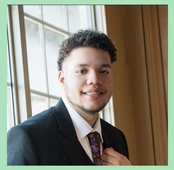
Knowball

Ben Cimini
CS



Blair Bowen
CS

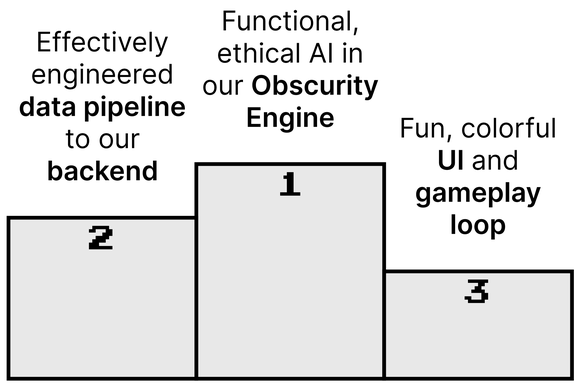
Stetson King
CS

Will Hawkins
Advisor

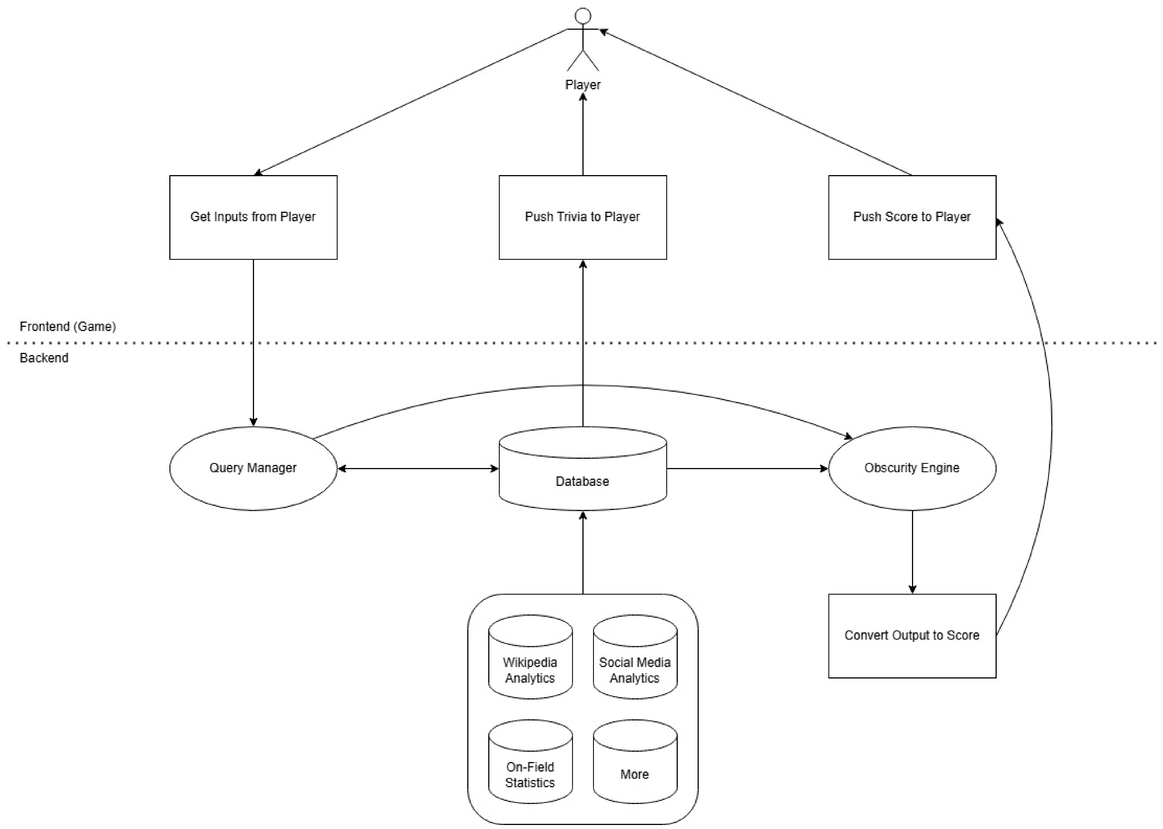
Overview

People who love sports, even rivals, have something in common: they could name obscure athletes all day. We are those people. So, we made Knowball, the **computer game** for that simple yet ubiquitous pastime. In it, you receive a daily trivia question for each of the NBA, EPL, and NFL. You must respond with five athletes to be scored on their obscurity. The scores are computed by our **Obscurity Engine**, an ML model trained on athletes' careers and fame. Play to find out if you know ball.

Accomplishments



Design



Future Work

- **Expand our dataset with more athletes and fame data.** This would improve our ML model and gameplay.
- **Live backend updates, live obscurity computations.** Sporting events should propagate automatically.
- **Host Knowball on the web.** We want the game to be publicly playable and shareable.

Challenges

Sourcing data was our biggest challenge. **No free dataset or public API** had the necessary breadth and depth, so we had to compile our own. Our data engineering process was an undertaking and an accomplishment, not least in that it's compliant with all ToS. Developing the Obscurity Engine was rather difficult, too. In early versions, many athletes clustered around a **narrow range of obscurity scores**. We curbed the behavior, in part, by supplying only those parameters that a human might care about – resulting in a human-like readout.

Obscurity Engine

The Obscurity Engine is, perhaps, the most key subsystem in our design. It's the one that makes Knowball unique. At a high level, it's an ML model for quantifying obscurity among athletes, trained on their statistics, achievements, awards, web analytics, and social media following. Given an athlete, it computes a score between 0 and 100 (0 being least obscure and 100 being most). Knowball, the game, revolves around these computations.

Under the hood, the Obscurity Engine is powered by **k-means clustering**. Simply put, k-means clustering is an unsupervised algorithm that groups data based on similarities. We felt an unsupervised algorithm was best because it prevents personal biases being injected into the model. Iteratively, it selects centroids (sentinel datapoints) and partitions the data by Euclidean distance to them. For our data, that means athletes were partitioned, approximately, into benchwarmers, role players, regular starters, superstars, and all-time greats. Granular obscurity differences were applied by athletes' web analytics and social media following.

Technologies

WIKIPEDIA · SQL · STATHEAD · Microsoft Azure · React · tailwindcss · scikit learn

Backend → **Frontend**