NeAR Me

Interactive AR mobile app to enrich your daily life

Motivation

- "Open now near me" searches have grown globally by over 400% year-over-year [1].
- Most smartphone users continue to rely on maps and navigational applications.
- AR is gaining popularity worldwide [2].
- Very few public facing apps allow AR postings.

Objective

- Design an app that allows users to post and receive a topic-based AR content feed.
- Provide users with a natural way to view location-specific information.
- Enable users to access information about their surroundings on-the-go, without the need to stop and manually search for it.

Concepts

- Users see surrounding AR **messages** updated in real-time.
- Users can **subscribe** to specific **topics** to receive a personalized feed of AR messages.
- Users can create and manage region-specific topics to interact with others.
- Users can publish 2-D AR **comments** and 3-D AR **posts** to describe their surroundings.
- Users get personalized topic and message recommendations.

Acknowledgements

- Pouya Mehrannia (Consultant)

References

1. Ready, B. (2022, March). 2022 Retail Marketing Guide: Drive foot traffic and in-store sales. Google. Retrieved May 23, 2022, from https://www.thinkwithgoogle.com/consumer-insights/consumer-journey/increase-foot-traffic-and-in-store-sales/
2. Alsop, T. (2021, November 29). Number of mobile augmented reality (AR) active users worldwide from 2019 to 2024. Statista. Retrieved May 23, 2022, from https://www.statista.com/statistics/1098630/global-mobile-augmented-reality-ar-users

Design Overview

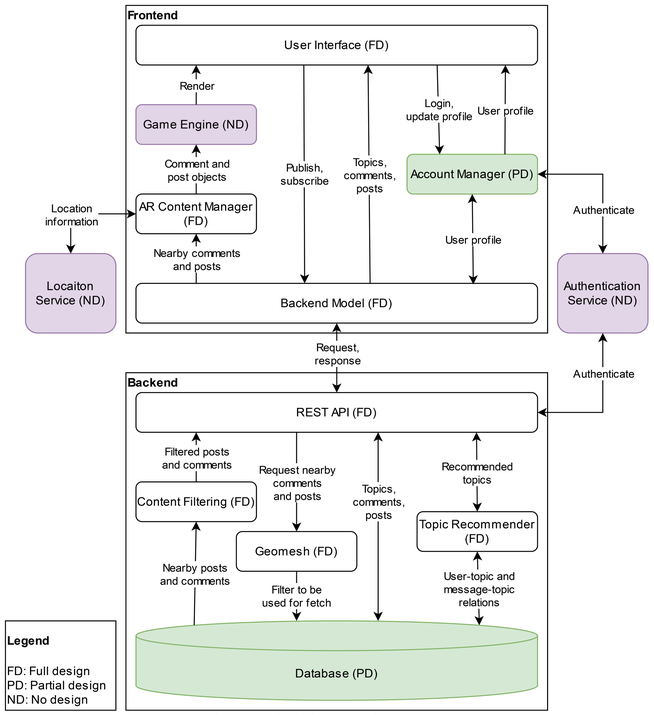

Figure 1: Block diagram of NeAR Me.

Design Highlights

Geomesh

- Adaptive geographical grouping of messages (larger mesh for sparse areas, smaller mesh for crowded areas).
- Time-saver for location-based message queries.

Content Filter

- Controls AR message volume to avoid overwhelming the frontend.
- Allows users to set their preference for AR display priority.

Topic Recommender

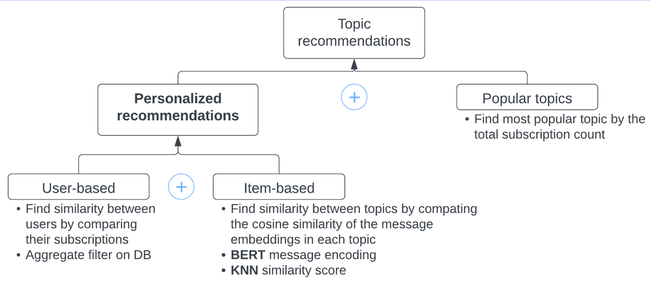

Figure 2: Topic recommender design.

Design Alternatives

Component	Alternatives	Advantages
Frontend	**1. Unity** 2. Unreal Engine	+ Good AR support + Compatible with Google APIs + Personal familiarity
Backend	**1. Java SpringBoot** 2. Python Django 3. Serverless cloud	+ Concurrent response speed + Level of control + Better security
Database	**1. MongoDB** 2. MySQL 3. Redis	+ Fast query speed + Flexible datatype + Affordable pricing + Easy to configure
Deployment Strategy	**1. Cloud - Kubernetes** 2. Cloud - VM 3. On-prem server	+ High scalability + High fault tolerance + Easy to configure

Analysis

Latency Analysis

- FE request -> Latency -> BE auth -> BE processing -> Latency -> FE rendering

Colors	Latency type	Stat acquisition	Estimate
Brown	User-based	uncontrollable	< 300ms
Black	Quick transaction	FE and BE logs	< 5ms
Blue	Processing delay	E2E simulation script	300~500ms

- E2E simulation: Simulate a moving user on the same host with resource-limited BE instance (the slow action)

Topic Recommender Analysis

- Topic similarity model is tested against a 20000-row Reddit dataset, with topics=subreddit and message=Reddit posts.
- User similarity model is tested against 100 generated users, sampling subscriptions from 20 random topics.

	Time (train)	Time (recommend)
Topic-based	7:51:54.987679	0:00:00.053471
User-based	-	0:00:00.086297

Presented by: Alice Ye, Aidan Foster, Michael Sawyer, Sean (Shizhen) Li.

Group 2023.12