

Project Progress Checkpoint

CSCE 436 - Spring 2017

April 4th, 2017

Team “Outrageous Privilege”

Cory Avra - coryavra@email.tamu.edu

Smitty Chandler - hschand@live.com

Zachary Daum - daum.zachary@gmail.com

Courtney Ericson - ceyevar@email.tamu.edu

Heming Tian - Tian.heming.2012@vjc.sg

Aggie Code of Honor

An Aggie does not lie, cheat or steal or tolerate those who do.

Problem Statement

What is the problem?

For the National Hockey League (NHL), as the season winds down, every fan who follows a team currently out of playoffs has one question: will my team make the playoffs? This answer to this simple question is actually rather complex, as there are a lot of factors that go into deciding which teams makes the playoffs and which teams don't. More specifically, for a team to make the playoffs, several conditions must be met. How many games does the team need to win? What are their chances of winning? Are certain games more important (weighted heavier) than others? And are there any games that the other, opposing teams need to lose for the mentioned fan's team to make playoffs?

Our problem statement encompasses answering all the questions above, visually representing a team's playoff odds in a user friendly interface. Currently, there are no available user-friendly approaches to this problem, as most approaches are bogged down by heavy-handed mathematical representations that are incomprehensible and intimidating to the average fan. Our application will solve this problem by providing an intuitive interface for viewing playoff odds that utilizes the design principles discussed in class.

What is the proposed solution?

Our solution shall provide a daily-changing interface for users to track the progress of their team into playoffs day-by-day. A user will be able to input the team they want to follow, and each day the application will provide a list of games scheduled. The list of games schedules will be ranked by importance, with importance being defined as how much impact a specific game can have on the team of the user making a playoff berth. In addition, a user will be able to view his team's current projected chances at playoffs and the Stanley Cup (which is the league trophy).

For the most part, the main emphasis of coding this application will lie in the user interface. The scraping of the data would be trivial, and the mathematic algorithms for calculating the percentages already exist in academic work. The main challenge will be formulating an app that is easy to use and organized in a fashion that doesn't confuse the users.

Why should people care?

Firstly, this field is relatively untouched by mobile applications. Few apps exist, and those that do are overwhelming with unnecessary stats and figures clouding the relevant data. By involving just the right amount of necessary figures - such as rank, percentages, ratings, and more advanced stats - our application will potentially draw a large and diverse audience. Our goal is to streamline the application in such a manner

that it's accessible and appealing to a range of fans, from the die-hards to those with little prior knowledge.

How does it solve the problem?

Our application will streamline relevant stats in an aesthetic, intuitive, and user-friendly fashion to visually represent a team's chances of making the playoffs utilizing the design principles learned in class. This allows its users to easily accomplish their goal of tracking their team's chances.

How does it differ (improve) existing approaches?

As previously mentioned, few other offer this service. Those that do are overly-complicated, so our more simplistic approach will make tracking teams easy.

Design Specification

What metaphors/design principles does the project embody?

Our application has been built around designing for the most optimal user experience through achieving and meeting the various goals specified by the different design principles. Most notably, the application centers around the quick output and presentation of our predicted game results to the user in a timely and convenient manner.

The visibility aspect of the design principle is achieved by placing what the users are most interested in at the forefront of the screen once the users log in. Therefore, the users would never have to navigate through our application just in order to look for the most relevant information they are looking for. Since our application's main objective is to deliver to the user the predicted future game play outcomes of their favorite teams, we have designed our UI to accommodate for a straightforward presentation of the results through incorporating an interactive design of a sliding window. This allows the user to swipe through the historical game plays as well as the future predicted game outcomes.

The feedback aspect of our application has been taken care of through setting the default number of gameplays in the sliding window to be only a certain number so that the user does not have to worry about loading all the historical records of the game which could be irrelevant information that potentially compromises the performance of the application at the same time.

The constraints aspect of our application makes sure that the user does not commit to an action that may have unwarranted results. For example we only display the information about the favourite team that was chosen instead of displaying the information of all the teams on the screen. Also, this adds value to the learnability

aspect of our application as well, because the user could easily follow the instructions/prompts to arrive at the results they are requesting, instead of having to search for them themselves.

The consistency aspect of our application has been carefully designed for, because we use the same library for GUI and what the user sees for the application is fairly consistent, meaning that the user could expect the same feedback from one given action without much changes. For example the user could always expect to review the past games and the future predictions by swiping through the window, and the consistency that is guaranteed here is that by swiping right one is going back to the previous games and swiping left is moving to the future game predictions.

The affordance aspect of our application is closely related to its learnability aspect, in the sense that it is rather easy to step through the process and get to learn how to use it. As soon as the user has made selection for his favourite team, the predictions will be tailored to only showcase the upcoming game predictions for that particular game. If more detailed information is needed about the overall likelihood for a user to proceed further in the tournament, for eg, winning his division, the user could click on the window right below the game predictions and be directed to another screen.

What is the overall design?

The overall design of our application features a user-friendly UI that allow the user to select the favourite NFL team and be presented with the future predictions of its upcoming games as well as the overall likelihood for it to make into any future conferences or playoffs. We only trim and present the relevant information that the user needs based on his selection of the favourite team.

What are the core features?

The core features of the game is to display the future predicted game outcomes of the user's favorite team as well as assessing the overall likelihood for the team to proceed further into the future tournament. Upon opening the application, the user will be prompted to select his/her favorite team from the drop down window that contains all the teams. After selecting the favourite team, the user will be led to the screen where he/she is presented with an interactive sliding window design. Right at the center of the sliding window will be the information the user is most likely to be interested in: the predicted outcome of the game in the nearest future. From there, the user is encouraged to swipe through the timeline of the possible games predictions to the right of the main window, or to see its past games to its left.

If in the event that the user wishes not only to access the information about his/her favourite team's individual gameplay outcomes, the user could click on the bottom of the screen currently containing the sliding window to be redirected to a detailed analysis and predictions of its likelihood to make into the future tournaments and Conferences.

What tools/technologies will you use and why?

The backend utilizes Amazon Web Services' Amazon Linux AML operating system image. An up-to-date database is maintained by using multiple technologies. The database is a MySQL database, which is an open-source relational database management system. To create and keep the database up-to-date, we used Python as our programming language and Scrapy, a fast and powerful scraping and web crawling framework. Using crontab, the scraper scrapes the current standings and each team's current data and adds it to the database. This process occurs every day at 2am CST. A screenshot of our database as viewed through phpMyAdmin, a tool for admin's to view their database over the world wide web, can be seen below.

MySQL

filter items by name

New

AHA

ARI

BOS

BUF

CAR

CBJ

CGY

CHI

COL

current_standings

DAL

DET

EDM

FLA

LAK

MINN

MTL

NJD

NSH

NYI

NYR

OTT

PHI

PIT

SJS

STL

TBL

TOR

VAN

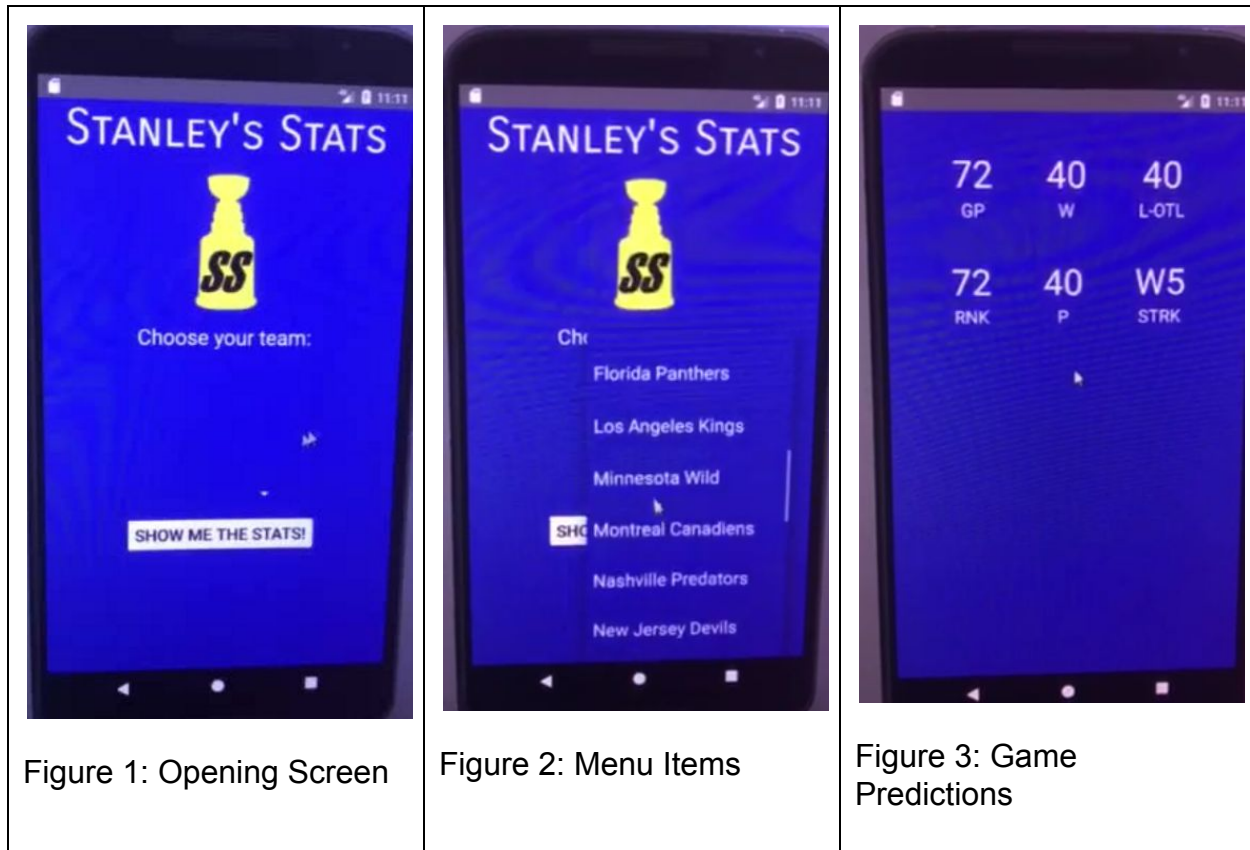
WPG

WSH

current_ranking	team_name	name_abbreviation	games_played	wins	losses	overtime	points	row	goals_for	goals_against	goal_differential	home_record	away_record	shootout_record	last_ten_record	streak	last_game	next_game	division	conference	
2	Anaheim	ANA	72	38	23	11	87	35	186	178		8	23-8-4	15-15-7	3-3	6-3-1	W1	Mar 18: ANA 2 - SJS 1	Mar 22 vs EDM	Pacific	Western
7	Arizona	ARI	71	26	36	9	61	20	172	227	-55	16-16-5	10-20-4	6-2	4-4-2	L1	Mar 18: ARI 0 - STL 3	Mar 20 @ NSH	Pacific	Western	
3	Boston	BOS	71	38	27	6	82	36	205	188		17	19-14-0	19-13-6	2-2	7-3-0	L1	Mar 16: BOS 4 - EDM 7	Mar 20 @ TOR	Atlantic	Eastern
7	Buffalo	BUF	72	29	31	12	70	27	180	210	-30	17-12-6	12-19-6	2-6	3-5-2	W1	Mar 17: BUF 2 - ANA 1	Mar 20 @ DET	Atlantic	Eastern	
6	Carolina	CAR	69	30	27	12	72	28	179	199	-20	21-10-4	9-17-8	2-5	5-2-3	W2	Mar 18: CAR 4 - NSH 2	Mar 19 @ PHI	Metropolitan	Eastern	
2	Columbus	CBJ	70	46	18	6	98	44	224	163		61	26-9-1	20-9-5	2-1	7-2-1	W3	Mar 18: CBJ 3 - NYI 2	Mar 19 @ NJD	Metropolitan	Eastern
4	Calgary	CGY	71	40	27	4	84	36	196	191		5	21-15-0	19-12-4	4-2	9-1-0	W1	Mar 17: CGY 3 - DAL 1	Mar 19 vs LAK	Pacific	Western
1	Chicago	CHI	71	46	20	5	97	43	212	175		37	23-9-4	23-11-1	3-1	8-2-0	W4	Mar 18: CHI 2 - TOR 1	Mar 19 vs COL	Central	Western
7	Colorado	COL	70	20	47	3	43	20	135	228	-93	11-22-2	9-25-1	0-1	3-7-0	L1	Mar 18: COL 1 - DET 5	Mar 19 @ CHI	Central	Western	
6	Dallas	DAL	71	28	33	10	66	28	195	233	-38	18-12-6	10-21-4	0-1	4-6-0	L1	Mar 17: DAL 1 - CGY 3	Mar 20 vs SJS	Central	Western	
8	Detroit	DET	70	28	31	11	67	20	177	210	-33	14-15-5	14-16-6	8-0	4-5-1	W2	Mar 18: DET 5 - COL 1	Mar 20 vs BUF	Atlantic	Eastern	
3	Edmonton	EDM	71	38	24	9	85	34	207	186		21	19-12-4	19-12-5	4-5	6-4-1	W3	Mar 18	Mar 20 vs	Pacific	Western

The application UI was developed through Android Studio with majority of the codes written in Python and Java. The Python KIVY framework was used in the development of the innovative features of our user interface. The core algorithm that we are implementing is based off the calculation from the ELO rating system, which gauges the respective overall likelihood for a team to win in the case of either winning or losing one particular game. The core algorithm for determining playoff odds is implemented entirely in Python.

Screenshots of the user interface



How does the UI satisfy the needs of your users?

We believe our application UI satisfy the possible needs of the user based on the fact that we give the the flexibility and the option to choose only the team that they are interested in at any given time. The information as well as the game outcomes predictions will all be based off the initial selection that the user makes. We have gotten rid of any unnecessary information associated with the user's query by only displaying the most relevant information we believe the user is interested in, so that he/she does not have to search through our application to look for what is needed. Most importantly, the UI is designed to accommodate for easy and responsive feedback for an action registered on the screen, this has to do with the implementation specifics of our algorithms. The results that are pulled out of our database is also presented in a way that is interactive to use and explore with the integration of a cascading/sliding window that enables the user to simply swipe through both the past and future game predictions. Thus overall our UI has accommodated not only for the usability aspect of the application by only displaying the most relevant information, it also allows for easy and interactive use that adds value to its affordance and learnability.

What is the current status of the implementation?

As of right now, each of the three major components of the application are each in product. The underlying algorithm evaluating playoffs odds can currently calculate a team's strength and the algorithm can calculate the projected win percentage of each team in a game. Still needing to be completed is the aggregation function that calculates the effect one specific game will have on the standings. The second main component, the scraper which queries the NHL.com website is currently completed. The scraper extracts data and updates the database everyday. The third main component, the UI is in the preliminary phase. As of right now, the basic format of data has been established and the UI can display basic elements of a team's record. The UI still needs to display playoff probabilities and be integrated with the algorithm and database backbone.

We have discussed and agreed upon the main features that should be presented to the user upon first login as well as the overall layout of our application. The general user experience that we are designing for will center around the "prediction" of the future games instead of looking in retrospect to the past games. We have gotten rid of the unnecessary information display of the chosen team by trimming the information displayed to be only the predictions of the future games' outcomes that the users are looking for. At this stage we have made good use of Android Studio to design and implement the UI of our application and a key surprise factor in our design will be the use of the sliding window as the forefront of our GUI that allows for users to interact with the chosen team's future game outcomes or previous games by swiping across the window. The default number of games displayed, both historical and future will be set to 7 to accommodate for optimal performance of the application.

The main GUI library we have used is KIVY, the framework that supports cross-platform Python development. Upon logging into the application and having selected the favorite team from the drop down window that contains all of the teams' names, the user will be led to the sliding window that displays first and foremost the predictions of the game that is the nearest in the future. A major revision here is that we decided to only display the most relevant information the user is interested upfront instead of having the user to navigate through the various pages to find what they are looking for. In this case we have gotten rid of displaying games that have already happened to make room for future predictions of game outcomes by allowing the user to simply swipe through the future games.

Work Plan

Summary

For the most part we have followed the original workplan. The scraper and algorithm that calculates the playoff odds both took less time than anticipated so more time has been allocated to the UI phase of implementation. One specific major revision we have

made that has affected our work is changing the overall structure of the UI. The last major revision to our work plan stems for the demands of integrating the python backbone of the application with the Android interface.

Due to the changes to the user interface, we are now using Kivy and are adding a cascading view of upcoming games. Learning a new GUI library (which very few of us have worked with) required us to designate one of the weeks that would have normally been dedicated to the design of the algorithm to familiarization with Kivy. The cascading view change to the UI is a more difficult implementation than a regular list-based view and therefore has required us to add another half-week period for implementing the cascade view which would have normally been done by now.

The integration needed between the Python backbone of the application and the android interface has required us to change our work plan. The UI development phase, which was initially expected to have the integration happen simultaneously with the UI formulation but has now been split up into two separate phases to increase simplicity. As of right now, we plan to design the UI initially using mock data and then, in a secondary phase 2 weeks from now, to substitute the actual Python backbone in place of the mock data.

Task List

1. Scraped NHL site for schedule, scores and organize into database
2. Created and updated ELO rating for each team according to wins and losses and store in db. Created interface for viewing.
3. Need to finish algorithms for estimating playoff probability
 - a. Calculate number of points needed in future games to qualify (i.e. how many games to win)
 - b. Obtain ELO rating of each team who you will play in remaining games
 - c. Use ELO rating of each team to calculate probability of win in each remaining game (so if the ELO rating of opponent is higher than your team, you have less probability of winning).
 - d. Aggregate percentage chances of winning of all remaining games and compare to points needed to qualify
 - e. **Output: X team has a 12% chance of qualify for playoffs based on their remaining schedule and the relative strength of their opponents**
4. Finish creating graphical interface for viewing daily games which:
 - a. Allows user to choose a favorite team
 - b. Sorts by relative importance of each game (which games have the biggest effect on the user's team making playoffs)
 - i. Shows results of the game if it has already happened
 - ii. Shows percentage swing of each game
 1. How much a win, tie, etc. will impact each team's playoff chances
 - c. Allows users to view current playoff picture (percent chance of qualifying & projected first-round matchup)

d. Have a working prototype to evaluate halfway through

Member Assignments

- Statistical backbone team
 - Courtney Ericson
- Data retrieval team
 - Zachary Daum
 - Heming Tian
- User interface team
 - Hayden Chandler
 - Cory Avra

Weekly Breakdown

Key for Table	
Task 1	
Task 2	
Task 3	
Task 4	
Redesign/Tweak	

Sun	Mon	Tues	Wed	Thurs	Fri	Sat
				<i>Proposal due</i>		
Sprint planning						
		UI mock-up	Sprint planning			
				UI integration with DB	Sprint planning	
				<i>Progress Checkpoint</i>		

				Sprint planning		
				UI prototype finished	Begin user trials with interface	
					Begin redesigning	More trials w/ interface
		<i>In-class Demo</i>	<i>In-class Demo</i>	<i>In-class Demo</i>		

Bibliography

How to Calculate the Elo Rating. Metin's Media and Math. 4th March 2015:
<https://metinmediamath.wordpress.com>

Stanley Cup Playoffs. Background Information on NHL. 2017:
<https://www.nhl.com/stanley-cup-playoffs>

My Sports Feed. Live Sports Data Feed API. 2017:
<https://www.mysportsfeeds.com/>

nhlscrapr: NHL Scraper API. PythonHosted. 2015:
<http://pythonhosted.org/nhlscrapr/>

GoBase.org - Introduction to the Elo Rating System. GoBase. 2017:
<http://gobase.org/studying/articles/elo/>

NFL Elo Ratings Are Back! FiveThirtyEight. 10th September 2015:
<https://fivethirtyeight.com/datalab/nfl-elo-ratings-are-back/>