CPEN 321 - M3

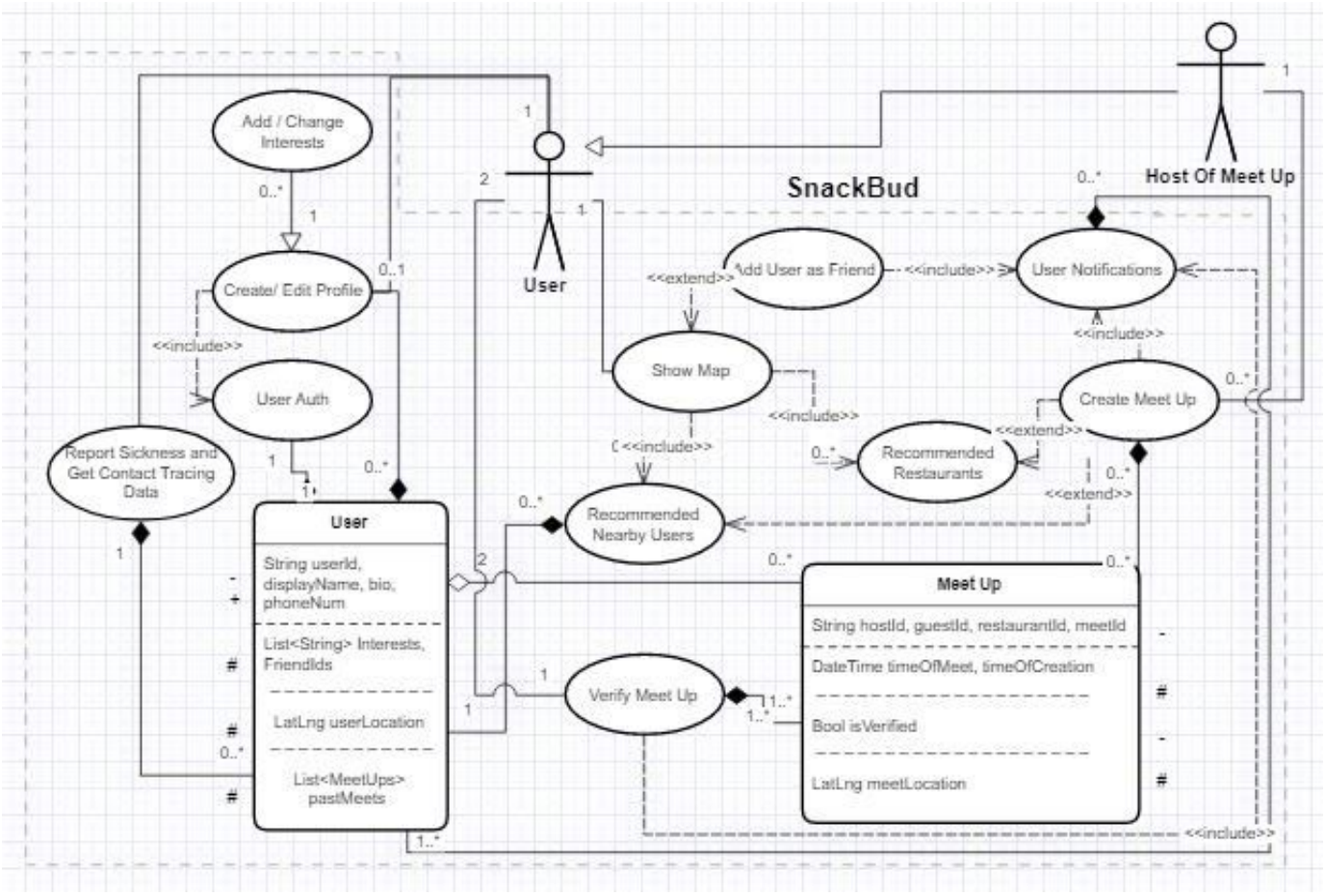
Parsa Riahi
72762974

SnackBud

Description:

With the current pandemic, people are no longer going out, and everything from their mental state to their university experience to the restaurant economy is tanking. SnackBud presents a new and spontaneous way to meet up with new and existing friends centered around a shared love between all people: food. The target audience initially will be students and youth in the age range of 18-24, which can expand to greater audiences as popularity grows. Restaurant owners are also a key market here, as they can promote themselves to new markets while providing some incentives to SnackBud and its users. To address the health concerns of COVID and to instantiate meetups, I will use the Google Maps API to allow for closeby users to safely interact with each other and schedule spontaneous meetups, which can then be inputted into their Google Calendars.

Use Case Diagram:



Non-Functional Requirements:

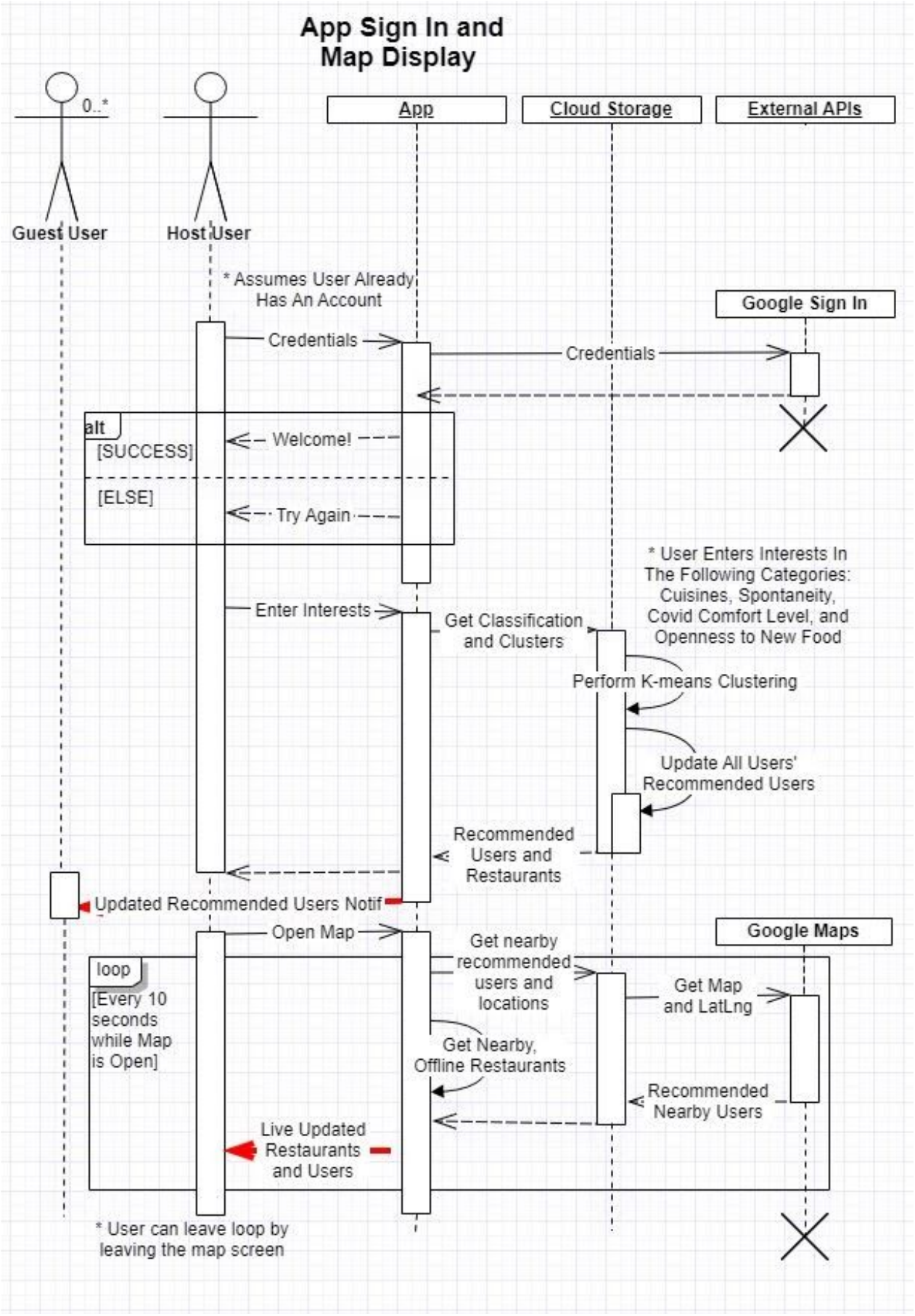
Create meet-up in 5 clicks	Quick use of the app adds to its reusability and dependability.
Update user location on the map every 10 seconds	Reliable location data is essential to creating spontaneous meets, adding to the performance and dependability of the app. The 10-second margin allows for better energy efficiency of the app due to fewer server pulls.
History of past meetups stored for 3 weeks	To add to the safety and security of the app, logs of data are to be kept for potential contact tracing and risk reporting.
Only show user info while on the app, but never their unique user ID	This adds to the security of personal data and privacy on the app.

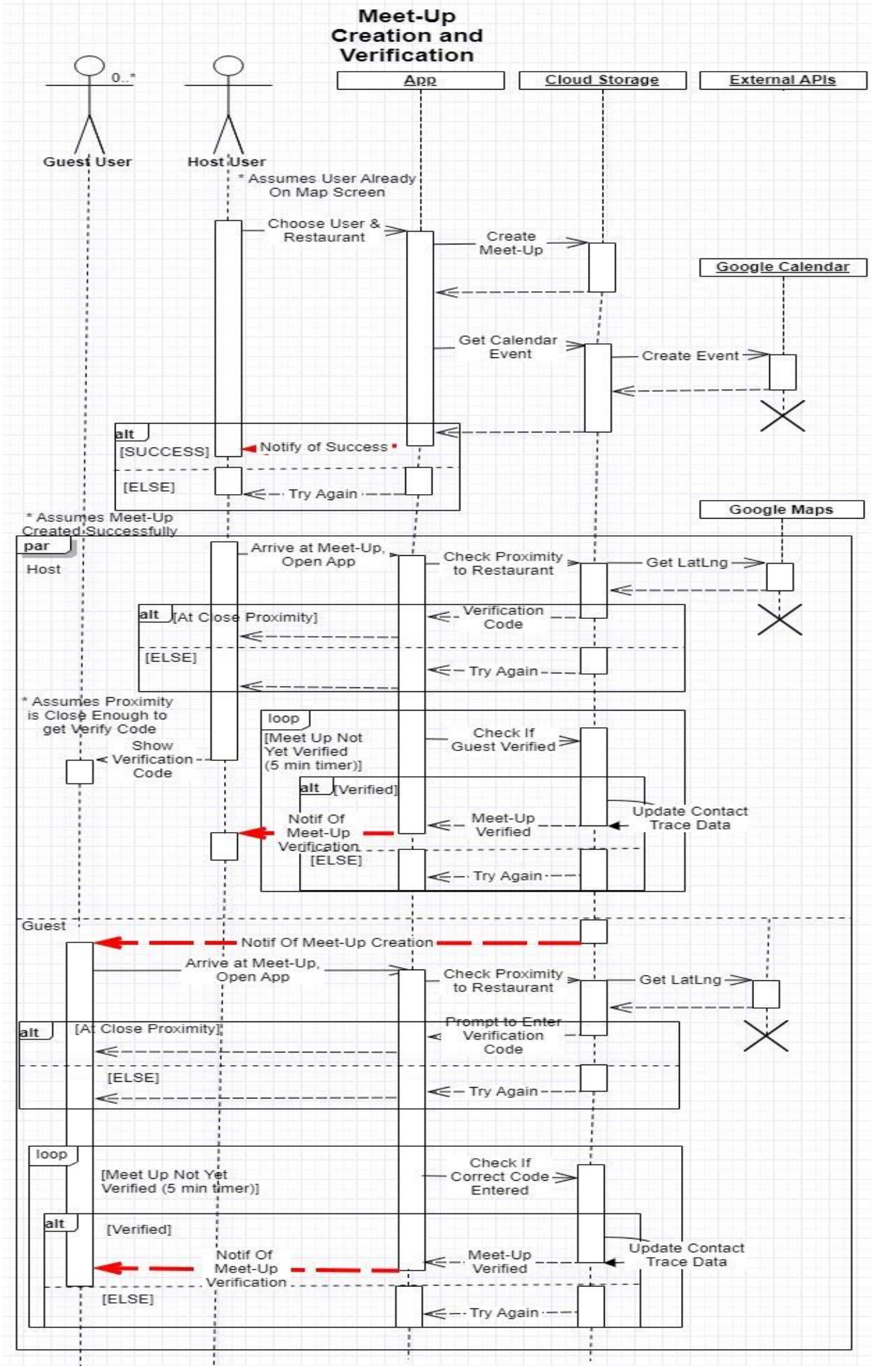
Only show location data of friends while within a 1km radius of them	This further adds to the security of personal data and privacy on the app.
List restaurants belonging to at least 20 cuisines	This adds to the scalability of the app in attracting personas who want to try new things to those who want to stick to their old favorites.

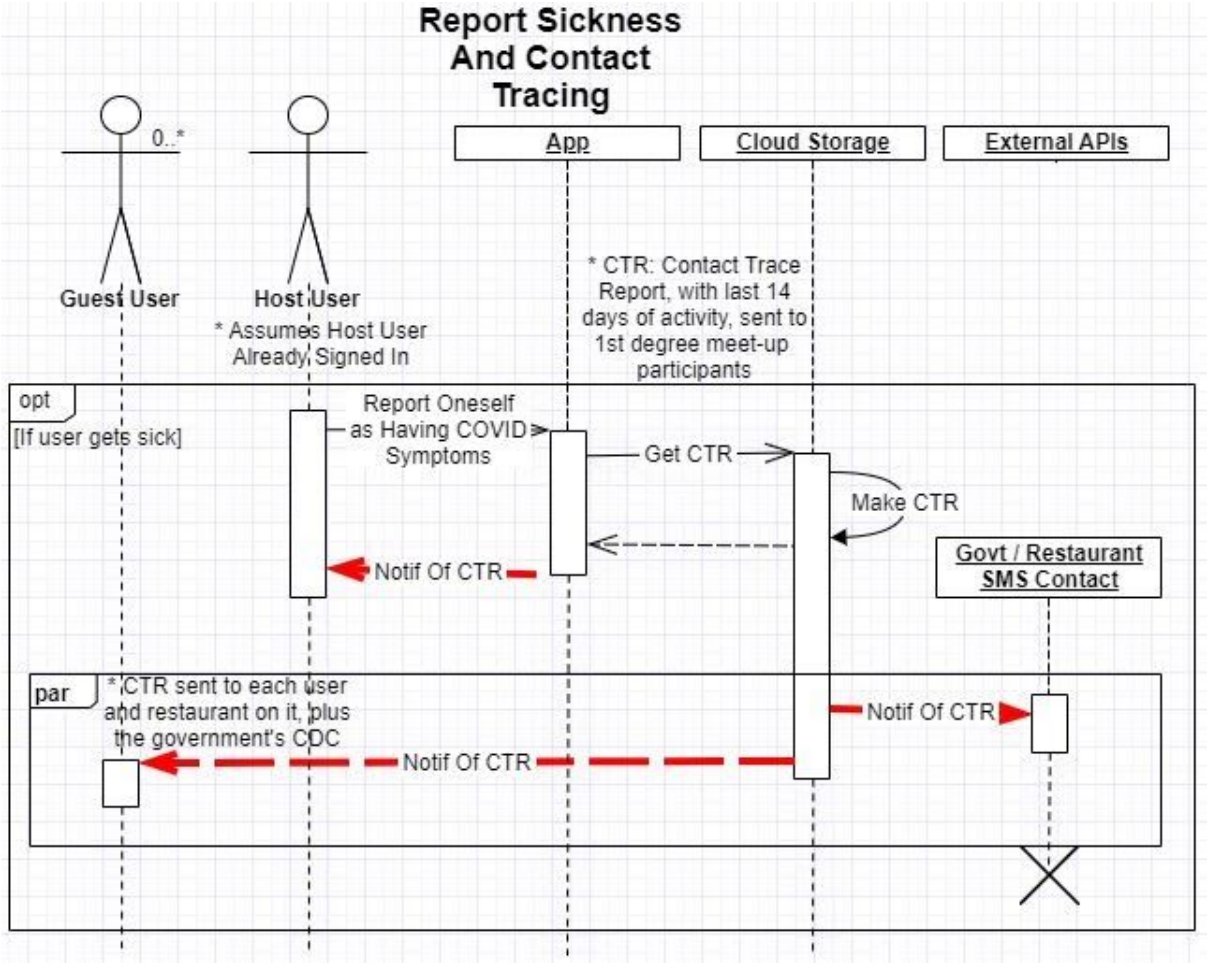
Main System Components:

This app has 5 main system components. The first is an advanced onboarding process involving Google Sign In and Clustering ML algorithms (ie. K-Means++) to categorize user interests in recommending new buddies and restaurants. This gets into the second use case, displaying recommended user and restaurants in a fixed radius around the user on an in-app map provided by the Google Maps API. The third use case is to create a Meet-Up with a user and restaurant from the map, which also involves sending Google Calendar events to all participants of the Meet-Up. The fourth case is the compliment of the third, Meet-Up Verification, whereby using Google Maps we determine the proximity of Meet-Up members to the restaurant, and if they are within a reasonable distance, they will be given the chance to enter a 3-digit verification code displayed on the host's phone into the app. The last component start when a Meet-Up is verified, as that Meet-Up is added to the user's Contact Trace Report (CTR), which records the past meets, users, and restaurants with which they interacted and attended, This report can be requested through the app and sent to all listed parties at 1 degree of separation as well as the CDC to provide a moderate level of contact tracing in order to prevent the further spread of COVID-19. These 5 components get combined into 3 main use cases: Startup - from onboarding to displaying the map with recommended entities, Meet-Up - creation and verification, and Blow-Up, where we attempt to limit the spread with CTRs.

Sequence Diagrams:




Meet-Up Creation and Verification
Guest User
0..*
Host User
App
Cloud Storage
External APIs
* Assumes User Already On Map Screen
Choose User & Restaurant
Create Meet-Up
Google Calendar
Get Calendar Event
Create Event
alt
[SUCCESS]
Notify of Success
[ELSE]
Try Again
Google Maps
* Assumes Meet-Up Created Successfully
par
Host
Arrive at Meet-Up, Open App
Check Proximity to Restaurant
Get LatLng
alt
[At Close Proximity]
Verification Code
[ELSE]
Try Again
* Assumes Proximity is Close Enough to get Verify Code
Show Verification Code
loop
[Meet Up Not Yet Verified (5 min timer)]
Check If Guest Verified
alt
[Verified]
Notif Of Meet-Up Verification
Meet-Up Verified
Update Contact Trace Data
[ELSE]
Try Again
Guest
Notif Of Meet-Up Creation
Arrive at Meet-Up, Open App
Check Proximity to Restaurant
Get LatLng
alt
[At Close Proximity]
Prompt to Enter Verification Code
[ELSE]
Try Again
loop
[Meet Up Not Yet Verified (5 min timer)]
Check If Correct Code Entered
alt
[Verified]
Notif Of Meet-Up Verification
Meet-Up Verified
Update Contact Trace Data
[ELSE]
Try Again


Report Sickness And Contact Tracing
0..*
Guest User
Host User
* Assumes Host User Already Signed In
App
Cloud Storage
External APIs
* CTR: Contact Trace Report, with last 14 days of activity, sent to 1st degree meet-up participants
opt
[If user gets sick]
Report Oneself as Having COVID Symptoms
Get CTR
Make CTR
Notif Of CTR
Govt / Restaurant SMS Contact
par
* CTR sent to each user and restaurant on it, plus the government's CDC
Notif Of CTR
Notif Of CTR

Sketch of Main App Screen: