{{/\* socials/GetRegisteredServices Retrieves the list of user registered services. Support legacy settings (root of params with service name as key) @author @regisphilibert @context Any (.) @access private @returns Slice of Maps - String (.name) String (.url) String (.label)? String (.color)? \*/}} {{ $registered\_services := slice }} {{/\* We first look for legacy settings that lives at the root of the site.Params as such (github: https://github.com/ theNewDynamic) and them to the list with key as .name and value as .url \*/}} {{ $user\_using\_legacy := false }} {{ $legacy\_api\_services := slice "facebook" "twitter" "instagram" "youtube" "github" "gitlab" "keybase" "linkedin" "medium" "mastodon" "slack" "stackoverflow" "rss" }} {{ range $name := $legacy\_api\_services }} {{ with $url := index site.Params . }} {{/\* If we can find a parameter matching the key with a set value, we add it with proper name and url \*/}} {{/\* We also note that user is using legacy for potential potential deprecation warnings \*/}} {{ $user\_using\_legacy = true }} {{ $registered\_services = $registered\_services | append (dict "name" $name "url" $url) }} {{ end }} {{ end }} {{/\* Then we go through the current way of registering services as per referenced in README \*/}} {{ with site.Params.ananke\_socials }} {{ range $service := . }} {{/\* Only if the service has a .name, we add it all its keys to the slice of registered services \*/}} {{ with .name }} {{ $registered\_services = $registered\_services | append $service }} {{ end }} {{ end }} {{ end }} {{ return $registered\_services }}