Project requirements:

- asynchronous loading
- user interactions to access specific data
- open and scrape data from **pdfs**

Best web scraping libraries for our requirements?

1. Scrapy: with scrapy splash and scrapy playwright
    a. Advantages:
        i. Designed specifically for web scraping
        ii. Can extract structured data
            1. data mining
            2. information processing
        iii. Scrapy splash: compatible with javascript rendered page content, handles asynchronous loading efficiently
        iv. Scrap playwright: can handle dynamic content and user interactions
    b. Disadvantages
        i. Steep learning curve: much harder to learn than beautifulsoup
2. Selenium
    a. Advantages:
        i. Useful for websites that require user interactions + asynchronous js
        ii. Can handle clicking on documents and navigating through a site's dynamically loaded content
    b. Disadvantages:
        i. Slower

BeautifulSoup?

- Mainly used for extracting data from HTML and XML
- Asynchronous loading:
    - Does not execute js
        - Can't access content that relies on js to load asynchronously
        - only parses the static HTML content on initial page load
- User Interactions:
    - Does not support simulating user interactions

In addition to choosing a scraping lib, we still need to implement pdf handling

- None of the web scraping tools directly handle pdf files
- pdf parsing library is needed in addition

Rough implementation roadmap (with scrapy as example):

1. Setup and config

    a. Install scrapy (with splash and playwright) or selenium
2. Scrapy for Initial Data Collection:
    a. Use scrapy to crawl SLO meetings calendar for things like notices of preparation and draft EIR notices
    b. Use splash to handle js rendered content
3. Define data structure
    a. Define data structure to capture info about notices of preparation, public hearing agendas, etc.
    b. Include fields like project number, document title, url
4. Create "spiders"
    a. Spiders define how data will be scraped
        i. defines how to perform the crawl: following links, how to extract structured data from their pages
    b. identify target urls
5. Process scraped data and save it (csv? RDBMS?)
6. Integrate JS content with splash
    a. Configure settings to support rendering
7. pdf handling
    a. Download pdfs
    b. Extract pdf content with pdf parsing library (pdfMiner)
8. Scheduling/automation?
    a. Create cron jobs to run spiders regularly
    b. Scrape new info automatically