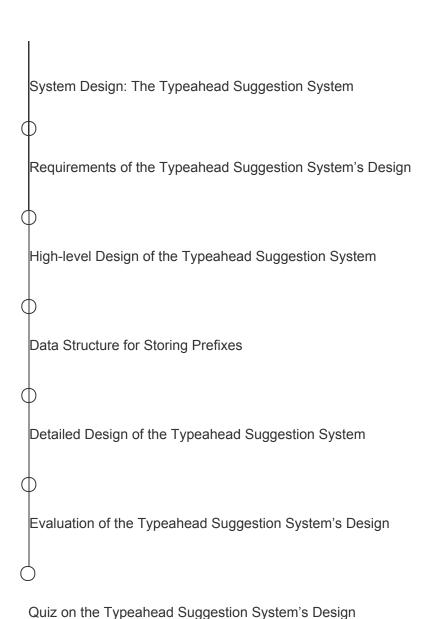
Join Log In
Back To Module Home
Design Problems 0% completed
RESHADED Approach for System Design
Design YouTube
Design Quora
Design Google Maps
Design a Proximity Service / Yelp
Design Uber
Design Twitter
Design Newsfeed System
Design Instagram
Design matagram

Design a URL Shortening Service / TinyURL

Design a Web Crawler

Design WhatsApp

Design Typeahead Suggestion



Design a Collaborative Document Editing Service / Google Docs

Conclusion

Mark Module as Completed

System Design: The Typeahead Suggestion System

Learn about the typeahead suggestion system and understand the important details related to its design process.

We'll cover the following

- Introduction
- How will we design a typeahead suggestion system?

Introduction#

Typeahead suggestion, also referred to as the autocomplete feature, enables users to search for a known and frequently searched query. This feature comes into play when a user types a query in the search box. The typeahead system provides a list of suggestions to complete a query based on the user's search history, the current context of the search, and trending content across different users and regions. Frequently searched queries always appear at the top of the suggestion list. The typeahead system doesn't make the search faster. However, it helps the user form a sentence more quickly. It's an essential part of all search engines that enhances the user experience.

The typeahead suggestion system in action

How will we design a typeahead suggestion system?#

We've divided the chapter on the design of the typeahead suggestion system into six lessons:

- 1. **Requirements:** In this lesson, we focus on the functional and non-functional requirements for designing a typeahead suggestion system. We also discuss resource estimations for the smooth operation of the system.
- 2. **High-level design:** In this lesson, we discuss the high-level design of our version of the typeahead suggestion system. We also discuss some essential APIs used in the design.
- 3. **Data Structure for Storing Prefixes:** In this lesson, we cover an efficient tree data structure called trie that's used to store search prefixes. We also discuss how it can be further optimized to reduce the tree traversal time.
- 4. **Detailed design:** In this lesson, we explain the two main components, the *suggestions* service and the *assembler*, which make up the detailed design of the typeahead suggestion system.
- 5. **Evaluation:** This lesson evaluates the proposed design of the typeahead suggestion system based on the non-functional requirements of the system. It also presents some client-side optimization and personalization that could significantly affect the system's design.
- 6. Quiz: In this lesson, we assess our understanding of the design via a quiz.

T 12 1 1 1	1 1 1.0	1	٠.	C	1	. 1	ı .• .
Let's start h	v identity	ang the re	annrement	s tor	designing	a typeaheac	l suggestion system.
LCC 5 Start 5	y lucifully	mg the re	quiicincin	101	acoisimis	a typeaneac	i buggestion by btem.

Back

Quiz on WhatsApp's Design

Next

Requirements of the Typeahead Sug...

Mark as Completed

Report an Issue