## Earth Station Controller Subsystem

### Introduction

#### Purpose

The purpose of this Software Requirements Specifications (SRS) document is to provide a detailed description of the functionalities of the Earth Station Controller subsystem. This document will cover each of the subsystem’s intended features, as well as offer a preliminary glimpse of the software application’s User Interface (UI). The document will also cover hardware, software, and various other technical dependencies.

#### Intended Audience and Reading Suggestions

This document is intended for all individuals participating in and/or supervising the Earth Station Controller. Readers interested in a brief overview of the product should focus on the rest of the Introduction, as well as the Overall Description section which provides a brief overview of each aspect of the project as a whole.

Readers who wish to explore the features of the Earth Station Controller in more detail should read on to the System Features, which expands upon the information laid out in the main overview. External Interface Requirements offers further technical details, including information on the User Interface as well as the hardware and software platforms on which the application will run.

Readers interested in the non-technical aspects of the project should read Other Nonfunctional Requirements, which covers performance, security, and various other attributes that will be important to users. Readers who have not found the information they are looking for should check Other Requirements, which includes any additional information which does not fit logically into the other sections.

#### Project Scope

The Earth Station Controller Subsystem consists of three major components. The first component is an application server which runs on the main site and serves as a proxy to all other sites. It is intended to be the entrance point to all users receiving data from more than one site. The second component is a series of remote streaming applications responsible for providing a connection to a data stream through the application server. The final component is a client application or user interface running on a web browser on the users’ computer.

### Overall Description

#### Product Perspective

The Earth Station Controller Subsystem consists of three major components: a server, streamers, and a font end client. The front end client component is served by the server component, a Node.js server application that is a proxy to all streamer locations. The front end client is intended to be the entrance point for all users receiving data from more than one streamer. Each streamer will provide a series of data streams to the application server, which is responsible for providing the appropriate connections to the front end client running in the web browser on the user’s computer.



Figure 6.4‑1 Earth Station Controller Block Diagram

The Earth Station Controller Subsystem provides access to the system on-site or remotely. The user can observe current or past activities, control the system mode of operation, view system status, and have access to all of the services the subsystem has to provide. It is the entity responsible to receive reports from other subsystems and communicate them to either the authorized NOAA personnel or the carrier as needed. It also provides a vehicle to store and access generated intrusion reports as well as forensic data containing detailed information about the intrusion.

One of the most critical functions of the Earth Station Controller Subsystem is to determine the action to be taken as a result of interference events. The actions are dictated by the end user and are meant to be automated on the event of intrusion detection.

1. Core Features

* Login Page
  + Users can request a new account
  + Users can reset a lost password
  + Users credentials are checked and validated
  + Authorization and authentication takes place here
* Status Page
  + Quick assessment of the system’s health
  + Easy navigation to all sites connected
  + Continuous display of internal messages
* Operation Mode
  + Displays current running mode
  + Operation mode can be overwritten based on user’s credentials
  + Depicts data stream as is being seen by the system
* Stream Details
  + Shows more details about a single site
  + Data streams from the selected are shown
  + Internal messages from that site only are displayed
* Data Query
  + Allows the user to query previous results
  + Can be used to generate reports quickly
  + Stores user’s queries and reports
* System Diagnostics
  + Allows user to run additional diagnostic tests
  + Provides access to test log files for analysis
  + Generates reports
* Signal Analysis
  + Used to analyze anomalies in the system
  + Mostly used by the system developers
  + Allows quick access to resources to assess system’s behavior
* Intrusion Reporting
  + Notifies users of intrusions detected by the system
  + Notification mechanism is determined by the site

1. Additional Features

* Users Notifications
  + Allows users to receive notifications of events
  + Different events and actions can be set by the user
* Setting Page
  + Allows users to set application preferences for customizable behavior
  + Provides a simple way for users to manage their account
* Administrator Tools
  + Allows administrators to manage streamer connections
  + Provides a way for administrators to manage users
  + Offers administrators access to the application’s general configuration
* System Running Mode
  + System can be configured to run on different modes based on needs
  + The modular design allows for quick and easy decoupling of the main components
* Help
  + Displays a list of topics covering the different components
  + Offers detailed information on each feature, menu, etc.
  + Can be accessed at any time via the drop down menu

#### Product Perspective

#### Operating Environment

The Earth Station Controller Server and the Streamer components will run on server nodes hosted by Linux Red Hat operating system. Both components will rely on several applications built into Linux services. One of the main dependencies the system has is Node.js. Node.js is the technology used to run the components and interfaces with the Linux environment to access the system.

The Earth Station Controller Client has a different set of dependencies as it runs on the client’s web browser. The client application uses AngularJS 2.0 and Bootstrap as the framework. This combination offers the best option to allow the application to send and receive data efficiently as well as being modular enough for a better architecture.

#### Design and Implementation Constraints

The primary design constraint is the geographical locations of all the different Earth Stations sites. Each site will have one or more connections streaming data near real time. To minimize the use of network traffic the Earth Station Controller will follow the Star Network Topology (Figure 1.1.2.3-1). The architecture is flexible enough to allow for a Fully Connected Topology (Figure 1.1.2.3-1).



**Figure 1.1.2.3-1 Network Topologies comparison**

As shown in the picture above the number of connections in the Fully Connected Network is almost 3 times as many as the Star Network Topology. Each Earth Station Controller can be set as the main site and therefore the designated main site could be strictly based on location and/or network accessibility.

#### User Documentation

After installation the team will provide training to the system’s end users on each earth station site. The training will contain step-by-step explanation of each of the main components of the user interface. Also, the Earth Station Controller Client application will include a Help Section containing a collection of topics covering each of the application’s menus, features, etc. At any time, the user can navigate to the Help menu and select any of these topics to obtain more information.

#### Assumptions and Dependencies

Earth Station Controller is designed to work on a semi-automatic mode with little or no external interaction other than monitoring the system. The monitoring and reporting features of the system can be reached assuming the following dependencies are met.

1. External Connections

Earth Station Controller key requirements include being able to connect remotely as well as sending intrusion notifications to carriers. In order to meet these requirements the system needs to have network access.

1. Carrier Notifications

Carriers may have different ways to receive information about intrusions incurred by their systems. The Earth Station Controller needs to meet those requirements and therefore it has a dependency on their notification Application Programming Interface (API) specifications.

1. Remote Operators

The system is designed not only to allow operators to monitor and control it, but also to set preferences to receive notifications. The notifications are triggered by events and the action to take when that happens can be set in the user’s preference page. The choices include: computer notifications, emails, and SMS text messages. The last two notification options require that the system has the capacity of doing so by being properly configured.

1. Supporting Packages

The Earth Station Controller uses MySQL to store events and results and therefore it must be accessible. Preferably the database should be host within the Earth Station system, but other locations can be supported as well. All the internal communication between the subsystems is done through Java and C++ Message Service (JMS/CMS) which requires a messaging server. ActiveMQ has been selected as the server of choice due to its high performance and ease of use.

### System Features

The Earth Station Controller features are divided into two main categories: core features and additional features. Core features form the body of the application and include any features that are essential to the functionality of the subsystem. These features must be implemented in order to have a fully-functioning application. Additional features are not required for the subsystem to function. They include enhancements and features to facilitate the system automation.

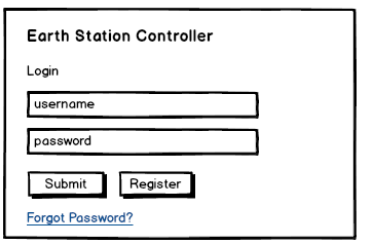
#### Core Features

1. Welcome and Login Page

The user is greeted by a welcome page when he or she goes to the Earth Station Controller main page. In that page the user has the choice of login in, register, or reset the password

1. Login

If the user has previously registered he or she would only need to provide the username and password and click on the submit button to get access to the system.

  
**Figure 1.1.3.1.1-1 Login Screen**

1. Register

If the user does not have a login username and password, he or she can register using the same web application. The application will ask for the user’s name, email address, password, and account role. The account role determines the level of access to the system. All users need to be verified and approved by an administrator before the user can access the system. The only exception is if the user selects Monitor role. This role allows you to monitor the system, but nothing else.

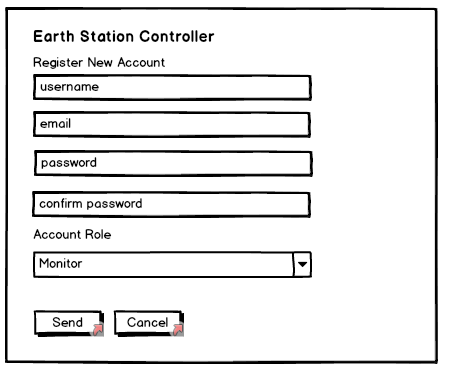
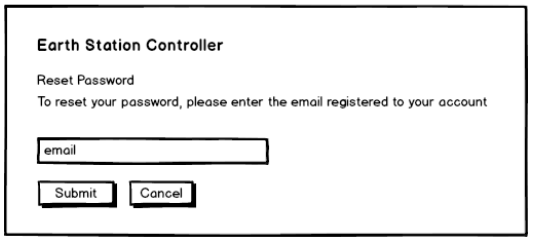


Figure 1.1.3.1.2-1 Registration Screen

1. Password Reset

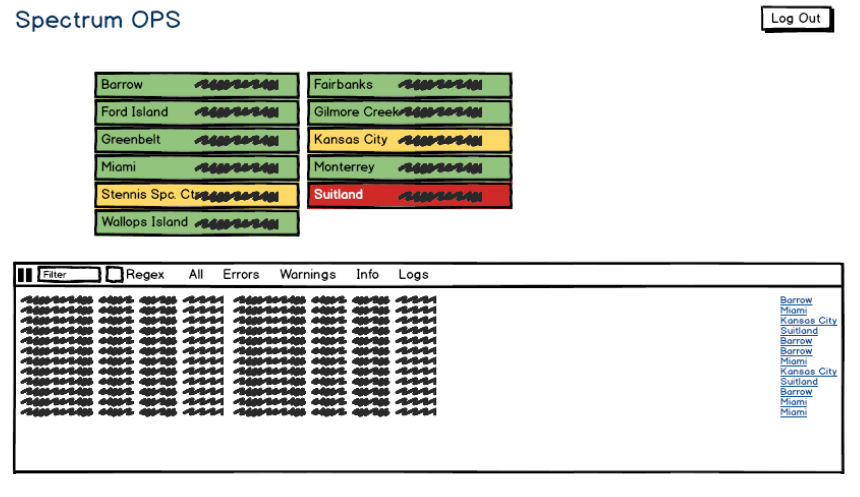
In cases where the user forgets his or her password, the application allows him to reset it. Once the user selects the “Forgot Password?” link, he or she will be forwarded to the password reset page. There the user can enter the email address associated with his account and a unique browser link is sent to his or her email to allow creation of a new password.

  
Figure 1.1.3.1.3-1 Password Reset Screen

1. Status Page

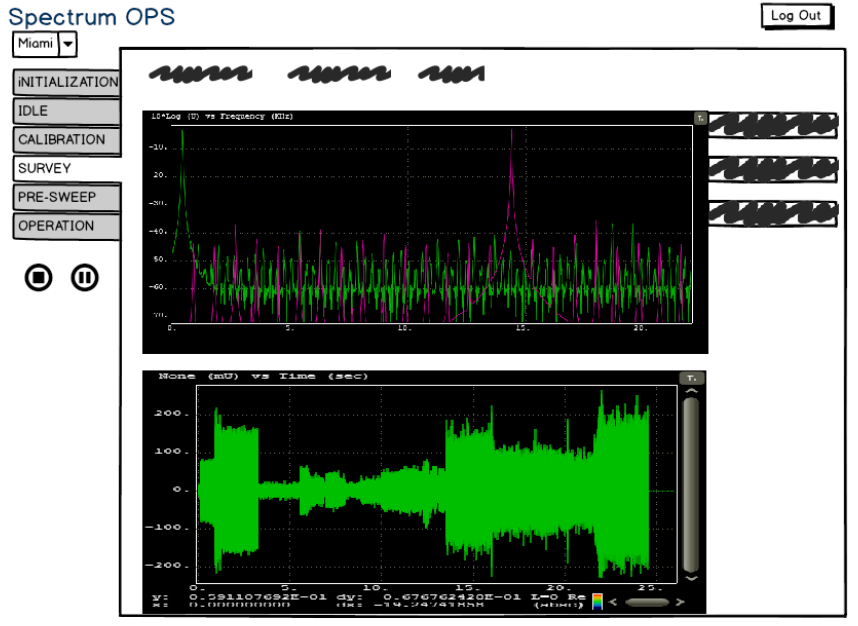
The status page is designed to give users a quick assessment of the system. It contains a color-coded set of labels with the name of each site being monitored. The color on the labels indicates the corresponding site’s system health. If the user clicks on one of the labels he or she will be redirected to the site details page where more information about the status of a particular site can be found.

The status page will also contain a rolling log window depicting relevant messages being sent regarding all connected Earth Station systems. Messages can be filtered to display only those that are of interest to the user.

  
Figure 1.1.3.2-1 Status Page

1. Operation Mode

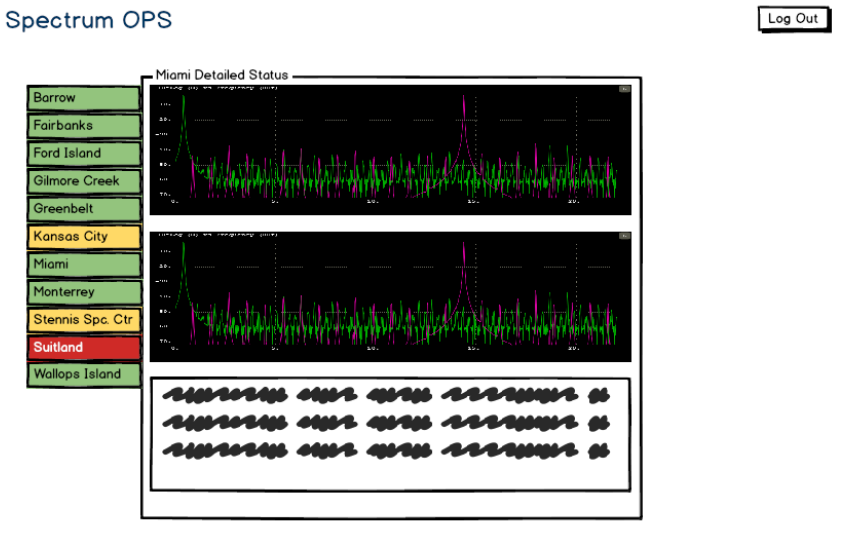
This page allows the user to view the current operation mode of a specific Earth Station system. Under normal operation the mode is set automatically, but the operator can override it as long as he or she has the appropriate role. The Operation Mode page depicts streams of data being produced and rendered near real-time. Real time streams are a great source of information to quickly determine anomalies in the system. The application will allow users to view different sites one at the time based on their account roles.

  
**Figure 1.1.3.3-1 Operation Mode**

1. Earth Station Details

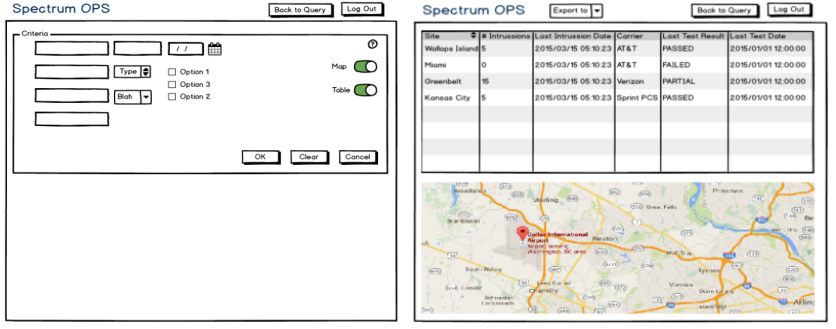
It is important having a good understanding of the environment affecting a particular Earth Station site. The Earth Station Details will provide information about such environment as well as specific operational details of the system itself. Environmental details include, but not limited to, nearby cellular phone towers, emitters, and other possible sources of interference. It will also include information such as number of detections found, near real time streaming data based on the operation mode, etc.

The details page will also contain a rolling log window depicting relevant messages being sent regarding the specific Earth Station system. Messages can be filtered to display only those that are of interest to the user.

  
**Figure 1.1.3.4-1 Operation mode**

1. Results Query

The Earth Station Controller will allow users to query its database for previous results. Results will be depicted either on a tabular form, on a map, or both. Also, the user will be able to export the results as a comma separated value (CSV) or as keyhole markup language (KML) file. The application will store queries in the user’s profile to simplify recurring report generations and queries. Authorized users will have access to intrusion detections, sweep and survey results as well as other relevant information.

  
**Figure 1.1.3.5-1 Database query and results**

1. System Diagnostics

Authorized users will be able to run system diagnostic tests that are not part of regular operation. The application will execute the tests as soon as the system is in capacity of doing so. In other words, if the system is busy and the test would interfere with its operation, then the test will be scheduled but not executed. Users will be able to quickly determine the last time a test was ran as well as be able to retrieve details or log files from selected test(s). This will provide useful information to assure the system is performing as expected and to diagnose its behavior if otherwise.

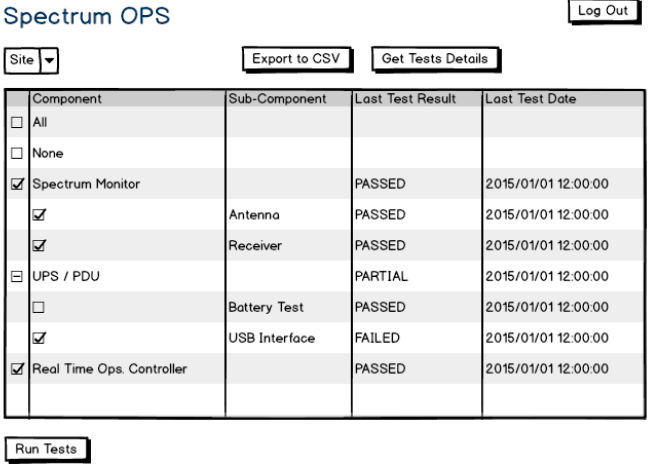
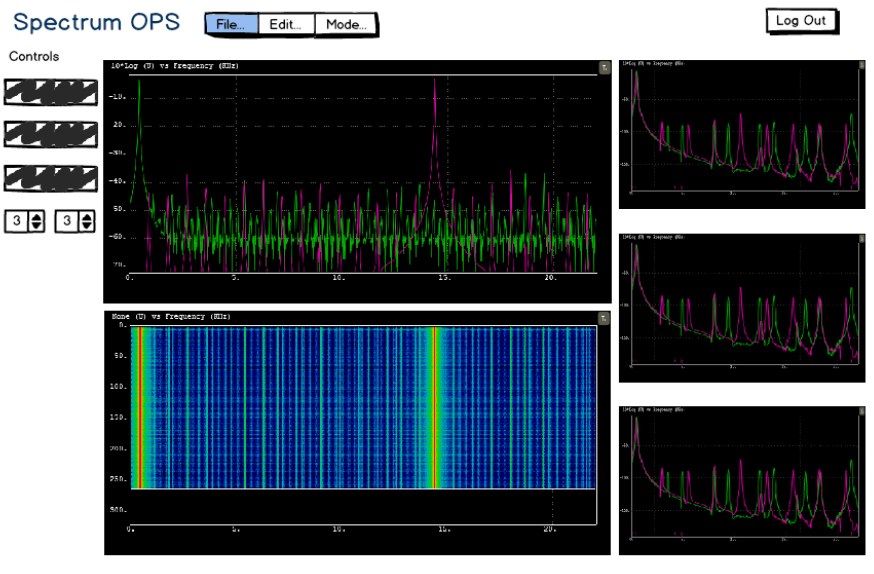
  
**Figure 1.1.3.6-1 System diagnostics page**

Figure 1.1.3.6-1 shows a mock page where users can select the tests to run or export results. The results can be exported as a Comma Separated Values (CSV) file containing all the test results. Users could also request the log file from a particular test or tests for a more in depth assessment.

1. Signal Analysis

There might be situations where the system is behaving oddly and a diagnosis is needed. In such cases, having an analysis page helps facilitates the process. The application will have the capability to perform simple signal analysis remotely in order to quickly test different scenarios. It will also allow users to download and play snapshot files containing particular scenarios. The intention of this page is to be accessible only to developers and analysts rather than general users.

  
**Figure 1.1.3.7-1 Signal analysis page**

1. Intrusion Reporting

When an intrusion is detected, the system is required to report the incident in a timely manner. The event is handled by the action manager which is the entity responsible for sending notifications to the interested parties. The actions taken when such events occur include notifying the carrier, notifying the user(s) monitoring the system, and storing the event on the database. The system will also store signal data containing information about the intrusion that could potentially be used for forensic investigations.

#### Additional Features

1. User Notifications

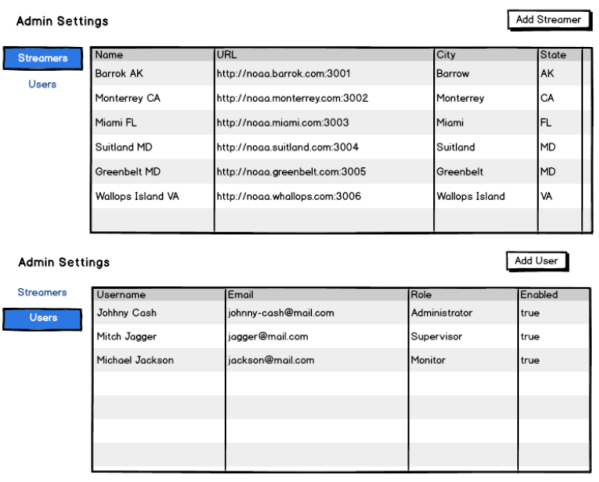
During normal operation the system generates events such as changing operation mode, sweep and survey results, activities, etc. The application will allow users to subscribe to these types of events and set the way the system will notify it. The method of notifying user of such events includes pop-up windows on the user’s workstation, SMS messages, and emails. Other methods could be added to the system based on requirements.

1. Settings Page

Customizing an application gives you the opportunity to automate routines that you would otherwise have to do manually on a daily basis. Having a settings page allows operators to store data in their profile which in turn will increase their productivity. Also, automating these events will minimize the chances of missing an event while the operator is looking away from their workstation.

1. Administrator Tools

Administrators need to be able to configure the system as required on each Earth Station. The Administrative page will allow them to add, remove, edit users as well as which streams of data to subscribe. Only those users will administrative role will be able to access this page. The management of users and streamers can be handled externally if requested. Removing this capability is a straight forward action with low impact to the system due to its modular design.

  
**Figure 1.1.3.2.3-1 Administrative tools**

1. Help

As mentioned previously, the application will contain a Help section where users will be able to get detailed information about specific features. The help page will be accessible to any authorized user. The help page will have a contents menu that users can use to quickly select what they need to review.

### External Interface Requirements

#### Hardware Interfaces

Earth Station Controller is intended as a web enabled application for the Linux platform. The underlying hardware is not relevant to the application as long as is supported by Linux. The application requires connectivity to other internal components as well as external and therefore requires network support hardware. Processing requirements are driven by the number of simultaneous connections the server has to endure at any given time. As a minimum the server should have 4 cores and 64 MB of RAM memory.

#### Software Interfaces

The Earth Station Controller uses different software packages to run the client as well as the server. There are different software packages requirements based on whether the software is being developed or deployed. We are including software packages required for operational purposes. The following is a list of the most critical packages including the minimum version required:

* Node.js 4.4.3
* Node Package Management (NPM) 3.6.0
* Angular 2.0.0-rc.0
* Boostrap 4.0 & JQuery
* SocketIO 1.4
* ExpressJS 4.13

Users need a modern web browser in order to support the technologies used by the Earth Station Controller Client. Angular2 supports Internet Explorer 9 and above as well as reasonable current versions of Chrome and Firefox.

The Earth Station Controller subsystem also depends on ActiveMQ 5.13.2 to handle all the internal messages sent across all the Earth Station subsystems. Finally, results and other data required for day to day operations will be stored in a MySQL 5.7.12 database.

#### Communication Interfaces

The Earth Station Controller will have a network server and streamers that are web-enabled using Node.js technology. These applications exist to retrieve information from the database and also to stream data from multiple sites to the web server. Each Earth Station is composed of multiple subsystems. The communication among them relies on having and internal network connectivity to send and receive JMS/CMS messages and/or to send packets of streamed data.

### Other Nonfunctional Requirements

#### Performance Requirements

The Earth Station Controller should be responsive and able to handle multiple streams of data simultaneously. Performance will be more heavily influenced by network connectivity than any other factor. Designating a particular Earth Station site to host the Earth Station Server is critical to assure responsiveness on the client side. In cases where remote sites have a limited connectivity, users will be able to connect directly to that site, but will be able to monitor and control that site only.

#### Security Requirements

The Earth Station Controller will follow all the security guidance and requirements to assure the integrity of the system as a whole. It will include algorithms to validate inputs from users to avoid different types of hacking attacks. It will also use different techniques to validate each connection request before granting it. Each account role will have a specific timeout determined by inactivity. This timeout will log out the current user to avoid others taking over his or her session.

The Earth Station Controller will also follow all the recommendations suggested by our security team. It will meet items relevant to the system to be in compliance with NIST 800-18 and NIST 800-53 guidelines.

Appendix A

Glossary

* **Earth Station Controller (ESC):**  A series of components used to command, control, and monitor an Earth Station site.
* **Component**: a "section" of the full Earth Station Controller stack (streamer, server, gui) which may be made up of multiple running applications.
* **Application**: an individual code executable, be it a server, database, or other utility.
* **Streamer (ESC Streamer)**: a Node.js application running in juncture with the Earth Station Controller hardware to stream information to and from the Earth Station Controller web server.
* **Server: (ESC Server)**: the server providing the front end client application (ESC client) and it's connection to streamer data & the database.
* **Site (ESC deployed site)**: A physical location where a deployed Earth Station Controller hardware & streamer are hosted (optionally a ESC server as well).
* **Node.js**
* Subsystem
* System:
* AngularJS 2.0:
* Bootstrap:
* SocketIO
* ExpressJS
* MySQL
* ActiveMQ
* Comma Separated Value (CSV):
* Keyhole Markup Language (KML):
* Carrier