### **Common Utilities (common\_utils.py) Analysis**

This table breaks down the utility functions used across the Streamlit application for common tasks like validation, file management, and user notifications.

| **Function/Method Name** | **Location of the API/Function** | **Input Parameters** | **Output [Datatype]** | **Comments** |
| --- | --- | --- | --- | --- |
| check\_field\_validation | client\_utils.py | field\_name: str, field\_value: str, is\_mandatory: bool | bool | A validation helper that returns True if a mandatory field is empty, signaling that a warning should be shown. |
| show\_field\_warning | client\_utils.py | field\_name: str | None | A UI helper that displays a formatted HTML warning message in Streamlit for a mandatory field that has failed validation. |
| normalize\_url | client\_utils.py | url: str | str | A data cleaning utility that ensures a URL string starts with http:// or https://. |
| save\_uploaded\_file\_and\_get\_path | client\_utils.py | uploaded\_file, logger, client\_enterprise\_name: str | str (file path) | Saves a file uploaded via Streamlit to a specific directory named after the client. **Note: This function appears three times with slight variations; this is the most complete version.** |
| get\_urls\_list | client\_utils.py | company\_name: str | List[str] | An asynchronous wrapper that calls the get\_urls agent to fetch a list of website URLs for a given company name. |
| get\_roles\_list | client\_utils.py | None | List[str] | Returns a hardcoded, static list of common executive job titles (CEO, CTO, etc.). |
| get\_priority\_suggestions | client\_utils.py | None | List[dict] | Returns a hardcoded, static list of business priority suggestions, each with a title, description, and icon. |
| get\_editable\_content | client\_utils.py | None | str | Returns a hardcoded, static string of sample text content. |
| get\_pain\_items | client\_utils.py | file, company\_name: str | (Depends on get\_pain\_points) | A wrapper function that calls the get\_pain\_points function from the pain\_points\_extractor module to analyze a file. |
| save\_uploaded\_file | client\_utils.py | uploaded\_file, save\_dir: str | str (file path) | A simplified file-saving function that saves an uploaded file to a specified directory. |
| clear\_global\_error | client\_utils.py | None | None | A state management utility that removes the global\_message and showing\_dialog keys from st.session\_state to hide dialogs. |
| is\_message\_expired | client\_utils.py | None | bool | Checks the timestamp on a global message in the session state to see if its display duration has passed. |
| show\_error\_dialog | client\_utils.py | message\_text: str, various optional params for buttons and functions | None | A UI function decorated with @st.dialog that displays a standardized error message box with configurable buttons and callback functions. |
| show\_warning\_dialog | client\_utils.py | message\_text: str, various optional params | None | A UI function decorated with @st.dialog that displays a standardized warning message box. |
| show\_success\_dialog | client\_utils.py | message\_text: str, various optional params | None | A UI function decorated with @st.dialog that displays a standardized success message box. |
| set\_global\_message | client\_utils.py | message\_text: str, message\_type: str, various optional params | None | A primary helper function that orchestrates showing a dialog. It checks if a dialog is already active and then calls the appropriate show\_\*\_dialog function. |
| display\_global\_message | client\_utils.py | None | None | Reads message details from st.session\_state and calls the appropriate show\_\*\_dialog function to display it. (Seems to have overlapping purpose with set\_global\_message). |
| get\_scraped\_data | client\_utils.py | url: str, scrape\_technique: str | (Depends on scraper) | A wrapper that calls one of two different web scraping functions (bs4 or googleai) based on the scrape\_technique parameter. |