High Level Design

**CAP Theorem**

It states that that a distributed system can deliver only two of three desired characteristics: consistency, availability, and partition tolerance (the ‘C.” ‘A’ and ‘P’ in CAP).

A distributed system is a network that stores data on more than one node (physical or virtual machines) at the same time. Because all cloud applications are distributed systems, it’s essential to understand the CAP theorem when designing a cloud app so that you can choose a data management system that delivers the characteristics your application needs most.

1. **Consistency**: Consistency means that all clients see the same data at the same time, no matter which node they connect to. For this to happen, whenever data is written to one node, it must be instantly forwarded or replicated to all the other nodes in the system before the write is deemed ‘successful.’
2. **Availability**: Availability means that any client making a request for data gets a response, even if one or more nodes are down. Another way to state this—all working nodes in the distributed system return a valid response for any request, without exception.
3. **Partition tolerance**: A partition is a communications break within a distributed system—a lost or temporarily delayed connection between two nodes. Partition tolerance means that the cluster must continue to work despite any number of communication breakdowns between nodes in the system.

Strangler Pattern

Saga Pattern: It is of 2 types: Choreography & Orchestrator

CQRS Pattern

Low Level Design