JCI Virtual Branch

**Issues Faced**

|  |
| --- |
|  |
| **Document Version / Details : Ver 1.60/16-Jan-09** |
|  |





**Record of Release**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Version No. | Modified By | Reviewed By | Authorized By | Release Date | Modifications Done |
| 1.1 | Sheetal Shinde |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |

Contents

[1.0 Objective 3](#_Toc532571032)

[2.0 Issues Faced 3](#_Toc532571033)

[2.1 CORS: 3](#_Toc532571034)

[2.2 Meta Tag for IE 3](#_Toc532571035)

[2.3 Caching on IE 3](#_Toc532571036)

[2.4 Image Binding From Server 3](#_Toc532571037)

[2.5 Mozilla Issue: 4](#_Toc532571038)

[2.6 Image Caching on Browser: 4](#_Toc532571039)

# Objective

This document mentions the issues faced while developing the Angular 6 based Virtual Branch App

# Issues Faced

## CORS:

Following headers were added on the server side in every webservice response.

* Access-Control-Allow-Origin
* Access-Control-Allow-Credentials
* Access-Control-Expose-Headers
* Access-Control-Max-Age
* Access-Control-Allow-Methods
* Access-Control-Allow-Headers

## Meta Tag for IE

Following meta tag need to be added in index.html to ensure that the app gets loaded only on latest IE versions.

<meta http-equiv="X-UA-Compatible" content="IE=edge">

## Caching on IE

We faced an issue where the webservice response was getting cached on IE. But the same was not happening on Chrome. There was a webservice which was called before login. Once the user logs in the same webservice was called which was then returning the logged in user data. But on IE it was returning the cached response of not logged in user. To fix this following headers were added on the server side:

*Set to expire far in the past.* [*response.setHeader*](http://response.setHeader)*("Expires", "Mon, 23 Aug 1982 12:00:00 GMT");*

*Set standard HTTP/1.1 no-cache headers.* [*response.setHeader*](http://response.setHeader)*("Cache-Control", "no-store, no-cache, must-revalidate");*

*Set IE extended HTTP/1.1 no-cache headers (use addHeader).* [*response.addHeader*](http://response.addHeader)*("Cache-Control", "post-check=0, pre-check=0");*

*Set standard HTTP/1.0 no-cache header.* [*response.setHeader*](http://response.setHeader)*("Pragma", "no-cache");*

## Image Binding From Server

We were facing an issue where when we bind an image url from server and if the image does not exist then the request used to time out after a long time. Thus if there were multiple images in a page then there will be multiple requests pending. On Chrome this creates an issue, as there can be a fixed amount of requests (~6) in the pending state. So if the count goes above 6 then unless the request count reaches 6 or less the next request will remain in stalled state, thus delaying UI response.

Since, angular universal was used in the project, the node was getting 404 error for the image but it was not sent to the browser as there was not registered component to handle this.

To fix the same we added a dummy component in the routing which had the image path (generic), since all the images were placed in a single folder. This avoided the wait period and stalling of requests as well.

## Mozilla Issue:

Keypress event was not working on Mozilla. So, instead of using 'keypress', use 'keyup' and 'keydown' events for input value.

## Image Caching on Browser:

Everytime a new build was deployed for any css changes or the product images were changed on the server, the users had to clear the browser cache to see the changes. In order to avoid this, a meta tag “build time” was added to the index.html file.

**<meta http-equiv="build-time" content="Tue Nov 27 2018 05:30:41 GMT-0530">**

This tag will tell the browser to fetch fresh files from the server and not to use the cached version.