

DEPARTMENT OF COMPUTER AND INFORMATION SCIENCE

CYS401: FUNDAMENTALS OF CYBER SECURITY

CYS401 - Website Scraper Project

Instructor's Name: Dr. Rabia Latif

Section 1249 - 1242

Prepared by:

Name: Haifa Zein Eddin **ID:** 220410581

Name: Maria Alrebdi **ID:** 220511210

Name: Ghadah Saad ALBugami **ID:** 217410489

Table of Contents

1.0 Introduction	3
1.1 Security Concerns	3
1.2 Assess Common Security Risks, Threats, Attacks, and Vulnerabilities in an organization	4
2.0 Literature Review	6
2.1 Existing Methods of Implementing the Solution	6
2.2 Substitute Methods of Implementing the Solution	7
2.3 Security Incidents	7
3.0 Implementation and Requirements	8
3.1 Code & Run Overview	9
4.0 Application and Recommendations to Overcome Threats	14
5.0 Conclusion	15
6.0 Appendix	16
6.2 Code	17
7.0 References	20

1.0 Introduction

Website scraping refers to the process of automatically collecting wide ranging amounts of online data from related websites in a short amount of time using scraping software. The ability to scrape or collect large amounts of data is done by two continuous fragments, the crawler and the scraper. The crawler searches across the internet for the specific wanted data and collects the URLs of them in a queue whereas the scraper enters the URLs located by the crawler and scrape or extract the wanted data. The two components are important in extracting the needed data across the internet's web pages. Say you are working on your senior research paper, however you need to collect specific data information of that topic, by using web scraping you will be able to extract substantial amounts of data in a minuscule amount of time while obtaining all the needed information for your said senior research paper.

In this documentation, we will further elaborate on the relation security concerns, assess the common risks, attacks, vulnerabilities as well as the implementation of the documented project and an incorporated substitute method of the implemented solution.

1.1 Security Concerns

Addressing the potential cyber attacks that may occur due to attackers encountering unprotected valuable data assets to an organization is a principal concern in cybersecurity in today's time of widespread internet usage; therefore, it is essential to implement high level security control and authentication measures.

The main security concerns that accompany web scraping are as follows:

1. Data Breaches

Although the scraper is sent to extract desired data, an organization may have unprotected data that has been unintentionally visible to the public as the scraper simultaneously extracts the data that may include highly sensitive information such as personal information, ID, credit card information, etc. Therefore, an attacker may target them which leads to compromised restricted data and exposing the organizational website to legal risks.

2. Malicious Data Injection

Since web scraping is widely used by companies, attackers may inject malicious data onto websites such as worms and viruses which may harm the integrity of the data once inspected by the company.

3. Denial of Service Attack

The implementation of scrapers on a website may cause harm to the performance of the website as it is being targeted by many scrapers which affects real time users of the service of the website.

4. Intellectual Property Infringement

One of the main security concerns is the property infringement on websites such as copyrights, ownership of texts, images, sounds, etc. that may be included in a website that lacks the need of authentication which ultimately leads to legal risks.

Such concerns can be easily mitigated and handled when a high level security control measures are implemented onto the website as well as ensuring law regulations within the implementation.

1.2 Assess Common Security Risks, Threats, Attacks, and Vulnerabilities in an organization.

A scraped website cannot be inevitable to common risks, threats, attacks, and vulnerabilities that may follow up with the process.

Assessing those points may be elaborated as followed:

1. Data Privacy Violations

Risk: Loss of the important information. Scraping confidential information and information disclosure without the consent of the client.

Threats: Potentially access the data directly or inject malicious software to the data.

Attacks: Tampering through the confidentiality of the database and archives of the infrastructure.

Vulnerability: Loss of trustworthiness of the infrastructure due to the inability to implement proper safeguards to protect the clients.
Common victims of data privacy violations are the users of the website being used.

2. Social Engineering Attacks

Risk: Ability to spoof as an individual in the organization which can be extracted and figured out by the attacker.

Threats: Possible harm to the database of the infrastructure if acquired by the attacker as an impersonator to harvest, manipulate, alter information credentials, misinterpret activities.

Attacks: Ability to collect valuable confidential information and encrypted data and elevate privileges.

Vulnerability: Exploit the relationship between both parties of the infrastructure which causes lack of cybersecurity systems.

Common victims of social engineering attacks are local businesses.

3. Price Scraping Attack

Risk: Prices of an organization can be easily scraped by an attacker which results in property theft, trademark infringement, scraped copyrights.

Threats: Attackers can use the information for their own benefit by comparing their own product and remain competitive which targets data integrity

Attacks: Exploitation of unauthorized data to unauthorized users to improve and recondition their websites to gain maximal revenue

Vulnerability: Server crashes which leads to a poor performance usage and degrades usability to users.

Common victims of price scraping attacks are travel agencies.

4. Content Scraping Attack

Risk: Exploitation of complete content of data which leads to harming the organization of the website by exposing unsolicited data

Threats: Potentially scraping personal information, customers data, user profile

Attacks: Broadcasting the database content in a structured format of sensitive data which can be republished or rephrased by the attackers.

Vulnerability: Spamming bots into the website which defines the lack of security and breaches of sensitive data.

Common victims of content scraping are large corporations.

Website scraping can be easily manipulated if the proper mitigation security protocols were not implemented which causes great harm to the organization's assets as well as the users therefore the trusting relationship between those two roles may fluctuate due to these attacks.

2.0 Literature Review

2.1 Existing Methods of Implementing the Solution

To achieve a secure website that reduced the capabilities of website scraping is the ultimate deal for organizations and infrastructures as it enhances the security of the website and the unexpected approach of potential threats and attacks which sources the harm of the website therefore, implementing existing methods and reusing them is a beneficial alternative to implementing a secure web scraping solution.

The implementation of an existing/previous method of a secure web scraping solution include:

- Accessing the needed data from the official website' API if provided.
- Restrict the site terms and conditions to prevent malicious web scraping.
- Implement a rate limiting algorithm that limits the frequency number of user requests.
For instance, a human scrolling through a website tends to have a familiar structure unlike bots therefore implementing a rate limit can reduce the number of IP address requesting within a timeframe
- Apply CAPTCHAS, as they allow legitimate users to complete a given task that is considered easily for humans as opposed to computerized machines.
- Monitor and detect the spiral growth of the website's volume or overload which indicates the movements of non-human activities being done to the website.
- Implementation of bot prevention softwares (such as Arkose Labs, Cheq) which ultimately has the analytical capabilities to reduce the access of bots to the website and prevents the occurrence of bot traffic.
- A common method used by site owners is "Robots.txt" files, which are guidelines specified by the website itself in which it permits distinct parts of the website and

disallows other parts. This method concedes to a secure level of security measures as it does not allow trespassing as the scrapers are enforcing themselves to extract.

- Implementing a tool that distinguishes between legitimate users and bots to reduce web scraping.

2.2 Substitute Methods of Implementing the Solution

- Implement strong Firewall softwares onto the website that tracks and detects the movements of unauthorized bots or users or perhaps overload traffic that may come across the network.
- Upload sensitive and confidential data locally on the client's side instead of the server's side, this method will ultimately reduce the potential risk of extracting the data.
- Implement the usage of proxy which can assist in spinning the IP addresses by re-routing the scraper onto another IP address which helps in reducing the risks of data breaches and improve the efficiency.
- Modify the HTML periodically as the attackers usually access the most requested pages that are consistently available and written in clear sight.

2.3 Security Incidents

According to (Fibbe) 2004, A company titled as Bidder's Edge, Inc. has been disputing web scraping bots onto a well profited website titled as Ebay. Bidder's Edge initially has desired to come to terms with Ebay's licensing terms, however Ebay did not comply with BE's terms and declined the offer which resulted in the trespassing of their website as a cause of non agreement discourse between two companies. BE has resorted to list Ebay's auction items that are within its website as its own by implementing a web scraper on Ebay's items, which caused huge overload traffic onto the auction items and the performance within the usage as well an escalation of the bandwidth usage. Despite Ebay's technical abilities to reduce the extraction and high volumes that are affecting their website, done by BE's unauthorized access, was merely inevitable and unsuccessful. "BE's software robots accounted for approximately 1.5 percent of the traffic on

Ebay's website" (George F. Fibbe, 2004). Ebay had decided to take legal actions against BE's disputing of web scraping where the court had acknowledged the harm caused against Ebay as it reduced its performance levels and damage to the reputation of the website as well as the fact that BE's scraping may have been outdated or inaccurate information. Therefore, the court ruling against BE had been recognized as the invasion of Ebay's valuable intellectual property and rule over the possession of their property interests in its server capacity, as BE has been acquiring the said server capacity and harvesting resources of it as competitors move against another. It has been shown that the court issued a preliminary injunction against BE (George F. Fibbe, 2014). This court case encouraged several other companies to come forward with their court cases against their opponent companies as it has been causing extreme damage to the breaching of data and unauthorized access to their sensitive information. A similar case was done by Tickets.com against Ticketmaster which had opposing results to Ebays. According to (George) 2014, " No evidence was before the court that Tickets.com's use interfered with Ticketmaster's regular business, and, unlike in Ebay, there was no specter of "dozens or more parasites joining the fray" Also, the court did not recognize a substantial commercial harm caused by tickets.com's activity where ticket buyers were sent to ticketmasters site to make their purchases".

3.0 Implementation and Requirements

In this paper, we have implemented a tool (facebook_scraper) which is inspired originally by (twitter-scraper) which is a twitter scraping tool. Scraping data from websites such as Facebook and Twitter has gained popularity due to the multiple tasks which can be performed using such data. Furthermore, NLP (Natural - Language - Processing) and specifically sentiment analysis can be easily performed by utilizing data which can be acquired through Facebook and Twitter. These are known platforms where people can express their sentiments regarding a recurrent issue or problem.

When performing website scraping on Facebook, Facebook has a certain mechanism to limit unauthorized scraping. Moreover, these mechanisms emphasize mainly on technical mitigations against scraping. For example, adding a rate limit capping the number of times a user can interact with products in a given amount of time, data limits which keep people from getting more data than they should need to use Facebook products normally, and blocking an account with abnormal activities specifically automated computer activities through pattern recognition which we faced in one of our attempts to scrape data from Facebook.

The tool “facebook_scraper” allows you to scrape data from public pages or groups on Facebook. We scraped the comments of a public page’s post which is why it wasn’t considered as an illegal action. To elaborate, you could directly scrape the comments of a post however: if you are planning to scrape comments for multiple posts and pages in a single run, then you need to extract cookies from the browser after logging into Facebook. For instance you can use Firefox extensions such as Cookie Quick Manager.

3.1 Code & Run Overview

We will delve into the code and explain it in the following points:

Tools and Libraries:

1. re (Regular Expressions):

- A Python module which provides support for regular expressions.
- It provides functionalities such as pattern matching and manipulation of strings based on specified rules.

2. csv (Handling .csv files):

- A python module which is used for reading/writing CSV (Comma Separated Values) files.
- It provides functionalities to handle data in tabular format and is often used for database management.

3. tkinter (Tkinter GUI Toolkit):

- The Standard GUI (Graphical User Interface) toolkit for Python.
- It provides classes and functions to create graphical desktop applications with widgets like buttons, labels, entry fields, etc.

4. facebook_scraper:

- A Python library for scraping data from Facebook without needing an API key.
- It provides functionalities to scrape data from Facebook in various forms such as posts, comments, reactions, etc.
- Often used to simplify the process of accessing Facebook data programmatically for research, analysis and many other purposes.

Code overview:

1. User Input:

The user will enter the Post ID, maximum number of comments to be extracted, and the name of the CSV file the extracted comments should be saved to.

2. Validation:

The code will validate the format of the entered Post ID using a regular expression to ensure it matches the expected format.

- If the format of the Post ID is invalid, an error message will be displayed to the user.

3. CSV Filename Formatting:

In this part of the code, we ensured that if the user was to forget to type the “.csv” extension, it shall be added by default to the file name provided by the user so that the file is saved in “.csv” format.

4. Scraping Process:

- “get_posts()” function from “facebook_scraper” library will be called with passing the following parameters: {post_urls = [post_id], options = {“Comments” : max_comments, “progress” : True}}, where the ID of the post, maximum number of comments are specified, and the last parameter “progress” set to True to enable progress tracking.
- The first post matching the provided Post ID will be retrieved from the generator returned by “get_posts()”.
- The comments which belong to the post will be extracted alongside the replies if found.

5. CSV File Handling:

- A CSV file will be opened in write mode with UTF-8 encoding and the field names for the CSV file (‘Comment’, ‘Reply’) will be defined and written to the header row.
- Each comment extracted will be iterated over and written under the ‘Comment’ column in the CSV file. Similarly for the replies and will be written under the ‘Reply’ column.
- The file will be closed once all comments and replies have been written.

6. User Feedback:

Case 1: If the scraping is successful, a message dialog will be displayed to inform the user that the comments have been successfully scraped and saved to the CSV file.

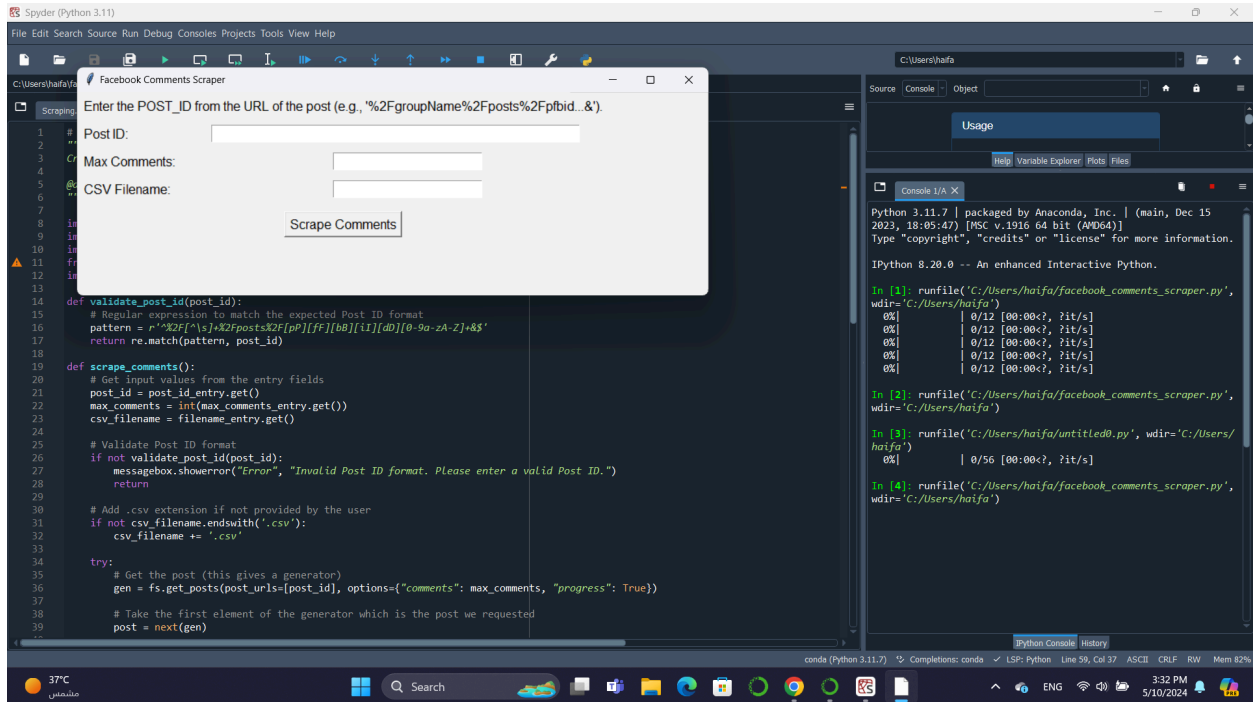
Case 2: If the scraping is unsuccessful, a message dialog will be displayed to provide the details of the error.

7. Tkinter GUI Interaction:

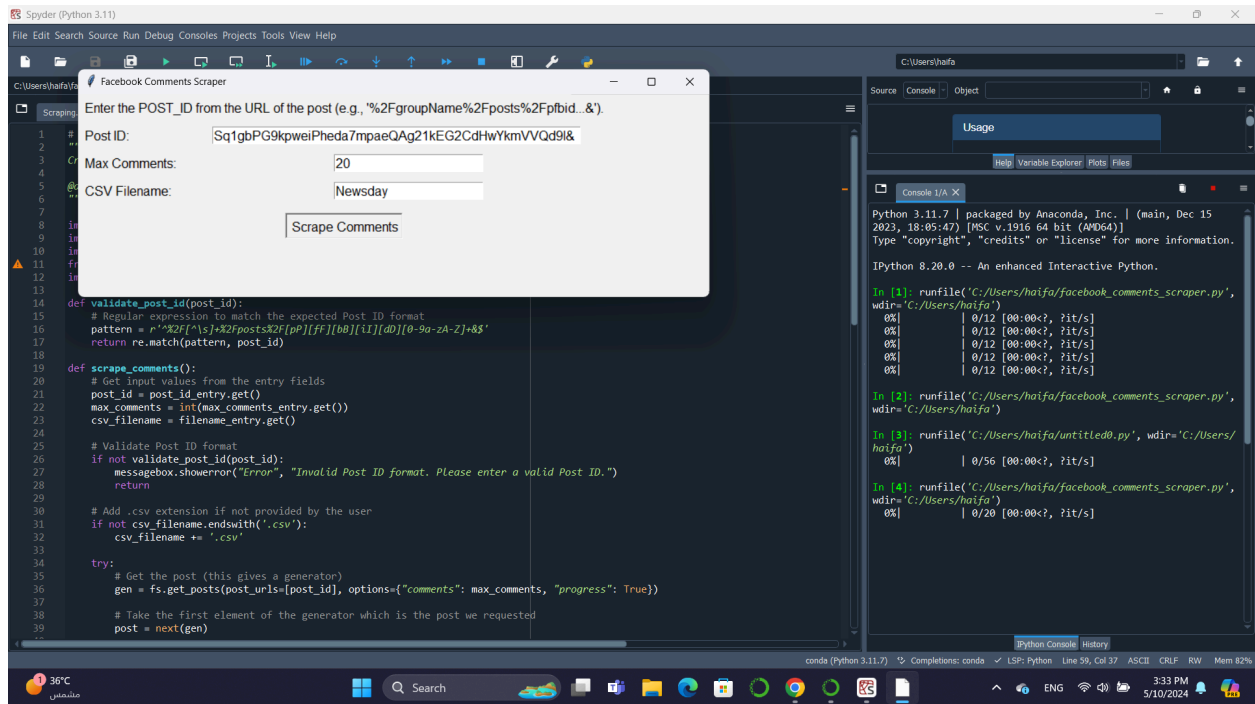
- A part of the code will create a GUI using “tkinter” tool, allowing users to interact and enter the required input to initiate the scraping process.
- Widgets such as entry fields for Post ID, maximum number of comments, and CSV file name, in addition to a button to start the scraping will be displayed inside the window.

- The Tkinter event loop runs, enabling user interaction and handling GUI events until the window is closed.

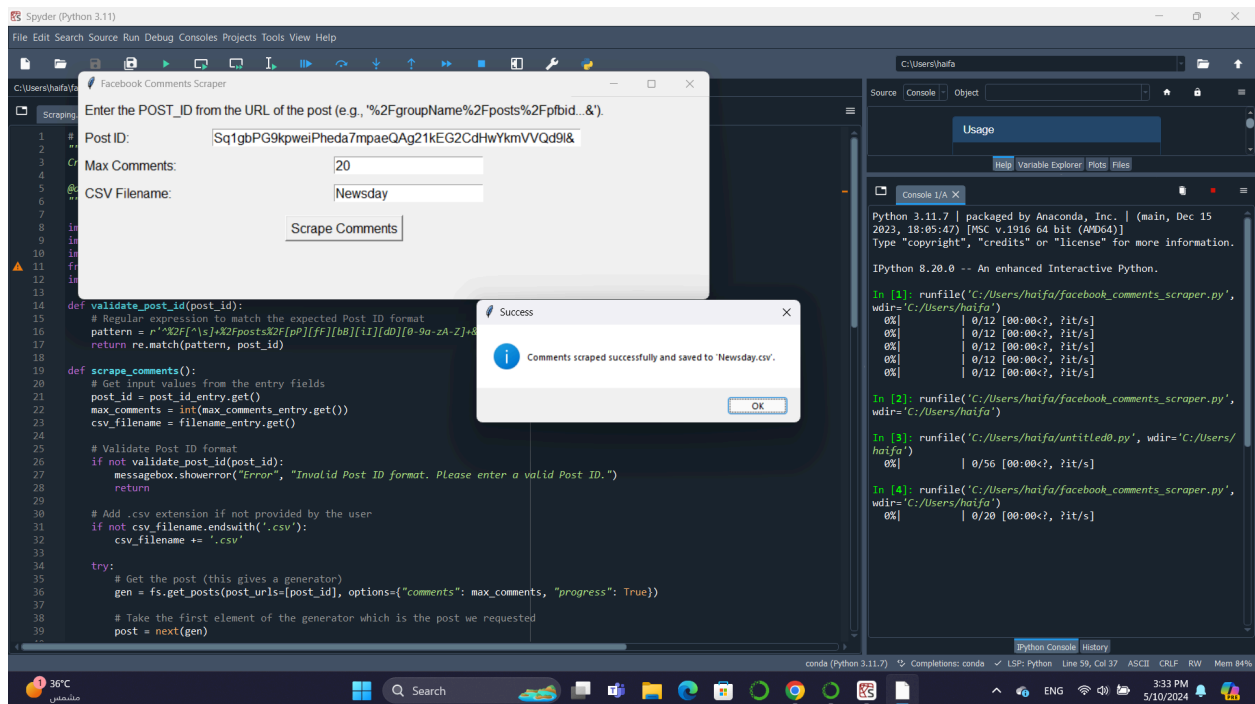
Run with Screenshots of possible scenarios:



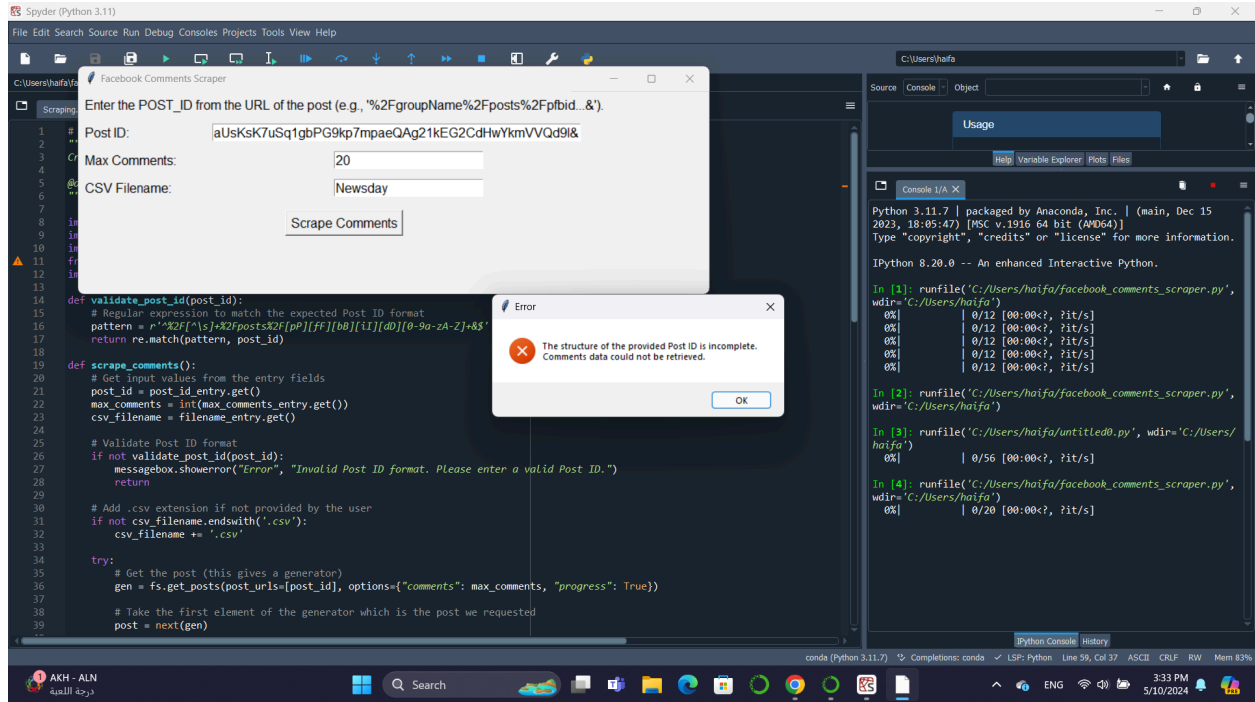
(GUI Run)



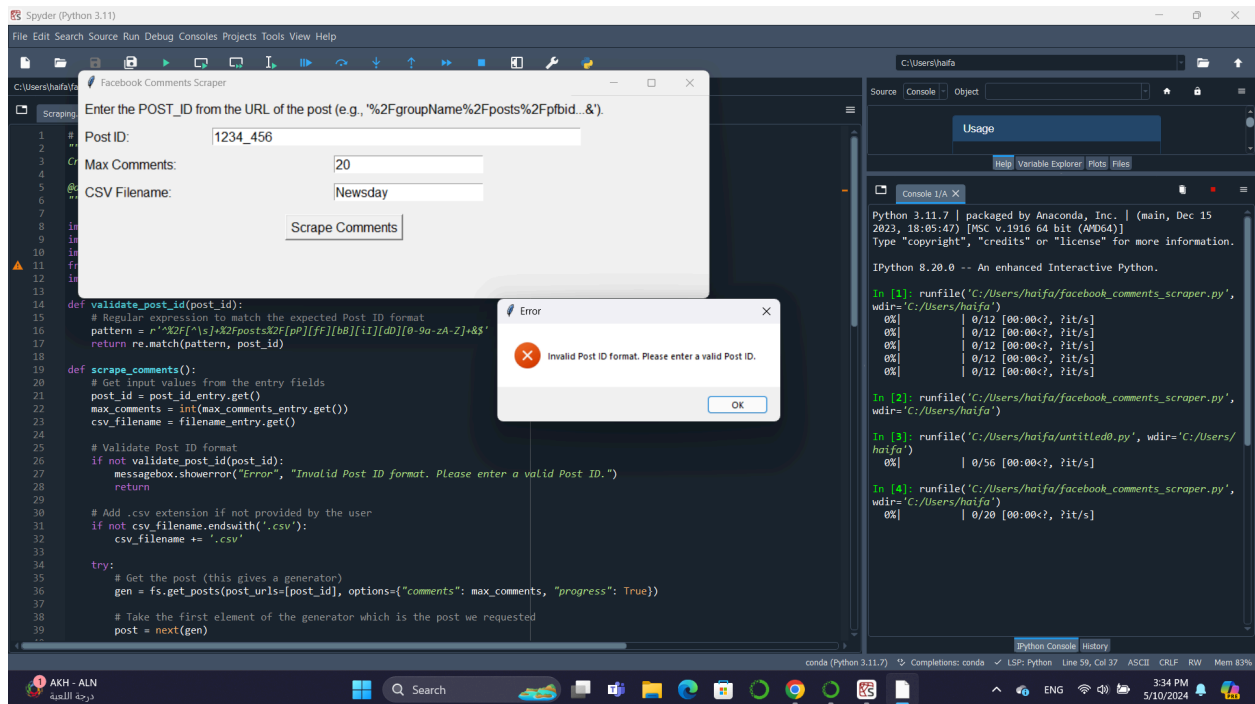
(Entering Input)



(Successful Scraping)



(Unsuccessful Scrapping - Missing Post ID structure)



(Unsuccessful Scrapping - Invalid Post ID)

Run Time Analysis and Complexity:

1. **Input Parsing:** Takes constant time complexity $O(1)$.
2. **Validation:** Takes linear time complexity $O(n)$, where n is the size of the input Post ID string.

3. **Facebook Scraping:** Its complexity depends on the number of comments to scrape which is $O(m)$.
4. **Writing to a CSV:** Its complexity is proportional to the total number of comments and replies which is $O(c)$.
5. **Exception Handling:** Doesn't have a significance on the time complexity.

The Total runtime = $O(1) + O(n) + O(m) + O(c)$, where $O(m)$ is likely to be the most dominant term. Furthermore, the complexity for scraping is heavily dependent on other factors such as network latency and response time from Facebook's servers.

4.0 Application and Recommendations to Overcome Threats

As society evolves in a technology driven world in today's modern day, web scraping has been unanimously relied on by organizations and business for its great use of harvesting data from opponent websites. Moreover, the mere dependency effect on web scraping for gathering information has its own challenges and extreme legal issues as opposed to its effectiveness as scrapers main target of data is usually public, it is essential to understand the potential threats and attacks that may overlap within its usage as well as the application methods and strategies industries are required to understand in order to mitigate the potential damage that may overcome within.

Maintaining the following recommended Facebook settings to prevent your data from unauthorized scraping possess great usage over the applications and recommendations to FaceBook web scraping.

- Navigate through your "Privacy Settings" in order to utilize your viewing options of who can see your public information or what is being shared.
- Walk through your "How People Find and Contact You" option on your settings page to utilize who can find your account through email and phone number.
- Utilize your basic information on your Facebook profile to ensure the visibility of you basic information.
- Customize your basic information on your Facebook page to control what is being shared with whom when posting.
- Follow up with the "Privacy Matters" page on Facebook for continuously informing yourself on the privacy initiatives at Meta Inc.

- You can also detect the possibility of your own data breaches by inspecting your data on the 'Have I Been Pwned' website.

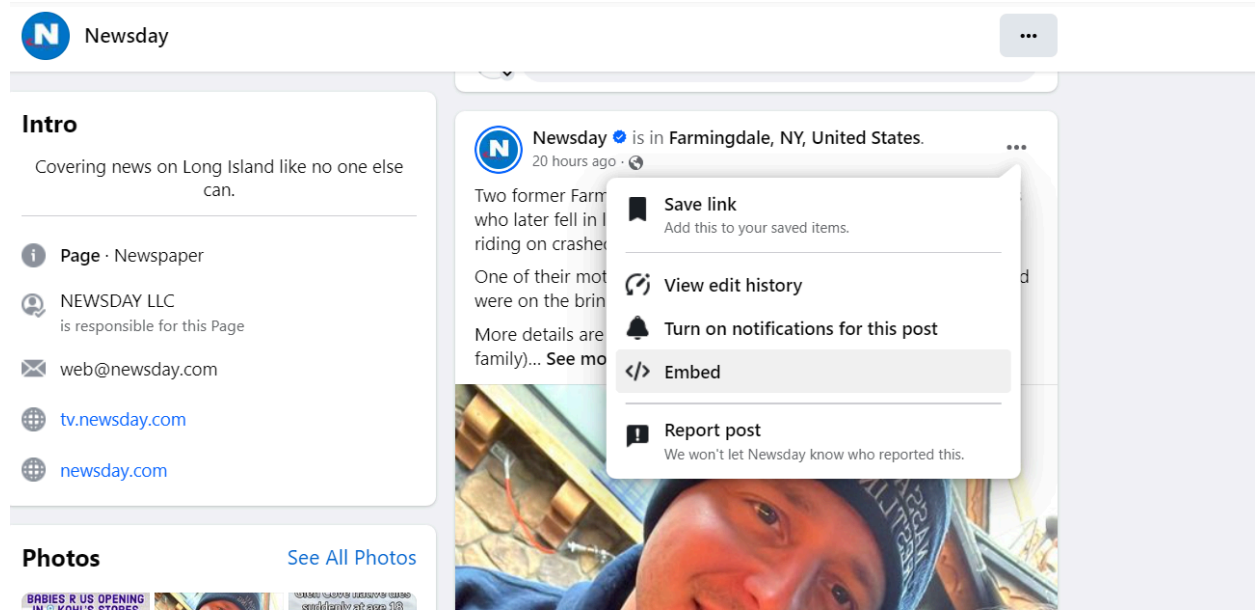
5.0 Conclusion

As the internet expands within the years of technology, it encounters advantages and disadvantages with tools that are implemented onto its websites. Throughout this project, we have explored web scraping to be a valuable software tool that helps in harvesting and collecting data from related websites and online resources which transforms it into structured datasets efficiently, which can be further used for various purposes as it can be a reason to support or oppose an infrastructure with the collected data; Moreover, we have addressed the security concerns that may accompany the scraped website which can be solved and acquire control over these threats by several discussed strategies such as the implemented Facebook methods or the previous methods discussed in the literature review. We have also addressed the common security risks, threats, attacks, and vulnerabilities that may convolve the websites and how it will affect it. Moving forward, the literature review discusses several methods in overcoming the challenges that may overbare the scraping where a conducted security incident elaborated throughout the document which discusses how web scraping has been affecting several organizations negatively as well as positively for both sides. In addition, we have implemented a code on Python that demonstrates how web scraping is done in action on Facebook website while also implementing a graphical user interface (GUI) that demonstrates the scraping could be done in action as well.

Overall, web scraping has a valuable application in the real world however it is mainly concealed for its downsides as it affects several cybersecurity protocols and methods which can cause damage and several threats and attacks for the scraped website, it is important to protect and implement security tools and create a safe environment to provide confidentiality, integrity, and authentication.

6.0 Appendix

6.1 Guide on how to find Post ID in the right format



The screenshot shows a Facebook page for Newsday. The post is from Newsday, located in Farmingdale, NY, United States, posted 20 hours ago. The post text reads: "Two former Farmingdale High School students who later fell in love while riding on crashed motorcycles. One of their motorcycles was on the bridge. More details are in the family... See more". The 'Embed' option is selected in the menu.

Intro
Covering news on Long Island like no one else can.

Page · Newspaper

NEWSDAY LLC is responsible for this Page

web@newsday.com

tv.newsday.com

newsday.com

Photos See All Photos

Embed Post

Copy and paste this code into your website. ☒ Include full post

%2Fnewsday%2Fposts%2Ffbid02VK7HVbuJMu1baUs [Copy Code](#)

Advanced settings

The advanced settings section shows a preview of the embedded post, which includes the Newsday logo, the location (Farmingdale, NY, United States), the post text, and the photo of the couple.

6.2 Code

```
import re
import csv
import tkinter as tk
from tkinter import messagebox, ttk
import facebook_scraper as fs

def validate_post_id(post_id):
    # Regular expression to match the expected Post ID format
    pattern = r'^%2F[^\s]+%2Fposts%2F[pP][fF][bB][iI][dD][0-9a-zA-Z]+&$'
    return re.match(pattern, post_id)

def scrape_comments():
    # Get input values from the entry fields
    post_id = post_id_entry.get()
    max_comments = int(max_comments_entry.get())
    csv_filename = filename_entry.get()

    # Validate Post ID format
    if not validate_post_id(post_id):
        messagebox.showerror("Error", "Invalid Post ID format. Please enter a valid Post ID.")
        return

    # Add .csv extension if not provided by the user
    if not csv_filename.endswith('.csv'):
        csv_filename += '.csv'

    try:
        # Get the post (this gives a generator)
        gen = fs.get_posts(post_urls=[post_id], options={"comments": max_comments, "progress":
True})

        # Take the first element of the generator which is the post we requested
        post = next(gen)

        # Check if the 'comments_full' key exists in the post object
        if 'comments_full' not in post:
            messagebox.showerror("Error", "The structure of the provided Post ID is incomplete.
Comments data could not be retrieved.")
            return
```

```

# Extract the comments part
comments = post['comments_full']

# Open the CSV file in write mode
with open(csv_filename, 'w', newline="", encoding='utf-8') as csvfile:
    # Define fieldnames for the CSV file
    fieldnames = ['Comment', 'Reply']
    writer = csv.DictWriter(csvfile, fieldnames=fieldnames)

    # Write the header row
    writer.writeheader()

    # Process comments and replies
    for comment in comments:
        # Write comment to CSV
        writer.writerow({'Comment': comment['comment_text'], 'Reply': ''})

    # Process replies
    for reply in comment['replies']:
        # Write reply to CSV
        writer.writerow({'Comment': '', 'Reply': reply['comment_text']})

    messagebox.showinfo("Success", "Comments scraped successfully and saved to  
'{}'.".format(csv_filename))
except Exception as e:
    messagebox.showerror("Error", "An error occurred: {}".format(str(e)))

# Create the main window
root = tk.Tk()
root.title("Facebook Comments Scraper")

# Set window size
root.geometry("600x250")

# Increase font size for text and tabs
font_style = ("Arial", 12)

# Create and pack widgets

```

```

tk.Label(root, text="Enter the POST_ID from the URL of the post (e.g.,
'%2FgroupName%2Fposts%2Fpfbid...&').", font=font_style).grid(row=0, column=0,
columnspan=2, padx=5, pady=5)
tk.Label(root, text="Post ID:", font=font_style).grid(row=1, column=0, sticky="w", padx=5,
pady=5)
post_id_entry = tk.Entry(root, width=50, font=font_style)
post_id_entry.grid(row=1, column=1, sticky="w", padx=5, pady=5)

tk.Label(root, text="Max Comments:", font=font_style).grid(row=2, column=0, sticky="w",
padx=5, pady=5)
max_comments_entry = tk.Entry(root, font=font_style)
max_comments_entry.grid(row=2, column=1, padx=5, pady=5)

tk.Label(root, text="CSV Filename:", font=font_style).grid(row=3, column=0, sticky="w",
padx=5, pady=5)
filename_entry = tk.Entry(root, font=font_style)
filename_entry.grid(row=3, column=1, padx=5, pady=5)

scrape_button = tk.Button(root, text="Scrape Comments", command=scrape_comments,
font=font_style)
scrape_button.grid(row=4, column=0, columnspan=2, pady=10)

# Start the Tkinter event loop
root.mainloop()

```

7.0 References

- Fibbe, G. H. (2004). Mercer Law Review Vol. 055 Issue 03-041 pg. 1011-Screen-Scraping and Harmful Cybertrespass after Intel.
- *Web scraping protection: How to prevent web scraping.* (2022. Nov 7) DataDome.
<https://datadome.co/learning-center/scrapper-crawler-bots-how-to-protect-your-website-against-intensive-scraping/>
- *What is data scraping: Techniques, Tools & Mitigation: Imperva.* (2023, December 20). Imperva a Thales Company.
<https://www.imperva.com/learn/application-security/data-scraping/>
- *What is web scraping and how to use it?.*(2024, March 7). GeeksforGeeks.
<https://www.geeksforgeeks.org/what-is-web-scraping-and-how-to-use-it/>