

Test Report created by Content Team for Enhancement of open-edX HTML editor(Before Integration)

Test Case Id	Test Case	Test Procedure	Expected Result	Actual Result	Status	Classification of Bugs	Comments	LINKS
TCT01.1 FAILED TAGS	check the working of audio tag in the HTML editor {<audio> </audio>}	1. run the tinymce editor on the django server. 2. open the raw html editor. 3. inside the body part of html add audio tag as in the format. <audio src="" "> </audio>	audio should be playing	no audio is being played	Fail	Major	for making Audio tag work we need to enable the media plugins in the tinymce init page	—
TCT01.2	check the working of video tag in an editor {<video> </video>}	1. run the tinymce editor on the django server. 2. open the raw html editor. 3. inside the body part of html add video tag as in the format. <video src="" "> </video>	video should be playing	no video is being played	Fail	Minor	for making video tag work we need to enable the media plugins in the tinymce init page. Also openedx platform has got an inbuilt video component so there is no need of video tag to work	—

TCT01.3	check the working of script tag in an editor {<script> </script>}	1. run the tinymce editor on the django server. 2. open the raw html editor. 3. inside the body part of html add audio tag as in the format. <audio src=" "> </audio>	internal java script should be enabled	internal JAVASCRIPT is not working only inline JAVASCRIPT is working	Fail	Critical	Possibility of a security threat if the script tag is enabled as it may casue redirection to a malicious site	—
TCT01.4	check the working of a checkbox input type in an editor {<input type="checkbox">	1. run the tinymce editor on the django server. 2. open the raw html editor. 3. inside the body part of html add navigation tag as in the format. <input type="checkbox" name="xyz" value="abc"> Sample string
	Should define a full functioning checkbox options	Works but is browser dependent	partial pass	Trivial	Doesn't work with Mozilla firefox	—

TCT01.5	check the working of select tag in an editor {<select > </select>}	1. run the tinymce editor on the django server. 2. open the raw html editor. 3. <select name="xyz"> <option value="abc">ABC</option> ></select>	Should define a full functioning dropdown list	Works but is browser dependent	partial pass	Trivial	Supported and wells work with Google Chrome, but it fails with Mozilla Firefox	–
TCT01.6	check the working of radio button input type in an editor {<input type="radio button">	1. run the tinymce editor on the django server. 2. open the raw html editor. 3. inside the body part of html add navigation tag as in the format. <input type="radio" name="gender" value="male" checked> Male
 <input type="radio" name="gender" value="female"> Female
	Should define a full functioning radio buttons	Works but is browser dependent	partial pass	Trivial	Supported and wells work with Google Chrome, but it fails with Mozilla Firefox	–

TCT01.7	check the working of <menu> and <menuitem> tag in the editor	<p>1. run the tinymce editor on the django server.</p> <p>2. open the raw html editor.</p> <p>3. inside the body part of html add menu and menuitem tag as in the format. <menu type="context" id="mymenu"> <menuitem label=" " onclick=" "></menuitem> </menu></p>	defines a command/menu item that the user can invoke from a popup menu.	Works but is browser dependent	partial pass	Trivial	Supported and wells work with Mozilla Firefox, but it fails with Google Chrome	—
TCT01.8	check the working of <track> tag in the editor	<p>1. run the tinymce editor on the django server.</p> <p>2. open the raw html editor.</p> <p>3. inside the body part of html add track tag as in the format. <video width=" " height=" " controls> <source src=" " type=" "> <track src=" " kind=" " srclang=" " label=" "> </video></p>	<track> tag specifies text tracks for media elements (<audio> and <video>).	no subtitles for a track are being visible.	fail	Minor	track tag works only if the media plugins are enabled. it is required only for the audio and video contents.	—

TCT01.9	check the working of <source> tag in the editor	1. run the tinymce editor on the django server. 2. open the raw html editor. 3. inside the body part of html add source tag as in the format. <audio controls> <source src=" " type=" "> </audio>	<i>The <source> tag is used to specify multiple media resources for media elements, such as <video>, <audio>, and <picture>.</i>	<i>working only for <picture> tag doesnt work with <audio> & <video> tag</i>	<i>partial pass</i>	<i>critical</i>	<i>source tag works for <audio> & <video> only if the media plugins are enabled. and it works pretty fine with the <picture> tag</i>	-
TCT02 WORKING TAGS	check the working of HTML5 tags in the editor	1. run the tinymce editor on the django server. 2. open the raw html editor. 3. inside the body part of html add the required code snippet	<i>Full functionality of the tags should be visible</i>	<i>Works as expected</i>	<i>pass</i>	<i>Nil</i>	<i>These attributes work as expected</i>	Link is here

<p>TCA03.1 FAILED ATTRIBUTES</p>	<p>check the working of ASYNC attribute in the editor</p>	<p>1. run the tinymce editor on the django server. 2. open the raw html editor. 3. inside the body part of html add async attribute as in the format. <script src=" " async> </script></p>	<p><i>Async attribute specifies that the script will be executed asynchronously as soon as it is available</i></p>	<p><i>working is not seen as external JAVASCRIPT cant be enabled</i></p>	<p><i>fail</i></p>	<p><i>critical</i></p>	<p><i>The async attribute is only for external scripts (and should only be used if the src attribute is present).</i></p>	<p>—</p>
<p>TCA03.2</p>	<p>check the working of AUTOPLAY attribute in the editor</p>	<p>1. run the tinymce editor on the django server. 2. open the raw html editor. 3. inside the body part of html add async attribute as in the format. <audio controls autoplay> </audio></p>	<p><i>on presence of autoplay attribute in the audio/video tag , the audio/video will automatically start playing as soon as it can do so without stopping</i></p>	<p><i>not working</i></p>	<p><i>fail</i></p>	<p><i>Minor</i></p>	<p><i>works when the media plugins are enabled cause it is used only with <audio> & <video></i></p>	<p>—</p>

TCA03.3	check the working of CONTROLS attribute in the editor	1. run the tinymce editor on the django server. 2. open the raw html editor. 3. inside the body part of html add controls attribute as in the format. <audio controls > </audio>	<i>it specifies that audio/video controls should be displayed.</i>	<i>not working</i>	<i>fail</i>	<i>Minor</i>	<i>works when the media plugins are enabled cause it is used only with <audio> & <video></i>	—
TCA03.4	check the working of DATA_* attribute in the editor	1. run the tinymce editor on the django server. 2. open the raw html editor. 3. inside the body part of html add data_* attribute	<i>used to store custom data private to the page or application.</i>	<i>not working</i>	<i>fail</i>	<i>Critical</i>	<i>functionality is seen when script tag works</i>	—
TCA03.5	check the working of DEFER attribute in the editor	1. run the tinymce editor on the django server. 2. open the raw html editor. 3. inside the body part of html add defer attribute <script src="dem0.js" defer></script>	<i>specifies that the script is executed when the page has finished parsing.</i>	<i>not working</i>	<i>fail</i>	<i>Critical</i>	<i>functionality is seen when internal script is enabled</i>	—

TCA03.6	check the working of DRAGGABLE attribute in the editor	1. run the tinymce editor on the django server. 2. open the raw html editor. 3. inside the body part of html add draggable attribute in the following format <p id=" " draggable="true" ondragstart=" "> ... </p>	<i>specifies whether an element is draggable or not.</i>	<i>not working</i>	<i>fail</i>	<i>Critical</i>	<i>functionality is seen when internal script is enabled</i>	–
TCA03.7	check the working of DROPZONE attribute in the editor	1. run the tinymce editor on the django server. 2. open the raw html editor. 3. inside the body part of html add dropzone attribute	<i>specifies whether the dragged data is copied, moved, or linked, when it is dropped on an element.</i>	<i>not working</i>	<i>fail</i>	<i>Critical</i>	<i>functionality is seen when internal script is enabled</i>	–

TCA03.8	check the working of FOR attribute in the editor	<p>1. run the tinymce editor on the django server.</p> <p>2. open the raw html editor.</p> <p>3. inside the body part of html add navigation tag as in the format. <code><form></code> <code><label for="male"></code><code>Male</code><code></label></code> <code></form></code></p>	<p>When used together with the <code><label></code> element, the for attribute specifies which form element a label is bound to.</p> <p>When used together with the <code><output></code> element, the for attribute specifies the relationship between the result of the calculation, and the elements used in the calculation.</p>	Works but is browser dependent	partial pass	Trivial	Supported and wells work with Google Chrome, but it fails with Mozilla Firefox	—
TCA03.9	check the working of LOOP attribute in the editor	<p>1. run the tinymce editor on the django server.</p> <p>2. open the raw html editor.</p> <p>3. inside the body part of html add loop attribute as in this format <code><audio controls loop></code><code></audio></code></p>	<p>it specifies that the audio will start over again, every time it is finished.</p>	not working	fail	Minor	functionality is seen when media plugins are enabled as this attribute is used with <code><audio></code> & <code><video></code>	—

TCA03.10	check the working of KIND attribute in the editor	1. run the tinymce editor on the django server. 2. open the raw html editor. 3. inside the body part of html add kind attribute as in this format <video><source src=" " type="video/ogg"> <track src=" " kind="subtitles" srclang=" " label=" "> </video>	<i>The kind attribute specifies the kind of text track</i>	<i>not working</i>	<i>fail</i>	<i>Minor</i>	<i>functionality is seen when media plugins are enabled as this attribute is used with <video></i>	—
TCA03.11	check the working of MEDIA attribute in the editor	1. run the tinymce editor on the django server. 2. open the raw html editor. 3. inside the body part of html add media attribute with the following tags <a> <area> <link> <source> <style>	<i>specifies what media/device the linked document is optimized for. This attribute is used to specify that the target URL is designed for special devices (like iPhone) , speech or print media.</i>	<i>works only with style tag</i>	<i>fail</i>	<i>Minor</i>	<i>functionality is seen when media plugins are enabled as this is dealt with print media</i>	—

TCA03.12	check the working of MUTED attribute in the editor	1. run the tinymce editor on the django server. 2. open the raw html editor. 3. inside the body part of html add kind attribute as in this format <video controls muted></video>	<i>it specifies that the audio output of the video should be muted</i>	<i>not working</i>	<i>fail</i>	<i>Minor</i>	<i>functionality is seen when media plugins are enabled as this attribute is used with <video></i>	—
TCA03.13	check the working of POSTER attribute in the editor	1. run the tinymce editor on the django server. 2. open the raw html editor. 3. inside the body part of html add kind attribute as in this format <video controls poster></video>	<i>The poster attribute specifies an image to be shown while the video is downloading, or until the user hits the play button</i>	<i>not working</i>	<i>fail</i>	<i>Minor</i>	<i>functionality is seen when media plugins are enabled as this attribute is used with <video></i>	—

TCA03.14	check the working of PRELOAD attribute in the editor	1. run the tinymce editor on the django server. 2. open the raw html editor. 3. inside the body part of html add kind attribute as in this format <video controls preload></video> <audio controls preload></audio>	<i>The preload attribute specifies if and how the author thinks that the media file should be loaded when the page loads.</i>	<i>not working</i>	<i>fail</i>	<i>Minor</i>	<i>functionality is seen when media plugins are enabled as this attribute is used with <video> <audio></i>	–
TCA03.15	check the working of SRCLANG attribute in the editor	1. run the tinymce editor on the django server. 2. open the raw html editor. 3. inside the body part of html add kind attribute as in this format <video><source src=" " type="video/ogg"> <track src=" " kind="subtitles" srclang=" " label=" "> </video>	<i>The srclang attribute specifies the language of the track text data. This attribute is required if kind="subtitles".</i>	<i>not working</i>	<i>fail</i>	<i>Minor</i>	<i>functionality is seen when media plugins are enabled as this attribute is used with <video></i>	–

TCA03.16 script related	check the working of html5events attribute in the editor	1. run the tinymce editor on the django server. 2. open the raw html editor. 3. inside the body part of html add the required code snippet	<i>Full functionality of the attributes should be visible</i>	<i>not working</i>	<i>fail</i>	<i>Major</i>	<i>functionality is seen when script tag is enabled</i>	Link is here
TCA04 WORKING ATTRIBUTES	check the working of HTML5 attributes in the editor	1. run the tinymce editor on the django server. 2. open the raw html editor. 3. inside the body part of html add the required code snippet	<i>Full functionality of the attributes should be visible</i>	<i>Works as expected</i>	<i>pass</i>	<i>Nil</i>	<i>These attributes work as expected</i>	Link is here
TCC05 CSS	check the working of CSS in the editor	1. run the tinymce editor on the django server. 2. open the raw html editor. 3. inside the head part of html add the required code snippet	<i>Full functionality of the CSS should be visible</i>	<i>Works as expected</i>	<i>pass</i>	<i>Nil</i>	<i>CSS completely works fine</i>	Link is here