One Semester Individual Project

CM3203 Final Report

Enhanced Company Data Compilation for Marketing Analysis

Done by:
Md Md Farid
Author Student Number: 21071339
Supervisor: Dr. Federico Liberatore
Moderator: Dr. Walter Colombo
Co-Supervisor: Mr. Devaang Misra
School and Institution: School of Computer Science and Informatics, Cardiff University
5-9-2024

Abstract

This project report highlights the process of data extraction, which is needed to obtain facts from various sources in the physical world or across the World Wide Web. On top of that, this report shows how compiling data in a structurally manner can result in an informative tool, with details describing specific features of a company that has proven their survivability and how others can boost their organisations' survivability by learning from the former company's data. An attempt to develop this tool of compiled data with researched techniques and mixed test results was documented throughout this report, along with considerable future upgrades other individuals could actualize to enhance the product.

Acknowledgements

As the author, I want to thank my supervisor, Dr. Liberatore along with moderator Dr. Colombo, and stakeholder and co-supervisor Mr. Misra, for their efforts to evaluate my submissions. All of them showed their dedication by monitoring my progress invariably. They communicated straightforwardly with their feedback, pushing me to add more substance to the project, such as extending research to find more useful information or modifying my computer programs to fulfil a user requirement.

Table of Contents

1. Introduction

This project focused on two essential components that led to breakthroughs in the 21st century varying from scientific discoveries to commercial growth, which are data and marketing. Data is a raw figure that can appear in forms like numbers 1 and 3 or words such as ‘Subject’, ‘Physics’ and ‘Engineering’. Structuring allows data to be organized, adding meaning to the data. With the examples above, you could put ‘Physics’ as ‘Subject 1’ and ‘Engineering’ as ‘Subject 3’, declaring two modules that are to be taken as a student.

With structuring of data, it becomes useful information, and can play a diverse range of roles in the technological industry. For instance, a data analyst of a video game company can recommend a company to develop and sell FPS (First Person-Shooter) video games towards the gaming community to attract a big number of customers who would purchase and play them after finding out that FPS video games are the most popular video game genre based on the video game genres’ product sales data. For this project, the featured type of data was company data, which are facts that give a general profile of a company.

A business needs to go through multiple processes to thrive commercially, such as product development to create the goods and services for customers to experience and accounting to monitor the financial health of a business. A key business process singled out for this project was marketing. It is the process of finding out about a customer’s requirements and taking actions to make a product satisfy their wishes. A great marketing strategy can convince consumers to purchase products from a company repeatedly, gaining huge profits in return.

To understand the customer’s preferences, information is required. For instance, a survey can collect the type of food regular customers of a restaurant business prefer to consume. Learning about the surveyed information, the restaurant can alter their dishes to satisfy the customers more and make use of digital marketing tools to reach out to both regular and new customers with campaigns like an official website to promote their new menu. Other than using a survey to understand the customer’s preferences or advertising lucrative products on a website, companies can also produce short videos to showcase the capabilities of a product, like how Apple uses videos to demonstrate the features of its smartphones, and this strategy could attract technological enthusiasts or retain Apple customers.

Merging data and marketing as a solution, companies can promote themselves efficiently by analysing patterns in a digital and comprehensive resource of data to figure out the requirements of their customers, and follow-up with a marketing decision. For this project, the overall aim was to develop a program which can retrieve company data from a variety of sources across the World Wide Web and collate them to assemble a database. Based on the characteristics of a relational Database Information System (Oracle 2021), this database contains connected instances of information that marketeers can utilize to heighten campaigns and strategies for the benefit of their companies.

The work plan started with deeper research into the marketing field and suitable technologies to develop and deploy a database. Technologies include a programming language and a platform that offers database structuring. A design and analysis phase followed in which the author set the project requirements for the computer program and conceptualized its flowchart. In addition to the program, a relevant schema for marketing exercises would be devised to make use of as many data sources from the World Wide Web as possible. Next, the author trained himself to adapt to the technologies. Becoming comfortable with the technologies lessened time spent on implementation and debugging of the program. The author took an approach of mastering programming techniques and repeating implementations of the program, as concisely stated in the initial plan, until a satisfactory version of the program has been developed.

At the end of this project, the author aimed to deliver at least one programmable file that extracts data, along with some required technologies to make the programs work. As no datasets were intended to be submitted, a sample scenario of applying database information compiled from those sources to heighten a marketing campaign was documented in the Final test phase subsection under Evaluation. A video demonstration was submitted outside PATS (Project Allocation & Tracking System) as the file size was too big.

2. Background

2.1. Expounding Marketing and the Data Problem

Marketing is essential for companies to succeed as it helps maintain a lasting relationship with customers by tailoring product characteristics to suit their likings. Without good marketing strategies, newcomers may not be inclined to buy products like, for instance, an advertised dish that contains an ingredient majority of the regular consumers are opposed to. Companies would not generate a steady revenue stream from product sales and their survivability could be severed. Therefore, it is important for companies to contrive marketing strategies, and it can be enhanced by gathering relevant information, which could include more than just customer preferences.

The procedure of data-driven marketing is ineffective if the data itself is not contextualised, let alone collected. Raw data without any context itself does not provide any benefit to anyone. Words like 'Toyota' and 'Alfa-Romeo' may not be understood by a reader who has no prior knowledge of cars. If these words are part of a structured database and clustered under an attributed column titled 'CarName' to indicate that they are names of car companies, the reader will understand the context of the source and may use it to note names not suitable for a new car company to avoid a civil lawsuit. Adding bits of raw data to a dataset, coming up with descriptions to add meaningful context to and combining it all together to form a schema for data-driven marketing tasks sounds simple, but there are gaps of data that could either render the entire information ineffective or even fruitless.

To elaborate further from the previous statement, a database containing instances of information from multiple companies may lack the location (e.g. country where the company is based). Without the location, marketeers would not be able to decide which country can be a potential spot to sell their company's products and avoid competition from other businesses. Omitting the CLV (Customer Lifetime Value) from a dataset can also affect a marketing decision. A CLV determines how loyal a customer is and how much revenue is generated from the same customer for the company (Jain 2023). Without the CLV, no one knows whether the products customers purchase are appealing and strategies to retain customers either by regularly vending the same product or something new are futile.

Evidently, a complete marketing campaign or strategy requires information that covers almost if not, all angles of a company.

2.2. Current Options

Datarade is an online marketplace reputed for its global access to data, including company data, which are offered for sale as premium data by various data providers. Looking into the website, the author found company data providers like Plugindata, which provided diverse details of a company along with the total annual sales, a very useful insight for marketing strategies. However, after an email exchange with the supervisor, premium data was declared to be out of the project scope. This was because money needed to be spent just to attain more data, which may have been deemed unnecessary by the supervisor. Instead, the author was directed to find open-source data across the World Wide Web.

A great free dataset originates from Big Picture. However, compared to a premium dataset that Big Picture offers as well, the free dataset does not contain a variety of details to understand a company cohesively, only showing the basic details like a company's industry to state what type of commercial products it provides and the city in which the company's headquarters is located.

Aim:

With a database containing extracted company data, this solution was aimed at resolving gaps of information to explore the detailed angles of a company and give an advantage to marketeers in their campaigns or strategies.

Research Question(s):

What goes through the mind of a marketeer? With so many details to conjure a brilliant campaign, there are a few that, when combined, can lead a company in the profitable direction. Along with deciding what sources of data provide the details a marketeer wants, it is also essential to understand what the best practices of marketing are, to design an ethical platform for the marketeers and ensure data is to be retrieved permissibly.

There are multiple data extraction methods, but which of them is ideal for the pursuit of company data? As they may differ from other types of data semantically, the experiments would reveal the methods fit for extracting company data from any source of diverse information.

How could data be taken from a source and used to build a database efficiently whilst practicing a good code of marketing conduct? Knowing how to integrate marketing practices and data extraction methods in the finest structure possible would determine the quality of the final deliverables.

3. Technical Execution

3.1. Methodology

The overall technical execution of this project was based on the Agile methodology. After researching the available methodologies for software development, the author's first pick was the Waterfall methodology, as he could go through each phase of the SDLC (Software Development Life Cycle) once to be time-efficient and implement database technologies without making big changes post-development. However, after completing basic programming tutorials and prototyping a simple web scraping program (See Test B1 in the

Basic test phase subsection) during week 4, the author figured he could upgrade the programs each week quickly and make big post-development changes without severely affecting the project because building the first web scraper and making modifications to adjust the functionality did not cost a huge amount of time.

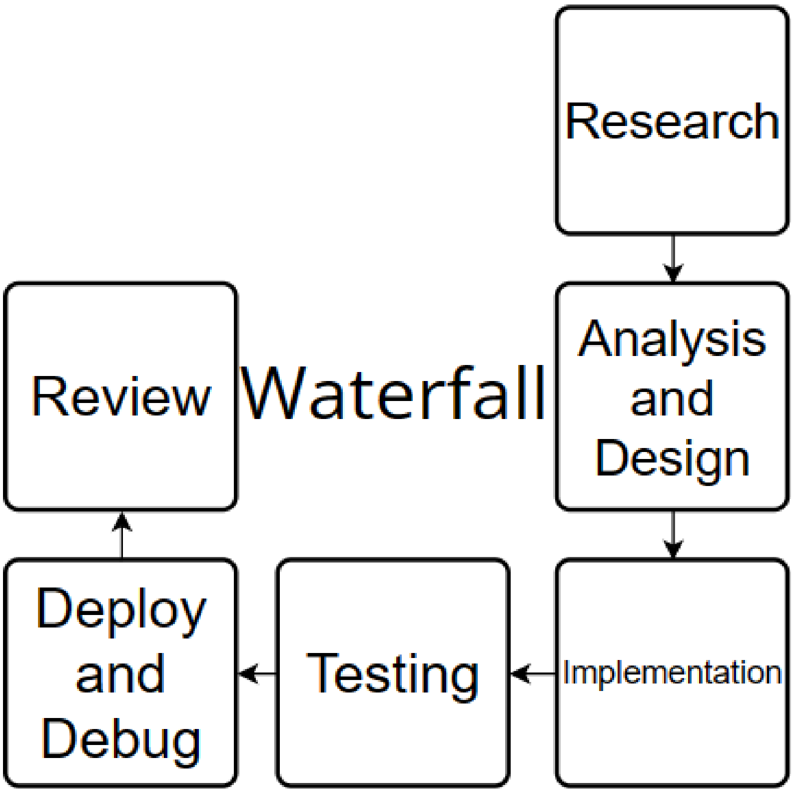


FIGURE 1.1: WATERFALL METHODOLOGY

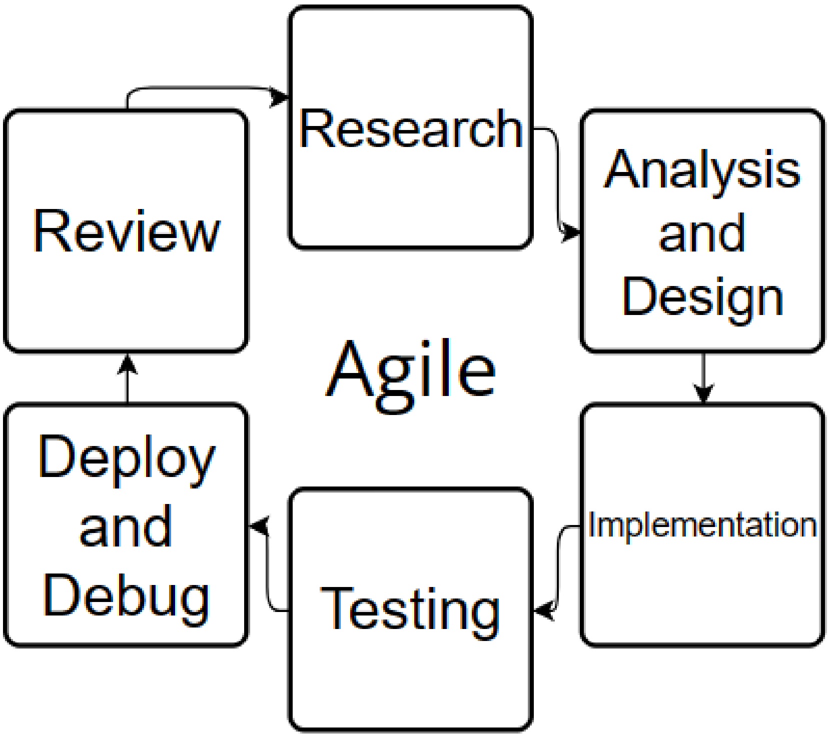


FIGURE 1.2: AGILE METHODOLOGY

With iterations made possible using Agile (as shown in Figure 1.2 compared to 1.1), efficiency could be achieved whilst making extensive modifications to the programs, and this notion aligned more with the Agile methodology, hence it was selected over the Waterfall methodology.

Following the Agile manifesto, 16 values were incorporated into the overall technical execution to accommodate every stage of software development, split by 4 key values and 12 additional values. While the 4 key values provided a robust management of this project ensuring progress towards completion is continuously pursued, abiding by 12 other values allowed the author to become more adaptable towards rapid project changes including adjusting all the project requirements based on a new research module.

Four key values

Individuals over tools and techniques: The supervisor, moderator and stakeholder had extensive domain knowledge in the fields of marketing and data, therefore their integral perspectives on the project quality mattered more than the technological edges of one powerful programming language researched by the author, which may have affected the project in the form of a disbenefit that the other individuals were aware of.

Functioning software over comprehensive documentation: This report documented every important detail of the program and its features, but none of it was as essential as a program that proved to satisfy the project requirements with its ability to retrieve data from various sources and collate them to form a structured database for marketeers without major functional errors.

Customer collaboration over contract negotiation: The author considered his supervisor, moderator, and stakeholder to be customers who could provide guidance in the right direction of the project execution. Through regular meetings in which constructive feedback was provided, the author responded by completing a task due in the upcoming meeting rather than trying to make a negotiation, which could have been adverse for the well-maintained relationship between the author and the customers.

Replying to change over following a plan: This key value prompted the author to select the Agile methodology over the Waterfall methodology as Agile offers a very flexible framework, allowing important adjustments to be made to program implementations without disrupting the project. Waterfall does not make adequate time for moving back to a previous SDLC phase, which meant researching and planning an additional upgrade for the programs after implementation was highly difficult, if not, impossible. Implementing more features for the programs without affecting the project flow would be smoother using Agile.

Twelve additional values

Satisfy customers through early, continuous advancement and delivery: With the feedback and new tasks assigned by customers after every meeting, the author delivered continuous research findings or an upgraded program at the next meeting, demonstrating good listening towards feedback and building confidence with the customers.

Embrace spontaneous requirements, including late ones in the project: The author acknowledged spontaneity to be part of the project and embraced rapid changes, some of which had to be made to adjust a data extraction program so it could get around its technical weaknesses, and some of which are explained in the Evaluation’s subsections.

Frequent value provision: Together with the first principle, the advanced research and programs did not only satisfy the customers but showcase valuable data extraction techniques that could be applied and upgraded in the future beyond this project.

Stepping out of the silos of the project: The author sought advice from the customers even outside scheduled meetings, admitting lacking an understanding of data extraction at times and proving the project could not be well-developed on his own.

Develop projects centring determined individuals: The author and customers may have had different external motivations for the completion of the project, but all of them aimed to have a functional company data extraction program by the end of the project schedule to support marketeers in their campaigns. This common goal prevented venturing into extraction of non-company data and deviating from the overall project aim.

Face-to-face communication: Although every customer meeting was held online using Microsoft Teams, computer cameras were used to facilitate face-to-face communication. The meetings proved to be effective for communicating feedback and understanding how each collaborator felt regarding progression of the project. For example, if the supervisor did not feel that conducted research or program implementation was sufficient, his face would show scepticism, and the author understood from the supervisor's expression that either more research needed to be done or the program needed to be modified.

Functioning software as the product: The main priority of this project was to deliver data extraction programs that could function as required. Extensive software development and testing was conducted with that priority in mind.

Work at a sustainable pace: The author of this project strove to work at a considerable pace that was not very fast or extremely occupied, to avoid suffering from a burnout. Taking median breaks between project continuation allowed the author to rest. Resting gave the author a refreshed mindset, and he could come up with upgrades for the programs afterwards. This balanced pace of working and recharging proved to be effective for the author.

Continuous excellence of project quality: Through presentations of program implementations, there were signs of excellence noted by the customers such as retrieving data and compiling the database within a short program running duration. If the programs were deemed to be excellent at a certain stage of the project schedule, future upgrades could maintain, if not, enhance the quality of the solutions as evaluations by the customers gave the author a clear indication of traits that made the programs excellent in quality.

Keep it simple: The data extraction methods of this project were intended to be built using a few lines of code. Excess programming could have resulted in a slower program and potentially, a database of lower quality in which company data is not structured tidily. In the end, only 3 methods of data extraction were tried and tested, all of which are elaborated in the Program Components subsection.

Be proactive as a team: The author would either resolve a technical issue independently or seek advice from the customers before having a scheduled meeting with them. For instance, one of the customers recommended the author another data source to enhance the database 5 days before a supervised meeting. In return, the author updated the programs following the feedback and reviewed the source. If there were any further problems with the data source, the author would have attempted to resolve it proactively before the supervised meeting as well, either by finding an alternate data source or edit the programs to bridge compatibility with the recommended source.

Reflection and adjustments to ways of working: With regular supervised meetings and customer feedback, there were multiple opportunities for the author to reflect on improvements that could be made for the project, such as research on extensive technologies to implement in the programs and how debugging needs to be exacted to fix a data extraction problem spotted by the customers.

3.2. Requirements

This subsection specifies the functional and non-functional requirements of the data extraction deliverables. These requirements made the technological objectives of the programs clear and concise for the author. Irrespective of the differentiating type of data extraction program by method, these overarching requirements accord the general scope of the project, so every programmable deliverable needed to fulfil them.

Functional requirements

Functional requirements determine the activities a user can perform with the data extraction programs. Splitting the functional requirements into 3 categories was appropriate for the marketing theme of the project and allowed more improvements to the functional quality of the deliverables through distinct parts of the programs.

Business requirements

The business requirements link to the overall aim of this project which was to provide marketeers with a database of company information to strengthen their campaigns and strategies. As many business background constraints are elaborated in the next subsection, the key requirements are summarised here.

- Structured company data needs to be available in a database.
- The data extraction techniques need to be implemented with the limited availability of technological resources.
- The deliverables need to accord ethical marketing principles.
- The data usage needs to be ethical.

User requirements

The user requirements detail the activities user can perform when the data extraction program is running. These include inputting some details to retrieve data from a specific source.

- User must input data source, which varies depending on the type of data extraction program selected.
- User must be able to create or load a database file.
- User must be able to exit the program manually.

System requirements

System requirements comprise the actions the data extraction programs perform themselves without human intervention. These include the automated structuring of company data after a user manually loads a data source.

- Data Extraction program must be opened after being selected on the user interface.
- Program must load one data source at a time.
- Program must extract data from the loaded source.
- Program must structure data based on user's (customers') specifications.
- Program must load structured data into the main database source.
- Program must create database source if it does not exist.
- Program must overwrite database source if it exists.
- Program must run (minimally on Windows operating systems for portability).

Non-functional requirements

The non-functional requirements comprise the constraints and technological abilities which could provide a great experience for the program user (Puzhevich 2021, para. 7). Like the general functional requirements, the deliverables need to fulfil all of them as none of these requirements are exclusive to a single type of data extraction program.

- Database file must have a sizeable capacity for storage of company data.

General acceptance criteria

These two short statements cement the overall conditions for the acceptable quality of the deliverables.

Given a computer program for data extraction,

When I use the program,

Then I can create a new database.

Given a computer program for data extraction and an existing database,

When I use the program,

Then I can update the existing database.

3.3. Associated Principles and Constraints

Data Life Cycle

This project accorded with the first four stages of the Data Life Cycle, a comprehensive framework selected due to its supervisory application of data in any project. They are:

Data generation: Data needs to be generated and made available for this project to commence. Across the World Wide Web, external parties involved in the use of company data have provided huge amounts of data, either publicly available or requiring premium subscription to access them. With the available sources online, no manual generation of data from the author was required.

Data collection: Data is then to be collected from the online sources by any technological means whilst paying attention to the associated principles and constraints to ensure that good practices of data usage are not overlooked. This is the quintessential step taken towards developing the database file deliverable as they determine what kind of information is going to be loaded and displayed in the database product. In this stage, the sources are to be loaded into the program as lines of code, then the bits of data can be accessed for the next data stage leading towards the database product.

Data processing: With this stage, the bits of extracted data will be assembled into a structured database according to the marketing schema. The schema itself is designed to fill the gaps of company information as intended in this project.

Data Storage: After the data extraction program fulfils its purpose to supplement the database and comes to a halt, company data remains stored in the database.

Data Ethics Framework

Constraints on the data extraction process were mainly based on the Data Ethics Framework, a written tool offered by the Central Digital & Data Office to ensure data is used sensibly. By following the framework, the author assures that the data used in the project were compliant with the legalities, and he did as much as possible to mitigate bias of any findings. The 3 principles of the Data Ethics Framework are also used in ethical marketing (Foster 2022).

Accountability: A principle ensuring that the project was under strict supervision and every progressive action taken to build the database did not go unnoticed. If the author performed anything transgressive intentionally or unintentionally, the supervisor, moderator and stakeholder were able to warn him in time to correct his mistakes. As the author was not well-versed in the field of data extraction prior to this project, having experts like Dr. Liberatore kept his actions in line with both ethical use of data and the objectives of the deliverable database. On a scale of 0-5 for accountability, 4 was given for this project. Although the author had a supervisor, moderator, and additional stakeholder to scrutinize his actions by reviewing this report and the deliverables, he was not under complete governance by the general public. No outsiders of the project influenced the author's actions.

Transparency: A principle to ensure that the project can be inspected publicly. As mentioned before, there were 3 individuals who reviewed the project completely, but the author wanted to make the project publicly accessible on the PATS (Project Allocation & Tracking System) website after the final submission and marking process. With full public disclosure, anyone can investigate the methods utilised in the deliverables, along with this report. Considering this decision to publicise the project on PATS, 5 was scaled for transparency.

Fairness: A principle to ensure that any biasness in the project is mitigated to prevent any unjust effects on the general public or distinctive social groups. As this project centres on company data, a potential bias would be building a database to contain fitness companies only. To maintain fairness in this project, the author only obtained data from free and open sources, as the data owners have given permission to use them. He also made the database contain information of companies with a diverse range of products apart from fitness services. However, as this project was based on existing data, a completely fair marketing coverage for companies across regions was not possible as some instances of company data were still unfound. 3 was scaled for fairness of data use.

7Ps

Apart from regulating the use of data, another few fundamental pointers needed to be accounted for to accommodate the entire marketing process from the start to the end. Through research, the 7Ps model was selected as it is an eclectic concept that covers the marketing fundamentals, rendering it a helpful framework. The author strove to apply the 7 Ps of marketing within database development. They are:

Product: There are free online data sources considered to be useful products. However, not all of them are customized for this project's purposes because some do not contain company data. Extracting data from sources like an online supermarket or travel booking system do not make sense as the data found on those sources include product details but not the details of the company itself.

Price: The pricing of the product influences the inclination of the customers. If the price of the product is high, then the quality of the product is expected to be high as well. The pricing reflects the company's image as well. Companies like Ferrari and Lamborghini produce very expensive cars, but they are still attractive to car enthusiasts due to the high quality and performance these companies offer. As this project was under an academic undertaking, along with premium data considered out of scope, this 'P' trait was not integrated into the final deliverables. The deliverables are not for sale as well.

Place: Whilst looking for online data sources, some of them possess details of a company's operational or base location, which can be extracted to supplement the project's database and enhance the quality of marketing information. The location of a business operation can be very beneficial for a company. As mentioned before, a company can sell products in a country where little to none of its competitors are operating.

Promotion: With the final compilation of the deliverables for this project, the author promoted the data extraction programs to the customers, showcasing the functions and how it benefits marketeers and their tactics.

People: As the supervisor, moderator and stakeholder assisted in ensuring that the project is both on schedule and legally compliant with the use of data, they were technically part of the development process, monitoring the author's progress on building the program and making sure that the deliverables are suitable for marketing purposes.

Process: In this project, programming was a key task of implementation to build the data extraction programs. Poor programming would lead to poor programs which may not extract company data properly, and the final deliverables may not be of any value to a marketeer. Paying attention to the process of programming, including tasks of code refactoring and debugging, was important to ensure a database solution of excellent quality.

Physical evidence: If a marketeer was to access the database of this project, the descriptive schema would be displayed on the UI (User Interface) and give the marketeer a great idea of the list of company details that can be learned about.

Seven more ethical marketing principles

There are 10 ethical principles of marketing (Foster 2022), though only 7 of the principles are elaborated below, as the rest have already been as principles of the Data Ethics Framework.

Honesty: Data supplemented into the database file were not fabricated or falsified by any methods, warranting marketing activities to be based on accurate company details. Slight adjustments to the data include shifting certain facts to a relevant attributed column.

Social responsibility: As web scraping, if misused or implemented improperly, could disrupt a website's services and prevent consumers from using them, the amount of programming done on the web scrapers was minimal, and attempts to scrape a website were made to extract data respectfully.

Privacy: As the data fetched online have been declared to be public, mutual consent has been given to share specific company data. Some attributes of the final data source of the prototype database were based on Big Picture's dataset, which does not contain sensitive data and demonstrates awareness of data privacy. With the submitted deliverables, no sensitive data has been revealed.

Sustainability: Computer hardware usage was minimal throughout the project, so energy consumption and CO_2 (Carbon Dioxide) emissions in the proximity of the project's occurrence was reduced. The author only used his laptop to write the data extraction programs, and occasionally used library desktop computers to conduct further research at hourly rates and avoid overusing his laptop. Using the laptop too much would degrade its performance faster and affect the overall maintenance of energy.

Respect for customers: No illicit actions were made with customers involved in the project or any individuals outside the project scope.

Data collation transparency: Any data collections by the companies for marketing purposes have to be informed to the customers or website users, and vice-versa. When a user registers on the website, there is a 'Terms and Conditions' or 'Privacy Policy' section to detail the user on how their data is going to be used whilst aligning with the GDPR. Big Picture, which was the main source of data collection for the database and its schema design, required the author to register on the website and declare his download of their datasets.

Compliance with legalities: Along with the GDPR, there were other laws the author complied with, such as the Open Data Commons Attribute License to access Big Picture's dataset. The datasets used to supplement the database were extracted from open and free online sources, therefore no payments were made to comply with any legalities of data usage, though the American Yellowpages web scraper, Apify, was a freemium solution used to scrape a very small amount of data from the Yellowpages website for free.

3.4. Program Components

This subsection details how the programs were developed to accustom the project requirements specified previously, along with how the data was extracted accordingly. The components planned to be implemented in the data extraction programs are expounded in detail, along with some supplementary computing theories surrounding each part of the final technological deliverables.

Data sources

Due to the principles of honesty and fairness, no dummy data was created to augment a database as marketeers would be using either falsified information or information that could lead up to a discriminatory bias or fraudulent practice in analytics. All the data was to be retrieved from existing online data sources while subject to 3 primary general conditions.

1. Access to the data source was free with no need for monetary actions. No amount of money was spent for this project.

2. The data source contains relevant information about companies.
3. The materials formats needed to be compatible with the data extraction programs.

Methods of data extraction

The data extraction process is the means of collecting data. As stated before in the second stage of the Data Life Cycle, these collection methods are quintessential towards construction of the database. The quality of the structured database heavily relies on the quality of the data, which can only be determined through data extraction. Though research led to many options to extract data, only 3 methods were actualized in this project. Not only were these 3 methods considered to be among tested data extraction procedures (Misra 2024), but the author was able to grasp them the fastest within the limited duration of the project schedule.

1. Manual extraction

Some websites offer their own datasets, which had to be manually downloaded in a file format suitable for the computer programs, such as CSV (Comma-Separated Value) for a Microsoft Excel database file. Attempting to bypass the download method in any other form would violate the website's terms of use. After downloading the datasets onto the laptop, a computer program can be written to further extractions from the datasets and compile the data in a structured format that would match the final database schema (See Main database design in the Program Design subsection) to make the database look comprehensive for a marketeer.

2. Web scraping

Web scraping was the second method used to retrieve data from some websites which permit the action in accordance with their terms of use. With a web scraping program, the program would look through a web page's source and extract the HTML (HyperText Markup Language) code. The HTML code would then be searched for company data located within tags, like the company's name which could be found in a 'h1' HTML tag, for instance. Once bits of company data are found within a tag, it is appended to a database file.

3. API integration.

API (Application Programming Interface) integration was the third and last method used to retrieve data from some websites. It was an additional method combined with web scraping to bypass anti-scraping features on websites. To use an API and enhance the web scraping programs, a freemium API key was added as a line of code.

Bringing the data together

With the data sources and extraction methods, all of them need to be compiled together into a single database file for easy comprehension. More data sources would make the total information of a company as varied as possible, and a single file reduces time needed to select a digital tool. The structuring method designed to compile the data into a common format and match the main database's schema can be found in the Program Implementation subsection under 'Selected data sources'.

3.5. Program Design

This subsection shows the visual concept of the programs designed with the use of Visual Paradigm, a UX/UI tool that was free to use and had options for a flowchart. The flowcharts show what each of the computer programs were intended to perform.

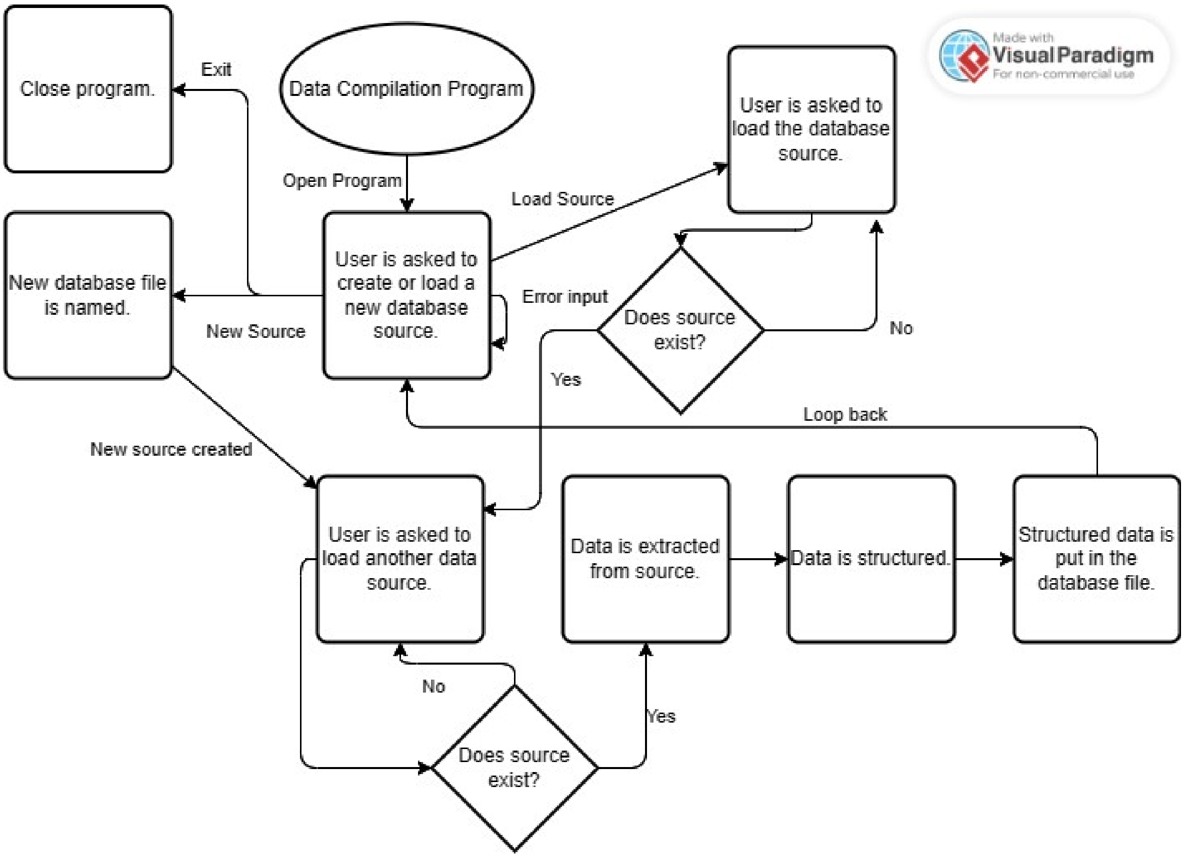


FIGURE 2: FLOWCHART FOR MANUAL DATA EXTRACTION AND COMPILATION

After downloading multiple datasets, they could be extracted further and combined with a single written computer program to save time, hence the file is aliased as a data compilation program. The looping allows a user to continuously find a dataset that exists before appending data from that source into either a new database file the user wants to create or an existing database file.

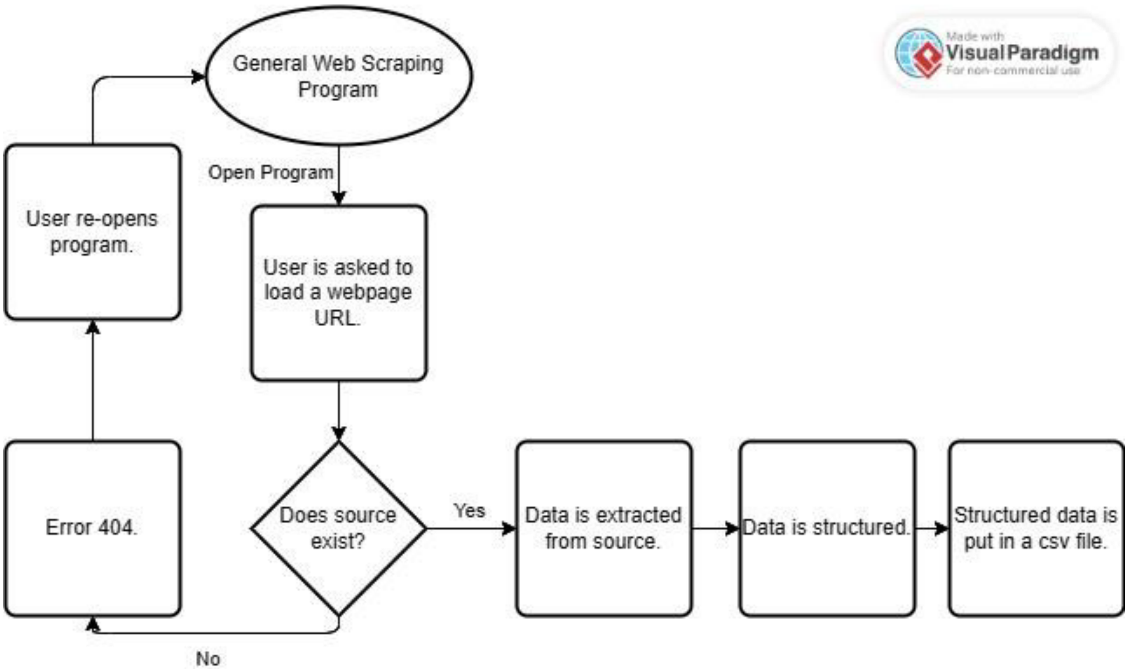


FIGURE 3: FLOWCHART FOR WEB SCRAPING AND COMPILATION

Data will be scraped using a website link. The link leads to a page containing the web data. After scraping the data, it will undergo several structuring processes to suit the schema of the primary database file. An initial file would be created to store every successful data extract of a single web scraping attempt. This initial file can then be used in the manual extractor/compiler program to append the scraped data to the primary database. The 'Error 404.' and 'User re-opens program.' encourage a user after finding out a link does not exist to immediately restart the program and resend a URL (Uniform Resource Locator).

Main database design

Main database schema

(handle, name, website, phone, industry, size, type, founded, address, city, state, post_code, country_code, rating, hours)

SCREENSHOT 1: FINAL SCHEMA WITH 15 ATTRIBUTES

The final database schema was based on the attributes of multiple data sources integrated in this project. The author used Big Picture's free dataset as a starting point, adding every attribute in the free edition to the final schema design. Though the premium edition contained more attributes and therefore more company details, there were free data sources outside Big Picture that could serve as an alternative data supplement for the additional attributed columns (See 'Selected data sources' for details of used data sources) and provide a comprehensive dataset more like the premium edition.

With the data extraction programs, this database included additional attributes inspired by other data sources that offer extensive information to cover more aspects of a company, such as the specific address of one of the company's offices which can be found on Yelp or Yell.

This final database schema, however, is also considered the biggest limiter. The computer programs were hardcoded to create any new database file and structure all company data from used sources according to the fixed schema. If you wanted to modify the schema by altering or adding more attributes, you would have to reprogram the files themselves to develop a database with your new schema concept. Despite this, the author wanted to show that this pre-coded schema could still be a comprehensive tool for marketing exercises with the varying angles of information it conveys to a marketeer. An example of applying the database schema in various marketing tactics is shown in test F3 of the 'Final test phase' subsection.

3.6. Program Implementation

This subsection expounds the technologies used to develop the program.

Selected data sources

After conducting intensive research and finding multiple data sources, those stated below were used for the project and experimented with the data extraction programs for further reasons.

Big Picture: This is an online organization that provides very basic but essential details of company data similarly to Datarade. As explained in the previous subsection, the dataset from this company was used as both the initial concept schema and data appending start point for the database deliverable. All the attributes from Big Picture's schema were used in the main database schema as every detail was useful as a starter kit for a marketing activity. A company's name, location, number of employees and type of industry are a few of the attributes that a marketeer can begin with to know more about a company's ingredients for successful business operations. To download a dataset from Big Picture, only registration on the website needs to be done. Then access to the dataset is given.

Companies House: It is a government-owned agency offering public datasets of a good general overview of company information, particularly from the United Kingdom. As the schema in the datasets closely matches Big Picture's schema, along with providing more details like a lengthier description of a company's address (location), this data source was the second pick for data extraction. Unlike Big Picture, the datasets can be downloaded directly from the website without account registration.

Yell (short for Yellowpages): It is a website that offers information for each company with a specific URL (Uniform Resource Locator) to locate useful company data sources. For instance, searching for Italian restaurants in Cardiff and extracting relevant instances of restaurant information necessitates the URL 'https://www.yell.com/ucs/UcsSearchAction.do?scrambleSeed=931839746&keywords=italian+restaurants&location=cardiff'. With this URL, you can find all the restaurants and use a filter or sorter to determine your marketing needs like finding out about the top 5 highly rated restaurants by customer reviews. You could also find the webpage of a single restaurant from the list, like Giovanni's Restaurant located in The Hayes, Cardiff with the URL 'https://www.yell.com/biz/giovanni-s-restaurant-cardiff-241344/', to extract data of a single company.

Yell is considered to be one of the biggest and most well-known business directories on the Internet for its comprehensive company data. As the data is publicly ready, there was no need for any ethical approval. For this website, web scraping was needed to retrieve the data on the webpages.

Yelp: It offers information similarly to Yellowpages, in the form of listed companies according to specific criterion and that web scraping was the method for data extraction. That means a URL is required as well. To find the best restaurants near or in Cardiff, use the URL 'https://www.yelp.co.uk/search?find_desc=Restaurants&find_loc=Cardiff'. Yelp is another big business directory on the Internet. Yelp's datasets allow researchers to find industries that are leading the market, so an effective product marketing strategy can be backed up with data from Yelp.

As Yelp's and Yell's webpage layouts differ by the styling and wording of the HTML tags, two separate web scraping programs were built to retrieve data from those sources based on their distinctive designs. Both websites do not offer the same amount of company details as either Big Picture or Companies House, so the details extracted from Yelp and Yell would instead fill up the remaining gaps of company information missing from the former two data sources.

Programming & Platforms

After exploring various programming languages to perform web scraping, Python was deemed the most suitable option for this project. Finding documentations to learn Python and build a web scraping functionality was easy. To add up, Python could extract data at a quick pace with minimal expense (ProWebScraper 2023). As the 'CSV' module was used to perform all the compilations, each of the datasets that were downloaded manually needed to be converted or already be in the '.csv' format. Although it may sound tedious, using the module in the manual compilation program was tested to be quick with several subsampled datasets.

PyCharm IDE Community Edition is a free and comprehensive platform that allows programming in the Python language. With it, experiments were done to test the programs regularly and gauge certain factors, such as:

- How fast the program runs.
- How much data can be retrieved.
- How effective is the compilation of the data.
- Any bugs that occur during each test.

PyCharm offers a good amount of storage for additional libraries needed to install to enhance the program. As PyCharm was utilized as an experimental sandbox, the author learned about functional errors which could be corrected prior to deployment on the CLI (Command Line Interface). This tactic proved to be efficient as less time was wasted on testing if PyCharm was used; you could debug and rerun the program on one platform.

MySQL and Microsoft Excel were the only two database storing options evaluated throughout the project. Microsoft Excel was selected as the final database storing platform for all of the compiled data. MySQL had a greater capacity for data storage. However, learning to implement MySQL upgrades in the Python files to strengthen the data compilation process was not straightforward. This, along with failed tests shown in the Evaluation section, led to a decision to switch to Microsoft Excel. The author had more experience working with Microsoft Excel, therefore he learned how to implement upgrades faster in the Python files and had more ideas to modify the data structuring to accommodate the differing datasets using Excel.

With the file, scraped data can be compiled and displayed as structured data on an Excel spreadsheet. The user-friendly interface of Excel also makes it easier for a marketeer to read the company details.

The computer programs

Ultimately, 6 Python files were developed throughout the project. Instead of relying on just one programmable file, the author wanted several files to allow adding instances of company information from a different type of source. A single program can be slow if there is a lot of code in it. Having more than one program reduced the amount of code in each file, and each program offers a quick running duration.

For descriptions/instructions on how to use the different program files, some of them may overlap due to similar functionality, but are still explained to describe how each program file works on its own. All the programs were named in lowercasing for String convenience. Having some uppercasing would require a little more attention to typing detail when a user wants to use a program using the CLI. The programs were categorized by data extraction methods.

1. Manual compiler programs

The manual extraction/compiler programs were the base programs implemented to build a new database file or update an existing one. Having both features would allow a user to create several instances of database files to accommodate more than one marketing strategy that would require a specific list of company information. Here is a list of general functions found in most programs.

New function: Every program will have a ‘New’ function to create a new database file.

Load function: Every program will also have a 'Load’ function, but this will require the name of an existing database file.

Select source: After selecting either ‘New’ or ‘Load’, a user selects which data source to extract from to supplement a new database or modify an existing one using Python input.

Select extra source: Allow a user to select another data source to extract data to be appended to the main source. Every extra dataset would undergo different versions of automated data structuring to fit the schema of the main database, ending with more company information added to a single file.

Looping: The functions above could be repeated as many times as desired until the user wants to close the program.

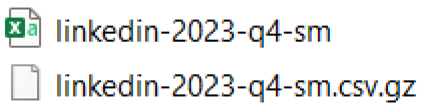
The primary constraint is that only one database file can be opened or modified at a time through the new/load function. To append data, multiple data sources can be used but, like the database, only one can be accessed to append data to the main database file.

List of manual compiler programs:

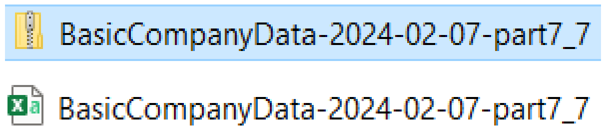
sample1appendtest.py: This prototype program was built to test for a successful connection between any data extraction/compiler program and any database and table in MySQL. It contains the string 'sample1' as the name of the targeted MySQL database table in the author's device was 'sample1'. With this program, data could be appended from any data source, though the 'sampleCategories.csv' file was hardcoded to try out appending data from that source only, as the appending test was carried out following a previous web scraping test with the 'webscrapesampler.py' file (See webscrapesampler.py). With minimal functionality evaluations, it does not have majority of the general functions stated above.

manualcompiler.py: This program was built to extract and compile data from multiple files into a single database file with the general manual compiler functions stated above, along with the utilization of MySQL, the first choice for a database storing platform. This and the above program need MySQL to be installed in your device or any hardware you would like to test the programs on and modify the code to change the host of the database; the program was hardcoded to connect to the database stored on the author's device. This program offered a password security feature to protect the database. Microsoft Excel cannot store as much data as MySQL, so the latter's scalability allows more company data to be stored with this program.

manualcompilerexcel.py: This program was built to extract and compile data into a single database file as well with the utilization of Microsoft Excel, the final choice for a database storing platform. With Excel, it was less troublesome to implement the compilation process, as no login action was needed to connect to a MySQL database at the expense of security, and no SQL queries had to be specified. With successful initial tests on this program, more data structuring techniques were implemented, making this program quite different from the above programs.



SCREENSHOT 2.1: NAMING CONVENTION FOR BIG PICTURE DATASETS



SCREENSHOT 2.2: NAMING CONVENTION FOR COMPANIES HOUSE DATASETS

Each structuring technique was based on the name of the file. With filenames that were standardized by their data sources, such as filenames containing 'linkedin' for Big Picture datasets and 'BasicCompanyData' for Companies House datasets, the string found in the filename was used to pick a distinct attribute structuring for each file. The cells of data would be arranged based on the inserted data source's schema to match the main database's schema, as stated in the 'Select extra source' function description above.

As mentioned in the implemented data sources, Yell and Yelp could only be used to append extra data therefore files with string 'yell' or 'yelp' are to be inputted if you select 'Load' for an existing database and the former files as the extra source.

2. Web scraping programs

For these programs which utilize the method of web scraping, data will be extracted from a website populated with business data like Yell and Yelp. Instead of using filenames as data sources, a URL of a webpage is to be inserted. For flexibility and recommended efficiency, users can copy the URL of a website he/she wants to scrape and paste it in the input line of the CLI. Once the web scraper retrieves the company data, it is then parsed, structured, and put in a Microsoft Excel file. Preparing the file to contain the scraped data before using the manual compiler to append the data to the main database file aligned with the program implementation goals of running speed.

The biggest challenge of web scraping was to bypass anti-bot scraping mechanisms. Data sources requiring the method of web scraping have set up defensive actions to prevent bots from scraping huge amounts of data at a rapid pace. API integration, non-looping and delayed scraping were chosen attempts to bypass those security features.

Input URL: The user must input a URL into the CLI. The fastest way to do it is by copying and pasting the URL link.

Request data: With a URL inputted by the user, the programs send a request for the webpage's information.

Parse data: Arrange the retrieved information in a format to allow company data to be extracted by specified HTML tags.

Compile data (on its own, separately from the manual compiler program): Assemble the parsed data into a 'csv' file suitable for the manual compilation program. Compiling the data in one program before using another reduces the probability of a program crash and improves the speed of running either program to provide an efficient experience for the user.

List of web scraping programs:

webscrapesampler.py: This was the simplest version of a web scraper. It was developed to try out the web scraping method on a webpage without any anti-bot scraping security features ('Books to Scrape') and compile the data into a comprehensive format without any additional upgrades. If this program failed to scrape any data, the author would not be able to develop upgraded versions of a web scraper, all of which are listed next. Theoretically, any web scraping upgrades could be implemented in this program, but this prototype was minimally coded to prove web scraping some useful data was feasible. More time was available for deeper research into marketing and database technologies for the other files below, which contain more complex code.

yellscrape.py: As there are a lot of pages with company information, scraping excess information could trigger security features, such as bot detection and lead to a blocked attempt. This file was customized to allow users to scrape data from a single webpage of Yell, as scraping from one source is faster and less prone to getting blocked by the anti-bot scraping features. The HTML tags hardcoded in the file match the tags the bits of company data were wrapped around on the Yell webpage.

yelpscrape.py: Like Yell, Yelp has a lot of pages with information, so scraping several pages using the general program could trigger anti-bot security features as well. This file's lines of code are tailored to scrape data according to Yelp's source format, which may use different tags and classes for its content. Like the previous file, it is only capable of extracting data from a single page to avoid being blocked.

Apify

The author tried to use a ready-made freemium web scraper online called Apify. It allowed him to bypass the bot detection techniques effectively and had a preset format to extract company data from certain HTML tags. Although a successful evaluation was done in which data was extracted using Apify and appended into the main database file using the manual compiler, the company data was revealed to be based in USA (United States of America), which was not advised for this project as the customers preferred company data from the UK. Nevertheless, this Apify evaluation directed the author to develop manual web scraping programs to extract company data based on the data's attributes from the freemium solution's successfully scraped dataset.

Abandoned implementations

Prior to evaluation tests, there were some technologies researched on and considered for the implementations. However, they were discarded before full implementation due to technical disbenefits, which would impede the quality of the final deliverables.

MongoDB: This is a very scalable and robust platform to develop and maintain databases. However, it's non-relational feature made it a less popular option, even among the customers, as the feature defeats the intended structuring component of the company data. With the aim of making a database comprehensive with relational attributes alluded to a marketing strategy, MongoDB was an impractical option.

SQLite: SQLite was the second choice to develop and maintain databases after MongoDB was scrapped. It was slightly better to learn SQLite due to its relational feature, using SQL to perform relational appending of company data. DB Browser, an open-source tool that is built on SQLite, was initially installed for the project. However, upon finding out about MySQL provides a distinct feature which aligned better with the project, SQLite was scrapped as well, and that feature is scalability (VishwashVishwakarma 2023). MySQL can house a large volume of company data. With many data sources and chunks of data, SQLite was not considered effective for large-scale storage.

Proxy rotation technique with Smartproxy: To bypass anti-bot scraping features, proxy rotation could be done with Smartproxy, which offers multiple proxies, and you would use each one alternatively to evade the defensive mechanism. Although subscription is required, the cost was relatively cheap and reasonable for a 3-month period of the project. After being advised to stay out of premium subscriptions along with premium data, the author opted out of this bypass attempt. There were some alternate free proxy rotation solutions online, but they were suspicious and less reliable than premium proxy solutions like Smartproxy.

4. Evaluation

Evaluations of the programs were split in several phases to document development progress throughout the project schedule. This section showed efforts made in timely stages to attain certain milestones as drafted in the initial plan or additional goals as discussed in meetings with the supervisor and stakeholder. All test subsections below contain formatted test cases based on a free template (Eby 2019), with a few modifications. The priorities in each test case followed a numeric order set by Browser Stack, with 1 indicating the most important tests, and ascending numbers decrease the priority.

4.1. Basic test phase

The first phase of testing evaluated the data extraction programs with the most basic functionality required to retrieve data. This included being able to access a (CSV) file using the program and scraping data from a web page on the World Wide Web. Tests shown below were stated to prove a successful implementation of a data extraction method at the lowest sufficient level. Without the success of these tests, it would be meaningless to think of any program upgrades. This phase was conducted during weeks 6 and 7.

<table>
<tr><th colspan="2">Test Title</th><th colspan="2">Priority</th><th>Test Case ID</th><th>Test Number</th></tr>
<tr><td colspan="2">Sample web scraping with non-commercial website.</td><td colspan="2">1</td><td>B1</td><td>1</td></tr>
<tr><th colspan="2">Test Description</th><th colspan="2">Test Designed and Executed By</th><th colspan="2">Execution Date</th></tr>
<tr><td colspan="2">Scrape non-company data from a website to test web scraping feature.</td><td colspan="2">Md Md Farid</td><td colspan="2">14 February 2024</td></tr>
<tr><th colspan="2">Test Dependencies</th><th colspan="2">Test Conditions</th><th colspan="2">Test Control</th></tr>
<tr><td colspan="2">- ‘Books to Scrape’ website.
- Microsoft Excel application.
- Sample web scraping program (webscrapesampler.py).</td><td colspan="2">- Single web page specified by URL.</td><td colspan="2">- Sample web scraping program tested on Command Prompt terminal.</td></tr>
<tr><th>Step ID</th><th>Step Description</th><th>Expected Result</th><th>Actual Result</th><th>Pass/Fail</th><th>Remarks</th></tr>
<tr><td>B1S1</td><td>Input a URL for a web page.</td><td>The web page is scrutinized.</td><td>The web page is scrutinized as expected.</td><td>Pass</td><td>Quick tutorial of web scraping allowed test to be executed early before implementation week 4.</td></tr>
<tr><td>B1S2</td><td>Content parsing for extraction.</td><td>The content on the ‘Books to Scrape’ web page are parsed for data extraction.</td><td>The content on the ‘Books to Scrape’ web page are parsed as expected.</td><td>Pass</td><td>Evident on the Command Prompt terminal, the HTML lines displayed show the test executor what can be extracted from a web page.</td></tr>
<tr><td>B1S3</td><td>Content structuring.</td><td>Structured extracted content into a database file.</td><td>Structured extracted content shown in a database file.</td><td>Pass</td><td>Evident on a file titled ‘sampleCategories.csv’ as content is put in attributed columns.</td></tr>
</table>

Comment: With the successful web scrape, this tested method can be replicated or modified in more case-specific web scraping programs. For instance, the method's code can be edited slightly to accustom a web scrape attempt from Yell's webpage.

<table>
<tr><th colspan="2">Test Title</th><th colspan="2">Priority</th><th>Test Case ID</th><th>Test Number</th></tr>
<tr><td colspan="2">Basic data appending with MySQL.</td><td colspan="2">1</td><td>B2</td><td>2</td></tr>
<tr><th colspan="2">Test Description</th><th colspan="2">Test Designed and Executed By</th><th colspan="2">Execution Date</th></tr>
<tr><td colspan="2">Append scraped data from sample data source to the MySQL database table using a MySQL data appending program and prove a connection between any computer program and MYSQL is successful.</td><td colspan="2">Md Md Farid</td><td colspan="2">13 March 2024</td></tr>
<tr><th colspan="2">Test Dependencies</th><th colspan="2">Test Conditions</th><th colspan="2">Test Control</th></tr>
<tr><td colspan="2">- Existing database (samples) and table (sample1) in MySQL Client.
- Manual MySQL data appending program (sample1appendtest.py) is available.
- MySQL application.
- 'sampleCategories.csv' file.</td><td colspan="2">- Connect to a database specified using the MySQL data appending program.
- MySQL login details are valid.</td><td colspan="2">- Only one database connection can be made.
- Connection tested on Python program using Command Prompt.
- MySQL Client terminal used to evaluate database result.</td></tr>
<tr><th>Step ID</th><th>Step Description</th><th>Expected Result</th><th>Actual Result</th><th>Pass/Fail</th><th>Remarks</th></tr>
<tr><td>B2S1</td><td>Login to MySQL.</td><td>An automatic data append follows.</td><td>An automatic data append follows as expected.</td><td>Pass</td><td></td></tr>
<tr><td>B2S2</td><td>Open MySQL Client to find database table with appended data.</td><td>Database table shows appended data.</td><td>Database table shows appended data as expected.</td><td>Pass</td><td></td></tr>
</table>

Comment: With this successful connection between MySQL and a computer program, you could simply retrieve data from any source and add it into a MySQL database using just one custom program instead of having to type multiple MySQL queries, saving time taken to complete a tedious task and compile data from many sources.

As stated in the 'webscrapesampler.py' description, more time was spent on research, thus explaining the lengthy gap between the two basic tests' execution dates. Considering the milestone of developing a prototype before 25 March 2024, this was achieved with one data extraction (web scraping) and compilation method with MySQL.

4.2. Mid test phase

The Mid phase of testing was only feasible with 3 available weeks during the Easter break to conduct the tests. These tests allowed the author to experiment with any upgrades to the basic features tested during the Basic test.

<table>
<tr><th colspan="2">Test Title</th><th colspan="2">Priority</th><th>Test Case ID</th><th>Test Number</th></tr>
<tr><td colspan="2">Manual MySQL database table creation.</td><td colspan="2">2</td><td>M1</td><td>3</td></tr>
<tr><th colspan="2">Test Description</th><th colspan="2">Test Designed and Executed By</th><th colspan="2">Execution Date</th></tr>
<tr><td colspan="2">Creating new data source using MySQL data extraction file.</td><td colspan="2">Md Md Farid</td><td colspan="2">25 March 2024</td></tr>
<tr><th colspan="2">Test Dependencies</th><th colspan="2">Test Conditions</th><th colspan="2">Test Control</th></tr>
<tr><td colspan="2">- Primary data source with schema of company information to standardize MySQL database table.
- MySQL application.
- 'manualcompiler.py' program.</td><td colspan="2">- Using schema based on Big Picture's dataset to standardize MySQL database table.
- MySQL login details are valid.</td><td colspan="2">- Python program tested on Command Prompt terminal.
- MySQL Client terminal used to evaluate database result.</td></tr>
<tr><th>Step ID</th><th>Step Description</th><th>Expected Result</th><th>Actual Result</th><th>Pass/Fail</th><th>Remarks</th></tr>
<tr><td>M1S1</td><td>Choose to create a data source.</td><td>Program asks for a new name for the main data source.</td><td>Program asks for a new name for the main data source as expected.</td><td>Pass</td><td></td></tr>
<tr><td>M1S2</td><td>Login to MySQL.</td><td>Schema is created.</td><td>Schema is created as expected.</td><td>Pass</td><td>No error shown on Command Prompt terminal, so step</td></tr>
</table>

					was presumed to have passed. The schema was hardcoded.
M1S3	Open MySQL Client to find database table with schema.	After successful login on MySQL Client, database table shows schema.	After successful login on MySQL Client, database table shows schema as expected.	Pass	Shown on MySQL Client terminal.

Test Title		Priority		Test Case ID	Test Number
Manual MySQL data append data.		2		M2	4
Test Description		Test Designed and Executed By		Execution Date	
Append data from an additional data source into the main database table using MySQL data extraction file.		Md Md Farid		25 March 2024	
Test Dependencies		Test Conditions		Test Control	
- Main database with table containing the primary schema on MySQL Client. - Additional data source with instances of company information. - MySQL application. - ‘manualcompiler.py’ program.		- Using only one additional data source to append data into the MySQL database table. - MySQL login details are valid.		- MySQL database and table is created. - Use only a small, subsampled amount of data from the additional data source. - Python program tested on Command Prompt terminal. - MySQL Client terminal used to evaluate database result.	
Step ID	Step Description	Expected Result	Actual Result	Pass/Fail	Remarks

M2S1	Choose to load a database.	Program asks to load a table.	Program asks to load a table as expected.	Pass	
M2S2	Choose to load a table from the database.	Program asks to load a database table.	Program asks to load an additional data source as expected.	Pass	
M2S3	Select an additional data source.	User is asked to enter login details.	User is asked to enter login details as expected.	Pass	
M2S4	Login to MySQL.	After successful login on MySQL data extraction program, data append is performed.	After successful login on MySQL data extraction program, data append performs as expected.	Pass	On the Command Prompt terminal which ran the Python program, no error was shown regarding a data append, so it was presumed to have passed.
M2S5	Open MySQL Client to find database table with appended data.	After successful login on MySQL Client, database table shows appended data as new instance rows.	After successful login on MySQL Client, database table did not contain any appended data as expected.	Fail	Due to this evident fail on a different terminal (MySQL Client), the author switched to Excel as the database platform.

<table>
<tr><td colspan="2">Test Title</td><td colspan="2">Priority</td><td>Test Case ID</td><td>Test Number</td></tr>
<tr><td colspan="2">Database creation with Excel.</td><td colspan="2">1</td><td>M3</td><td>5</td></tr>
<tr><td colspan="2">Test Description</td><td colspan="2">Test Designed and Executed By</td><td colspan="2">Execution Date</td></tr>
<tr><td colspan="2">Create new database file using Excel.</td><td colspan="2">Md Md Farid</td><td colspan="2">25 March 2024</td></tr>
<tr><td colspan="2">Test Dependencies</td><td colspan="2">Test Conditions</td><td colspan="2">Test Control</td></tr>
<tr><td colspan="2">- A data source with instances of company information and the intended schema.
- Microsoft Excel application.
- ‘manualcompilerexcel.py’ program.</td><td colspan="2">Database creation with a data source.</td><td colspan="2">- Use only a small, subsampled amount of data from the main database file.
- Python program tested on Command Prompt.</td></tr>
<tr><td>Step ID</td><td>Step Description</td><td>Expected Result</td><td>Actual Result</td><td>Pass/Fail</td><td>Remarks</td></tr>
<tr><td>M3S1</td><td>Choose to create a new data source.</td><td>Program asks for a main data source name.</td><td>Program asks for a main data source name as expected.</td><td>Pass</td><td></td></tr>
<tr><td>M3S2</td><td>Select a data source.</td><td>Program creates a database file.</td><td>Program creates a database file as expected.</td><td>Pass</td><td>File is found on the File Explorer application on the tested device.</td></tr>
</table>

Note: As evident in the appendix, every newly created database’s schema matches the schema of Big Picture’s dataset.

<table>
<tr><td>Test Title</td><td>Priority</td><td>Test Case ID</td><td>Test Number</td></tr>
<tr><td>Append data with Excel.</td><td>1</td><td>M4</td><td>6</td></tr>
<tr><td>Test Description</td><td>Test Designed and Executed By</td><td colspan="2">Execution Date</td></tr>
<tr><td>Append data from an additional data source into the main database file using Excel.</td><td>Md Md Farid</td><td colspan="2">25 March 2024</td></tr>
<tr><td>Test Dependencies</td><td>Test Conditions</td><td colspan="2">Test Control</td></tr>
</table>

- Main database file with schema. - Additional data source with instances of company information. - Microsoft Excel application. - ‘manualcompilerexcel.py’ program.		Using only one additional data source to append data into the database file.		- Use only a small, subsampled amount of data from the main database file. - Use only a small, subsampled amount of data from the additional data source. - Python program tested on Command Prompt.	
Step ID	**Step Description**	**Expected Result**	**Actual Result**	**Pass/Fail**	**Remarks**
M4S1	Choose to load a data source.	Program asks to load a file.	Program asks to load a file as expected.	Pass	
M4S2	Select an existing database file.	Program asks to load an additional data source.	Program asks to load an additional data source as expected.	Pass	
M4S3	Select an additional data source.	Database file shows appended data put on additional rows in the spreadsheet.	Database file shows appended data put on additional rows in the spreadsheet as expected.	Pass	File is evidenced using Microsoft Excel.

Test Title	Priority	Test Case ID	Test Number
Basic company database different instance structuring.	2	M5	7
Test Description	**Test Designed and Executed By**	**Execution Date**	
Structure information instances extracted from an additional data source and	Md Md Farid	26 March 2024	

<table>
<tr><td colspan="2">appended to the main database file according to its schema.</td><td colspan="2"></td><td colspan="2"></td></tr>
<tr><th colspan="2">Test Dependencies</th><th colspan="2">Test Conditions</th><th colspan="2">Test Control</th></tr>
<tr><td colspan="2">- Main database file with schema.
- Big Picture sample instances of company information (subSample1.csv).
- Companies House sample instances of company information (companiesHouseSubSample.csv).
- Algorithms to match data based on the most suitable attribute from the schema.
- Microsoft Excel application.
- ‘manualcompilerexcel.py’ program.</td><td colspan="2">- Structuring of information with a basic attribute matching technique.</td><td colspan="2">- Use only a small, subsampled amount of data from the main database file.
- Use only a small, subsampled amount of data from the additional data source.
- Python program tested on Command Prompt.</td></tr>
<tr><th>Step ID</th><th>Step Description</th><th>Expected Result</th><th>Actual Result</th><th>Pass/Fail</th><th>Remarks</th></tr>
<tr><td>M5S1</td><td>Choose to load a data source.</td><td>Program asks to load a file.</td><td>Program asks to load a file as expected.</td><td>Pass</td><td></td></tr>
<tr><td>M5S2</td><td>Select an existing database file.</td><td>Program asks to load an additional data source.</td><td>Program asks to load an additional data source as expected.</td><td>Pass</td><td></td></tr>
<tr><td>M5S3</td><td>Select an additional data source.</td><td>Database file shows appended data structured according to its schema.</td><td>Database file shows appended data structured according to its schema as expected.</td><td>Pass</td><td>File is evidenced using Microsoft Excel.</td></tr>
</table>

<table>
<tr><th colspan="2">Test Title</th><th colspan="2">Priority</th><th>Test Case ID</th><th>Test Number</th></tr>
<tr><td colspan="2">Sample web scraping with Apify.</td><td colspan="2">3</td><td>M6</td><td>8</td></tr>
<tr><th colspan="2">Test Description</th><th colspan="2">Test Designed and Executed By</th><th colspan="2">Execution Date</th></tr>
<tr><td colspan="2">Scrape company data from a website using Apify.</td><td colspan="2">Md Md Farid</td><td colspan="2">8 April 2024</td></tr>
<tr><th colspan="2">Test Dependencies</th><th colspan="2">Test Conditions</th><th colspan="2">Test Control</th></tr>
<tr><td colspan="2">- Yellowpages website.
- Yellow Pages Scraper actor.
- Microsoft Excel application.</td><td colspan="2">- Single web page specified by URL.
- Only 3 instances of company data to be extracted.</td><td colspan="2">- Run on the Apify console.</td></tr>
<tr><th>Step ID</th><th>Step Description</th><th>Expected Result</th><th>Actual Result</th><th>Pass/Fail</th><th>Remarks</th></tr>
<tr><td>M6S1</td><td>Input details for the Yellow Pages Scraper actor.</td><td>UK Company data is retrieved based on details.</td><td>Company data is retrieved, though the companies are based in California, USA, not Cardiff, UK.</td><td>Fail</td><td>‘Cardiff’ was put as a ‘Location’ to extract company data from that area, but the country UK was not specified.</td></tr>
<tr><td>M6S2</td><td>Content structuring.</td><td>Structured extracted content into a database file in Excel.</td><td>Structured extracted content shown in a database file in Excel.</td><td>Pass</td><td>Evident on a file titled ‘dataset_yellow-pages-scraper_2024-04-08_04-00-20-842’ as content is put in attributed columns.</td></tr>
</table>

Comment: Despite having company data originating from the USA, the dataset provided gave the author an idea of some attributes that could be put for the final database schema. This test was useful and thus, documented. If someone else wanted to extract company data from the American Yellowpages webpage, Apify is a great freemium solution.

4.3. Final test phase

The final conducted phase of testing, starting from week 10 and reaching the end of the project schedule. This phase was intended to evaluate the programs nearing the end of the schedule and noting what was missed out on for the project.

<table>
<tr><th colspan="2">Test Title</th><th colspan="2">Priority</th><th>Test Case ID</th><th>Test Number</th></tr>
<tr><td colspan="2">Yell Scrape 1.</td><td colspan="2">2</td><td>F1</td><td>9</td></tr>
<tr><th colspan="2">Test Description</th><th colspan="2">Test Designed and Executed By</th><th colspan="2">Execution Date</th></tr>
<tr><td colspan="2">Scrape company data from Yell using a manual program.</td><td colspan="2">Md Md Farid</td><td colspan="2">23 April 2024</td></tr>
<tr><th colspan="2">Test Dependencies</th><th colspan="2">Test Conditions</th><th colspan="2">Test Control</th></tr>
<tr><td colspan="2">- Yell website.
- ‘yellscrape.py’ program.
- Microsoft Excel application.</td><td colspan="2">- Single web page specified by URL.
- Single company data extract.
- Only 3 attributes of company data to be extracted (name, rating and company number).</td><td colspan="2">- Python program tested on Command Prompt.
- Company chosen for web scraping was ‘I Say Supported Living Services Ltd’ as the tester had existing data of the company, and this scrape could add more company details.</td></tr>
<tr><th>Step ID</th><th>Step Description</th><th>Expected Result</th><th>Actual Result</th><th>Pass/Fail</th><th>Remarks</th></tr>
<tr><td>F1S1</td><td>Input details for the Yell Scraper actor.</td><td>UK Company data is retrieved based on input.</td><td>Company data was not retrieved.</td><td>Fail</td><td>Error 403 was received. The web scrape attempt was blocked by anti-bot scraping measures.</td></tr>
</table>

<table>
<tr><th>Test Title</th><th>Priority</th><th>Test Case ID</th><th>Test Number</th></tr>
<tr><td>Yell Scrape 2.</td><td>2</td><td>F2</td><td>10</td></tr>
<tr><th>Test Description</th><th>Test Designed and Executed By</th><th colspan="2">Execution Date</th></tr>
</table>

Scrape company data from Yell using a manual program but with an API key.		Md Md Farid		23 April 2024	
Test Dependencies		Test Conditions		Test Control	
- Yell website. - 'yellscrape.py' program. - Microsoft Excel application. - API key.		- Single web page specified by URL. - Single company data extract. - Only 3 attributes of company data to be extracted (name, rating and company number).		- Python program tested on Command Prompt. - Company chosen for web scraping was 'I Say Supported Living Services Ltd' as the tester had existing data of the company, and this scrape could add more company details.	
Step ID	Step Description	Expected Result	Actual Result	Pass/Fail	Remarks
F2S1	Input details for the Yell Scraper actor.	UK Company data is retrieved based on input.	Company data was not retrieved.	Fail	Error 500 was received. The web scrape attempt returned with no data extracts. There was a hidden issue with the web scrape.

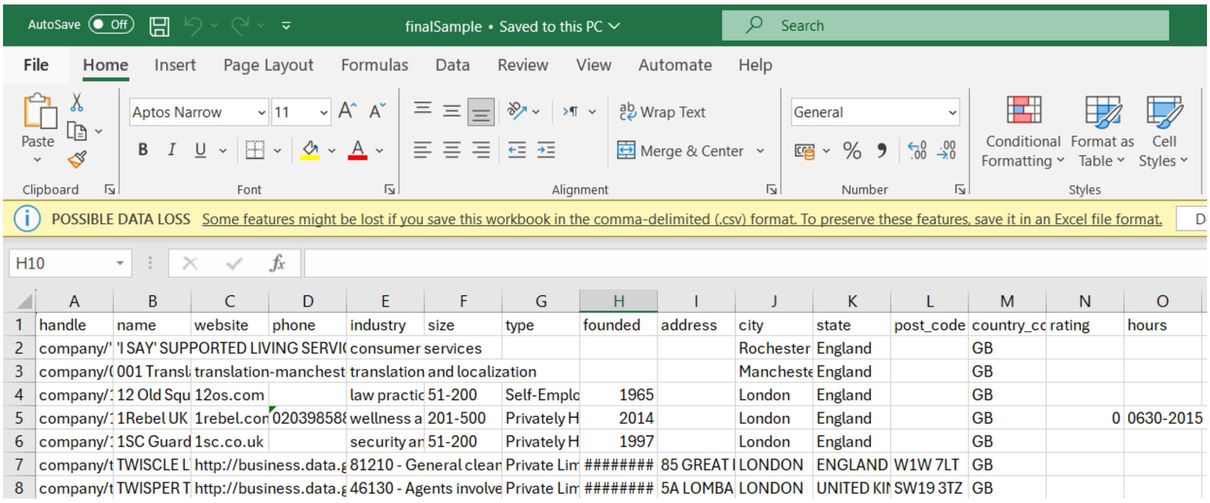
Comment: Both tests F1 and F2 used the same 'Yellscrape.py' program, differing by just the use of an integrated API key. However, both tests failed and witnessed by the supervisor and stakeholder on 23 April 2024, and the remaining duration left of the project was too short for the author to fix this data extraction method.

Test Title	Priority	Test Case ID	Test Number
Final test of data compilation.	1	F3	11
Test Description	Test Designed and Executed By	Execution Date	

<table>
<tr><td colspan="2">Structure information instances extracted from 3 data sources and append them to the main database file continuously.</td><td colspan="2">Md Md Farid</td><td colspan="2">7 May 2024</td></tr>
<tr><th colspan="2">Test Dependencies</th><th colspan="2">Test Conditions</th><th colspan="2">Test Control</th></tr>
<tr><td colspan="2">- Latest database schema of 15 attributes.
- Big Picture sample instances of company information (linkedinSubSample.csv).
- Companies House sample instances of company information (BasicCompanyDataSubSample.csv).
- Single manually inputted Yell UK company information (yellSingleExtract.csv) matching website https://www.yell.com/biz/1rebel-london-10130941/.
- Multiple algorithms to match data based on the most suitable attributes from the schema.
- Microsoft Excel application.
- ‘manualcompilerexcel.py’ program.</td><td colspan="2">- Structuring of information with basic attribute matching techniques, differing by filename.</td><td colspan="2">- Use only a small, subsampled amount of data from the main database file.
- Use only a small, subsampled amount of data from the additional data sources.
- Python program tested on Command Prompt.</td></tr>
<tr><th>Step ID</th><th>Step Description</th><th>Expected Result</th><th>Actual Result</th><th>Pass/Fail</th><th>Remarks</th></tr>
<tr><td>F3S1</td><td>Choose to create a new data source.</td><td>Program asks for a main data source name.</td><td>Program asks for a main data source name as expected.</td><td>Pass</td><td></td></tr>
<tr><td>F3S2</td><td>Select a data source.</td><td>Program creates a database file.</td><td>Program creates a database file as expected.</td><td>Pass</td><td>File is found on the File Explorer application on the tested device.</td></tr>
<tr><td>F3S3</td><td>Loop and append data from another source.</td><td>Program asks for another data source.</td><td>Program asks for another data source as expected.</td><td>Pass</td><td></td></tr>
</table>

F3S4	Select another data source.	Program appends data to main database file.	Program appends data to main database file as expected.	Pass	
F3S5	Loop and append data from another source.	Program asks for another data source.	Program asks for another data source as expected.	Pass	
F3S6	Select another data source.	Program appends data to main database file.	Program appends data to main database file as expected.	Pass	
F3S7	Exit program.	Program shuts down.	Program shuts down as expected.	Pass	

Final test comment: The company data from Yell UK was manually typed into the 'csv' file following the URL that contained the same details. Manual downloading of datasets and compiling them with 'manualcompilerexcel.py' was the only functional option to satisfy the acceptable criterion.



SCREENSHOT 3: FINALSAMPLE.CSV

This was the result of integrating company data from 3 sources. With 15 attributes, a marketeer could set some strategies, such as starting up a company outside London since majority of the companies shown in the screenshot are based there or keeping a company as private limited like the companies shown.

5. Conclusion

The biggest challenge of this project was the search for data sources that were not only relevant for marketing exercises, but also contained extensive attributes to cover a wide range of company details, such as the number of employees and the total turnover. Majority of the data sources, if not, all of them had narrow angles of company information. There were more websites found by research, but not enough time was available to customize the computer programs to structure their company data according to the primary database schema.

As the deliverables fulfilled the overall aim of the project with only a few methods, there was not a huge depth of data extraction and compilation. With several ideas explained in this section, upgrading the computer programs and enlarging the technological scope of data extraction and compilation could be achieved in the future.

Future data sources

The table below contains website names of additional data sources and what kind of company information they offered, but they were never explored to develop a data extraction program to collect data from these sources.

Website name	Data that could be found on this source	Method of data extraction
Yahoo Finance	Public datasets of company financial data.	Web scraping.
Nasdaq	More diverse financial datasets which could be appended to all instances of company information in a database.	Manual extraction.
Annual Reports	Data found in reports that could be downloaded as PDF (Portable Document Format) files.	Manual download and Document scraping.
Open Corporates	Global company registers with URL links to each company register where you could find company data from a particular country apart from the United Kingdom.	Varying according to the country the company register is based in.
ICAEW	Online research guide highlighting online sources for reports other than Annual Reports.	Varying according to each online source but Document scraping is theorized to be the primary data extraction method.

OECD	OECD offers a public dataset for the level of GDP per capita, and productivity categorized by countries. On the database, this could be matched with the country each company is based in.	Manual extraction.
Overview International Database	This website offers more URLs to company data productivity sources.	Varying according to source but Manual extraction is theorized to be the primary data extraction method.
The Conference Board	This website provides datasets containing economic data, such as the global economic indicator, which could also be matched with each company by country.	Manual extraction.
DataCite Commons	This online repository offers basic details of companies that may not be registered with Companies House. However, loading the data for retrieval from this website takes a very long duration, so it may not be ideal for data extraction.	Manual extraction.

If data could be retrieved from premium sources or websites listed above, the database would not only be sharper in terms of information quality but look more wonderful for a marketeer to explore. A larger variety of data could give them a cutting edge for marketing campaigns. Even with API integration, which is a great technique for large-scale data extraction, the author could not extract a single instance of company data due to being blocked. His only successful attempt to perform web scraping was with Apify. There may be the figured possibility of extracting at least one, if not, copious amounts of data from the World Wide Web in a future implementation with both API integration and avoiding a blocked scrape.

As the programs' respective functions were developed based on the data sources' formats online, such as having similar column attributes or a consistent HTML format for a web page, flexibility was intended to be achieved for the data extraction process but reached only as far as it could accommodate the retrieved data. Modifying company information from Companies House's dataset to match Big Picture's dataset based on the most similar attributes was only possible due to the consistencies of the dataset filenames. Companies House has more than one dataset with the exact schema, and an algorithm was customized to fit their file's specification.

In the future, someone could contrive a means of collating data with less, if not, no limitations based on the format of the data sources. Furthering from the previous example, appending data without restricting the feature to just 'csv' files would be a big step forward in developing a comprehensive data structuring platform. A file converter which can go through 2 or more different spreadsheets and structure them identically would be ideal for the future of this project's aim, as compared to the author's programs which could only go through a single data source of the 'csv' format.

A technique only picked up by research but not actualized was an alternate type of scraping called document scraping, as mentioned in the data sources table above. This technique extracts data from a document and structures it in a format depending on the specification by the program developer. After the 3 tried data extraction methods, this could be a very useful technique to find more relevant company data that is only found in written reports.

For advanced research, a technique could be developed to segment information in the database further to accommodate very specific marketing campaign types. The database itself contains a fairly comprehensive range of company details but lacks the feature of filtering data such as company location and type of service to suit a product-based marketing strategy, for example. With a database marketing type filter, marketeers could retrieve only relevant details to be more exact in their strategies. From the database itself, it only gives a general overview of company details.

6. Project Reflection

Starting from week 2, the author conducted heavy research on the marketing field, not anticipating any negative impact on his progress until discovering that his amount of research for the database technologies and data extraction fundamentals was insufficient. The author knew his research was imbalanced because too much attention was paid to the marketing research compared to learning about data essentials and technologies. The author did not research marketing and data equally, and it meant he had to gather more details for database implementation on the same week in which the author had to start designing the conceptual database model to fit extracted data.

To improve on future research quality, the author needs to equalize his focus on two or more fields together. He can set a limited amount of targeted information from multiple fields. If the author has written two paragraphs of marketing ethics, he can write two paragraphs of data ethics for balance.

At the end of week 4, the author realized he skipped the crucial SDLC stage of identifying and laying out the program requirements, which was technically meant to be done prior to the data model design stage. Like his early research, there was too much focus on the conceptual design and functional flow of the programs, which led to ignoring the requirements that had to be set first. Ultimately, if the author had followed the SDLC properly and not skipped a stage, he could have laid more requirements than what was finalized. More upgrades could have been implemented into the data extraction programs as well.

During week 6, the author performed evaluations on his program designs and considered potential database platforms that could store the data. However, he was not confident in selecting a platform, and the lack of decision delayed upgrades implementation by a week. This was a hard lesson for him in terms of confidence and being decisive. Having both traits would save up time and give room for multiple experiments. A possible experiment could be carried out to give MongoDB a morning trial and if it failed, SQLite could be given an afternoon trial and Microsoft Excel could have an evening trial if the second platform failed as well. This concept of deciding on 3 trials within a day is better than abstractedly wondering which of these technologies provided the best output without starting a single trial, the latter of which just leads to procrastination.

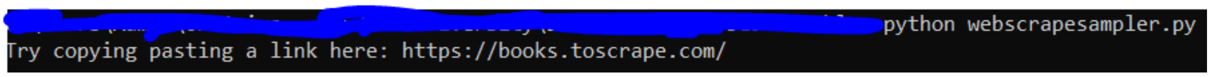
In contrast to the non-desirable experiences, the author found it easy to implement the first version of a web-scraping program because of his prior experience with Python from previous projects, along with a certain understanding of database systems in the form of an Excel spreadsheet. Reflecting on it, the author aims to be more open with learning new technologies even with no set goals in mind, as they may come in handy when a new challenge emerges and requires them.

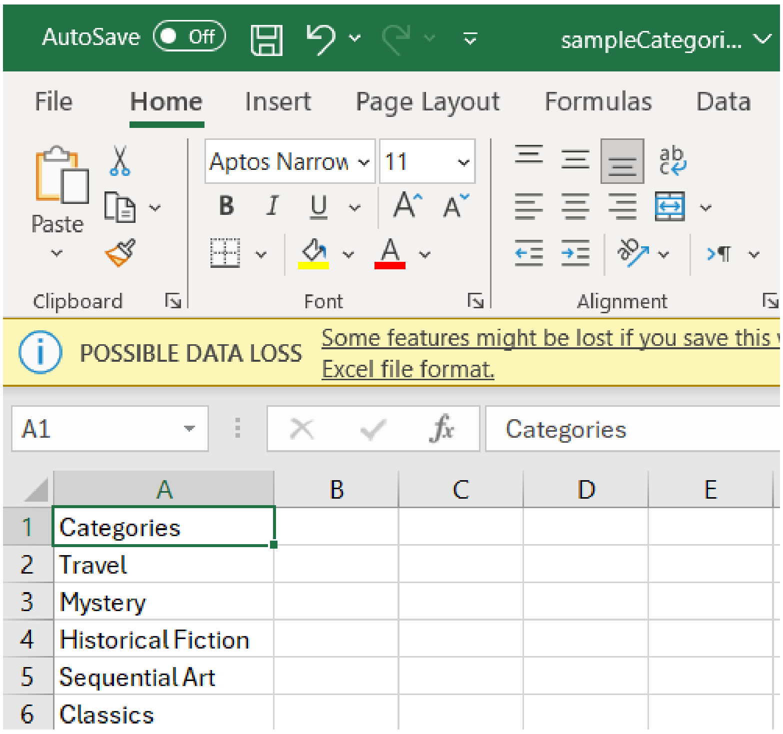
The author learned that going for the most optimum theoretical solution would not necessarily fit every situation, especially for database structuring. Thinking MySQL was better due to its scalability incentive and attempting to optimize the deliverables with it led to a hindrance in progress. At that time, data appending with MySQL failed and the author could not think of a debug with the reduced amount of time on schedule. So, he immediately switched to Microsoft Excel and settled on finalizing the project deliverables with this technology. The final decision on a database storing platform was appropriate since deliverables could be submitted with no major technical errors.

In one feedback session, the supervisor requested the author to gather more data sources, which caused him to panic as he had to perform an extra task apart from planned implementations for the computer programs. Although worried about setting aside time to do more work and progress would not be made on time with the project schedule, the author learned to trust his customers even more, acknowledging that they were the ones with more experience and would evaluate the project submissions. Even with the extra task, it turned out that the author could complete more tasks than he thought. The extra efforts led to a database with a greater quantity of company information.

In retrospect, sustainability was only achievable in this project because the author did not need to use a lot of computer hardware to create the deliverables. If many of the ideas mentioned above require computers with larger storage or performance capabilities, the contrasting effects on an environment with CO_2 emissions from more energy consumption need to be considered.

7. Appendices





SCREENSHOT GROUP A: TEST B1 (DATA SCRAPED AND COMPILED IN A DATABASE FILE)

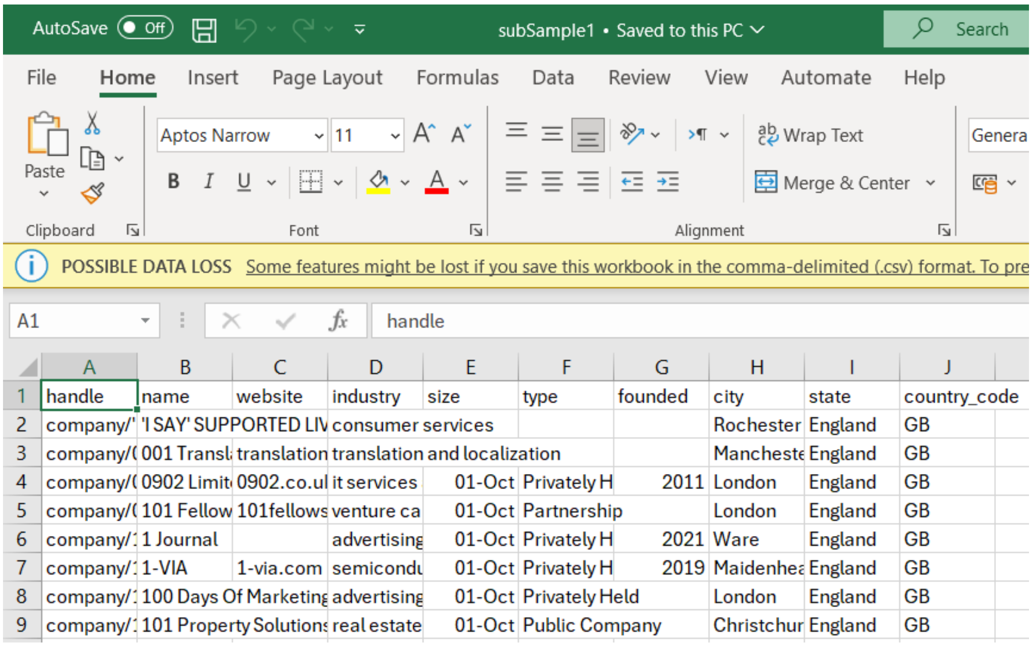
```
mysql> SELECT * FROM sample1;
Empty set (0.00 sec)

mysql> desc sample1;
+--------------+--------------+------+-----+---------+-------+
| Field        | Type         | Null | Key | Default | Extra |
+--------------+--------------+------+-----+---------+-------+
| PersonID     | int          | YES  |     | NULL    |       |
| bookCategory | varchar(255) | YES  |     | NULL    |       |
+--------------+--------------+------+-----+---------+-------+
2 rows in set (0.00 sec)
```

```
[redacted]>python sample1appendtest.py
Enter username![redacted]
Enter password!
```

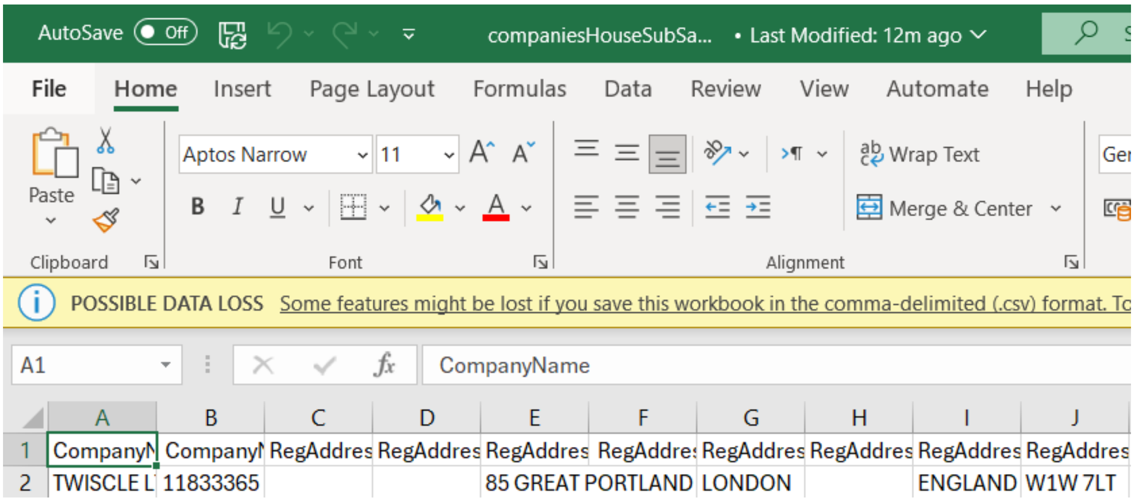
```
mysql> SELECT * FROM sample1;
+----------+-------------------------+
| PersonID | bookCategory            |
+----------+-------------------------+
|        1 | ['Travel']              |
|        2 | ['Mystery']             |
|        3 | ['Historical Fiction']  |
```

SCREENSHOT GROUP B: TEST B2 (APPENDING THE SCRAPED DATA INTO A MYSQL DATABASE TABLE)



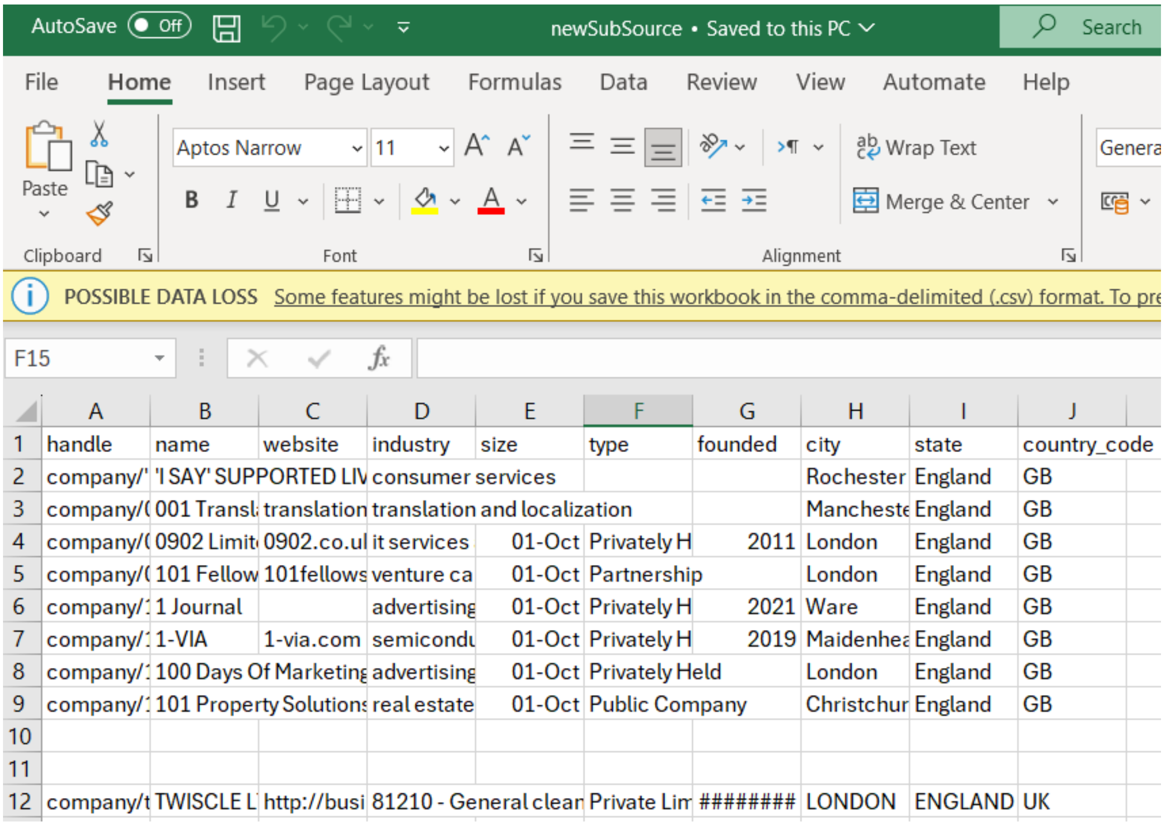
	A	B	C	D	E	F	G	H	I	J
1	handle	name	website	industry	size	type	founded	city	state	country_code
2	company/'	'I SAY' SUPPORTED LIV		consumer services				Rochester	England	GB
3	company/(	001 Transl	translation	translation and localization				Mancheste	England	GB
4	company/(	0902 Limit	0902.co.ul	it services	01-Oct	Privately H	2011	London	England	GB
5	company/(	101 Fellow	101fellows	venture ca	01-Oct	Partnership		London	England	GB
6	company/1	1 Journal		advertising	01-Oct	Privately H	2021	Ware	England	GB
7	company/1	1-VIA	1-via.com	semicondu	01-Oct	Privately H	2019	Maidenhea	England	GB
8	company/1	100 Days Of Marketing		advertising	01-Oct	Privately Held		London	England	GB
9	company/1	101 Property Solutions		real estate	01-Oct	Public Company		Christchur	England	GB

```
[redacted]python manualcompilerexcel.py
Type 'new' to create a new database source or type 'load' to load an existing source. new
Type new name of file. No need to type extension as csv format is defaulted. newSubSource
Enter data source filename to append company data: subSample1.csv
```



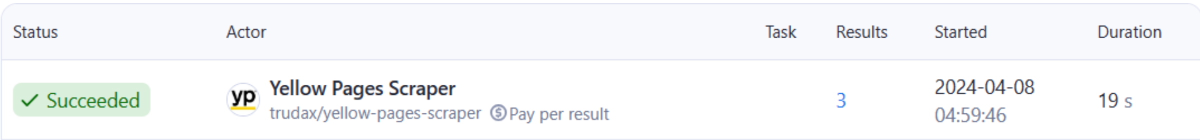
	A	B	C	D	E	F	G	H	I	J
1	CompanyN	Company	RegAddres	RegAddres	RegAddres	RegAddre:	RegAddres	RegAddres	RegAddres	RegAddres
2	TWISCLE L	11833365			85 GREAT PORTLAND		LONDON		ENGLAND	W1W 7LT

```
[redacted]python manualcompilerexcel.py
Type 'new' to create a new database source or type 'load' to load an existing source. newSubSource.csv
Invalid choice typed. Enter 'new' to create a new database source or type 'load'to load an existing source.
Type 'new' to create a new database source or type 'load' to load an existing source. load
Type name of existing csv file: newSubSource.csv
Enter data source filename to append company data: companiesHouseSubSample.csv
```

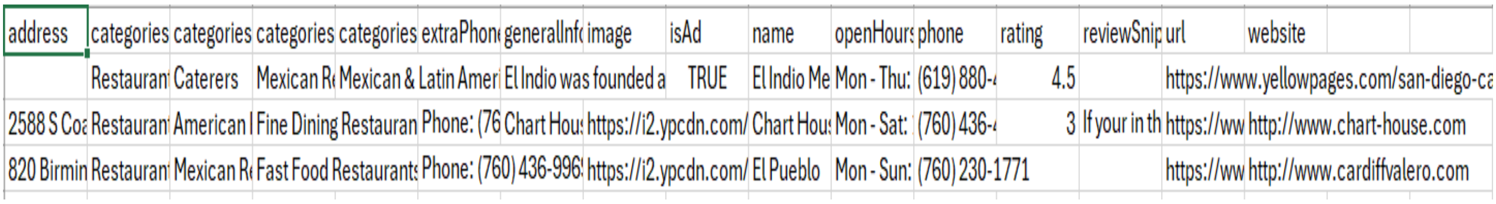


	A	B	C	D	E	F	G	H	I	J
1	handle	name	website	industry	size	type	founded	city	state	country_code
2	company/'	'I SAY' SUPPORTED LIV		consumer services				Rochester	England	GB
3	company/(	001 Transl	translation	translation and localization				Mancheste	England	GB
4	company/(	0902 Limit	0902.co.ul	it services	01-Oct	Privately H	2011	London	England	GB
5	company/(	101 Fellow	101fellows	venture ca	01-Oct	Partnership		London	England	GB
6	company/1	1 Journal		advertising	01-Oct	Privately H	2021	Ware	England	GB
7	company/1	1-VIA	1-via.com	semicondu	01-Oct	Privately H	2019	Maidenhea	England	GB
8	company/1	100 Days Of Marketing		advertising	01-Oct	Privately Held		London	England	GB
9	company/1	101 Property Solutions		real estate	01-Oct	Public Company		Christchur	England	GB
10										
11										
12	company/t	TWISCLE L	http://busi	81210 - General clean		Private Lim	########	LONDON	ENGLAND	UK

SCREENSHOT GROUP C: TEST M5 (COMPILING COMPANY DATA FROM MULTIPLE SOURCES FOLLOWING AN INITIAL SCHEMA OF 10 ATTRIBUTES)

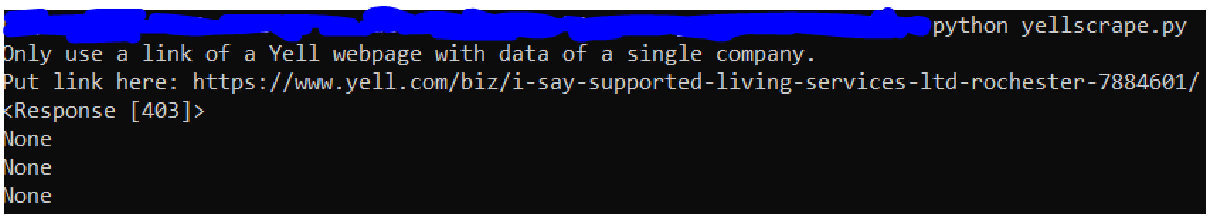


Status	Actor	Task	Results	Started	Duration
Succeeded	Yellow Pages Scraper trudax/yellow-pages-scraper Pay per result		3	2024-04-08 04:59:46	19 s

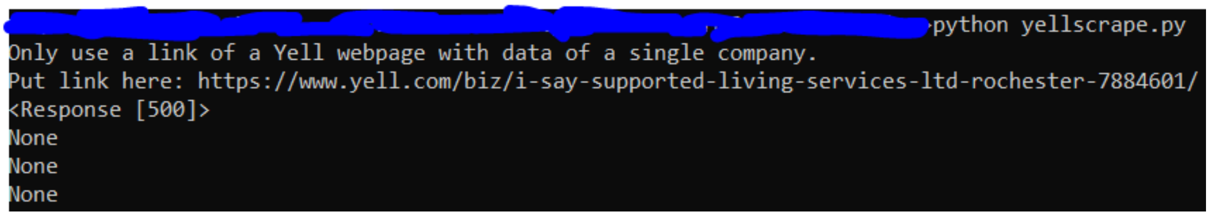


address	categories	categories	categories	categories	extraPhone	generalInfo	image	isAd	name	openHours	phone	rating	reviewSnip	url	website
	Restaurant	Caterers	Mexican R	Mexican & Latin Ameri		El Indio was founded a		TRUE	El Indio Me	Mon - Thu:	(619) 880-4	4.5		https://www.yellowpages.com/san-diego-ca	
2588 S Coa	Restaurant	American I	Fine Dining Restauran		Phone: (76	Chart Hous	https://i2.ypcdn.com/		Chart Hous	Mon - Sat:	(760) 436-4	3	If your in th	https://ww	http://www.chart-house.com
820 Birmin	Restaurant	Mexican R	Fast Food Restaurants		Phone: (760) 436-996		https://i2.ypcdn.com/		El Pueblo	Mon - Sun:	(760) 230-1771			https://ww	http://www.cardiffvalero.com

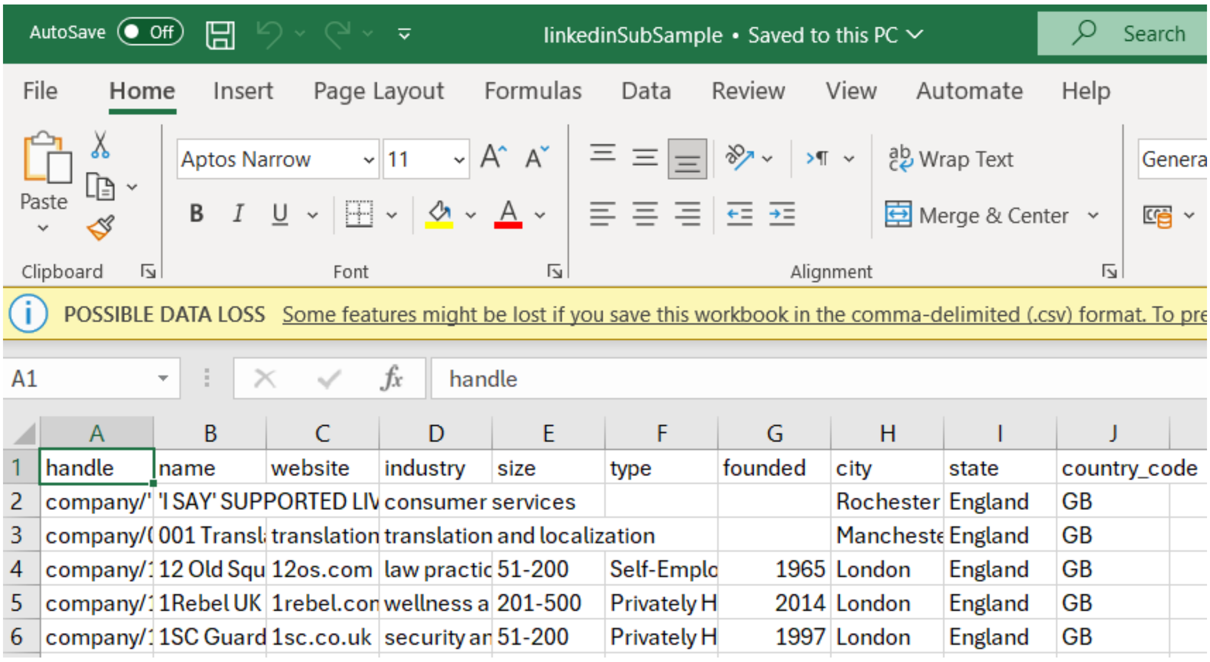
SCREENSHOT GROUP D: TEST M6 (SUCCESSFUL APIFY WEB SCRAPE ON YELLOW PAGES USA WITH INSPIRATIONAL ATTRIBUTES FOR THE FINAL SCHEMA)

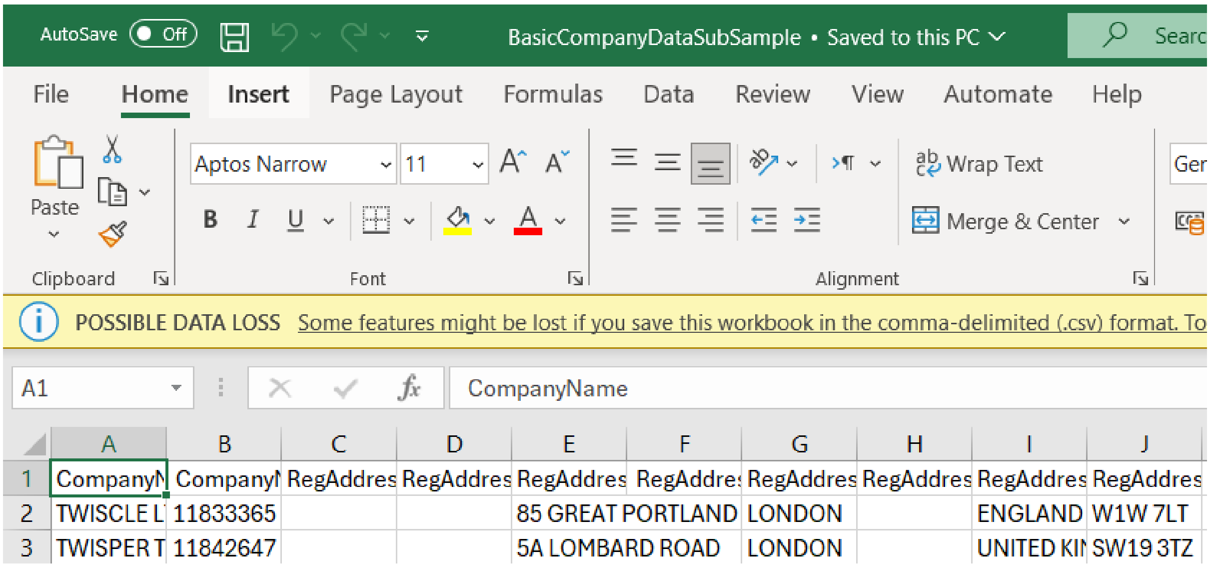


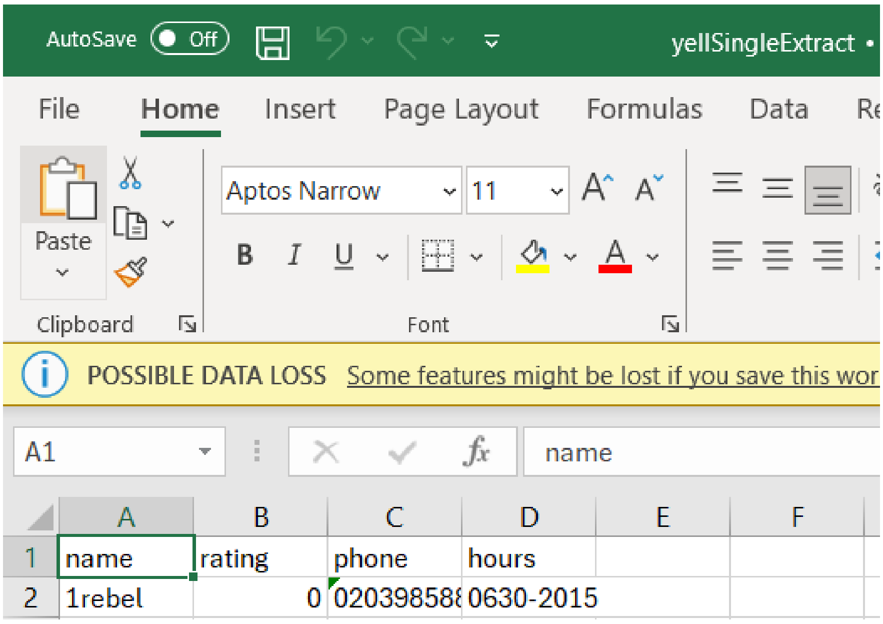
```
[redacted]python yellscrape.py
Only use a link of a Yell webpage with data of a single company.
Put link here: https://www.yell.com/biz/i-say-supported-living-services-ltd-rochester-7884601/
<Response [403]>
None
None
None
```

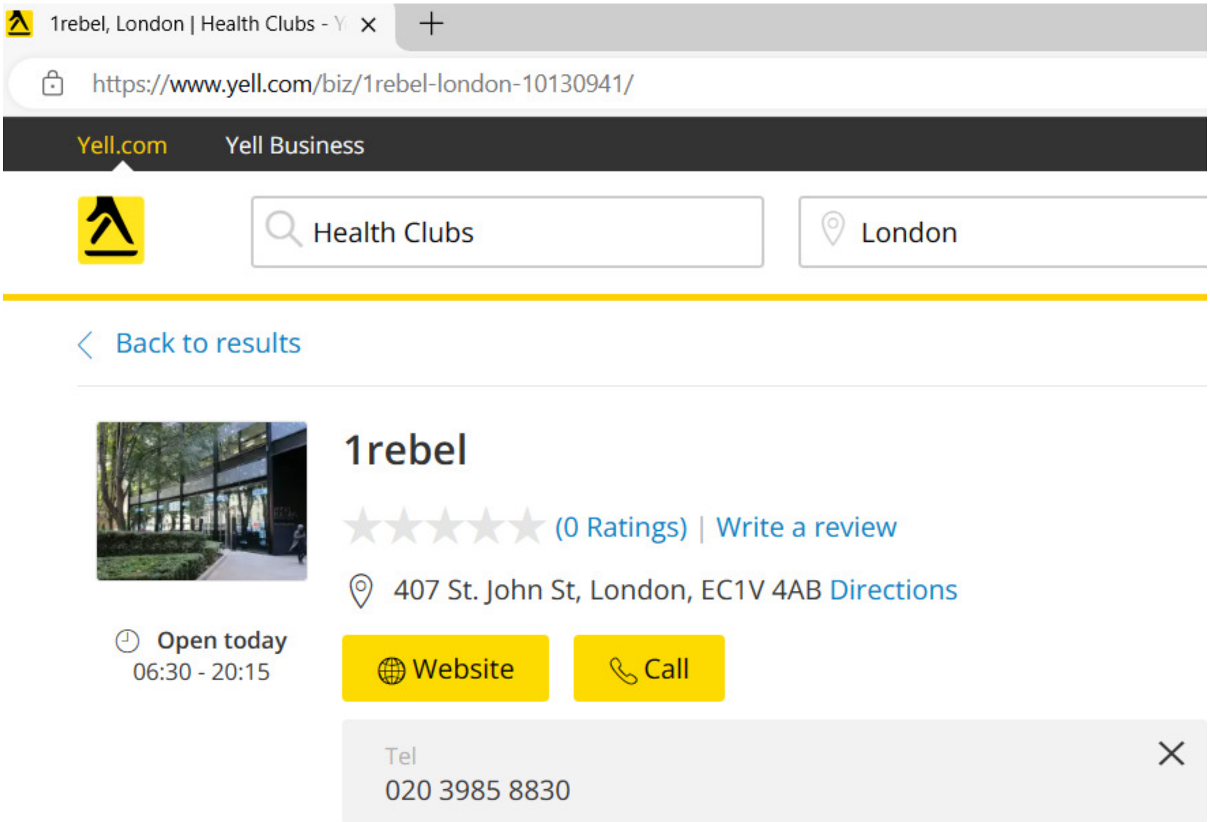


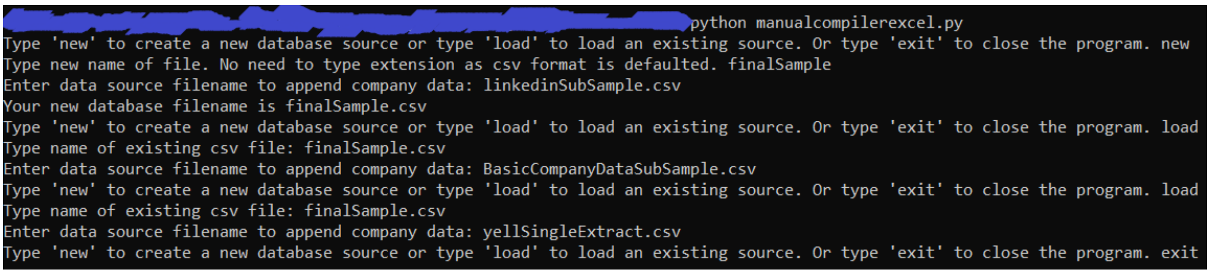
SCREENSHOT GROUP E: FAILED WEB SCRAPE ATTEMPTS ON YELL UK











Screenshot Group F: Final Test F3 with verified data sources

8. References:

Browser stack. 2024. What is Test Case Prioritization? | BrowserStack. Available at: https://www.browserstack.com/test-management/features/test-run-management/what-is-test-case-prioritization [Accessed: 18 March 2024].

Eby, K. 2019. Free Test Case Templates | Smartsheet. Available at: https://www.smartsheet.com/test-case-templates-examples#:~:text=1%20Confirming%20login%20functionality%20on%20a%20username%20and,either%20the%20username%20or%20password%20component%20More%20 [Accessed: 18 March 2024].

Foster, A. 2022. 10 Ethical Marketing Principles. Ethical marketing December 13. Available at: https://www.ethical.marketing/blog/10-ethical-marketing-principles [Accessed: 10 February 2024].

Jain, A. 2023. What is Customer Lifetime Value (CLV), and How to Calculate It. Available at: https://www.gartner.com/en/digital-markets/insights/what-is-customer-lifetime-value [Accessed: 30 January 2024].

Misra, N. 2024. Data Extraction Types & Techniques: A Complete Guide. Nanonets April 30. Available at: https://nanonets.com/blog/data-extraction-types-techniques/ [Accessed: 9 May 2024].

Oracle. 2021. What Is a Database | Oracle. Available at: https://www.oracle.com/database/what-is-database/ [Accessed: 24 February 2024].

ProWebScraper. 2023. The 5 Best Programming Languages for Web Scraping – ProWebScraper. ProWebScraper November 28. Available at: https://prowebscraper.com/blog/best-programming-language-for-web-scraping/ [Accessed: 7 February 2024].

Puzhevich, V. 2021. Functional vs Non-Functional Requirements: Key Differences. Available at: https://scand.com/company/blog/functional-vs-non-functional-requirements/ [Accessed: 25 February 2024].

VishwashVishwakarma. 2023. Difference between MySQL and SQLite – GeeksforGeeks. Available at: https://www.geeksforgeeks.org/difference-between-mysql-and-sqlite/ [Accessed: 10 March 2024].

9. Bibliography:

Apify. Apify: Full-stack web scraping and data extraction platform. https://apify.com/ [Accessed: April 2024]. Useful web scraping solution with a freemium package.

Asana. Work, Task & Project Management Resources • Asana. https://asana.com/resources [Accessed: February 2024]. Website with extensive knowledge of software development methodologies.

Baker, K. The Ultimate Guide to Product Marketing in 2024. https://blog.hubspot.com/marketing/product-marketing [Accessed: February 2024]. Guide for applying marketing knowledge to your products.

Baweja, C. Python and MySQL Database: A Practical Introduction – Real Python. https://realpython.com/python-mysql/ [Accessed: March 2024]. Tutorial on combining Python and MySQL for application development.

Big picture. Free Dataset – Companies | BigPicture Documentation. https://docs.bigpicture.io/docs/free-datasets/companies/ [Accessed: February 2024]. Great starting kit for company data.

Cambridge dictionary. Cambridge English Dictionary: Meanings & Definitions. https://dictionary.cambridge.org/dictionary/english/ [Accessed: April 2024]. Online dictionary for words. Useful to find definitions of unknown words you have just learned about.

Central digital and data office. Data Ethics Framework – GOV.UK. https://www.gov.uk/government/publications/data-ethics-framework [Accessed: February 2024]. Guide to using data principally.

CIM. Marketing Qualifications, Training and Membership | CIM. https://www.cim.co.uk/ [Accessed: February 2024]. Website offering a variety of marketing knowledge, including the '7Ps' model.

Companies house. Companies House – GOV.UK. https://www.gov.uk/government/organisations/companies-house [Accessed: February 2024]. Free datasets of company details are available on this website.

Cooke, R. Marketing for the 21st Century (Industry Insights) – ALLBOUND | MARKETING. https://allbound.marketing/marketing-for-the-21st-century/ [Accessed: March 2024]. Online article offering insightful knowledge on digital marketing, which encompasses data-driven marketing.

DataOx. Document Scraping and Parsing Solutions. https://data-ox.com/documets-scraping-parsing [Accessed: April 2024]. Introduction to document scraping, a possible future method for data extraction.

Decker, A. The Ultimate Guide to Marketing Campaigns. https://blog.hubspot.com/marketing/marketing-campaigns [Accessed: February 2024]. Describes a variety of campaigns marketeers would develop, including product marketing.

Department for science, innovation and technology, Office for artificial intelligence and Centre for Data Ethics and Innovation. Understanding artificial intelligence ethics and safety – GOV.UK. https://www.gov.uk/guidance/understanding-artificial-intelligence-ethics-and-safety [Accessed: April 2024]. Additional guide to the principal use of data, especially in Artificial Intelligence.

Gutteridge, A. Apple's Extended Marketing Mix: How Successful Is Apple's Marketing? | by Abby Gutteridge | Medium. https://medium.com/@abbygutteridge/apples-extended-marketing-mix-how-successful-is-apple-s-marketing-e120c3251b3e [Accessed: February 2024]. Online article on a successful company's application of marketing principles.

Investopedia. Investopedia. https://www.investopedia.com/ [Accessed: February 2024]. Website with extensive knowledge of finance and marketing for your guidance.

Jain, V. What is a Test Condition in Software Testing? https://testsigma.com/blog/test-condition/ [Accessed: March 2024]. Describes the use of test conditions in software testing.

Khorikov, V. Unit Testing Dependencies: The Complete Guide · Enterprise Craftsmanship. https://enterprisecraftsmanship.com/posts/unit-testing-dependencies/ [Accessed: April 2024]. Describes test dependencies in software testing.

Lebovic, D. and Rivera, J. How To Protect A Non-Trademark Business Name | LegalMatch. https://www.legalmatch.com/law-library/article/protecting-my-small-business-name.html [Accessed: February 2024]. How you can protect your business's name from being used by others.

Lopresti, J. What Is a Business Process? | Definition, Importance and Examples. https://www.villanovau.com/articles/bpm/what-is-a-business-process/ [Accessed: March 2024]. Concise introduction to required processes in a business, including marketing.

Minalpandey6899. MongoDB Architecture – GeeksforGeeks. https://www.geeksforgeeks.org/mongodb-architecture/ [Accessed: April 2024]. Quick introduction to MongoDB.

Perham, G. Data-Driven Marketing: A Comprehensive Guide to Success. https://www.accudata.com/blog/data-marketing/ [Accessed: February 2024]. Tips to using data to strengthen marketing.

Python.org. Welcome to Python.org. https://www.python.org/ [Accessed: February 2024]. Contains informative guides to using Python, including additional modules.

Sahu, V. K. and Paul, H. What is Test Monitoring and Test Control in Software Testing. https://www.lambdatest.com/learning-hub/test-monitoring-and-test-control [Accessed: March 2024]. Describes the use of test monitoring and control in software testing.

Scrum alliance. Acceptance Criteria: Everything You Need to Know Plus Examples. https://resources.scrumalliance.org/Article/need-know-acceptance-criteria [Accessed: April 2024]. Useful summary of detailing acceptance criteria for software development.

Sławińska, M. How to Import a CSV File to a MySQL Database | LearnSQL.com. https://learnsql.com/blog/import-csv-mysql-database/ [Accessed: 25 March 2024]. Guide to importing 'csv' data sources into MySQL databases.

Smartproxy. Smartproxy – Best Proxy Provider & Data Collection Solutions. https://smartproxy.com/?adgroupid=1271035812895802&utm_source=bing&utm_medium=cpc&utm_campaign=Bing-Global-Brand&msclkid=4a29a8a33df217feebcdce8a7ef95c51&utm_term=smartproxy&utm_content=Smartproxy%20brand [Accessed: February 2024]. Premium proxy solution to bypass anti-bot web scraping features.

SQLite. DB Browser for SQLite. https://sqlitebrowser.org/ [Accessed: March 2024]. Considerable option for building your database.

Stobierski, T. 8 Steps in the Data Life Cycle | HBS Online. https://online.hbs.edu/blog/post/data-life-cycle [Accessed: March 2024]. Introductory article from a reputable school on the data phases.

Susmit_sekhar_bhakta. What is Data Extraction? – GeeksforGeeks. https://www.geeksforgeeks.org/what-is-data-extraction/ [Accessed: February 2024]. Concise introduction to data extraction and some methods.

User39685. What is the difference between requirements and specifications? – Software Engineering Stack Exchange. https://softwareengineering.stackexchange.com/questions/121289/what-is-the-difference-between-requirements-and-specifications [Accessed: February 2024]. Query with explained differences between requirements and specifications for software development.

VisualParadigm online. Visual Paradigm – Online Productivity Suite. https://online.visual-paradigm.com/ [Accessed: March 2024]. Free online tool for digital design.

Yell. Yell – a leading marketplace for great local businesses. https://www.yell.com/ [Accessed: February 2024]. Trendy online directory for company data scraping.

Yelp. Cardiff Restaurants, Dentists, Bars, Beauty Salons, Doctors – Yelp. https://www.yelp.co.uk/cardiff [Accessed: February 2024]. Additional online directory for company data scraping.