Foodmate

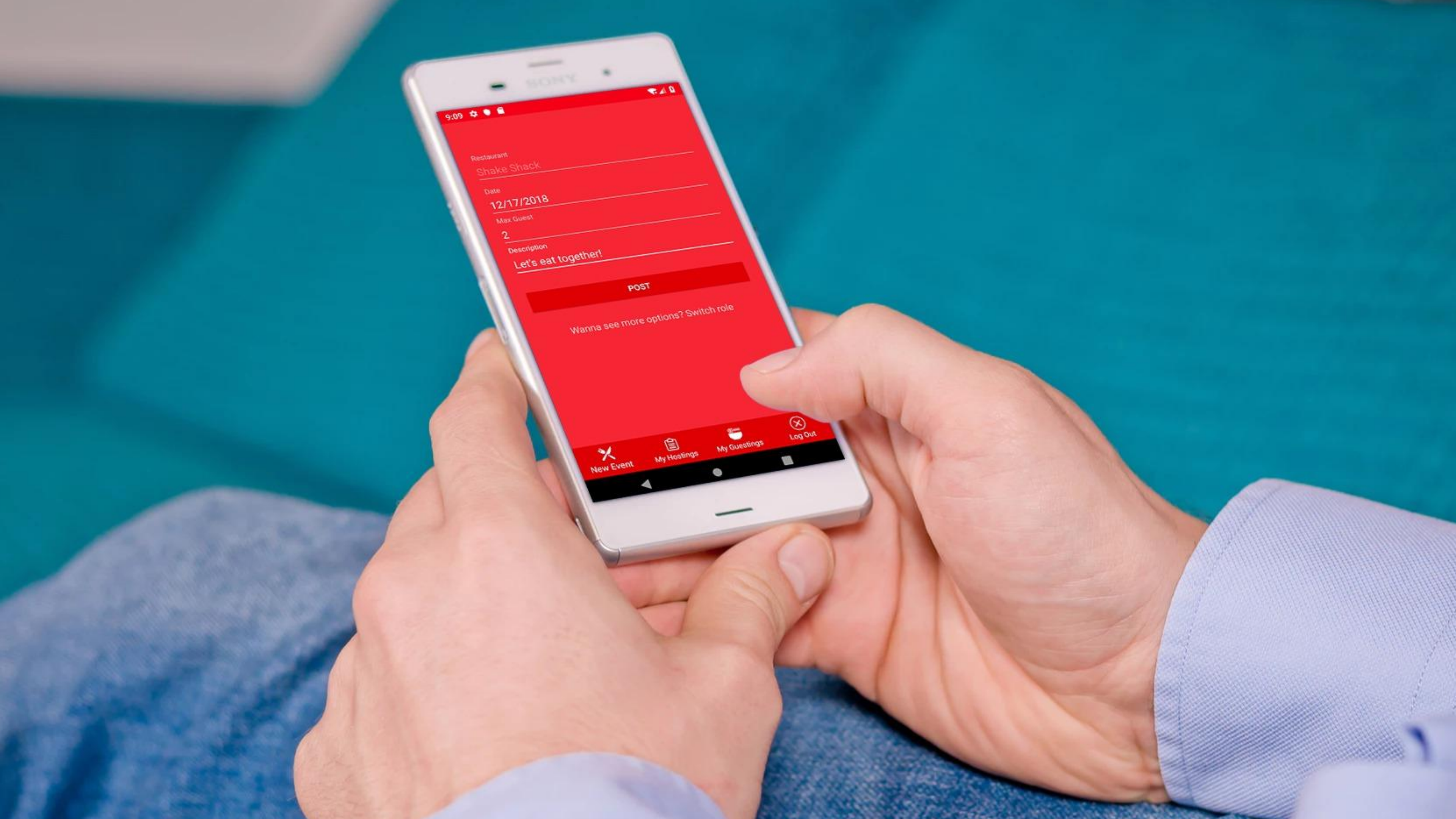
Group 18

Yujia Liu, Xiangge Meng, Yi Zhang,
Tianshi Liao, Zhongyi Sheng

Overview

- Motivation
- UML & Development tools
- Recommender System
- Live Demo

FoodMate



Restaurant
Shake Shack

Date
12/17/2018

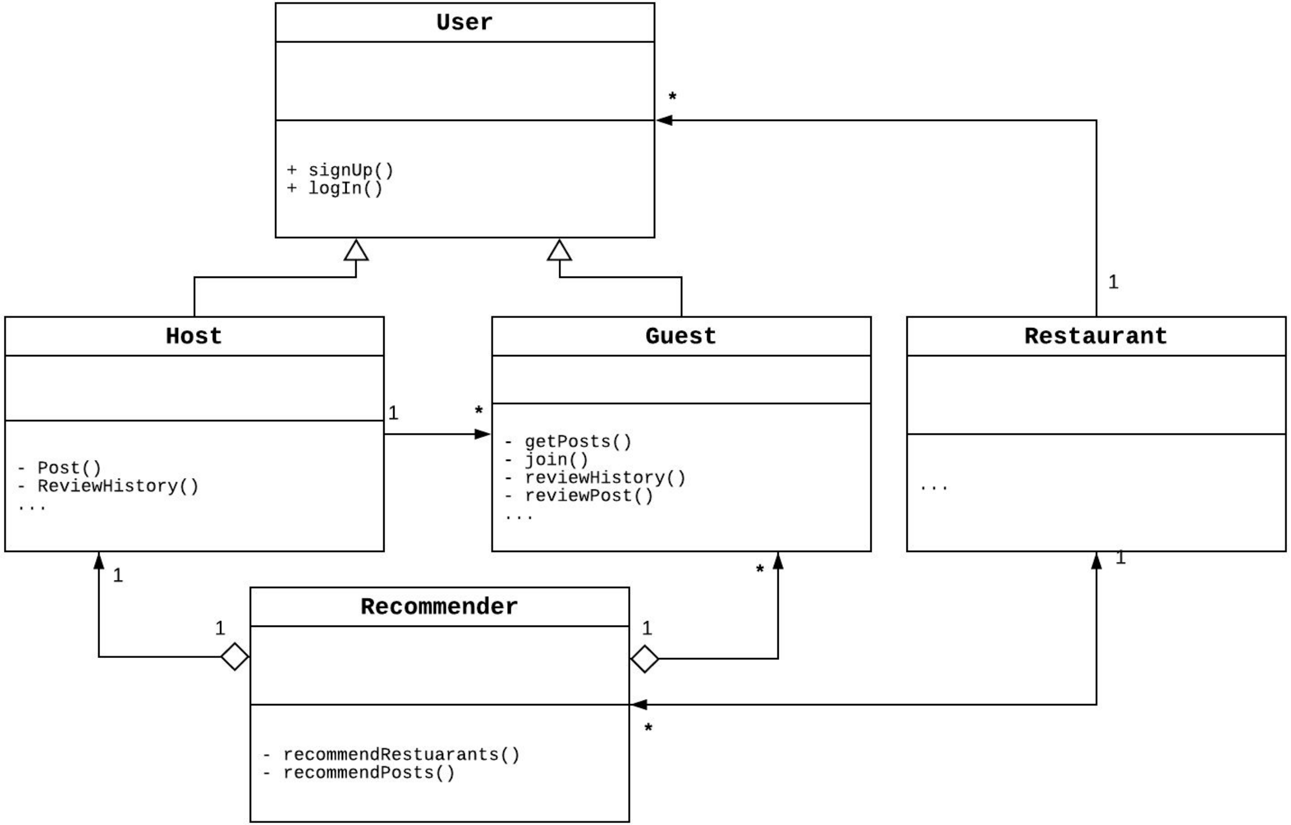
Max Guest
2

Description
Let's eat together!

Shake Shack
Xiangge
New
My Hostings
My Guestings
Log Out
SAMSUNG

Components

- Users: Host / Guest
- Host: post invitation
- Guest: browse, accept invitation
- Recommender system



User

+ signUp()
+ logIn()

Host

- Post()
- ReviewHistory()
...

Guest

- getPosts()
- join()
- reviewHistory()
- reviewPost()
...

Restaurant

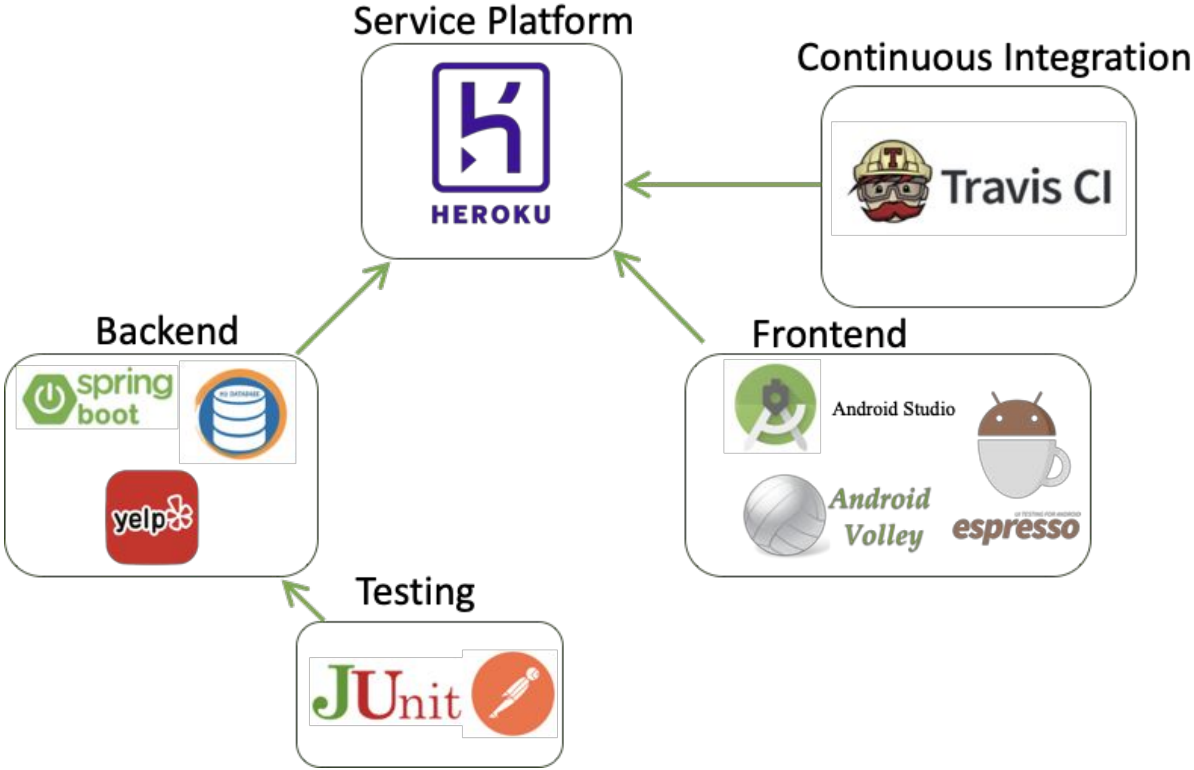
...

Recommender

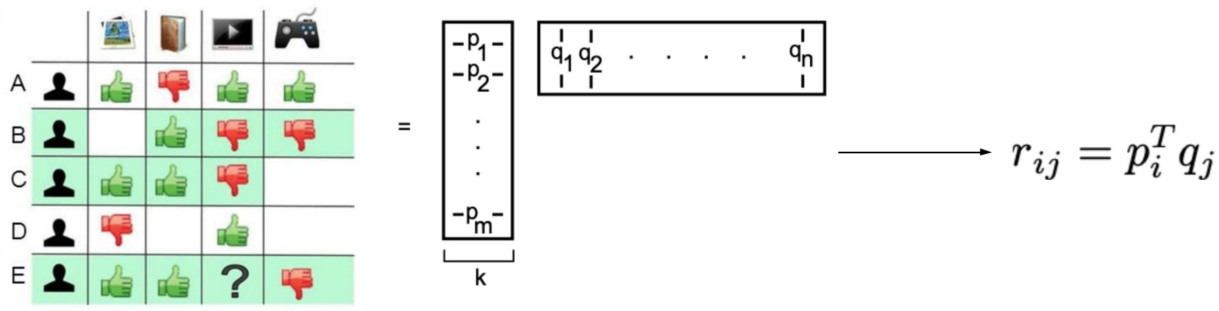
- recommendRestuarants()
- recommendPosts()

Development Tools

- Spring Boot, Maven
- H2 Database
- Yelp API
- JUnit
- PostMan
- Heroku
- Travis-CI
- Android Studio



Recommender System: Algorithm



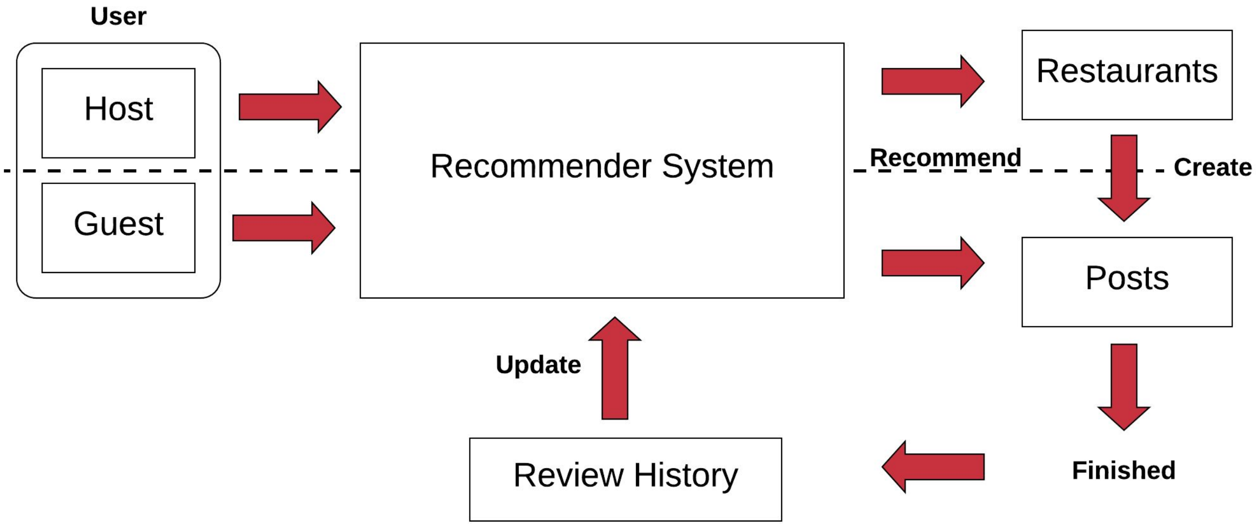
A	👍	👎	👍	👍
B		👍	👎	👎
C	👍	👍	👎	
D	👎		👍	
E	👍	👍	?	👎

Rating Matrix **Decomposition** **Inference**

Our implementation: Matrix Factorization-based Collaborative Filtering

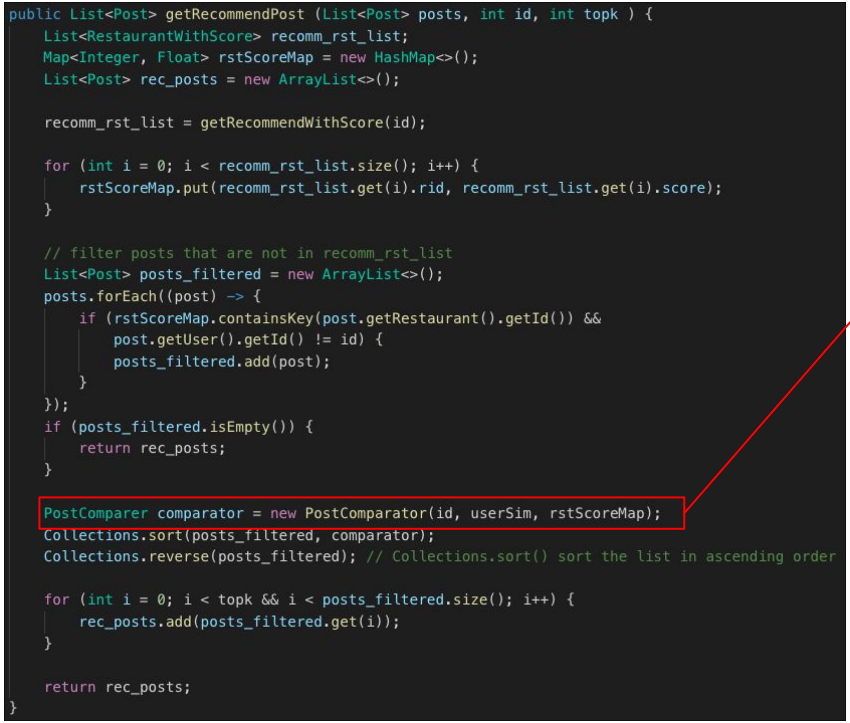
- Collaborative filtering framework
- Basic idea - predict user preference by collecting preference information from other users
- Advantages
  - No domain knowledge required
  - Relying directly on user behavior
  - Allows uncovering complex and unexpected patterns

Recommender System: Pipeline



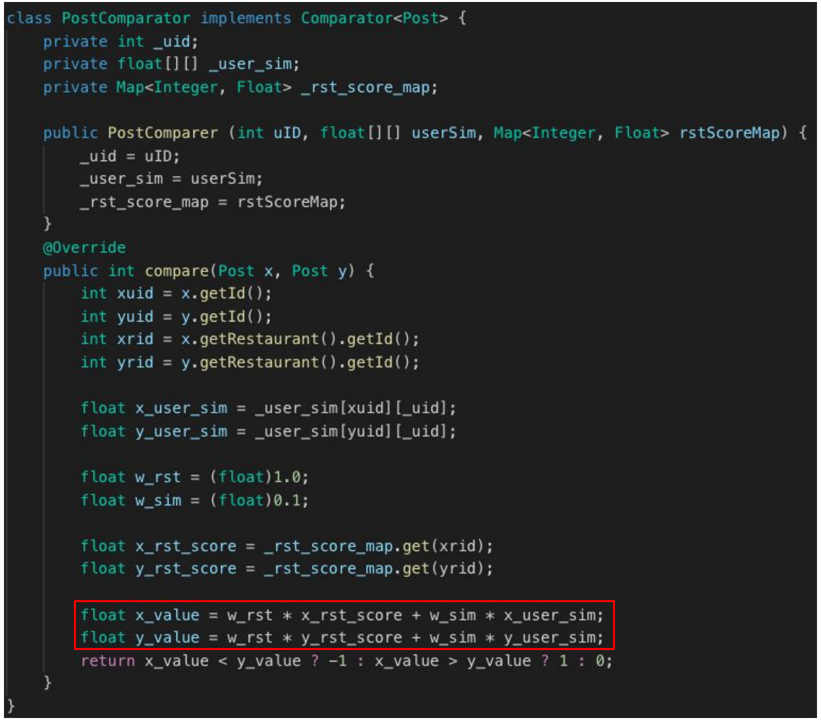
- Recommendation will become more accurate as a user keeps making more reviews.

Recommender System: Code Snippets



```
public List<Post> getRecommendPost (List<Post> posts, int id, int topk ) {
    List<RestaurantWithScore> recomm_rst_list;
    Map<Integer, Float> rstScoreMap = new HashMap<>();
    List<Post> rec_posts = new ArrayList<>();

    recomm_rst_list = getRecommendWithScore(id);

    for (int i = 0; i < recomm_rst_list.size(); i++) {
        rstScoreMap.put(recomm_rst_list.get(i).rid, recomm_rst_list.get(i).score);
    }

    // filter posts that are not in recomm_rst_list
    List<Post> posts_filtered = new ArrayList<>();
    posts.forEach((post) -> {
        if (rstScoreMap.containsKey(post.getRestaurant().getId()) &&
            post.getUser().getId() != id) {
            posts_filtered.add(post);
        }
    });
    if (posts_filtered.isEmpty()) {
        return rec_posts;
    }

    PostComparer comparator = new PostComparator(id, userSim, rstScoreMap);
    Collections.sort(posts_filtered, comparator);
    Collections.reverse(posts_filtered); // Collections.sort() sort the list in ascending order

    for (int i = 0; i < topk && i < posts_filtered.size(); i++) {
        rec_posts.add(posts_filtered.get(i));
    }

    return rec_posts;
}
```

Method: getRecommendPost()

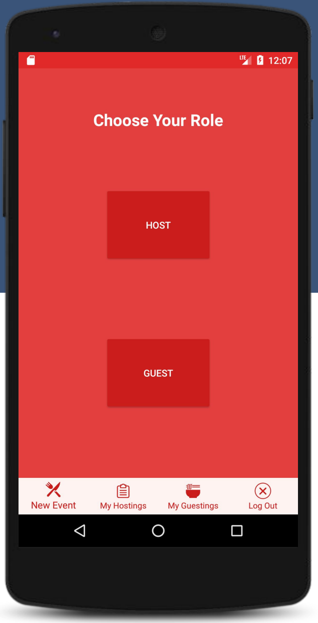


```
class PostComparator implements Comparator<Post> {
    private int _uid;
    private float[][] _user_sim;
    private Map<Integer, Float> _rst_score_map;

    public PostComparer (int uID, float[][] userSim, Map<Integer, Float> rstScoreMap) {
        _uid = uID;
        _user_sim = userSim;
        _rst_score_map = rstScoreMap;
    }
    @Override
    public int compare(Post x, Post y) {
        int xuid = x.getId();
        int yuid = y.getId();
        int xrid = x.getRestaurant().getId();
        int yrid = y.getRestaurant().getId();

        float x_user_sim = _user_sim[xuid][_uid];
        float y_user_sim = _user_sim[yuid][_uid];

        float w_rst = (float)1.0;
        float w_sim = (float)0.1;

        float x_rst_score = _rst_score_map.get(xrid);
        float y_rst_score = _rst_score_map.get(yrid);

        float x_value = w_rst * x_rst_score + w_sim * x_user_sim;
        float y_value = w_rst * y_rst_score + w_sim * y_user_sim;
        return x_value < y_value ? -1 : x_value > y_value ? 1 : 0;
    }
}
```

class: PostComparer()

Foodmate App Overview

HOST


12:07

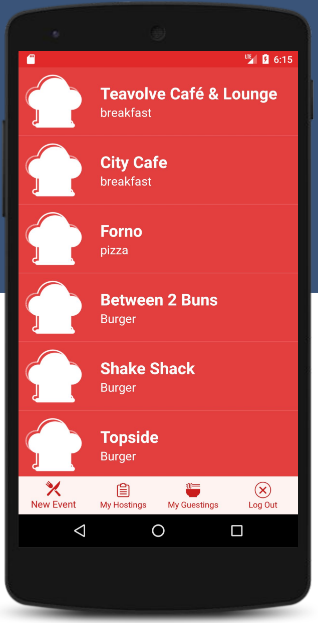
Choose Your Role

HOST

GUEST

New Event My Hostings My Guestings Log Out

Be a HOST!


6:15

Teavolve Café & Lounge
breakfast

City Cafe
breakfast

Forno
pizza

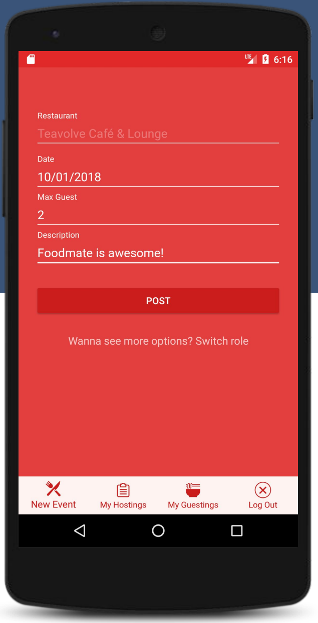
Between 2 Buns
Burger

Shake Shack
Burger

Topside
Burger

New Event My Hostings My Guestings Log Out

Restaurant recommended by Foodmate


6:16

Restaurant
Teavolve Café & Lounge

Date
10/01/2018

Max Guest
2

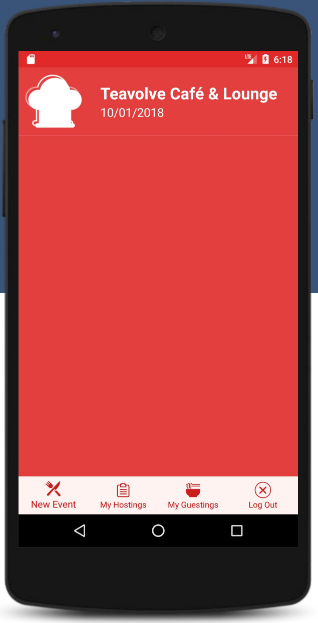
Description
Foodmate is awesome!

POST

Wanna see more options? Switch role

New Event My Hostings My Guestings Log Out

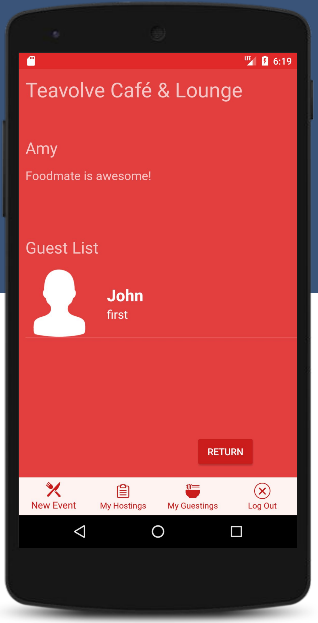
Start to invite someone!


6:18

Teavolve Café & Lounge
10/01/2018

New Event My Hostings My Guestings Log Out

Check on posts created


6:19

Teavolve Café & Lounge

Amy

Foodmate is awesome!

Guest List

John
first

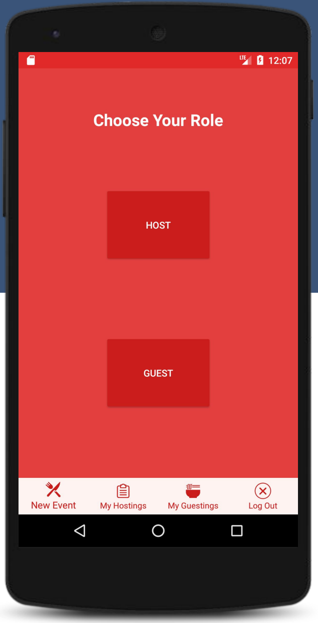
RETURN

New Event My Hostings My Guestings Log Out

Detailed information of your post

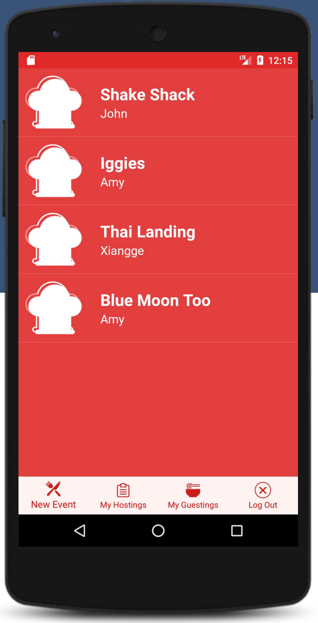
Foodmate App Overview

GUEST



Choose Your Role

Be a GUEST!



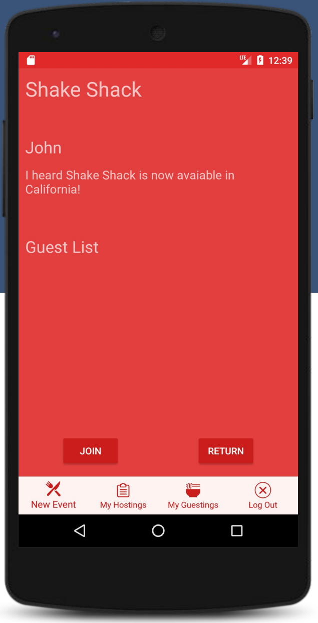
Shake Shack
John

Iggies
Amy

Thai Landing
Xiangge

Blue Moon Too
Amy

If no one has posted yet...



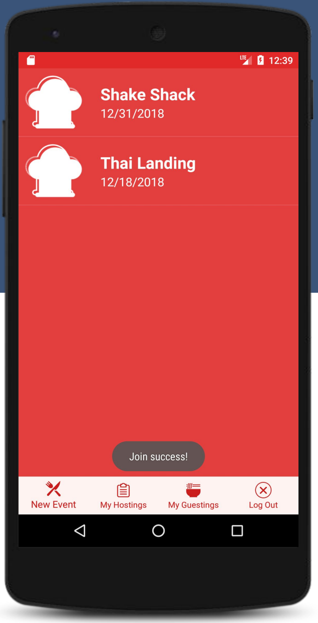
Shake Shack

John

I heard Shake Shack is now avaiable in California!

Guest List

New
posts!

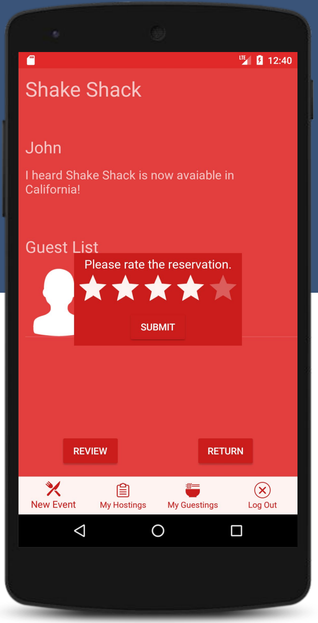


Shake Shack
12/31/2018

Thai Landing
12/18/2018

Join success!

Check joined
posts



Shake Shack

John

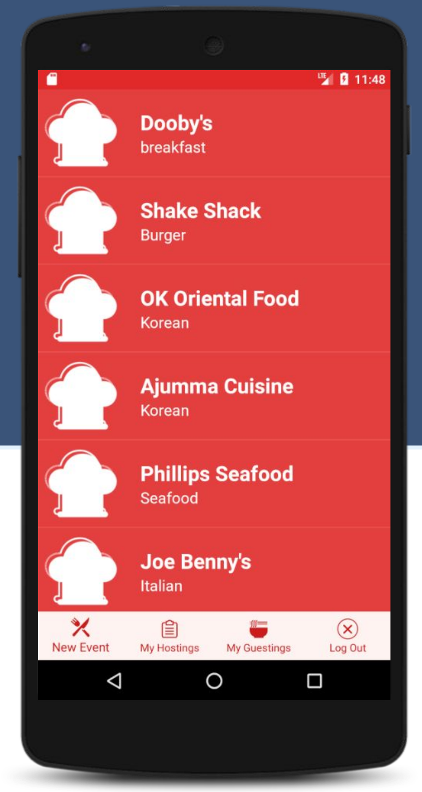
I heard Shake Shack is now avaiable in California!

Guest List

Please rate the reservation.

★★★★☆

Review
completed posts



Dooby's
breakfast

Shake Shack
Burger

OK Oriental Food
Korean

Ajumma Cuisine
Korean

Phillips Seafood
Seafood

Joe Benny's
Italian

Social App

Foodmate App

Foodmate is an app that connects people through meals. Do not want to eat alone? Start hosting a meal or joining invitations from someone else using *Foodmate*!

Restaurant
Find your favorite place

Food
Check out your favorite food

Friends
Make new friends

Interest
Mark your interests