System Requirements Specification

for

MBSE Avionics System Capstone

Version 0.6.1

Prepared by
Walter Hernandez
Shawn Merrick Miller
Luke Newcomb
William Pate

Embry-Riddle Aeronautical University

March 2, 2024

Table of Contents

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
Walter H	10/18/23	Section 4	0.1.4
Luke N	10/18/23	Section 4	0.1.5
Luke N	10/31/23	Section 2.1, Section 4	0.2.0
Shawn M	10/31/23	Section 1.4, Section 4	0.2.1
William P	10/31/23	Section 2.2, 4, reviewing previous sections	0.2.2
Walter H	10/31/23	Section 5	0.3.0
William P	11/5/23	Reviewing Feedback and setting up V3 material	0.3.1
Walter H	11/9/23	Section 4.1, 4.2	0.3.2
Shawn M	11/14/23	Redo section 2.2,2.4,2.5,5	0.3.3
William P	11/20/23	Finalize document for end of semester	0.3.4
Luke N	11/21/23	Review changes and edit formatting	0.4.0
Walter H	11/21/23	Review changes and edit formatting	0.4.0
Luke N	2/3/24	Add changes to sections 2 and 4	0.5.0
Walter H	2/3/24	Review changes	0.5.1
William P	2/4/24	Section 5	0.5.2
Shawn M	2/4/24	Changes to section 2	0.5.3
Luke N	3/1/24	Added an introduction and reorganized the information in Section 1	0.6.0
Luke N	3/2/24	Filled in the references section	0.6.1
Luke N	3/2/24	Replaced figure 1 with the updated version	0.6.1
Luke N	3/2/24	Rewrote some vague nonfunctional requirements to be more specific functional requirements	0.6.1
Luke N	3/2/24	Added new nonfunctional requirements	0.6.1
Luke N	3/2/24	Added descriptions for terms in the glossary	0.6.1

1. Introduction

This project is designed to explore and evaluate the effectiveness of the MagicGrid approach to Model-Based Systems Engineering (MBSE). This approach uses the Systems Modeling Language (SysML) to provide an approach to modeling systems. The goal of this project is to model a system using MagicGrid and use that model to perform Failure Mode and Effects Analysis (FMEA) on the resultant model in order to determine the effectiveness of modeling. The MBSE tool that is being used for this project is the Magic System of Systems Architect by Dassault Systemes.

1.1 Purpose

The purpose of this document is to define all requirements relating to the MBSE Avionics Senior Capstone (MASC) project. This document covers the process used to create the model which is derived from following the MagicGrid process. The requirements listed in this document define what the model should look like according to this process. The requirements are not for the System of Interest (SOI) being modeled. The requirements specified in this document are applicable at the document's date, and are superseded by any following revisions.

1.2 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 System of Systems (SoS) and generic spacecraft avionics systems or in the general modeling use case. This document also details requirements which can be used to verify the process of FMEA item generation. 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 3 through 6.
Sections 1 and 2 describe what the project is and the general methodology of the project.
Sections 3 through 6 contain the detailed requirements which are used to verify if the process is being followed correctly.

1.3 Product Scope

The MASC product will include an evaluation of system modeling using the MagicGrid framework to capture the problem domain and failure modes for a notional Crewed Cislunar Station (CCS). The focus of this model is on the avionics subsystem and generating safety and reliability requirements through FMEA. This project aims to improve understanding of 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. The CCS being modeled by MASC is a spacecraft avionics system which has the mission of orbiting the moon and performing both crewed and uncrewed missions for long durations. This project is meant to serve as a ground-up modeling of a system that closely mirrors the goals of the NASA Gateway system. This system to model was chosen for its complexity and the team's lack of subject knowledge in the area to facilitate a more exploratory approach.

1.4 References

[1] MagicGrid Book of Knowledge, 2nd Edition, from
https://discover.3ds.com/magicgrid-book-of-knowledge
[2] Describing FMEA Items Guide, from
https://docs.nomagic.com/display/CSRA190/Describing+FMEA+Items

2. Overall Description

2.1 Product Perspective

The product specified in this document is the complete process followed to create a new, self-contained model of the CCS. The model of the avionics system will be defined in relation to the other subsystems of the CCS, and is meant to be developed as an evaluation of the MagicGrid process. Figure 1 captures the CCS system context. This diagram captures the interaction of multiple external systems in relation to the spacecraft avionics system. This diagram exemplifies the product itself because it was generated as part of the MagicGrid process to model the system.

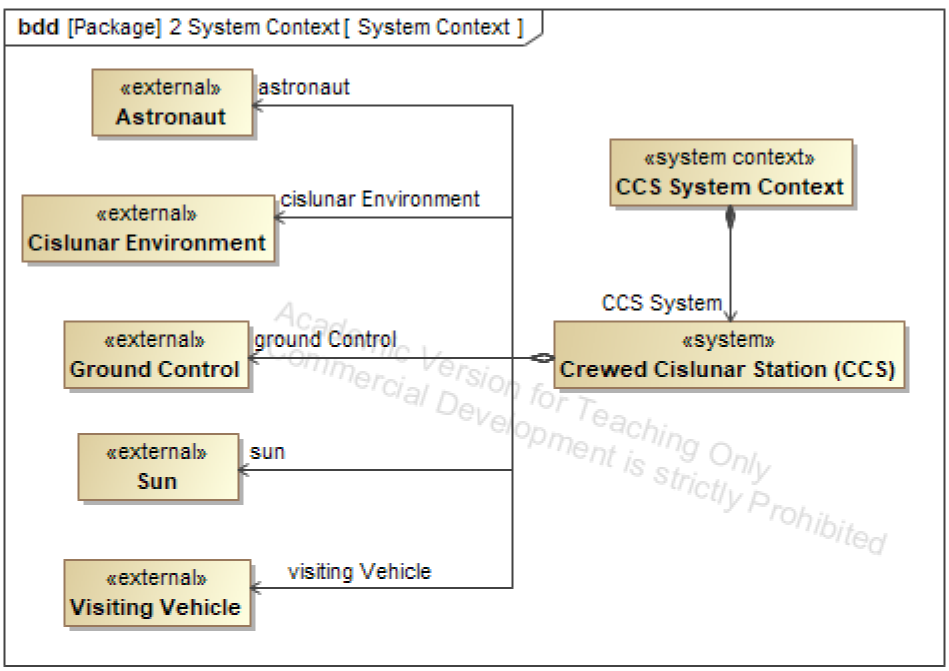

Figure 1: CCS System Context Model

2.2 Product Functions

- Functions of the methodology:
  - Follow the Zachman style matrix as defined within the MagicGrid framework.
  - Allow for failure modes effect analysis to be performed on a base model.
- Functions of our created system:
  - Function as a minimal model of a spacecraft avionics system within the context of a CCS.

- Deliver a problem domain analysis of the system.
- Be generic enough to provide value to any specified CCS to be modeled within the tool.

2.3 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 and the sub-systems contained in its construction. Furthermore, the systems engineers use products generated by the model and will impose requirements for data aggregation.

2.4 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 10 laptops which contain greater than or equal to 8 gigabytes of RAM, and have at least 50 gigabytes of free disk space available.

2.5 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.

2.6 User Documentation

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

2.7 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.

3. External Interface Requirements

3.1 User Interfaces

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

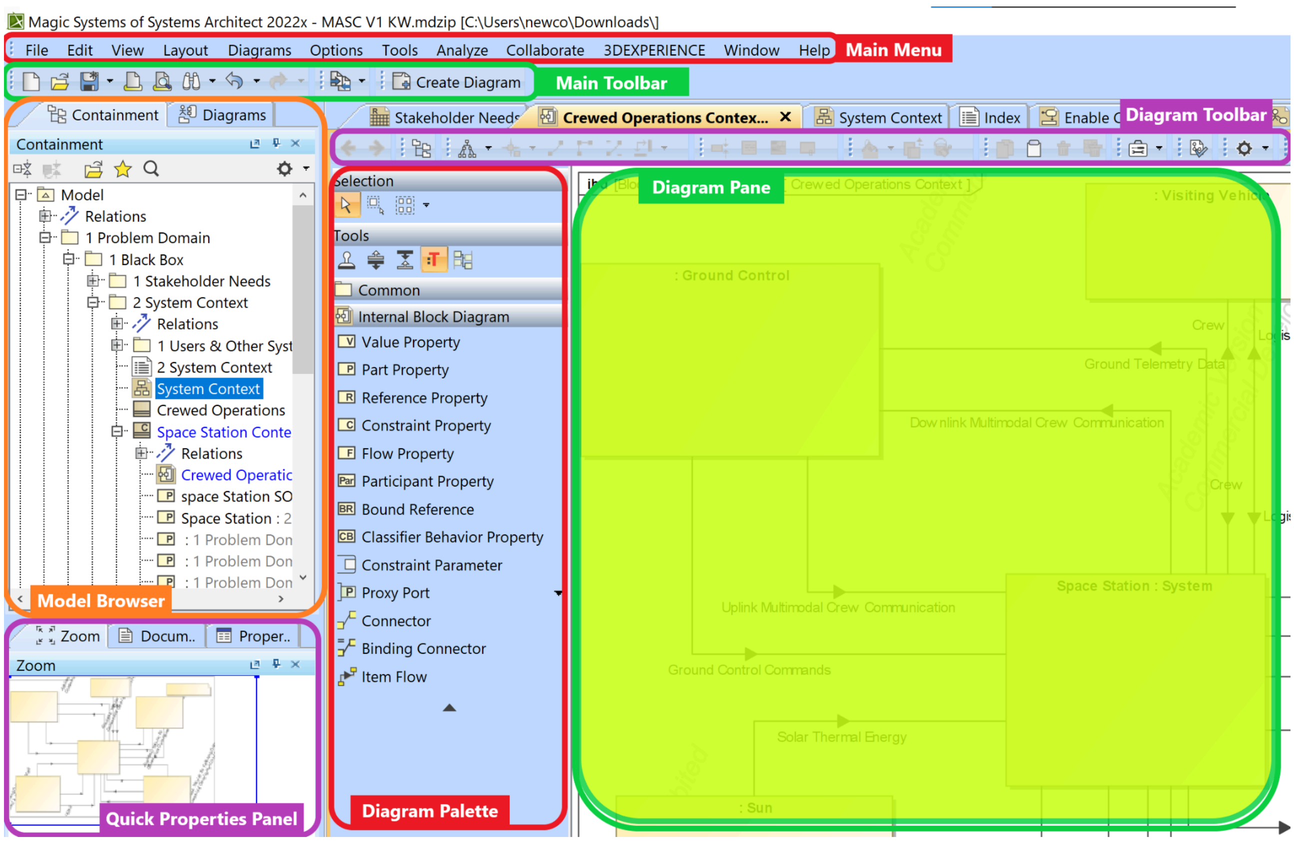

Figure 2: Magic System of Systems Architect User Interface

- **Main Menu:** The main menu located at the top of the modeling tool window and contains main functions for file management.
- **Main Toolbar:** Contains commands for basic project-related tasks and are located at the top of the modeling tool window, right under the main menu.

- **Diagram Toolbar:** Contains commands for working with diagram symbols and are located at the top of the diagram pane.
- **Model Browser:** Provides a visual representation of the hierarchy of your model elements.
- **Diagram Pane:** Provides display function of diagram or file being modified.
- **Diagram Palette:** Contains modeling elements applicable to the diagram or file being modified.
- **Quick Properties Panel:** Contains High Level overview of the diagram pane for navigation functions.

3.2 Hardware Interfaces

The product currently maintains no hardware interface requirements.

3.3 Software Interfaces

This product uses the 2022x Magic Model Analyst Plugin from Dassault Systems Inc.

3.4 Communications Interfaces

The product currently maintains no communication interface requirements.

4. Product Features

Our product is the evaluation of the effectiveness of the MagicGrid framework process as applied to the modeling of our SoI. The SoI model is a byproduct of the process which we are following. The following requirements define our process and measure the adherence to the MagicGrid framework and the effectiveness at capturing the CCS mission requirements.

4.1 Model Mission Alignment

4.1.1 Description and Priority

The system model reflects the mission of a notional CCS and describes the related context. This is a high priority.

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 model shall facilitate human crewed missions to cislunar space including capabilities that enable surface missions.

REQ-1.2: The model shall provide capabilities to meet scientific requirements for lunar discovery and exploration.

REQ-1.3: The model shall enable, demonstrate, and prove technologies that are enabling for deep space missions.

REQ-1.4: The model shall capture the need for manual control of flight dynamics.

REQ-1.5: The model shall capture the need for automatic maintenance of station orbit.

REQ-1.6: The model shall capture the need for the station to produce, store, and regulate its own power.

REQ-1.7.: The model shall capture the need to keep the crew alive and safe.

REQ-1.8: The model shall capture the need to accommodate extended crew mission durations.

REQ-1.9: The model shall capture the need to allow the crew to perform extra-vehicular activity.

REQ-1.10: The model shall capture the need to allow for visiting vehicles to dock.

REQ-1.11: The model shall capture the need to accept the transferring of crew and cargo.

REQ-1.12: The model shall capture the need to provide communications to the lunar surface.

REQ-1.13: The model shall capture the need to have the ability to support multiple self, commercial, and international partner objectives.

REQ-1.14: The model shall capture the need to accommodate up to and including 4 crew members.

REQ-1.15: The model shall capture the need to enable 30 to 90 days of a single crew mission duration.

REQ-1.16: The model shall capture the need to provide easy access from Earth with current launch vehicles.

REQ-1.17: The model shall capture the need to have continuous communication with Earth.

4.2 System MagicGrid Compliance

4.2.1 Description and Priority

This section describes how various parts of the model should be constructed to adhere to the MagicGrid framework as laid out in [1]. Requirements can refer to a diagram being created as either a Block Definition Diagram (BDD) or an Internal Block Diagram (IBD).

4.2.2 Stimulus/Response Sequences

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

4.2.3 Functional Requirements

REQ-2.1: The system shall capture the stakeholder needs using a table which holds Requirement items.

REQ-2.2: The system shall represent the stakeholder needs at the context level within the System Context model as a BDD.

REQ-2.3: The system shall refine the functional stakeholder needs with the Use Case diagram and Use Case scenarios.

REQ-2.4: The system shall refine the non-functional stakeholder needs in the system-level Measures of Effectiveness (MoEs) diagram.

REQ-2.5: The system shall identify the expected behavior of the main use case with Functional Analysis diagrams.

REQ-2.6: The system shall decompose the system into conceptual subsystems using a BDD with directed composition relationships from the system to the subsystems.

REQ-2.7: The system shall specify subsystem-level MoEs for one or more conceptual subsystems within the model.

REQ-2.8: The system shall capture generated requirements in a table which contains Requirement items.

REQ-2.9: The system shall capture traceability of requirements through the creation of diagrams which use the Trace relationship between system requirements and stakeholder needs.

REQ-2.10: The system shall capture traceability of requirements through the creation of diagrams which use the Derive relationship between system requirements and use cases.

REQ-2.11: The system shall decompose the system into major distinct subsystems and represent them on the conceptual subsystems BDD.

REQ-2.12: The system shall decompose the avionics conceptual subsystem into conceptual components using a BDD with directed composition relationships from the subsystem to the components.

REQ-2.13: The system shall decompose the avionics subsystem into major distinct components and represent them on the avionics conceptual components BDD.

REQ-2.14: The system shall organize Functional Analysis diagrams using Vertical Swimlanes which correspond to each conceptual subsystem involved in the function.

REQ-2.15: The system shall organize Use Case Scenario diagrams using Vertical Swimlanes which correspond to the system and each external system involved in the use case.

REQ-2.16: The system shall organize non top-level Functional Analysis diagrams using Vertical Swimlanes which correspond to each conceptual subsystem and conceptual component involved in the function.

4.3 System FMEA Process Compliance

4.2.1 Description and Priority

This section describes how various parts of the model should be constructed to adhere to the standard process of FMEA item identification and usage using the guide in [2].

4.2.2 Stimulus/Response Sequences

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

4.2.3 Functional Requirements

REQ-3.1: The system shall capture possible failure items by creating unique FMEA Item objects.

REQ-3.2: The system shall classify FMEA Items as either electrical, software, or mechanical failures.

REQ-3.3: The system shall identify which model item (block, subsystem, etc) is identified as the item that fails due to an FMEA Item.

REQ-3.4: The system shall identify the failure mode associated with an FMEA Item by creating Failure Mode objects.

REQ-3.5: The system shall identify the local effect of a failure associated with an FMEA Item by creating Local Effect of Failure objects.

REQ-3.6: The system shall categorize local effects of a failure as effects which impact the system item identified by the model item attached to the FMEA Item.

REQ-3.7: The system shall identify the final effect of a failure associated with an FMEA Item by creating Final Effect of Failure objects.

REQ-3.8: The system shall categorize final effects of a failure as effects on the customer(s) of a system.

REQ-3.9: The system shall identify the cause of failure associated with an FMEA Item by creating Cause of Failure objects.

5. Other Nonfunctional Requirements

Section 5 details the current known non-functional requirements associated with both the system to model and the modeling framework (MagicGrid).

5.1 Model Mission Alignment

5.1.1 *Performance Requirements*

REQ-1.1: The system model shall have captured the System Context, Use Cases, MoE's, Functional Analysis, and Conceptual Subsystems within the problem domain in no more than 8 months.

REQ-1.2: The system model shall have at least 2 external entities identified on the system context diagram.

REQ-1.3: The system model shall be decomposed into at least 2 conceptual subsystems.

REQ-1.4: The system model avionics subsystem shall be decomposed into at least 2 conceptual components.

REQ-1.5: The system model shall contain at least one 1 top-level use case scenario.

REQ-1.6: The system model shall contain at least 1 top-level avionics use case scenario functional analysis diagram.

REQ-1.7: The system model shall contain at least 2 sub-level avionics functional analysis diagrams.

5.1.2 *Safety Requirements*

REQ-2.1: The system model shall support the function of safely deleting and modifying existing models without any disruption of the preexisting model views or specifications.

5.2.3 *Business Requirements*

REQ-2.1: The system model shall follow MagicGrid compliance standards by interlinking components between separate models.

REQ-2.2: The system model shall be verified by the customer as following the MagicGrid framework.

6. Other Requirements

REQ-1.1: 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.

Appendix A: Glossary

CCS - Crewed Cislunar Station. The system that is being modeled by using the process explored in this project. A space station that orbits the moon which can support a crew.
MBSE - Model-Based Systems Engineering. The current alternative to document-based systems engineering.
FMEA - Failure Mode and Effects Analysis. The process of finding and characterizing various ways that the system could fail in order to drive the creation of new safety and reliability requirements for the system.
SOI - System of Interest. The system that is being modeled by MBSE.
BDD - Block Definition Diagram. A diagram which defines how a block interacts with other blocks in an external manner.
IBD - Internal Block Diagram. A diagram which describes processes or components which are internal to a block.
RAM - Random Access Memory

Appendix B: To Be Determined List

N/A