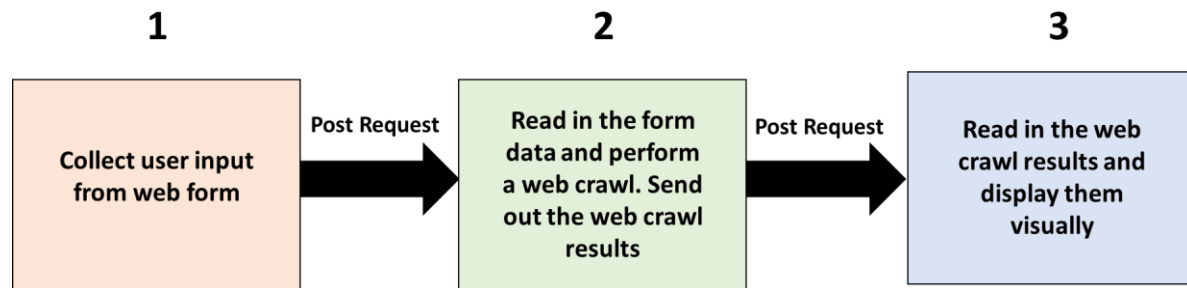


Structure of Web Crawler App

This is a high-level description of the web crawler app with a focus on how data is transferred from one part of the app to another. There are three main parts to the app as displayed in the image below.



I. Collecting User Input

A web form is used to collect user input. JavaScript value-checking along with defined form fields ensure that the user enters valid data in the form. When the user clicks submit, the form data is passed to the Python code in a post request.

II. Reading the User Data and Performing a Web Crawl

The Python program in the second part of the app has three main parts:

getFormData() gets form data from the post request and uses it to populate the Python variables corresponding to each of the input fields. The Python function `cgi.FieldStorage` is used to create an object holding the data received from the post request.

crawler.py is an imported module which does the web crawl using the Scrapy web crawling framework.

dataTransfer() creates a non-displayed form with a single field, a text area. The text area is populated with the JSON-formatted results from the web crawl, and the form is then auto-submitted as a post request to the visualizer.

III. Displaying the Web Crawl Results

The web crawl results are displayed through a three-step process:

1. On the server side, the post data sent from the Python program is placed in a non-displayed div on the page.
2. On the client side, the post data is accessed from the non-displayed div by JavaScript and saved in the variable `webcrawlResults`.
3. The `webcrawlResults` variable is then passed to the visualizer code to create the visual display.