Carleton University

Fantasy Hockey Helper

Comp 4905 – Honours Project

By: Jeremy DenHartogh

Supervisor: Dr. Tony White

December 10, 2018

Abstract

Fantasy sports function on the idea that everybody has an opinion on which players they think are the best. However, a lot people are willing to acknowledge that their opinion is not the best out there, and as a result many look to experts or tools online to help them in their fantasy leagues. Fantasy hockey is no exception to this.

The goal of this project involves making competitive projections, as well as an easy to use, mobile-friendly web application to host these projections and to assist users in all the decisions involved in playing fantasy hockey. It will assist users with drafting, picking up free agents, and managing their roster on both a week-to-week and long-term basis.

The final application successfully achieved the goal of the project, in that its projections proved competitive with some of the best expert projections, as well as the physical web application achieved all the functionality goals. Furthermore, the goal to keep it mobile-friendly and easy to use was also achieved.

Acknowledgments

I would like to thank Dr. Tony White for all of his assistance with this project

Table of Contents

List of Figures

List of Tables

Introduction

Context

Fantasy sports in the age of the internet have continued to gain popularity and are now played by millions of people. Yahoo Fantasy Hockey alone has over 120000 leagues set up this year. The draw for fantasy sports can differ from person to person, some enjoy the extra depth, beyond just a viewing experience, that it can bring, while others may just enjoy the competition. Regardless of why they play however, everybody loves winning, and many seek any edge they can possibly get over their competition.

Outline

This report will contain the following sections: Motivation, Methodology, User Interface, Results and Conclusion. The Motivation section will discuss why I chose to do this project and give some background to the problem I had with existing fantasy hockey tools. The Methodology section will discuss the technology used for the project, the methods to obtain and store the required input data, the methods for projecting player statistics and developing an evaluation metric, and finally it will discuss the methods used to build the application. The User Interface section outlines the design decisions made when creating the user interface. The Results section compares the performance of the projections created against three other sources, and the performance of the application against the top fantasy hockey website, FantayHockeyGeek.com. Finally, the Conclusion section will summarize the report and make a final conclusion about the web application.

Motivation

Background

Plenty of experts have created their own projections to try and assist fantasy users, but few experts go past this. The applications that do are either slow and bulky, expensive, or take a lot of work to use. The most widely used tool today (FantasyHockeyGeek.com) has in some compacity all three of these weaknesses. This site will be the source of my main comparisons, and as part of my results I will evaluate my application's strengths and weaknesses when compared to this site.

To evaluate my projections, I will use the projections from Dobber, Yahoo Fantasy and Scott Cullen. Yahoo is generally considered a lower quality projection, Dobber's projections are considered mid-tier, and Scott Cullen is generally considered one of the best for projections. In the results section, more details on the comparative methods used will be outlined.

Objectives

This project has two primary objectives. The first objective is to gain an understanding of machine learning so that a model can be produced for every standard fantasy hockey statistic. These projections should have an accuracy level that is at the very least comparable to the projections of various fantasy hockey experts. The second primary objective is creating a mobile friendly application with many fantasy hockey assistance tools. Some sub-objectives here contain having the application run smoothly with little to no errors, an easy to use user interface, and storing no unnecessary date on the user.

Methodology

Technology

The primary portion of the application was written in Ruby on Rails. It contains the user interface for the clients, as well as the scripts necessary to pull information from the both the Yahoo and NHL API's and the scripts necessary to convert the projections into a format useable by the application.

The machine learning and statistical analysis was done via Python and the scikit-learn[1] machine learning python library. This library was used to help simplify the machine learning process, and this library proved to be more than capable. An attempt to use a Ruby library was used, however no libraries proved to be as good at machine learning, as the python options.

Amazon Web Services and their Cloud 9 service were also used to do all of my development. The use of these services allowed for coding to be done on any computer and allowed for quick deployment of the application to the web via an EC2 instance. Amazon's Route 53 service was also used to obtain the application's domain, allowing for a cleaner deployment.

OAuth was used to make API calls with Yahoo possible and was a key piece in allowing for the collection of the user's league and roster data. Specifically, the Ruby gems OAuth2[2] and omniauth-yahoo_auth[3] were used. The Yahoo and NHL API calls also both relied on the Ruby gem rest-client[4] which was used to make the API calls.

[1] https://scikit-learn.org/stable/
[2] https://github.com/oauth-xx/oauth2
[3] https://github.com/ikylin/omniauth-yahoo_auth
[4] https://github.com/rest-client/rest-client

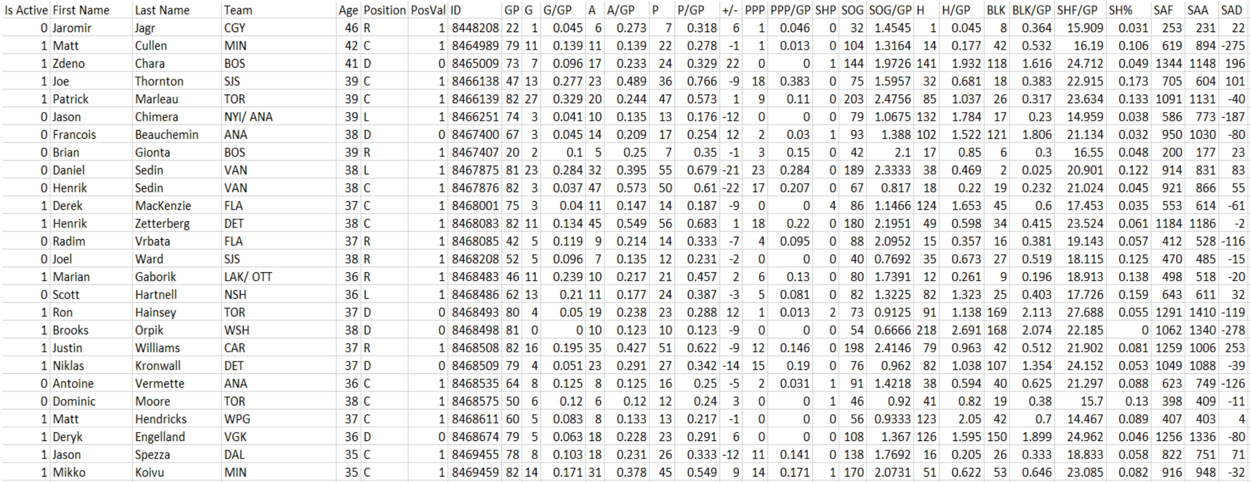
Data Collection

Yahoo's documented API calls[5] were used to collect all the necessary league and roster data once a user could be authenticated. This data included league information, league standings, the user's roster, the roster of every other team in the league, the leagues available free agents, and if the league was in pre-draft mode, the available players left in the draft. This data was not stored anywhere as it would require large databases to store it, and most of the data becomes useless the day after it is obtained. The data can also be obtained from Yahoo in under a few seconds, so no speed benefits would be gained by storing it.

For the NHL API the calls needed were found via multiple sources[6] [7], as well as by personal trial and error, to find some undocumented calls, and they all make use of the leagues publicly available data. Two different types of calls were made, the first set of which were used to obtain various tracked players statistics. All player data was stored in a handful of CSV files that would be used later by the machine learning algorithms. These CSV files included the players stats for each of the 2015-2016, 2016-2017 and 2017-2018 seasons, as well CSV files with the three-year totals and averages for the player at the end of each of those seasons. More stats were stored than necessary to allow for every stat that will be projected to read from the same set of data.

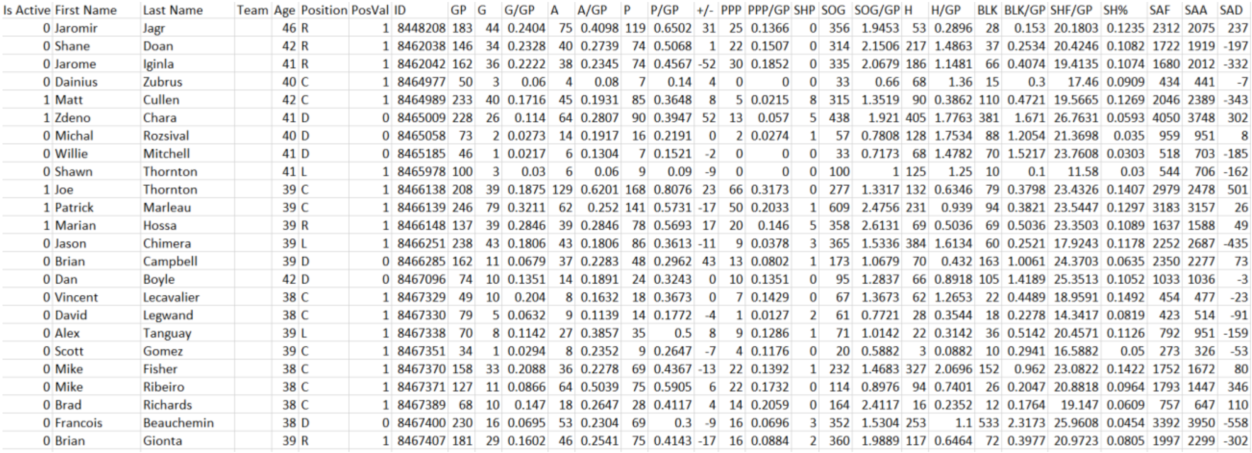
The second set of calls made to the NHL API, collected teams schedule information, which allowed for tracking the days every player plays. The team schedule information was also stored in csv file, to allow for the application to quickly obtain a player's schedule information. This was important information for the application as it shows users how many games and on what days players play each week. The information stored here included what days a team played, who they played, and whether they played on the road or at home, as all of this information could be considered valuable to user.

[5] https://developer.yahoo.com/fantasysports/guide/
[6] https://github.com/dword4/nhlapi
[7] https://www.reddit.com/r/hockey/comments/7p7o70/documenting_the_nhl_api/dsfuy54/



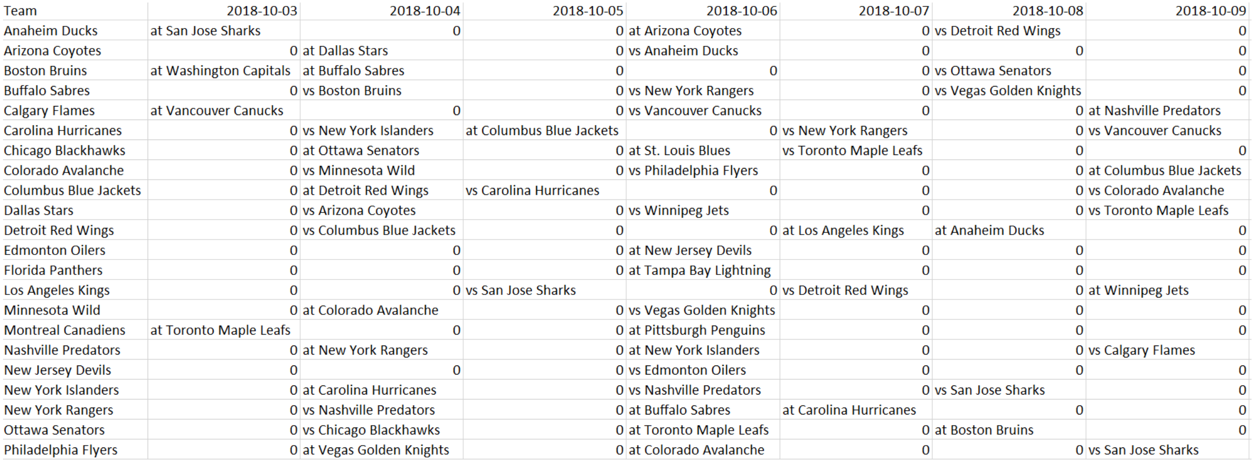
Is Active	First Name	Last Name	Team	Age	Position	PosVal	ID	GP	G	G/GP	A	A/GP	P	P/GP	+/-	PPP	PPP/GP	SHP	SOG	SOG/GP	H	H/GP	BLK	BLK/GP	SHF/GP	SH%	SAF	SAA	SAD
0	Jaromir	Jagr	CGY	46	R	1	8448208	22	1	0.045	6	0.273	7	0.318	6	1	0.046	0	32	1.4545	1	0.045	8	0.364	15.909	0.031	253	231	22
1	Matt	Cullen	MIN	42	C	1	8464989	79	11	0.139	11	0.139	22	0.278	-1	1	0.013	0	104	1.3164	14	0.177	42	0.532	16.19	0.106	619	894	-275
1	Zdeno	Chara	BOS	41	D	0	8465009	73	7	0.096	17	0.233	24	0.329	22	0	0	1	144	1.9726	141	1.932	118	1.616	24.712	0.049	1344	1148	196
1	Joe	Thornton	SJS	39	C	1	8466138	47	13	0.277	23	0.489	36	0.766	-9	18	0.383	0	75	1.5957	32	0.681	18	0.383	22.915	0.173	705	604	101
1	Patrick	Marleau	TOR	39	C	1	8466139	82	27	0.329	20	0.244	47	0.573	1	9	0.11	0	203	2.4756	85	1.037	26	0.317	23.634	0.133	1091	1131	-40
0	Jason	Chimera	NYI/ ANA	39	L	1	8466251	74	3	0.041	10	0.135	13	0.176	-12	0	0	0	79	1.0675	132	1.784	17	0.23	14.959	0.038	586	773	-187
0	Francois	Beauchemin	ANA	38	D	0	8467400	67	3	0.045	14	0.209	17	0.254	12	2	0.03	1	93	1.388	102	1.522	121	1.806	21.134	0.032	950	1030	-80
0	Brian	Gionta	BOS	39	R	1	8467407	20	2	0.1	5	0.25	7	0.35	-1	3	0.15	0	42	2.1	17	0.85	6	0.3	16.55	0.048	200	177	23
0	Daniel	Sedin	VAN	38	L	1	8467875	81	23	0.284	32	0.395	55	0.679	-21	23	0.284	0	189	2.3333	38	0.469	2	0.025	20.901	0.122	914	831	83
0	Henrik	Sedin	VAN	38	C	1	8467876	82	3	0.037	47	0.573	50	0.61	-22	17	0.207	0	67	0.817	18	0.22	19	0.232	21.024	0.045	921	866	55
1	Derek	MacKenzie	FLA	37	C	1	8468001	75	3	0.04	11	0.147	14	0.187	-9	0	0	4	86	1.1466	124	1.653	45	0.6	17.453	0.035	553	614	-61
1	Henrik	Zetterberg	DET	38	C	1	8468083	82	11	0.134	45	0.549	56	0.683	1	18	0.22	0	180	2.1951	49	0.598	34	0.415	23.524	0.061	1184	1186	-2
0	Radim	Vrbata	FLA	37	R	1	8468085	42	5	0.119	9	0.214	14	0.333	-7	4	0.095	0	88	2.0952	15	0.357	16	0.381	19.143	0.057	412	528	-116
0	Joel	Ward	SJS	38	R	1	8468208	52	5	0.096	7	0.135	12	0.231	-2	0	0	0	40	0.7692	35	0.673	27	0.519	18.115	0.125	470	485	-15
1	Marian	Gaborik	LAK/ OTT	36	R	1	8468483	46	11	0.239	10	0.217	21	0.457	2	6	0.13	0	80	1.7391	12	0.261	9	0.196	18.913	0.138	498	518	-20
0	Scott	Hartnell	NSH	36	L	1	8468486	62	13	0.21	11	0.177	24	0.387	-3	5	0.081	0	82	1.3225	82	1.323	25	0.403	17.726	0.159	643	611	32
1	Ron	Hainsey	TOR	37	D	0	8468493	80	4	0.05	19	0.238	23	0.288	12	1	0.013	2	73	0.9125	91	1.138	169	2.113	27.688	0.055	1291	1410	-119
1	Brooks	Orpik	WSH	38	D	0	8468498	81	0	0	10	0.123	10	0.123	-9	0	0	0	54	0.6666	218	2.691	168	2.074	22.185	0	1062	1340	-278
1	Justin	Williams	CAR	37	R	1	8468508	82	16	0.195	35	0.427	51	0.622	-9	12	0.146	0	198	2.4146	79	0.963	42	0.512	21.902	0.081	1259	1006	253
1	Niklas	Kronwall	DET	37	D	0	8468509	79	4	0.051	23	0.291	27	0.342	-14	15	0.19	0	76	0.962	82	1.038	107	1.354	24.152	0.053	1049	1088	-39
0	Antoine	Vermette	ANA	36	C	1	8468535	64	8	0.125	8	0.125	16	0.25	-5	2	0.031	1	91	1.4218	38	0.594	40	0.625	21.297	0.088	623	749	-126
0	Dominic	Moore	TOR	38	C	1	8468575	50	6	0.12	6	0.12	12	0.24	3	0	0	1	46	0.92	41	0.82	19	0.38	15.7	0.13	398	409	-11
1	Matt	Hendricks	WPG	37	C	1	8468611	60	5	0.083	8	0.133	13	0.217	-1	0	0	0	56	0.9333	123	2.05	42	0.7	14.467	0.089	407	403	4
1	Deryk	Engelland	VGK	36	D	0	8468674	79	5	0.063	18	0.228	23	0.291	6	0	0	0	108	1.367	126	1.595	150	1.899	24.962	0.046	1256	1336	-80
1	Jason	Spezza	DAL	35	C	1	8469455	78	8	0.103	18	0.231	26	0.333	-12	11	0.141	0	138	1.7692	16	0.205	26	0.333	18.833	0.058	822	751	71
1	Mikko	Koivu	MIN	35	C	1	8469459	82	14	0.171	31	0.378	45	0.549	9	14	0.171	1	170	2.0731	51	0.622	53	0.646	23.085	0.082	916	948	-32

FIGURE 1 – Sample of single year data



Is Active	First Name	Last Name	Team	Age	Position	PosVal	ID	GP	G	G/GP	A	A/GP	P	P/GP	+/-	PPP	PPP/GP	SHP	SOG	SOG/GP	H	H/GP	BLK	BLK/GP	SHF/GP	SH%	SAF	SAA	SAD
0	Jaromir	Jagr		46	R	1	8448208	183	44	0.2404	75	0.4098	119	0.6502	31	25	0.1366	0	356	1.9453	53	0.2896	28	0.153	20.1803	0.1235	2312	2075	237
0	Shane	Doan		42	R	1	8462038	146	34	0.2328	40	0.2739	74	0.5068	1	22	0.1507	0	314	2.1506	217	1.4863	37	0.2534	20.4246	0.1082	1722	1919	-197
0	Jarome	Iginla		41	R	1	8462042	162	36	0.2222	38	0.2345	74	0.4567	-52	30	0.1852	0	335	2.0679	186	1.1481	66	0.4074	19.4135	0.1074	1680	2012	-332
0	Dainius	Zubrus		40	C	1	8464977	50	3	0.06	4	0.08	7	0.14	4	0	0	0	33	0.66	68	1.36	15	0.3	17.46	0.0909	434	441	-7
1	Matt	Cullen		42	C	1	8464989	233	40	0.1716	45	0.1931	85	0.3648	8	5	0.0215	8	315	1.3519	90	0.3862	110	0.4721	19.5665	0.1269	2046	2389	-343
1	Zdeno	Chara		41	D	0	8465009	228	26	0.114	64	0.2807	90	0.3947	52	13	0.057	5	438	1.921	405	1.7763	381	1.671	26.7631	0.0593	4050	3748	302
0	Michal	Rozsival		40	D	0	8465058	73	2	0.0273	14	0.1917	16	0.2191	0	2	0.0274	1	57	0.7808	128	1.7534	88	1.2054	21.3698	0.035	959	951	8
0	Willie	Mitchell		41	D	0	8465185	46	1	0.0217	6	0.1304	7	0.1521	-2	0	0	0	33	0.7173	68	1.4782	70	1.5217	23.7608	0.0303	518	703	-185
0	Shawn	Thornton		41	L	1	8465978	100	3	0.03	6	0.06	9	0.09	-9	0	0	0	100	1	125	1.25	10	0.1	11.58	0.03	544	706	-162
1	Joe	Thornton		39	C	1	8466138	208	39	0.1875	129	0.6201	168	0.8076	23	66	0.3173	0	277	1.3317	132	0.6346	79	0.3798	23.4326	0.1407	2979	2478	501
1	Patrick	Marleau		39	C	1	8466139	246	79	0.3211	62	0.252	141	0.5731	-17	50	0.2033	1	609	2.4756	231	0.939	94	0.3821	23.5447	0.1297	3183	3157	26
1	Marian	Hossa		39	R	1	8466148	137	39	0.2846	39	0.2846	78	0.5693	17	20	0.146	5	358	2.6131	69	0.5036	69	0.5036	23.3503	0.1089	1637	1588	49
0	Jason	Chimera		39	L	1	8466251	238	43	0.1806	43	0.1806	86	0.3613	-11	9	0.0378	3	365	1.5336	384	1.6134	60	0.2521	17.9243	0.1178	2252	2687	-435
0	Brian	Campbell		39	D	0	8466285	162	11	0.0679	37	0.2283	48	0.2962	43	13	0.0802	1	173	1.0679	70	0.432	163	1.0061	24.3703	0.0635	2350	2277	73
0	Dan	Boyle		42	D	0	8467096	74	10	0.1351	14	0.1891	24	0.3243	0	10	0.1351	0	95	1.2837	66	0.8918	105	1.4189	25.3513	0.1052	1033	1036	-3
0	Vincent	Lecavalier		38	C	1	8467329	49	10	0.204	8	0.1632	18	0.3673	0	7	0.1429	0	67	1.3673	62	1.2653	22	0.4489	18.9591	0.1492	454	477	-23
0	David	Legwand		38	C	1	8467330	79	5	0.0632	9	0.1139	14	0.1772	-4	1	0.0127	2	61	0.7721	28	0.3544	18	0.2278	14.3417	0.0819	423	514	-91
0	Alex	Tanguay		39	L	1	8467338	70	8	0.1142	27	0.3857	35	0.5	8	9	0.1286	1	71	1.0142	22	0.3142	36	0.5142	20.4571	0.1126	792	951	-159
0	Scott	Gomez		39	C	1	8467351	34	1	0.0294	8	0.2352	9	0.2647	-7	4	0.1176	0	20	0.5882	3	0.0882	10	0.2941	16.5882	0.05	273	326	-53
0	Mike	Fisher		38	C	1	8467370	158	33	0.2088	36	0.2278	69	0.4367	-13	22	0.1392	1	232	1.4683	327	2.0696	152	0.962	23.0822	0.1422	1752	1672	80
0	Mike	Ribeiro		38	C	1	8467371	127	11	0.0866	64	0.5039	75	0.5905	6	22	0.1732	0	114	0.8976	94	0.7401	26	0.2047	20.8818	0.0964	1793	1447	346
0	Brad	Richards		38	C	1	8467389	68	10	0.147	18	0.2647	28	0.4117	4	14	0.2059	0	164	2.4117	16	0.2352	12	0.1764	19.147	0.0609	757	647	110
0	Francois	Beauchemin		38	D	0	8467400	230	16	0.0695	53	0.2304	69	0.3	-9	16	0.0696	3	352	1.5304	253	1.1	533	2.3173	25.9608	0.0454	3392	3950	-558
0	Brian	Gionta		39	R	1	8467407	181	29	0.1602	46	0.2541	75	0.4143	-17	16	0.0884	2	360	1.9889	117	0.6464	72	0.3977	20.9723	0.0805	1997	2299	-302

FIGURE 2 – Sample of three-year average data



Team	2018-10-03	2018-10-04	2018-10-05	2018-10-06	2018-10-07	2018-10-08	2018-10-09
Anaheim Ducks	at San Jose Sharks	0	0	at Arizona Coyotes	0	vs Detroit Red Wings	0
Arizona Coyotes	0	at Dallas Stars	0	vs Anaheim Ducks	0	0	0
Boston Bruins	at Washington Capitals	at Buffalo Sabres	0	0	0	vs Ottawa Senators	0
Buffalo Sabres	0	vs Boston Bruins	0	vs New York Rangers	0	vs Vegas Golden Knights	0
Calgary Flames	at Vancouver Canucks	0	0	vs Vancouver Canucks	0	0	at Nashville Predators
Carolina Hurricanes	0	vs New York Islanders	at Columbus Blue Jackets	0	vs New York Rangers	0	vs Vancouver Canucks
Chicago Blackhawks	0	at Ottawa Senators	0	at St. Louis Blues	vs Toronto Maple Leafs	0	0
Colorado Avalanche	0	vs Minnesota Wild	0	vs Philadelphia Flyers	0	0	at Columbus Blue Jackets
Columbus Blue Jackets	0	at Detroit Red Wings	vs Carolina Hurricanes	0	0	0	vs Colorado Avalanche
Dallas Stars	0	vs Arizona Coyotes	0	vs Winnipeg Jets	0	0	vs Toronto Maple Leafs
Detroit Red Wings	0	vs Columbus Blue Jackets	0	0	at Los Angeles Kings	at Anaheim Ducks	0
Edmonton Oilers	0	0	0	at New Jersey Devils	0	0	0
Florida Panthers	0	0	0	at Tampa Bay Lightning	0	0	0
Los Angeles Kings	0	0	vs San Jose Sharks	0	vs Detroit Red Wings	0	at Winnipeg Jets
Minnesota Wild	0	at Colorado Avalanche	0	vs Vegas Golden Knights	0	0	0
Montreal Canadiens	at Toronto Maple Leafs	0	0	at Pittsburgh Penguins	0	0	0
Nashville Predators	0	at New York Rangers	0	at New York Islanders	0	0	vs Calgary Flames
New Jersey Devils	0	0	0	vs Edmonton Oilers	0	0	0
New York Islanders	0	at Carolina Hurricanes	0	vs Nashville Predators	0	vs San Jose Sharks	0
New York Rangers	0	vs Nashville Predators	0	at Buffalo Sabres	at Carolina Hurricanes	0	0
Ottawa Senators	0	vs Chicago Blackhawks	0	at Toronto Maple Leafs	0	at Boston Bruins	0
Philadelphia Flyers	0	at Vegas Golden Knights	0	at Colorado Avalanche	0	0	vs San Jose Sharks

FIGURE 3 – Sample of team schedule data

Projecting Skater Statistics

First Attempt

For my first attempt at projecting skater statistics I attempted to use the ruby gem ruby-linear_regression[8]. While it helped with developing models, this gem was slow, and produced unreliable data. After further research I decided that Ruby was not a good language for machine learning, and that it fell short in compared to the available Python libraries that were available. After failing to get the desired results and level of accuracy from ruby, further research led to a conclusion that the best course of action was to switch languages to Python.

Second Attempt

For my second attempt python was used as well as the previously mentioned scikit-learn Python library. Next a model had to be determined for each statistic that was to be predicted. First, the input data had to be sorted in a way such that a model could be developed. To do this I combined the data for a year, with the three-year totals/averages for that year as the input data, and the data for the next year as the output data. As an example, 2016-2017 skater statistics combined with 2014-2017 skater statistics to create the input data, and the output data was the 2017-2018 skater statistics. By doing this the possibility was left open that depending on the stat being predicted, a combination of single and three-year statistics could be used. After cutting out players who didn’t play in both the predictor and predicted years, I was left with just under 1000 players rows to develop a model from.

The next step was to choose an algorithm. Since a quantity was being predicted, with a limited number of samples, two specific algorithms stood out, Ridge Regression, and Support Vector Regression. After experimenting with both, Ridge Regression appeared to produce more accurate predictions, and ran at a faster speed, so it was selected as the algorithm of choice.

[8] https://github.com/daugaard/linear-regression

Ridge regression is a generalized linear model that is similar to Ordinary Least Squares, in which it fits a linear model with coefficients in attempt to minimize the residual sum of squares between the inputted data and the resulting prediction, however it differs by imposing a penalty on the size of the coefficients.

With the learning algorithm selected, the next objective was to obtain a variable selection algorithm. A version of Stepwise Regression was selected here, as it proved the most successful in trials at obtaining a predictive subset of variables. Due to some limitations in the scikit-learn library however, some slight variances to regular Stepwise Regression were made, but the general concept remained the same.

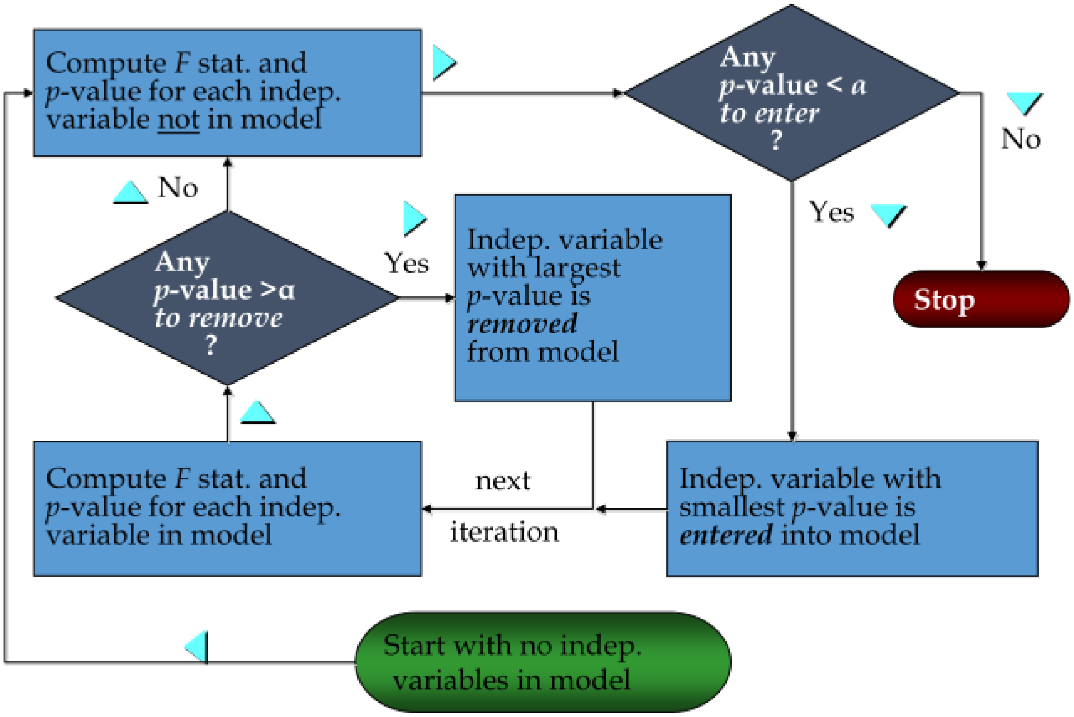


FIGURE 4 – Stepwise regression flowchart

With all the algorithms in place, the next step was to start actually selecting the variables. To do this the data was randomly split up into two groups, training and validation. The potential models (selected via Stepwise Regression) were then fit (using Ridge Regression) using the training data, and then predictions were made using the validation data. The necessary scores were then recorded. To reduce randomization, I ran over two thousand tests on each model, randomly selecting different subsets of the data to serve as the training and validation group. The averages of all their scores were stored, and compared, to eventually work my way to the best possible subset of available variables and eventually a model for every skater statistic.

One other thing that became clear during all of this, was more accuracy could be obtained by projecting the players per game averages, rather than season total. This likely is because the amount of games a player plays differs from year to year based on injuries and other factors. As a result, all projections were made on the skaters per game averages, and then extrapolated based on the assumption that every player would play 82 games. Obviously not every player will play 82 games but trying to predict who will get injured is a near impossible task. (The application can adjust its projections if a player does get injured during the season, but that will be discussed later in this paper.)

With all the models and algorithms in place, all that was left for the skaters was creating the projections. Using the skater data from 2017-2018 seasons, and the three-year total/average data from 2015-2018, statistical projections were produced for the 2018-2019 season for every active skater. These projections were then stored initially in one large CSV file, but then also split and stored into separate CSV files depending on position eligibility.

Projecting Goalie Statistics

First Attempt

For the first attempt at projecting goalie statistics, the methods outlined in Projecting Skater Statistics were used. This did not work for multiple reasons, the first of which is the small sample size. There are only 31 NHL teams, with two goalies per team, meaning that there are 62 professional goaltenders at any given time. Additionally, most teams have only one goalie they trust to start most games. For example, only 53 goalies averaged 20 or more games per season over the last three seasons and of these 53 only 15 averaged more than 50 games per season. This caused an issue with the machine learning because there just simply was not enough data to work with.

The second issue with this method was that two of the main stats for goalies that needed to be predicted, wins and shutouts, are dependent on how many games goalies play. Games played for goalies can often fluctuate greatly from season to season, making predicting how many games they will play very difficult.

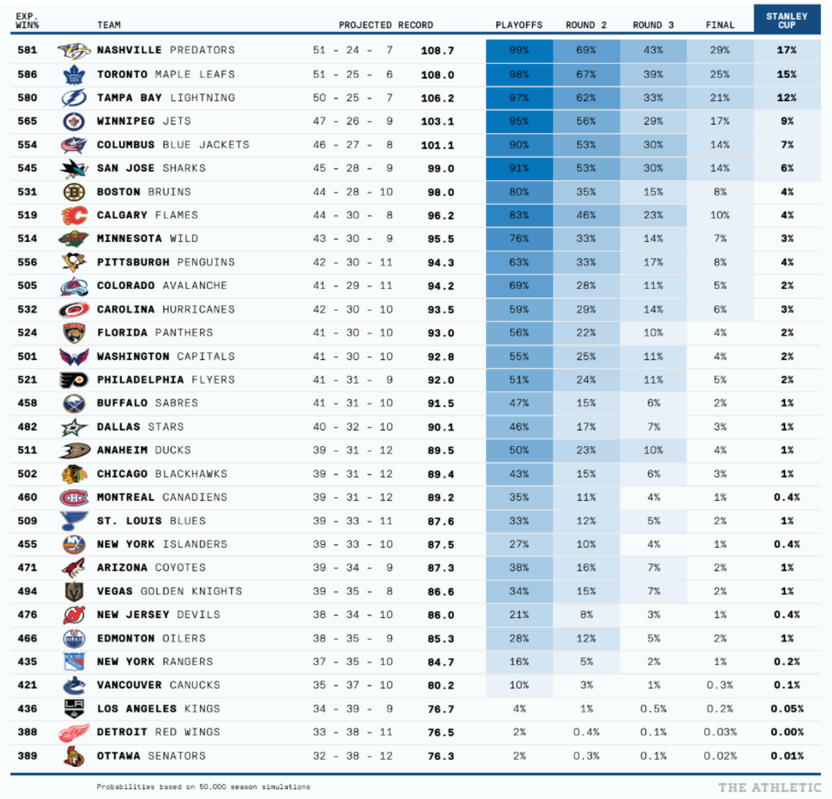
Finally, three of the four goaltending stats (wins, shutouts and goals against average) depend very highly on the performance of the team they play for. For example, a goalie who plays for a team that allows an above average amount of shots will likely have a high goals against average, and smaller wins and shutouts per games played ratios. Other team factors like how good their offense is, how many penalties they take, and how many high danger shots they allow.

Second Attempt

With the second attempt a decision was made to take projecting goalies in a different direction. The new plan for projecting goalies would come in three steps. Firstly, project use information known about each lineup to project how many games each goalie would play. Secondly, use existing team performance projections to obtain projections on how many wins each team will get, and how many shots they are expected to face per game. Finally, use the existing team stats, projected games played and the goalies three-year average even strength save percentage to project the stats for every goaltender.

The first step, to project how many games each goalie would play, mostly had to be done manually. Getting these projections required research into every team individually and determining what share of the games all of their goalies would get. For teams who have the same starting goaltender as last, this task was fairly easy, since it is easy to just project that the goalies split would be relatively the same. For teams who acquired a new goaltender, or whose goalie was injured the previous season, this task took a little more research. For the most part, with the teams who had injured goalies the previous year, looking at the past three years instead can help project who will get the larger share of the games. For the other teams, where they acquired a new goaltender, or lost an old goaltender, the research mostly came from media reports on who the experts think will get the larger share of the games.

For the second step, team projections were needed, however, projecting how well teams would do was slightly outside the scope of this project and could easily be a project on its own. Instead, the team projections from industry expert Dom Luszczyszyn[9] were used since he updates his projections consistently enough throughout the season that they would allow for updates to the projections mid-season, and he also projects total wins, rather than just total points in the standings. To obtain the expected shots against per game average, the average shots against per game over the last three seasons for each team was used.



EXP. WIN%	TEAM	PROJECTED RECORD		PLAYOFFS	ROUND 2	ROUND 3	FINAL	STANLEY CUP
581	**NASHVILLE** PREDATORS	51 - 24 - 7	**108.7**	99%	69%	43%	29%	**17%**
586	**TORONTO** MAPLE LEAFS	51 - 25 - 6	**108.0**	98%	67%	39%	25%	**15%**
580	**TAMPA BAY** LIGHTNING	50 - 25 - 7	**106.2**	97%	62%	33%	21%	**12%**
565	**WINNIPEG** JETS	47 - 26 - 9	**103.1**	95%	56%	29%	17%	**9%**
554	**COLUMBUS** BLUE JACKETS	46 - 27 - 8	**101.1**	90%	53%	30%	14%	**7%**
545	**SAN JOSE** SHARKS	45 - 28 - 9	**99.0**	91%	53%	30%	14%	**6%**
531	**BOSTON** BRUINS	44 - 28 - 10	**98.0**	80%	35%	15%	8%	**4%**
519	**CALGARY** FLAMES	44 - 30 - 8	**96.2**	83%	46%	23%	10%	**4%**
514	**MINNESOTA** WILD	43 - 30 - 9	**95.5**	76%	33%	14%	7%	**3%**
556	**PITTSBURGH** PENGUINS	42 - 30 - 11	**94.3**	63%	33%	17%	8%	**4%**
505	**COLORADO** AVALANCHE	41 - 29 - 11	**94.2**	69%	28%	11%	5%	**2%**
532	**CAROLINA** HURRICANES	42 - 30 - 10	**93.5**	59%	29%	14%	6%	**3%**
524	**FLORIDA** PANTHERS	41 - 30 - 10	**93.0**	56%	22%	10%	4%	**2%**
501	**WASHINGTON** CAPITALS	41 - 30 - 10	**92.8**	55%	25%	11%	4%	**2%**
521	**PHILADELPHIA** FLYERS	41 - 31 - 9	**92.0**	51%	24%	11%	5%	**2%**
458	**BUFFALO** SABRES	41 - 31 - 10	**91.5**	47%	15%	6%	2%	**1%**
482	**DALLAS** STARS	40 - 32 - 10	**90.1**	46%	17%	7%	3%	**1%**
511	**ANAHEIM** DUCKS	39 - 31 - 12	**89.5**	50%	23%	10%	4%	**1%**
502	**CHICAGO** BLACKHAWKS	39 - 31 - 12	**89.4**	43%	15%	6%	3%	**1%**
460	**MONTREAL** CANADIENS	39 - 31 - 12	**89.2**	35%	11%	4%	1%	**0.4%**
509	**ST. LOUIS** BLUES	39 - 33 - 11	**87.6**	33%	12%	5%	2%	**1%**
455	**NEW YORK** ISLANDERS	39 - 33 - 10	**87.5**	27%	10%	4%	1%	**0.4%**
471	**ARIZONA** COYOTES	39 - 34 - 9	**87.3**	38%	16%	7%	2%	**1%**
494	**VEGAS** GOLDEN KNIGHTS	39 - 35 - 8	**86.6**	34%	15%	7%	2%	**1%**
476	**NEW JERSEY** DEVILS	38 - 34 - 10	**86.0**	21%	8%	3%	1%	**0.4%**
466	**EDMONTON** OILERS	38 - 35 - 9	**85.3**	28%	12%	5%	2%	**1%**
435	**NEW YORK** RANGERS	37 - 35 - 10	**84.7**	16%	5%	2%	1%	**0.2%**
421	**VANCOUVER** CANUCKS	35 - 37 - 10	**80.2**	10%	3%	1%	0.3%	**0.1%**
436	**LOS ANGELES** KINGS	34 - 39 - 9	**76.7**	4%	1%	0.5%	0.2%	**0.05%**
388	**DETROIT** RED WINGS	33 - 38 - 11	**76.5**	2%	0.4%	0.1%	0.03%	**0.00%**
389	**OTTAWA** SENATORS	32 - 38 - 12	**76.3**	2%	0.3%	0.1%	0.02%	**0.01%**

Probabilities based on 50,000 season simulations

THE ATHLETIC

[9] https://theathletic.com/565136/2018/11/20/2018-19-nhl-point-projections-and-playoff-probabilities-updated-daily/

FIGURE 5 – Dom Luszczyszyn project standings

With the required stats obtained, projections for goalies could be made. For wins, projections were created from a combination of expected games played for the goalie, and expected win total of the team. A simple direct relationship was used here, where goalies were projected to get the exact same share of the wins as they did games played. For example, if a goalie plays half of a team's games, they will be projected to get half of a teams wins.

The next stat to project was shutouts. For this stat the ratio of wins to shutouts for all goalies over the last three years was calculated. That ratio was then used in combination with the projected wins for the goaltender to project how many shutouts they were expected to achieve this upcoming season.

The third stat to project was save percentage. The stat with the best correlation to save percentage, was the goalie's three-year average even strength save percentage. This is because powerplay and penalty-kill save percentages are very unreliable from year to year and have almost no correlation. So instead a correlation between the three-year average even strength save percentage, and the three-year average total save percentage was found, and then used to project the save percentage of each goaltender for the upcoming season.

Finally, the last statistics to calculate was goals against average. For this stat a simple formula was used ((1-expected save percentage) x (teams expected shots against)). This formula seemed the most logical way to project goals against average, since goals against average is simply the number of goals each goalie is expected to let in during the average game. After this projection was completed, this and the other projections were all stored in csv file.

Evaluating Players Value

To compare the players to one each other a metric had to be created to assign value to each statistic. Since every category has equal value in standard Yahoo fantasy hockey leagues, every category was assigned the same value when developing the value metric. There are also three main positions in fantasy hockey, forwards, defensemen, and goalies.

For evaluating a player's worth in every category, all players were grouped by position, and then their stats were only compared to players who play the same position. This was necessary since goalies have their own categories, and forwards tend to perform better in most categories when compared to defensemen. Since in standard Yahoo fantasy hockey leagues you must have at least four defensemen, it made sense logically to split them up from the forwards, and evaluate their worth compared to other defensemen.

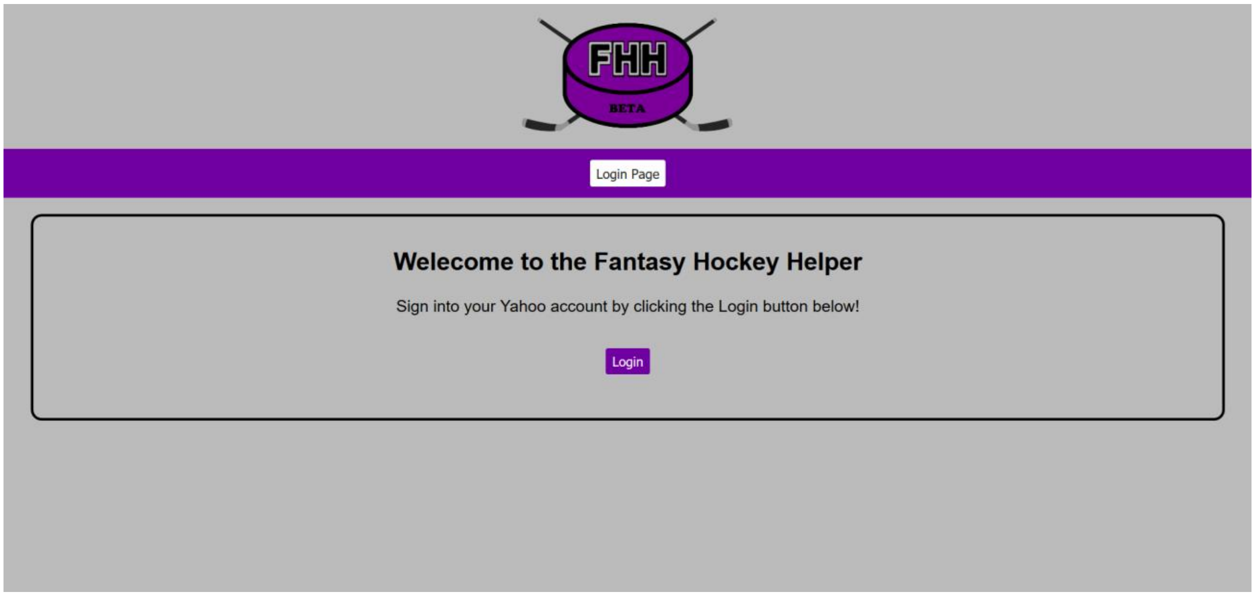
The next step in creating a metric, was to calculate the mean and standard deviation in every stat category for every position group. Then it was calculated how many standard deviations from the mean every player was, this number was then added or subtracted from the player's score. Once done for every category, the scores were altered to fit onto a scale from 0-100. The player in each group with lowest score, was given a 0, and the one with highest score was assigned a value of 100, every other player was sorted somewhere in the middle.

Since all player groups were now assigned values from 0-100, the three groups could now be compared to one another, with their scores representing how much better they were on average when compared to players of the same group as them. This would allow for the creation of rankings, which would be necessary when recommending to users which moves or draft selections they should make.

Creating the Base Application

When creating the application, the first steps were to choose an environment and a programming language. For the environment, an AWS Cloud9 server was used, as it allowed for the ability to code from any computer, which allowed for more flexibility when creating the application. For the language, Ruby on Rails was selected, this was because of personal experience with the language, allowing for less time to be spent on learning syntax.

Once the application was created, a Git repository was setup to allow for version control with my application. Then using Amazon's Route 53 tool, I set up a public domain for my web application, so that testing would be made simpler. After this I designed a temporary home page, that would eventually serve as the login page, once OAuth with Yahoo was set up.



Welcome to the Fantasy Hockey Helper

Sign into your Yahoo account by clicking the Login button below!

FIGURE 6 – Login Page for Fantasy Hockey Helper

Connecting the Application to Yahoo

To allow for the application to obtain a user’s league and roster information, it was a requirement to connect to Yahoo via OAuth2[10]. OAuth2 is an authorization protocol that allows for applications to obtain read or write privileges from another application. In the case of gaining a user’s league and roster information from Yahoo, read privileges of the user’s fantasy sports data was needed. To gain these privileges the URL for the application, FantasyHockeyHelper.com, was registered as an application to the Yahoo Developer Network, with request to gain read privileges to fantasy sports data.

Once the application was registered, the ruby gems previously mentioned (oauth2 and omniauth-yahoo_auth) were used to login the user and obtain a token. This token was then saved into a cookie on the user’s browser and set with a one-hour expiry time. After the token was obtained, the application would use the user’s token to get the required data from Yahoo. The decision to store the user’s token in a cookie rather than in a database was made because it allowed for less data to be stored on the server, and allowed for quicker API calls, since the application would not need to search the database for a token before making the call.

Obtaining a User’s League Information

With the user authenticated, the next step was to use their token to make Yahoo API calls, to find all the data necessary on the user and their leagues. Yahoo had a good amount of documentation for all their API calls, which helped a lot for obtaining the majority of the information the application would need. Some API calls that were needed however were not fully documented so some trial and error was necessary to get the specific information the application required.

[10] https://oauth.net/2/

The first API call the application would use was one that did not have a lot of useful documentation, thus requiring some work into getting an API call that would obtain the information required. This is the call that eventually was used: `"https://fantasysports.yahooapis.com/fantasy/v2/users;use_login=1/games;game_keys=nhl/leagues;season=2018"`. The goal of this call was to obtain a list of all the leagues the user is apart of as well as the league id of this league, so that the user could easily select which league they want to view.

The league information page makes two Yahoo API calls, the first of which obtains a response variable that contains all the basic information about the current league the user is viewing. It obtains league settings, like the roster positions, stat categories and schedule information. The seconds API call obtains a list of teams in the league. The controller then searches through this list and finds the team the user owns and stores in a cookie the id of this team.

The standings page has only one API call and it obtains information about all the teams in the league, however the controller only stores the id of these teams, and their current record. It then sorts the teams by their record, to obtain the standings of the league. This page serves mostly just as a list of the links for the user to view other teams.

Building the Rankings Tool

The first tool for the application was the rankings tool. This tool includes a master list of every player, and their corresponding value ranking. To obtain this list it simply reads the data from the available CSV projection files and creates a table. As a result, it is the only one of the tools that do not rely on Yahoo, and thus is the only tool available to players before a user selects their league. This tool can change which CSV it reads from, depending on the position the user chooses view. Overall, it is the simplest of all the tools, but is also the most available, as you do not need to be in a league to use it.

Building the Free Agent / Draft tool

The next tool for the application built was the Free Agent / Draft Tool. While these were initially planned to be two separate tools, after research into the Yahoo API calls, it made more sense to combine them into one dynamic tool. The tool is dynamic because depending on the leagues state (pre-draft, draft or post-draft), the way the tool works will alter slightly. The Yahoo API call used for all forms of this tool returns a list of all currently available players in the league.

The first step of building the free agent and draft tools, was to obtain a list of available players. To do this the controller makes use of Yahoo API calls, however it only obtains the available players in groups of 25 players at a time. As a result of this, the tool has to make multiple API calls, and merge the results together. The controller also uses the available player projection CSV files and creates a list of all its projections. The next step is to merge these two lists together and create a 2D list of available players and their projections.

Once the available players list is complete, the next list created was the schedule information for each player. This is done by looking at a player's team and using the schedule CSV to obtain information on what days in the viewed week the player plays. It also collects how many games they play total in each week, and how many games remaining in the week they have left.

The application also has multiple routes, allowing for the user to filter by specific position. Depending on the route provided, the Yahoo API calls will add an extra requirement, causing the returning available players list to only be players that are eligible to play in the provided position. This route will also change which CSV the controller reads from. The routes have no impact on the schedule portion, since the position of a player has zero impact on this list.

By default, the tool will collect information from the schedule CSV on the current ongoing week. The user however has the option to select a different week, doing this causes the application to read from a different subsection of the schedule CSV file. If a week other than the current ongoing week is selected, the tool no longer tracks how many games are remaining in the week, since the week is not in progress and it would not make sense to collect this info.

The tool also creates two recommendations for the user on what player it thinks the player should pick up. The first recommendation is a long-term recommendation, and it is the same regardless of what week the user is viewing and is essentially just based on player value. The second recommendation however is based on the week the user is viewing. The tool in the second case looks at how many games a player plays in the viewed upon week, on top of their value, to determine the best player for that week. If the user is looking at the current week, the second recommendation for the user changes as the week progresses adjusting to games already played.

Building the Roster Tool

The Roster Tool was the last tool built. This tool only makes one API call to Yahoo, which results in all the roster information for a selected team, including the team's players, and the roster spots they have those players in. After searching in the projections file for all of the roster players, a new 2D players array was built that was a combination of the players information, and their projections. Using the schedule CSV and the player's team information, another 2D array was built, that included the players schedule information for the viewed week. This array will help users see what days they have holes in their roster, so they know what to look for in free agency.

The only filter option available to the user in the Roster Tool is the ability to select which week they want to view their roster for. In the Free Agent / Draft Tool, when the user selects a week, a parameter is sent to the controller. The controller then uses this parameter to select which days it will display to the user. If no parameter is sent the tool should display the current ongoing week. Position filtering was excluded from this tool, since the maximum players on any roster in any Yahoo league is 30, so it did not seem logical to allow for filtering by position. Users, however, are still able to sort their roster by position.

Finally, the controller builds a drop recommendation list for the player. It includes eight total recommendations, four long-term recommendations, and four weekly recommendations. It builds these lists the exact opposite way that the Free Agent / Draft Tool does, as it looks for the lowest value players. The four long-term recommendations in the list are simply the lowest value player at each position, and the roster as a whole. Meanwhile the weekly recommendations list is the same, but also accounts for the amount of games a player has left in the viewed week. For example, it will recommend dropping a player who has a value of 55 but plays 3 games over a player who has a value of 50 but plays 4 games. This is because the value difference is not significant enough between the two players, and the extra game is more valuable.

Updating Projections Mid-season

With more time, more work would have been done to make this section better. What is in place is a method that when generating projections, accounts for a player’s current statistics. For example, if the season is exactly half way done, half of a player’s value and projections will come from their performance so far, and the other half will come from the projections. This is not a perfect method, as it would be better if their in-season performance affected their projections, but unfortunately this would have required another round of machine learning and model building, which due to time, was not accomplished.

Inserting a Playoff Tool

Initially, a separate playoff tool was intended to be created that would help users specifically in their playoff weeks. This idea was scrapped however once the idea to put in weekly recommendations into the other tools. The ability to sort by week, including playoff weeks, made the playoff tool obsolete, as it did exactly what the playoff tool would have done, except that it works for every week, instead of just the playoff weeks.

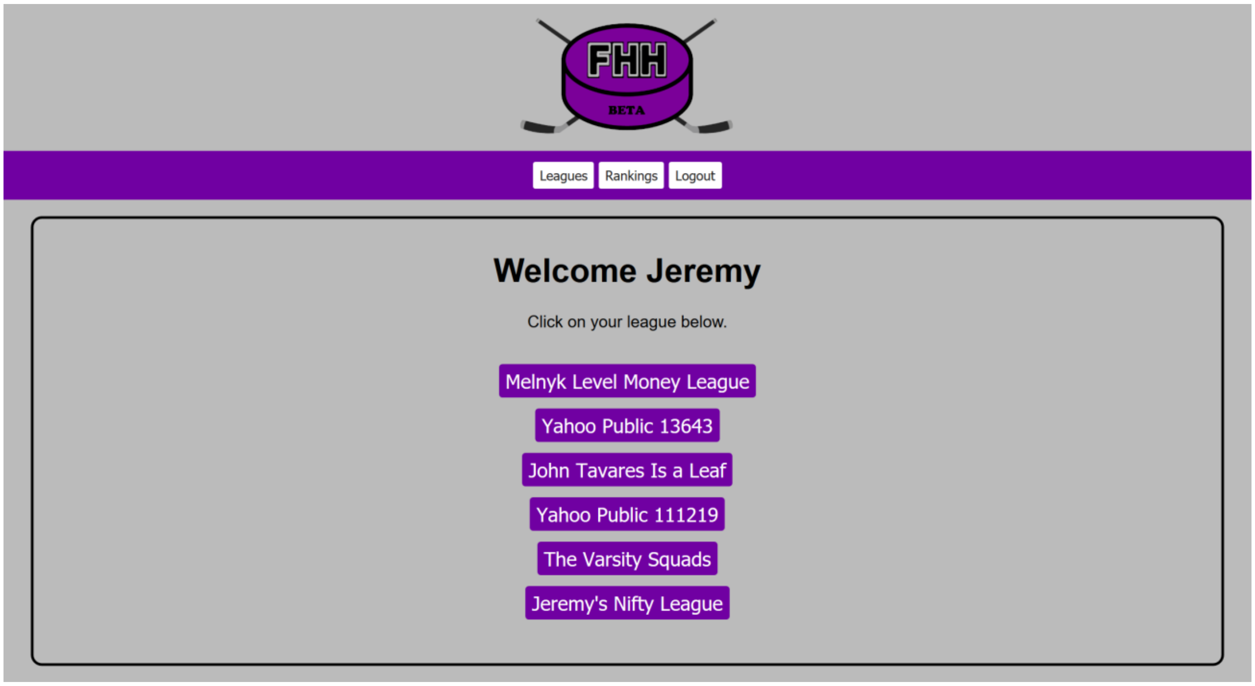
User Interface

User Interface Design Choices

The pages were all made to be easily readable, with big buttons wherever possible. This was done so that the web application would be easy to use on mobile. Since the app is built for Yahoo Fantasy Hockey, the decision was made to use purple and gray as the base colours of the application, these were used to make the application feel familiar to users since Yahoo relies heavily on these colours as well. The projections and schedules for players are separated into two tables on any page where both would make sense to exist. The decision to separate them was to make the application more vertical rather than horizontal, thus allowing for easier use on mobile devices.

League Information Pages

The leagues page is the first page the user sees when they log in, the page just display’s a list of buttons, clickable by the user, to allow them to select a league. When a user clicks on one of the leagues, it takes them to a page with more information on that league and stores the league they clicked in a cookie on the user’s browser. To select a different league, the user can just click on the “Leagues” header and make a different selection.

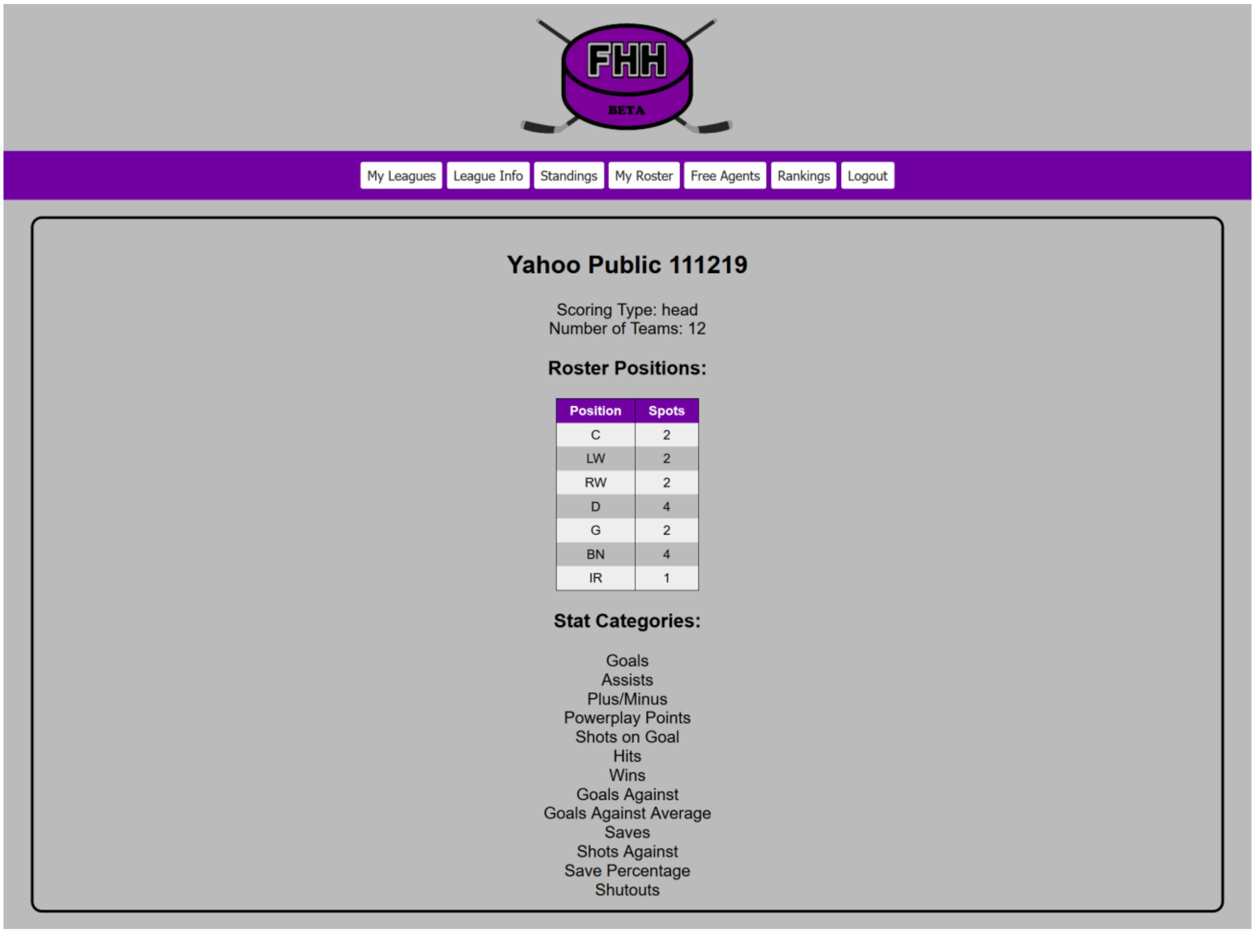


Welcome Jeremy

Click on your league below.

FIGURE 7 – List of user’s leagues – Leagues Page

The league information page provides further information on the user's selected league. Here they will see the scoring system of their league, the number of teams in their league, the stat categories of their league, and the roster positions their league uses. This page serves mostly an in between page between the league selection, and all of the fantasy tools available the user, while also just providing more information about the league they are viewing.



Yahoo Public 111219

Scoring Type: head
Number of Teams: 12

Roster Positions:

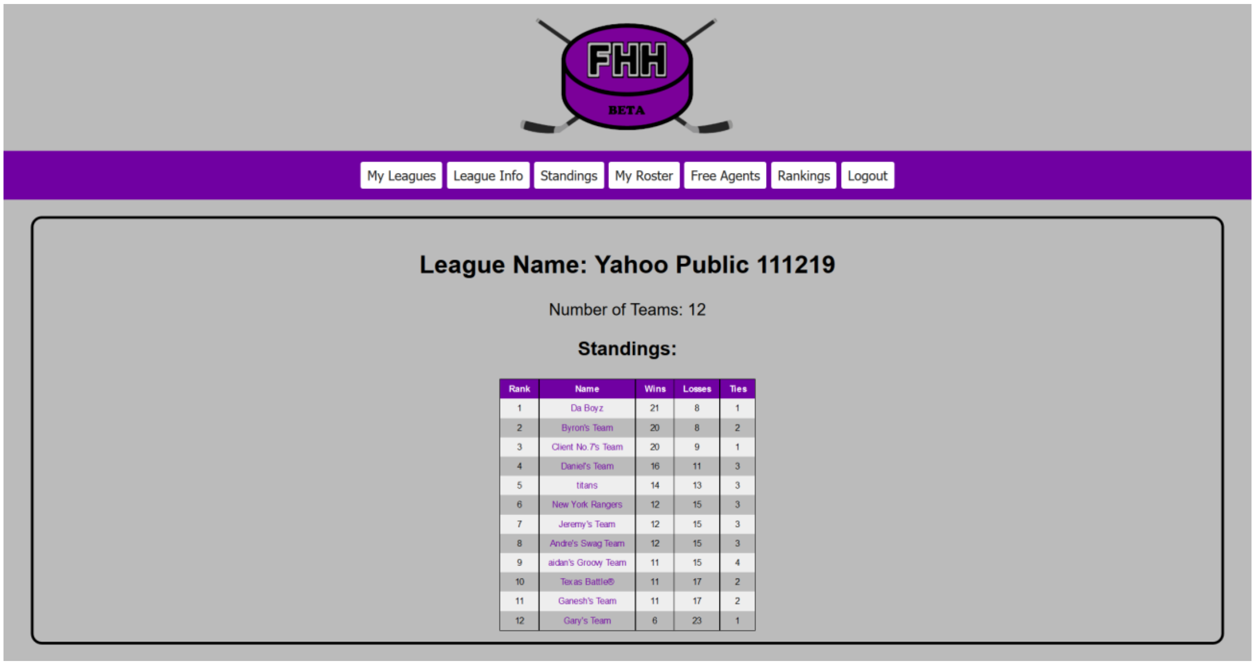
Position	Spots
C	2
LW	2
RW	2
D	4
G	2
BN	4
IR	1

Stat Categories:

Goals
Assists
Plus/Minus
Powerplay Points
Shots on Goal
Hits
Wins
Goals Against
Goals Against Average
Saves
Shots Against
Save Percentage
Shutouts

FIGURE 8 – League Information Page

Finally, the last league information page is the standings page. By clicking on the standings button in the header, the user would be directed to a list of teams in their league, and their current standings. The main purpose for including this page is to include links to the other teams in the league, so that a user can view their opponents' rosters. This is useful to user as it allows them to view the strength of their opponent's roster, and if the user is looking to make a trade, it can help the user determine who on their opponent's team would be most worth targeting.



League Name: Yahoo Public 111219

Number of Teams: 12

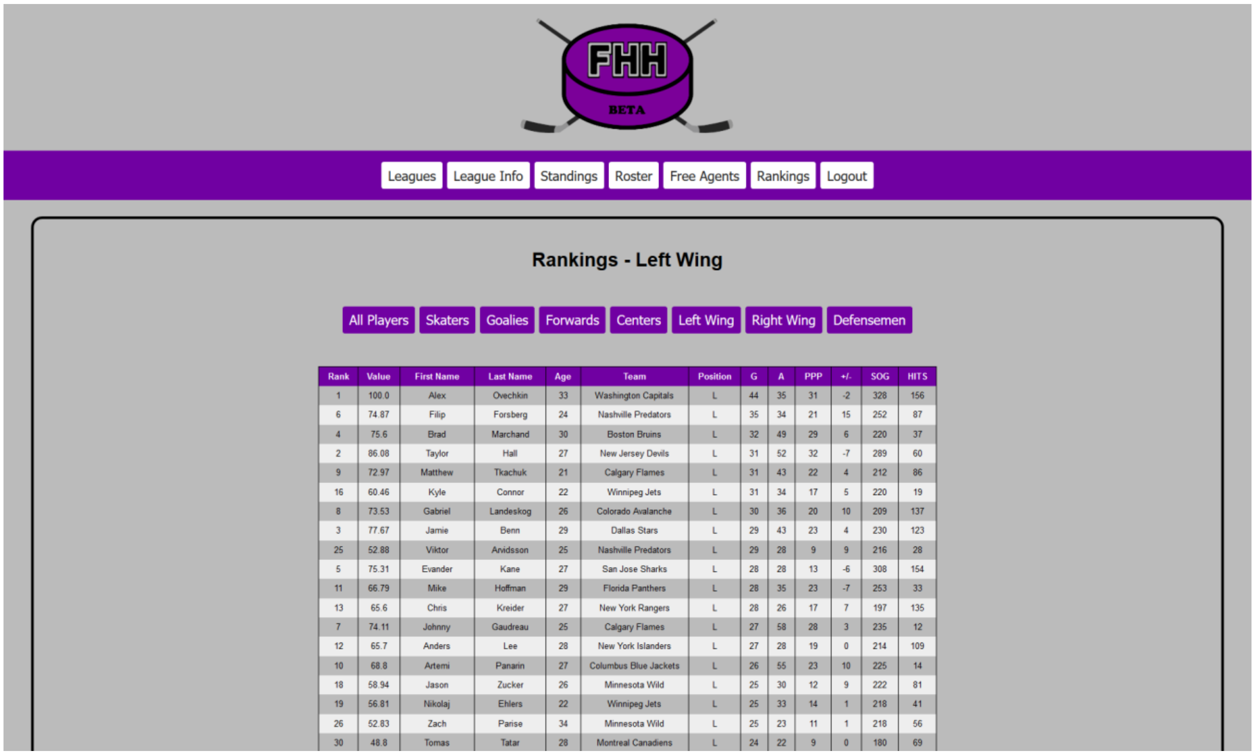
Standings:

Rank	Name	Wins	Losses	Ties
1	Da Boyz	21	8	1
2	Byron's Team	20	8	2
3	Client No.7's Team	20	9	1
4	Daniel's Team	16	11	3
5	titans	14	13	3
6	New York Rangers	12	15	3
7	Jeremy's Team	12	15	3
8	Andre's Swag Team	12	15	3
9	aidan's Groovy Team	11	15	4
10	Texas Battle®	11	17	2
11	Ganesh's Team	11	17	2
12	Gary's Team	6	23	1

FIGURE 9 – League Standings Page

Rankings Page

The rankings page is a table of all the projections made on players. On top of this, is the ability to filter players by position, this is useful for comparing players of the same position easier. The table is also sortable by every column, so that it is easy for the user to find players who are strong in specific stat categories, as a user may not care about every single category.



Rankings - Left Wing

Rank	Value	First Name	Last Name	Age	Team	Position	G	A	PPP	+/-	SOG	HITS
1	100.0	Alex	Ovechkin	33	Washington Capitals	L	44	35	31	-2	328	156
6	74.87	Filip	Forsberg	24	Nashville Predators	L	35	34	21	15	252	87
4	75.6	Brad	Marchand	30	Boston Bruins	L	32	49	29	6	220	37
2	86.08	Taylor	Hall	27	New Jersey Devils	L	31	52	32	-7	289	60
9	72.97	Matthew	Tkachuk	21	Calgary Flames	L	31	43	22	4	212	86
16	60.46	Kyle	Connor	22	Winnipeg Jets	L	31	34	17	5	220	19
8	73.53	Gabriel	Landeskog	26	Colorado Avalanche	L	30	36	20	10	209	137
3	77.67	Jamie	Benn	29	Dallas Stars	L	29	43	23	4	230	123
25	52.88	Viktor	Arvidsson	25	Nashville Predators	L	29	28	9	9	216	28
5	75.31	Evander	Kane	27	San Jose Sharks	L	28	28	13	-6	308	154
11	66.79	Mike	Hoffman	29	Florida Panthers	L	28	35	23	-7	253	33
13	65.6	Chris	Kreider	27	New York Rangers	L	28	26	17	7	197	135
7	74.11	Johnny	Gaudreau	25	Calgary Flames	L	27	58	28	3	235	12
12	65.7	Anders	Lee	28	New York Islanders	L	27	28	19	0	214	109
10	68.8	Artemi	Panarin	27	Columbus Blue Jackets	L	26	55	23	10	225	14
18	58.94	Jason	Zucker	26	Minnesota Wild	L	25	30	12	9	222	81
19	56.81	Nikolaj	Ehlers	22	Winnipeg Jets	L	25	33	14	1	218	41
26	52.83	Zach	Parise	34	Minnesota Wild	L	25	23	11	1	218	56
30	48.8	Tomas	Tatar	28	Montreal Canadiens	L	24	22	9	0	180	69

FIGURE 10 – Rankings Page – Filtered by Left Wing – Sorted by Goals

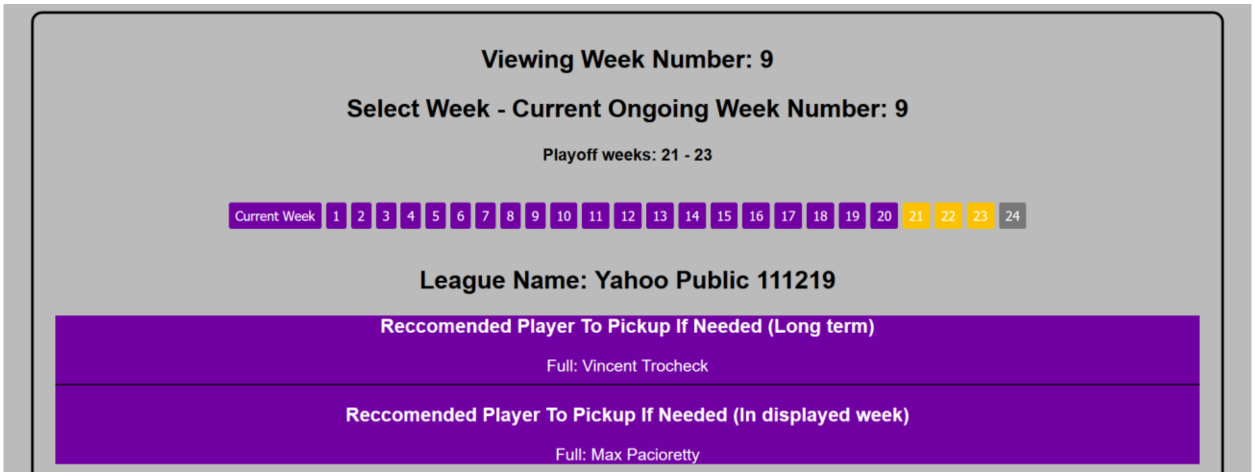
Free Agents/Draft Page

In both versions of the tool, a sortable table showing all the player projections is available, below this is a sortable table showing a schedule for the players in the viewed week. The users have the ability to click on a week at the top of the page and view a different week. The options include the current week, regular season weeks (shown in purple), and playoff weeks (shown in yellow). Finally, similar to the rankings tool, the user can filter by position in case they are looking for a player of a specific position.

The tool also displays two recommendations to the user. The first is a long-term value-based recommendation and the second is a week-based recommendation, specific to the week the user is viewing. If the user is viewing a specific position, the recommendation changes to a player of that viewed position.

If the league is in draft mode, the page can be refreshed after every team makes a draft pick, to see an updated list of available players. This is useful as it allows for players to use this tool while drafting and consistently see an updated list of available players.

Finally, when the league is in post draft, it shows a list of available players, as well as what type of availability they are. In Yahoo Standard leagues there are two kinds of available players, free agents, which can be picked up by anybody instantly, or waiver players, who are players that multiple teams can put claims on, but only one wins out. This distinction between these types is an important piece of information for the user when the tool makes recommendations, as one of the two types of players won't be instantly available to the user.



Viewing Week Number: 9

Select Week - Current Ongoing Week Number: 9

Playoff weeks: 21 - 23

League Name: Yahoo Public 111219

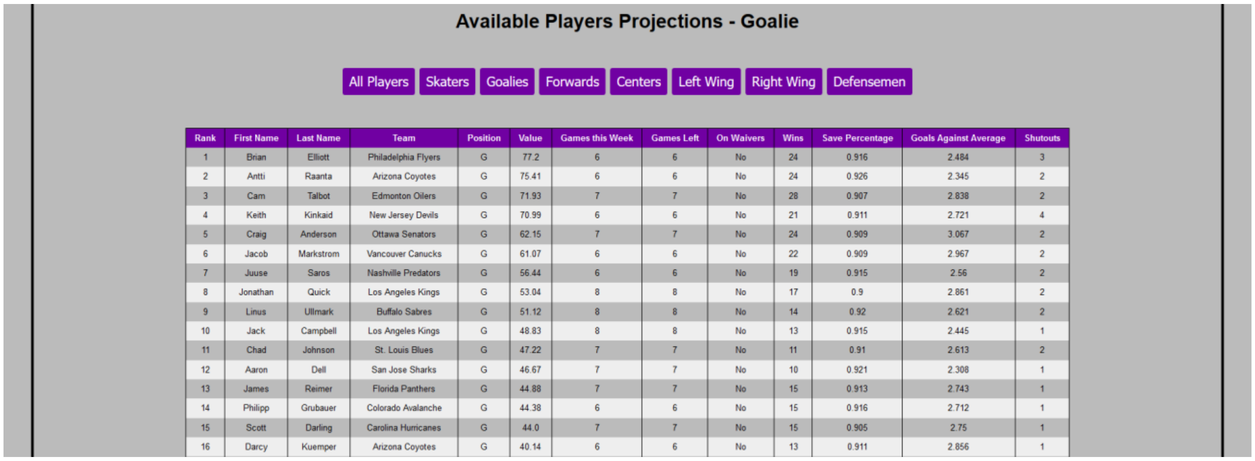
Reccomended Player To Pickup If Needed (Long term)

Full: Vincent Trocheck

Reccomended Player To Pickup If Needed (In displayed week)

Full: Max Pacioretty

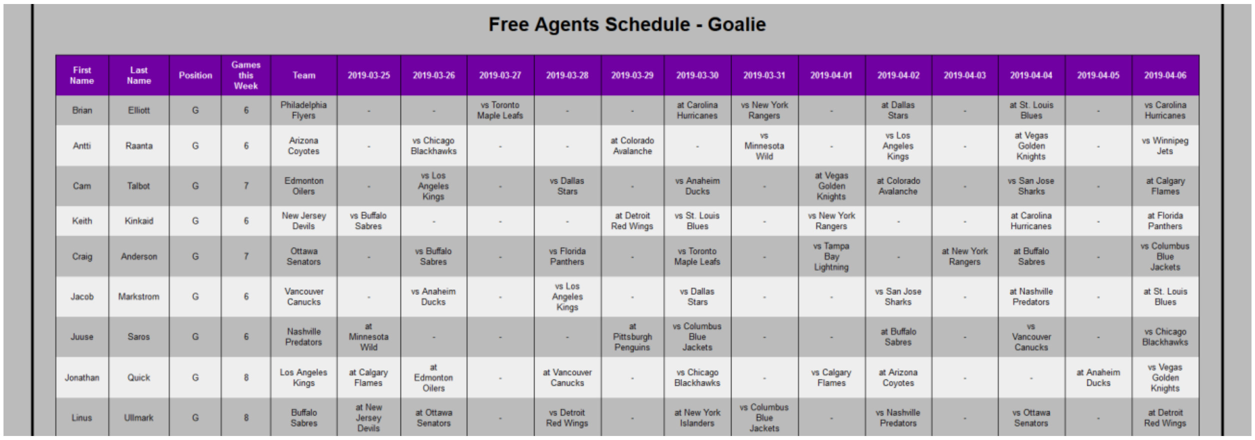
FIGURE 11 – Free Agents / Draft Tool Page– Viewing Recommendations and Week Selections for Full Roster



Available Players Projections - Goalie

Rank	First Name	Last Name	Team	Position	Value	Games this Week	Games Left	On Waivers	Wins	Save Percentage	Goals Against Average	Shutouts
1	Brian	Elliott	Philadelphia Flyers	G	77.2	6	6	No	24	0.916	2.484	3
2	Antti	Raanta	Arizona Coyotes	G	75.41	6	6	No	24	0.926	2.345	2
3	Cam	Talbot	Edmonton Oilers	G	71.93	7	7	No	28	0.907	2.838	2
4	Keith	Kinkaid	New Jersey Devils	G	70.99	6	6	No	21	0.911	2.721	4
5	Craig	Anderson	Ottawa Senators	G	62.15	7	7	No	24	0.909	3.067	2
6	Jacob	Markstrom	Vancouver Canucks	G	61.07	6	6	No	22	0.909	2.967	2
7	Juuse	Saros	Nashville Predators	G	56.44	6	6	No	19	0.915	2.56	2
8	Jonathan	Quick	Los Angeles Kings	G	53.04	8	8	No	17	0.9	2.861	2
9	Linus	Ullmark	Buffalo Sabres	G	51.12	8	8	No	14	0.92	2.621	2
10	Jack	Campbell	Los Angeles Kings	G	48.83	8	8	No	13	0.915	2.445	1
11	Chad	Johnson	St. Louis Blues	G	47.22	7	7	No	11	0.91	2.613	2
12	Aaron	Dell	San Jose Sharks	G	46.67	7	7	No	10	0.921	2.308	1
13	James	Reimer	Florida Panthers	G	44.88	7	7	No	15	0.913	2.743	1
14	Philipp	Grubauer	Colorado Avalanche	G	44.38	6	6	No	15	0.916	2.712	1
15	Scott	Darling	Carolina Hurricanes	G	44.0	7	7	No	15	0.905	2.75	1
16	Darcy	Kuemper	Arizona Coyotes	G	40.14	6	6	No	13	0.911	2.856	1

FIGURE 12 – Free Agents / Draft Tool Page– Viewing Available Goalie Projections Sorted by Value



Free Agents Schedule - Goalie

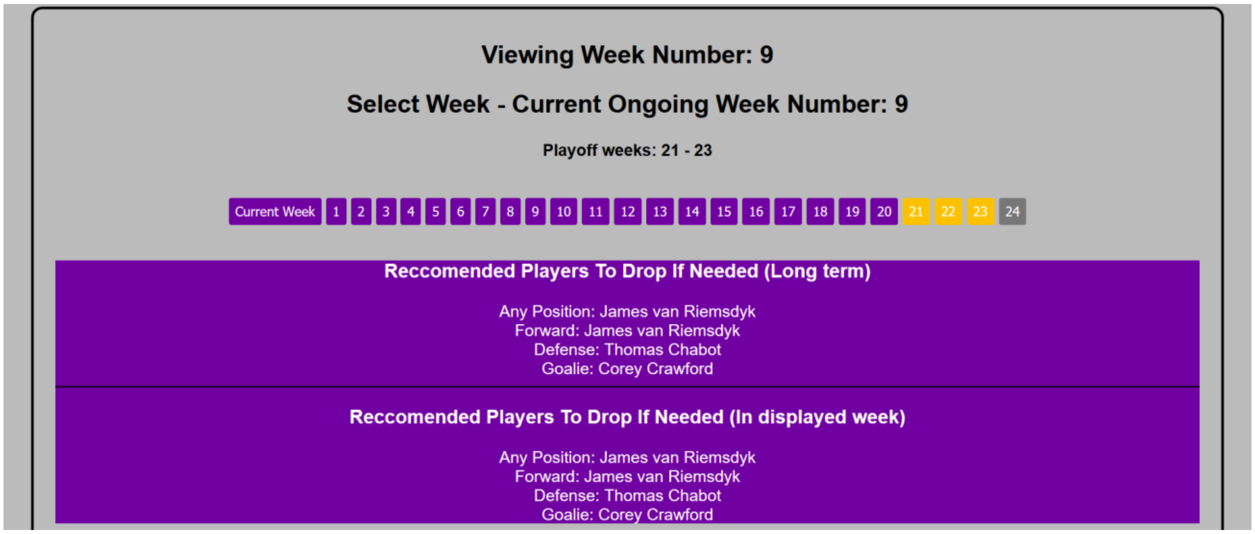
First Name	Last Name	Position	Games this Week	Team	2019-03-25	2019-03-26	2019-03-27	2019-03-28	2019-03-29	2019-03-30	2019-03-31	2019-04-01	2019-04-02	2019-04-03	2019-04-04	2019-04-05	2019-04-06
Brian	Elliott	G	6	Philadelphia Flyers	-	-	vs Toronto Maple Leafs	-	-	at Carolina Hurricanes	vs New York Rangers	-	at Dallas Stars	-	at St. Louis Blues	-	vs Carolina Hurricanes
Antti	Raanta	G	6	Arizona Coyotes	-	vs Chicago Blackhawks	-	-	at Colorado Avalanche	-	vs Minnesota Wild	-	vs Los Angeles Kings	-	at Vegas Golden Knights	-	vs Winnipeg Jets
Cam	Talbot	G	7	Edmonton Oilers	-	vs Los Angeles Kings	-	vs Dallas Stars	-	vs Anaheim Ducks	-	at Vegas Golden Knights	at Colorado Avalanche	-	vs San Jose Sharks	-	at Calgary Flames
Keith	Kinkaid	G	6	New Jersey Devils	vs Buffalo Sabres	-	-	-	at Detroit Red Wings	vs St. Louis Blues	-	vs New York Rangers	-	-	at Carolina Hurricanes	-	at Florida Panthers
Craig	Anderson	G	7	Ottawa Senators	-	vs Buffalo Sabres	-	vs Florida Panthers	-	vs Toronto Maple Leafs	-	vs Tampa Bay Lightning	-	at New York Rangers	at Buffalo Sabres	-	vs Columbus Blue Jackets
Jacob	Markstrom	G	6	Vancouver Canucks	-	vs Anaheim Ducks	-	vs Los Angeles Kings	-	vs Dallas Stars	-	-	vs San Jose Sharks	-	at Nashville Predators	-	at St. Louis Blues
Juuse	Saros	G	6	Nashville Predators	at Minnesota Wild	-	-	-	at Pittsburgh Penguins	vs Columbus Blue Jackets	-	-	at Buffalo Sabres	-	vs Vancouver Canucks	-	vs Chicago Blackhawks
Jonathan	Quick	G	8	Los Angeles Kings	at Calgary Flames	at Edmonton Oilers	-	at Vancouver Canucks	-	vs Chicago Blackhawks	-	vs Calgary Flames	at Arizona Coyotes	-	-	at Anaheim Ducks	vs Vegas Golden Knights
Linus	Ullmark	G	8	Buffalo Sabres	at New Jersey Devils	at Ottawa Senators	-	vs Detroit Red Wings	-	at New York Islanders	vs Columbus Blue Jackets	-	vs Nashville Predators	-	vs Ottawa Senators	-	at Detroit Red Wings

FIGURE 13 – Free Agents / Draft Tool Page– Viewing Available Goalies Schedule Information

Roster Page

The roster page helps users view the value of all the players on their roster and make decisions about whether or not there is any players they should consider dropping. The roster is displayed in a table and shows all of the roster's projections. The table is sortable and allows users to sort by any individual stat category so that they can compare their players in any category. The page also shows the schedule information for the user's roster in the viewed upon week. Similar to the Free Agent / Draft page, the user can select which week they wish to view schedule information for, as well as recommendations for

Also like the Free Agent / Draft page, the roster page displays two kinds of recommendations. Once again, they are long-term full season recommendations, and short-term weekly recommendations. The difference for the roster page recommendations is that it displays a recommendation of both types for every position. This was necessary, since the roster can not be filtered by position.



Viewing Week Number: 9

Select Week - Current Ongoing Week Number: 9

Playoff weeks: 21 - 23

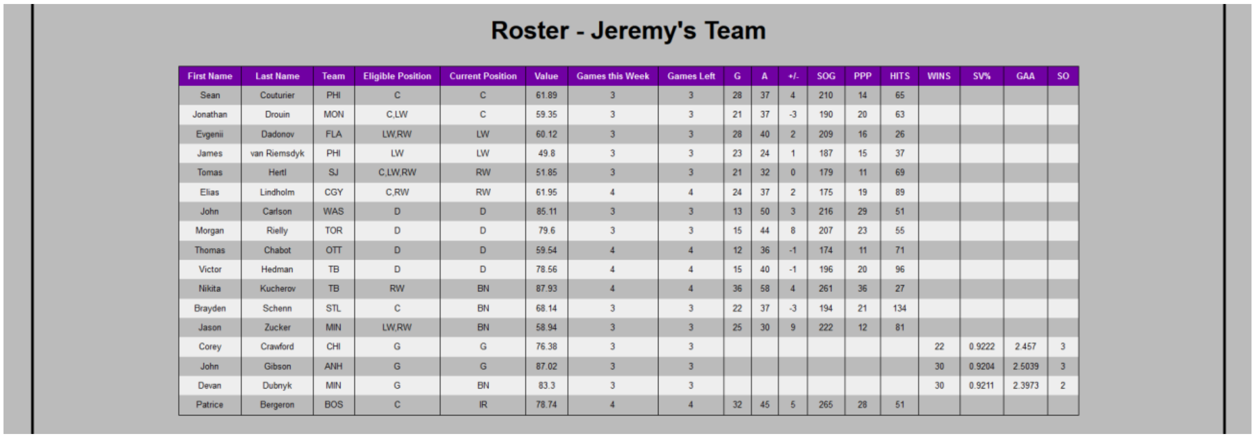
Reccomended Players To Drop If Needed (Long term)

Any Position: James van Riemsdyk
Forward: James van Riemsdyk
Defense: Thomas Chabot
Goalie: Corey Crawford

Reccomended Players To Drop If Needed (In displayed week)

Any Position: James van Riemsdyk
Forward: James van Riemsdyk
Defense: Thomas Chabot
Goalie: Corey Crawford

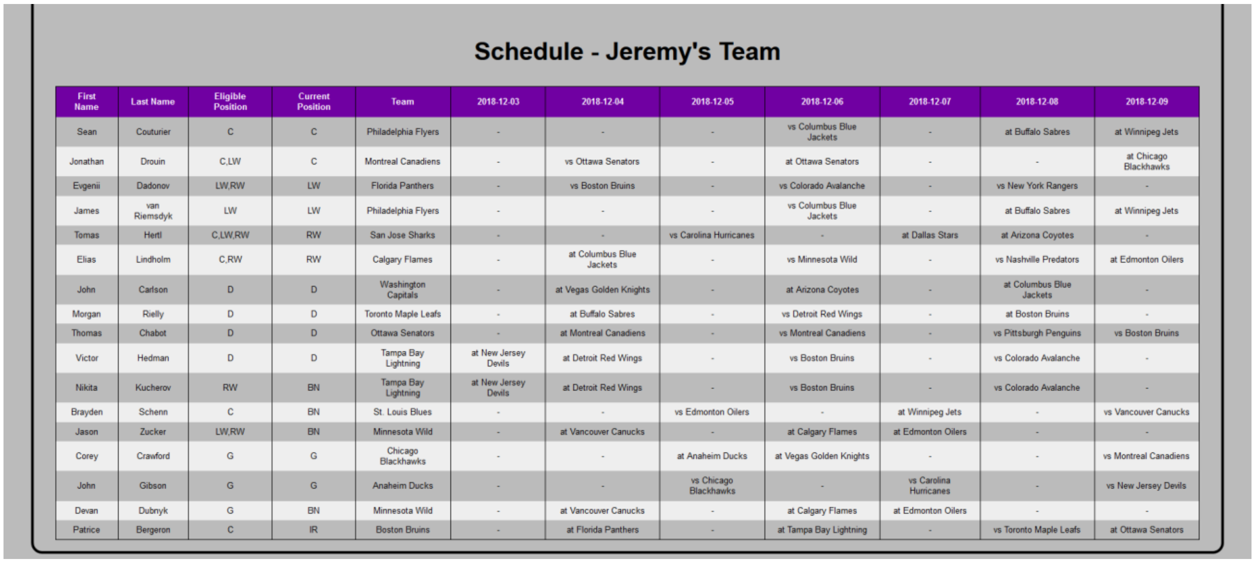
FIGURE 14 – Roster Tool Page – Viewing Week selection and Recommendations



Roster - Jeremy's Team

First Name	Last Name	Team	Eligible Position	Current Position	Value	Games this Week	Games Left	G	A	+/-	SOG	PPP	HITS	WINS	SV%	GAA	SO
Sean	Couturier	PHI	C	C	61.89	3	3	28	37	4	210	14	65				
Jonathan	Drouin	MON	C,LW	C	59.35	3	3	21	37	-3	190	20	63				
Evgenii	Dadonov	FLA	LW,RW	LW	60.12	3	3	28	40	2	209	16	26				
James	van Riemsdyk	PHI	LW	LW	49.8	3	3	23	24	1	187	15	37				
Tomas	Hertl	SJ	C,LW,RW	RW	51.85	3	3	21	32	0	179	11	69				
Elias	Lindholm	CGY	C,RW	RW	61.95	4	4	24	37	2	175	19	89				
John	Carlson	WAS	D	D	85.11	3	3	13	50	3	216	29	51				
Morgan	Rielly	TOR	D	D	79.6	3	3	15	44	8	207	23	55				
Thomas	Chabot	OTT	D	D	59.54	4	4	12	36	-1	174	11	71				
Victor	Hedman	TB	D	D	78.56	4	4	15	40	-1	196	20	96				
Nikita	Kucherov	TB	RW	BN	87.93	4	4	36	58	4	261	36	27				
Brayden	Schenn	STL	C	BN	68.14	3	3	22	37	-3	194	21	134				
Jason	Zucker	MIN	LW,RW	BN	58.94	3	3	25	30	9	222	12	81				
Corey	Crawford	CHI	G	G	76.38	3	3							22	0.9222	2.457	3
John	Gibson	ANH	G	G	87.02	3	3							30	0.9204	2.5039	3
Devan	Dubnyk	MIN	G	BN	83.3	3	3							30	0.9211	2.3973	2
Patrice	Bergeron	BOS	C	IR	78.74	4	4	32	45	5	265	28	51				

FIGURE 15 – Roster Tool Page – Viewing Team Roster Sorted by Position



Schedule - Jeremy's Team

First Name	Last Name	Eligible Position	Current Position	Team	2018-12-03	2018-12-04	2018-12-05	2018-12-06	2018-12-07	2018-12-08	2018-12-09
Sean	Couturier	C	C	Philadelphia Flyers	-	-	-	vs Columbus Blue Jackets	-	at Buffalo Sabres	at Winnipeg Jets
Jonathan	Drouin	C,LW	C	Montreal Canadiens	-	vs Ottawa Senators	-	at Ottawa Senators	-	-	at Chicago Blackhawks
Evgenii	Dadonov	LW,RW	LW	Florida Panthers	-	vs Boston Bruins	-	vs Colorado Avalanche	-	vs New York Rangers	-
James	van Riemsdyk	LW	LW	Philadelphia Flyers	-	-	-	vs Columbus Blue Jackets	-	at Buffalo Sabres	at Winnipeg Jets
Tomas	Hertl	C,LW,RW	RW	San Jose Sharks	-	-	vs Carolina Hurricanes	-	at Dallas Stars	at Arizona Coyotes	-
Elias	Lindholm	C,RW	RW	Calgary Flames	-	at Columbus Blue Jackets	-	vs Minnesota Wild	-	vs Nashville Predators	at Edmonton Oilers
John	Carlson	D	D	Washington Capitals	-	at Vegas Golden Knights	-	at Arizona Coyotes	-	at Columbus Blue Jackets	-
Morgan	Rielly	D	D	Toronto Maple Leafs	-	at Buffalo Sabres	-	vs Detroit Red Wings	-	at Boston Bruins	-
Thomas	Chabot	D	D	Ottawa Senators	-	at Montreal Canadiens	-	vs Montreal Canadiens	-	vs Pittsburgh Penguins	vs Boston Bruins
Victor	Hedman	D	D	Tampa Bay Lightning	at New Jersey Devils	at Detroit Red Wings	-	vs Boston Bruins	-	vs Colorado Avalanche	-
Nikita	Kucherov	RW	BN	Tampa Bay Lightning	at New Jersey Devils	at Detroit Red Wings	-	vs Boston Bruins	-	vs Colorado Avalanche	-
Brayden	Schenn	C	BN	St. Louis Blues	-	-	vs Edmonton Oilers	-	at Winnipeg Jets	-	vs Vancouver Canucks
Jason	Zucker	LW,RW	BN	Minnesota Wild	-	at Vancouver Canucks	-	at Calgary Flames	at Edmonton Oilers	-	-
Corey	Crawford	G	G	Chicago Blackhawks	-	-	at Anaheim Ducks	at Vegas Golden Knights	-	-	vs Montreal Canadiens
John	Gibson	G	G	Anaheim Ducks	-	-	vs Chicago Blackhawks	-	vs Carolina Hurricanes	-	vs New Jersey Devils
Devan	Dubnyk	G	BN	Minnesota Wild	-	at Vancouver Canucks	-	at Calgary Flames	at Edmonton Oilers	-	-
Patrice	Bergeron	C	IR	Boston Bruins	-	at Florida Panthers	-	at Tampa Bay Lightning	-	vs Toronto Maple Leafs	at Ottawa Senators

FIGURE 16 – Roster Page – Viewing Roster Schedule Information

Results

Projections Performance

To evaluate the performance of the projections, they were compared to the projections of three other sources. Yahoo Fantasy Hockey creates their own projections, they have been largely considered poor, and were used as the low-level performance bracket for this comparison. Dobber has created projections for years and is well respected in the industry. His projections are generally middle to high level in quality but will be used as the mid-tier. Scott Cullen is largely considered the best in business with projections, and his projections will be used as the top-tier.

To compare all the projections, the same subset of players was used for all sources. Since not all players are projected by every source, but top players usually are, the top 300 players ranked from the projections created by the application will be used. Any players not projected for all four sources will be discarded.

Not all sources project games played, but some do, so for this comparison all projections were prorated to 82 games for every source. So, for example if a source says they project 20 goals in 41 games, for this comparison that number was adjusted to 40 goals in 82 games. In essence, the pace projections of the player were compared rather than the physical projections.

Finally, since the season is only about one third of the way done, the players actual statistics were also prorated to 82 games as well, since this will allow for the easiest comparisons. Skaters who have played less than 20 games or goalies who have played less than 10 games so far this season were excluded from the comparisons due to lack of sample size. The Plus/Minus stat was also excluded, since not all sources project this.

The average differences in each statistic category, between the projections for every source and the actual performance of the players were then recorded to be compared.

	Projection Source			
Statistic	Fantasy Hockey Helper	Dobber	Scott Cullen	Yahoo Fantasy Hockey
G	8.019	7.894	7.770	8.081
A	10.700	10.481	10.644	11.077
PPP	6.866	6.765	6.722	6.753
SOG	28.077	31.071	28.514	29.699
HITS	21.798	21.344	21.800	23.400
W	7.529	8.559	8.206	7.735
SV%	0.014	0.014	0.014	0.021
GAA	0.442	0.511	0.430	0.414
SO	2.706	2.853	2.765	2.824

TABLE 1 – Average Difference Between Projections and Actual Performance

	Projection Source			
Statistic	Fantasy Hockey Helper	Dobber	Scott Cullen	Yahoo Fantasy Hockey
Rank G	3	2	1	4
Rank A	3	1	2	4
Rank PPP	4	3	1	2
Rank SOG	1	4	2	3
Rank HITS	2	1	3	4
Rank W	1	4	3	2
Rank SV%	3	2	1	4
Rank GAA	3	4	2	1
Rank SO	1	4	2	3
Rank(AVG)	2.33	2.78	1.89	3.00
Ranking	2	3	1	4

TABLE 2 – Average Rank Comparison

	Projection Source			
Statistic	Fantasy Hockey Helper	Dobber	Scott Cullen	Yahoo Fantasy Hockey
Standard G	19.958	60.042	100	0
Standard A	63.239	100	72.643	0
Standard PPP	0	69.757	100	78.323
Standard SOG	100	0	85.409	45.824
Standard HITS	77.899	100	77.829	0
Standard W	100	0	34.286	80
Standard SV%	93.651	94.444	100	0
Standard GAA	71.042	0	83.625	100
Standard SO	100	0	60	20
Score(AVG)	69.532	47.138	79.310	36.016
Ranking	2	3	1	4

TABLE 3 – Average Standardized Score Comparison

TABLE 2 (Ranks Comparison) and TABLE 3 (Standard Score Comparison) reached the same conclusion for the ranking of the four projections sources. Outside the projections created by the application, the three outside projections sources ranked in the exact order predicted. Scott Cullen’s projections ranked the highest, Dobber’s projection ranked in the middle, and Yahoo’s projections ranked last. In comparison the created projections ranked right between Scott Cullen’s and Dobber’s projections, although when standardized, ranked closer to Scott Cullen. Therefore, as the tables display, the projections developed by the application ranked above average. Although they are not the best, they performed on a similar level as other respected projection sources.

Application Performance and Evaluation

The Fantasy Hockey Helper web application was compared to the currently most popular tool, Fantasy Hockey Geek, to determine its strengths and weaknesses. The pros and cons of both applications were compared, as well as the functionality and speed of the applications.

Fantasy Hockey Helper		Fantasy Hockey Geek	
Pros	Cons	Pros	Cons
Capable of giving advice in live drafts	Non-customizable stat categories	Customizable stat categories	Cost money
Free	Limited to Yahoo	Connected with Yahoo and Fantrax	To slow to help in live drafts
Mobile friendly	Does not allow users to use custom projections	Custom projections permitted	Not mobile friendly
Weekly help provided	No trade analyzing tool	Trade analyzing tool included	No week-to-week tool, only full season analysis
Schedule information for players provided			No player schedule information provided
Gives advice to users			Does not give any advice
One click to almost anywhere in the application			Takes multiple clicks to get or do most things

TABLE 4 – Pros and Cons comparison with Fantasy Hockey Geek

The results from the pros and cons comparison show there are clearly strengths to both tools. The weekly assistance feature as well as the schedule information provided by the created Fantasy Hockey Helper application helps assist the user long past draft day. This gives the user more help throughout the season and has more long-term usefulness to users. Furthermore, while both tools provide draft help, the speed and ability for Fantasy Hockey Helper to give live advice during the draft by updating after every draft pick, is a lot more helpful to the user. This is because as players get drafted, Fantasy Hockey Geek does not remove them from their tables, making it hard to keep track of which players are still available. On top of this the user-interface is a lot simpler with Fantasy Hockey Helper, allowing you to get to almost anywhere in the application in one click. Comparatively Fantasy Hockey Geek is a lot more confusing, require multiple clicks to do things like update rosters, or get refresh the projections.

Fantasy sports are also starting to trend more and more towards mobile and according to Oath, Yahoo's parent company, up to 92% of fantasy sport interactions occur on mobile[11]. Since user's are playing fantasy sports on mobile, it's increasingly important that any fantasy tools also be mobile friendly. This is where Fantasy Hockey Helper shines, the user interface was designed entirely with mobile in mind, designed vertically instead of horizontally. The user can also add the application to their home screen through their browser, and it will load like an application. This is due to the application being created as a Progressive Web Application. Comparatively, Fantasy Hockey Geek struggles on mobile, and you cannot even login without switching to desktop mode on your phone.

Conversely, Fantasy Hockey Geek's main strength is that it allows for custom stat categories, and thus is not restricted to the standard yahoo stat categories. This allows for a lot of flexibility in users who are not in standard leagues. On top of this Fantasy Hockey Geek also has a Fantrax implementation that allows for a whole other website of users to use the site. The site also has a trade analyzer tool, that Fantasy Hockey Helper does not include, but it comes at an even larger price than just the base application.

[11] https://www.oath.com/insights/lexus-snickers-bars-and-universal-pictures-play-to-win-with-ya/

Source	Fantasy Hockey Helper		Fantasy Hockey Geek	
Page Load Time	Desktop	Mobile (LTE)	Desktop	Mobile (LTE)
Home Page	Instant	1.2 Seconds	2 Seconds	3.5 Seconds
Get League Information for first time	1.5 Seconds	3 Seconds	50 Seconds	52 Seconds
Acquire and display updated available players list	3 Seconds	4 Seconds	32 Seconds	34 Seconds
View Updated Projections	Instant	2 Seconds	190 Seconds	195 Seconds

TABLE 5 – Comparing Load Speed (Fantasy Hockey Helper vs. Fantasy Hockey Geek)

The table above displays conclusively that Fantasy Hockey Helper is much faster than Fantasy Hockey Geek. This is a very important benefit to the created Fantasy Hockey Helper application as most users would prefer not to spend upwards of three minutes doings something like updating projections. This large amount of time would definitely have a negative impact on the user experience.

Finally, on of the main differences between the two applications is cost, Fantasy Hockey Helper is free, and is one of the only tools online that is. Fantasy Hockey Geek however cost $16 for the base tools, plus another $7 for all of its tools. This is a total cost of $23, which is much higher than the average fantasy hockey user is likely willing to pay.

In summary, it is clear to see the benefits of Fantasy Hockey Helper over what is commonly referred to as the best fantasy hockey assistance application out there. Fantasy Hockey Helper is faster, cheaper, and easier to use. However, it falls short in customizability, and this is where the application would have to improve before being able to conclusively call it the best fantasy hockey assistance application.

Conclusion

In conclusion, all of the goals set out for this project were achieved. The project had two main goals and three sub goals. The first main goal was to create projections that were on the same level as other the projections from existing fantasy hockey experts. Creating these projections proved more difficult to create than expected. This left me with a lot of respect for the people that put the work into creating new ones every single year. This was definitely the hardest part of the project, as finding the correct subset of variables to serve as the model for each statistic was a complicated task. Overall however the projections performed above expectations, and although they did not prove the best, they still performed well enough to achieve the first goal of this project.

The second main goal of this project was to create a web-based application that was both mobile friendly and had multiple tools to assist fantasy hockey players. The hardest part of creating the web-application was setting up the Yahoo API calls. Due to the authentication methods required to make any of these calls it took a lot of research into the best ways to make this work. To keep the application mobile friendly the application was designed with big buttons, and to fit on a vertical mobile screen, adjusting where necessary. It was also created as a progressive web application so that mobile users could use their mobile browsers "Add to Home Screen" button and get a mobile app-like version of the application. Multiple different tools were made available to the users as well including tools that assists with roster decisions, free agent decisions and draft decisions. Overall, this second goal was also achieved, the application accomplished almost everything it set out to, and even proved to be able to compete with one of the top fantasy hockey assistance applications out there.

The three sub-goals of keeping the application error-free, user-friendly and to not store unnecessary data on user were also all achieved. In terms of keeping the application error-free, although there are likely still errors somewhere, any that came up through testing were fixed, and any error routes redirect the user to an appropriate page. To keep the application user-friendly everything was kept one click away. This allowed the user to find all the most important tools and information with one click. This also helped with mobile friendliness as well. Finally, the application stored zero data on the users, achieving the final sub-goal. This was done by making use of cookies, allowing users to store any necessary data on their own browsers. API calls take only seconds to make, so it made more sense to have the users store their yahoo authentication tokens and any other necessary information.

One future goal for the project, if there was more time would include having a more sophisticated mid-season adjustment for projections to allow for more accurate projections as the season progresses. The other goal would be to allow for custom stat categories as this would make the application more useful to a wider audience.

Overall, everything that was set to be achieved was, and a lot was learned about the process of both creating web applications for scratch, as well as using to machine learning and analytics to create projections from models.

References

[1] “scikit-learn Machine Learning in Python” Internet: https://scikit-learn.org/stable/

[2] “oauth-xx/oauth2: A Ruby wrapper for the OAuth 2.0 protocol” Internet: https://github.com/oauth-xx/oauth2

[3] “ikylin/omniauth-yahoo_auth: Yahoo OAuth2 Strategy for OmniAuth (Rails)” Internet: https://github.com/ikylin/omniauth-yahoo_auth

[4] “rest-client/rest-client: Simple HTTP and REST client for Ruby, inspired by microframework syntax for specifying actions” Internet: https://github.com/rest-client/rest-client

[5] “Fantasy Sports API – Yahoo Develop Network” Internet: https://developer.yahoo.com/fantasysports/guide/

[6] “dword4/nhlapi: Documenting the publicly accessible portions of the NHL API” Internet: https://github.com/dword4/nhlapi

[7] “Documenting the NHL API” Internet: https://www.reddit.com/r/hockey/comments/7p7o70/documenting_the_nhl_api/dsfuy54/

[8] “daugaard/linear-regression: Linear regression implemented in Ruby” Internet: https://github.com/daugaard/linear-regression

[9] “2018-19 NHL point projections and playoff probabilities” Internet: https://theathletic.com/565136/2018/11/20/2018-19-nhl-point-projections-and-playoff-probabilities-updated-daily/

[10] “OAuth 2.0 - OAuth” Internet: https://oauth.net/2/

[11] “Lexus, SNICKERS® Bars and Universal Pictures Play to Win with Yahoo Sports” Internet: https://www.oath.com/insights/lexus-snickers-bars-and-universal-pictures-play-to-win-with-ya/