# CICERONE

## Explore with a knowledgeable friend

***Midterm presentation***

by Team 4: Björn Bebensee, Eric Lindgren, Sigrid Marita Kvamme, Jungwook Kim

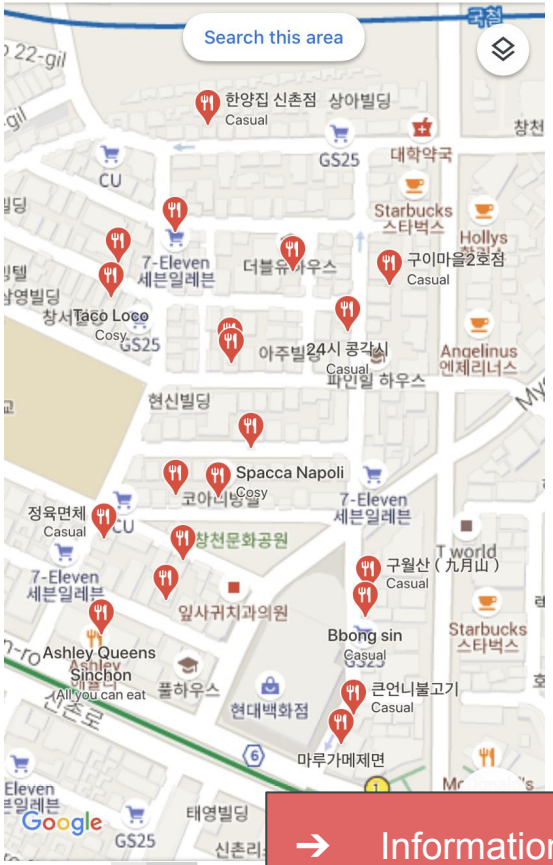
# Motivation & Problem

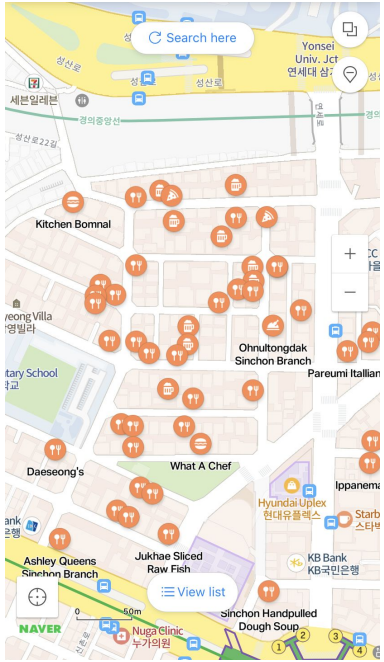
***TARGET GROUP*** Tourists, people who are new in a city (e.g. expats)

***SETTING***

You are in a new city and want to meaningfully explore without planning every detail in advance.

- Very time-consuming
- Can forget interesting info by the time you get there




➔ Information overload

- ➔ You might miss out on good places because there is too much choice!
- ➔ You might not think to look up places that you walk past!
- ➔ Have too look up history/interesting points about places on your own (which you are likely not to do on the go)

This is not the way in which you would be recommended a city by a friend!

They would show you around as you go, not the bird's eye view.

➔ How to provide this automatically without having to look up yourself?

# Key solution

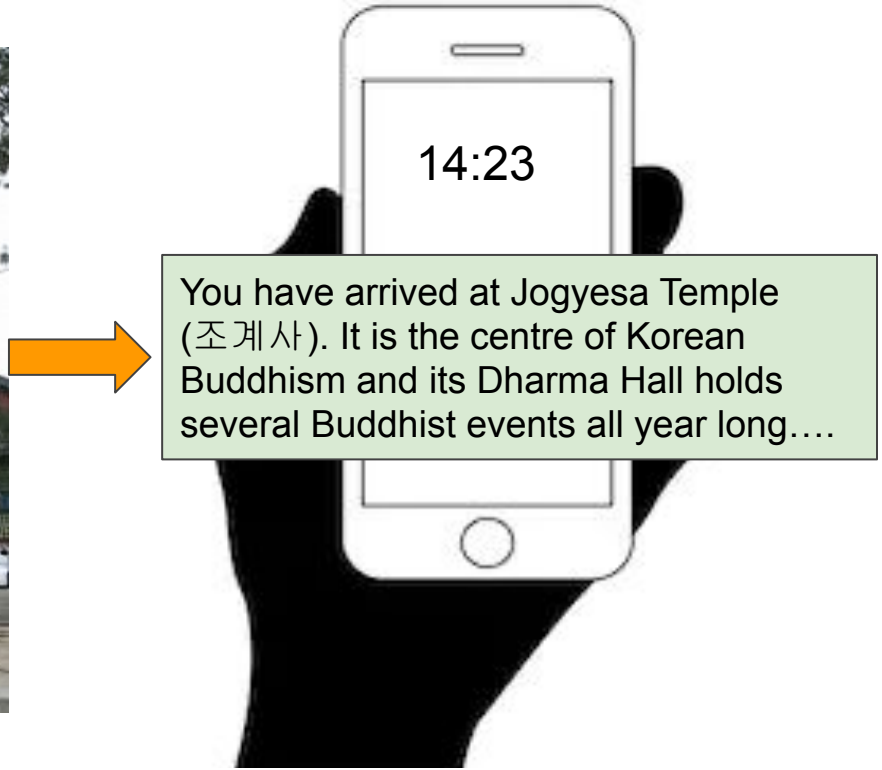
**Exploration vs. planned out itinerary**: key information for the user *as they go,* based on the user context *right now*.

- Provide specific information (such as history, etc.) about POIs as you're walking by
- Hear a voice clip, see key information at a glance or save the place to your places to visit

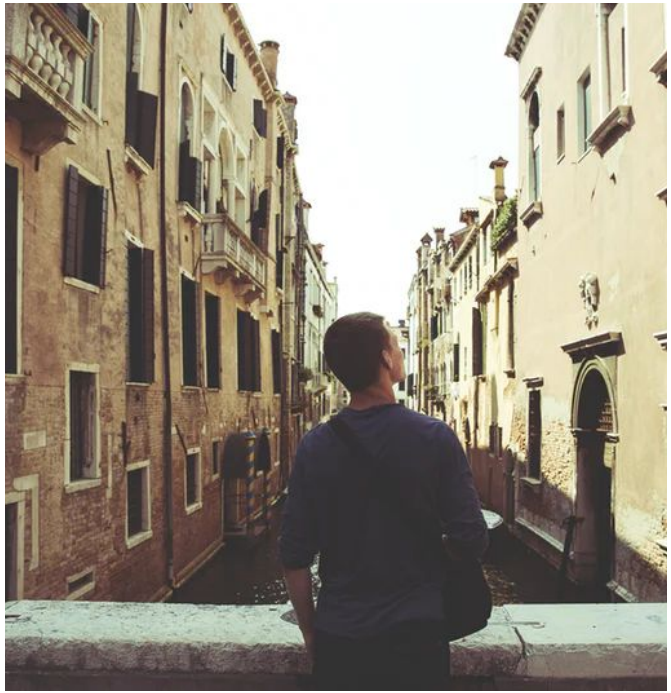
POIs could be anything: Restaurants, shopping malls, historical places etc.

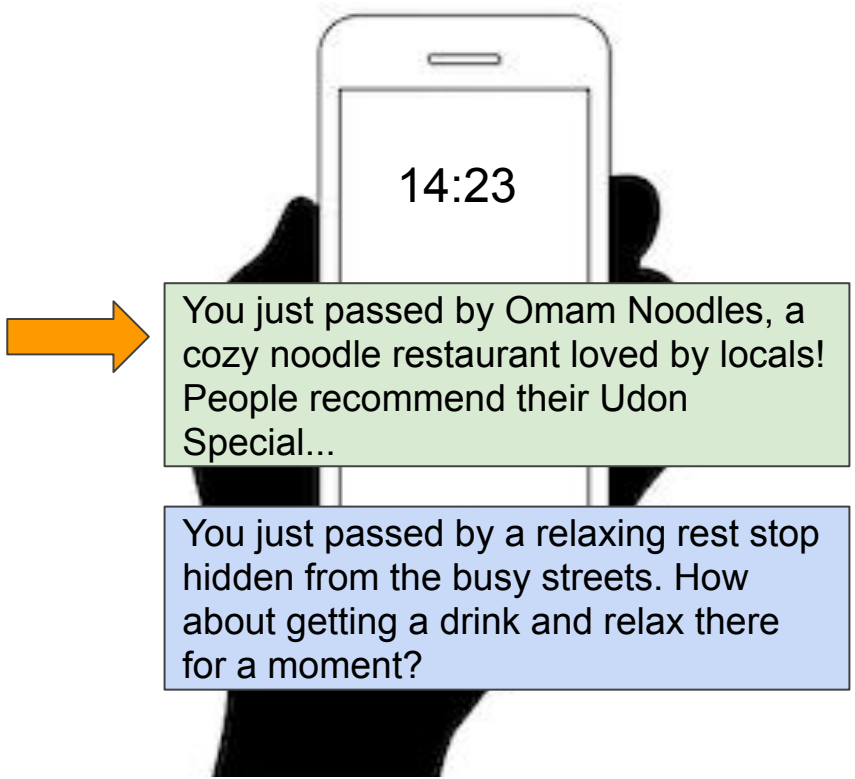
*Discover new places as you explore the city!*

# Usage scenarios

# Usage scenarios





# Distinguishing features

We aim to use **user context**: e.g. location, time, season, how busy a place is to provide you with the information you didn't know you wanted

Much like audio guides in museums, we can provide **location-specific context information** as you go, e.g. historical information, closing time, ratings.

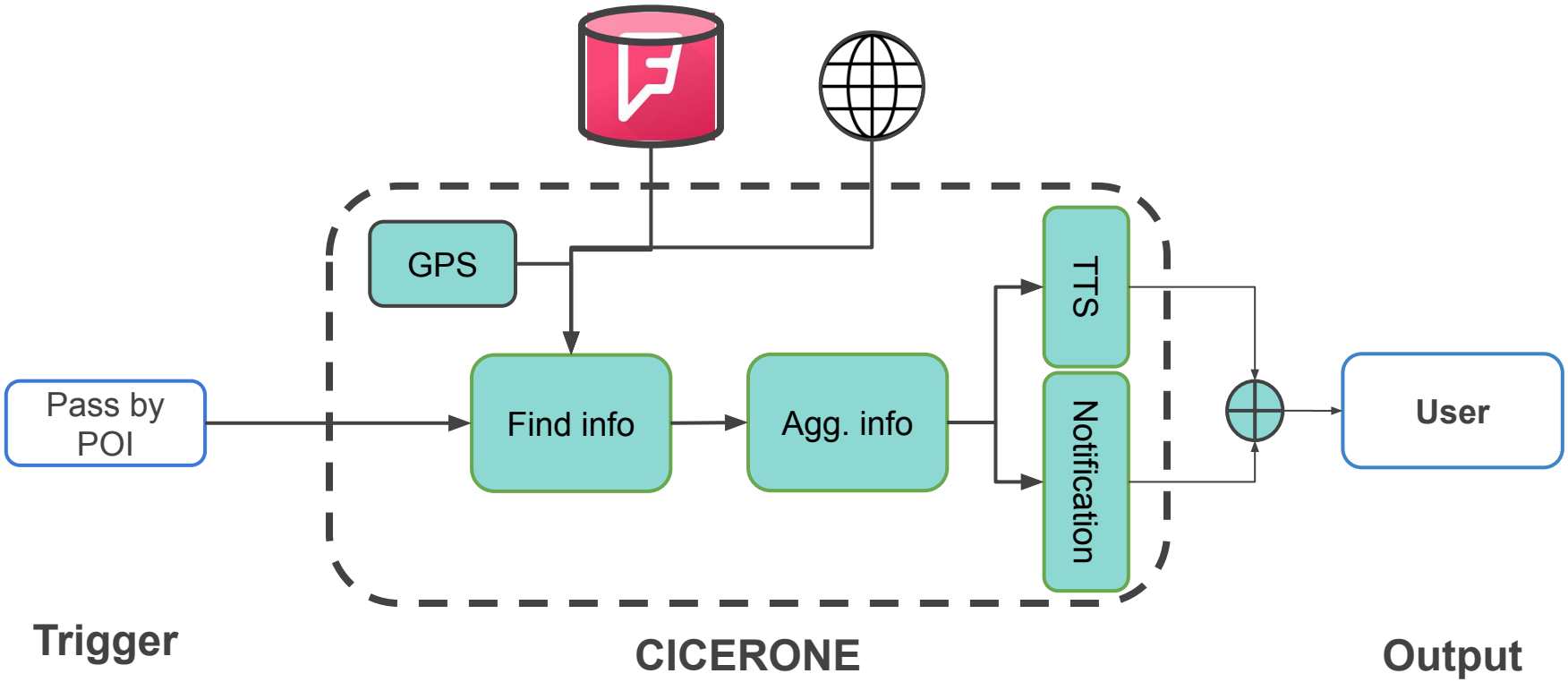
**Seamless** and no need to look it up yourself:

Receive information you didn't know you wanted while passing by.

# System architecture overview

1. Obtain user **location** - Google Play services (*FusedLocationProvider*)
2. Query for **nearby POIs** - Foursquare API. Basic information (*location, name, category...*)
3. Setup **geofences** around obtained POIs - Android Geofencing API
4. Fetch **detailed information** when a geofence is triggered - Wikipedia API *(summary, ...)*
5. Send **notification** to user with basic info - When pressed leads to a POI activity
6. POI activity **displays** the **detailed information** on the POI - TODO add TTS
7. **Remove** the triggered **geofence** (TODO)

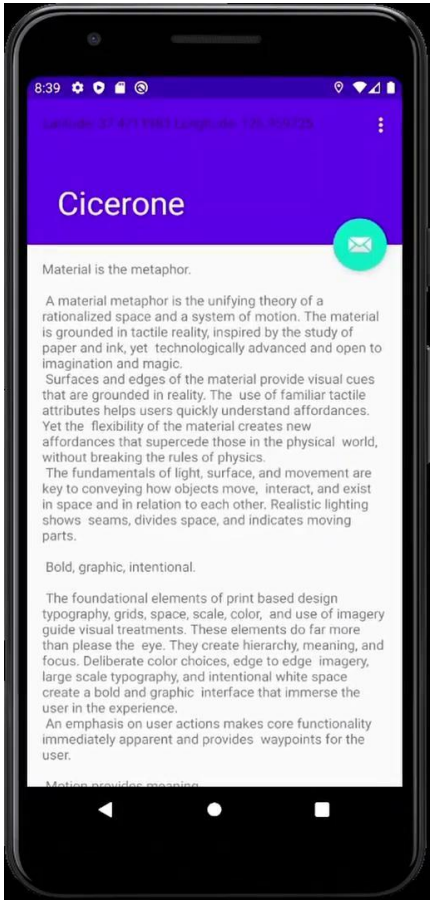
# System overview



# System demo

Cicerone:

1. Retrieves nearby POI from Foursquare
2. Sets up geofence
3. Triggering of geofence sends notification
4. Information is displayed in the POI activity



# Solved technical challenges

- **Retrieving** user location when application is in the **foreground**
  - Utilizes Google Play services API
- **Designing an API** for the app to retrieve information on nearby POIs
  - Foursquare API - Implemented
  - Wikipedia API - Mostly completed
  - Kakao API - In progress
- **Setup geofences** based on nearby POIs
  - Utilizes Android Geofence API
- Send **notification** containing information on the triggered geofence
  - Contains the name of the geofence - to let the user decide if it is interesting or not




# Remaining technical challenges

- **Retrieving** user location when application is in the **background**
  - In Android 10, location is updated only sporadically (a few times an hour) when in background.
  - Geofence location updates are more frequent (every few minutes)
  - *Possible solution*: Bulk fetch a lot of POIs in a large area around the user when location is available
- **Overlapping** geofences
  - *Current solution*: Set a threshold for how close geofences can be, and dynamic radius.
- **Aggregating** information into one piece of information
  - Combining data from Foursquare, Wikipedia and ev. Kakao dynamically.
- **Filtering** of POIs: Recommend only top-rated place in a small distance
  - E.g. based on Kakao ratings. Problem: Foursquare supplies ratings only in Premium version.
- Using text-to-speech to **generate audio tracks**

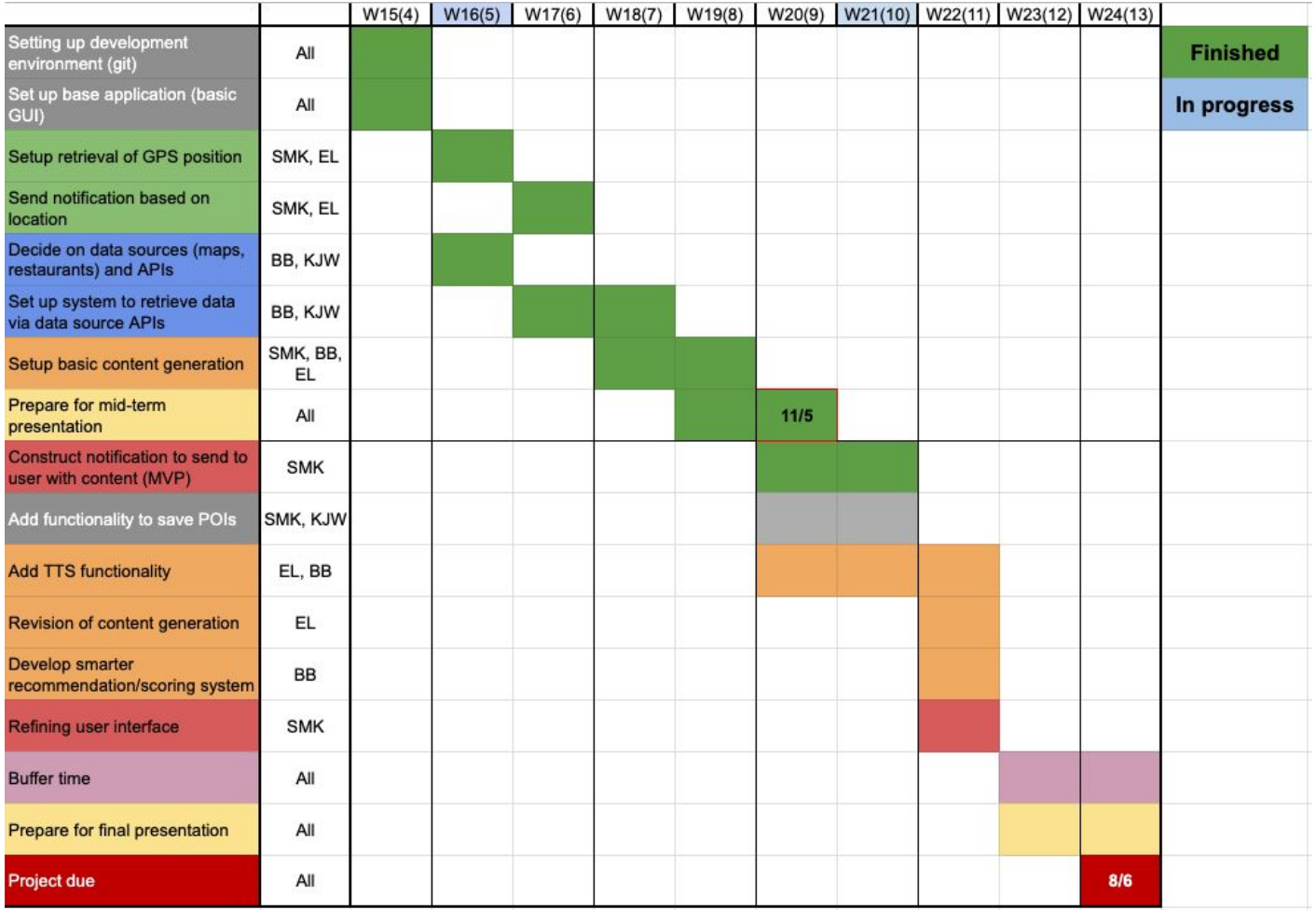




# User experience challenges

- How do we display the detailed information on the POI in a clear way?
  - *Current solution*: POI activity
- How frequently do we send recommandations?
  - *Current solution*: As soon as a geofence is triggered
- How do we formulate the aggregated texts?
  - *Current solution*: Title from Foursquare + Body text from Wikipedia
- How do the user access a previously displayed POI?
  - *Possible solution*: Display history of visited geofences
- How can the user get more information?
  - *Possible solution*: Hyperlinks to Wikipedia if available

# Project plan




Full schedule:
https://docs.google.com/spreadsheets/d/1mL5yhuGn5g0Rd8uT927GOU4tvzUW-SXN7fr32s-0OY8/edit

# Final deliverable

An Android **app** with the following **core functionality**

- **Extract** nearby POI:
  - At least use GPS
- **Find information** on POI
  - At least one of the following: name, history, rating
- **Aggregate** information
  - Should be a coherent piece of text/content
- **Present** text to user
  - At least in the form of a push notification, possibly TTS.

# Success criteria

**Success criteria** - the criteria are considered fulfilled if:

1. User gets notification with information on POI within reasonable time of passing it, information is accurate
2. The system is seamless and easy to use while on-the-go

# Potential threats

- Limited usefulness if we can’t use app reliably in background

- Might not be able to get ahold of ratings for recommendation for MVP
  - Kakao does not supply API for ratings -> Can scrape, non-trivial task
  - Foursquare hides ratings behind paywall
  - Google maps hides all queries behind paywall, but has ratings

- Foursquare does not hold the best information in Korea
  - Our app is in English, but place names are usually only in Korean

- Difficult to get Wikipedia article with Foursquare name
  - Foursquare has some info behind paywall

# Thank you for your attention!

Questions, comments?