Thang note:

* Favourite point: for where user check in , need Name, description, address, lat, lng, note, giá cả trung bình, hình ảnh đính kèm. Tao nghĩ cần thiết kế sao để khi nó cần share thông tin favourite đó cho user khác, hoặc cả cộng đồng, hoặc mình có thể tham khảo điểm này của nó để bổ sung vào place(location) trong db của mình. Trường hợp này dùng trong một số trường hợp mình đi về quê hay phượt ở đâu đó có thể làm review cái điểm đó, sau đó chia sẽ hoặc dùng lại trong lần đi sau. Vì thường thường mình đi phường thường nhờ dân địa phương dẫn đi, gặp nhìu quán ngon nhưng sau này ko còn nhớ quán đó ở đâu, đây là tính năng vô cùng hữu ích . (cái này tao thấy ko khác gì Object Location cả, thêm 1 thuộc tính Favourite cho user là đc),

Re: tách ra de phan biet may, sau nay favourite co them cac thu khac nua , ko nhap chung vao dc dau.

* message: invitation, message text , cần 1 message type để sau này có thể mở rộng (cái này từ từ đã cu)
* Location: đây là nơi chứa thông tin về các địa điểm đã được hệ thống mình xác thực là có thật và sẽ dùng gợi ý cho user, hiện sẽ có location về ở, ăn, chơi. Một câu hỏi đặt ra là có cần phải thiết kế mỗi loại là 1 bảng hay không, vì các địa điểm nghỉ ngơi nó các thuộc tính khác, các địa điểm ăn nó có thuộc tính khác, tương tự điểm chơi cũng vậy, còn có 1 cái type về điểm tham quan là các chùa chiềng, vì ngày nay du lịch phật giáo đang phát triển. (Theo tao ko nên tách ra từng bảng riêng, mà phải tìm cách tổng quát hóa nó lên vì nếu tách bảng sẽ tốn chi phí, đồng thời sẽ phải update liên tục nếu có những thuộc tính mới. Có 1 vấn đề là 1 Plan sẽ bao gồm nhiều Location. Như thế user có 2 quyền, 1 là copy (tức là copy Location ấy về Plan của mình, lúc này thuộc tính public của Location này sẽ phụ thuộc hoàn toàn vào user mới); 2 là shared (lúc này thuộc tính public sẽ phụ thuộc vào user sở hữu plan được share)

Re: location tao nghĩ ko cần public hay private, location này là những điểm thuộc hệ thống mình đã được mình XÁC THỰC,còn những điểm user tự tạo đó nó thuộc về riêng user đó, nên mới cần favourite cho user, chỉ có plan mới cần public hay privated

Object:

def LocationType {

Hotel 0

Restaurant 1

Sight-seeing 2

… (định nghĩa thêm)

}

Location {

String name;

String description;

LocationType type;

Double longitude;

Double latitude;

List<String> imageUrls;

String planID;

Boolean isShared;

Boolean isPublic; // depends on plan

Date dateCreated;

Date dateModified;

List<Like> likes;

List<Comment> comments;

List<Rating> ratings;

}

Plan {

String name;

String description;

Date time;

List<Location> location;

Integer noMembers;

String vehicle;

Integer budget;

String contact;

Integer visited;

List<String> ownerId;

Boolean isPublic;

Date dateCreated;

Date dateModified;

List<Like> likes;

List<Comment> comments;

List<Rating> ratings;

String others;

}

Database: Plan

Key: String id;

Value: Object Plan;

All Plan

Key: all\_plan;

Value: List<String> id;

Database: Location

Key: String id;

Value: Object Location;

All Location

Key: all\_location

Value: List<String> id;

Object:

User {

String username;

String password;

Map <Role, Plan> roles;

Date age;

String email;

String nationality;

List<Location> favouriteLocation;

Map<String, Like> likes;

Map<String, Rating> ratings;

Map<String, Comment> comments;

}

Database: User

Key: String id;

Value: Object User;

All Plan

Key: all\_user;

Value: List<String> id;

Object:

Like {

String planLikeId;

String userLikeId;

Date dateLike;

}

Database: Like

Key: String id;

Value: Object Like;

Object:

Comment {

String planCommentId;

String userCommentId;

String content;

Date dateComment;

}

Database: Comment

Key: String id;

Value: Object Comment;

Object:

Rating {

String planRatingId;

String userRatingId;

Integer rating;

Date dateRating;

}

Database: Rating

Key: String id;

Value: Object Rating;