Owen Chastain
Penney Fagan
Brent Evans
Jan-Erik Hustrulid
Akata Patel

SRS for

Simple Satellite Tracker 2003

Janusz Zalewski, Ph.D

ISM4331, Information Systems Design

Fall 2003

SRS – Table of Contents

Introduction

1.1 Purpose

The purpose of this SRS is to explain the requirements for our satellite tracking application. The intended audience of our SRS is designers and project managers.

1.2 Scope

The scope of this document provides an overall description of the software and its specific requirements.

1.3 Definitions, Acronyms, & Abbreviations

1.3.1 database. A file that contains the specific coordinates of the satellite unit of time.

1.3.2 RS-232-C. Hardware specification for PC serial ports.

1.3.3 satellite. Objects that we can track with our telescope application. [1]

1.3.4 telescope. The Meade LX200 model telescope. [1]

1.4 References

[1] Meade Instruments Corporation, Meade LX200 Electronic Manual, Meade Corporation, Irvine, California, September 21, 2003, http://www.meade.com/manuals/lx200/index.html

[2] The Institute of Electrical and Electronics Engineers, Inc., IEEE Recommended Practice for Software Requirements Specifications (IEEE Std 830-1998), The Institute of Electrical and Electronics Engineers, Inc., New York, NY, June 25, 1998

1.5 Overview

The remainder of this document contains an overall description of the software including product perspective, product functions, user characteristics, constraints, assumptions, and dependencies. This document also contains specific requirements for the application.

2 Overall description

2.1 Product perspective

2.1.1 System Interfaces

2.1.1.1 This product interfaces with a telescope using a RS-232-C serial specification. [1]

2.1.2 User Interfaces

2.1.2.1 The interface of the software is a form that requests the location and name of the satellite.

2.1.3 Hardware Interfaces

2.1.3.1 This product will require a computer with free RS-232-C serial port and a Meade LX200 telescope. [1]

2.1.4 Software Interfaces

2.1.4.1 No additional software is required for the use of our application.

2.1.5 Communications Interfaces

2.1.5.1 The application will communicate with the telescope via the RS-232-C serial port. [1]

2.1.6 Memory Constraints

2.1.6.1 Given the state of current hardware and software technologies, there are no constraints on the use of primary and secondary memory or storage.

2.1.7 Operations

2.1.7.1 The initial state of the application after execution is stand-by mode.

2.1.7.2 Upon entering coordinates of a satellite and clicking the “Track it!” button, the application will validate the input data, and enter tracking mode.

2.1.7.3 The application will remain in tracking mode until the application is terminated.

2.1.8 Site Adaptation Requirements

2.1.8.1 There are no initialization requirements for the software.

2.1.8.2 There are no local adaptation requirements.

2.2 Product Functions

2.2.1 The software will find a satellite based upon coordinates supplied by the user.

2.2.2 The software will receive updated coordinates from the telescope, display them on the screen, and record them in a database after a predetermined period of time. (Fig. 1)

2.3 User Characteristics

2.3.1 Any user with an interest in tracking satellites will be able to use this program.

2.4 Constraints

2.4.1 The database is viewable with a text-based authoring utility.

2.5 Assumptions and Dependencies

2.5.1 It is assumed that this application will be installed to an operating system that meets the minimum specifications of its respective platform.

2.6 Apportioning of Requirements

2.6.1 Graphical representations of the satellite's path will be delayed and reserved for future versions.

3 Specific Requirements

3.1 External interfaces (Fig. 1)

3.1.1 Software shall accept coordinates from the keyboard.

3.1.2 Software shall validate coordinates and submit them to the telescope. (Fig 2)

3.1.3 Software shall accept changing coordinates from the telescope. (Fig. 3)

3.1.4 Application shall display the updated coordinates after a predefined period of time.

3.1.5 Application shall write the updated coordinates to the database after a predefined period of time. (Fig. 3)

3.2 Functions

3.2.1 Input Requirements

3.2.1.1 The software shall verify that all data entered into its form are real numbers.

3.2.1.2 The software shall verify that all data entered into its form are coordinates.

3.2.1.3 The software shall accept all changing coordinates from the telescope without validation.

3.2.2 Output Requirements (Fig. 3)

3.2.2.1 The software shall output the coordinates on the screen as they are received.

3.2.2.2 The software shall replace the existing coordinates with new coordinates as they are received.

3.2.2.3 The software shall output the coordinates to the database as they are received.

3.2.2.4 Coordinates written to the database shall terminate with a carriage return and line feed character.

3.2.2.5 Velocity of the satellite will be written to the form.

3.2.2.6 The system shall provide an error message if invalid data is entered into the form.

3.2.2.7 The system shall provide an error message if the “Track” button is clicked before relevant parameters have been entered.

3.2.2.8 The system shall provide an error message if the telescope is unable to find an object at the specified coordinates.

3.2.3 Processing Requirements

3.2.3.1 The software shall calculate the velocity of the satellite according to the coordinates received from the telescope.

3.3 Performance Requirements

3.3.1 Reliability

3.3.1.1 The application shall perform its task while the computer and telescope are properly powered.

3.3.2 Availability

3.3.2.1 The application shall not have any availability issues due to the lack of multi-user, multithreading, or network interfaces.

3.3.3 Security

3.3.3.1 The application shall not implement any security constraints.

3.3.4 Maintainability

3.3.4.1 The application shall not have any maintainability requirements.

3.3.5 Portability

3.3.5.1 The application shall run on portable computers if desired.

3.4 Organizing the Specific Requirements

3.4.1 System Modes

3.4.1.1 The initial program state upon execution of the software is a stand-by mode where the application will prompt the user to enter the coordinates of a satellite.

3.4.1.2 The software will validate the coordinates once they are entered. If the coordinates are valid, the software will go into tracking mode.

3.4.2 User Class

3.4.2.1 There are no administrative functions associated with this application; therefore it is not necessary to implement user classes.

3.4.3 Objects

3.4.3.1 The only external objects associated with this program are the telescope, the satellites it tracks, and the user who operates the software.

3.4.4 Feature

3.4.4.1 The initial version of the application will be stand-alone and completely independent of any external systems or processes.

3.4.5 Stimulus

3.4.5.1 Poor weather or cloudy conditions may affect the ability of the telescope to track a satellite.

3.4.6 Response

3.4.6.1 The only response of this application is the recording of coordinates at predetermined lengths of time as supplied by an external telescope.

3.4.7 Functional hierarchy

3.4.7.1 The application requests coordinates from the end user.

3.4.7.2 The application validates coordinates received from the end user.

3.4.7.3 If the coordinates are valid, the software will send those coordinates to the telescope.

3.4.7.4 If the telescope is able to find the object at those coordinates, the software displays the coordinates on the screen and enters tracking mode.

3.4.7.5 The application will write the changing coordinates to a database at predetermined moments until the application is terminated.

*** END SRS DOCUMENT ***