# Project Phase I: KWIC Requirement Specification

Versions 2.0

CS/SE 6362 Advanced Software Architecture (Fall 2015)

**Submitted to:**

**Dr. Lawrence Chung,**

**Associate Professor,**

**Department of Computer Science,**

**The University of Texas at Dallas,**

**Richardson, TX -75080**

**Submitted By:** Team Name: Quick Search

**Sruthi Chappidi,** [**sxc105920@utdallas.edu**](mailto:sxc105920@utdallas.edu)**,**

**Barbara Maweu,** [**bmk101020@utdallas.edu**](mailto:bmk101020@utdallas.edu)

**Maryellen Oltman,** [**mco130030@utdallas.edu**](mailto:mco130030@utdallas.edu)

**Twinkle Sharma,**[**txs151730@utdallas.edu**](mailto:txs151730@utdallas.edu)**.**

**Team Website:** [**www.utdallas.edu/~maryellen.oltman**](http://www.utdallas.edu/~maryellen.oltman)

## **Revision History**

|  |  |  |  |
| --- | --- | --- | --- |
| **Date** | **Version** | **Description** | **Author(s)** |
| 9/29/2015 | 1.0 | Preliminary version of K.W.I.C system architecture | All |
| 10/15/2015 | 2.0 | Updated version with professor suggestion from interim presentation | All |

Table of Contents

[1. Introduction 3](#_Toc433091738)

[1.1 Purpose 3](#_Toc433091739)

[1.2 Scope 3](#_Toc433091740)

[1.3 Definitions, Acronyms, and Abbreviations 4](#_Toc433091741)

[1.4 Project Deliverables 4](#_Toc433091742)

[2. Requirement Specification 4](#_Toc433091743)

[i. Functional Requirements 4](#_Toc433091744)

[ii. Non-Functional Requirements 5](#_Toc433091745)

[iii. Traceability Matrix 5](#_Toc433091746)

# Introduction

The Project that our team will be working on is a web search engine.

A web search engine is a web-based tool that is designed to search and locate information on the World Wide Web. Popular examples includes Google, Yahoo! and Bing. These Search engines utilize automated software applications (referred to as robots, bots, crawlers or spiders) that travel along the Web, following links from page to page, site to site. The information gathered by the spiders is used to create a searchable index of the Web. The search results are generally presented in a line of results often referred to as search engine results pages (SERPs).

## Purpose

The purpose of our project is to develop KWIC index System (Keyword In Context) proposed by David Parnas in early 70’s using Java Applet. This System provides a convenient search mechanism for information in a long list of lines, such as book titles, or online documentation entries.

Parnas described the KWIC problem as follows:

*“The KWIC index system accepts an ordered set of lines; each line is an ordered set of words, and each word is an ordered set of characters. Any line may be “circularly shifted” by repeatedly removing the first word and appending it at the end of the line. The KWIC index system outputs a list of all circular shifts of all lines in alphabetical order.”*

In his paper of 1972, Parnas used the problem to contrast different criteria for decomposing a system into modules. Our team followed the same phenomena to implement KWIC system, by designing the system with 5 highly cohesive modules (input, line storage, circular shift, alphabetical sort, and output). We analyzed functional and nonfunctional requirement, design architecture styles, implement using Java applet and test the system. The KWIC system architecture style shall be an Abstract Data Type (ADT) style as this will provide clear object oriented structure with desire qualities of high cohesion and low coupling.

## Scope

The KWIC system shall be designed, implemented and tested to satisfy list of functional and nonfunctional requirements. Based on design specification, the system should be implemented using Java applet. Finally, we describe user manual as a guideline for using KWIC system. All the materials of project should be uploaded on our web site.

## Definitions, Acronyms, and Abbreviations

* KWIC: Key Word In Context describes a way to display related and accurate results to a specific search query.
* Architecture acronym if applicable
* UML(Unified Modeling Language): This provides a way to describe structure, behavior and architecture of application along with business process and data structure

## Project Deliverables

**Phase 1**:

Phase 1.1: Interim Project I

Deliverables: Preliminary Definition, PPT, and Presentation

Due Date: September 29th

Team Leader: Barbara Maweu

Phase 1.2: Final Part I

Deliverables: Project Report, Presentation, and Design Plans  
 Due Date: October 15th

Team Leader: Sruthi Chappidi

**Phase 2**:

Phase 2.1: Interim Project II

Deliverables: Outline, Project Plan, Presentation

Due Date: November 10th

Team Leader: Maryellen Oltman

Phase 2b: Final Part II

Deliverables: Presentation and Demo  
 Due Date: December 1st

Team Leader: Twinkle Sharma

## Requirement Specification

## Functional Requirements

Here we define the function of the KWIC system and its components. These functional requirements will drive the choice of software architecture of the KWIC system.

* FR1.0 – The KWIC system shall provide an input field to accept an ordered set of lines.
* FR2.0 – The KWIC system shall accept an ordered set of lines.
  + FR2.1 – Each ordered set of lines are an ordered set of words.
  + FR2.2 – Each ordered set of words are an ordered set of characters.
* FR3.0 – The KWIC system shall perform a “circular shift” on each the inputted ordered set of lines by repeatedly removing the first word and appending it at the end of the line.
* FR4.0 – The KWIC system shall output a listing of all circular shifts of all ordered set of lines in ascending alphabetical order.

## Non-Functional Requirements

Here we define the characteristics or quality attributes of the KWIC system.

* NFR1.0 – User-Friendly
  + Display should use easy to understand icons.
* NFR2.0 – Accuracy
  + The circular shift of each ordered set of lines must be > 95% accurate
  + The output must be a sorted result of circular shift lines
* NFR3.0 – Modifiability
  + There must be separation of concern in each process module
* NFR4.0 – Performance
  + Each input line must be output in <5 seconds

## Traceability Matrix

This traceability matrix correlates that the system through the functional requirements satisfies the non-functional requirements needed for the system.

|  |  |  |  |
| --- | --- | --- | --- |
|  | **FR1.0** | **FR2.0** | **FR3.0** |
| **NFR1.0** | X |  |  |
| **NFR2.0** | X | X | X |
| **NFR3.0** | X | X | X |
| **NFR4.0** | X | X | X |