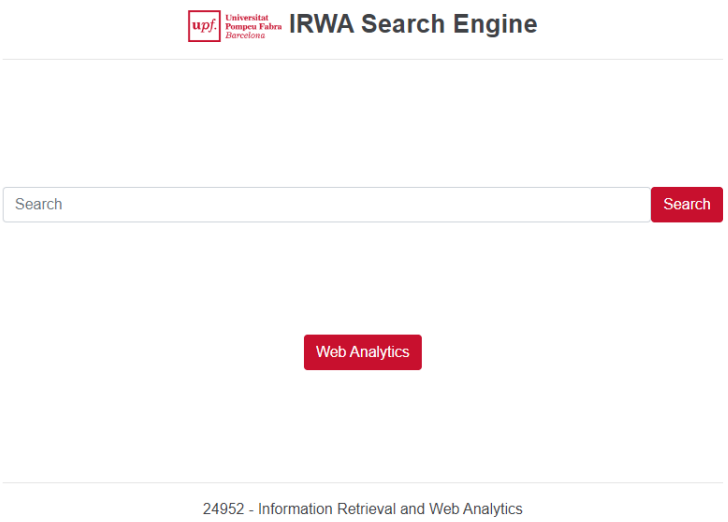
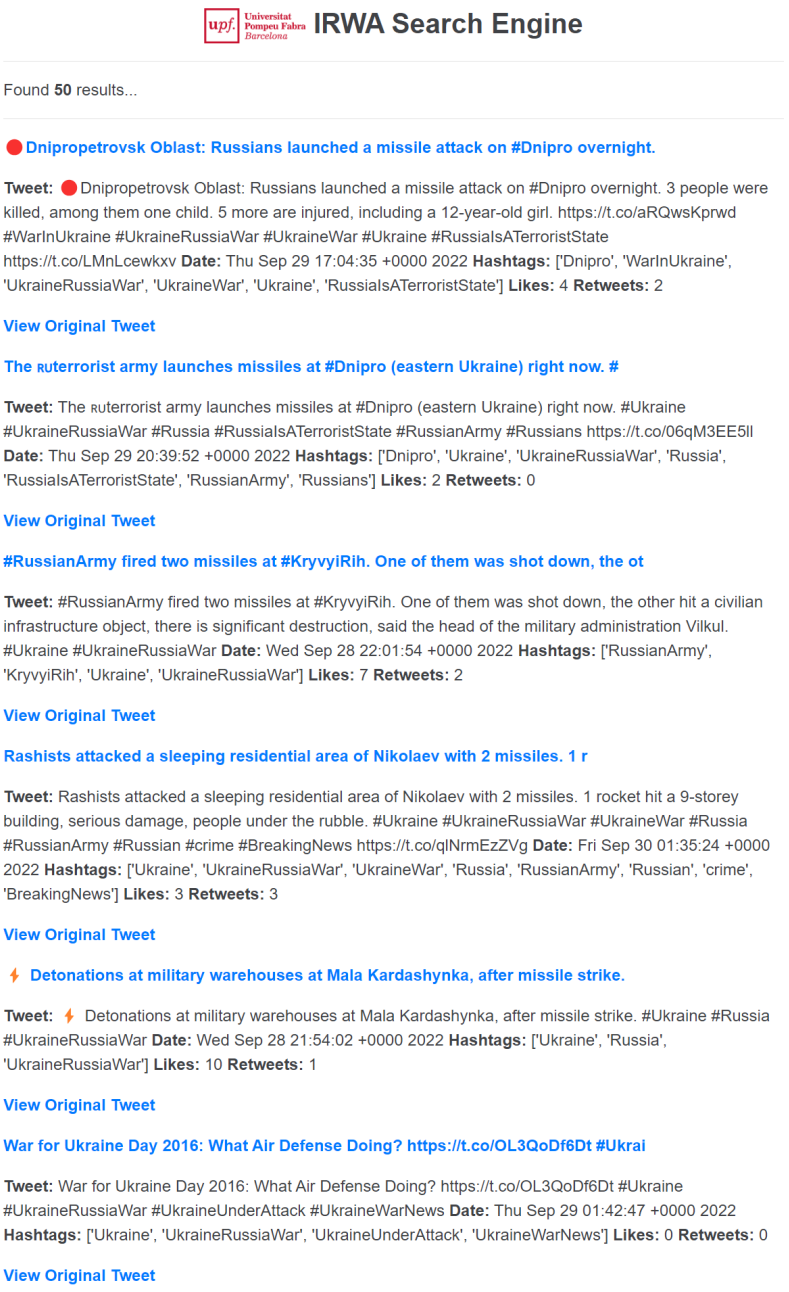
Report Lab 4

Guillem Escriba Molto - David Pérez Carrasco - Nicolás Vila Alarcón

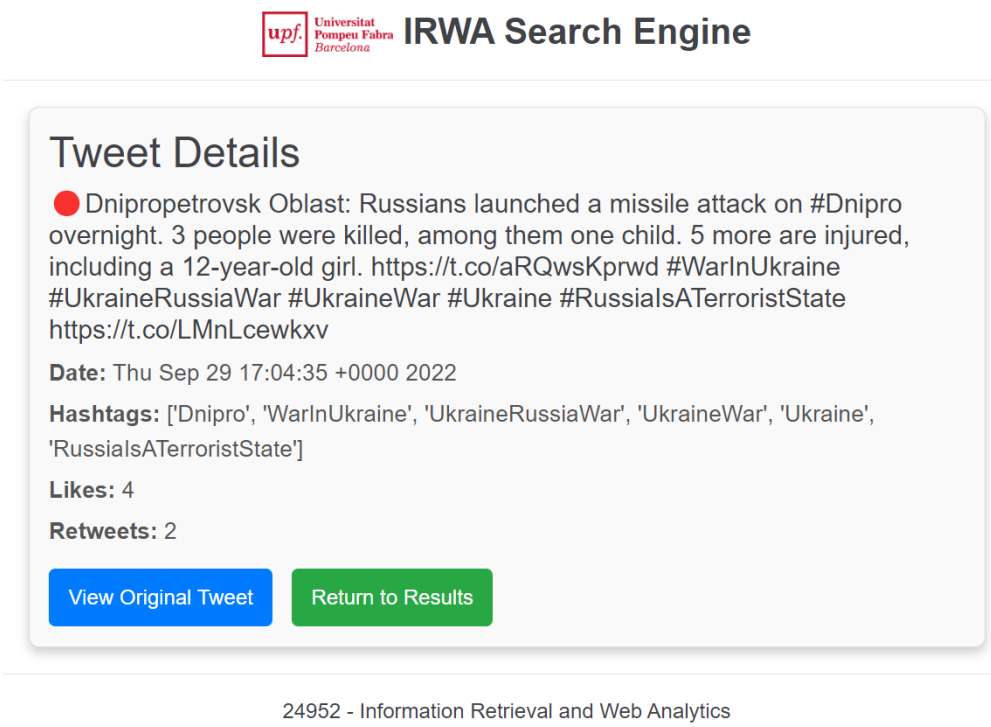
The UI of the search engine was crafted to provide a simple and efficient user experience. The search interface was designed with clarity in mind, featuring a straightforward layout with a prominent search bar, enabling users to enter their queries with minimal distraction. This design choice was motivated by the need to create an environment where users could focus on their search tasks without unnecessary complexity or clutter.



In displaying search results, attention was paid to both aesthetics and functionality. Each result was presented in a clean and organized manner, making it easy for users to scan through the information and identify relevant results. The use of well-defined sections, clear typography, and spacing contributed to the readability and overall user-friendliness of the results page. This approach not only enhanced the visual appeal of the application but also improved usability, a key factor in user satisfaction.



Moreover, the integration of click tracking into the UI was handled seamlessly. Clickable elements in the search results were designed to be intuitive, encouraging user interaction while simultaneously facilitating the collection of valuable click data. This feature was subtly incorporated into the UI, ensuring that it did not disrupt the user experience or detract from the primary functionality of the search engine.



The addition of an analytics dashboard further exemplified the application's user-centric design. The dashboard was created to be informative and easy to navigate, displaying key metrics and analytics in a clear and concise format. The use of graphs and visualizations helped in presenting complex data in an understandable way, making it accessible even to users who might not be technically inclined.

In summary, the user interface of the Flask web application was developed with a keen focus on user experience. By combining a clean and straightforward design with functional features, the UI effectively supported the application's core functionalities while ensuring a pleasant and engaging experience for users. The thoughtful design of the UI, in conjunction with the robust backend development, resulted in a cohesive and user-friendly application that effectively met the project's educational and functional objectives.

For data collection and storage, an in-memory storage mechanism was chosen. This decision was made to simplify the replication process bypassing the complexities associated with setting up and managing a database. Custom data models for Sessions, Clicks, and Requests

were developed within the Python and Flask framework, reflecting a preference for a straightforward and accessible development environment conducive to rapid prototyping. While this approach was ideal for demonstration and educational purposes, it was recognized that a persistent database solution would be more appropriate for real-world applications where data durability and scalability are crucial.

Session management was enhanced to ensure accurate and reliable user tracking. This involved creating unique session IDs for each user visit and implementing logic to prevent the creation of duplicate session entries for the same user. The use of UUIDs guarantees the uniqueness of each session, which is critical for tracking user activities accurately. The rationale behind this was to avoid the inflation of user session counts and ensure that the web analytics data reflected genuine user interactions.

Click tracking was integrated to provide detailed insights into user interactions with the search results. This involved recording each click on a search result, capturing associated data such as the query, the rank of the clicked document, and generating a unique identifier for the click event. The search results template was updated to route user clicks through a dedicated Flask endpoint, which handled the recording of click data before redirecting to a detailed view page. This feature is crucial for understanding user preferences and the effectiveness of the search results. The decision to acknowledge the complexity of accurately calculating dwell time, an important metric in web analytics, was a nod to the project's educational context and the need for more advanced techniques that blend server-side and client-side programming.

In conclusion, the development choices made in the project were carefully considered to provide a balance between educational value, ease of implementation, and the potential for offering meaningful insights into user behavior and application performance. The result was a functional Flask web application prototype, equipped with fundamental web analytics capabilities, and serving as a solid foundation for more advanced developments.

The selection of metrics and the motivation behind data collection in the Flask web application were grounded in the objectives of enhancing user experience, improving search functionality, and providing meaningful insights into user behavior.

Importance of Chosen Metrics

Session Data (User IP, Location, User Agent): Collecting session data is fundamental for understanding user demographics and behavior. Information like user IP, geographical location, and user agent provides insights into the diversity of the user base, their geographical distribution, and the devices or browsers used. This data is crucial for optimizing the application for different user segments and ensuring compatibility across various platforms.

Request Data (Queries, Timestamps): Tracking search queries and their timestamps enables the analysis of search trends, popular search terms, and user engagement over time. Understanding what users are searching for is vital for refining the search algorithm and ensuring that it aligns with user needs and expectations. Timestamp data helps in analyzing peak usage times, which can inform infrastructure scaling decisions and maintenance scheduling.

Click Data (Document Clicks, Related Queries, Ranking): Monitoring which documents are clicked, the associated queries, and their rankings in the search results yields valuable insights into the effectiveness of the search algorithm. It helps in assessing whether the most relevant and useful results are being presented to the users. This data is also instrumental in improving search result ranking algorithms.

In summary, the chosen metrics and the rationale for data collection were driven by a comprehensive approach to understand and enhance user interactions with the application. The ultimate goal was to create a more effective, user-friendly, and optimized search engine, capable of meeting the diverse needs of its user base.