UNICORN

Naming conventions for uu5 components.

Outline

- Naming conventions
- Component example

Referencies

- UAF – Naming&Code Conventions

Calls	Template	Example
POST call - method name	{methodName}	addItem
GET call - method name	get{MethodName}	getItem
	list{MethodName}	listItems
File names, routes	Template	Example
folders/src modules	{multi_word_name}	/main/src/bricks/, /main/src/uu-forum/
routes	/{name}	/home, /app-info
JS/JSX files	{multi-word-name}.js	data-grid.js
CSS files	{multi-word-name}.css	data-grid.css
LESS files	{multi-word-name}.less	data-grid.less
VUC	Template	Example
statics.tagName	{Vendor}.{SubApp}.Vuc{Name}	UU.MyApp.VucHome
statics.vucName	{name}	home
statics.classNames.main	{vendor}-{subapp}-vuc{name}	uu-myapp-vuchome
Bricks	Template	Example
statics.tagName	{Vendor}.{SubApp}.{Name}	UU.Trainee.TodoList
statics.classNames.main	{vendor}-{subapp}-{name}	uu-trainee-todolist
export name	{Name}	export default TodoList
custom classNames	{bricks|element}-{context}(-{additional_context})	todo-text-input
props / states	{propertyOrState}	items, labelPosition
interface class methods	{methodName}	getItems()
private class methods	_{methodName}	_getItems()
overriding class methods	{methodName}_	onLoadSuccess_()
variables	{variableName}	let dtoIn;

Prefere listItems over getItems, getItem over loadItem (load is reserved word in Ruby)

Prefere CRUD names - add over insert, update over modify, delete over remove.

Interface methods should always return a value. If there is no value to return, it should return component itself through `return this`.

Component example

```
//@@viewOn:imports
import React from "react";
import UU5 from 'uu5';
import Common from "../common/common.js";
import "./home.css";
```

```
//@@viewOff:imports

const VucHome = React.createClass({
 //@@viewOn:mixins
 mixins: [
   UU5.Common.BaseMixin,
   UU5.Common.VucMixin
 ],
 //@@viewOff:mixins

 //@@viewOn:statics
 statics: {
  tagName: 'UU.AppWorkspace.VucHome',
  vucName: 'home',
  classNames: {
   main: 'uu-appworkspace-vuchome'
  },
  // vucName: 'Info' // same for all languages or like lsi object
  vucName: {
   en:"Home",
   cs:"Úvod"
  }
 },
 //@@viewOff:statics

 //@@viewOn:propTypes
 //@@viewOff:propTypes

 //@@viewOn:getDefaultProps
 //@@viewOff:getDefaultProps

 //@@viewOn:standardComponentLifeCycle
 //@@viewOff:standardComponentLifeCycle

 //@@viewOn:interface
 //@@viewOff:interface

 //@@viewOn:componentSpecificHelpers
 //@@viewOff:componentSpecificHelpers

 //@@viewOn:render
 render() {
  return (
   <UU5.Layout.Root
    header={{
     element: (
      <Common.Header
       labelLsi={this.constructor.vucNameLsi}
       menuProfileCodes={this.state.profiles}
       colorSchema="primary"
      />
     )
    }}
    footer={{ element:<Common.Footer/> }}
   >
    <Common.UnderConstruction />
   </UU5.Layout.Root>
  );
 }
 //@@viewOff:render
});

export default VucHome;
```