**Document Control**

**Approval**

The Guidance Team and the customer will approve this document.

**Document Change Control**

<table>
<thead>
<tr>
<th>Initial Release:</th>
<th>1.0</th>
</tr>
</thead>
<tbody>
<tr>
<td>Current Release:</td>
<td>1.9</td>
</tr>
<tr>
<td>Indicator of Last page in Document:</td>
<td>Page 36</td>
</tr>
<tr>
<td>Date of Last Review:</td>
<td>2/16/2014</td>
</tr>
<tr>
<td>Date of Next Review:</td>
<td>2/21/2014</td>
</tr>
<tr>
<td>Target Date for Next Update:</td>
<td>2/21/2014</td>
</tr>
</tbody>
</table>

**Distribution List**

This following list of people will receive a copy of this document every time a new version of this document becomes available:

- **Guidance Team:** Dr. Ann Gates
  Elsa Tai
  Aditi Barua

- **Customer:** Dr. Nicholas Del Rio

- **Software Teams:** TBA
  Leaf Development
  Development Technologies
  Atom
  Team 5
  Cases

**Change Summary**

The following table details changes made between versions of this document:

<table>
<thead>
<tr>
<th>Version</th>
<th>Date</th>
<th>Modifier</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.0</td>
<td>2/10/2014</td>
<td>O. Ochoa</td>
<td>Initial creation and edits</td>
</tr>
<tr>
<td>1.1</td>
<td>2/12/2014</td>
<td>N. Del Rio</td>
<td>Interface screen shots</td>
</tr>
<tr>
<td>1.2</td>
<td>2/15/2014</td>
<td>O. Ochoa</td>
<td>Updated use case diagram</td>
</tr>
<tr>
<td>1.3</td>
<td>2/15/2014</td>
<td>N. Del Rio</td>
<td>Added scenarios</td>
</tr>
<tr>
<td>1.4</td>
<td>2/15/2014</td>
<td>O. Ochoa</td>
<td>Edits and requirements related to the interface</td>
</tr>
<tr>
<td>1.5</td>
<td>2/16/2014</td>
<td>N. Del Rio</td>
<td>Added recover password and activation specs</td>
</tr>
<tr>
<td>1.6</td>
<td>2/16/2014</td>
<td>O. Ochoa</td>
<td>Stimulus Requirements</td>
</tr>
<tr>
<td>1.7</td>
<td>2/16/2014</td>
<td>N. Del Rio</td>
<td>Functional and Software Interface Requirements</td>
</tr>
<tr>
<td>1.8</td>
<td>2/16/2014</td>
<td>N. Del Rio</td>
<td>Related Real World Objects</td>
</tr>
<tr>
<td>1.9</td>
<td>2/6/2014</td>
<td>O. Ochoa</td>
<td>Functional Requirements Edits</td>
</tr>
</tbody>
</table>
# TABLE OF CONTENTS

**DOCUMENT CONTROL** ........................................................................ II

- APPROVAL .................................................................................................. II
- DOCUMENT CHANGE CONTROL .......................................................... II
- DISTRIBUTION LIST ................................................................................ II
- CHANGE SUMMARY .................................................................................. II

## 1. INTRODUCTION .......................................................... 4

- 1.1. PURPOSE AND INTENDED AUDIENCE ........................................... 4
- 1.2. SCOPE OF PRODUCT ....................................................................... 4
- 1.3. DEFINITIONS, ACRONYMS, AND ABBREVIATIONS ..................... 4
  - 1.3.1. Definitions .................................................................................. 4
  - 1.3.2. Acronyms .................................................................................. 5
  - 1.3.3. Abbreviations ............................................................................ 6
- 1.4. OVERVIEW ...................................................................................... 6
- 1.5. REFERENCES ................................................................................... 6

## 2. GENERAL DESCRIPTION .................................................. 7

- 2.1. PRODUCT PERSPECTIVE ................................................................ 7
- 2.2. PRODUCT FEATURES ..................................................................... 7
- 2.3. ACTORS ............................................................................................ 8
- 2.4. USE CASES ..................................................................................... 9
- 2.5. VIKO API ......................................................................................... 9
- 2.6. SCENARIOS .................................................................................... 9
- 2.7. USER CHARACTERISTICS ............................................................ 19
- 2.8. GENERAL CONSTRAINTS ............................................................... 19
- 2.9. ASSUMPTIONS AND DEPENDENCIES ......................................... 19

## 3. SPECIFIC REQUIREMENTS .............................................. 20

- 3.1. EXTERNAL INTERFACE REQUIREMENTS .................................... 20
  - 3.1.1. User Interfaces ......................................................................... 20
  - 3.1.2. Hardware Interfaces ................................................................. 40
  - 3.1.3. Software Interfaces ................................................................. 40
  - 3.1.4. Communications Interfaces .................................................... 41
- 3.2. BEHAVIORAL REQUIREMENTS .................................................. 41
  - 3.2.1. Same Class of User .................................................................. 42
  - 3.2.2. Related Real-world Objects ...................................................... 42
  - 3.2.3. Stimulus .................................................................................... 42
  - 3.2.4. Related Features ..................................................................... 52
  - 3.2.5. Functional ................................................................................ 52
- 3.3. NON-BEHAVIORAL REQUIREMENTS .......................................... 54
  - 3.3.1. Performance Requirements .................................................... 54
  - 3.3.2. Qualitative Requirements ......................................................... 54
  - 3.3.3. Design and Implementation Constraints .................................. 55
- 3.4. OTHER REQUIREMENTS ............................................................ 55
  - 3.4.1. Operations ................................................................................ 55
  - 3.4.2. Site Adaptation ....................................................................... 55

## 4. APPENDIX DIAGRAMS .................................................. 57
1. Introduction

1.1. Purpose and Intended Audience

The purpose of this document, the Software Requirements Specifications (SRS), is to provide a clear and precise description of the functionality and behavior that the Visualization Knowledge (VisKo) system must have. The intended audiences of this document are the client, the guidance team and the software engineering teams. This document acts as an agreement between the intended audiences over the requirements of the VisKo system.

1.2. Scope of Product

The VisKo system is a framework that supports the answering of visualization queries that describe what visualizations users want generated rather than specifying how they should be generated. Relying on a knowledge base of visualization services, VisKo can translate user queries into equivalent visualization pipelines that generate the visualization specified in the query. Therefore, users of the VisKo system can generate visualizations while remaining hidden from the complexities associated with visualization pipeline assembly, a practice that encompasses the themes: visualization transformation theory, computer graphics, semiotics, and performance. Currently, VisKo is only accessible through an application-programming interface and requires a more appropriate user interface. Therefore, it is desirable that VisKo be improved by adding a set of perspectives that can support different interfaces and related functionality that would greatly assist in the widespread adoption of VisKo.

1.3. Definitions, Acronyms, and Abbreviations

This section describes definitions, acronyms and abbreviations used throughout this document.

1.3.1. Definitions

Provided below is a list of terms and definitions that may be required, or be helpful to the intended audience.

<table>
<thead>
<tr>
<th>TERM</th>
<th>DEFINITION</th>
</tr>
</thead>
<tbody>
<tr>
<td>Application Programming Interface (API)</td>
<td>An application-programming interface describes the names and signatures of a software or hardware system with the purpose of describing how to invoke or use it.</td>
</tr>
<tr>
<td>Regular User</td>
<td>A user that comes to the system to get desired visualizations based on some given data.</td>
</tr>
<tr>
<td>Data Flow Diagram</td>
<td>A diagram used to model functions of an informative system and how data flows from function to function.</td>
</tr>
<tr>
<td>Deployment</td>
<td>An activity that allows a user to prepare a software system for use.</td>
</tr>
<tr>
<td>Feasibility Report</td>
<td>A document that ensures a project can be complete through the dimensions of technology, finance, time, and resources.</td>
</tr>
<tr>
<td>Knowledge Base</td>
<td>A knowledge base is composed of visualization toolkit operators including rules and facts for how operators are chained together to form pipelines.</td>
</tr>
<tr>
<td>Listbox</td>
<td>A Graphical User Interface (GUI) element that allows the user to select one or more items from a list contained within a static, multiple line text box.</td>
</tr>
<tr>
<td>Navigation Side Bar</td>
<td>A user interface element within a webpage that contains links to other sections of the website.</td>
</tr>
<tr>
<td>Object</td>
<td>An instance of a class, used to interact with one another to design applications and computer programs</td>
</tr>
<tr>
<td>Pipeline</td>
<td>An ordered sequence of executable web services that generate visualizations. The VisKo API translates user submitted Visualization Queries into sets of Pipelines.</td>
</tr>
<tr>
<td><strong>Privileged User</strong></td>
<td>A user that is defined as the owner of the system. A privileged user sets up the system as well as manages the system through various tasks.</td>
</tr>
<tr>
<td>--------------------</td>
<td>----------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td><strong>Provenance Information</strong></td>
<td>Intermediate information that tells the origin or source of the content, any changes that may have taken place since it was originated, and who has had custody of it since it was originated.</td>
</tr>
<tr>
<td><strong>Query used</strong></td>
<td>A special language that allows the user to say what visualization they want generated. A query gets translated into a pipeline.</td>
</tr>
<tr>
<td><strong>Semiotics</strong></td>
<td>A general philosophical theory of signs and symbols that deals especially with their function in both artificially constructed and natural languages and comprises syntactic, semantics, and pragmatics</td>
</tr>
<tr>
<td><strong>Services</strong></td>
<td>A web-based tool kit used to generate visualizations.</td>
</tr>
<tr>
<td><strong>SPARQL</strong></td>
<td>A query language able to retrieve and manipulate data stored in a Resource Description Framework.</td>
</tr>
<tr>
<td><strong>Structured Query Language (SQL)</strong></td>
<td>A special-purpose programming language designed for managing data held in a relational database management system.</td>
</tr>
<tr>
<td><strong>Syntax highlighting</strong></td>
<td>A fully functional self-contained code syntax highlighter developed in JavaScript.</td>
</tr>
<tr>
<td><strong>Toggle Switch</strong></td>
<td>A switch that allow the change from one setting to another.</td>
</tr>
<tr>
<td><strong>Unique identifier</strong></td>
<td>A numeric or alphanumeric string that is associated with a single entity within the VisKo system.</td>
</tr>
<tr>
<td><strong>Use Case Diagram</strong></td>
<td>A description of sequence of interactions between a system and an external actor. A Use Case is a stand-alone activity that an actor can perform to achieve some outcome of value.</td>
</tr>
<tr>
<td><strong>User</strong></td>
<td>A negotiator, a human agent, who uses or will use the VisKo system. Two different types; privileged user and registered user.</td>
</tr>
<tr>
<td><strong>Visualization Knowledge (VisKo)</strong></td>
<td>A system that enables users to generate visualizations, such as graphs or static images, by submitting queries.</td>
</tr>
<tr>
<td><strong>Visualization Abstraction</strong></td>
<td>A visualization geometry such as isosurfaces, isolines (i.e., contour lines), points, volumes, polygons, grids, etc.</td>
</tr>
<tr>
<td><strong>VisKo API</strong></td>
<td>A VisKo application-programming interface version 7.0 that specifies how specific hardware components work with each other.</td>
</tr>
<tr>
<td><strong>Visualization Query</strong></td>
<td>A formal request for the generation of a visualization described in terms of a Visualization Abstraction, Viewer Set, Input Format, and Input Data Type. A query and a pipeline provide equal forms of expressivity, but the query is declarative.</td>
</tr>
<tr>
<td><strong>Visualization Type</strong></td>
<td>A finished visualization of a query.</td>
</tr>
<tr>
<td><strong>Visualization Pipeline</strong></td>
<td>A process of creating visual representations of data.</td>
</tr>
<tr>
<td><strong>Visualization Viewer</strong></td>
<td>A is a program that presents visualizations.</td>
</tr>
<tr>
<td><strong>Visualization Viewer Set</strong></td>
<td>A group of viewers.</td>
</tr>
<tr>
<td><strong>Visualization</strong></td>
<td>A presentation of information that is being presented through graphs and static images, depending on the service. Visualizations are generated by queries in this context.</td>
</tr>
<tr>
<td><strong>Visualization (Viz) Wall</strong></td>
<td>A matrix of 45 40-inch computer monitors stacked five high and nine across that is capable of displaying 93-megapixel images. Located at the Cyber-Share at UTEP.</td>
</tr>
<tr>
<td><strong>Wrapped services</strong></td>
<td>Services that can be added to the system</td>
</tr>
</tbody>
</table>

### 1.3.2. Acronyms

Listed below are the acronyms used in this document and their associated definitions.
<table>
<thead>
<tr>
<th>TERM</th>
<th>DEFINITION</th>
</tr>
</thead>
<tbody>
<tr>
<td>API</td>
<td>Application Programming Interface</td>
</tr>
<tr>
<td>DFD</td>
<td>Data flow diagram</td>
</tr>
<tr>
<td>FAQ</td>
<td>Frequently Asked Questions</td>
</tr>
<tr>
<td>GB</td>
<td>Gigabyte</td>
</tr>
<tr>
<td>GHz</td>
<td>GigaHertz</td>
</tr>
<tr>
<td>GMT</td>
<td>Generic Mapping Tools</td>
</tr>
<tr>
<td>GUI</td>
<td>Graphical User Interface</td>
</tr>
<tr>
<td>HEX</td>
<td>A hexadecimal representation of a color.</td>
</tr>
<tr>
<td>HTML</td>
<td>Hyper-text Markup Language</td>
</tr>
<tr>
<td>IDE</td>
<td>Integrated Development Environment</td>
</tr>
<tr>
<td>SQL</td>
<td>Structured Query Language</td>
</tr>
<tr>
<td>SRS</td>
<td>Software Requirements Specification</td>
</tr>
<tr>
<td>UML</td>
<td>Unified Modeling Language</td>
</tr>
<tr>
<td>UTEP</td>
<td>University of Texas at El Paso</td>
</tr>
<tr>
<td>VisKo</td>
<td>Visualization Knowledge</td>
</tr>
<tr>
<td>VTK</td>
<td>Visualization Toolkit</td>
</tr>
</tbody>
</table>

1.3.3. Abbreviations

Provided below is a list of used abbreviations and their associated definitions in this document.

<table>
<thead>
<tr>
<th>TERM</th>
<th>DEFINITION</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fig</td>
<td>Figure</td>
</tr>
<tr>
<td>Viz</td>
<td>Visualization</td>
</tr>
<tr>
<td>Req</td>
<td>Requirement</td>
</tr>
<tr>
<td>e.g.</td>
<td>For example</td>
</tr>
<tr>
<td>i.e.</td>
<td>That is</td>
</tr>
</tbody>
</table>

1.4. Overview

The Software Requirement Specifications document is divided into three main sections: Introduction, General Description, and Specific Requirements. The first section introduces the purpose of the document, the intended audience and definitions, acronyms and abbreviations used throughout this document. The second section describes in general the product perspective and features, the characteristics of the users, general constrains and assumptions and dependencies within the VisKo system. The last section describes requirements for external interfaces, behavioral requirements, non-behavioral requirements and other requirements.

1.5. References

2. **General Description**

2.1. **Product Perspective**

The VisKo approach and resulting application programming interface (API) is founded on a visualization query language that was designed with the goal of hiding the associated complexities of visualization pipeline construction. By posing visualization queries, users can declaratively specify what visualizations they want generated from their data and quickly explore a variety of results that may only be generated using complex combinations of modules possibly supported across different toolkits.

The VisKo system described in this document provides a graphical user interface to the existing VisKo API as well as capabilities for managing statistical information regarding usage of the system.

2.2. **Product Features**

The following figure depicts a high-level use case diagram for the VisKo system. The use case diagram models the interactions between different actors, which are outside entities, and the system. Each oval describes a use of the system and the lines refer to actor interactions. The dashed boxes around use cases signify that each of those contained use cases are related to any other use case connected to the dashed box.
section 2.3. **Actors**

The three main actors of the system are as follows:

- **Privileged User**: A privileged user is defined as the owner of the system. The privileged user is able to see the activities executed by a common user, and flag invalid services. Privileged users may view statistics detailing the performance and usage of the VisKo system.

- **Regular Users**: A regular user is defined as a user that uses the generation facilities of the VisKo system. They submit queries for the system to assemble and generate visualizations.

- **VisKo API**: The VisKo API is the Application Programming Interface that allows the system to generate visualizations and register services. It is the core of the system and maintains a knowledge base of all registered Visualization Services and pipeline composition rules.
2.4. Use Cases

The following are each of the use cases of the VisKo system:

- **Log in**: An include use case of the system; this permits users to log in to VisKo.
- **Access Database**: This use case models the use of database storage within the system.
- **Visualize**: Regular users submit visualization queries to the system. The system, in turn, returns a set of executable pipelines that are configured to generate visualizations that satisfy the query constraints.
- **Manage Service**: Regular users add, edit, or delete visualization services, which include: Viewer Sets, Viewers, Transformers, Converters, Filter, and Mappers.
- **Search History**: Regular Users search for their previously generated visualizations.
- **Configure Account**: Regular Users edit settings of their account, including password and registered email address.
- **Register**: This use case is an extension of Login; guests have the option to register with the VisKo system.
- **Perform Search Task**: Privileged users can search regular users’ submitted queries, executed pipelines, registered services, and accounts.
- **Perform Analyze Task**: Privileged users can access statistical information regarding regular users’ activities, submitted queries, and executed pipelines.
- **Control Task**: Activate and suspend user accounts and registered services.

2.5. VisKo API

The VisKo API is available as a set of Java JAR files. Instructions for how to download, build, and interface with the VisKo API can be found at the following GitHub repository: https://github.com/nicholasdelrio/visko.

2.6. Scenarios

The following section describes the scenarios for the VisKo system.

2.6.1.1. Privileged User Registration Scenario

<table>
<thead>
<tr>
<th>Use Case Name</th>
<th>Register</th>
</tr>
</thead>
<tbody>
<tr>
<td>Actors</td>
<td>Privileged User</td>
</tr>
<tr>
<td>Brief Description</td>
<td>This scenario describes how a privileged user can create an account the VisKo system. There can be many privileged users but they must all register from the same physical machine that the VisKo system is running.</td>
</tr>
<tr>
<td>Pre-Condition</td>
<td>The system is published on the Web</td>
</tr>
<tr>
<td>Trigger Condition</td>
<td>The regular user clicks the Register button on the guest navigation window and the IP of the request is the same as the IP of the machine hosting the VisKo system.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Normal Flow of Events:</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. The system presents a new user registration form</td>
</tr>
<tr>
<td>2. The user completes and submits the form</td>
</tr>
<tr>
<td>3. The system verifies that the form is complete and the submitted information is valid</td>
</tr>
<tr>
<td>4. The system creates a new “suspended” privileged user account in the database (ALT 1)(ALT 2)</td>
</tr>
<tr>
<td>5. The system sends an “activate account” email to the registered user containing an activate account link request</td>
</tr>
<tr>
<td>6. End of use case</td>
</tr>
</tbody>
</table>

(ALT 1)
## 2.6.1.2. Regular User Registration Scenario

<table>
<thead>
<tr>
<th>Use Case Name:</th>
<th>Register</th>
</tr>
</thead>
<tbody>
<tr>
<td>Actors:</td>
<td>Regular User</td>
</tr>
<tr>
<td>Brief Description:</td>
<td>This scenario describes how a regular user can create an account the VisKo system.</td>
</tr>
<tr>
<td>Pre-Condition:</td>
<td>The system is published on the Web</td>
</tr>
<tr>
<td>Trigger Condition:</td>
<td>The regular user clicks the Register button on the guest navigation window</td>
</tr>
</tbody>
</table>

**Normal Flow of Events:**
1. The system presents a new user registration form
2. The user completes and submits the form
3. The system verifies that the form is complete and the submitted information is valid
4. They system creates a new “suspended” regular user account in the database (ALT 1)(ALT 2)
5. The system sends an “activate account” email to the registered user containing an activate account link request
6. End of use case

(ALT 1)
1.1. The system responds indicating that required fields are missing
1.2. End of use case

(ALT 2)
2.1. The system responds indicating the information violates a condition (e.g., email is invalid, confirmation email does not match, confirmation password does not match)
2.2. End of use case

## 2.6.1.3. Account Activation Scenario

<table>
<thead>
<tr>
<th>Use Case Name:</th>
<th>Register</th>
</tr>
</thead>
<tbody>
<tr>
<td>Actors:</td>
<td>Regular User, Privileged User</td>
</tr>
<tr>
<td>Brief Description:</td>
<td>This scenario describes how a regular and privileged user can activate a new account.</td>
</tr>
<tr>
<td>Pre-Condition:</td>
<td>The system is published on the Web and the user has created an account.</td>
</tr>
<tr>
<td>Trigger Condition:</td>
<td>The user submits an activate account request</td>
</tr>
</tbody>
</table>

**Normal Flow of Events:**
7. The system updates the user’s status from suspended to activated (ALT 1)
8. The system presents the “activated account” page
9. End of use case

(ALT 1)
1.1. The system indicates that the account is already in an activated state
1.2. End of use case
### 2.6.1.4. Log in Scenario

**Use Case Name:** Log in  
**Actors:** Regular User, Privileged User  
**Brief Description:** This scenario describes how a regular and privileged user can log into the VisKo system.

**Pre-Condition:** The regular and privileged users have existing accounts.  
**Trigger Condition:** The user clicks the Login button on the guest navigation window

**Normal Flow of Events:**  
1. The system authenticates the user based on the provided email and password (ALT 1)  
2. The system identifies the account as being regular (ALT 2)  
3. The system displays the regular user home page  
4. End of use case  

(ALT 1)  
1.1. The system presents a message indicating that the required input email and/or password is missing  
1.2. End of use case  

(ALT 2)  
2.1. The system identifies the account as being privileged  
2.2. The system displays the privileged user home page  
2.3. End of use case

### 2.6.1.5. Search History Scenario

**Use Case Name:** Search History  
**Actors:** Regular User  
**Brief Description:** This scenario describes how a regular user accesses his/her visualization generation history.

**Pre-Condition:** The regular user needs to be logged into the system.  
**Trigger Condition:** The regular user clicks the Search History link on the regular user navigation bar

**Normal Flow of Events:**  
5. The system presents visualization search options to the user  
6. The user specifies search criteria by configuring search options  
7. The user requests for a search based on the specified criteria  
8. The system searches the user’s visualization history in the database for matches  
9. The system responds with a set of visualizations that satisfy the search criteria (ALT 1)  
10. The user selects a visualization search result  
11. They system presents the details associated with the selected result  
12. End of use case  

(ALT 1)  
1.1. The system responds indicating that no results were found that satisfied the search criteria  
1.2. End of use case
2.6.1.6. Visualize Scenario

<table>
<thead>
<tr>
<th>Use Case Name:</th>
<th>Visualize</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Actors:</strong></td>
<td>Regular User, VisKo API</td>
</tr>
<tr>
<td><strong>Brief Description:</strong></td>
<td>This scenario describes how a regular user generates visualizations in the VisKo System.</td>
</tr>
<tr>
<td><strong>Pre-Condition:</strong></td>
<td>A regular user needs to be logged into the system. The associated user account must be in an “activated” state.</td>
</tr>
<tr>
<td><strong>Trigger Condition:</strong></td>
<td>A regular user clicks the Visualize link on the regular user navigation bar.</td>
</tr>
</tbody>
</table>

**Normal Flow of Events:**

1. The user opts to input a visualization query manually (ALT 1)
2. A user submits a visualization query
3. The system logs the input query
4. The system requests a set of pipelines from the VisKo API, given the user input query
5. The system removes any pipelines containing services flagged as “suspended,” or “deleted”
6. The system responds by presenting a set of visualization pipelines to the user (ALT 2) (ALT 3)
7. A user selects a set of response pipelines to be executed
8. The system logs the pipelines selected for execution in the database
9. The system logs execution details of every service involved in a pipeline execution, including errors, in the database
10. The system initiates the execution of pipelines in the order specified by the user
11. The system logs all visualization results of the completed pipeline executions
12. The system presents a thumbnail of all visualizations generated by a pipeline (ALT 4)
13. A user selects a resultant visualization thumbnail
14. The system presents a full-sized version of the visualization result
15. End of use case

(ALT 1)
1.1. The user opts for query generation guidance
1.2. The user selects a visualization abstraction
1.3. The system presents the set of visualization query parameters
1.4. The user configures the set of visualization query parameters
1.5. The user submits the parameter configurations
1.6. Go to 3

(ALT 2)
2.1. The system determines the query to be syntactically incorrect
2.2. The system responds with a message stating the query is malformed
2.3. End of use case

(ALT 3)
3.1. The system is unable to derive pipelines that can satisfy the query
3.2. They system responds with a message stating that no query solution is possible
3.3. End of use case

(ALT 4)
4.1. A pipeline execution fails
4.2. The system logs the service the pipeline failed on as well as the error: execution error or time out
4.3. The system indicates that the pipeline execution has failed
4.4. End of use case
2.6.1.7. Register Executable Pipeline Component

**Use Case Name:** Manage Service  
**Actors:** Regular User, VisKo API  
**Brief Description:** This scenario describes how a regular user registers Transformers, Filters, Converters, and Mappers, which are supported by a WSDL service. Therefore, it is assumed that users know the URL to the WSDL describing their service.

**Pre-Condition:** The regular user needs to be logged into the system. The associated user account must be in an "activated" state.

**Trigger Condition:** The regular user clicks the Manager Services link on the regular user navigation bar

**Normal Flow of Events:**
1. The system presents a set of service types to be registered
2. The user selects to register a Transformer (ALT 1) (ALT 2) (ALT 3) (ALT 5)
3. The system presents a Transformer form for the user to complete, which includes a WSDL URL input
4. The user submits the completed form
5. The system submits the form information to the VisKo API (ALT 4)
6. The system logs the form information in the database
7. The system indicates that the registration was successful
8. End of use case

(ALT 1)
1.1. The user selects to register a Filter
1.2. The system presents a Filter form for the user to complete, which includes a WSDL URL input
1.3. Go to 4

(ALT 2)
2.1. The user selects to register a Converter
2.2. The system presents a Converter form for the user to complete, which includes a WSDL URL input
2.3. Go to 4

(ALT 3)
3.1. The user selects to register a Mapper
3.2. The system presents a Mapper form for the user to complete, which includes a WSDL URL input
3.3. Go to 4

(ALT 4)
4.1. The VisKo API rejects the registration request due to an error processing the WSDL document
4.2. The system displays a message indicating that there was an error processing the WSDL document
4.3. End of use case

2.6.1.8. Edit Executable Pipeline Component

**Use Case Name:** Manage Service  
**Actors:** Regular User, VisKo API  
**Brief Description:** This scenario describes how a regular user edits previously registered Transformers, Filters, Converters, and Mappers, which are supported by a WSDL service. Therefore, it is assumed that users know the URL to the WSDL describing their service.

**Pre-Condition:** The regular user needs to be logged into the system. The associated user account must be in an "activated" state.

**Trigger Condition:** The regular user clicks the Manager Services link on the regular user navigation bar
## Normal Flow of Events:

1. The system presents a set of services previously registered by the user
2. The user selects to edit a Transformer (ALT 1) (ALT 2) (ALT 3) (ALT 5)
3. The system presents a pre-populated Transformer form for the user to modify
4. The user submits the modified form
5. The system submits the form information to the VisKo API (ALT 4)
6. The system logs the form information in the database
7. The system indicates that the registration was successful
8. End of use case

(ALT 1)
1.1. The user selects to edit a Filter
1.2. The system presents a pre-populated Filter form for the user to modify
1.3. Go to 4

(ALT 2)
2.1. The user selects to edit a Converter
2.2. The system presents a pre-populated Converter form for the user to modify
2.3. Go to 4

(ALT 3)
3.1. The user selects to edit a Mapper
3.2. The system presents a pre-populated Mapper form for the user to modify
3.3. Go to 4

(ALT 4)
4.1. The VisKo API rejects the registration request due to an error processing the WSDL document
4.2. The system displays a message indicating that there was an error processing the WSDL document
4.3. End of use case

### 2.6.1.9. Register Non-executable Pipeline Component

<table>
<thead>
<tr>
<th>Use Case Name:</th>
<th>Manage Service</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Actors:</strong></td>
<td>Regular User, VisKo API</td>
</tr>
<tr>
<td><strong>Brief Description:</strong></td>
<td>This scenario describes how a regular user registers Viewer Sets and Viewers, which are not executable although necessary for the VisKo API to construct pipelines.</td>
</tr>
<tr>
<td><strong>Pre-Condition:</strong></td>
<td>The regular user needs to be logged into the system. The associated user account must be in an “activated” state.</td>
</tr>
<tr>
<td><strong>Trigger Condition:</strong></td>
<td>The regular user clicks the Manager Services link on the regular user navigation bar</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Normal Flow of Events:</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. The system presents a set of service types to be registered</td>
</tr>
<tr>
<td>2. The user selects to register a Viewer Set (ALT 1)</td>
</tr>
<tr>
<td>3. The system presents a Viewer Set form for the user to complete</td>
</tr>
<tr>
<td>4. The user submits the completed form</td>
</tr>
<tr>
<td>5. The system submits the form information to the VisKo API</td>
</tr>
<tr>
<td>6. The system logs the form information in the database</td>
</tr>
<tr>
<td>7. The system displays a message indicating that registration was successful</td>
</tr>
</tbody>
</table>

(ALT 1)
1.1. The user selects to register a Viewer
1.2. The system presents a Viewer form for the user to complete
1.3. Go to 4
2.6.1.10. Edit Non-executable Pipeline Component

**Use Case Name:** Manage Service

**Actors:** Regular User, VisKo API

**Brief Description:** This scenario describes how a regular user edits previously registered Viewer Sets and Viewers, which are not executable although necessary for the VisKo API to construct pipelines.

**Pre-Condition:** The regular user needs to be logged into the system. The associated user account must be in an “activated” state.

**Trigger Condition:** The regular user clicks the Manager Services link on the regular user navigation bar

**Normal Flow of Events:**
1. The system presents a set of services previously registered by the user
2. The user selects to register a Viewer Set (ALT 1)
3. The system presents a pre-populated Viewer Set form for the user to modify
4. The user submits the modified form
5. The system submits the form information to the VisKo API
6. The system logs the form information in the database
7. The system displays a message indicating that registration was successful

(ALT 1)
1.1. The user selects to register a Viewer
1.2. The system presents a pre-populated Viewer form for the user to modify
1.3. Go to 4

2.6.1.11. Update Password

**Use Case Name:** Configure Account

**Actors:** Regular User

**Brief Description:** This scenario describes how a regular updates their password

**Pre-Condition:** The regular user needs to be logged into the system. The associated user account must be in an “activated” state.

**Trigger Condition:** The regular user clicks the Configure Account link on the regular user navigation bar

**Normal Flow of Events:**
1. The system presents a fields for updating username and password
2. The user updates the password
3. They system responds indicating the update was a success (ALT 1)

(ALT 1)
2.1. The system responds indicating the new password does not match the new password confirmation
2.2. End of use case

2.6.1.12. Update Email Address

**Use Case Name:** Configure Account

**Actors:** Regular User

**Brief Description:** This scenario describes how a regular updates their email address.

**Pre-Condition:** The regular user needs to be logged into the system. The associated user account must be in an “activated” state.
### Trigger Condition
The regular user clicks the Configure Account link on the regular user navigation bar

### Normal Flow of Events:

<table>
<thead>
<tr>
<th>Step</th>
<th>Event</th>
</tr>
</thead>
<tbody>
<tr>
<td>4.</td>
<td>The system presents a fields for updating username and password</td>
</tr>
<tr>
<td>5.</td>
<td>The user updates the email</td>
</tr>
<tr>
<td>6.</td>
<td>They system responds indicating the update was a success (ALT 1) (ALT 2)</td>
</tr>
</tbody>
</table>

(ALT 1)

1. The system responds indicating the new email does not match the new email confirmation
2. End of use case

(ALT 2)

1. The system responds indicating the new email is not valid
2. End of use case

---

#### 2.6.1.13 Search Queries Scenario

**Use Case Name:** Perform Search Tasks

**Actors:** Privileged User

**Brief Description:** This scenario describes how a privileged user searches for previously posed queries

**Pre-Condition:** The privileged user needs to be logged into the system.

**Trigger Condition:** The privileged user clicks the Search Queries link on the privileged user navigation bar

### Normal Flow of Events:

<table>
<thead>
<tr>
<th>Step</th>
<th>Event</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>The system presents query search options to the user</td>
</tr>
<tr>
<td>2.</td>
<td>The user specifies search criteria by configuring search options</td>
</tr>
<tr>
<td>3.</td>
<td>The user requests for a search based on the specified criteria</td>
</tr>
<tr>
<td>4.</td>
<td>The system searches the logs of every users’ query submission in the database for matches</td>
</tr>
<tr>
<td>5.</td>
<td>The system responds with a set of past-submitted queries that satisfy the search criteria (ALT 1)</td>
</tr>
<tr>
<td>6.</td>
<td>The user selects a query search result</td>
</tr>
<tr>
<td>7.</td>
<td>They system presents the query text of the selected query</td>
</tr>
<tr>
<td>8.</td>
<td>End of use case</td>
</tr>
</tbody>
</table>

(ALT 1)

1.1. The system responds indicating that no results were found that satisfied the search criteria
1.2. End of use case

---

#### 2.6.1.14 Search Pipelines Scenario

**Use Case Name:** Perform Search Tasks

**Actors:** Privileged User

**Brief Description:** This scenario describes how a privileged user searches for previously executed pipelines

**Pre-Condition:** The privileged user needs to be logged into the system.

**Trigger Condition:** The privileged user clicks the Search Pipelines link on the privileged user navigation bar

---
Normal Flow of Events:

9. The system presents pipelines search options to the user
10. The user specifies search criteria by configuring search options
11. The user requests for a search based on the specified criteria
12. The system searches the logs of every users’ associated pipeline executions in the database for matches
13. The system responds with a set of past-executed pipelines that satisfy the search criteria (ALT 1)
14. The user selects a pipeline search result
15. They system presents the details associated with the pipeline search result
16. End of use case

(ALT 1)
1.1. The system responds indicating that no results were found that satisfied the search criteria
1.2. End of use case

2.6.1.15. Search Services Scenario

Use Case Name: Perform Search Tasks

Actors: Privileged User

Brief Description: This scenario describes how a privileged user searches for registered services and modifies their accessibility status

Pre-Condition: The privileged user needs to be logged into the system.

Trigger Condition: The privileged user clicks the Search Services link on the privileged user navigation bar

Normal Flow of Events:

1. The system presents services-related search options to the user
2. The user specifies search criteria by configuring search options
3. The user requests for a search based on the specified criteria
4. The system searches the logs of every users’ registration records in the database
5. The system responds with a set of registered services that satisfy the search criteria (ALT 1)
6. The user selects to view the details of a search result (ALT 2)(ALT 3)
7. They system presents the details associated with the service search result
8. End of use case

(ALT 1)
1.1. The system responds indicating that no results were found that satisfied the search criteria
1.2. End of use case

(ALT 2)
2.1. The user selects to toggle the status of a service from “suspended” to “active” or “active” to “suspended”
2.2. The system tags the new status of the service in the database
2.3. End of use case

(ALT 3)
3.1. The user selects to delete a service
3.2. The system tags the new status of the service in the database
3.3. End of use case

2.6.1.16. Search Users Scenario

Use Case Name: Perform Search Tasks

Actors: Privileged User
**Brief Description:** This scenario describes how a privileged user searches for regular users and modifies their account status.

**Pre-Condition:** The privileged user needs to be logged into the system.

**Trigger Condition:** The privileged user clicks the Search Users link on the privileged user navigation bar.

**Normal Flow of Events:**

9. The system presents users-related search options to the user
10. The user specifies search criteria by configuring search options
11. The user requests for a search based on the specified criteria
12. The system searches for matching user accounts in the database
13. The system responds with a set of users that satisfy the search criteria (ALT 1)
14. The user browses the user search results (ALT 2)
15. End of use case

(ALT 1)
1.1. The system responds indicating that no results were found that satisfied the search criteria
1.2. End of use case

(ALT 2)
2.1. The user selects to toggle the status of a user from “suspended” to “active” or “active” to “suspended”
2.2. The system tags the new status of the user in the database
2.3. End of use case

### 2.6.1.17. View Pipeline Statistics Scenario

**Use Case Name:** Perform Analysis Tasks

**Actors:** Privileged User

**Brief Description:** This scenario describes how a privileged user browses statistics associated with pipeline executions.

**Pre-Condition:** The privileged user needs to be logged into the system.

**Trigger Condition:** The privileged user clicks the Analyze Pipelines on the privileged user navigation bar.

**Normal Flow of Events:**

16. The system presents various statistics of related to pipeline execution by processing the pipeline execution logs of every user

### 2.6.1.18. View Service Statistics Scenario

**Use Case Name:** Perform Analysis Tasks

**Actors:** Privileged User

**Brief Description:** This scenario describes how a privileged user browses statistics associated with registered services.

**Pre-Condition:** The privileged user needs to be logged into the system.

**Trigger Condition:** The privileged user clicks the Analyze Services on the privileged user navigation bar.

**Normal Flow of Events:**

17. The system presents various statistics of related to registered services by processing the service execution logs
2.6.1.19. View Query Statistics Scenario

<table>
<thead>
<tr>
<th>Use Case Name:</th>
<th>Perform Analysis Tasks</th>
</tr>
</thead>
<tbody>
<tr>
<td>Actors:</td>
<td>Privileged User</td>
</tr>
<tr>
<td>Brief Description:</td>
<td>This scenario describes how a privileged user browses statistics associated with user query submissions.</td>
</tr>
<tr>
<td>Pre-Condition:</td>
<td>The privileged user needs to be logged into the system.</td>
</tr>
<tr>
<td>Trigger Condition:</td>
<td>The privileged user clicks the Analyze Queries on the privileged user navigation bar</td>
</tr>
</tbody>
</table>

Normal Flow of Events:
18. The system presents various statistics related to query submission by processing the query submission logs of every user.

2.7. User Characteristics
The VisKo system will only support two types of users, the Regular User and the Privileged User. The Privileged User is a person that wishes to install VisKo and manage it. Using the system to manage and analyze the use of the system by the regular user. The Regular User is the main user of the system, this user comes to VisKo primarily to generate visualizations of scientific data. A requirement of the system is that the regular user must be registered.

2.8. General Constraints
The constraints for the development of VisKo are as follows:
- The system will be built by May of 2014.
- The design, implementation, and verification will satisfy the requirements as defined within this SRS.

2.9. Assumptions and Dependencies
The following are the assumptions and dependencies for the construction of the VisKo system:
- The client will provide the VisKo API as described in previous interviews and demonstrations.
- The client will provide the VisKo logo to the development teams.
3. Specific Requirements

3.1. External Interface Requirements
This section describes requirements pertaining to the external interface of the system. For these interface requirements, a picture from a prototype that was used to elicit requirements is shown. The general intent of this section is to guide the developer in creating an interface that matches the ones described here.

3.1.1. User Interfaces
This section will display the different views of the VisKo system and what they must look like.

3.1.1.1. Guest Heading
[Req 1] The system shall contain a Guest Heading as displayed below heading every Guest page.

3.1.1.2. Guest Welcome
[Req 2] The system shall contain a Guest Welcome page as displayed below when first entering the system.

3.1.1.3. Register
[Req 3] The system shall contain a Register page as displayed below to support registration to the system.
3.1.1.4. Account Activation

[Req 4] The system shall contain an Account Activation confirmation page as displayed below.

Greetings, your account is now activated!
You can now login and start visualizing your data with VisKo!

3.1.1.5. Password Recovery

[Req 5] The system shall contain a Password Recovery page as displayed below.

Your password will be sent to the email address above if it was registered with this VisKo installation.

3.1.1.6. Regular User Heading

[Req 6] The system shall contain a Regular User Heading as displayed below heading every Regular User page.
3.1.1.7. Regular User Navigation

[Req 7] The system shall contain a Regular User Navigation as displayed below, positioned on the left side of every Regular User page.

![Regular User Navigation Diagram]

[Req 8] The Regular User Navigation shall contain five elements: Home, Search History, Visualize, Manage Services and Configure Account as shown above.

3.1.1.8. Regular User Home

[Req 9] The system shall contain a Regular User Home page as displayed below for users that have registered and successfully logged into the system.
3.1.1.9. Regular User Tasks

The following section describes the UI elements that will appear when the Regular User interacts with the Regular User Navigation.

3.1.1.9.1. Search History

[Req 10] The system shall contain two pages under Search History: the Specify Criteria page and the View Details page.

3.1.1.9.1.1. Specify Criteria

[Req 11] The system shall contain a Specify Criteria page as displayed below.

VisKo
3.1.1.9.1.2. View Details

[Req 12] The system shall contain a View Details page as displayed below.

![View Details Page]

### Visualize

[Req 13] The system shall contain five pages under Visualize: the Choose Query Style page, the Select Pipelines page, the Set Query Parameters page, the Edit Pipeline Parameters page, and the View Result page.

#### 3.1.1.9.2.1. Choose Query Style

[Req 14] The system shall contain a Choose Query Style page as displayed below.
3.1.1.9.2.2. Select Pipelines
[Req 15] The system shall contain a Select Pipelines page as displayed below.

3.1.1.9.2.3. Set Query Parameters
[Req 16] The system shall contain a Set Query Parameters page as displayed below.
3.1.1.9.2.4. Edit Pipeline Parameters

[Req 17] The system shall contain an *Edit Pipeline Parameters* page as displayed below.

**Editing Parameters for Pipeline <<ID>>**

- **Service 1**
  - Parameter 1 = 12
  - Parameter 2 = 12312.23423
  - Parameter 3 = 0.0002

- **Service 2**
  - Parameter 4 = 0

- **Service 3**
  - Parameter 5 = 109.8
  - Parameter 6 = 90.5

[Save]

3.1.1.9.2.5. View Result

[Req 18] The system shall contain a *View Result* page as displayed below.
3.1.1.9.3. Manage Services

[Req 19] The system shall contain seven pages under Manage Services: the Select Operation page, the Edit Mapper page, the Edit Converter page, the Edit Transformer page, the Edit Viewer page, the Edit Viewer Set page, and the Edit Filter page.

3.1.1.9.3.1. Select Operation

[Req 20] The system shall contain a Select Operation page as displayed below.

3.1.1.9.3.2. Edit Mapper
[Req 21] The system shall contain an *Edit Mapper* page as displayed below.

![Edit Mapper](image1)

### 3.1.1.9.3.3. Edit Converter

[Req 22] The system shall contain an *Edit Converter* page as displayed below.

![Edit Converter](image2)

### 3.1.1.9.3.4. Edit Transformer

[Req 23] The system shall contain an *Edit Transformer* page as displayed below.
3.1.1.9.3.5. Edit Viewer

[Req 24] The system shall contain an *Edit Viewer* page as displayed below.

3.1.1.9.3.6. Edit Viewer Set

[Req 25] The system shall contain an *Edit Viewer Set* page as displayed below.
3.1.1.9.3.7. Edit Filter
[Req 26] The system shall contain an Edit Filter page as displayed below.

3.1.1.9.4. Configure Account
[Req 27] The system shall contain one page under Configure Account as shown below.
3.1.1.10. Privileged User Heading

[Req 28] The system shall contain a Privileged User Heading as displayed below heading every Privileged User page.

3.1.1.11. Privileged User Navigation

[Req 29] The system shall contain Privileged User Navigation as displayed below, positioned on the left side of every Privileged User page.
[Req 30] The Privileged User Navigation shall contain seven elements: Search Pipelines, Search Queries, Search Services, Search Users, Analyze Pipelines, Analyze Queries, and Analyze Services as shown above.

3.1.1.12. Privileged User Home

[Req 31] The system shall contain a Privileged User Home page as displayed below for users that have successfully logged into the system.
3.1.1.13. Search Tasks

The following section describes the UI elements that will appear when the Privileged User interacts with the Privileged User Navigation search task section.

3.1.1.13.1. Search Pipelines

[Req 32] The system shall contain two pages under Search Pipelines: the Specify Criteria page, and the View Details page.

3.1.1.13.1.1. Specify Criteria

[Req 33] The system shall contain a Specify Criteria page as displayed below.
3.1.13.1.2 View Details

[Req 34] The system shall contain a View Details page as displayed below.
3.1.13.2. Search Queries

[Req 35] The system shall contain one page under Search Queries page as displayed below.
3.1.1.13.3. Search Services

[Req 36] The system shall contain three pages under Search Services: the Specify Criteria page, the View Details page, and the View Arguments page.

3.1.1.13.3.1. Specify Criteria

[Req 37] The system shall contain a Specify Criteria page as displayed below.

3.1.1.13.3.2. View Details
[Req 38] The system shall contain a View Details page as displayed below.

![View Details Page]

3.1.1.13.4. Search Users

[Req 39] The system shall contain one page under Search Users page as displayed below.

![Search Users Page]

3.1.1.14. Analysis Tasks

The following section describes the UI elements that will appear when the Privileged User interacts with the Privileged User Navigation analysis tasks section.
3.1.1.14.1. Analyze Pipelines
[Req 40] The system shall contain one page under Analyze Pipelines page as displayed below.

3.1.1.14.2. Analyze Queries
[Req 41] The system shall contain one page under Analyze Queries page as displayed below.
3.1.1.14.3. Analyze Services

[Req 42] The system shall contain one page under Analyze Services page as displayed below.
3.1.2. Hardware Interfaces

There are no hardware interface requirements at this time.

3.1.3. Software Interfaces

This section describes requirements related to interfacing with the VisKo API.

[Req 43] The system shall interface with the VisKo API as described under documentation at https://github.com/nicholasdelrio/visko.

3.1.3.1. User Interface Population

For the following requirements, every method required to populate a user interface component i.e., drop down lists, is contained in the VisKo API Java class “edu.utep.trustlab.visko.sparql.ViskoTripleStore”.

[Req 44] The listing of Visualization Abstractions presented in pages: Choose Query Style and Edit Mapper shall be populated by the result returned by the method “getVisualizationAbstractions(“).

[Req 45] The listing of Viewer Sets presented in pages: Set Query Parameters and Edit Viewer shall be populated by the result returned by the method “getViewerSets(“).

[Req 46] The listing of Data Formats presented in pages: Set Query Parameters, Edit Viewer, Edit Mapper, Edit Filter, Edit Transformer, and Edit Converter shall be populated by the result returned by the method “getInputFormats(“).

[Req 47] The listing of Data Types presented in pages: Set Query Parameters, Edit Viewer, Edit Mapper, Edit Filter, Edit Transformer, and Edit Converter shall be populated by the result returned by the method “getInputDataTypes(“).
[Req 48] The listing of Toolkits presented in pages: Edit Viewer Set, Edit Mapper, Edit Filter, Edit Transformer, and Edit Converter shall be populated by the result returned by the method “getToolkits().”

### 3.1.3.2. Query Processing
For the following requirements, the VisKo API package containing the set of classes that manage query processing is: “edu.utep.trustlab.visko.planning”.

[Req 49] The Choose Query Style shall rely on the functions provided by “Query” to verify the correctness of input query syntax

[Req 50] The Select Pipelines page shall rely on the functions provided by “QueryEngine” to derive a set of executable pipelines from a query.

### 3.1.3.3. Pipeline Processing
For the following requirements, the VisKo API package containing the set of classes that manage pipeline execution is: “edu.utep.trustlab.visko.execution.”

[Req 51] The Select Pipeline Page shall rely on the functions provided by “edu.utep.trustlab.visko.execution.”PipelineExecutorJob” to prepare a pipeline for execution.

[Req 52] The Select Pipeline Page shall rely on the functions provided by “edu.utep.trustlab.visko.execution.”PipelineExecutor” to execute a pipeline

[Req 53] The Select Pipelines page shall rely on the functions provided by “edu.utep.trustlab.visko.execution.”PipelineExecutorJobStatus” to populate the pipeline execution status.

[Req 54] The Search History Details, Edit Pipeline Parameters, and Search Pipeline Details pages shall rely on the functions provided by “edu.utep.trustlab.visko.planning.pipelines.Pipeline.getParameterBindings()” to populate the parameter bindings listing.

### 3.1.3.4. Registration
This section describes requirements for assembling a “visko-module” which is structured Java code that VisKo can execute to install a service.

[Req 55] The Edit Viewer Set Page shall bundle the registration information in the form of a visko-module and forward the resulting module to the VisKo module installer script.

[Req 56] The Edit Viewer Page shall bundle the registration information in the form of a visko-module and forward the resulting module to the VisKo module installer script.

[Req 57] The Edit Mapper Page shall bundle the registration information in the form of a visko-module and forward the resulting module to the VisKo module installer script.

[Req 58] The Edit Transformer Page shall bundle the registration information in the form of a visko-module and forward the resulting module to the VisKo module installer script.

[Req 59] The Edit Filter Page shall bundle the registration information in the form of a visko-module and forward the resulting module to the VisKo module installer script.

[Req 60] The Edit Converter Page shall bundle the registration information in the form of a visko-module and forward the resulting module to the VisKo module installer script.

### 3.1.4. Communications Interfaces
There are no communications interfaces requirements at this time.

### 3.2. Behavioral Requirements
This section describes the behavioral requirements of VisKo.
3.2.1. Same Class of User
This section contains requirements related to the same class of users.

3.2.1.1. Regular User
[Req 61] The system shall require the Regular User to register and be activated before granting access to the system.

3.2.1.2. Privileged User
[Req 62] There shall only be one Privileged User per installation of the system.
[Req 63] The Privileged User will activate Regular User accounts.

3.2.2. Related Real-world Objects
This section describes requirements related to the Real world Objects of the VisKo system.

[Req 64] A Viewer Set shall be a named set of Viewers associated with a Toolkit
[Req 65] A Toolkit shall be the name of an external software application that supports a set of Viewers, Mappers, Transformers, Filters, and Converters.
[Req 66] A Format shall be the name of a file-encoding scheme.
[Req 67] A Type shall be the name of a data type that is encoded in some specific Format.
[Req 68] A Viewer shall be the description of a piece of external software that can render a dataset of a given Format and Type.
[Req 69] A Transformer shall be a service that transforms an input dataset to a dataset with a different Type and Format.
[Req 70] A Mapper shall be a type of Transformer that generates a Visualization Abstraction.
[Req 71] A Filter shall generate a dataset with elements removed from the input.
[Req 72] A Converter shall change the Format of an input dataset.
[Req 73] The system shall adhere to the following rules for designating a service as a Transformer, Filter, and Mapper:

<table>
<thead>
<tr>
<th>Operator Type</th>
<th>Type Condition</th>
<th>Format Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>is Filter</td>
<td>if input.type == output.type</td>
<td>and if input.format == output.format</td>
</tr>
<tr>
<td>is Converter</td>
<td>if input.type == output.type</td>
<td>and if input.format != output.format</td>
</tr>
<tr>
<td>is Transformer</td>
<td>if input.type != output.type</td>
<td>inconsequential</td>
</tr>
</tbody>
</table>

[Req 74] A pipeline shall be an ordered sequence of services.
[Req 75] A service shall be a WSDL service.
[Req 76] A parameter shall control the behavior of a service.

3.2.3. Stimulus
This section will describe the requirements related to stimulus of the VisKo interface.
3.2.3.1. Guest Heading and Guest Welcome

[Req 77] When the Login button is clicked, the system shall capture what is entered into the Email and Password field and submit for login validation.

[Req 78] When the Register button is clicked, the system shall display the Register page.

[Req 79] If the user cannot be logged into the system, the system shall display a message stating that the user could not be logged in and suggesting to register if they do not have an account.

3.2.3.2. Register

[Req 80] When the Create Account button is clicked, the system shall ensure that all the required fields are entered, if the required fields are not entered a message shall be displayed stating that the required fields are not entered.

[Req 81] When the Create Account button is clicked, the system shall check that the passwords and emails match, if the emails and passwords do not match a message shall be displayed stating that the emails or passwords do not match.

3.2.3.3. Password Recovery

[Req 82] When the Forgot Password link is clicked on the Guest Welcome page, the system shall display the Password Recovery page.

[Req 83] When the Recover button is clicked, the system shall check that the email is associated with an existing account and if not, the system will display a non-existing account message.

[Req 84] When the Recover button is clicked and if an account exists, the system shall email the user the password associated with that account.

3.2.3.4. Regular User Heading

[Req 85] Upon logging into the system, the Welcome <<user@email.com>> shall be replaced with the corresponding user email address.

[Req 86] When the Logout button is clicked, the system shall log the user off from the system and display the Guest Welcome page.

3.2.3.5. Regular User Navigation

[Req 87] When the Home element is clicked, the system shall display the Regular User Home page.

[Req 88] When the Search History element is clicked, the system shall display the Specify Criteria page.

[Req 89] When the Visualize element is clicked, the system shall display the Choose Query Style page.

[Req 90] When the Manage Services element is clicked, the system shall display the Select Operation page.

[Req 91] When the Configure Account element is clicked, the system shall display the Configure Account page.

3.2.3.6. Regular User Tasks

The following section describes the stimulus response to the UI elements for the Regular User Tasks.

3.2.3.6.1. Search History

This section describes the stimulus for Search History tasks.

3.2.3.6.1.1. Specify Criteria

[Req 92] When the Abstraction drop down list is clicked, the system shall display all the visualization abstractions used previously by this user within the system.
[Req 93] When the Input URL drop down list is clicked, the system shall display all the URLs used previously by this user within the system.

[Req 94] When the Viewer Set drop down list is clicked, the system shall display all the Viewers used previously by this user within the system.

[Req 95] When the Source Format drop down list is clicked, the system shall display all the source formats used previously by this user within the system.

[Req 96] When the Source Type drop down list is clicked, the system shall display all the source types used previously by this user within the system.

[Req 97] When the Target Format drop down list is clicked, the system shall display all the target formats used previously by this user within the system.

[Req 98] When the Target Type drop down list is clicked, the system shall display all the target types used previously by this user within the system.

[Req 99] When a date is clicked in the Start Date section of the Date Range Calendar, the system shall highlight that date.

[Req 100] When a date is clicked in the End Date section of the Date Range Calendar, the system shall highlight that date.

[Req 101] When the Search button is clicked, the system shall use the entered drop down information, the date range selected as Search Criteria and display the results of the search in the Results section.

[Req 102] When a visualization result is clicked, the system shall display the View Details page for the clicked result.

3.2.3.6.1.2. View Details

[Req 103] When the View Details page loads, the system shall populate all the fields in the page with the information that was used to create that visualization.

3.2.3.6.2. Visualize

This section describes the stimulus for Visualize tasks.

3.2.3.6.2.1. Choose Query Style

[Req 104] When the Submit button is clicked, the system shall use the entered query in the Option 1 area to generate pipelines which shall be displayed in the Select Pipelines page.

[Req 105] When the Submit button is clicked, if there is no text in the Option 1 area, the system shall display an error message stating that the text area is empty.

[Req 106] When the expand(triangle pointing right)/collapse(triangle pointing down) buttons are clicked in the Option 2 area, the system shall display the associated text areas with the query options.

[Req 107] When the user clicks on the image associated with one of the options, the system shall display the Set Query Parameters page.

3.2.3.6.2.2. Select Pipelines

[Req 108] When the expand(triangle pointing right)/collapse(triangle pointing down) buttons are clicked in the Select Pipeline area, the system shall display the associated text areas with the Pipelines options.

[Req 109] When user clicks on the Edit button associated with a Pipeline, the system shall display the Edit Pipeline Parameters page.

[Req 110] When user clicks on the Run button associated with a Pipeline, the system shall execute that pipeline and append it to be displayed at the bottom of the Pipeline Execution Queue.

[Req 111] When user clicks on the Remove button associated with a Pipeline row in the Pipeline Execution Queue, the system shall remove that pipeline from the Pipeline Execution Queue.
Software Requirements Specification

[Req 112] When a pipeline is executing, the Result being displayed shall be an image signifying that it is currently being processed, and a message shall popup with the text “Executing <<Service Name>>”.

[Req 113] When a pipeline fails its execution, the Result being displayed shall be an image signifying that it failed, and a message shall popup with the text “Error Executing <<Service Name>>”.

[Req 114] When the user clicks on an image Result on the Pipeline Execution Queue area, the system shall display the View Result page.

3.2.3.6.2.3. Set Query Parameters
[Req 115] When the user clicks on the Viewer Set drop down list, the system shall display the list of all supported Viewer Sets.

[Req 116] When the user clicks on the Input Data Format drop down list, the system shall display the list of all supported Input Data Formats.

[Req 117] When the user clicks on the Input Data Type drop down list, the system shall display the list of all supported Input Data Types.

[Req 118] When the user clicks on the Submit button, the system shall capture the contents of the drop down lists and the entered Input Data URL and display the Select Pipelines page.

3.2.3.6.2.4. Edit Pipeline Parameters
[Req 119] When the user clicks on the Save button, the system shall store the entered parameter values and return to the Select Pipelines page.

3.2.3.6.2.5. View Result
[Req 120] When the user clicks on the Save button, the system shall prompt the user for a location to save the generated Visualization in the client local file system.

3.2.3.6.3. Manage Services
This section describes the stimulus for Manage Services tasks.

3.2.3.6.3.1. Select Operation
[Req 121] When the user clicks on the Service Type drop down list, the system shall display the list of all the supported Service Types.

[Req 122] When the user clicks on the Add button, the system shall display the appropriate blank Edit Service (Viewer Set, Viewers, Filter, Transformer, Converter, or Mapper) page based on the value of the drop down list.

[Req 123] When the user clicks on the Viewer Sets drop down list, the system shall display the list of all the Viewer Sets that this user has added.

[Req 124] When the user clicks on the Viewer Sets drop down list, the system shall display the list of all the Viewer Sets that this user has added.

[Req 125] When the user clicks on the Viewers drop down list, the system shall display the list of all the Viewers that this user has added.

[Req 126] When the user clicks on the Filters drop down list, the system shall display the list of all the Filters that this user has added.

[Req 127] When the user clicks on the Transformers drop down list, the system shall display the list of all the Transformers that this user has added.

[Req 128] When the user clicks on the Converters drop down list, the system shall display the list of all the Viewer Sets that this user has added.

[Req 129] When the user clicks on the Mappers drop down list, the system shall display the list of all the Mappers that this user has added.
[Req 130] When the Edit button is clicked, the system shall capture what drop down list value is associated with the radio button that is selected and display the appropriate populated Edit Service (Viewer Set, Viewers, Filter, Transformer, Converter, or Mapper) page based on the value of the drop down list.

[Req 131] When the Delete button is clicked, the system shall capture what drop down list value is associated with the radio button that is selected and delete that service.

3.2.3.6.3.2 Edit Mapper

[Req 132] When the user clicks on the Operation drop down list, the system shall display the list of all the operations that the system supports.

[Req 133] When the user clicks on the Supported Abstraction drop down list, the system shall display the list of all the supported abstractions that the system supports.

[Req 134] When the user clicks on the Input Format drop down list, the system shall display the list of all the input formats that the system supports.

[Req 135] When the user clicks on the Output Format drop down list, the system shall display the list of all the output formats that the system supports.

[Req 136] When the user clicks on the Input Data Type drop down list, the system shall display the list of all the input data types that the system supports.

[Req 137] When the user clicks on the Output Data Type drop down list, the system shall display the list of all the output data types that the system supports.

[Req 138] When the user clicks on the Toolkit drop down list, the system shall display the list of all the toolkits that the system supports.

[Req 139] When the Commit button is clicked, the system shall capture all of the values in each of the drop down lists and the URL entered in the WSDL URL field, and add this new service under this user.

[Req 140] When the Commit button is clicked, if the WSDL URL field is empty the system shall display an error message stating that the WSDL URL field is empty.

[Req 141] When the Commit button is clicked, if the URL entered in the WSDL URL field is malformed, then the system shall display an error message stating that the URL in the WSDL URL field is a malformed WSDL document.

3.2.3.6.3.3 Edit Converter

[Req 142] When the user clicks on the Operation drop down list, the system shall display the list of all the operations that the system supports.

[Req 143] When the user clicks on the Input Format drop down list, the system shall display the list of all the input formats that the system supports.

[Req 144] When the user clicks on the Output Format drop down list, the system shall display the list of all the output formats that the system supports.

[Req 145] When the user clicks on the Toolkit drop down list, the system shall display the list of all the toolkits that the system supports.

[Req 146] When the Commit button is clicked, the system shall capture all of the values in each of the drop down lists and the URL entered in the WSDL URL field, and add this new service under this user.

[Req 147] When the Commit button is clicked, if the WSDL URL field is empty the system shall display an error message stating that the WSDL URL field is empty.

[Req 148] When the Commit button is clicked, if the URL entered in the WSDL URL field is malformed, then the system shall display an error message stating that the URL in the WSDL URL field is a malformed WSDL document.

3.2.3.6.3.4 Edit Transformer
[Req 149] When the user clicks on the Operation drop down list, the system shall display the list of all the operations that the system supports.

[Req 150] When the user clicks on the Input Format drop down list, the system shall display the list of all the input formats that the system supports.

[Req 151] When the user clicks on the Output Format drop down list, the system shall display the list of all the output formats that the system supports.

[Req 152] When the user clicks on the Input Data Type drop down list, the system shall display the list of all the input data types that the system supports.

[Req 153] When the user clicks on the Output Data Type drop down list, the system shall display the list of all the output data types that the system supports.

[Req 154] When the user clicks on the Toolkit drop down list, the system shall display the list of all the toolkits that the system supports.

[Req 155] When the Commit button is clicked, the system shall capture all of the values in each of the drop down lists and the URL entered in the WSDL URL field, and add this new service under this user.

[Req 156] When the Commit button is clicked, if the WSDL URL field is empty the system shall display an error message stating that the WSDL URL field is empty.

[Req 157] When the Commit button is clicked, if the URL entered in the WSDL URL field is malformed, then the system shall display an error message stating that the URL in the WSDL URL field is a malformed WSDL document.

### 3.2.3.6.3.5>Edit Viewer

[Req 158] When the user clicks on the Input Format drop down list, the system shall display the list of all the input formats that the system supports.

[Req 159] When the user clicks on the Input Data Type drop down list, the system shall display the list of all the input data types that the system supports.

[Req 160] When the user clicks on the Part of ViewerSet drop down list, the system shall display the list of all the viewer sets that the system supports.

[Req 161] When the Commit button is clicked, the system shall capture all of the values in each of the drop down lists and the text entered in the Name field, and add this new service under this user.

[Req 162] When the Commit button is clicked, if the Name field is empty the system shall display an error message stating that the Viewer name field is empty.

### 3.2.3.6.3.6>Edit Viewer Set

[Req 163] When the user clicks on the Toolkit drop down list, the system shall display the list of all the toolkits that the system supports.

[Req 164] When the Commit button is clicked, if the Name field is empty the system shall display an error message stating that the Viewer Set name field is empty.

[Req 165] When the Commit button is clicked, if the Brief Description field is empty the system shall display an error message stating that the Brief Description field is empty.

[Req 166] When the Commit button is clicked, if the Documentation URL field is empty the system shall display an error message stating that the Documentation URL field is empty.

[Req 167] When the Commit button is clicked, the system shall capture all of the values in each of the fields and the Toolkit drop down list, and add this new service under this user.

### 3.2.3.6.3.7>Edit Filter

[Req 168] When the user clicks on the Operation drop down list, the system shall display the list of all the operations that the system supports.
[Req 169] When the user clicks on the Input Format drop down list, the system shall display the list of all the input formats that the system supports.

[Req 170] When the user clicks on the Input Data Type drop down list, the system shall display the list of all the input data types that the system supports.

[Req 171] When the user clicks on the Toolkit drop down list, the system shall display the list of all the toolkits that the system supports.

[Req 172] When the Commit button is clicked, the system shall capture all of the values in each of the drop down lists and the URL entered in the WSDL URL field, and add this new service under this user.

[Req 173] When the Commit button is clicked, if the WSDL URL field is empty the system shall display an error message stating that the WSDL URL field is empty.

[Req 174] When the Commit button is clicked, if the URL entered in the WSDL URL field is malformed, then the system shall display an error message stating that the URL in the WSDL URL field is a malformed WSDL document.

3.2.3.6.4. Configure Account
[Req 175] When the Submit Changes button is clicked, if the Current Password is incorrect, then the system shall display an error message stating that the Incorrect Current Password was entered.

[Req 176] When the Submit Changes button is clicked, if the New Password text does not match the New Password Confirmed text, then the system shall display an error message stating that the Passwords do not match.

[Req 177] When the Submit Changes button is clicked, if the New Email Address text does not match the New Email Address Confirmed text, then the system shall display an error message stating that the Emails do not match.

[Req 178] When the Submit Changes button is clicked, if there is text on the Change Password section of the page, then the system shall capture the value of the New Password field and save these changes for the user.

[Req 179] When the Submit Changes button is clicked, if there is text on the Change Email section of the page, then the system shall capture the value of the New Email field and save these changes for the user.

3.2.3.7. Privileged User Heading
[Req 180] Upon logging into the system, the Welcome <<user@email.com>> shall be replaced with the corresponding user email address.

[Req 181] When the Logout button is clicked, the system shall log the user off from the system and display the Guest Welcome page

3.2.3.8. Privileged User Navigation
[Req 182] When the Home element is clicked, the system shall display the Privileged User Home page.

[Req 183] When the Search Pipelines element is clicked, the system shall display the Specify Criteria (for searching pipelines) page.

[Req 184] When the Search Queries element is clicked, the system shall display the Specify Criteria (for searching queries) page.

[Req 185] When the Search Services element is clicked, the system shall display the Specify Criteria (for searching services) page.

[Req 186] When the Search Users element is clicked, the system shall display the Specify Criteria (for searching users) page.

[Req 187] When the Analyze Pipelines element is clicked, the system shall display the Analyze Pipelines page.

[Req 188] When the Analyze Queries element is clicked, the system shall display the Analyze Queries page.
[Req 189] When the Analyze Services element is clicked, the system shall display the *Analyze Services* page.

### 3.2.3.9. Privileged User Home

This section will describe the Search and Analyze tasks stimulus.

### 3.2.3.10. Search Tasks

The following section describes the stimulus response to the search tasks.

#### 3.2.3.10.1. Search Pipelines

The following section describes the stimulus when searching for pipelines.

**3.2.3.10.1.1. Specify Criteria**

- [Req 190] When the Abstraction drop down list is clicked, the system shall display all the visualization abstractions used within the system.
- [Req 191] When the Input URL drop down list is clicked, the system shall display all the URLs used within the system.
- [Req 192] When the Viewer Set drop down list is clicked, the system shall display all the Viewer Sets used within the system.
- [Req 193] When the Source Format drop down list is clicked, the system shall display all the source formats used within the system.
- [Req 194] When the Source Type drop down list is clicked, the system shall display all the source types used within the system.
- [Req 195] When the Target Format drop down list is clicked, the system shall display all the target formats used within the system.
- [Req 196] When the Target Type drop down list is clicked, the system shall display all the target types used within the system.
- [Req 197] When a date is clicked in the Start Date section of the Date Range Calendar, the system shall highlight that date.
- [Req 198] When a date is clicked in the End Date section of the Date Range Calendar, the system shall highlight that date.
- [Req 199] When the Error drop down list is clicked, the system shall display all the possible error types defined within the system.
- [Req 200] When the Search button is clicked, the system shall use the entered drop down information, the date range selected as Search Criteria and display the results of the search in the Results section.
- [Req 201] When the View Details button is clicked within a Results row, the system shall display the *View Details* page populated with the information for the clicked result.

#### 3.2.3.10.2. Search Queries

The following section describes the stimulus when searching for queries.

**3.2.3.10.2.1. Specify Criteria**

- [Req 202] When the Abstraction drop down list is clicked, the system shall display all the visualization abstractions used within the system.
- [Req 203] When the Input URL drop down list is clicked, the system shall display all the URLs used within the system.
- [Req 204] When the Viewer Set drop down list is clicked, the system shall display all the Viewer Sets used within the system.
[Req 205] When the Source Format drop down list is clicked, the system shall display all the source formats used within the system.

[Req 206] When the Source Type drop down list is clicked, the system shall display all the source types used within the system.

[Req 207] When the Target Format drop down list is clicked, the system shall display all the target formats used within the system.

[Req 208] When the Target Type drop down list is clicked, the system shall display all the target types used within the system.

[Req 209] When a date is clicked in the Start Date section of the Date Range Calendar, the system shall highlight that date.

[Req 210] When a date is clicked in the End Date section of the Date Range Calendar, the system shall highlight that date.

[Req 211] When the Error drop down list is clicked, the system shall display all the possible error types defined within the system.

[Req 212] When the Search button is clicked, the system shall use the entered drop down information, the date range selected as Search Criteria and display the results of the search in the Results section and the query in the Query Details section.

[Req 213] When the View Details button is clicked within a Results row, the system shall display the View Details page populated with the information for the clicked result.

3.2.3.10.3. Search Services
The following section describes the stimulus when searching for services.

3.2.3.10.3.1. Specify Criteria
[Req 214] When the Service Type drop down list is clicked, the system shall display all the service types used within the system.

[Req 215] When the Abstraction drop down list is clicked, the system shall display all the visualization abstractions used within the system.

[Req 216] When the Source Format drop down list is clicked, the system shall display all the source formats used within the system.

[Req 217] When the Source Type drop down list is clicked, the system shall display all the source types used within the system.

[Req 218] When the Target Format drop down list is clicked, the system shall display all the target formats used within the system.

[Req 219] When the Target Type drop down list is clicked, the system shall display all the target types used within the system.

[Req 220] When a date is clicked in the Start Date section of the Date Range Calendar, the system shall highlight that date.

[Req 221] When a date is clicked in the End Date section of the Date Range Calendar, the system shall highlight that date.

[Req 222] When the Error drop down list is clicked, the system shall display all the possible error types defined within the system.

[Req 223] When the Status drop down list is clicked, the system shall display all the possible status types defined within the system.

[Req 224] When the Search button is clicked, the system shall use the entered drop down information, the date range selected as Search Criteria and display the results of the search in the Results section.
[Req 225] When the View Details button is clicked within a Results row, the system shall display the View Details page populated with the information for the clicked result.

[Req 226] When the Toggle button is clicked within a Results row, the system shall set the status of this service as Active or Suspended, depending on its current status (Active toggles to Suspended, Suspended toggles to Active).

[Req 227] When the Delete button is clicked within a Results row, the system shall delete the associated service.

3.2.3.10.3.2.View Details

[Req 228] When the View Arguments button is clicked, the system shall display the View Details page populated with the associated service execution information.

3.2.3.10.4.Search Users

The following section describes the stimulus when searching for users.

3.2.3.10.4.1.Specify Criteria

[Req 229] When the Account Status drop down list is clicked, the system shall display all the possible account status types defined within the system.

[Req 230] When the Affiliation drop down list is clicked, the system shall display all the possible affiliations defined within the system.

[Req 231] When the Search button is clicked, the system shall use the entered drop down information, the date range selected as Search Criteria and display the results of the search in the Results section.

[Req 232] When the Toggle button is clicked within a Results row, the system shall set the status of this user account as Active or Suspended, depending on its current status (Active toggles to Suspended, Suspended toggles to Active).

3.2.3.11.Analysis Tasks

The following section describes the stimulus response to the analysis tasks.

3.2.3.11.1.Analyze Pipelines

[Req 233] Upon displaying the Analyze Pipelines page, the system shall populate statistics for the usage of Pipelines by replacing each of the <<placeholders>> with the actual up to date calculated values.

[Req 234] Upon displaying the Analyze Pipelines page, the system shall display a bar graph containing the up to date frequency of pipeline errors.

[Req 235] Upon displaying the Analyze Pipelines page, the system shall populate and display the Most Popular Pipeline and the services that compose it with the associated parameter values.

3.2.3.11.2.Analyze Queries

[Req 236] Upon displaying the Analyze Queries page, the system shall populate statistics for the usage of Queries by replacing each of the <<placeholders>> with the actual up to date calculated values.

[Req 237] Upon displaying the Analyze Queries page, the system shall display a bar graph containing the up to date frequency of query errors.

[Req 238] Upon displaying the Analyze Queries page, the system shall populate and display the Most Popular Query.

3.2.3.11.3.Analyze Services

[Req 239] Upon displaying the Analyze Services page, the system shall populate statistics for the usage of Services by replacing each of the <<placeholders>> with the actual up to date calculated values.
3.2.4. Related Features
There are no related features requirements at this time.

3.2.5. Functional
This section describes the functional requirements of the system.

3.2.5.1. User Status
[Req 242] The system shall log a regular user’s account status as either one of the following:
- Active: the regular user can access all the regular users of the system.
- Suspended: the regular users is unable to access the functions of the system.

3.2.5.2. Service Status
[Req 243] The system shall log a registered service’s availability status as either one of the following:
- Active: the service is available for execution in a pipeline.
- Suspended: the service will cause a pipeline to be filtered from the set of pipeline results presented to the user in the Select Pipelines Page.

3.2.5.3. Query Construction
[Req 244] The Set Query Parameters page shall construct together a visualization query in terms of the selected components and structure according to the query grammar below:

<table>
<thead>
<tr>
<th>QUERY</th>
<th>(PREFIX) SOURCE VISUAL DATA {PARAM}</th>
</tr>
</thead>
<tbody>
<tr>
<td>PREFIX</td>
<td>“PREFIX” {alphanumeric-character} namespace</td>
</tr>
<tr>
<td>SOURCE</td>
<td>“VISUALIZE” url</td>
</tr>
<tr>
<td>VISUAL</td>
<td>“AS” (visualization-abstraction</td>
</tr>
<tr>
<td>DATA</td>
<td>“WHERE TYPE =” type “AND FORMAT =” format</td>
</tr>
<tr>
<td>PARAM</td>
<td>“AND” param “=” value</td>
</tr>
</tbody>
</table>

3.2.5.4. Errors
[Req 245] The system shall define three types of query errors, as follows:
- Syntax Error
- No Pipeline Results Error
- Malformed URI Error

[Req 246] The system shall define three types of pipeline execution errors, as follows:
- Service Execution Error
- Service Timeout Error
- Inaccessible Input Data URL Error

[Req 247] The system shall define three types of service execution errors, as follows:
- Service Execution Error
• Service Timeout Error
• Inaccessible Input Data URL Error

[Req 248] The system shall log regular users’ submission of syntactically incorrect queries.
[Req 249] The system shall log regular users’ submission of queries that yield an empty pipeline set.
[Req 250] The system shall log regular users’ submission of queries containing malformed URIs.
[Req 251] The system shall log regular users’ pipelines that fail execution due to a service execution error.
[Req 252] The system shall log regular users’ pipelines that fail execution due to a service timeout error.
[Req 253] The system shall log regular users’ pipelines that fail execution due to inaccessible input data URL.
[Req 254] The system shall log the association between a pipeline that failed execution and the pipeline error that caused the failure.
[Req 255] The system shall log regular users’ registered services that yield execution errors.
[Req 256] The system shall log regular users’ registered services that yield time out errors.
[Req 257] The system shall log the association between a specific service that failed execution and the service error that caused the failure.
[Req 258] The system shall populate the error drop down lists in Search Queries Criteria with the set of query error types.
[Req 259] The system shall populate the error drop down lists in Search Pipelines Criteria and Search Services Criteria with the set of pipeline execution error types.

3.2.5.5. Pipeline Statistics

[Req 260] The system shall calculate the total number of pipelines executed based on users’ pipeline execution histories.
[Req 261] The system shall identify the most generated visualization abstraction from pipelines based on users’ pipeline execution histories.
[Req 262] The system shall identify the most generated output format from pipelines based on users’ pipeline execution histories.
[Req 263] The system shall identify the most ingested input format from pipelines based on users’ pipeline execution histories.
[Req 264] The system shall identify the most ingested input type from pipelines based on users’ pipeline execution histories.
[Req 265] The system shall identify the most generated output type from pipelines based on users’ pipeline execution histories.
[Req 266] The system shall calculate the total number of service execution errors based on users’ pipeline execution error histories.
[Req 267] The system shall calculate the total number of service input data errors based on users’ pipeline execution error histories.
[Req 268] The system shall calculate the total number of service time outs based on users’ pipeline execution error histories.

3.2.5.6. Service Statistics

[Req 269] The system shall count the total number of services registered based on users’ service registration histories.
[Req 270] The system shall identify the most common visualization abstraction supported based on users’ service registration histories.

[Req 271] The system shall identify the most common input format based on users’ service registration histories.

[Req 272] The system shall identify the most common input data type based on users’ service registration histories.

[Req 273] The system shall identify the most common input data type based on users’ service registration histories.

[Req 274] The system shall calculate the total number of service execution errors based on users’ pipeline execution error histories.

[Req 275] The system shall calculate the total number of service input data errors based on users’ pipeline execution error histories.

[Req 276] The system shall calculate the total number of service time outs based on users’ pipeline execution error histories.

[Req 277] The system shall calculate the total number of pipeline input data errors based on users’ pipeline execution error histories.

[Req 278] The system shall calculate the total number of pipeline time outs based on users’ pipeline execution error histories.

3.2.5.7. Query Statistics

[Req 279] The system shall identify the most common visualization abstraction specified based on users’ query submission histories.

[Req 280] The system shall identify the most common Viewer Set specified based on users’ query submission histories.

[Req 281] The system shall identify the most common input format specified based on users’ query submission histories.

[Req 282] The system shall identify the most common input data type specified based on users’ query submission histories.

[Req 283] The system shall calculate the total number of query syntax errors on users’ query execution error histories.

[Req 284] The system shall calculate the total number of zero pipeline results errors based on users’ query execution error histories.

[Req 285] The system shall calculate the total number of malformed URI errors based on users’ query execution error histories.

3.3. Non-behavioral Requirements

This section describes non-behavioral requirements that the VisKo system must implement.

3.3.1. Performance Requirements

There are no performance requirements at this time.

3.3.2. Qualitative Requirements

This section describes the qualitative requirements of the system.

3.3.2.1. Availability

This section describes requirements related to availability.
[Req 286] The system shall be available to users as long as VisKo’s web server is running and a connection to it can be established from the user.

3.3.2.2. Security
This section describes the security requirements of the system.

[Req 287] The system shall encrypt the registered user’s information using an MD5 algorithm.
[Req 288] The system shall use the email address of the user as a password recovery mechanism.

3.3.2.3. Maintainability
This section describes the maintainability requirements of the system.

[Req 289] The system shall be compatible with all browsers that support HTML 5 and JavaScript 1.8.
[Req 290] The system shall generate error logs for unexpected errors during normal operations and installation of the system.

3.3.2.4. Portability
This section describes the portability requirements of the system.

[Req 291] The system shall be viewable through mobile web browsers.

3.3.3. Design and Implementation Constraints
[Req 292] The system shall adhere to the documentation style used in the VisKo API.

3.4. Other Requirements
This section contains additional requirements.

3.4.1. Operations
There are no operations requirements at this time.

3.4.2. Site Adaptation
This section describes the requirements related to the site adaptation of the system.

[Req 293] The installation and setup of the system for the Privileged User shall be fully documented.
4. Appendix Diagrams

Teams should refer to the diagrams that they developed in CS 4310.