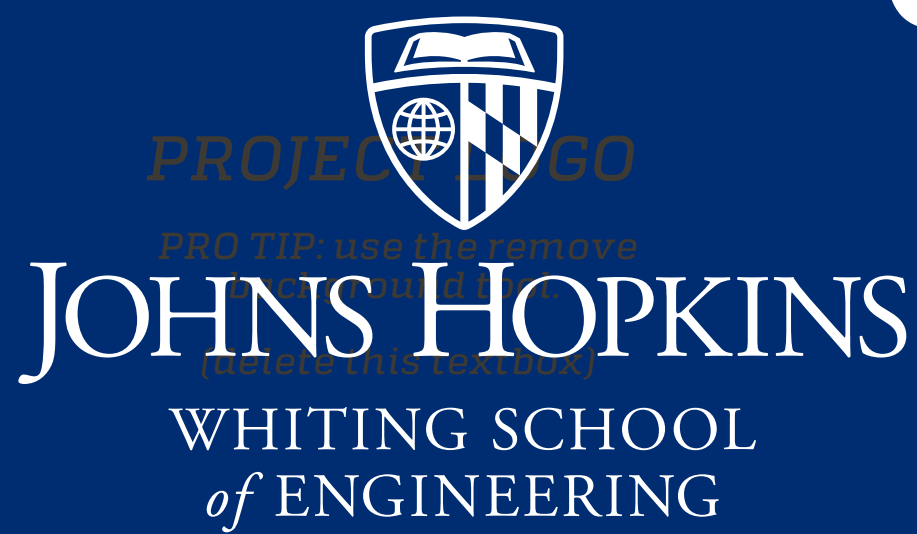
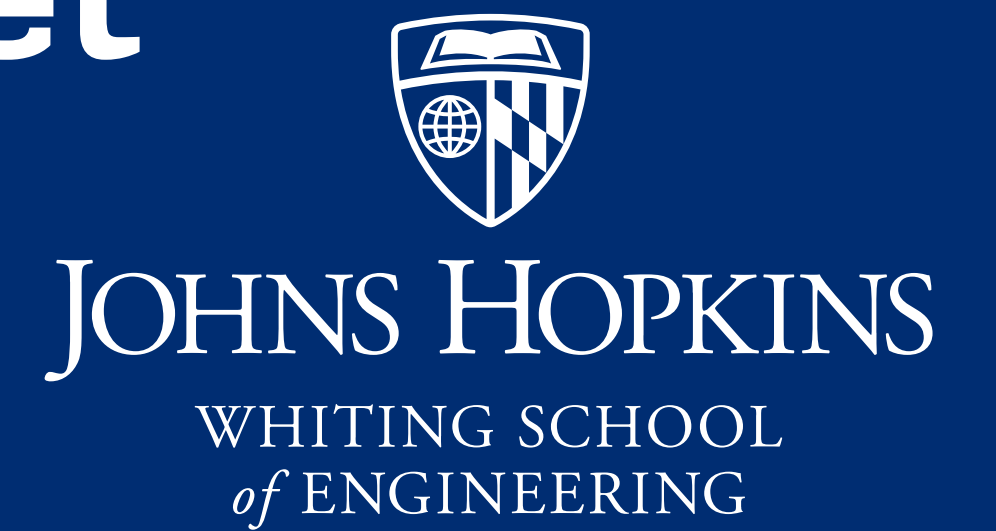


# Outfielder Optimization Widget



Helping teams in the ALPB (Atlantic League of Professional Baseball) place outfielders using data analytics from historical data. Visit [sluggeroutfielderoptimizer.pythonanywhere.com](http://sluggeroutfielderoptimizer.pythonanywhere.com)



By James Tsunman To as a part of the Sports Analytics Research Group @ Johns Hopkins with mentoring from Dr. Anton Dahbura developed for teams in the Atlantic League of Professional Baseball

## Background

Positioning of outfielders in baseball is **traditionally** derived from scouting reports, observations, and the experiences of the coaches. While this strategy has worked in the past, the **efficacy** of subjective coaching practices comes into question. These strategies often differ from team to team and when games fall apart, it raises the question of whether it was a coaching issue or just an outlier in a game. We believe that by taking a **quantitative approach** to this issue, we can create solutions **grounded in data**.

## The Idea

The SLUGGER outfielder optimization widget takes a more **analytical approach** to this problem using **Trackman** data to optimize the positioning of outfielders. The system takes into account years of data from the Atlantic League of Professional Baseball and cleans the data that are only **relevant** for outfielders. The data is then evaluated through an algorithm and the results of this data is represented by a **visualization**. The visualization includes all outfield hits from a batter, optimized positions of outfielders, with a real baseball field as the canvas for our data and results.

## Behind the scenes

Our program uses a **Python** based optimizer using kinematics and a weighting formula for outs, singles, and doubles to determine the positions for the three outfielders. The optimized position of outfielders **minimizes** the runs from the given batter.

As for our tech stack, we used **Flask** for our backend, **Trackman API** to extract our data, and **Pandas** to transform it. For our front end, a simple **React** interface was used for users to interact with our program.

## Our Solution

**1** The Slugger Outfielder Optimization widget is a **Python**-based program that turns **spray data** into **quantifiable results**.

At its core, the widget utilizes **Flask** as our backend framework and **React** as our frontend, allowing coaches to analyze **historical batting data** to make better-informed decisions on placing outfielders. Below is how our program works from a macroscopic perspective.

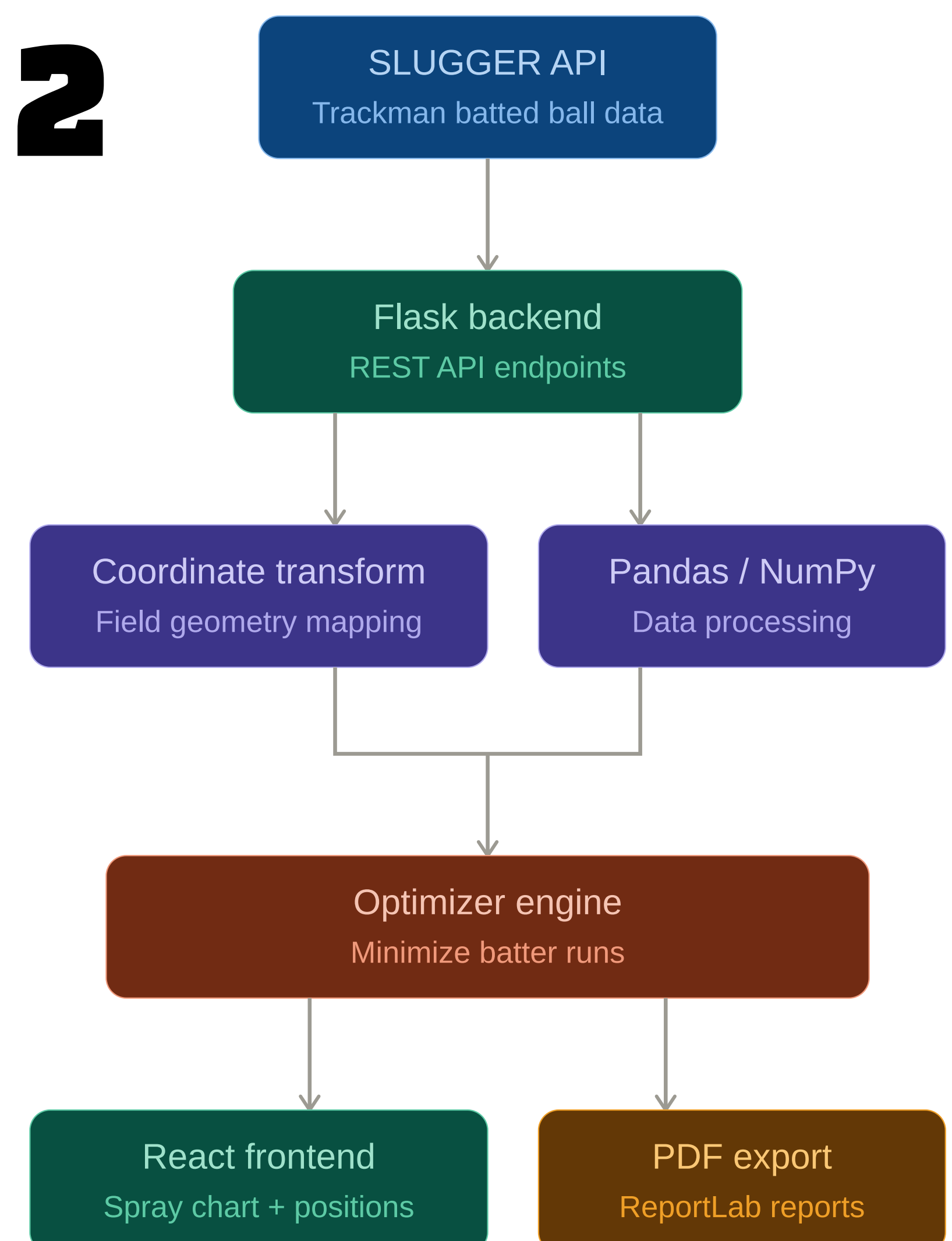
**Data** - Pulled from Slugger Trackman API, allowing retrieval of live data from the ALPB

**Cleaning** - Filtering data relevant for outfielders for each player in the league

**Optimizing** - Data is aggregated and outfielders' positions are placed to minimize runs

**Results** - Findings are displayed through detailed Matplotlib graphs

**2**



**3** How to access the widget:

- Access** - Coaches and users can visit our program at [sluggeroutfielderoptimizer.pythonanywhere.com](http://sluggeroutfielderoptimizer.pythonanywhere.com) 24/7
- Select** - Select your batter and pitcher handedness, then click generate to display a visual of batter spray data and optimal positions for outfielders
- Download** - Use the PDF download feature to save any batter's data for future use or for printing. This is great for coaches who need access to data without needing a laptop or Wi-Fi connection.



**4** What users see when you visit:

**Outfield Positioning Optimizer**

Pitcher Handedness:

Batter:

173 players with data

PDF download button →



Visualization

## Conclusion

Our program serves as the stepping stone for ALPB coaches placing outfielders with greater context. It's a tool not to replace but to **assist** coaches in understanding data in an **intuitive** way.

This program is a part of **SLUGGER**, a series of widgets designed with the ALPB in mind. You can check out the SLUGGER platform by scanning the QR code below.

