

Team Noname:

Carlos Cardenas
 Jonathan Morgan
 Josue Lozano
 Jeremy Miles
 Jeff Caldwell

Communication Medium:

Github Projects

BARRIOS

Ginger Kerrick
 Devin Vyain

Project Proposal

Our team proposes to create an interactive UI (user interface) capable of presenting, evaluating, and predicting the following fields: consumable usage rates, resupply logistics limitations, the accuracy of predictions, resupply schedules, incoming and outgoing crew members schedules, minimum supply threshold and historical actuals. The interface will be derived from a forecast time-series model that will provide insights on the optimum times to resupply, and help to prepare for unanticipated problems that may occur.

Based on various inputs, we plan to create a graph-like interface that presents data for set periods and changes over time. Once the program has data input for said time, based on historical actuals, our team will develop formulas to predict the projected ISS storage supply for a given time period. The UI will present the target supply threshold, the minimum supply threshold, the predicted actual threshold in total, and various consumables. Users will be able to adjust launch dates and shipment payloads and see how this affects the ISS consumable supply over time.

These features together will allow users to view the estimated amount of time ISS crew can survive without a resupply along with any unexpected consumable shortages and make adjustments using a simple input field.

Features and Functions

Predictive Model**Jonathan Morgan**

- Import data from server
- Display data in categories based on date and component
- Implement the “prophet” algorithm to hypothesize about future conditions based on historical data
- Analyze the data to show daily, monthly and yearly patterns
- Predict optimal or conflicting times in the ISS resupply schedule

Refinement System**Carlos Cardenas**

- Allow client to plug variables into the predictive model
- Build an equation to give a ‘weight’ to variables based on previous data
- See in real time the projected outcome of adding variables
- A rigid “bad or good” projected outcome that they can make decisions from
- Real time suggestions such as “too much weight” or “not enough granola bars”

REST API Server**Jeremy Miles**

- Receive/respond to HTTP requests from browser client
- React to requests via controllers, that then interact with modeling and analytics algorithms
- Persist/retrieve datasets and, possibly, stored modeling results to database
- Serialize raw data from database and send JSON response via HTTP
- Parse JSON request data from client

Potential Loss Evaluations**Josue Lozano**

- Highlighting the potential loss will allow users to make informed decisions to mitigate risks
- Using time-series model to forecast potential losses
- Based on potential loss percentages, we can utilize the time-series model to simulate various supply chain scenarios
- Display the estimated percentage chance of experiencing losses over the upcoming year
- Display the potential losses as a result of supply delays

Web Application UI**Jeff Caldwell**

- Graphical “Timeline” Interface - based on the “Baseline Flight Plan”.
- Ability to add or remove shipments graphically
- Ability to adjust shipment times, payloads, and crew arrivals or departures
- Ability to scroll through flight plan timeline at varying levels of detail
- Visualization of predicted supply levels based on current levels and anticipated shipments, and payloads