

Requirements Specification - LTU Search Engine

Fullstack Developer .NET

December 29, 2025

1 Functional Requirements

Table 1: Functional Requirements List

RQ ID	Description	Test Method
<i>Crawler (1000 Series)</i>		
FRQ-1001	The system shall crawl web pages starting from a given seed URL.	
FRQ-1002	The system shall follow internal links recursively.	
FRQ-1013	Domain-based crawling and rate limiting. The crawler shall restrict all crawling to domains explicitly listed in a configurable domain whitelist. The crawler shall enforce a configurable maximum number of concurrent HTTP requests per domain (<code>maxConcurrencyPerDomain</code>) and a configurable minimum delay (<code>minDelayMs</code>) between consecutive requests to the same domain. The system shall identify crawling targets by their domain name (e.g., <code>ltu.se</code>) rather than resolved IP addresses to ensure consistent behavior if underlying hosting or IP mappings change.	
FRQ-1014	The crawler shall log the active rate limiting configuration values (<code>maxConcurrencyPerDomain</code> , <code>minDelayMs</code>) at startup for verification and debugging purposes.	
FRQ-1004	The crawler must parse and adhere to the “robots.txt” file of the target domain.	
FRQ-1006	The crawler shall avoid crawling the same URL more than once per execution.	
FRQ-1007	The crawler shall ignore non-relevant resources (images, CSS, JS).	
FRQ-1009	The system must support adding new domains to a whitelist automatically.	
FRQ-1010	The crawler must detect linked PDF files and include them in the index.	
FRQ-1011	The crawler must not follow/crawl links found inside PDF documents.	
<i>Indexing (2000 Series)</i>		
FRQ-2001	The system shall extract textual content from HTML pages.	

Continued on next page

Table 1 – continued from previous page

RQ ID	Description	Test Method
FRQ-2002	The system shall store indexed terms together with page references (inverted index).	
FRQ-2003	The system shall support incremental updates of the index.	
FRQ-2004	The system shall ignore non-textual content (images, videos, binaries).	
<i>Searching (3000 Series)</i>		
FRQ-3001	Query Terms. The system shall support queries composed of terms (e.g., "cats and dogs") and operators (e.g., "cats" AND "dogs").	
FRQ-3002	Single Term Support. The system shall support single-term queries, where a term consists of one word (e.g., test, hello).	
FRQ-3003	Phrase Support. The system shall support phrase queries, defined as multiple words enclosed in double quotation marks (e.g., "hello dolly").	
FRQ-3005	Operator Case Sensitivity. The system shall require Boolean operators to be specified in uppercase letters.	
FRQ-3006	Boolean OR Logic. The system shall support the disjunctive operator to match documents containing at least one specified term. Syntax: 1. Keyword: OR (case-sensitive) 2. Symbol: 3. Implicit: Whitespace between terms implies OR (Default behavior).	
FRQ-3009	Boolean AND Logic. The system shall support the conjunctive operator to match documents containing all specified terms. Syntax: 1. Keyword: AND (case-sensitive) 2. Symbol: &&	
FRQ-3014	Required Term Operator. The system shall support the required (+) operator, indicating that the term must exist in a matching document. (e.g, If the query searches for "+are cats" dog", results must include items containing "are cats" and can but does not have to contain "dog").	
FRQ-3011	Exclusion Logic (NOT). The system shall support operators to exclude documents containing specific terms. Syntax: 1. Keyword: NOT (case-sensitive) 2. Symbol: ! or - Constraint: The exclusion operator must be preceded by a positive term (e.g., "cat -dog"). Standalone exclusion queries shall return an error or zero results.	

Continued on next page

Table 1 – continued from previous page

RQ ID	Description	Test Method
FRQ-3016	<p>Grouping with Parentheses. The system shall support grouping of query expressions using parentheses to control operator precedence.</p> <p>(e.g, If the query searches for ("cat" AND "dog") OR "fish", results should include items containing either both "cat" and "dog", "fish" or both clauses.).</p>	
FRQ-3017	<p>Special Character Escaping. The system shall support escaping of special query characters using a backslash (\).</p> <p>Supported Escapable Characters:</p> <p>+ - && ! () { } [] ^ " ~ * ? : \</p>	
FRQ-3018	Literal Search with Escaping. The system shall correctly interpret escaped characters as literal values within a query.	
FRQ-3019	Results of queries must be paginated when more than 10 results are found.	
FRQ-3020	The results should contain the headline of the context found.	
FRQ-3021	The results should contain a snippet with keywords highlighted.	
FRQ-3022	<p>Search results shall be ranked by relevance using a combination of TF/IDF scores and PageRank.</p> <ul style="list-style-type: none"> • Documents with higher TF/IDF scores for the query terms must appear before documents with lower scores. • When TF/IDF scores are equal, documents with higher PageRank shall appear first. <p>Example: For the query "cat dog", a document containing both "cat" and "dog" with high term frequency and appearing on a highly linked page shall be ranked above a document containing only "cat" or appearing on a low-ranked page.</p>	

User Interface (4000 Series)

FRQ-4001	The system shall allow users to enter search queries and view results.
FRQ-4002	The UI shall indicate the current page (pagination state).
FRQ-4003	When a search query returns zero results, the UI shall display a visible message stating that no results were found (e.g., "No results found")

2 Low-Priority Functional Requirements

RQ ID	Description	Test Method
L-FRQ-5001	The query should be able to handle wildcards.	

L-FRQ-5002	An estimation of the completed query should be suggested (Autocomplete).
------------	--

3 Non-Functional Requirements

RQ ID	Description	Test Method
NFRQ-6001	A query should take no longer than 10 seconds.	
NFRQ-6002	The search engine should search the whole LTU-Domain (with exceptions).	
NFRQ-6003	The system shall only index publicly available HTML pages.	
NFRQ-6004	The system shall provide a web-based search interface.	
NFRQ-6005	The system shall provide a search API for the UI.	
NFRQ-6006	Rate limiting parameters shall be configurable via configuration file or environment variables and shall take effect after restart, without requiring code changes.	

4 Non-Testable RQ

- FRQ-2005 The system shall use a clearly defined data structure (e.g., ER diagram).
- FRQ-3001 A query containing 2-3 words should return the correct result.
- FRQ-3004 Use same syntax as other providers (standard search syntax).