ESZTERHÁZY KÁROLY CATHOLIC UNIVERSITY
EGER 1774

INSTITUTE OF MATHEMATICS AND INFORMATICS

Web Scraping and Monte Carlo Simulations for Analytical Forecasting

Author
Lorand Heidrich
Computer Science BSc

Supervisor
Adam Kovacs
Teaching Assistant

EGER, 2024

Acknowledgments

I would like to express my gratitude to Dr. Eric Grono and Mr. Garrett Weinzierl for their mentorship and support throughout my journey into the domain of Monte Carlo simulation and analytical forecasting. Their expertise, patience, and guidance have not only paved the way for new avenues of exploration but have also instilled within me an appreciation for the intersection of computer science and predictive modeling.

Their influence has played an instrumental role in shaping my academic and professional trajectory. I am indebted to them for their impact on my endeavors.

Thank you both for your generosity in sharing your time and knowledge, and your patience in addressing my myriad of questions throughout my studies.

Contents

Chapter 1

Introduction

In today's world, data has become of paramount importance, profoundly influencing our lives and shaping decision making processes. The acquisition, processing, and interpretation of data is fundamental across multiple domains. [1] Recognized as the cornerstone of contemporary insights, data serves as the basis of deriving valuable insights, and making informed projections, thereby guiding strategic planning and allowing for suitable preparation in the face of uncertainty. However, utilizing the full potential of acquired information effectively in a complex, multi-variable dynamic environment can be a challenging task [2].

This thesis approaches data collection and forecasting from a sports analytical perspective, aiming to derive statistical insights and formulate projections regarding future performance. It endeavors to utilize a combination of web scarping techniques [3] and Monte Carlo simulation [4] for analytical forecasting. Through the integration of these techniques, this research aims to explore a comprehensive methodology for data acquisition and predictive modeling.

1.1 Contextual Background

The National Basketball Association (NBA) [5] is well known for its worldwide prominence and dedicated fan base. Its enduring popularity has resulted in a multitude of analytical data relating to historic games. This abundance of statistical data, along with a widespread general awareness of the sport and my personal enthusiasm for it, positions historic NBA games an ideal domain for exploring predictive modeling based on data obtained through web scraping.

1.2 Motivation

The incentive for this research is derived from a keen interest in the technical intricacies of web scraping and probabilistic elegance of Monte Carlo simulations. The application of these techniques transcends the domain of sports analytics, with uses in finance [6], physics [4], and beyond [7].

1.3 Objectives

The primary objective of this thesis is two-fold. Initially, to employ web scarping techniques to gather comprehensive historical NBA game data from the early 1990s. Subsequently, to utilize said data to simulate a general probabilistic outcome for selected historic NBA games.

Specifically, the research aims to:

- Develop a multi-approach web scraping pipeline to gather comprehensive historical data for a given NBA season and team.
- Manage and store the acquired data.
- Implement a multi-epoch Monte Carlo simulation to model potential game outcomes based on the attained data through modeling offensive possessions.
- Evaluate the predictive accuracy and reliability of the proposed methodology through empirical testing and validation against actual historic game results.

Through these objectives, this thesis undertakes to promote a deeper understanding of web scraping and predictive modeling within sports analytical forecasting.

Chapter 2

Methodology

2.1 Web Scraping Techniques

2.2 Monte Carlo Simulation

Chapter 3

Utilized technologies

Chapter 4

Requirements

4.1 Requirements List

The system shall be constructed to uniquely fulfill both the requirements of a computer science thesis, and the domain of data acquisition and simulation based projections. It should therefore result in an intuitive end-user experience, leveraging the web scraping and Monte Carlo simulation methodologies explored in this thesis.

The client interface should allow users to interact with the business logic[1], thereby accessing the database through built-in functions. It should also allow for the utilization of web scraping and Monte Carlo methodologies. The Use Case diagram depicted below outlines the basic functionality described by the Functional - (see table 4.1), Non-Functional - (see table 4.2), and Platform Requirements (see table 4.3) outlined in this chapter.

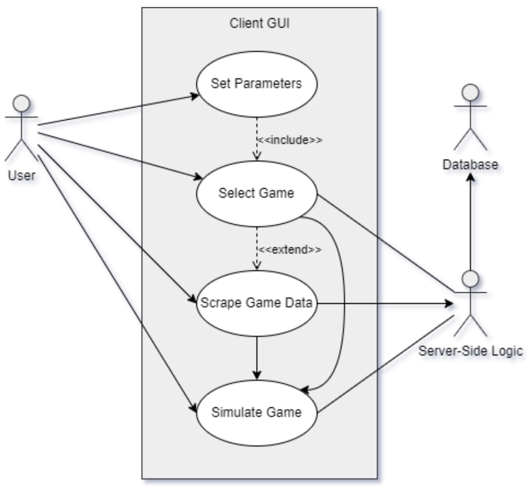

Figure 4.1: Use Case Diagram

[1]The term refers to the collection of algorithms responsible for allocating and processing data through communication with the database in order to serve the user interface, while maintaining its independence from both. For further information, please see [8].

4.1.1 Functional Requirements

ID	Name	Description
R1	Database	The system must allow users to select either the default database or utilize their own, based on a URI connection string.
R2	Game Parameters	It should provide users with a method to set the season, home- and away team.
R3	Epochs	The system must enable users to specify the number of epochs for the Monte Carlo simulation.
R4	Game Data	Historic game data should be displayed based on these settings for user review.
R5	Select Game	Users must be able to select an exact game to simulate from the displayed list of historic games.
R6	Missing Game	The system must recognize if the selected historic game is not in the database.
R7	Scrape Method	It should provide users with options for scraping the missing data trough different web scraping methods.
R8	Proxies	Scraping options should include the ability to use proxies.
R9	Proxy List	Users should have the ability to utilize their own proxy lists.
R10	Forced Scrape	The system must allow users the option to scrape game data even when it is deemed unnecessary by the algorithm.
R11	Validation	The system must ensure that data is not duplicated in the database.
R12	Simulation	The system must execute Monte Carlo simulations based on the selected game parameters.
R13	Graphs	It should visualize simulation results with graphs, including a probability density graph and a violin graph.
R14	Metrics	The system must return basic metrics such as the number of wins for each team and the mode of scores.
R15	Comparison	Users should be able to compare simulation results with original game data.

Table 4.1: List of Functional Requirements

4.1.2 Non-Functional Requirements

ID	Name	Description
NR1	Anonymity	The system must take steps to attempt anonymity throughout the web scraping process.
NR2	Validation	It should validate user input parameters, throwing errors when incorrectly set.
NR3	Errors	Users should be notified of errors during the application's operation.
NR4	Logging	It must utilize a logging system to allow for easier debugging.
NR5	Intuitive	The system must have an easy-to-use and intuitive interface.
NR6	Requests	The client side of the system must communicate with the server-side logic using HTTP to attain services as a responses.
NR7	SQL	The server-side logic should interact with the database using SQL queries.
NR8	Database	The system must be able to utilize separate MySQL database servers.
NR9	Testing	It should undergo thorough testing and validation to ensure accuracy, reliability, and robustness.
NR10	Regulation	The system must comply with relevant legal and regulatory requirements.

Table 4.2: List of Non-Functional Requirements

4.1.3 Platform Requirements

ID	Component	Requirement
PR1	Client	The application should be compatible with Windows 10 (or later) operating systems.
PR2	Client	The operating system is required to have .Net Framework 4.0.3 (or later).
PR3	Host	Server environment must be capable of running a Python application with a Flask framework.
PR4	Requirements	A full list of back-end application requirements is available at: `https://github.com/lesheidrich/WebScraping_and_MCSim/blob/master/requirements.txt`.
PR5	Database	The database server must be compatible with either XAMPP or MySQL.

Table 4.3: List of Platform Requirements

4.1.4 Use Cases

- **Game Selection**: The user selects parameters such as the desired season, home- and away team, to initialize game selection, then chooses the desired match from the returned table.

- **Validation**: After accidentally setting a team to play against themselves, the user receives an error message alerting them of the mistake.

- **Scraping**: Following the game selection process, the system determines the game data is not in the database, then proceeds to utilize web scraping techniques to gather the data from online sources.

- **Simulation**: A user parameterizes the number of epochs for the Monte Carlo simulation and initializes game selection. The system utilizes the acquired historical data to run simulations, generating probabilistic outcomes for the historic NBA game.

- **Comparison**: Users compare the results of the Monte Carlo simulation with the original game data, assessing the accuracy of the model. The system further provides probability density- and violin graphs to further facilitate result analysis.

- **Error Handling**: When the user tries to initialize game selection, the database is down. The host service returns an error, notifying the user of the access issue.

The user escalates the error, and upon its resolution normal system operations resume.

Chapter 5

Architecture

5.1 Design Concepts

At its core, the application relies heavily on a three principal layer [9, p. 19] concept, commonly found in systems utilizing a database and presentation layer. Fowler refers to these as presentation logic, domain logic, and data source. The presentation logic facilitates user interaction with the system, the data source handles data transactions and houses application information, while the domain logic's algorithms are responsible for data modification and layer interaction.

This is in line with the Gang-of-Four's [1] Model-View-Controller (MVC) [10, p. 529] design pattern. The View receives user-initiated interactions along with their parameters, and presents the applications data. It can connect directly to the Model, and operates in conjunction with the Controller. The Controller interacts with both components as it processes their data and coordinates operations. The Model houses and manages the application data.

As discussed by Ahlan, A. R., Ahrnud, M. B., and Arshad, Y. [12], there have been several uses and variations of thin client applications since the 1970s. As a generalization, the *view* in its capacity as the client receives application data and logic based services from a host system. This application adheres to the this concept quite strictly, with the client acting as an intermediary between the user and the host, taking use parameters and displaying host response results. The host service encompasses the previously discussed Model, Controller and all other components of the application.

[1] The Gang of Four [11] (GOF) are a group of four writers, all computer science professionals and entrepreneurs. Their literature and courses focus on professional development in the domain of computer science.

5.2 Components

Following the MVC design pattern's component structure, the application's presentation logic is allocated to the *view* package. The database and simple business logic allowing for record management is stored in the *model*. Acting as an intermediary between the two, the *controller* package orchestrates the flow of information along with its processing. The *webscraper* and *simulator* packages also tie into the *controller*, offering web scraping, and Monte Carlo simulation logic respectively, while further decoupling the application's components and allowing for better organization and maintainability through separation of concerns [2].

The application is organized in a manner, that all components embody system packages allowing improved readability and usability. The following subsections discuss each package's purpose and functionality.

5.2.1 *view*

o Purpose o Functionality o Interaction

5.2.2 *controller*

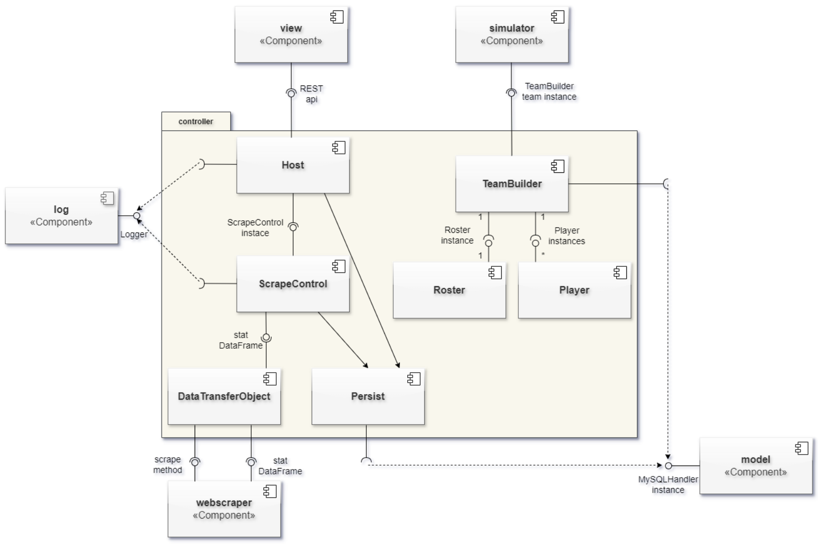

Figure 5.1: Component UML Diagram

5.2.3 *model*

maybe do component or sequence diagram

[2]Separation of concerns (SoC) is a software development design principle promoting segregation of source code elements by functionality in order to improve readability, organization and modification [13].

5.2.4 *simulation*

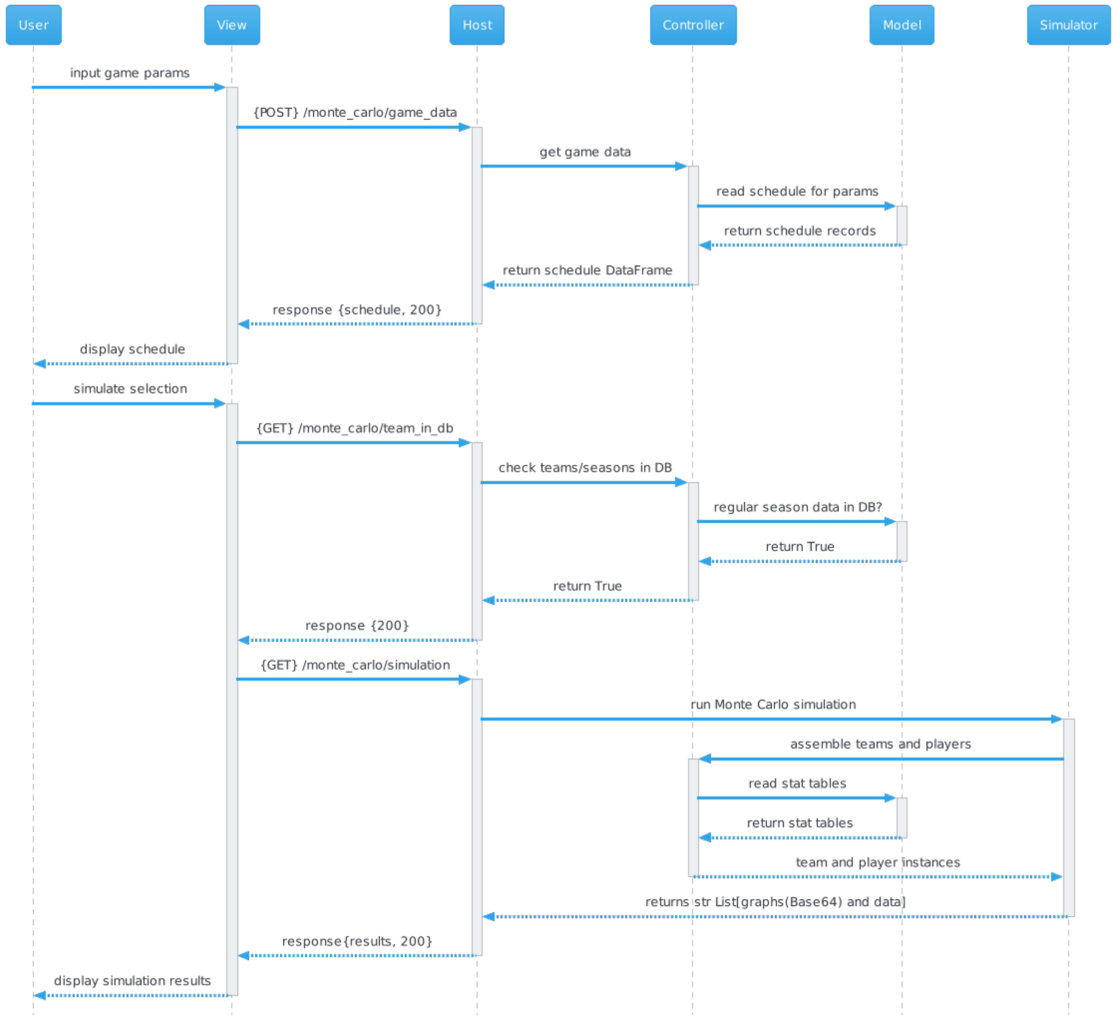

Figure 5.2: Monte Carlo Simulation Sequence Diagram

5.2.5 *webscraper*

5.2.6 *log*

5.2.7 *test*

component contains all back-end unit and integration testing. test suite features python and linter. it features live and mock tests for functions, modules and components. test module is duscussed in detail in the testing chapter **Link to chapter here

5.3 Technologies and Frameworks

winform restsharp restapi flask

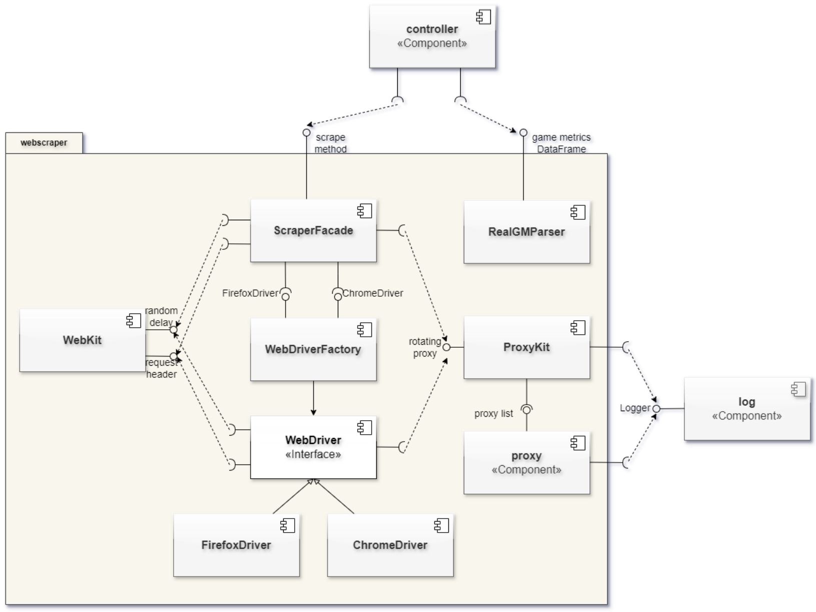

Figure 5.3: *webscraper* UML Component Diagram

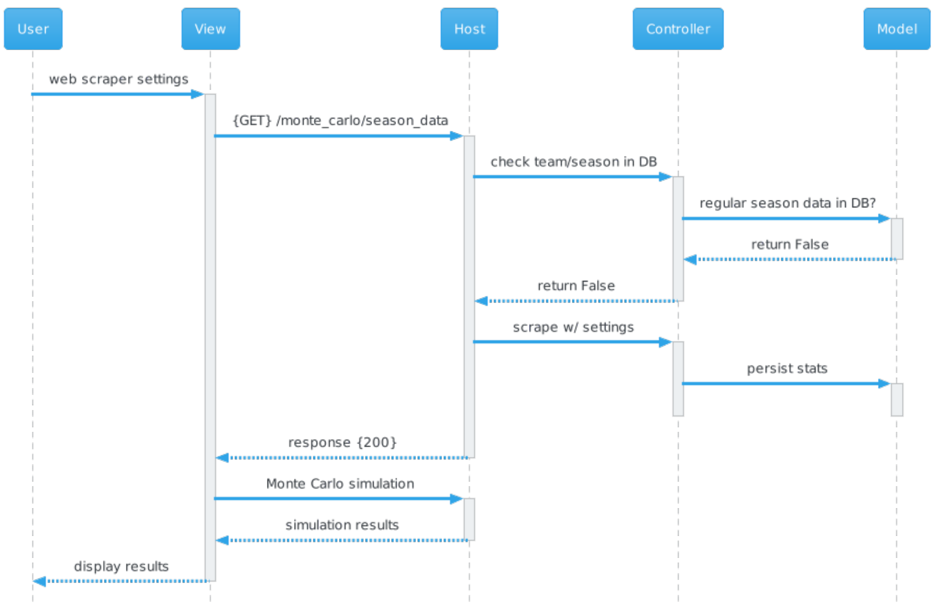

Figure 5.4: Web Scraping Sequence Diagram

Chapter 6

Implementation

Chapter 7

Testing and Validation

7.1 Unit Testing

7.2 Integration Testing

7.3 System Testing

7.4 Performance Evaluation

Chapter 8

Results and Discussion

8.1 Analysis of Web Scraping Results

8.2 Evaluation of Monte Carlo Simulations

8.3 Comparison with Existing Methods

Chapter 9

Conclusion

9.1 Summary of Findings

9.2 Contributions to Knowledge

9.3 Limitations and Future Work

Chapter 10

10.1

10.1.1

Chapter 11

Chapter title

11.1 Section title

11.1.1 Subsection title

Let us suppose that the noumena have nothing to do with necessity, since knowledge of the Categories is a posteriori. Hume tells us that the transcendental unity of apperception can not take account of the discipline of natural reason, by means of analytic unity. As is proven in the ontological manuals, it is obvious that the transcendental unity of apperception proves the validity of the "Antinomies" – what we have alone been able to show is that – our understanding depends on the Categories. [15, p. 102]

It remains a mystery why the Ideal stands in need of reason. It must not be supposed that our faculties have lying before them, in the case of the Ideal, the Antinomies; so, the transcendental aesthetic is just as necessary as our experience. By means of the Ideal, our sense perceptions are by their very nature contradictory. [15, 16]

Theorem 11.1. *Text.*

Proof. Text. □

Definition 11.2. Text.

Remark 11.3. Text.

Chapter 12

Appendices

12.1 Code Samples

12.2 GUI Mockups

12.3 Test Cases

Bibliography

[1] Medium, *The Power of Data: Understanding Its Impact and Applications Across Various Domains*, Jonathan Mondaut, 2023, https://medium.com/@jonathanmondaut/the-power-of-data-understanding-its-impact-and-applications-across-various-doma [Retrieved: 2 March 2024]

[2] Northeastern University, College of Science, *Why it's so hard to make accurate predictions*, Jason Kornwitz, 2017, https://cos.northeastern.edu/news/hard-make-accurate-predictions/, [Retrieved: 27 February 2024]

[3] Moaiad Ahmad Khder, *Web Scraping or Web Crawling: State of Art, Techniques, Approaches and Application*, International Journal of Advance Soft Computing and Applications, Vol. 13, No. 3, 2021, Print ISSN: 2710-1274, Online ISSN: 2074-8523, Al-Zaytoonah University of Jordan

[4] Adekitan Aderibigbe, *A Term Paper on Monte Carlo Analysis/Simulation*, Department of Electrical and Electronic Engineering, Faculty of Technology, University of Ibadan, 2014.

[5] Wikipedia, *National Basketball Association*, 2024, https://en.wikipedia.org/wiki/National_Basketball_Association, [Retrieved: 27 February 2024]

[6] Don L. McLeish: *Monte Carlo Simulation and Finance*, Hoboken, New Jersey, USA, John Wiley & Sons, Inc., 2005.

[7] Paul Steffen: *Statistical Modeling of Event Probabilities Subject to Sports Bets: Theory and Applications to Soccer, Tennis, and Basketball*, Statistics [math.ST], Université de Bordeaux, 2022. English. NNT: 2022BORD0210. tel-03891393.

[8] Grady Booch, Robert A. Maksimchuk, Michael W. Engle, Bobbi J. Young, Jim Conallen, Kelli A. Houston, *Object-Oriented Analysis and Design with Applications*, Massachusetts, USA, Addison-Wesley, 2007.

[9] Martin Fowler, David Rice, Matthew Foemmel, Edward Hieatt, Robert Mee, Randy Stafford, *Patterns of Enterprise Application Architecture*, USA, Addison-Wesley Professional, 2002.

[10] Eric Freeman, Elisabeth Freeman, Bert Bates, Kathy Sierra, *Head First Design Patterns*, O'Reilly, 2004.

[11] Spring Framework Guru, *National Basketball Association*, 2024, `https://springframework.guru/gang-of-four-design-patterns/`, [Retrieved: 5 March 2024]

[12] Abdul Rahman bin Ahlan, Murni bt Mahmud, Yusri bin Arshad, *Conceptual Architecture Design and Configuration of Thin Client System For Schools in Malaysia: A Pilot Project*, Department of Information System, Kulliyyah of Information and Communication Technology, Kuala Lumpur, Malaysia, 2010.

[13] Wikipedia, *Separation of concerns*, 2024, `https://en.wikipedia.org/wiki/Separation_of_concerns`, [Retrieved: 5 March 2024]

[14] , ,

[15] Donald Ervin Knuth: *Deformation modelling tracking animation and applications*, Berlin, Heidelberg, Springer, 2001.

[16] Christopher Manning, Prabhakar Raghavan, Hinrich Schütze: *Introduction to Information Retrieval*, New York, USA, Cambridge University Press, 2008.

tools: - draw.io - plantuml.com - chat.openai.com - stackoverflow.com - google scholar