

Enhancement to the Open edX HTML Editor

Ashutosh Sathe (2ashutoshbs@gmail.com)

Srijan Roy Choudhury (roychoudhursrijan442@gmail.com)

Pravallika Podugu (pravallikapodugu22@gmail.com)

June 18, 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 , CodeMirror 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 (Since EdX uses django as well)
- Test this django integration
- Set up a working environment of OpenEdx platform
- Integrate the hybrid editor with the OpenEdx platform
- Testing the complete product

Working over Approach #1

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 link \(1\)](#)
- Source code of this integration can be found here [on link \(2\)](#)
- Wrapping content inside iframe(for internal CSS) is implemented in django backend(Check source code for more info)

Links :

- I <https://docs.google.com/spreadsheets/d/1AJQWfAG2NljpXFZt2fnLPIKdhP2-q35KZ0DOmdb-F0w/edit?ts=5b2744d4#gid=0>
- II <https://github.com/ashutoshbsathe/codemirror-in-tinymce/tree/django-test>

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

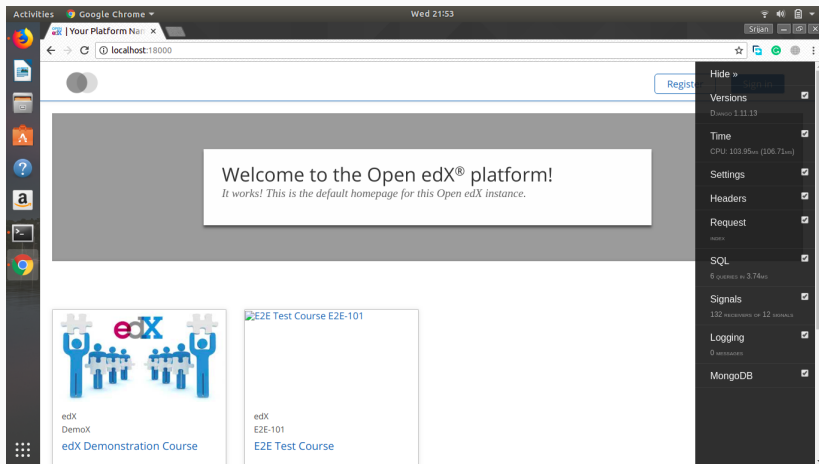
- **Proxy** : Just disable the proxy altogether and use a non-proxy network for downloading everything you need
- **Docker installation** : Follow the steps given on the docker site very very carefully
- **Building ReactJS assets** : Explicitly configure "*babel-loader*" to build JSX

Getting a working environment

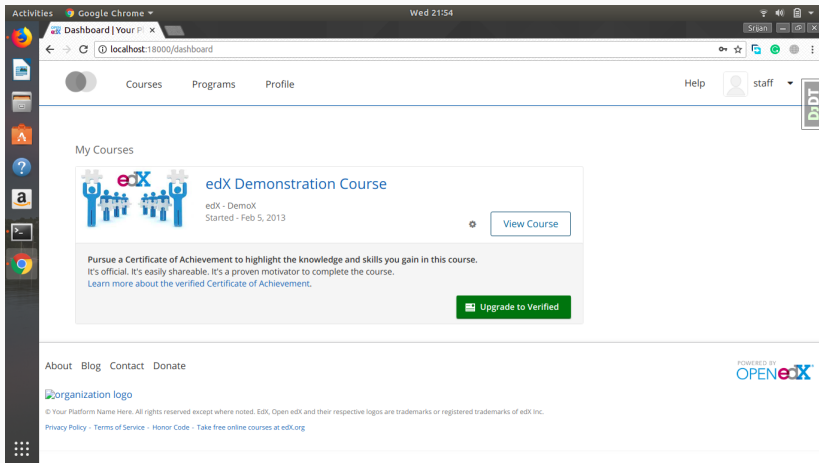
Dealing with the issues

- **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

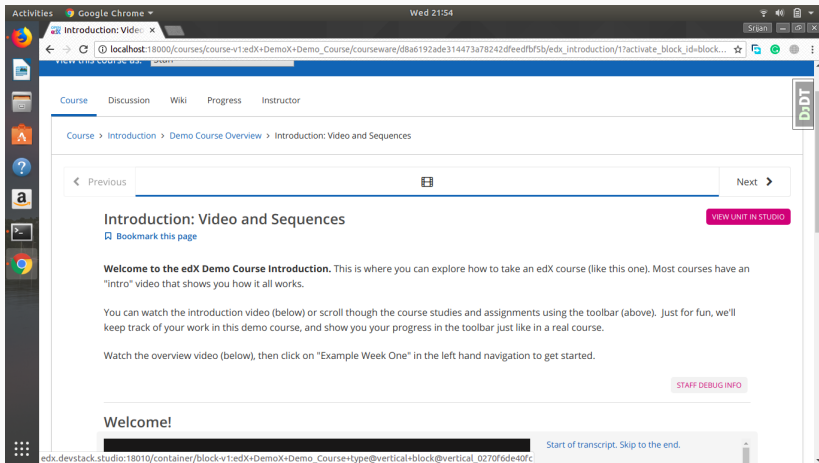
Screenshots of working devstack



Screenshots of working devstack



Screenshots of working devstack



Screenshots of working devstack

The screenshot shows a Google Chrome browser window at a local development URL. The page is titled "Introduction: Video and Sequences" and is in a "Draft" state. A caution message at the top states: "Caution: The last published version of this unit is live. By publishing changes you will change the student experience." The main content area is a "Blank HTML Page" with the following text:

Welcome to the edX Demo Course Introduction. This is where you can explore how to take an edX course (like this one). Most courses have an "intro" video that shows you how it all works.

You can watch the introduction video (below) or scroll through the course studies and assignments using the toolbar (above). Just for fun, we'll keep track of your work in this demo course, and show you your progress in the toolbar just like in a real course.

Watch the overview video (below), then click on "Example Week One" in the left hand navigation to get started.

Below this text is a "Welcome!" heading and a video player. The video player has a "Start of transcript. Skip to the end." button.

On the right side of the page, there is a "Draft (Unpublished changes)" panel. It shows the release date as "Jan 01, 1970 at 05:00 UTC" and lists the visibility settings as "Staff and Learners". There is a "Publish" button and a "Discard Changes" link.

Working over Approach #1

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 \(1\)](#)

Issues we face currently :

Our CSS styling is not getting saved by OpenEdx backend. We may need to discuss this issue with some backend developer

Links :

1. <https://github.com/ashutoshbsathe/tree/updated-codemirror>

Summary

Summary

- Devstack is a set of services

Summary

- Devstack is a set of services
- Each service corresponds to one docker container

Summary

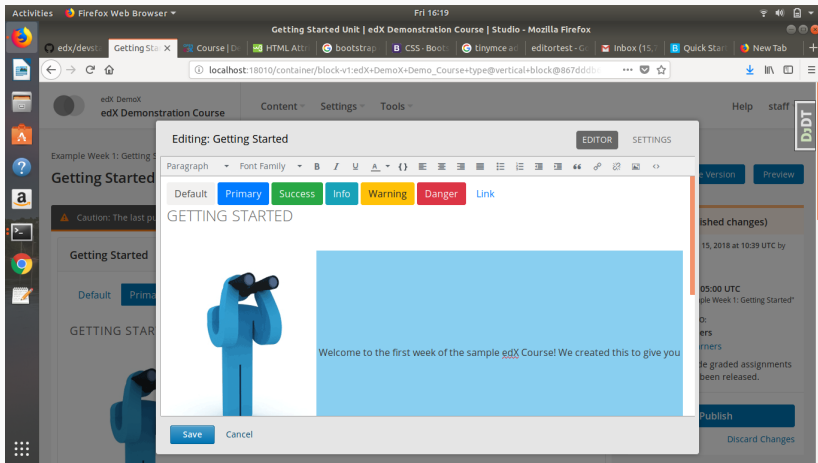
- Devstack is a set of services
- Each service corresponds to one docker container
- Each docker container is made up of docker image and docker volume

Summary

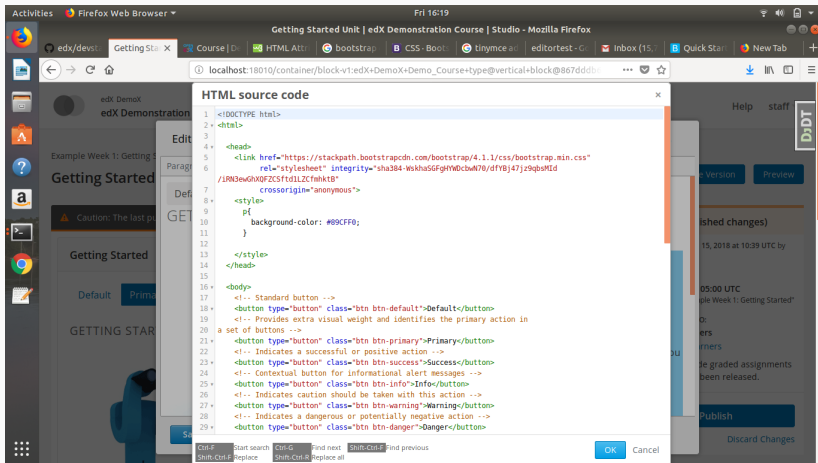
- Devstack is a set of services
- Each service corresponds to one docker container
- Each docker container is made up of docker image and docker volume
- Docker images are pulled from OpenEdx repositories

We also wanted to create a pictorial representation but couldn't do neatly, We will make sure to include that in final documentation

Screenshots of our issues



Screenshots of our issues



Screenshots of our issues

Activities Firefox Web Browser Fri 16:20

Getting Started Unit | edX Demonstration Course | Studio - Mozilla Firefox


edx/devst... GettingSta... Course | D... HTML Attr... bootstrap CSS-Boot... tinymce ad... editoritest - G... Inbox (15,7 B Quick Start New Tab +

localhost:18010/container/block-v1:edX+DemoX+Demo_Course+type@vertical+block@867ddb6f55d410caaa9c1eb9c6743ecr

Getting Started EDIT

Default Primary Success Info Warning Danger Link

GETTING STARTED



Welcome to the first week of the sample edX Course! We created this to give you "the basics" of how courses work at edX. Almost all edX courses have weeks, lessons, lectures, homework assignments, and exams. This demo course has three example weeks – most courses last 4 to 12 weeks.

localhost:18010/container/block-v1:edX+D...+block@867ddb6f55d410caaa9c1eb9c6743ecr

Draft (Unpublished changes)

Draft saved on Jun 15, 2018 at 10:39 UTC by staff

RELEASE:
Feb 05, 2013 at 05:00 UTC
with Section "Example Week 1: Getting Started"

WILL BE VISIBLE TO:
Staff and Learners
☐ Hide from learners

Note: Do not hide graded assignments after they have been released.

Publish

Discard Changes

Previously published

Unit Location

LOCATION ID
867ddb6f55d410caaa9c1eb9c6743ec

To create a link to this unit from an HTML component in this course, enter "jump_to_id" as the URL value.

Future Work

- Consult OpenEdX backend developer
- Completing the integration of hybrid editor
- Rigorous testing of the completed product
- Report making

THANK YOU!

Questions ?