Qupid

The teams matchmaking service

Content Table

About Qupid

1

Is this Section a Pitch Deck?

About the Project

Qupid is a matchmaking service that is responsible for creating efficient teams for workplaces and classes.

Why?

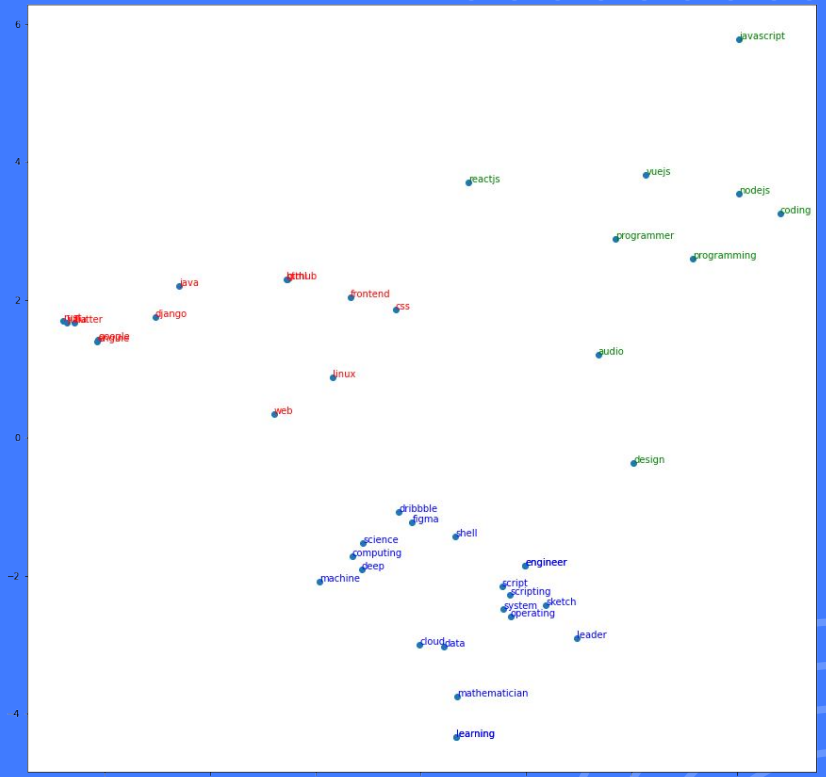
It is always difficult to come up with efficient teams when we are working on projects. These are some problems that we face :

- Teams are not diverse
- Teams are not balanced in skill level
- Teams have higher distribution members belonging to one major skill.

With Qupid our goals is to deal with all these problems and make teams that are diverse and balanced in skill level.

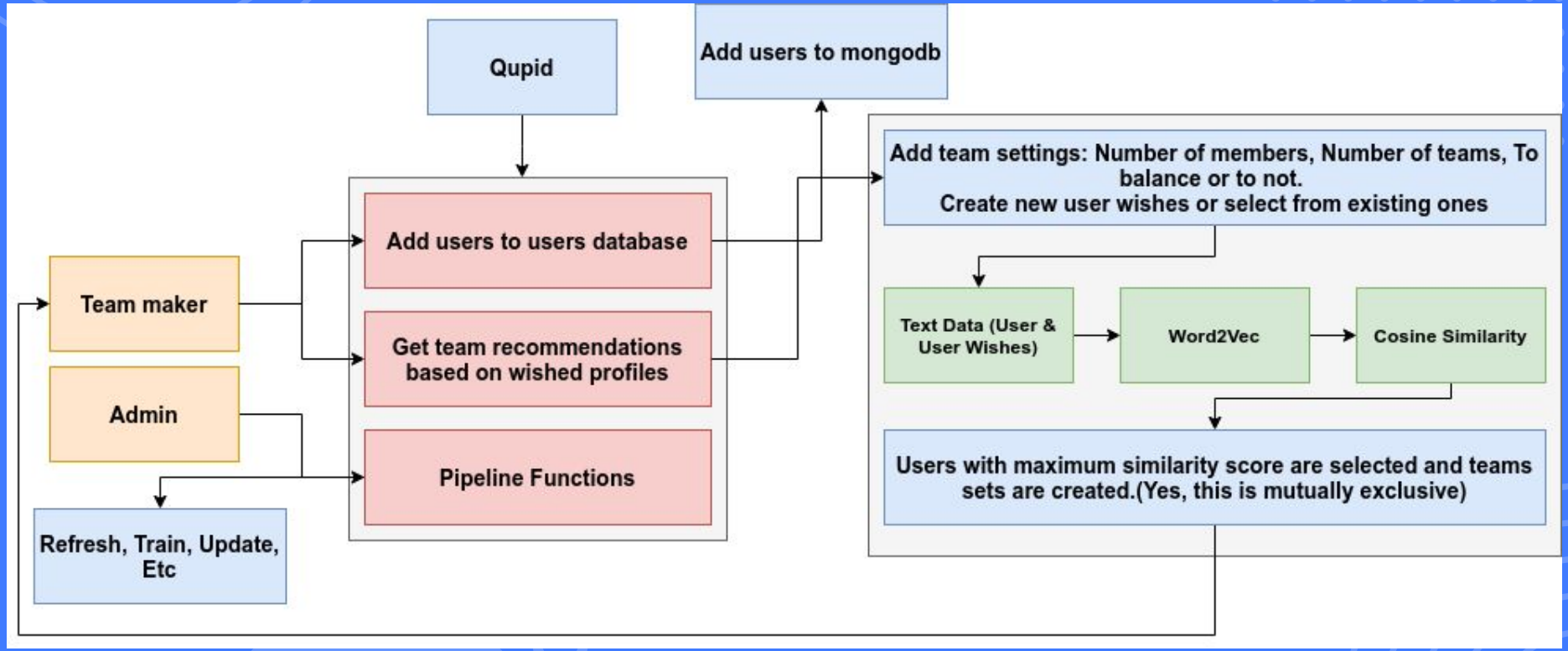
How?

Qupid uses a custom algorithm designed by me. At its core this is based on *word embeddings* created from *scraping tweets* related to data appearing in user profiles and then using *cosine similarity* to make teams that are just like the *team that the team maker has wished*. This is explained in detail in section Under the Hood.



3 cluster Kmeans plot of 2 major dimensions (PCA) of trained Word2Vec Model. Notice how similar words are grouped together

Flowcharts are great!



Project Overview

2

Let’s view the project from front!

4 Main Sections of Qupid

Home

Add Users

Make Teams

Settings

Home

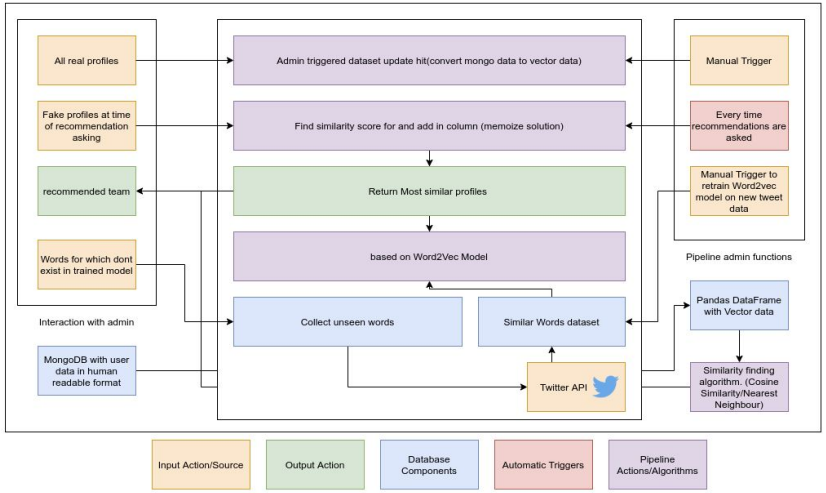
Qupid: The teams matchmaker!

Welcome to Qupid!

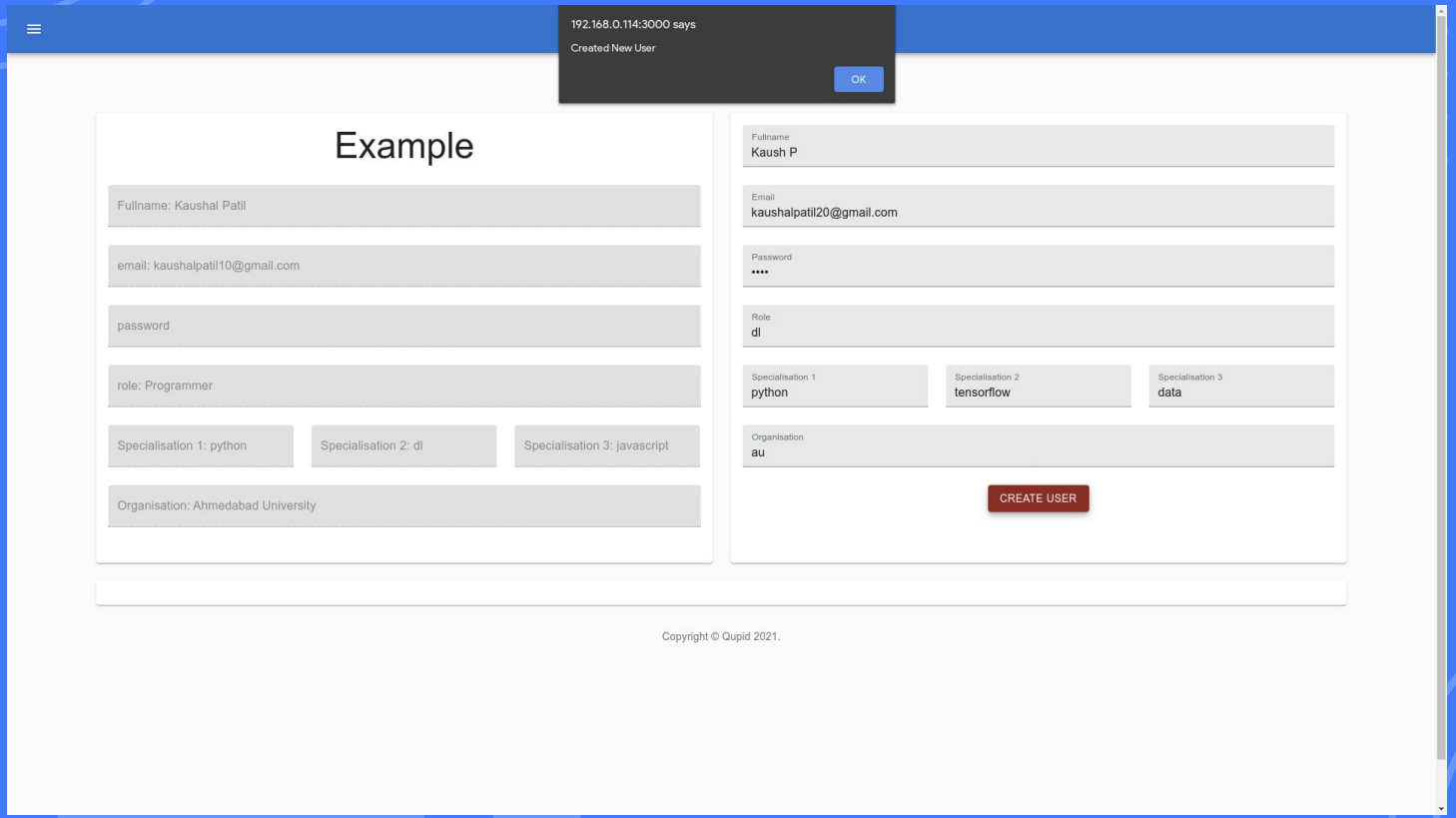
The teams matchmaker!

Qupid is a matchmaking service that is responsible for creating efficient teams for workplaces and classes. It is based on a custom made content based recommendation algorithm that uses a self trained word2vec model at its core. This model is trained from scrapping tweet data that has terms related to team users profiles.

Data Pipeline Model



Add Users


192.168.0.114:3000 says
Created New User
OK
Example
Fullname: Kaushal Patil
email: kaushalpatil10@gmail.com
password
role: Programmer
Specialisation 1: python
Specialisation 2: dl
Specialisation 3: javascript
Organisation: Ahmedabad University
Fullname
Kaush P
Email
kaushalpatil20@gmail.com
Password
••••
Role
dl
Specialisation 1
python
Specialisation 2
tensorflow
Specialisation 3
data
Organisation
au
CREATE USER
Copyright © Qupid 2021.

How it works ?

1. You add details such as fullname, email, password, role, speciality 1, speciality 2, speciality 3 and organisation and click on create button.
2. A new user is created in database.

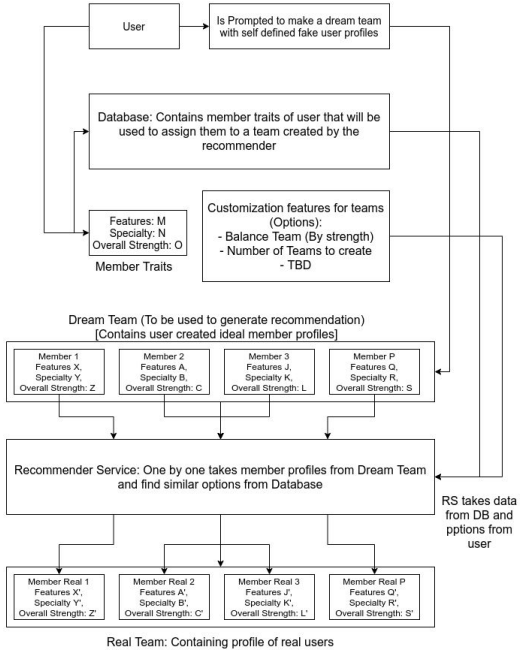
What to do next?

- To make your user appear in pipeline dataframe you got to settings and click on update users in dataset button. (This is kept this way so multiple users can be added once and the pipeline data can be updated at once(to save time on atomic requests)).

Make Teams: Page 1

Qupid: The teams matchmaker!

How Team Creation Works.



Team Settings

Number of members in a team

2

Number of Teams

2

Balance Teams

BEGIN PROCESS

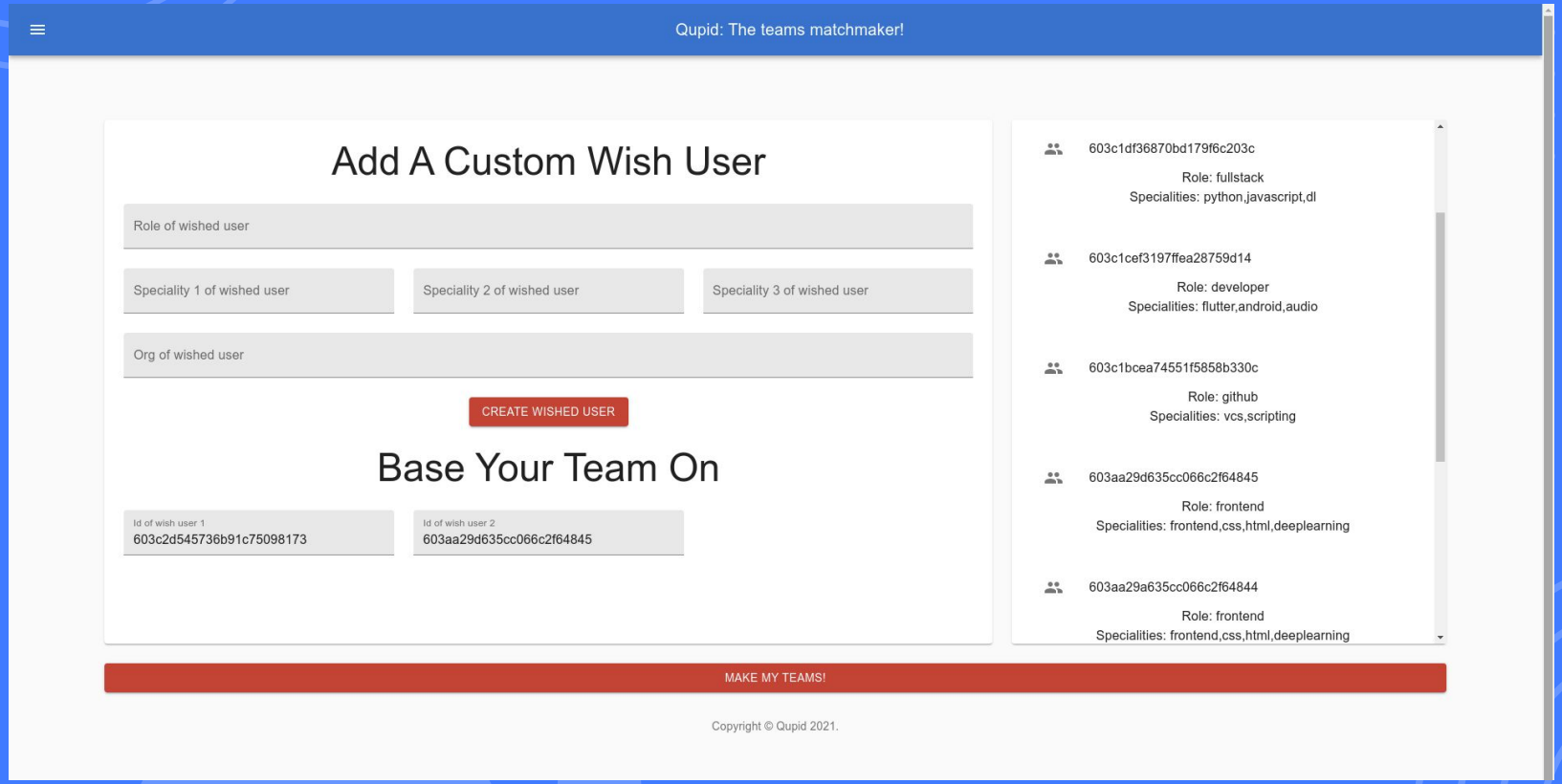
How it works ?

1. You add teams details such as number of members in a team and number of teams that you want to create in one call.
2. You select if you want to balance teams or not.

What to do next?

- Wish for certain kind of users and create one wish if the already existing don’t match your requirement.

Make Teams: Page 2



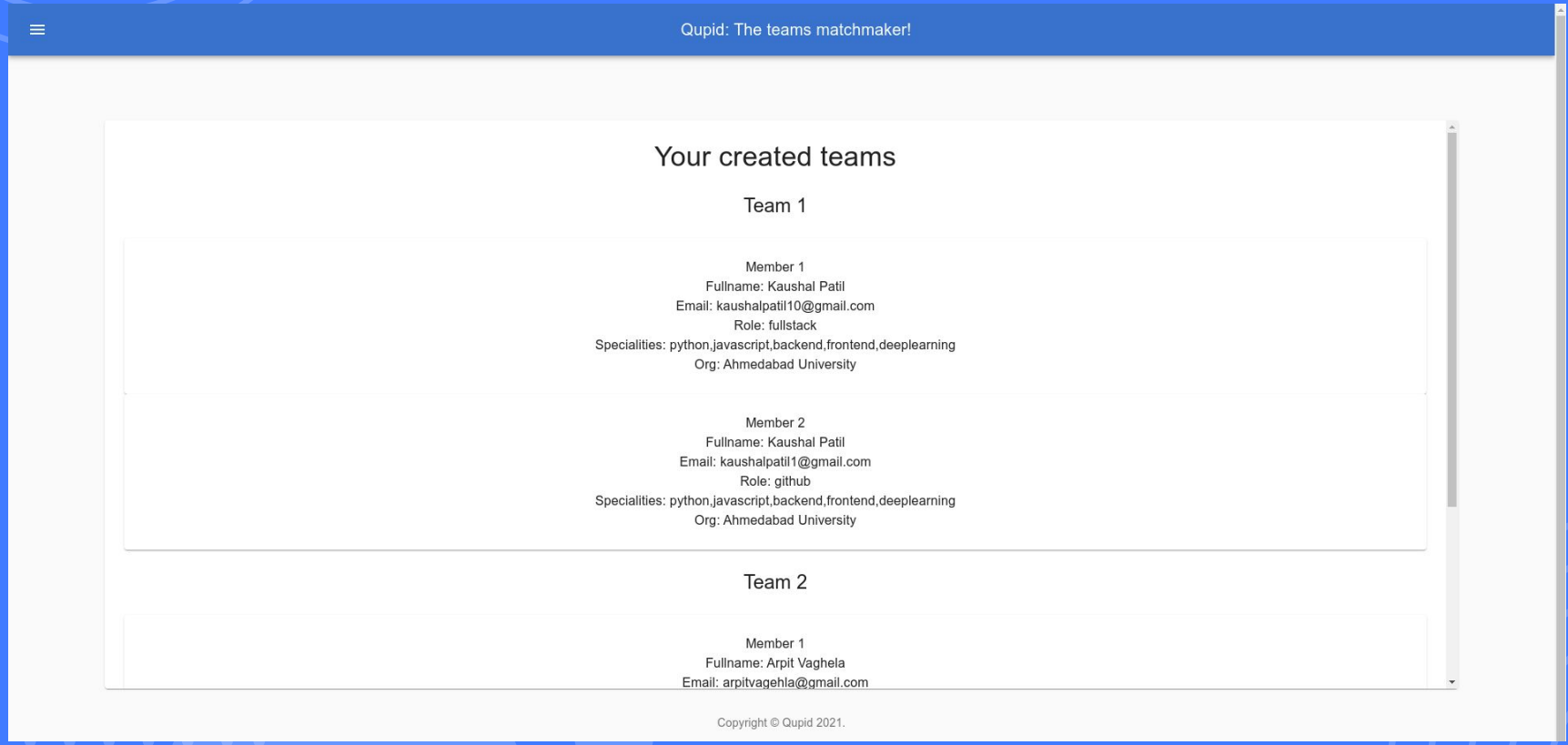
How it works ?

1. You copy ID of a user wish from right side into the Base your team section ID.
2. If a suitable user wish doesn't exist already you create one using th create from.

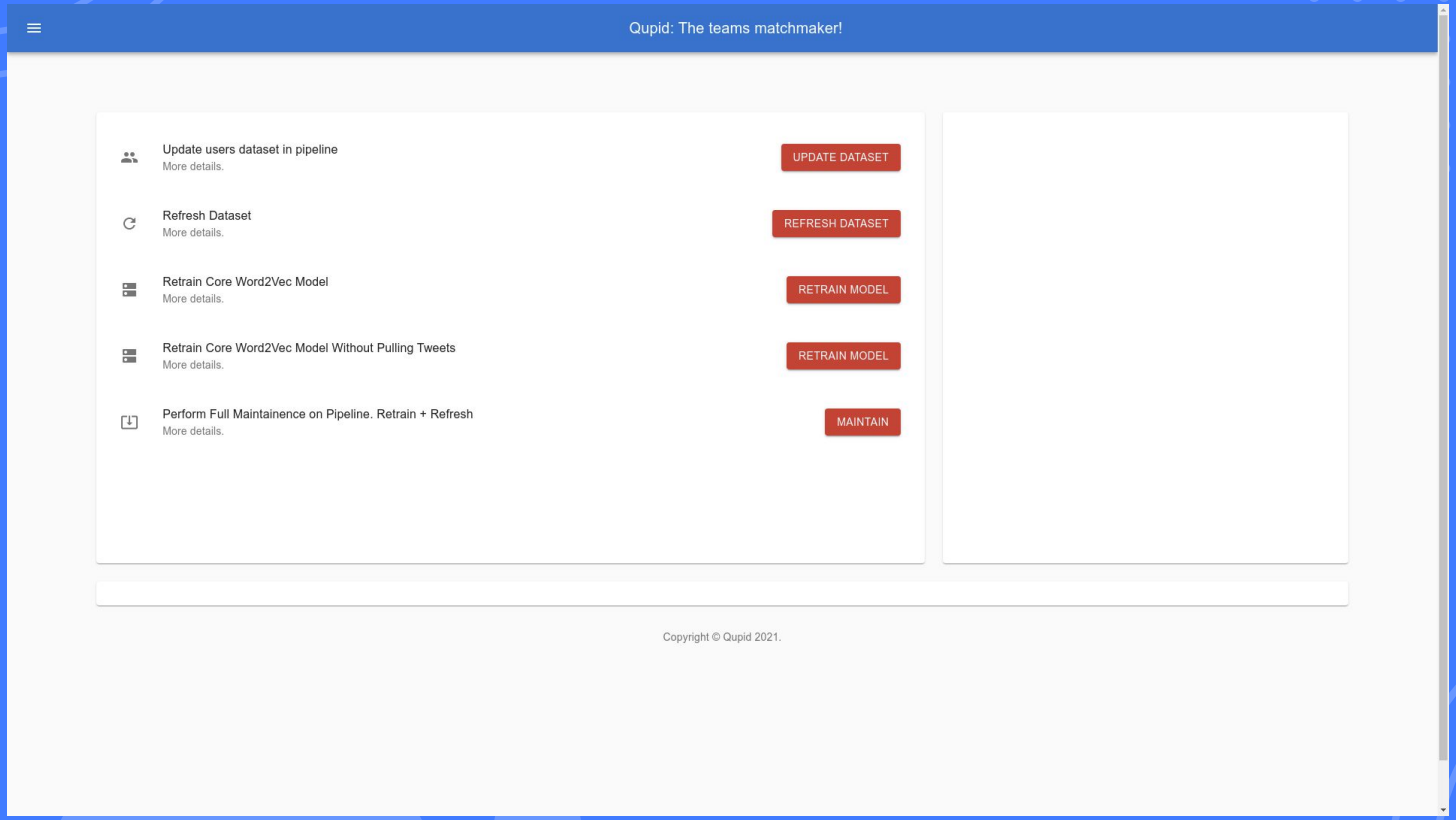
What to do next?

- That's it! Your teams will be cooked and readily server in 3..2..1.. Poof!

Make Teams: Here are your built Teams!



Settings Page



How it works ?

1. This page has some pipeline options. They are to be used carefully since they are data heavy operations. Majorly these are related to:
    a. Data Source Cleanup
    b. Retraining Model
    c. Maintaining Pipeline

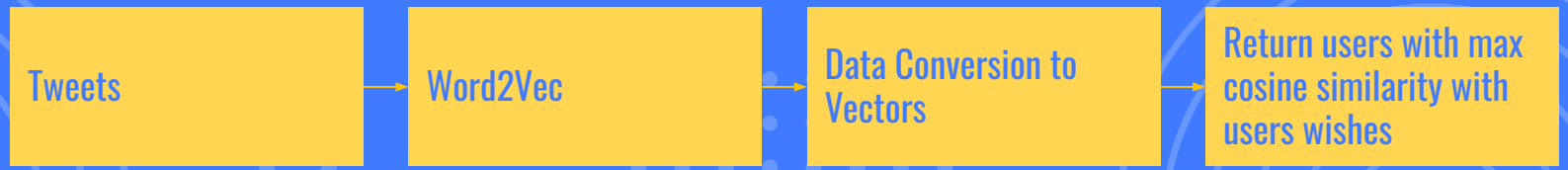
What to do next?

- Get a coffee and wait. Some of these operation take a lot of time. Specially the retrain with tweets scrap one.(Fun fact approximately 20,000 tweets are scrapped for training)

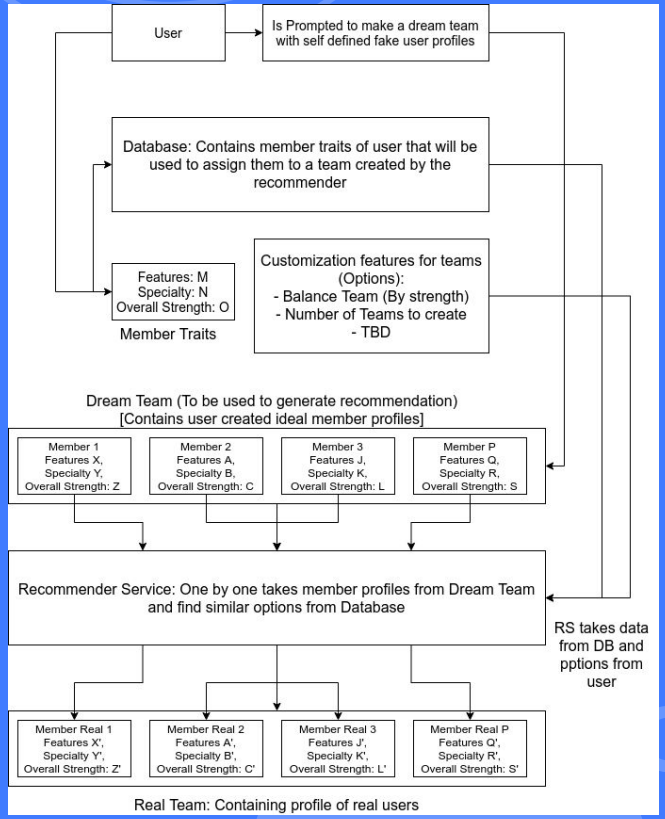
Under the Hood: Algorithm

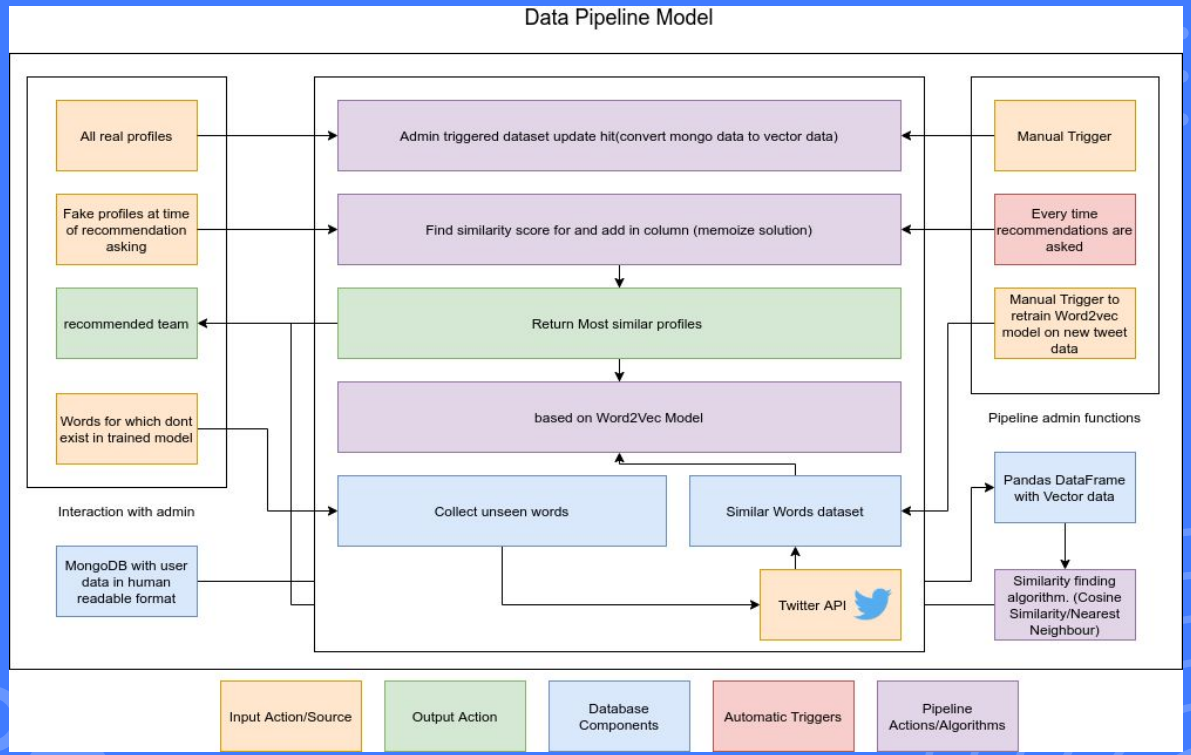
3

All hail draw.io



Prototype Algorithm & Under the Hood Algorithm





Under the Hood: Tech

4

Show me the cool code! Major cool code written related to AOBD: *Train Model*, *Generate Word to Vec Insights*, *Cosine Similarity*, *Teams Creation*

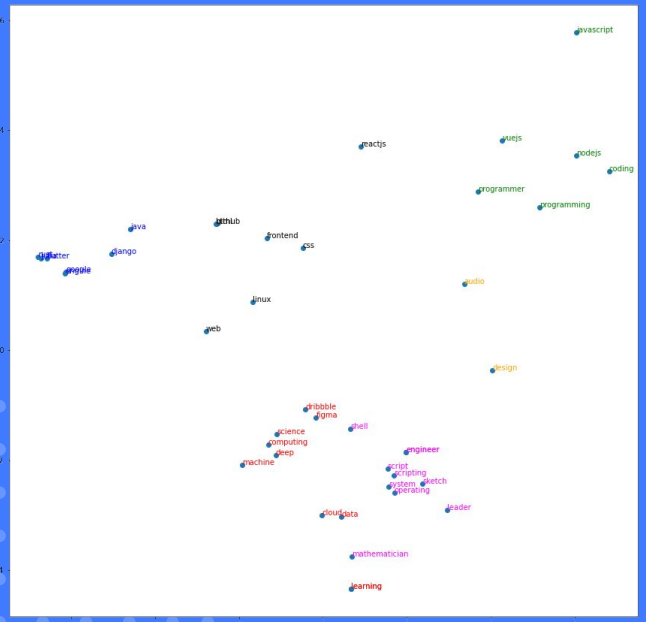
Train Model and Generate Insights

Code Link:
https://github.com/Kaushal1011/Qupid/blob/main/app/server/routes/model/train.py

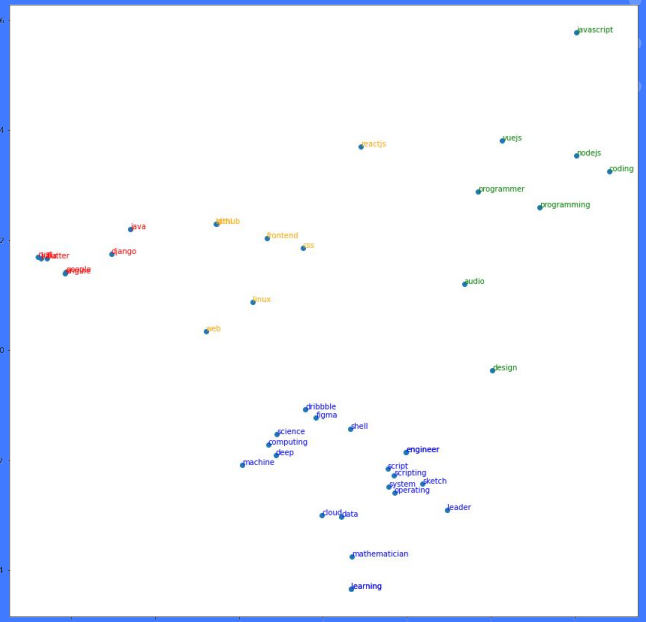
What does it do :

1. Scraps Tweets (based on dynamic oob terms)
2. Clean Tweets (Removes stop words and special chars)
3. Creates a dataframe with all text tweet data.
4. Tokenizes Tweet Data.
5. Trains a gensim Word2Vec model on Tokenized data.
6. Generates Model Insight
    a. Performs 3 Groups K Means Cluster.
    b. Performs 2 PC PCA on selected word vectors
    c. Scatter plots selected word vectors

Different Word2Vec Model Insights



N Clusters = 6

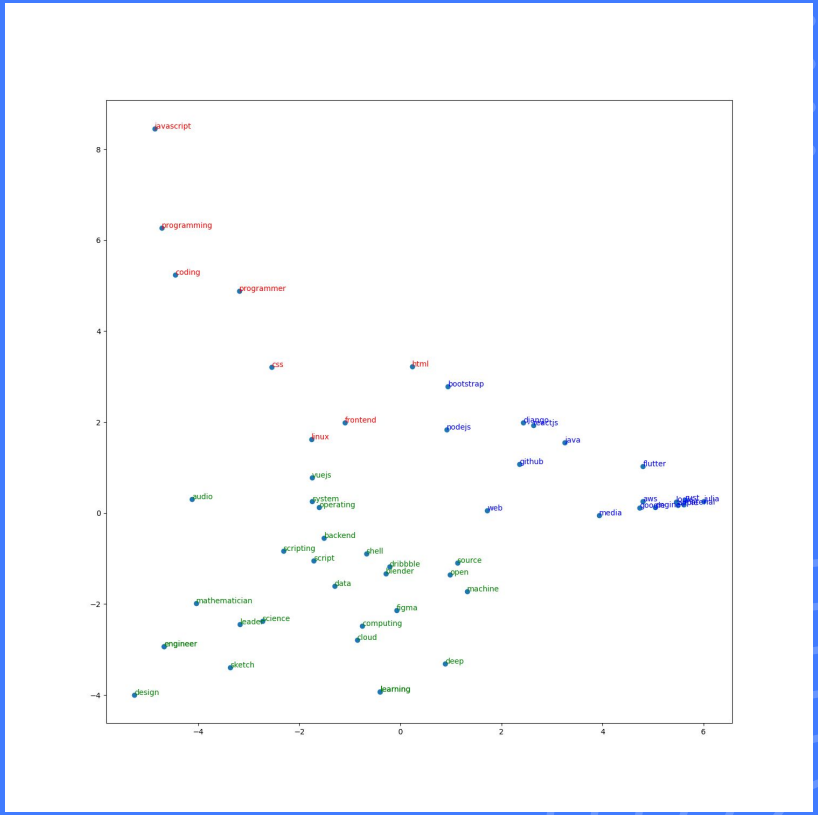


N Clusters = 4

Current Word2Vec Model Insight

This is the model that's currently being used by the application. Some insights:

1. Notice how similar words are closer
   a. ReactJS, NodeJS, Django
   b. Frontend, Html, CSS
2. Notice how some words are at same distance from some words:
   a. Javascript→ Node JS ~= Javascript→ ReactJS
   b. deep→learning ~= machine→learning



Recommend Code & Cosine Similarity

Code Link:
https://github.com/Kaushal1011/Qupid/blob/main/app/server/routes/pipeline.py

What does it do :

1. Takes user wishes' vector data and finds cosine similarity with user data.
   a. SimScore
      formula=(cossim(role)*1.3+cossim(sp1)+cossim(sp2)+cossim(sp3)*0.7)/4
      (More weight given to role over speciality 3).
2. Finds mutually exclusive sets of users for all teams by maintain a hashtable.
3. Balance teams. (Balancing is basically finding X best users for X teams and then assigning each user to a team randomly, this makes sure that team 1 doesn't always gets best matches)

How does our data look?

User Entered Data

Vector Data from W2V

Cached Similarity Score

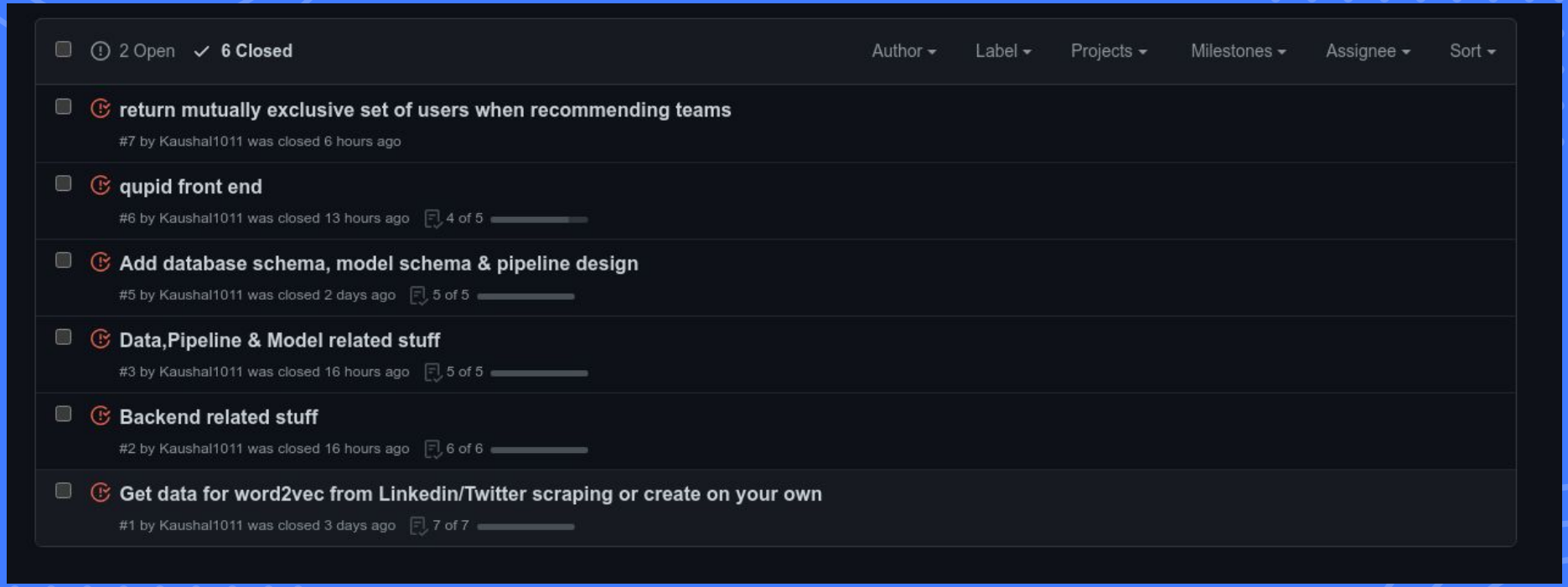
```
[2]: df.head()
```

[2]:

fullname	email	role	specialities	organisation	rolevec	sp1	sp2	sp3	sim603953ab4f8181a538bf3088	sim603aa29a635cc066c2f64844	sim603c1df36870bd179f6c20
Kaushal Patil	kaushalpatil10@gmail.com	fullstack	[python, javascript, backend, frontend, deeple...	Ahmedabad University	[1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, ...	[-1.9521664, -1.1455659, -0.35911146, -0.51699...	[-3.590196, -1.7386764, 0.30715978, -0.7908878...	[-1.1487315, -0.9598047, -1.4830381, -1.308684...	1.000000	0.586689	0.8106
Kaushal Patil123	kaushalpatil20@gmail.com	fullstack	[python, javascript]	Bright Day School	[1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, ...	[-1.9521664, -1.1455659, -0.35911146, -0.51699...	[-3.590196, -1.7386764, 0.30715978, -0.7908878...	[1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, ...	0.810697	0.439924	1.0000
Kaushal Patil	kaushalpatil1@gmail.com	github	[python, javascript, backend, frontend, deeple...	Ahmedabad University	[-0.7634714, -0.5412413, -0.51123375, -0.44693...	[-1.9521664, -1.1455659, -0.35911146, -0.51699...	[-3.590196, -1.7386764, 0.30715978, -0.7908878...	[-1.1487315, -0.9598047, -1.4830381, -1.308684...	0.658122	0.919329	0.4688
Kaushal Patil	kaushalpatil10@gmail.com	frontend	[html, frontend, css, frontend, deeplearning]	Ahmedabad University	[-1.6952097, -0.983109, -0.8618005, -1.0766659...	[-1.729509, -0.86467075, -0.38770512, -0.75679...	[-1.6952097, -0.983109, -0.8618005, -1.0766659...	[-2.2691832, -1.2222046, -0.84275925, -1.21728...	0.603163	0.986846	0.4375
Kaushal Patil 2	kaushalpatil10@gmail.com	fullstack	[python, javascript, backend, frontend, deeple...	Ahmedabad University	[1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, ...	[-1.9521664, -1.1455659, -0.35911146, -0.51699...	[-3.590196, -1.7386764, 0.30715978, -0.7908878...	[-1.1487315, -0.9598047, -1.4830381, -1.308684...	1.000000	0.586689	0.8106

Project Stages

A whole lot of issues were created and closed on github


2 Open
6 Closed
Author
Label
Projects
Milestones
Assignee
Sort
return mutually exclusive set of users when recommending teams
#7 by Kaushal1011 was closed 6 hours ago
qupid front end
#6 by Kaushal1011 was closed 13 hours ago
4 of 5
Add database schema, model schema & pipeline design
#5 by Kaushal1011 was closed 2 days ago
5 of 5
Data,Pipeline & Model related stuff
#3 by Kaushal1011 was closed 16 hours ago
5 of 5
Backend related stuff
#2 by Kaushal1011 was closed 16 hours ago
6 of 6
Get data for word2vec from Linkedin/Twitter scraping or create on your own
#1 by Kaushal1011 was closed 3 days ago
7 of 7

Future and Challenges

6

Will qupid really become the angel that makes teams?

Challenges

1. Integrating Recommendation algorithm in a pipeline sense.
2. Creating Mutually Exclusive Set of teams required some thinking and use of hashtables
3. Usage of csv to manage vector data. This was later changed to pickle file.
4. Writing code the pandas/python way.

Future

1. A lot of optimisation tasks can be done in the pipeline to make it faster and scale up this project
2. Use of data scraped from something such as linkedin can work better over tweets as linkedin has professional user data and profile.
3. Collaborative filtering can be used to recommend user wishes over showing all user wishes at one place.

Thanks!

Do you have any questions?

Follow the project updates

https://github.com/Kaushal1011/Qupid

https://www.linkedin.com/in/kaushal1011/