

Non-Functional Requirements

Performance

The queries made to the Yahoo Fantasy API are constantly monitored for any mistake, raising the respective exception whenever needed.

Methods from the Yahoo Fantasy Sports API Wrapper (YFPY) library were adapted in order to obtain faster and more precise data extraction.

Reliability

The Yahoo Fantasy Sports API Wrapper (YFPY) library was used to query the Yahoo Fantasy API and extract the necessary data.

A SetUp class was created to contain all the necessary information that the other modules will need to work properly.

Availability

The user may extract and visualize the data for the present season as well as for any past season, at any day of the week.

Maintainability

The program was divided into 4 different modules, SetUp, Extraction, Organizer, Visualizer, this way each functionality can be executed separately. The addition of future adaptations is easily done.

Abstract classes were used for each module that can be extended, this way, the basic organization for new sub-classes is maintained.

Recoverability

The extracted data will be saved into the project directory, being available for the user to access and use further on.

Capacity

Multiple leagues can be initiated and visualized, saved data contains league name for differentiability.

Serviceability

The Jupyter Notebook can be easily adapted to show another league's information.

Security

A Yahoo developer app must be created to safely connect with the Yahoo Fantasy API through OAuth authentication.

Data Integrity

Abstract classes were used for the data interactions between each module, the basic data flow between new sub-classes of each module is maintained.

Interoperability

The modules were created as .py files, so they can be imported to a Jupyter Notebook (.ipynb) or used within a python environment.

Usability

A Jupyter Notebook was created to act as a follow along for the necessary steps to use the Fantasy Intelligence Services. It includes the importing of the modules and the creation of each object in the expected order of usage. This makes it easier for a non-experienced user to interact with the program.