Software Requirements Specification

for

DocFinder

Version 1.0 approved

Prepared by Group 5

7th May 2023

Table of Contents

Revision History

Name	Date	Reason For Changes	Version

1. Introduction

1.1 Purpose

The document outlines the details requirements for developing a web scraping tool to collect accurate and up-to-date data on doctors from various medical websites and databases to support healthcare services for example patients can search for doctors anytime and look for specific things like reviews, prices, experiences and other more, also this data may then be utilized for a variety of reasons, including medical research, marketing, and analysis.

1.2 Document Conventions

This document has been typed using 'Times' font, with a font size of 12 and line spacing of 1.5. The document follows standard MLA format.

1.3 Intended Audience and Reading Suggestions

This SRS document is intended for product designers, developers, marketing personnel, and product testers. This paper will provide them with a thorough grasp of the software requirements, allowing them to design, sell, and test a user-friendly solution at the end. We recommend reading the paper from beginning to end to acquire a thorough understanding of the final product requirements.

1.4 Product Scope

The final product is intended to collect information about doctors from different trustworthy medical/healthcare websites and to allow patients to look for doctors by searching about specific things such as reviews, clinics, experiences and other more.

1.5 References

- IEEE standard 610.12-1990 IEEE standard Glossary of Software engineering terminology
- IEEE standard 730-1998 IEEE standard for Software Quality Assurance plans
- IEEE standard 830-1990 IEEE Recommended Practice for Software requirements specification.

2. Overall Description

2.1 Product Perspective

The context and origin of this project in this SRS is a new, self-contained web scraping tool designed specifically for collecting doctor data from various sources on the internet.

This tool is not a replacement for any existing system or a follow-on member of a product family, but rather a standalone solution designed to meet the growing need for doctor data collection in the healthcare industry.
The tool will be able to scrap data from various websites that contain information about doctors, including their names, specialties, contact information, education, and professional affiliations. While the tool is a self-contained product, it may be integrated with larger systems or used as a component of a larger software solution in the healthcare industry.

2.2 Product Functions

The main functions of the system are as follows:

- Search for doctors based on various criteria, such as name, specialty, location, or medical condition.
- View doctors profiles containing information such as their names, specialties, education, experience, patient ratings, and reviews.
- Contact doctors directly through the website, using secure messaging or email systems, or by phone or fax.
- Read articles and resources or blogs related to healthcare and wellness, offering users additional information and insights about various medical topics.
- Provide patient feedback or ratings on doctors, helping other users make informed decisions and improving the quality of care.
- Manage patient accounts on the website, allowing them to save their searches, appointments, and preferences, and to receive personalized recommendations and updates.
- Ensure data security and privacy of user data, complying with relevant regulations and standards in the healthcare industry.

2.3 User Classes and Characteristics

The following are the anticipated user classes:

System Admin: is the person who manages the database of the system.
System crawler: is the person who scraps websites to update doctors' data.
System User: is the website visitor who searches for doctors' information.

2.4 Operating Environment

- The web scraping tool will operate in a computing environment that includes a modern web browser and an internet connection. The tool will be designed to run on multiple hardware platforms, including desktops, laptops, and servers. The specific hardware requirements will depend on the amount of data being collected and the frequency of data scraping.

- The software will be compatible with multiple operating systems, including Windows, macOS, and Linux. The web scraping tool will be designed to operate independently of other software applications, but it may need to peacefully coexist with other components in certain cases. For example, the tool may need to be integrated with a database management system to store the collected data. In such cases, the tool will be designed to seamlessly exchange data and information with the database system.

- The web scraping tool will be developed using modern programming languages and libraries that are widely used in the industry such as python. It will be designed to operate in a secure environment and follow best practices for software development and security. The software will also be designed to scale easily as the amount of data being collected increases over time.

2.5 Design and Implementation Constraints

The following items or issues may limit the options available to the developers:

- Corporate policies: The software must comply with any corporate policies related to data privacy, security, and use of third-party components.
- Regulatory policies: The software must comply with any relevant regulatory policies related to healthcare data, privacy, and security.
- Hardware limitations: The software must operate within the hardware limitations, including timing and memory requirements, of the target environment.
- Interface limitations: The software must be compatible with other applications or systems that it interfaces with, and must comply with any interface limitations imposed by those systems.
- Technology limitations: The software must be developed using specific technologies, tools, and databases specified by the client.
- Language requirements: The software must support specific languages, as required by the project proposal.
- Communications protocols: The software must comply with any communications protocols specified by the client or required by the target environment.
- Security considerations: The software must comply with any security requirements specified by the client, including authentication, authorization, and data encryption.
- Design conventions and programming standards: The software must comply with any design conventions and programming standards specified.

2.6 User Documentation

In this Project we offer the user a searchable database of frequently asked questions (FAQs) and troubleshooting tips to assist users in resolving issues with the tool.
This user documentation component will be delivered in digital format, accessible through the web scraping tool's user interface. The documentation will be delivered in compliance with standard formatting and delivery standards, such as **HTML** format, to ensure compatibility with a wide range of devices and operating systems.

2.7 Assumptions and Dependencies

Assumed factors that could affect the requirements stated in the SRS include:

- The availability and quality of the doctor data sources on the internet.
- The stability and reliability of the internet connection during web scraping.
- The compatibility of the web scraping tool with the web browsers and operating systems used by potential users.

- The legal and ethical considerations around web scraping for doctor data.
- The availability of funding and resources to support the development and maintenance of the web scraping tool.
- The potential need for future updates or modifications to the web scraping tool based on changes to websites or data sources.

Dependencies of the project on external factors include:

- Compatibility with web browsers and operating systems used by potential users.
- Availability of third-party libraries or tools for web scraping.
- Compliance with legal and ethical guidelines related to web scraping for doctor data.

3. External Interface Requirements

3.1 Hardware Interfaces

Database Interface: The database interface provides a logical connection between the software product and the physical database. The logical characteristics of this interface include query languages such as **SQL** that are used to retrieve and store data. The physical characteristics include the database server, storage devices, and network connections.

Operating System Interface: The operating system interface provides a logical connection between the software product and the physical hardware of the computer system. The logical characteristics of this interface include system calls and APIs that are used to interact with the operating system. The physical characteristics include the CPU, RAM, storage devices, and other peripherals.

API Interface: The API interface provides a logical connection between the software product and other software applications. The logical characteristics of this interface include the API calls that are used to send and receive data. The physical characteristics include the hardware components and networking infrastructure that enable communication between the software applications.

3.2 Software Interfaces

Database: The web scraping product for doctors may require a database to store the scraped data. The database may be a relational database such as MySQL or a NoSQL database such as MongoDB. The product may use SQL or other query languages to retrieve and store data in the database.

Operating System: The web scraping product for doctors may be developed to run on a specific operating system such as Windows or Linux. The product may use system calls and APIs provided by the operating system to interact with hardware components and other software applications.

Web Scraping Tools/Libraries: The web scraping product for doctors may use specific web scraping tools and libraries such as **BeautifulSoup** or Scrapy to extract data from websites. These tools and libraries provide APIs for parsing **HTML** and **XML** documents and extracting relevant data.

3.3 Communications Interfaces

Web Browser: The web scraping product require communication with web browsers to access and extract data from websites. The product require a specific version of web browsers, such as Chrome or Firefox. The communication use **HTTP** or **HTTPS** protocols to retrieve web pages.

Network Server: The web scraping product require communication with network servers to retrieve and store data. The product may require access to specific servers or network shares, and may use **FTP**, **SFTP**, or other protocols to transfer data between the server and the product.

Electronic Forms: The web scraping product require communication with electronic forms to retrieve or submit data. The product may use **HTML** forms or other formats to interact with electronic forms.

4. System Features

4.1 Authentication

4.1.1 Description and Priority

The authentication system is an important feature for the website that requires users to identify themselves before being able to add a review to the doctor. It has a high priority to prevent fake reviews and spam.

4.1.2 Stimulus/Response Sequences

Users should be able to register on the website by providing their basic details, such as username, email address, and a password. Then each time the user logs in to the system, a verification code will be sent to their email.

4.1.3 Functional Requirements

REQ-1: User shall be able to sign in into the system
REQ-2: User should be able to create a new account
REQ-3: System shall be able to send verification code to user's email
REQ-4: User shall be able to reset their password

4.2 View doctors' information

4.1.1 Description and Priority

The view doctor feature is one of the core features of the system, where users shall be able to view doctor's detailed information/profile. It has a high priority so the users can find a healthcare professional who meets their medical needs.

4.1.2 Stimulus/Response Sequences

Users will be able to search for a doctor on the website. Once they find a doctor that matches their criteria, they can view the doctor's profile, which will contain important information such as the doctor's qualifications, experience, areas of expertise, and patient ratings and reviews.

4.1.3 Functional Requirements

REQ-1: User shall be able to view top rated doctors
REQ-2: User will be able to view doctor’s profile
REQ-3: User should be able to view doctor’s area of expertise
REQ-4: User shall be able to view doctor’s clinic

4.3 Search for doctors

4.1.1 Description and Priority

Search for doctor feature is a critical aspect of any healthcare website or application that aims to connect patients with medical professionals. This feature enables users to search for doctors based on various criteria, such as location, specialty, availability, and rating, and helps them find a medical professional who can meet their healthcare needs.

4.1.2 Stimulus/Response Sequences

On the homepage of the website, users will have access to a search bar that they can use to enter keywords. Additionally, an advanced filter option will be available, enabling users to refine their search results based on specific criteria such as location, specialty, or rating.

4.1.3 Functional Requirements

REQ-1: User should be able to search doctors by name, location, and field
REQ-2: User shall be able to filter doctors by hospitals and clinics
REQ-3: User shall be able to sort doctors by their reviews

4.4 Doctors’ reviews

4.1.1 Description and Priority

Adding a review for a doctor is a high priority core feature that allows patients to share their experiences with a medical professional and provide feedback on the quality of care they received. It helps other patients make informed decisions about which doctor to choose.

4.1.2 Stimulus/Response Sequences

Once a user has searched for and viewed a doctor's profile, there will be a review button for adding a review, if they click it, they will have the ability to submit a review about the doctor if it meets the website's criteria for validity.

4.1.3 Functional Requirements

REQ-1: User shall be able to add a review for a doctor

REQ-2: User should be able to view doctor’s reviews
REQ-3: User will be able to compare doctors by reviews
REQ-4: System shall be able to detect fake and spam reviews

4.5 Scraper

4.1.1 Description and Priority

This feature can allow the healthcare website or application to provide a comprehensive list of doctors in each area or specialty, even if they are not listed on the website itself. It's the highest priority feature, because it should be designed to collect accurate and up-to-date information on doctors and should be regularly maintained to ensure that the data remains relevant.

4.1.2 Stimulus/Response Sequences

The website administrator will have the ability to initiate the web scraper tool, select the desired area of expertise to gather data on, and then the scraper will automatically extract doctors' information from other healthcare websites and store it in the website's database.

4.1.3 Functional Requirements

REQ-1: Scraper will be able to scrap doctor’s profile information
REQ-2: Scraper should be able to scrap doctor’s area of expertise
REQ-3: Scraper should be able to get doctor’s location
REQ-4: Scraper must be able to scrap multiple wesbites at once

4.6 Admin

4.1.1 Description and Priority

An admin feature allows the website administrator to manage and monitor the website's content, user accounts, and system settings. This feature is critical for ensuring that the website operates smoothly, securely, and in compliance with legal and ethical standards.

4.1.2 Stimulus/Response Sequences

After logging into the admin page, the administrator will have access to a wide range of actions they can perform on the website. These actions could include adding or deleting user accounts, managing doctor profiles, moderating user-generated content, configuring system settings, and generating analytics reports.

4.1.3 Functional Requirements

REQ-1: Admin must be able to add users
REQ-2: Admin shall be able to keep track of new doctors added to the system
REQ-3: Admin shall be able view all users of the system
REQ-4: Admin should be able to view all reviews added by users
REQ-5: Admin must be able to track fake and spam reviews
REQ-6: Admin shall be able to ban/remove users with spam or fake reviews

5. Other Nonfunctional Requirements

5.1 Performance Requirements

- Throughput: The web scraping product may be required to process a large volume of data in a short amount of time. For example, the product may be required to scrape data from hundreds of web pages in under an hour. This requirement is important to ensure that the product can handle large-scale data extraction efficiently.

- Scalability: The web scraping product should be able to handle increasing amounts of data without compromising performance. For example, the product should be able to scrap data from hundrers of web pages without slowing down or crashing. This requirement is important to ensure that the product can meet the needs of a growing user base and handle increasing amounts of data.

5.2 Safety Requirements

- Data Privacy: The web scraping product should not violate any laws or regulations related to data privacy, such as the General Data Protection Regulation (GDPR). The product should not extract any personal information about doctors without their explicit consent or in violation of privacy policies.

- Use of Third-Party Libraries: The web scraping product should not use any third-party libraries that violate safety or security standards. The product should only use libraries that have been thoroughly tested and approved for use by the development team.

5.3 Security Requirements

- User Authentication: The web scraping product should require users to authenticate themselves before using the product. User authentication may involve username and password authentication, multi-factor authentication, or other methods that verify the identity of the user.

- Data Encryption: The product should use encryption algorithms to secure any sensitive data transmitted between the user and the server, as well as any data stored on the server or user's device.

- Secure Data Storage: The product should store extracted data in a secure and encrypted manner to prevent unauthorized access or data breaches. The product should also have mechanisms in place to protect against data loss or corruption.

5.4 Software Quality Attributes

- Scalability: The ability to handle an increasing amount of data without a significant decrease in performance. For example, the product should be able to handle scraping data from a large number of websites without crashing or slowing down.

- Maintainability: The ease with which the product can be maintained, modified or updated. This includes code readability, use of clear documentation, and well-structured code.

- Reusability: The extent to which components of the product can be used in other contexts or products. For example, if the product includes a component that extracts data from a particular type of website, it should be possible to reuse that component for other websites of the same type.

- Usability: The ease of use and navigation of the product. This includes features such as a clear and intuitive user interface, clear error messages, and easy-to-understand documentation.

5.5 Business Rules

- Only authorized users with valid login credentials can access and use the product.

- The product should only scrape publicly available data and not violate any laws or regulations.

- The product should scrape data in a manner that is respectful of the websites being scraped, avoiding excessive traffic or server load.

- The product should prioritize accuracy and completeness of scraped data, and alert users if data is missing or incomplete.

- The product should store scraped data securely and protect user privacy.

Appendix A: Glossary

- ***DocFinder** – Doctor Finder*
- ***HTML** – Hyper-text Markup Editor*
- ***API** – Application Program Interface*
- ***MacOS** – Mac Operating System*
- ***SRS** – Software Requirement Specification*
- ***SDS** – Software Design Specification*
- ***UI** – User Interface*
- ***HTTP** – Hyper-text Transfer Protocol*
- ***FTP** – File Transfer Protocol*
- ***SQL** – Structured Query Language*

- ***CPU** – Center Processor Unit*
- ***OS** – Operating System*

Appendix B: Analysis Models

Will be found in SDS document

Appendix C: To Be Determined List