

SOFTWARE REQUIREMENTS SPECIFICATION (SRS) DOCUMENT

CSDLC

Team-5

Members: Kowshik, Chaitanya, Anandhini, Shri
Vidhatri

BRIEF PROBLEM STATEMENT

Building Dashboard, Adapters for registry sites(17), building containers for each adapter, schedule containers, map/merge data into a unified data model, and build data transformation scripts for each of the registries to normalise the scraped HTML content into JSON format.

SYSTEM REQUIREMENTS

- Linux based containers(Docker)
- Python
- Linux CLI tools/commands
- AWS
- Apache SOLR
- Pup for converting HTML to JSON
- JSTL for converting raw JSON to JSON
- HTML
- XML
- yq,xq,jq for XML.
- requests,lxml,beautiful soup,scrapy,selenium

USERS PROFILE

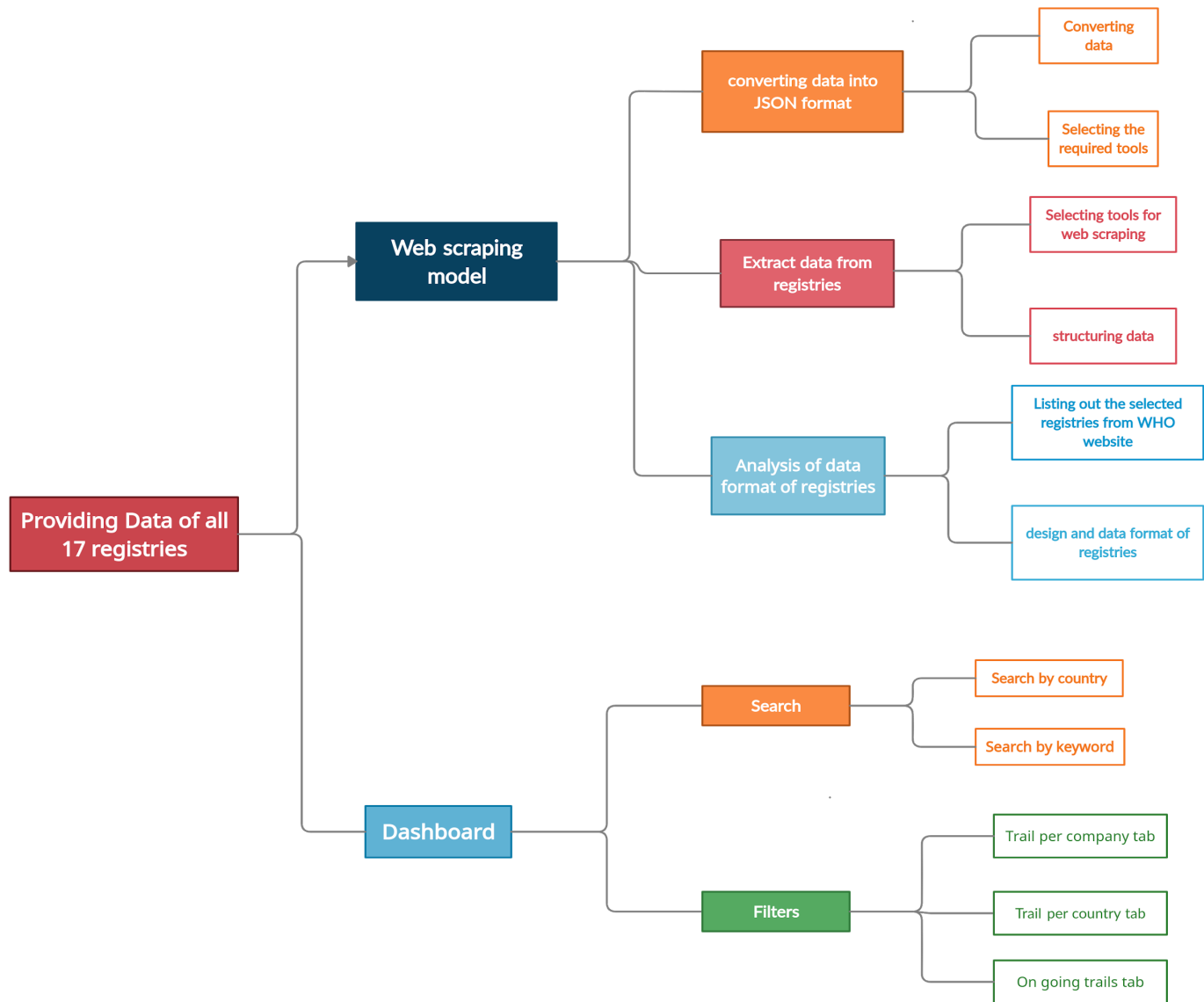
- Researchers
- Apothecaries
- Medical companies
- Colleges
- Students
- Patients

FEATURE REQUIREMENTS (DESCRIBED USING USE CASES)

No .	User Case Name	Description	Release
1.	Analysing data from registries	Document the data pattern and finalise the list of design patterns for data adapters/web scrapers and document each of them.	Initial draft on data patterns due on 04/02/2021
2.	web scraping data	Build web scrapers for each of the registries listed to collect information about all trials listed.	
3.	Structuring Data	Structuring data obtained from 17 different registries	
4.	Normalising data (JSON)	Converting different data formats to JSON form Build data transformation scripts for each of the registries to collect information about all trials listed there	
5.	Dashboard	optional if time permits as told by client.It involves search,filter,and other operations	

We are doing the only backend (getting data from registries in JSON format to be used in the existing website).

USE CASE DIAGRAM



USE CASE DESCRIPTION:

Use Case Number:	UC-1
Use Case Name:	Analysis of data from registries
Overview:	Document the data pattern and finalise the list of design patterns for data adapters/web scrapers and document each of them.
Actors:	Client
Precondition:	The registry must be mentioned in the WHO website.
Flow:	1.Pick 17 registries 2.divide the registries 3.analyse the data pattern and url patterns
Post Condition:	All the trials details must be accessible from the website

Use Case Number:	UC-2
Use Case Name:	web scraping data
Overview:	Build web scrapers for each of the registries listed to collect information about all trials listed.
Actors:	Client,Developers
Precondition:	All the trials details must be accessible from the website
Flow:	1. Selecting a tool for web scraping 2. Testing the tool 3. Scraping the data
Post Condition:	Data must be available individually

Use Case Number:	UC-3
Use Case Name:	Structuring Data
Overview:	Structuring data extracted from registries by web scraping.(Analyzing the different fields of extracted data, combining and grouping common fields)
Actors:	Client,Developers
Precondition:	Data must be available discreetly
Flow:	<ol style="list-style-type: none"> 1. Analysing the data 2. Creating containers for each registry
Post Condition:	Data must be convertible to JSON format

Use Case Number:	UC-04
Use Case Name:	Normalising data
Overview:	Converting all data extracted from registries to json form data
Actors:	Developers, Clients.
Precondition:	Web scrapers should extract data from the registries.
Flow:	Main (success) Flow: <ol style="list-style-type: none"> 1. Identifying the data format. 2. Using python libraries like pup,JSTL,xq,yq,jq to convert different formats like XML,HTML,raw JSON to JSON for the given data format

	Alternate Flows: <ol style="list-style-type: none"> 1. Identifying the data format. 2. Separating each Data Format. 3. Separately converting same type data formats using different programs
Post Condition:	Maintaining the database in the mainframe server.