

---

# **Vision and Scope Document**

**for**

# **Spell Checker for KnowledgeTrac**

**Version 1.0 approved**

**Prepared by Eric Engquist**

**December 8, 2004**

## Table of Contents

Table of Contents .....	ii
Revision History .....	ii
1. Business Requirements .....	1
1.1. Background .....	1
1.2. Business Opportunity .....	1
1.3. Business Objectives and Success Criteria .....	1
1.4. Customer or Market Needs .....	1
1.5. Business Risks .....	1
2. Vision of the Solution .....	2
2.1. Vision Statement .....	2
2.2. Major Features .....	2
2.3. Assumptions and Dependencies .....	3
3. Scope and Limitations .....	3
3.1. Scope of Initial Release .....	3
3.2. Scope of Subsequent Releases .....	3
3.3. Limitations and Exclusions .....	4
4. Business Context .....	4
4.1. Stakeholder Profiles .....	4
4.2. Project Priorities .....	5
4.3. Operating Environment .....	5

## Revision History

Name	Date	Reason For Changes	Version
Eric Engquist	12/08/04	Initial Revision	1.0
Mark Biddlecom	01/06/05	Refined iteration 1 requirements after discussion with stakeholders.	1.2

# **1. Business Requirements**

The business requirements provide the foundation and reference for all detailed requirements development

## **1.1. Background**

The Spell Checker for KnowledgeTrac is intended to increase the efficacy of the KnowledgeTrac search functionality by evaluating search terms submitted by the user, offering suggested replacements for misspelled terms, and to assist the user in finding search results relevant to their topic of interest.

## **1.2. Business Opportunity**

This product will be valuable to the project sponsors because it adds value to existing KnowledgeTrac search technology and makes the existing search capacities of KnowledgeTrac more useful to its end users.

## **1.3. Business Objectives and Success Criteria**

No quantifiable success criteria have been identified, but a complete solution will be effective in assisting the end user to find search results useful to them without significantly compromising the performance of the KnowledgeTrac search itself. Suggested alternate search terms will be consistently intuitive and related to the original search words or phrases submitted. The project sponsors will derive value from this project by improving the usefulness of its KnowledgeTrac product to its end users.

## **1.4. Customer or Market Needs**

KnowledgeTrac satisfies the need of an internet user to perform directory-based searches to obtain useful information as efficiently as possible. Spell Checker will provide a 'shortcut' for users who have entered inexact or vague search terms, and offer suggestions of search terms that would be useful to them.

Specifically, the completed system will satisfy the following requirements:

1. The system will run as a service or a Web service.
2. The system will load dictionary of terms at system start and sit and wait for an incoming request, then return the best user definable number of matches. The calling process will then decide how to handle the results.
3. The system will have the ability to easily add words to a dictionary.
4. The system will have the ability to handle submissions via an application or a Web page.
5. The system will more heavily weight the best match toward a business name.
6. When adding words, the system will ignore duplicates.
7. The system will have the ability to learn the 'best match' based upon past user selections
8. The system will associate a commonly misspelled word with a 'best match' until a history of selections has been built.

## **1.5. Business Risks**

No major business risks have been identified.

## 2. Vision of the Solution

This section establishes a long-term vision for the system to be built to address the business objectives. This vision will provide the context for making decisions throughout the course of the product development life cycle.

### 2.1. Vision Statement

Spell Checker for KnowledgeTrac is a software component designed to assist users of TCN's KnowledgeTrac software in refining search terms to increase their success in finding web and telephone directory results relevant to their search terms. Our software will stand alone from the larger KnowledgeTrac suite, analyze search terms, and compare them to existing dictionaries containing English words, business names, and domain specific terminology to suggest potentially more successful search terms. These dictionaries must support dynamic addition of new phrases and terms. Spell Checker's functionality will be transparent to the end user and will silently analyze search terms, only offering alternate suggestions if the search terms are clearly unlikely to return useful search results.

Time permitting, Spell Checker will support recognition of common word/term misspellings and suggestion of related topical phrases to the terms submitted. Other enhancements may be identified as the project progresses.

### 2.2. Major Features

Functional Requirements:

- Look up search terms in a dictionary
- Suggest replacements for misspelled terms (closest match)
  - Support acronyms
    - Be able to enter acronyms as search strings (results will include expanded spelling)
    - Be able to enter expanded spelling (results will include acronyms)
  - Support company/contact names
  - Support English words
    - Default language can be modified by changing loaded dictionaries
- Add new terms to dictionary
- Process phrases (as opposed to single words)
- Support multiple dictionaries
  - Generic English dictionary (by default)
  - Domain specific terminology
  - Business names / contact info
- Results should favor non-generic dictionaries

Non-functional Requirements:

- Object-oriented design to be implemented with VB.NET
- Written as a Web Service project using .NET technologies
- Adaptability
  - Must support ability to work with different data stores (e.g., SQL DB or text file)
  - Must support the addition of new components (e.g., learning and synonym finding)
- Performance
  - Analysis of a search string cannot take longer than one second.
  - Expected hardware: Multiprocessors at 2.0GHz each; 2GB RAM
- Maintainability
  - Must be able to hand off code to TCN engineers
  - Code must be readable and understandable

## 2.3. Assumptions and Dependencies

The following technologies have been identified as technical constraints. The system must be compatible with:

- Windows Server 2000
- MS IIS 5.0
- MS SQL Server 2000

The system is to be developed as a web service using VB.NET.

## 3. Scope and Limitations

The project scope defines the concept and range of the proposed solution.

### 3.1. Scope of Initial Release

An initial release has been planned to serve as a proof-of-concept for the final deliverable product. This initial release (R1) will consist of a working demonstration of the framework on which the remainder of the project will be built. R1 will include:

Iteration 1: February 1<sup>st</sup> 2005

- Complete system design for system iterations 1-3
- Instructions for installation and integration with TCN client software
- Research
  - Analysis of historic search strings and business names from TCN
  - Dictionaries (common words)
  - Word search algorithms
    - Tokenizer
    - Letter frequency
    - Degree-of-difference calculus
    - Phonetic analysis
- Software
  - Can check for the presence of a word in the dictionary using a quick binary search; will support the addition of other arbitrary algorithms and rules governing their application
  - Works from a base dictionary only, but can support an arbitrary number of dictionaries
  - Will load dictionaries into process memory to minimize processor time devoted to I/O
- Database integration
- Testing

### 3.2. Scope of Subsequent Releases

Iteration 2: February 18<sup>th</sup> 2005

- Suggest replacements for words not in the dictionary
- Addition of a new search algorithm (as yet undecided) to provide more intelligent searches
- Using multiple dictionaries
- Unit Testing for all written code

Iteration 3: March 21<sup>st</sup> 2005

- Dynamically add words/phrases to the dictionary
- Support phrase searching
- Addition of further search algorithms
- GUI Configuration tool

Iteration 4 (time permitting):

- Learning

Iteration 5 (time permitting):

- Suggest related words/phrases

### 3.3. Limitations and Exclusions

TCN will take care of getting their ASP search system to work with our web service. We will deliver a working web service.

## 4. Business Context

This section summarizes some of the business issues around the project, including profiles of major customer categories, assumptions that went into the project concept, and the management priorities for the project.

### 4.1. Stakeholder Profiles

Stakeholders are individuals, groups, or organizations that are actively involved in a project, are affected by its outcome, or can influence its outcome. The stakeholder profiles identify the customers for this product and other stakeholders, and states their major interests in the product.

<b>Stakeholder</b>	<b>Major Value</b>	<b>Attitudes</b>	<b>Major Interests</b>	<b>Constraints</b>
<i>TCN executives</i>	<i>increased revenue</i>	<i>See product as avenue to increase in market share</i>	<i>richer feature set than competitors; time to market</i>	<i>N/A</i>
<i>TCN engineers</i>	<i>Integration with existing software</i>	<i>Desire a product that is intuitive, maintainable, and customizable</i>	<i>Good documentation, no undue complexity, and reliability of services</i>	<i>Must be developed with technologies engineers are familiar with</i>
<i>RIT Senior Project Team</i>	<i>Educational value</i>	<i>Consider project an opportunity to apply academic and co-op expertise</i>	<i>Completion of objectives using learned software engineering disciplines</i>	<i>Time (20-week senior project duration)</i>
<i>RIT staff advisor</i>	<i>Harmony between project team and sponsors</i>	<i>Receptive as resource for project success</i>	<i>Timely completion of project and meeting of all sponsor objectives</i>	<i>N/A</i>
<i>TCN customers</i>	<i>Utility of Spell Checked searches</i>	<i>Require ease of use and maximization of usefulness</i>	<i>Software must be simple to use and provide easily</i>	<i>Unpredictability of search input</i>

			<i>perceived search value</i>	
--	--	--	-------------------------------	--

## 4.2. Project Priorities

<i>Dimension</i>	<i>Driver (state objective)</i>	<i>Constraint (state limits)</i>	<i>Degree of Freedom (state allowable range)</i>
<i>Schedule</i>	<i>Release 1 to be available by 2/1/04, future releases as described in section 3.2</i>		
<i>Features</i>		<i>100% of base functionality must be implemented in R1; other functionality to be further refined</i>	
<i>Adaptability</i>		<i>Spell Checker must be modular and support addition of supplemental database formats and functionality</i>	
<i>Maintainability</i>		<i>Spell Checker must be developed to be easily modified and maintained by TCN staff</i>	
<i>Performance</i>			<i>Spell Checker must not delay existing search functionality by more than several seconds</i>
<i>Staff</i>		<i>Team size is 5 developers</i>	
<i>Time</i>			<i>Schedule slippage of more than 48 hours requires schedule revision</i>

## 4.3. Operating Environment

This system will operate on a Windows Server 2000 machine.