**What others use?**

#Swiggy

Backend: Java, Scala, Python, Go, Rust, NodeJS

Frontend: ReactJS

App: Kotlin (Android), Swift(iOS)

IAAS: AWS

DB’s: MySQL, Postgres, ScyllaDB

Caches: Redis, Aerospike

Message brokers: Kafka, Rabbitmq

ServiceMesh: Kubernetes ecosystem, Mesos, Consul

Load balancing: Haproxy, Kong

Data platform: S3, Presto, Druid, Snowflake, Flume, Flink, Hive, Spark, Storm

Search: Solr, Elasticsearch

CI/CD: Custom built tool over Jenkins

Monitoring: Thanos (by improbable), NewRelic (APM), Custom analytics systems built in-house

The tech stack is very heterogeneous.

#Zomato

VARNISH

acts as http accelerator (powers our cdn servers)

HAPROXY

load balances our app servers

APACHE/PHP

acts as app servers (we do not use any php framework)

MEMCACHE

caches our query result sets and reduces load to our databases

MYSQL

for persistence (we use MySQL Community Edition in master/slave setup)

SOLR

indexes our content and powers our search

REDIS

Acts as messaging server (pub/sub) and as cache server for social relationships, recommendations and notifications

And managing realtime analytics information

NODEJS

Serving realtime notifications

HYPERTABLE

Used internally for tracking and logging changes to our content

**Efficient way**

* **SEARCH**Full text search with mongodb is possible. Let’s see if it can bit elastic.

Ref link https://docs.mongodb.com/manual/core/link-text-indexes/

I will also try the following

With Mongoosastic, simultaneous indexation to both mongodb and elastic, is possible

<https://code.likeagirl.io/5-different-ways-to-synchronize-data-from-mongodb-to> elasticsearch-d8456b83d44f

I will clone data to elastic specifically for search purpose. Otherwise continue with mongodb full text search.

* **TRANSACTION**I will validate if mongodb can handle transaction. If transaction queries are not flexible I will otherwise move to oracle.

Online tools for mongo db and sql conversion Ref: https://www.site24x7.com/tools/sql-to-mongodb.html

* **SESSION CACHE HANDLING**  
  Redis for cache or session storage is preferred since others use it. I don't still know its real time use. But as the project goes further the familiarity might give me the intuition behind it.

<https://www.objectrocket.com/blog/how-to/top-5-redis-use-cases/>

**Real time scenario**

**User registers**

Data goes to mongo dB

**User logs in**

Data goes to redis. Every time app validates if user session persists by fetching redis.

**Listing of items**

Data fetched from mongo dB

**User posts cook**

Data goes to mongo dB.

**User orders food**

Data goes to mongo dB. Mongo dB is said to be ACID after v4. Let’s search with v4 otherwise Oracle is a better option.

**User searches for food**

Data is fetched from mongo dB using full text search api, if search is not ok, elastic search is an option.