**Group One**

**Development Team Project: Executive Summary**

**October 23, 2023**

**The Design and Implementation of a Real Estate Listings Database for Manhattan**

**Module:** Deciphering Big Data

TABLE OF CONTENTS

[Introduction 1](#_Toc148986206)

[Objectives 2](#_Toc148986207)

[General Objective 2](#_Toc148986208)

[Specific Objectives 2](#_Toc148986209)

[Methodology 3](#_Toc148986210)

[Data Collection 3](#_Toc148986211)

[Data Cleaning 5](#_Toc148986212)

[Database Design 6](#_Toc148986213)

[Database Implementation 8](#_Toc148986214)

[Discussion 10](#_Toc148986215)

[The Relational Database Model 10](#_Toc148986216)

[The Real Estate Database 11](#_Toc148986217)

[Legal and Data Privacy Policy Compliance Requirements 11](#_Toc148986218)

[Conclusion 13](#_Toc148986219)

[References 14](#_Toc148986220)

[Appendices 16](#_Toc148986221)

[Appendix 1: Scripts for Database Implementation 16](#_Toc148986222)

[Appendix 2: Github link to Web Scraping Code 17](#_Toc148986223)

**TABLE OF FIGURES**

[Figure 1: Data Management Pipeline 3](#_Toc148986324)

[Figure 2: Search Results Page 4](#_Toc148986325)

[Figure 3: Entity Relationship Diagram 6](#_Toc148986326)

[Figure 4: Entities 7](#_Toc148986327)

[Figure 5: Relationships 7](#_Toc148986328)

[Figure 6: SQL Script 1.1 16](#_Toc148986329)

[Figure 7: SQL Script 1.2 17](#_Toc148986330)

Introduction

Manhattan, located in New York City, contains a prestigious real estate hub housing over 1.6 million residents, and stands as a beacon for global investors (PropertyShark, 2023). Over the years, the Manhattan real estate market has experienced economic shifts and urban growth, heightening its allure. However, investors grapple with identifying ideal investment opportunities in this dense Manhattan market known for its lucrative returns. To navigate this complex landscape, investors rely on platforms like Zillow for comprehensive up-to-date data on property listings (Mashvisor, 2022).

This report delves into the design and implementation of a relational database, which serves as a foundation for data-informed decisions and analysis related to investment in Manhattan’s real estate market. The intricacies of the relational database model and how it aligns with the specific needs of investors are also explored. The implementation process is discussed detailing how data was collected, cleaned, and integrated into the database. Finally, the report addresses the legal and data privacy policies related to the data management pipeline processes, ensuring data privacy and legal considerations are observed.

Objectives

General Objective

To develop a real estate database for investors in the Manhattan market, sourcing data from Zillow platform through web scraping.

Specific Objectives

* To design and discuss a logical design for the real estate database.
* To construct the real estate database informed by the logical design.
* To critically assess the database model, discussing its advantages and constraints.
* To examine legal and data privacy policies pertinent to the data management pipeline.

Methodology

The methodology is presented step by step in the main phases of the data management process, as shown in Figure 1.

A black and white image of a couple of objects

Description automatically generated

Figure 1: Data Management Pipeline

Data Collection

This phase involved obtaining real estate listings data from Zillow through web scraping. Web Scraping, as described by Kazil and Jarmul (2016), is a data extraction technique that makes use of specific python libraries to explore web content, search for relevant data and subsequently collect it.

However, instead of traditional direct web scraping techniques, a more refined approach was adopted by using the Scrapeak Web Scraping Application Programming Interface (API). This was due to its simple usage via the endpoints provided and structured data response in JavaScript Object Notation (JSON) format.

The endpoint provision feature was utilised to collect specific data, attuned to cater to the investors’ needs. Two endpoints: ‘listing’ and ‘property details’ were used. The ‘listing’ endpoint was pivotal in fetching the results from the search for property listings. This endpoint takes two parameters: the API key and search Uniform Resource Locator (URL). The API key is an authorisation tool, provided by Scrapeak upon user registration, to identify and authenticate users. The search URL corresponds to the specific listing page returned by Zillow, upon searching for listed properties as shown by the arrow in Figure 2 below.

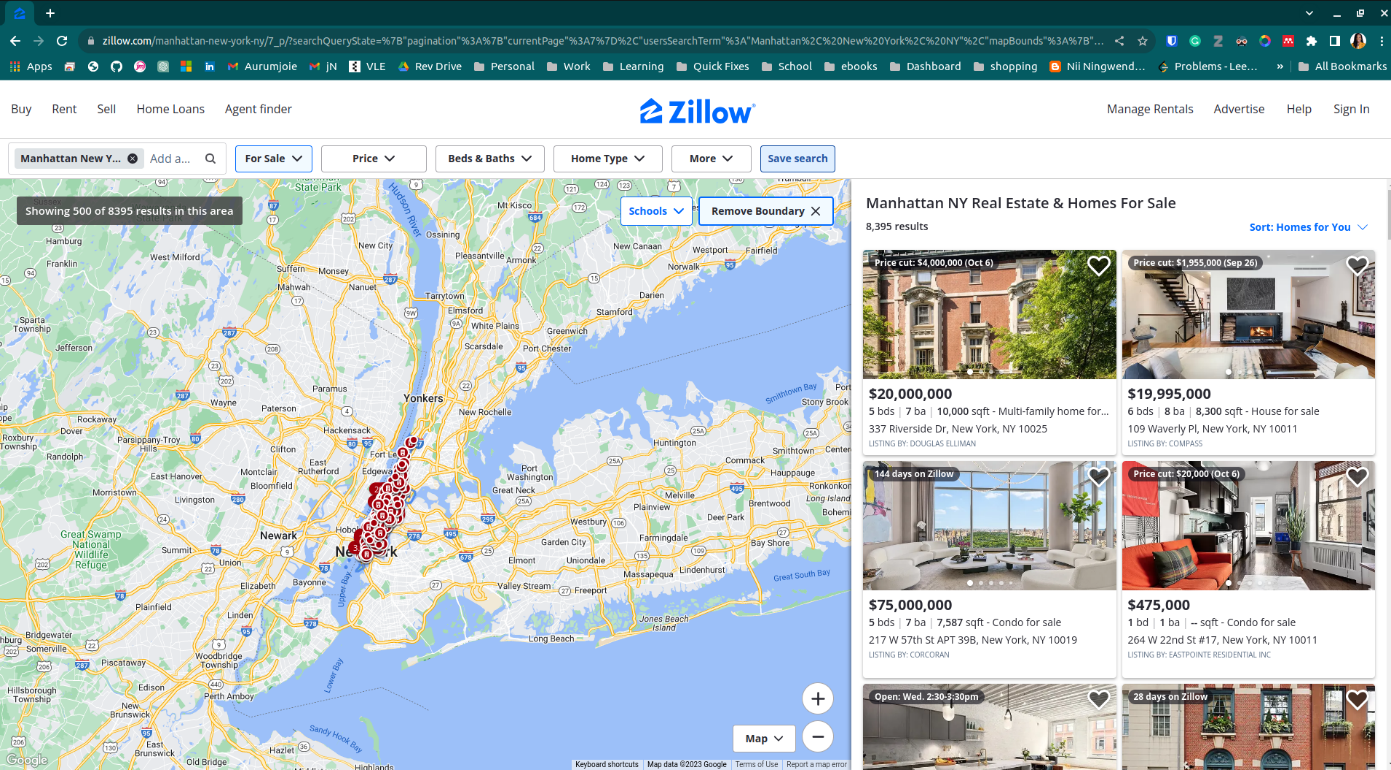


Figure 2: Search Results Page

The ‘listing’ endpoint returned a JSON-formatted response containing a list of properties available for sale. This format allowed seamless inspection of the data and led to the observation that the listing endpoint response lacked a comprehensive breakdown of all the expected property attributes, such as agent, agency details and price history, which were important in catering to the investors need of understanding the market. Hence the adoption of the ‘property details’ endpoint.

The ‘property details’ endpoint returns a comprehensive breakdown of each property by taking in two parameters, the API key, and Zillow Property ID (ZPID). ZPID is a unique identifier assigned to each property listing in Zillow and was extracted from the response returned by the ‘listing’ endpoint. The integration of these two endpoints facilitated the enrichment of the quality and depth of the data collected.

Data Cleaning

The data collected consisted of nested objects defining various attributes of a property e.g., the address object within the JSON had, street, city, state, zip code, community, and subdivision. The data showed this high dimensionality across the entire JSON response and once transferred to a data frame had 255 columns.

Once the data frame was closely examined the following columns were picked as containing the most useful and relevant data: 'zpid', 'bedrooms', 'bathrooms', 'yearBuilt', 'homeType', 'price', 'currency', 'livingArea', 'description', 'address', 'neighborhoodRegion', 'attributionInfo'. The columns chosen did not all have standard data types such as string or integer as some were JSON objects and required further cleanup.

To better represent the location, the ‘address’ column that contained a JSON object with ‘street’, ‘city’, ‘state’, ‘zipcode’, ‘community’ and ‘subdivision’ was split into two new columns by picking the ‘zipcode’ and ‘street’ for each property. These two columns were picked because they contained more unique attributes specific to each property compared to ‘city’ and ‘state’, which were the same for all properties, i.e., New York. The community and subdivision columns did not contain any data and were therefore dropped.

The ‘neighborhoodRegion’ contained values in the form of {'name': 'Harlem'} and to clean this column the value from the key ‘name’ was extracted, i.e., ‘Harlem’. The ‘attribution’ column had several nested values inside a JSON object and within them were the agent and agency details. This data was extracted then added to the data frame as separate columns i.e., 'agentName', 'agentPhoneNumber', 'agencyName', 'agencyPhoneNumber' and the broader ‘attribution’ column was dropped.

The next steps involved cleaning up the data frame by enforcing datatypes. Columns such as ‘yearBuilt’ and ‘bedrooms’ were converted to integer and all empty or null values replaced by the appropriate value e.g., 0. The columns were then appropriately named e.g., removing filler names by renaming ‘neighborhoodRegion’ to ‘neighborhood’ and finally re-ordered to match the intended destination table structure.

The ‘priceHistory’ data went through similar cleanup steps and the data prepared for import into the database.

Database Design

A screenshot of a computer

Description automatically generated

Figure 3: Entity Relationship Diagram (Harrington, 2016)

The database design is represented by the Entity Relationship Diagram (ERD) above which shows a unified view of the database and its relational model. Grounded on the data collected in the previous steps, this design captures entities that encapsulate the various facets of the real estate market in Manhattan.

The design identifies four primary entities:

A close-up of a blue and white text

Description automatically generated

Figure 4: Entities

To ensure the distinctness of each record within the entities, a 'Primary Key' (denoted as 'PK' in the ERD) is integrated. Originally, a unified key named 'zpid' interconnected the records. However, to optimize the relational model's efficiency, a normalization process segregated the data into distinct entities, each getting assigned auto-incremented primary keys: ‘agentID’, ‘agencyID’, and ‘priceHistoryID’.

These primary keys were fundamental in the creation of relationships between the data entities. The following are the relationships created:

A blue and white text on a white background

Description automatically generated

Figure 5: Relationships

Database Implementation

With the database design defined, the implementation of the database proceeded. To implement the relational database model My Structured Query Language (MySQL) was used as the Relational Database Management System (RDBMS).

The first step was to create the database and the tables using MySQL Workbench; a Graphical User Interface (GUI) tool that allows creating, modelling, and managing of MySQL databases (Letkowski, 2014).This involved writing SQL scripts to create the database schema and tables, specifying the attributes, data types, and relationships which were defined in the database design.

The scripts included the necessary constraints such as primary keys, foreign keys, unique constraints, and default values to enforce data integrity and maintain consistency within the database. Furthermore, indexes were created to optimise search query operations, for example the property table has indexes on the columns ’homeType’, ’zipcode’, and ’neighborhood’, this speeds up the queries that filter, sort, or group by these columns.

The next step was to populate the database with data collected from Zillow. To do this, SQLAlchemy, a popular SQL toolkit and Object-Relational Mapping (ORM) tool, was used. This ORM tool provides a simpler process of inserting, updating, and querying the data in the database. Using SQLAlchemy, Python scripts were written to transform the scraped data, which after data cleaning, was in data frames, into structured Python objects. Furthermore, using PyMySQL, a python package used to create an API interface to access MySQL databases, a connection was established by defining the credentials of the database like database name, host name, user, and password.

With the connection to the database established and objects for the entities in the database created, the data was inserted into the database by mapping the corresponding objects to the entities and executing the script to convert the data frames into SQL tables.

Discussion

The design and implementation of the real estate database resulted in several insights and findings discussed in this section.

The Relational Database Model

Codd (1970) introduced the relational model as a way of describing data in its natural structure, without superimposing any additional structure for machine representation purposes. Jatana et al. (2012) further augment this notion by characterising a relational database model as a systematised assembly of data set out in tables. These tables, bearing resemblance to spreadsheets, ensure that data points are not only uniquely organized but also interconnected through specific relationships that define their association.

The attributes of the relational model described by Codd and Jatana et al became the foundation for its selection for the real estate database. The model’s emphasis on data independence ensured that potential alterations in data structure or representation would not disrupt any operations or user interactions. Such resilience is vital in a dynamic environment like real estate, where listings, property details, and market trends can frequently change.

However, the lack of scalability of the relational database model presents a potential limitation for the future of the database. Relational databases, as noted by Győrödi et al. (2015), do not efficiently support high scalability beyond a certain point. This inherent limitation can restrict the performance and responsiveness of the database as the volume of real estate data grows. On the other hand, transitioning to cloud database services offers a promising avenue to mitigate these scalability concerns. By leveraging cloud infrastructure and services, the real estate database could benefit from features like automatic scaling and handling of replicas. These services not only support larger data volumes but also ensure the database remains responsive for more simultaneous users.

The Real Estate Database

The real estate database has been constructed to empower investors with data-driven insights. The historical price data emerges as a pivotal component. It offers investors a lens through which they can trace property value fluctuations over time, enabling them to strategise based on historical trends and make more informed investment choices. In addition to offering a temporal view of price change, the price history entity dives deeper, pinpointing the causes of these shifts within the 'event' attribute. Furthermore, the 'priceChangeRate' attribute quantifies these changes, presenting them as a percentage variation over defined durations. This granular data is important, particularly for investors in Manhattan's high-end real estate market, where the modus operandi often revolves around buying to eventually resell, rather than to retain for rental income (Santarelli, 2023). They, therefore, can predict what times are best to put their properties back on the market to make profits.

Legal and Data Privacy Policy Compliance Requirements

To ensure the integrity and legality of the data collection process, Zillow’s terms and conditions were examined and adhered to. The terms and conditions (Zillow Group, 2023) explicitly state that scraping is permissible solely for individual and non-commercial purposes. Any intentions to use or distribute Zillow's data for commercial ends, necessitate a distinct agreement. Importantly, it is strictly prohibited to utilise scraped data for the establishment or sustenance of any rival online real estate platforms.

As such, the main objective to collect and store data on Manhattan for-sale properties to aid investors to make data-informed decisions follows these terms and conditions.

Stringent adherence to data privacy policies and laws was also paramount. The New York Privacy Act (NYPA) played a pivotal role given the geographical focus of the data extraction i.e., since Manhattan is not in Europe the General Data Protection Regulation (GDPR) did not apply. The NYPA allows scraping of publicly available data but underscores the importance of safeguarding personal data collected publicly. Personal data according to the NYPA (2023) is any information that identifies or can reasonably be linked, directly or indirectly, to a specific individual or household. In this case, this would be the names and contact details of agents and agencies. To collect personal data, notice and consent is required from the affected parties. However, NYPA exempts personal data that is already publicly available from these requirements. This reiterates the foundational precept of 'No privacy in public', a principle echoed by the broader United States Privacy Law (Xiao, 2021).

Conclusion

The evolving landscape of Manhattan's real estate market underscores the necessity for investors to access precise, comprehensive, and timely data. This report has highlighted the significance of a well-structured real estate database, as seen in the implementation of the logical design, in enabling data-informed investment decisions. By adhering to legal guidelines and harnessing tools like Zillow in compliance with their terms and conditions, the database ensures legitimacy and reliability. Emphasising crucial entities like historical price data further cements its relevance, providing investors with a clear lens to gauge market trends. As the real estate market continues to transform, data-driven insights will undeniably remain at the heart of successful investment strategies. Future adaptations, potentially integrating cloud SQL services, could further enhance the database's scalability, ensuring it remains a pivotal resource for investors in Manhattan's intricate real estate market.

References

Codd, E. F. (1970). A relational model of data for large shared data banks. *Communications of the ACM*, *13*(6), 377–387. Available from: <https://doi.org/10.1145/362384.362685> [Accessed 22 October 2023].

Győrödi, C., Győrödi, R., & Sotoc, R. (2015). A Comparative Study of Relational and NonRelational Database Models in a Web- Based Application. *International Journal of Advanced Computer Science and Applications*, *6*(11), 78–83. Available from: <https://www.researchgate.net/publication/285752873_A_Comparative_Study_of_Relational_and_Non-Relational_Database_Models_in_a_Web-_Based_Application> [Accessed 22 October 2023].

Harrington, J. L. (2016). ‘Entities and Relationships,’ in: Harrington, J. L. *Relational Database Design and Implementation* (4th ed.). Morgan Kaufmann. Available from: <https://learning.oreilly.com/library/view/relationaldatabasedesign/9780123747303/B9780123747303000048.xhtml> [Accessed 14 October 2023].

Jatana, N., Puri, S., Ahuja, M., Kathuria, I., & Gosain, D. (2012). A Survey and Comparison of Relational and Non-Relational Database. *International Journal of Engineering Research & Technology*, *1*(6). Available from: <https://www.researchgate.net/publication/316498950_Comparative_Analysis_of_Relational_and_Nonrelational_Databases_in_the_Context_of_Performance_in_Web_Applications> [Accessed 22 October 2023].

Kazil, J., & Jarmul, K. (2016). ‘Web Scraping: Acquiring and Storing Data from the Web’, in: *Data Wrangling with Python : Tips and Tools to Make Your Life Easier* (1st ed., pp. 275–312). O’Reilly Media, Incorporate. Available from: <https://ebookcentral.proquest.com/lib/universityofessexebooks/reader.action?docID=4543981> [Accessed 12 October 2023].

Letkowski, J. (2014). Doing database design with MySQL. *Journal of Technology Research*, *6*. Accessed from: <https://www.researchgate.net/publication/271910489_Doing_database_design_with_MySQL> [Accessed 12 October 2023].

New York Privacy Act, (2023). Accessed from: <https://nyassembly.gov/leg/?default_fld=&leg_video=&bn=S00365&term=2023&Summary=Y&Actions=Y&Text=Y> [Accessed 22 October 2023].

PropertyShark. (September 2023). *2023 Home Prices & Sales Trends | New York, NY Real Estate Market*. Available from: <https://www.propertyshark.com/mason/market-trends/residential/nyc/manhattan> [Accessed 12 October 2023].

Santarelli, M. (September 22, 2023). *Manhattan Real Estate Market: Prices, Trends, Forecast 2023*. Norada Real Estate Investments. Available from: <https://www.noradarealestate.com/blog/manhattan-real-estate-market/> [Accessed 12 October 2023].

Xiao, G. (2021). Bad Bots: Regulating the Scraping of Public Personal Information. *Harvard Journal of Law & Technology*, *34*(2). Available from: <https://jolt.law.harvard.edu/assets/articlePDFs/v34/6.-Xiao-Bad-Bots-Regulating-the-Scraping-of-Public-Personal-Information-edit.pdf> [Accessed 22 October 2023].

Zillow Group. (2023). *Terms of Use*. Available from: <https://www.zillowgroup.com/terms-of-use/> [Accessed 22 October 2023].

Appendices

Appendix 1: Scripts for Database Implementation

A screenshot of a computer program

Description automatically generated

Figure 6: SQL Script 1.1

A screenshot of a computer code

Description automatically generated

Figure 7: SQL Script 1.2

Appendix 2: Github link to Web Scraping Code

[**https://github.com/Ngugi-Joy-Grace/real-estate-property-listings-scraper**](https://github.com/Ngugi-Joy-Grace/real-estate-property-listings-scraper)