

The scraping tool used in this project was Web Scraper.io, a browser-based Chrome extension designed for structured scraping using visual selectors. Key features utilized included the selector graph for a hierarchical setup, element and text selectors, element scroll for infinite scrolling, and data preview and export options (CSV/JSON).

In terms of methodology, the first step involved page structure analysis. The target review page structure was inspected using Chrome Developer Tools, and it was found that each review was enclosed in a distinct HTML container (div) with identifiable class names. Following this, the selector configuration was designed using the following components: a scroll selector to simulate scrolling and load dynamic content, a reviewWrapper element to serve as the parent container for each review, text selectors for extracting the reviewer's name and full review text, an elementAttribute selector to extract the star rating from the alt attribute of the star image, and a text selector to extract the date of the review. These selectors were nested to ensure that each data field was a child of its corresponding reviewWrapper, ensuring row-wise consistency.

Regarding scroll vs pagination, the review platform employed infinite scroll, meaning that more reviews load automatically as the user scrolls down. A scroll selector was used to trigger this dynamic content loading. If the website had used pagination instead, such as "Next" buttons, a pagination selector would have been added to iterate through pages.

Several assumptions were made during the scraping process. It was assumed that the CSS selectors would remain stable, and that the class names and DOM structure would not change, as any such changes could break the scraper. Additionally, it was assumed that all reviews would be client-rendered, meaning they were not hidden behind JavaScript obfuscation or protected APIs. Each review was also assumed to contain the four fields mentioned earlier, with any missing fields resulting in empty cells in the scraped data. Lastly, it was assumed that the review website would not aggressively block traffic or rate-limit Web Scraper.io during moderate usage.