Requirements

REQ-1.1.1 The NLP model will accept a source of text to be analyzed. Ambiguity: what form may the text take (link, tweet, plaintext, news page, etc)?

REQ-1.1.2 The NLP model will identify colonialist language within a given body of text.

REQ-1.1.3 The NLP model will report the results of its analysis to the user. Ambiguity: How does this report occur? What is displayed?

REQ-1.1.4 (Optional) The NLP model will suggest decolonialized alternatives to colonialist language uses within an analyzed body of text.

REQ-1.2.1 The web scraper will collect the requisite data from a given website.

REQ-1.2.2 Requisite data (as described in REQ-1.2.1) may take the form of a news article, peer-reviewed study, or social media post.

REQ-1.2.3 The web scraper will store the data collected from sources for use in training the NLP model.

REQ-1.2.4 The web scraper will pass the data collected from sources for analysis by the trained NLP model.

REQ-1.2.5 The web scraper will accept a target to be scraped. Ambiguity: how flexible must the scraper be in what it can collect data from?

REQ-1.2.6 The web scraper will parse a web page via traversing the displayed HTML to collect requisite text. Ambiguity: should the scraper traverse the DOM or only parse raw HTML? Ambiguity: what tags are relevant to the scraper's collection process?

REQ-1.2.7 The web scraper will collect data from a given web page upon activation from the user interface by a given user.

REQ-1.2.8 The web scraper will complete its collection of data for a given website within 2s of user initiation.

REQ-1.2.9 The web scraper will provide visible feedback to the user upon completion of its processing.

REQ-1.3.1 The NLP model will accept input of text from the web scraper application.

REQ-1.4.1 The web scraper user interface will have no more than [10] interactable points immediately visible on the landing page.

REQ-1.4.2 The web scraper user interface will maintain continuity between all pages by placing common elements across pages in the same location on each page.

REQ-1.4.3 The web scraper user interface will achieve [insert quantifiable measurement of usability here].

REQ-1.4.4 The web scraper user interface will only have interaction points of a minimum size of [10x10px] to ensure ease of use.

REQ-1.4.5 (Optional) The web scraper user interface will implement accessibility features, including keyboard navigation support, screenreader support, and additional language options (French?).

REQ-1.4.6 The web scraper user interface will maintain a consistent color palette across all pages.

REQ-1.4.7 The web scraper user interface will use consistent terminology to refer to various features across all pages.

REQ-1.4.8 The web scraper user interface will provide immediate feedback on all interactable points (ex. slight changes in appearance of the interactable point, or a loading indicator).

REQ-1.4.9 The web scraper user interface will validate any data input into it prior to passing that input to other services.

REQ-1.4.10 The web scraper user interface will indicate to the user when erroneous inputs have been supplied to the user interface.

REQ-1.5.1 The web scraper will provide meaningful responses in the event of internal errors.

REQ-1.5.2 The web scraper will not expose stack traces or similarly detailed error reports to outside users.

REQ-1.6.1 The web scraper will have the following pages: a landing page, a scraper application page, and a page listing notable outside software tools.

REQ-1.7.1 The scraper application page will have the following interactable surfaces: a textbox for inputting a URL to scrape from, a button to scrape the formatted text from the URL, a button to scrape the raw text from the formatted URL, and a button to scrape highlighted text.

REQ-1.7.2 The scraper application page will have a header indicating the purpose of the page as a web scraper tool.

REQ-1.7.3 Once a URL has been input by the user, that URL will be displayed on the page separately from the textbox used to input the URL.

REQ-1.8.1 All clickable surfaces in the web scraper will change their appearance when hovered over.

REQ-1.8.2 All dropdown menus in the web scraper will indicate their purpose with a downward arrow indicator on the right side of the menu.

REQ-1.9.1 The web scraper application will use the following colors for the primary elements of the user interface in Light mode: #212529, #6c757d, #adb5bd, #f8f9fa, #ffffff

REQ-1.9.2 The web scraper application will use the following colors for the primary elements of the user interface in Dark mode: #121212, #1e1e1e, #2c2c2c, #f8f9fa, #ced4da, #495057, #66b2ff

REQ-1.9.3 The web scraper application will use the following colors for the primary elements of the user interface in Blue mode: #e3f2fd, #bbdefb, #90caf9, #0d47a1, #1976d2, #1565c0, #0d47a1

REQ-1.9.4 The web scraper application will use the following colors for the primary elements of the user interface in Disco mode: #ff00ff, #00ced1, #c8bca7, #4b0082, #8d6e63, #6d4c41, #7fff00, #5d40037

REQ-1.10.1 Any user data handled by the web scraper must be minimized, securely stored, and disposed of once it is no longer needed.

Requirements that will likely need to change to reflect changes in our project: 1.2.2, 1.2.8