

Low-Level Design Document

Universal Website Scraper

Name – Subham Sahoo

Email id – subhamsahoo9861@gmail.com

Table of Contents :-

1. Introduction
2. System Architecture
3. User Interface
4. Web Scraping Logic
5. Data Storage
6. Error Handling
7. Deployment

1. Introduction

The purpose of this Low-Level Design Document is to provide technical details for the implementation of the Web Scraping GUI and Data Saving project. This document outlines the architecture, components, and technical considerations for the project.

2. System Architecture

The system comprises two main components:

- **GUI:** Developed using the Tkinter library in Python, the GUI allows users to input a URL and initiate web scraping.
- **Web Scraping Logic:** Implemented using the Requests, BeautifulSoup, and Selenium libraries, this component handles HTTP requests, data extraction, and saving to Excel.

3. User Interface

The GUI consists of the following elements:

- Entry field for the URL input.
- "Scrape & Save to Excel" button for initiating the scraping process.
- Appropriate labels and visual elements for user interaction.

4. Web Scraping Logic

- The requests library is used to make HTTP requests to the provided URL.
- BeautifulSoup (beautifulsoup4) is employed to parse the HTML content and extract relevant data.
- Selenium (selenium) is used to handle dynamic content and interaction with web pages.
- Error-checking is implemented to handle cases where elements cannot be found on the page.

5. Data Storage

- Extracted data is saved to an Excel file using the **openpyxl** library.
- A new worksheet is created in the Excel file, and the data is written to specific cells.
- The Excel file is saved with a predefined name ("flipkart_product_data.xlsx").

6. Error Handling

- The system checks for errors during web scraping and handles exceptions gracefully.
- Error messages are displayed in pop-up message boxes to inform the user.
- Specific error types include HTTP request errors, missing elements, and file-saving issues.

7. Deployment

- The project can be deployed on Windows, macOS, or Linux platforms.
- Users need to have Python and the required libraries installed.
- Ensure that the appropriate WebDriver (e.g., Chrome WebDriver) is installed for Selenium.