

# Enhancement to the Open edX HTML Editor

---

Ashutosh Sathe (email@example.com)

Srijan Roy Choudhury (email@example.com)

Pravallika Podugu (email@example.com)

June 17, 2018

## What is OpenEdx ?

An open source MOOC platform developed by professionals at Harvard and MIT. OpenEdx platform consists of modules called XBlocks. In short, EdX platform is "Xblock Runtime"

## Main Components :

- LMS
- CMS
- Catalog
- Analytics
- Notes API
- Ecommerce

## Why do we need HTML Editor ?

- Adding content to course
- Discussion Forums
- Email

## Solutions adopted by OpenEdx :

- TinyMCE : WYSIWYG
- CodeMirror : Advanced editor for TinyMCE
- WMD : Mainly Email and Discussion Forums

## Current versions :

- TinyMCE : 4.0.20
- CodeMirror : 3.27
- WMD : Unknown version from 2012

## Issues :

- Editors are very old
- HTML code is not well indented and it is very hard to edit
- Internal CSS is not supported

## Technologies

- **Python, Django** for web applications
- **MongoDB** database for courses
- **MySQL** for pre-learner data
- **Amazon S3/YouTube** for videos

## Observations

- Latest version of CodeMirror has good features like code folding, indentation etc.
- **CodeMirror-For-TiynMCE** : A plugin for TinyMCE to replace TinyMCE's raw HTML Editor by CodeMirror

# Our approach

## The indentation problem in raw HTML Editor

- Download the latest version of TinyMCE , CodeMirroa and CodeMirror-For-TinyMCE plugin
- Integrate and test the plugin on updated editors
- Use this updated editor as raw HTML Editor in OpenEdx system

## Expected Result

TinyMCE opening the instance of latest codemirror as it's raw HTML editor

## **Alternative approach to indentation solution**

- Trying other online editors such as brackets, quill, contenttools
- Porting the indentation logic from other editors such as Notepad++ to the current editor



# Our approach

## Supporting internal CSS

- Wrap the entire HTML content in an iframe
- Set the width of iframe to 100%
- Set the height of iframe equal to the height of content of iframe
- Set the border of iframe to zero/none
- Open all hyperlinks within the iframe in new tab

## Expected Result

An invisible iframe holding our HTML content

## Alternative approach for supporting internal CSS

- Use javascript to apply CSS
- Separate out CSS part from HTML content
- Wrap the entire content in a div with unique id
- Select each and every child of the div and apply styles to it

## Expected Result

An invisible iframe holding our HTML content

## Advantages of Approach #1 of both the problems

- Upgrading something is always better than integrating something new altogether
- The iframe approach also allows the course content creator to include third party CSS such as bootstrap, w3css which will make the course even more beautiful
- Wrapping data inside the iframes containerizes the data ensuring no mixing of CSS between 2 HTML blocks in LMS

# Working over Approach #1

## Steps for completing Approach #1

- Download the latest versions of both the editors
- Test each editor individually
- Integrate both the editors using *codemirror for tinymce* plugin. We call this integration as hybrid version of tinymce
- Test this hybrid editor
- Integrate this hybrid editor with django (similar steps must be done in Edx as well so we will get the idea where to look for code in backend Remove this afterwards)
- Test this django integration
- Set up a working environment of OpenEdx platform
- Integrate the hybrid editor with the OpenEdx platform
- Testing the complete product

## Summary of local django integration

- Most of the HTML tags and their attributes work
- Most of the CSS styles and attributes work
- CSS animations and external CSS such as bootstrap works
- Report of this integration can be found [on this link](#)
- Source code of this integration can be found here [on this link](#)
- Wrapping content inside iframe(for internal CSS) is implemented in django backend(Check source code for more info)

# Working over Approach #1

## Preparing for the integration

- We keep the TinyMCE at it's same version
- We pull our CodeMirror plugin in the *edx-platform/common/static/js/vendor* without touching the current CodeMirror folder
- We configure the TinyMCE to use updated CodeMirror
- We modify the backend (exact location still unknown) to integrate our CSS solution

# Working over Approach #1

## Getting a working environment

OpenEdx can be installed in one of the following 3 ways

- Devstack
- Fullstack
- Native Ubuntu Installation

For developers, as the name suggests, devstack is suggested

## Getting a devstack up and running

- Virtualbox + Vagrant system(Deprecated now)
- Docker based system

## Issues we faced

- A LOT OF PROXY CONFIGURATION ISSUES !
- Docker installation and configuration
- Building ReactJS assets
- Docker container issues



## Dealing with the issues

- **Proxy** : Just disable the proxy altogether and use a non-proxy network for downloading everything you need

## Dealing with the issues

- **Proxy** : Just disable the proxy altogether and use a non-proxy network for downloading everything you need
- **Docker installation** : Follow the steps given on the docker site very very carefully

## Dealing with the issues

- **Proxy** : Just disable the proxy altogether and use a non-proxy network for downloading everything you need
- **Docker installation** : Follow the steps given on the docker site very very carefully
- **Building ReactJS assets** : Explicitly configure "*babel-loader*" to build JSX

# Getting a working environment

## Dealing with the issues

- **Proxy** : Just disable the proxy altogether and use a non-proxy network for downloading everything you need
- **Docker installation** : Follow the steps given on the docker site very very carefully
- **Building ReactJS assets** : Explicitly configure "*babel-loader*" to build JSX
- **Docker container** : Set the environment variables very very carefully and preserve them

## Integrating our hybrid editor with OpenEdx Devstack

- The work on this is partially done
- We were successful in getting the devstack to open updated instance of CodeMirror
- The integration source code can be found [here](#)