

# Evaluation – Element-Frameworks Projekt: DigitalSchoolNotes

**Projekt Team:** Adler, Brinnich, Hohenwarter, Karic, Stedronsky

**Version 2.0**

**12.10.2015**

**Status: [RELEASE]**

	Datum	Name	Unterschrift
Erstellt	05.10.2015	ThomasStedronsky	
Geprüft	12.10.2015	Thomas Stedronsky	
Freigegeben			
<b>Git-Pfad:</b> /doc/evaluation <b>Dokument:</b> evaluation_element.doc			

# 1 Changelog

Version	Datum	Status	Bearbeiter	Kommentar
0.1	2015-10-01	Erstellt	Thomas Stedronsky	Dokument erstellt
1.0	2015-10-07	Geprüft	Selina Brinnich	QA
1.1	2015-10-12	Bearbeitet	Adin Karic	Formatierung, Fehler ausgebessert, Abbildungsverzeichnis
2.0	2015-10-12	Geprüft	Thomas Stedronsky	QA

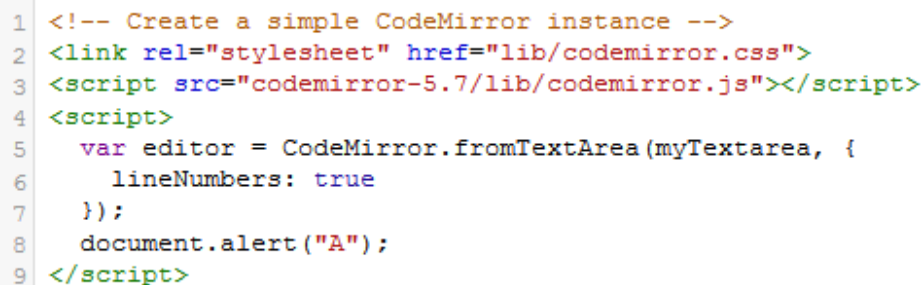
## 2 Programmcode Element CodeMirror

*„CodeMirror is a versatile text editor implemented in JavaScript for the browser. It is specialized for editing code, and comes with a number of language modes and addons that implement more advanced editing functionality.*

*A rich programming API and a CSS theming system are available for customizing CodeMirror to fit your application, and extending it with new functionality.”[1][2]*

Hier kann man die Funktionsweise von CodeMirror (das Syntax-Highlighting) sehen:

```
<form style="position: relative; margin-top: .5em;">
  <textarea id=demotext>
    <!-- Create a simple CodeMirror instance -->
    <link rel="stylesheet" href="lib/codemirror.css">
    <script src="codemirror-5.7/lib/codemirror.js"></script>
    <script>
      var editor = CodeMirror.fromTextArea(myTextarea, {
        lineNumbers: true
      });
    </script>
  </textarea>
```



```
1 <!-- Create a simple CodeMirror instance -->
2 <link rel="stylesheet" href="lib/codemirror.css">
3 <script src="codemirror-5.7/lib/codemirror.js"></script>
4 <script>
5   var editor = CodeMirror.fromTextArea(myTextarea, {
6     lineNumbers: true
7   });
8   document.alert("A");
9 </script>
```

**Figure 1: CodeMirror Funktionsweise**

Durch die Einbindung des Frameworks ist es möglich ein Textfeld einzubinden, welches erlaubt Zeilennummern einzublenden. Der eingegebene Text wird hierbei automatisch hervorgehoben.

Es werden diverse Sprachen unterstützt, man muss lediglich den Mode wechseln, um die jeweilige Sprache hervorzuheben.

### Browser Support:

- Firefox Version 4 und höher
- Chrome jede Version
- Safari Version 5.2 und höher
- Internet Explorer Version 8 und höher
- Opera Version 9 und höher

Dieses Framework läuft unter der MIT Lizenz und darf somit kommerziell genutzt werden.

### 3 Skizzen Element Sketch.js

Dieses Tool kann mittels JavaScript eingebunden werden. Durch diverse Parameter können Farbe und Größe des Pinsels eingestellt werden. Außerdem ermöglicht das Framework einen reibungslosen Download der angefertigten Zeichnung. Dies geschieht in diversen Formaten. (In diesem Beispiel PNG)[3][4]

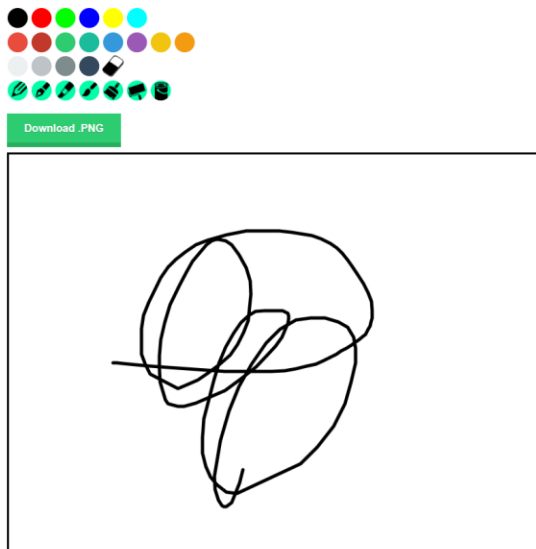


Figure 2: Sketch.js Funktionsweise

#### Browser Support:

- Firefox
- Chrome
- Safari
- Internet Explorer
- Opera

Dieses Framework ist für kommerzielle Nutzung freigegeben. [4]

```
<a href="#SketchPad" data-color="#000000" title="Black">
  
</a>
<a href="#SketchPad" data-size="3">
  
</a>
<canvas id="SketchPad" width="800" height="600"></canvas>
<script type="text/javascript">
  $(function() {
    $('#SketchPad').sketch();
  });
</script>
<a href="#SketchPad" data-download='png' id="DownloadPng">
  Download .PNG
</a>
```

## 4 CKEditor

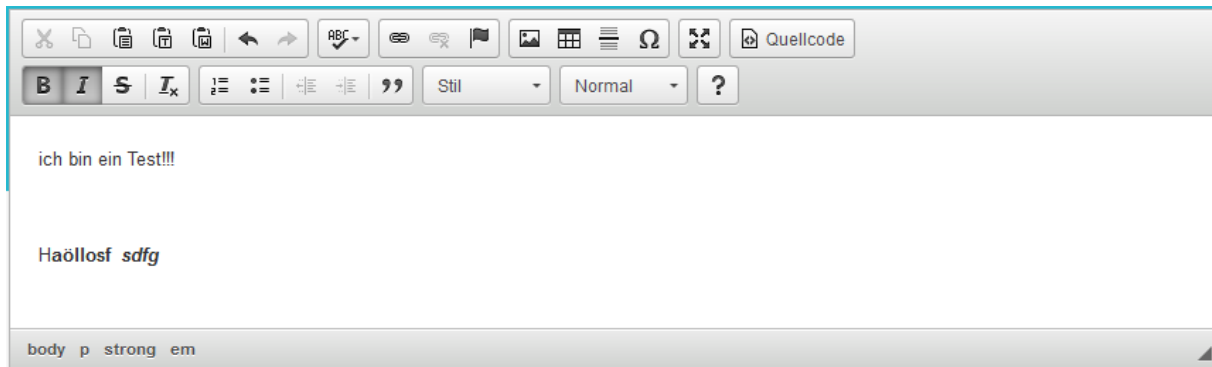


Figure 3: CKEditor Funktionsweise

Der Texteditor lässt sich sehr leicht in die Website einbinden und verwenden. Durch dieses Framework ist es möglich einen Textblock zu erstellen und diesen individuell zu bearbeiten. (bold, italic, underline, ...)[5][6]

Außerdem gibt es die Möglichkeit, seinen eigenen *CKEditor* zu erstellen. Die Website erlaubt es in 3 Schritten den Editor für die eigenen Bedürfnisse anzupassen.

```
<div class="adjoined-bottom">
  <div class="grid-container">
    <div class="grid-width-100">
      <div id="editor">
        <h1>Hello world!</h1>
        <p>Text </p>
      </div>
    </div>
  </div>
</div>
```

Der Editor wird mit der ID Editor gekennzeichnet und durch diese ID wird ein Editor Fenster erzeugt.

### Browser Support:

- Firefox
- Chrome
- Safari
- Internet Explorer
- Opera

Dieses Produkt ist für kommerzielle Nutzung freigegeben. (GPL, LGPL, CDL, MPL)[5]

## 5 Photoswipe



**Figure 4: Photoswipe Galerie Ansicht**

Die Slideshow erlaubt Bilder in Vollbild anzuzeigen und durch diese zu navigieren. Dieses Framework hat gute Grundstrukturen braucht allerdings für unseren Anwendungsbereich kleine Abänderungen. [7][8]

### **Browser Support:**

- Firefox
- Chrome
- Safari
- Internet Explorer
- Opera

Dieses Produkt ist für kommerzielle Nutzung freigegeben. (MIT-Lizenz) [7]

## 6 Quellen

[1], CodeMirror, „CodeMirror“, <https://codemirror.net/>, zuletzt besucht: 05.10.2015

[2], CodeMirror, „Download CodeMirror“, <http://codemirror.net/codemirror.zip>, zuletzt besucht: 05.10.2015

[3], Filip Danic, „How to Make an HTML5 Drawing Board with Sketch.js and HTML5 Canvas“, <http://danicfilip.com/make-html5-drawing-board-sketch-js-html5-canvas-source-code/>, zuletzt besucht: 05.10.2015

[4], Michael Bleigh and Intridea, Inc., „Sketch.js - Simple Canvas-based Drawing for jQuery“, <http://intridea.github.io/sketch.js/>, zuletzt besucht am 05.10.2015

[5], CKSource, „CKEditor“, <http://ckeditor.com/>, zuletzt besucht: 06.10.2015

[6], CKSource, „Download CKEditor“, <http://ckeditor.com/download>, zuletzt besucht: 06.10.2015

[7], Dmitry Semenov, „PhotoSwipe: Responsive JavaScript Image Gallery“, <http://photoswipe.com/>, zuletzt besucht: 06.10.2015

[8], GitHub, „GitHub PhotoSwipe“, <https://github.com/dimsemenov/photoswipe>, zuletzt besucht: 06.10.2015

## 7 Abbildungsverzeichnis

Figure 1: CodeMirror Funktionsweise.....	3
Figure 2: Sketch.js Funktionsweise .....	4
Figure 3: CKEditor Funktionsweise.....	5
Figure 4: Photoswipe Galerie Ansicht .....	6