**Title: Baseball Pitch Analytics: A Geospatial Web Database for Querying of MLB Player Data**

**Github Page:** [**PPxiaoP/Nothing**](https://github.com/PPxiaoP/Nothing)

## **Introduction**

Sabermetrics, the empirical analysis of baseball statistics, has revolutionized the sport by transforming how teams evaluate talent, strategize, and optimize performance. Rooted in Bill James’ pioneering work and popularized by *Moneyball*, modern sabermetrics leverages advanced metrics to quantify player value and predict outcomes with unprecedented precision. Teams now rely on data-driven insights to draft players, shift defensive alignments, and tailor pitching strategies to exploit hitter weaknesses. Beyond front offices, sabermetrics has reshaped fan engagement, media commentary, and player development, as athletes refine their mechanics using real-time analytics from tools like Statcast.

## **The Critical Role of Databases in Modern Baseball**

Databases serve as the backbone of modern baseball operations, enabling the storage, organization, and retrieval of vast datasets that drive decision-making at every level of the sport. From tracking millions of individual pitches (velocity, spin, location) to logging player biometrics, game conditions, and historical performance, databases transform raw data into structured, queryable information. For instance, a relational database allows teams to cross-reference a pitcher’s arm angle with the outcomes of specific pitch sequences or correlate weather data with home run rates in humid vs. dry environments. Advanced indexing and SQL querying enable rapid analysis of trends, such as identifying a batter’s slump against left-handed pitchers in night games or optimizing bullpen usage based on historical fatigue patterns. Moreover, geospatial databases enhance this capability by anchoring statistics to real-world contexts—for example, mapping a hitter’s fly-ball distance to a ballpark’s fence geometry to calculate "would-be home runs" in other venues. Without databases, the explosion of sabermetrics and real-time analytics—from Statcast’s tracking systems to wearable sensor data—would be unmanageable. By centralizing and contextualizing data, databases empower teams to draft smarter, deploy shifts strategically, and even negotiate contracts using predictive analytics, making them indispensable in baseball’s evolution from a game of instinct to one of precision.

## **Objectives**

1. Develop a comprehensive baseball web applicaiton dataset and interactive querying platform.
2. Design and implement an online relational database to store pitch-level data, including geospatial coordinates of ballpark locations.
3. Develop a web interface allowing users to query and overlaid on interactive ballpark maps.
4. Integrate geospatial tools to display the ballpark as one of the query elements, enhancing visual analysis and contextualization.

## **Questions to Answer**

* How can we relate the query tool on an interactive map.
* In what ways can geospatial visualization of ballpark data enhance the analysis of player?
* How can an interactive web interface improve user engagement by allowing dynamic filtering and mapping of MLB team and player data?

## **Database Design**

The project will implement a relational database using a subset of the Lahman Baseball Database. Key tables include:

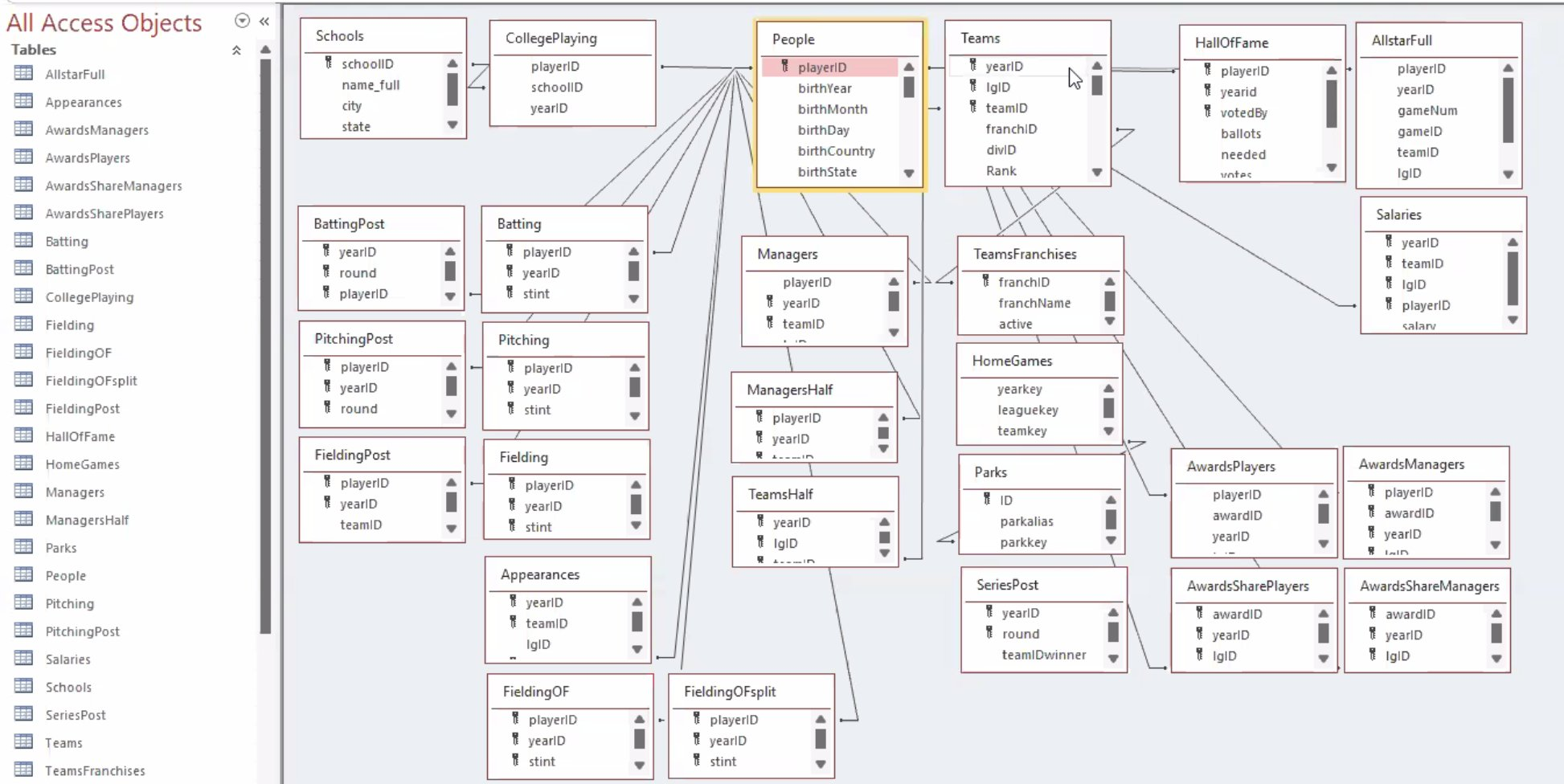
People

Batting

Pitching

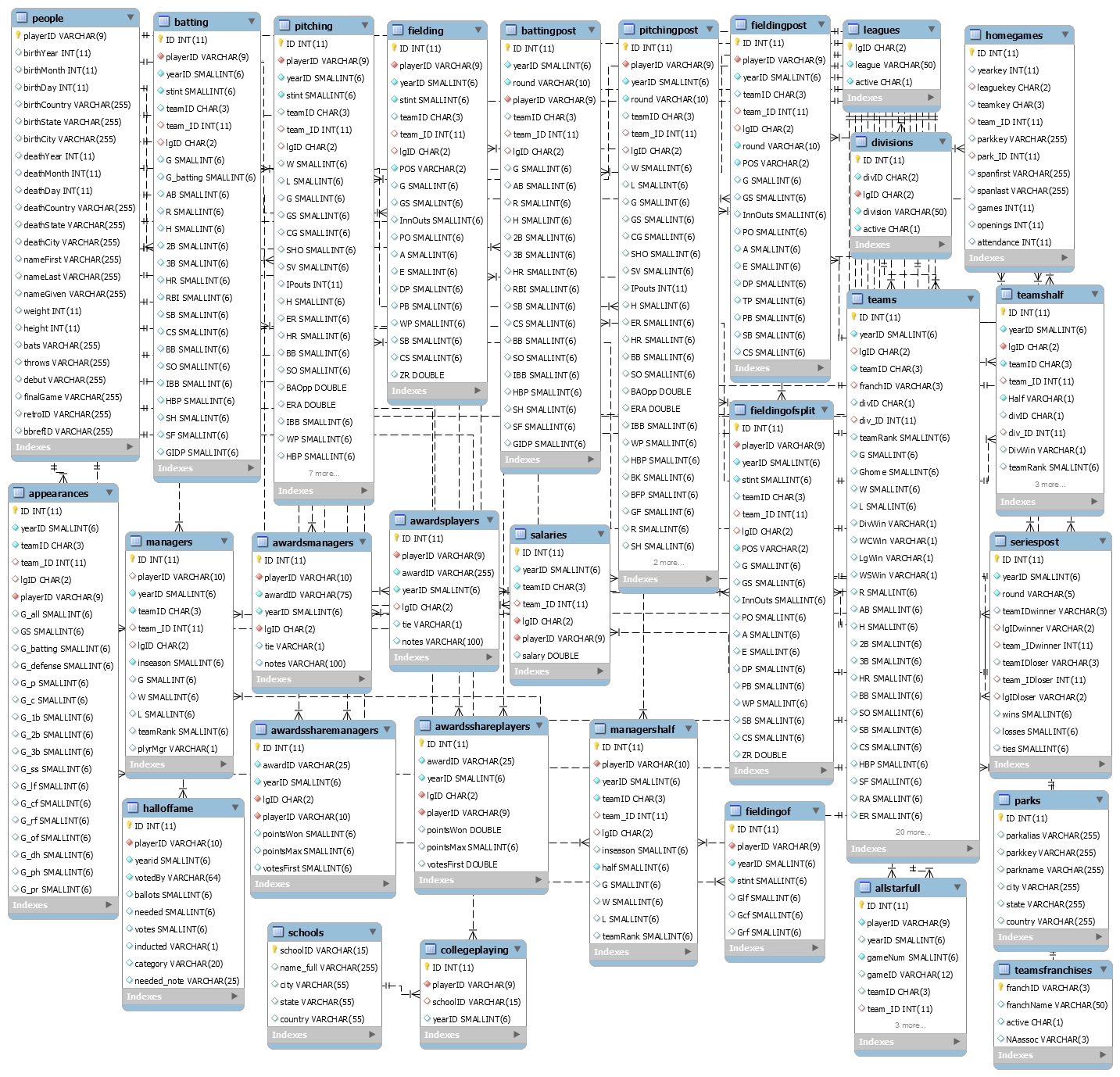
Teams

Parks



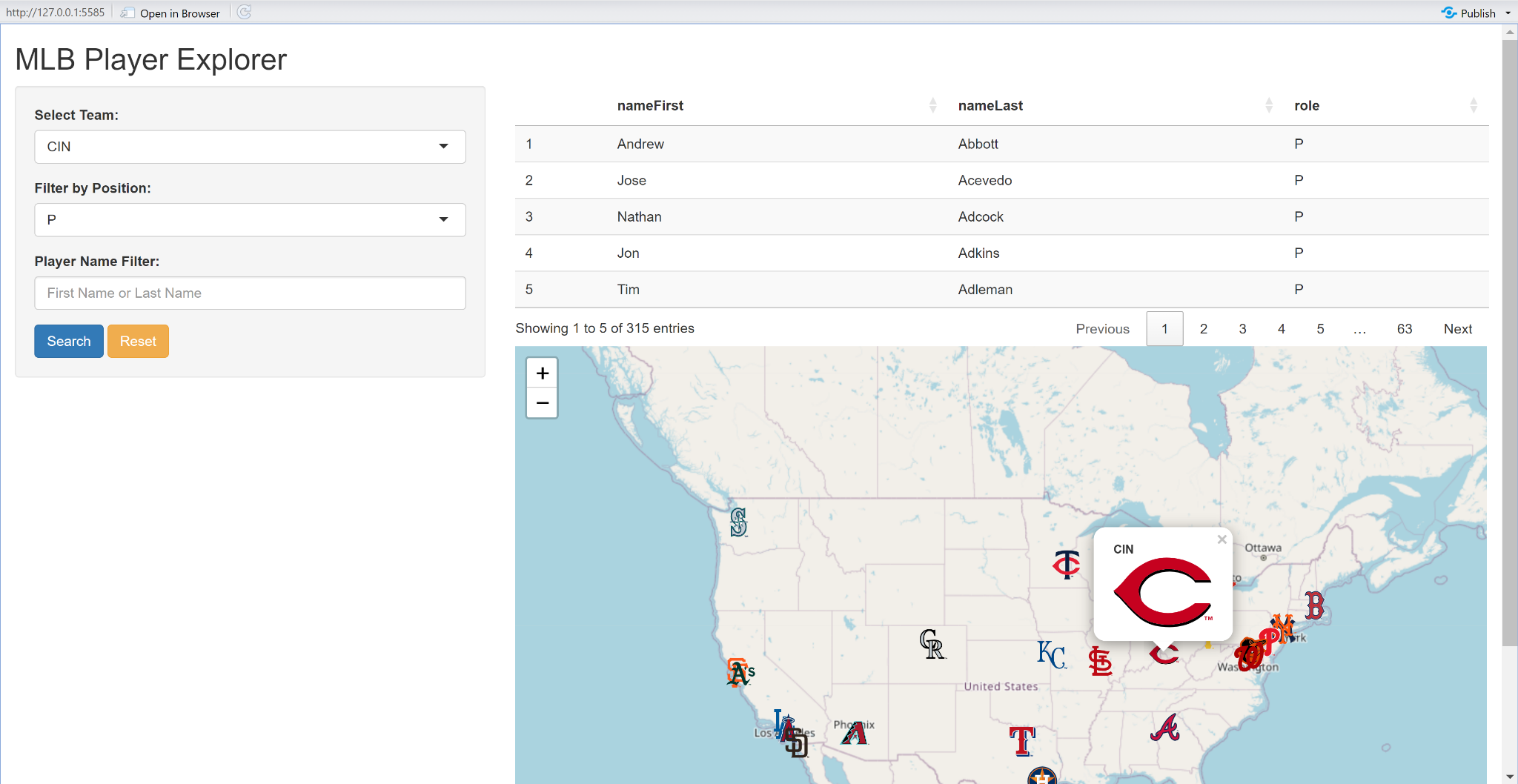
The Lahman Baseball Database is a comprehensive collection of Major League Baseball statistics, covering pitching, hitting, and fielding data from 1871 to present. It encompasses information from the American and National Leagues, as well as four other historical major leagues and the early National Association. This database serves as a valuable resource for researchers, analysts, and enthusiasts interested in baseball history and statistical analysis.

Although it contains numerous tables, this web application only involves some useful tables of it to avoid intensive workload, which are “People”, “Batting”, “Pitching”, “Teams” and “Parks”.



**Key Features:**

* Team and Player Search: Users can select an MLB team and filter players by position or name, facilitating quick access to specific player information.
* Interactive Map Interface: The application features an interactive map displaying team locations. Clicking on a team's logo updates the player list accordingly, enhancing user engagement.
* Data Visualization: Player data is presented in a clear, tabular format, allowing users to easily analyze and compare statistics.
* Reset Functionality: A reset button enables users to clear all filters and return to the default view, ensuring a seamless browsing experience.

****

## **Data Description**

### **Primary Data Source:**

* **Sean Lahman’s Baseball Database**A historical dataset covering MLB statistics from 1871 to the present, including pitching, batting, and fielding records.

### **Additional Data Sources:**

* **Team Logos:** Sourced from Major League Baseball Logos History and Chris Creamer's Sports Logos Page.
* **Ballpark Field Data:** Derived from MLB Ballparks – Fair & Foul Territory.

The integration of these data sources will allow cross-referencing between player statistics and geospatial information, enhancing both analytical depth and visual context.

### **System Architecture Overview**

1. **User Interface (UI) – Client Side**
   * **Components:**
     + **Sidebar Panel:** Provides filter inputs such as team selection, player role, and player name, plus search and reset buttons.
     + **Main Panel:** Displays the interactive data table (using the **DT** package) and the interactive map (using **Leaflet**) with team markers and logos.
   * **Technology:**
     + Runs in a web browser using Shiny's front-end framework.
2. **Shiny Server – Application Logic**
   * **Data Processing Module:**
     + **Data Integration:** Merges data from multiple Lahman tables (Fielding, Batting, Pitching, People) and computes each player's primary position.
     + **Filtering & Querying:** Uses **dplyr** to filter and join data based on user inputs.
   * **Reactive Programming Layer:**
     + **Event Handling:** Uses reactive expressions to update the data table and map when users search, click markers, or reset filters.
   * **Caching:**
     + Uses the **memoise** package to cache static geospatial data (team locations and logos), reducing redundant computations and speeding up app responsiveness.
3. **Data Sources – Backend Data**
   * **Lahman Baseball Database:**
     + Provides historical MLB statistics for player performance (batting, pitching, fielding) and biographical information (from the People table).
   * **Geospatial & Static Data:**
     + Manually defined data for MLB team locations (latitude and longitude) and corresponding logo URLs, which are merged and cached for efficiency.
4. **Rendering Components – Output Delivery**
   * **Data Table (DT):**
     + Displays filtered player records in an interactive, paginated table.
   * **Interactive Map (Leaflet):**
     + Plots team locations with custom markers (team logos) and provides pop-ups with additional team information.
5. **Deployment**
   * **rsconnect:**
     + The app can be deployed to a Shiny server or hosted on platforms like RStudio Connect for public access.

## **Conclusion**

This project will develop a comprehensive web application that integrates pitch-level MLB data with geospatial information. By leveraging modern database techniques and interactive visualization tools, the application aims to provide actionable insights into player performance and team strategies, thereby enhancing both analytical and user engagement capabilities in the realm of baseball analytics.