

Test Case Specification

Group 1

January 5, 2026

1 Introduction

This document describes test cases for the LTU Search Engine project.

2 Test Case: TC-FRQ-1009

Related Requirements

FRQ-1009

Description

Verify that the crawler detects hyperlinks that point to PDF files and adds those PDF documents to the index so that they become searchable.

Preconditions

- Crawler component is operational
- Indexing component is operational
- Crawling of external resources (PDF files) is permitted
- At least one HTML page exists that contains a hyperlink to a PDF file
- The search index is initially empty.

Test Steps

1. Configure the crawler with the HTML page as the seed URL.
2. Start the crawling and indexing process.
3. Allow the crawler to fetch the HTML page.
4. Detect and follow the hyperlink to the linked PDF file.
5. Allow the indexing component to process the discovered PDF file.
6. Inspect the contents of the search index.

Expected Results

- The crawler detects the hyperlink pointing to the PDF document.
- The PDF document is scheduled for crawling or downloading.
- The PDF document is added to the index.
- Terms extracted from the PDF document become searchable.
- No errors occur as a result of handling the PDF file.

3 Test Case: TC-FRQ-2001

Related Requirements

FRQ-2001, FRQ-2004

Description

Verify that the system extracts only textual content from HTML pages and ignores all non-textual content during the indexing process.

Preconditions

- An HTML page containing visible text content
- The same page contains images, videos, scripts, and binary files
- Search index is empty

Test Steps

1. Submit the HTML page to the indexing component.
2. Parse the HTML document.
3. Extract all indexable content.
4. Inspect extracted data before storage.
5. Store extracted content in the search index.

Expected Results

- Only visible textual content is extracted.
- Images, videos, scripts, and binary resources are ignored.
- No non-textual data is stored in the search index.
- Extracted text is suitable for term-based indexing.

4 Test Case: TC-FRQ-2002

Related Requirements

FRQ-2002

Description

Verify that indexed terms are stored together with references to the pages in which they appear using an inverted index structure.

Preconditions

- Two or more HTML pages containing overlapping keywords
- Search index is empty

Test Steps

1. Submit all pages to the indexing component.
2. Extract and tokenize textual terms from each page.
3. Normalize extracted terms.
4. Store terms in the search index.
5. Inspect the internal index structure.

Expected Results

- Each unique term is stored exactly once in the index.
- Each term maps to one or more page references (URLs or document IDs).
- Pages containing the same term are associated with that term.
- The index follows the inverted index model.

5 Test Case: TC-FRQ-2003

Related Requirements

FRQ-2003

Description

Verify that the system supports incremental updates of the search index without rebuilding the entire index.

Preconditions

- Existing search index populated with indexed pages
- One existing page is modified or a new page is added

Test Steps

1. Run the indexing process on the initial dataset.
2. Modify an existing page or add a new page.
3. Run the indexing process again.
4. Monitor which pages are re-indexed.
5. Compare the index state before and after the update.

Expected Results

- Only new or modified pages are re-indexed.
- Unchanged indexed pages remain untouched.
- No full index rebuild occurs.
- The index remains consistent and searchable after the update.