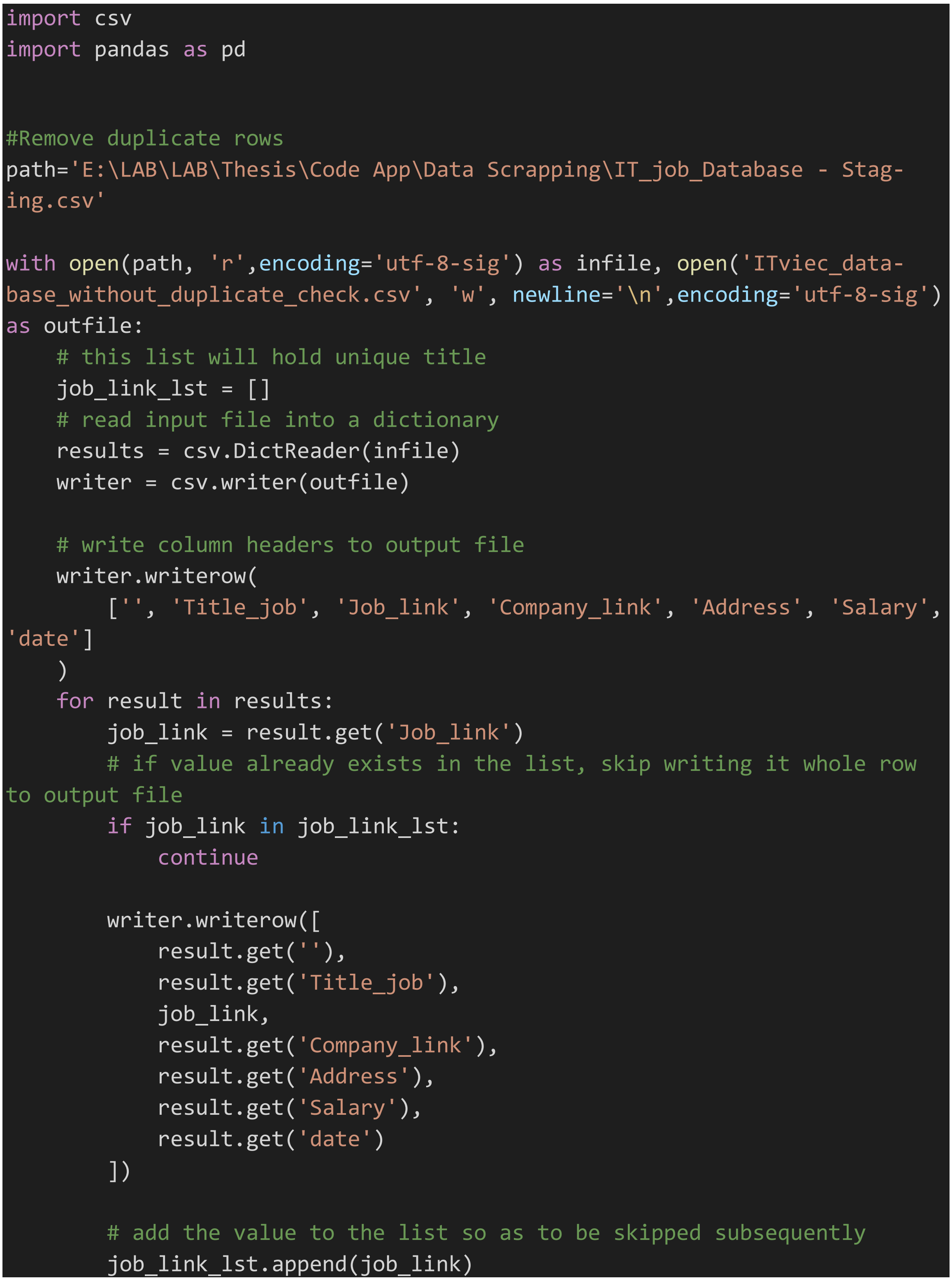


	A	B	C	D	E	F	G
1		Title_job	Job_link	Company_link	Address	Salary	date
2	0	5 NodeJS Developers/ReactJS/ VueJS	/it-jobs/5-nodejs-developers-reactjs-vuejs-hdwebsoft-2725	/companies/hdwebs	Ho Chi Minh	Attractive salary	8/12/2021
3	1	Java Developer (Spring, AWS)	/it-jobs/java-developer-spring-aws-incepit-1739	/companies/incepit	Ho Chi Minh	1,200 - 2,000 USD	8/12/2021
4	2	Jr. Mobile Developer (Swift, Kotlin)	/it-jobs/jr-mobile-developer-swift-kotlin-go2joy-2312	/companies/go2joy	Ho Chi Minh	You'll love it	8/12/2021
5	3	Senior Java Developer	/it-jobs/senior-java-developer-home-credit-vietnam-3443	/companies/home-c	Ho Chi Minh	You'll love it	8/12/2021
6	4	Senior Oracle Developer	/it-jobs/senior-oracle-developer-home-credit-vietnam-0332	/companies/home-c	Ho Chi Minh	You'll love it	8/12/2021
7	5	Omni-channel Technical leader	/it-jobs/omni-channel-technical-leader-central-retail-0740	/companies/central-	Ho Chi Minh	1,500 - 3,000 USD	8/12/2021
8	6	Product Quality Assurance Specialist	/it-jobs/product-quality-assurance-specialist-peeba-5311	/companies/peeba	Ho Chi Minh	You'll love it	8/12/2021
9	7	PHP Developer (Laravel) | Sign-on Bonus	/it-jobs/php-developer-laravel-sign-on-bonus-titan-technology-corporation-015	/companies/titan-te	Ho Chi Minh	You'll love it	8/12/2021
10	8	Senior Tester (QA QC/Web)	/it-jobs/senior-tester-qa-qc-web-b2be-gss-vietnam-4455	/companies/b2be-gs	Ho Chi Minh	1,000 - 1,500 USD	8/12/2021
11	9	Application Developer (JAVA, Oracle)	/it-jobs/application-developer-java-oracle-vietnam-international-bank-vib-0625	/companies/vietnam	Ha Noi	Competitve	8/12/2021
12	10	Mid/Senior Web Frontend Engineer	/it-jobs/web-frontend-engineer-reactjs-vuejs-baemin-vietnam-woowa-bros-02	/companies/baemin-	Ho Chi Minh	You'll love it	8/12/2021
13	11	Automation Tester	/it-jobs/software-tester-vietnam-international-bank-vib-0046	/companies/vietnam	Ha Noi	Competitive	8/12/2021
14	12	Senior Java Engineer - Up to $3000++	/it-jobs/senior-java-engineer-up-to-3000-line-vietnam-1739	/companies/line-vie	Ho Chi Minh	You'll love it	8/12/2021
15	13	Senior QA Engineer	/it-jobs/senior-qa-engineer-line-vietnam-2022	/companies/line-vie	Ho Chi Minh	You'll love it	8/12/2021
16	14	Frontend Dev (JavaScript, ReactJS)	/it-jobs/frontend-dev-javascpit-nodejs-reactj-missing-corner-3748	/companies/missing	Ha Noi	1,200 - 2,000 USD	8/12/2021
17	15	Data Engineer (Python/NodeJS/Java)~40M	/it-jobs/data-engineer-python-nodejs-java-40m-knorex-5036	/companies/knorex	Ho Chi Minh	700 - 2,000 USD	8/12/2021
18	16	Solution Architect/Presales Manager	/it-jobs/solution-architect-presale-manager-hitachi-vantara-5312	/companies/hitachi-	Ho Chi Minh	You'll love it	8/12/2021
19	17	Frontend Developer (Angular/ReactJS)	/it-jobs/frontend-developer-angular-reactjs-hitachi-vantara-4830	/companies/hitachi-	Da Nang	You'll love it	8/12/2021
20	18	Senior QA QC Engineer / Manual Tester	/it-jobs/senior-manual-qa-engineer-leapxpert-0547	/companies/leapxpe	Ho Chi Minh	1,300 - 1,900 USD	8/12/2021
21	19	Fullstack Developer (ReactJS,NodeJS/PHP)	/it-jobs/fullstack-developer-reactjs-nodejs-php-golden-owl-consulting-compan	/companies/golden-	Da Nang	You'll love it	8/12/2021
22	20	Front-end Tech Lead (JS, ReactJS)	/it-jobs/front-end-tech-lead-js-reactjs-ekino-vietnam-5710	/companies/ekino-v	Ho Chi Minh	Attractive salary	8/12/2021

Image 21. Screenshot from CSV's file: the interface of staging table after the web scraping process

5.2 Phase 2: Data preparation

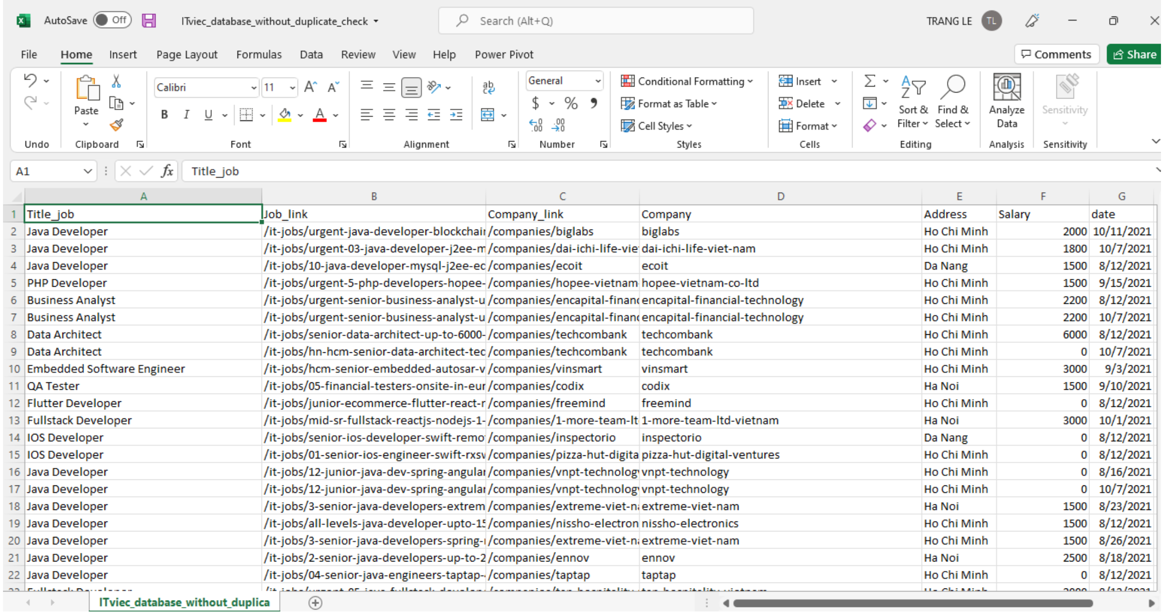
The second stage of the building process is data preparation. Pandas is the main data cleaning library. Job postings can be made at various times to ensure those job seekers are aware of them. When they run the scraping script every day, it results in duplicate records in the table. The job link column helps to distinguish the rows. If a job link in one row is identical to a job link in another row, the job link in the first row will be removed, leaving only one row. After cleaning the data, it will be saved in a new CSV file for future maintenance. Each recruitment website stores data in a different type and format. As a result, there is one extra step required to manage it manually via Excel. Storing data without duplicates in a separate file will provide coders with a backup file to compare and restore data when manual consolidation errors occur. Image 22 below shows the code scripts for eliminating duplicate rows using the Pandas library.



```
import csv
import pandas as pd


#Remove duplicate rows
path='E:\LAB\LAB\Thesis\Code App\Data Scrapping\IT_job_Database - Staging.csv'

with open(path, 'r',encoding='utf-8-sig') as infile, open('ITviec_database_without_duplicate_check.csv', 'w', newline='\n',encoding='utf-8-sig') as outfile:
    # this list will hold unique title
    job_link_lst = []
    # read input file into a dictionary
    results = csv.DictReader(infile)
    writer = csv.writer(outfile)

    # write column headers to output file
    writer.writerow(
        ['', 'Title_job', 'Job_link', 'Company_link', 'Address', 'Salary', 'date']
    )
    for result in results:
        job_link = result.get('Job_link')
        # if value already exists in the list, skip writing it whole row to output file
        if job_link in job_link_lst:
            continue

        writer.writerow([
            result.get(''),
            result.get('Title_job'),
            job_link,
            result.get('Company_link'),
            result.get('Address'),
            result.get('Salary'),
            result.get('date')
        ])

        # add the value to the list so as to be skipped subsequently
        job_link_lst.append(job_link)
```

Image 22. The code script of web application in the data cleaning stage

Furthermore, there are various issues with the table that must be addressed before it can be viewed. According to the theoretical in the sub-chapter 3.3.2, the table has frequent issues such as human errors, incomplete information, and data format. To begin with, when it comes to human errors, each recruiter will have a unique style while posting a job. It can be written in either English or Vietnamese. But it should be translated entirely into the common language that is English. The job titles are different since they can add some additional unexpected value to attract candidates, but it has no value for analysis. Thus, the unexpected words of the job title must delete. Additionally, several records lack pay information. It may contain incomplete information. In this case, it will be replaced by the value 0. The salary data is recorded as a string data type. It should be transformed into integer data types in order to they may be calculated in the chart. The candidates are also interested in company information, so it will be extracted into a new column by Excel formulas from the company link column. Following the data cleaning procedure, the result as seen in image 23 below is ready for the data visualization stage.



	A	B	C	D	E	F	G
1	Title_job	Job_link	Company_link	Company	Address	Salary	date
2	Java Developer	/it-jobs/urgent-java-developer-blockchai	/companies/biglabs	biglabs	Ho Chi Minh	2000	10/11/2021
3	Java Developer	/it-jobs/urgent-03-java-developer-j2ee-m	/companies/dai-ichi-life-vie	dai-ichi-life-viet-nam	Ho Chi Minh	1800	10/7/2021
4	Java Developer	/it-jobs/10-java-developer-mysql-j2ee-ec	/companies/ecoit	ecoit	Da Nang	1500	8/12/2021
5	PHP Developer	/it-jobs/urgent-5-php-developers-hopee-	/companies/hopee-vietnam	hopee-vietnam-co-ltd	Ho Chi Minh	1500	9/15/2021
6	Business Analyst	/it-jobs/urgent-senior-business-analyst-u	/companies/encapital-financ	encapital-financial-technology	Ho Chi Minh	2200	8/12/2021
7	Business Analyst	/it-jobs/urgent-senior-business-analyst-u	/companies/encapital-financ	encapital-financial-technology	Ho Chi Minh	2200	10/7/2021
8	Data Architect	/it-jobs/senior-data-architect-up-to-6000-	/companies/techcombank	techcombank	Ho Chi Minh	6000	8/12/2021
9	Data Architect	/it-jobs/hn-hcm-senior-data-architect-tec	/companies/techcombank	techcombank	Ho Chi Minh	0	10/7/2021
10	Embedded Software Engineer	/it-jobs/hcm-senior-embedded-autosar-v	/companies/vinsmart	vinsmart	Ho Chi Minh	3000	9/3/2021
11	QA Tester	/it-jobs/05-financial-testers-onsite-in-eur	/companies/codix	codix	Ha Noi	1500	9/10/2021
12	Flutter Developer	/it-jobs/junior-ecommerce-flutter-react-r	/companies/freemind	freemind	Ho Chi Minh	0	8/12/2021
13	Fullstack Developer	/it-jobs/mid-sr-fullstack-reactjs-nodejs-1-	/companies/1-more-team-lt	1-more-team-ltd-vietnam	Ha Noi	3000	10/1/2021
14	IOS Developer	/it-jobs/senior-ios-developer-swift-remo	/companies/inspectorio	inspectorio	Da Nang	0	8/12/2021
15	IOS Developer	/it-jobs/01-senior-ios-engineer-swift-rxsv	/companies/pizza-hut-digita	pizza-hut-digital-ventures	Ho Chi Minh	0	8/12/2021
16	Java Developer	/it-jobs/12-junior-java-dev-spring-angula	/companies/vnpt-technolog	vnpt-technology	Ho Chi Minh	0	8/16/2021
17	Java Developer	/it-jobs/12-junior-java-dev-spring-angula	/companies/vnpt-technolog	vnpt-technology	Ho Chi Minh	0	10/7/2021
18	Java Developer	/it-jobs/3-senior-java-developers-extrem	/companies/extreme-viet-n	extreme-viet-nam	Ha Noi	1500	8/23/2021
19	Java Developer	/it-jobs/all-levels-java-developer-upto-15	/companies/nissho-electron	nissho-electronics	Ho Chi Minh	1500	8/12/2021
20	Java Developer	/it-jobs/3-senior-java-developers-spring-	/companies/extreme-viet-n	extreme-viet-nam	Ho Chi Minh	1500	8/26/2021
21	Java Developer	/it-jobs/2-senior-java-developers-up-to-2	/companies/ennov	ennov	Ha Noi	2500	8/18/2021
22	Java Developer	/it-jobs/04-senior-java-engineers-taptap-	/companies/taptap	taptap	Ho Chi Minh	0	8/12/2021

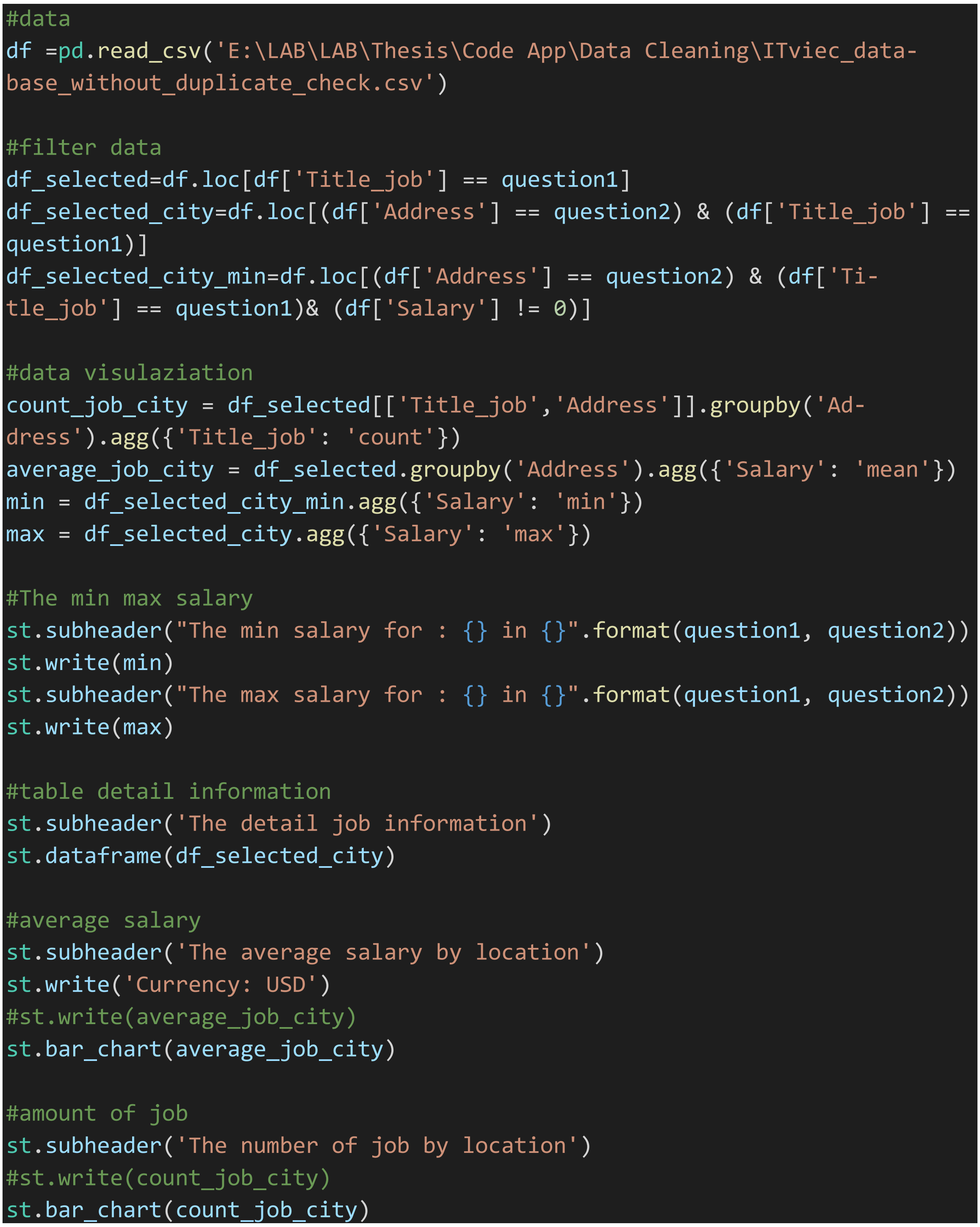
Image 23. Screenshot from CSV's file: the interface of the output after the data cleaning process

5.3 Phase 3: Data visualization

After they have a clean database, they may move on to the third step of establishing a web application process, which is data visualization. According to the wireframe, the web application will employ two types of charts: bar charts and table charts. The web application will display four charts, which are as follows:

- The min of salary of selected job in the selected location
- The max of salary of selected job in the selected location
- The average salary of the selected job by location
- The number of jobs of the selected job by location

The fundamental framework for making the chart is Streamlit. Each chart is displayed using a unique filter database. In the navigation pane, users can select the job title and location they want to search for. As a result, the first two charts are given in the table, along with the selected job and location. The rest of the chart is a bar chart with simply the job title as a filter. st.subheader() displays the hearer of each chart. The st.bar_chart() function is used to display the bar chart. st.write() and st.dataframe() are similar functions to display the table information. The code script for the data visualization procedure is shown in image 24 below.

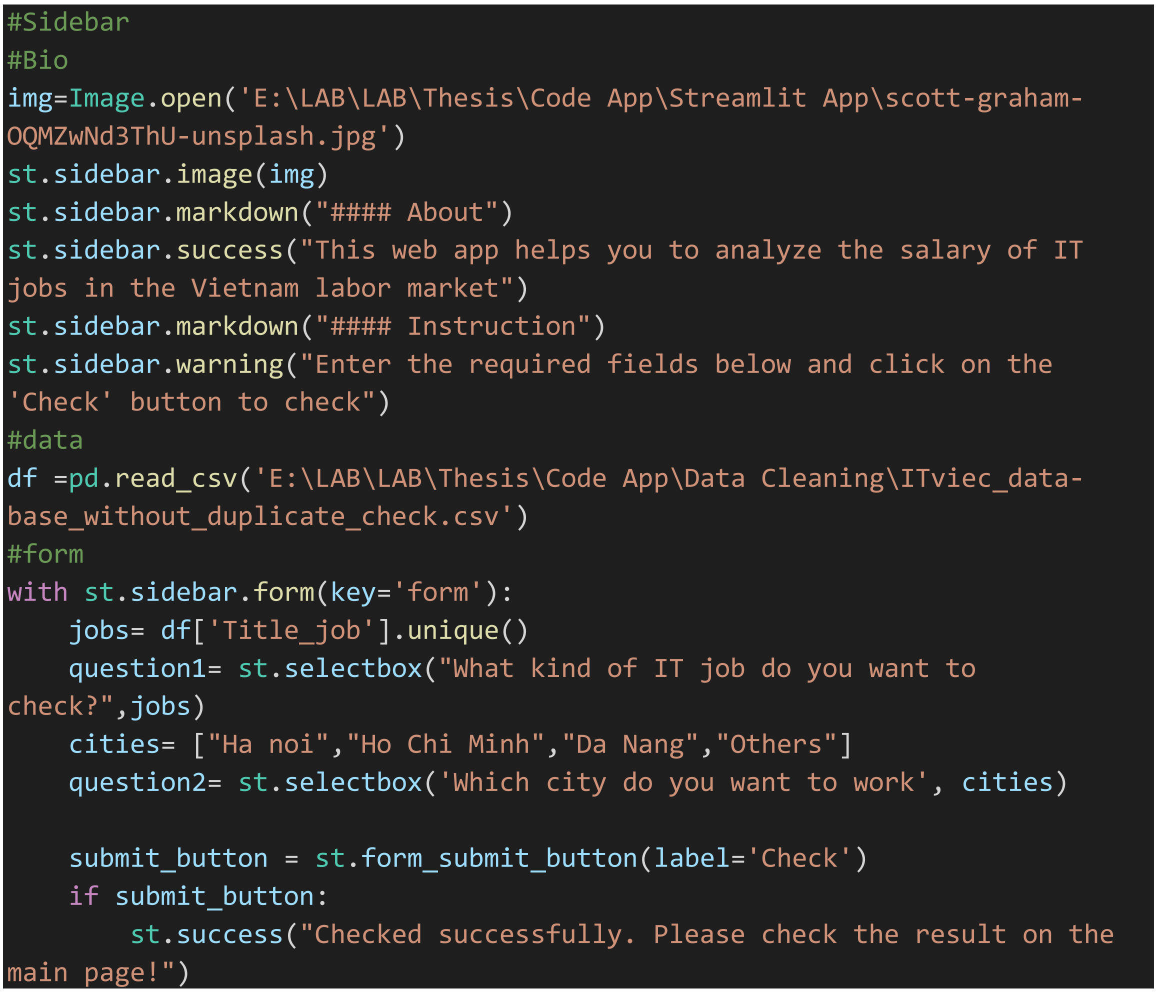


```
#data
df =pd.read_csv('E:\LAB\LAB\Thesis\Code App\Data Cleaning\ITviec_database_without_duplicate_check.csv')

#filter data
df_selected=df.loc[df['Title_job'] == question1]
df_selected_city=df.loc[(df['Address'] == question2) & (df['Title_job'] == question1)]
df_selected_city_min=df.loc[(df['Address'] == question2) & (df['Title_job'] == question1)& (df['Salary'] != 0)]

#data visulaziation
count_job_city = df_selected[['Title_job','Address']].groupby('Address').agg({'Title_job': 'count'})
average_job_city = df_selected.groupby('Address').agg({'Salary': 'mean'})
min = df_selected_city_min.agg({'Salary': 'min'})
max = df_selected_city.agg({'Salary': 'max'})

#The min max salary
st.subheader("The min salary for : {} in {}".format(question1, question2))
st.write(min)
st.subheader("The max salary for : {} in {}".format(question1, question2))
st.write(max)

#table detail information
st.subheader('The detail job information')
st.dataframe(df_selected_city)

#average salary
st.subheader('The average salary by location')
st.write('Currency: USD')
#st.write(average_job_city)
st.bar_chart(average_job_city)

#amount of job
st.subheader('The number of job by location')
#st.write(count_job_city)
st.bar_chart(count_job_city)
```

Image 24. The code script of web application in the data visualization stage

5.4 Phase 4: Data publishing

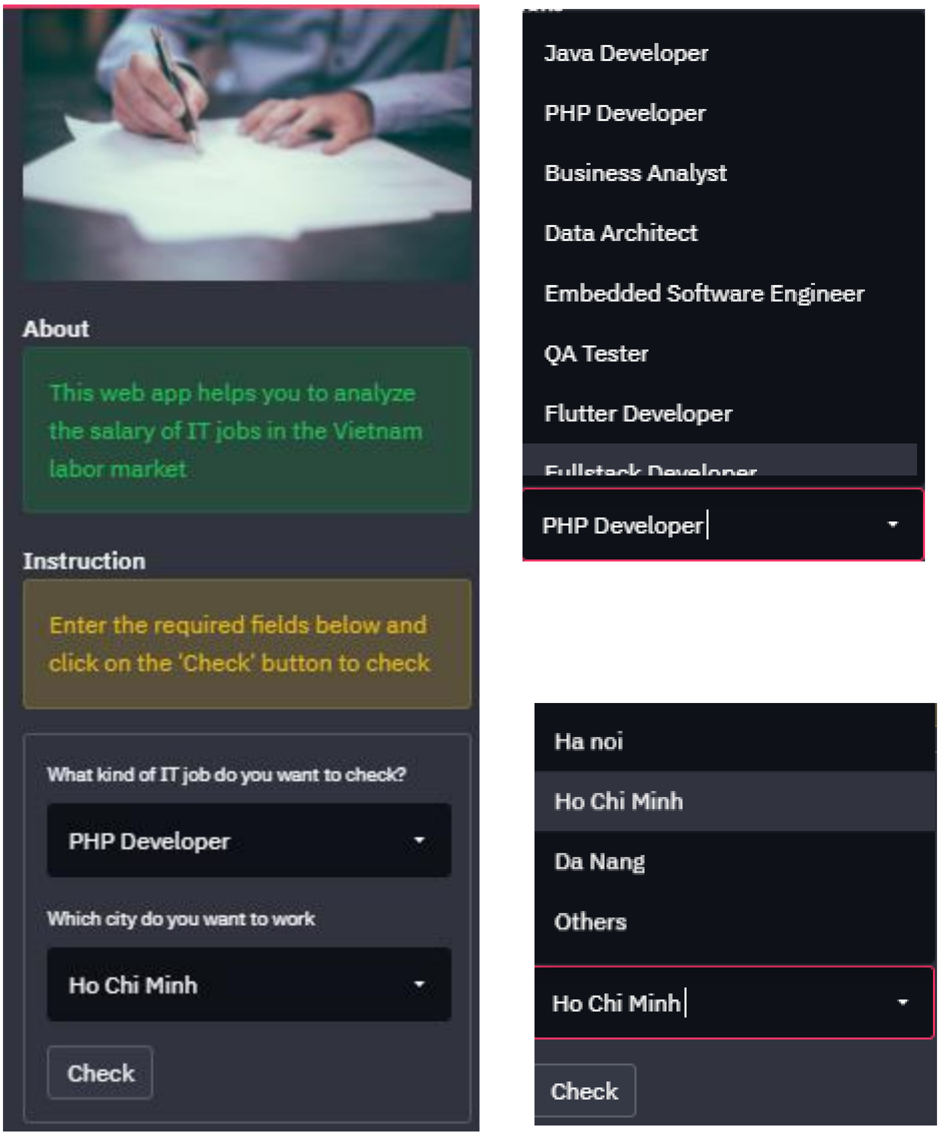
The final step in the process is data publishing. As discussed in sub-chapter 3.4, Streamlit is a framework for developing a web application for data analysis. The web application will be deployed all functions into an interactive website by Streamlit. According to the wireframe, the web application has three major components. The first component of the web application is the navigation pane. Widgets may be used to not only give interactivity to a report but also to organize it into a sidebar using the st.sidebar.element. Each element supplied to the st.sidebar is pinned to the left, allowing visitors to concentrate on the content of their application. (Streamlit 2021c.) The code script for creating the navigation pane is shown in image 25 below. The entire code script is available in Appendix 2 or the Github link at the end of this thesis.



```
#Sidebar
#Bio
img=Image.open('E:\LAB\LAB\Thesis\Code App\Streamlit App\scott-graham-
OQMZwNd3ThU-unsplash.jpg')
st.sidebar.image(img)
st.sidebar.markdown("#### About")
st.sidebar.success("This web app helps you to analyze the salary of IT
jobs in the Vietnam labor market")
st.sidebar.markdown("#### Instruction")
st.sidebar.warning("Enter the required fields below and click on the
'Check' button to check")
#data
df =pd.read_csv('E:\LAB\LAB\Thesis\Code App\Data Cleaning\ITviec_data-
base_without_duplicate_check.csv')
#form
with st.sidebar.form(key='form'):
    jobs= df['Title_job'].unique()
    question1= st.selectbox("What kind of IT job do you want to
check?",jobs)
    cities= ["Ha noi","Ho Chi Minh","Da Nang","Others"]
    question2= st.selectbox('Which city do you want to work', cities)

    submit_button = st.form_submit_button(label='Check')
    if submit_button:
        st.success("Checked successfully. Please check the result on the
main page!")
```

Image 25. The navigation pane’s code script in the data publishing stage

When running the web application's code script, the navigation pane will have the interface as shown in image 26 below. It includes a brief instruction as well as two drop-down lists for users to select the job title and location.



About

This web app helps you to analyze the salary of IT jobs in the Vietnam labor market

Instruction

Enter the required fields below and click on the 'Check' button to check

What kind of IT job do you want to check?

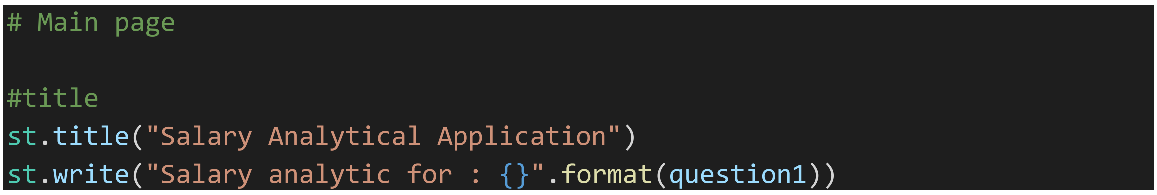
PHP Developer

Which city do you want to work

Ho Chi Minh

Image 26. Screenshot from web application: the interface of navigation pane part

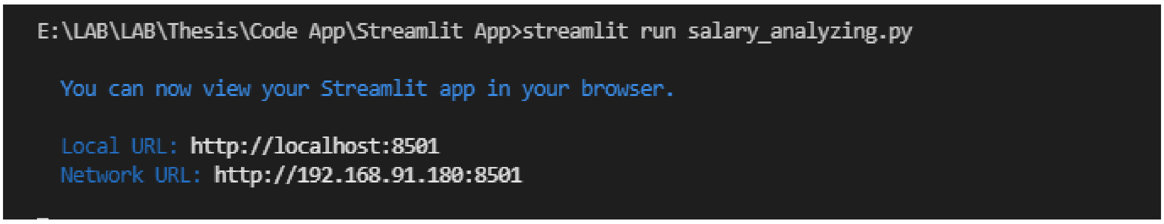
Furthermore, the web application features a layout for displaying its name and the name of its search. The text in title format is shown by st.title(). In this scenario, st.write() displays the content as a sentence line in standard text formatting. The creating header’s code script is shown in image 27 below.



```
# Main page

#title
st.title("Salary Analytical Application")
st.write("Salary analytic for : {}".format(question1))
```

Image 27. The header's code script in the data publishing stage

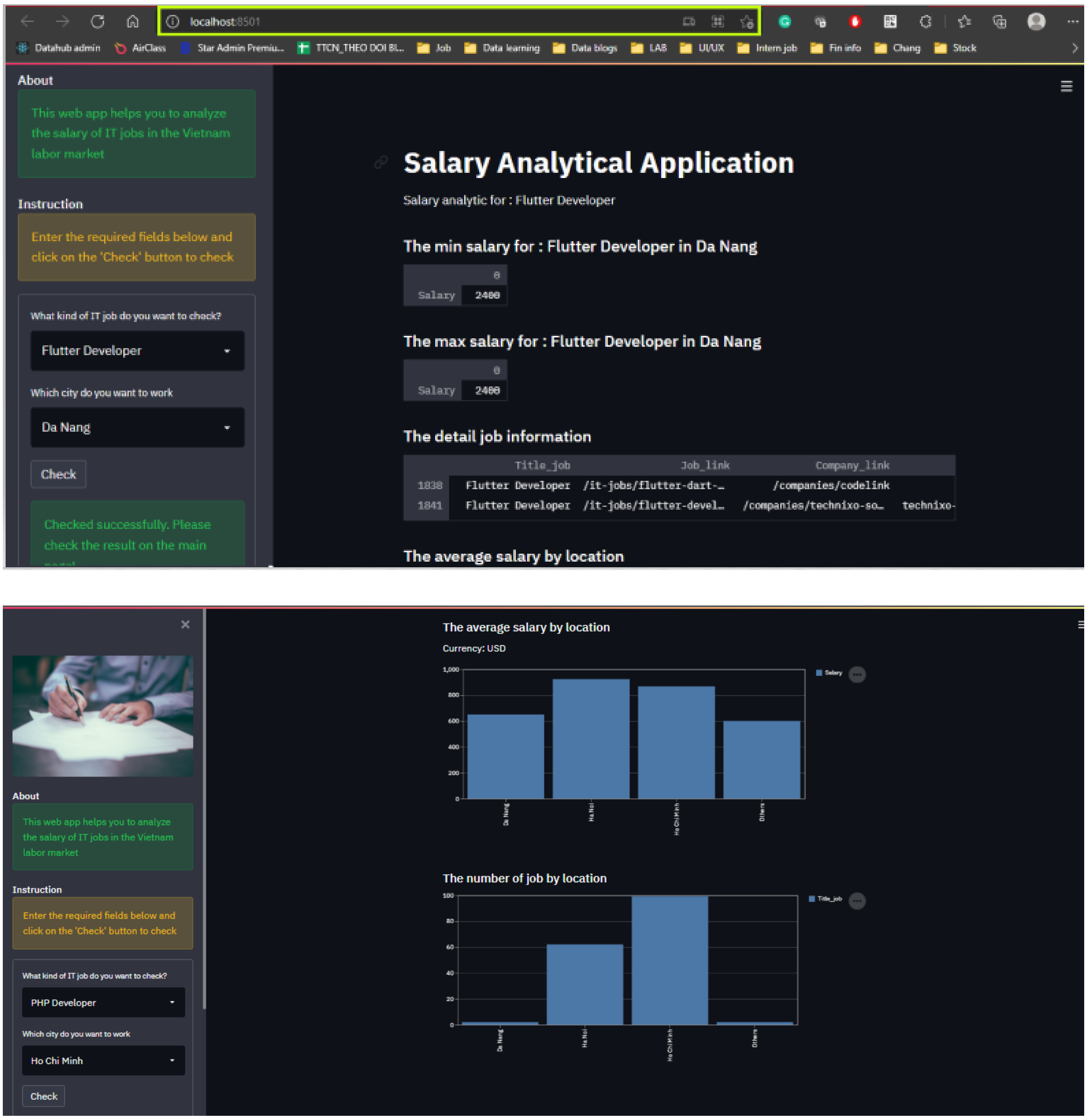
The chart area is the final web application component. It is covered in more detail in the sub-chapter on data visualization. When the code script is finished, it is time to run the script. The command line 'streamlit run path.py' will be used to run the application script through the command prompt. It will generate two distinct URLs: local URL and network URL.



```
E:\LAB\LAB\Thesis\Code App\Streamlit App>streamlit run salary_analyzing.py

  You can now view your Streamlit app in your browser.

  Local URL: http://localhost:8501
  Network URL: http://192.168.91.180:8501
```

Image 28. The command line of running the web application script

A web browser will open the web application automatically. The web application is run on the web browser in the local URL as shown in image 29 below.



About

This web app helps you to analyze the salary of IT jobs in the Vietnam labor market

Instruction

Enter the required fields below and click on the 'Check' button to check

What kind of IT job do you want to check?

Flutter Developer

Which city do you want to work

Da Nang

Check

Checked successfully. Please check the result on the main panel

Salary Analytical Application

Salary analytic for : Flutter Developer

The min salary for : Flutter Developer in Da Nang

	0
Salary	2400

The max salary for : Flutter Developer in Da Nang

	0
Salary	2400

The detail job information

	Title_job	Job_link	Company_link	
1838	Flutter Developer	/it-jobs/flutter-dart-…	/companies/codelink	
1841	Flutter Developer	/it-jobs/flutter-devel…	/companies/technixo-so…	technixo-

The average salary by location

About

This web app helps you to analyze the salary of IT jobs in the Vietnam labor market

Instruction

Enter the required fields below and click on the 'Check' button to check

What kind of IT job do you want to check?

PHP Developer

Which city do you want to work

Ho Chi Minh

Check

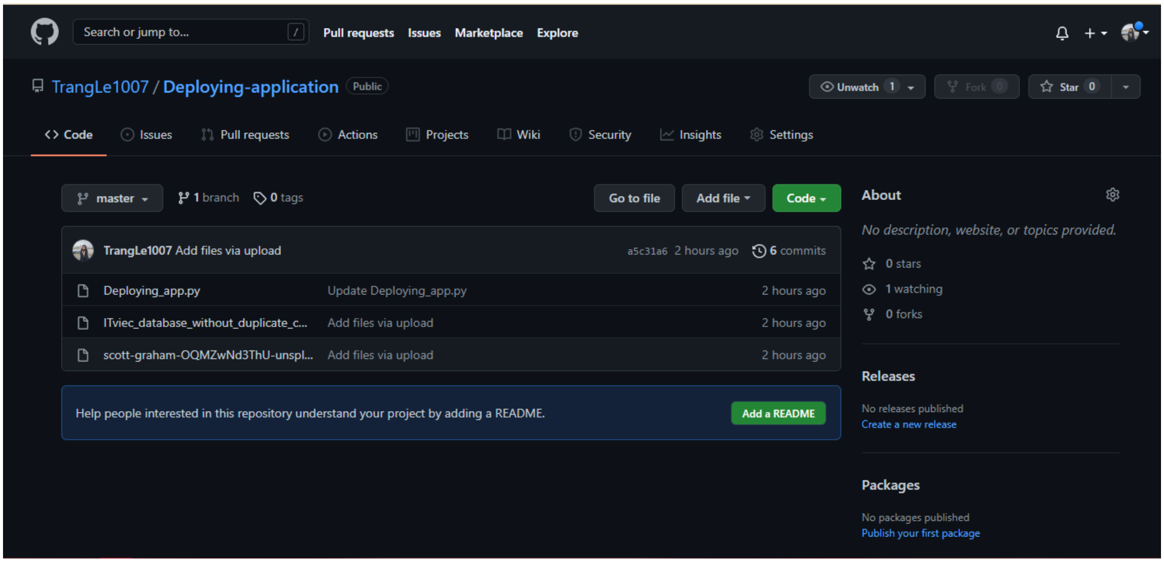
The average salary by location

Currency: USD

The number of job by location

Image 29. Screenshot from web application: the interface of salary analysis web application in local URL

Based on the theoretical instruction, a repository called deploying the application is created on Github to hold web application code scripts and related materials. The interface of the repository is shown in image 30 below.



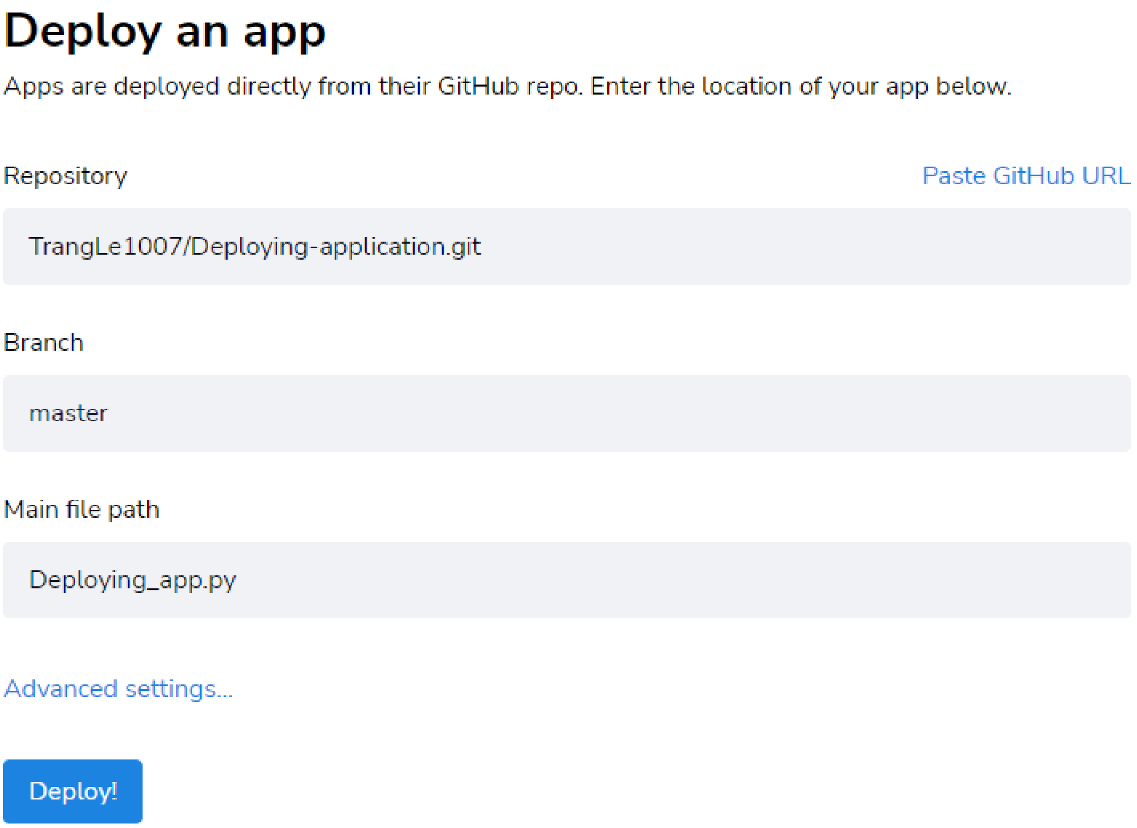
TrangLe1007 Add files via upload a5c31a6 2 hours ago 6 commits

Deploying_app.py	Update Deploying_app.py	2 hours ago
ITviec_database_without_duplicate_c...	Add files via upload	2 hours ago
scott-graham-OQMZwNd3ThU-unspl...	Add files via upload	2 hours ago

Help people interested in this repository understand your project by adding a README.

Image 30. Screenshot from Github’s website: the interface of web application GitHub repository

After that, developers run the web application script to open on the web browser. By clicking the deploy an app button on the local web application site, the data is entered into each box in the deploying window option as shown in image 31 below. When a web application is deployed to the Streamlit cloud, end users can now access and interact with the final version.



Deploy an app

Apps are deployed directly from their GitHub repo. Enter the location of your app below.

Repository: TrangLe1007/Deploying-application.git

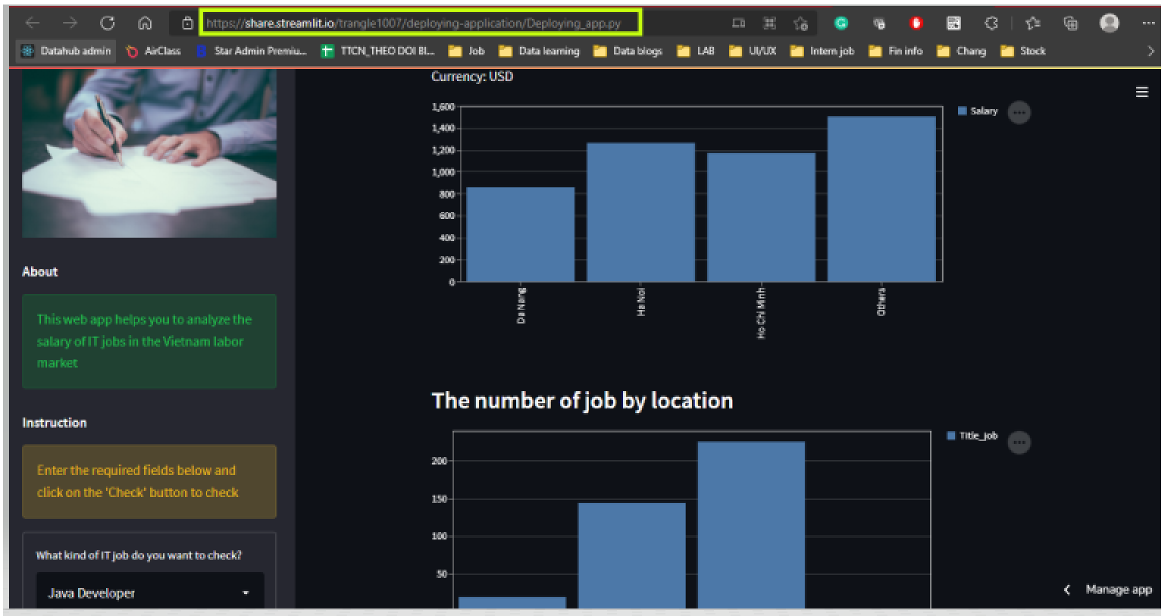
Branch: master

Main file path: Deploying_app.py

Image 31. Screenshot from web application: the interface of deploying application setting window

After hitting the deploy button, end-users can access the web application by clicking the public URL link below. The web application UI on the public URL is similar to the web application interface on the localhost as shown in image 32 below.

Public URL: https://share.streamlit.io/trangle1007/deploying-application/Deploying_app.py



About

This web app helps you to analyze the salary of IT jobs in the Vietnam labor market

Instruction

Enter the required fields below and click on the 'Check' button to check

What kind of IT job do you want to check?

Java Developer

The number of job by location

Image 32. Screenshot from web application: the interface of salary analysis web application in public URL

6 Conclusion

6.1 Validity and limitations

The current chapter evaluates the credibility of the research using validity and reliability. The current research was exploratory and descriptive. As a result, a few unfavorable characteristics have an impact on the research's validity and reliability, as well as the feasibility of putting the findings into reality. In addition, interpreting qualitative data is vulnerable to the author's bias. Furthermore, the sample does not appropriately represent the target demographic. (Dudovskiy 2021.)

Due to the study's scope, the research was conducted with a small community's limited response. The responses collected in the demand gathering are consistent, and many of them agreed with the establishing application. However, the small number of respondents, which prevented further implementation of the development plan and model solution. As a result, the study's validity is questionable.

Because Streamlit's history is short, the web application's stability throughout the operating time requires more time to verify. As a result, the thesis's validity is fairly low in the actual world when applied to a broader area of study, although it has potential for future research and advancement. Qualitative research was conducted utilizing a combination of scholarly publications, books, and journals, as well as internet documentation and blog postings which provided a diverse source of information and in-depth analysis of the research topic. These are the results of academic research and should be regarded as reputable references. It leads to a rise in the reliability of the thesis.

6.2 Answers to the research questions

As stated in the introduction of this thesis, this study needs to answer the two primary questions. The following paragraph solves the first main question.

What indicators are job seekers interested in?

There is a wealth of information given on each employment website, but job seekers are only interested in a subset of it. The result of the survey and interviews explained the customer's interest in the salary analysis web application. The end users are people who have less than 2 years of experience. The main indicators must be included in the web application are the job title, salary, company, and location. It should contain additional information regarding the company review and job description because job seekers look for more detailed information. To assist them in gaining additional valuable insights, the salary chart should be visualized in monthly and yearly time values. There is also a list of major recruitment websites that have been relocated as an accurate source such as LinkedIn, Vietnamworks, Itviec, and TopCV. The answer to the second research question is provided below.

What is the most effective way to deploy a salary analysis application to maximize its benefits?

This study was able to address this subject through both the theoretical and practical aspects of related techniques. This thesis discussed a variety of advantages Python libraries and the Streamlit framework. It also introduced the case study project to demonstrate the possibilities of these. In terms of task requirements, Python and Streamlit are the ideal matches for developing a web application in the data analysis industry. It gives developers a plethora of options for lowering development time through the variety and adaptability of libraries. Furthermore, Streamlit provides a rapid approach to setting up a website without learning HTML, CSS, or other web programming languages. When compared to other related programming languages and frameworks, Python and Streamlit both offer advantages in terms of simplicity, popularity, and community support.

6.3 Suggestion for further research

The suggestion for further research provides to enhance more functions that serve the needs of the business in the future. It contains the following list:

- Researching about the suitable database for storing the data that scraping daily (NoSQL or Relation Database or Big Data)
- Deploying the automation ETL pipeline from running script for scraping and cleaning to storing data into database
- Conducting research in huge communities with various objects to identify additional potential capabilities in the web application
- Planing how to introduce the web application into the market and make profits.

References

Andrade, F. 2020. Web Scraping with Beautiful Soup, Selenium, or Scrapy?. Retrieved on 11 October 2021. Available at: https://towardsdatascience.com/web-scraping-with-beautiful-soup-selenium-or-scrapy-62c6f3545de7

Beautiful Soup documentation. 2021. Retrieved on 11 October 2021. Available at: https://beautiful-soup-4.readthedocs.io/en/latest/

Bhandari, P. 2020. An introduction to qualitative research. Retrieved on 26 September 2021. Available at: https://www.scribbr.com/methodology/qualitative-research/

Cass, S. 2021. Top Programming Languages 2021 Python dominates as the platform for new technologies. Retrieved on 2 October 2021. Available at: https://spectrum.ieee.org/top-programming-languages-2021

Cui, Y. 2020. Deploy a Public Streamlit Web App for Free — Here’s How: Google Sheets as its backend and hosted by Streamlit Sharing. Retrieved on 1 November 2021. Available at: https://towardsdatascience.com/deploy-a-public-streamlit-web-app-for-free-heres-how-bf56d46b2abe

Datacamp. 2021. Choosing Python or R for Data Analysis? An Infographic. Retrieved on 11 Dec 2021. Available at: https://s3.amazonaws.com/assets.datacamp.com/email/other/Python+vs+R.pdf

Donnabhain, C. 2019. How Python is used in Data Science. Retrieved on 2 October 2021. Available at: https://irishtechnews.ie/how-python-is-used-in-data-science/?__cf_chl_jschl_tk__=2a69af5765dbcb4076162c9604a19d2886051136-1580376157-0-%20GlIuzx.

Dudovskiy, J. 2021. Exploratory research. Business Research Methodology. Retrieved on 20 November 2021. Available at: https://research-methodology.net/research-methodology/research-design/exploratory-research/

GeeksforGeeks. 2021. Python Language advantages and applications. Retrieved on 2 October 2021. Available at: https://www.geeksforgeeks.org/python-language-advantages-applications/

Glassdoor. 2019. The Secrets Recruiters Won’t Tell You (But Really Want To). Retrieved on 26 September 2021. Available at: https://www.glassdoor.com/blog/8-secrets-recruiters-wont-tell-you/

IBM Team. 2021. Python vs. R: What's the Difference? Retrieved on 11 Dec 2021. Available at: https://www.ibm.com/cloud/blog/python-vs-r

Indeed. 2021. Salary Range: Definition and How It's Used by Employers. Retrieved on 11 September 2021. Available at: https://www.indeed.com/career-advice/pay-salary/range-salary

Interaction Design. 2021. Retrieved on 24 October 2021. Available at: https://www.interaction-design.org/literature/topics/wireframing

Karakan, B. 2020. Python and R for Data Sciences. Retrieved on 11 Dec 2021. Available at: https://towardsdatascience.com/python-vs-r-for-data-science-6a83e4541000

Kopf, B. 2021. The Power of Figma as a Design Tool. Retrieved on 24 October 2-21. Available at: https://www.toptal.com/designers/ui/figma-design-tool

Kuhlman, D. 2012. A Python Book: Beginning Python, Advanced Python, and Python exercises. United States: Platypus Global Media

Kumar, N. 2021. Implementing Web Scraping in Python with BeautifulSoup. Retrieved on 11 October 2021. Available at: https://www.geeksforgeeks.org/implementing-web-scraping-python-beautiful-soup/

Kung, S. 2020. Python vs R: The Basics. Retrieved on 2 October 2021. Available at: https://towardsdatascience.com/python-vs-r-the-basics-d754c45c1596

Locke, E. A. 2007. The Case for Inductive Theory Building. Journal of Management vol. 33. New York: Sage Publications, pp. 880-885.

Molin, S. 2019. Hans-On Data Analysis with Pandas: Efficiently perform data collection, wrangling, analysis, and visualization using Python. Second Edition. Birmingham: Packt Publishing Ltd

Nola, R. & Sankey, H. 2007. Theories of Scientific Method. Stocksfield: Acumen Publishing Limited.

Oracle 2021. What is a database?. Retrieved on 9 November 2021. Available at: https://www.oracle.com/database/what-is-database/

Pandas Documentation. 2021a. How do I read and write tabular data? Retrieved on 16 October 2021. Available at: https://pandas.pydata.org/pandas-docs/stable/getting_started/intro_tutorials/02_read_write.html#min-tut-02-read-write

Pandas Documentation. 2021b. What kind of data do pandas handle? Retrieved on 16 October 2021. Available at: https://pandas.pydata.org/pandas-docs/stable/getting_started/intro_tutorials/01_table_oriented.html#min-tut-01-tableoriented

PitchBook. 2021. Streamlit. Retrieved on 11 October 2021. Available at: https://pitchbook.com/profiles/company/327074-23#overview

Python. 2021. Applications for Python. Retrieved on 2 October 2021. Available at: https://www.python.org/about/apps/

Rosenzweig, E. 2015. Successful User Experience: Strategies and Roadmaps. San Francisco: Elsevier Science & Technology

Schmitt, M. 2020. Streamlit vs. Dash vs. Shiny vs. Voila vs. Flask vs. Jupyter: Comparing data dashboarding tools and frameworks. Retrieved on 1 November 2021. Available at: https://towardsdatascience.com/streamlit-vs-dash-vs-shiny-vs-voila-vs-flask-vs-jupyter-24739ab5d569

Simpson-Wolf, A. 2021. Customer Needs Identification. Retrieved on 16 November 2021. Available at: https://sites.tufts.edu/eeseniordesignhandbook/2013/customer-needs-identification-2/

Streamlit. 2021a. Welcome to Streamlit. Retrieved on 11 October 2021. Available at: https://docs.streamlit.io/en/stable/#welcome-to-streamlit

Streamlit. 2021b. Deploy an app. Retrieved on 11 Dec 2021. Available at: https://docs.streamlit.io/streamlit-cloud/get-started/deploy-an-app

Streamlit. 2021c. st.sidebar. Retrieved on 20 November 2021. Available at: https://docs.streamlit.io/library/api-reference/layout/st.sidebar

Tatman, R. 2021. Data Cleaning. Retrieved on 17 October 2021. Available at: https://www.kaggle.com/learn/data-cleaning

Tech terms. 2021. Table. Retrieved on 9 November 2021. Available at: https://techterms.com/definition/table

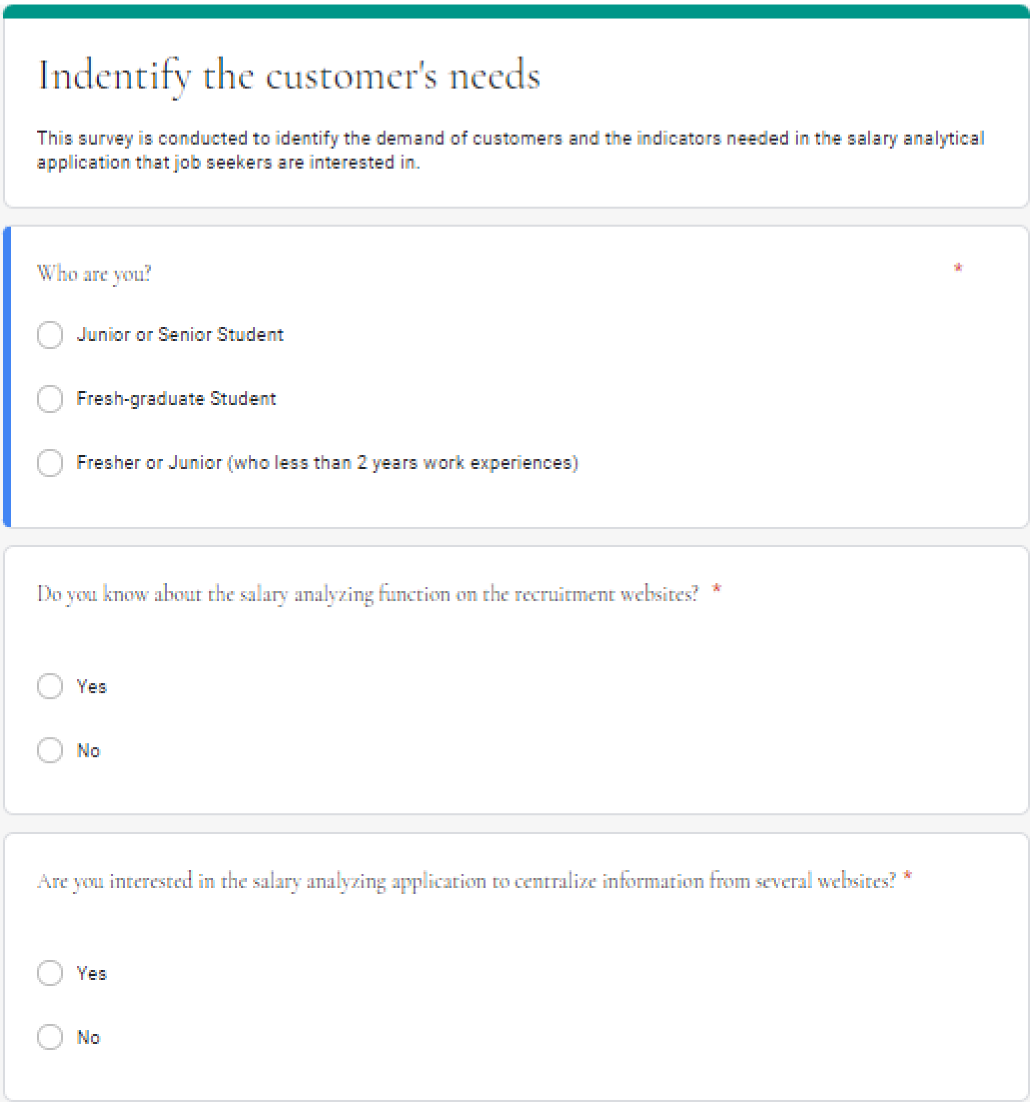
Ulrich, K & Eppinger, S. 2012. Product design and development. New York: McGraw-Hill

Vietnam Briefing. 2019. Payroll in Vietnam: A Guide to Compensation, Bonuses, and Benefits. Retrieved on 26 September 2021. Available at: https://www.vietnam-briefing.com/news/managing-payroll-vietnam.html/

Wieringa, J. 2012. Intro to Beautiful Soup. Retrieved on 11 October 2021. Available at: https://programminghistorian.org/en/lessons/retired/intro-to-beautiful-soup

Wilson, J. 2014. Essentials of business research: a guide to doing your research project: A guide to Doing Your Research Project. Second Edition. London: Sage Publication Ltd, pp. 176-177

Appendix 1. Questionnaire of identifying the customer needs



Indentify the customer's needs

This survey is conducted to identify the demand of customers and the indicators needed in the salary analytical application that job seekers are interested in.

Who are you? *

○ Junior or Senior Student

○ Fresh-graduate Student

○ Fresher or Junior (who less than 2 years work experiences)

Do you know about the salary analyzing function on the recruitment websites? *

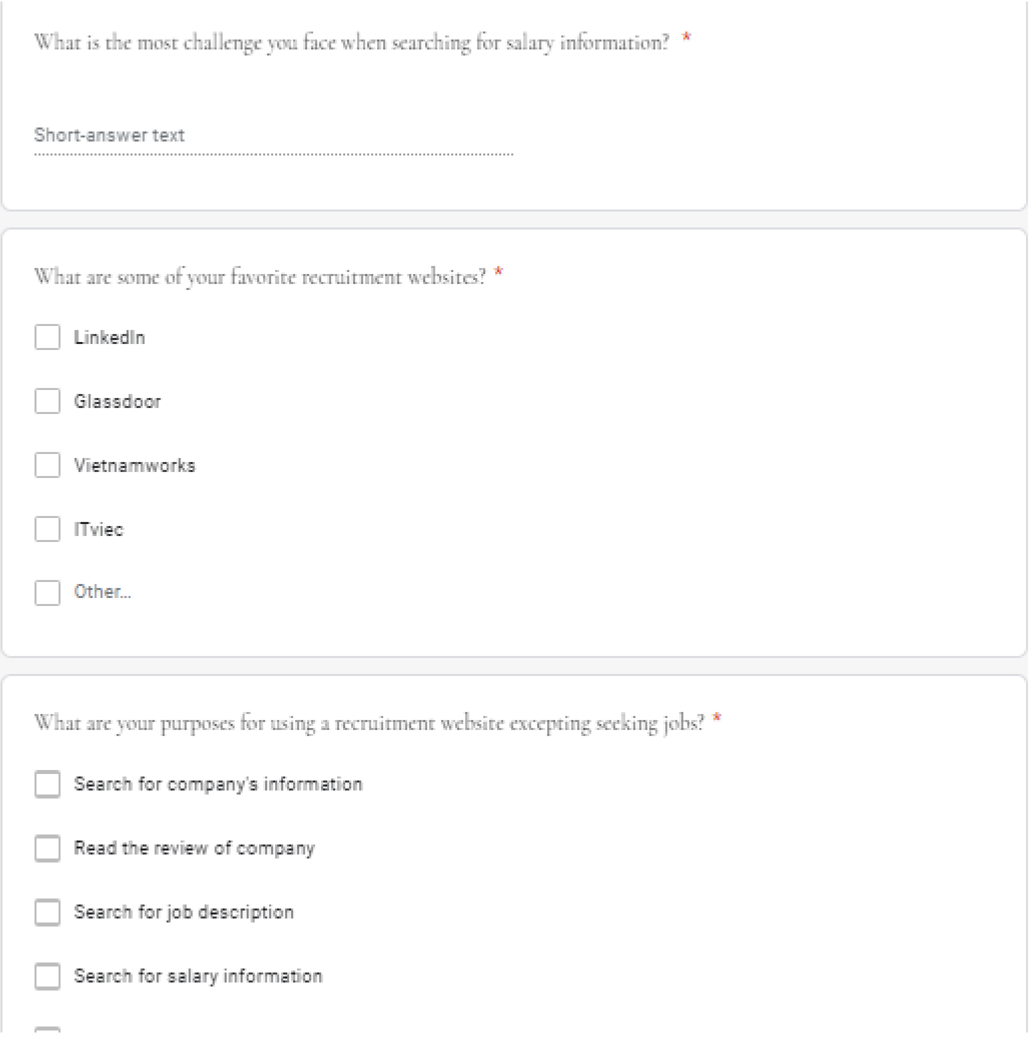
○ Yes

○ No

Are you interested in the salary analyzing application to centralize information from several websites? *

○ Yes

○ No



What is the most challenge you face when searching for salary information? *

Short-answer text

What are some of your favorite recruitment websites? *

□ LinkedIn

□ Glassdoor

□ Vietnamworks

□ ITviec

□ Other...

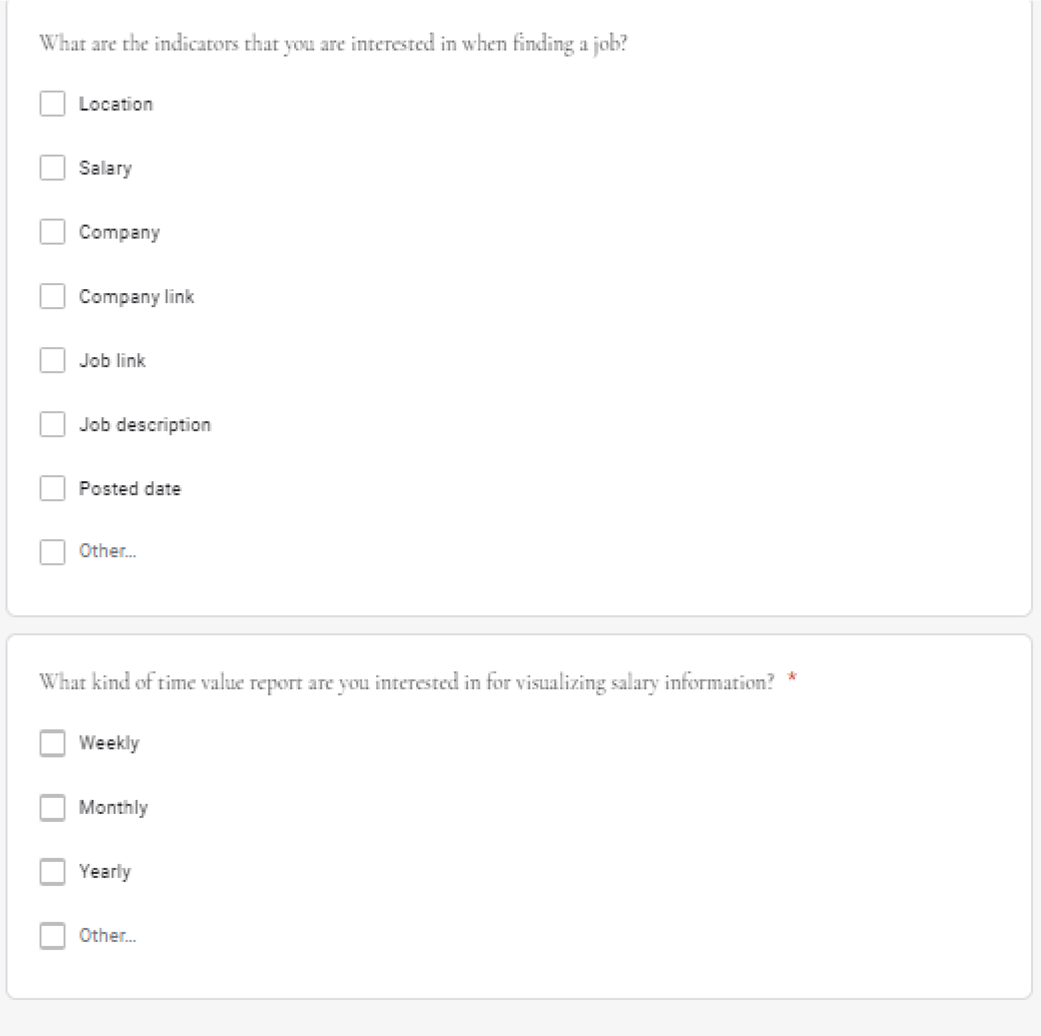
What are your purposes for using a recruitment website excepting seeking jobs? *

□ Search for company's information

□ Read the review of company

□ Search for job description

□ Search for salary information



What are the indicators that you are interested in when finding a job?

□ Location

□ Salary

□ Company

□ Company link

□ Job link

□ Job description

□ Posted date

□ Other...

What kind of time value report are you interested in for visualizing salary information? *

□ Weekly

□ Monthly

□ Yearly

□ Other...

Appendix 2. The code script of the web application

GitHub repository: https://github.com/TrangLe1007/Salary-Analytical-Application

```
import streamlit as st
from PIL import Image
import pandas as pd


#Sidebar
#Bio
img=Image.open('E:\LAB\LAB\Thesis\Code App\Streamlit App\scott-graham-OQMZwNd3ThU-unsplash.jpg')
st.sidebar.image(img)
st.sidebar.markdown("#### About")
st.sidebar.success("This web app helps you to analyze the salary of IT jobs in the Vietnam labor market")
st.sidebar.markdown("#### Instruction")
st.sidebar.warning("Enter the required fields below and click on the 'Check' button to check")
#data
df =pd.read_csv('E:\LAB\LAB\Thesis\Code App\Data Cleaning\ITviec_database_without_duplicate_check.csv')
#form
with st.sidebar.form(key='form'):
    jobs= df['Title_job'].unique()
    question1= st.selectbox("What kind of IT job do you want to check?",jobs)
    cities= ["Ha noi","Ho Chi Minh","Da Nang","Others"]
    question2= st.selectbox('Which city do you want to work', cities)

    submit_button = st.form_submit_button(label='Check')
    if submit_button:
        st.success("Checked successfully. Please check the result on the main page!")

# Main page

#title
st.title("Salary Analytical Application")
st.write("Salary analytic for : {}".format(question1))

#filter data
df_selected=df.loc[df['Title_job'] == question1]
df_selected_city=df.loc[(df['Address'] == question2) & (df['Title_job'] == question1)]
df_selected_city_min=df.loc[(df['Address'] == question2) & (df['Title_job'] == question1)& (df['Salary'] != 0)]
```

```
#data visualization


count_job_city = df_selected[['Title_job','Address']].groupby('Address').agg({'Title_job': 'count'})
average_job_city = df_selected.groupby('Address').agg({'Salary': 'mean'})
min = df_selected_city_min.agg({'Salary': 'min'})
max = df_selected_city.agg({'Salary': 'max'})

#The min max salary
st.subheader("The min salary for : {} in {}".format(question1, question2))
st.write(min)
st.subheader("The max salary for : {} in {}".format(question1, question2))
st.write(max)

#table detail information
st.subheader('The detail job information')
st.dataframe(df_selected_city)

#average salary
st.subheader('The average salary by location')
st.write('Currency: USD')
#st.write(average_job_city)
st.bar_chart(average_job_city)

#amount of job
st.subheader('The number of job by location')
#st.write(count_job_city)
st.bar_chart(count_job_city)
```