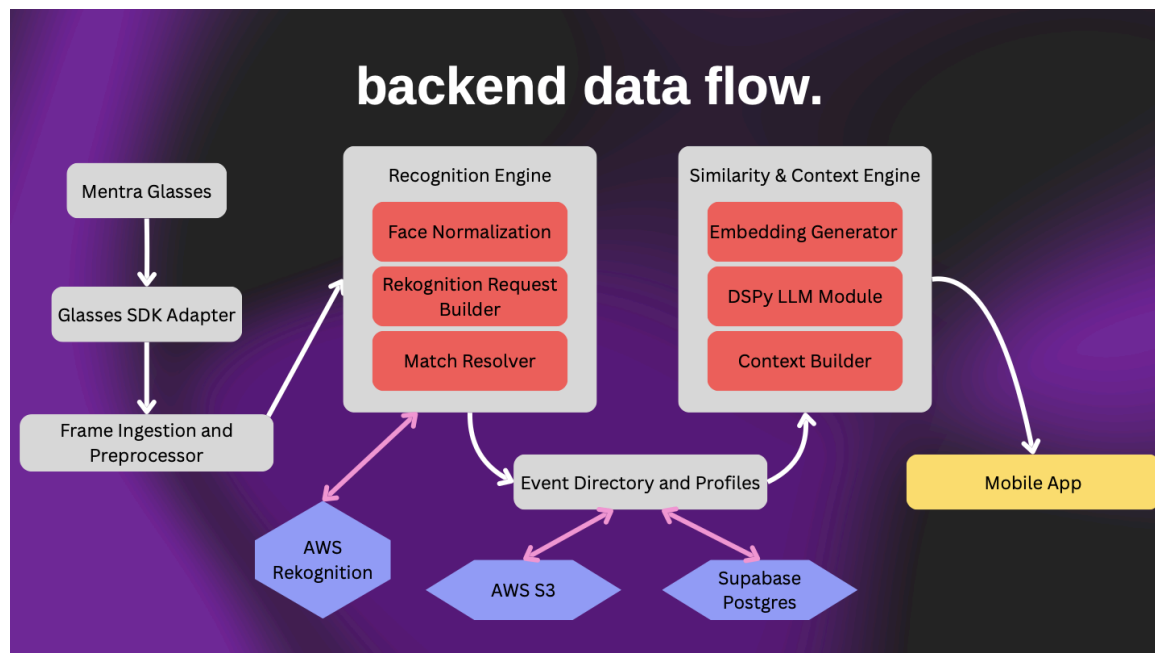
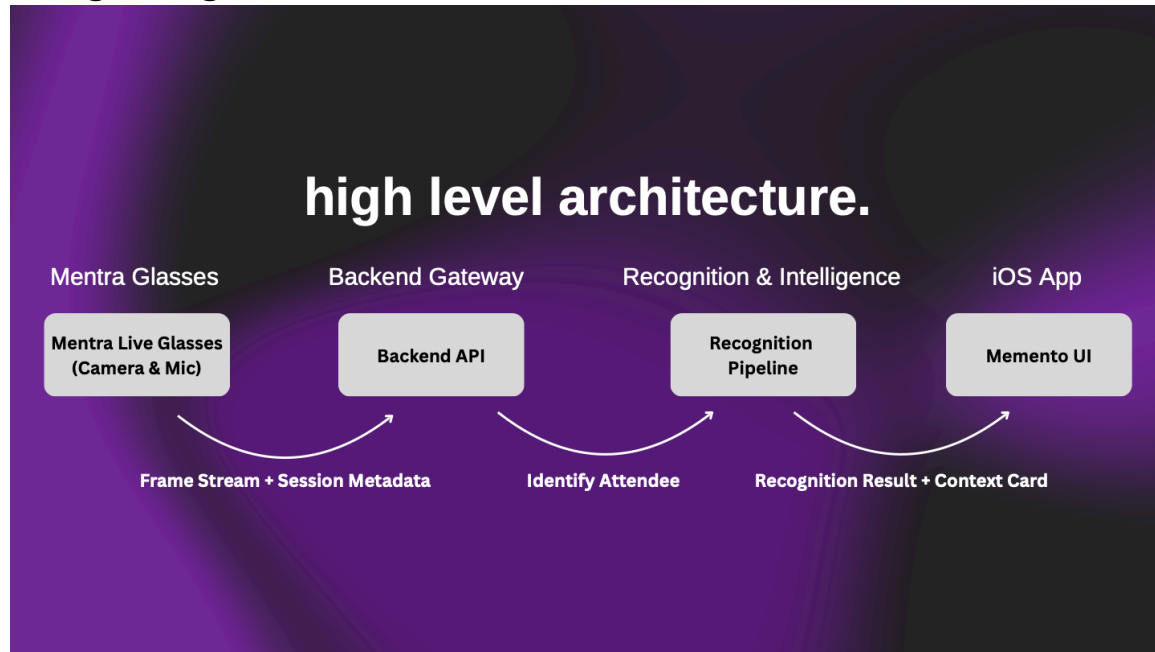


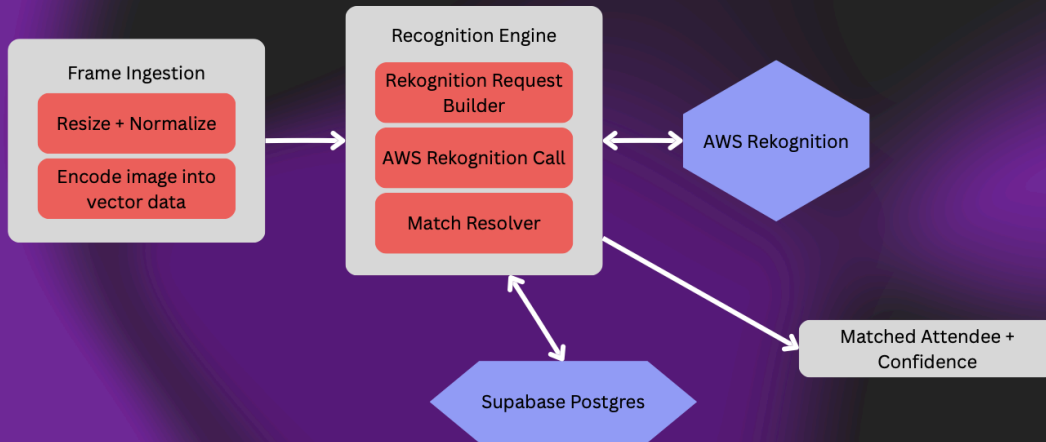
ECE 49595 - Team 5 Design Document

Fall 2025

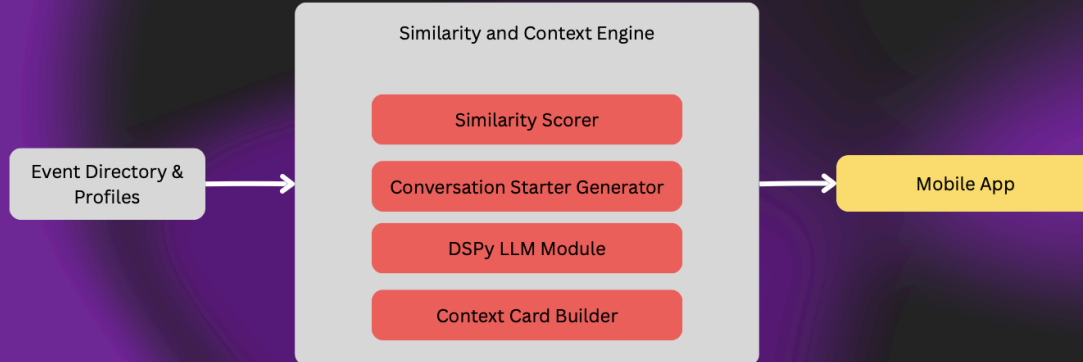
Design Diagrams

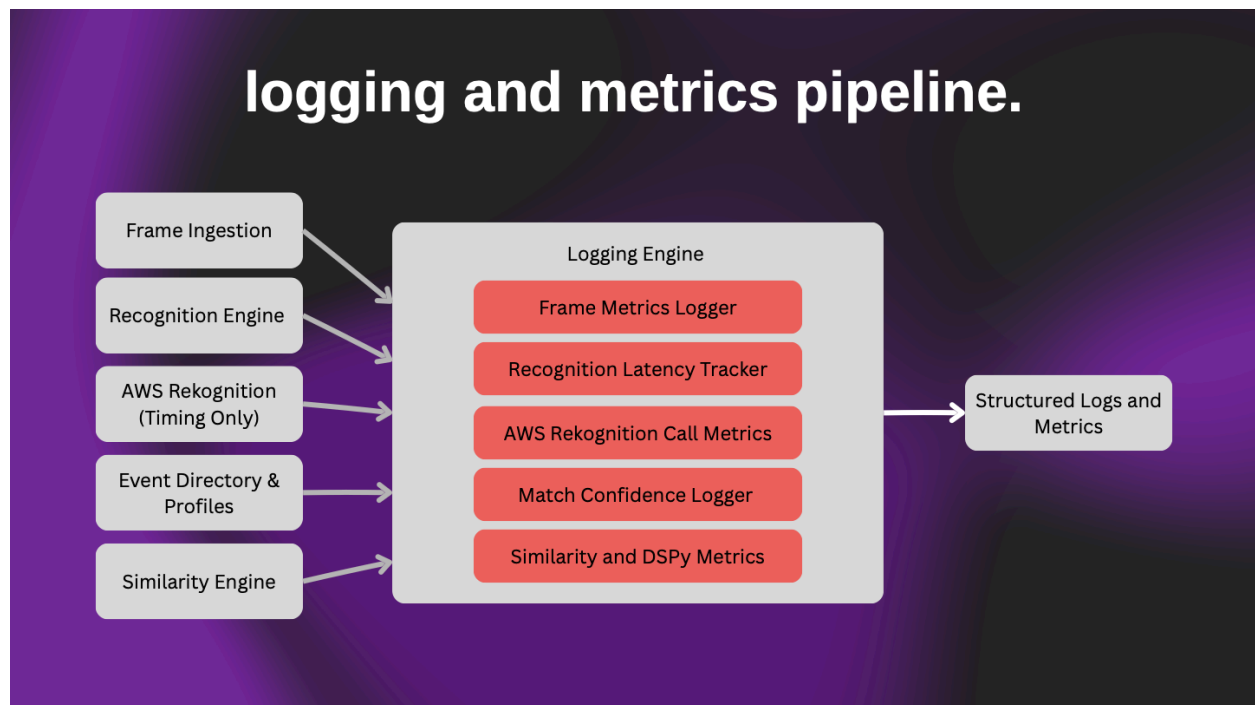


recognition pipeline.



similarity and context engine.





Figma

<https://www.figma.com/design/P7YPY86SRqWdDHq58Y8NHq/UI-UX?node-id=295-199&t=dep3KFpNcqFwZQNj-1>

Design Trade

Each team member completed a design trade study for a distinct subsystem. The following summarizes each trade study, including the design alternatives considered, the evaluation results, and the final design decision.

Image Preprocessing (Pre-Rekognition)

Design alternatives considered:

ID1 – On-Device Fast OpenCV: Image preprocessing (auto-rotation, denoising, resizing, contrast adjustment, and EXIF stripping) is performed directly on the phone before sending the frame to AWS Rekognition. This approach was considered to minimize latency and reduce upload size while preserving privacy.

ID2 – Cloud Preprocess Service: Raw images are sent to a backend server where OpenCV preprocessing is performed before forwarding the cleaned image to Rekognition. This was considered to provide consistent behavior across devices and simplify parameter tuning.

ID3 – Phone–Cloud Hybrid: The phone performs quick filtering and resizing, then sends the best frame to a server for final preprocessing before Rekognition. This was considered to balance speed, consistency, and bandwidth usage.

Option	Latency (50)	Accuracy (30)	Privacy (10)	Resource Mgmt (10)	Total
ID1 – On-Device Fast OpenCV	9 (450)	7 (210)	9 (90)	8 (80)	830
ID2 – Cloud Preprocess Service	6 (300)	8 (240)	7 (70)	6 (60)	670
ID3 – Phone–Cloud Hybrid	8 (400)	8 (240)	8 (80)	7 (70)	790

ID1 was selected because it best satisfies the system’s strict real-time latency requirements by avoiding additional network hops. Performing preprocessing locally also improves privacy by preventing server-side storage of raw frames and enabling immediate EXIF removal. While cloud-based approaches offer easier tuning, their added latency and privacy surface area were unacceptable for a live interaction system. ID1 provides the best balance of speed, simplicity, and recognition accuracy improvement.

Face Recognition Module

Design alternatives considered:

Alternative 1 – Cloud-Based Recognition (AWS Rekognition): Frames are sent to AWS Rekognition for detection and matching against an event-specific collection. This was considered due to its maturity, accuracy, and scalability.

Alternative 2 – On-Device Recognition (Edge AI): Face embedding and matching are performed locally without cloud calls. This was considered to maximize privacy and minimize latency.

Alternative 3 – Hybrid Recognition: A lightweight on-device step filters or detects candidates before sending reduced data to Rekognition for confirmation. This was considered to balance latency, privacy, and accuracy.

Criterion	Weight	Alt 1	Alt 2	Alt 3
Recognition Accuracy	25	9 (225)	7 (175)	9 (225)
Latency	20	7 (140)	10 (200)	9 (180)
Privacy & Security	20	8 (160)	10 (200)	9 (180)
Scalability	15	10 (150)	6 (90)	8 (120)
Implementation Complexity	10	9 (90)	6 (60)	7 (70)
Cost per Inference	10	6 (60)	10 (100)	8 (80)
Total Score	100	825	825	855

The hybrid approach was selected because it provides near cloud-level accuracy while reducing bandwidth usage and exposure of raw image data. Local pre-filtering improves privacy and latency without sacrificing recognition reliability at scale. Fully on-device recognition struggled with accuracy and deployment complexity, while fully cloud-based recognition increased privacy and cost concerns. The hybrid solution offers the strongest overall trade-off for the current system goals.

PostgreSQL Data Layer (Supabase)

Design alternatives considered:

Design A – Single Shared Schema with RLS: All events and users share common tables, with Row-Level Security enforcing consent and event access. This was considered due to its simplicity and fast joins.

Design B – Per-Event Partitioned Schema: Every record includes an event identifier, enforcing event scoping at the schema level in addition to RLS. This was considered to reduce cross-event leakage risk and simplify data retention.

Criterion	Weight	Design A	Design B
Query Latency	30%	8 (2.4)	8 (2.4)
Access Control / Privacy	25%	7 (1.75)	9 (2.25)
Implementation Complexity	15%	9 (1.35)	7 (1.05)
Auditability & Traceability	15%	7 (1.05)	8 (1.20)
Scalability	15%	7 (1.05)	8 (1.20)
Total Score	100%	7.8	8.1

Design B was selected because it enforces privacy and event isolation directly at the data-model level, reducing reliance on complex RLS rules alone. It also supports straightforward event-level data purging to meet retention requirements. Query performance remains comparable due to indexed event-scoped lookups. Although slightly more complex to implement, Design B provides stronger privacy guarantees and better long-term scalability.

Facial Recognition Processing Architecture

Design alternatives considered:

A1 – Fully Cloud-Based Recognition: All recognition is handled in the cloud, minimizing on-device compute and maximizing scalability. This was considered as the most reliable and production-ready option.

A2 – Hybrid Edge–Cloud Recognition: Initial processing occurs on the device, with final matching in the cloud. This was considered to reduce bandwidth and improve privacy.

A3 – Fully Edge-Based Recognition: Recognition runs entirely on-device, with cloud used only for updates. This was considered to maximize privacy and reduce cloud dependence.

Criterion	Weight	A1: Cloud	A2: Hybrid	A3: Edge
Latency	25	8 (200)	9 (225)	6 (150)
Accuracy	20	9 (180)	8 (160)	7 (140)
Privacy & Security	15	6 (90)	8 (120)	10 (150)
Scalability	15	10 (150)	8 (120)	5 (75)
Reliability	10	9 (90)	8 (80)	7 (70)
Energy Efficiency	5	10 (50)	8 (40)	6 (30)
Implementation Complexity	10	9 (90)	7 (70)	5 (50)
Total Score	100	850	815	665

The fully cloud-based architecture was selected because it best meets current performance, reliability, and development timeline constraints. It leverages mature cloud infrastructure to provide high accuracy and scalability while keeping wearable compute requirements minimal. Although cloud processing increases privacy considerations, these are mitigated through explicit consent, event scoping, and audit logging. For the current prototype, this approach provides the most dependable path to meeting all must-have requirements.