Document Download Toggle Amendments

This document outlines the implementation of a new feature that allows administrators to toggle document download functionality on and off in the Danswer application. The feature was developed by modifying three key files SearchResulstDisplay.tsx, page.tsx, chat_backend.py across the frontend and backend.

Server Implemented on:
(**http://ec2-54-79-231-211.ap-southeast-2.compute.amazonaws.com**)

(**Site: http://54.79.231.211**)

1) (chat_backend) Backend - File Download Management:

Path: cd danswer/backend/danswer/server/query_and_chat/chat_backend.py:

Summary: The **backend manages** the **status** of the **download functionality**. It serves **API endpoints** for **toggling download state**, checking the **current status**, and handling file **download requests**.

❖ Key Modifications:

- **Status Management:** The **backend** is **responsible** for managing the **download state**. It **reads** and **writes** the **status** to a **file** (**flag.json**) **located** at **/app/danswer/server/documents/flag.json**. This **status** is checked before **file downloads**.

- **Download Endpoint:** The **/download/{uuid}/{file_name} endpoint serves** the file to the **user**, but only if **downloads** are **enabled**. If **downloads** are **disabled**, a **403 error** is **returned**, **preventing** access.

- **Toggle Endpoint:** The **/admin/downloads-toggle endpoint** allows the **admin** to change the **download status**. It updates the **status** in **flag.json** based on the **admin's input**.

- **Error Handling:** The **backend includes error** handling for **scenarios** such as **missing files**, **read/write failures**, and **invalid status configurations**. Appropriate **HTTP responses** (eg. **403** for **disabled downloads** or **404** for **missing files**) ensure clear communication with the **frontend**.

2) (page.tsx) Frontend - Admin Panel Download Toggle:

Path: cd danswer/web/src/app/admin/indexing/status/page.tsx

Summary: The **admin panel** provides a **toggle button** to **enable** or **disable** the **document download** functionality. The **toggle state** is **fetched** from and **updated** in the **backend**.

❖ **Key Modifications:**

● **Toggle Button:** A **new toggle button** has been **added** to the **admin panel**, **allowing** the **admin** to **enable** or **disable document downloads**. This button **dynamically updates** based on the **current state** fetched from the **backend** using **/api/chat/download/status.**

● **Backend Communication:** The **admin toggle calls** the **/api/manage/admin/downloads-toggle API** to **update** the **download status**. **Toast notifications provide feedback** to the admin, indicating whether **downloads** have been **enabled** or **disabled**.

● **State Handling:** The **download status** is stored in a **local state** within the **Main components**, and the **toggle reflects** the **current state dynamically**.

3) (SearchResultsDisplay.tsx) Frontend - Search Results Display:

Path: cd danswer/web/src/components/search/SearchResultsDisplay.tsx

Summary: The **document download functionality** in the **search results** is **governed** by the **backend's status**. If **downloads** are **enabled**, users can **download documents** by **clicking** on them. If **disabled**, an **error message** is shown.

❖ **Key Modifications:**

● **Download Button Logic:** The **frontend** now **checks** whether **downloads** are **enabled** before **allowing** users to **download documents**. It calls the **API endpoint /api/chat/download/status** to fetch the **current download status**.

● **File Download Mechanism:** If **downloads** are **enabled**, the file is **fetched** from the **backend**, and the **file-saver package** is used to **save** the **file locally**. If **downloads** are **disabled**, a **toast notification informs** the **user** that **downloads** are **currently unavailable**.

- **Toast Notification: Success** and **error messages** are **displayed** to the **user** during the **download process** to enhance **user experience**.

- ❖ The **code** was took from **this server** and was **modified** with **additional key features** (ubuntu@ec2-13-236-50-121.ap-southeast-2.compute.amazonaws.com) **Implementing** on our **testing server** and **successful achieving** it on (http://ec2-54-79-231-211.ap-southeast-2.compute.amazonaws.com).

5) <mark>Overall Summary:</mark>

- The file chat_backend.py (Backend Control) serves as the central authority, managing the status of the download feature. It controls the file download functionality and communicates the state to both the admin panel and search results display.

- The (Admin Panel) page.tsx fetches the current status from the backend chat_backend.py and allows the administrator to toggle the download functionality. The state is stored in a JSON file, which is used by the backend to manage document download permission.

- The (Frontend Display) SearchResultsDisplay.tsx checks the current download status, fetched from the backend,before allowing users to download documents. If the download feature is disabled, users are blocked from downloading any file, and appropriate notifications are shown.

<mark>Note: For more detailed information, please refer to our other document titled "Document Download Function Amendments".</mark>

<mark>To access the Danswer application as an admin, please log in as:</mark>
<mark>Email: noelshallum@gmail.com</mark>
<mark>Password: 12345</mark>

- **Ibrahim Sultan**