#### **1. Introduction**

The web scraping methodology detailed herein is a foundational component of the TERAH (The Epic Retirement AI Helper) project. Developed in collaboration with The Epic Retirement Institute, TERAH is designed as a conversational AI chatbot to provide Australians with accessible and trustworthy information on retirement. The core of TERAH’s functionality is its ability to ground responses in a curated knowledge base derived from official, verified sources. This document outlines the process for using Power Automate Desktop to extract, clean, and prepare this data, ensuring factual accuracy and compliance with project goals.

The web scraping process is a critical part of the modular, LLM-agnostic architecture, providing the chatbot with a consistent source of truth. The extracted datasets, sourced from reputable government and industry bodies, are crucial for enabling TERAH to offer reliable information without providing regulated financial advice.

#### **2. Data Sources and Acquisition Strategy**

The project will acquire data from a range of official and industry sources, as approved by our industry partner, Bec Wilson. The data is primarily comprised of a mix of static information, paragraphs, Q&A sections, and structured tables.

The primary data sources include:

* Australian Government: ATO and Services Australia (Centrelink) pages on retirement planning, employment, and income support. This includes standard web pages as well as downloadable datasets from data.gov.au.
* Industry & Research: Websites for the Australian Bureau of Statistics (ABS), Super Consumers Australia, and Money Smart.

To maintain the factual accuracy and relevance of TERAH's knowledge base, a structured scraping schedule will be implemented. Web scraping will be performed on a quarterly to bi-yearly basis, strategically aligned with the Australian financial year (July-June) to capture any major regulatory or policy updates. This schedule balances the need for up-to-date information with the stability of the source data.

For the initial phase, the scraped data volume is not precisely determined but will be managed to remain within the data ingestion limits of the ChatGPT API.

#### **3. Web Scraping Methodology: Power Automate Flow**

The web scraping process will be automated using a dedicated Power Automate Desktop flow, designed for robust and efficient data extraction from diverse web content.

The core methodology involves:

* **Modular Flow Design**: The process utilizes a main flow and specialized subflows for content extraction, data cleaning, and error handling, enhancing reusability and maintainability.
* **Targeted Content Extraction**: Power Automate actions will extract both structured (tables, lists) and unstructured (paragraphs, Q&A) data, employing Extract data from web page and Get details of web page actions as appropriate.
* **Comprehensive Data Cleaning**: A dedicated CleanText subflow processes raw scraped data by stripping all HTML tags and attributes using a multi-line regular expression (<[\s\S]\*?>), consolidating whitespace and newlines, and removing irrelevant content. This process aims for structural compression and clarity, ensuring the text is "straight to the point" by removing non-essential formatting and redundant elements.
* **Flexible Data Storage**: Cleaned data will be saved as plain .txt files, with dynamic file location selection based on predefined conditions.

#### **4. Web Scraping Efficiency and Proper Use**

Efficient and responsible web scraping is paramount for TERAH, impacting both its operational performance and ethical standing. Our approach focuses on minimizing resource impact and maximizing data acquisition quality.

* **Efficiency Focus**:
  + **Optimized Execution Time**: Power Automate flows will minimize scraping duration by optimizing selector logic, utilizing Wait for UI element actions instead of fixed delays, and structuring subflows for streamlined execution. The frequency of flow runs is optimized for quarterly/bi-yearly updates, balancing data freshness with operational overhead.
  + **Reduced Resource Consumption**: Flows will operate leanly, avoiding unnecessary browser interactions or excessive data processing during the scrape itself to minimize load on the scraping machine and associated computational costs. This contributes directly to positive financial savings.
  + **Optimized Network Calls**: Strategies will prevent overwhelming target servers by careful pacing of requests and implementing pauses between page navigations.
  + **Structured and Compressed Output**: The text output is designed to be structured (via cleaning) and compressed (by removing verbose HTML and redundant whitespace), ensuring minimal file sizes while retaining all necessary information. This optimizes ingestion into the LLM and contributes to more efficient information delivery to users.
* **Proper Use and Ethical Considerations**:
  + **Server Load Management & Bot Detection Avoidance**: Request throttling (adding randomized delays between actions and pages, typically 2-5 seconds) will be implemented to prevent excessive load on government and other source servers, minimizing the risk of IP blocking. While Power Automate Desktop doesn't natively support advanced IP rotation, flows will mimic human browsing behavior, including setting realistic User-Agent headers (if allowed by the 'Launch new browser' action) and avoiding unnaturally fast actions that could trigger bot detection.
  + **Data Minimization & Compliance**: Only data directly relevant to retirement support will be targeted, adhering to the principle of data minimization and avoiding financial advice.
  + **Transparency & Attribution**: Although scraped data is for internal LLM grounding, the overall project maintains transparency about its data sources, aligning with ethical data practices.
* **Measuring Efficiency**: Monitoring of the scraping process will include:
  + **Execution Time Logging**: Power Automate's built-in logging will track the duration of each scraping run to identify bottlenecks.
  + **Output File Size (Before/After Compression)**: Tracking the file size after raw extraction versus after cleaning and structural compression will quantify the efficiency of the data preparation.
  + **Success/Failure Rates**: The existing alert system (detailed in Section 6) will provide immediate notification of failures, allowing for quick investigation into issues related to efficiency or blocking.
  + **Data Completeness & Accuracy**: Test cases during the curation phase directly assess if the required information is fully and correctly extracted.

By prioritizing efficiency and adhering to ethical best practices, the web scraping process supports TERAH's robust performance, ensures positive financial savings by automating manual tasks, and enhances user experience through efficient and accurate information delivery, reinforcing its reputation as a trustworthy, compliant AI helper.

#### **5. Data Curation and Quality Assurance**

To ensure that TERAH's knowledge base remains accurate and reliable, a dual-method approach to data sourcing will be employed: a combination of automated web scraping and manual imports. This strict adherence ensures the bot uses the texts as its authoritative data source, preventing misquotations.

The quality assurance process is as follows:

* **Data Overwriting**: For now, each new scrape will completely overwrite the previous dataset. This simple approach ensures that the knowledge base is always fully refreshed with the latest information, eliminating the risk of using outdated content.
* **Accuracy Testing**: The scraped data, once cleaned, will be subject to a series of test cases. A range of pre-set questions will be run against TERAH to verify that its responses are factually accurate and directly grounded in the new data.
* **Manual Review**: In the event of a scraping failure or a negative result from the test cases, the scraped data will be manually reviewed and supplemented with content from manual imports if necessary, ensuring the data integrity remains uncompromised.

#### **6. Error Handling and Future Maintenance**

To maintain the integrity and reliability of the data scraping process, a robust error handling and maintenance plan will be in place.

* **Alert System**: In the event that a UI element is not found, a webpage returns an error, or the scraping flow fails for any reason, an automated alert will be triggered. This alert will notify the development team so they can manually review the issue and perform a manual scrape if needed.
* **Incremental Updates (Future Development)**: A key future development for the project is to transition from a full overwrite model to an incremental update model. This would involve adding logic to the flow that could recognize repetitive data and only update or add new content, eliminating the need for full data replacement and improving the efficiency of the scraping process.
* **Schema Integration**: The current output is a plain .txt file, but a future enhancement involves storing the data in a structured schema. This would improve data organization and facilitate more complex data ingestion for advanced LLM architectures.

#### **7. Conclusion**

The web scraping methodology is a critical and well-defined component of the TERAH project, providing a dynamic and reliable source of truth for the LLM. By using Power Automate, the project can efficiently acquire, clean, and prepare data from trusted government and industry sources. The combination of automated scraping, manual review, and rigorous testing ensures that TERAH's knowledge base remains accurate, compliant, and ready to provide valuable information to users.