



**CoreLogic**<sup>®</sup>

# Trestle™ 4.24 Release Notes

**EXTERNAL**

**March 25, 2020**

## Contents

Dashboard Updates .....	2
API Updates .....	2

©2018 CoreLogic. CORELOGIC, the CoreLogic logo, and Matrix are trademarks of CoreLogic, Inc. and/or its subsidiaries. All other trademarks are the property of their respective owners.

This document may not be furnished, in whole or in part, in any form to any other person or entity without the written consent of CoreLogic. The contents of this document are deemed accurate but not guaranteed and are not intended for contractual purposes.

In Canada, the trademarks MLS®, Multiple Listing Services® and the associated logos are owned by The Canadian Real Estate Association (CREA) and identify the quality of services provided by real estate professionals who are members of CREA.

## Dashboard Updates

### **TRES-4770 – When clicking preview contract, the button does nothing – Update additional signers on page**

Previously, the fields on the contract signers' interface did not allow commas and hyphens in the signers field. Submissions were being silently rejected by the system, with no feedback to the user. We've now made these fields support all relevant special characters and display any error messages generated by the submission.

## API Updates

### **TAPI-1147 – Add HTTP response headers to allow cross-domain requests**

This change enables API consumers to build web pages that call the Trestle API directly. This is the final evolution of the access control to allow truly real-time API access. A data consumer now never needs to touch the data at all. They can now construct web pages that query Trestle directly from the consumer's browser window.

For security purposes, the OAuth2 token endpoint is not available from 3<sup>rd</sup> party clients. The workflow should be:

1. Authenticate via code on the web server and store the token in a stand-alone .js file. The token is valid for 8 hours, so re-run this auth periodically.
2. On the web browser code, bring in the auth token as an included .js file.
3. Make OData queries directly from the JavaScript on the page, including using Media.MediaURL values as image src'es, etc

This approach carries some security risk, as it necessarily exposes your OAuth2 authentication token in the client code. For at least a bit of security, please include your OAuth2 token as an external javascript (i.e. `<script src="">`) file. *Please do not include your auth token visibly in the HTML source.*

We'll be providing code to demonstrate this feature on docs-trestle.corelogic.com over the next few weeks.

**Content Patch #133 is included in this release; details in a separate document.**