

sog^o

Developer's Guide

SOGGo v5.10.0

Table of Contents

1. UI and UX	2
2. Frameworks & Tools	3
2.1. GNUstep	3
2.2. SOPE	3
2.3. JavaScript libraries	3
3. Code Style	4
3.1. Objective-C	4
3.2. HTML	4
3.3. JavaScript	4
3.4. SASS/CSS	5
4. Building frontend	6
4.1. Defining an alternate color theme	6
5. Version Control	8
6. Testing	9
6.1. JSON API	9

Version 5.10.0 - février 2023

Copyright © 2023 Inverse inc. / Alinto

Permission is granted to copy, distribute and/or modify this document under the terms of the GNU Free Documentation License, Version 1.2 or any later version published by the Free Software Foundation; with no Invariant Sections, no Front-Cover Texts, and no Back-Cover Texts. A copy of the license is included in the section entitled "GNU Free Documentation License".

The fonts used in this guide are licensed under the SIL Open Font License, Version 1.1. This license is available with a FAQ at: <http://scripts.sil.org/OFL>

Copyright © Łukasz Dziedzic, <http://www.latofonts.com>, with Reserved Font Name: "Lato".

Copyright © Raph Levien, <http://levien.com/>, with Reserved Font Name: "Inconsolata".

᠑ᠡᠢᠨᠠᠯᠢᠨᠲᠤᠨᠢ
Alinto

1. UI and UX

- <https://material.io/design/>

2. Frameworks & Tools

2.1. GNUstep

2.2. SOPE

2.3. JavaScript libraries

- <https://angularjs.org/>
- <https://material.angularjs.org/>
- <http://angular-ui.github.io/ui-router/>
- <http://lodash.com/>
- <https://github.com/nervgh/angular-file-upload>
- <https://github.com/a5hik/ng-sortable>

3. Code Style

3.1. Objective-C

- To document the Web APIs, we follow the [APIDOC](#) standards.

```
apidoc -f ".*\.\.m" -i UI -o Documentation/JSON-API
```

- When returning JSON from an object extending `UINavigationController`, use the following method:
 - `(WOResponse *) responseWithStatus: (unsigned int) status andJSONRepresentation: (NSObject *) contentObject;`

3.2. HTML

- Localize your strings:

```
<var:string label:value="I'm localized!"/>
```

- Localize your attributes:

```
<input type="text" label:placeholder="I'm localized!"/>
```

- Reuse existing localized strings as much as possible. Otherwise don't forget to update the English `Localizable.strings` file of the appropriate module.

3.3. JavaScript

- <http://eslint.org/>
- <http://standardjs.com>
- <https://github.com/toddmotto/angularjs-styleguide>
- <https://github.com/angular/material/blob/master/docs/guides/CODING.md>
- <http://jshint.com/>
- <https://github.com/angular/material/tree/master/docs/guides>
- We conform to [Airbnb coding standards](#). Use `jscs` with the `airbnb` preset to validate your code style:

```
jscs -p airbnb *.js
```

- For documentation, we follow the [JSDoc3](#) standards:

```
jsdoc
UI/WebServerResources/js/{Common,Contacts,Mailer,Preferences,Scheduler}/*.js
-d Documentation/JavaScript-API
```

3.3.1. Models (facades)

- Move business logic into models and share them.
- Keep it simple, separate server interaction and error handling from the model. That way model only handle data processing and code is easier to maintain.

3.3.2. Controllers

- Functions not exposed in the controller must be prefixed with an underscore.

3.4. SASS/CSS

- <http://componentcss.com/>
- <https://github.com/styleguide/css>
- <https://smacss.com/>
- <http://cssguidelin.es/>
- <http://sixrevisions.com/css/css-methodologies/>
- <https://css-tricks.com/what-a-css-code-review-might-look-like/>
- <http://una.im/classy-css/>
- <http://stylelint.io/>

A **@mixin** is like a stamp: it creates a duplicated version of the property block (optionally) with arguments provided. An **@extend** appends the element you are extending to the property block. It is your "yes, and _" statement.

- For documentation, we follow [SassDoc](#) annotations.
- FlexBox compatibility <http://caniuse.com/#feat=flexbox>

3.4.1. Fonts

- [Mozilla Fira](#)
- [Material icons](#)

4. Building frontend

- Install the latest stable release of [Node.js](#)
- Install grunt

```
npm install -g grunt-cli
npm install -g bower
```

- We need the SASS files of Angular Material to build our CSS. The git repository of Angular Material is included as a submodule of SOGo:

```
git submodule init
git submodule update
```

- Generate the JavaScript and CSS files

```
cd UI/WebServerResources
npm install
bower install
grunt build
```

4.1. Defining an alternate color theme

SOGo relies on the [theming system of Angular Material](#) to define the colors of the Web interface.

To overwrite the default theme in SOGo, set the following parameter in `/etc/sogo/sogo.conf`:

```
SOGoUIAdditionalJSFiles = (js/theme.js);
```

Edit `theme.js` under `/usr/lib64/GNUstep/SOGo/WebServerResources/js` or `/usr/lib/GNUstep/SOGo/WebServerResources/js` depending on your platform and restart `sogod`.

If the configuration parameter `SOGoUIxDebugEnabled` is unset or set to `NO` in `/etc/sogo/sogo.conf`, you'll need to generate a new `theme-default.css` stylesheet for the new theme.

1. Temporarily set `SOGoUIxDebugEnabled` to `YES`;
2. Restart `sogod`;
3. From your favorite browser, open the JavaScript console and type the following:

```
copy([], slice.call(document.styleSheets)
  .map(e => e.ownerNode)
  .filter(e => e.hasAttribute('md-theme-style'))
  .map(e => e.textContent)
  .join('\n')
)
```

4. Overwrite the content of `WebServerResources/css/theme-default.css` with the content of the clipboard;
5. Restore the value of `SOGoUIxDebugEnabled` (NO or unset);
6. Restart `sogod`.

5. Version Control

- <https://devcharm.com/articles/46/improve-your-git-workflow/>
- <https://github.com/angular/material/blob/master/docs/guides/CONTRIBUTING.md#-commit-message-format>
- Each commit should cover only one thing;
- Begin the commit message with a single short (less than 50 characters) line summarizing the change, followed by a blank line and then a more thorough description;
- When fixing a bug, commit to the devel branch as well as the maintenance branch of the latest release version (named maintenance/x.y). When a ticket is associated to the bug, add to the description a line saying **Fixes #1234**.
- `git pull` may introduce [inconsistencies and problems](#). Replace it with the following alias:

```
git config --global alias.up '!git remote update -p; git merge --ff-only @u}'
```

6. Testing

- <https://github.com/angular/protractor>
- <http://karma-runner.github.io/>

6.1. JSON API

One practical way to test the JSON API is to use `curl`. To do so, you need to enable `S0GoTrustProxyAuthentication` and configure HTTP authentication in Apache. You can pipe the result to `jq` to nicely format and filter the output:

```
curl -u username:password  
http://localhost/S0Go/so/username/Calendar/calendarslist | jq '.'
```

```
curl -u username:password -H 'Content-Type: application/json' -d '{}'  
http://localhost/S0Go/so/francis/Calendar/personal/71B6-54904400-1-  
7C308500.ics/save
```