Improving Visualization Knowledge (VisKo)

Candidate Classes and Responsibilities

Version <1.3>

2/28/14

Document Control

Approval

The Guidance Team and the customer shall approve this document.

Document Change Control

|  |  |
| --- | --- |
| Initial Release: | February 20, 2014 |
| Current Release: | V 1.2 |
| Indicator of Last Page in Document: |  |
| Date of Last Review: | February 25, 2014 |
| Date of Next Review: | February 29, 2014 |
| Target Date for Next Update: | February 29, 2014 |

Distribution List

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

Guidance Team Members:

Dr. YoonsikCheon

Bhanukiran Gurijala

Customer: Dr. Nicholas Del Rio

Software Team Members:

Maria Cortes – V&V

Manuel Dosal – Systems Designer

Mario Flores – Lead Programmer 1

Rebekah Gruver –Programmer 2

Hector Quintana – Systems Architect

Marcela Vazques – Systems Analyst

Change Summary

The following table details changes made between versions of this document

|  |  |  |  |
| --- | --- | --- | --- |
| Version | Date | Modifier | Description |
| 1.0 | 2/20/14 | Maria Cortes, Manuel Dosal, Marcela Vazquez | Added candidate classes, scope of product and section 1.3, definitions, acronyms and abbreviations. |
| 1.1 | 2/21/2014 | Hector Quintana, Mario Flores, Rebekah Gruver | Added candidate responsibilities, section 3.2 |
| 1.2 | 2/23/2014 | Leaf Development | Compiled all sections of the document. |
| 1.3 | 2/25/2014 | Leaf Development | Revise the entire document, and added missing information. |

Table of Contents

Document Control ii

Approval ii

Document Change Control ii

Distribution List ii

Change Summary ii

1. Introduction 1

1.1. Purpose and Intended Audience 1

1.2. Scope of product 1

1.3. Definitions, acronyms and abbreviations 1

1.3.1. Definitions 1

1.3.2. Acronyms 2

1.3.3. Abbreviations 2

1.4. Overview 2

1.5. References 3

2. Process Description 4

2.1. Candidate Classes Process 4

2.2. Candidate Responsibilities Process 4

2.3. CRC Cards Process 5

2.4. Techniques, Heuristics, Assumptions and Constraints 5

2.4.1. Techniques 5

2.4.2. Heuristics 6

2.4.3. Assumptions and Constraints 7

3. Candidate Classes and Responsibilities 8

3.1. Candidate Classes 8

3.2. Candidate Responsibilities 18

3.3. CRC Cards 37

# Introduction

## Purpose and Intended Audience

The purpose of the Candidate Classes and Responsibilities (CRC) document is to use an object-oriented technique to organize the management of the system. The organization is formatted to find the different features of the system (candidate classes), along with their associated tasks and knowledge of the classes (candidate responsibilities). The candidate classes and responsibilities will be written on index cards in order to allow easy modification, maintainability, as well as organization.

The intended audience of this document is the guidance team, designers, architect, analyst, programmers and V&V. These cards will keep the team on track, as well as make it easy for them to make modifications to the classes and responsibilities during the development process.

## 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 [3]. 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 [3]. 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 [3].

## Definitions, acronyms and abbreviations

This section is divided into 3 subsections: Definitions, Acronyms, and Abbreviations. Each subsection will help the reader understand definitions, acronyms and abbreviations used within this document.

### Definitions

|  |  |
| --- | --- |
| **Term** | **DEFINITION** |
| Application Programming Interface (API) | 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. |
| Navigation Side Bar | A user interface element within a webpage that contains links to other sections of the website. |
| Pipeline | An ordered sequence of executable web services that generate visualizations. The VisKo API translates user submitted Visualization Queries into sets of Pipelines. |
| Privileged User | 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. |
| Query | 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. |
| Regular User | A user that comes to the system to get desired visualizations based on some given data. |
| Services | A web-based tool kit used to generate visualizations. |
| User | A negotiator, a human agent, who uses or will use the VisKo system. Two different types; privileged user and registered user. |
| Viewer | A is a program that presents visualizations. |
| Viewer Set | A group of viewers. |
| VisKo API | A VisKo application-programming interface version 7.0 that specifies how specific hardware components work with each other. |
| Visualization | 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. |
| Visualization Abstraction | Visualization geometry such as isosurfaces, isolines (i.e., contour lines), points, volumes, polygons, grids, etc. |
| Visualization Knowledge (VisKo) | A system that enables users to generate visualizations, such as graphs or static images, by submitting queries. |
| Visualization Type | A finished visualization of a query. |
| Wrapped Services | Services that can be added to the system |

### Acronyms­­

|  |  |
| --- | --- |
| **TERM** | **DEFINITION** |
| CRC | Class Responsibilities Collaboration |
| GUI | Graphical User Interface |
| ID | Identifier |
| URL | Uniform Resource Locator |
| VisKo | Visualization Knowledge |

### Abbreviations

|  |  |
| --- | --- |
| **TERM** | **DEFINITION** |
| App | Application |
| e.g. | For example |
| etc. | Etcetera |
| i.e. | In other words |

## Overview

The CRC is divided into three main sections: Introduction, Process Description, and Candidate Classes and Responsibilities.

Section 1: the **Introduction**: describes the project and the audience for the document. It includes a brief description of the purpose of the project, references, the scope of the project, and an overview of the CRC document. In addition, a glossary, which details any unique terminology used in this document, is included in this section.

Section 2: the **Process Description:** describes the process used to create the CRC cards, and the process to identify the candidate classes and responsibilities of the VisKo system. It also describes the techniques, heuristics, assumptions and constraints that were made during the process.

Section 3: the **Candidate Classes and Responsibilities:** describes the candidate classes, candidate responsibilities and the CRC cards. Two tables are provided Table 1 contains candidate classes; each belongs to a category (obvious, discarded or unsure). Comments state the reasons of the assigned category. Table 2 contains candidate responsibilities; each belongs to a category (obvious, discarded or unsure). Comments state the reasons of the assigned category. Finally, the section includes the CRC card, which gives each class the responsibilities that best fit.

## References

[1] Cheon, Yoonsik,*Classes, Responsibilities and Collaborations (CRC) Overview.* [Microsoft Power Point] El Paso, TX, 2014.

[2] Guidance Team, “Classes, Responsibilities, and Collaborations Template,” El Paso, TX, 2014.

[3] Guidance Team, “VisKo Software Requirements Specification,” El Paso, TX, 2014

# Process Description

This section describes the process used to create the CRC cards, and process used to identify the candidate classes and responsibilities of the VisKo system. It also describes the techniques, heuristics, assumptions and constraints that were made during the process.

## Candidate Classes Process

This section describes the process Leaf Development used to identify candidate classes.

The process used to identify classes in our system is as follows:

* The team assigned a group in charge of identifying the objects/classes
* The group consisted of three members: Analyst, Designer and V&V
* As a group they divided the software requirements into three equal parts, in order to go through all the requirements: each individual was in charge of revising 97 requirements
* Each member read their assigned requirements, and looked for noun phrases
* Each noun phrase was listed in the section **Noun Phrase** of the Table 1
* Once all nouns were identified, these were separated into obvious candidates, uncertain candidates, and discarded candidates
* After two days of revision, the three members combined all of the noun phrases together
* The noun phrases were arranged in alphabetical order, to order to remove any noun phrase that repeated
* Once all the noun phrases were put in alphabetical order, the three group members refined a list of candidate classes by following a Guideline:
* Model physical objects – e.g., disks, printers, sensors.
* Model conceptual objects – e.g., windows, files, shots, picks.
* Choose one word for one concept – what does it mean within the domain?
* Beware of adjectives – does it really signal a separate class?
* The members justified each candidate class:
* Why it was obvious?
* Why it was unsure?
* Why it was discarded?
* Finally the members named all the obvious classes, and arranged them in alphabetical order

## Candidate Responsibilities Process

This section describes the process Leaf Development used to identify candidate responsibilities.

The process used to identify responsibilities in our system is as follows:

* The team assigned a group in charge of identifying the responsibilities
* The group consisted of three members: Programmer I, Programmer II, and Architect
* As a group they revised the software requirements together, using a single computer to list them all at once.
* Two members kept track of the verbs, while another typed them into a list
* Each verb was listed in the section **Verb Phrase** of the Table 2
* Once all verbs were identified, they were separate into obvious candidates, uncertain candidates, and discarded candidates
* Once all the verb phrases were put together, Programmer I refined a list of responsibilities by the following Guideline:
* Public Services – e.g., knowledge an object maintains and provides.
* Record Services – Responsibilities that fulfill roles within the system.
* Record Knowledge – Maintenance and manipulation of information in the system.
* The members justified each candidate responsibility:
* Why it was obvious?
* Why it was unsure?
* Why it was discarded?
* Finally the members named all the obvious responsibilities, and arrranged them in alphabetical order

## CRC Cards Process

This section describes the process Leaf Development used to create the CRC cards.

The process to create the CRC cards is as follows:

* One team member (V&V) wrote each class onto its own individual card
* All the cards were placed into a Google Document, in order to facilitate the access of all members to the cards
* All the members gave descriptions to each class
* In order to assign responsibilities to the cards, the responsibilities were divided into six equal parts
* Each member had to assign their assigned responsibilities to the most appropriate class, by using the heuristics described in section 2.4.2
* Once all team members had assigned all their responsibilities to the appropriate class, the team checked if all classes were described, and if all had responsibilities
* If a class did not have a responsibility, the team discussed why this happened, and assigned a responsibility that best fit the class
* If there were some responsibilities that did not fit into any of the classes, a new class was created for those responsibilities
* The class was given a name that best described the responsibilities it had
* Again, all the members reviewed the classes that did not have responsibilities, and reviewed the responsibilities to see if some fit these classes. At the same time, the SRS was reviewed, to find possible responsibilities for these classes.
* Finally, all the CRC cards were reviewed and approved by all the group members

## Techniques, Heuristics, Assumptions and Constraints

This section describes the techniques, heuristics, assumptions and constrains that were used/made during the process of identifying classes, responsibilities, and creating the CRC cards.

### Techn­­iques

The techniques used in the processes to identify responsibilities, classes, and create CRC cards are as follow:

* To identify classes the team divided the software requirements in three equal parts, in order to go through all the requirements: each individual was in charge of revising 97 requirements
* Each member read their assigned requirements, and looked for noun phrases that would be a candidate classes
* Each noun phrase was listed in the section **Noun Phrase** of the Table 1
* Once all nouns were identified, these were separated into obvious candidates, uncertain candidates, and discarded candidates
* The noun phrases were arranged in alphabetical order, to remove any noun phrase that repeated
* The team refined a list of candidate classes by following a Guideline:
* Model physical objects – e.g., disks, printers, sensors.
* Model conceptual objects – e.g., windows, files, shots, picks.
* Choose one word for one concept – what does it mean within the domain?
* Be wary of adjectives – does it really signal a separate class
* To identify responsibilities, the team revised the software requirements together, using a single computer to list them all at once.
* Two members kept track of the verbs, while another one typed them in the list
* Each verb was listed in the section **Verb Phrase** of the Table 2
* All verbs were identified, these were separated into obvious candidates, uncertain candidates, and discarded candidates
* All the verb phrases were refined to a list of responsibilities by the following a Guideline:
  + - Public Services – e.g., knowledge an object maintains and provides.
    - Record Services – Responsibilities that fulfill roles within the system.
    - Record Knowledge – Maintenance and manipulation of information in the system
* The team gave a meaningful name to all obvious classes and responsibilities
* The team ordered the candidate classes, and candidate responsibilities alphabetically
* The team justified each candidate responsibility and each candidate class, as to why it was obvious, why it was unsure, and why it was discarded
* The team created a Google document, in order to have easy access to all the CRC cards, and easily modify them
* The responsibilities were divided into six equal parts among the members
* Each member assigned their responsibilities to the most appropriate class
* If responsibilities that did not fit to any of the classes, a new class was created for those responsibilities
* The class was given a name that best described the responsibilities it had
* The CRC cards were reviewed and approved by all the members

### Heuristics

Assigning responsibilities to classes is not an easy process; therefore our team used the following heuristics discussed in class:

*H1: Distribute intelligence*

* Intelligence is what the system knows, actions that it can perform, and impact it has on other systems and users with which it interacts [1].
* In our project the team used this heuristic when deciding to assign responsibilities to classes. The team had to carefully assign limited responsibilities to the classes, this in order to keep the class with low intelligence, and make it depend in other classes. Making them have a more flexible modularity, and more cohesive This is because if a class has to many responsibilities, then it’s very intelligent, and makes modification difficult, and leaves many classes depend highly on intelligent classes. **Example**: the Statistics class had itself too many responsibilities, therefore, some of it’s responsibilities were distributed among CommonStatistics class

*H2: State responsibilities as generally as possible.*

* The concept of finding common responsibilities that can be shared, to make responsibilities reusable.
* In our project the team used this heuristic when naming responsibilities. Some of the responsibilities were named as general as possible, in order to reuse them in other classes. **Example**: the responsibility populateFields() was given a very genera name, instead of naming it populatePipelineFields(), therefore this method is reused in other classes.

*H3: Keep behavior with related information.*

* If information changes, no update messages must be sent between classes [1].
* The class that needs to know that something has changed will know it**.**
* In sum, this makes responsibilities efficient.
* In our project the team used this heuristic when assigning responsibilities to classes. In other words we kept the responsibility with the related data, all data the responsibility uses should be in same class the method uses. **Example**: the responsibility getToolkit () was assigned to the Toolkit class, in order to keep the data and responsibility together.

*H4: Keep information about one thing in one place.*

* The concept of keeping information about one class in one place (i.e. class User, responsibility, createUser)
* Distribution of information often leads to a duplication, and thus inconsistency, it makes responsibilities safe [1].
* In our project the team used this heuristic when assigning responsibilities to classes. When assigning a responsibility the team tried to keep information about a class in one place, in order to avoid inconsistency. **Example**: the responsibilities encryptUserInformationMD5() and setUserStatus() were assigned to the User class in order to keep information about one class in one place.

### Assumptions and Constraints

Some assumptions may appear in any activity where processing information is needed. These are the assumptions our team found while finding classes and responsibilities.

* The team assumed that each page of the GUI needed its own class to make it easier to build.
* The team assumed that no subclasses were going to be used at this time.
* The team assumed that the user’s navigation and header were not the same; therefore, different classes were made for them.
* The team assumed that all nouns needed to be reviewed, and discard the ones that were irrelevant.
* The team assumed anything displayed with HTML wouldn’t be a responsibility.
* The team assumed that if a responsibility did not fit in any class, new classes would be created, and given a name that best described the responsibilities it was given. As a result CommonStatistics Class was made to put around 5 responsibilities that did not fit anywhere.

# Candidate Classes and Responsibilities

This section describes the candidate classes, candidate responsibilities and the CRC cards.

## Candidate Classes

Table 1 shows each candidate class found in the SRS, the classes are listed alphabetically. Each candidate belongs to a one category (obvious, discarded or unsure). Comments state the reasons of the assigned category.

**Table 1** Candidate Classes

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Class** | **Noun Phrase** | **Obvious** | **Unsure** | **Discarded** | **Comments** |
| CommonStatistics | Common operations used, the most common visualization abstraction, etc. | ✓ |  |  | A class containing all information of most common operations used within the system, and most common parameters used |
| Database | Update DB | ✓ |  |  | Handles all the database calls |
| Error | Error | ✓ |  |  | Class that contains all errors in the system. |
| ExecutionQueue | Pipeline Execution Queue | ✓ |  |  | Class that contains all executed pipelines |
| GUIAccountActivationConfirmation | Account activation confirmation page | ✓ |  |  | GUI class for account activation confirmation page |
| GUIAnalizePipelines | Analyze Pipelines element/page | ✓ |  |  | GUI class for analyze pipelines page |
| GUIAnalizeQuieries | Analyze Queries element/page | ✓ |  |  | GUI class for analyze queries page |
| GUIAnalizeServices | Analyze Services element/page | ✓ |  |  | GUI class for analyze services page |
| GUIChooseQueryStyle | Choose Query Style page | ✓ |  |  | GUI class for choose query style page |
| GUIConfigureAccount | Configure Account page | ✓ |  |  | GUI class for configure account page |
| GUIEditConverter | Edit Converter page | ✓ |  |  | GUI class for edit converter page |
| GUIEditFilter | Edit Filter page | ✓ |  |  | GUI class for edit filter page |
| GUIEditMapper | Edit Mapper page | ✓ |  |  | GUI class for edit mapper page |
| GUIEditPipelineParameters | Edit Pipeline Parameters page | ✓ |  |  | GUI class for edit pipeline parameters page |
| GUIEditTransformers | Edit Transformer page | ✓ |  |  | GUI class for edit transformers page |
| GUIEditViewer | Edit Viewer page | ✓ |  |  | GUI class for edit viewer page |
| GUIEditViewerSet | Edit Viewer Set page | ✓ |  |  | GUI class for edit viewer set page |
| GUIGuestWelcome | Guest Welcome page | ✓ |  |  | GUI class for Guest Welcome page |
| GUIPasswordRecovery | Password Recovery page | ✓ |  |  | GUI class for Password Recovery page |
| GUIPrivilegedNavigation | Privileged User Navigation | ✓ |  |  | GUI class that implements the Privileged User navigation |
| GUIPrivilegedUSerHome | Privileged User home page | ✓ |  |  | GUI class for privileged user Home page |
| GUIPrivilegeHeading | Privilege user heading | ✓ |  |  | Class for privilege User Heading |
| GUIRegister | Register page | ✓ |  |  | GUI class for Register page |
| GUIRegularHeading | Regular user heading | ✓ |  |  | Class for Regular User Heading |
| GUIRegularNavigation | Regular User Navigation | ✓ |  |  | GUI class that implements the Regular User Navigation. |
| GUIRegularUserHome | Regular User Home Page | ✓ |  |  | GUI class for Regular User Home page |
| GUISearchQueries | Search Queries element/page | ✓ |  |  | GUI class for Search Queries page |
| GUISearchUSers | Search Users element/page | ✓ |  |  | GUI class for Search Users page |
| GUISelectOperation | Select Operation page | ✓ |  |  | GUI class for Select Operation page |
| GUISelectPipeline | Select Pipelines page | ✓ |  |  | GUI class for Select Pipeline page |
| GUISetQueryParameters | Set Query Parameters page | ✓ |  |  | GUI class for Set Query Parameters |
| GUISpecifyHistory | Specify Criteria page (history) | ✓ |  |  | GUI class for Specify History Criteria page |
| GUISpecifyPipeline | Specify Criteria page (pipelines) | ✓ |  |  | GUI class for Specify Pipeline Criteria page |
| GUISpecifyService | Specify Criteria page (Services) | ✓ |  |  | GUI class for Specify Service Criteria page |
| GUIViewArguments | View Arguments page | ✓ |  |  | GUI class for View Arguments page |
| GUIViewDetailsHistory | View Details page (history) | ✓ |  |  | GUI class for View History Details page |
| GUIViewDetailsPipeline | View Details page (pipeline) | ✓ |  |  | GUI class for View Pipeline Details page |
| GUIViewDetailsService | View Details page (service) | ✓ |  |  | GUI class for View Service Details page |
| GUIViewResult | View Result page | ✓ |  |  | GUI class for View (visualization) Results page |
| Pipeline | Pipeline | ✓ |  |  | Class to contain pipeline information, etc. Defined as sequence of services |
| PrivilegedUser | Privileged User | ✓ |  |  | Subclass of User. Class to contain privileged user information, and special features. |
| Query | Query | ✓ |  |  | Class to contain query information, and special features of query. |
| RegularUser | Regular User | ✓ |  |  | Subclass to contain information for regular user. |
| Service | Service | ✓ |  |  | Class that contains all services information, and special features of service. |
| Rule | Rules | ✓ |  |  | A class containing a set of rules for designating a service as a transformer, filter or mapper. |
| Statistics | Statistics | ✓ |  |  | Class containing system statistics, such as most common queries, pipelines, etc. |
| Toolkit | Toolkit, toolkits |  |  | ✓ | Attribute of Service class |
| User | Users, User, |  |  | ✓ | An abstract class that contains information of users but not used at this time |
| VisKoAPIInterface | VisKo API | ✓ |  |  | Interface class, to interact with VisKo API |
| Visualization | Visualization | ✓ |  |  | A class containing all information of a visualization. |
| Visualization Query | Visualization query | ✓ |  |  | A class containing all information of a visualization query |
|  | Abstraction dropdown list |  |  | ✓ | Attribute of GUISpecifyPipeline class, GUISearchQuery class, and GUISpecifyService class, and GUI SpecifyHistory class |
|  | Account status |  |  | ✓ | Attribute GUISearchUsers class and RegularUser |
|  | Account Status drop down list |  |  | ✓ | Attribute GUISearchUsers class |
|  | Active |  |  | ✓ | Attribute of RegularUser |
|  | Affiliation drop down list |  |  | ✓ | Attribute GUISearchUsers class |
|  | Affiliation/s |  |  | ✓ | Attribute of RegularUser |
|  | Analyze Queries Element |  |  | ✓ | Attribute of GUIPrivilegeNavigation class |
|  | Analyze Pipelines Element |  |  | ✓ | Attribute of GUIPrivilegeNavigation class |
|  | Analyze Users Element |  |  | ✓ | Attribute of GUIPrivilegeNavigation class |
|  | Bar graph |  |  | ✓ | Attribute of Statistics class |
|  | Brief Description field |  |  | ✓ | Attribute of GUIViewerSet class |
|  | Browsers |  |  | ✓ | Requirement about implementation (not relevant to classes) |
|  | Calculated values |  |  | ✓ | Attribute of Pipeline class |
|  | Change Email section |  |  | ✓ | Attribute of GUIConfigureAccount class |
|  | Change Password section |  |  | ✓ | Attribute of GUIConfigureAccount class |
|  | Collapse button |  |  | ✓ | Attribute of GUIChooseQueryStyle |
|  | Commit button |  |  | ✓ | Attribute of GUIEditMapper class |
|  | Converter |  |  | ✓ | Attribute of GUIEditConverter class and Service class |
|  | Configure Account Element |  |  | ✓ | Attribute of GUIRegular Navigation class |
|  | Converters drop down list |  |  | ✓ | Attribute of GUISelectOperation class |
|  | Create Account button |  |  | ✓ | Attribute GUIRegister class |
|  | Current Password |  |  |  | Attribute of User class |
|  | Data Format dropdown list |  |  | ✓ | Attribute of Edit Mapper, Edit Converter, Edit Transformer, and Edit Filter pages |
|  | Data Types, Type, Dataset |  |  | ✓ | Attribute of Edit Mapper, Edit Converter, Edit Transformer, and Edit Filter |
|  | Date frequency |  |  | ✓ | Attribute of Pipeline class |
|  | Date Range Calendar, Date, date range, |  |  | ✓ | Attribute of GUISpecifyPipeline, GUISpecifyServices, GUISpecifyHistory, GUISpecifyQuery classes |
|  | Delete button |  |  | ✓ | Attribute GUISpecifyService class, GUISelectOperation class |
|  | Documentation URL field |  |  | ✓ | Attribute GUIEditMapper class, GUIEditConverter class, GUIEditTranformer class, GUIEditFilter class |
|  | Edit button |  |  | ✓ | Attribute GUISpecifyService class, GUISelectOperation class |
|  | Edit service page |  |  | ✓ | It’s very general and its functions are implemented in other classes. |
|  | Email, email address |  |  | ✓ | Attribute of User class |
|  | Empty pipeline set |  |  | ✓ | Attribute of Error class |
|  | End Date section |  |  | ✓ | Attribute of GUISpecifyPipeline, GUISpecifyServices, GUISpecifyHistory, GUISpecifyQuery classes |
|  | Entered drop down information |  |  | ✓ | Attribute of GUISpecifyPipeline, GUISpecifyServices, GUISpecifyHistory, GUISpecifyQuery classes |
|  | Error drop down list |  |  | ✓ | Attribute of GUISpecifyPipeline, GUISpecifyServices, GUISpecifyHistory, GUISpecifyQuery classes |
|  | Error types |  |  | ✓ | Attribute Error Class |
|  | Error logs |  |  | ✓ | General Error Message used for all errors [incorrect password, other errors, etc] |
|  | Expand button |  |  | ✓ | Attribute of GUIChooseQueryStyle |
|  | Filter |  |  | ✓ | Attribute of Service |
|  | Filters drop down list |  |  | ✓ | Attribute of GUIEditFilter |
|  | Forgot Password link |  |  | ✓ | Attribute of GUIGuestWelcome class, and GUIAccountActivation |
|  | Frequency of service errors |  |  | ✓ | Attribute of Statistics class |
|  | Functions of the system |  |  | ✓ | Very general, not a class. |
|  | Home element |  |  | ✓ | Attribute of GUIPrivilegedNavigation class, and GUIRegularNavigation |
|  | Image |  |  | ✓ | Attribute of ExecutionQueue |
|  | Inaccessible input data URL error |  |  | ✓ | Attribute of Error class |
|  | Incorrect Current Password |  |  | ✓ | Attribute of Error class |
|  | Input data format |  |  | ✓ | Attribute of GUISpecifyPipeline,GUISpecifyServices, GUISpecifyHistory, GUISpecifyQuery classes |
|  | Input data type |  |  | ✓ | Attribute of GUISpecifyPipeline, GUISpecifyServices, GUISpecifyHistory, GUISpecifyQuery classes |
|  | Input data type drop down list |  |  | ✓ | Attribute of GUISpecifyPipeline, GUISpecifyServices, GUISpecifyHistory, GUISpecifyQuery classes |
|  | Input data URL |  |  | ✓ | Attribute of Error class |
|  | Input dataset |  |  | ✓ | Attribute GUIEditMapper class, GUIEditConverter class, GUIEditTranformer class, GUIEditFilter class |
|  | Input format drop down list |  |  | ✓ | Attribute GUIEditMapper class, GUIEditConverter class, GUIEditTranformer class, GUIEditFilter class |
|  | Input URL dropdown list |  |  | ✓ | Attribute of GUISpecifyPipeline, GUISpecifyServices, GUISpecifyHistory, GUISpecifyQuery classes |
|  | Local file system |  |  | ✓ | Attribute of GUIViewResult class |
|  | Location |  |  | ✓ | Attribute of GUIViewResult |
|  | Login button |  |  | ✓ | Attribute of GUIGuestWelcome class |
|  | Logout button |  |  | ✓ | Attribute of GUIRegularHeading, GUIPrivilegeHeading class |
|  | Malformed URI error |  |  | ✓ | Attribute of Error class |
|  | Manage Services element |  |  | ✓ | Attribute of GUIRegularNavigation class |
|  | Mapper |  |  | ✓ | Attribute of Service class |
|  | Mappers drop down list |  |  | ✓ | Attribute of GUIEditMapper class |
|  | MD5 algorithm |  |  | ✓ | Requirement about implementation, not a class. |
|  | Mobile web browsers |  |  | ✓ | Requirement about implementation, not a class. |
|  | Most common input data type |  |  | ✓ | Attribute of Statistics class |
|  | Most common input data type specified |  |  | ✓ | Attribute of Statistics class |
|  | Most common input format |  |  | ✓ | Attribute of Statistics class |
|  | Most common input format specified |  |  | ✓ | Attribute of Statistics class |
|  | Most common viewer set specified |  |  | ✓ | Attribute of Statistics class |
|  | Most common visualization abstraction |  |  | ✓ | Attribute of Statistics class |
|  | Most common visualization abstraction specified |  |  | ✓ | Attribute of Statistics class |
|  | Most generated output format |  |  | ✓ | Attribute of Statistics class |
|  | Most generated output type |  |  | ✓ | Attribute of Statistics class |
|  | Most generated visualization abstraction |  |  | ✓ | Attribute of Statistics class |
|  | Most ingested input format |  |  | ✓ | Attribute of Statistics class |
|  | Most ingested input type |  |  | ✓ | Attribute of Statistics class |
|  | Most popular pipeline |  |  | ✓ | Attribute of Statistics class |
|  | Most popular query |  |  | ✓ | Attribute of Statistics class |
|  | No pipeline results error |  |  | ✓ | Attribute of Error |
|  | Operation drop down list |  |  | ✓ | Attribute GUIEditMapper class, GUIEditConverter class, GUIEditTranformer class, GUIEditFilter class |
|  | Output format drop down list |  |  | ✓ | Attribute of GUIEditTranformer class |
|  | Output Data Type drop down list |  |  | ✓ | Attribute of GUIEditTranformer class |
|  | Parameter values |  |  | ✓ | Too general, parameters are broken down, in different classes. |
|  | Password |  |  | ✓ | Attribute of User class |
|  | Pipeline errors |  |  | ✓ | Attribute of Error class |
|  | Pipeline row |  |  | ✓ | Attribute of GUISelectPipelines |
|  | Pipeline Statistics |  |  | ✓ | Attribute of statistics class |
|  | Pipelines options |  |  | ✓ | Attribute of GUISelectPipelines class |
|  | Privileged User page |  |  | ✓ | Very general, is divided into classes of GUI. |
|  | Query Details section. |  |  | ✓ | Attribute of GUISearchQuery |
|  | Query errors |  |  | ✓ | Attribute of Error class |
|  | Query parameters |  |  | ✓ | Attributes of Query class |
|  | Radio button |  |  | ✓ | Attribute of GUISelectOperation class |
|  | Recover button |  |  | ✓ | Attribute of GUIPasswordRecovery class |
|  | Register button |  |  | ✓ | Attribute of GUIGuestWelcome Page class |
|  | Registration Information |  |  | ✓ | Attribute of GUIRegister class |
|  | Regular User Heading |  |  | ✓ | Attribute of Regular User page |
|  | Regular user’s account status |  |  | ✓ | Attribute of RegularUser class |
|  | Remove button |  |  | ✓ | Attribute of GUISelectPipelines class |
|  | Required field/s |  |  | ✓ | Attribute for different classes, that need required fields (i.e. GUIRegister) |
|  | Result |  |  | ✓ | Attribute of Visualization |
|  | Results row |  |  | ✓ | Attribute of GUISpecifyPipeline, GUISpecifyServices, GUISpecifyHistory, GUISpecifyQuery classes |
|  | Results section |  |  | ✓ | Attribute of GUISpecifyPipeline, GUISpecifyServices, GUISpecifyHistory, GUISpecifyQuery classes |
|  | Run button |  |  | ✓ | Attribute of GUISpecifyPipeline, GUISpecifyServices, GUISpecifyHistory, GUISpecifyQuery classes |
|  | Save button |  |  | ✓ | Attribute of ofGUIEditPipelineParemeters |
|  | Search button |  |  | ✓ | Attribute of GUISpecifyPipeline, GUISpecifyServices, GUISpecifyHistory, GUISpecifyQuery classes |
|  | Search Criteria |  |  | ✓ | Attribute GUISearchUsers |
|  | Search History element |  |  | ✓ | Attribute of GUIRegularNavigation class |
|  | Search Pipelines |  |  | ✓ | Attribute of GUISpecifyPipeline class |
|  | Search pipelines criteria |  |  | ✓ | Attribute of GUISpecifyPipeline class |
|  | Search services criteria |  |  | ✓ | Attribute of Error class |
|  | Search Services element |  |  | ✓ | Attribute GUIPrivilegedNavigation class |
|  | Search Pipelines element |  |  | ✓ | Attribute GUIPrivilegedNavigation class |
|  | Search Queries element |  |  | ✓ | Attribute GUIPrivilegedNavigation class |
|  | Search Users element |  |  | ✓ | Attribute GUIPrivilegedNavigation class |
|  | Service execution error |  |  | ✓ | Attribute of Error class |
|  | Service Statistics |  |  | ✓ | Attribute of Statistics class |
|  | Service timeout error |  |  | ✓ | Attribute of Error class |
|  | Service Type drop down list |  |  | ✓ | Attribute of GUISpecificError class |
|  | Services used |  |  | ✓ | Attribute of Pipeline class |
|  | Source Format dropdown list |  |  | ✓ | Attribute of GUISpecifyPipeline, GUISpecifyServices, GUISpecifyHistory, GUISpecifyQuery classes |
|  | Source Formats |  |  | ✓ | Attribute of Query class |
|  | Source Type drop down list |  |  | ✓ | Attribute of GUISpecifyPipeline, GUISpecifyServices, GUISpecifyHistory, GUISpecifyQuery classes |
|  | Start Date section |  |  | ✓ | Attribute of GUISpecifyPipeline, GUISpecifyServices, GUISpecifyHistory, GUISpecifyQuery classes |
|  | Status, Status types, status of this service |  |  | ✓ | Attribute of RegularService class |
|  | Status drop down list |  |  | ✓ | Attribute of GUISpecifyService class |
|  | Supported abstraction drop down list |  |  | ✓ | Attribute of GUIEditMapper class |
|  | Suspended, status |  |  | ✓ | Attribute of RegularUser class |
|  | Syntax error |  |  | ✓ | Attribute of Error |
|  | Target Format drop down list |  |  | ✓ | Attribute of GUISpecifyPipeline, GUISpecifyServices, GUISpecifyHistory, GUISpecifyQuery classes |
|  | Target Format drop down list, target formats |  |  | ✓ | Attribute of GUISpecifyPipeline, GUISpecifyServices, GUISpecifyHistory, GUISpecifyQuery classes |
|  | Target Type drop down list |  |  | ✓ | Attribute of GUISpecifyPipeline, GUISpecifyServices, GUISpecifyHistory, GUISpecifyQuery classes |
|  | Target Type drop down list |  |  | ✓ | Attribute of GUISpecifyPipeline, GUISpecifyServices, GUISpecifyHistory, GUISpecifyQuery classes |
|  | Target types |  |  | ✓ | Attribute of GUISpecifyPipeline, GUISpecifyServices, GUISpecifyHistory, GUISpecifyQuery classes |
|  | The system |  |  | ✓ | The whole system is not a class by itself. |
|  | Toggle button |  |  | ✓ | Attribute of GUISpecifyService class, and GUISearchUsers |
|  | Toolkit drop down list |  |  |  | Attribute of GUIEditTransformer class |
|  | Total number of malformed URI errors |  |  | ✓ | Attribute of Statistics class |
|  | Total number of pipeline input data errors |  |  | ✓ | Attribute of Statistics class |
|  | Total number of pipeline time outs |  |  | ✓ | Attribute of Statistics class |
|  | Total number of query syntax errors |  |  | ✓ | Attribute of Statistics class |
|  | Total number of service execution errors |  |  | ✓ | Attribute of Statistics class |
|  | Total number of service input data errors |  |  | ✓ | Attribute of Statistics class |
|  | Total number of service time outs |  |  | ✓ | Attribute of Statistics class |
|  | Total number of services registered |  |  | ✓ | Attribute of Statistics class |
|  | Total number of zero pipeline results errors |  |  | ✓ | Attribute of Statistics class |
|  | Transformer |  |  | ✓ | Attribute of Service class |
|  | URIs |  |  | ✓ | Attribute of Service class |
|  | URL |  |  | ✓ | Attribute of Service class |
|  | User account |  |  | ✓ | Attribute of Regular User class |
|  | Users’ pipeline execution histories |  |  | ✓ | Attribute of Statistics class |
|  | Values |  |  | ✓ | Attribute of GUIEditMapper class |
|  | View Arguments button |  |  | ✓ | Attribute of GUIViewServiceDetails class |
|  | View Details button |  |  | ✓ | Attribute of GUISpecifyPipeline, GUISpecifyServices, GUISpecifyHistory, GUISpecifyQuery classes |
|  | Viewer Set |  |  | ✓ | Attribute of Query class |
|  | Viewer set dropdown list |  |  | ✓ | Attribute of GUISpecifyPipeline, GUISpecifyServices, GUISpecifyHistory, GUISpecifyQuery classes |
|  | Viewers |  |  | ✓ | Attribute of Query |
|  | Visko module installer script, module |  | ✓ |  | A possible attribute of Service class, still need to review. |
|  | VisKo’s web server |  |  | ✓ | Not relevant to our system. |
|  | Visualization Abstraction |  |  | ✓ | Attribute of Query class |
|  | Visualize element |  |  | ✓ | Attribute of GUIRegularHeading class |
|  | Error message |  |  | ✓ | General Error Message used for all errors [incorrect password, other errors, etc] |
|  | Regular User Page |  |  | ✓ | Broken down into small GUI classes in the system, therefore is not a class. |
|  | Query Statistics |  |  | ✓ | Attribute of Statistics class. |

## Candidate Responsibilities

Table 2 shows each responsibility found in the SRS, these are listed alphabetically. Each responsibility belongs to a one category (obvious, discarded or unsure). Comments state the reasons of the assigned category.

**Table 2** Candidate Responsibilities

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Verb Phrase** |  | **Obvious** | **Unsure** | **Discarded** | **Comments** |
| Activate | ActivateAccount() | ✓ |  |  | activate user account |
| Add service | addService() | ✓ |  |  | Add a new service under a new user |
| Bundle | bundleRegistrationAsViskoModule() | ✓ |  |  | Bundle the registration information in the form of a visko-module |
| Calculate Pipeline Errors | calculatePipelineInputDataErrors() | ✓ |  |  | Calculates the total number of pipeline input data errors |
| Calculate tall query syntax errors | calculateQuerySyntaxErrors() | ✓ |  |  | Calculate the total number of query syntax errors |
| Calculate Service Execution Errors | calculateServiceExeutionErrors() | ✓ |  |  | Calculate the total number of service execution errors |
| Calculate total number service input error | calculateServiceInputErrors() | ✓ |  |  | Calculate the total number of service input errors |
| Calculate total service time out | calculateServiceTimeOut() | ✓ |  |  | Calculate the total number of service time outs |
| Calculate total Pipelines Executed | calculateTotalPipelinesExecuted() | ✓ |  |  | Calculate the total number of pipelines executed |
| Calculate toal service Input Errors | calculateTotalServiceInputDataErrors() | ✓ |  |  | Calculate the total number of service input data errors |
| Calculate total number of zero pipelines | calculateZeroPipelineResults() | ✓ |  |  | Calculate the total number of zero pipeline results |
| Captures all values in each dropdown list | captureDropdownValueAndTextEntered() | ✓ |  |  | Capture all the values in each of the drop down list and the text entered in the name field |
| Capture | captureValuesAndDropDownURL() | ✓ |  |  | Capture all the values in each of the drop down list and the URL entered |
| Change | changeFormatOfInputDataset() | ✓ |  |  | Change the Format of an input dataset. |
|  | checkEmailAccountAssociation() | ✓ |  |  | Check that the email is associated with an existing account |
| Check | checkPasswordAndEmailMatch() | ✓ |  |  | Check that the passwords and emails match |
| Clicked | Click button |  |  | ✓ | This is a user action, however it could be a method using the onClick() call |
| Construct a visualization query | constructVisualizationQuery() | ✓ |  |  | Construct together a visualization query in term of the components and structure |
|  | Contain a Choose Query Style page |  |  | ✓ | Part of the HTML. |
| Contain | Contain a Guest Heading |  |  | ✓ | Containing something can’t really be seen as a method. It’s part of the HTML. |
| Create | createVisualization() | ✓ |  |  | Create the visualization |
| Define | defineErrorType() | ✓ |  |  | Define three types of query errors |
|  | defineTypesofPipelineErrors() | ✓ |  |  | Define three types of pipeline execution errors |
|  | defineTypesofServicesErrors() | ✓ |  |  | Define three types of service execution errors |
| Delete | deleteService() | ✓ |  |  | Delete the associated service |
| Derive | derivePipelinesFromQuery() | ✓ |  |  | Derive a set of executable pipelines from a query |
| Encrypt | encryptUserInformationMD5() | ✓ |  |  | Encrypt the registered user’s information using MD5 algorithm |
| Encrypt user information | encryptUserInformationMD5() | ✓ |  |  | Encrypt the registered user’s information using MD5 algorithm |
| Execute | executePipeline() | ✓ |  |  | Execute a pipeline |
| Forward | forwardResultModuleToVisKoScript() | ✓ |  |  | Forward the resulting module to the VisKo module installer script |
| Generate | generateDatasetWithElementsRemovedFromInput() | ✓ |  |  | Generate a dataset with elements removed from the input. |
|  | generateErrorLogs() | ✓ |  |  | Generate error logs |
| Highlight | highlightDate() | ✓ |  |  | Highlight the date |
|  | identifMostCommonInputDataType() | ✓ |  |  | Identify the most common input data type |
|  | identifyMostCommonInputFormat() | ✓ |  |  | Identify the most common input format |
|  | identifyMostCommonViewerSet() | ✓ |  |  | Identify the most common viewer set |
| Interface | Interface with the VisKo API |  | ✓ |  | Not exactly a way to put it in a method, although it does have to interface with the VisKo API. |
|  | logAssociationBetweenPipelineAndError() | ✓ |  |  | Log the association between a pipeline that failed execution in the pipeline error that caused the failure |
|  | logAssociationBetweenSpecificServiceAndError() | ✓ |  |  | Log the association between a specific service that failed execution in the service error that caused the failure |
| Submit | loginValidation() | ✓ |  |  | Submit for login validation |
|  | logRegisteredServiceStatus() | ✓ |  |  | Log a registered service availability status |
|  | logRegularUserPipelineExecutionInaccessibleDataURL() | ✓ |  |  | Log regular users’ pipelines that fail execution due to inaccessible data URL |
|  | logRegularUserPipelinesExecutionServiceError() | ✓ |  |  | Log regular users’ pipelines that fail execution due to a service execution error |
|  | logRegularUserPipelinesServiceTimeOutError() | ✓ |  |  | Log regular users’ pipelines that fail execution due to a service time out error |
|  | logRegularUserRegisteredServicesErrors() | ✓ |  |  | Log regular users’ registered services that yield execution errors |
|  | logRegularUserRegisteredServicesTimeOutErrors() | ✓ |  |  | Log regular users’ registered services that yield time out errors |
|  | logRegularUserStatus() | ✓ |  |  | Log a regular user’s account status |
|  | logRegularUserSubmissionOfIncorrectSyntaxQueries() | ✓ |  |  | Log regular users’ submission of syntactically incorrect queries |
|  | logRegularUserSubmissionOfMalformedURL() | ✓ |  |  | Log regular users’ submission of queries of malformed URLs |
|  | logRegularUserSubmissionOfQueriesWithEmptPipeline() | ✓ |  |  | Log regular users’ submission of queries that yield an empty pipeline set |
| Identify most common visualization abstraction | mostCommomVisualizationAbstraction() | ✓ |  |  | Identify the most common visualization abstraction |
| Identify most common input data type | mostCommonInputDataType() | ✓ |  |  | Identify the most common input data type |
| Identify most common input format | mostCommonInputFormat() | ✓ |  |  | Identify the most common input format |
| Identify most common viewer set | mostCommonViewerSet() | ✓ |  |  | Identify the most common viewer set |
| Identify most common visualization support | mostCommonVisualizationSupport() | ✓ |  |  | Identify the most common visualization abstraction support |
| Identify most common generated output format | mostGeneratedOutputFormat() | ✓ |  |  | Identify the most generated output format |
| Identify most common generated output type from pipelines | mostGeneratedOutputTypePipelines() | ✓ |  |  | Identify the most generated output type from pipelines |
| Identify most common visualization | mostGeneratedVisualzations() | ✓ |  |  | Identify the most generated visualization abstractions |
| Idenfity most common input format | mostIngestedInputFormat() | ✓ |  |  | Identify the most ingested input format |
| idendify most common pipeline | mostIngestedInputTypePipelines() | ✓ |  |  | Identify the most ingested input type from pipelines |
|  | Populated by the result returned by the method “getInputDataTypes()” | ✓ |  |  | Requirement contain method itself. |
|  | Populated by the result returned by the method “getInputFormats()” | ✓ |  |  | Requirement containmethod itself. |
|  | Populated by the result returned by the method “getToolKits()” | ✓ |  |  | Requirement contain method itself. |
|  | Populated by the result returned by the method “getViewerSet()” | ✓ |  |  | Requirement contain method itself. |
| Populated | Populated by the result returned by the method “getVisualizationAbstraction()” | ✓ |  |  | Requirement contain method itself. |
|  | populateErrorDropDownPipelineExecutionTypes() | ✓ |  |  | Populate the error drop down lost in search pipelines criteria and search service criteria with a set of pipeline execution types |
|  | populateErrorDropDownQueryTypes() | ✓ |  |  | Populate the error drop down list in search queries criteria with a set of query error types |
|  | populateFields() | ✓ |  |  | Populate all the fields in the page with the information that was used to create that visualization |
|  | populateParamtersBindingListing() | ✓ |  |  | Populate the parameters binding listing |
|  | populatePipelineExecutionStatus() | ✓ |  |  | Populate the pipeline execution status |
| Popup | popUpExecuting(String serviceName) | ✓ |  |  | Popup with the text “Executing <Service Name>” |
|  | popupExecutingError(String serviceName) | ✓ |  |  | Popup with the text “Executing Error <Service Name>” |
| Prepare | preparePipelineExecution() | ✓ |  |  | Prepare a pipeline for execution |
| Remove | removePipelineFromExecutionQueue() | ✓ |  |  | Remove the pipeline from the pipeline execution queue |
| Replace | replaceCorrespondingEmail() | ✓ |  |  | Replace with the corresponding email address |
|  | replaceEachPlaceHolderWithValues() | ✓ |  |  | Replace each of the <place holders> with the actual up to date calculated values |
|  | saveChanges() | ✓ |  |  | Save the changes for the user |
| Save | saveGeneratedVisualization() | ✓ |  |  | Save the generated visualization in the client local file system |
| Email | sendPasswordToAssociatedAccount() | ✓ |  |  | Email password associated with the account |
| Set service status | setServiceStatus() | ✓ |  |  | Set the status of the service as active or suspended |
| Set user status | setUserStatus() | ✓ |  |  | Set the status of this user account as active or suspended |
|  | stateEmailsDoNotMatch() | ✓ |  |  | State that emails do not match |
| Stating | stateTextAreaIsEmpty() | ✓ |  |  | State that the text area is empty |
| Count | totalServices() | ✓ |  |  | Count the total number of services registered |
| Transform | transformInputDatasetToDifferentTypeFormat() | ✓ |  |  | Transforms an input dataset to a dataset with a different Type and Format |
|  |  |  |  |  |  |
| Verify | verifyQuerySyntax() | ✓ |  |  | Verify the correctness of input query syntax |
| Used | Used previously by the user within the system |  |  | ✓ | User action |
| Adhere to the following rules for designating a service as a Transformer, Filter, and Mapper |  |  | ✓ |  | Unsure to what it means by this, is it a rule the system has to follow? |
| Contain a Guest Welcome |  |  |  | ✓ | Part of the HTML. |
| Contain a Password Recovery |  |  |  | ✓ | Part of the HTML. |
| Contain a Privileged User Heading |  |  |  | ✓ | Part of the HTML. |
| Contain a Privileged User Home page |  |  |  | ✓ | Part of the HTML. |
| Contain a Register Page |  |  |  | ✓ | Part of the HTML. |
| Contain a Regular User Heading |  |  |  | ✓ | Part of the HTML. |
| Contain a Regular User Home |  |  |  | ✓ | Part of the HTML. |
| Contain a Regular User Navigation |  |  |  | ✓ | Part of the HTML. |
| Contain a Select Operation page |  |  |  | ✓ | Part of the HTML. |
| Contain a Select Pipelines page |  |  |  | ✓ | Part of the HTML. |
| Contain a Set Query Parameters page |  |  |  | ✓ | Part of the HTML. |
| Contain a Specify Criteria page |  |  |  | ✓ | Part of the HTML. |
| Contain a Specify Criteria page |  |  |  | ✓ | Part of the HTML. |
| Contain a Specify Criteria page |  |  |  | ✓ | Part of the HTML. |
| Contain a View Details page |  |  |  | ✓ | Part of the HTML. |
| Contain a View Details page |  |  |  | ✓ | Part of the HTML. |
| Contain a view Details page |  |  |  | ✓ | Part of the HTML. |
| Contain a View Result page |  |  |  | ✓ | Part of the HTML. |
| Contain an Account Activation confirmation page |  |  |  | ✓ | Part of the HTML. |
| Contain an Edit Converter page |  |  |  | ✓ | Part of the HTML. |
| Contain an Edit Filter page |  |  |  | ✓ | Part of the HTML. |
| Contain an Edit Mapper page |  |  |  | ✓ | Part of the HTML. |
| Contain an Edit Pipeline Parameters Page |  |  |  | ✓ | Part of the HTML. |
| Contain an Edit Transformer page |  |  |  | ✓ | Part of the HTML. |
| Contain an Edit Viewer page |  |  |  | ✓ | Part of the HTML. |
| Contain an Edit Viewer Set page |  |  |  | ✓ | Part of the HTML. |
| Contain five elements |  |  |  | ✓ | Part of the HTML. |
| Contain five pages under Visualize |  |  |  | ✓ | Part of the HTML. |
| Contain one page under Analyze Pipelines |  |  |  | ✓ | Part of the HTML. |
| Contain one page under Analyze Queries page |  |  |  | ✓ | Part of the HTML. |
| Contain one page under Analyze Services page |  |  |  | ✓ | Part of the HTML. |
| Contain one page under Configure Account |  |  |  | ✓ | Part of the HTML. |
| Contain one page under Search Queries |  |  |  | ✓ | Part of the HTML. |
| Contain one page under Search Users page |  |  |  | ✓ | Part of the HTML. |
| Contain Privileged User Navigation |  |  |  | ✓ | Part of the HTML. |
| Contain seven elements |  |  |  | ✓ | Part of the HTML. |
| Contain seven pages under Manage Services |  |  |  | ✓ | Part of the HTML. |
| Contain three pages under Search Services |  |  |  | ✓ | Part of the HTML. |
| Contain two pages under Search History |  |  |  | ✓ | Part of the HTML. |
| Contain two pages under Search Pipelines |  |  |  | ✓ | Part of the HTML. |
| Described under documentation at <https://github.com/nicholasdelrio/visko> |  |  |  | ✓ | Req. 43 |
| Display a bar graph containing the up to date frequency of pipeline errors |  |  |  | ✓ | Part of HTML |
| Display a bar graph containing the up to date frequency of query errors |  |  |  | ✓ | Part of HTML |
| Display a bar graph containing the up to date frequency of services used |  |  |  | ✓ | Part of HTML |
| Display a bar graph containing the up to date frequency of service errors |  |  |  | ✓ | Part of HTML |
| Display a list of all the input data types |  |  |  | ✓ | Part of HTML |
| Display a list of all the input formats |  |  |  | ✓ | Part of HTML |
| Display a list of all the output data types |  |  |  | ✓ | Part of HTML |
| Display a list of all the output formats |  |  |  | ✓ | Part of HTML |
| Display a list of all the toolkits |  |  |  | ✓ | Part of HTML |
| Display a non-existing account message |  |  |  | ✓ | Part of HTML |
| Display al the possible error types defined within the system |  |  |  | ✓ | Part of HTML |
| Display all the possible accounts status types |  |  |  | ✓ | Part of HTML |
| Display all the possible affiliations |  |  |  | ✓ | Part of HTML |
| Display all the possible status types |  |  |  | ✓ | Part of HTML |
| Display all the service types |  |  |  | ✓ | Part of HTML |
| Display all the source formats |  |  |  | ✓ | Part of HTML |
| Display all the source formats used previously by this user within the system |  |  |  | ✓ | Part of HTML |
| Display all the source types |  |  |  | ✓ | Part of HTML |
| Display all the target formats |  |  |  | ✓ | Part of HTML |
| Display all the target formats used previously by this user within the system |  |  |  | ✓ | Part of HTML |
| Display all the target types |  |  |  | ✓ | Part of HTML |
| Display all the target types used previously by this user within the system |  |  |  | ✓ | Part of HTML |
| Display all the urls |  |  |  | ✓ | Part of HTML |
| Display all the URLs used previously by this user within the system |  |  |  | ✓ | Part of HTML |
| Display all the Viewer Sets |  |  |  | ✓ | Part of HTML |
| Display all the Viewers used previously by this user within the system |  |  |  | ✓ | Part of HTML |
| Display all the visualization abstraction |  |  |  | ✓ | Part of HTML |
| Display all the visualization abstractions used previously by this user within the system |  |  |  | ✓ | Part of HTML |
| Display an error message stating that the Brief Description name field is empty |  |  |  | ✓ | Part of HTML |
| Display an error message stating that the documentation URL name field is empty |  |  |  | ✓ | Part of HTML |
| Display an error message stating that the emails do not match |  |  |  | ✓ | Part of HTML |
| Display an error message stating that the Incorrect Current Password was entered |  |  |  | ✓ | Part of HTML |
| Display an error message stating that the Passwords do not match |  |  |  | ✓ | Part of HTML |
| Display an error message stating that the text area is empty |  |  |  | ✓ | Part of HTML |
| Display an error message stating that the Viewer name field is empty |  |  |  | ✓ | Part of HTML |
| Display an error message stating that the Viewer Set name field is empty |  |  |  | ✓ | Part of HTML |
| Display an error message stating that the WSDL URL field is empty |  |  |  | ✓ | Part of HTML |
| Display an error message stating the URL in the WSDL URL field is a malformed WSDL document |  |  |  | ✓ | Part of HTML |
| Display emails or passwords do not match |  |  |  | ✓ | Part of HTML |
| Display the appropriate blank edit Service page |  |  |  | ✓ | Part of HTML |
| Display the associated text areas with the Pipelines options |  |  |  | ✓ | Part of HTML |
| Display the associated text areas with the query options |  |  |  | ✓ | Part of HTML |
| Display the Choose Query Style page |  |  |  | ✓ | Part of HTML |
| Display the Configure Account page |  |  |  | ✓ | Part of HTML |
| Display the Edit Pipeline Parameters page |  |  |  | ✓ | Part of HTML |
| Display the Guest Welcome page |  |  |  | ✓ | Part of HTML |
| Display the list of all supported Input Data Formats |  |  |  | ✓ | Part of HTML |
| Display the list of all supported Input Data Types |  |  |  | ✓ | Part of HTML |
| Display the list of all supported viewer Sets |  |  |  | ✓ | Part of HTML |
| Display the list of all the Filters that the user has added |  |  |  | ✓ | Part of HTML |
| Display the list of all the Mappers the user has added |  |  |  | ✓ | Part of HTML |
| Display the list of all the operations that the system supports |  |  |  | ✓ | Part of HTML |
| Display the list of all the supported abstractions that the system supports |  |  |  | ✓ | Part of HTML |
| Display the list of all the supported Service Types |  |  |  | ✓ | Part of HTML |
| Display the list of all the Transformers that the user had added |  |  |  | ✓ | Part of HTML |
| Display the list of all the Viewer Sets that the user has added |  |  |  | ✓ | Part of HTML |
| Display the list of all the Viewers that this user has added |  |  |  | ✓ | Part of HTML |
| Display the Password Recovery page |  |  |  | ✓ | Part of HTML |
| Display the Register page |  |  |  | ✓ | Part of HTML |
| Display the Regular User Home page |  |  |  | ✓ | Part of HTML |
| Display the required fields are not entered |  |  |  | ✓ | Part of HTML |
| Display the results of the search |  |  |  | ✓ | Part of HTML |
| Display the results of the search in the Results Section |  |  |  | ✓ | Part of HTML |
| Display the Select Operation page |  |  |  | ✓ | Part of HTML |
| Display the Select Pipelines page |  |  |  | ✓ | Part of HTML |
| Display the Set Query Parameters page |  |  |  | ✓ | Part of HTML |
| Display the Specify Criteria page |  |  |  | ✓ | Part of HTML |
| Display the View Details page |  |  |  | ✓ | Part of HTML |
| Display the View Details page |  |  |  | ✓ | Part of HTML |
| Display the View Result page |  |  |  | ✓ | Part of HTML |
| Displayed at the bottom of the Pipeline Execution Queue |  |  |  | ✓ | Part of HTML |
| Displayed in the Select Pipelines page |  |  |  | ✓ | Part of HTML |
| Displayed shall be an image signifying that it is currently being processed |  |  |  | ✓ | Part of HTML |
| Displayed shall be an image signifying that it is failed |  |  |  | ✓ | Part of HTML |
| Enter the parameter values |  |  |  | ✓ | More of a store responsibility rather than enter |
| Granting access to the system |  |  |  | ✓ | This is a privileged user action |
| Log into the system |  |  |  | ✓ | These are for users, unless you meant log as in log data? The only “log into the system” I found in the SRS is actually “Logged” which is meant for users. |
| Log off the system |  |  |  | ✓ | User action |
| Presented in pages: Set Query Parameters and Edit Viewer |  |  |  | ✓ | Presented is similar to display. |
| Register and be activated |  |  |  | ✓ | This is a user action. |
| Rely on the functions provided by “edu.utep.trustlab.visko.execution.” PipelineExecutionJob” |  |  |  | ✓ |  |
| Rely on the functions provided by “edu.utep.trustlab.visko.execution.” PipelineExecution” |  |  |  | ✓ | System relies in functions previously mentioned. |
| Rely on the functions provided by “edu.utep.trustlab.visko.execution.” PipelineExecutionJobStatus” |  |  |  | ✓ | System relies in functions previously mentioned. |
| Rely on the functions provided by “edu.utep.trustlab.visko.planning.pipelines.Pipelines.getParametersBindings()” |  |  |  | ✓ | System relies in functions previously mentioned. |
| Rely on the functions provided by “Query Engine” |  |  |  | ✓ | System relies in functions previously mentioned. |
| Rely on the functions provided by “Query” |  |  |  | ✓ | System relies in functions previously mentioned. |
| Select page |  |  | ✓ |  | User Action |
| Support HTML5 and Javascript 1.8 |  |  |  | ✓ | This is more about the developers following the way the system should be designed. |

## CRC Cards

This section provides all the CRC cards. Each card is labeled with a class name, the class description, and the class responsibilities.

|  |  |
| --- | --- |
| **Class Name:** CommonStatistics |  |
| **Description:** A class containing all information of most common operations used within the system, most common parameters used, and most common outputs (i.e. visualization) |  |
| **Responsibilities:**  mostCommomVisualizationAbstraction()  mostCommonInputDataType()  mostCommonInputFormat()  mostCommonViewerSet()  mostCommonVisualizationSupport()  mostGeneratedOutputFormat()  mostGeneratedOutputTypePipelines()  mostGeneratedVisualzations()  mostIngestedInputFormat()  mostIngestedInputTypePipelines() | **Collaborations:**  << Not used at this time >> |
| **Comments:**  A class containing all information of most common operations used within the system, and most common parameters |  |

|  |  |
| --- | --- |
| **Class Name:** Database |  |
| **Description:** Class that handles database calls |  |
| **Responsibilities:**  saveUserChanges()  saveGeneratedVisualization()  saveUserInformation()  changeFormatOfInputDataset()  encryptUserInformationMD5() | **Collaborations:**  << Not used at this time >> |
| **Comments:**  Handles all the database calls |  |

|  |  |
| --- | --- |
| **Class Name:** Error |  |
| **Description:** Class that manages, and contains all errors within the system. |  |
| **Responsibilities:**  defineErrorType()  defineTypesofPipelineErrors()  defineTypesofServicesErrors()  logRegularUserSubmissionOfIncorrectSyntaxQueries()  logRegularUserSubmissionofQueriesWithEmptyPipeline()  logRegularUserSubmissionOfMalformedURL()  logRegularUserPipelinesExecutionServiceError()  logRegularUserPipelinesServiceTimeOutError()  logRegularUserPipelineExecutionInaccessibleDataURL()  logRegularUserPipelineExecutionInaccessibleDataURL()  logRegularUserRegisteredServicesTimeOutErrors() | **Collaborations:**  << Not used at this time >> |
| **Comments:**  Class that contains and handles all errors in the system. |  |

|  |  |
| --- | --- |
| **Class Name:** ExecutionQueue |  |
| **Description:** Class that manages, and contains all executed pipelines. |  |
| **Responsibilities:**  removePipeline()  popUpExecution(String serviceName)  preparePipelineExecution()  executePipeline()  populatePipelineExecutionStatus() | **Collaborations:**  << Not used at this time >> |
| **Comments:**  Class that contains all executed pipelines |  |

|  |  |
| --- | --- |
| **Class Name:** GUIAccountActivationConfirmation |  |
| **Description:** GUI class for Account Activation Confirmation page. |  |
| **Responsibilities:** activateAccount() | **Collaborations:**  << Not used at this time >> |
| **Comments:**  GUI class for account activation confirmation page |  |

|  |  |
| --- | --- |
| **Class Name:** GUIAnalizePipelines |  |
| **Description:** GUI class for Analyze Pipelines page. |  |
| **Responsibilities:**  checkParameters()  populateGraph() | **Collaborations:**  << Not used at this time >> |
| **Comments:**  GUI class for analyze pipelines page |  |

|  |  |
| --- | --- |
| **Class Name:** GUIAnalizeQuieries |  |
| **Description:** GUI class for Analyze Queries page. |  |
| **Responsibilities:**  checkSyntax()  populateGraph() | **Collaborations:**  << Not used at this time >> |
| **Comments:**  GUI class for analyze queries page |  |

|  |  |
| --- | --- |
| **Class Name:** GUIAnalizeServices |  |
| **Description:** GUI class for Analyze services page. |  |
| **Responsibilities:**  populateGraph() | **Collaborations:**  << Not used at this time >> |
| **Comments:**  GUI class for analyze services page |  |

|  |  |
| --- | --- |
| **Class Name:** GUIChooseQueryStyle |  |
| **Description:** GUI class for Choose Query Style page |  |
| **Responsibilities:**  verifyQuerySyntax()  captureVisualizationAbstraction() | **Collaborations:**  << Not used at this time >> |
| **Comments:**  GUI class for choose query style page |  |

|  |  |
| --- | --- |
| **Class Name:** GUIConfigureAccount |  |
| **Description:** GUI class for Configure Account page |  |
| **Responsibilities:**  checkEmailAccountAssociation()  checkPasswordMatch()  checkEmailMatch()  replaceCorrespondingEmail()  replaceCorrespondingPassword() | **Collaborations:**  << Not used at this time >> |
| **Comments:**  GUI class for configure account page |  |

|  |  |
| --- | --- |
| **Class Name:** GUIEditConverter |  |
| **Description:** GUI class for Edit Converter page |  |
| **Responsibilities:**  checkURLDropDownValues() | **Collaborations:**  << Not used at this time >> |
| **Comments:**  GUI class for edit converter page |  |

|  |  |
| --- | --- |
| **Class Name:** GUIEditFilter |  |
| **Description:** GUI Class for Edit Filter page |  |
| **Responsibilities:**  checkURLDropDownValues() | **Collaborations:**  << Not used at this time >> |
| **Comments:**  GUI class for edit filter page |  |

|  |  |
| --- | --- |
| **Class Name:** GUIEditMapper |  |
| **Description:** GUI class for Edit Mapper page |  |
| **Responsibilities:**  checkURLDropDownsValues() | **Collaborations:**  << Not used at this time >> |
| **Comments:**  GUI class for edit mapper page |  |

|  |  |
| --- | --- |
| **Class Name:** GUIEditPipelineParameters |  |
| **Description:** GUI class for Edit Pipeline Parameters class |  |
| **Responsibilities:**  checkParameters() | **Collaborations:**  << Not used at this time >> |
| **Comments:**  GUI class for edit pipeline parameters page |  |

|  |  |
| --- | --- |
| **Class Name:** GUIEditTransformers |  |
| **Description:** GUI class for Edit Transformers page |  |
| **Responsibilities:**  captureValues()  checkURLDropDown() | **Collaborations:**  << Not used at this time >> |
| **Comments:**  GUI class for edit transformers page |  |

|  |  |
| --- | --- |
| **Class Name:** GUIEditViewer |  |
| **Description:** GUI class for Edit Viewer page |  |
| **Responsibilities:**  checkDropdownValues()  checkTextEntered() | **Collaborations:**  << Not used at this time >> |
| **Comments:**  GUI class for edit viewer page |  |

|  |  |
| --- | --- |
| **Class Name:** GUIEditViewerSet |  |
| **Description:** GUI class for Edit Viewer Set page |  |
| **Responsibilities:**  checkText()  CheckDropdown() | **Collaborations:**  << Not used at this time >> |
| **Comments:**  GUI class for edit viewer set page |  |

|  |  |
| --- | --- |
| **Class Name:** GUIGuestWelcome |  |
| **Description:** GUI class for Guest Welcome page |  |
| **Responsibilities:**  loginValidation() | **Collaborations:**  << Not used at this time >> |
| **Comments:**  GUI class for Guest Welcome page |  |

|  |  |
| --- | --- |
| **Class Name:** GUIPasswordRecovery |  |
| **Description:** GUI class for Password Recovery page |  |
| **Responsibilities:**  checkEmailAccountAssociation()  sendPasswordToAssociatedAccount() | **Collaborations:**  << Not used at this time >> |
| **Comments:**  GUI class for Password Recovery page |  |

|  |  |
| --- | --- |
| **Class Name:** GUIPrivilegedNavigation |  |
| **Description:** GUI class for Privileged User Navigation Sidebar |  |
| **Responsibilities:**  logout() | **Collaborations:**  << Not used at this time >> |
| **Comments:**  GUI class that implements the Privileged User navigation |  |

|  |  |
| --- | --- |
| **Class Name:** GUIPrivilegedUserHome |  |
| **Description:** GUI class for Privileged User Home page |  |
| **Responsibilities:**  logout() | **Collaborations:**  << Not used at this time >> |
| **Comments:**  GUI class for privileged user Home page |  |

|  |  |
| --- | --- |
| **Class Name:** GUIRegister |  |
| **Description:** GUI class for Register page |  |
| **Responsibilities:**  checkPasswordMatch()  heckEmailMatch()  stateTextAreaIsEmpty()  stateEmailsDoNotMatch() | **Collaborations:**  << Not used at this time >> |
| **Comments:**  GUI class for Register page |  |

|  |  |
| --- | --- |
| **Class Name:** GUIRegularNavigation |  |
| **Description:** GUI class for Regular User navigation |  |
| **Responsibilities:**  logout() | **Collaborations:**  << Not used at this time >> |
| **Comments:**  GUI class that implements the Regular User Navigation. |  |

|  |  |
| --- | --- |
| **Class Name:** GUIRegularUserHome |  |
| **Description:** GUI class for Regular User home |  |
| **Responsibilities:** | **Collaborations:**  << Not used at this time >> |
| **Comments:**  GUI class for Regular User Home page |  |

|  |  |
| --- | --- |
| **Class Name:** GUISearchQueries |  |
| **Description:** GUI class for Privileged User search queries option |  |
| **Responsibilities:**  captureDateRange()  checkDropDowns() | **Collaborations:**  << Not used at this time >> |
| **Comments:**  GUI class for Search Queries page |  |

|  |  |
| --- | --- |
| **Class Name:** GUISearchUSers |  |
| **Description:** GUI class for Search Users page |  |
| **Responsibilities:**  checkTextFields()  captureDateRange()  checkDropDowns() | **Collaborations:**  << Not used at this time >> |
| **Comments:**  GUI class for Search Users page |  |

|  |  |
| --- | --- |
| **Class Name:** GUISelectOperation |  |
| **Description:** GUI class for Select Operation page for Regular User |  |
| **Responsibilities:**  checkDropDowns() | **Collaborations:**  << Not used at this time >> |
| **Comments:**  GUI class for Select Operation page |  |

|  |  |
| --- | --- |
| **Class Name:** GUISelectPipeline |  |
| **Description:** GUI class for Select Pipeline page for Visualize Regular User option |  |
| **Responsibilities:**  derivePipelinesFromQuery()  consrtructVisualizationQuery() | **Collaborations:**  << Not used at this time >> |
| **Comments:**  GUI class for Select Pipeline page |  |

|  |  |
| --- | --- |
| **Class Name:** GUISetQueryParameters |  |
| **Description:** GUI class for Set Query Parameters |  |
| **Responsibilities:**  checkURLInput()  getViewerSet()  getInputFormats()  getInputDataTypes() | **Collaborations:**  << Not used at this time >> |
| **Comments:**  GUI class for Set Query Parameters |  |

|  |  |
| --- | --- |
| **Class Name:** GUISpecifyHistory |  |
| **Description:** GUI class for specify history criteria |  |
| **Responsibilities:**  highlightDate()  checkDropdownValues() | **Collaborations:**  << Not used at this time >> |
| **Comments:**  GUI class for Specify History Criteria page |  |

|  |  |
| --- | --- |
| **Class Name:** GUISpecifyPipeline |  |
| **Description:** GUI class for Specify Pipeline Criteria Page |  |
| **Responsibilities:**  highlightDate()  checkDropdownValues() | **Collaborations:**  << Not used at this time >> |
| **Comments:**  GUI class for Specify Pipeline Criteria page |  |

|  |  |
| --- | --- |
| **Class Name:** GUISpecifyService |  |
| **Description:** GUI class for Specify Service Criteria page |  |
| **Responsibilities:**  highlightDate()  checkDropdownValues() | **Collaborations:**  << Not used at this time >> |
| **Comments:**  GUI class for Specify Service Criteria page |  |

|  |  |
| --- | --- |
| **Class Name:** GUIViewDetailsHistory |  |
| **Description:** GUI class for View History Details page |  |
| **Responsibilities:**  populateFields() | **Collaborations:**  << Not used at this time >> |
| **Comments:**  GUI class for View History Details page |  |

|  |  |
| --- | --- |
| **Class Name:** GUIViewDetailsPipeline |  |
| **Description:** GUI class for View Pipeline Details Page |  |
| **Responsibilities:**  populateFields() | **Collaborations:**  << Not used at this time >> |
| **Comments:**  GUI class for View Pipeline Details page |  |

|  |  |
| --- | --- |
| **Class Name:** GUIViewDetailsService |  |
| **Description:** GUI class for View Service Details page |  |
| **Responsibilities:**  populateFields() | **Collaborations:**  << Not used at this time >> |
| **Comments:**  GUI class for View Service Details page |  |

|  |  |
| --- | --- |
| **Class Name:** GUIViewResult |  |
| **Description:** GUI class for View Results page |  |
| **Responsibilities:**  populateFields() | **Collaborations:**  << Not used at this time >> |
| **Comments:**  GUI class for View (visualization) Results page (views some visualization pipeline result) |  |

|  |  |
| --- | --- |
| **Class Name:** Pipeline |  |
| **Description:** Class to contain pipeline information |  |
| **Responsibilities:**  logAssociationBetweenPipelineAndError()  removePipelineExecutionQueue() | **Collaborations:**  << Not used at this time >> |
| **Comments:**  Class to contain pipeline information, etc. Defined as sequence of services |  |

|  |  |
| --- | --- |
| **Class Name:** PrivilegedUser |  |
| **Description:** Subclass to contain information attributes, and special features for privileged user. |  |
| **Responsibilities:**  (leave alone) | **Collaborations:**  << Not used at this time >> |
| **Comments:**  Subclass of User. Class to contain privileged user information, and special features. |  |

|  |  |
| --- | --- |
| **Class Name:** Query |  |
| **Description:** Class containing all information and behavior of a query. |  |
| **Responsibilities:**  transformInputDataTypeToDifferentTypeFormat()  generateDataSetWithElementsRemovedFromInput() | **Collaborations:**  << Not used at this time >> |
| **Comments:**  Class to contain query information, and special features of query. |  |

|  |  |
| --- | --- |
| **Class Name:** RegularUser |  |
| **Description:** Subclass to contain information attributes, and special features for regular user. |  |
| **Responsibilities:**  logRegularUserStatus() | **Collaborations:**  << Not used at this time >> |
| **Comments:**  Subclass to contain information, attributes, and special features for regular user. |  |

|  |  |
| --- | --- |
| **Class Name:** Service |  |
| **Description:** Class containing service knowledge and behavior |  |
| **Responsibilities:**  addService()  deleteService()  setServiceStatus()  logRegisteredServiceStatus()  logAssociationBetweenSpecificServiceAndError()  getToolKit() | **Collaborations:**  << Not used at this time >> |
| **Comments:**  Class that contains all services information, and special features of service. |  |

|  |  |
| --- | --- |
| **Class Name:** Rule |  |
| **Description:** Set of rules to designate a service |  |
| **Responsibilities:**  setRules() | **Collaborations:**  << Not used at this time >> |
| **Comments:**  A class containing a set of rules for designating a service as a transformer, filter or mapper. |  |

|  |  |
| --- | --- |
| **Class Name:** Statistics |  |
| **Description:** Class containing system statistics, such as statistics on errors, total number of users, queries, etc. |  |
| **Responsibilities:**  calculateTotalPipelinesExecuted()  totalServices()  calculatePipelineInputDataErrors()  calculateServiceExeutionErrors()  calculateServiceInputErrors()  calculateZeroPipelineResults()  calculateQuerySyntaxErrors()  calculateServiceTimeOut()  replaceEachPlaceHolderWithValues()  generateErrorLogs()  populateErrorDropDownQueryTypes()  populateErrorDropDownPipelineExcecutionTypes() | **Collaborations:**  << Not used at this time >> |
| **Comments:**  Class containing system statistics, such as statistics on errors, total number of users, queries, etc. |  |

|  |  |
| --- | --- |
| **Class Name:** VisKoAPIInterface |  |
| **Description:** Interface class, to interact with VisKo API |  |
| **Responsibilities:**  bundleRegistrationAsVisKoModule()  forwardResultModuleToVisKoScript() | **Collaborations:**  << Not used at this time >> |
| **Comments:**  Interface class, to interact with VisKo API |  |

|  |  |
| --- | --- |
| **Class Name:** Visualization |  |
| **Description:** A class containing all information of a visualization. |  |
| **Responsibilities:**  createVisualization()  saveGeneratedVisualiztion() | **Collaborations:**  << Not used at this time >> |
| **Comments:**  A class containing all information of a visualization. |  |

|  |  |
| --- | --- |
| **Class Name:** VisualizationQuery |  |
| **Description:** A class containing all information of a visualization query |  |
| **Responsibilities:**  constructVisualizationQuery() | **Collaborations:**  << Not used at this time >> |
| **Comments:**  A class containing all information of a visualization query |  |