GuaranaUI View Object

see article GuaranaUI-Overview for the overview

Inheritance

```
Nokia.View
    |- Nokia.Class (basic class)
```

1 - Description

Basic class to build views using GuaranaUI.

2 - Visual Design

Not Applicable for this Component.

3 - Component Dependencies

3.1 CSS

Not Applicable for this Component.

3.2 Javascript

```
<!-- jQuery file -->
<script src="/lib/jquery/jquery.js" type="text/javascript" charset="utf-8"></script>
<!-- Guarana file -->
<script src="/lib/Guarana.js" type="text/javascript" charset="utf-8"></script>
```

4 - Functional Specifications/Requirements

4.1 - Options

- container: 'selector'
  - jQuery selector or the DOM reference to become the container of the component.

4.2 - Events - Callbacks

- show: function(arguments)
  - Scope: [this.element]
- hide: function(arguments)
  - Scope: [this.element]

4.3 - Methods

- init: function()
  - View initializer
  - return void
- fireCallback: function(type)
  - View Object function to call view callbacks
  - return void
- renderUI: function()
  - This defines renderUI function for the view. More details in view lifecycle.
  - return void
- bindUI: function()
  - This defines bindUI function for the view. More details in view lifecycle.

(C) Copyright Nokia 2013. All rights reserved.

  - return void
- syncUI: function()
  - This defines the syncUI function for the view. More details in view lifecycle.
  - return void
- setContainer: function()
  - This sets the view container.
  - return void
- getContainer: function()
  - This gets the view container.
  - return [jQuery Object]
- setTemplate: function()
  - This sets a template for the view.
  - return void
- getContainer: function()
  - This gets the template for the view.
  - return [jQuery collection]
- isRendered: function()
  - This returns true if view is already rendered.
  - return boolean
- show: function()
  - This defines the show function for the view. More details in view lifecycle.
  - return void
- hide: function()
  - This defines the hide function for the view. More details in view lifecycle.
  - return void
- hide: function()
  - This defines the hide function for the view. More details in view lifecycle.
  - return void
- destroy: function()
  - This defines the destroy function for the view
  - return void

4.4 - Default options

Not Applicable for this Component.

5 - The Guarana View Lifecycle

The Guarana UI has a lifecycle approach for views objects.

Guarana View Class provides a renderer method implementation, for each view this need to be over-ridden. This render method is shown below:

```
render: function() {
    this.renderUI();
    this.bindUI();
    this.syncUI();
}
```

The renderUI, bindUI and syncUI are abstract (empty) methods in the view class. The intended role of each of these methods is described below, and most widgets will simply implement these methods based on their expected roles:

- **renderUI**: This method is responsible for creating and adding the HTML nodes which the view needs. It is usually the point at which the DOM is first modified by the view.
- **bindUI**: This method is responsible for attaching event listeners which bind the UI to the view.
- **syncUI**: This method is responsible for synchronizing the state of the UI based on the current state of the view.

6 - Component Demo

6.1 HTML

```
<div id="myview"></div>
```

5.2 Javascript

```
/*
* User Impl
* - First View
*/

var myFirstView = Nokia.View.extend({
      /*
       * Lifecycle
```


(C) Copyright Nokia 2013. All rights reserved.

```
 */
      init: function() {
            alert("initialize FirstView view");
      },

      // Developer can put HTML Creation here
      renderUI: function() {

            this._title = Nokia.dom.parseHTML('<span>Nokia Developer</span>');

            Nokia.dom.append(this.getContainer(), this._title);
      },

      // Developer can put events binding here
      bindUI: function() {

            this._title.click(function() {

                  alert('Clicked on Header');

            });
      },
      // Developer can put syncUI rules here
      syncUI: function() {

            this._title.css('color', 'red');

      },

      show: function() {

            alert('show FirstView');

      }
});

//Creating and rendering the view

var firstView = new myFirstView ({
      container: '#myview'
});

firstView.render();
```


(C) Copyright Nokia 2013. All rights reserved.

(C) Copyright Nokia 2013. All rights reserved.