AutoLocalize 🡪 R-Loc New Scope and Requirements

# Scope

* The project supports clients who are not already using the Singularity or Dynamic UI frameworks. These frameworks already handle the localization workflows for the websites using them.

# Front-end: R-Loc Script

* The R-Loc script is an app that runs as a script in the Command Line Interface (CLI).
* The script should scan through all HTML, JS, and other files that contain hard-coded translatable strings and extract them into JSON (or any other suitable format for the project) files for TMS to pick-up within the website's source code on any dedicated repository (Git or others).
* The script should extract all translatable strings from all the UI files and populate them in one JSON file per locale for the entire website. The script must also handle any number of locales requested for translation. For example, if there are 3 locales requested for translation, the script must make 3 copies of the JSON file and name each file according to its locale.
  + The above is better handled as follows: make copies of the first JSON file after all strings are localized properly. Then make sure that each copy is named after the locale it is intended to support.
* TMS is expected to connect to and collect the JSON files from the website's source code repository for translation and push back the files after translation into the repository.
* The workflow between the script and TMS is expected to take place in the website's source code.
* TMS is expected to handle the automatic detection of updates in the JSON files and collect the updates for translation and reintegration into the source code.
* The client is expected to manually Integrate the updated source code with the latest translations into their live website.
* The client is expected to adopt the localization structure adopted by Rakuten to localize their website. This means that the client should now be adding content in the JSON files instead of directly hard coding the strings in the HTML or JS files.
* My project is expected to handle the manual introduction of a function that allows the website visitors to switch among the available languages the website is now supporting. The selected language option from the drop-down menu should automatically change the website's content source from the corresponding JSON file in the “translations” directory.
* The continuous localization, translation reviews and validations, automatic updates of translated files are expected to be handled by TMS.

# Backend: R-Loc Plugin

* The plugin is expected to work with multiple CMSs or otherwise a version of the plugin is developed for each CMS supported.
* The plugin automatically collects saved data from the CMS’s database and sends it to a dedicated repository where TMS can pick it up for translation and put it back for manual integration in the database.
* The plugin is expected to create or upgrade the database schema to support the new locales enabled on the CMS.
  + Example: creating new tables form existing tables’ schemas but suffixing the locale names to corresponding table names. Adjusting the CMS’s functionality to read from the specific tables based on user language choice from a drop-down menu.
* The plugin should handle the enabling of new locales on the CMS.
* The plugin should be able to collect new translated strings from TMS and post them to their respective tables.