This is where I dump ideas, so there may be some redundancies.

Use-Case Driven UIs

Mid-Season Draft Tool

- The Player List UI 1 (Fig. 1), but modified so that each row allows some extra info to be displayed, including one or more of
  - some summary representation of several player ranking measures
  - a small message or number conveying extremely salient contextual information (injured: projected return date, teammate projected return date)
  - would-be-if-drafted team measures, or
  - the difference between $m(T)$ and $m(T - \{p\})$ (where T is user's team and p is the player) for currently-selected measure m, or
  - a summary representation of the above-defined difference for multiple measures.
- Clicking on a player will display a pop-up with details
  - performance stats for season games to date
  - additional contextual fields in each row of game data - roster, quarterback, target share
  - Potentially also some creative "timeline" display against it, showing team states and their duration.

Trade Tools

Trade Manual Analysis Could be integrated into previously-mentioned roster UI styled after the typical "Matchup" UIs (Fig. 2). We want to show comparison of two values: $m(t_0)$ and $m(t_1)$, where m is the currently-selected measure, t_0 and t_1 are user's team before and after the trade, respectively.

Trade Finder Search league for trade that would optimize some user-specified objective function $f(X)$ and satisfy a set of user-specified constraints $g_1(X) < c_1, ..., g_n(X) < c_n$, $X \in D$. We could define the feasible set D in multiple ways, but for the moment let X be some matrix representation of both teams' before and after states. (Obviously, the UI will enable the user to specify these in a more intuitive way. No mathematical optimization experience required.) For example

- "mutually advantageous"

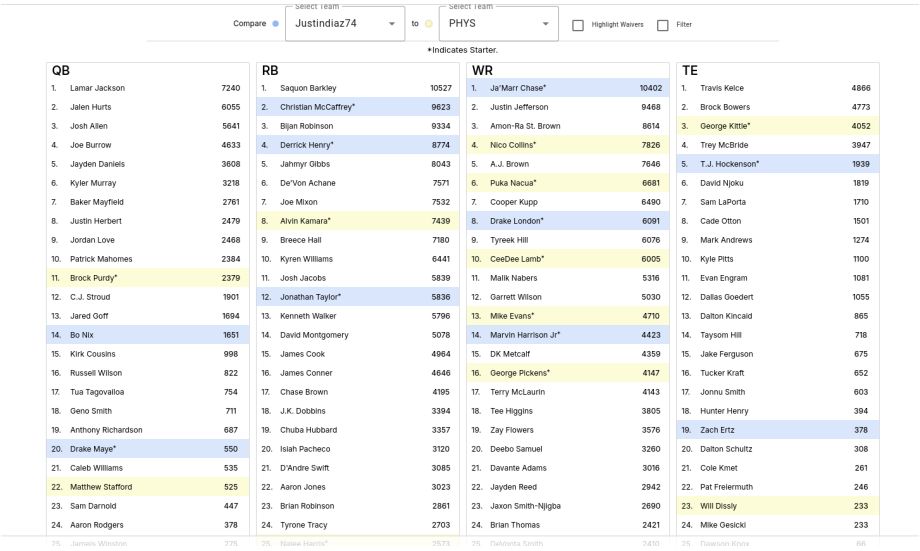

Figure 1: Player List UI 1 - "Highlighted Subsets"

  - objective function: $f(X) = gain(t_{user}, \texttt{trade})$
  - constraints: $gain(t_{opponent}, \texttt{trade}) > 0$,
  - where $gain(t, \texttt{trade})$ is how many more fantasy points team t is projected to score, rest-of-season, after making `trade`. (where we consider `trade` to be a function over the collection of sets of players, $gain(t, f) = p(f(t)) - p(t)$ where f is the trade and p is projected rest-of-season fantasy points.)
  - i.e. maximize our benefit whilst still being beneficial to the other team.

- "maximizing difference between apparent and actual opponent-team gain"
  - e.g.
    - apparent: free rankings provided online that don't take into account league settings
    - actual: projected points using league settings
    - this wouldn't be a great thing to optimize, but would be interesting to see. A better set up would be "maximize actual user-team gain while keeping apparent opponent-team gain over a minimum threshold"

We'll have to show the above defined optimization problem can be solved, or approximated to some degree of tolerance, reasonably efficiently. On second thought, it might be best to think of the criteria/constraints ourselves and give the user a set of pre-defined options. Or just present the trades we find

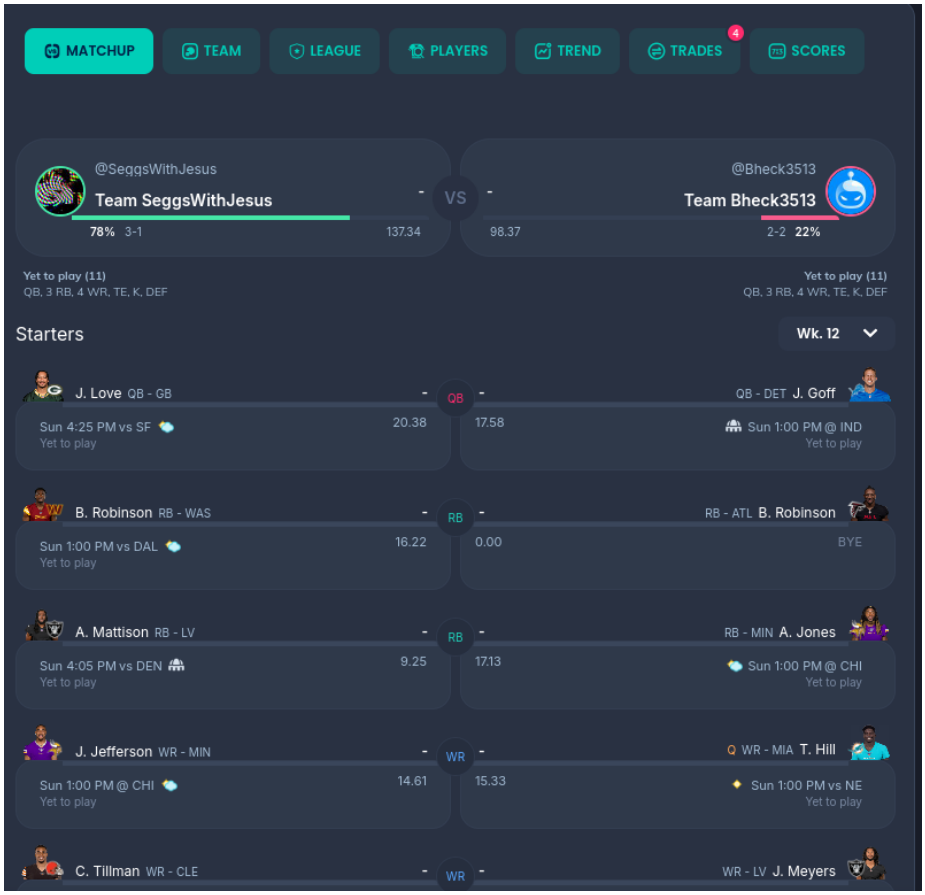

Figure 2: League Team Roster UI 1 - “Matchup”

using our hard-coded optimization problems to the user. I do think an easy-to-understand (potentially simplified) explanation of what was optimized would be very beneficial.

Trade Retrospective This would be a *great* way to draw people to site. Go through past trades and enable evaluation of their fairness/quality based on

- information available at the time
- how they actually affected teams afterwards

General Entity-Representation UIs

Player

We want to have a page that conveys a lot of information about a player at one time, including

- how that player's projections have been trending over the last week
- their performance history
  - raw performance metrics
  - fantasy points, using league settings
- context for past performances - Look into timeline UIs
  - their injuries (obviously)
  - teammates' injuries
  - team roster changes
  - strength of matchups
- performance projections
  - stats projections
  - ScoringConfig (stats projections) -> fantasy projections
- context for future
  - returning teammates
  - strength of matchups

NFL Teams

- fantasy points and stats allowed - to position - drill down into wide receiver types
- fantasy points and stats allowed - to play type - short/long pass, qb rush
- fantasy points and stats allowed - to (passing, rushing, receiving)
- defensive roster changes

League Teams

- 2-Team Comparison of current players' past performance
- 2-Team Comparison of current players' projected performance
- Able to change ranking measure for each of listed UIs
  - ranking measures
    - (past/projected performance) by (points/stats)
    - projected performance add'l inputs
      - can use raw-rankings projections for standard-scoring (PPR/HALF/STD), as there are *a lot* more opinions available than with stats projections
      - next week/ROS
  - UIs
    - PlayerListUI1 (Fig 1.)
    - "Matchup"-style UI (Fig 2.) with players' values next to them, and totals at op
      - some sort of summary comparison