Spring 2020, SNU | Mobile and Ubiquitous Computing

Cicerone

Explore with a knowledgeable friend

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

Final presentation

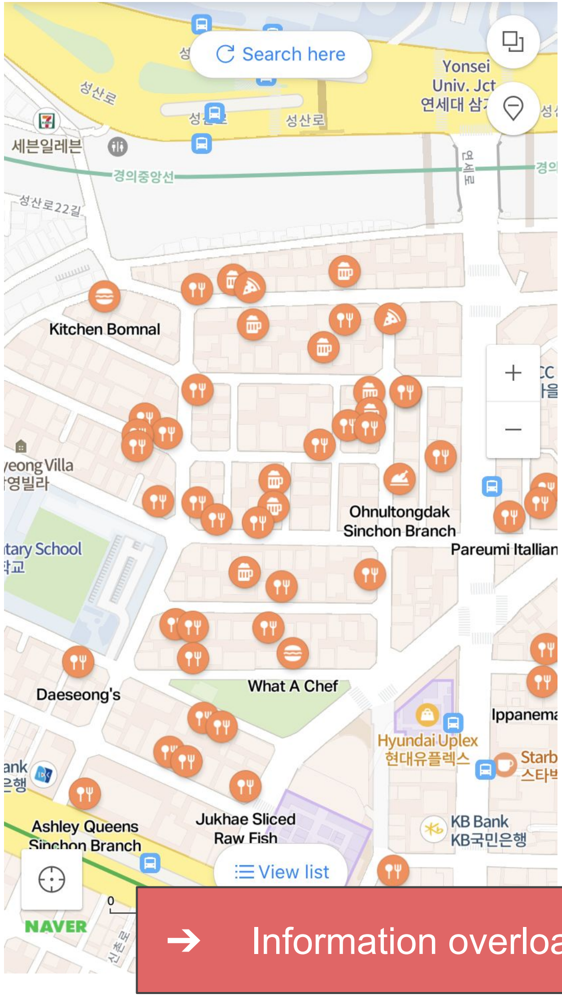
Motivation & Problem

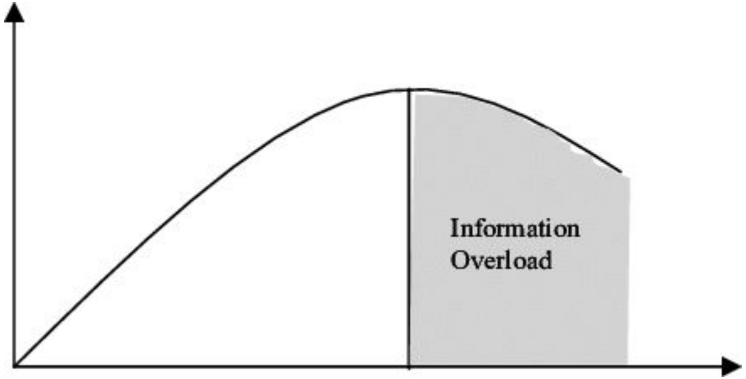
Cicerone

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 simply by going out without having to plan in advance.

Cicerone





→ Information overload

- ➔ Too much choice!
- ➔ Might miss places that you walk past!
- ➔ Have to look up actively look up information

Bird’s eye view is not how you would be introduced to a city by a friend!

➔ A friend would simply show you around as you go out and explore!

Key solution

Exploration vs. planned out itinerary: recommendations *as you go*, based on context *right now.* As you're walking by:

- See key information at a glance, listen to a voice clip or save the place to visit it later

POIs could be anything: Restaurants, historical sites, shops etc.

Discover new places as you explore the city!

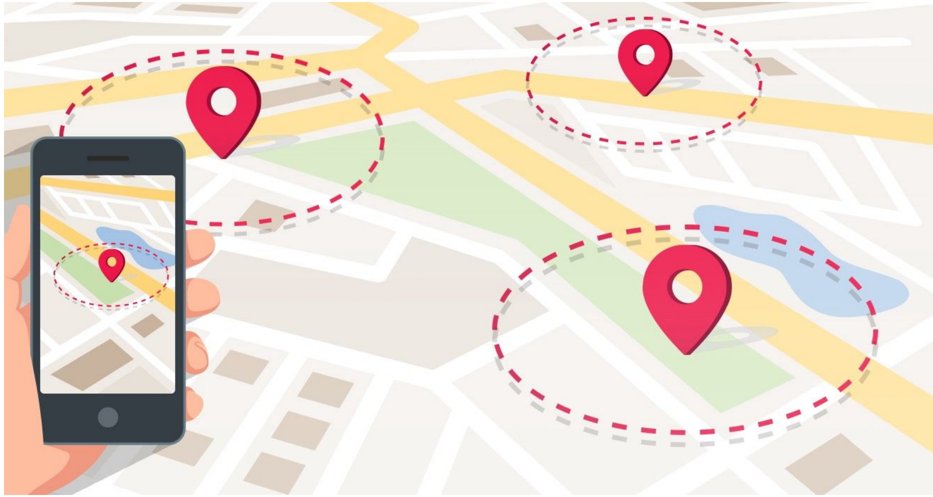
Distinguishing features

Cicerone

We use **location** information and **preferences** to provide you with the information you didn’t know you wanted.

As you pass by we provide **location-specific context information:** *historical background, closing time, ratings, and even tips.*

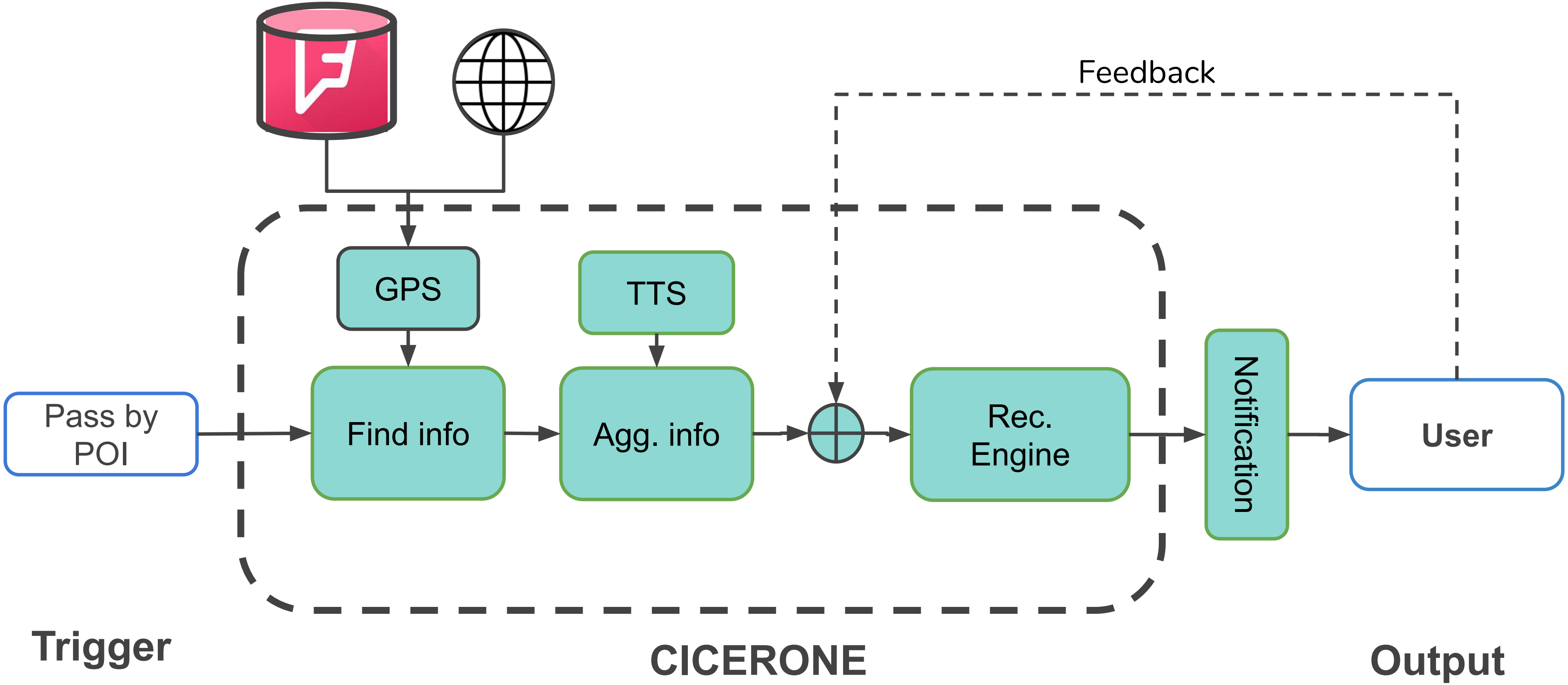
The entire process is **seamless**.



System architecture overview

1. Obtain user **location** - Google Play services (*FusedLocationProvider*)
2. Query for **nearby POIs** - Foursquare API. Basic information (*location, name, category…*)
3. **Filter** POIs by **recommendation score** - Recommendation algorithm
4. Setup **geofences** around obtained POIs - Android Geofencing API
5. Fetch **detailed information** when a geofence is triggered - Wikipedia API *(summary, …)*
6. Send **notification** to user with basic info - When pressed leads to a POI activity
7. POI activity **displays** the **detailed information** on the POI
8. Add POI to **history**, give option to add it to **saved places**
9. **Update recommendation scores** with user feedback

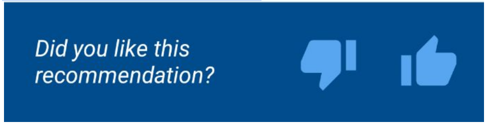
System overview



Recommendation algorithm

Cicerone

User feedback:



Did you like this recommendation?

👎 👍

Problem: **What POI to recommend to user?**

Solution: We chose a *Bayesian formulation* based on user feedback *per category*,

i.e. What is the probability a user likes a place given that it is from a certain category?

We can compute a POI's recommendation score as the posterior

$$P(Liked|Category) = \frac{P(Category|Liked)\ P(Liked)}{P(Category|Liked)\ P(Liked) + P(Category|Disliked)\ P(Disliked)}$$

Priors are chosen and may be updated with new data. **Likelihoods** are obtained from the database, i.e. query for fraction of category out of all liked/disliked POIs.

Pseudocounts are added to deal with cold start. *Increase/decrease to adjust initial sensitivity.*

Recommendation algorithm

Assume we have a set of POIs $P = \{p_1, ..., p_n\}$ and each p_i belongs to a category C_j .

Then the set of all POIs $P = C_1 \cup C_2 \cup ... \cup C_m$

We want to **choose** k **POIs** weighted by their respective category's probability score.

We can accomplish this as follows:

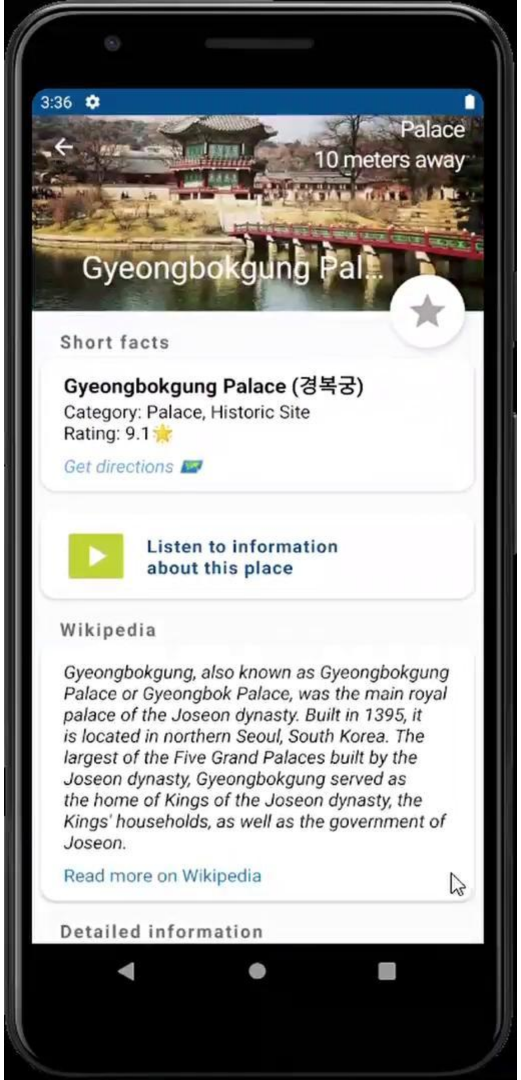
1. Obtain category scores by normalizing over all scores $P(Liked|Category)$
2. Draw a category, probabilities are weighted by their respective score
3. Draw a POI from the chosen category with uniform probability.
4. Repeat above steps k times.

System demo

- *Optimal case*

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, and TTS is generated.



3:36

Palace
10 meters away

Gyeongbokgung Pal...

Short facts

Gyeongbokgung Palace (경복궁)
Category: Palace, Historic Site
Rating: 9.1
Get directions

Listen to information about this place

Wikipedia

Gyeongbokgung, also known as Gyeongbokgung Palace or Gyeongbok Palace, was the main royal palace of the Joseon dynasty. Built in 1395, it is located in northern Seoul, South Korea. The largest of the Five Grand Palaces built by the Joseon dynasty, Gyeongbokgung served as the home of Kings of the Joseon dynasty, the Kings' households, as well as the government of Joseon.

Read more on Wikipedia

Detailed information

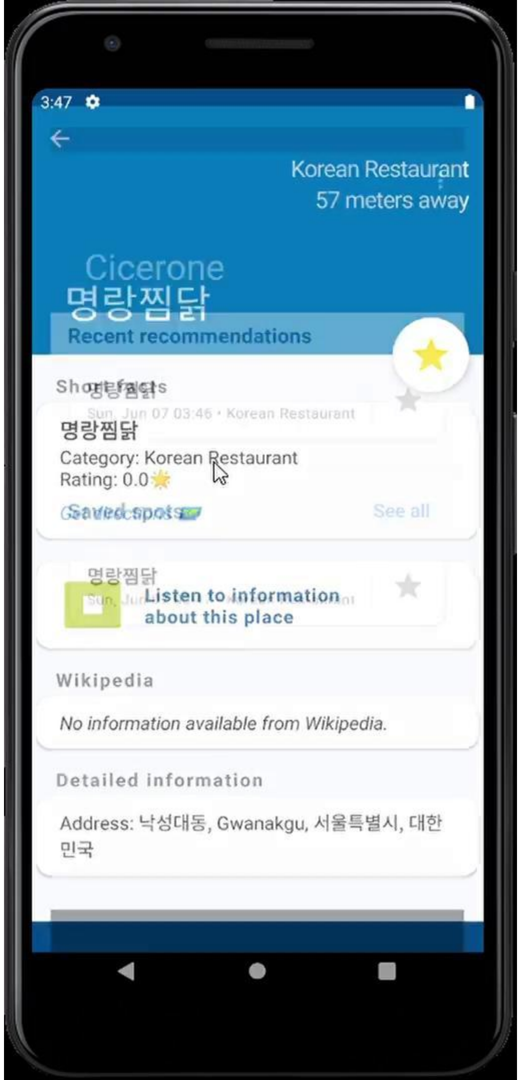
Cicerone

System demo

- *Suboptimal case*

Cicerone:

1. Retrieves nearby POI from Foursquare
2. Sets up geofence
3. Triggering of geofence sends notification
4. Available information is displayed in the POI activity
5. Placeholder texts/TTS are displayed where information is not available.



Korean Restaurant
57 meters away

명랑찜닭

Short facts

Sun, Jun 07 03:46 · Korean Restaurant

명랑찜닭
Category: Korean Restaurant
Rating: 0.0

Wikipedia

No information available from Wikipedia.

Detailed information

Address: 낙성대동, Gwanakgu, 서울특별시, 대한민국

Cicerone

Solved technical challenges (since mid-term)

- **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)
  - *Current solution*: Bulk fetch a lot of POIs in a large area around the user when location is available.
- **Filtering** of POIs: Recommend best POI to user ✓
  - Bayesian recommendation algorithm based on user feedback - like/dislike categories for POIs.
  - Categories: *restaurant, palace, museum...*
- **Overlapping** geofences ✓
  - Greedy selection algorithm, where the POI which has the highest category score is selected.
- **Aggregating** information into one piece of information ✓
  - Combines information from Foursquare and Wikipedia dynamically.
- Using text-to-speech to **generate audio tracks** ✓

User experience challenges

- ✓ ➔ How do we formulate the aggregated texts?
    - ◆ *Solution*: Text boxes with information from Wikipedia & Foursquare
- ✓ ➔ How frequently do we send recommandations?
    - ◆ *Solution*: As soon as a geofence is triggered after timeout of 15 min
- ✓ ➔ How do the user access a previously displayed POI?
    - ◆ *Solution*: Display history of visited geofences and saved POIs
- ✓ ➔ How can the user get more information?
    - ◆ *Solution*: Hyperlinks to Wikipedia and Google Maps

- ✓ ➔ How do we display the detailed information on the POI in a clear way?
    - ◆ *Solution*: POI detail activity with clear sections with information

Evaluation - User study

“I’ve been in the situation a lot of times where I see something when walking in the street and then going to it ***without knowing what it is****. Then it would make good sense to have this kind of* ***guide with you all the time*** *that is connected to your location”.*

- Interviewed user on what purpose they think Cicerone fulfills

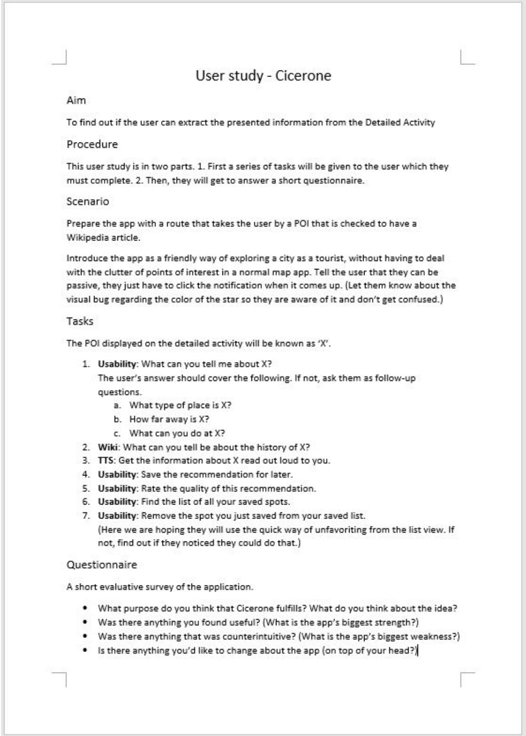
Evaluation - User study

Setup

- 2 users in the study group. User's interacted with Cicerone in **Android Emulator**.
- Users were given tasks to complete - "*What is the name of this place?", "Describe the history of this place"* etc.
- Short evaluative questionnaire afterwards

Results

- **Both users solved all tasks** => Indicates information is displayed in a clear way to the user.
- **Feedback**: Easy way to get recommendations of nice spots without any effort. Similar to recommendations from Google Maps, but Cicerone is more centered towards tourists.
- **Improvement points**: Make TTS/hyperlink to Google Maps more obvious, as well as more feedback on like/dislike.



User study - Cicerone

Aim

To find out if the user can extract the presented information from the Detailed Activity

Procedure

This user study is in two parts. 1. First a series of tasks will be given to the user which they must complete. 2. Then, they will get to answer a short questionnaire.

Scenario

Prepare the app with a route that takes the user by a POI that is checked to have a Wikipedia article.

Introduce the app as a friendly way of exploring a city as a tourist, without having to deal with the clutter of points of interest in a normal map app. Tell the user that they can be passive, they just have to click the notification when it comes up. (Let them know about the visual bug regarding the color of the star so they are aware of it and don't get confused.)

Tasks

The POI displayed on the detailed activity will be known as 'X'.

1. **Usability**: What can you tell me about X?
   The user's answer should cover the following. If not, ask them as follow-up questions.
   a. What type of place is X?
   b. How far away is X?
   c. What can you do at X?
2. **Wiki**: What can you tell be about the history of X?
3. **TTS**: Get the information about X read out loud to you.
4. **Usability**: Save the recommendation for later.
5. **Usability**: Rate the quality of this recommendation.
6. **Usability**: Find the list of all your saved spots.
7. **Usability**: Remove the spot you just saved from your saved list.
   (Here we are hoping they will use the quick way of unfavoriting from the list view. If not, find out if they noticed they could do that.)

Questionnaire

A short evaluative survey of the application.

- What purpose do you think that Cicerone fulfills? What do you think about the idea?
- Was there anything you found useful? (What is the app's biggest strength?)
- Was there anything that was counterintuitive? (What is the app's biggest weakness?)
- Is there anything you'd like to change about the app (on top of your head?)

Result - 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.

BONUS: Our app is not limited to South-Korea and might perform even better in English-speaking countries due to the sources we are using at the moment.

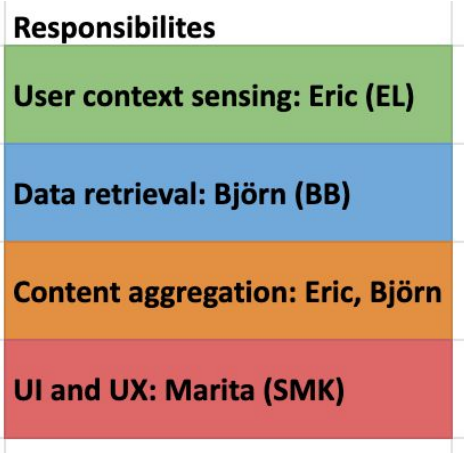
Result - Success criteria

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

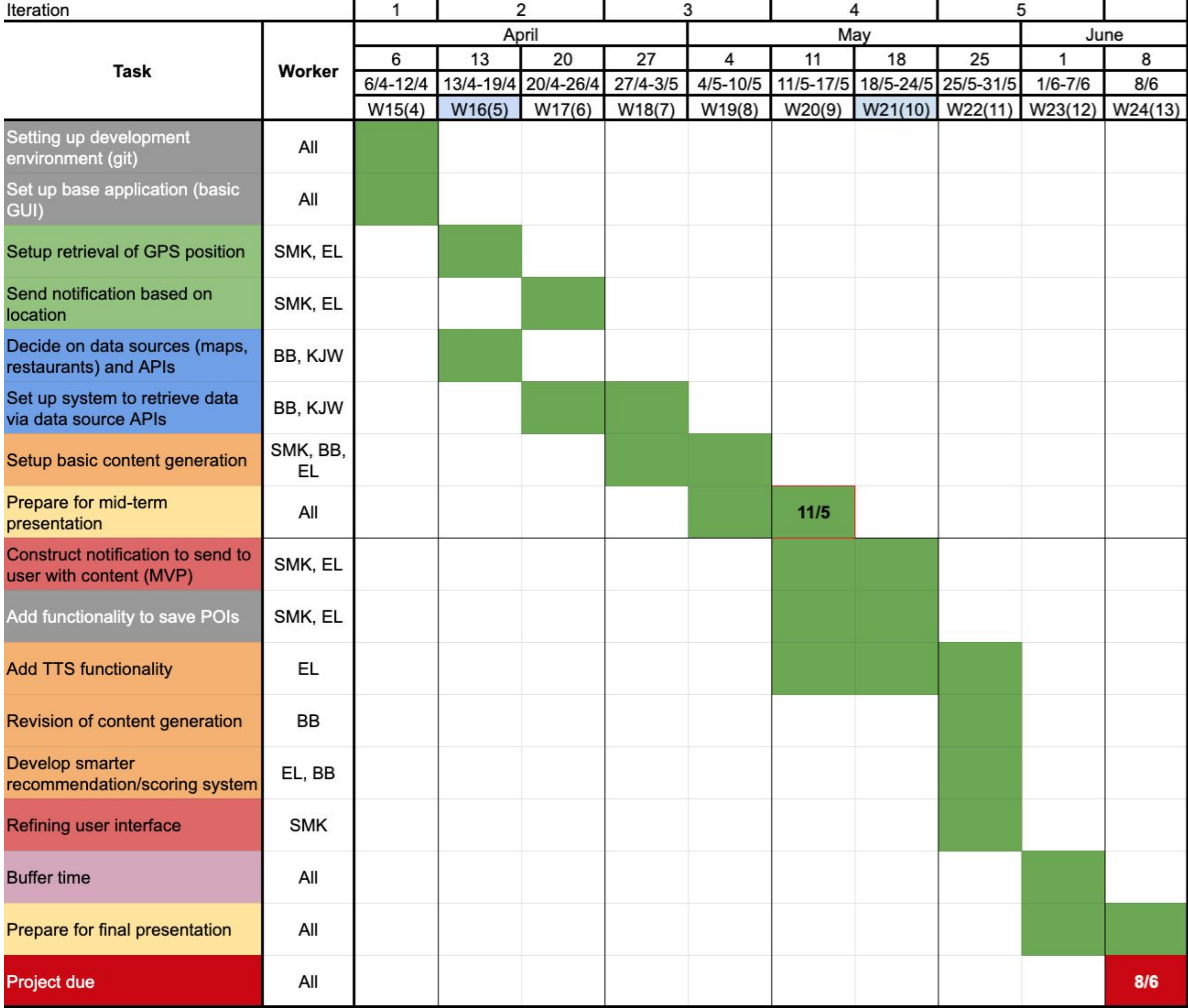
✓ 2. The system is seamless and easy to use while on-the-go

We tested on a real Samsung S10 device and it works as intended.



Responsibilites
User context sensing: Eric (EL)
Data retrieval: Björn (BB)
Content aggregation: Eric, Björn
UI and UX: Marita (SMK)

Project plan



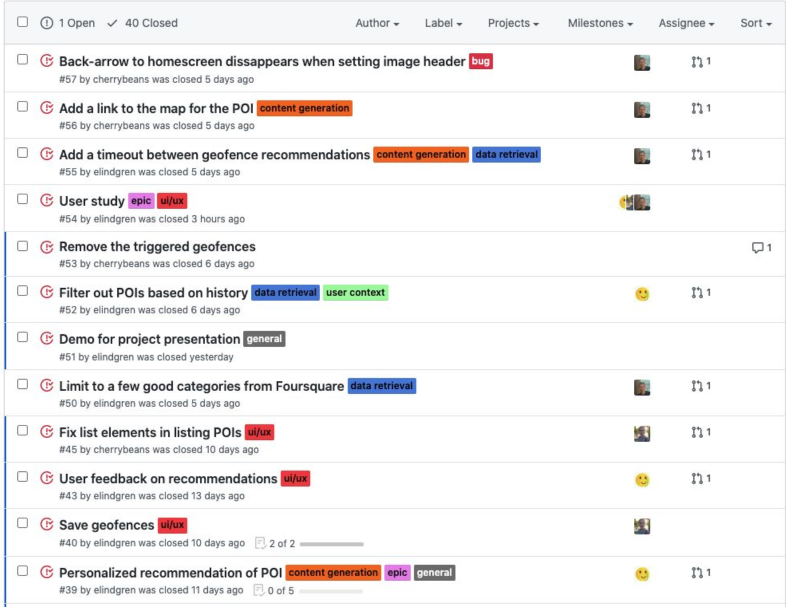
Iteration		1	2		3		4		5		
Task	Worker	April				May				June	
		6	13	20	27	4	11	18	25	1	8
		6/4-12/4	13/4-19/4	20/4-26/4	27/4-3/5	4/5-10/5	11/5-17/5	18/5-24/5	25/5-31/5	1/6-7/6	8/6
		W15(4)	W16(5)	W17(6)	W18(7)	W19(8)	W20(9)	W21(10)	W22(11)	W23(12)	W24(13)
Setting up development environment (git)	All	🟩									
Set up base application (basic GUI)	All	🟩									
Setup retrieval of GPS position	SMK, EL		🟩								
Send notification based on location	SMK, EL			🟩							
Decide on data sources (maps, restaurants) and APIs	BB, KJW		🟩								
Set up system to retrieve data via data source APIs	BB, KJW			🟩	🟩						
Setup basic content generation	SMK, BB, EL				🟩	🟩					
Prepare for mid-term presentation	All					🟩	11/5				
Construct notification to send to user with content (MVP)	SMK, EL						🟩	🟩			
Add functionality to save POIs	SMK, EL						🟩	🟩			
Add TTS functionality	EL						🟩	🟩	🟩		
Revision of content generation	BB								🟩		
Develop smarter recommendation/scoring system	EL, BB								🟩		
Refining user interface	SMK								🟩		
Buffer time	All									🟩	
Prepare for final presentation	All									🟩	🟩
Project due	All										8/6

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

Cicerone

Project stats

- We used the GitHub issue tracker and project board



□ ⓘ 1 Open ✓ 40 Closed Author ▾ Label ▾ Projects ▾ Milestones ▾ Assignee ▾ Sort ▾

□ Back-arrow to homescreen dissappears when setting image header bug 1
#57 by cherrybeans was closed 5 days ago

□ Add a link to the map for the POI content generation 1
#56 by cherrybeans was closed 5 days ago

□ Add a timeout between geofence recommendations content generation data retrieval 1
#55 by elindgren was closed 5 days ago

□ User study epic ui/ux
#54 by elindgren was closed 3 hours ago

□ Remove the triggered geofences 1
#53 by cherrybeans was closed 6 days ago

□ Filter out POIs based on history data retrieval user context 1
#52 by elindgren was closed 6 days ago

□ Demo for project presentation general
#51 by elindgren was closed yesterday

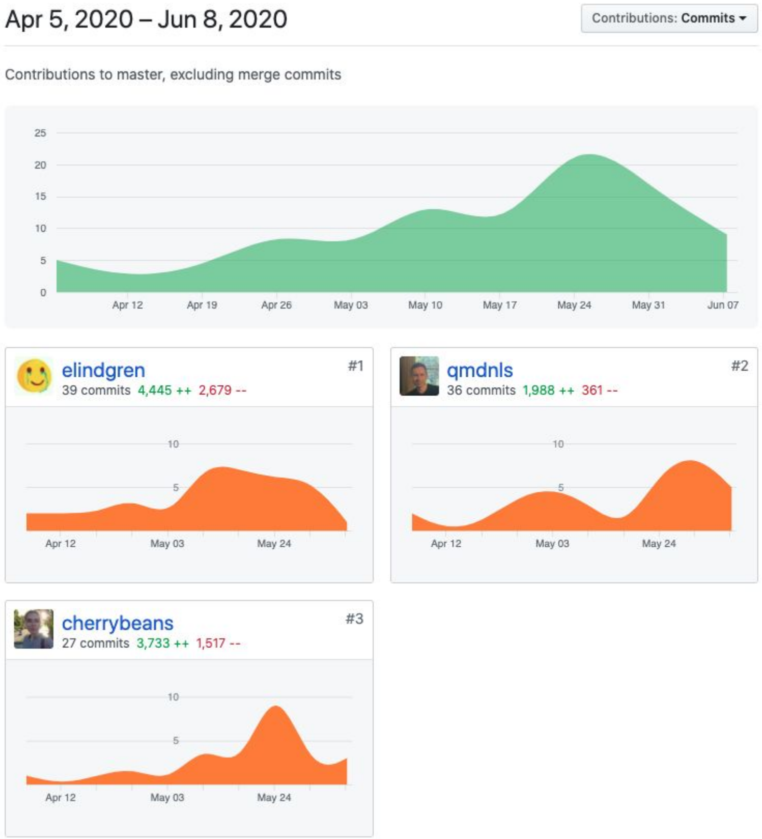
□ Limit to a few good categories from Foursquare data retrieval 1
#50 by elindgren was closed 5 days ago

□ Fix list elements in listing POIs ui/ux 1
#45 by cherrybeans was closed 10 days ago

□ User feedback on recommendations ui/ux 1
#43 by elindgren was closed 13 days ago

□ Save geofences ui/ux
#40 by elindgren was closed 10 days ago 2 of 2

□ Personalized recommendation of POI content generation epic general 1
#39 by elindgren was closed 11 days ago 0 of 5



Apr 5, 2020 – Jun 8, 2020

Contributions: **Commits** ▾

Contributions to master, excluding merge commits

elindgren #1
39 commits 4,445 ++ 2,679 --

qmdnls #2
36 commits 1,988 ++ 361 --

cherrybeans #3
27 commits 3,733 ++ 1,517 --

Reflecting on our project

- Dividing tasks into smaller issues is very helpful for keeping focus
- We had a lot of difficulties with finding out how to asynchronous data from database and APIs in Kotlin
- Good communication and planning helped the project go smoothly (productive meetings). Easy to have meetings because of Zoom.

Future work and refinements

- Add more information sources for more useful content to the user
- Improve recommendation system - more sources of feedback to the system
  - E.g. When user installs app, can select some of their interests
  - Using paid API gives us more information to use as basis for recommendation.
- Add more visual feedback to the user
- Pull-to-refresh for loading new POIs manually
- Give users the option to see some nearby POIs.
- Test it on more users, refine app and remove bugs

Thank you for your attention!

Questions, comments?

Cicerone

by Team 4