NASA Path: Phase 3

Software Engineering Project   SWEN 670 9040

F. Cenal, J. Dostal, C. Farley, L. Powell, D. Varma and T. Weldesemayat

Sponsor: Extravehicular Activity (EVA), Johnson Space Center, NASA
Liaison: Daren Welsh, EVA Instructor & Flight Controller | darenwelsh@gmail.com

Overview

- Accomplishments with Live Demo
- Lessons Learned

A Proving Ground in Space (NASA, 2017).

ACCOMPLISHMENTS & LIVE DEMO

NASA PATH PHASE 3

Accomplishments

Add Distances Between Each Handrail Pair In Each Path

Initial Problem

- The Path Results tab provides the route information for each of the three paths and provides the total path distance in inches with two decimal place.
- Distances between the path was not displayed
- The input for wingspan is as precise as an integer of inches whereas the output was two decimal place.

Stakeholder Request

- Stakeholder desired to display the handrail distances.
- For any set of n handrails shall display n-1 distances
- Stakeholder desired the output for the total path distance and handrail distance shall be one decimal place at most.

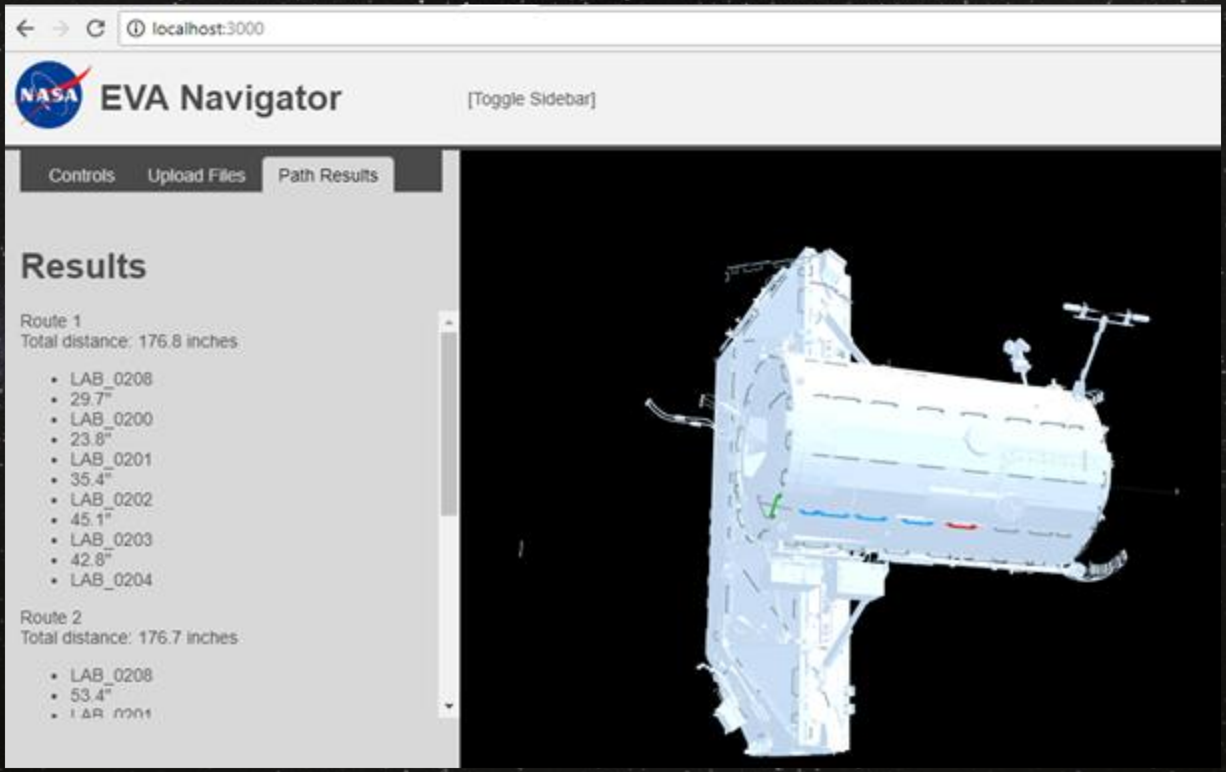
Solution

- Distance between each handrail pairs in each path accounting from the starting point to the end of the handrails are provided.
- The distances are calculated and displayed in one decimal precision.
- The changes are made to EVA Navigator project to accomplish the addition of distance between handrails in the Container.js and App.java files.

Accomplishments

Continued…

The screenshot display the added distances between the handrail pair in each path and the distances shown are on one decimal precision.



Accomplishments

Display tether routing

Initial Problem

- The model does not display tether routing.

Stakeholder Request

- Stakeholder desired to display the safety tether routing in the model.

Requirement Analysis

- The project shall show how astronauts tether routes from the Airlock to them, as they translate around.
- The tether shall be represented as a line starting at the Airlock and following the route along the ISS surface to the astronaut, based on how they translated out to their current position. So that the user can determine a route from point A to point B and can predict where their tether will drag and if it contacts anything hazardous using this application.

Accomplishments

Continued…

What are tether

- A tether is a cord or cable that restrains any hardware or person, to prevent it from "floating" away.
- There are two basic types: One is for hardware and the other is for people. The tethers used to keep people restrained have more safety features.
- Each astronaut carries with them a reel with 85 feet of coiled cable. One end is connected to their space suit. The other end is connected just outside the EV hatch, on the Airlock. As they translate away from the Airlock, the reel tends out more and more cable. If they need to translate more than 85 feet, they bring a second tether and transfer onto it.
- Typically, the tether will naturally tend along the structure as they translate out from the Airlock. But in some cases, it might span a valley if the astronauts don't proactively pin it down to structure (called a "fairlead").
- At times, their tether may drape across hardware that is sensitive to contact or the hardware may articulate during the EVA. In these scenarios, the astronaut must proactively route their tether around such hazards to avoid damaging the hardware or the tether.

Accomplishments

Continued…

Impediments

- Current model does not include all of ISS.
- This project shall only consider the safety tethers
- The current model does not display the Airlock position to calculate the route from airlock to the astronaut's position.
- The current model does not display the Airlock position to display the tether route from airlock to the astronaut.

Accomplishments

Add indications of when axial direction and plane changes (e.g. from port to zenith or from face 1 to face 2)

Initial Problem

- The model does not indicate when axial direction and plane changes.

Stakeholder Request

- Stakeholder desired to add indications of when axial direction and plane changes.

Requirement Analysis

- The project shall not consider the external references during EVA that is the z axis.
- The application shall identify when an astronaut changes direction during translation. Thus, if the astronauts are set on a path that will have them move only along the y axis, along a single face of the truss, by implementing this requirement it shall be very simple to determine the direction during translation.
- But if astronauts are to go part way along the y axis and then turn so they can translate nadir along the z axis, then they must change their momentum.
- The changes in direction shall be flagged or highlighted.

Accomplishments

Continued….

- When an astronaut translates over the edges between these faces, they must rotate their suit. This also requires redirection of their momentum. These types of movements shall be displayed in the model(UI) and there shall be visual indication for the change in momentum.
- The term "zenith" is used for going in the negative direction on the z axis and "Nadir" is used for going in the positive z direction. Zenith is generally away from Earth and nadir is toward Earth.

Impediments

- Current model does not include all of ISS required to implement this feature.
- The current model does not have the movement controls to allow for 3-axis movement and 3-axis rotation for implementing this feature.

Accomplishments

Camera Orientation using OrbitControls

Initial Problem

- Rotation of the ISS model is limited on x- and y- axes
- Cannot orient model by switching x- and y- axes of trust

Stakeholder Request

- Determine if constraints exists preventing full orientation/rotation of model

Proposed Solution

- Project limited by using OrbitControls
  - Locks the y-axis and prevents tilting
- Suggested using TrackballControls

Accomplishments

Camera Orientation using OrbitControls

OrbitControls example

https://threejs.org/examples/misc_controls_orbit.html

http://jsfiddle.net/Stemkoski/ddbTy/

TrackballControls examples

https://threejs.org/examples/misc_controls_trackball.html

https://benchung5.github.io/trackball-controls-three-js/

Accomplishments

Camera Orientation using OrbitControls

SHOW DEMO

Accomplishments

Handrail and Routes Recoloring

Initial Problem

- Start and end handrails not obvious for end users
  - Blue for starting handrail
  - Purple for ending handrail
- Users unable to discern the start and end handrails

Stakeholder Request

- Stakeholder desired extra highlighting or halo effect

Proposed Solution

- Make start and end handrails more logically obvious
  - Green for starting handrail
  - Red for ending handrail
- Route color modification due to existing route 2 color being green
- Evaluate existing color choices for fitness with accessibility standards (WCAG 2.1)

Accomplishments

Handrail and Routes Recoloring

SHOW DEMO

Accomplishments

No Logging Framework Support

Initial Problem

- All logging of EVA Navigator was directed only to console
- No persistence of stacktrace
- Once console is closed, program output and history is gone
- Either remember to copy console output or replicate error again for stacktrace

Accomplishments

No Logging Framework Support

SHOW DEMO

Accomplishments

Handrail Selection using 3D Model

Initial Problem

- Option to select start and end handrails only via drop-downs
- Users may not know handrail name but know location on model

Stakeholder Request

- Stakeholder desired allowing users to click a handrail in the 3D model to select start and end points

Proposed Solution

- Raycast from mouse point to handrail mesh to select handrail object
- If no selection is chosen in start handrail dropdown and user clicks handrail, populate start handrail dropdown
- If selection is chosen in start handrail dropdown, no selection is chosen for end handrail dropdown and user clicks handrail, populate end handrail dropdown
- Populating dropdowns should fire React onChange event to change handrail color

Accomplishments

Handrail Selection using 3D Model

SHOW DEMO

Accomplishments

User Interface and path calculation update for 2 crew members

Current system:

- The system allows for only one user to enter handrail selections and search for the shortest path
- The system provides three route choices for only one user

Stakeholder Request:

- Add UI on the current system to allow for 2 crew members to select handrails and find the shortest routes
- Update route calculation to deconflict routes accounting for 2 crew members

Solution Approach:

1. ***User Interface Updates:*** update the structures of the different controls under the left navigation to accommodate for two user selection.
    - Add two tabs under the Controls tab for Crew Member 1 and 2 respectively
    - Adjust the existing controls panel to display under Crew Member 1 tab
    - Add the start and end handrail dropdown selections for Crew Member 2
    - Add UI to select the wingspan for Crew Member 2
    - Add the three visible routes to select and deselect from for Crew Member 2
    - Add the Go and Reset buttons for Crew Member 2

Accomplishments

Continued ...

2. **Update Route calculation:**
    - Update the current route calculation to account for two crew members
    - Save the route results of crew member 1 in a state
    - Calculate routes for crew member 2
    - Compare route results of crew member 2 with the route results obtained from crew member 1
    - Discard and recalculate path results for crew member 2 if the results are on similar paths as that of crew member 1

Impediments:

The User interface update of this requirement is completed while the route calculation accounting two crew members is still in progress due to the following difficulties the team faced:

- Needed familiarization and deep understanding of the front end programing language already in use (ReactJs)
- Needed familiarization as the choices of technologies are pretty new
- Lack of time to experiment
- Time constraint as submissions shift focus

Accomplishments

User Interface and path calculation update for 2 crew members

SHOW DEMO

NASA's Science Aircraft Flies Over Thomas Fire in California, (NASA, 2017).

LESSONS LEARNED

NASA PATH PHASE 3

Lessons Learned

Communication

- Group meetings
  - Ensure all meetings are preannounced
  - Fit to group members' schedules
- Stakeholder contact
  - Continue to keep stakeholder updated and engaged
  - Goal is to build satisfaction with the product
- Collaborative tools
  - Standardize chosen tools early during project team forming
  - Helped to gain adoption and familiarization

Lessons Learned

Project Management

- Team roles using Kanban's roleless strategy
  - Removed need to create silo-like project roles
  - Allowed project team to focus on priority concerns
  - Fostered groupthink activities to solve problems

Lessons Learned

Schedule Estimation

- External priorities and scheduling
  - Keep project expectations small and manageable
  - Remember external priorities when assigning tasks
  - Keep project team productive by decomposing broad tasks
- Familiarity training
  - Utilize project team members' strengths and occupational knowledge
  - Fill knowledge gaps through collaboration versus creating knowledge silos

Lessons Learned

Design and Implementation

- High-level backlog items
  - Kanban adoption facilitated stakeholder dialogue for task decomposition
- Volume of backlog items
  - It is imperative to settle stakeholder expectations when presented 22 items
  - Focus on priority project concerns and tasks
  - Avoid work which do not return value to the stakeholder
- Use of development tools
  - Standardization is important to retain use
  - Lock in agreed tools early

Lessons Learned

Closing

- Milestone wrap-up
  - Identify all artifacts for milestone delivery early
  - Begin work preemptively
  - Keep all artifacts up-to-date

Summary

- Accomplishments with Live Demo
- Lessons Learned

QUESTIONS?