



# High-level Design of a Distributed Cache

Learn how we can develop a high-level design of a distributed cache.

We'll cover the following



- Requirements
  - Functional
  - Non-functional requirements
- API design
  - Insertion
  - Retrieval
- Design considerations
  - Storage hardware
  - Data structures
  - Cache client
  - Writing policy
  - Eviction policy
- High-level design

In this lesson, we'll learn to design a distributed cache. We'll also discuss the trade-offs and design choices that can occur while we progress in our journey towards developing a solution.

## Requirements



Let us start by understanding the requirements of our solution.



## Functional



The following are the functional requirements:

- **Insert data:** The user of a distributed cache system must be able to insert an entry to the cache.
- **Retrieve data:** The user should be able to retrieve data corresponding to a specific key.

