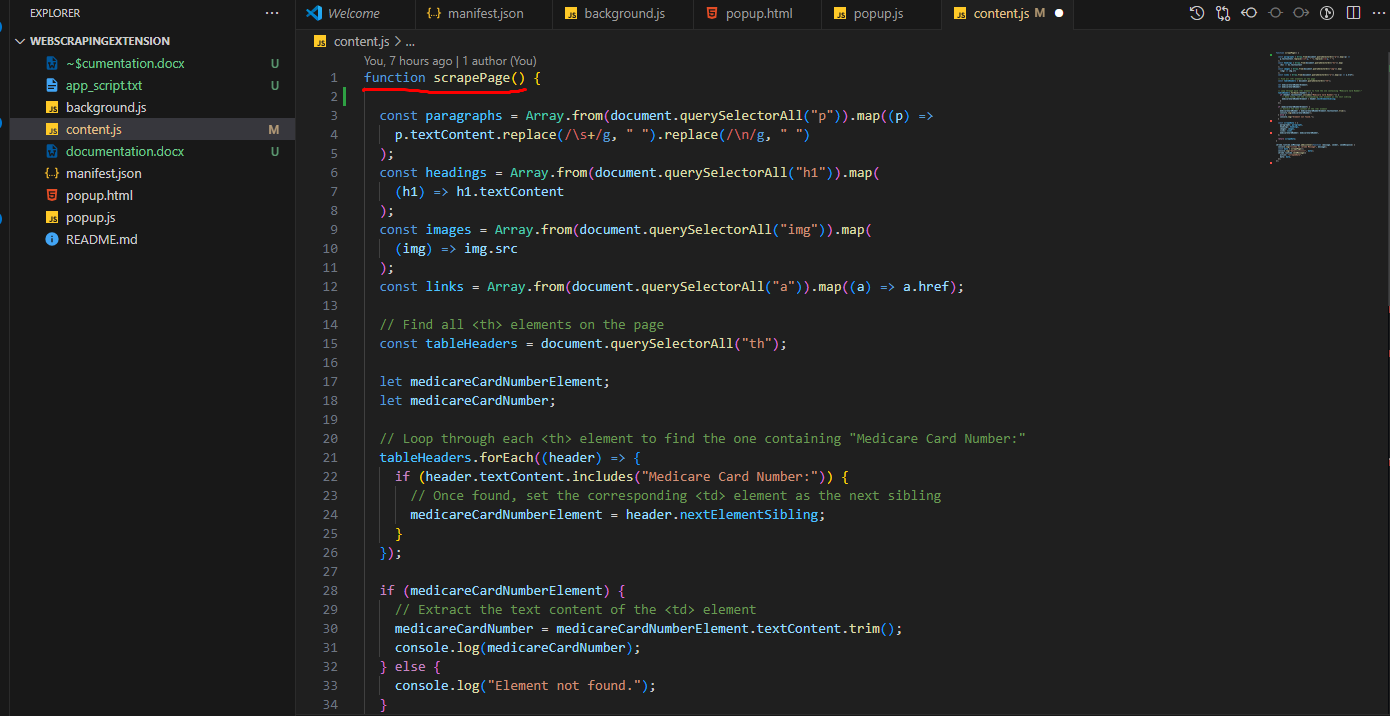
For updating, you can change scrapePage function in context.js.



**Description:**

This function extracts various data elements from a webpage using DOM manipulation techniques. It gathers paragraphs, headings, images, links, and optionally a Medicare Card Number if present.

**Fields Extracted:**

paragraphs: Array of text content extracted from <p> tags.

headings: Array of text content extracted from <h1> tags.

images: Array of image URLs extracted from <img> tags.

links: Array of URLs extracted from <a> tags.

medicareCardNumber: Extracted Medicare Card Number (if available).

**Algorithm:**

Extract paragraphs, headings, images, and links by selecting corresponding HTML elements.

Locate the "Medicare Card Number:" label within <th> elements to extract the associated value from the subsequent <td> element.

**Returns:**

An object containing the following fields:

paragraphs: Array of strings.

headings: Array of strings.

images: Array of strings (URLs).

links: Array of strings (URLs).

medicareCardNumber: String representing the extracted Medicare Card Number, or "Element not found" if absent.

**For updating:**

If you want to scrape other spec elements, you can use following pattern:

* By using tag name

const temp = document.querySelectorAll(“tag\_name”)

* Element with class  
  <div class="mdl-layout-title">Service and Support Portal</div>

const temp = document.querySelector(“.mdl-layout-title”)

* Element with id

<div id="mdl-layout-title">Service and Support Portal</div>

const temp = document.querySelector(“#mdl-layout-title”)

* Element with other attributes

<div aria-label="mdl-layout-title">Service and Support Portal</div>

const temp = document.querySelector(“div[aria-label=’mdl-layout-title’]”)

<div other\_attribute="mdl-layout-title">Service and Support Portal</div>

const temp = document.querySelector(“div[other\_attribute=’mdl-layout-title’]”)

* Some spec fields

To scrape email addresses:

const emails = Array.from(document.querySelectorAll("a[href^='mailto:']")).map((emailLink) => emailLink.href.replace('mailto:', ''));

To scrape phone numbers:

const phoneNumbers = document.body.innerText.match(/\(?\d{3}\)?[-.\s]?\d{3}[-.\s]?\d{4}/g);

To scrape product prices:

const prices = Array.from(document.querySelectorAll(".product-price")).map((priceElement) => priceElement.textContent.trim());

**Note: To scrape data, you need to analysis DOM of target web page**.