****

**Keyword Search application**

**Software Requirement Specification (SRS) Document**

**Sprint 1 Implementation**

**Project Timeline: 07.09.2022 to 14.09.2022**

**INDEX**

1. Introduction

1.1 Purpose -------------------------------------------------- 4

1.2 Intended audience -------------------------------------------------- 4

1.3 Intended use -------------------------------------------------- 4

1.4 Scope -------------------------------------------------- 4

2. Overall description -------------------------------------------------- 5

2.1 Assumptions and dependency ------------------------------------------------- 5

3. System feature and requirements -------------------------------------------------- 5

3.1 Functionality -------------------------------------------------- 5

3.1.1 File\_Parsing Function — ----------------------------------------------- 5

3.1.2 File\_Validation Function -------------------------------------------------- 5

3.1.3 Display\_Invalid\_Files function -------------------------------------------------- 5

3.1.4 Extract\_Project\_Details function ----------------------------------------------5

3.1.5 Finding\_Unique\_Keywords function -------------------------------------------- 6

3.1.6 Keyword\_Search\_Main Function ------------------------------------------- 6

3.1.7 Display\_Result function -------------------------------------------------- 6

3.1.8 Store\_Result function -------------------------------------------------- 6

3.2 System requirement -------------------------------------------------- 6

3.2.1 Tools to be used -------------------------------------------------- 6

3.3 Non-Functional Requirements ------------------------------------------------- 6

3.4 System feature -------------------------------------------------- 6

4. Data Flow Diagram

4.1 DFD level 0 ------------------------------------------------ 8

4.2 DFD level 1 ------------------------------------------------ 9

### **1.** **Introduction: -**

The introduction of the software requirement specification provides an overview of the entire software. The entire SRS with overview description purpose, scope, tools used and basic description. The aim of this document is to gather, analyze and give an in-depth insight into the complete Keyword Search for Research paper application by defining the problem statement in detail. The detailed requirements of the Keyword Search for Research Paper application is provided in this document.

**1.1** **Purpose**: **-**The purpose of this document is to show the requirements for the Keyword Search application, which can be useful in finding research papers with specific keywords.

**1.2** **Intended Audience: -**This document is intended to be read by, Client.

**1.3** **Intended Use: -**

* Development Team
* Maintenance Team
* Clients

Since this a general-Purpose Software any one can access it.

**1.4** **Scope: -** This application can be very helpful to people for finding information a by using specific keyword. This application displays all research papers containing “keyword” attribute = given keyword. So, with this application it will fetch Numerous research papers as results, which contains the keyword that we have searched .

This application displays all research papers containing “keyword” attribute = given keyword.

**2. Overall Description: -**

It is keyword search application used to keep track of unique keywords occurring in multiple research papers. With this application user can find project ID's associated with entered keyword. This application takes research papers as command line input. The program starts by validating the input files. After successful validation the file is been parsed and useful information (which help in searching) are stored. Simultaneously we will be tracking unique keywords. Once the above processes are completed, we will find and store all the project id's associate with unique keywords. The main purpose of this application is to keep track of keywords and it's project id, which can make search efficient.

**2.1 Assumptions and Dependency: -**

* System should have Ubuntu Linux installed.
* · System should have either 2GB or more RAM.
* The service is used preferably on a desktop or laptop.

**3.System Features and Requirements: -**

**3.1 Functionality: -**

**3.1.1 KS\_01-> File\_Parsing:** This is the first main menu level function that allows the user to parse a particular file into the program, and further read and write the contents of file.

**3.1.2 KS\_02-> File\_Validation:** This function validates whether the given research paper is in the required format. The research paper should contain Project ID, Project Name, Keywords, Abstract, Author and References.

**3.1.3 KS\_03->** **Display\_Invalid\_Files:**  After the validation of the given file, if it isn't satisfying the input given format of the Research Paper, it will display ERROR message.

**3.1.4 KS\_04-> Extract\_Project\_Detail:** This function will find the relevant project ID and project name from the input specified format of the research paper.

**3.1.5 KS\_05-> Finding\_Unique\_Keywords:** As the name suggests, this function will find the unique keywords from all the research papers.

**3.1.6 KS\_06-> Keyword\_Search\_Main:** This function is the core of the application that searches one of the unique keywords on the wish of the user, returning with the project id and project name of the research paper in which the keyword was found.

**3.1.7 KS\_07-> Display\_Result:** This function’s task is to show the results that we have obtained . It will shows the project id ,project title where the keyword is found.

**3.1.8 KS\_08-> Store\_Result:** This function will show the results in the separate file in the results Folder. This File will be text file which can be viewed to see the results obtained from the application.

**3.2 Technical Requirements: -**

### **3.2.1. Tools to be used:**

* Pthread Library
* C File Handling
* C Language
* System Programming

**3.3 Non-Functional Requirements:**

* C Unit to automate unit testing
* Valgrind to detect memory leak
* Make file
* Multi file multi directory solution with two step compilation process.

### **3.4 System Features: -**

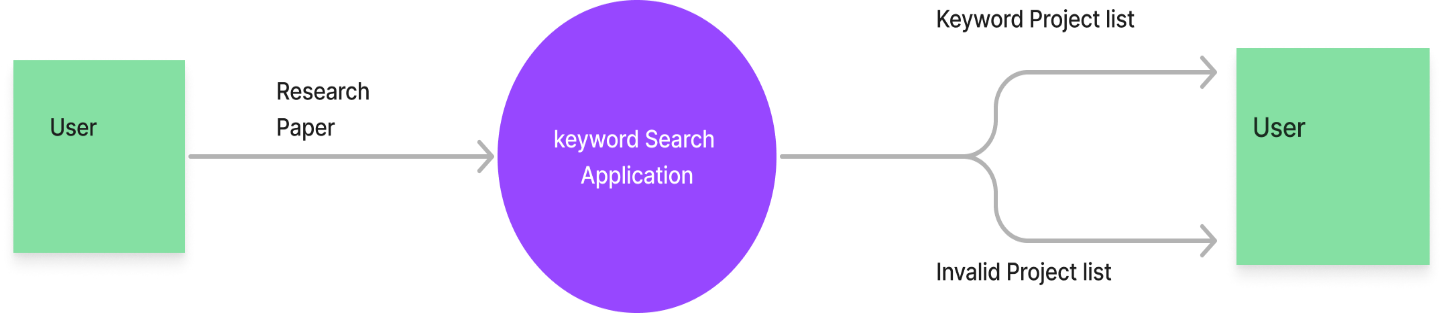
### Supportability:The system is easy to use.

* Design Constraints: The system is built using only C language.
* Usability:The automatic bowling scorer application can be used to replace the old means. Recording score by displaying various information to the player such as number of pins knocked down by each ball, the frame number which is currently going on, the cumulative score gained throughout the the end of each frame and the total number of points scored by the player after the completion of a total of ten frames.
* Reliability & Availability**:** The system is available 24/7 that is whenever the user would like to use the system, they can use it up to its functionalities.

### Performance: The system will work on the user’s terminal**.**

**4. DataFlow Diagram:**

**4.1 DFD Level 0 -**



**4.1 DFD Level 1 -**

