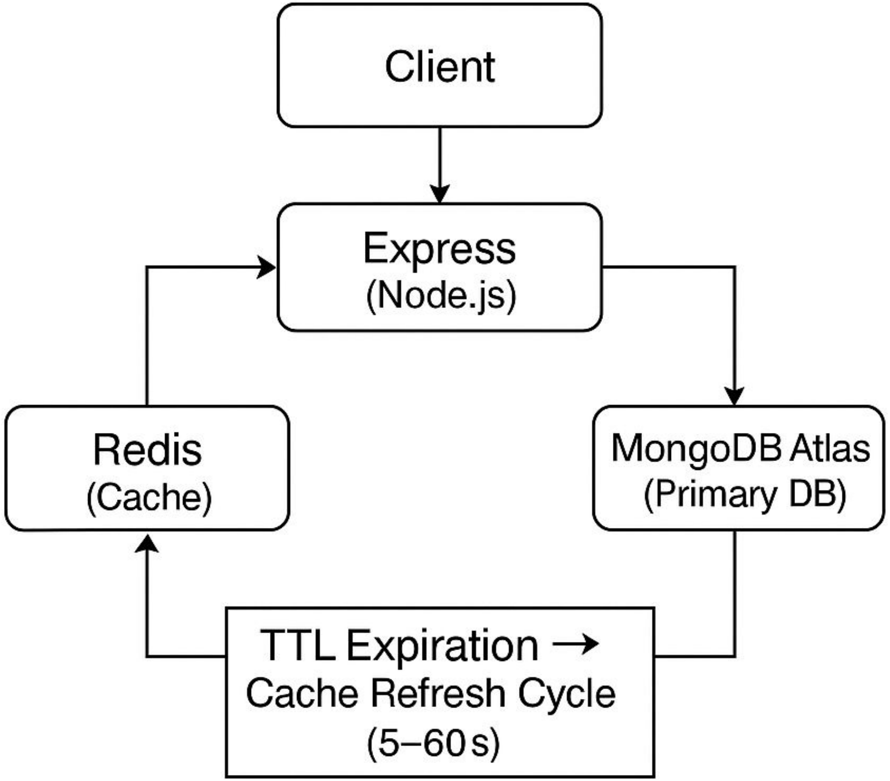
Lab 4

Omkar Prathamesh Punekar

Redis–Mongo Caching pipeline

fig.1



Lessons Learned

Cache-Aside: Reduced latency by fetching from Redis on repeat queries.

Read-Through: Simplified logic—cache layer transparently fetches & stores on miss.

TTL-Based Expiration: Controlled freshness; automatic invalidation.

Performance improved from ~14 ms (cold) → ~1.4 ms (cached).

Demonstrated integration of multiple containers (Redis + Node) via Docker Compose.

Observed smooth connection with MongoDB Atlas after IP whitelisting.

Challenges & Results

Challenges:

Docker + WSL setup and network bridging for Atlas.

Cache invalidation timing & TTL tuning.

Handling JSON serialization/deserialization in Redis.

Understanding when to refresh vs reuse cache.

What Worked Well:

Simple Express endpoints for /cache-aside, /read-through, /ttl.

Bench script automated timing tests (cold, warm, post-expire).

Redis CLI confirmed connection (PONG), and health endpoint verified Mongo+Redis.

Compass verified 2,000 seeded records by meta.author:"Omkar".