**System Requirements Specification**

**for**

MBSE Avionics System Capstone

**Version 0.1.3 approved**

**Prepared by**Walter Hernandez  
Shawn Merrick Miller  
Luke Newcomb  
William Pate

**Embry-Riddle Aeronautical University.**

**September 26, 2023**

**Table of Contents**

**Table of Contents ii**

**Revision History ii**

**1. Introduction 1**

1.1 Purpose 1

1.2 Document Conventions 1

1.3 Intended Audience and Reading Suggestions 1

1.4 Product Scope 1

1.5 References 1

**2. Overall Description 2**

2.1 Product Perspective 2

2.2 Product Functions 2

2.3 User Classes and Characteristics 2

2.4 Operating Environment 2

2.5 Design and Implementation Constraints 2

2.6 User Documentation 2

2.7 Assumptions and Dependencies 3

**3. External Interface Requirements 3**

3.1 User Interfaces 3

3.2 Hardware Interfaces 3

3.3 Software Interfaces 3

3.4 Communications Interfaces 3

**4. System Features 4**

4.1 System Feature 1 4

4.2 System Feature 2 (and so on) 4

**5. Other Nonfunctional Requirements 4**

5.1 Performance Requirements 4

5.2 Safety Requirements 5

5.3 Security Requirements 5

5.4 Software Quality Attributes 5

5.5 Business Rules 5

**6. Other Requirements 5**

**Appendix A: Glossary 5**

**Appendix B: Analysis Models 5**

**Appendix C: To Be Determined List 6**

**Revision History**

| **Name** | **Date** | **Reason For Changes** | **Version** |
| --- | --- | --- | --- |
| Shawn M | 9/26/23 | Initialize Document; Section 1.1,1.4 | 0.1.0 |
| William P | 9/26/23 | Initialize Document; Section 1.2,1.3,1.5 | 0.1.0 |
| Luke N | 9/26/23 | Section 2.2, 2.4 | 0.1.1 |
| Shawn M | 9/29/23 | Section 1, 2, 3, 4, 5, 6, Appendix A | 0.1.2 |
| Luke N | 9/29/23 | Appendix C | 0.1.3 |

# **Introduction**

## **Purpose**

The product requirements presented in this document apply to the Model-Based System Engineering (MBSE) Avionics Senior Capstone (MASC) model. The requirements herein specified are applicable at the document’s date, and are superseded by any following revisions. This SRS applies to the complete System of Interest (SOI) model and does not cover details of the SOI. The SOI covered by MASC is a generic exploration of a spacecraft’s avionics system.

## **Document Conventions**

Within our Software Requirement Specification (SRS) document, text that is derived from our template document is italicized and denoted with “<>” on either side, it will be removed in later revisions. Each section of our document follows a numbering system with the left-most number denoting the document section, the second number denoting the chapter within the section, and the third number notating specific and distinct labels within a chapter.

## **Intended Audience and Reading Suggestions**

This document is intended to be read by developers, project managers, or educators who are new or familiar to the MBSE process The contents of this document provide a reference for the design and requirement specification process for a System of Systems (SoS) and generic spacecraft avionic systems or in the general modeling use case. Those who are looking for the full overview of our project as it pertains to the senior design capstone class will want to read sections 1 and 2 before viewing the rest of the document. Alternatively, readers who are solely interested in seeing the application of MBSE onto the developed system may only read sections 4 and 5.

## **Product Scope**

The MASC product should be a generic system model that captures hardware and software configurations derived from high-level system requirements, which can be used to analyze system behavior and identify failure modes to expand Failure Mode Effects Analysis (FMEA) capability. This product would improve failure mode identification in the system design process, with the objective of demonstrating the solution agnostic to the SOI, and demonstrating a path forward for this capability. Using a system model in this way highly aligns with industry needs and can apply to failure analysis in any complicated system. For more information, please refer to the Product Vision Statement.

## **References**

No current references, but more will be added following completion of the literature review.

# **Overall Description**

## **Product Perspective**

The product specified in this document is a new, self-contained model created with the Dassault, Magic System of Systems tool. This model captures the interaction of multiple systems in the context of a spacecraft avionics system of systems. The diagram below captures a simplified representation of the systems.

<Product perspective figure to be inserted when licenses acquired>

## **Product Functions**

The product detailed in this document must:

* Function as a minimal model of a spacecraft avionics system
* Be generic enough to provide value to any specific spacecraft which specializes the model
* Allow failure analysis to be performed on the base model

<Product function figure to be inserted when licenses acquired>

## **User Classes and Characteristics**

*Product Owner*

The product owner receives the completed model and guides the implementation of the model and requirements for the SOI. The product owner will demonstrate the technical expertise on the SOI and satisfy input data requirements for the engineering team.

*Responsible Engineer*

The responsible engineer owns the accuracy of the model and its implementation. The responsible engineer assumes ownership of the highest level model requirements and makes decisions on the best implementation practices for the model product.

*System Engineers*

The system engineers are responsible for building the model to satisfy the requirements outlined in this document. Furthermore, the system engineers are responsible for using the model of the SOI to generate data for other stakeholders.

*Discipline Engineers*

The discipline engineers are at the boundary of the product’s functionality, and provide input and requirements for the accuracy of the data contained in its construction. Furthermore, the systems engineers use products generated by the model and will impose requirements for data aggregation.

## **Operating Environment**

The environment in which the product will operate is within the Magic System of Systems Architect 2022x software. The software will be running on Windows laptops which contain greater than or equal to 8 gigabytes of RAM, and have at least 50 gigabytes of free disk space available.

## **Design and Implementation Constraints**

The largest constraint facing the product is the licensing for the model software. The Magic System of Systems Architect requires a license to interact with the model in a meaningful way. Furthermore, the spacecraft avionics system model will be generated on an academic license, and therefore will only be able to use open source information and cannot share the product with any commercial entity. Lastly, the product will likely only be forward compatible with future versions of the modeling software, and may not be supported by most teams that aren’t using the 2022x version.

## **User Documentation**

Currently, there are no additional documentation components other than the System Design Document.

## **Assumptions and Dependencies**

The accuracy of the system model is highly dependent on the design decisions made for the SOI. The results from this product model will vary based on the assumptions and decisions made in constructing a generic avionics system. Niche or solution specific dependencies will not be captured by this model. The model requirements will help construct a logical representation of the SOI and will be bound by the logical assumptions made. Lastly, the system design process is highly dependent on the MagicGrid framework, and the modeling requirements will use the processes applicable to the SOI.

# **External Interface Requirements**

## **User Interfaces**

The user shall only interface with the model through the Magic System of Systems tool and its software user interface.

## **Hardware Interfaces**

The product currently maintains no hardware interface requirements.

## **Software Interfaces**

This product uses the 2022x Magic Model Analyst Plugin from Dassault Systems Inc., which is a plugin available from the software company.

## **Communications Interfaces**

The product currently maintains no communication interface requirements.

# **System Features**

*<This template illustrates organizing the functional requirements for the product by system features, the major services provided by the product. You may prefer to organize this section by use case, mode of operation, user class, object class, functional hierarchy, or combinations of these, whatever makes the most logical sense for your product.>*

## The Spacecraft Avionics Model

4.1.1 *Description and Priority*

The system of interest for the modeling product is a generic exploration spacecraft avionics system model.

4.1.2 *Stimulus/Response Sequences*

The model will be invoked through manipulation and query with the Magic System of Systems tool.

4.1.3 *Functional Requirements*

REQ-1.1: The system model shall capture the structure of a generic exploration spacecraft avionics system

REQ-1.2: The system model shall capture the behavior of a generic exploration spacecraft avionics system

## Example **System Feature 1**

*<Don’t really say “System Feature 1.” State the feature name in just a few words.>*

4.1.1 Description and Priority

*<Provide a short description of the feature and indicate whether it is of High, Medium, or Low priority. You could also include specific priority component ratings, such as benefit, penalty, cost, and risk (each rated on a relative scale from a low of 1 to a high of 9).>*

4.1.2 Stimulus/Response Sequences

*<List the sequences of user actions and system responses that stimulate the behavior defined for this feature. These will correspond to the dialog elements associated with use cases.>*

4.1.3 Functional Requirements

*<Itemize the detailed functional requirements associated with this feature. These are the software capabilities that must be present in order for the user to carry out the services provided by the feature, or to execute the use case. Include how the product should respond to anticipated error conditions or invalid inputs. Requirements should be concise, complete, unambiguous, verifiable, and necessary. Use “TBD” as a placeholder to indicate when necessary information is not yet available.>*

*<Each requirement should be uniquely identified with a sequence number or a meaningful tag of some kind.>*

REQ-1:

REQ-2:

## **System Feature 2 (and so on)**

# **Other Nonfunctional Requirements**

## **Performance Requirements**

REQ-2.1: The system model shall be detailed to the <TBD 1> order of functional detail.

REQ-2.2: The system model shall demonstrate <TBD 2> behavioral interaction between multiple systems.

## **Safety Requirements**

The product currently maintains no safety requirements.

## **Security Requirements**

REQ-3.1: The product shall not violate the Magic System of Systems academic licensing agreement.

## **Software Quality Attributes**

REQ-4.1: The product shall be adaptable to support changes in the SOI scope.

REQ-4.2: The model shall be reconfigurable given new or updated design information.

## **Business Rules**

The product currently maintains no business requirements.

# **Other Requirements**

REQ-4.1: The work completed using the MagicDraw tool for the senior capstone class was done under an academic license provided to Embry-Riddle Aeronautical University by Dassault Systems Incorporated. Work completed under this license is not to be used for any commercial or personal purposes, but only in accordance with the university's policies and <TBD 3>.

**Appendix A: Glossary**

MBSE - Model-Based Systems Engineering

FMEA - Failure Mode and Effects Analysis

SOI - System of Interest

RAM - Random Access Memory

**Appendix B: Analysis Models**

N/A

**Appendix C: To Be Determined List**

*<Collect a numbered list of the TBD (to be determined) references that remain in the SRS so they can be tracked to closure.>*

TBD 1: The number of orders of functional detail to include in the generic model. Determined by learning more about MagicDraw/MagicGrid.

TBD 2: The number of behavioral interactions between multiple systems in the generic model. Determined by learning more about MagicDraw/MagicGrid.

TBD 3: Dassault Systems’ or other applicable regulations on the team’s licenses.