

## INTERACTIVE DASHBOARD USING SHINY IN R

**Team member: Babatunde Olawale John**

This project intends to design an **interactive dashboard using Shiny in R** to analyze crime data in Chicago, United States from January 1st, 2024 to December 31st, 2024. The dashboard provides insights into crime trends, patterns, and spatial distributions across different communities, and time frames, and its includes interactivity such as filters for crime type, and community.

The dashboard includes features such as:

- ❖ Total crime occurrences.
- ❖ Monthly crime occurrence trends.
- ❖ Crime distribution by days of the week.
- ❖ Arrests made.
- ❖ Top five communities with the highest crime occurrences.
- ❖ Crime occurrences by day or night.
- ❖ Map of Chicago showing crime frequency in different areas.

The dashboard helped answer questions about:

- i. Where crimes occur in Chicago.
- ii. When (time) crimes occur in Chicago.
- iii. The effectiveness of crime fighting in the city of Chicago.

### Data Source

The data for this study was obtained from the City of Chicago public database [here](#). The data was filtered to include only crime data for the last year, i.e. January 1, 2024, to December 31, 2024. The data contains 21 columns (attributes).

The spatial attributes (latitude, longitude, district, and community) and temporal attributes (date) will help answer the questions of where and when crimes are most commonly committed in Chicago city. Additionally, attribute such as arrest helped assess how effective the Chicago Police Department is in combating crime.

The dataset was explore to visualize:

- When crimes most commonly occur (e.g., time of the day, days of the week and months).

- Where crimes occur most frequently (e.g., hotspots or neighborhoods) and which types of crimes are most prevalent.

## Programming Language:

Shiny and R

## Expected Output

- R script code (.r)
- Processed and cleaned Chicago crime dataset (.csv)
- GitHub repository link

## Declaration

The dashboard was previously designed using ArcGIS Dashboard for a prior course. The Shiny dashboard is intended to address deficiencies in the ArcGIS Dashboard and provide a more sophisticated solution. Please view the ArcGIS Dashboard [here](#)

## Explanation of the R code chunk

### Library Installation

The required libraries were installed in code lines 1-7. These libraries include:

- **dplyr**: For data manipulation and cleaning.
- **ggplot2**: For plotting and chart visualization.
- **leaflet**: For creating interactive maps.
- **lubridate**: For handling date-time data, particularly for splitting the date and time into separate columns for days of the week, day, month, and year
- **shiny**: For building the interactive web application.

### Data Cleaning, Manipulation and Enrichment

Code lines 10-50 were dedicated entirely to cleaning the dataset to make it useful and fit for our purpose. The dataset was read from a CSV file, and unnecessary columns such as *Case.Number*, *IUCR*, *Block*, *Beat*, *Ward*, *X.Coordinate*, *Y.Coordinate*, *Year*, *Updated.On*, *Location*, and all rows with missing data was removed.

Similarly, the *Date* column was split into five distinct columns as follows;

- ❖ *Date\_only*
- ❖ *Time\_only*
- ❖ *Day\_of\_week*
- ❖ *Month*
- ❖ *Time\_of\_day* classified into Daytime and Nighttime.

Lastly, the dataset was enriched with another dataset to replace the community numbers with their corresponding names using a Left join on the *Community.Area* column.

### **User Interface Layout**

The UI layout is structured in code lines 53-108. The layout includes a sidebar panel that hosts the dropdown inputs for filtering data by community and crime type, as well as the main panel, which consists of:

- ❖ Text (Total Crime Count).
- ❖ Charts (such as monthly crime trends, arrests made, top 5 communities, crime by day of the week, and crime by time of the week).
- ❖ A map visualizing crime occurrence with Leaflet.

The UI layout also defines how the charts, text, and map are arranged within the dashboard.

### **Server Logic**

The code lines 111-119 define the reactive filtering, which enables the filtering of data and chart based on user input or selections from the sidebar panel provided in the UI section.

### **What does not work?**

The interactive filtering function on the sidebar panel did not work. Initially, I wrote the code within a Quarto document but encountered issues when introducing filtering. It was later discovered that Quarto is not designed to support interactivity. As such, I was advised to use Shiny instead. However, despite consulting various documentation sources, I still do not understand why the filtering function failed to work.

### **Use of AI**

ChatGPT was used but was sparingly use, particularly when I encountered the issue of filtering not been to work. I was able to understand that Quarto was not optimized for interactive dashboard. As such, I changed to Shiny.