¢Ï ♣\$ Ï Þ 4 3/4 3/4 Ĭ Þ 1 T 4 T × \$ 7 8 3/4 4

This document presents a test plan/test procedure for the CoastsideARC PayPal Dues Renewal software.

Overview

This software consists of 2 major components: The client side component and the server side component.

Client Side Overview

The client side component may be accesses immediately from https://coastsidearc.org/www.coastsidearc.org/PayPalDuesRenewal.php. This component consists of a form that must be filled out by the user. Upon completing the form, the user presses the 'Submit to Paypal' button. The client looks up the full user name based on the user's Callsign, and all of the collected data is bundled into a JSON object that is

transmitted to the server-side component via transmission on PORT 5556.

Server side overview

The server-side component is located at ~/node-test-01/carc-paypal/server/server.js. This is a node.js module that runs continuously on the DreamHost server vps56613. Upon receipt of the JSON packet, the payment components are stripped off, and a data record is formed for the payment transaction, which is inserted into the payment database, and the 'status' field of the transaction is initially marked 'pending'. See https://coastsidearc.org/www.coastsidearc.org/Test_getLastEntries.php to see the record structure of this database.

Testing Approach

The testing approach is to carry out 3 phases of testing. The first phase will test the client-side code. In this phase we will set breakpoints at the point in the code as the Cient is about to transfer control over to the Server. A procedure for doing that is shown below.

Procedure taken to debug the client side code

The following debug steps were taken. They are described here in some detail because I am not that familiar with debugging in the chrome browser.

- 1. Open the Chrome Browser.
- 2. Enter "CoastsideARC.org in the search bar.

- 3. Click on PayPal Dues Renewal on the menu
- 4. Enter the following information on the form
 - a. Callsign: AI6BB
 - b. Repeater Fund: \$0.50
 - c. On the line "Include PayPal Fee?" check the checkbox
- 5. Note that PayPal receives \$0.53 and that Total Charges were \$1.03
- 6. Invoke the Chrome Debugger by hitting 'Ctrl+Shift+I' key combination
- 7. Click the forward pointing triangle in front of '<div class="submit">' in the LH panel.
- 8. Click on the '<button id = "altButton">Submit to PayPal</button> == \$0' line
- 9. In the RH panel, click on the forward pointing triangle in front of 'click'
- 10. Right Click on the text 'button#altButton' and click on 'Reveal in Sources panel'
- 11. Click on the line callsign = jsondict['callsign']; to set a breakpoint
- 12. Click on the 'Submit to PayPal button on the PayPal Dues Renewal page
- 13. Note that the breakpoint is hit, and the message 'Paused on Breakpoint' is shown in the RH panel.
- 14. Click on the forward pointing triangle in front of the text 'Global' in the RH panel.
- 15. Scroll down to the forward pointing triangle in front of the text 'jsondict'
- 16. Note that the following fields are set in the jsondict object:

```
callsign: "AI6BB"
                                           pay_paypal: "yes"
                                           paypalfee: "$
callsigns: "AI6BB"
clubreceives: "$
                  0.50"
                                          pp_total: 1.03
                                          primary: "$ 0.00"
date: <current date & time>
                                           repeater: "$ 0.50"
digipeater: "$ 0.00"
family: "$ 0.00"
                                           subtotal: "$ 0.50"
                                           total: "$ 1.03"
ncallsigns: 1
newmember: "no"
                                           transaction status: "pending"
```

- 17. Click on the line 'insert data into the PPTnxTable(jsondict).then(() => { to set a breakpoint and then hit the 'Resume script Execution 'button' above the RH panel.
- 18. Note that the program jumped to the line (386) indicating Paused on promise rejection: Error: Error fetching FullName [object ProgressEvent] at xhr.onerror = (error) => reject(new Error(`Error fetching FullName: \${error}`));

That rejection will need to be further investigated