

Design Specification Review

March, 2019

Team15

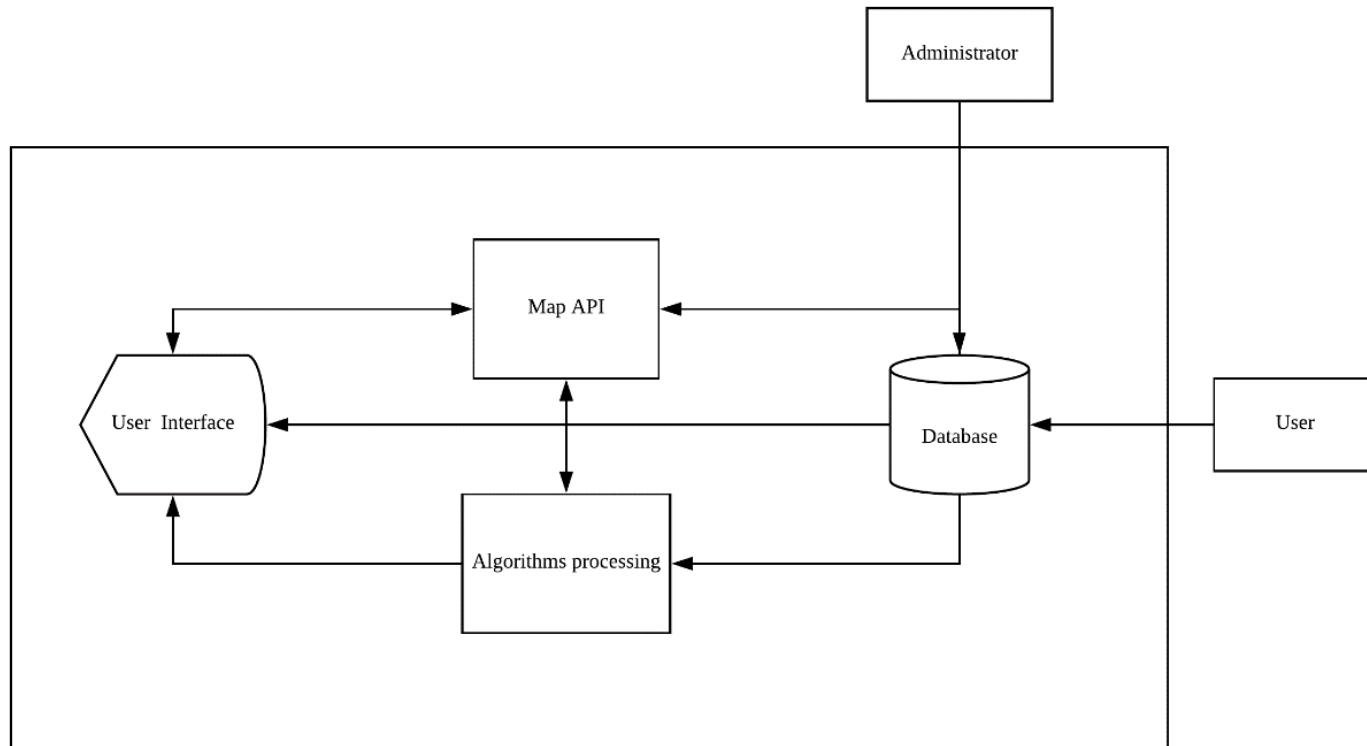
Summary of Proposal

- Inspiration
- Feature
- Project Goal

Features

- Standard Recommendation
- Preference Recommendation
- Ultimate Recommendation

System Boundary Diagram

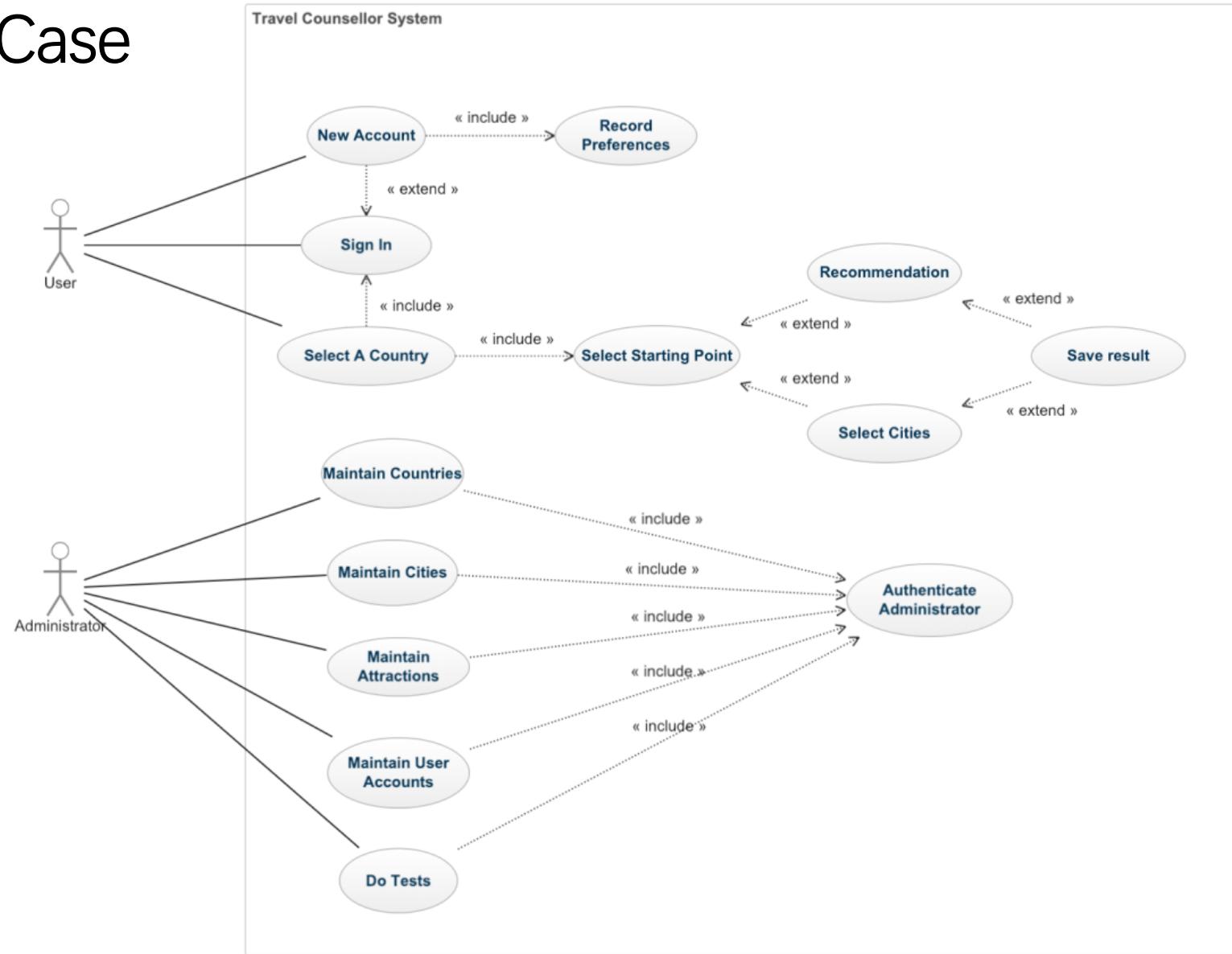


Database: preference, history queries, destination, attractions

Algorithms: route solution(ant colony Optimization), Travelling time, distance, Sorting Algorithm, city recommendation

Map API: external navigation application

Use Case



User Views

Data	Access Type	User	Administrator
Countries	Maintain		X
	Search	X	X
	Report		X
Cities	Maintain		X
	Search	X	X
	Report		X
Attractions	Maintain		X
	Search	X	X
	Report		X
Preference	Maintain	X(registered)	X
	Search		X
	Report		X

Data Structure Used in the System

- Pandas – Dataframe
- Two dimensional tabular data structure
- Size mutable
- Support operations align on both row and column labels
- Realizable for further extrapolation algorithms

Tourism Attractions Info Data Tables

	Country				
Country	Continent	Cities	Nearby Country	Official Website	Travel Label
China	Asia	China[...]	China[...]		
Japan	Asia	Japan[...]	Japan[...]		
UK	Europe	UK[...]	UK[...]		
USA	America	USA[...]	USA[...]		

Specification of City Info Data Tables

Attractions							
Attraction	ID of attraction	Latitude	Longitude	Address	Category	Detail Info Website	Abstract Stay Time
The Forbidden City							
Tiananmen Square							
Beijing Bird's Nest							

User Account Info Data Table

User Account							
Name	Email	Password	ID	Preference	City	Accomodations	History
User 1							
User 2							
.....							

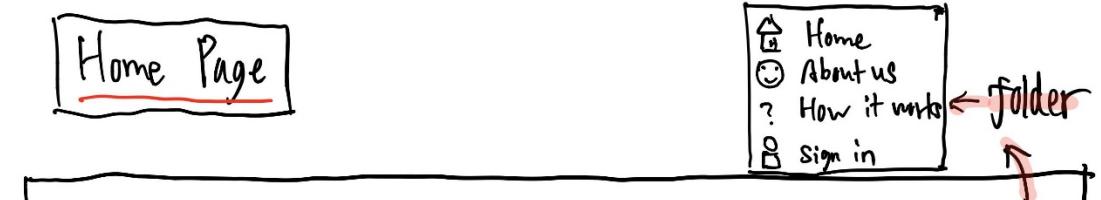
History				
Name	Time	Route	Duriation	Result URL
User 1				
User 2				
.....				

Algorithm to manipulate the Data Structure:

- Retrieve and Screen the Data
- Update the Value in the Data Frame
- Delete Data in the Data Frame
- Insert New Data Record into the Data Frame
- Traverse the Data Efficiently
- Sort the Data
- Handle the Null Value in the Data Frame

Design of intended user interface (UI)

- Design sketches
- Major scenarios of user-system interactions



1

(Logo)

Language ▾

(Image here)

— YOUR TRAVEL COUNSELLOR —

MAKE TRAVEL EASIER

Brief introduction

START

COUNTRY (E.g. UK) ▾

Customise your route

START PLACE

Liverpool ▾

END PLACE

London ▾

2

CITIES COVERED

Manchester	<input type="checkbox"/>	Birmingham	<input type="checkbox"/>
Leeds	<input type="checkbox"/>	Lancaster	<input type="checkbox"/>
York	<input type="checkbox"/>	Newcastle	<input type="checkbox"/>

START DATE

June 1, 2019 ▾

END DATE

June 10, 2019 ▾

Want to customise further?

Tell us more

fold if not click

3

(Preference)

NUMBER OF STOPS

A few

Good range

A lot

ACTIVITY TYPE

GO

4

LINKS

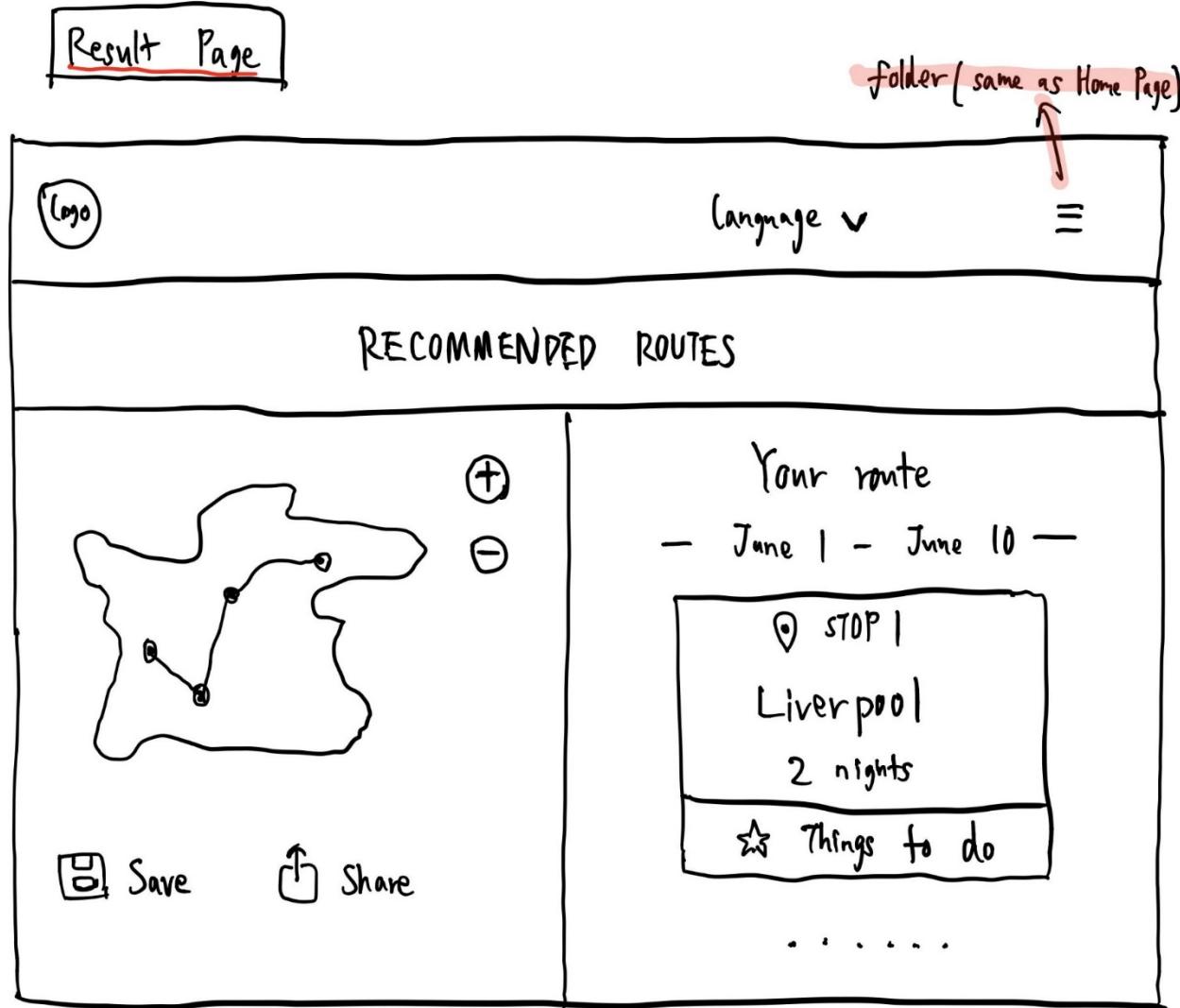
Book hotel

Book flight

GET IN TOUCH

Email

Phone



“Links” part is same as HomePage (quick link to booking websites)

Evaluation of the System

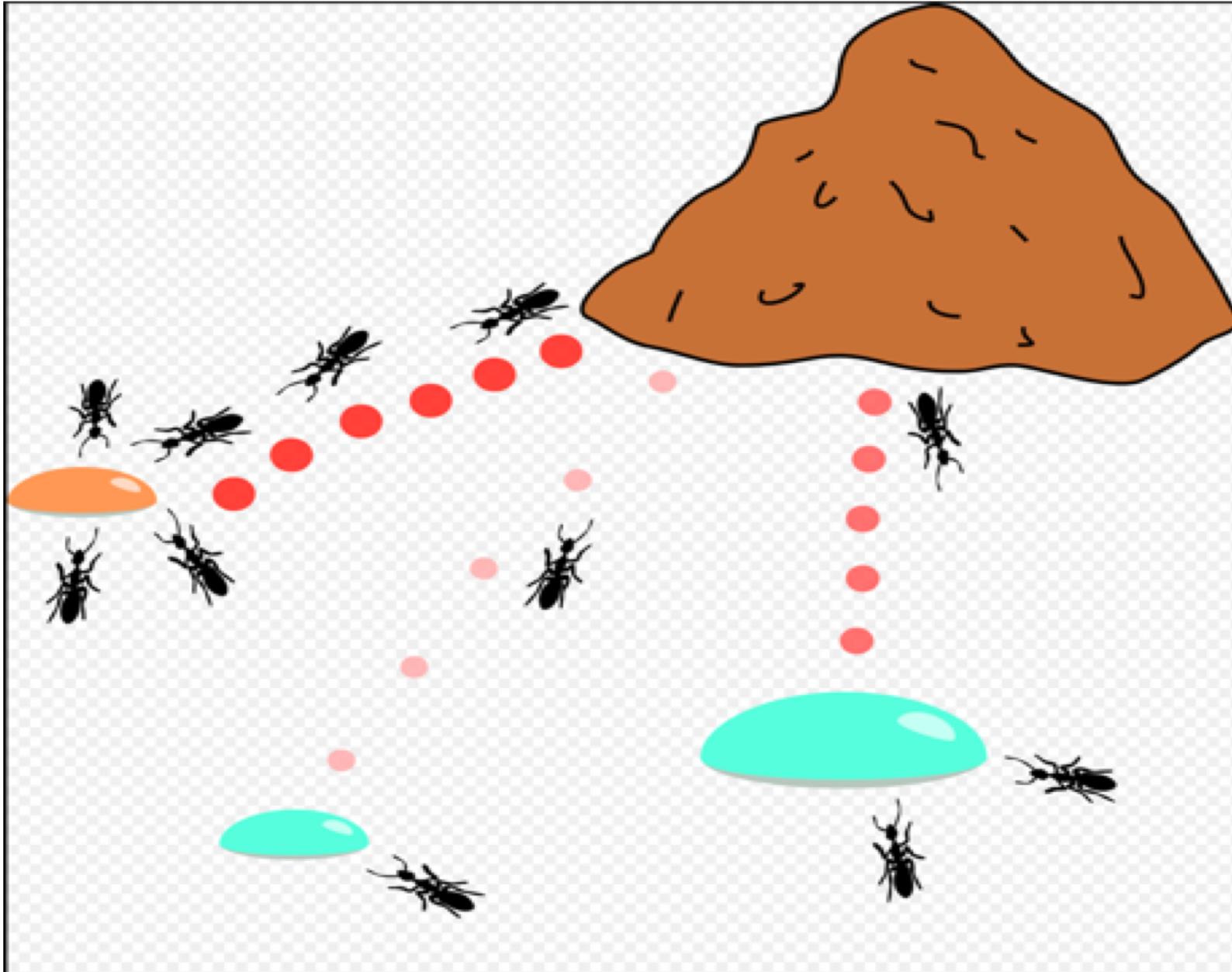
- Basic functionality
- Complexity
- Testing methods
- Accuracy of key features
- User experience

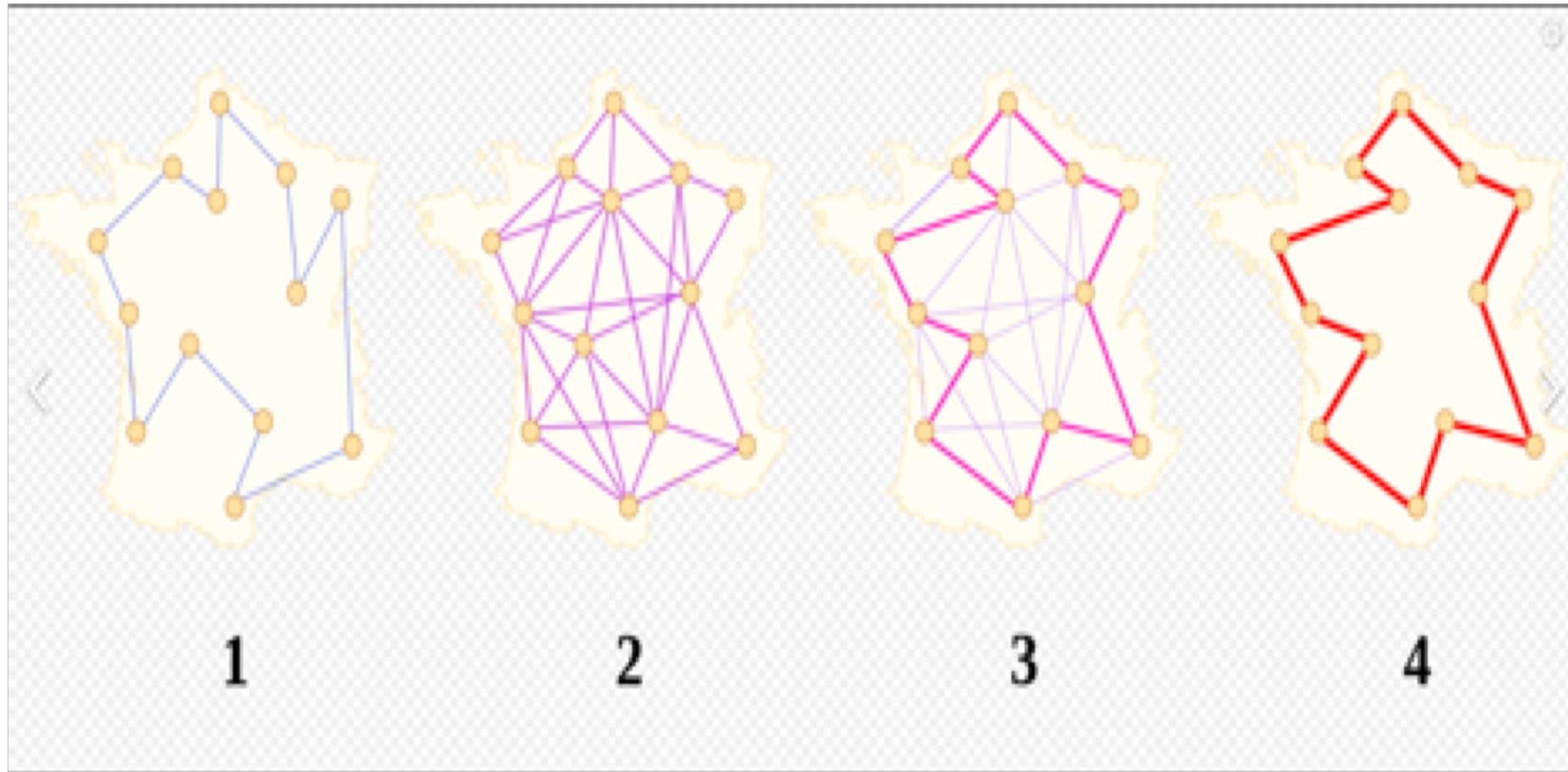
Introduction of Key Method

- Ant Colony Optimization
- Sorting Method
- Process of Calculating Similarity
- Recommend Travel Route

Ant Colony Optimization

- Find shortest path through graph
- Normal ant releases pheromones, directing each other to resources to path
- “Smart” ant record route and make judgement about whether route is appropriate
- After many times iterations, small ant locate better solution





ANTCOLONYOPTIMIZATION(S, E, G, M)

▷Input: Location of start city S , location of end city E , $G(V, E)$ of all the cities selected and Set M of ants.

▷Output: Path route P of the result.

```
1   for  $i \leftarrow 1$  to  $|C|$  do
2        $distmat[i][j] \leftarrow$  distance between city  $i$  and city  $j$ 
3   procedure SETINITINFORMATION
4       for  $\forall k \in M$  do
5           Let  $r_{k1}$  be the starting city for ant  $k$ 
6            $J_k(r_{k1}) \leftarrow V - \{r_{k1}\}$ 
7           /* the set to be visited for ant  $k$  in city  $r$ */
8            $r_k \leftarrow r_{k1}$ 
9           /* city the ant  $k$  located in */
10  end procedure
11
12  procedure ROUTES
13      for  $i \leftarrow 1$  to  $|V| - 1$  do
14          for  $\forall k \in M$  do
15              Select next city  $s_k$  from the formula mentioned
16              add  $edge(r^k, s^k)$  to  $Tour_k$ 
17  end procedure
```

```

procedure UPDATE
18          Compute  $L_k \forall k \in M$ 
19          /* the length of tour of ant  $k$  */
20          Update  $\tau_{r,s}$  from the formula mentioned
21 end procedure

22 procedure MAIN
23     for  $\forall edge(r,s) \in E$  do
24          $\tau_{r,s} \leftarrow \tau_0$ 
25          $\eta_{r,s} \leftarrow 1/distmat[i][j]$ 
26     while Not End_Condition do
27         SETINITINFORMATION
28         ROUTES
29         UPDATE
30 end procedure

```

Sorting Method

- Mark cities based on data or comment of main travel platform.
- Mark user preference based on answer of questionnaire.
- calculate similarity.

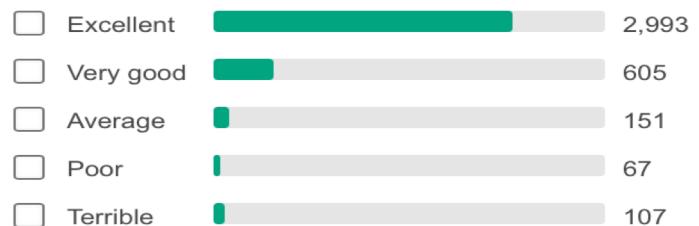
Process of calculating similarity

- calculating Euler distance between user vector and city vector.
- Definition of similarity

Reviews (4,450)

[Write a Review](#)

Traveller rating



Traveller type

- Families
- Couples
- Solo
- Business
- Friends

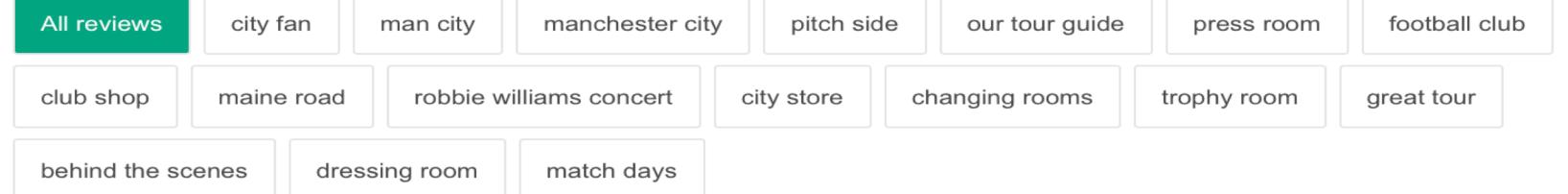
Time of year

- Mar-May
- Jun-Aug
- Sep-Nov
- Dec-Feb

Language

- All languages
 - English (3,923)
 - Chinese (Sim.) (115)
 - Chinese (Trad.) (104)
- [More languages ▾](#)

Show reviews that mention



1 - 10 of 3,923 reviews



santibolanos7
Quito, Ecuador



Reviewed today

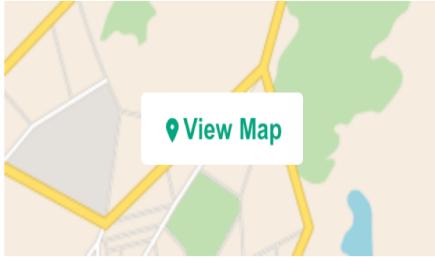
Amazing Experience for all Football Lovers

The place is packed with amazing stories and fun facts along with a stunning atmosphere. The tour guides have a vast knowledge of football and usually share their own City experiences, making it a little more personal. The place looks beautiful in every single corner.... [More](#)

Date of experience: March 2019

Thank santibolanos7





Types of Things to Do

Top Attractions

Shop Tours & Tickets

Types of Attractions

- Museums (31)
- Fun & Games (76)
- Sights & Landmarks (62)
- Nightlife (219)
- Concerts & Shows (28)
- Traveller Resources (10)
- Shopping (67)

More ▾

Airports Nearby

Manchester Intl Airport (4)

Shop Tickets And Tours (25)

Shop Now



Tours & Sightseeing



Cultural & Theme Tours



Multi-day & Extended Tours

Top Things to do (164)

See all



Etihad Stadium

 4,479 Reviews

Arenas & Stadiums



Manchester Three Ri...

 380 Reviews

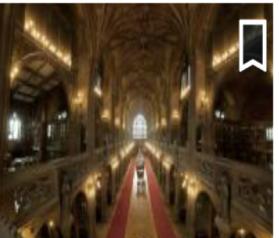
Distilleries



Manchester Climbing ...

 162 Reviews

Sports Complexes



The John Rylands Lib...

 4,982 Reviews

Libraries

Fun & Games (76)

See all



CITYSORT(U, C)

▷Input: Preference of the user U which stores an unidimensional array, and $C[i][j]$ array stores the degree of factors for all the cities in a country.

▷Output: Sorted list S of cities

```
1      difference[] = 0
2      for  $i \leftarrow 1$  to  $|C|$  do
3           $sum \leftarrow 0$ 
4          for  $j \leftarrow 1$  to  $|C[1]|$  do
5               $sum \leftarrow sum + (U[j] - C[i][j])^2$ 
6               $sum \leftarrow sum^{1/2}$ 
7              /* L2 norm of the difference between user preference
8                 and the city */
9              difference[ $i$ ]  $\leftarrow sum$ 
10
11      procedure MERGESORT(difference[0.. $n-1$ ])
12      if  $n > 1$  then begin
13          copy difference[0.. $\lfloor n/2 \rfloor - 1$ ] to  $B[0.. \lfloor n/2 \rfloor - 1]$ 
14          copy difference[ $\lfloor n/2 \rfloor .. n-1$ ] to  $C[0.. \lceil n/2 \rceil - 1]$ 
15          MERGESORT( $B[0.. \lfloor n/2 \rfloor - 1]$ )
16          MERGESORT( $C[0.. \lceil n/2 \rceil - 1]$ )
17          MERGE( $B, C, difference$ )
18      end procedure
```

```
18  procedure MERGE( $B[0..p-1]$ ,  $C[0..q-1]$ ,  $A[0..p+q-1]$ )
19      Set  $i \leftarrow 0$ ,  $j \leftarrow 0$ ,  $k \leftarrow 0$ 
20      while  $i < p$  and  $j < q$  do
21          begin
22              if  $B[i] \leq C[j]$  then set  $A[k] = B[i]$  and increase  $i$ 
23              else set  $A[k] \leftarrow C[j]$  and increase  $j$ 
24           $k \leftarrow k+1$ 
25      end
26      if  $i \leftarrow p$  then copy  $C[j..q-1]$  to  $A[k..p+q-1]$ 
27      else copy  $B[i..p-1]$  to  $A[k..p+q-1]$ 
```

Recommend travel route

- Based on user preference and distance between different cities.
- confirm starting point
- list first $2*n$ cities
- calculate distance between origin place and cities, select city which has shortest path to origin place

CITYRECOMENDATION (N, L, S, P)

▷Input: N: Number of cities N selected. L: Location of all the cities. S: sorted list of cities according to user preference. P: Starting point of the route

▷Output: R: route path

city \leftarrow Select first $N*2$ cities from array S

present \leftarrow P

Add present to R

 while city $\neq \emptyset$

 do

 Remove the closest city c with present

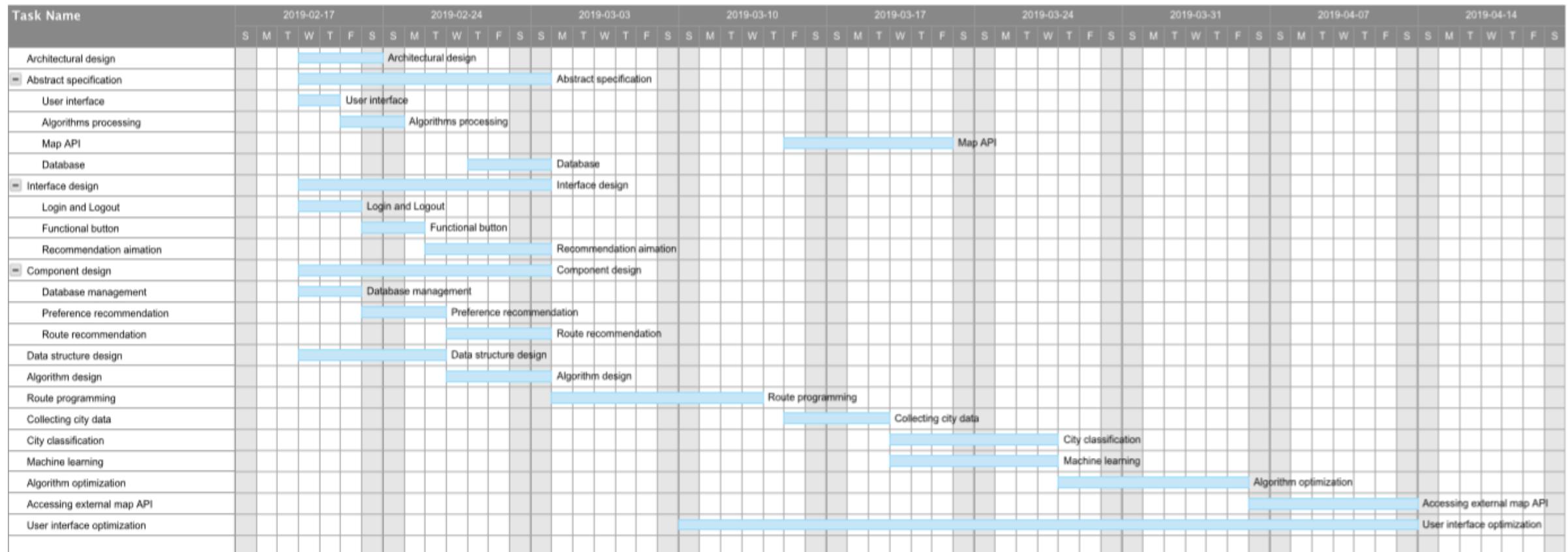
 Add c to R

 Return the set R

Review Against Plan

- Gantt Chart
- Current progress
- Outlook of Implementation Stage

Gantt Diagram



Outlook of Implement stage

- System: Python with pandas library, SQL
- Function Realizations: Queries
- Main Algorithm Implementation
- User Interface Design: HTML & CSS, JavaScript

Thanks

@Group 15
2019.3.20