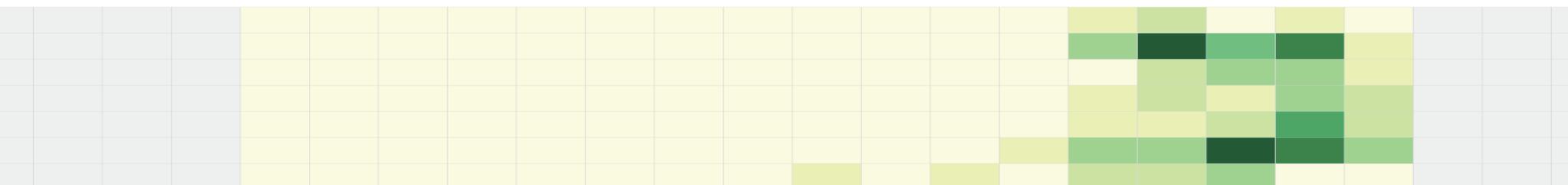


Process Book

CS 171 - Visualization

Noorain Ahmed
Gaurav Gupta
Yingyi Wang



Introduction

Our project tries to visualize the data of Yelp reviews and check-in.

The goal is to let the user quickly find the best suitable restaurant around them.



When we use Yelp to find a restaurant to eat, we are great influenced by several options.

We care about the distance. So we start with the map that people can find the county they are located and give sum of the restaurants on it. After we get into the county level, it will list all the restaurants inside it and update the information of all the restaurants.

Each circle represents one restaurant. Obviously, we need to know whether it is open or not at this moment. Restaurant that open will come out compared to those are closed with the color difference. They are clustered according to the price now because we want to know the price . We can recategorize them by food type if they want to choose one or several types of food that we prefer.

Most likely we will choose the restaurant with the highest reviews. And the radius of each circle is related to their reviews. But the average score sometimes is not reliable and people still will go through the detailed reviews.

So we embed the most important function in our visualization that people could recalculate the score of the restaurant according to the votes of the reviews. They can put weight on different votes which affects the calculation. For example, if people put 2 weights on the useful reviews, so the review which has 10 votes may be calculated twice.

With that, when we select one restaurant, it will update the detailed information of the restaurant and the check-in heatmap which indicates the busy time of it. So people will know whether it is the best time to go because it is not busy now.

So this is our visualization, which focuses on more frequent used features and help users to quickly filter the restaurants they want.

Introduction

Milestone in developing process

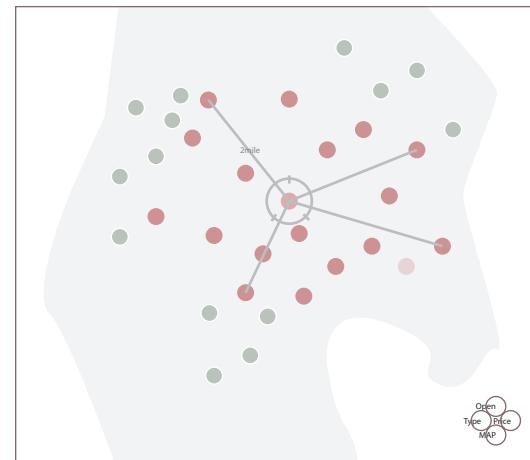
Milestone 1

The concept
Check-in Heatmap



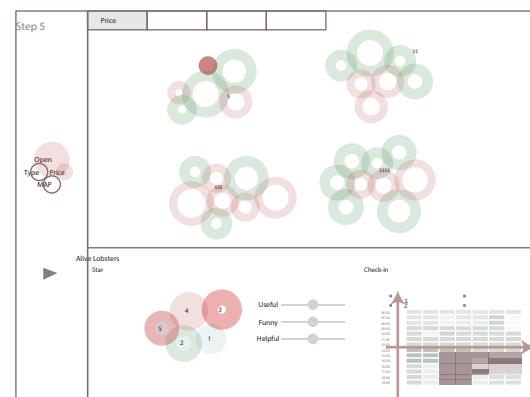
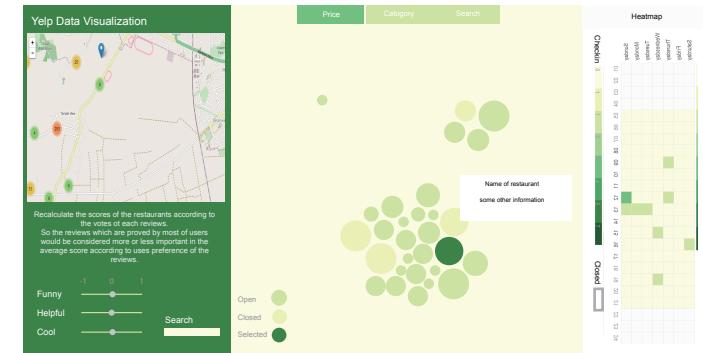
Milestone 2

Selection of data
Reviews become more important



Milestone 3

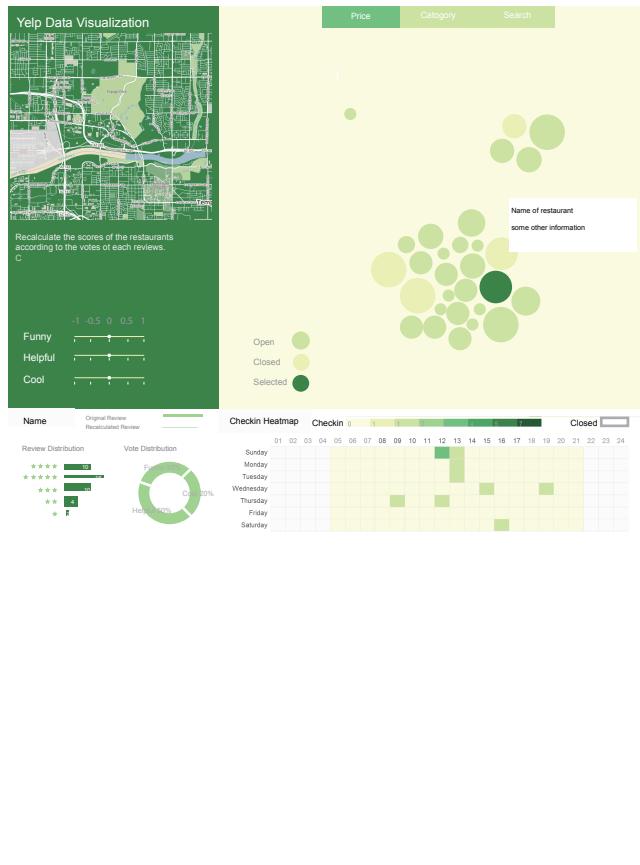
Color Selection
Merge steps to one interface



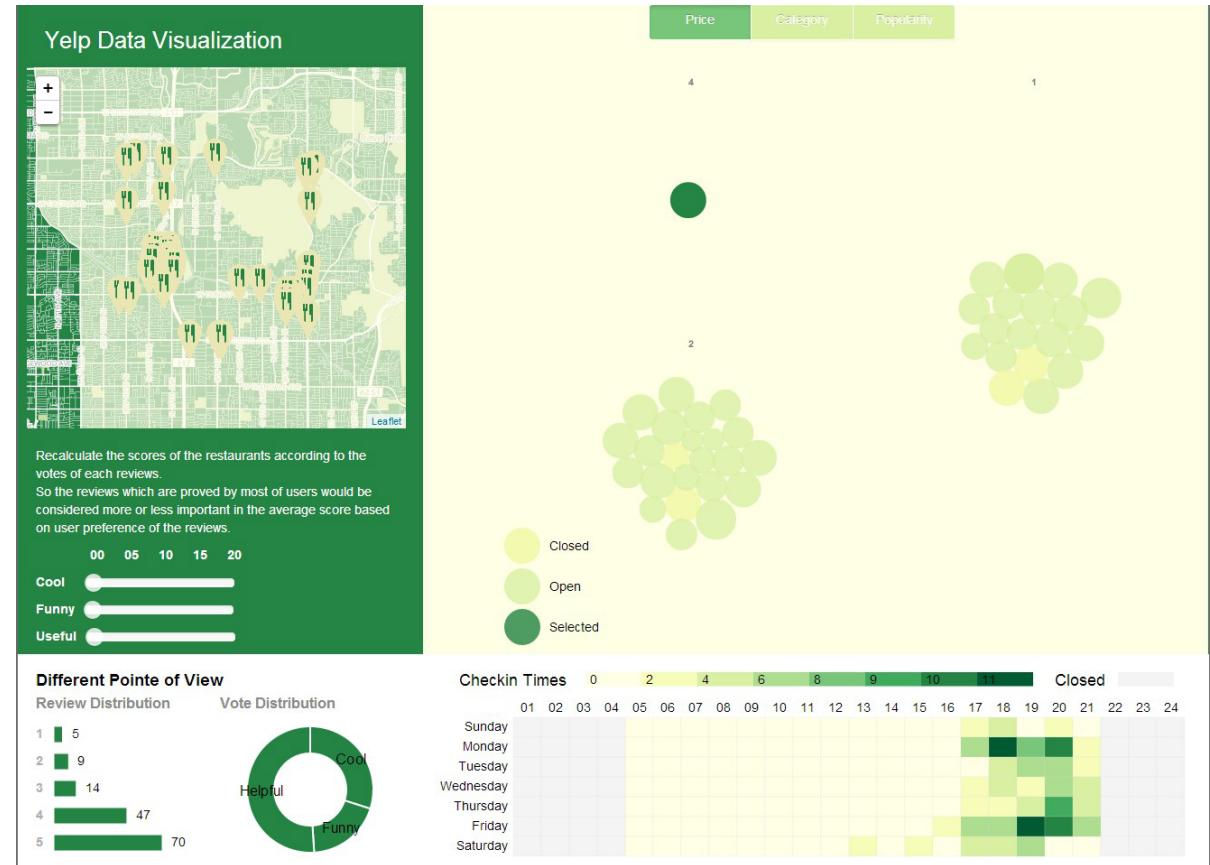
Preview

Final Diagram

Layout we want to achieve



Screen Cast for Final Result

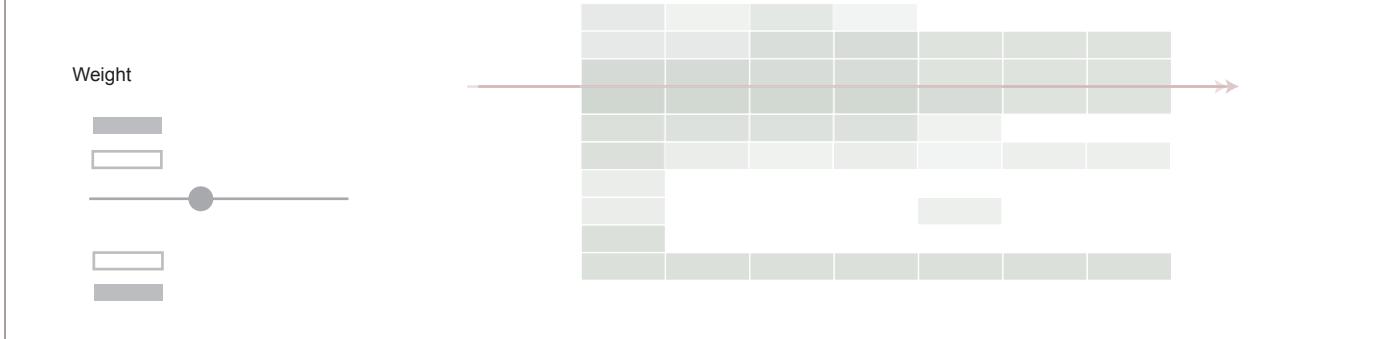


Milestone 1

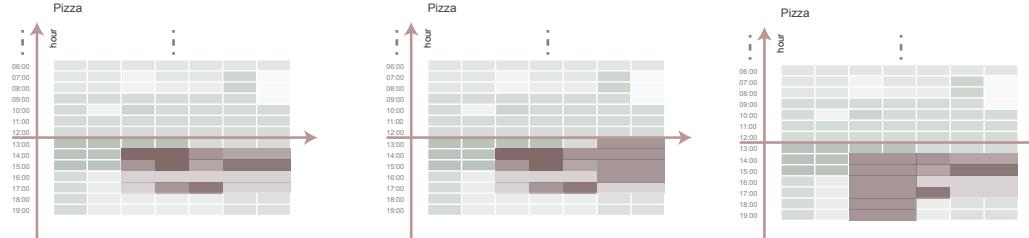
Checkin heatmap



Heat map or recalculated result of the business map according to time

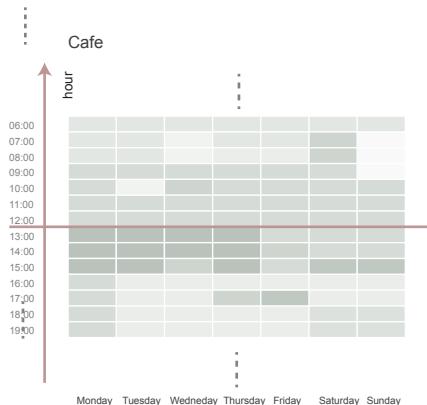


Solutions



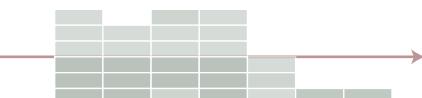
Algorithm for heatmap overlay

Generate heat map of check-in data according to our selection of restaurants. (For example, according to restaurant type)



Filter Top 30%
➡

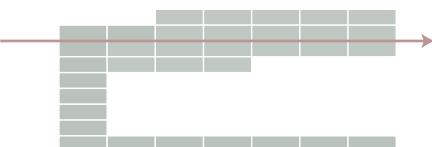
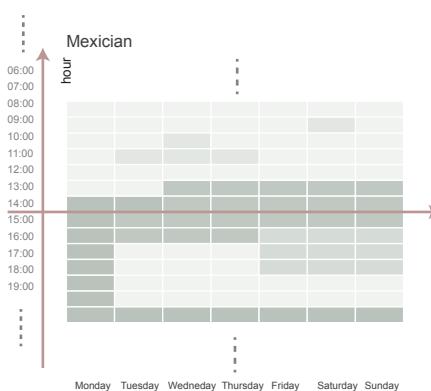
Generate heat map of check-in data according to our selection of restaurants. (For example, according to restaurant type)



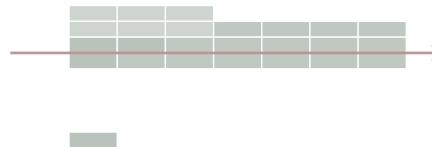
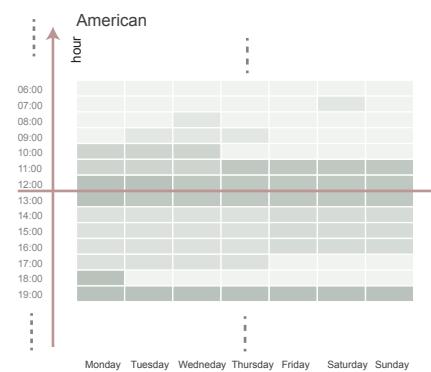
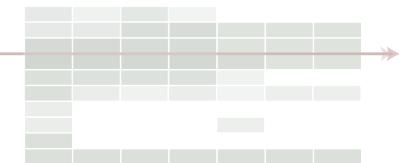
Overlap the different data
➡

To see the missing part, which could be an opportunity to open one restaurant which performs quite good in the time area.

We could offer some potential solutions according to our analysis.



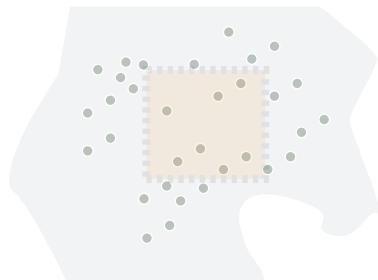
➡



➡

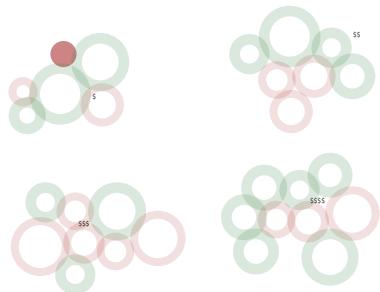
Milestone 2

Step 1&2



We divide our project into 5 steps. The 1&2 steps would be a Martini Glass which let the user to choose the restaurants in their areas.

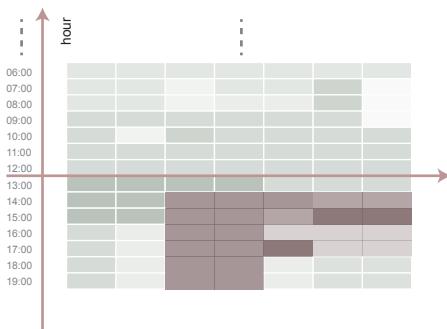
Step 3



And in 3rd step then we visualize them and categorize them in different ways to let the user see how several restaurants they are interested share how many features they would prefer.

Because the problem for the Yelp is sometimes you would miss a very suitable restaurant because you put one more limitation and not that flexible in a way. And in 4th step we visualize the detailed information for the restaurants that people want to see.

Step 4



And 5th step allows people to recalculate the score of the restaurant according to the weights they put in the votes of the reviews.

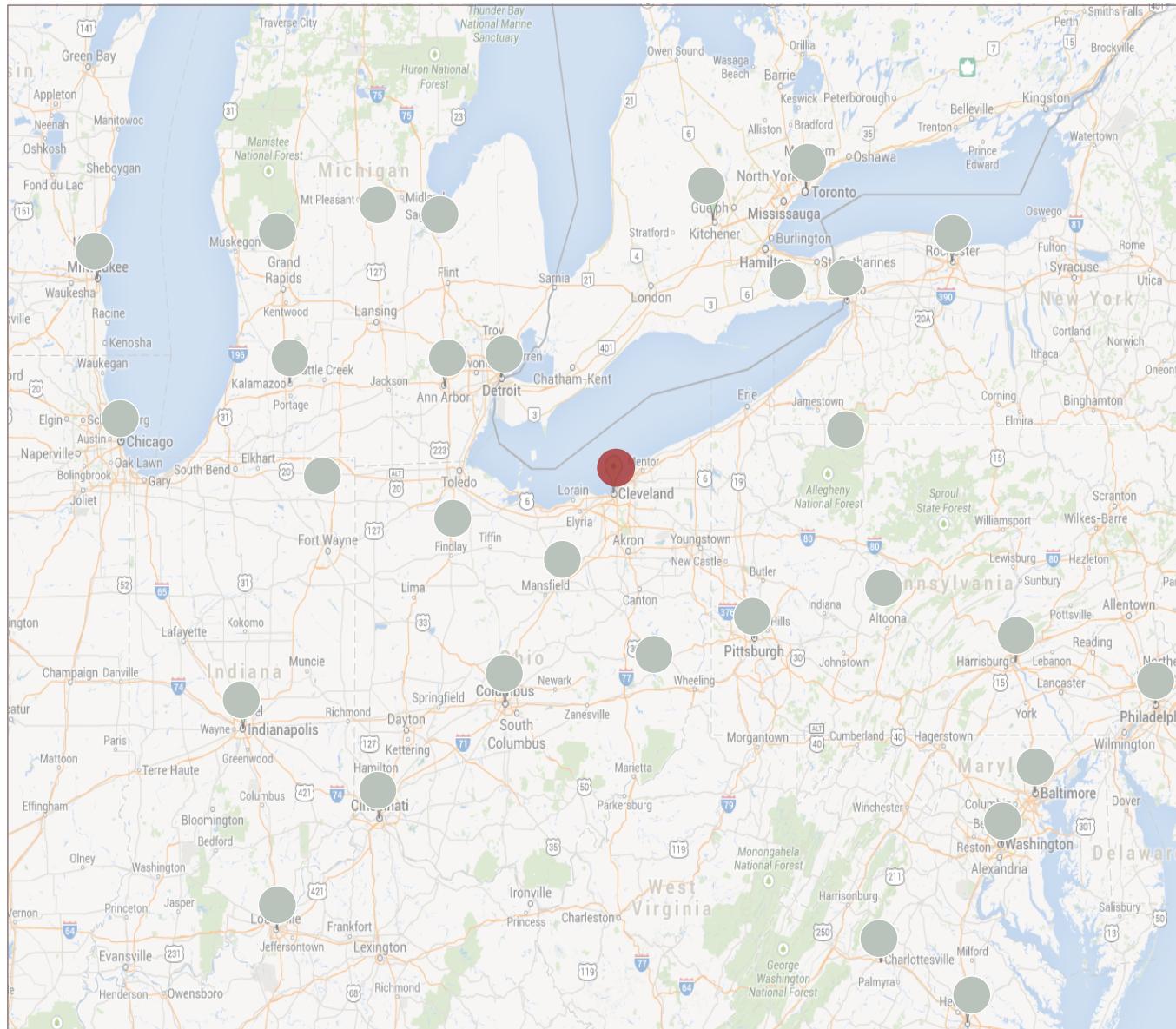
And detailed information will be explained with diagrams in later page.

Step 5



Milestone 2

Step 1

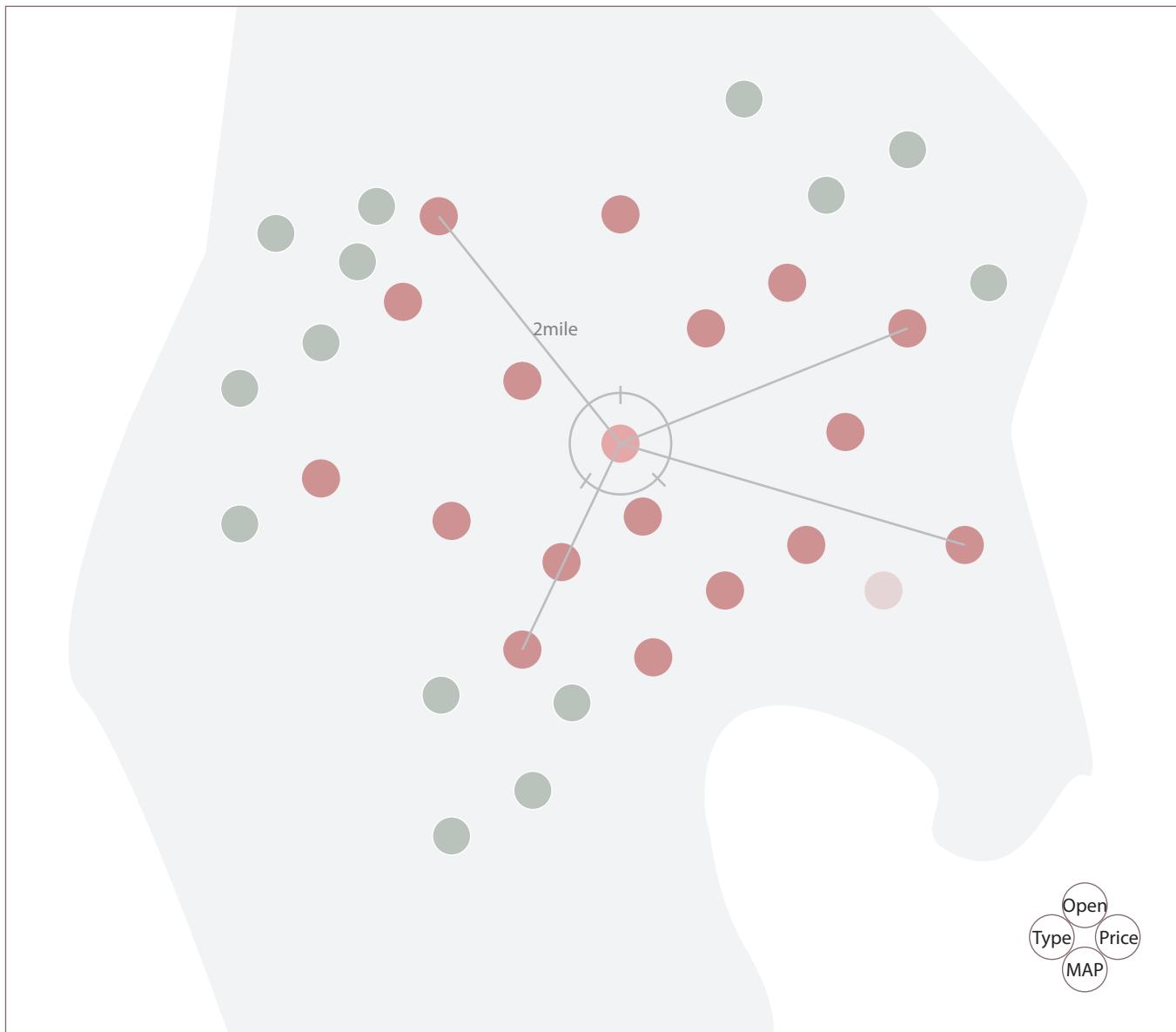


The circle represents the center
of the counties in Phonix

The chosen county
Will transition to next page

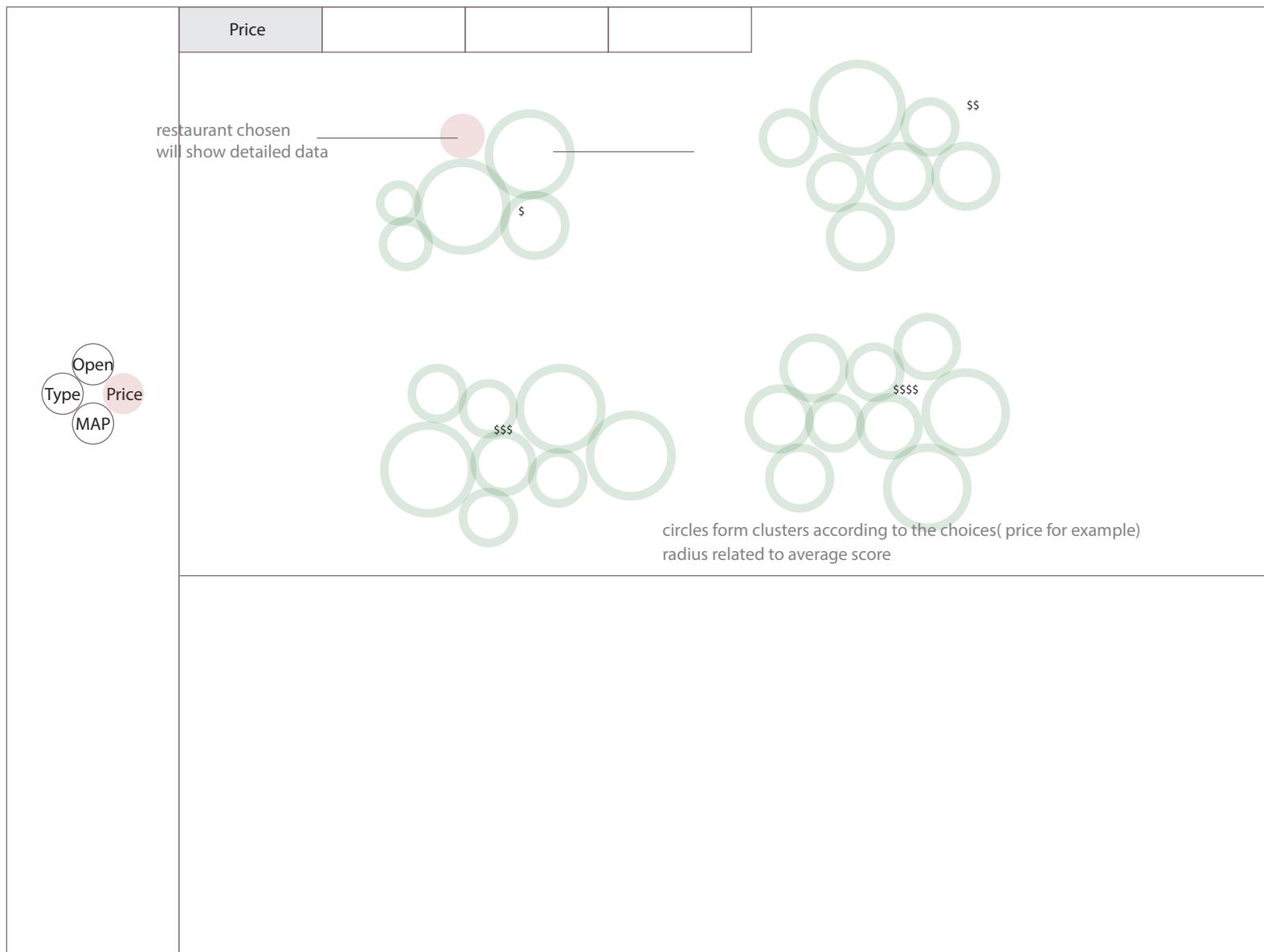
Milestone 2

Step 2



Milestone 2

Step 3



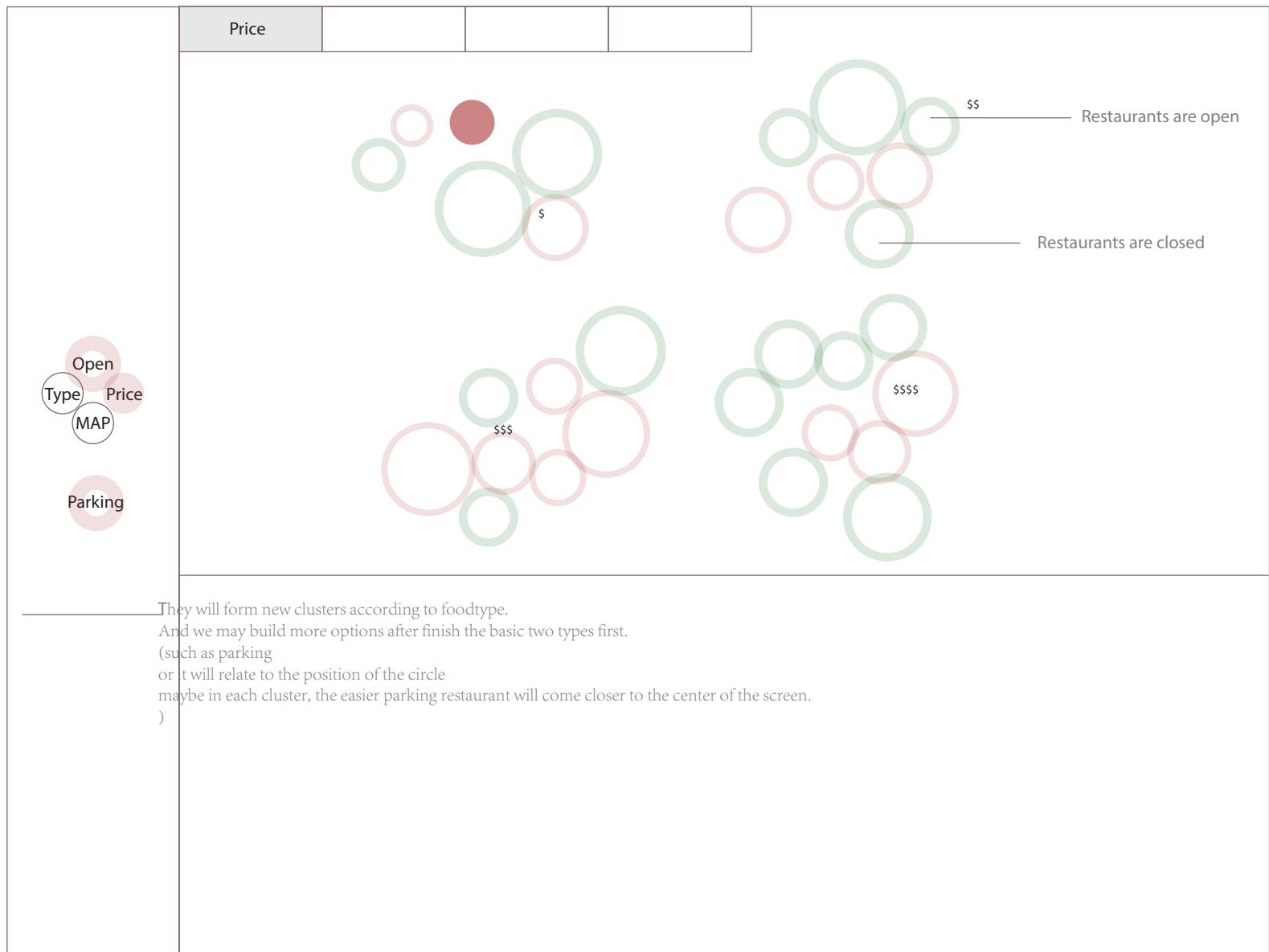
Milestone 2

Step 3_1



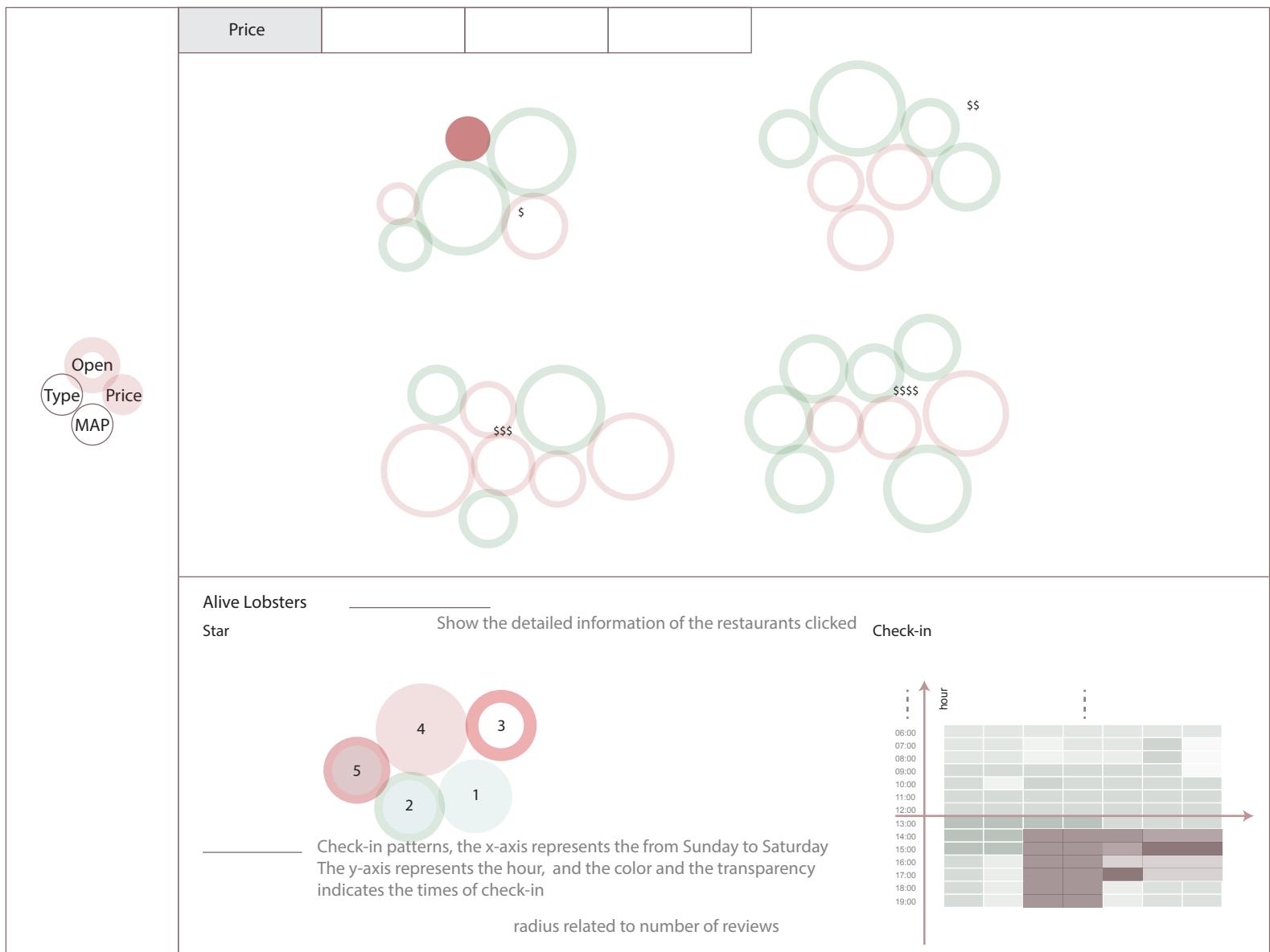
Milestone 2

Step 3_2



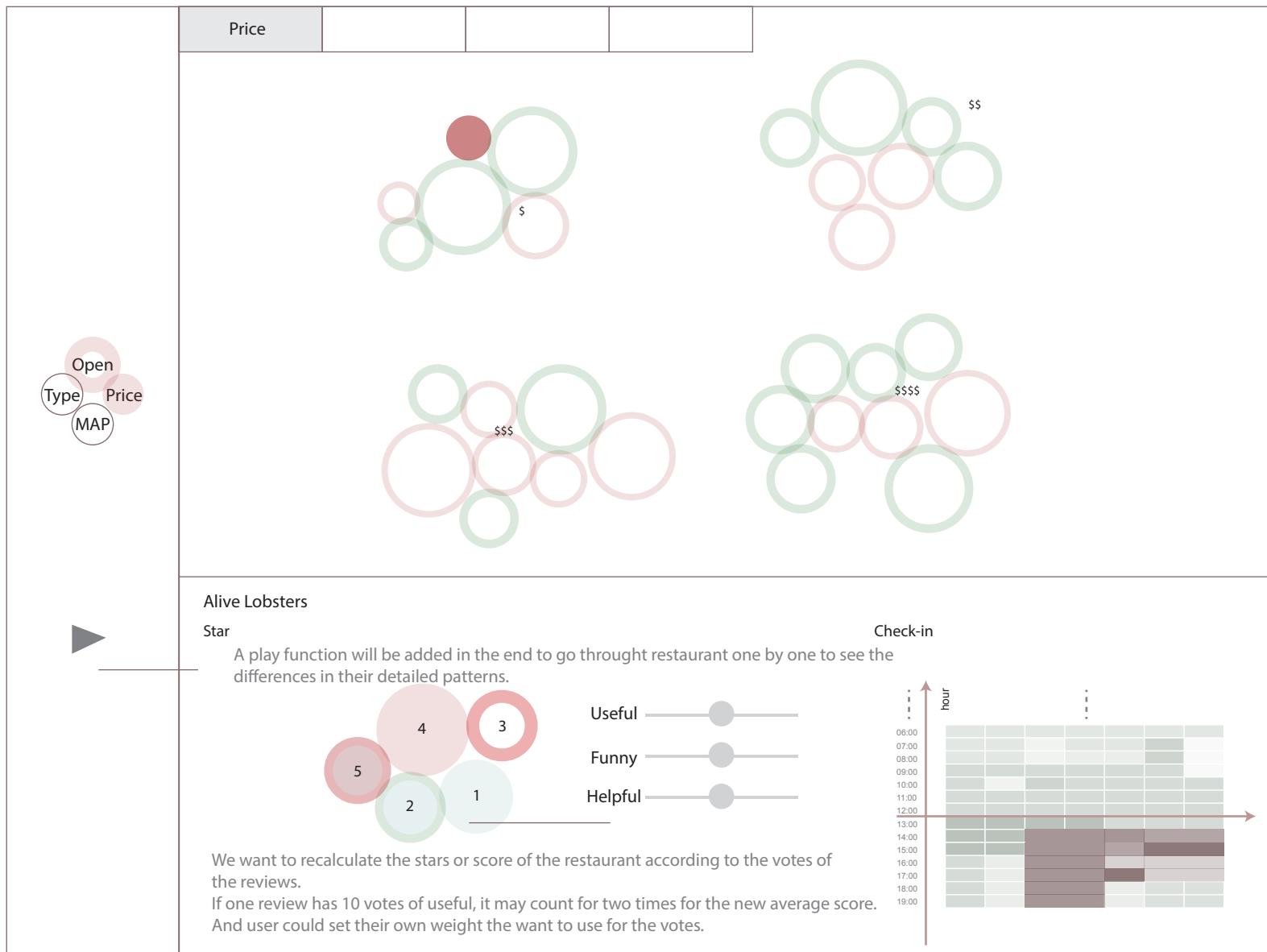
Milestone 2

Step 4



Milestone 2

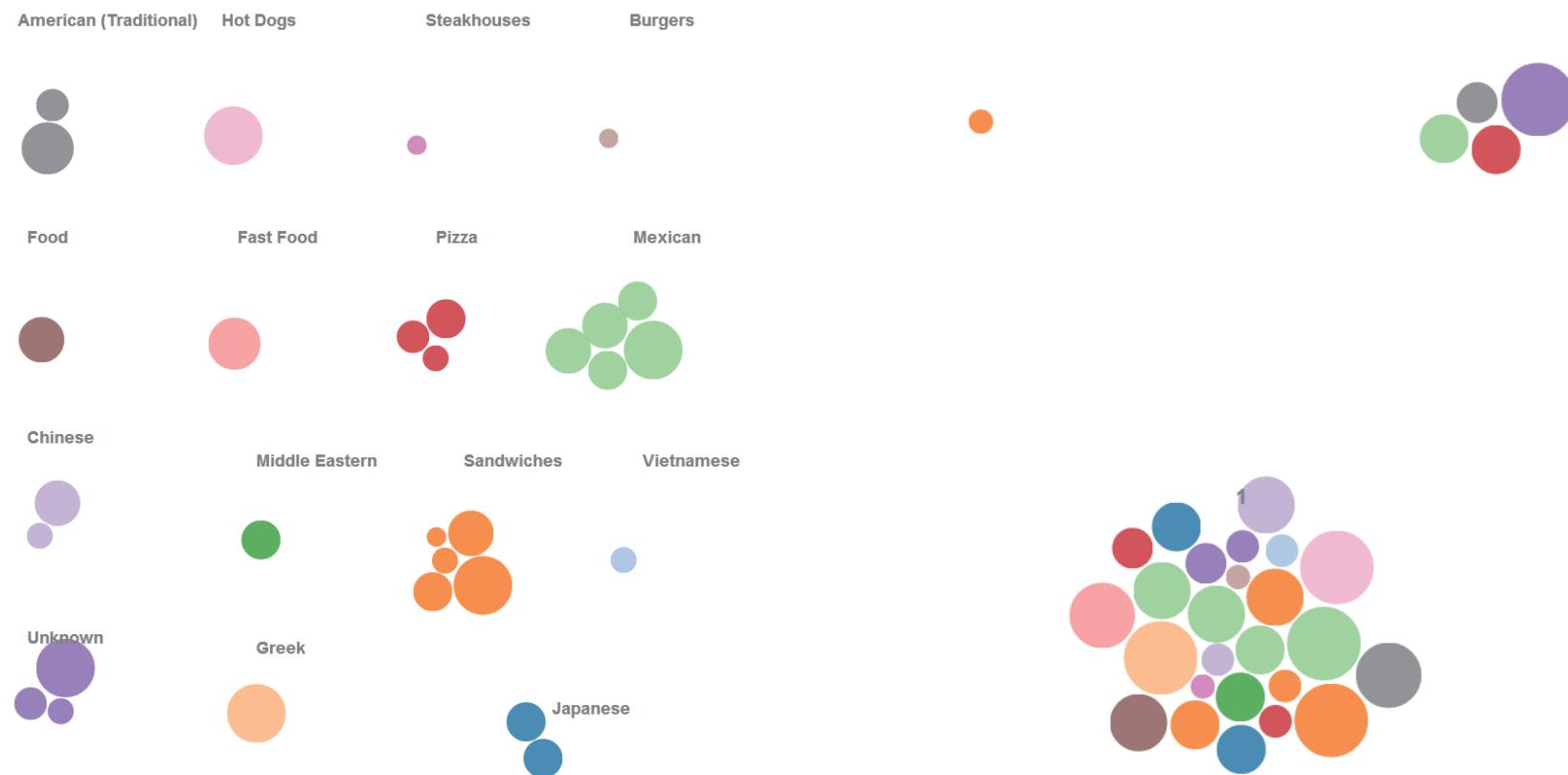
Step 5



Milestone 3

In milestone 3, we merge several steps into one layout.

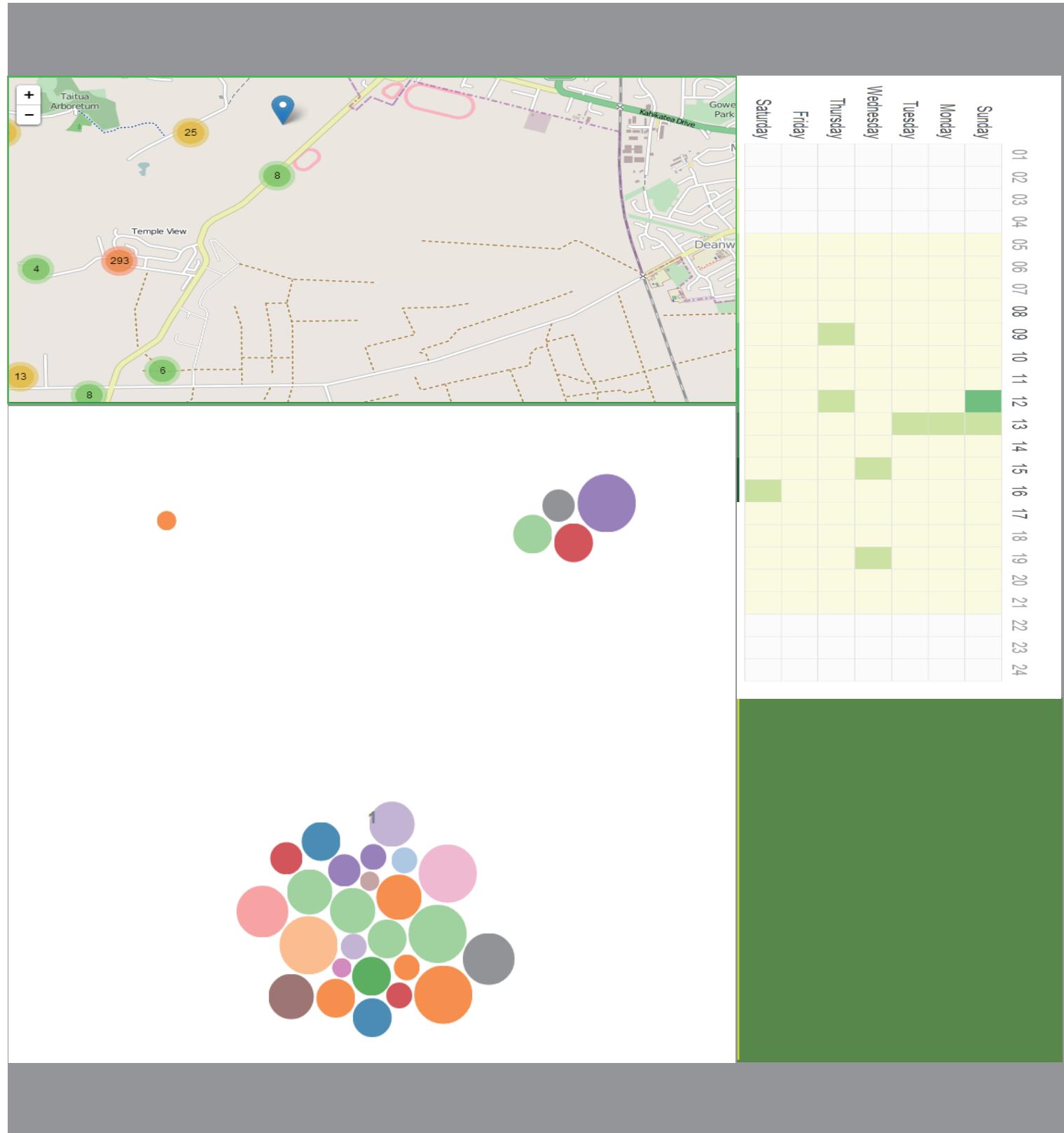
To choose several important options to display the data and clustered them by price, food type and popularity.



Milestone 3

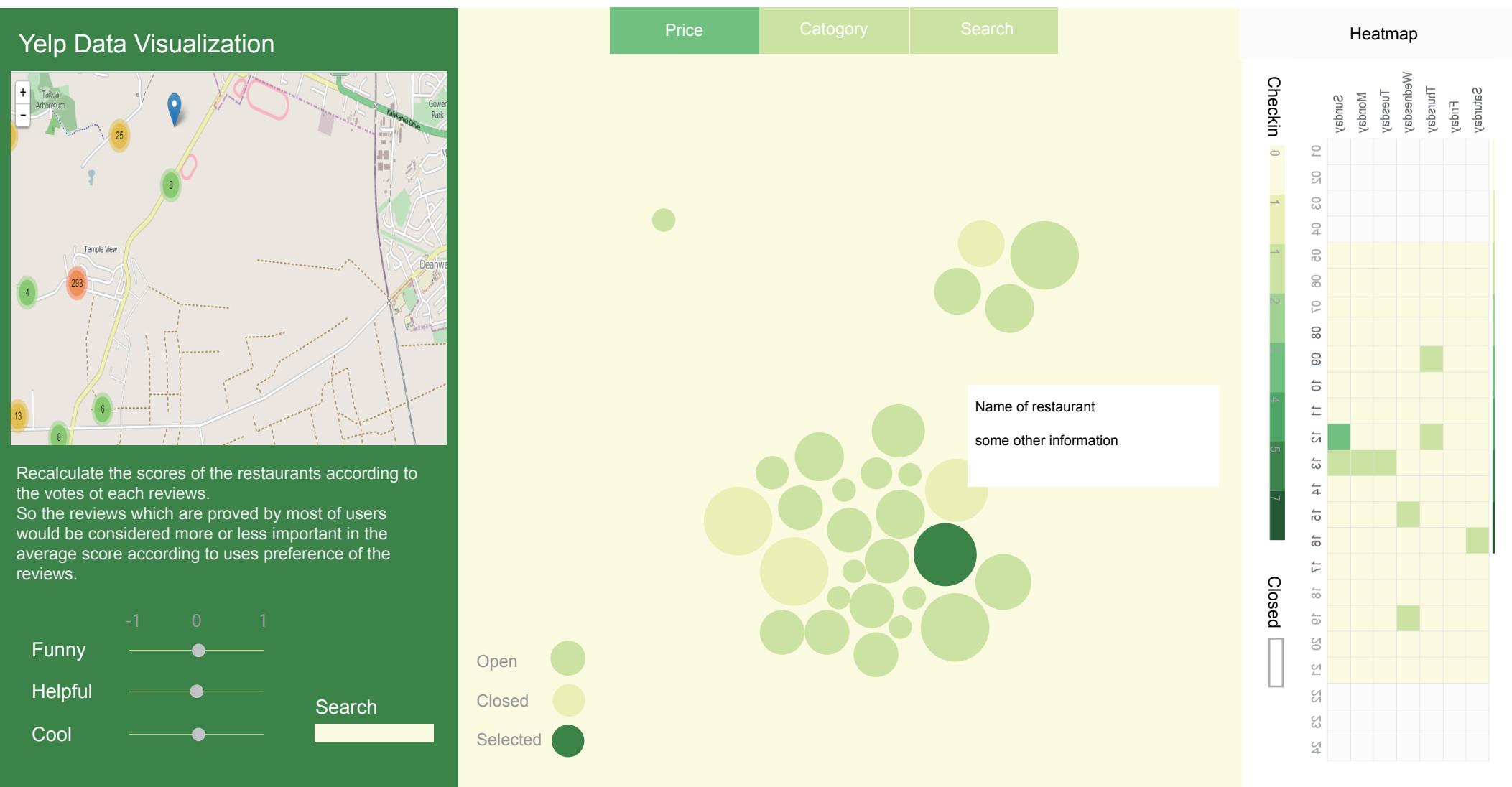
Layout trial

We decide to use one set of color which is color-blind friendly instead of two.
And tried several different layout.



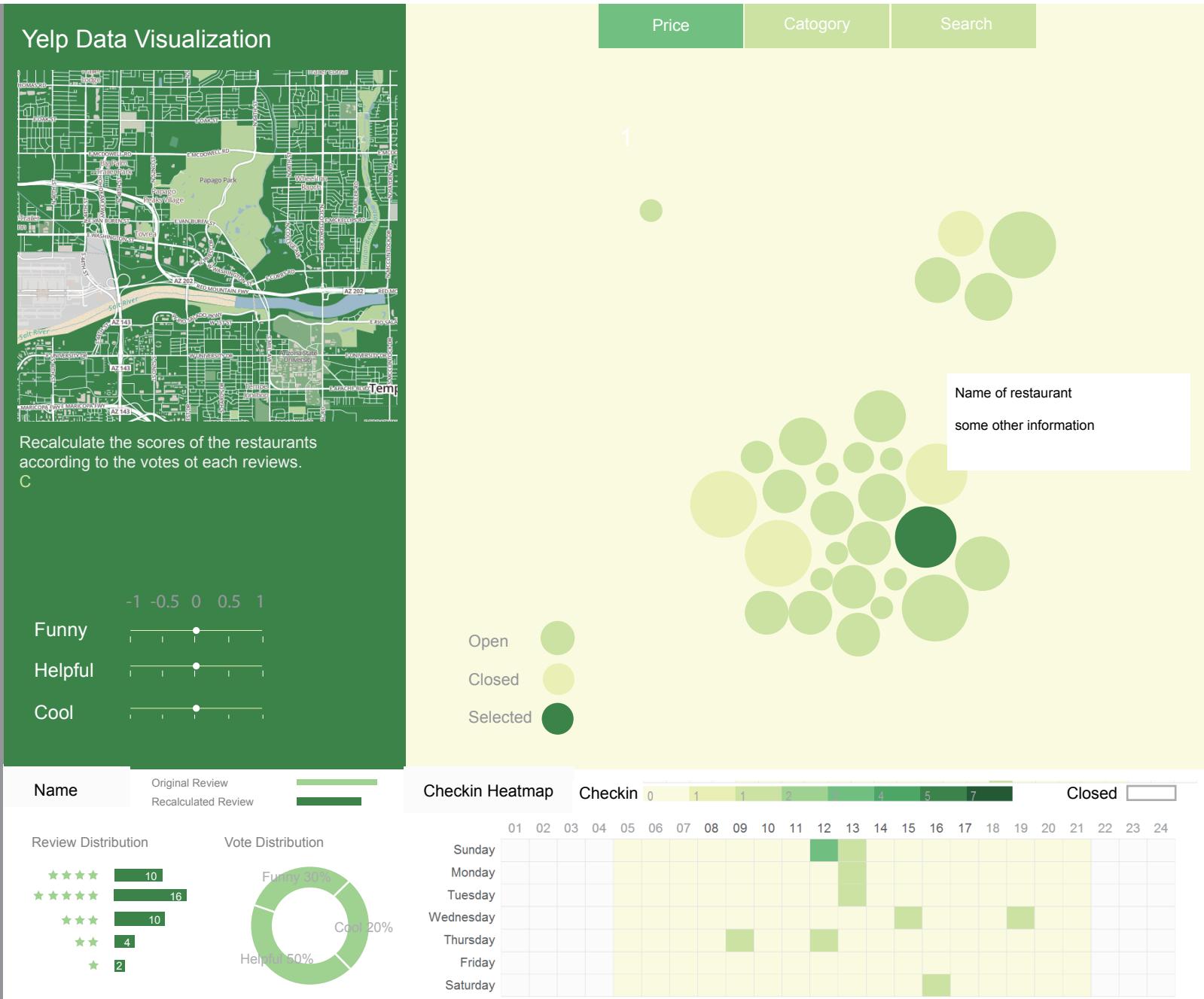
Milestone 3

Layout trial



Milestone 3

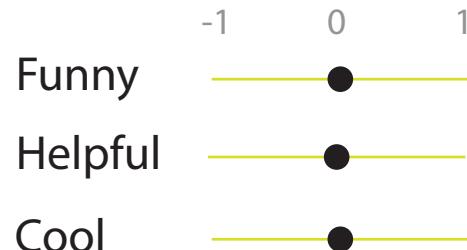
Layout trial



Milestone 3

We want to find a better way to present the reviews of the restaurants.

So we want to use votes to recalculate the score which avoids the trouble to go through all the detailed reviews.



Original review

```
reviewtotal= (review1+review2+review3+...+reviewN)
```

```
reviewN=N;
```

```
reviewaverage=reviewtotal/reviewN
```

Recalculate review

```
Num1funny= vote for funny
```

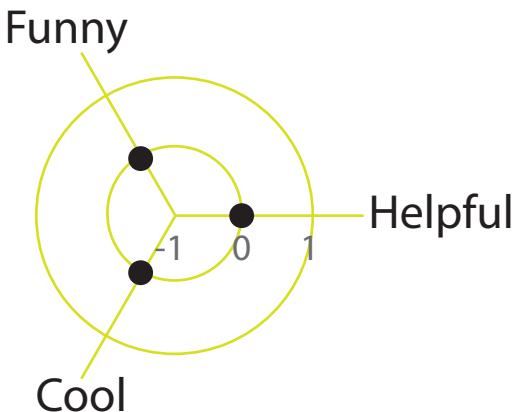
Wfunny is a slider range from -1 to 1 (0 is default)

```
review1recal= (1+Num1funny*Wfunny+Num1useful*Wuseful+Num1cool*Num1cool)
```

```
reviewN=review1recal+review2recal+...+reviewNrecal
```

```
reviewtotal=review1*review1recal+review2*review2recal+...+reviewN*reviewNrecal;
```

```
reviewaverage=reviewtotal/reviewN
```



Milestone 3

Algorithm update

Original review

```
reviewtotal= (review1+review2+review3+...+reviewN)
```

```
reviewN=N;
```

```
reviewaverage=reviewtotal/reviewN
```

Recalculate review

Num1funny= vote for funny

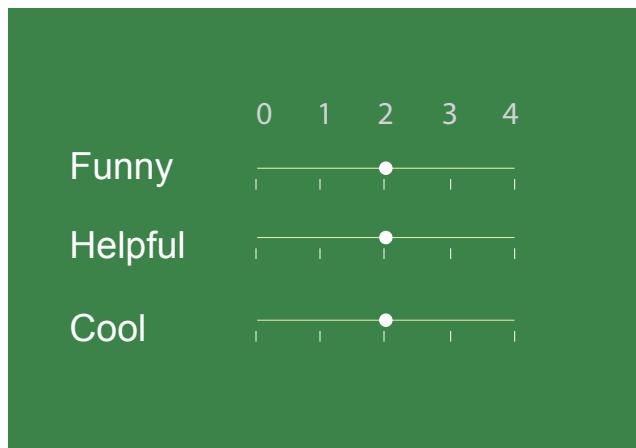
Wfunny is a slider range from 0 , 1, 2, 3, 4 (0 is default)

```
review1recal= (1+Num1funny*Wfunny+Num1useful*Wuseful+Num1cool*Num1cool)
```

```
reviewtotal=review1*review1recal+review2*review2recal+...+reviewN*reviewNrecal;
```

```
reviewN=review1recal+review2recal+...+reviewNrecal
```

```
reviewaverage=reviewtotal/reviewN
```



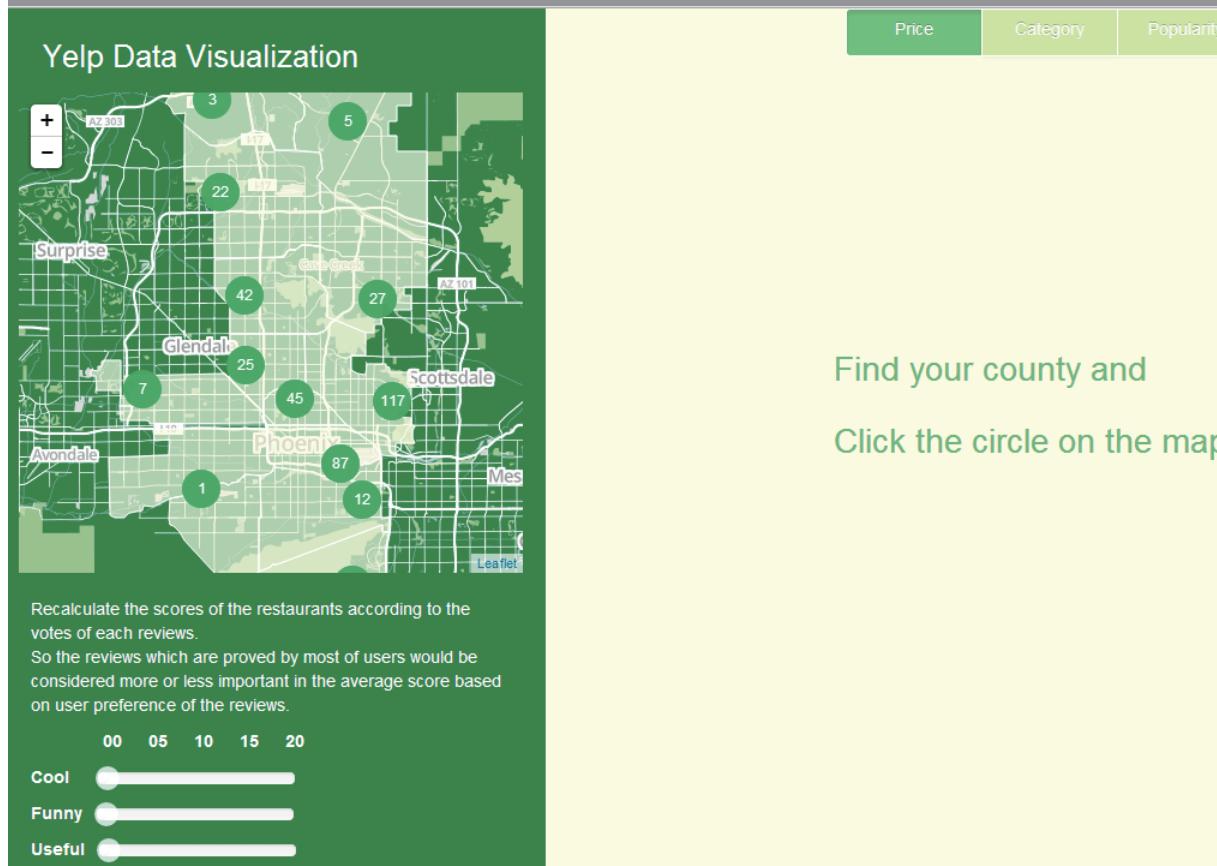
We update the algorithm based on the first trial which makes more sense for the data.

Final

Layout

The final result seems almost the same with what we designed.

This is the loading page.



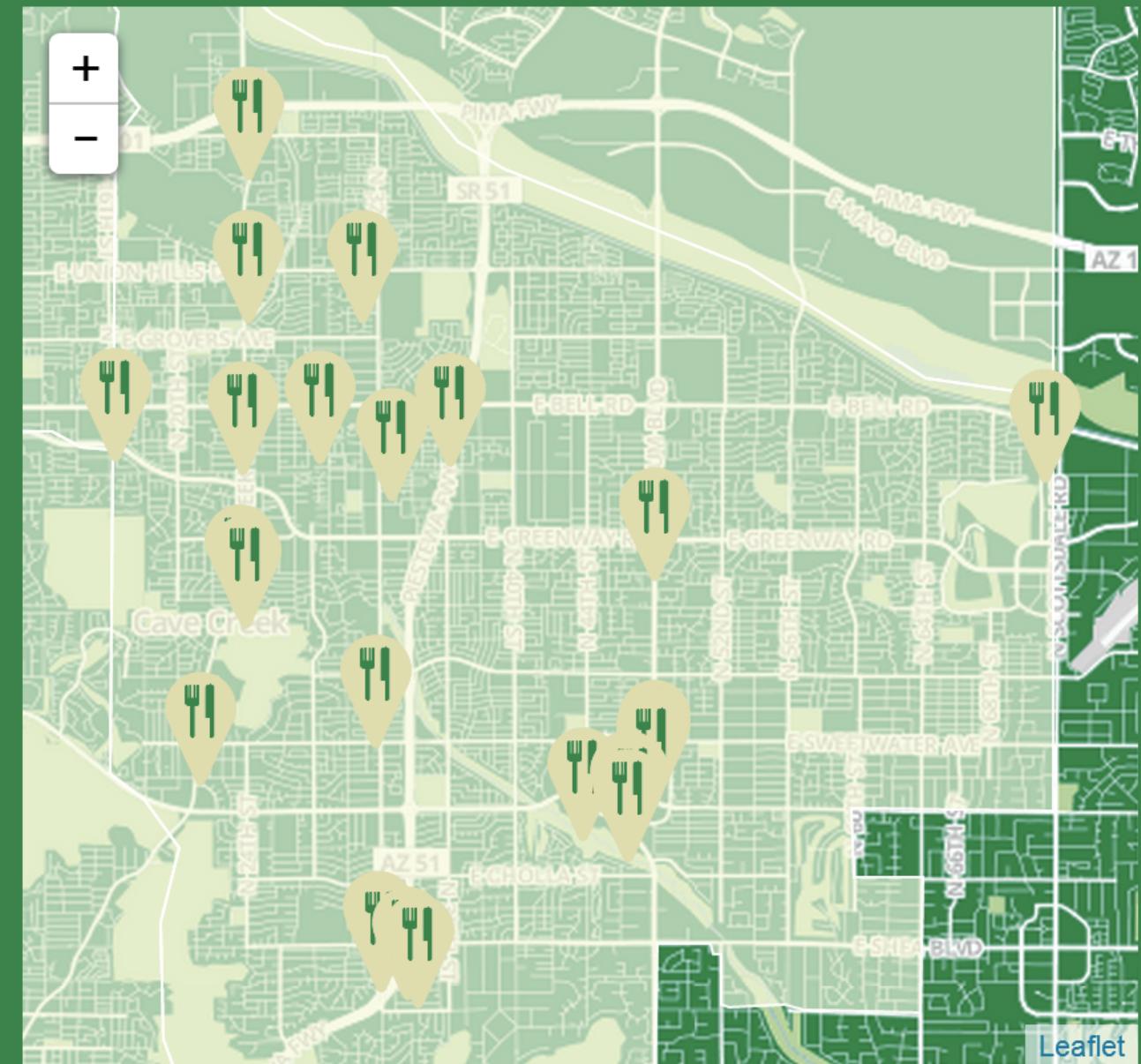
Final

Yelp Data Visualization

Map API

When click, it will go into the neighbourhood level which updates all the restaurants inside.

We design icons for our restaurants which includes most of events triggered by users.



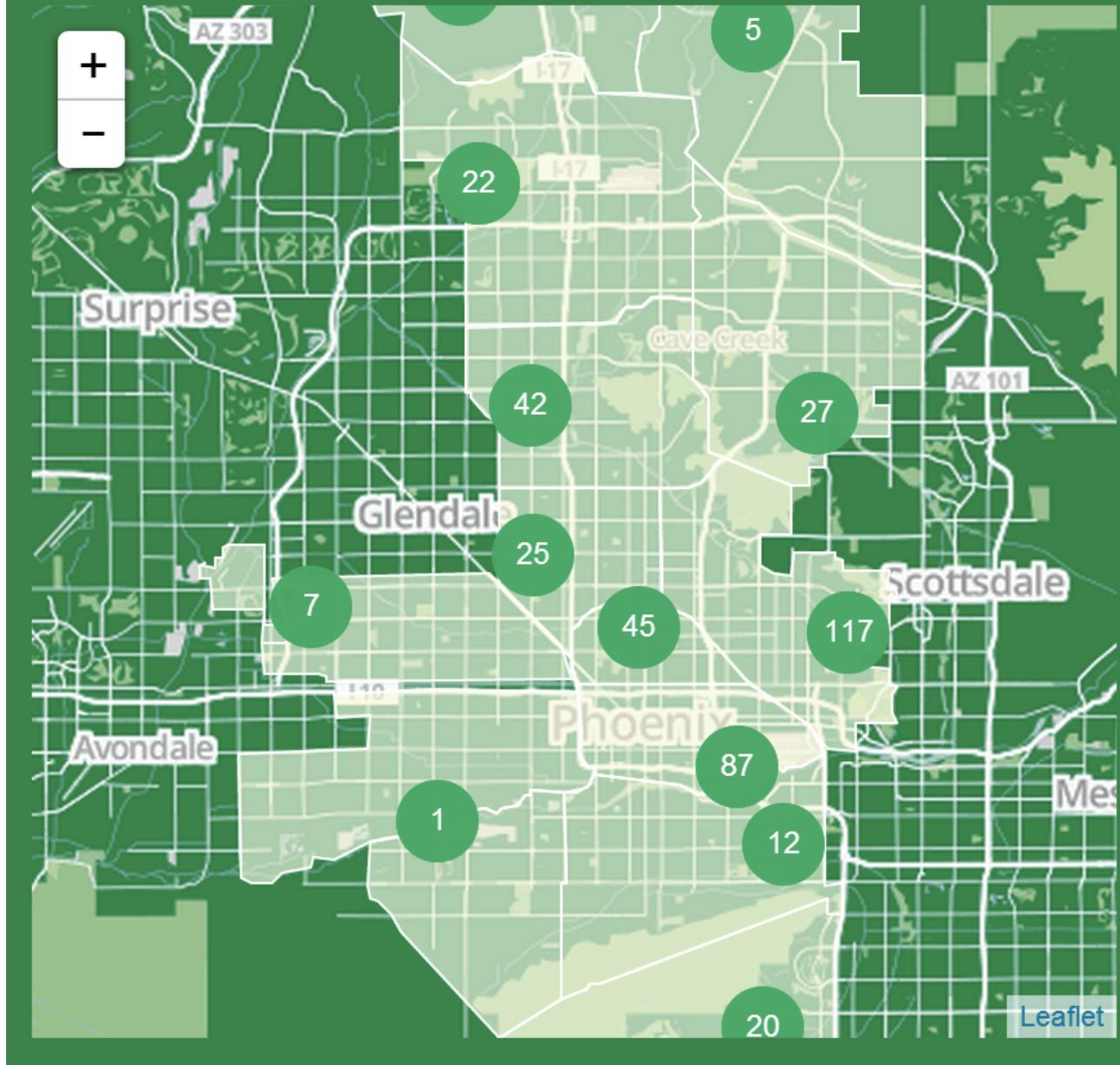
Final

Map API

We use leaflet.js and Mapbox to customize our map which enables to cluster the restaurants on map by neighbourhood.

And we add the geojson layer on the map to define the boundary area.

Yelp Data Visualization



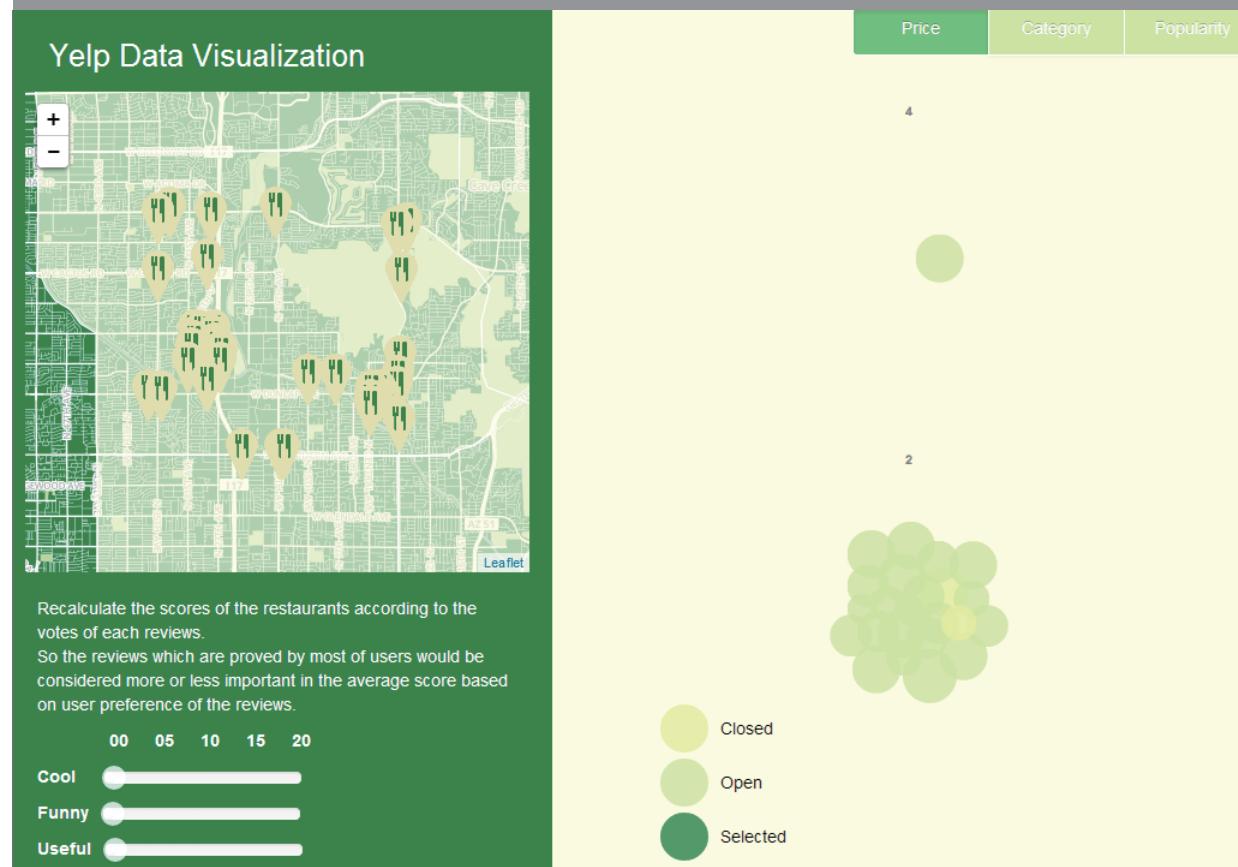
Final

Restaurant Cluster

The restaurants in the neighbourhood will update in the right window.

And they will clustered according to people's choice.

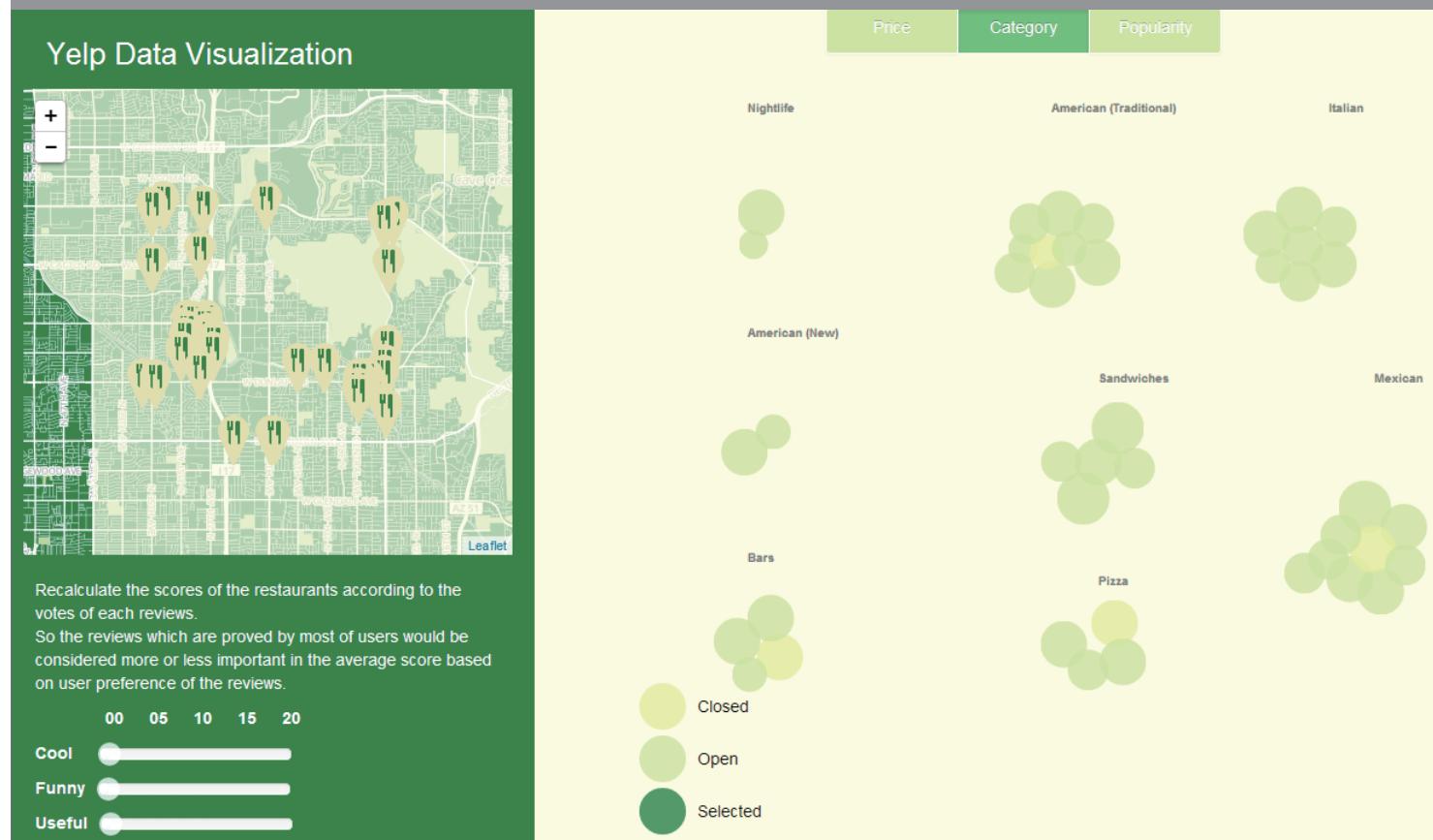
Price is always important.



Final

Restaurant Cluster

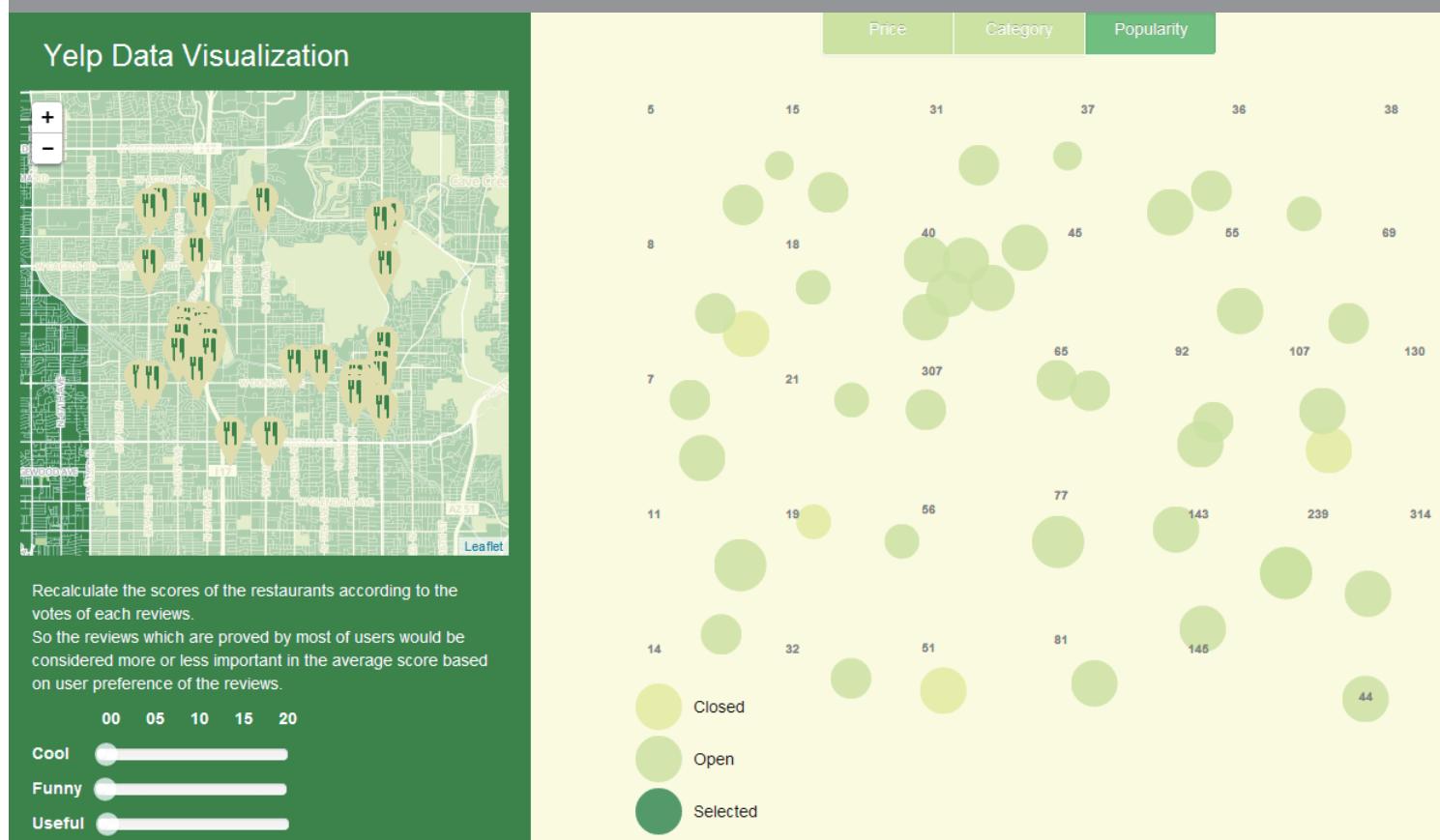
We can recategorize them by food type if users want to choose one or several types of food that we prefer.



Final

Layout trial

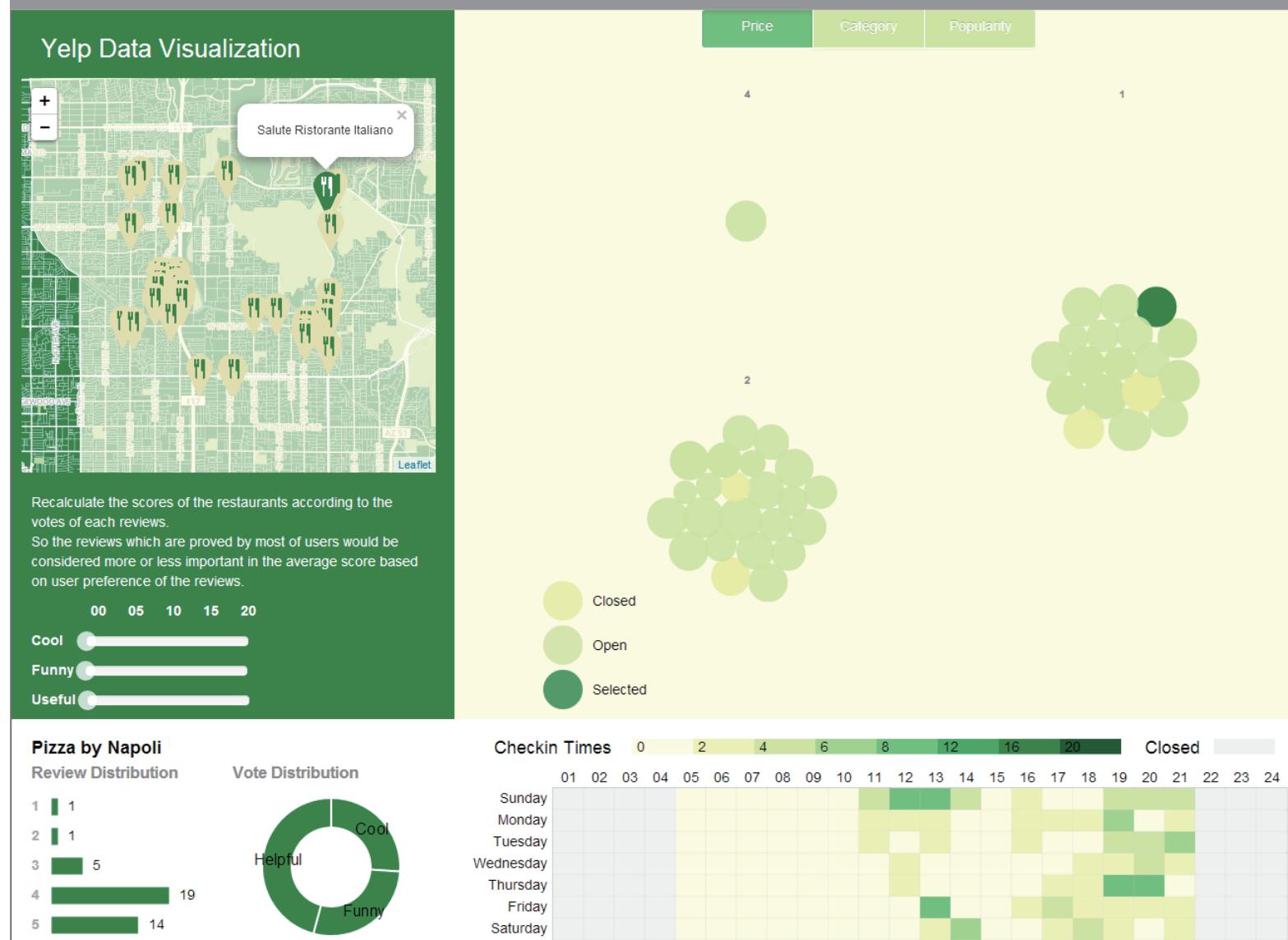
They can also be listed with review numbers, which indicate the popularity of the restaurant and the reliability of the scores.



Final

Layout trial

Most likely we will choose the restaurant with the highest reviews. And the radius of each circle is related to their reviews. But the average score sometimes is not reliable and people still will go through the detailed reviews.



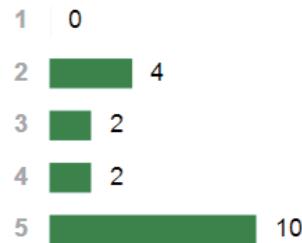
Final

Layout trial

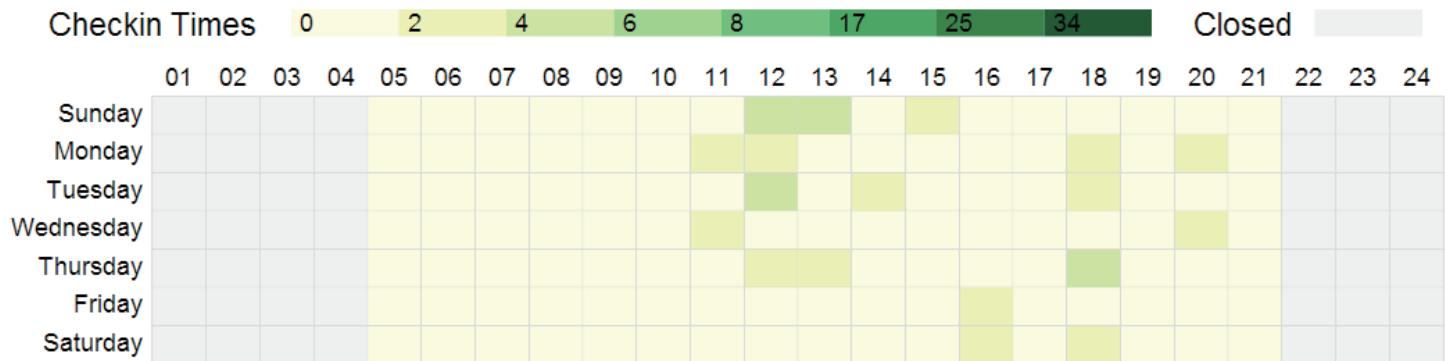
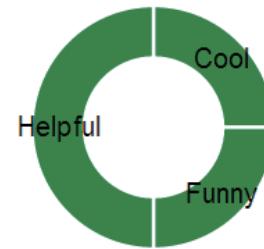
When we select one restaurant, it will update the detailed information of the restaurant and the check-in heatmap which indicates the busy time of it. So people will know whether it is the best time to go because it is not busy now.

Jersey Mike's

Review Distribution



Vote Distribution



Final

Layout trial

So we embed the most important function in our visualization that people could recalculate the score of the restaurant according to the votes of the reviews. They can put weight on different votes which affects the calculation. For example, if people put 10 weights on the useful reviews, so the review which has 10 votes may be calculated twice.



Recalculate the scores of the restaurants according to the votes of each reviews.

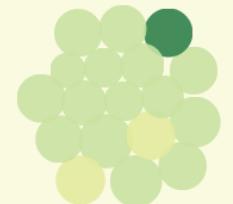
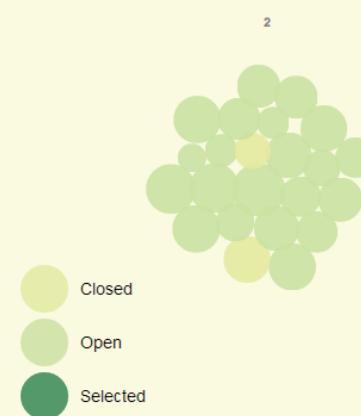
So the reviews which are proved by most of users would be considered more or less important in the average score based on user preference of the reviews.

00 05 10 15 20

Cool

Funny

Useful



Recalculate the scores of the restaurants according to the votes of each reviews.

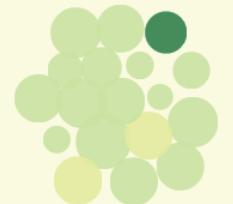
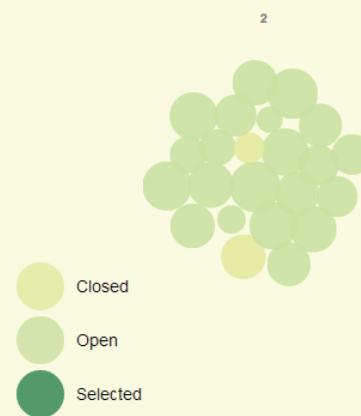
So the reviews which are proved by most of users would be considered more or less important in the average score based on user preference of the reviews.

00 05 10 15 20

Cool

Funny

Useful



Conclusion



We are still developing our project and will continue working on it even after the class.

We want to use the project to take part in the Yelp Challenge Dataset Competition this year. The project could be improved a lot but we think we develop our idea quite clearly in the project so far. And we want to use this visualization as a start point and analysis tool to find better algorithms for the recalculation according the result.

We really appreciate that CS171 gives us the opportunity to meet and work with people from different background such as our team. The combination of designer and computer scientist really makes the project forward. Thank you!