# William and Mary's Union Central Automation Project

Andrew Choi
*Data Science*
*College of William and Mary*
Williamsburg, Virginia
ajchoi02@wm.edu

David Cho
*Data Science, Math*
*College of William and Mary*
Williamsburg, Virginia
dhcho@wm.edu

Carlo Mehegan
*Computer Science, Data Science*
*College of William and Mary*
Williamsburg, Virginia
cjmehegan@wm.edu

## I. Project Description

### A. Project Statement

Union Central, located on the first floor of the Sadler Center, is a dedicated place at the College of William and Mary that offers various types of games, consoles, and fun activities. This is an important place for students as they are able to take time off from their stressful work and relax by themselves or with friends by playing some games and other sort of activities. In previous work with Union Central employees, a project was done where occupancy and managing equipment rentals were logged to be turned into usable data. The start of the data collection during the shifts included information about the use of different game consoles, rentals of various items, and overall occupancy trends. While this project has addressed the start of incorporating data driven analysis for Union Central, such valuable data still remains underutilized.

This project aims to build beyond the scope of the previous project by automating the process of data analysis and visualization for Union Central's website. By creating specific Python scripts and utilizing Google Sheet's API, we plan on building a pipeline that pulls new data, and after cleaning it, returning it back, ideally in real time, to the website for the viewer to see. Our goal with this project is to be able to showcase the ways automation can bring forth progress in a data-driven subject.

### B. Project Delivery

The main deliverable will be an interactive visualization dashboard. This visualization is a rough floor plan representing the different areas of Union Central. Users click on different areas or the boxed elements on the right to see updated statistics about that section. For example, clicking on the game console will show data on their use. Clicking on the game console area will show data on how the consoles compare to each other. This pop-up information is in the form of pie charts, bar graphs, or other small visualizations along with descriptions. These visualizations will update either at the press of a button or automatically to incorporate new data.

We aim to use the data to create automated visualizations for us. This important task will help to not only provide strong and usable visuals for the audience to see, but also act as a great resource for the Union Central team as this will offer ready to use visualizations that either employees or professional staff can use freely in company-wide meetings each semester.

## II. Data Description

### A. Dataset

The dataset currently in use comes from Union Central data collected from Fall 2023, Spring 2024, and Fall 2024. We have currently incorporated each of the semesters' data, separated by filters, into the up-to-date website.

This data is collected by Union Central employees where all the data is saved through Google Sheets. The key features in the rental logs include student names and IDs. Because the data is sensitive with student names, we anonymized this data by removing the names entirely and gave each student their own identifier (starting with 1, 2, etc.) in order to maintain the relationship between renter and rentals while removing personal identifiers. With the data being taken of William and Mary students, the dataset used in the project is not able to be shared for public use and replication of this project.

### B. Data Features

The Union Central dataset is separated into multiple tables. Here is an excerpt from the occupancy data. The key features here are the day, date, headcount, and time of record. Time between occupancy recordings varies, from ten minutes to an hour. Occupancy is typically taken every 20-30 minutes.

| Day | Date | Headcount | Time of Record |
|---|---|---|---|
| Friday | 8/25/2023 | 3 | 7:30 |
| Friday | 8/25/2023 | 0 | 7:50 |
| Friday | 8/25/2023 | 2 | 8:14 |
| Friday | 8/25/2023 | 5 | 8:39 |
| Friday | 8/25/2023 | 14 | 9:11 |
| Friday | 8/25/2023 | 9 | 9:25 |
| Friday | 8/25/2023 | 13 | 9:33 |
| Friday | 8/25/2023 | 12 | 9:52 |
| Friday | 8/25/2023 | 3 | 7:30 |

Fig. 1. Sample of 2,441 Occupancy Records

Pool, Shuffleboard, Air Hockey, and Foosball data is collected in another table. The key features here are the date, game, time in, time out, and the table number.

| Date | Name | ID | Game | In | Out | Table # |
|---|---|---|---|---|---|---|
| 8/25 | | | Pool | 12:14 | 12:25 | 1 |
| | | | Foosball | 12:14 | 12:30 | |
| | | | Pool | 12:41 | 12:55 | 3 |
| | | | Pool | 12:58 | 1:07 | 2 |
| | | | Pool | 1:02 | 1:36 | 3 |
| | | | Pool | 1:09 | 1:20 | 3 |
| | | | Pool | 1:12 | 1:18 | 1 |

Fig. 2. Sample of 3,598 rental records for table games

Here is an excerpt, with names and IDs removed. We used a python script to systematically remove the names and IDs, and replace them with unique identifiers. The date in the first column is only recorded for the first entry of each day. We took this into account when parsing the data. The "Table #" column is only used if the game is Pool. The reason we do this is because there are multiple pool tables, but only one shuffleboard, air hockey, and foosball table.

Video game data is collected in another table. Names and IDs have been removed. The key features here include the date, type of console, type of game, number of controllers used, as well as the time in and time out.

| Date | Name | ID | Console | Game | Controllers | In | Out |
|---|---|---|---|---|---|---|---|
| 9/2 | | | Xbox | Minecraft | 1 | 12:23 | 1:15 |
| | | | Wii | Mario Party 8 | 4 | 1:37 | 2:00 |
| | | | Wii | Mario Kart | 2 | 2:55 | 5:02 |
| | | | Xbox | Minecraft | 2 | 3:45 | 5:00 |
| | | | Wii | Splatoon | 1 | 5:15 | 5:45 |
| | | | Xbox | Forza Horizon 5 | 1 | 7:02 | 9:44 |

Fig. 3. Sample of 448 rental records for video games

Board game data is collected in another table. The key features here are the date, type of game, the time in and time out, and finally the notes. Names and IDs have been removed. The "Game" column here is a drop down, which only contains the nine most popular board games.

| Date | Name | ID | Game | In | Out | Notes |
|---|---|---|---|---|---|---|
| 9/9 | | | Other (specify in Notes) | 7:14 | 8:08 | Chess |
| | | | Other (specify in Notes) | 9:12 | 9:31 | Set |
| 9/10 | | | Other (specify in Notes) | 6:40 | 7:45 | Taboo |
| 9/15 | | | Deck of Cards | 6:31 | 6:34 | |
| | | | Deck of Cards | 7:53 | 8:08 | |
| 9/19 | | | Uno | 7:18 | 7:55 | |

Fig. 4. Sample of 113 rental records for board games

## III. Methods

The main method for this project started with several Python scripts which were developed to automate the data preparation process. The initial phase involved gathering data from Union Central staff and analyzing records stored in Google Sheets. To ensure the data was easily parsable and ready for cleaning we used Google Sheets API to help convert the Google Sheets data into a workable CSV format.

The Python script sheets_to_csv.py was specifically designed to handle the first automation step of this project. It processed Google Sheets files containing four pages of data, each representing a distinct data category (board games, table games, video games, and occupancy). The script included a function, get_current_semester, which dynamically determined the current semester based on the date. This function helped appropriately name the raw data files generated. After initializing authentication with the Google API, the script established a connection to the Google Sheets API, iterated through the required pages, and saved the extracted data into a designated raw_data folder. It is important to note that the credentials and configuration details in the script were taken off as that is private information used by Union Central staff for their specific use case.

The next step was executed through the uc_parsing.py script. This script transformed the raw CSV files into cleaned data files where helper functions helped to read CSV files, process rows into lists, and save cleaned data back into CSV format. The script handled data cleaning for different datasets by employing specialized functions for each category of data. For example, the clean_occupancy function was tailored to clean occupancy data by removing extraneous columns and empty rows, formatting time entries, and returning a cleaned and proper dataset to be used for visualizations

Similarly, the clean_games function parsed and processed data for board games, video games, and table games. This function resolved issues such as handling the "other" category by parsing information from the Notes column and updating relevant fields. This function also made sure missing values were filled, columns were standardized, and names were anonymized by replacing them with unique IDs. By addressing various data-specific cleaning requirements, this script provided a comprehensive solution for handling the diverse datasets collected by Union Central.

The cleaned CSV files from the parsing script were then used as input for the visualization script, update_viz.py. This script generated essential visual representations for the Union Central website. The script allowed for customizable visuals based on specific criteria, ensuring the images meet the requirements of the live website. The visualizations were automatically updated, improving the accessibility and efficiency of data presentation.

Finally, the project culminated in an interactive GitHub-hosted webpage showcasing Union Central's data. The website's structure, built using HTML and CSS, provided an intuitive interface. The main content was divided with a top-down map of Union Central on the left and a navigation panel on the right. The index HTML file structured the site with two main sections: the head section, containing metadata and links, and the body section, displaying the webpage's content. Clicking on a map element follows up with visualizations for the corresponding game or occupancy data, enabling users to explore analytical insights interactively.

## IV. Results and Output

HTML and CSS code were combined to create an interactive user interface showcasing automated data visualizations. In

Figure 5, we have the live webpage's layout features filled with clickable sections for each semester at the top. These sections enable users to view data specific to their selected semester.

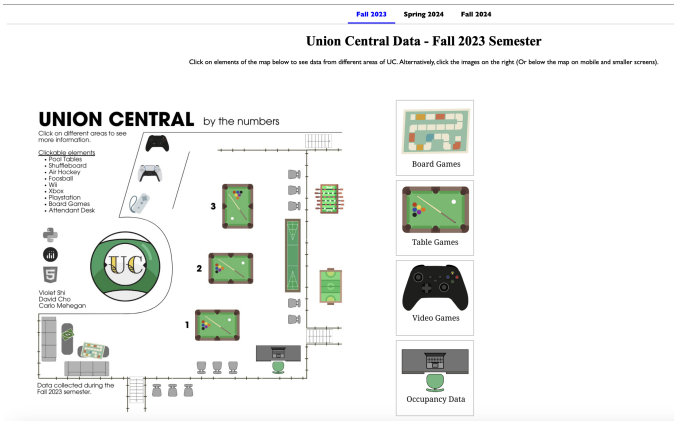


Fig. 5. Live Webpage Layout

Beneath the title, a top-down map of Union Central is displayed on the left. Interactive images on the map, when clicked, provide visualizations for specific game or occupancy data. To enhance usability, a navigation tool on the right categorizes data into board games, table games, video games, and occupancy metrics. Clicking on a category displays relevant data visualizations similar to those triggered by the map's interactive elements.

Figures 6 and 7 illustrate the visualizations generated for different datasets. When "board games" is clicked, it reveals visualizations highlighting analytical findings specific to board game data (Figure 6), while selecting "video games" generates visualizations for video game data (Figure 7). Each dataset's visualizations are tailored from the cleaned data prepared through the automated scripts.

This interactive webpage effectively showcases Union Central's data while enabling users to explore and analyze specific datasets seamlessly.

## V. Future Work

Significant progress has been achieved through this project, but several areas for future improvement remain. For example, integrating advanced platforms like Heroku could allow real-time updates of semester-specific data on the website. This would enable live visualizations as new data, such as Spring 2025 metrics, become available.

In conclusion, this Automations and Workflows project demonstrates the potential of automation in data-driven applications. By combining Python scripts with web development techniques, the project offers a robust and scalable solution for managing Union Central's data. This initiative highlights the value of leveraging automation to improve operational efficiency and accessibility for William and Mary's Union Central student center.

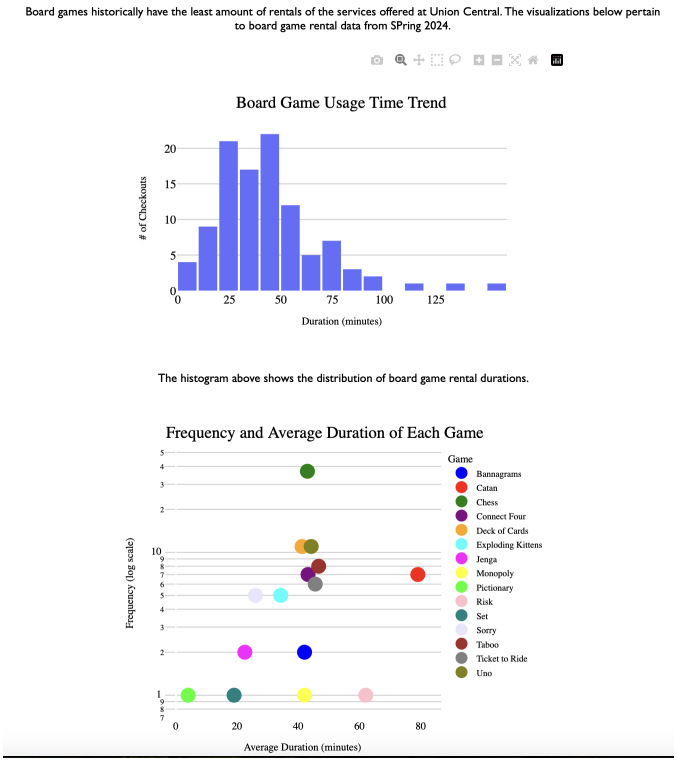


Fig. 6. Board Games Data Visualization

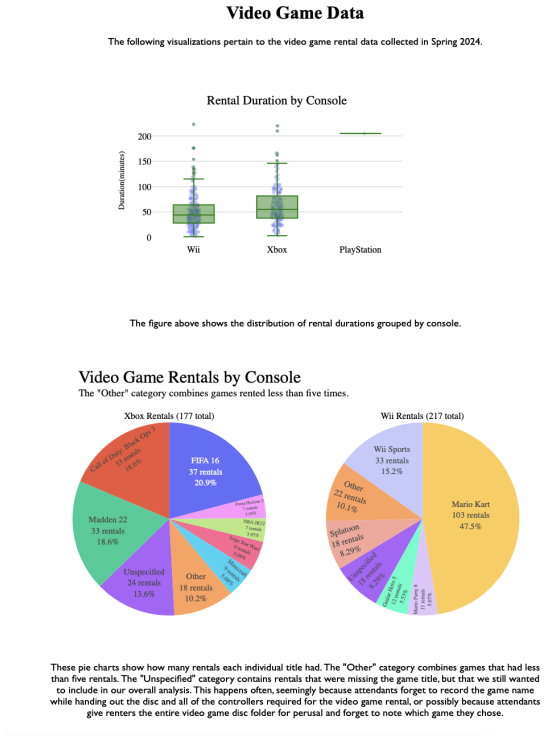


Fig. 7. Video Games Data Visualization

## VI. Group Work

Andrew Choi: Using Google Sheets API to automatically take the Google Sheets data and convert that into a usable CSV data file. Adding and updating the website data.
David Cho: Creating automatic visualizations based on the updated data. Making sure specific criteria can be specific to create different kinds of images (like png or html)
Carlo Mehegan: Creating the automatic script that will clean and parse through the updated data. Making sure to account for all forms of issues that may come with the Union Central Data
Everyone: Developing functions for analyzing the different datasets into visualizations. Each person will focus on a different subsection of the data and add their sections to the main deliverable and this project proposal.

## References


[1] Google. (n.d.). *Key concepts for the Google Sheets API*. Retrieved December 18, 2024, from https://developers.google.com/sheets/api/guides/concepts

[2] Union Central Data. (2024). *UC Data Project 2024*. Retrieved December 18, 2024, from https://tinyurl.com/ucdata2024

[3] W3Schools. (n.d.). *HTML and CSS introduction*. Retrieved December 18, 2024, from https://www.w3schools.com/html/html$_{c}ss.asp$

[4] GeeksforGeeks. (n.d.). *How to automate data cleaning in Python*. Retrieved December 18, 2024, from https://www.geeksforgeeks.org/how-to-automate-data-cleaning-in-python/