SYST699



SPEC Innovations Innoslate™ System Engineering Management Tool Test Evaluation & Analysis

Version 1.0

December 11, 2012

Aron Ceely

Justin Mathews

Kate Stevenson

Bruck Woldie

[Page Left Blank Intentionally]

Executive Summary 7

1 Introduction 10

1.1 Background 10

1.2 Problem Statement 11

1.3 Objectives 12

1.4 Approach 12

1.4.1 Project Requirements 12

1.4.2 Sample Systems Engineering Project 14

1.4.3 Innoslate™ Features 14

1.4.4 Test and Evaluation Plan 15

1.4.5 Analysis 15

2 Test Plan and Evaluation 16

2.1.1 Test Objective 16

2.1.2 Test Scope 16

2.1.3 Test Strategy 17

2.1.4 Test Schedule 22

2.1.5 Test Cases 23

3 Test Case Results 28

3.1 Database Viewer 28

3.1.1 Data Entry 29

3.1.2 File Import/Upload/Export/Deletion 30

3.1.3 Search/Filter/Sort 30

3.1.4 Summary 30

3.2 Requirements Viewer 31

3.2.1 Requirements Creation/Saving/Editing/Deleting 32

3.2.2 Requirements Validation 33

3.2.3 Requirements Reports 33

3.2.4 Summary 34

3.3 Document Analyzer 34

3.3.1 Analyzer 35

3.3.2 Manual Parser 36

3.3.3 Summary 36

3.4 Report Generation 37

3.4.1 Concept of Operations 38

3.4.2 Class Based Report 38

3.4.3 Department of Defense Architecture Framework Report 38

3.4.4 Joint Capabilities Integration Development System 38

3.4.5 Summary 39

3.5 Miscellaneous 39

3.5.1 Automatic Logout 39

3.5.2 Collaboration 40

4 Recommendations 41

4.1 Database View 41

4.1.1 Entity Creation/Saving/Editing/Deletion 41

4.1.2 Importing/Uploading/Exporting/Downloading 41

4.2 Requirements Viewer 42

4.2.1 Requirements Entry/Saving/Editing/Deleting 42

4.2.2 Requirements Validation 42

4.2.3 Requirements Reports 42

4.3 Document Analyzer 43

4.4 Report Generation 43

4.5 Miscellaneous 43

4.5.1 Security 43

4.5.2 Collaboration 44

4.5.3 Other Aspects 44

4.6 Future Recommended Innoslate™ Capabilities 45

4.6.1 Project Management 45

4.6.2 Systems Engineering Lifecycle (SELC) Workflow 46

5 Summary 46

Appendix A: References 49

Appendix B: Requirements Traceability – Functional 50

Appendix C: Requirements Traceability – Non-Functional 55

Appendix D: Test Case Results – Database Viewer 57

Appendix E: Test Case Results – Requirements Viewer 62

Appendix F: Test Case Results – Document Analyzer 65

Appendix G: Test Case Results – Reports Generator 68

Executive Summary

The current market space for suites of software tools supporting the system engineer in developing the artifacts that comprise deliverables for a given system is as diverse as it is confusing. These tools tend to be highly sophisticated products that exist to support large organizations and are tailored to address architecture frameworks and align with capability maturity models. These frameworks and models have steep learning curves, but are generally valued aides in the management of complex, expensive projects. Alternatively, other projects are managed through word documents, spreadsheets, project management scheduling tools, and graphical drawing tools (Microsoft Office Professional Suite). Thus, systems engineers can be frustrated and perplexed when assessing a toolset that best meets their needs while providing utility and efficiency, and enabling productivity in the course of their project lifecycle tasks and activities. The engineer, architect, project manager or user of project management tools desires tools that are lightweight (thin-client), user-friendly, shield the complexity of systems engineering processes and structures, are extensible and scalable, and agile enough to support changes within standard industry practice.

SPEC Innovations is a company that is in the process of developing a toolset based upon the idea of a foundational systems engineering management application that can provide everything that the system engineer might need to develop and maintain work product artifacts in a centralized easily accessible location. Innoslate™ is a cloud-based web application that SPEC Innovations is developing to support this idea of a holistic, comprehensive systems engineering management tool implemented using the software-as-a-service (SaaS) model.

SPEC Innovations was founded as a systems engineering and information technology consulting company and its capabilities as a software product company are relatively immature. This limited experience in developing and testing software products presents an opportunity for our project team to assist them in an evaluation of the Innoslate™ toolset and deliver feedback in the form of tool usage results and recommendations for improvements to existing features/functions and suggestions for new capabilities.

The team decided that the best approach in developing a critical evaluation of Innoslate™ was to utilize a previously developed set of artifacts based on a proposed system that was completed for another class, SYST513 Total Systems Engineering, Re-engineering, and Enterprise Integration. SYST513 is a course offering that is meant to provide foundational learning in the principles of strategic system quality, organizational learning, reengineering approaches, architecture driven system characteristics, and transition management of legacy systems.[[1]](#footnote-2) One aspect of the course involved developing work products and artifacts encompassing the aforementioned principles specific to a fictitious system problem and integrated into a final report. The final report was selected to provide a basis for developing a test strategy and test cases and offer a set of data inputs to conduct a critical analysis of the Innoslate™ tool based on specific system engineering artifacts that would be developed within the Systems Development/Engineering Life Cycle.

Utilizing a sample system enabled the scope of the project to be limited to a subset of artifacts and assets that addressed the concept visioning, requirements and design phases of the system development lifecycle while also ensuring that our team could complete the Innoslate™ system testing and provide detailed feedback and recommendations within the course time constraint.

To aid in the evaluation of Innoslate™ the team adopted an approach that included a test strategy and functional tool recommendations as outputs of the testing execution. The test strategy that was developed included an initial evaluation of the core functional components of Innoslate™ that assisted in the crafting of test evaluation criteria. The evaluation criteria were utilized as the basis for constructing detailed test cases that served as the script for the team while conducting testing of the tool.

As Innoslate™ is a web-based tool operating in Internet “cloud”, the team selected Google Chrome and Mozilla Firefox as the web browsers utilized to conduct testing. The team was split into two teams (each testing with one browser variant) and the test cases were executed by each team to ensure proper coverage across browser types. The test results were captured within a documented test case template as pass or fail with associated comments or issues noted. Each team member documented their respective test results using the noted templates and then one set of results was integrated based on the testing behavior that was most frequently exhibited to eliminate any tool anomaly bias. In addition to findings that were documented within the test cases, the team also documented test findings in a narrative form that provided additional detail regarding the test execution experience. The test results, both in pass/fail and narrative forms, served as the basis for completing the scoring evaluation of the tool.

Lastly a set of recommendations are offered that were derived based on the team’s experience testing the tool, using Firefox and Chrome, and associated working experience developing and delivering system engineering related work products. The report additionally offers in a final summary section our team’s overall impression of the tool, the value it offers to the system engineer, and potential opportunities for SPEC Innovations to refine and advance the tool’s capabilities and value to the system and software engineering marketplace.

# Introduction

The sections that follow will provide introductory detail regarding the background of the team’s project sponsor, the problem definition as stated by the sponsor, the objectives established by the team to address the stated problem, and the approach the team developed to meet the objectives. The project approach includes sub-sections that detail the requirements for the project effort and strategy for successfully completing the project objectives.

## Background

Systems and Proposal Engineering Company (SPEC Innovations) is a company founded in 1993, which offers a diversity of technical, and proposal development consulting services to government and commercial customers. SPEC Innovations’ specific areas of technical expertise include advanced concept technology demonstrations, enterprise architecture consulting, system design and development, and system operations and maintenance support. SPEC Innovations focuses its service offerings particularly in the area of system architectures for programs specific to architecture frameworks (DoDAF), capability maturity models, software development and training. The functional areas for which their customers reside include airborne reconnaissance, counter-proliferation, radar target generators, and ballistic missile defense.

Innoslate™ is the company’s first foray into commercial software product development, and encompasses a cloud-based web application that offers a range of capabilities to manage all aspects of the systems engineering and development lifecycle from concept visioning and requirements derivation to system operational transition.

Innoslate™ was envisioned by SPEC Innovations as the aggregation of several different tools and concepts utilized in the capture and development of system engineering artifacts, requirements, models, and designs over the past three decades. The tool has been designed to provide a simpler and more intuitive mechanism to capture the information detail necessary to manage systems development, operations and support.[[2]](#footnote-3) The application was designed to feel like a drawing tool with underlying semantic meaning to enable the development of an executable model(s) in support of system design. Innoslate™ includes functional capabilities in the areas of requirements management, project collaboration, system design and modeling, artifact reporting, and discrete event simulation. The tool’s fundamental objective is to make the system engineer’s job easier, more productive and efficient, and ultimately increasingly more successful.

## Problem Statement

SPEC Innovations, as has been previously noted, is a mature organization in the areas of technical and proposal services that demonstrate its wealth of expertise and knowledge related to the systems engineering lifecycle and the tools and processes involved with developing system artifacts and deliverables. The company’s competencies in the field of software product development and delivery are much less mature and as such they have determined it would be very beneficial toward the success of Innoslate™ to have a critical review and analysis be conducted to present independent feedback and recommendations that speak to the tools ability to assist in system engineering tasks. The primary motivation for a critical analysis and review of Innoslate™ stems from the need to better understand, from an independent perspective, whether the tool’s usability and functionality address the needs of systems engineers in conducting their job tasks.

## Objectives

The key objectives for this project effort as it relates to the evaluation of Innoslate™ include the following:

* Derive requirements that address the sponsor’s problem specific to reviewing and testing the Innoslate™ tool.
* Document systems test cases and develop a test plan for evaluating Innoslate™.
* Execute the documented test plan in support of a critical evaluation and analysis of the capabilities offered by Innoslate™.
* Compile and detail test results and recommendations as part of a report to the sponsor.
* Deliver the final project report and present project details and findings to the sponsor and George Mason University (GMU) faculty.

## Approach

The SPEC Innovations Team selected a technical approach that allows the team to achieve the objectives for evaluating the Innoslate™ tool. The approach is composed of three parts: (1) identify a sample project to develop systems engineering life cycle stage specific deliverables, (2) scope Innoslate™ capabilities to test and evaluate, (3) describe a test plan that outlines the test’s scope, strategy, evaluation criteria, and test cases. Part of the approach was identifying a sample project to enable the team to use to evaluate the tools capabilities.

### Project Requirements

The functional, non-functional, and recommendations requirements sections that follow will detail the mandated aspects of the team’s project that assisted in developing the approach for meeting the stated project objectives.

#### Functional Requirements

##### **Entity creation and persistence**

The creation, persistence, and lifecycle management of the following entities will be tested:

1. Requirements
2. Assets
3. Artifacts (CONOPS, Requirements Documents, JCIDS reports, DoDAF reports)

##### **Analyzer tool**

The Analyzer tool helps to extract information from Adobe Acrobat and Microsoft Word files into the database. It can be used to extract information into any class of artifacts. Function of this tool will be tested using both MS Word and PDF documents containing artifact definitions for the following classes:

1. Requirements
2. Assets
3. Artifacts

##### **Project Management, Import, and Export Capabilities**

Innoslate™ promotes team work and collaboration by adding a project sharing, import, and export options. Currently Innoslate™ supports importing or exporting of whole projects only – not individual elements. These features will be evaluated problems (if any) will be identified and documented.

#### Non-Functional Requirements

The non-functional aspects of the software will be evaluated during this phase. Mainly the following two aspects of the software will be evaluated.

##### **Usability**

The usability aspects of Innoslate™ will be evaluated, including the following attributes:

1. GUI: Appearance
2. GUI: Organization of Screens and Input Elements
3. Overall Ease of Use

#### Feature Recommendations

Create a wish-list of functionalities that need to be added or recommend modifications to existing functionalities to make it more user-friendly and efficient.

### Sample Systems Engineering Project

The sample project selected to offer data inputs for testing Innoslate™ was derived from a final project report delivered as a requirement for taking Systems (SYST) 513 Total Systems Engineering, Re-engineering, and Enterprise Integration. The sample project entailed the development of a system named Meeting and Event Planning Assistant System (MEEPAS) whose purpose was to design an efficient mechanism for requesting, planning, executing, and monitoring group meetings (both face-to-face and at-a-distance). The MEEPAS Project was decomposed into seven components that spanned all stages of the Systems Engineering Life Cycle (SELC). The team chose to focus on a series of engineering artifacts that a Systems Engineer may be able to develop In Innoslate™ throughout the SELC. These artifacts include:

* Concept of Operations (CONOPS)
* Functional and Non-Functional Requirements Documents
* Joint Capabilities Integration Development System (JCIDS)
* Department of Defense Architecture Framework (DoDAF) reports

### Innoslate™ Features

In addition to selecting a sample project, the team isolated the features within Innoslate™ that could (1) be used to develop the scoped deliverables from the project and (2) be evaluated within the timeframe given for the capstone project.

The core features of Innoslate™ selected by the team for testing include:

* The Document Analyzer is used to import documents into the tool’s Database, allowing both an automatic as well as a manual parsing of the inputted requirements and statements.
* The Database Viewer which allows persisted content entity storage, searching, sorting, and filtering.
* The Collaboration feature allows interactive communication such as project sharing, live-chat, and version control.
* The Requirements Viewer feature of Innoslate™ provides a document-like view of statements and requirements, enabling assignment of quality-scores for each requirement.
* The Reports Generation feature enables users to generate various systems engineering reports based on the inputted objects into the Database, such as class-based and DODAF reports.

### Test and Evaluation Plan

The team developed a test and evaluation plan that describes the testing scope, strategy, evaluation criteria, and test cases. The strategy outlines how the team will execute the test defining evaluation criteria, types of browsers to be used, and how the team will be structured to complete the tests defined in the test cases. The test plan also describes how results will be captured and applied towards the analysis activity of this project.

### Analysis

For this report, the team decided to not use a scoring and weighting evaluation method due to the maturity of the tool. It is felt by the team that at this stage of development and marketing, the team offer SPEC Innovations user experience related results and recommendations that may lead the company to focus on the marketing of Innoslate™’s strengths and continue maturing weak points in the capabilities the team has chosen to test.

As a result, the proposed recommendations will be based on the experiences and the findings the team had as users of the tool. The team will analyze test case results from Mozilla Firefox and Google Chrome users to identify overarching enhancements that can lend a better user experience with the Innoslate™ tool and identify the benefits the tool currently offers to a user in the systems engineering field.

# Test Plan and Evaluation

This section describes the outlined sections of the SPEC Innovations test and evaluation plan, which defines the objectives, scope, strategy and test cases for conducting testing of the Innoslate™ systems engineering management tool.

### Test Objective

As the motivation behind the SPEC Innovations Innoslate™ Project is to provide a critical analysis and recommendations pertaining to the Innoslate™ tool, testing is concerned with both the usability and functionality aspects of the tool. Specifically, the test outcomes are expected to enable the team to determine whether the tool’s usability and functionality address the needs of current and future System Engineers in performing their duties. To that end, each member of the SPEC Innovations project team will be engaged in the testing process including the planning, requirements, test case authoring, test execution and test result compilation.

Testing involves development of test cases, execution of the test cases, as well as evaluation of the test results in order to provide recommendations for improvements. Vehicles of communication with the sponsor involve providing feedback through the Innoslate™ Feature Tracker tool, as well as compiling the detailed test results and recommendations as part of a report to the sponsor. The Innoslate™ tool itself also will be utilized to capture the documented test cases and execution results.

### Test Scope

The scope of the SPEC Innovations Innoslate™ project is limited to a subset of assets and artifacts that address the concepts of visioning, requirements and design, and as such only selected features of Innoslate™ will be tested. These functions will enable the team to better understand whether or not the tool’s usability and functionality address the needs of Systems Engineers in conducting their tasks (Please see Section 1.4.3 for the list of Innoslate™ features that are selected for testing). The decision to focus on a subset of the available Innoslate™ features was made following recommendations from our instructor, and the specific features were selected following discussions within the project team members.

The following features of Innoslate™ to be tested are:

* *Database Viewer*, including storing of Assets, Artifacts, etc.
* *Requirement Viewer*, including capturing of requirements for our SYST 699 project as well as the MEEPAS project, our template project.
* *Document Analyzer*, including automated versus manual parsing of inputted documents.
* *Report Generation,* based on the inputted objects into the Database.
* *Collaboration,* project sharing and user permission evaluation.

### Test Strategy

As it has been mentioned, the overall approach in testing the Innoslate™ tool is establishing a series of functional and usability requirements the team expects the tool to meet, identify and document test cases for each Innoslate™ feature based on how the team expects the feature to perform, and document the results that demonstrate whether the features pass or fail in meeting the team’s documented requirements. Results will then be used to develop recommendations and future features that SPEC Innovations may consider in future versions of Innoslate™.

#### Functional Requirements

For each of the selected Innoslate™ features to test, *functional requirements* have been developed to define what functionalities the team expects each feature to perform. These requirements will be mapped to a test case, where the team will evaluate whether the requirements is met by the tool or not. The team has defined the following requirements below and their test case mappings:

|  |  |  |
| --- | --- | --- |
| **UID** | **Requirements** | **Test Case** |
| **F-1** | **Database Viewer** |  |
| F-1.1 | The Innoslate™ tool shall enable users to manage project data. | DB-001 |
| F-1.1.1 | The Innoslate™ tool shall allow users to create class entities. | DB-001 |
| F-1.1.2 | The Innoslate™ tool shall allow users to store class entities. | DB-002 |
| F-1.1.3 | The Innoslate™ tool shall allow users to edit class entities. | DB-001 |
| F-1.1.4 | The Innoslate™ tool shall allow users to view all class entities in a single location. | DB-003 |
| F-1.1.5 | The Innoslate™ tool shall allow users to filter class entities using a class type or a label. | Db-006 |
| F-1.1.6 | The Innoslate™ tool shall allow users to sort class entities using name, class type, ID, dates. | DB-005 |
| F-1.1.7 | The Innoslate™ tool shall allow users to sort class entities in descending and ascending orders. | DB-005 |
| F-1.1.8 | The Innoslate™ tool shall allow users to search class entities using the entities’ full name and partial name. | DB-004 |
| F-1.1.8.1 | Search results shall be filterable and sortable. | DB-004, DB-005 |
| F-1.1.10 | The Innoslate™ tool shall allow users to delete class entities from the tool. | DB-008 |
| F-1.1.10.1 | The Innoslate™ tool shall prompt users to confirm that their selected class entities are being deleted. | DB-008 |
| F-1.1.10.2 | The Innoslate™ tool shall archive deleted class entities for 30 days to enable users to cross check that deleted class entities were not pertinent to the success of a project. | DB-008 |
| F-1.1.10.3 | For large batch deletions, the Innoslate™ tool shall enable users to delete at least 100 class entities at a time. | DB-008 |
| F-1.1.14 | The Innoslate™ tool shall allow users to import class entities using an XML file. | DB-009 |
| F-1.1.15 | The Innoslate™ tool shall allow users to export class entities into an XML file. | DB-010 |
| **F-2** | **Requirements Viewer** |  |
| F-2.1 | The Innoslate™ tool shall allow users to manage requirements. |  |
| F-2.1.1 | The Innoslate™ tool shall allow users to create requirement entities. | RQ-001 |
| F-2.1.2 | The Innoslate™ tool shall allow users to store requirement entities. | DB-002 |
| F-2.1.3 | The Innoslate™ tool shall allow users to edit requirement entities. | RQ-003 |
| F-2.1.4 | The Innsolate™ tool shall allow users to view all requirements in a view separate from database entities. |  |
| F-2.2 | The Innoslate™ tool shall assess the quality of a requirement using a scale of 100%. | RQ-004, RQ-005 |
| F-2.2.1 | The Innoslate™ tool shall assign a quality value to a requirement. | RQ-004, RQ-005 |
| F-2.2.2 | The Innsolate™ tool shall display the quality score next to the requirement. | RQ-004, RQ-005 |
| F-2.3 | The Innoslate™ tool shall enable users to decompose requirements. | RQ-005 |
| F-2.4 | The Innoslate™ tool shall allow users to display the decomposition of the requirements hierarchy. | RQ-005 |
| F-2.5 | The Innsolate tool shall enable users to generate unique identifiers for each requirements. | RQ-001 |
| F-2.6 | The Innoslate™ tool shall allow users to label the requirements as specific requirement types, i.e., functional, non-functional, operational, system. | RQ-001 |
| F-2.7 | The Innsolate™ tool shall enable users to map requirements to architecture components and other systems engineering components. | RQ-002 |
| F-2.8 | The Innoslate™ tool shall enable users to populate the Requirements Report template with existing requirement entities. | RG-001 |
| **F-3** | **Document Analyzer** |  |
| F-3.1 | The Innoslate™ tool shall enable users to upload existing systems engineering artifacts using Word or PDF formats. | DA-001 |
| F-3.2 | The Innoslate™ tool shall allow users parse documents using automated and manual methods. | DA-002, DA-003 |
| F-3.3 | For automated parsing, the Innsoalte™ tool shall examine the documents template and parse using the same document framework. | DA-002 |
| F-3.3.1 | The Innoslate™ tool shall recognize document framework headers and body text. | DA-002 |
| F-3.3.2 | The Innoslate™ tool shall automatically create new entities with IDs, names, and content using content of uploaded document. | DA-002 |
| F-3.3.3 | The Innsolate™ tool shall use document’s headers and body text to determine IDs, names, and entity content. | DA-002 |
| F-3.3.4 | The Innoslate™ tool shall recognize document’s sections and sub-sections to establish parent-child relationships between new entities. | DA-002 |
| F-3.4 | For manual parsing, the Innoslate™ tool shall display uploaded documents in a viewer enabling user to see the document while creating entities. | DA-003 |
| F-3.4.1 | The Innsolate™ tool shall allow users to highlight content within the document and select where in the entity fields the content should be added. | DA-003 |
| **F-4** | **Reports Generator** |  |
| F-4.1 | The Innoslate™ tool shall enable users to generate project reports. | RG-001 |
| F-4.1.1 | The Innsolate™ tool shall generate Requirements Reports. | RG-001 |
| F-4.1.1.1 | The Innoslate™ tool shall offer user to generate different Requirements Report templates, i.e., Joint Capabilities Integration Development System (JCIDS), Operational, Functional, Systems. | RG-001, RG-004 |
| F-4.1.1.2 | The Innoslate™ tool shall allow users to assign existing requirement entities to sections within a Requirements Report. | RG-001 |
| F-4.1.1.3 | The Innsolate™ tool shall store Requirement Report templates within the tool. | RG-001 |
| F-4.1.1.4 | The Innoslate™ tool shall export Requirement Reports in Microsoft Word. | RG-005 |
| F-4.1.2 | The Innoslate™ tool shall enable generating a Concept of Operations (CONOPS). | RG-002 |
| F-4.1.2.1 | The Innoslate™ tool shall generate a CONOPS template. | RG-002 |
| F-4.1.2.2 | The Innoslate™ tool shall allow users to assign existing entities to sections within the CONOPS template. | RG-002 |
| F-4.1.2.3 | The Innoslate™ tool shall store the CONOPS template and associated content within the tool. | RG-002 |
| F-4.1.2.4 | The Innoslate™ tool shall export the CONOPS in Microsoft Word. | RG-005 |
| F-4.1.3 | The Innoslate™ tool shall enable users to generate an enterprise architecture report, i.e., Department of Defense Architecture Framework (DODAF). | RG-003 |
| F-4.1.4 | The Innoslate™ tool shall enable users to generate Project Management Plans. | RG-001 |
| F-4.1.5 | The Innoslate tool shall enable users to generate Risk Management Plans. | RG-001 |
| F-4.1.6 | The Innsolate™ tool shall enable users to generate Test and Evaluation Master Plans (TEMP). | RG-001 |
| F-4.1.7 | The Innsolate™ tool shall enable users to generate Implementation Plans. | RG-001 |
| F-4.2 | The Innsolate™ tool shall store exported reports in Microsoft Word in the tool. | RG-005 |
| **F-5** | **Collaboration** |  |
| F-5.1 | The Innoslate™ tool shall enable users to share projects. | DB-010 |
| F-5.1.1 | The Innsolate™ tool shall allow users to search for other Innoslate™ users/project members. | DB-010 |
| F-5.1.2 | The Innoslate™ tool shall allow users to assign user roles to projects, i.e., Read Only, Read/Write, Owner. | DB-010 |
| F-5.1.2.1 | Users with Owner permissions shall have full control of the shared project. | DB-010 |
| F-5.1.2.2 | Users with Read/Write permissions shall have the ability to add, modify, and delete project content. | DB-010 |
| F-5.1.2 | The Innsolate tool shall notify Read/Write users of project changes (e.g., modifications, deletions, additions). | DB-010 |
| F-5.1.3 | The Innoslate™ tool shall notify the Owner users of a project for any changes (e.g., modifications, deletions, additions). | DB-010 |
| F-5.1.4 | Users with Read Only permissions shall only read content. | DB-010 |

Table . Functional Requirements

#### Non-Functional Requirements

The team defined a series of non-functional requirements to assess the operations of the Innoslate™ against its expected behavior. The team has defined the non-functional requirements as follows:

|  |  |  |
| --- | --- | --- |
| **UID** | **Requirements** | **Test Case** |
| **NF-1** | **Accessibility** |  |
| NF-1.1 | The Innoslate™ tool shall be accessible on multiple browser platforms. | N/A |
| NF-1.2 | The Innoslate™ tool shall be accessible on Google Chrome. | N/A |
| NF-1.3 | The Innoslate™ tool shall be accessible on Mozilla Firefox. | N/A |
| NF-1.4 | The Innoslate™ tool shall be accessible on Internet Explorer version 8 and newer. | N/A |
| **NF-2** | **Response Time** |  |
| NF-2.1 | The Innoslate™ tool shall enable users to add and remove large batches of entities into/from the database. Large batch is defined as containing at least 100 entities. | DB-008 |
| NF-2.2 | The Innoslate™ tool shall enable users to import and export documents in less than 30 seconds. | DB-009, DB-010, DA-001, RG-005 |
| NF-2.3 | The Innolsate™ tool shall generate report templates and documents in less than 30 seconds. | RG-001, RG-004, RG-002 |
| **NF-3** | **Robustness** |  |
| NF-3.1 | In cases where inputs are invalid, the Innolsate™ tool shall not terminate prematurely, but instead display an error message describing the problem along with a troubleshooting solution. | DB-001, DB-004, DB-008, DB-009, DB-010, RQ-001, RQ-004, RQ-005, DA-001, DA-002, DA-003, RG-001, RG-002, RG-003 |
| **NF-4** | **Usability** |  |
| NF-4.1 | Users of Innoslate™ shall be required to have a minimal amount of training or experience in using systems engineering management tools to be able to operate the Innoslate™ tool. | N/A |
| NF-4.2 | Users shall be able to complete the following systems engineering activities within the Innoslate™: Project Management, Requirements Elicitation and Management, Concept Development, System Design and Architecture. | RQ-001, RQ-002, RQ-003, RQ-004, RQ-005, RG-001, RG-002, RG-003, RG-004 |

Table . Non-Functional Requirements

#### Test Cases

To test whether the tool is meeting the requirements defined above, the team developed a series of test cases to observe if the system is working correctly or not. Test results will be collected as pass or fail based on whether it met the requirements or not, and the results will be used by the team to make comprehensive recommendations to the sponsor.

#### Test Execution

Since Innoslate™ is a web-based tool; the tests will be executed using internet browsers to access to tool. Currently, the tool only operates using two specific browser platforms: Google Chrome and Mozilla Firefox. As a result, the team will form into two teams where each team will test against an assigned browser. Each team will capture the results of their tests in table formatted templates using a word processor tool. These results will be used to evaluate outcomes, identify gaps between browsers, and commonalities. This analysis will then be used to develop recommendations for the SPEC Innovations sponsor.

#### Test Results and Compilation

The team will execute and evaluate each of the features in pairs using the requirements traceability table in section 2.1.3.1 and 2.1.3.2. In the case where the team encounters problems during testing, the team will contact SPEC Innovations for troubleshooting support and provide further guidance the user guide may lack. To compile results for review and incorporation, analysis, and incorporation into the final report, pairs will use a pre-developed template where testers can comment on their results from testing the tool, which will then be evaluated and scored during analysis.

### Test Schedule

|  |  |  |  |
| --- | --- | --- | --- |
| **Testing Activity** | **Start Date** | **End Date** | **Group Member** |
| Develop Test Plan and Test Cases | 10/4/2012 | 10/12/2012 | Woldie |
| Conduct Functionality Testing | 10/14/2012 | 10/22/2012 | Ceely  Matthews Stevenson  Woldie |
| Conduct Usability Testing | 10/23/2012 | 10/28/2012 | Ceely  Matthews Stevenson  Woldie |
| Evaluate Test Results | 10/29/2012 | 11/11/2012 | Ceely  Matthews Stevenson  Woldie |

Table . Innoslate Test Schedule

### Test Cases

The Test Cases for each of the four Innoslate™ features selected for testing are presented below. Each test case will be executed using both the Google Chrome and the Mozilla Firefox web browsers.

#### Database Viewer

|  |  |  |
| --- | --- | --- |
| **Test Case Number** | **Description** | **Expected Result** |
| DB-001 | Creating, Saving, and Viewing Class Entities:  .Sign in into Innoslate  .Click on the “Create Entity” icon  .Select an entity to create (example, “Action”)  .Fill in the Name, Number, and Description  .Click on the “Save and Close” icon  .Follow the above steps to create all of the other available entities | . After doing a “Save and Close” for the last entity that you have created, make sure that all of the created entities are visible from the Database View.  .Namely, you should see the following class entities in the Database View: Action, Artifact, Asset, Characteristic, Cost, Decision, Input/Output, Link, Measure, Physical, Requirement, Resource, Risk, Software Interface, Statement, and Time |
| DB-002 | Searching the Database  .From the Database view, locate the Search textbox (Top-Right corner of screen)  .Type in the Full-Name of one of the entities you created during step DB-001  .Click on the Search Icon (the Blue Square icon with magnifying lens)  .Do the above step by only typing-in part of a name of an entity (example, if the Full Name is “My First Action”, type in only one of the words: “Action” or “My” or “First” or a combination of these | Should be able to search entities that are contained in the Database using both the Full-Name as we Part of the Name of the entity |
| DB-003 | Sorting Search Results from Database  .From the Database View, Click on each of the Attribute Columns (Number, Name, Class, Modified) | The search results are expected to be sorted by each respective attribute type that has been clicked for sorting (Number, Name, Class, Modified) |
| DB-004 | Filtering the Database View  .From the Database view, click on the “Filter Classes” command (Top-Left area of view)  .From the drop-down list, click on each entity types that appears | For each of the entity-type that has been clicked, all of the created entities for that specific type need to be returned by the database. For example, if you happen to have 2 Action entities previously created, both of these Action entities should be returned when selecting an Action entity from the drop-down list |
| DB-005 | Scrolling the Database View’s Window  .Make sure that you have more than 30 entities created in your test project  .Scroll up and down the Database window a few times | While scrolling down the Database window, once you reach the 30th entity, the remaining entities (the 31st and up) need to be added to the window’s view for viewing |
| DB-006 | Deleting Entities from Database View  .From the Database view, click on the square-box next to some of the entities  .Click the Delete command (Red Label on top of page menu)  .When asked “Are you sure you wish to delete?”, click on Confirm | The selected entities are expected to be deleted from the Database View |
| DB-007 | Deleting Entities from Inside the Entities view  .From the Database View, click on an entity (this takes you to the entity’s view)  .Click the Delete command (Red Label on top of page menu  .When asked “Are you sure you wish to delete?”, click on Confirm | The selected entities are expected to be deleted from the Database View |
| DB-008 | Importing Documents into the Tool  .From the Database View, click on the down-arrow next to the Import command (top of page)  .Click on “File” from the drop-down list  .Using the “Upload File” command, upload each file type (Xml, MPP, and KML) from your local machine into the tool | Expected to be able to import all 3 file types- Xml, MPP, and KML- successfully into the tool |
| DB-009 | Exporting Documents from the Tool  -From Database View, click on the Export command (top of page)  -Export by doing a filter-by Classes, Labels, and None (no filter applied) | Expected to export data from the tool, filtering by Classes as well as Labels. |
| DB-010 | Project Sharing  .Click the “Share” command (top right of screen)  .On project share window, enter the email address of user that you would like to share with and set their editing rights for the project (Owner, Read, Write)  .Click “Share”. The person’s Name and Email Address should pop-up under the list of users. Click “Done”  .When the person comes online; you should see a notification under the “Online Users” section (bottom-right corner).  .Click on “Online Users”. You should see the person in the list, and if he/she is online & available to chat.  .Click on the person’s name. A separate chat window should open, which allows you to chat with the person | Expected to allow chatting in real-time with an online user, while both parties are able to view the shared project |

Table . Database Viewer Test Case

#### Requirement Viewer

|  |  |  |
| --- | --- | --- |
| **Test Case Number** | **Description** | **Expected Result** |
| RQ-001 | Creating, Saving, and Viewing Requirement Entities:  *NOTE: In Innoslate, Requirements can be created in two ways – either from Database view by clicking ‘Create Entity’ or from ‘Requirements’ view by clicking ‘Create Requirement’.*  .Sign in into Innoslate  .Click on the “Create Entity” icon  .Select “Requirement” from the popup  .Fill in the Name, Number, and Description  .Click on the “Save and Close” icon  .Follow the above steps to create all of the requirements for MEEPAS project | After doing a “Save and Close” for the last requirement that you have created, make sure that all of the created requirements are visible from the Database View and Requirements view. |
| RQ-002 | Establishing Relationship with an Action:  .From the Database view, click on the “Create Entity” icon  .Select “Action” from the popup  .Fill in the Name, Number, and Description  .On the right side click ‘Add’ on ‘traced from Statement’ row  .From the popup select the requirement that this Action is satisfying  .Click on the “Save and Close” icon | After completing the Add operation you should be able to see the Requirement on the right side. You should be able to switch to the Requirement from Action screen and vice versa. |
| RQ-003 | Editing Requirements: From the Requirements view, Click ‘Edit’ on the row of the requirement that you want to edit. Modify Name, Number, and/or Description  .Click ‘Save and Close’. | The requirements view should reflect the change that you applied |
| RQ-004 | Obtaining Quality Scores for Requirements:  1.From the Requirements view, click on the “Validate Requirements”  2.From Requirements view select the requirement that you want to view/edit the quality score for and manually edit one or more of the nine quality score parameters (Clear, Complete, Consistent, Correct, etc.) | 1. Innoslate should calculate the quality score for each requirement based on a nine point quality check  2. Innoslate should automatically recalculate the quality score for the requirement and display it |
| RQ-005 | Create a requirement hierarchy: Create a top level requirement by following steps in RG-001. Using the same steps create a few child level requirements (decomposing the top level requirement). Edit the top level requirement and select ‘Add’ on ‘decomposed by Children’ at the right side. Select the child requirements | The child level requirements should added at the right side under the section ‘decomposed by Children’  Clicking ‘View’ on any of the child requirements should switch to the child requirement and should show the parent requirement under ‘decomposes Parent’ section |

Table . Requirements Viewer Test Case

#### Document Analyzer

|  |  |  |
| --- | --- | --- |
| **Test Case Number** | **Description** | **Expected Result** |
| DA-001 | **Uploading Documents using the Automated parser**  .Make sure you have a DOCX and a PDF file on your local computer ready to upload  .From Database View, click on the drop-down arrow next to the “Import” menu  .Choose “Analyzer”  .Click on “Upload Document” (Right Panel)  .Select the document to upload(either the PDF or the DOCX file)  .When prompted “Would you like to Auto parse this Document using Analyzer?”, choose a Starting Number (example 1.0) and click YES  .Once uploading completes, list of the created entities shows. Click on “Save Entities”  .Follow the same steps to upload the 2nd file sample (either the PDF or the DOCX file) | The uploaded documents are expected to be broken-down into separate paragraphs, with Names and Parent-Child relationships created based on the documents’ Titles and Numbering Schemes |
| DA-002 | **Uploading Documents using the Manual Parser**  .Follow the same steps as Step# DA-001 to upload both the DOCX and PDF sample, but when asked to auto parse using the Analyzer, choose NO  .Once uploading completes, you should see blank entity fields on left panel, and the uploaded document on right panel. Selecting a Text from the document (right panel) causes a hanger to appear, showing Name, Number, and Description entities. Clicking any of these entities will automatically insert them into their corresponding blank entity fields at left panel.  .Fill in the blank entity fields (Name, Number, and Description) using the steps described above  .Click on “Save and Reset”  . Follow the same steps to upload the 2nd file sample (either the PDF or the DOCX file | Expected to allow manually parsing the uploaded files |

Table . Document Analyzer Test Case

#### Report Generator

|  |  |  |
| --- | --- | --- |
| **Test Case Number** | **Description** | **Expected Result** |
| RG-001 | **Class-Based Reports:**  1. In Database View click ‘Reports’.  2.From the ‘Generate Report’ dialog select ‘Class Based Report’  3.From the ‘Class Based Report’ list, select ‘Requirement Reports  4.Pick ‘Requirement Class Summary Report’ from the dropdown  5.Click ‘Submit’  6. Repeat steps 1 to 5 selecting ‘Requirements Class Entity Report’ and ‘Requirement Class Selected Entity Report’ at step 4. | Must be able to generate, download, and view reports on individual and all class entities |
| RG-002 | **General Reports:**  1. In Database View click ‘Reports’.  2.From the ‘Generate Report’ dialog select ‘General Report’  3.From the ‘General Report’ list, select ‘CONOPS’  *NOTE: If Concept of Operations has not been defined yet, Innoslate will invoke the CONOPS Data Entry Wizard. The report generation will continue after you finish the CONOPS data entry*  4. Select Generate CONOPS from the dropdown  5. Click ‘Submit’ | Must be able to generate, download, and view general reports |
| RG-003 | **DODAF Reports:**  1. In Database View click ‘Reports’.  2.From the ‘Generate Report’ dialog select ‘DoDAF Reports’  3. Select ‘DoDAF OV-1’  4. Click Submit | Must be able to generate, download, and view DoDAF reports |
| RG-004 | **JCIDS Reports:**  1. In Database View click ‘Reports’.  2.From the ‘Generate Report’ dialog select ‘JCIDS Reports’  3. Select ‘JCIDS CDD Report’  4. Click ‘Submit’  *Note: If JCIDS Capability Development Document has not been created yet, Innoslate will launch the JCIDS CDD creation Wizard. Report generation will continue after the CDD is created.* | Must be able to generate, download, and view JCIDS reports |
| RG-005 | **View Reports:**  1. Go to ‘Database’ view  2. Scroll to the bottom  3. Select one of the reports just generated  4. Click ‘Download’  5. Verify that the report contains expected information | Should be able to open the Report entity and view in the appropriate format |

Table . Report Generator Test Case

# Test Case Results

To best document the results of the team’s test cases, results from both web browsers were evaluated and documented as the overall experience. The team feels this demonstrates areas where the tool is consistent across both browsers. To highlight where the team observed differences, those experiences will be specifically called out throughout the results. In addition, completed requirements traceability matrices demonstrating whether features of Innoslate™ passed or failed in meeting the functional and non-functional requirements is available in Appendices B and C.

## Database Viewer

An Innoslate project consists of a collection of Systems Engineering entities and their inter-relationships stored in a database. Hence the Database Viewer is a major capability of Innoslate using which a user can create, edit, view, search, filter, sort, delete, import, or export entities, as illustrated in Figure 1. In addition, user can upload a structured document – Microsoft Word document or PDF - that contains entity definitions into Innoslate. Innoslate can parse the document and generate entities based on the contents of the document.

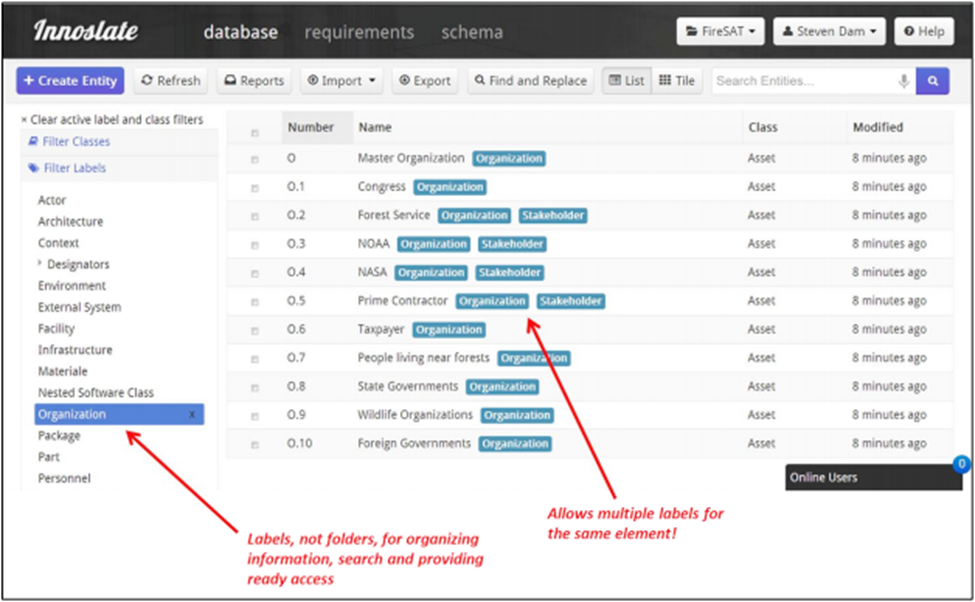


Figure 1. Innoslate Database Viewer [1]

### Data Entry

The database capability enabled users to easily input entities into the tool. It was observed that while inputting entities, users had to be knowledgeable about specific attributes for the data that was to be entered into the tool, i.e. deciding naming conventions and numbering hierarchies were required fields when entering data. However, users did agree that the context sensitive help (mouse-over dialog windows) related to entity creation was very helpful.

Store/Save/Delete –The database capability enabled users perform the key functions of saving, storing, and viewing all type of class entities. Users observed issues with deleting large numbers of entries from within the database. In addition, users observed that after completing the entity deletion for a smaller number of rows, the delete button remained inactive and the text read "Deleting". This behavior required a refresh of the Database View several times before additional items could be selected for deletion.

### File Import/Upload/Export/Deletion

Users found that performing a file import/upload associated with a particular entity, i.e., artifact, statement, etc, to be user-friendly, however, there is no visual cue or reference that indicates a file is attached once import/upload is completed.  Users observed that it is not clear how files attached to a particular entity would be represented in a generated report or export of a series of entities. In addition, users had difficulty deleting files attached to entities within attached files associated with an entity. Based on testing of the file upload function within the entity edit screen, only initial and overwritten uploads appear to be possible.

### Search/Filter/Sort

Users were able to successfully search the database’s contents using full as well as part of names and filtering by classes and labels established by the user within the tool. Users observed that sorting on modified date only displayed newest to oldest instead of the typical sorting behavior that orders items ascending and descending.

Users also tested the sorting capability on searched entities discovering that the tool was unable to sort on the ‘Name’ and ‘Class’ entities. For entities that are sortable, the tool only allowed users to sort data in one direction.

### Summary

The Database functionality of Innoslate™ was found to be a robustly designed and developed feature that allows user to create, save, and view various types of class entities such as Assets, Artifacts, and Requirement statements. In addition to allowing users to dynamically sort and search the created entities, it enables filtering the displayed view to limit the number of class entities that are returned by the view, which becomes significantly beneficial when having large number of entities in a given project. For relatively small number of entities, the user can simply scroll-down the view screen to locate the desired entity from the list.

Moreover, the feature allows importing external documents that are associated with particular class entities, as well as exporting existing entities from the tool to a desired location under the user’s local environment. A particular function that was found to be useful was the existence of two ways of saving a newly created class entity- ‘Saving’ versus ‘Saving and Closing’; whereas the former records any changes to the database and returns the user to the database view, the latter records changes to the database while simultaneously allows the user to make further changes to the just-created entity.

Last but not least, the database feature allows the ability to share projects and workspaces with other Innoslate™ users.

## Requirements Viewer

The Requirements View provides a simple, document-like view of Statements and Requirements entities. As noted in the figure below these entities cover most of the window in the view. A user may select one or more entities using the check boxes to the left of the information and a red “Delete” button will appear on the button bar above. This feature enables you to delete multiple entities all at once, rather than one by one. A user may also click into the name, number and description fields and edit them directly in this view, or select the “Edit” button on the far right of the screen to view and edit all the entity attributes. One item of note is the quality score that is provided, for which a user may obtain automatically (for several of the quality attributes, such as Clear and Complete) by selecting the “Validate Requirements” button. A user may also adjust the quality factors by editing the specific attributes related to scoring a requirement within the entity edit screen. The Requirements View is very helpful in organizing and partitioning system needs from the Database View, which includes all known entities for a given Innoslate™ project [1].

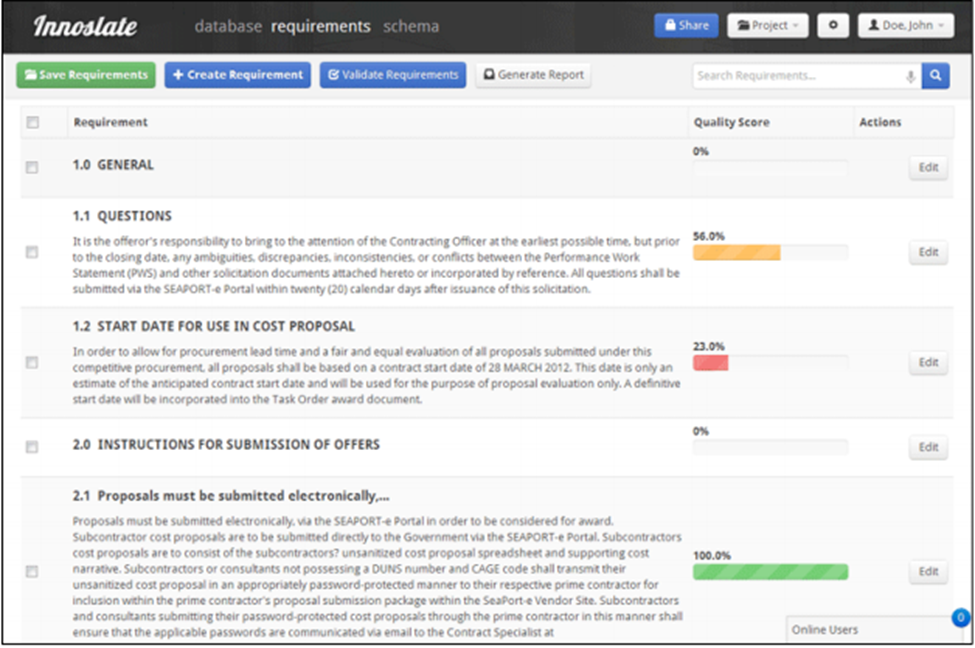


Figure 2. Innoslate Requirements Viewer [1]

### Requirements Creation/Saving/Editing/Deleting

Users were able to create and save requirements successfully; however users did observe instances where a requirement did not save properly or was not displayed in the refreshed requirements view. Users observed that upon entering requirements, there was no apparent way to validate a requirement’s identification value was unique, which could lead users to developing requirements with duplicate unique identifiers. This scenario required the user to navigate back and forth from the requirements view browse screen to the requirement edit screen to validate the unique identifier value. Users were also able to edit requirements successfully by clicking on the requirement description opens an editor for modifying the text, however, clicking out of the editor caused the underlying HTML tags to be exposed to the user. As a result, users were required to click the “Refresh” button at the top of screen to re-display the requirement description text properly, which reset the view. While editing a previously created requirement, users noted that the requirement description editor displayed odd behavior when hitting the tab key as it caused all of the text to change to italics. As a test to remove a large number of requirements, users attempted to delete a large number of requirements to observe if the same issue that occurred within the Database View occurred. Users observed that the deletion of a large number of selected entries (more than 100) was not successful, as the application hung and did not respond after a lengthy period of time (more than 5 minutes). However, users did observe that the deletion of requirements entries in smaller batches (50) was successful.

### Requirements Validation

Users observed that when clicking the validate button the tool appears to hang. The button display changes as though it is working, but nothing further happens and requires exiting out of the requirement edit view to the browse view. Selecting the newly created requirement for edit does not provide any particular guide as to whether any scoring of the requirement as the output of the validate function was successfully completed. In addition, users found that tool did not validate requirements consistently either from the requirements view or create/edit screens. On several instances when clicking the “Validate” button from the browse view and edit screen, the process appeared to hang or not complete successfully resulting in the perception that the function was slow or not operational. The validate requirements scoring result appeared to be inconsistent, as two requirements that with almost identical details with one minor difference received very different scores. The requirement validation and scoring capability entry may allow for bias if the requirements author is performing the scoring. This is essentially the ability for the author of a requirement to modify the scoring if the automated validation does not provide the result that is desired or needed.

### Requirements Reports

Users successfully generated the requirements document sections using the report function, however, users felt that it was not intuitive on how to actually get requirement entities already established within the tool entered into the requirement document template generated by the tool. The task of moving requirements entities from the Requirements View into the requirements document template is not discussed in depth in the user guide.

### Summary

Based on the overall experience in using the Requirements Viewer, users feel that the capability still needs to evolve to offer more robust functionality for collecting and managing project requirements. Users did find the validation feature extremely useful enabling systems engineers evaluate the quality of their requirements, which is imperative when the quality of a requirement may impact the design of a solution. The ability to easily create predefined and custom relations to other entities is a very powerful feature that helps to build traceability into the project. It would have been even more useful if there were better charting and diagramming capabilities to visually represent the various relations thus created. Another useful addition would be the capability to add a batch of requirements.

## Document Analyzer

The Analyzer feature enables users to import and parse information from PDF and Word files into the Innoslate database. Users can extract information and organize it by selecting any of the classes offered within the Innoslate. In addition to the analyzer, users can manually parse documents, extracting information and organizing it similar to the analyzer function [1].

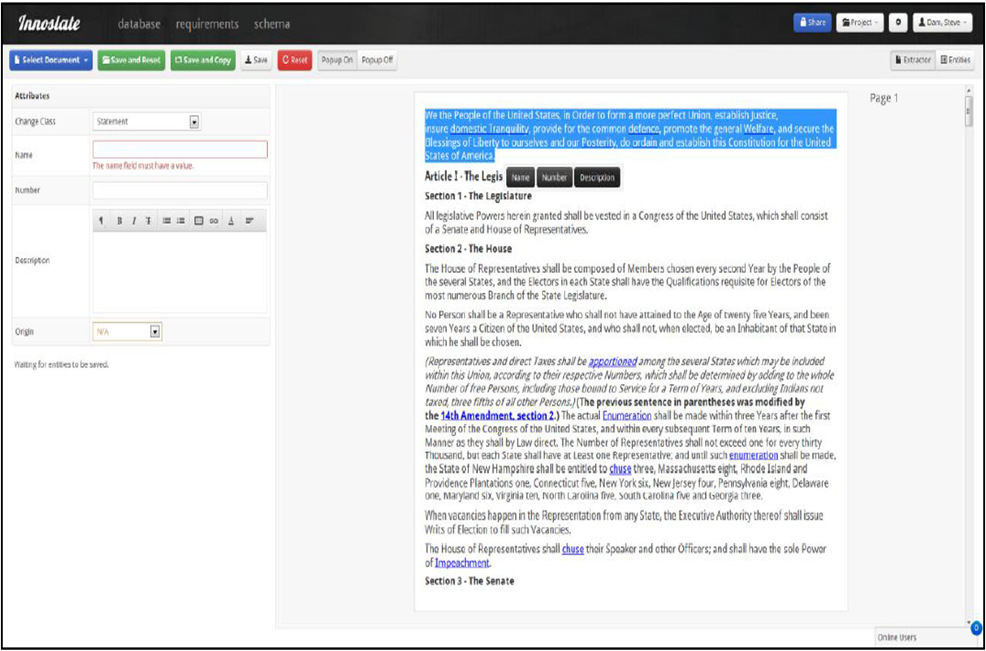


Figure 3. Innoslate Analyzer View [1]

### Analyzer

Users of the Document Analyzer found that with minimal use of the User Guide, the tool seemed intuitive about how to import a document. The users observed that the “Import” button was easy to locate on the Database View. While importing PDF files, Google Chrome users observed that the Analyzer does not import the entire document. It only imports three sections of the document and the parsed sections seem random. Innoslate™ successfully stores the PDF and allows users to download the document while opening the PDF in a browser. PDF appears to maintain its integrity once downloaded from Innoslate™.

Users successfully imported Word files where the Analyzer parses the document into statement entities, but users could not analyze the results of the parsing because the tool gives a quota error. In addition, the tool seems to freeze while trying to save and analyze any statements the tool identified. Once an MS-Word file is saved in Innoslate™, users were able to successfully download and open the file. Users observed that the integrity of the document, i.e., formatting, is maintained.

### Manual Parser

Users successfully imported and manually extracted all of the sections of the MS-Word file. In the manual parser, users were able to highlight section titles, paragraphs, and numbers to create entities that were saved into the Database View. Users found the parser easy to use, but a little labor intensive. To reopen the imported document in extractor mode, users observed that it was not intuitive on how to do this and the user guide does not provide guidance on how to navigate back in order to add more entities to the database. After more extensive testing with the tool, users found the “Analyze” button, which allowed users to enter the extractor mode and begin adding more entities.

For PDFs, users successfully imported a PDF file, but the file does not appear in the extractor mode preventing users from manually extracting sections from the document. Users tried to open the artifact and reanalyze the document, but the document does not appear on the right-hand side screen display to manually pull sections from.

### Summary

Users found the Analyzer View to offer the functionality as described in the user guide. As described in the previous sections, users did observe some bugs in the capability, which limited the users’ ability to fully evaluate the tool. However, based on the overall experience in using the capability, users feel that this capability enables systems engineers to easily archive systems engineering artifacts in a single location. Using such a capability may alleviate systems engineers from acquiring and managing other content management tools such as SharePoint to store and update documents. It also offers users the ability to easily leverage content from existing artifacts, establish relationships between artifacts and other systems engineering deliverables, i.e., requirements traceability to CONOPS content, architecture mapping to requirements/operational concepts.

## Report Generation

Innoslate™ has a number of reports available, and are grouped into various categories including: Class Based Reports that provide reports on individual and all class entities; General Reports that provide CONOPS, Requirements Documents, Risk Characterization, and Schema Report; DoD Architecture Framework (DoDAF) Reports, such as the CV-1 through 7, OV-3, PV-1 through PV-3, SV-3, SV-5, SV-8 and Services View versions); and Joint Capabilities Integration Development System (JCIDS) documents, such as the ICD, CDD, CPD, and DCR.

As many of these reports require a complicated set of inputs and proper storage of the information to generate the report, we have developed specific report “wizards” to help capture the information needed, put the information in the correct classes, make the appropriate relationships, and attach the necessary labels automatically. The screen noted below shows a CONOPS example. All reports are captured as Artifacts in the database and may be opened and viewed (download) as a Microsoft Word file in DOCX format or Comma Separated Values (CVS) format [1].

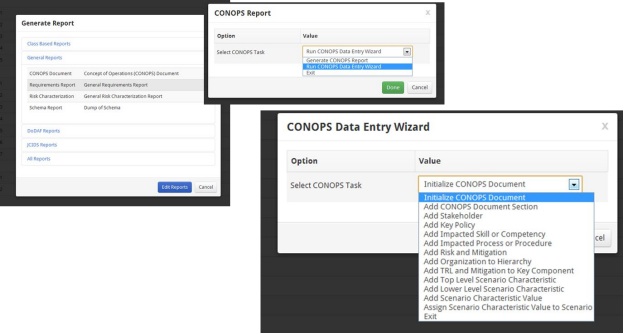


Figure 4 - Report Generation

### Concept of Operations

While initializing a CONOPS document utilizing the CONOPS Data Entry Wizard, users initially observed that the tool goes into an endless loop of prompts. However, after looking at the User Guide and playing with the tool, users discovered that the continuous prompts were used to have the user begin selecting tasks to complete the CONOPS, i.e., add CONOP sections, adding tables to stakeholder section, etc. The CONOPS Wizard does establish a straw man template, however, users observed that using the wizard to complete tasks in developing the CONOPS, but all of tasks that were selected from the list lead to an error. In addition, users couldn’t identify how to incorporate existing CONOPS statement entities into the auto-generated template.

### Class Based Report

Users were successful at generating a Class Based Summary Report, saving it in Innoslate™, and downloading the document into a Word file.

### Department of Defense Architecture Framework Report

Users observed that the Department of Defense Architecture Framework (DoDAF) report failed to generate a document. Users discovered that the DoDAF views are not yet implemented in the current version of Innoslate™.

### Joint Capabilities Integration Development System

Innoslate has support for Joint Capabilities Integration Development System (JCIDS) Reports.

These are the JCIDS reports that are supported:

1. Capability Development Document (CDD)
2. Capability Production Document (CPD)
3. Initial Capabilities Document (ICD)
4. Joint DOTMLPF-P Change Recommendation (DCR)

Users tested the JCIDS Capability Development Document (CDD) report and the result was satisfactory. Before generating the report, Innoslate checks to see if a JCIDS Capability Development Document exists. If it has not been created yet, Innoslate will launch the JCIDS CDD creation Wizard. Report generation will continue after the CDD is created. As with the other reports, the generated report gets added to the Database view. The user will have to scroll down the list of entities in the Database view or specify a filter on Artifacts to view the report. It would be helpful if there was an easier mechanism for opening a report that has just been generated.

### Summary

Users found the Report Generation feature offered the functionality described in the user guide. As described in the previous sections, users did observe some bugs in the capability, which limited the users’ ability to fully evaluate the tool. However, based on the overall experience in using the capability, users feel that this capability offers an impressive list of reports and diagrams that can be easily generated tailored to meet a systems engineers. However, not all reports in the list have been implemented, i.e., DoDAF views and reports. A neat feature of the Reporting capability of Innoslate is that it can create the underlying document framework required for reports like JCIDS on the fly. Innoslate detects if the required document framework exists for the selected report. If it does not exist, Innoslate prompts the user to create and initialize the framework first before proceeding with the generation of the report. Users found the report generation feature user friendly and generated reports in a timely fashion, and found the storing of the report in the Innoslate database appropriate for easy content management.

## Miscellaneous

### Automatic Logout

It was noted that the application does not log you out after a certain amount of idle time or prompt the user that their session will end due to idleness, which can be a security concern.

### Collaboration

collaborate_screen.tiffFigure 5 - Project Collaboration

#### Project Administration

Users can easily create projects and share with other members of their team. Collaborating is straight forward with the “Share” button being easily located at the top of the Database View. Once the Share button is selected, users can identify themselves as the owner of the project and add collaborators by entering team members email addresses. Users can then define the team members’ user role, i.e., Owner, Read/Write, and Read, allowing for easy management of user access.

#### Project Sharing

Innoslate™ successfully shared team members’ projects and implemented user roles appropriately.

#### Project Modification and Deletion

While sharing a project with other team members, it was observed that while team members assigned the “owner” role in Innoslate™ deleted projects, the tool deleted all data associated with the project without confirmation. In addition, this caused other team members to lose their work as the project was deleted from their account as well.

# Recommendations

## Database View

### Entity Creation/Saving/Editing/Deletion

1. It would be useful when viewing or editing an entity (artifact, class, requirement) that has an existing file(s) attached to it to add some sort of graphic display that indicates a file has been previously uploaded/attached. Currently there is no way for the user to know an entity has a file attached to it other than clicking the Download link.
2. To facilitate efficient entity creation there need to be a way to create multiple entities without having to exit to Database/Requirements view. For example when a user creates requirements usually he/she needs to create a requirement hierarchy with multiple requirements. In the current version of Innoslate™, after creating each requirement user is taken back to the Database/Requirements view. User needs to click Create Entity or Create Requirement multiple times to complete requirements creation, which results in very inefficient use of time.
3. During entity creation or editing, if the user navigates away from the create/edit screen, all data entered on the screen is lost. Currently there is no warning alert if user navigates away from create/edit screen. There need to be an alert to caution the user about data loss, when he/she tries to navigate away from the page.
4. A checklist for necessary entities to support specific reports that can be generated would be helpful. This could be a task list or set of required entities as a trigger for reports that are enabled.

### Importing/Uploading/Exporting/Downloading

1. Database export feature need to be fine-tuned to allow exporting of entities selectively, so that selected entities can be exchanged between projects.
2. Enable users to download artifacts/files from Database View; currently users must open edit view to download artifacts/files
3. Enable users to view imported/uploaded/exported artifacts/files within Innoslate™ without downloading artifacts/files to view them

## Requirements Viewer

### Requirements Entry/Saving/Editing/Deleting

1. To enable efficient creation of requirements within Innoslate™, it is recommended that an auto-generated unique identifier mechanism be implemented that auto-populates the ID field for a requirements. Auto-generated values should be based on the parent-child relationship established by the user while creating the requirements. In addition or as an alternative, a visual display noting the current numbering scheme would enable the user to visually identify where they are in the requirements hierarchy during requirements creation.
2. To enable the user to better display the requirements in a hierarchical manner, it is recommended that the boxes in the hierarchical chart be auto-fit to allow the title of the requirements to be completely visible; currently the requirements-title are cutoff in the chart causing the user to manually increase the size of the box.

### Requirements Validation

To enable the user to have situational awareness about where the tool is in terms of completion the requirements validation process, it is recommended that a status display or progress bar be implemented offering a visual of how much longer the process will take; currently the validation operates by disabling the Validate button and changes the text to “Validating” until complete, however it is not clear how long the process will take to complete.

### Requirements Reports

To enable the user to efficiently generate a requirements document using existing requirements entries within Innoslate™, it is recommended that the User Guide be updated to further describe in detail how to utilize the report artifact tools to generate a requirements document that is based primarily on entered requirements entries; users found it challenging to understand how to get the report function working properly to integrate entered requirements into a pre-generated requirements document. The need to add multiple requirements at once is applicable for requirements also.

## Document Analyzer

At this time, the only recommendation for the document analyzer is to address the issues with uploading and parsing PDFs, and correcting the quota error, which prevents users from reviewing the parsed Word document. Once these are corrected, recommendations in regards to the capabilities functionality can be better addressed.

## Report Generation

1. DODAF is a commonly used enterprise architecture (EA) framework used across the Federal Government therefore, it is recommended that this capability completed and implemented soon to allow for user feedback.
2. To improve the users’ accessibility to newly generated reports, it is recommended that a ‘View Report’ button be provided on the 'Report Completed' dialog, which would directly take the user to the report just generated.

## Miscellaneous

### Security

Implement a measure that ensures users are not logged in continually and provide a prompt to continue a session after a certain period of idleness. This would alleviate security concerns from a data sensitivity standpoint, and additionally may be wise from an overall system performance perspective.

### Collaboration

1. To ensure appropriate team members are receiving collaborate requests at the appropriate email address, it is recommended that a global list of Innoslate™ users be available and searchable.
2. To ensure that pertinent content cannot be deleted from the project, it is recommended that:
   1. Only owners of a project have the ability to delete content or implement a request statement where a user with Read/Write access must request permission to delete content; or
   2. A flagging mechanism be implemented that allows the Read/Write user to flag content for deletion and the owner can approve it, therefore automatically removing it from the tool; or
   3. Implement an archive where the owner of the project can see a record of what has been deleted with an associated deletion date.

### Other Aspects

#### Document Object Model

It is recommended that Innoslate™ be able to publish a Document Object Model (DOM) and expose the DOM to an Application Programming Interface (API) using a script language like JavaScript so that an SE project and its entities can be created and manipulated programmatically which will considerably boost productivity of Systems Engineers.

#### Systems Engineering Task List

It is recommended that the Innoslate™ a feature that is cross-functional that provides a checklist or task list to the user offering a guide for entering data/entities into the tool that will track to the systems/software engineering lifecycle (SELC) model or architecture frameworks to assist in the development of artifact outputs would be very useful in acclimating to using the tool to complete systems engineering tasks. This could be a project specific category selection that pre-determines a set of required entities that need to be added before certain kinds of reports can be generated might be another way of guiding users as it can be a bit overwhelming when using the tool for the first time.

## Future Recommended Innoslate™ Capabilities

### Project Management

One particular area that Innoslate™ does not address that is important to systems engineers during the course of their work on project and program is project management. The following potential features have been identified to address project management aspects of systems engineering and assist in developing Innoslate™’s maturity.

#### Project Schedule Capability

Implement a project management scheduling capability that includes functions to build a project schedule, develop project tasking, assign tasks to users that are collaborating on the same project, and generate output reports that assist in project tracking (Gantt chart, critical path analysis chart or Program Review and Evaluation (PERT) chart).

#### Project Schedule Monitoring Capability

Integrate a capability within the tool that is associated with project management scheduling and task tracking that aligns a project schedule, tasks, and individuals that are responsible for those tasks to the defined deliverables and artifacts that are required as outputs of the project or program. This capability would include functions that could automatically assess the completeness of artifacts that in the midst of development based on existing entities and reports that have been initiated within Innoslate™. The assessment would then have a reporting component that would allow project manager to get a more accurate picture of the progress for deliverables that are associated with the milestone within the project schedule. The information provided would assist in mitigating areas of schedule risk and enable project teams to better focus their attention to tasks that may have issues or roadblocks that are causing delays.

#### Cost Estimation

Integrating or developing a cost estimate tool that could auto generate estimates based on requirements, use cases, models, and other data entities that are relevant against known system development cost estimate heuristics and past performance to enable more efficient and accurate cost estimation processes.

### Systems Engineering Lifecycle (SELC) Workflow

In addition to implementing more project management features, the team also recommends the Innoslate have a SELC workflow feature implemented in future versions to enable a Systems Engineer to execute activities and develop artifacts specific to a given SELC phase. A user of tool would be able to open Innoslate™ and be able to select what phase of the life cycle he/she is in. Once selected, users would be offered a list of recommended foundational SELC documents that one should consider developing during this phase. In addition, the tool could highlight what artifacts are dependent upon another and indicate what content could be leverages or mapped to other SELC artifacts. This may enable Systems Engineers to more efficiently implement the practice of traceability, which can be lacking in a systems engineering project.

In addition to implementing this capability, if not doing so already, may want to consider tailoring this feature to specific customers enabling them to add to the list of required documents required for their organization (e.g., Department of Defense (DoD) Acquisition and SELC artifacts and activities). This allows an organization to better prepare their engineers to meet the acquisition requirements put in place to acquire a system.

# Summary

Innoslate introduces a refreshingly new paradigm for defining and managing Systems Engineering projects. The cloud based approach makes it easy for a team of Systems Engineers to collaboratively create and maintain artifacts on a large scale project. Both the screen layouts and procedures are consistent throughout the tool helping in easy learning. Some of the features need polishing to be effective. Certain features like DoDAF views are not fully implemented yet. But overall users felt that with this tool SPEC Innovations has taken a step in the right direction. With some improvements in usability and fine tuning of the capabilities mentioned within this report, Innoslate has the potential to become a valuable tool in the toolset of a Systems Engineer. Listed below are some of the major pros and cons of the current version of the tool:

Pros:

* Consistent look and feel across create/edit screens for various entities and artifacts
* Promotes collaboration among team members
* Offers a single location to create, store, and management systems engineering artifacts
* Capability to define custom schemas is a very powerful feature

Cons:

* Users run into too many bugs, which hinder their ability to complete tasks
* Lacks robustness in handling a large number of entities and importing big documents using document analyzer tool
* Need to improve usability by aligning itself with the way SELC workflow is implemented
* Limited audience of users because the tool is only usable on two browser platforms, many organizations operate on Internet Explorer, specifically older version

In addition to the summarized pros and cons listed above, the team wanted to highlight some next steps that SPEC Innovations may want to consider for moving forward, if they have not considered these already.

* User Forums – To continue maturing Innoslate™ to meet a broader systems engineering audience, it is recommended that SPEC Innovations establish or host user forums. Attendees should be members of industry and government to identify growing trends in the field, i.e., complex systems, and how Innoslate™ can continue meetings those needs. In addition, these forums could be used to discuss issues in using the current capabilities of the tool and receive feedback from the community about what’s working and what is not.
* Market Analysis – If SPEC Innovations has not already developed a market analysis to identify systems engineering management tool competitors in the market, it is recommended that this be done. This market analysis can enable SPEC Innovations to identify their strong points in the field and focus their efforts on mastering capabilities Innoslate™ excels at. In addition, the market analysis may also drive future features that could be included in the tool based on trending topics.
* Comparative Trade Study Evaluation – one opportunity for a possible follow-on GMU Master of Systems Engineering capstone project could be an independent comparative analysis of Innoslate against leading systems engineering management tools (e.g., IBM Rational DOORS, Caliber, Enterprise Architect). This kind of a trade study may offer feedback as to Strengths, Weaknesses, Opportunities, and Threats (SWOT) and assist a future team in evaluating the list of recommended modifications and enhancements to Innoslate™ that have been described within this report that offering the most potential value.

# Appendix A: References

[1] Spec Innovations. *Innoslate™ User Guide*. Version 1.0, October 2012. [http://www.Innoslate.com/wp-content/uploads/2012/08/users\_guide.pdf](http://www.innoslate.com/wp-content/uploads/2012/08/users_guide.pdf)

# Appendix B: Requirements Traceability – Functional

|  |  |  |  |
| --- | --- | --- | --- |
| **UID** | **Requirements** | **Test Case** | **Pass/Fail** |
| **F-1** | **Database Viewer** |  |  |
| F-1.1 | The Innoslate™ shall enable users to manage project data. | DB-001 |  |
| F-1.1.1 | The Innoslate™ shall allow users to create class entities. | DB-001 | Pass |
| F-1.1.2 | The Innoslate™ shall allow users to store class entities. | DB-002 | Pass |
| F-1.1.3 | The Innoslate™ shall allow users to edit class entities. | DB-001 | Pass |
| F-1.1.4 | The Innoslate™ shall allow users to view all class entities in a single location. | DB-003 | Pass |
| F-1.1.5 | The Innoslate™ shall allow users to filter class entities using class type or label. | Db-006 | Pass |
| F-1.1.6 | The Innoslalte™ shall allow users to sort class entities using name, class type, ID, dates. | DB-005 | Pass |
| F-1.1.7 | The Innoslate™ shall allow users to sort class entities in descending and ascending orders. | DB-005 | Fail |
| F-1.1.8 | The Innoslate™ shall allow users to search class entities using the entities’ full name and partial name. | DB-004 | Pass |
| F-1.1.8.1 | Search results shall be filterable and sortable. | DB-004, DB-005 | Fail |
| F-1.1.10 | The Innoslate™ shall allow users to delete class entities from the tool. | DB-008 | Fail – Based on deletion requirements set by the users |
| F-1.1.10.1 | The Innoslate™ shall prompt users to confirm that their selected class entities are being deleted. | DB-008 | Fail |
| F-1.1.10.2 | The Innoslate™ shall archive deleted class entities for 30 days to enable users to cross check that deleted class entities were not pertinent to the success of a project. | DB-008 | Fail |
| F-1.1.10.3 | For large batch deletions, the Innoslate™ shall enable users to delete at least 100 class entities at a time. | DB-008 | Fail |
| F-1.1.10.4 | The Innoslate™ shall allow users to delete small batches of class entities. Small batches is defined as 50 and less. | DB-008 | Pass |
| F-1.1.15 | The Innoslate™ shall allow users to import class entities using an XML file. | DB-009 | Pass |
| F-1.1.16 | The Innoslate™ shall allow users to export class entities into an XML file. | DB-010 | Pass |
| **F-2** | **Requirements Viewer** |  |  |
| F-2.1 | The Innoslate™ shall allow users to manage requirements. |  |  |
| F-2.1.1 | The Innoslate™ shall allow users to create requirement entities. | RQ-001 | Pass |
| F-2.1.2 | The Innoslate™ shall allow users to store requirements entities. | DB-002 | Pass |
| F-2.1.3 | The Innoslate™ shall allow users to edit requirement entities. | RQ-003 | Pass |
| F-2.1.4 | The Innoslate™ shall allow users to view all requirements in a view separate from database entities. |  | Pass |
| F-2.2 | The Innoslate™ shall assess the quality of a requirement using a scale of 100%. | RQ-004, RQ-005 | Pass |
| F-2.2.1 | The Innoslate™ shall assign a quality value to a requirement. | RQ-004, RQ-005 | Pass |
| F-2.2.2 | The Innsolate™ shall display the quality score next to the requirement. | RQ-004, RQ-005 | Pass |
| F-2.3 | The Innoslate™ shall enable users to decompose requirements. | RQ-005 | Pass |
| F-2.4 | The Innoslate™ shall allow users to display the decomposition of the requirements hierarchy. | RQ-005 | Pass |
| F-2.5 | The Innsolate shall auto generate unique identifiers for each requirements as entered into the system and mapped to a parent requirement. | RQ-001 | Fail |
| F-2.6 | The Innoslate™ shall allow users to label the requirements as specific requirement types, i.e., functional, non-functional, operational, system. | RQ-001 |  |
| F-2.7 | The Innsolate™ shall enable users to map requirements to architecture components and other systems engineering components. | RQ-002 | Pass |
| F-2.8 | The Innoslate™ shall enable users to populate the Requirements Report template with existing requirement entities. | RG-001 | Pass |
| **F-3** | **Document Analyzer** |  |  |
| F-3.1 | The Innoslate™ shall enable users to upload existing systems engineering artifacts using Word or PDF formats. | DA-001 | Pass |
| F-3.2 | The Innoslate™ shall allow users parse document using automated and manual methods. | DA-002, DA-003 | Pass |
| F-3.3 | For automated parsing, the Innsoalte™ shall examine the documents template and parse using the same document framework. | DA-002 | Fail – Only parses PDF files partially |
| F-3.3.1 | The Innoslate™ shall recognize document framework headers and body text. | DA-002 | Fail – Only parsed Word files using headers and body text |
| F-3.3.2 | The Innoslate™ shall automatically create new entities with IDs, names, and content using content of uploaded document. | DA-002 |  |
| F-3.3.3 | The Innsolate™ shall use document’s headers and body text to determine IDs, names, and entity content. | DA-002 | Fail – Only parsed Word files using headers and body text |
| F-3.3.4 | The Innoslate™ shall recognize document’s sections and sub-sections to establish parent-child relationships between new entities. | DA-002 |  |
| F-3.4 | For manual parsing, the Innoslate™ shall display uploaded document in a viewer enabling user to see the document while creating entities. | DA-003 | Fail – PDF files did not display in the extractor; only MS Word files appeared |
| F-3.4.1 | The Innsolate™ shall allow users to highlight content within the document and select where in the entity fields the content should be added. | DA-003 | Fail – Only MS Word files were extractable; unable to test PDFs |
| **F-4** | **Reports Generator** |  |  |
| F-4.1 | The Innoslate™ shall enable users to generate project reports. | RG-001 |  |
| F-4.1.1 | The Innoslate™ shall generate Requirements Reports. | RG-001 | Pass |
| F-4.1.1.1 | The Innoslate™ shall offer user to generate different Requirements Report templates, i.e., Joint Capabilities Integration Development System (JCIDS), Operational, Functional, Systems. | RG-001, RG-004 | Pass |
| F-4.1.1.2 | The Innoslate™ shall allow users to assign existing requirement entities to sections within a Requirements Report. | RG-001 | Fail – Only because users were unable to determine how move existiny DB entities into report templates; not described in user manual |
| F-4.1.1.3 | The Innsolate™ shall store Requirement Report templates within the tool. | RG-001 | Pass |
| F-4.1.1.4 | The Innoslate™ shall export Requirement Reports in Microsoft Word. | RG-005 | Pass |
| F-4.1.2 | The Innoslate™ shall generate a Concept of Operations (CONOPS). | RG-002 | Pass |
| F-4.1.2.1 | The Innoslate™ shall generate a CONOPS template. | RG-002 | Pass |
| F-4.1.2.2 | The Innoslate™ shall allow users to assign existing entities to sections within the CONOPS template. | RG-002 | Fail (refere to Req. F-4.1.1.2 results) |
| F-4.1.2.3 | The Innoslate™ shall store the CONOPS template and associated content within the tool. | RG-002 | Pass |
| F-4.1.2.4 | The Innoslate™ shall export the CONOPS in Microsoft Word. | RG-005 | Pass |
| F-4.1.3 | The Innoslate™ shall enable users to generate an enterprise architecture report, i.e., Department of Defense Architecture Framework (DODAF). | RG-003 | Fail – Not implemented yet |
| F-4.1.4 | The Innoslate™ shall enable users to generate Project Management Plans. | RG-001 | Fail – Not offered |
| F-4.1.5 | The Innoslate shall enable users to generate Risk Management Plans. | RG-001 | Fail – Not offered |
| F-4.1.6 | The Innsolate™ shall enable users to generate Test and Evaluation Master Plans (TEMP). | RG-001 | Fail – Not offered |
| F-4.1.7 | The Innsolate™ shall enable users to generate Implementation Plans. | RG-001 | Fail – Not offered |
| F-4.2 | The Innsolate™ shall store exported reports in Microsoft Word in the tool. | RG-005 | Pass |
| **F-5** | **Collaboration** |  |  |
| F-5.1 | The Innoslate™ shall enable users to share projects. | DB-010 | Pass |
| F-5.1.1 | The Innsolate™ shall allow users to search for other Innoslate™ users/project members. | DB-010 | Fail – Currently have to know members emails addresses and manually enter |
| F-5.1.2 | The Innoslate™ shall allow users to assign user roles to projects, i.e., Read Only, Read/Write, Owner. | DB-010 | Pass |
| F-5.1.2.1 | Users with Owner permissions shall have full control of the shared project. | DB-010 | Pass |
| F-5.1.2.2 | Users with Read/Write permissions shall have the ability to add, modify, and delete project content. | DB-010 | Pass |
| F-5.1.2 | The Innsolate shall notify Read/Write users of project changes (e.g., modifications, deletions, additions). | DB-010 | Fail – Feature currently not available |
| F-5.1.3 | The Innoslate™ shall notify the Owner users of project changes (e.g., modifications, deletions, additions). | DB-010 | Fail – Feature currently not available |
| F-5.1.4 | Users with Read Only permissions shall only read content. | DB-010 | Pass |

# Appendix C: Requirements Traceability – Non-Functional

|  |  |  |  |
| --- | --- | --- | --- |
| **UID** | **Requirements** | **Test Case** | **Pass/Fail** |
| **NF-1** | **Accessibility** |  |  |
| NF-1.1 | The Innoslate™ shall be accessible on multiple browser platforms. | N/A | Fail – Not accessible to IE 8 and older users |
| NF-1.2 | The Innoslate™ shall be accessible on Google Chrome. | N/A | Pass |
| NF-1.3 | The Innoslate™ shall be accessible on Mozilla Firefox. | N/A | Pass |
| NF-1.4 | The Innoslate™ shall be accessible on Internet Explorer version 8 and newer. | N/A | Fail – Only accessible to IE 9 users. |
| **NF-2** | **Response Time** |  |  |
| NF-2.1 | The Innoslate™ shall enable users to add and remove large batches of entities into/from the database. Large bacth is defined as at least 100 entities. | DB-008 | Fail |
| NF-2.2 | The Innoslate™ shall enable users to import and export documents in less than 30 seconds. | DB-009, DB-010, DA-001, RG-005 | Pass |
| NF-2.3 | The Innolsate™ shall generate report templates and documents in less than 30 seconds. | RG-001, RG-004, RG-002 | Pass |
| **NF-3** | **Robustness** |  |  |
| NF-3.1 | In cases where inputs are invalid, the Innolsate™ shall not terminate, but display an error message describing the problem and troubleshooting solution. | DB-001, DB-004, DB-008, DB-009, DB-010, RQ-001, RQ-004, RQ-005, DA-001, DA-002, DA-003, RG-001, RG-002, RG-003 | Fail |
| **NF-4** | **Usability** |  |  |
| NF-4.1 | Users of Innoslate™ shall require minimal training or experience in using systems engineering management tools to operate the Innoslate™. | N/A | Pass |
| NF-4.2 | Users shall be able to complete the following systems engineering activities within the Innoslate™: Project Management, Requirements Elicitation and Management, Concept Development, System Design and Architecture. | RQ-001, RQ-002, RQ-003, RQ-004, RQ-005, RG-001, RG-002, RG-003, RG-004 | Pass – Only activities that cannot be completed is PM activities |

# Appendix D: Test Case Results – Database Viewer

| **Detailed Procedure Steps** | | | | |
| --- | --- | --- | --- | --- |
| **Step #** | **Step Description** | **Expected Results/Method of Recording** | **Actual Results (Firefox)** | **Actual Results (Chrome)** |
| DB-001 | Creating, Saving, and Viewing Class Entities:  .Sign in into Innoslate  .Click on the “Create Entity” icon  .Select an entity to create (example, “Action”)  .Fill in the Name, Number, and Description  .Click on the “Save and Close” icon  .Follow the above steps to create all of the other available entities | . After doing a “Save and Close” for the last entity that you have created, make sure that all of the created entities are visible from the Database View.  .Namely, you should see the following class entities in the Database View: Action, Artifact, Asset, Characteristic, Cost, Decision, Input/Output, Link, Measure, Physical, Requirement, Resource, Risk, Software Interface, Statement, and Time | Was able to successfully create, save, and view all of the aforementioned class entities | I entered many artifact and statement entities associated with the MEEPAS project Concept of Operations.  I entered many artifact entities associated with the MEEPAS to-be architecture diagrams (OV-2, OV-5, OV-6).  I was able successfully create, save and view these entities within the Database view. |
| DB-002 | Searching the Database  .From the Database view, locate the Search textbox (Top-Right corner of screen)  .Type in the Full-Name of one of the entities you created during step DB-001  .Click on the Search Icon (the Blue Square icon with magnifying lens)  .Do the above step by only typing-in part of a name of an entity (example, if the Full Name is “My First Action”, type in only one of the words: “Action” or “My” or “First” or a combination of these | Should be able to search entities that are contained in the Database using both the Full-Name as we Part of the Name of the entity | Was able to search the existing entities using both their Full Names as well as Parts of their Names | I was able to search on entities created as part of step DB-001 and successfully returned the expected entity results. |
| DB-003 | Sorting Search Results from Database  .From the Database View, Click on each of the Attribute Columns (Number, Name, Class, Modified) | The search results are expected to be sorted by each respective attribute type that has been clicked for sorting (Number, Name, Class, Modified) | Was able to sort the search results using Number and Modified attributes, but can only sort in one direction. For example, cannot sort “Earliest to Latest” on the Modified field (Can only sort “Latest to Earliest”).  Cannot sort at all using the other 2 attributes (Name and Class) | I sorted the Database View list using the column headings number, name, class and modified. The order of entities was displayed as expected in ascending and descending order. |
| DB-004 | Filtering the Database View  .From the Database view, click on the “Filter Classes” command (Top-Left area of view)  .From the drop-down list, click on each entity types that appears | For each of the entity-type that has been clicked, all of the created entities for that specific type need to be returned by the database. For example, if you happen to have 2 Action entities previously created, both of these Action entities should be returned when selecting an Action entity from the drop-down list | Was able to filter the Database View for selected entities, as expected | I filtered the Database view by selecting the relevant entity types of artifact, statement, and requirement. The resulting list of entities was displayed based on the filter criteria as expected. |
| DB-005 | Scrolling the Database View’s Window  .Make sure that you have more than 30 entities created in your test project  .Scroll up and down the Database window a few times | While scrolling down the Database window, once you reach the 30th entity, the remaining entities (the 31st and up) need to be added to the window’s view for viewing | Was able to view the 31st and over entities while browsing the DB window | Scrolling the database view interface when more than 30 entities exist/have been saved returns additional rows until the end of the list is reached. It would appear this limit is 50 rows at a time. |
| DB-006 | Deleting Entities from Database View  .From the Database view, click on the square-box next to some of the entities  .Click the Delete command (Red Label on top of page menu)  .When asked “Are you sure you wish to delete?”, click on Confirm | The selected entities are expected to be deleted from the Database View | Was able to delete a set of entities from the Database view | Selected one and many entities to be deleted and selected the delete button. Received a confirmation to continue with removing the data values permanently and clicked yes to confirm. The entities were removed from the Database View. One thing to note is that when trying to delete more than 100 entities in a batch task, the tool hung and did not complete successfully. |
| DB-007 | Deleting Entities from Inside the Entities view  .From the Database View, click on an entity (this takes you to the entity’s view)  .Click the Delete command (Red Label on top of page menu  .When asked “Are you sure you wish to delete?”, click on Confirm | The selected entities are expected to be deleted from the Database View | Was able to delete entities from the entities’ views with no issues | Selected a particular entity to access the Edit entity view and clicked the Delete button and successfully deleted the entity. |
| DB-008 | Importing Documents into the Tool  .From the Database View, click on the down-arrow next to the Import command (top of page)  .Click on “File” from the drop-down list  .Using the “Upload File” command, upload each file type (Xml, MPP, and KML) from your local machine into the tool | Expected to be able to import all 3 file types- Xml, MPP, and KML- successfully into the tool | There was an error importing an XML file, a file that has been created by the Innoslate tool itself. Error Message says - “From Database: no relationships” | Tested the entity file upload function via the entity edit screen and was able to successfully upload file attachments and download them to verify.  The import document function requires existing sample data for which was not tested in this test case. |
| DB-009 | Exporting Documents from the Tool  -From Database View, click on the Export command (top of page)  -Export by doing a filter-by Classes, Labels, and None (no filter applied) | Expected to export data from the tool, filtering by Classes as well as Labels. | Was able to successfully export, filtering by Classes |  |
| DB-010 | Project Sharing  .Click the “Share” command (top right of screen)  .On project share window, enter the email address of user that you would like to share with and set their editing rights for the project (Owner, Read, Write)  .Click “Share”. The person’s Name and Email Address should pop-up under the list of users. Click “Done”  .When the person comes online; you should see a notification under the “Online Users” section (bottom-right corner).  .Click on “Online Users”. You should see the person in the list, and if he/she is online & available to chat.  .Click on the person’s name. A separate chat window should open, which allows you to chat with the person | Expected to allow chatting in real-time with an online user, while both parties are able to view the shared project | PENDING |  |

# Appendix E: Test Case Results – Requirements Viewer

| **Detailed Procedure Steps** | | | | |
| --- | --- | --- | --- | --- |
| **Step #** | **Step Description** | **Expected Results/Method of Recording** | **Actual Results (Firefox)** | **Actual Results (Chrome)** |
| RQ-001 | Creating, Saving, and Viewing Requirement Entities:  *NOTE: In Innoslate, Requirements can be created in two ways – either from Database view by clicking ‘Create Entity’ or from ‘Requirements’ view by clicking ‘Create Requirement’.*  .Sign in into Innoslate  .Click on the “Create Entity” icon  .Select “Requirement” from the popup  .Fill in the Name, Number, and Description  .Click on the “Save and Close” icon  .Follow the above steps to create all of the requirements for MEEPAS project | . After doing a “Save and Close” for the last requirement that you have created, make sure that all of the created requirements are visible from the Database View and Requirements view. | Was able to successfully create, save, and view all requirement entities | The first requirement was saved as expected and displayed in the Requirements Main View.  There was one instance where data was not saved and had to be re-entered. |
| RQ-002 | Establishing Relationship with an Action:  .From the Database view, click on the “Create Entity” icon  .Select “Action” from the popup  .Fill in the Name, Number, and Description  .On the right side click ‘Add’ on ‘traced from Statement’ row  .From the popup select the requirement that this Action is satisfying  .Click on the “Save and Close” icon | . After completing the Add operation you should be able to see the Requirement on the right side. You should be able to switch to the Requirement from Action screen and vice versa. | Was able to trace from a requirement to an Action and vice versa | Performed requirements validation, the process took approximately 30 seconds the first run through and then less than 5 seconds on subsequent runs; Validate requirements scoring is inconsistent, as two requirements that were almost identical were given very different scores. |
| RQ-003 | Editing Requirements:  .From the Requirements view, Click ‘Edit’ on the row of the requirement that you want to edit  .Modify Name, Number, and/or Description  .Click ‘Save and Close’. | The requirements view should reflect the change that you applied | The Requirements view was updated to reflect the change applied on the Edit screen | Edit requirements saved data as expected. The data was stored and reflected in both the requirements view and the requirement entry screen. |
| RQ-004 | Obtaining Quality Scores for Requirements:  1.From the Requirements view, click on the “Validate Requirements”  2.From Requirements view select the requirement that you want to view/edit the quality score for and manually edit one or more of the nine quality score parameters (Clear, Complete, Consistent, Correct, etc.) | 1. Innoslate should calculate the quality score for each requirement based on a nine point quality check  2. Innoslate should automatically recalculate the quality score for the requirement and display it | Innoslate did the Quality Score computation and displayed the result |  |
| RQ-005 | Create a requirement hierarchy:  .Create a top level requirement by following steps in RG-001  .Using the same steps create a few child level requirements (decomposing the top level requirement)  .Edit the top level requirement and select ‘Add’ on ‘decomposed by Children’ at the right side  .Select the child requirements | The child level requirements should added at the right side under the section ‘decomposed by Children’  Clicking ‘View’ on any of the child requirements should switch to the child requirement and should show the parent requirement under ‘decomposes Parent’ section | The child requirements were shown under ‘decomposed by Children’ section on parent screen and on the child screen, the parent requirement was shown under ‘decomposes Parent’ section |  |

# Appendix F: Test Case Results – Document Analyzer

| **Detailed Procedure Steps** | | | | |
| --- | --- | --- | --- | --- |
| **Step #** | **Step Description** | **Expected Results/Method of Recording** | **Actual Results (Firefox)** | **Actual Results (Chrome)** |
| DA-001 | **Uploading Documents using the Automated parser**  .Make sure you have a DOCX and a PDF file on your local computer ready to upload  .From Database View, click on the drop-down arrow next to the “Import” menu  .Choose “Analyzer”  .Click on “Upload Document” (Right Panel)  .Select the document to upload(either the PDF or the DOCX file)  .When prompted “Would you like to Auto parse this Document using Analyzer?”, choose a Starting Number (example 1.0) and click YES  .Once uploading completes, list of the created entities shows. Click on “Save Entities”  .Follow the same steps to upload the 2nd file sample (either the PDF or the DOCX file) | The uploaded documents are expected to be broken-down into separate paragraphs, with Names and Parent-Child relationships created based on the documents’ Titles and Numbering Schemes | For large DOCX as well as PDF samples (example, a 600 KB docx file with 24 pages, or a 2000KB PDF file with 300 pages), the upload progress bar runs indefinitely, not completing even after 30 minutes. If user goes back to the Database view while the upload bar is still pending, it turns out that the document has in fact been uploaded, which indicates that the issue is with the Upload Progress bar’s design. | Overall: With minimal use of the User Guide, the tool is fairly intuitive about how to import a document. The “Import” button was easy to find on the DB view. Once in the import portion of the tool, it’s easy to upload the document. It is unclear as to what class a user should select for their document. Other fields used to label the imported document seem straightforward.  Tool seems to get hung when saving uploaded document.  PDF: Imported the MEEPAS Project Report using a PDF and labeled it under the class “Statement”. Once uploaded, the analyzer did not import the entire document. It only imported three parts of the document and seems to parse those parts randomly.  Innoslate does store the PDF and when I download the document from Innoslate, the PDF opened in a browser and appears to have maintained the integrity of the document.  Word:  I’ve loaded a word document into the tool and tool seems to parse the document into statement entities, but I can’t analyze the result of the parsing because the tool gives me quota error and gets hung up in trying to save and analyze any requirements the tool identified.  Innoslate downloads the Word version downloads and appears to have maintained the integrity of the document. |
| DA-002 | **Uploading Documents using the Manual Parser**  .Follow the same steps as Step# DA-001 to upload both the DOCX and PDF sample, but when asked to auto parse using the Analyzer, choose NO  .Once uploading completes, you should see blank entity fields on left panel, and the uploaded document on right panel. Selecting a Text from the document (right panel) causes a hanger to appear, showing Name, Number, and Description entities. Clicking any of these entities will automatically insert them into their corresponding blank entity fields at left panel.  .Fill in the blank entity fields (Name, Number, and Description) using the steps described above  .Click on “Save and Reset”  . Follow the same steps to upload the 2nd file sample (either the PDF or the DOCX file | Expected to allow manually parsing the uploaded files | Same issue as above (DA-001) when uploading large-sized files (both DOCX and PDF files). In addition, when uploading PDF files, the uploaded document is not visible at the right panel (only the blank entities are shown on the left panel). This 2nd issue is okay when uploading DOCX files (only happens for PDF samples) | Word: I’ve loaded a word document into the tool and the manual extractor extracts all of the MEEPAS document. In the extractor mode, I am able to highlight section titles, paragraphs, and number to create entities that can be saved into the DB. This is a little heavy lifting, but it is allowing me to insert the sections of my document into the tool.  To reopen the document in extractor mode, it’s not intuitive by the user how to do this and the user guide doesn’t provide guidance on how to get back to this to add more entities to the DB. After playing with the tool, I found the “Analyze” button, which allowed me to enter the extractor mode and begin adding more entities.  The Word version downloads and appears to have maintained the integrity of the document. |

# Appendix G: Test Case Results – Reports Generator

| **Detailed Procedure Steps** | | | | |
| --- | --- | --- | --- | --- |
| **Step #** | **Step Description** | **Expected Results/Method of Recording** | **Actual Results (Firefox)** | **Actual Results (Chrome)** |
| RG-001 | **Class-Based Reports:**  1. In Database View click ‘Reports’.  2.From the ‘Generate Report’ dialog select ‘Class Based Report’  3.From the ‘Class Based Report’ list, select ‘Requirement Reports  4.Pick ‘Requirement Class Summary Report’ from the dropdown  5.Click ‘Submit’  6. Repeat steps 1 to 5 selecting ‘Requirements Class Entity Report’ and ‘Requirement Class Selected Entity Report’ at step 4. | Must be able to generate, download, and view reports on individual and all class entities | Was able to successfully generate, download, and view reports | Able to initialize a CONOP document using the wizard process; however did get stuck in an endless loop of confirmation prompts. The core set of CONOPS entities were created.  Note: This isn’t an endless loop. It seems the tool brings the user back to the prompt screen, so they can select the next task, i.e., add CONOP sections, adding tables to the stakeholder section, etc. Ran into an error when trying to develop the stakeholder table. Doesn’t seem that anything gets saved in the database, so I am not sure what the result of adding a stakeholder is. Overall, I can’t add anything to the CONOPs framework using the Wizard without receiving an error. I can’t seem to find anything that I’ve added in the database view, so I can’t see how the tool is handling my input. |
| RG-002 | **General Reports:**  1. In Database View click ‘Reports’.  2.From the ‘Generate Report’ dialog select ‘General Report’  3.From the ‘General Report’ list, select ‘CONOPS’  *NOTE: If Concept of Operations has not been defined yet, Innoslate will invoke the CONOPS Data Entry Wizard. The report generation will continue after you finish the CONOPS data entry*  4. Select Generate CONOPS from the dropdown  5. Click ‘Submit’ | Must be able to generate, download, and view general reports | Was able to generate, download, and view CONOPS report | Was unable to generate a CONOPS report. It was not offered as an option in the version being used. |
| RG-003 | **DODAF Reports:**  1. In Database View click ‘Reports’.  2.From the ‘Generate Report’ dialog select ‘DoDAF Reports’  3. Select ‘DoDAF OV-1’  4. Click Submit | Must be able to generate, download, and view DoDAF reports | Received confirmation dialog saying the report is complete. But could not find the report under ‘Database’ view | Attempted several times to create an OV-1 and OV-5 report based on existing entities, however nothing was generated. |
| RG-004 | **JCIDS Reports:**  1. In Database View click ‘Reports’.  2.From the ‘Generate Report’ dialog select ‘JCIDS Reports’  3. Select ‘JCIDS CDD Report’  4. Click ‘Submit’  *Note: If JCIDS Capability Development Document has not been created yet, Innoslate will launch the JCIDS CDD creation Wizard. Report generation will continue after the CDD is created.* | Must be able to generate, download, and view JCIDS reports | Was able to generate, download, and view JCIDS CDD Report | Could not generate JCIDS reports |
| RG-005 | **View Reports:**  1. Go to ‘Database’ view  2. Scroll to the bottom  3. Select one of the reports just generated  4. Click ‘Download’  5. Verify that the report contains expected information | Should be able to open the Report entity and view in the appropriate format | Was able to download and view the report |  |

1. Adapted from the GMU SEOR website specific to graduate curriculum - http://seor.gmu.edu/courses/syst-grad-courses.html [↑](#footnote-ref-2)
2. Innoslate User Guide – www.Innoslate.com [↑](#footnote-ref-3)