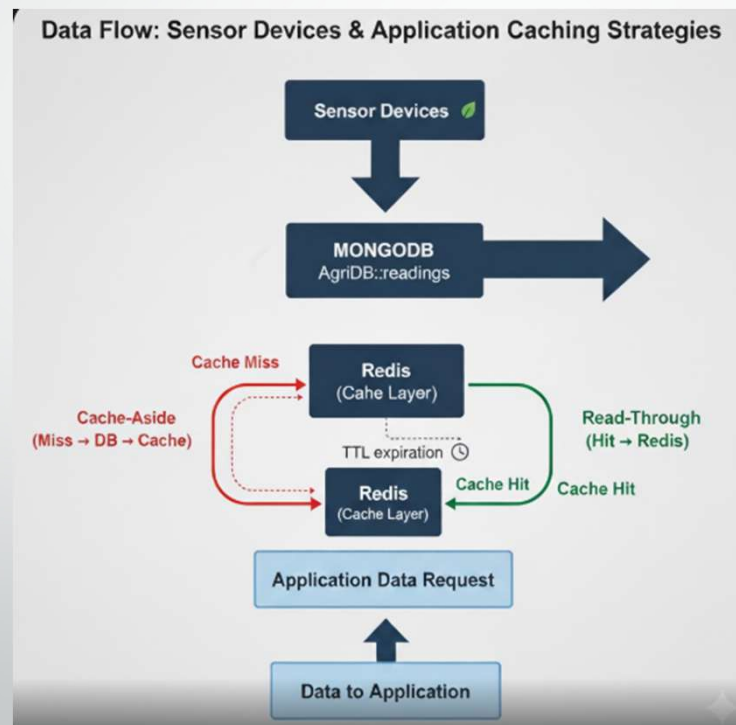# Lab-4

Submitted By: Bhavanand Jupalli

Architecture

Lessons Learnt

- Implementing **Cache-Aside** drastically improved response times on repeated queries.
- **TTL expiration** is useful to keep cached data fresh without manual invalidation.
- Using **composite cache keys** (sensor ID+author) avoids collisions and ensures data integrity
- Monitoring cache hit/miss rates is critical for evaluating performance improvements.

Challenges

- **What worked well:**
- Cache-Aside and Read-Through strategies reduced average response time from ~120ms → ~15ms on cache hits.
- Node.js + Redis integration was straightforward.
- **What did not work well:**
- First-time cache misses still hit the database, so caching is effective only after initial access.
- Managing TTL required careful testing to avoid unexpected reloads.