****System Requirements Specifications for the Volt & Pepper System (VPS)****

Sponsor

**The Department of Electrical, Computer, Software & Systems Engineering at Embry-Riddle Aeronautical University**

Released September 18, 2014

**Volt & Pepper Development Team**

**Abstract**: The System Requirements Specifications (SyRS) for the Volt & Pepper System are detailed within this document. These requirements serve as a contract between the customer of the Volt & Pepper System and the Volt & Pepper Development Team. All stakeholders are recognized and the extent of each party’s respective involvement is thoroughly detailed. This document is compliant with the Institute of Electrical and Electronics Engineers (IEEE) Std. 1233-1998 [4] and the IEEE Recommended Practice for Software Requirements Specifications [5].

# Revision History

|  |  |  |
| --- | --- | --- |
| **Date** | **Reason for Change** | **Version** |
| Sep. 5, 2014 | Initial draft of document | 0.0.1 |
| Sep. 7, 2014 | Added requirements and user stories | 0.0.2 |
| Sep. 8, 2014 | Revised requirements, definitions added | 0.0.3 |
| Sep. 9, 2014 | Updated Definitions | 0.0.4 |
| Sep. 11, 2014 | Defined document sections | 0.0.5 |
| Sep. 12, 2014 | Compiled components of SyRS together | 0.0.6 |
| Sep. 14, 2014 | Refined document sections and styles | 0.0.7 |
| Sep. 15, 2014 | Abstract, references added | 0.0.8 |
| Sep. 16, 2014 | Use cases added, numbering adjusted | 0.0.9 |
| Sep. 17, 2014 | All sections editted | 0.0.10 |

**Table of Contents**

Revision History ii

Table of Contents iii

List of Figures v

List of Tables vi

1. Introduction 1

1.1 Purpose 1

1.2 Mission Statement 1

1.3 Scope 1

1.4 Team Roles 1

1.5 Overview 2

2. General Descriptions 2

2.1 Product Perspective 3

2.2 Product Functions 3

2.3 User Characteristics 3

2.4 General Constraints 3

2.5 Assumptions, Dependencies and Risks 4

2.5.1 Dependencies 4

2.5.2 Assumptions 4

2.5.3 Risks 4

2.6 Use Cases 4

Use Case 1: Startup 5

Use Case 2: Monitor Red LED 6

Use Case 3: Navigation 7

Use Case 4: Simon Carabiner 9

Use Case 5: Etch A Sketch 10

Use Case 6: Rubik’s Cube 11

Use Case 7: Playing Card 12

Stakeholders and Interests 12

Preconditions 12

Postconditions 12

Main Success Scenario 12

Frequency of Occurrence 12

2.7 Sequence Diagrams 13

3. Requirements 19

3.1 Functional Requirement 19

3.2 Nonfunctional Requirements 19

A. Appendicies 20

A.1. Appendix A 20

Glossary 22

Acronyms & Abbreviations 23

References 24

# List of Figures

# List of Tables

# Introduction

<words>

## Purpose

The purpose of this document is to provide a detailed account of the scope, high-level description, and system requirements of the Volt & Pepper System, henceforth known as VPS. The requirements include functional and nonfunctional requirements and system constraints. This document is aimed toward the customer of the VPS. This document is meant to capture the high-level requirements of the VPS.

## Mission Statement

To create a fully autonomous robot that can traverse over a path marked by a white line, and complete four tasks. These tasks include playing Simon for 15 seconds, drawing IEEE on an Etch A Sketch, rotating one row of a Rubik’s cube 180 degrees, and picking up and carrying a playing card to the end of the course.

## Scope

The VPS is intended to compete in the IEEE SoutheastCon 2015 Hardware [7]. SoutheastCon is the annual IEEE Region 3 Technical, Professional, and Student Conference. It brings together Computer Scientists, Electrical, and Computer Engineering professionals, faculty and students to share the latest information through technical sessions, tutorials, and exhibits [6].

## Team Roles

The following table presents all members of the Volt & Pepper System Development Team (VPSDT) and respective role assignments. Each member is accountable for the overesight and advancement of the positions held.

Table 1—Team roles

|  |  |
| --- | --- |
| **Name** | **Role** |
| Nezar Bahksh | Team Leader  Scrum Master  Development Team |
| Greg Carkin | Software Configuration Manager  Development Team |
| Gary Roach | Development Leader  Development Team |
| Brittany Rompa | Testing Leader  Prodct Owner  Development Team |

## Overview

This document is compliant with the standards established in IEEE Std. 1233, 1998 Edition [4]. The document has been divided into three sections. Section 1 serves as a introduction to the VPS, which describes the scope of the project and the team involved. Section 2 contains the general VPS description which includes the product stakeholders, functions of the VPS, and proposed use cases, and Section 3 contains the VPS functional and nonfunctional requirements.

The glossary contains all ambiguous words and phrases, as well as industry terms used in the document. Appendix A serves as the index for all diagrams, tables, and pictures used in the document.

# General Descriptions

The information in this section is provided to improve understanding of the VPS. The areas covered include stakeholders, product perspective, product functions, user characteristics, general constraints, assumptions, dependencies, risks, use cases, and sequence diagrams.

**Stakeholders**

The following is a comprehensive list of individual parties that have a stake in the development, production, and operation of the VPS.

1. **Volt & Pepper System Development Team**

The VPSDT will be graded on the adequate completion of the system by customers (see 2.1.2). Grading of the project includes the VPS itself, along with all artifacts created throughout the 2014-2015 Senior Design course.

1. **Dr. Barott, Dr. Seker, Jorge Torres**

As custumers of the VPS, Dr. Barott, Dr. Seker, and Jorge Torres are interested in the completion of the VPS and all artifacts created throughout the 2014-2015 Senior Design course. Additionally, Dr. Barott and Dr. Seker are interested in the VPSDT meeting the standards set forth by the Accredation Board of Engineering and Technology (ABET), for a Capstone Senior Design project.

1. **Embry-Riddle Aeronautical University**

The VPS will be one of three contending teams from Embry-Riddle Aeronautical University (ERAU) competing to be sent to the IEEE SoutheastCon 2015 Hardware Competition [7]. If chosen, the final product produced by the VPSDT will uphold the prestige of ERAU. In doing so, the VPS must conform to the standards of ERAU as defined in the 2014-2015 Student Handbook **Error! Reference source not found.**

1. **Department of Electrical Computer Software & Systems Engineering**

The Department of Electrical Computer Software & Systems Engineering (ECSSE) at ERAU is interested in the product being delivered on time and within budget, as specified by the budget document for this product (TBD as of 9/18/14).

1. **Nova Southeastern University & Broward College**

Nova Southeastern University & Broward College is the institution hosting the IEEE SoutheastCon Hardware Competition. They are interested in the product complying with all regulations for the competition. It is essential to maintain a safe environment by assurring no product will cause harm to the institiution, or any persons which may come into contact with the product.

1. **Accredation Board of Engineering and Technology, Inc.**

The product must abide by the standards of ABET in order to receive credit for completion of this two-semester course.

## Product Perspective

The VPS is intended to be a self-propelled, autonomous robot that can complete a series of challenges for the 2015 IEEE SoutheastCon Hardware Competition [7].

## Product Functions

The functionality of the VPS is divided into seven major functions: (1) The VPS startup function, referred to as the setup throughout this document, (2) The VPS navigation function, referred to as navigation throughout this document, (3) The VPS Simon challenge function, referred to as Simon throughout this document, (4) The VPS Etch A Sketch challenge function, referred to as Etch A Sketch throughout this document, (5) The VPS Rubik’s Cube function, referred to as Rubik’s Cube throughout this document, (6) The VPS playing card challenge function, referred to as playing card throughout this document, (7) The VPS shut down function, referred to as shut down throughout this document, These functions do not impose a design constraint on the VPS, but are instead used to facilitate the requirements engineering process.

## User Characteristics

The VPS is proposed by an undergraduate senior design team, the VPSDT, as a graduation requirement. It is inteneded to show the high standards asked of all seniors graduating from the Deptartment of ECSSE at ERAU. The VPS is intended for any user capable of powering the system on.

## General Constraints

The VPS is constrained by the IEEE regulations for the Southeast Con competetion, such regulations includes the following:

* The physical size of the VPS can’t exceed 1ft. x 1 ft. x 1 ft.
* The VPS must remain as one unit while operating
* The VPS can’t have any outside influence while operating
* While operating, the VPS can’t move any object outside its station
* The VPS can’t hold any flammable substances
* While operating, the robot must cover the white line underneath it at all time
* Given the course dimensions the size constrain denies us by default from executing multiple challenges simultaneously
* The VPS is constrained to finish all tasks in a time interval less than five minutes

## Assumptions, Dependencies and Risks

## Dependencies

* The VPS depends on IEEE regulations; as the IEEE regulations change, the VPS requirements will also have to update and change correspondingly.
* Funding from the Department of Electrical, Computer, Software, & Systems Engineering at Embry-Riddle Aeronautical.
* All member of the development team being collaborative

## Assumptions

* The VPSDT assumes the VPS can complete all the requirements without running out of power.
* The VPSDT assumes the VPS can be designed, developed, tested and operated without harming any user.
* The VPSDT assumes the VPS can complete all the requirements without falling apart.

## Risks

The VPS might deviate from the white line.

* The VPS might fall off the course.
* The VPS might be misaligned with a challenge.
* The VPS might damage an object on the course.
* The VPS might run out of onboard power while operating.
* The VPS might

## Use Cases

The following use cases demonstrate the proper functionality of the autonomous VPS. These use cases represent the required operation in order to receive maximum points for the competition. Deviation from the normal operation of the VPS may result in a loss of points or disqualification.

## Use Case 1: Startup

Scope: VPS

Level: User goal

Primary Actors: Volt and Pepper team member

Stakeholders and Interests

<insert stakeholders that apply to this use case>

Preconditions

1. The VPS is not on (power to the VPS is disabled)
2. The VPS is not within the 1 ft. x 1 ft. starting square

Postconditions

1. The VPS is on (power to the VPS is enabled)
2. The VPS is within the 1 ft. x 1 ft. starting square

Main Success Scenario

* + - 1. The team member places the VPS within the 1 ft. x 1 ft. starting square, ensuring that the VPS is facing towards the desired path of travel
      2. The team member activates the VPS (enables power to the VPS)

Extensions (Alternate Flows)

<insert alternates>

Frequency of Occurrence

This use case will occur at the start of each round of the competition. Amount of rounds will vary based on amount of participants, successful completion of the course, as well as the time it takes to achieve successful completion of the course.

## Use Case 2: Monitor Red LED

Scope: VPS

Level: User goal

Primary Actors: Red LED

Stakeholders and Interests

<insert stakeholders that apply to this use case>

Preconditions

1. The VPS is on (power to the VPS is enabled) as described in Use Case 1: Startup
2. The VPS is within the 1 ft. x 1 ft. starting square as described in Use Case 1: Startup
3. The Red LED light is on (the light is illuminated)

Postconditions

1. The VPS will begin the task of navigation, as described in Use Case 3: Navigation
2. The VPS is within the 1 ft. x 1 ft. starting square, as described in Use Case 1: Startup
3. The red LED light is off (the light is not illuminated)

Main Success Scenario

1. The VPS will wait for the red LED light to turn off
2. The red LED light will turn off
3. The VPS will begin the task of navigation, as described in Use Case 3: Navigation

Frequency of Occurrence

This use case will occur once during each round of the competition. Amount of rounds will vary based on amount of participants, successful completion of the course, as well as the time it takes to achieve successful completion of the course.

## Use Case 3: Navigation

Scope: VPS

Level: User goal

Primary Actors: White line

Stakeholders and Interests

<insert stakeholders that apply to this use case>

Preconditions

1. The VPS is on (power to the VPS is enabled), as described in Use Case 1: Startup
2. The VPS is within the 1 ft. x 1 ft. starting square or in front of one of the 4 challenges, as described in Use Case 1: Startup, Use Case 4: Simon Carabiner, Use Case 5: Etch A Sketch, Use Case 6: Rubik’s Cube, and Use Case 7: Playing Card
3. The Red LED light is off (the light is not illuminated) or the challenge the VPS is currently in front of has been successfully completed, as described in Use Case 2: Monitor Red LED, Use Case 4: Simon Carabiner, Use Case 5: Etch A Sketch, Use Case 6: Rubik’s Cube, and Use Case 7: Playing Card

Postconditions

1. The VPS will be halted at the finish line or in front of one of the 4 challenges, as described in Use Case 4: Simon Carabiner, Use Case 5: Etch A Sketch, Use Case 6: Rubik’s Cube, and Use Case 7: Playing Card
2. The VPS will be touching the white line of the finish line or the painted white line which makes up the box that the challenge resides in

Main Success Scenario

1. The VPS will follow the white line, as it heads to the finish line or the next challenge
2. The VPS will cover the white line at all times on route to its destination
3. The VPS will visit and complete each challenge, as described in Use Case 4: Simon Carabiner, Use Case 5: Etch A Sketch, Use Case 6: Rubik’s Cube, and Use Case 7: Playing Card
4. The VPS will, after completion of each challenge(as described in Use Case 4: Simon Carabiner, Use Case 5: Etch A Sketch, Use Case 6: Rubik’s Cube, and Use Case 7: Playing Card), will navigate to and halt at the finish line
5. The VPS will remain halted at the finish line

Frequency of Occurrence

This use case will occur five times during each round of the competition. The use case will be used each time the VPS must navigate to the next challenge (four challenges in total, described in Use Case 4: Simon Carabiner, Use Case 5: Etch A Sketch, Use Case 6: Rubik’s Cube, and Use Case 7: Playing Card), then once more to navigate to the finish line. Amount of rounds will vary based on amount of participants, successful completion of the course, as well as the time it takes to achieve successful completion of the course.

## Use Case 4: Simon Carabiner

Scope: VPS

Level: User goal

Primary Actors: Simon Carabiner

Stakeholders and Interests

<insert stakeholders that apply to this use case>

Preconditions

1. The VPS is on (power to the VPS is enabled), as described in Use Case 1: Startup
2. The VPS is directly in front of the Simon Carabiner game, touching the painted white square in which the game resides, as described in Use Case 3: Navigation

Postconditions

1. The VPS is on (power to the VPS is enabled), as described in Use Case 1: Startup
2. The VPS is directly in front of the Simon Carabiner game, touching the painted white square in which the game resides, as described in Use Case 3: Navigation

Main Success Scenario

1. The VPS will depress (push) the center button on the game to start Simon Carabiner
2. The VPS will wait for the visual and audible pattern emitted from the game.
3. The VPS will then duplicate this pattern by depressing (pushing) the corresponding buttons on the Simon Carabiner game.
4. The VPS will then wait for the next visual and audible pattern to be emitted from the game.
5. The VPS will once again duplicate this pattern by depressing (pushing) the corresponding buttons on the Simon Carabiner game.
6. The VPS, after 15 seconds had passed since depressing (pushing) the center button to start the game, will then depress (push) the center button on the game to stop Simon Carabiner
7. The VPS will then begin navigation, as described in Use Case 3: Navigation

Frequency of Occurrence

This use case will occur once during each round of the competition. Amount of rounds will vary based on amount of participants, successful completion of the course, as well as the time it takes to achieve successful completion of the course.

## Use Case 5: Etch A Sketch

Scope: VPS

Level: User goal

Primary Actors: Etch A Sketch

Stakeholders and Interests

<insert stakeholders that apply to this use case>

Preconditions

1. The VPS is on (power to the VPS is enabled), as described in Use Case 1: Startup
2. The VPS is directly in front of the Etch A Sketch game, touching the painted white square in which the game resides, as described in Use Case 3: Navigation
3. The Etch A Sketch will be blank (nothing has been drawn on the Etch A Sketch display)

Postconditions

1. The VPS is on (power to the VPS is enabled), as described in Use Case 1: Startup
2. The VPS is directly in front of the Etch A Sketch game, touching the painted white square in which the game resides, as described in Use Case 3: Navigation
3. The Etch A Sketch will display “IEEE” (the letters “IEEE” have been drawn on the Etch A Sketch display)

Main Success Scenario

1. The VPS will begin to turn the knobs of the Etch A Sketch
2. The VPS will continue to turn the knobs in a manner in which the Etch A Sketch will display the letters “IEEE”
3. The VPS will then begin navigation, as described in Use Case 3: Navigation

Frequency of Occurrence

This use case will occur once during each round of the competition. Amount of rounds will vary based on amount of participants, successful completion of the course, as well as the time it takes to achieve successful completion of the course.

## Use Case 6: Rubik’s Cube

Scope: VPS

Level: User goal

Primary Actors: Rubik’s Cube

Stakeholders and Interests

<insert stakeholders that apply to this use case>

Preconditions

1. The VPS is on (power to the VPS is enabled), as described in Use Case 1: Startup
2. The VPS is directly in front of the Rubik’s Cube game, touching the painted white square in which the game resides, as described in Use Case 3: Navigation
3. The Rubik’s Cube will be solved (no rows are turned from its initial, in package, condition)

Postconditions

1. The VPS is on (power to the VPS is enabled), as described in Use Case 1: Startup
2. The VPS is directly in front of the Rubik’s Cube game, touching the painted white square in which the game resides, as described in Use Case 3: Navigation
3. The Rubik’s Cube will have one row turned 180 degrees (from its initial, in package, condition)

Main Success Scenario

1. The VPS will begin to turn a row on the Rubik’s Cube
2. The VPS will continue to turn the row on the Rubik’s Cube until it has turned 180 degrees
3. The VPS will then begin navigation, as described in Use Case 3: Navigation

Frequency of Occurrence

This use case will occur once during each round of the competition. Amount of rounds will vary based on amount of participants, successful completion of the course, as well as the time it takes to achieve successful completion of the course.

## Use Case 7: Playing Card

Scope: VPS

Level: User goal

Primary Actors: Deck of Playing Cards

Stakeholders and Interests

<insert stakeholders that apply to this use case>

Preconditions

1. The VPS is on (power to the VPS is enabled), as described in Use Case 1: Startup
2. The VPS is directly in front of the deck of playing cards, touching the painted white square in which the game resides, as described in Use Case 3: Navigation
3. The deck of cards will contain 52 cards

Postconditions

1. The VPS is on (power to the VPS is enabled), as described in Use Case 1: Startup
2. The VPS is directly in front of the deck of playing cards, touching the painted white square in which the game resides, as described in Use Case 3: Navigation
3. The VPS will have possession of one playing card
4. The deck of cards will contain 51 cards

Main Success Scenario

1. The VPS will pick up the playing card
2. The VPS will maintain possession of the playing card
3. The VPS will then begin navigation, as described in Use Case 3: Navigation

Frequency of Occurrence

This use case will occur once during each round of the competition. Amount of rounds will vary based on amount of participants, successful completion of the course, as well as the time it takes to achieve successful completion of the course.

## Sequence Diagrams

The following diagrams provide a sequence of actions in order to complete a task. The tasks are based on the requirements needed to complete the course. Initially, the VPS must be turned on and be setup to await the starting signal. Once the VPS recieves the start signal it then transitions into a navigation phase where it remains on the white line as it moves to a challenge. The navigation phase occurs between each of the challenges. The four challenges, Simon, Etch A Sketch, Rubik’s cube, and the playing card each are considered their own tasks. The components of the VPS are broken up into the RobotController, sensors, interactors, and MovementSystem. The RobotController acts as the central hub of the system where commands are given out. The sensors are the sensors for the VPS. The interactors are the physical components used to touch the challenges. Lastly the MovementSystem is the components of the VPS dedicated to traversing the course.

The setup sequence diagram depicts the sequence of actions needed for the VPS to complete its setup of components and start of navigation. The RobotController sends a message to each of the components to setup. Once the sensors, interactors and MovementSystems have setup the Robotontroller sends a message to the sensors to sense the LED for a start signal. Once the start signal is received the RobotController sends messages to the sensors and MovementSystem to start navigation.

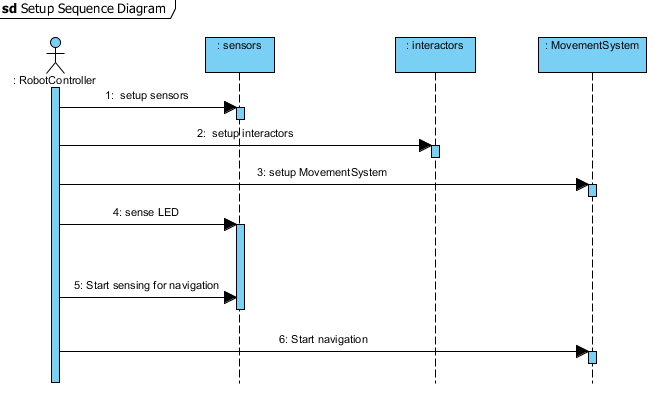


Figure 1—Setup sequence diagram

The navigation sequence diagram depicts the sequence of actions needed for the VPS to successfully traverse the course. Durring this sequence the RobotController is constantly messaging the sensors and MovementSystem to ensure the VPS is staying on the white line.

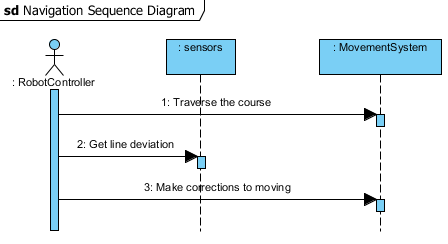


Figure 2—Navigation sequence diagram

The simon sequence diagram depicts the sequence of actions needed for the VPS to successfully complete the simon challenge and enter navigation. Initially the RobotController sends a message to the sensors to find simon. Once the VPS has found simon the RobotController sends messages to the MovementSystem and the interactors to align with simon. Once the VPS is aligned the RobotController sends messages to the sensors and interactors to play simon until the challenge is complete. After the VPS has completed the simon challenge the RobotController sends messages to the sensors, and interactors to stop sensing and interacting with simon. Lastly the RobotController sends messages to the sensors and MovementSystem to start navigation.

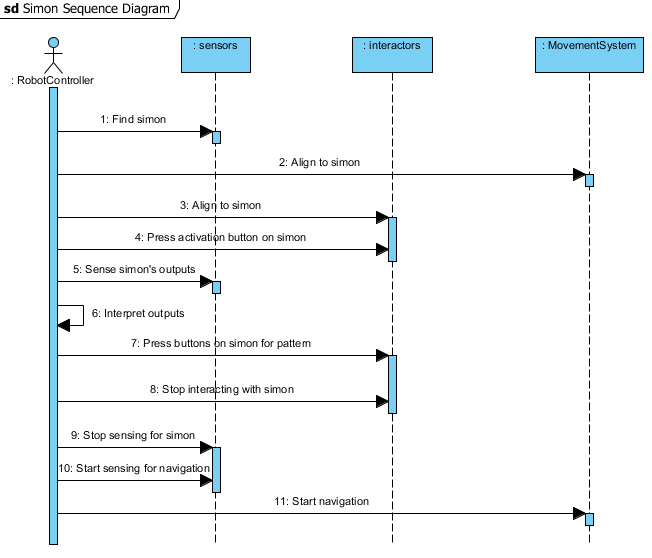
****

Figure 3—Simon sequence diagram

The Etch A Sketch sequence diagram depicts the sequence of actions needed for the VPS to successfully complete the Etch A Sketch challenge and enter navigation. Initially the RobotController sends a message to the sensors to find the Etch A Sketch. Once the VPS has found the Etch A Sketch the RobotController sends messages to the MovementSystem and the sensors to align with the Etch A Sketch. Once the VPS is aligned the RobotController sends messages to the interactors to print IEEE onto the Etch A Sketch. After the VPS has printed IEEE on the Etch A Sketch the RobotController sends messages to the sensors, and interactors to stop sensing and interacting with the Etch A Sketch. Lastly the RobotController sends messages to the sensors and Movement system to start navigation.

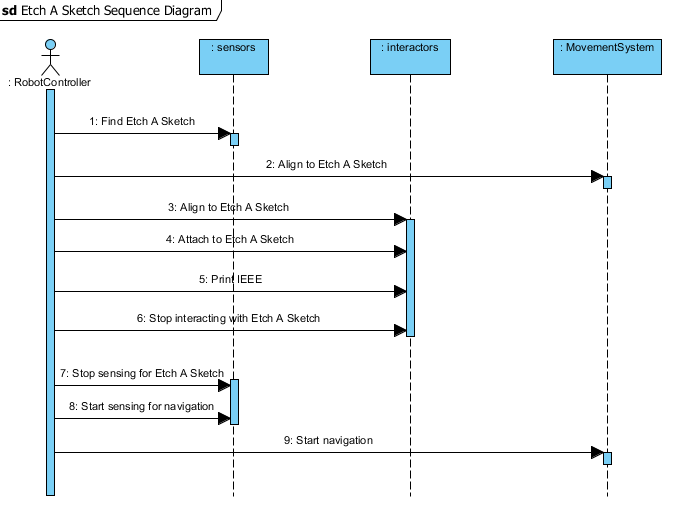


Figure 4—Etch A Sketch sequence diagram

The Rubik’s Cube sequence diagram depicts the sequence of actions needed for the VPS to successfully complete the Rubik’s Cube challenge and enter navigation. Initially the RobotController sends messages to the sensors to find the cube. Once the sensors locate the cube the RobotController sends messages to the MovementSystem and the interactors to align with the cube. Once the VPS is aligned the RobotController sends messages to the interactors to rotate the cube. After the VPS has rotated the cube the RobotController sends messages to the sensors and interactors to stop sensing and interacting with the cube. The RobotController then sends messages to the sensors and MovementSystem to start navigation.

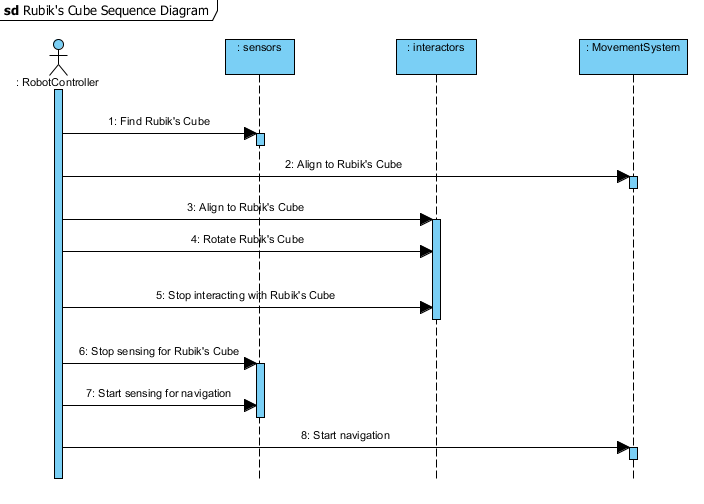


Figure 5—Rubik's Cube sequence diagram

The card sequence diagram depicts the sequence of actions needed for the VPS to successfully complete the card challenge and enter navigation. Initially the RobotController sends a message to the sensors to find the playing card. Once the VPS has found the card the RobotController sends messages to the MovementSystem and the interactors to align with the card. After the VPS is aligned the RobotController sends a message to the interactors to take hold of a card. The RobotController then sends messages to the sensors stop sensing for the card. Lastly the RobotController sends messages to the sensors and the MovementSystem to start navigation.

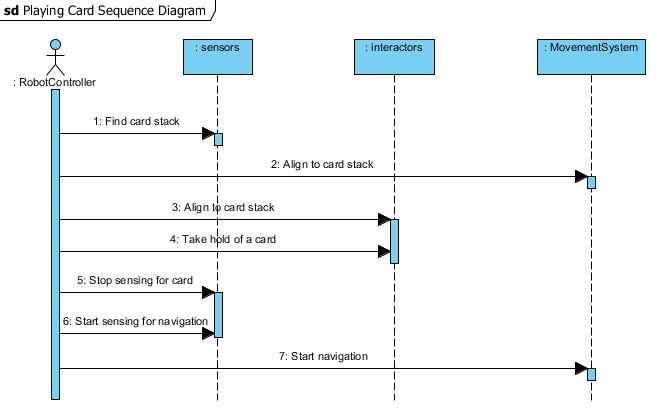


Figure 6—Playing card sequence diagram

# Requirements

## Functional Requirement

* + 1. The VPS shall exit the 1 ft. x 1 ft. starting box if the red LED is off.
    2. The VPS shall traverse the white line on the course.
       1. The VPS shall remain on the white line for at least TBD % of the course.
    3. The VPS shall play the Simon game for 15 seconds.
       1. The VPS shall press the middle button on the Simon game.
       2. The VPS shall react to the Simon game in real-time.
       3. The VPS shall press the buttons on the Simon game in the correct sequence.
       4. The VPS shall respond to the Simon game in TBD seconds.
       5. The VPS shall leave the Simon game in the Simon game area.
    4. The VPS shall draw the letters “IEEE” on the Etch A Sketch in a TBD lettering style.
       1. The VPS shall rotate the Etch A Sketch knobs to draw the letters “IEEE”.
       2. The VPS shall leave the Etch A Sketch in the Etch A Sketch area.
    5. The VPS shall rotate one row of the Rubik’s Cube 180 degrees relative to the other two rows.
       1. The VPS shall leave the Rubik’s Cube in the Rubik’s Cube area.
    6. The VPS shall obtain one playing card from the stack of cards on the course.
       1. The VPS shall cross the finish line holding the obtained playing card.
       2. The VPS shall leave the remaining cards in the playing card area.
    7. The VPS shall cross the finish line after completing all games.

## Nonfunctional Requirements

* + 1. The VPS shall fit within 1 ft. x 1 ft. x 1 ft. if it is in the starting square.
    2. The VPS shall fit within 1 ft. x 1 ft. x 1 ft. to cross the finish line.
    3. The VPS shall operate autonomously.
    4. The VPS shall operate with an on-board power supply.
    5. The VPS shall remain within the bounds of the course.
    6. The VPS shall cross the finish line in at most 5 minutes from the LED turning off.
    7. The VPS shall remain in the 1 ft. x 1 ft. square while the LED is on.

# A. Appendicies

## A.1. Appendix A

The following figures are a supplemental visual aid of the IEEE SoutheastCon 2015 Hardware Competition course and challenge components.

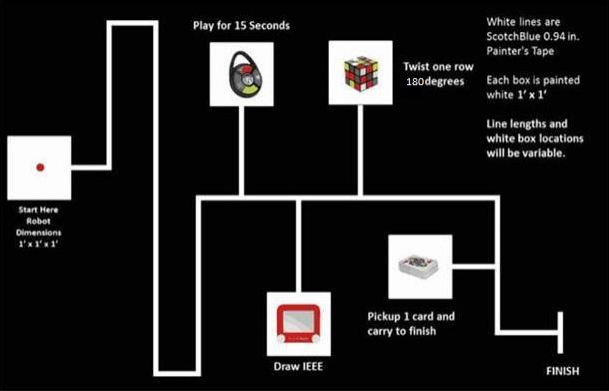
[](#_Table_of_Figures)

Figure 1—Course for IEEE SoutheastCon 2015 Hardware Competition [7]



Figure 2—Etch A Sketch [9]



Figure 3—Standard 52-deck of playing cards [10]



Figure 4—Rubik’s 3x3 Cube [1]



Figure 5—Simon Carabiner [11]

# Glossary

|  |  |  |
| --- | --- | --- |
| **Entry** | **Definition** | **Alias** |
| autonomous | When activated, it is independent with no outside influence |  |
| challenge | One of the four tasks- Simon, Etch A Sketch, Rubik’s Cube, or playing card | Task |
| course | 5/8 in. x 4 ft. x 8 ft. Sanded Pine Plywood |  |
| deck of cards | Standard 52-card deck (see Appendix A.1, Figure 3) | Deck |
| draw | To produce or print |  |
| emergency | Unexpected occurrence requiring human intervention |  |
| Etch A Sketch | Pocket Etch A Sketch by: Ohio Art (see Appendix A.1, Figure 2) |  |
| finish line | Refer to “FINISH” (see Appendix A.1, Figure 1) | Finish |
| interact | Physically affecting by executing the functional requirments |  |
| line | Scotch Blue 0.94 in. x 60 yd. Painter’s Tape |  |
| obtain | To have possesion of |  |
| playing card | A card from the standard 52-card deck (see Appendix A.1, Figure 3) | Card |
| robot | The platform being built for the IEEE SoutheastCon 2015 Hardware Competition. | VPS |
| Rubik’s Cube | Rubik’s 3x3 Cube (see Appendix A, Figure 4) |  |
| sequence | Simon, Etch A Sketch, Rubik’s Cube, playing card, finish line |  |
| Simon | Simon Carabiner (see Appendix A, Figure 5) |  |
| traverse | To move across |  |

# Acronyms & Abbreviations

|  |  |
| --- | --- |
| **Entry** | **Expanded Phrase** |
| ABET | Accredation Board for Engineering and Technology, Inc. |
| ERAU | Embry-Riddle Aeronautical University |
| ECSSE | Electrical, Computer, Software & Systems Engineering |
| IEEE | Institution of Electrical and Electronics Engineers, Inc. |
| SyRS | System Requirements Specifications |
| VPS | Volt & Pepper System |
| VPSDT | Volt & Pepper System Development Team |

# References

1. Booyabazooka. *Rubik’s Cube.* March 5, 2008. http://commons.wikimedia.org/wiki/File:Rubik%27s\_cube.svg (accessed September 13, 2014).
2. ERAU. “Student Handbook - Embry-Riddle Aeronautical University.” *ERAU - Daytona Beach, FL.* 2014. http://daytonabeach.erau.edu/Assets/daytonabeach/forms/daytonabeach-student-handbook.pdf (accessed September 13, 2014).
3. IEEE. *IEEE Citation Reference.* September 9, 2014. http://www.ieee.org/documents/ieeecitationref.pdf (accessed 2014).
4. IEEE. IEEE Guide for Developing System Requirements Specifications. 1998. New York, NY: IEEE, Decembeer 22, 1998.
5. IEEE. *IEEE Recommended Practice for Software Requirements Specifications.* New York, NY: Institute of Electrical and Electronics Engineers, Inc., 1998.
6. —. *IEEE Region 3.* 2014. http://www.ewh.ieee.org/reg/3/southeastcon/ (accessed 2014).
7. —. “SoutheastCon 2015 Hardware Competition Rules (DRAFT).” *IEEE.* March 19, 2014. http://sites.ieee.org/sb-unfc/files/2014/07/hardwareComp2015.pdf (accessed September 13, 2014).
8. IEEE Standards Association. *2014 IEEE-SA Standards Style Manual.* New York, NY: IEEE, 2014.
9. Ohio Art. *Pocket Etch A Sketch - Red.* Ohio Art. 2014. http://www.toysrus.com/buy/etch-a-sketch-doodle-pro/pocket-etch-a-sketch-red-5163-2395954 (accessed September 13, 2014).
10. Plank Fitness. *Deck of Cards “Anywhere” Workout.* January 3, 2014. http://plankavl.com/wp-content/uploads/2014/01/deck-cards.png (accessed September 13, 2014).
11. The Bridge Direct, Inc. *Simon: Carabiner Edition.* 2012. http://www.thebridgedirect.com/sim\_prd\_carabiner.php (accessed September 13, 2014).