TEST ID	Catagory	Method	Description	Test Cases	Steps	Expected Results/Behavior	Actual Results	Pass/Fail
1	Search Bar	handleSearchTextChange()	handles changes to the search input ar based on the new search text.	"ISS" "Cap" "ZZZ"	Type test case in search bar	"ISS (Zarya)" "CAPE-3" ""	"ISS (Zarya)" "CAPE-3" ""	Pass Pass Pass
2		handleSuggestionClick()	handles a suggestion click event by up setting the search text to the selected s	"ISS"	1. Type in test case in search bar 2. Click A Suggestion	console.log(Satellite != null) => true console.log(SearchText) => ISS (Zarya)	True ISS (Zarya)	Pass Pass
3		handleClickAway()	This function clears the suggestions an away and did not select a satellite.	"ISS"	1. Type in test case in search bar 2. Click away without selecting a satellite	console.log(suggestions) => undefined	undefined	Pass
4	Rotator Connector	handleConnect()	attempts to connect to a submitted IP a the rotator object if successful, and dis messages accordingly.	"slakdgo" "102.2.4.2.4" "123.123.123.123" "192.168.50.100"	1. Click "Connect a Rotator" 2. setup server to 192.168.50.100 3. enter test case 4. Press "Connect"	"Invalid IP" "Invalid IP" "Timed Out" "Connected Sucessfully"	"Invalid IP" "Invalid IP" "Timed Out" "Connected Sucessfully"	Pass Pass Pass Pass
5		handleCancel()	clears the IP address input field, closes any errors.	N/A	1. Click "Connect a Rotator" 2. Press "Cancel"	console.log(openDialog) => false console.log(error) => ""	false ""	Pass Pass
6		handleClick()	opens a dialog box for connecting to a connected rotator, but prevents disconr currently tracking.	isTracking= False connected = False isTracking= False connected = True isTracking= True connected = True	1. reach test case state 2. press "Connect/Disconnect"	Dialog Box Opens Disconnects Rotator Nothing	Dialog Box Opens Disconnects Rotator Nothing	Pass Pass Pass
7		handleInputChange()	This function sets the IP address state change event.	"123.123.123"	1. Press "Connect" 2. type in test case into field	console.log(ipAddress) => "123.123.123"	"123.123.123"	Pass
8	Mercator Map	handleToggleMapSettings()	opens or closes a map settings popup to the current event target.	N/A	1. Mount application 2. Click Map Setting Icon (wrench) 3. Click the Icon again to Close	Map Settings pops up to the left of the I Popup closes	Map Settings pops up to the left of the Popup closes	Pass Pass
9	MapSettings	handleClose()	closes the map settings popup by settir	N/A				
10		handleShowGridChange()	changes the visibility of a grid layer and showGrid.	N/A	1. Mount application 2. Click Map Setting Icon (wrench) 3. Click "Grid" to show 4. Click "Grid" to Close	grid shows on switch right grid dissapears on switch left	grid shows on switch grid dissapears on switch	Pass Pass
11		handleShowDayNightChange()	toggles the visibility of the day/night ter corresponding state variable.	N/A	1. Mount application 2. Click Map Setting Icon (wrench) 3. Click "DayNight" to show 4. Click "DayNight" to Close	terminator shows on switch right terminator dissapears on switch left	terminator shows on switch terminator dissapears on switch	Pass Pass
12		handleShowGroundTrackChan	toggles the visibility of the ground track graphic is not null.	N/A	1. Mount application 2. Type in known satellite in search bar 3. Click on suggestion 2. Click Map Setting Icon (wrench) 3. Click "Ground Track" to show 4. Click "Grid" to Close	Ground track shows on switch right Ground track dissapears on switch left	Ground track shows on switch right Ground track dissapears on switch lef	Pass Pass
13	Map Helper	getGround()	generates a coordinate path to compute satellite based on given coordinates.	Longitude =100 Latitude = 45 Longitude = 500 Latitude = -2993	1. Run MapHelper.test.js	Coordinates of polyline that intersect with Satellite "Out of Range" Error	Coordinates of polyline that intersect with Satellite "Out of Range" Error	Pass Pass
14		getGrid()	generates a grid of coordinates with sp and longitude.	N/A		Coordinates of Latitude/Longitude at 15 degree interval	Coordinates of Latitude/Longitude at degree interval	Pass
15		getDayNight()	calculates and returns the coordinates separating day and night) on Earth at t	N/A		Coordinates of day/night terminator retu	Coordinates of day/night terminator re	Pass
16		renderGrid()	renders a grid on a map viewport using returns the grid graphic.	N/A		Grid shows on map	Grid shows on map	Pass
17		renderDayNight()	creates and adds a day/night graphic to	N/A		Terminator shows on map	Terminator shows on map	Pass
18		renderGroundTrack()	renders a ground track graphic on an E longitude and latitude coordinates.	Longitude =100 Latitude = 45 Longitude = 500 Latitude = -2993		Polyline rendered intersecting (100,45) "Out of Range" Error	Polyline rendered intersecting (100,45 "Out of Range" Error	Pass Pass
19		renderObserver()	renders an observer graphic on an Esri longitude and latitude coordinates, and	Longitude =100 Latitude = 45 Longitude = 500 Latitude = -2993		Marker rendered at (100,45) "Out of Range" Error	Marker rendered at (100,45) "Out of Range" Error	Pass Pass
20		renderSatellite()	enders a satellite on a map view using creating a graphic with a specific symt view's graphics layer.	Longitude =100 Latitude = 45 Longitude = 500 Latitude = -2993		Satellite Icon rendered at (100,45) "Out of Range" Error	Satellite Icon rendered at (100,45) "Out of Range" Error	Pass Pass
21	Helper	getSatellite()	retrieves orbital data from Celestrak for name, constructs a satellite object with name, time, longitude, latitude, altitude, orbital data (OMM), and returns the sat	Name = ISS (Zarya) Name = aksfhsdf	1. Run Helper.test.js	console.log(Satellite != null) => true console.log(Satellite != null) => false	true false	Pass Pass
22		getRotator()	retrieves rotator information from a give constructs a rotator object with relevar IP address, call sign, model, local time look angles (initialized as null), and trac rotator object; if the IP address is inval appropriate error messages are return	"slakdgo" "102.2.4.2.4" "123.123.123.123" "192.168.50.100"		console.log(Rotator != null) => false console.log(Rotator != null) => false console.log(Rotator != null) => false console.log(Rotator != null) => true	false false false true	Pass Pass Pass Pass
23		fetchOrbitalData()	fetches orbital data for a satellite from specified format (either TLE or JSON), the satellite name and format, sends a returns the fetched data as either TLE depending on the format requested; an the fetch are logged to the console.	Name = ISS (Zarya) Format = "JSON" Name = aksfhsdf Fortmat = sdfgs		OBJECT_NAME: "ISS (ZARYA)" OBJECT_ID: "1998-067A" EPOCH: "2023-04-25T12:46:22.350432" MEAN_MOTION: 15.50274552 ECCENTRICITY: 0.0005628 INCLINATION: 51.6404 RA_OF_ASC_NODE: 229.4846 ARG_OF_PERICENTER: 240.7961 MEAN_ANOMALY: 207.2997 EPHEMERIS_TYPE: 0 CLASSIFICATION_TYPE: "U" NORAD_CAT_ID: 25544 ELEMENT_SET_NO: 999 REV_AT_EPOCH: 39362 BSTAR: 0.00048547 MEAN_MOTION_DOT: 0.00027629 MEAN_MOTION_DDOT: 0 Error	OBJECT_NAME: "ISS (ZARYA)" OBJECT_ID: "1998-067A" EPOCH: "2023-04-25T12:46:22.350432" MEAN_MOTION: 15.50274552 ECCENTRICITY: 0.0005628 INCLINATION: 51.6404 RA_OF_ASC_NODE: 229.4846 ARG_OF_PERICENTER: 240.7961 MEAN_ANOMALY: 207.2997 EPHEMERIS_TYPE: 0 CLASSIFICATION_TYPE: "U" NORAD_CAT_ID: 25544 ELEMENT_SET_NO: 999 REV_AT_EPOCH: 39362 BSTAR: 0.00048547 MEAN_MOTION_DOT: 0.00027629 MEAN_MOTION_DDOT: 0 Error	Pass Pass
24		fetchJson()	fetches JSON data from a specified IP request type, with an optional payload, data; it includes a timeout mechanism longer than 10 seconds, and throws ar invalid.	"slakdgo" "102.2.4.2.4" "123.123.123.123" "192.168.50.100"		"Invalid IP Error" "Invalid IP Error" "IP Timeout Error" Name: "David's Rotator" IP: "192.168.50.100" Call_Sign: "W5QZ" Model: "Yaesu G-5500" Time: "21:58:18" Longitude: "30.6280" Latitude: "-96.3344" ObserverGd: Object Look_Angles: null isTracking: false Template: Object	"Invalid IP Error" "Invalid IP Error" "IP Timeout Error" Name: "David's Rotator" IP: "192.168.50.100" Call_Sign: "W5QZ" Model: "Yaesu G-5500" Time: "21:58:18" Longitude: "30.6280" Latitude: "-96.3344" ObserverGd: Object Look_Angles: null isTracking: false Template: Object	Pass Pass Pass Pass
25		getPositionGd()	calculates the geodetic coordinates (latitude, longitude, and altitude) of a sa surface based on its Earth-Centered In the current date and time.	x: 3049.86548, y: -472.53814, z: -6332.4052		longitude: -0.84379 latitude: 0.23418 height: 35856.7679	longitude: -0.84379 latitude: 0.23418 height: 35856.7679	Pass
26		getSatRec()	takes a Two-Line Element (TLE) string the line 1 and line 2 TLE data, converts object using the twoline2satrec library,	RIGIDSPHERE 2 (LCS 4) 05398U 71067E 23115.5114748 0001221 00000+0 41465-3 0 99 5398 87.6149 296.8478 006482 852 284.6547 14.347658967089		{error: 0, satnum: "05398", epochyr: 23, epochdays: 115.51147481, ndot: 0.00001221…}	{error: 0, satnum: "05398", epochyr: 23, epochdays: 115.51147481, ndot: 0.00001221…}	Pass
27		getLookAngles()	calculates and returns the azimuth, ele of a satellite with respect to a given obs (observerGd), using the satellite record between Earth-centered inertial (ECI) a coordinate systems.	{error: 0, satnum: "05398", epochyr: 23, epochdays: 115.51147481, ndot: 0.00001221..		longitude: 2.7705218839347348, latitude: -0.03913181951360097, height: 768.8038068041697}	longitude: 2.7705218839347348, latitude: -0.03913181951360097, height: 768.8038068041697}	Pass
28		isValid()	takes an input string representing an IP and uses a regular expression to valida returning a boolean value indicating its	"slakdgo" "102.2.4.2.4" "123.123.123.123" "192.168.50.100"		false false true true	false false true true	Pass Pass Pass Pass

29		getLocalTime()	calculates and returns the local time at (in degrees), by adjusting the current difference from Greenwich Mean Time object representing the local time.	N/A		Time: "22:04:16"	Time: "22:04:16"	Pass
30		getTimeUTC()	calculates and returns the current Coor Time (UTC) based on the Greenwich M and the observer's geographic coordina the current date and time to account fo Date object representing the UTC time.	N/A		12:52:55	12:52:55	Pass
31	Dropdown Menu	handleToggleMenu()	handles a menu toggle event and displ the control menu based on whether a been selected.	Satellite = null Rotator = null Satellite != null Rotator = null Satellite = null Rotator != null Satellite != null Rotator != null	1. Reach the test case state by connecting to rotator/selecting satellite 2. Click Control Menu button in the top left corner (Slider Icon)	" Please select a Satellite and connect a rotator to begin tracking" " Please select a Satellite and connect a rotator to begin tracking" " Please select a Satellite and connect a rotator to begin tracking" Control Menu Opens	" Please select a Satellite and connect a rotator to begin tracking" " Please select a Satellite and connect a rotator to begin tracking" " Please select a Satellite and connect a rotator to begin tracking" Control Menu Opens	Pass Pass Pass Pass
32		handleCloseMenu()	handles the event to close a control me and error messages related to azimuth	Azimuth = 120 Elevation = 10 Azimuth = 1210 Elevation = 110	1. Connect a rotator and choose a satellite 2. Click Control Menu button in the top left corner (Slider Icon) 3. enter test case into text field 4. click away from control menu	Azimuth = 120 Elevation = 10 Azimuth = "" Elevation = ""	Azimuth = 120 Elevation = 10 Azimuth = "" Elevation = ""	Pass Pass
33		handleCommand()	sends a start or stop tracking command whether the tracking is already on or of success or error message accordingly. of the satellite object or azimuth/elevat whether auto or manual tracking mode	isTracking = False command = "Start" isTracking = False command = "Stop" isTracking = False command = "" isTracking = True command = "Start"	1. Connect a rotator and choose a satellite 2. Click Control Menu button in the top left corner (Slider Icon) 3. click "autoTrack" 4. reach test case state by using commands	"Rotator has started auto-tracking" Client Request:: StartAuto "Rotator has stopped tracking" Client Request:: Stop "HTTP/1.1 404 Not Found" No Change	"Rotator has started auto-tracking" Client Request:: StartAuto "Rotator has stopped tracking" Client Request:: Stop "HTTP/1.1 404 Not Found" No Change	Pass Pass Pass Pass
34		handleCloseSnack()	clears the snack message popup and c	N/A	1. Click Control Menu button in the top left corner (Slider Icon)	Snack bar message displays and disap	Snack bar message displays and disa	Pass
35		handleAzimuth()	validates and sets the azimuth input va if it is out of range.	Azimuth = 120 Elevation = 10 Azimuth = 1210 Elevation = 110	1. Connect a rotator and choose a satellite 2. Click Control Menu button in the top left corner (Slider Icon) 3. enter test case	True False	True False	Pass Pass
36		handleElevation()	validates and sets the elevation input v if it is out of range.	Azimuth = 120 Elevation = 10 Azimuth = 1210 Elevation = 11330	1. Connect a rotator and choose a satellite 2. Click Control Menu button in the top left corner (Slider Icon) 3. enter test case	True False	True False	Pass Pass
37		handleToggleAutoTrack()	toggles the state of the tracking control and elevation input values.	isTracking = False Azimuth = 120 Elevation = 10 isTracking = True	1. Connect a rotator and choose a satellite 2. Click Control Menu button in the top left corner (Slider Icon) 3. click Autotrack Switch	AutoTrack Switch Toggles On Azimuth = "" Elevation = "" AutoTrack Switch Toggles Off	AutoTrack Switch Toggles On Azimuth = "" Elevation = "" AutoTrack Switch Toggles Off	Pass Pass