### **Seller Tab (Seller/) Script Analysis**

This table provides a comprehensive breakdown of the SellerTabState dataclass and all the functions responsible for rendering the UI, handling user interactions, and managing the state for the "Seller" information tab.

| **Class/Scope** | **Function/Method Name** | **Input Parameters** | **Output [Datatype]** | **Comments** |
| --- | --- | --- | --- | --- |
| **SellerTabState** | (Dataclass) | Various fields (e.g., seller\_enterprise\_name: str) | SellerTabState instance | The central state management object for the entire tab. It holds all data related to the seller, from their company name and website to the content of uploaded documents and selected services. |
| **SellerTabState** | to\_session\_state / from\_session\_state | self / cls | None / SellerTabState | Core methods for persisting the tab's data in Streamlit's session state, allowing information to be saved across user interactions and tab switches. |
| **SellerTabState** | update\_field | self, field\_name: str, value: Any | None | Updates a single field in the state and syncs it to the session. |
| **SellerTabState** | add\_processed\_file | self, file\_key: str, filename: str, services: dict, file\_path: str | None | A specialized method to add the results of a document analysis to the state, updating the list of extracted services. |
| **SellerTabState** | validate\_required\_fields | self | Dict[str, bool] | Checks if the mandatory fields for the seller (name, website, details) have been filled out. |
| **Main UI** | seller\_tab | is\_locked: bool | SellerTabState instance | **The main entry point for the tab.** It initializes the state, applies all necessary CSS, and calls the various \_render\_\* functions to build the complete user interface. |
| **Utility** | save\_uploaded\_file\_and\_get\_path | uploaded\_file | str or None | A utility function that saves a file uploaded by the user into a predefined directory (SELLER\_DOCUMENTS\_DIR) and returns the path. |
| **UI Rendering** | \_render\_top\_section | seller\_state, is\_locked: bool, seller\_name\_provided: bool | None | A layout function that creates a two-column layout and places the seller name input and website URL sections within it. |
| **UI Rendering** | \_render\_seller\_name\_input | seller\_state, is\_locked: bool | None | Renders the text input for the seller's company name and the associated "Find Website" button. |
| **Event Handling** | \_handle\_find\_urls\_button | seller\_state, seller\_enterprise\_name: str, is\_locked: bool | None | Contains the logic for the "Find Website" button. When clicked, it sets processing flags in the state and triggers a rerun to initiate the URL search. |
| **Event Handling** | \_display\_url\_search\_status | seller\_state, seller\_enterprise\_name: str | None | This function runs during a rerun triggered by the find URLs button. It displays a spinner, calls the external get\_urls\_list function, and updates the state with the results. |
| **UI Rendering** | \_render\_website\_url\_section | seller\_state, is\_locked: bool, seller\_name\_provided: bool | None | Renders the website URL dropdown, the refresh button, and the "Get Details" (scrape) button. |
| **Event Handling** | \_handle\_pending\_scraping | seller\_state | None | Similar to the URL search, this function runs during a rerun triggered by the "Get Details" button. It shows a spinner, calls the external get\_scraped\_data function, and updates the state with the scraped company details. |
| **UI Rendering** | \_render\_document\_upload\_section | seller\_state, is\_locked: bool | None | Renders the file uploader component for seller documents and the text area where scraped website data is displayed. |
| **Event Handling** | \_handle\_uploaded\_documents | seller\_state, seller\_documents\_upload, is\_locked: bool | None | Manages the display of uploaded files and renders the "Analyse documents" button. |
| **Event Handling** | \_process\_all\_documents | seller\_state, seller\_documents\_upload | None | Contains the core logic for document analysis. When triggered, it iterates through uploaded files, saves them, calls the external get\_services function for each, and adds the results to the state. |
| **UI Rendering** | \_render\_enterprise\_details\_section | seller\_state, is\_locked: bool | None | Renders the final, complex UI component on the tab: a three-column selector that displays the extracted services and capabilities for the user to review and select. |
| **Utility** | \_get\_default\_data\_from\_services | seller\_state | dict or None | A helper function that aggregates all the services extracted from multiple uploaded documents into a single dictionary to populate the three-column selector. |