Meeting Notes

Project: LTU Search Engineee

Supervisor Meeting: 2026-01-08

Supervisor: Malte Kerl
Attendees: Emma, Cilla, Jean-Paul, Nattarintra, Linnea

Feedback on ER Diagram

- A question was raised regarding how the system represents pages that are no longer available. One possible solution is to simply remove such pages from the database. The solution does not need to be sophisticated, but the group should be able to reason about and motivate the chosen approach.
- The use of a **normalized representation of words** was recommended. At the moment, several different representations of the same word may exist. The purpose and benefit of normalization should be clearly understood and applied.
- Storing a **datetime for last crawled** was confirmed to be a good design choice.
- Clarification was requested regarding the purpose of **HTTP status**. If it is used to indicate removed or redirected pages (e.g. HTTP 302), then it is reasonable to keep it.
- The attribute `priority_integer` can be removed for now, as it is not necessary. Similarly, the `Done` status in the crawl queue can be removed.
- Overall, the ER diagram is considered largely complete.
- For handling concurrency in the crawler, tools such as **Redis** or **Celery** may be used (links were provided by Malte). The group is encouraged to attempt implementing concurrency. However, if it turns out to be too complex, it is acceptable to skip it.

Module Diagram and Architecture

- The module diagram is generally acceptable.
- Authentication should be removed.
- The paginator should be moved either to the database layer or to `Backend.Application`.
- The role and necessity of a `BackgroundService` should be reconsidered and clearly motivated.

Process and Testing

- The documentation repository and the current use of GitHub issues were reviewed and considered appropriate.

- Malte will check with Ulf whether the project should follow SCRUM, and if a backlog is required. It may be useful to define a few initial use cases at the beginning, until development of the search engine is underway.
- The project does not need to strictly follow Test Driven Development (TDD). However, testing is required, and tests should be written continuously alongside development, not postponed until after implementation.

Design and Readiness

- The module diagram should be used as the basis for completing the UML and class diagrams.
- Malte asked whether the group feels ready to start parallel coding. The group expressed that this is not yet the case, as parts of the overall structure and "big picture" are still unclear. This is expected to become clearer once the diagrams are finalized.
- The query syntax must be defined and documented.
- Contact information pages should be crawled at this stage. In the future, a solution for overriding `robots.txt` for such pages will be needed.

Next Steps

- Finalize UML and class diagrams and send them to Malte.
- Prepare for the next supervisor meeting on **Thursday, January 15 at 10:00**.

Meeting: 2026-01-08

Attendees: Cilla, Emma, Jean-Paul, Nattarintra, Linnea

Topics Discussed

- The group discussed questions to be raised with the supervisor regarding:
  - Whether the project should follow Test Driven Development (TDD).
  - If TDD would imply additional workload by writing tests before implementation and then iterating.
  - The need for clarification on what TDD means in practice for this project.
  - Whether contact information pages should be included in the crawling scope.
  - With respect to FRQ-1003, whether the crawler should support concurrent crawling.
  - If concurrency increases complexity or execution time.
  - The consequences of not using concurrency.
- It was noted that the group should organise the work in a more structured way, including clearer role and task distribution.
- The documentation is not yet complete. There are still test cases that need to be written for several requirements.
- On GitHub, issues exist for most backend-related tasks, while no issues have yet been created for frontend work.

Task Distribution

- Jean-Paul will continue writing test cases for crawler-related requirements (RQ 1000-series).
- Cilla has also started working on crawler test cases and will continue contributing in that area.
- Nattarintra will begin working on requirements and test cases in the RQ 3000-series.
- Emma, Cilla, and Linnea will conduct a short walkthrough of the development environment in VS Code to investigate and resolve issues encountered when running the program.

Next Steps

- Model the crawler and incorporate feedback from the supervisor meeting later the same day (15:00).
- Work towards clearer role definitions and task distribution within the group.

Meeting: 2026-01-05

Attendees:
Emma, Cilla, Jean-Paul, Nattarintra

Agenda

1. What was done during the holidays
2. Divide work between members.

Discussion and Notes

We discussed what we have been doing during holidays and how we should continue with the project.

Action Items

- **JP:** Summary of diagram
- **Emma:** Write testcase requirements
- **Cilla:** Write testcase requirements
- **Nattarintra:** Write testcase requirements

Meeting: 2025-12-19

Attendees:
Emma, Linnea, Jean-Paul

Agenda

1. What was done the day before.
2. Discuss reporting.
3. Divide work between members.

Discussion and Notes

Everyone present mentioned what they have done they day before. We discussed reporting and decided to send our current activity report in conjunction with a small "to do next week" list.

Action Items

- **JP:** Review meeting notes PR, Tests
- **Emma:** What to include in an index, Review Query syntax PR
- **Linnea:** How to limit crawler rate limit
- **Cilla:** Investigate different algorithms for searching
- **Nattarintra:** Test

Meeting: 2025-12-18

Attendees:
Emma, Cilla, Linnea, Jean-Paul, Nattarintra

Agenda

1. Divide work between members

Discussion and Notes

We focused on dividing the tasks for the search engine project. Everyone has been assigned a specific area to research or implement.

Action Items

- **JP:** Query syntax
- **Emma:** What to include in an index, upload meetingnotes to Git Repository
- **Linnea:** How to limit crawler rate limit
- **Cilla:** Investigate different algorithms for searching
- **Nattarintra:** Test

Meeting: 2025-12-17

Context: Updated directives from Malte (replaces parts of original spec).

Attendees:
Emma, Cilla, Linnea, Jean-Paul, Nattarintra

1. Technical Requirements (Scope)

- **Database:**
  - Free choice (Postgres, MariaDB, SQLite are all okay).
  - *Important:* Avoid MS SQL Server (due to Linux compatibility issues). Choose something easy to install/run.
  - Storage: 512 GB is available on the VM, but we likely only need a few hundred MB.
- **Web Crawler Behavior:**
  - **Whitelist:** Strict. Only crawl `ltu.se` and specific links provided. Do **not** follow external links (e.g., Facebook, Instagram).
  - **Robots.txt:** Generally respect it, *unless* it blocks "contacts" or "users". We must index contact pages so staff can be found.
  - **Content:** Skip images to reduce complexity. Focus on text.
  - **Rate Limiting:** No hard limit, but "don't DDoS". A few hundred requests per minute is acceptable.
- **Search Algorithm & Ranking:**
  - Use a combination of **PageRank** (link popularity) and **Word Frequency**.
  - We are encouraged to experiment with algorithms to find the best results.

2. Documentation & Requirements (Milestone 0)

- **Requirements Document:**
  - Add a **third column** for "Test Cases". (Be specific: How do we prove the requirement is met?)
  - Remove "Test Coverage" from the requirements doc (keep it in code, but not as a functional requirement).
- **Diagrams:** Create initial drafts for UML Diagrams (e.g. a Class Diagram), ER-Diagram (Database). Share these early ("Share everything").
- **Tools:** Documentation in LaTeX, Code in Git.

3. Project Structure & Team Roles

Total team size: 5 members.

- **1 Lead Developer** (Responsible for overall architecture).
- **2 Frontend Developers**.
- **2 Backend Developers**.

Workflow: Use Code Reviews and Pull Requests (PR). Clear separation of tasks between Frontend and Backend.

4. Reporting & Next Steps

- **Weekly Report:** Send an email **every Friday**.
  - Content: "Activity report" (what we did) and "Next to-dos" (what we will do).
  - *Note:* Send one immediately before the holidays.
- **Next Meeting:** January 8th at 15:00.

Meeting: 2025-12-16

Attendees:
Emma, Cilla, Linnea, Jean-Paul, Nattarintra

Agenda

1. Meeting for preparation

Discussion and Notes

We had a shorter meeting to prepare questions regarding tomorrows meeting with Malte.

Meeting: 2025-12-15

Attendees:
Emma, Cilla, Linnea, Jean-Paul, Nattarintra

Agenda

1. Meeting for preparation

Discussion and Notes

We had a shorter meeting to prepare questions regarding tomorrows meeting with Malte.

Meeting: 2025-12-11

Attendees:
Emma, Cilla, Linnea, Jean-Paul, Nattarintra

Agenda

1. Daily meeting we discussed what to focus on, Jean-Paul set up a Req document and we started to fill it up whit Test and priorities them.

Discussion and Notes

We had a shorter meeting to prepare questions regarding tomorrows meeting with Malte.

Meeting: 2025-12-10 (Project Kick-off)

Attendees: Team and Malte (Client)

Agenda

1. Review of project scope and limitations
2. Technical requirements clarification
3. Expectations for Milestone 0

Decisions & Directives (Summary)

1. Scope & Crawler Behavior

- **Whitelist:** Strictly `*.ltu.se`. Do not follow external links (Facebook, etc.). Future domains like `islab.se` might be added later.
- **Robots.txt:** strictly adhere to it for Milestone 0. (Exceptions for contact pages might be added later).
- **Dynamic Content:** LTU's dynamic content is server-side rendered, so no "headless browser" is required.
- **PDFs:** Crawler must detect PDF links. Indexing the *content* of PDFs is optional (nice-to-have).

2. Search Functionality

- **Queries:** Optimized for simple queries (2–3 words). Ranking priority: All terms > Some terms > One term.
- **Pagination:** Required (e.g., 10, 20 results per page).
- **Performance:** Response time should be under 10 seconds.
- **Optional features:** Wildcards (*), category filters, and boolean operators are *not* mandatory.

3. Technology & Deployment

- **UI:** Plain HTML/CSS/JS is perfectly acceptable. No complex frameworks (React/Vue) needed unless desired.
- **Deployment:** Local execution is enough. No server deployment required yet.

Deliverables for Milestone 0

- **Requirements Document:** Must include Stakeholder requirements, Functional requirements, and Non-functional requirements.
- **Diagrams:** UML Class diagram + ER Diagram (Database).

Meeting: 2025-12-09

Attendees:
Emma, Cilla, Linnea, Jean-Paul, Nattarintra

Agenda

1. Get to know eachother
2. Send a mail to Malte
3. Set up a Github project

Discussion and Notes

We had a presentation round and talked about the project and sent a email to Malte for scheduling a meeting. Finally we set up a Github Project