Goals

1. Provide users with reccomendations for trades in their ESPN Fantasy Football leagues.
2. Provide users with a simple portal where they can view their team stats, player stats, and recent NFL news

Target Demographic

ESPN Fantasy Football players

1. Generally men
2. Age range between 16-28
3. Reside in the United States

Data

1. League data (id, name, teams)
2. Team data (id, name, owner name, record, logo, stats, waiver position)
3. Player data (id, name, position, injury status, stats, NFL team, fantasy team, league, player image)
4. News data (newest ESPN NFL news links)

Approach

1. First, pull the necessary data from the API and find the best method to sort it
2. Create classes to store the data before storing in the db
3. Create db models
    a. Schema
        i. User
            1. id (PK SERIAL)
            2. email (non-null unique text)
            3. username (non-null unique str(20)
            4. password (non-null password)
            5. leagues (relationship with League)
        ii. TeamStat
            1. id (PK SERIAL)
            2. team_id (non-null int FK Team.id)
            3. league_id (non-null int FK League.id)
            4. stat_name (non-null text)
            5. stat_value (non-null float)
            6. team (backref relationship w/ Team)
        iii. PlayerStat
            1. id (PK SERIAL)
            2. player_id (non-null int FK Player.id)
            3. league_id (non-null int FK League.id)
            4. stat_name (non-null text)
            5. stat_value (non-null float)
            6. player (backref relationship w/ Player)
        iv. Player
            1. id (PK SERIAL)
            2. player_id (non-null int)
            3. team_id (non-null int FK Team.id)
            4. league_id (non-null int FK League.id)
            5. first_name (non-null text)
            6. last_name (non-null text)
            7. pro_team (non-null text)
            8. position (non-null text)
            9. points (non-null float)
            10. projected_points (non-null float)
            11. position_rank (non-null int)
            12. grade (text)
            13. stats (relationship with PlayerStat)

14. team (backref realtionship w/ Team)
15. league (backref relationship w/ League)

v. Team
1. id (PK SERIAL)
2. team_id (non-null int)
3. league_id (non-null int FK League.id)
4. accronym (text)
5. location (text)
6. nickname (text)
7. logo_url (text)
8. record (text)
9. waiver_position (int)
10. players (relationship w/ Player)
11. stats (relationship w/ TeamStat)
12. league (backref relationship w/ League)

vi. League
1. id (PK SERIAL)
2. league_id (non-null int)
3. user_id (non-null int FK User.id)
4. year (non-null int)
5. espn_s2 (optional cookie data)
6. swid (optional cookie data)
7. players (relationship w/ Player)
8. teams (relationship w/ Team)
9. user (backref relationship w/ User)

4. Write methods for storing class data into db
5. Create Forms
6. Write authentication for users (passwords, and espn cookies need to be secured)
7. Create get trade reccomendation functionality
8. Create get news functionality
9. User flow
    a. Login/Sign up
    b. Go to leagues page
    c. Add league or click one of user's leagues
    d. Visit team profiles
    e. Visit player profiles
        i. Get trade reccomendations
    f. Visit user profile
        i. Change email
        ii. Change username
        iii. Change password
        iv. Delete account
    g. Visit settings
        i. Turn on dark mode

Stretch Goals

1. Add support for private leagues (espn_s2 & swid cookies)
2. Add ability to commit a trade from the website
3. Store commited trades in db, and get user feedback
4. Post trade inquiry on reddit