

Bubbler: Personalized and In-Control Social Media

Erick Cuadra, Alyssa Hooper, Mauricio Moreno, Nishan Narain, Steven Fandino · Instructor: Prof. Masoud Sadjadi

Problem

- The user’s social media browsing is driven by an invisible algorithm, with no transparency.
- Everything is spoonfed to the user; there is no deliberation.
- Apps prioritize view time and comment engagement, meaning usage is based on anger instead of having fun.
- The user has little ability to truly control what they see on social media apps.

Stakeholders & Requirements

Stakeholders include our team and those interested in personalizing their social media experience.

Functional requirements:

- A new “graph” system of interacting with posts, designed from scratch.
- The ability to search other users’ posts.
- The ability to adjust algorithmic preferences, including preferred/blacklisted topics and graph strategies (similarity in candidate posts, randomness, etc.)
- A traditional feed view of posts sorted by topic.
- The ability to write posts, edit, and delete them, and classify posts by topic (or “General”).

Non-functional requirements:

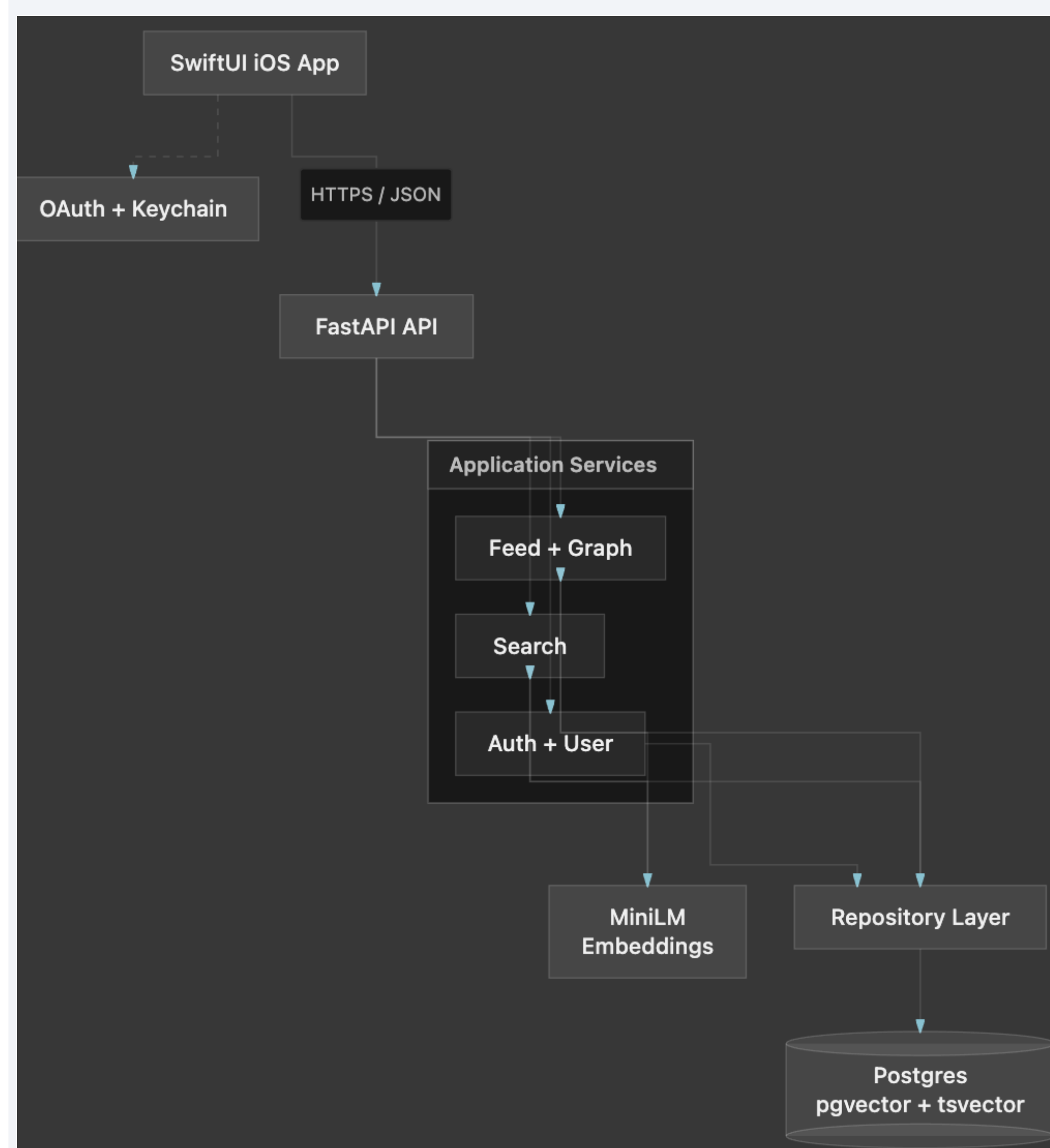
- Responsive <2sec feed, graph, search functions
- Ability to run on iOS devices
- Use requires a user account
- Scalable graph system (not hanging when significantly more posts)
- Easy to understand, fun graph view

Our Solution

We create a social media app which encourages users to diversify their browsing while also implementing a full revolutionary pipeline for users to interact with posts. In contrast to current social media apps, the design also ensures users have full control over what they are shown.

- Create adjustable, impactful algorithmic preferences
- Design a new social media view using Swift to encourage deliberate browsing
- Manage the full backend, frontend, database, and authentication stack
- Implement a graph-based post system designed to categorize posts by strategy and balance users’ preferences and healthy browsing

Architecture

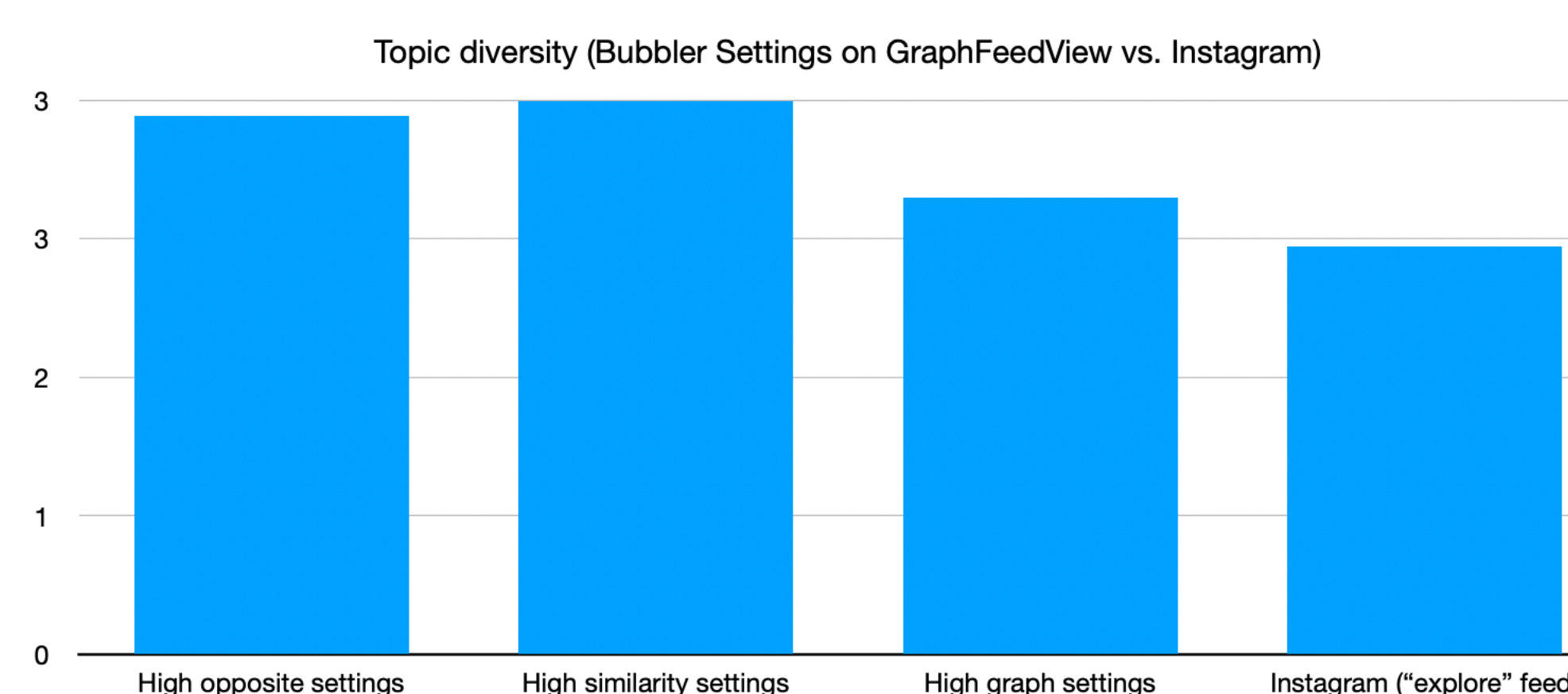


Demo & Screenshots



Two screenshots of the Bubbler GraphFeedView. Image 1 shows the current option “Bubbles”, shown by topic. Users can adjust settings to determine which Bubbles they’d like to see with how much diversity. The star above education shows it is a preferred topic and will appear more often. Image 2 shows the selected Education post; the user can select it, generating more Bubbles based on this post, or go back and select a different post. The “Explore” button allows for a hard refresh of the Bubbles.

Results & Evaluation



Across all settings, Bubbler showed a 32% increase in topic diversity, measured using Bubbler topics, compared to the equivalent post values in Instagram. Using similar and opposite settings, which unlike Graph do not emphasize post topics, showed a 12% increase in topic diversity over Graph itself, showing how these settings provide clear control over the topics they see.

Engineering Process (SO6)

Design decision	Why it matters	Testing highlight
Graph-based feed with tap-through paths	Replaces passive scrolling with deliberate post discovery.	Checkpoints traverse four unique connected posts and verify iOS graph wiring.
User-controlled recommendation weights	Lets users tune similarity, graph, opposite, random, diversity, and topic preferences.	Ranking checks cover stable and randomized ordering plus persisted settings.
Hard topic blacklisting	Keeps explicitly unwanted topics out of feed and graph results.	API and client checks confirm blacklist persistence and result filtering.
FastAPI service/repository separation	Keeps application logic independent from PostgreSQL access.	End-to-end checkpoints compare API feed content with database rows.
PostgreSQL hybrid retrieval	Combines pgvector semantic similarity with tsvector keyword search.	Search checks validate separate exact-match and related-result groups.
Backend-issued bearer auth with iOS Keychain storage	Centralizes authentication while preserving secure sessions across launches.	Checks cover register/login, protected routes, bearer headers, and the Keychain lifecycle.

What's Next

Many relevant social media features need to be added, such as:

- Comments (in some form)
- The ability to follow and block users
- A proper moderation system

There is also a need to expand out of hardcoded topics. Future plans involve an AI/ML system which can automatically identify and classify new/existing topics, allowing for diverse exploration based on current events or specific locations.

Acknowledgments

KFSCIS Capstone Program, Prof. Masoud Sadjadi.