Chapter 2

Software Requirements and Specifications of classification of unstructured data from online course web pages

A Software Requirements Specification (SRS) is a comprehensive description of the intended purpose and environment for software under development. The SRS fully describes what the software will do and how it will be expected to perform**.** An SRS minimizes the time and effort required by developers to achieve desired goals and also minimizes the development cost. A good SRS defines how an application will interact with system hardware, other programs and human users in a wide variety of real-world situations. Parameters such as operating speed, response time, availability, portability, maintainability, footprint, security and speed of recovery from adverse events are evaluated [41].

2.1 Overall Description

In this project, a system is considered which provides the user offline access to the various online courses available. For initial setup as well as subsequent updates over a defined time frame like a month, the system requires to access the Internet to perform crawling and indexing of web pages. Once this data is collected, efficient classification algorithms will be applied offline to segregate the pages and fill them into a database for the user to access in future.

2.1.1 Product Perspective

This project is aimed at providing the service of web page classification by making use of machine learning algorithms and improving the experience by using a very high quality of pre-processed data used for training. The technique proposed in this project is through improved random forest algorithm and compare it with other classification algorithms which have hitherto not been used with this particular type of data. This novel approach to classification also seeks to combine multiple features such as social tagging in multiple websites as well as spatial and temporal relevance of web pages. Users are provided access only to relevant courses using an interactive, fresh and intuitive front end that enhances user experience and is also portable enough to be used on mobile platforms.

2.1.2 Product Functions

The project aims at providing users easy offline access to relevant online courses available. Thus, the product performs the following functions:

- Crawls the web and identifies relevant online course web pages
- Provides the option for a user to find courses by subject, duration and university
- Enables a user to update the application over a long term span so that the information remains relevant
- Uses highly optimized machine learning algorithms to ensure the most accurate results without wastage of computational resources
- Allow code portability across multiple platforms.

2.1.3 User Characteristics

A user can be any person who is new to the service. The user might be very much familiar with the available functionalities or a complete novice. Hence the GUI of the application should be built in such a way that the user takes no time to get used to its interface. The target audience is everyone who wishes to obtain information about online courses right from students to universities and corporate organizations wishing to provide training to employees.

2.1.4 Constraints

This software has been designed keeping in mind the basic factors of user convenience and improved functionality together. However, there are various aspects that constrain the functioning of the software:

- Crawling the entire web is extremely time consuming and does not justify the cost of resources spent due to relevance of results. Hence the user must note that since there is no manual creation of results, the application can only strive for highest levels of accuracy but cannot guarantee the absence of any discrepancies.
- Slow internet connections or outdated versions of browsers might not be able to handle the website. Since the functionality of the software depends on the functioning of the server, this would be an additional constraint.
- User understanding and familiarity is another constraint upon which the usefulness of the software depends, something that may vary across systems.

2.1.5 Assumptions and Dependencies

For the service to function as expected, certain prerequisite conditions must exist on which the application can run. The presence of these conditions results in the most optimal performance. Thus, the following subsection discusses few assumptions and dependencies that help the application to run effectively.

The web service is assumed to be updated at frequent intervals such that all the information provided is most recent. This will enable the database to be more reliable and useful. Thus, a dependency exists on the regular updates of records for proper user information and proper automation.

2.2 Specific Requirements

This section contains the various requirements specific to each parameter of the entire project. The following section briefly describes the different project requirements essential for its proper working.

2.2.1 Functionality Requirements

The functional requirements for the product are as follows:

- Enable users to initially set up the application easily.
- Enable a user to easily access instructions related to the application.
- Enable a user to specify the parameters for selection of new courses.
- Enable the system to periodically update by re-crawling and classification of data obtained thereof.

2.2.2 Performance Requirements

The performance of the project mainly depends on the available internet bandwidth which goes hand-in-hand with the computer hardware requirements. The various performance requirements are detailed further in this chapter with respect to both the developers' and users' point of view.

2.2.3 Supportability

The web service will be supported on various operating systems as it is a web based system. It only requires a web browser to access its contents since web browsers provide UI to modify the database and use its functionalities. Therefore, supportability or maintainability of the system being built, including coding standards, naming conventions, class libraries, maintenance access and maintenance utilities can be enhanced by implementing it like the real world application projects.

2.2.4 Software requirements

Software requirements deal with defining software resource requirements and prerequisites that need to be installed on a computer to provide optimal functioning of an application. These requirements or prerequisites are generally not included in the software installation package and need to be installed separately before the software is installed.

- Java1.4 or higher
- Java Swing-front end
- JDBC-Database connectivity
- UDP-User Datagram Protocol
- TCP-Transmission Control Protocol
- Networking-Socket programming
- MySQL server on Windows/ Linux/ Mac OS X
- Windows 98 or higher-Operating System

2.2.5 Hardware requirements

The most common set of requirements defined by any operating system or software application is the physical computer resources, also known as hardware, A hardware requirements list is often accompanied by a Hardware Compatibility List (HCL), especially in case of operating systems. An HCL lists tested, compatible, and sometimes incompatible hardware devices for a particular operating system or application. The following sub-sections discuss the various aspects of hardware requirements.

All computer operating systems are designed for a particular computer architecture. Most software applications are limited to particular operating systems running on particular architectures. Although architecture-independent operating systems and applications exist, most need to be recompiled to run on a new architecture.

The power of the Central Processing Unit (CPU) is a fundamental system requirement for any software. Most software running on x86 architecture define processing power as the model and the clock speed of the CPU. Many other features of a CPU that influence its speed and power, like bus speed, cache, and MIPS (Million Instructions Per Second) are often ignored. This definition of power is often erroneous, as AMDAthlon and Intel Pentium CPUs at similar clock speed often have different throughput speeds. Hence the following are a recommended minimum hardware configuration:

- 10GB HDD
- 0.125 GB RAM
- Pentium P4 Processor 3.3GHz

2.2.6 Design Constraints

Due to the evolving technologies being used and with the growing demand for better technologies the conventions being used in programming languages is also changing. To support better GUI and User experience the designers have favoured the use of Java Swing front end which also provides good database connectivity. Hence, even though Java is becoming a ubiquitous part of almost all machines today, this design constraint mandates its presence for correct and complete functioning of the application.

2.2.7 Interfaces

The interface must be as user friendly as possible. Even a new user should easily understand the complete functionalities provided by the social networking service. This can only be achieved by providing a user-friendly GUI. The user will be provided with a HELP page with information on how to navigate the functions of the product. A possible improvement in subsequent versions would be the addition of a help video or a similar tutorial.

2.3 Other Non-functional Requirements

The other non-functional requirements include performance requirements that focus to accommodate user requirements in terms of usability. It also includes safety and security requirements along with software quality attributes.

2.3.1 Performance Requirements

There are a basic set of requirements that state the minimum performance expected from the system from the user's point of view to ensure a smooth and glitch-free operation of the system. This set of requirements is listed below:

- No user shall experience difficulty in understanding the rules and functioning of the application
- No user shall experience any difficulty in navigating through the various features of the application
- No user shall find it cumbersome to sort and use different search parameters for online courses
- No user will be subjected to a lot of menus to access all the relevant information.

2.3.2 Safety and Security Requirements

Safety and security of the system are crucial attributes that seek to preserve the integrity of the system. The system must be protected against all failures and attempts to unauthorized attack and access of information. The following requirements detail the same:

- Since the system seeks to function offline, it is recommended that client machines are installed with strong firewalls and antivirus software to ensure that the software can function smoothly. In addition to this, there should be adequate privacy and firewall authentication each time the application attempts to connect to the internet and update itself, but must request user permission for the same at all times.
- The database will be stored on the client machine hence reducing the possibility of server related downloading attacks

- To ensure higher security standards, it is recommended that the web interface is accessed using secure protocols such as the https protocol for secure data transmission between the client and the server machines.

2.3.3 Software Quality Attributes

The additional requirements detailed below are related to the quality of the software and list out the desired characteristic attributes for the software to have. These are not technically specified but encouraged to be adhered to in development. They are:

a) **Adaptability -** Since the software is dynamic in nature, provision must be made to include newer courses based on the need of the hour. Since certain courses are location-based or occupation based, the software must be able to adapt to these with minor modifications
b) **Availability -** The software must be running and able to deliver required services to the user at any given point in time.
c) **Correctness -** Since the information generated as output of the classification algorithm is meant to eliminate manual labour, the application must strive to be of highest level of accuracy when classifying web pages as relevant to user needs.
d) **Flexibility -** The types of users will vary in levels of experience, expertise, familiarity with the interface and multiple other factors. The product must be flexible enough to cater to this diversity and the user needs as well.
e) **Interoperability -** The product must be universally available on all platforms and operating systems.
f) **Maintainability -** The product must require minimal maintenance which should be easy to accomplish by means of simple code and detailed documentation.
g) **Portability -** The services offered by the product must be independent of location, platform and other location or system based parameters.
h) **Reliability -** The risk, likelihood and severity of potential failure of the product must be minimized. If failure occurs, the product should show adequate recovery from the failure state.
i) **Reusability -** The product must be developed using methods in accordance with software reuse like improving on a pre-existing version without having to start from scratch.

j) **Robustness -** The product must be able to function under different sets of constraints like high server load, incorrect input data and other potentially dangerous scenarios, with minimal chance of failure**.**

k) **Testability -** The product must be such that it's testing is simple, orderly and easy to document and improve upon. Unit testing, component testing, sub-system and system integration testing, as well as end-user and release testing must be done and the reports of the same must be documented.

l) **Usability -** The product must be easy to use and must not cause the end user any trouble in navigational or understanding its various functionalities and interfaces.

2.4 Summary

The second chapter of the report has detailed the software requirements specification of the project in terms of hardware, software, functional, non functional, security, quality and other desirable attributes of software. The entire product has been developed according to these specifications which keep the user need and developer convenience and quality uppermost at all times.