Project Report

ON

Scrape Sonic

SUBMITTED IN PARTIAL FULFILLMENT OF THE
REQUIREMENT FOR SEMESTER IV OF

S.E. (Information Technology)

SUBMITTED BY

Mr. Tejas Rokade

Mr. Pranav Raikar

Mr. Gaurang Raorane

Mr. Anurag Gaiwal

UNDER THE GUIDANCE OF

Ms. Sukanya Roychowdhury

DEPARTMENT OF INFORMATION TECHNOLOGY
V.E.S. INSTITUTE OF TECHNOLOGY
2022-23

Certificate

This is to certify that project entitled

"Scrape Sonic"

Group Members Names

Mr. Tejas Rokade
Mr. Pranav Raikar
Mr. Gaurang Raorane
Mr.Anurag Gaiwal

In partial fulfillment of degree of BE. (Sem IV) in Information Technology for Project is approved.

Ms. Sukanya Roychowdhury
Project Mentor

External Examiner

Dr.(Mrs.)Shalu Chopra
H.O.D

Dr.(Mrs.)J.M.Nair
Principal

Date:29/04/2023
Place: VESIT, Chembur

College Seal

Declaration

I declare that this written submission represents my ideas in my own words and where others' ideas or words have been included, I have adequately cited and referenced the original sources. I also declare that I have adhered to all principles of academic honesty and integrity and have not misrepresented or fabricated or falsified any idea/data/fact/source in my submission. I understand that any violation of the above will be cause for disciplinary action by the Institute and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been taken when needed.

- - - - - - - - - - -

- - - - - - - - - - -

- - - - - - - - - - -

- - - - - - - - - - -

(Signature)

Mr. Tejas Rokade
Mr. Pranav Raikar
Mr. Gaurang Raorane
Mr.Anurag Gaiwal

ACKNOWLEDGEMENT

The project report on "Scrape Sonic" is the outcome of the guidance, moral support and devotion bestowed on our group throughout our work. For this we acknowledge and express our profound sense of gratitude to everybody who has been the source of inspiration throughout project preparation. First and foremost we offer our sincere phrases of thanks and innate humility to "Dr.(Mrs.)Shalu Chopra", "Ms.Sukanya Roychowdhury" for providing the valuable inputs and the consistent guidance and support provided by them.We can say in words that we must at outset tender our intimacy for receipt of affectionate care to Vivekanand Education Society's Institute of Technology for providing such a stimulating atmosphere and conducive work environment.

Abstract

Scrapesonic is a web scraping tool developed using Python, Tkinter, and several libraries, including Beautiful Soup, Requests, PIL, Pandas, and NumPy. The tool extracts specific product information from the Flipkart website, including names, prices, ratings, and categories, and exports the extracted data to a CSV file. The Scrapesonic project aims to provide an intuitive and user-friendly GUI that can facilitate the data extraction and analysis process. By providing an open-source tool that can be improved and expanded upon by others in the community, we hope to promote collaboration and innovation in the field of web scraping and data analysis. Overall, the Scrapesonic project aims to provide a valuable resource for businesses and individuals seeking to analyze the e-commerce market and promote the growth of the open-source community.

Contents

1 Introduction **1**
- 1.1 Introduction . . . 1
- 1.2 Aim and Objectives . . . 1
- 1.3 Motivation for the Work . . . 1
- 1.4 Scope of Project . . . 2
- 1.5 Contribution . . . 2

2 Literature Survey **3**
- 2.1 Introduction . . . 3
- 2.2 Problem Definition . . . 4

3 Design Implementation **5**
- 3.1 Proposed System . . . 5
- 3.2 Requirement Gathering and Analysis . . . 5
- 3.3 Hardware Requirement . . . 6
- 3.4 Software Requirement . . . 6
- 3.5 Timeline Activity chart . . . 6

4 Results and Discussion **7**
- 4.1 Code . . . 7
- 4.2 Screen Shots . . . 10

5 Conclusion **17**
- 5.1 Summary . . . 17
- 5.2 Future Scope . . . 17

Chapter 1

Introduction

1.1 Introduction

In today's world the internet is best suited to obtain the most relevant data. Web scraping is the process of extracting and parsing data from websites in an automated fashion using a computer program. It's a useful technique for creating datasets for research and learning.

Review analysis is the act of going through customer and product reviews from a number of different channels and uncovering insights. These insights can then be used to improve products and services, create new ones, or enhance the overall customer experience.

It can also help in saving the time of user just by showing the relevant data he or she wants.

1.2 Aim and Objectives

- Businesses scrap product reviews. Some use it to improve their products, while others monitor their competition. Whatever the reason, scraping product reviews can provide valuable insights into customer sentiment.
- To serve the customers better, businesses need to be aware of their feedback. By collecting and analyzing the customers' reviews, businesses can know about insightful trends of customers and fine-tune their products and services accordingly.
- Through our software customers will be able to see the reviews of their favourite books at one place and in more structured and descriptive way.

1.3 Motivation for the Work

Our motivation for the Scrapesonic project was to create a tool that could efficiently extract specific data from the Flipkart website and provide businesses and individuals with insights into the e-commerce market. We aimed to develop an intuitive and user-friendly GUI and provide an open-source project that could be used and improved by others.

1.4 Scope of Project

- Extraction of product information such as names, prices, ratings, and categories from the Flipkart website.
- Development of a user-friendly GUI using the Tkinter library.
- Utilization of the Beautiful Soup, Requests, PIL, Pandas, and NumPy libraries to aid in the data extraction and analysis process.
- Exporting of extracted data to a CSV file.
- Implementation of error handling to ensure that the tool can handle exceptions and unexpected behavior.
- Provision of open-source code that can be improved and expanded upon by others in the community.

1.5 Contribution

Our Scrapesonic project is an tool that we developed with the intention of providing a valuable resource to the community of web scraping and data analysis enthusiasts. By sharing our code, we hope to enable others to extract and analyze e-commerce data accurately and efficiently.

Additionally, we aimed to provide a user-friendly GUI and clean code that could be easily understood and improved by others. We welcome contributions from the community, including suggestions for improvements, bug fixes, and additional features.

Overall, our contribution to the community through the Scrapesonic project is an effort to promote collaboration and innovation in the field of web scraping and data analysis.

Chapter 2

Literature Survey

2.1 Introduction

To know how the data extraction process has evolved, one must understand the techniques involved in this method of web scraping. The impact behind business web scraping has dependably been to pick up a simple business advantage and incorporate things like undermining a contender's special valuing, taking leads, commandeering promoting efforts and many more.[1]

Web scraping is widely utilized for a variety of purposes, including comparing prices online, observing changes in weather data, website change detection, research, integrating data from multiple sources, extracting offers and discounts, scraping job postings information from job portals, brand monitoring, and market analysis.[2]

The implementation part of the system starts by taking the URL of the website. System scraps all the links of the top stories. After taking the link of each top stories system scraps the content of those links. System keeps monitoring in the change in the contents of the site. System can be implemented using a python library known as beautiful soup. Beautiful soup can also be used to implement the monitoring of changes.[4]

Already existing web scraping techniques

- Traditional copy and paste [3]
- Regular expression(Regex)[1]
- Hyper Text Markup Language (HTML) Parsing [3]
- Web Scraping Software [3]

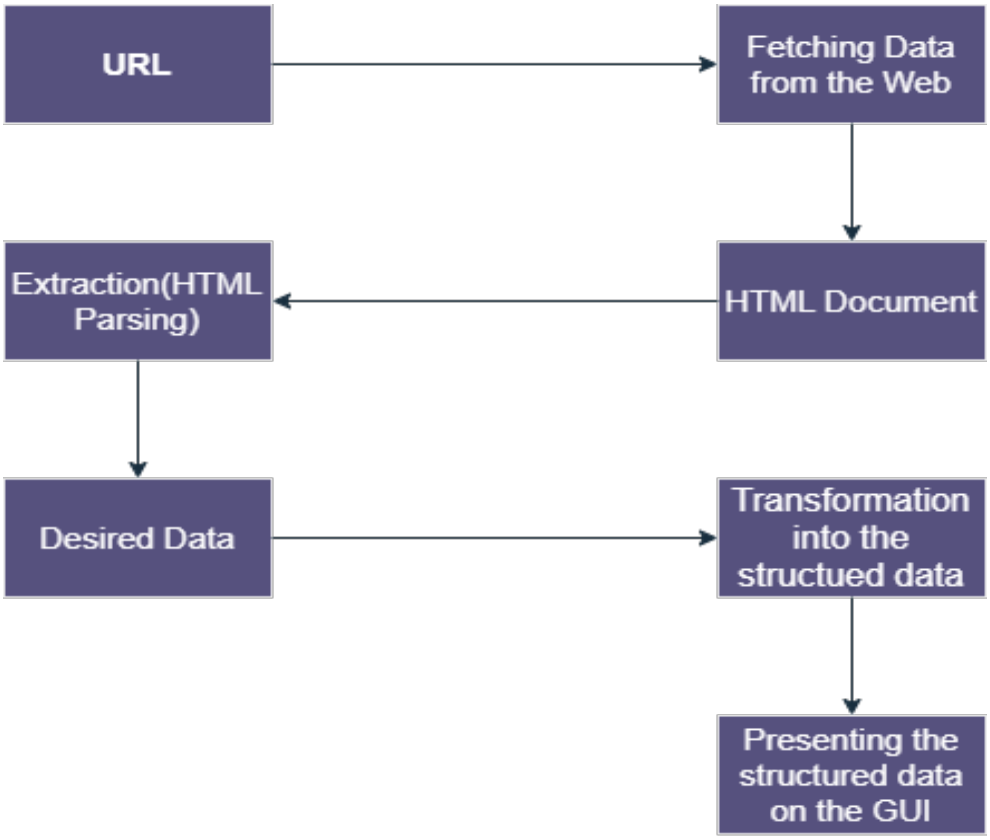
2.2 Problem Definition

Many people face problems in getting relevant data of a particular product, as a result they end up in a big confusion in preference of suitable product, which may leads to selecting a wrong product just because they didn't read the correct and relevant reviews. The online shopping experience on websites such as Amazon, Flipkart, Meesho, Myntra, and others can be challenging for buyers due to several factors. Customers tend to overspend, leading to financial losses, and often feel that they have limited choice because websites display products they want to sell. Moreover, browsing through multiple products and their information can make decision-making tough.

Chapter 3

Design Implementation

3.1 Proposed System



3.2 Requirement Gathering and Analysis

To ensure that our Scrapesonic project met its objectives, we conducted user interviews, analyzed similar applications, and studied project specifications to identify the necessary requirements. Key requirements included extracting specific data from the Flipkart website, designing an intuitive GUI, and ensuring efficient data handling. To analyze these requirements, we used methods such as creating use cases and defining user personas. This process helped us to prioritize and develop an application that met our objectives and provided a positive user experience.

3.3 Hardware Requirement

- Core i5 and above.
- Processor: Minimum 1 Ghz processor; (Recommended 2GHz or more).
- Hard Drive: Minimum 32 GB; (Recommended 64GB or more).
- Memory (RAM): Minimum 1 GB; (Recommended 4 GB or above).

3.4 Software Requirement

- Windows: 7 or newer
- MAC: OS X v10.7 or higher
- Linux: Ubuntu
- Any Python IDE.

3.5 Timeline Activity chart

Chapter 4

Results and Discussion

4.1 Code

Code for Individual links

```
from bs4 import BeautifulSoup
import requests


class Scraper:
    ''' Scraper Class
    '''

    def url_linker(self, total_pages=0, *args):
        self.urls = list()
        for page in range(1, total_pages+1):
            self.urls.append(self.url_builder(page, *args))
        return self.urls

    def url_builder(self, page=0, *args):
        self.parms = '+'.join(args)
        self.url = f'https://www.flipkart.com/search?q={self.parms}&page=
        return self.url

    def scrapper(self, urls: list, category, pages):

        self.names = list()
        self.data = list()

        for url in urls:
            print(url)

            response = requests.get(url).content
            soup = BeautifulSoup(response, 'html.parser')
            all_blocks = soup.findAll('div', class_='_1YokD2 _3Mn1Gg')
            for product in all_blocks:
```

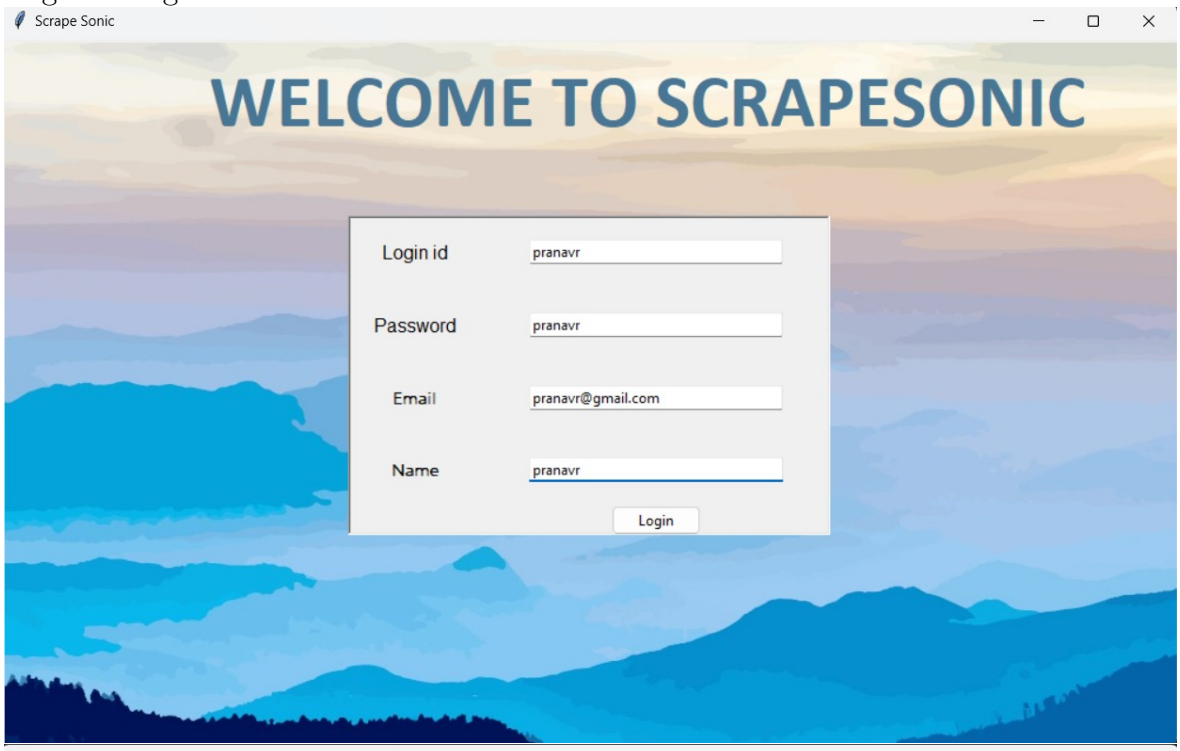
```
                try:
                    if product.find('a', class_='s1Q9rs').get_text():

                        name = product.find('a', class_='s1Q9rs').get('hr
                        flipkart = f'https://www.flipkart.com'
                        link = flipkart+name
                        self.names.append(link)
                        print("Name ", link)

                except Exception as e:
                    continue

        self.data = self.names  # data is a list which has our generated

        return self.data

    # this function sends the actual link of the product recieved from th
    def individuallink_scraper(self, category, pages=0):
        self.pages = pages
        self.category = category

        # url of page containing mainy blocks is obtained
        urls = self.url_linker(self.pages, self.category)
        # thus this function send actual link of a product in the form of
        url = self.scrapper(urls, self.category, self.pages)
        return url[0]
```

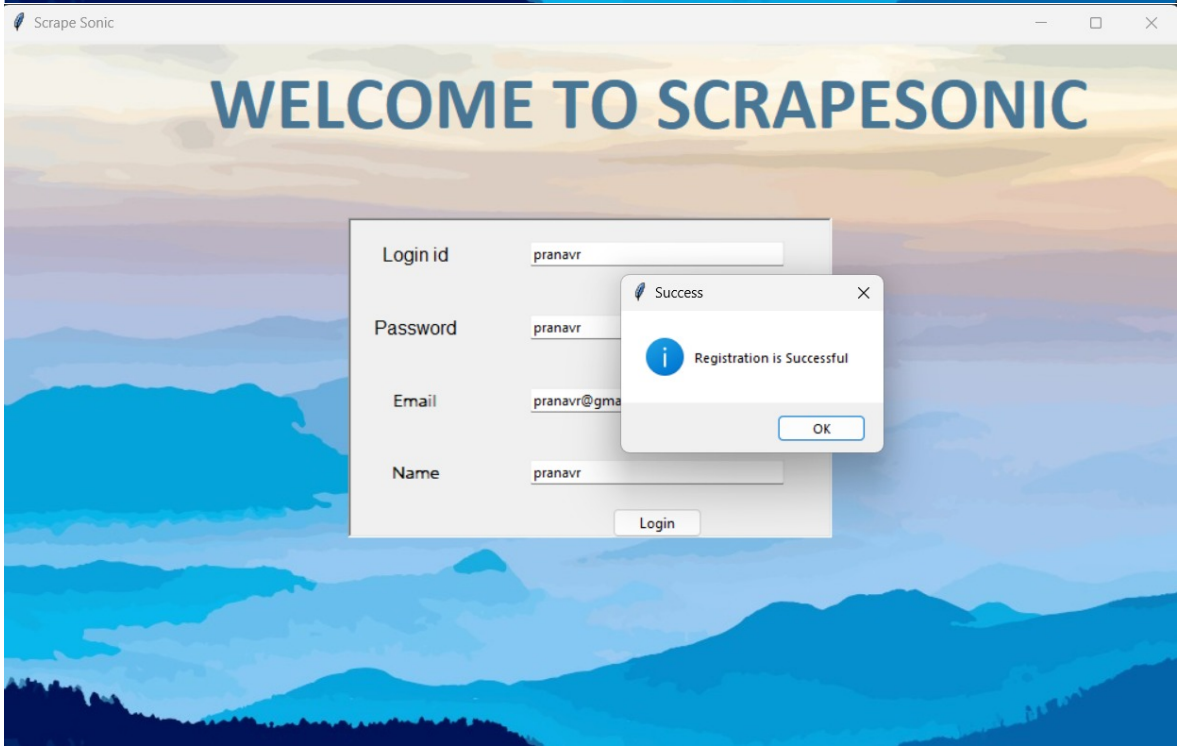
Code for Description

```
from bs4 import BeautifulSoup
import requests
from individuallink import Scraper


class Scraper1:
    def scrapper(self, category, pages):

      # creating a list to store the scrapped description
        self.descriptions = list()

        # urlrecieved is  not in the form of list but is a part of list w
        urlrecieved = Scraper().individuallink_scraper(category, pages)

        print(urlrecieved)

        response = requests.get(urlrecieved).content
        soup = BeautifulSoup(response, 'html.parser')
        # description_block  is the main block of description
        description_block = soup.findAll('div', class_='_1AtVbE col-12-1
        for description in description_block:
            try:

                if description.find('div', class_='_1mXcCf').get_text():
                    description = description.find(
                        'div', class_='_1mXcCf').get_text()
                    print(description)
                    self.descriptions.append(description)

            except Exception as e:
                continue

        return self.descriptions

    def flipkart_scraper(self, category, pages=0):
        self.pages = pages
        self.category = category

        data = Scraper1().scrapper(self.category, self.pages)
# data is  a list
        print(data)

        return data
```

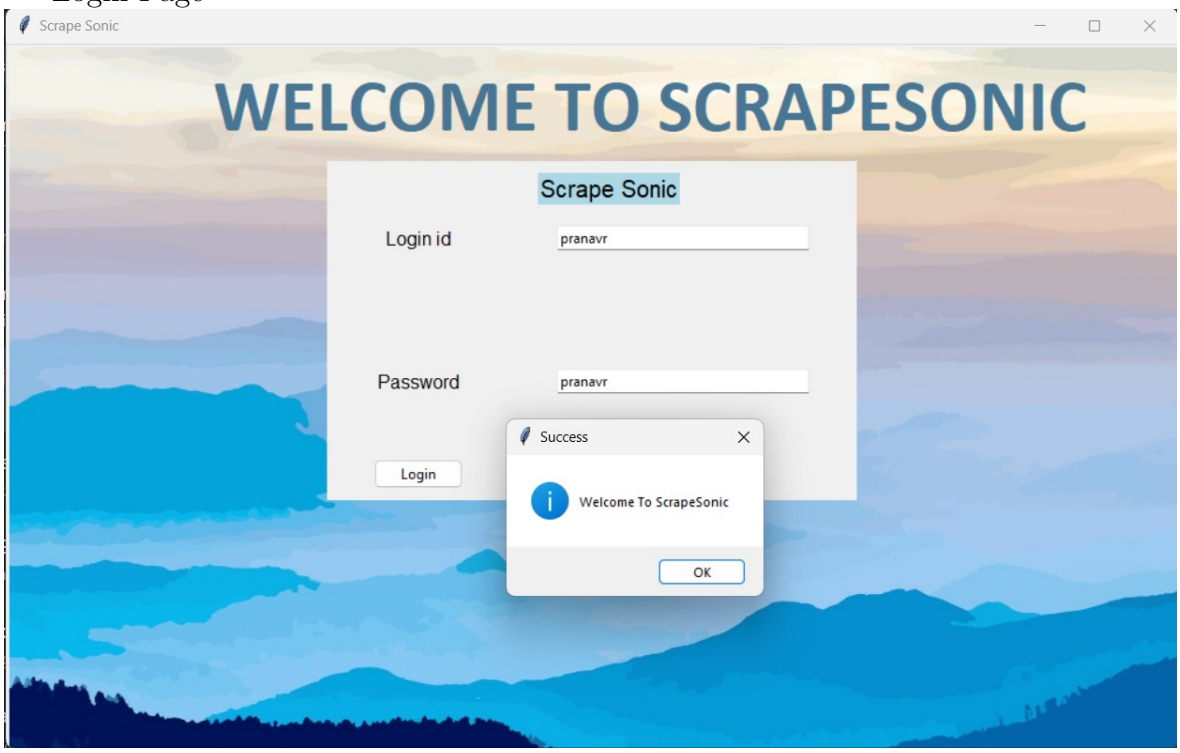
4.2 Screen Shots

Register Page



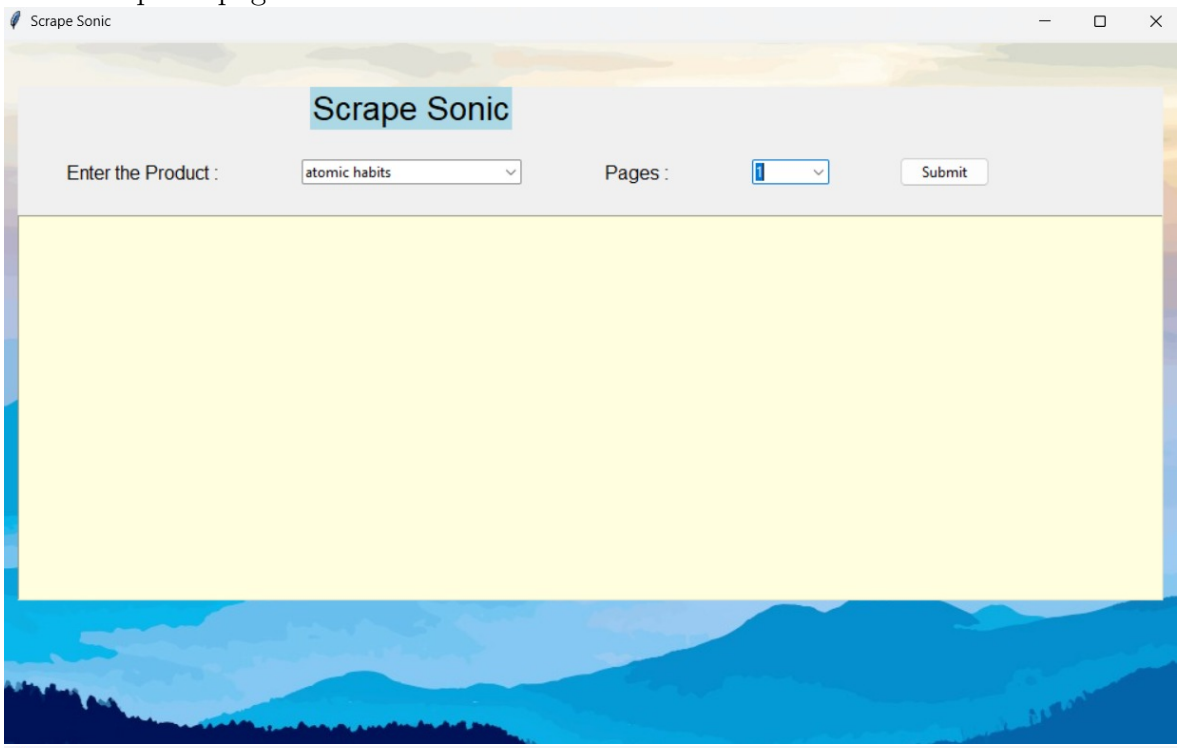


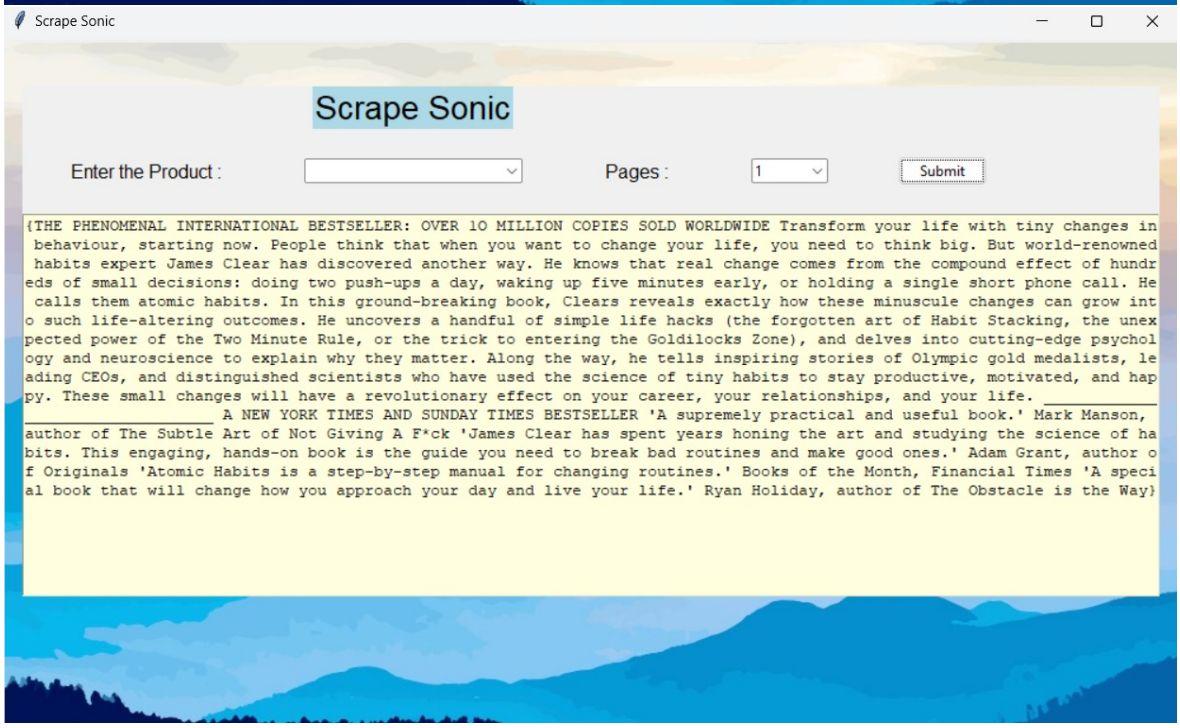
Login Page


Scrape Sonic
WELCOME TO SCRAPESONIC
Scrape Sonic
Login id
pranavr
Password
pranavr
Login
Success
Welcome To ScrapeSonic
OK

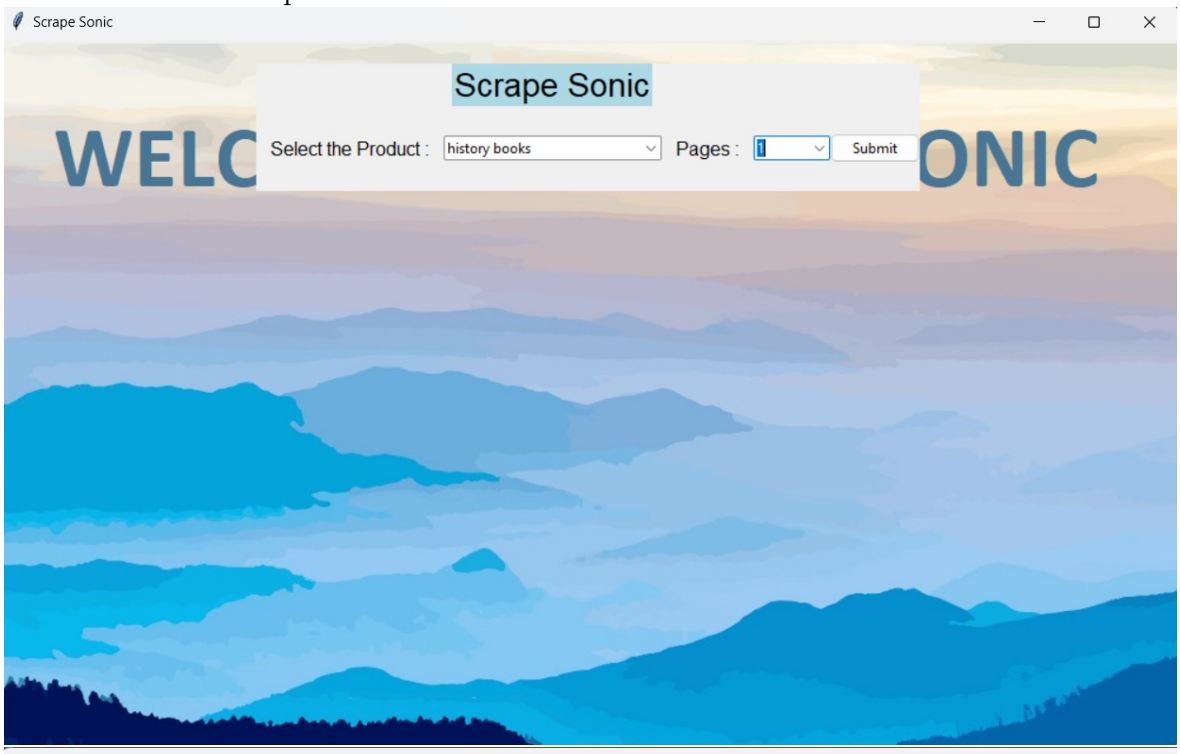
DIRECTED TO HOME

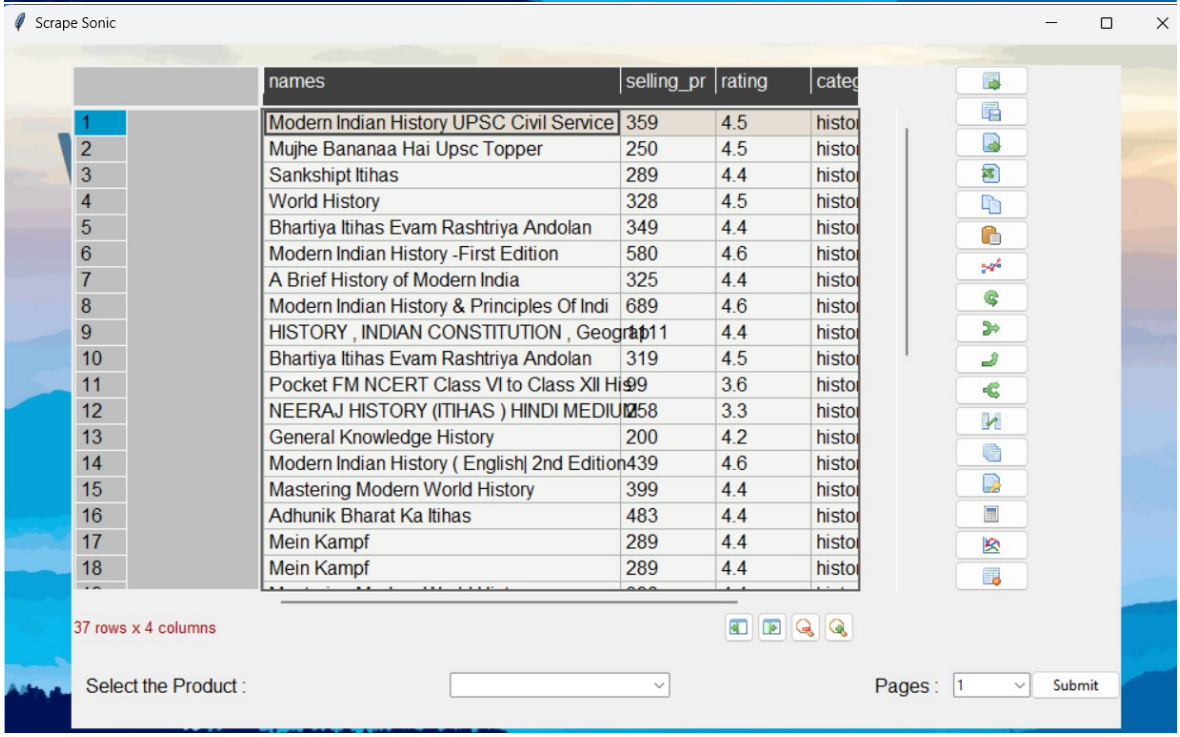
Description page



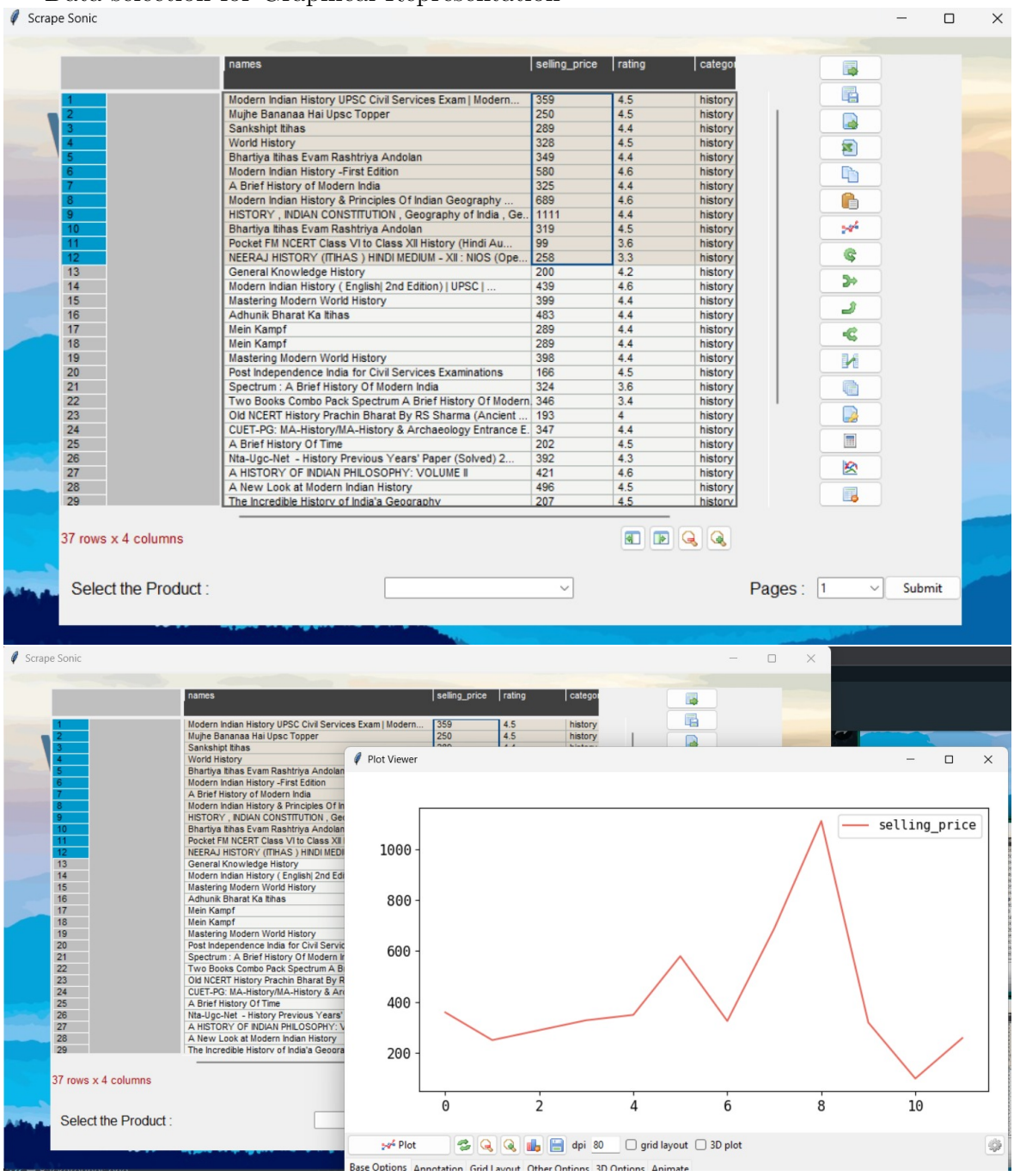


Tabular Data Representation

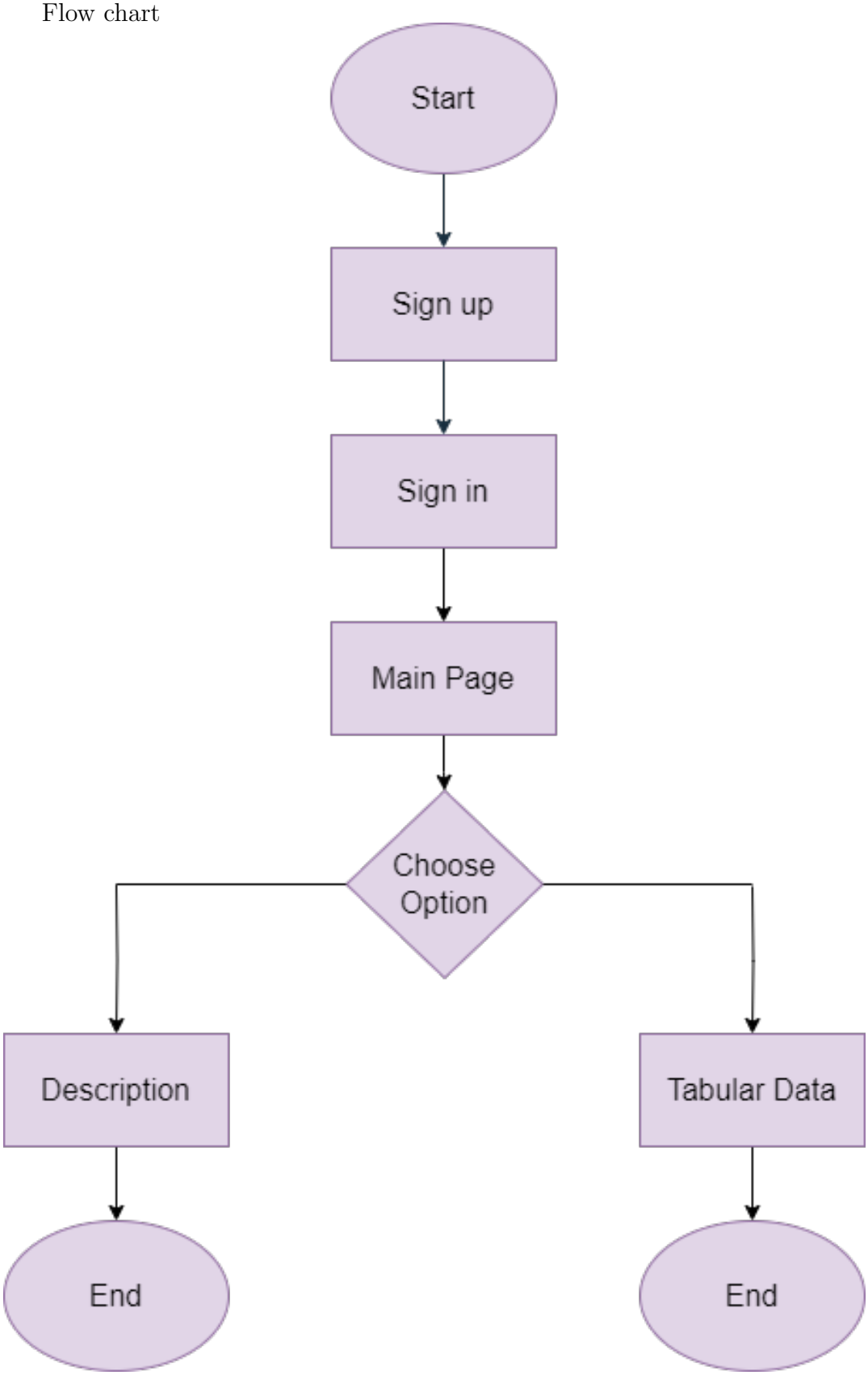




Data selection for Graphical Representation



Flow chart


Start
Sign up
Sign in
Main Page
Choose Option
Description
Tabular Data
End
End

Chapter 5

Conclusion

5.1 Summary

In the Scrapesonic project, we implemented web scraping techniques to extract data from the Flipkart website, and used Python libraries like Beautiful Soup, Requests, Pandas, and Tkinter to process and display the data in a user-friendly GUI. By doing so, we were able to provide users with easy access to product information, such as pricing, ratings, and categories, which can help inform their purchasing decisions.

In terms of societal issues, Scrapesonic addresses the problem of information overload that consumers face when trying to make informed purchasing decisions. By providing a user-friendly interface for accessing product information, Scrapesonic helps to simplify the decision-making process for consumers and promotes more informed choices. With the vast amount of information available online, it can be difficult for consumers to find relevant and accurate information about products.

5.2 Future Scope

The future scope of the Scrapesonic project is vast and diverse. Firstly, the project could be extended to include the development of a fake review detection system, which uses natural language processing and machine learning algorithms to analyze the scraped review data for suspicious patterns or inconsistencies in language use.

Additionally, the project could be expanded to include Amazon price forecasting and seller price undermining detection, which could analyze the scraped data to identify pricing trends and potentially fraudulent practices by third-party sellers. Furthermore, the project could be enhanced to provide sentimental review analysis, which could help identify the emotions expressed in reviews and provide insights into consumer preferences.

Lastly, there is potential for exploring the use of artificial intelligence and machine learning algorithms to scrape data from any website, regardless of the structure and layout.

References

[1] S.Dhoke,A.Sakhare,S.Sharma."EFFICIENT SCRAPING OF DATA FROM WEBSITES USING SELENIUM"jetirhttps://www.jetir.org/papers/JETIRFM06063.pdf (accessed Feb. 2, 2023).

[2] C.lotfi,S.Srinivasan,M.Srinivasan,I.Latrous."Web Scraping Techniques andApplications: A Literature Review"publications. https://www.publications.scrs.in/uploads/final_menuscript/863dc5628ae9215e611c22943d061742.pdf(accessed Feb. 2, 2023).

[3] S.Sirisuriya,."A Comparative Study on Web Scraping"kdu.ac.in http://ir.kdu.ac.lk/bitstream/handle/345/1051/com-059.pdf?sequence=1&isAllowed=y(accessed Feb. 2, 2023).

[4] M.Shreesha,S.Shrikara,R.Manjesh."A Novel Approach for News extraction using Web Scrapping"https://www.ijert.org/research/a-novel-approach-for-news-extraction-us pdf(accessed Feb. 2, 2023).