**Customizable Medical Quotations File Reader Software (Each increment’s Description)**

**21K-3372**

**21K-4529**

**21K-3153**

As our project is based on the Plan incremental model, following are descriptions of each increment.

**Increment 1: Integration of Beautiful Soup for HTML Document Scraping**

Objective: Incorporate Beautiful Soup (a Python library for scraping HTML documents) into the project

Tasks:

* Analyze the structure of HTML documents containing medical quotations to identify relevant tags and attributes.
* Integrate Beautiful Soup and utilize its parsing capabilities to extract data fields such as medicine name, discounts and costs.
* Implement error handling mechanisms to handle different types of HTML docs or unreadable names.

Testing: Address any parsing errors or inconsistencies discovered during testing. Check if correct data is being extracted, what data is not being extracted and figure out why. Fix issues.

Deployment: Present a demonstration to the primary stakeholder and start on increment 2.

**Increment 2: Implementation of Data Storage and Categorization in Excel Using Pandas** Objective: Develop a system to organize and categorize scraped data into an Excel document efficiently using the Pandas library.

Tasks:

* Utilize Pandas functionality to write the scraped data into an Excel spreadsheet.
* Implement sorting and filtering features within the Excel spreadsheet to easily find any medicine

Testing: Compare Excel spreadsheet to HTML documents to ensure data correctness and verification.

Deployment: Once the Excel sheet has been created and checked for correctness, show it to the primary stakeholder and ask for datasets in other file formats.

**Increment 3: Integration of Textract and PDFMiner for Scraping Multiple File Formats** Objective: Use Textract and PDFMiner libraries to enhance scraping functionality of various file formats such as PDF and CSVs.

Tasks:

* Research and analyze the structure of different file formats, including PDF and CSV.
* Implement parsing algorithms specific to each file format using Textract for PDFs and CSV parsing for CSV files.
* Utilize PDFMiner for additional PDF parsing capabilities if needed.

Testing: As with increment 1, ensure that the correct data was extracted, figure out what data was not extracted and fix any bugs that arise.

Deployment: Present a demonstration to the primary stakeholder and start on increment 4.

**Increment 4: Use Pandas to store results from Textract and PDFMiner in Excel doc**

Objective: Figure out how to store data extracted from various file formats, including those processed by Textract and PDFMiner into the Excel doc, using pandas.

Tasks:

* Utilize Pandas functionality to write the scraped data from other file formats into an Excel spreadsheet.
* Ensure that new data does not coincide with old data.
* Remove repeated medicine names, or if their discounts are different, differentiate the medicine by using alphanumeric identifiers.

Testing: Compare Excel spreadsheet to documents/ provided dataset to ensure data correctness and verification, to ensure correct names were extracted, correct discounts are being shown and so on.

Deployment: Once the Excel sheet has been created and checked for correctness, show it to the primary stakeholder, ask for feedback, and move on to finalization.

**Increment 5: Final Integration and Website Preparation**

Objective: Integrate stakeholder feedback, complete the integration of all modules, perform final testing, and prepare for the deployment of the system on a website.

Tasks

* Consider stakeholder feedback, fix issues, implement suggestions if feasible.
* Integrate all modules, including HTML scraping, Textract/PDFMiner processing, and Excel data storage, ensuring seamless communication and functionality.
* Conduct comprehensive testing to validate the integrated system, including end-to-end testing to ensure data flows correctly from extraction to storage.
* Implement a user-friendly interface for the website, incorporating features for data visualization, search, and export functionality.

Testing: Perform thorough testing of the system and website, prepare a set of beta users to check the system and ensure it is working as expected. Fix any lingering bugs and prepare for deployment.

Deployment: Prepare for project evaluation and show the final product to the primary stakeholder.