
Feature Documentation: Add Distance Between Handrails

for

NASA EVA Path Phase 3

Version 1.1

Developer

Deepali Varma

Product Owner

Daren Welsh

Test Team

Cameron Farley

Lincoln Powell

Tenadam Weldesemayat

University of Maryland University College
SWEN 670 9040 (2185)
Software Engineering Project
Dr. Michael Brown

July 22, 2018

Table of Contents

i. Revision History	2
1. Introduction	3
1.1 Background	3
1.2 Intent	3
1.3 Agreed Change(s)	4
2. Development	4
2.1 Code Additions or Modifications	4
3. Functional Testing	7
3.1 Proposed Functional Test Case	7
3.1.1 Test Case 1: Distance is added between each handrail pair for each path and route	7
3.2 Assumptions and Constraints	7
3.2.1 Assumptions	7
3.2.2 Constraints	7
3.3 Findings	8
References	8

i. Revision History

Revision	Author	Date	Description
1.0	Tenadam Weldesemayat	7/22/18	Initial document.
1.1	Lincoln Powell	7/22/18	Reviewed document.

1. Introduction

1.1 Background

The NASA EVA Navigator web application delivered from Phase 2 completed tailored backlog items designed by Daren to meet existing objectives for the product. One of the desired functionality remained in the backlog was the addition of distance between each handrail pair in each path. The application delivered from Phase 2 showed path results for three routes with the name of each of the consecutive handrails from start to finish as shown below:

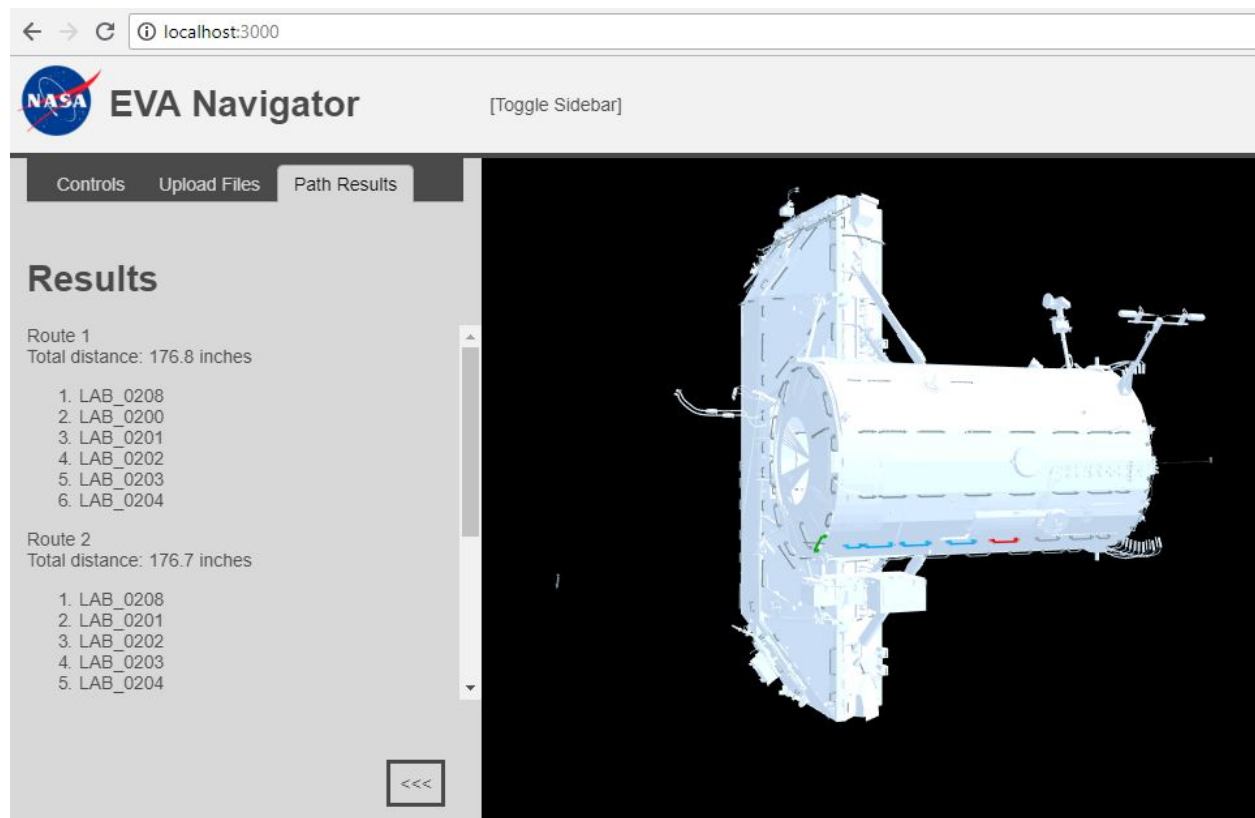


Figure 1. Path Results tab showing start to end handrails for each route from Phase 2.

1.2 Intent

Daren's intent for this change is to provide distance between each handrail pairs in each path accounting from the starting point to the end of the handrails of the EVA Navigator application. Toward this goal, he desired to provide better information in regards to each move a space walker will make in the International Space Station model.

1.3 Agreed Change(s)

The addition of distances between each handrail pair in each path in the path results section of the application.

2. Development

2.1 Code Additions or Modifications

The changes made to EVA Navigator project to accomplish the addition of distance between handrails were mainly to the Container.js and App.java files. The screenshots below show the code changes made to the project. The pink highlights demonstrate the lines removed and the light green once show additions. (see Figure 2 and 3 below).

Add Distance Between Handrails Browse files

Added distances between each handrail pair in each path
 Modified the file to truncate the distance and distancetotal to one decimal
 Modified file to change bullet numbers to dots

master (#4)

deepalivarma committed 9 days ago 1 parent 872dc87 commit 3fb9037b5f27c2ff61d0c00c68b0bfb5944b95d

Showing 2 changed files with 22 additions and 7 deletions. Unified Split

6 src/components/Container/Container.js View

```

@@ -8,6 +8,8 @@
8      // Added in-line documentation
9      // April 2018 - George -
10     // Modified file to include distance total in the results tab in the sidebar from App.java
11     +// July 2018 - Deepali -
12     +// Modified file to change bullet numbers to dots
13
14     import React from 'react';
15     import Renderer from 'components/Renderer/Renderer';
16
17     @@ -195,11 +197,11 @@ export default class Container extends React.Component {
195         <div key={routeI}>
196             <div>Route {routeI + 1}</div>
197             <div>Total distance: {route.distancetotal} inches</div>
198             - <ol>
199             + <ul>
200                 {route.nodes.map((node, nodeI) =>
201                     <li key={nodeI}>{node}</li>
202                 )}
203             - </ol>
204             + </ul>
205         </div>
206     }
207     </div>
  
```

Figure 2. Container.js file changes to add distance between handrails.

```

23  server/src/main/java/com/nasa/App.java
@@ -14,6 +14,12 @@
14 14  import fi.iki.elonen.NanoHTTPD;
15 15  // April 2018 - George - Added code to Add distances between each handrail pair in each path here and in Container.js
16 16  // NanoHTTPD > v3.0.0
17 17  +
18 18  +//July 2018 - Deepali Varma-
19 19  +//Added distances between each handrail pair in each path
20 20  +//Modified the file to truncate the distance and distancetotal to one decimal
21 21  +
22 22  +
17 23  //import org.nanohttpd.NanoHTTPD;
18 24
19 25  /*
@@ -128,22 +134,29 @@ public Response serve(IHTTPSession session) {
128 134      // Loop through each node to display handrail and calculate distance
129 135      for (Node node : nodes) {
130 136          String nodeId = node.getNodeId();
131 137          - nodeIds.add(nodeId);
132 138
133 139      // For each handrail after the first, calculate distance between previous and current handrail
134 140      // For each handrail distance tally up the total distance between the first handrail and the last
135 141      - // Truncate the distancetotal to two decimals
136 142      + // Truncate the distancetotal to one decimals
137 143      try {
138 144          if (nodeLast != null) {
139 145              distance = node.node_distance_formula(node, nodeLast);
140 146              - distance = ((double) Math.round(distance * 100)) / 100;
141 147              + distance = ((double) Math.round(distance * 10)) / 10;
142 148              distancetotal += distance;
143 149              - distancetotal = (double) Math.round(distancetotal * 100) / 100;
144 150              + distancetotal = (double) Math.round(distancetotal * 10) / 10;
145 151          }
146 152      } catch (Exception ex){
147 153          System.out.println("Error calculating handrail distance.");
148 154      }
149 155      -
150 156      +
151 157      //Added distances between each handrail pair in each path
152 158      +
153 159      +
154 160      + if(distance!=0.0)
155 161      + {
156 162      +     nodeIds.add(Double.toString(distance)+"\n");
157 163      + }
158 164      + nodeIds.add(nodeId);
159 165      +
160 166
161 167      // Output handrail name and distance from last.
162 168      System.out.println(nodeId + " [" + distance + " in.]");
163 169      nodeLast = node;
164 170

```

Figure 3. App.java file changes to add distance between handrails.

Provided are screenshots, see Figures 4 and 5 on page 6, which show the Path Results tab on the left with the previous (without distance between handrails) and updated (distance added between handrails) routes output windows respectively.

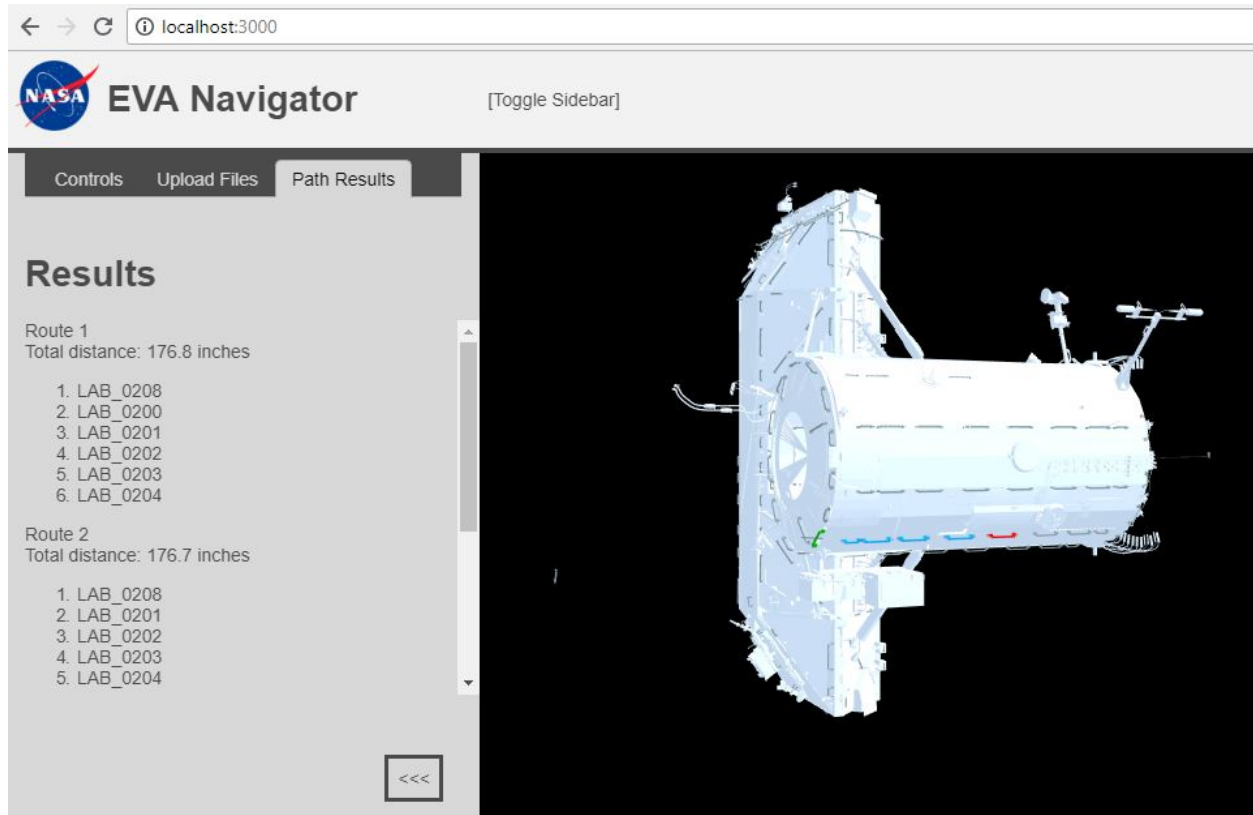


Figure 4. Path Results tab without distance between each path from Phase 2.

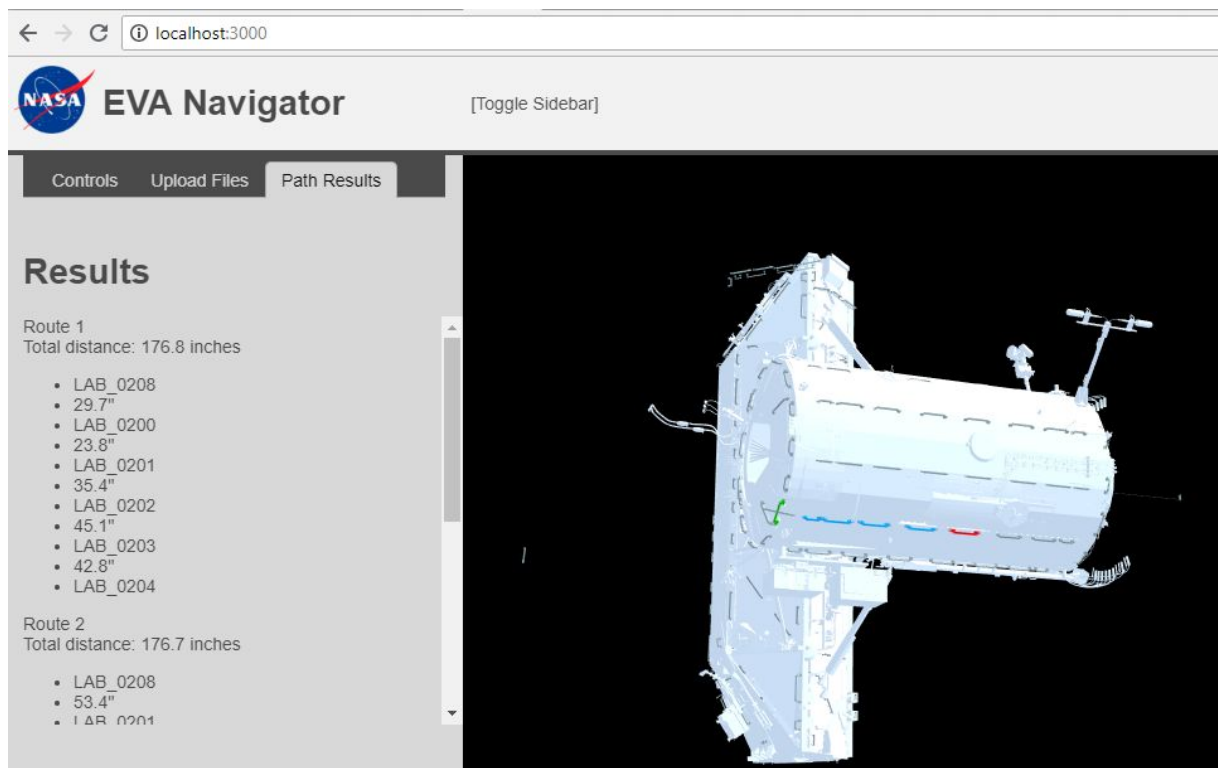


Figure 5. Path Results tab showing distance between each handrail in each path.

3. Functional Testing

3.1 Proposed Functional Test Case

3.1.1 Test Case 1: Distance is added between each handrail pair for each path and route

Description: The user should see distance between each handrail pairs under the path results for each route one when a starting and ending handrails are selected selected from the drop-down menu the “Go” button is clicked.

Requirements:

1. The route results of each route show distance between each handrail pairs.

Prerequisites:

1. Start handrail drop-down menu must work as intended.
2. End handrail drop-down menu must work as intended.
3. The routes are calculated when the “Go” button is clicked.

Steps:

1. Load “EVA Navigator” web application by navigating browser to <http://127.0.0.1:3000> or <http://localhost:3000>.
2. Select LAB_0208 as the starting handrail.
3. Select LAB_0204 as the ending handrail.
4. Select the “Go” button.
5. Click on the Path Results.
6. Check that the results for each route show distance in between the listings of each handrail.

Input: Click on the starting handrail drop-down menu and select the “LAB_0208” handrail. Click on the ending handrail drop-down menu and select the “LAB_0204” handrail. Click on the “Go” button. Click on the Path Results

Expected Output: Distance between each handrail path the is visible for each route in the Path Results tab.

Assumptions: None.

3.2 Assumptions and Constraints

3.2.1 Assumptions

It is assumed that the EVA Navigator web application has been setup and launched correctly, following the User_Manual.docx, section 3, Software Installation based on your operating system.

3.2.2 Constraints

None.

3.3 Findings

All functional tests worked as expected, resulting in the desired behavior.

References

None.