
FuretUI Documentation

Release 1.2.0

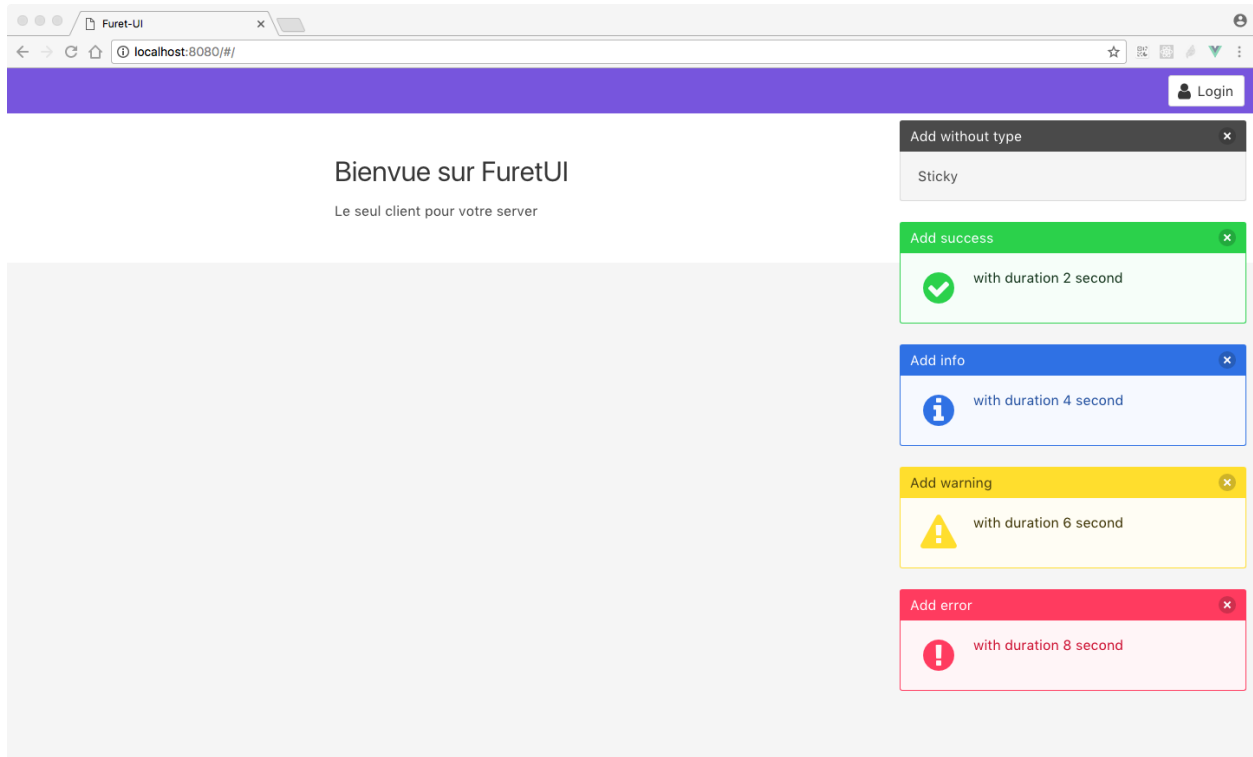
Jean-Sébastien Suzanne

Jul 23, 2017

Contents

1	Front Matter	3
1.1	Project Homepage	3
1.2	Project Status	3
1.3	Installation	3
1.4	Unit Test	4
1.5	Contributing (hackers needed!)	4
1.6	Author	4
1.7	Contributors	4
1.8	Bugs	4
2	Add FuretUI in your own project	5
3	Some picture	7
4	API	13
4.1	APP	13
4.2	Right and Left Menu	13
4.3	Space	14
4.4	Picture	15
4.5	Views	15
4.6	Field	18
4.7	Plugin	19
5	Store and Flow	21
5.1	client	21
5.2	leftmenu, rightmenu	22
5.3	global	23
5.4	data	25
5.5	View definition for store	29
6	Client / Server Api	33
6.1	/furetui/init/required/data	33
6.2	/furetui/init/optionnal/data	33
6.3	/furetui/homepage	34
6.4	/furetui/button/<buttonId>	34
6.5	/furetui/custom/view/<viewName>	34
6.6	/furetui/view/<viewId>	34

6.7	/furetui/space/<spaceId>	34
6.8	/furetui/action/<actionId>	34
6.9	/furetui/field/x2x/search	35
6.10	/furetui/list/x2m/get	35
6.11	/furetui/field/x2m/get/views	35
6.12	/furetui/data/create	36
6.13	/furetui/data/read	36
6.14	/furetui/data/read/<dataId>	36
6.15	/furetui/data/update	37
6.16	/furetui/data/delete	37
6.17	/furetui/data/search	37
6.18	Default route for login and logout	38
7	Template	39
8	RELEASE	41
8.1	1.2.0 (2017-07-23)	41
8.2	1.1.0 (2017-07-22)	41
8.3	1.0.0 (2017-07-22)	41
9	Mozilla Public License Version 2.0	45
9.1	1. Definitions	45
9.2	2. License Grants and Conditions	47
9.3	3. Responsibilities	48
9.4	4. Inability to Comply Due to Statute or Regulation	49
9.5	5. Termination	49
9.6	6. Disclaimer of Warranty	49
9.7	7. Limitation of Liability	50
9.8	8. Litigation	50
9.9	9. Miscellaneous	50
9.10	10. Versions of the License	50
9.11	Exhibit A - Source Code Form License Notice	51
9.12	Exhibit B - “Incompatible With Secondary Licenses” Notice	51



Contents

- *Front Matter*
 - *Project Homepage*
 - *Project Status*
 - *Installation*
 - *Unit Test*
 - *Contributing (hackers needed!)*
 - *Author*
 - *Contributors*
 - *Bugs*

Information about the FuretUI project.

Project Homepage

FuretUI is hosted on [github](#) - the main project page is at http://github.com/AnyBlok/furet_ui. Source code is tracked here using [GIT](#).

The most recent published version of this documentation should be at <http://furet-ui.readthedocs.io/en/latest/index.html>.

Project Status

FuretUI is currently in development status and is expected to be fairly stable. Users should take care to report bugs and missing features on an as-needed basis. It should be expected that the development version may be required for proper implementation of recently repaired issues in between releases; the latest master is always available at https://github.com/AnyBlok/furet_ui/archive/master.zip.

Installation

Install latest versions of FuretUI from github (need **npm**):

```
git clone https://github.com/AnyBlok/furet_ui.git
cd furet_ui
npm install
npm run build-prod
```

..warning:

```
``build-prod`` return a minimify build
```

Unit Test

Run the test with npm:

```
npm run test
```

AnyBlok is tested using [Travis](#)

Run the mock server from the project:

```
npm run build
virtualenv develop
source develop/bin/activate
pip install -r requirement.txt
python server.py
```

Contributing (hackers needed!)

FuretUI is at a very early stage, feel free to fork, talk with core dev, and spread the word!

Author

Jean-Sébastien Suzanne

Contributors

[Anybox](#) team:

- Jean-Sébastien Suzanne
- Simon ANDRÉ

Bugs

Bugs and feature enhancements to FuretUI should be reported on the [Issue tracker](#).

Contents

- *Add FuretUI in your own project*

CHAPTER 2

Add FuretUI in your own project

FuretUI don't give a final web client, You have to create the **html** page. In this page you must:

- 1. create page: (css, html, js)
- 2. load furetui components
- 3. load your own components
- 4. create the FuretUI Client

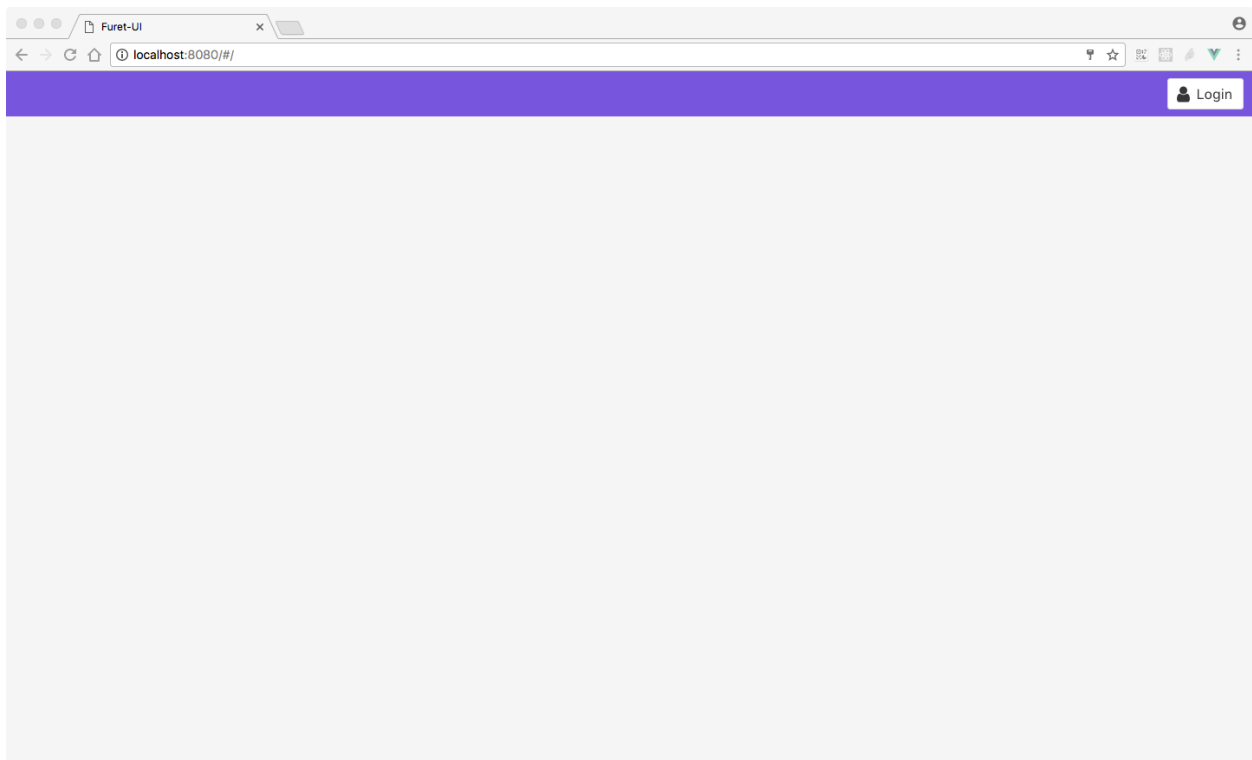
Example:

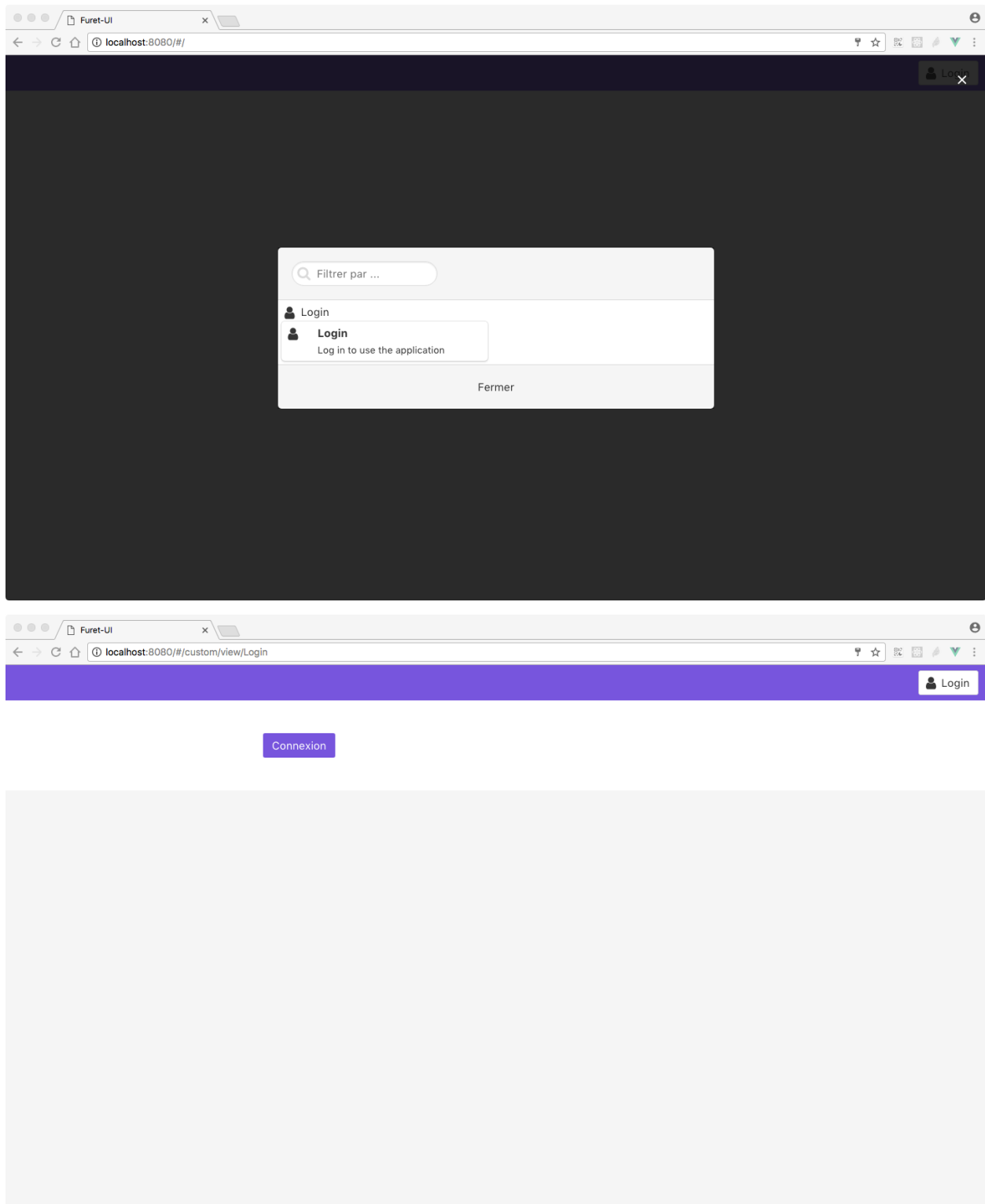
```
<!DOCTYPE html>
<html>
  <head>
    <meta http-equiv="content-type" content="text/html; charset=utf-8" />
    <meta name="viewport" content="width=device-width, initial-scale=1, maximum-
↪scale=1.0, user-scalable=no"/>
    <title>Your title</title>
    <!--
      Your css files
    -->
  </head>
  <body>
    <!-- the entry point of the client -->
    <div id="anyblok-furetui-app"></div>
    <!--
      Your js files
    -->
    <!--
      Furet UI bundles
    -->
    <!--
      Your components for FuretUI
    -->
    <!-- creation of the client -->
    <script type="text/javascript">
```

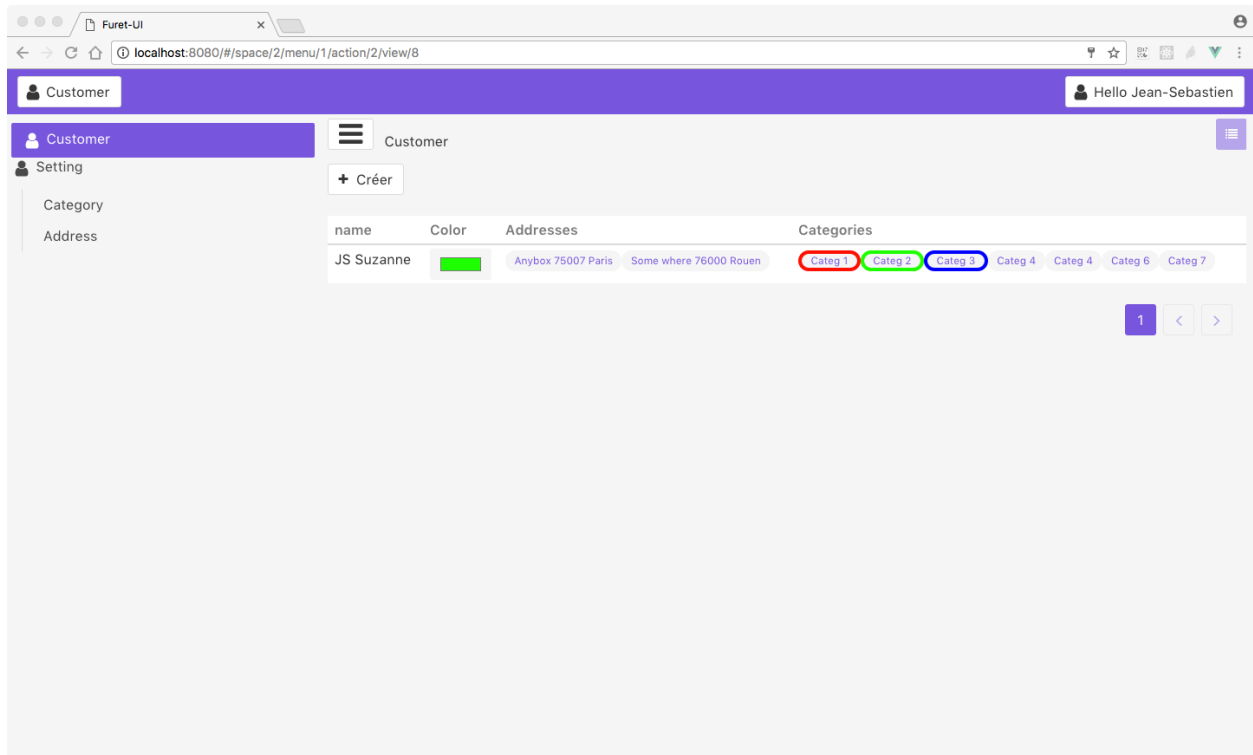
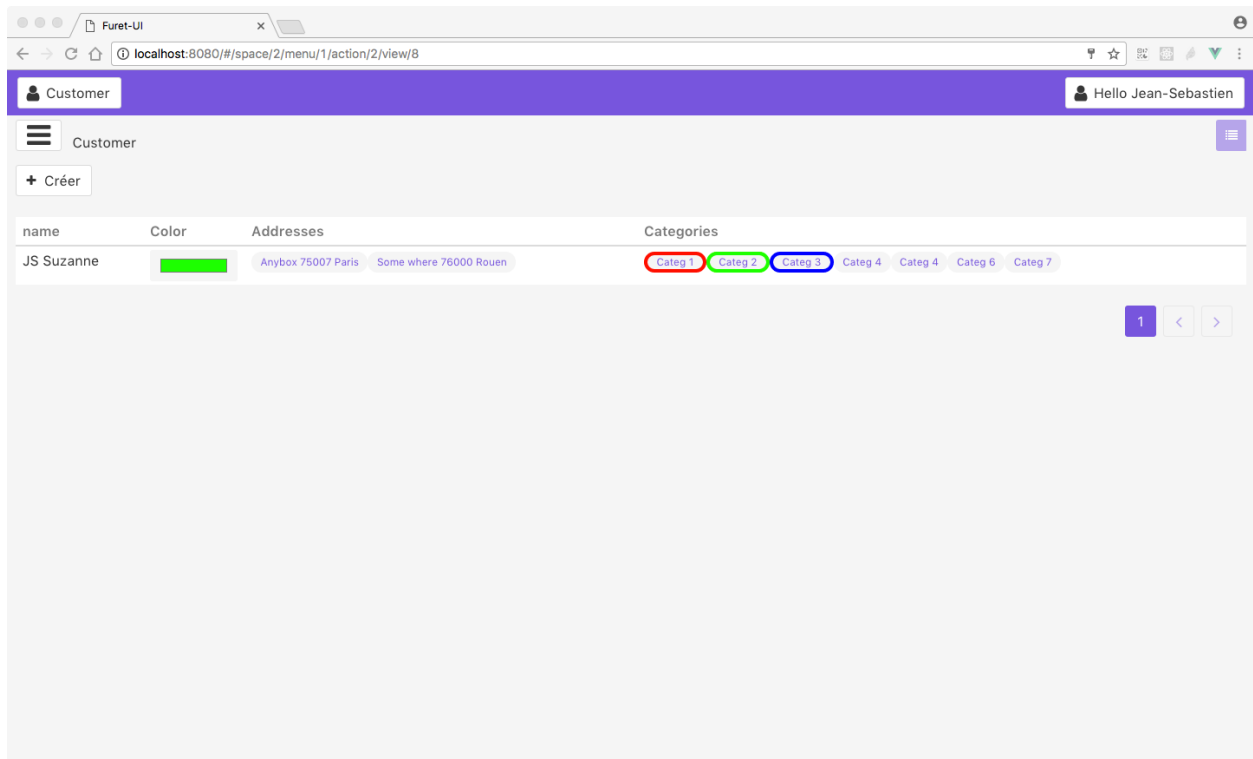
```
        createFuretUIClient('#anyblok-furetui-app');  
    </script>  
</body>  
</html>
```

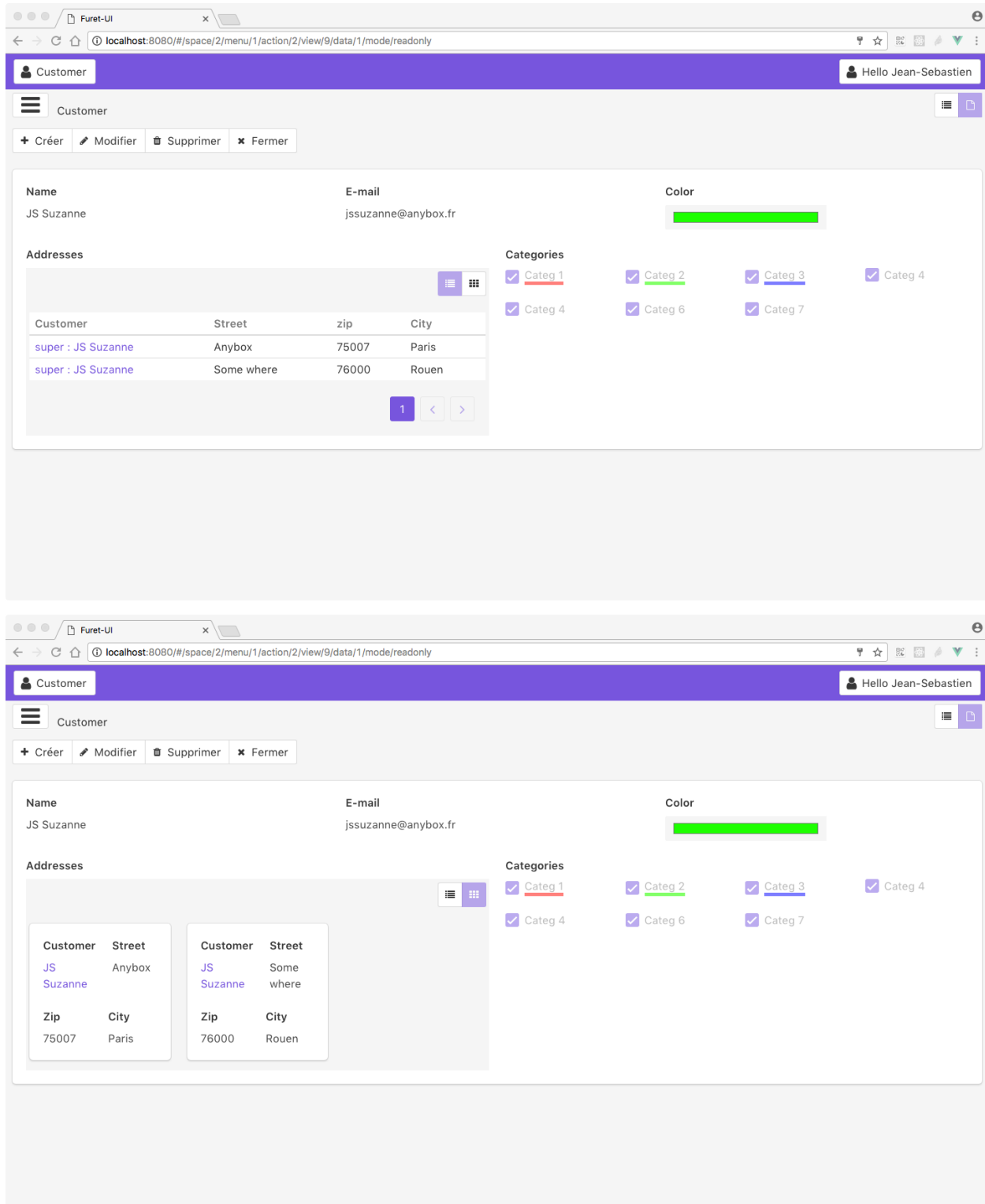
CHAPTER 3

Some picture









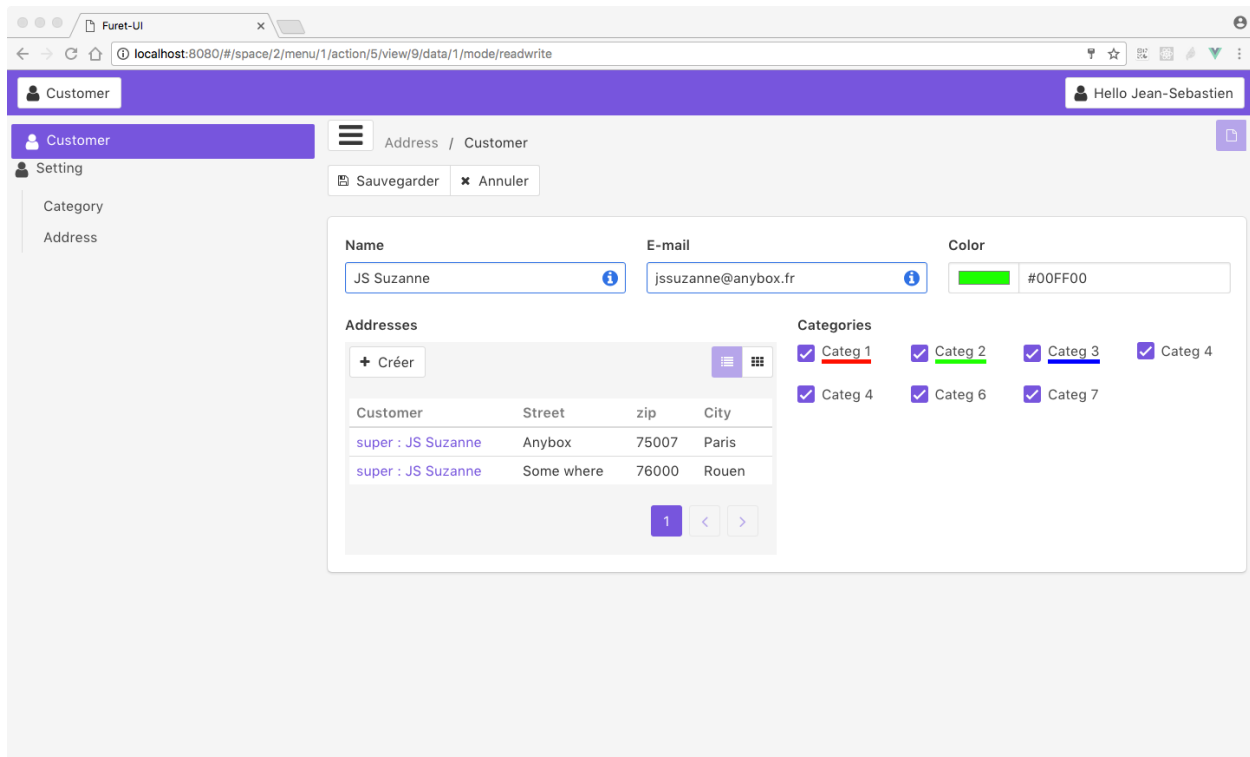
The screenshot shows a web browser window with the URL `localhost:8080/#/space/2/menu/1/action/2/view/9/data/1/mode/readonly`. The page title is "Customer". The user is logged in as "Hello Jean-Sebastien". The form contains the following fields:

- Name:** JS Suzanne
- E-mail:** jssuzanne@anybox.fr
- Color:** A green color picker.
- Addresses:** A list of addresses with a "Fermer" button. The first address is:

Customer	Street
JS Suzanne	Some where
Zip	City
76000	Rouen
- Categories:** A list of checkboxes for categories: Categ 1, Categ 2, Categ 3, Categ 4, Categ 4, Categ 6, and Categ 7.

The screenshot shows the same FuretUI interface but in "edit" mode. The URL is `localhost:8080/#/space/2/menu/4/action/4/view/13/data/1/mode/readonly`. The page title is "Customer". The user is logged in as "Hello Jean-Sebastien". The form contains the following fields:

- Customer:** JS Suzanne
- Street:** Anybox
- Zip:** 75007
- City:** Paris

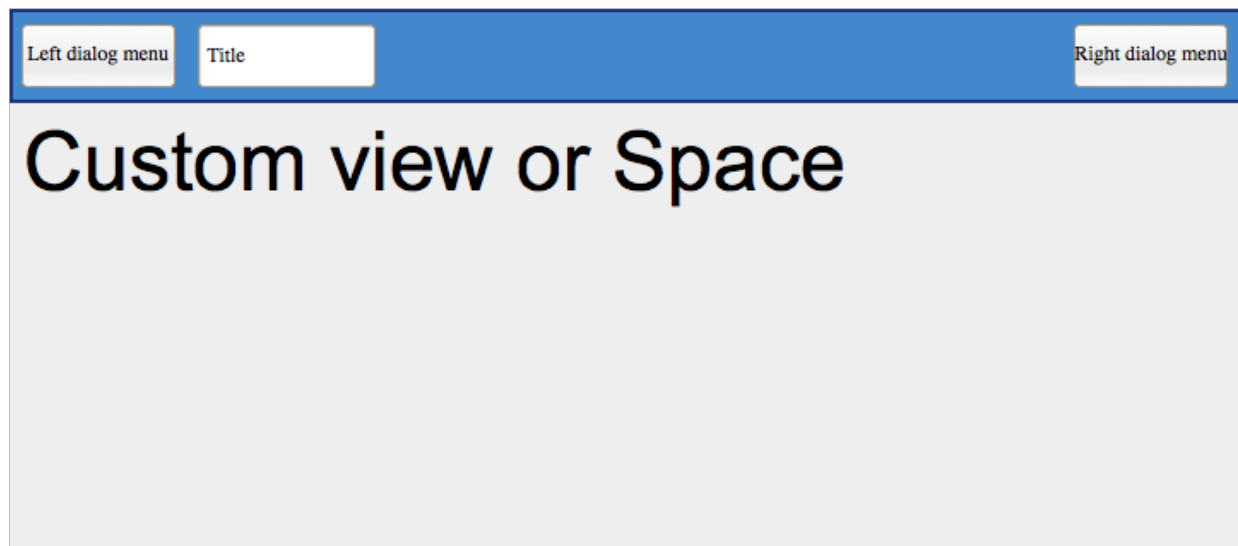


Contents

- *API*
 - *APP*
 - *Right and Left Menu*
 - *Space*
 - *Picture*
 - *Views*
 - * *Unknown*
 - * *List*
 - * *Form*
 - * *Thumbnail*
 - *Field*
 - * *Simple*
 - * *Relation ship*
 - *Plugin*

APP

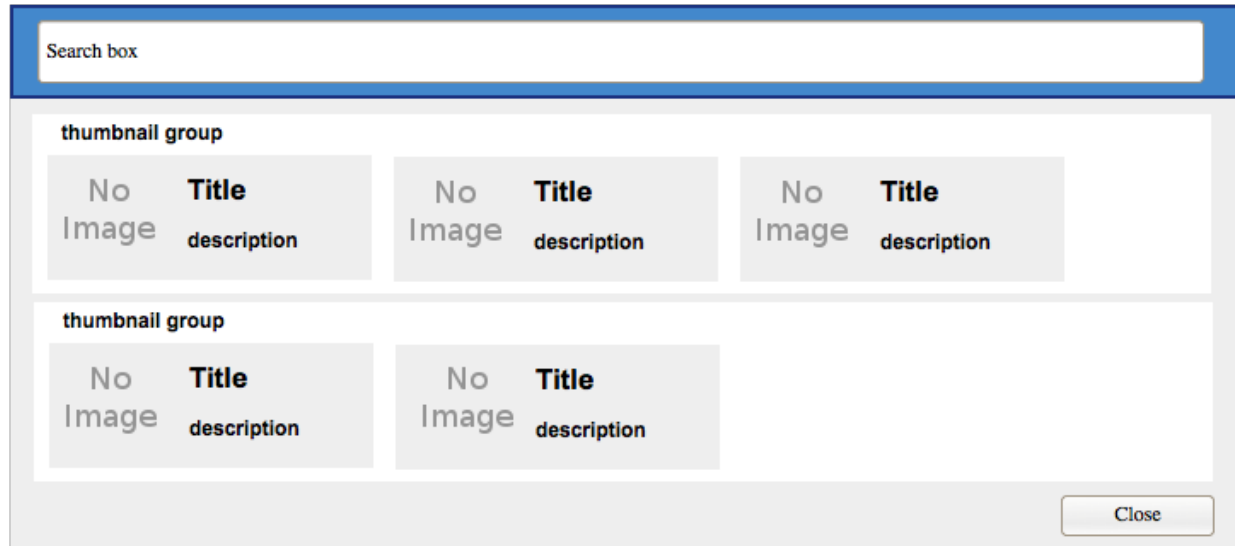
Render the application, need the providers for theme and redux storage



```
<furet-ui />
```

Right and Left Menu

It is a Modal menu. It is a thumbnail render, the thumbnail can be filtered by a searchBox



```
<furet-ui-appbar-left-menu />  
<furet-ui-appbar-right-menu />
```

Space

A space is an environnement for a specific behavior. Each space can defined menus, actions, views



```
<furet-ui-space
  v-bind:spaceId="..."
  v-bind:menuId="..."
  v-bind:actionId="..."
  v-bind:viewId="..."
/>
```

Picture

Render a picture, the picture can be come from:

- font-icon: font-awesome class

```
<furet-ui-picture type="font-icon" value="fa-user" />
```

Views

List the type of view available for the server.

The view type can be:

- standard: List, Form, Thumbnail

- custom: Login, Logout

All the view are present in the plugin system of FuretUI.

Unknown

Replace the wanted view if no view are available

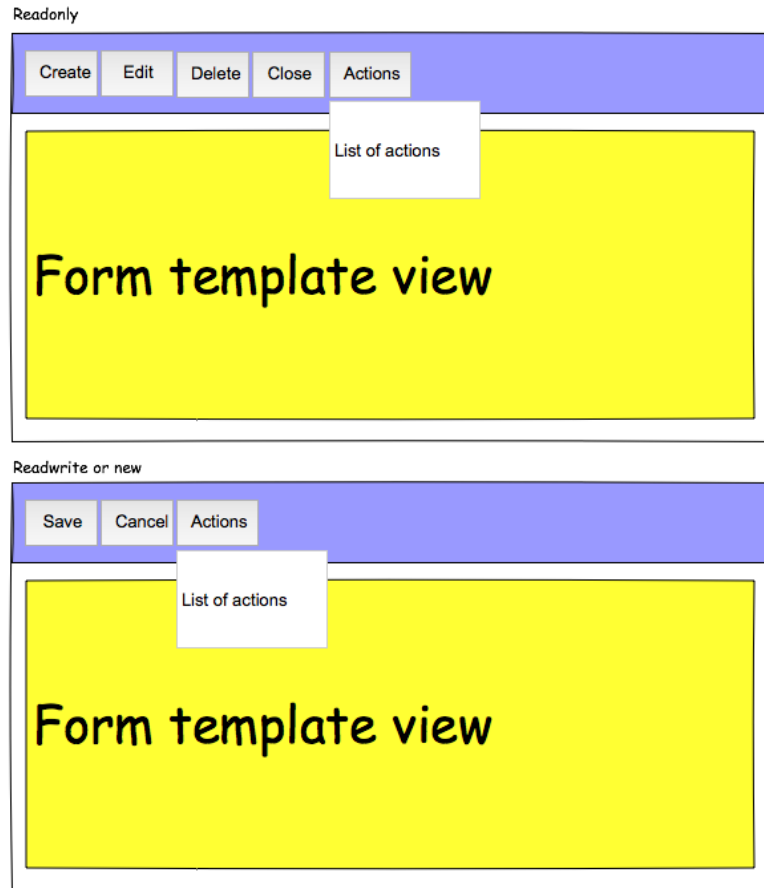
List

Render a list of data

The image displays two examples of a list view in FuretUI. The top example shows a header bar with a 'Create' button, an 'Actions' dropdown menu, and a search box. Below the header is a list of eight yellow rows, each labeled 'Line'. The bottom example shows a more complex header bar with 'Create', 'Delete', 'Actions', and 'More' buttons, each with a dropdown menu, and a search box. Below the header is a list of eight yellow rows, each labeled 'Line' and preceded by a blue checkmark icon. The 'Actions' and 'More' dropdown menus are open, showing a 'List of actions' option.

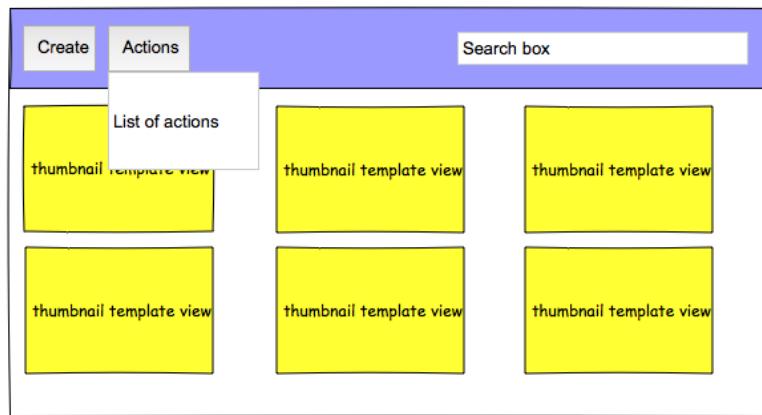
Form

Render a react template, the template come from serveur and stock in the redux store



Thumbnail

Render a react template, the template come from serveur and stock in the redux store



Field

Each field must be defined by views type.

Simple

The list are:

- Boolean
- Color
- Date
- DateTime
- Float
- Integer
- File
- JSON
- Mail

- Password
- RichText
- Selection
- String
- Text
- Time
- URL

Relation ship

Relation ship are particular field which link two model.

The list are:

- Many2One
- **Many2Many**
 - List,
 - Thumbnail
 - **Form**
 - * checkbox
 - * tags
- One2Many

Plugin

It is not a composent, it allow to save some function / object / class. This function / object / class are used and can be overwrite easily to change the behavior.

```
import plugin from './plugin';

const func = (props) => {...}
plugin(['path', 'to', 'save', 'in', 'plugin'], {Hello: func});

plugin.get(['path', 'to', 'save', 'in', 'plugin', 'Hello']);
```

Contents

- *Store and Flow*
 - *client*
 - * *UPDATE_CLIENT*
 - * *CLEAR_CLIENT*
 - *leftmenu, rightmenu*

- * *UPDATE_LEFT_MENU, UPDATE_RIGHT_MENU*
- * *CLEAR_LEFT_MENU, CLEAR_RIGHT_MENU*
- *global*
 - * *UPDATE_GLOBAL*
 - * *CLEAR_GLOBAL*
 - * *ADD_IN_BREADSCRUMB*
 - * *REMOVE_FROM_BREADSCRUMB*
 - * *CLEAR_BREADSCRUMB*
 - * *ADD_NOTIFICATION* (*mutation*)
 - * *ADD_NOTIFICATION* (*action*)
 - * *REMOVE_NOTIFICATION*
- *data*
 - * *UPDATE_ACTION*
 - * *UPDATE_VIEW*
 - * *UPDATE_DATA*
 - * *DELETE_DATA*
 - * *UPDATE_CHANGE*
 - * *REPLACE_CHANGE*
 - * *CLEAR_CHANGE*
 - * *CREATE_CHANGE_X2M*
 - * *UPDATE_CHANGE_X2M*
 - * *UPDATE_CHANGE_X2M_DELETE*
 - * *UPDATE_SPACE*
 - * *CLEAR_DATA*
- *View definition for store*
 - * *List*
 - * *Thumbnail*
 - * *Form*

client

Add any information about a custom client:

```
store.state.client: {  
  ``custom view name``: {  
    ``params of the custum view``,  
  },  
  ...  
}
```

UPDATE_CLIENT

Add params for a custom view:

```
{  
  type: 'UPDATE_VIEW_CLIENT',  
  viewName: ``custom view name``,  
  ``params ...``,  
}
```

CLEAR_CLIENT

Clear all the params:

```
{  
  type: 'CLEAR_CLIENT'  
}
```

leftmenu, rightmenu

It is the same for the both. The define dialog menu:

```
store.state.[ leftmenu | rightmenu ] = {
  value: {
    label: ``label of the selected menu``,
    image: {
      type: [ font-icon ],
      value: ``value in function of type to display the image``,
    },
  },
  values: [
    {
      label: ``label of the selected menu``,
      description: ``Description of the menu``,
      image: {
        type: [ font-icon ],
        value: ``value in function of type to display the image``,
      },
      type: [ SPACE | CLIENT ]
      id: ``space id or custom view name in function of type value``
    },
  ],
}
```

UPDATE_LEFT_MENU, UPDATE_RIGHT_MENU

Update the menu definition or selected menu:

```
{
  type: [ UPDATE_LEFT_MENU | UPDATE_RIGHT_MENU ],
  value: {
    label: ``label of the selected menu``,
    image: {
      type: [ font-icon ],
      value: ``value in function of type to display the image``,
    },
  },
  values: [
    {
      label: ``label of the selected menu``,
      description: ``Description of the menu``,
      image: {
        type: [ font-icon ],
        value: ``value in function of type to display the image``,
      },
      type: [ SPACE | CLIENT ]
      id: ``space id or custom view name in function of type value``
    },
  ],
}
```

Note: value and values is not required

CLEAR_LEFT_MENU, CLEAR_RIGHT_MENU

Clear the menu:

```
{
  type: [ CLEAR_LEFT_MENU | CLEAR_RIGHT_MENU ]
}
```

global

Define the context of the application:

```
store.state.global = {
  title: ``title of the application``,
  modal_custom_view: ``the current modal custom view``,
  breadscrums: [
    {
      path: ``the path of the previous action``,
      label: ``label display in the breadcrumb``,
      changes: ``the changes did during the action``,
      position: ``to order the Array``,
    },
    ...
  ],
}
```

UPDATE_GLOBAL

Update the global values:

```
{
  type: 'UPDATE_GLOBAL',
  ``params``,
}
```

Warning: Used for title or modal_custom_view.

CLEAR_GLOBAL

Clear all the parameter in global:

```
{
  type: 'CLEAR_GLOBAL',
}
```

ADD_IN_BREADCRUMB

Add new entry at the end of the breadcrumb:

```
{
  type: 'ADD_IN_BREADSCRUMB',
  path: ``the path to save``,
  label: ``the label to display``,
  changes: ``the state data changes to save``,
}
```

REMOVE_FROM_BREADSCRUMB

Remove the breadcrumb entries if the position is higher than the wanted position:

```
{
  type: 'REMOVE_FROM_BREADSCRUMB',
  position: ``the position to delete``
}
```

CLEAR_BREADSCRUMB

Clear all the breadcrumb entries:

```
{
  type: 'CLEAR_BREADSCRUMB',
}
```

ADD_NOTIFICATION (mutation)

Add new notification in the notification stack:

```
{
  type: 'ADD_NOTIFICATION',
  process: 'commit',
  id: ``unique id``
  title: ``title of the notification``
  message: ``html template of the message``,
  has_icon: ``display icon only for notification type success, info, warning or_
↪error``,
  notification_type: [ ' ' | success | info | warning | error ],
}
```

Warning: If you need duration, you must use the action not the mutation

ADD_NOTIFICATION (action)

Add new notification in the notification stack:

```
{
  type: 'ADD_NOTIFICATION',
  process: 'dispatch',
  id: ``unique id``
}
```

```

title: ``title of the notification``
message: ``html template of the message``,
has_icon: ``display icon only for notification type success, info, warning or_
↪error``,
notification_type: [ ' ' | success | info | warning | error ],
duration: ``number of the duration before remove, null for sticky``,
}

```

REMOVE_NOTIFICATION

Remove from notifications stack the notification id:

```

{
  type: 'REMOVE_NOTIFICATION',
  id: ``id of the notification``,
}

```

data

Save all the data for furet_ui:

```

store.state.data = {
  actions: {
    ``action id``: {
      label: ``Label of the action``,
      views: [
        {
          viewId: ``view id``,
          type: ``type of view``,
        },
        ...
      ],
    },
    ...
  },
  views: {
    ``view's id``: {
      model: ``model's name``,
      ``view params which depend of the type of the view``
    },
  },
  data: {
    ``Model name``: {
      ``data id``: {
        ``data``,
        ...
      },
      ...
    },
    ...
  },
  changes: {
    new: {
      ``Model name``: {

```

```
        ``data id``: [ 'DELETED' || { ``data`` } ],
        ...
    },
    ...
},
``Model name``: {
    ``data id``: [ 'DELETED' || { ``data`` } ],
    ...
},
...
},
spaces: {
    left_menu: [
        ``menu params``,
        ...
    ],
    right_menu: [
        ``menu params``,
        ...
    ],
}
}
```

the left and right menu definition is the same:

```
[
    {
        label: ``displayed label``,
        image: {
            type: [ 'font-icon' ],
            value: ``value of the icon``,
            actionId: ``action id``,
            id: ``id of the menu``,
            submenus: [
                ``Same menu definition, to use the submenus don't put actionId``,
                ...
            ],
        },
    },
]
```

UPDATE_ACTION

Add action definition:

```
{
    type: 'UPDATE_ACTION_MANAGER_ADD_ACTION_DATA',
    actionId: ``action id``,
    label: ``Label of the action``,
    views: [
        {
            viewId: ``view id``,
            type: ``type of view``,
        },
        ...
    ],
}
```

UPDATE_VIEW

Update the data of one view:

```
{
  type: 'UPDATE_VIEW',
  viewId: ``view id``,
  model: ``model name``,
  ``params of the view``,
}
```

UPDATE_DATA

Update the data:

```
{
  type: 'UPDATE_DATA',
  model: ``Model name``,
  data: {
    ``data id``: {
      ``params``,
    },
    ...
  },
}
```

DELETE_DATA

Delete some data:

```
{
  type: 'DELETE_DATA',
  data: {
    ``Model name``: [
      ``data id``,
    ],
  },
}
```

UPDATE_CHANGE

Modify current change:

```
{
  type: 'UPDATE_CHANGE',
  model: ``Model name``,
  dataId: ``data id``,
  fieldname: ``name of the field``,
  value: ``new value to save``,
}
```

REPLACE_CHANGE

replace the change by another:

```
{
  type: 'REPLACE_CHANGE',
  changes: ``object``,
}
```

CLEAR_CHANGE

replace the change by another:

```
{
  type: 'CLEAR_CHANGE',
}
```

CREATE_CHANGE_X2M

Add new change in the new entry, use by One2Many and Many2Many:

```
{
  type: 'CREATE_CHANGE_X2M',
  model: ``model name``,
  dataId: ``id of the data to create``
}
```

UPDATE_CHANGE_X2M

Modify current change in new if exist or in the normal place:

```
{
  type: 'UPDATE_CHANGE_X2M',
  model: ``Model name``,
  dataId: ``data id``,
  fieldname: ``name of the field``,
  value: ``new value to save``,
}
```

UPDATE_CHANGE_X2M_DELETE

Remove the change in the new entry if exist else add the 'DELETED' in the normal place for the model and the dataIds:

```
{
  type: 'UPDATE_CHANGE_X2M_DELETE',
  model: ``Model name``,
  dataIds: [
    ``data id``,
    ...
  ],
}
```


UPDATE_SPACE

Update the space data:

```
{
  type: 'UPDATE_SPACE',
  spaceId: ``
  left_menu: [
    ``menu params``,
    ...
  ],
  right_menu: [
    ``menu params``,
    ...
  ],
}
```

CLEAR_DATA

Clear all the data:

```
{
  type: 'CLEAR_DATA'
}
```

View definition for store

Note: the params are different for each view type

List

```
label: ``label of the view``,
model: ``model name``,
creatable: ``true display the create button``,
deletable: ``true display the delete button``,
selectable: ``true display the check box``,
onSelect: ``view id to use if the line is clicked``,
empty: ``html template displayed when no data``,
headers: [
  {
    name: ``column name``,
    label: ``Label of the column``,
    component: ``the vue component name``
    sortable: ``true: the column is sortable``
    numeric: ``true the value is a numeric``
    invisible: ``condition to determinate if the field is displayed or not``
  },
  ...
],
search: [
```

```
{
  key: ``key used by the server to filter the data``,
  label: ``Label display of the key``,
  type: ``what ever if you need to split the filter``,
  operator: ``what ever if you want define this``,
  value: ``the value``,
},
...
],
buttons: [
  {
    label: ``Label of the button``,
    buttonId: ``id of the button``,
  },
  ...
],
onSelect_buttons: [
  {
    label: ``Label of the button``,
    buttonId: ``id of the button``,
  },
  ...
],
fields: [
  ``column name``,
  ...
],
],
```

Thumbnail

```
label: ``label of the view``,
model: ``model name``,
creatable: ``true display the create button``,
deletable: ``true display the delete button``,
onSelect: ``view id to use if the line is clicked``,
border_fieldcolor: ``color to display can be hard coded or come from a field or eval_
↪some condition``,
background_fieldcolor: ``color to display can be hard coded or come from a field or_
↪eval some condition``,
search: [
  {
    key: ``key used by the server to filter the data``,
    label: ``Label display of the key``,
    type: ``what ever if you need to split the filter``,
    operator: ``what ever if you want define this``,
    value: ``the value``,
  },
  ...
],
template: ``template definition``,
buttons: [
  {
    label: ``Label of the button``,
    buttonId: ``id of the button``,
  },
  ...
],
```

```
],
fields: [
  ``column name``,
  ...
],
```

Note: For the template see the template page [Template](#)

Form

```
label: ``label of the view``,
model: ``model name``,
creatable: ``true display the create button``,
deletable: ``true display the delete button``,
editable: ``true display the edit button``,
onClose: ``view id to use if the close button is clicked``,
template: ``template definition``,
buttons: [
  {
    label: ``Label of the button``,
    buttonId: ``id of the button``,
  },
  ...
],
fields: [
  ``column name``,
  ...
],
```

Note: For the template see the template page [Template](#)

Contents

- *Client / Server Api*
 - `/furetui/init/required/data`
 - `/furetui/init/optionnal/data`
 - `/furetui/homepage`
 - `/furetui/button/<buttonId>`
 - `/furetui/custom/view/<viewName>`
 - `/furetui/view/<viewId>`
 - `/furetui/space/<spaceId>`
 - `/furetui/action/<actionId>`
 - `/furetui/field/x2x/search`
 - `/furetui/list/x2m/get`

- `/furetui/field/x2m/get/views`
- `/furetui/data/create`
- `/furetui/data/read`
- `/furetui/data/read/<dataId>`
- `/furetui/data/update`
- `/furetui/data/delete`
- `/furetui/data/search`
- *Default route for login and logout*
 - * `/furetui/client/login`
 - * `/furetui/client/logout`

CHAPTER 6

Client / Server Api

All the route use the **POST** method and the **JSON** rendering. They must all return list of store mutation. See the store page:

```
[
  ``store mutation``
    OR / AND
  ``translation definition``
    OR / AND
  ``locale``
    OR / AND
  ``client path for vue-router``
]
```

/furetui/init/required/data

First query called by FuretUI. The goal of this query is to initialize the client:

Warning: Return quick data, allow to display quickly the UI

/furetui/init/optionnal/data

Second call. The goal is to load heavy data to improve the client and allow a degradate mode

Note: The query can be slower than required data because the main data is already present and this application may be usable.

/furetui/homepage

Called, when the router path is /

/furetui/button/<buttonId>

Called, when the button is clicked with params in body:

- viewId
- model
- dataIds
- options

/furetui/custom/view/<viewName>

Called when the custom view is called, is not required to respond to this query

/furetui/view/<viewId>

Called when the view have been selected or changed.

/furetui/space/<spaceId>

Called when the view space been selected or changed.

optionnal values in body:

- spaceId
- menuId
- actionId
- viewId
- dataId
- mode

/furetui/action/<actionId>

Called when the view action been selected or changed.

optionnal values in body:

- spaceId
- menuId

- `actionId`
- `viewId`
- `dataId`
- `mode`

`/furetui/field/x2x/search`

Call to by Many2One and Many2ManyTags to search the remote value
the value in body are:

- `model`
- `value`
- `fields`

Warning: the wanted result is not a classical response

```
{
  ids: [ data id, ...
    ], data: [
      store mutation, OR / AND
      translation definition OR / AND
      locale OR / AND
      client path for vue-router
    ],
}
```

`/furetui/list/x2m/get`

Call to by One2Many and Many2Many to get children values
the value in body are:

- `model`
- `viewId`
- `dataIds`

`/furetui/field/x2m/get/views`

Call to by One2Many and Many2Many to get the children views
the value in body:

- `viewIds`

`/furetui/data/create`

Call to save an unexisting data

the value in body are:

- model
- data
- changes: all the change not only for this data
- fields
- **path: optionnal**
 - spaceId
 - menuId
 - actionId
 - viewId

Warning: Don't forget to change path to pass in RO mode

`/furetui/data/read`

Call to read entries in function of filter

the value in body are:

- model
- viewId
- filter
- fields

`/furetui/data/read/<dataId>`

Call to read only one entry

the value in body are:

- model
- viewId
- new
- fields

/furetui/data/update

Call to save an existing data

the value in body are:

- model
- dataId
- data
- changes: all the change not only for this data
- fields
- **path: optionnal**
 - spaceId
 - menuId
 - actionId
 - viewId

Warning: Don't forget to change path to use the good dataId and RO mode

/furetui/data/delete

Call to remove an existing data

the value in body are:

- model
- dataIds

Warning: Don't forget to call the mutation to remove the dataIds also on furetui client

/furetui/data/search

Call to remove an existing data

the value in body are:

- model
- search
- value

Warning: It is not a classical response, the goal it to define the available filter for search view

Default route for login and logout

The view can be overwriting this route can be deprecated.

`/furetui/client/login`

Call when the FuretUI user click on the `connection` button. The server may check the identity of the user and this access righth. The server:

`/furetui/client/logout`

Call when the FuretUI user click on the `Logout` thumbnail. The server may close the session of the user. The server:

Warning: Dont forget to remove all the data

Contents

- *Template*

CHAPTER 7

Template

The templates are used by `Form` and `Thumbnail` views, the template describe the view to display. The template come from the server and must apply some rules.

Contents

- *RELEASE*
 - *1.2.0 (2017-07-23)*
 - *1.1.0 (2017-07-22)*
 - *1.0.0 (2017-07-22)*

1.2.0 (2017-07-23)

- Add two / way instance creation to allowed import oter component
- Add documentation to start client

1.1.0 (2017-07-22)

- Fix some documentation
- Rename Text to RichText
- Add Field Text

1.0.0 (2017-07-22)

First release:

- Add some fields
- Add some views
- Add notifications
- Add vuejs, vuex, vue-i18n, vue-router
- Add Bulma, buefy

Contents

- | |
|---|
| <ul style="list-style-type: none">• <i>Mozilla Public License Version 2.0</i> |
|---|

- 1. *Definitions*
 - * 1.1. *“Contributor”*
 - * 1.2. *“Contributor Version”*
 - * 1.3. *“Contribution”*
 - * 1.4. *“Covered Software”*
 - * 1.5. *“Incompatible With Secondary Licenses”*
 - * 1.6. *“Executable Form”*
 - * 1.7. *“Larger Work”*
 - * 1.8. *“License”*
 - * 1.9. *“Licensable”*
 - * 1.10. *“Modifications”*
 - * 1.11. *“Patent Claims” of a Contributor*
 - * 1.12. *“Secondary License”*
 - * 1.13. *“Source Code Form”*
 - * 1.14. *“You” (or “Your”)*
- 2. *License Grants and Conditions*
 - * 2.1. *Grants*
 - * 2.2. *Effective Date*
 - * 2.3. *Limitations on Grant Scope*
 - * 2.4. *Subsequent Licenses*
 - * 2.5. *Representation*
 - * 2.6. *Fair Use*
 - * 2.7. *Conditions*
- 3. *Responsibilities*
 - * 3.1. *Distribution of Source Form*
 - * 3.2. *Distribution of Executable Form*
 - * 3.3. *Distribution of a Larger Work*
 - * 3.4. *Notices*
 - * 3.5. *Application of Additional Terms*
- 4. *Inability to Comply Due to Statute or Regulation*
- 5. *Termination*
 - * 5.1.
 - * 5.2.
 - * 5.3.
- 6. *Disclaimer of Warranty*

- *7. Limitation of Liability*
- *8. Litigation*
- *9. Miscellaneous*
- *10. Versions of the License*
 - * *10.1. New Versions*
 - * *10.2. Effect of New Versions*
 - * *10.3. Modified Versions*
 - * *10.4. Distributing Source Code Form that is Incompatible With Secondary Licenses*
- *Exhibit A - Source Code Form License Notice*
- *Exhibit B - “Incompatible With Secondary Licenses” Notice*

1. Definitions

1.1. “Contributor”

Means each individual or legal entity that creates, contributes to the creation of, or owns Covered Software.

1.2. “Contributor Version”

Means the combination of the Contributions of others (if any) used by a Contributor and that particular Contributor’s Contribution.

1.3. “Contribution”

Means Covered Software of a particular Contributor.

1.4. “Covered Software”

Means Source Code Form to which the initial Contributor has attached the notice in Exhibit A, the Executable Form of such Source Code Form, and Modifications of such Source Code Form, in each case including portions thereof.

1.5. “Incompatible With Secondary Licenses”

Means:

- **That the initial Contributor has attached the notice described in Exhibit B** to the Covered Software; or
- **That the Covered Software was made available under the terms of version 1.1** or earlier of the License, but not also under the terms of a Secondary License.

1.6. “Executable Form”

Means any form of the work other than Source Code Form.

1.7. “Larger Work”

Means a work that combines Covered Software with other material, in a separate file or files, that is not Covered Software.

1.8. “License”

Means this document.

1.9. “Licensable”

Means having the right to grant, to the maximum extent possible, whether at the time of the initial grant or subsequently, any and all of the rights conveyed by this License.

1.10. “Modifications”

Means any of the following:

- **Any file in Source Code Form that results from an addition to, deletion from,** or modification of the contents of Covered Software; or
- Any new file in Source Code Form that contains any Covered Software.

1.11. “Patent Claims” of a Contributor

Means any patent claim(s), including without limitation, method, process, and apparatus claims, in any patent Licensable by such Contributor that would be infringed, but for the grant of the License, by the making, using, selling, offering for sale, having made, import, or transfer of either its Contributions or its Contributor Version.

1.12. “Secondary License”

Means either the GNU General Public License, Version 2.0, the GNU Lesser General Public License, Version 2.1, the GNU Affero General Public License, Version 3.0, or any later versions of those licenses.

1.13. “Source Code Form”

Means the form of the work preferred for making modifications.

1.14. “You” (or “Your”)

Means an individual or a legal entity exercising rights under this License. For legal entities, “You” includes any entity that controls, is controlled by, or is under common control with You. For purposes of this definition, “control” means (a) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (b) ownership of more than fifty percent (50%) of the outstanding shares or beneficial ownership of such entity.

2. License Grants and Conditions

2.1. Grants

Each Contributor hereby grants You a world-wide, royalty-free, non-exclusive license:

- **Under intellectual property rights (other than patent or trademark)** Licensable by such Contributor to use, reproduce, make available, modify, display, perform, distribute, and otherwise exploit its Contributions, either on an unmodified basis, with Modifications, or as part of a Larger Work; and
- **Under Patent Claims of such Contributor to make, use, sell, offer for sale,** have made, import, and otherwise transfer either its Contributions or its Contributor Version.

2.2. Effective Date

The licenses granted in Section 2.1 with respect to any Contribution become effective for each Contribution on the date the Contributor first distributes such Contribution.

2.3. Limitations on Grant Scope

The licenses granted in this Section 2 are the only rights granted under this License. No additional rights or licenses will be implied from the distribution or licensing of Covered Software under this License. Notwithstanding Section 2.1(b) above, no patent license is granted by a Contributor:

- For any code that a Contributor has removed from Covered Software; or
- **For infringements caused by: (i) Your and any other third party's** modifications of Covered Software, or (ii) the combination of its Contributions with other software (except as part of its Contributor Version); or
- **Under Patent Claims infringed by Covered Software in the absence of its** Contributions.

This License does not grant any rights in the trademarks, service marks, or logos of any Contributor (except as may be necessary to comply with the notice requirements in Section 3.4).

2.4. Subsequent Licenses

No Contributor makes additional grants as a result of Your choice to distribute the Covered Software under a subsequent version of this License (see Section 10.2) or under the terms of a Secondary License (if permitted under the terms of Section 3.3).

2.5. Representation

Each Contributor represents that the Contributor believes its Contributions are its original creation(s) or it has sufficient rights to grant the rights to its Contributions conveyed by this License.

2.6. Fair Use

This License is not intended to limit any rights You have under applicable copyright doctrines of fair use, fair dealing, or other equivalents.

2.7. Conditions

Sections 3.1, 3.2, 3.3, and 3.4 are conditions of the licenses granted in Section 2.1.

3. Responsibilities

3.1. Distribution of Source Form

All distribution of Covered Software in Source Code Form, including any Modifications that You create or to which You contribute, must be under the terms of this License. You must inform recipients that the Source Code Form of the Covered Software is governed by the terms of this License, and how they can obtain a copy of this License. You may not attempt to alter or restrict the recipients' rights in the Source Code Form.

3.2. Distribution of Executable Form

If You distribute Covered Software in Executable Form then:

- **Such Covered Software must also be made available in Source Code Form, as** described in Section 3.1, and You must inform recipients of the Executable Form how they can obtain a copy of such Source Code Form by reasonable means in a timely manner, at a charge no more than the cost of distribution to the recipient; and
- **You may distribute such Executable Form under the terms of this License, or** sublicense it under different terms, provided that the license for the Executable Form does not attempt to limit or alter the recipients' rights in the Source Code Form under this License.

3.3. Distribution of a Larger Work

You may create and distribute a Larger Work under terms of Your choice, provided that You also comply with the requirements of this License for the Covered Software. If the Larger Work is a combination of Covered Software with a work governed by one or more Secondary Licenses, and the Covered Software is not Incompatible With Secondary Licenses, this License permits You to additionally distribute such Covered Software under the terms of such Secondary License(s), so that the recipient of the Larger Work may, at their option, further distribute the Covered Software under the terms of either this License or such Secondary License(s).

3.4. Notices

You may not remove or alter the substance of any license notices (including copyright notices, patent notices, disclaimers of warranty, or limitations of liability) contained within the Source Code Form of the Covered Software, except that You may alter any license notices to the extent required to remedy known factual inaccuracies.

3.5. Application of Additional Terms

You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Software. However, You may do so only on Your own behalf, and not on behalf of any Contributor. You must make it absolutely clear that any such warranty, support, indemnity, or liability obligation is offered by You alone, and You hereby agree to indemnify every Contributor for any liability incurred by such Contributor as a result of warranty, support, indemnity or liability terms You offer. You may include additional disclaimers of warranty and limitations of liability specific to any jurisdiction.

4. Inability to Comply Due to Statute or Regulation

If it is impossible for You to comply with any of the terms of this License with respect to some or all of the Covered Software due to statute, judicial order, or regulation then You must: (a) comply with the terms of this License to the maximum extent possible; and (b) describe the limitations and the code they affect. Such description must be placed in a text file included with all distributions of the Covered Software under this License. Except to the extent prohibited by statute or regulation, such description must be sufficiently detailed for a recipient of ordinary skill to be able to understand it.

5. Termination

5.1.

The rights granted under this License will terminate automatically if You fail to comply with any of its terms. However, if You become compliant, then the rights granted under this License from a particular Contributor are reinstated (a) provisionally, unless and until such Contributor explicitly and finally terminates Your grants, and (b) on an ongoing basis, if such Contributor fails to notify You of the non-compliance by some reasonable means prior to 60 days after You have come back into compliance. Moreover, Your grants from a particular Contributor are reinstated on an ongoing basis if such Contributor notifies You of the non-compliance by some reasonable means, this is the first time You have received notice of non-compliance with this License from such Contributor, and You become compliant prior to 30 days after Your receipt of the notice.

5.2.

If You initiate litigation against any entity by asserting a patent infringement claim (excluding declaratory judgment actions, counter-claims, and cross-claims) alleging that a Contributor Version directly or indirectly infringes any patent, then the rights granted to You by any and all Contributors for the Covered Software under Section 2.1 of this License shall terminate.

5.3.

In the event of termination under Sections 5.1 or 5.2 above, all end user license agreements (excluding distributors and resellers) which have been validly granted by You or Your distributors under this License prior to termination shall survive termination.

6. Disclaimer of Warranty

Warning: Covered Software is provided under this License on an “as is” basis, without warranty of any kind, either expressed, implied, or statutory, including, without limitation, warranties that the Covered Software is free of defects, merchantable, fit for a particular purpose or non-infringing. The entire risk as to the quality and performance of the Covered Software is with You. Should any Covered Software prove defective in any respect, You (not any Contributor) assume the cost of any necessary servicing, repair, or correction. This disclaimer of warranty constitutes an essential part of this License. No use of any Covered Software is authorized under this License except under this disclaimer.

7. Limitation of Liability

Warning: Under no circumstances and under no legal theory, whether tort (including negligence), contract, or otherwise, shall any Contributor, or anyone who distributes Covered Software as permitted above, be liable to You for any direct, indirect, special, incidental, or consequential damages of any character including, without limitation, damages for lost profits, loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses, even if such party shall have been informed of the possibility of such damages. This limitation of liability shall not apply to liability for death or personal injury resulting from such party's negligence to the extent applicable law prohibits such limitation. Some jurisdictions do not allow the exclusion or limitation of incidental or consequential damages, so this exclusion and limitation may not apply to You.

8. Litigation

Any litigation relating to this License may be brought only in the courts of a jurisdiction where the defendant maintains its principal place of business and such litigation shall be governed by laws of that jurisdiction, without reference to its conflict-of-law provisions. Nothing in this Section shall prevent a party's ability to bring cross-claims or counter-claims.

9. Miscellaneous

This License represents the complete agreement concerning the subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. Any law or regulation which provides that the language of a contract shall be construed against the drafter shall not be used to construe this License against a Contributor.

10. Versions of the License

10.1. New Versions

Mozilla Foundation is the license steward. Except as provided in Section 10.3, no one other than the license steward has the right to modify or publish new versions of this License. Each version will be given a distinguishing version number.

10.2. Effect of New Versions

You may distribute the Covered Software under the terms of the version of the License under which You originally received the Covered Software, or under the terms of any subsequent version published by the license steward.

10.3. Modified Versions

If you create software not governed by this License, and you want to create a new license for such software, you may create and use a modified version of this License if you rename the license and remove any references to the name of the license steward (except to note that such modified license differs from this License).

10.4. Distributing Source Code Form that is Incompatible With Secondary Licenses

If You choose to distribute Source Code Form that is Incompatible With Secondary Licenses under the terms of this version of the License, the notice described in Exhibit B of this License must be attached.

Exhibit A - Source Code Form License Notice

This Source Code Form **is** subject to the terms of the Mozilla Public License, v. 2.0. If a copy of the MPL was **not** distributed **with** this file, You can obtain one at <http://mozilla.org/MPL/2.0/>.

If it is not possible or desirable to put the notice in a particular file, then You may include the notice in a location (such as a LICENSE file in a relevant directory) where a recipient would be likely to look for such a notice.

Note: You may add additional accurate notices of copyright ownership.

Exhibit B - “Incompatible With Secondary Licenses” Notice

This Source Code Form is “Incompatible With Secondary Licenses”, as defined by the Mozilla Public License, v. 2.0.