

EI-Logger

Fundamentals

Search engines are the **first destination** for people to gain knowledge.

Searchers' **needs transcend** the old **information retrieval** paradigm and **promote exploratory search** that blends fact-finding in the context of learning exploration.

Search engines should strive to fulfil searcher's needs. **Evaluation** of exploratory search systems is a stepstone to achieve that.

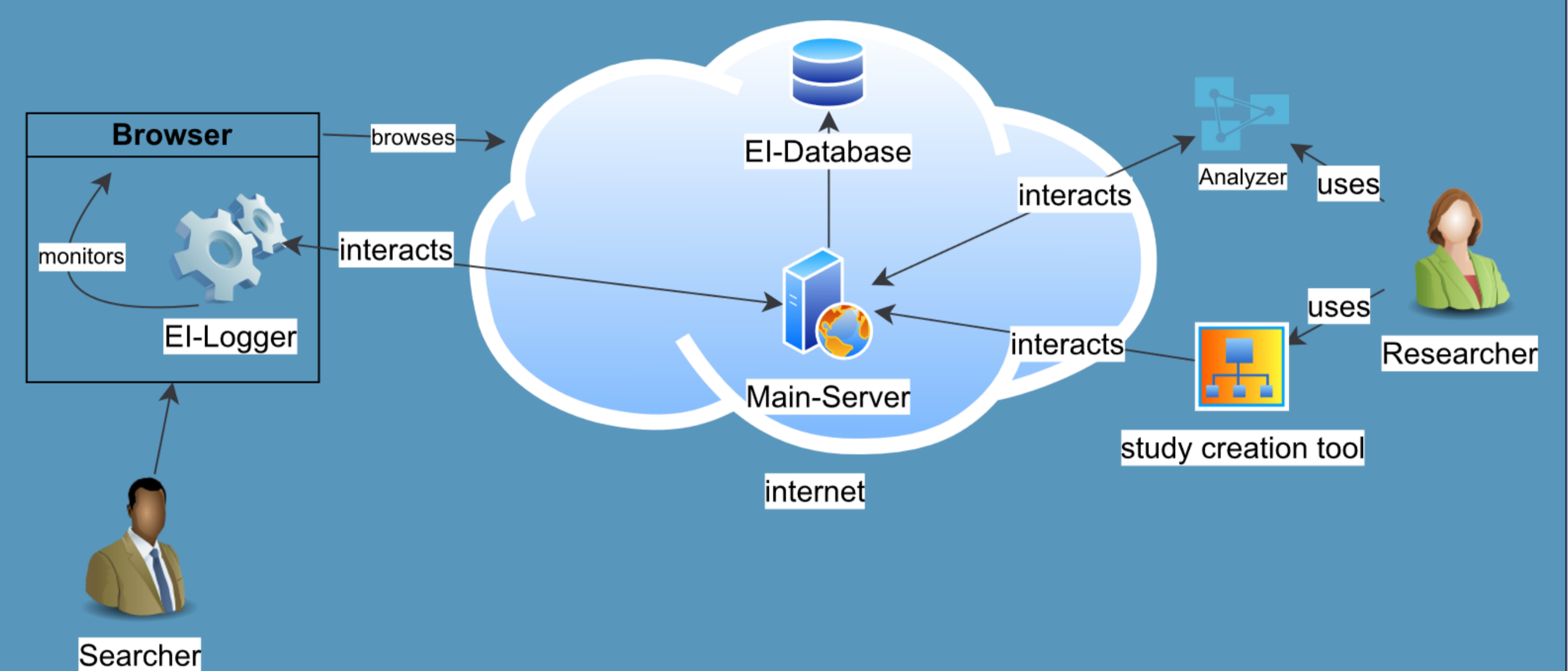
Evaluation tools must consider the **searcher** as an **integral part** of the **search process** and gather **implicit and explicit** searcher information.

Data gathered in the **background** are **Implicit** data.
Data gathered as direct **feedback** are **Explicit** data.

Design & Architecture

EI-Logger is a **browser extension** designed to be **flexible** and with the **exploratory search in mind**.

Built around the definition of the **exploratory search task** being an **open-ended, abstract** and **poorly defined information need** with a **multifaceted character**.



Background

Few tools have been developed to evaluate exploratory search systems.

Most existing tools are **not maintained** or **updated**, and some are **not even publicly accessible**.

Accessible tools either **exclude** the **searcher** from the evaluation context or **exclude their explicit input**.

Objectives

The objective is to build a system with the following properties:

- Modern, Modular, Intuitive, and Versatile.
- Puts focus on the searcher.
- Logs explicit and implicit data.
- Able to provide a precompiled set of search tasks.
- Able to provide a precompiled set of pre-, and/or post-questionnaires.
- Able to guarantee anonymity.
- Logs job title, gender, and date of birth.
- Logs user-browser interactions.
- Logs user-system interactions.
- Logs query, search engine, and SERP.
- Easily distributable.
- Clean code & architecture.

Implementation & Future work

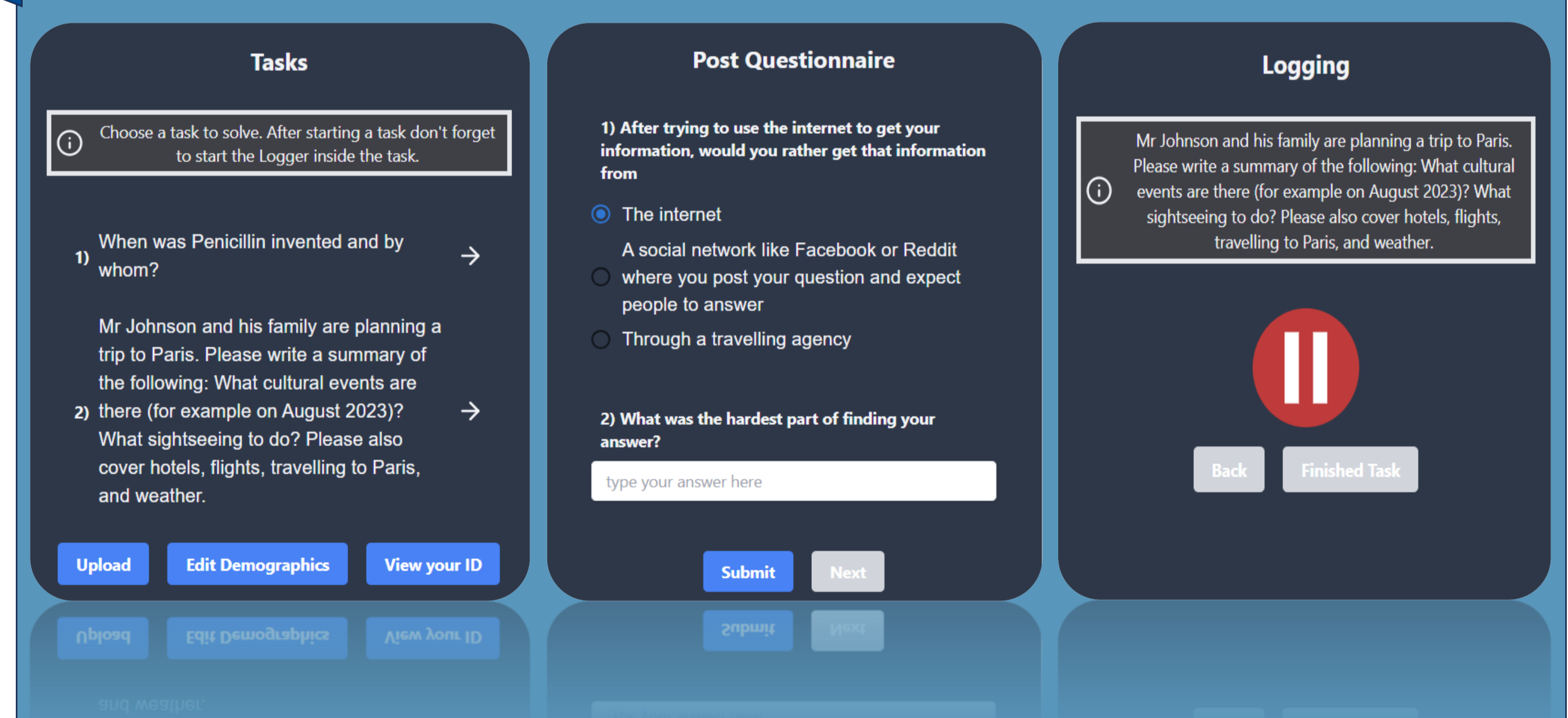
Implementation:

- **Typescript**: guarantees code safety and robustness.
- **React**: a modern and widespread frontend framework.
- **DexieDB**: offers a simple interface for the local browser IndexedDB.
- **GitHub Repository**: has the code for the extension.
- **Jira board**: has stories and tasks.

Future Work:

- Add RAT integration or its own backend.
- Log the index of the clicked search result item.
- Exclude information from the SERP Log that leads back to the searcher.
- Prevent logging on private or user-chosen Websites.

Screenshots



References

- Georg Singer, Ulrich Norbisch, Eero Vainikko, Hannu Kikkas, and Dirk Lewandowski. (2011). Search-logger analyzing exploratory search tasks. In Proceedings of the 2011 ACM Symposium on Applied Computing (SAC '11). 751–756.

- Gary Marchionini. (2006). Exploratory search: from finding to understanding. Commun. ACM. 49. 41-46.



GitHub

UNIVERSITÄT
DUISBURG
ESSEN

Open-Minded