Ex\_submit.js

// document.addEventListener('DOMContentLoaded', function () {

// var tabButtons = document.querySelectorAll('.nav-link');

// var tabContent = document.querySelectorAll('.tab-pane input');

// var selectedTabIndex = 0;

// var prevBtn = document.getElementById("prev");

// var nextBtn = document.getElementById("next");

// var submitBtn = document.getElementById("submit");

// var form = document.getElementById("form");

// var inputFields = form.querySelectorAll("input");

// var title = document.getElementById('title');

// var keywords = document.getElementById('keywords');

// var abstract = document.getElementById('abstract');

// var reference = document.getElementById('reference');

// for (var i = 1; i < tabButtons.length; i++) {

// tabButtons[i].disabled = true;

// }

// function updateButtonVisibility(index) {

// // Display buttons based on the current tab index

// if (index === 0) {

// prevBtn.style.display = "none";

// nextBtn.style.display = "inline-block";

// submitBtn.style.display = "none";

// }else if(index === 1){

// prevBtn.style.display = "inline-block";

// nextBtn.disabled = true;

// submitBtn.style.display = "none";

// }

// else if (index > 0 && index < tabButtons.length - 1) {

// prevBtn.style.display = "inline-block";

// nextBtn.style.display = "inline-block";

// submitBtn.style.display = "none";

// } else if (index === tabButtons.length - 1) {

// prevBtn.style.display = "inline-block";

// nextBtn.style.display = "none";

// submitBtn.style.display = "inline-block";

// }

// }

// function updateStyles() {

// tabButtons.forEach(function (btn, i) {

// if (i === selectedTabIndex) {

// btn.style.backgroundColor = "#0858a4";

// btn.style.color = "white";

// } else {

// btn.style.backgroundColor = "white";

// btn.style.border = "1px solid";

// btn.style.color = "#0858a4";

// }

// });

// }

// function updateButtonStates(index) {

// for (var i = index + 1; i < tabButtons.length; i++) {

// tabButtons[i].disabled = tabContent[i - 1].value === '';

// }

// }

// function switchToNextTab() {

// if (selectedTabIndex < tabButtons.length - 1) {

// inputFields = tabContent[selectedTabIndex];

// if (inputFields.value === '' || tabButtons[selectedTabIndex + 1].disabled) {

// Swal.fire({

// html: '<h4 style="color: #0858a4; font-family: font-family: Arial, Helvetica, sans-serif">Please read and check the guidelines to proceed</4>',

// icon: 'warning',

// })

// }

// tabButtons[selectedTabIndex + 1].click(); // Simulate a click on the next tab button

// }

// }

// function switchToPrevTab() {

// if (selectedTabIndex > 0) {

// tabButtons[selectedTabIndex - 1].click(); // Simulate a click on the previous tab button

// }

// }

// tabContent.forEach(function (input, index) {

// input.addEventListener('input', function () {

// updateButtonStates(index);

// updateStyles();

// });

// input.addEventListener('focus', function () {

// updateStyles();

// });

// input.addEventListener('blur', function () {

// // Don't change styles on blur

// });

// });

// tabButtons.forEach(function (button, index) {

// button.addEventListener('click', function () {

// selectedTabIndex = index;

// updateButtonVisibility(index);

// updateButtonStates(index);

// updateStyles();

// });

// // Set initial styles and button visibility for the first button

// if (index === 0) {

// button.style.backgroundColor = "#0858a4";

// button.style.color = "white";

// updateButtonVisibility(index);

// }

// });

// nextBtn.addEventListener('click', switchToNextTab);

// prevBtn.addEventListener('click', switchToPrevTab);

// });

document.getElementById('form').addEventListener('submit', function (event) {

const check = document.getElementById('check').value;

const title = document.getElementById('title').value;

const abstract = document.getElementById('abstract').value;

const keyword = document.getElementById('keyword').value;

const reference = document.getElementById('reference').value;

const notes = document.getElementById('notes').value;

const file\_name = document.getElementById('file\_name').value;

if (!check || !title || !abstract || !keyword || !reference || !notes || !file\_name) {

event.preventDefault();

alert("Please fill in all the required fields.");

}

});

var quill1 = new Quill('#editor', {

theme: 'snow'

});

var quill2 = new Quill('#editor2', {

theme: 'snow'

});

var titleInput = document.getElementById("title");

var keywordsInput = document.getElementById("keywords");

var abstractInput = document.getElementById('abstract');

var referenceInput = document.getElementById('reference');

const titleInputValidation = document.getElementById('title');

const titleValidation = document.getElementById('title-validation');

const keywordsInputValidation = document.getElementById('keywords');

const keywordsValidation = document.getElementById('keywords-validation');

const abstractInputValidation = document.getElementById('abstract');

const abstractValidation = document.getElementById('abstract-validation');

quill1.on('text-change', function() {

abstractInput.value = quill1.getText();

updateButtonState();

});

quill2.on('text-change', function() {

referenceInput.value = quill2.getText();

updateButtonState();

});

titleInput.addEventListener('input', function () {

inputValidation(titleInput, titleValidation, function (value) {

const wordCount = value.split(/\s+/).length;

return wordCount >= 10 && wordCount <= 20;

});

updateButtonState();

});

keywordsInput.addEventListener('input', function () {

inputValidation(keywordsInput, keywordsValidation, function (value) {

const commaCount = (value.match(/,/g) || []).length;

return commaCount >= 1 && commaCount <= 4;

});

updateButtonState();

});

abstractInput.addEventListener('input', function () {

inputValidation(abstractInput, abstractValidation, function (value) {

const wordCount = value.split(/\s+/).length;

return wordCount < 10;

});

updateButtonState();

});

// Periodically check if abstractInput and referenceInput have values

setInterval(function() {

updateButtonState();

}, 1000); // Adjust the interval as needed

function updateButtonState() {

const formFloat = document.getElementById('form-floating-2');

const formFloating = document.getElementById('form-floating');

const formFloating3 = document.getElementById('form-floating-3');

const qlToolbar = document.getElementsByClassName('ql-toolbar');

const isTitleValid = titleValidation.style.display === 'none';

const isKeywordsValid = keywordsValidation.style.display === 'none';

const isAbstractValid = abstractValidation.style.display === 'none';

if (

titleInput.value.trim() !== '' &&

keywordsInput.value.trim() !== '' &&

abstractInput.value.trim() !== '' &&

referenceInput.value.trim() !== '' &&

isTitleValid &&

isKeywordsValid &&

isAbstractValid

) {

formFloating.style.width = '60%';

formFloating3.style.width = '60%';

formFloat.style.display = 'inline-block';

} else {

formFloat.style.display = 'none';

}

document.getElementById('btn-okay').addEventListener('click', function (event) {

const buttonOkay = document.getElementById('btn-okay');

formFloat.style.display = 'none';

formFloating.style.width = '100%';

// abstractEditor.style.width = '100%';

// referenceEditor.style.width = '100%';

// qlToolbar.style.width = '100%';

keywordsInput.innerHTML = '';

quill2.setText('');

});

}

function inputValidation(inputElement, validationElement, conditionFunction) {

const inputValue = inputElement.value.trim();

const isValid = conditionFunction(inputValue);

if (isValid) {

validationElement.style.display = 'none';

} else {

validationElement.style.display = 'block';

}

}

var quill3 = new Quill('#editor3', {

theme: 'snow'

});

quill3.on('text-change', function() {

notes.value = quill3.getText();

});

function openFileModal2(fileNumber) {

document.getElementById(`file\_name${fileNumber}`).click();

}

function deleteFile(fileNumber) {

const fileNameElement = document.getElementById(`fileName${fileNumber}`);

const fileTypeElement = document.getElementById(`fileType${fileNumber}`);

fileNameElement.textContent = '';

fileTypeElement.textContent = '';

}

// Listen for file input changes

document.getElementById('file\_name1').addEventListener('change', function () {

displayFileInfo(1);

});

document.getElementById('file\_name2').addEventListener('change', function () {

displayFileInfo(2);

});

document.getElementById('file\_name3').addEventListener('change', function () {

displayFileInfo(3);

});

// Function to display file information

function displayFileInfo(fileNumber) {

const fileInput = document.getElementById(`file\_name${fileNumber}`);

const fileNameElement = document.getElementById(`fileName${fileNumber}`);

const fileTypeElement = document.getElementById(`fileType${fileNumber}`);

// Check if a file is selected

if (fileInput.files.length > 0) {

fileNameElement.textContent = fileInput.files[0].name;

const longFileType = fileInput.files[0].type;

const shortenedFileType = shortenFileType(longFileType);

fileTypeElement.textContent = shortenedFileType;

} else {

// Handle the case when no file is selected

fileNameElement.textContent = 'No file selected';

fileTypeElement.textContent = '';

}

}

function shortenFileType(longFileType) {

const fileTypeMappings = {

'application/vnd.openxmlformats-officedocument.wordprocessingml.document': 'docx',

// Add more mappings as needed

};

return fileTypeMappings[longFileType] || longFileType;

}

document.getElementById('contributor-btn').addEventListener('click', function (event) {

Swal.fire({

html:

'<h5 class="title9" id="title-9">Contributors</h5>' +

'<hr id="swal-d">' +

'<div id="fName"><label id="sub-16">First Name: </label><input id="input1" class="swal2-input"></div>' +

'<div id= "lName"><label id="sub-17">Last Name: </label><input id="input2" class="swal2-input"></div>' +

'<label id= "sub-18">Preferred Public Name: </label><input id="input3" class="swal2-input">' +

'<label id= "sub-19">Email: </label><input id="input4" class="swal2-input">' +

'<label id= "sub-20">ORCID: </label><input id="input5" class="swal2-input">',

footer: '<button id="confirmBtn">Add Contributor</button>',

showConfirmButton: false,

});

document.getElementById('confirmBtn').addEventListener('click', function () {

var input1Value = document.getElementById('input1').value;

var input2Value = document.getElementById('input2').value;

var contributorsValue = input1Value + ' ' + input2Value;

var table = document.getElementById('table-contributor').getElementsByTagName('tbody')[0];

var newRow = table.insertRow(table.rows.length);

var cellContributor = newRow.insertCell(0);

cellContributor.innerHTML = contributorsValue;

var cellActions = newRow.insertCell(1);

cellActions.innerHTML = '<button type="button" id="btn-update" class="btn btn-outline-primary btn-sm btn-update" style="margin-right: 10px;">Update</button>' +

'<button type="button" id="btn-delete" class="btn btn-danger btn-sm btn-delete">Delete</button>';

var updateBtn = cellActions.querySelector('.btn-update');

var deleteBtn = cellActions.querySelector('.btn-delete');

updateBtn.addEventListener('click', function (event) {

Swal.fire({

html: '<h5 class="title10" id="title-10">Update Contributor</h5>' +

'<hr id="swal-d">' +

'<div id="fName"><label id="sub-21">First Name: </label><input id="updateInput1" class="swal2-input" value="' + input1Value + '"></div>' +

'<div id= "lName"><label id="sub-22">Last Name: </label><input id="updateInput2" class="swal2-input" value="' + input2Value + '"></div>' +

'<label id= "sub-23">Preferred Public Name: </label><input id="updateInput3" class="swal2-input">' +

'<label id= "sub-24">Email: </label><input id="updateInput4" class="swal2-input">' +

'<label id= "sub-25">ORCID: </label><input id="updateInput5" class="swal2-input">',

footer: '<button id="update-cont">Update</button>',

showConfirmButton: false

});

document.getElementById('update-cont').addEventListener('click', function () {

// Get the updated values from the modal

var updatedInput1Value = document.getElementById('updateInput1').value;

var updatedInput2Value = document.getElementById('updateInput2').value;

// Update the displayed value in the table

contributorsValue = updatedInput1Value + ' ' + updatedInput2Value;

cellContributor.innerHTML = contributorsValue;

// Close the update modal

Swal.close();

});

});

deleteBtn.addEventListener('click', function () {

var rowIndex = this.closest('tr').rowIndex;

table.deleteRow(rowIndex - 1);

console.log(contributorsValue);

});

Swal.close();

});

});

var titleInput = document.getElementById("title");

var keywordsInput = document.getElementById("keywords")

var fileInput = document.getElementById("file\_name");

var contributorInput = document.getElementById("contributor").textContent;

var input5 = document.getElementById("input5f1");

var input6 = document.getElementById("input6");

var input7 = document.getElementById("input7");

var input8 = document.getElementById("input8");

var input9 = document.getElementById("input9");

var input10 = document.getElementById("input10");

var input11 = document.getElementById("input11");

var input12 = document.getElementById("input12");

var input13 = document.getElementById("input13");

var input14 = document.getElementById("input14");

var input15 = document.getElementById("input15");

titleInput.addEventListener('input', function() {

input5.value = titleInput.value;

input11.value = titleInput.value;

});

keywordsInput.addEventListener('input', function() {

input6.value = keywordsInput.value;

input12.value = titleInput.value;

});

quill1.on('text-change', function() {

input7.value = quill1.getText();

input13.value = quill1.getText();

});

quill2.on('text-change', function() {

input8.value = quill2.getText();

input14.value = quill2.getText();

});

fileInput.addEventListener('input', function() {

if (fileInput.files.length > 0) {

input9.value = fileInput.files[0].name;

} else {

input9.value = "";

}

});

input10.value = contributorInput;

document.getElementById('update-cont-2').addEventListener('click', function (event) {

Swal.fire({

html: '<h5 class="title14" id="title-14">Update Article Details</h5>' + '<hr id="swal-d-2">' + '<label class="sub30" id="sub-30">Title: <input type="text" class="form-control" id="input11" value="'+ titleInput.value +'"></label>' + '<label class="sub31" id="sub-31">Keywords: <input type="text" class="form-control" id="input12" value="'+ keywordsInput.value +'"></label>' + '<label class="sub32" id="sub-32">Abstract: <input type="text" class="form-control" id="input13" id="input12" value="'+ quill1.getText() +'"></label>' + '<label class="sub33" id="sub-33">Reference: <input type="text" class="form-control" id="input14" value="'+ quill2.getText() +'"></label>',

footer: '<button id="btn-article-update">Update</button>',

showConfirmButton: false,

});

document.getElementById('btn-article-update').addEventListener('click', function () {

var newTitle = document.getElementById('input11').value;

var newKeywords = document.getElementById('input12').value;

var newAbstract = document.getElementById('input13').value;

var newReference = document.getElementById('input14').value;

titleInput.value = newTitle;

keywordsInput.value = newKeywords;

input5.value = newTitle;

input6.value = newKeywords;

input7.value = newAbstract;

input8.value = newReference;

try {

quill1.setText(newAbstract);

} catch (error) {

console.error("Unable to update Abstract Content:", error);

}

// i add two separate try catch because its not working when putting them in one :>>>

try {

quill2.setText(newReference);

} catch (error) {

console.error("Unable to update Reference Content:", error);

}

Swal.close();

});

});

document.getElementById('update-cont-3').addEventListener('click', function (event) {

Swal.fire({

html: '<h5 class="title15" id="title-15">Update File Content</h5>' + '<hr id="swal-d-3">'

+ '<label id="sub-34">File Name: </label>' +

'<button type="button" class="btn btn-primary btn-sm" id="newFile">Select File</button> '

+ '<input type="text" class="form-control" id="input15" readonly></input>',

showConfirmButton: false

})

document.getElementById('newFile').addEventListener('click', function() {

document.getElementById('file\_name').click();

});

document.getElementById('file\_name').addEventListener('change', function () {

var fileName = this.files[0].name;

document.getElementById('input15').value = fileName;

});

});

function inputValidation(inputElement, validationElement, conditionFunction) {

inputElement.addEventListener('blur', function () {

const inputValue = inputElement.value.trim();

const isValid = conditionFunction(inputValue);

if (isValid) {

validationElement.style.display = 'none';

} else {

validationElement.style.display = 'block';

}

});

}

// const titleInputValidation = document.getElementById('title');

// const titleValidation = document.getElementById('title-validation');

// inputValidation(titleInputValidation, titleValidation, function (value) {

// const wordCount = value.split(/\s+/).length;

// return wordCount >= 10 && wordCount <= 20;

// });

// const keywordsInputValidation = document.getElementById('keywords');

// const keywordsValidation = document.getElementById('keywords-validation');

// inputValidation(keywordsInputValidation, keywordsValidation, function (value) {

// const commaCount = (value.match(/,/g) || []).length;

// return commaCount >= 1 && commaCount <= 4;

// });

// const abstractInputValidation = document.getElementById('abstract');

// const abstractValidation = document.getElementById('abstract-validation');

// inputValidation(abstractInputValidation, abstractValidation, function (value) {

// const wordCount = value.split(/\s+/).length;

// return wordCount < 10;

// });

// function passValidation(){

// }

Ex\_submit.php

<?php

session\_start();

?>

<!DOCTYPE html>

<html>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>Submit Article</title>

<link href="../CSS/ex\_submit.css" rel="stylesheet">

<link href="https://fonts.googleapis.com/css?family=Raleway" rel="stylesheet">

<link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/6.4.2/css/all.min.css">

<link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.2/dist/css/bootstrap.min.css" rel="stylesheet">

<link href="https://cdn.quilljs.com/1.3.6/quill.snow.css" rel="stylesheet">

<body>

<div class="header-container" id="header-container">

</div>

<nav class="navigation-menus-container" id="navigation-menus-container">

</nav>

<form action="ex\_submit\_con.php" method="post" id="form" enctype="multipart/form-data">

<ul class="nav nav-tabs" id="myTab" role="tablist">

<li class="nav-item" role="presentation" style="margin-left: -10px;">

<button class="nav-link active" id="privacy-tab" data-bs-toggle="tab" data-bs-target="#privacy-tab-pane" type="button" role="tab" aria-controls="privacy-tab-pane" aria-selected="true">Privacy</button>

</li>

<li class="nav-item" role="presentation">

<button class="nav-link" id="article-tab" data-bs-toggle="tab" data-bs-target="#article-tab-pane" type="button" role="tab" aria-controls="article-tab-pane" aria-selected="false">Article Details</button>

</li>

<li class="nav-item" role="presentation">

<button class="nav-link" id="file-tab" data-bs-toggle="tab" data-bs-target="#file-tab-pane" type="button" role="tab" aria-controls="file-tab-pane" aria-selected="false">Upload File</button>

</li>

<li class="nav-item" role="presentation">

<button class="nav-link" id="contributors-tab" data-bs-toggle="tab" data-bs-target="#contributors-tab-pane" type="button" role="tab" aria-controls="contributors-tab-pane" aria-selected="false">Contributors</button>

</li>

<li class="nav-item" role="presentation">

<button class="nav-link" id="comment-tab" data-bs-toggle="tab" data-bs-target="#comment-tab-pane" type="button" role="tab" aria-controls="comment-tab-pane" aria-selected="false">Notes</button>

</li>

<li class="nav-item" role="presentation">

<button class="nav-link" id="review-tab" data-bs-toggle="tab" data-bs-target="#review-tab-pane" type="button" role="tab" aria-controls="review-tab-pane" aria-selected="false">Preview</button>

</li>

</ul>

<div class="tab-content" id="myTabContent">

<div class="tab-pane fade show active" id="privacy-tab-pane" role="tabpanel" aria-labelledby="privacy-tab" tabindex="0">

<p class="h5 pt-5" id="title-1">Submission Checklist</p>

<p class="h6" id="sub-1">Indicate that this submission is ready to be considered by this journal by checking off the following (comments to the editor can be added below).</p>

<div class="descript-1 pt-3">

<div class="form-check" id="checkList-1">

<input type="checkbox" id="check-1">

<p class="st-1">The submission has not been previously published, nor is it before another journal for consideration (or an explanation has been provided in Comments to the Editor).</p>

</div>

<div class="form-check" id="checkList-2">

<input type="checkbox" id="check-2">

<p class="st-2">Note that your paper will be submitted to iThenticate.com (Plagiarism Detection Software) to check the similarity score.</p>

</div>

<div class="form-check" id="checkList-3">

<input type="checkbox" id="check-3">

<p class="st-3">The submission file is in Microsoft Word document file format. Please do NOT submit in pdf.</p>

</div>

<div class="form-check" id="checkList-4">

<input type="checkbox" id="check-4">

<p class="st-4">Where available, URLs for online references have been provided.</p>

</div>

<div class="form-check" id="checkList-5">

<input type="checkbox" id="check-5">

<p class="st-4">Where available, URLs for online references have been provided.</p>

</div>

<div class="form-check" id="check-6">

<input type="checkbox" id="check-6">

<p class="st-5" style="width: 90%; display: inline-block">The text is single-column; single-spaced; uses a 11-point font (except for section headings which is 12); employs italics, rather than underlining (except with URL addresses); and all illustrations, figures, and tables are placed within the text at the appropriate points, rather than at the end. The Book Antiqua font should be used for all text in the paper. Use A4 page set-up. The top and bottom margins are 1.4 inches and the right/left margins are 1.2 inches.</p>

</div>

<div class="form-check" id="check-7">

<input type="checkbox" id="check-7">

<p class="st-6">The text adheres to the stylistic and bibliographic requirements outlined in the Author Guidelines, which is found in About the Journal.</p>

</div>

</div>

<hr id="line-1">

<p class="h5 pt-3" id="title-2">Copyright Notice</p>

<p class="h6 pt-3" id="sub-2">Authors who publish with this journal also agree to the following terms: </p>

<p class="h6 pt-3" id="sub-3">Authors are able to enter into separate, additional contractual arrangements for the non-exclusive distribution of the journal's published version of the work (e.g., post it to an institutional repository or publish it in a book), with an acknowledgement of its initial publication in this journal.</p>

<p class="h6 pt-3" id="sub-4">Authors are permitted and encouraged to post their work online (e.g., in institutional repositories or on their website) as this can lead to productive exchanges, as well as earlier and greater citation of published work.</p>

<hr id="line-2">

<p class="h5 pt-3" id="title-3">Journal's Privacy Statement</p>

<p class="h6" id="sub-5">The names and email addresses entered in this journal site will be used exclusively for the stated purposes of this journal and will not be made available for any other purpose or to any other party</p>

<div class="descript-2 pt-3">

<div class="form-check">

<input type="checkbox" id="check" name="check" value="1">

<p class="st-7">The authors agree to the terms of this Copyright Notice, which will apply to this submission if and when it is published by this journal (comments to the editor can be added below).</p>

</div>

</div>

</div>

<div class="tab-pane fade" id="article-tab-pane" role="tabpanel" aria-labelledby="article-tab" tabindex="0">

<div class="article-details">

<div class="details">

<p class="h5 pt-5" id="title-4">Article Details</p>

<p class="h6" id="sub-6" style="#0858a4">Please provide the following details to help us manage your submission in our system.</p>

</div>

<div class="input-details">

<div class="form-floating" id="form-floating">

<h6 id="sub-9">Title</h6>

<input class="form-control" type="text" id="title" name="title">

<p id="title-validation" style="color: red; display: none;">Title should consist of 10 words and less than 20 words\*</p>

<h6 id="sub-11">Abstract</h6>

<div id="editor">

</div>

<input class="form-control" type="text" id="abstract" name="abstract" style="display: none;">

<p id="abstract-validation" style="color: red; display: none;">Maximum of 1000 words\*</p>

</div>

<div class="form-floating-2" id="form-floating-2">

<!-- <h5 id="duplication-title">Checking of Details</h5> -->

<div class="duplicated-article">

<h6 class="checker-titles">Duplication Checker</h6>

<h6></h6>

<!-- <label id="label-title">Article: </label> -->

<p id="label-title">Article:</p>

<div id="similar-title"></div>

<div id ="similar-abstract"></div>

<!-- <label id="label-result">Result: </label> -->

<p id="label-result">Result:</p>

<div id="result-duplication"></div>

</div>

<div class="journal-type-container">

<h6 class="checker-titles">Journal Classification</h6>

<select class="form-select" name="journal-type" id="journal-type">

<option value="1" id="gavel">The Gavel</option>

<option value="2" id="lamp">The Lamp</option>

<option value="3" id="star">The Star</option>

</select>

<p class="suggestion-title">QOAJ can suggest journal based on your article</p>

</div>

<button type="button" class="btn btn-primary btn-sm" id="check-duplication">Check</button>

<button type="button" class="btn btn-primary btn-sm" id="btn-okay">Okay</button>

</div>

</div>

<div class="input-details-2 mt-3" id="form-floating-3">

<h6 id="sub-10">Keywords</h6>

<input class="form-control" type="text" id="keywords" name="keywords">

<p id="keywords-validation" style="color: red; display: none;">Keywords should consist at least 1 and a maximum of 4 commas</p>

<h6 class="sub-12 mt-5" id="sub-12">Reference</h6>

<div id="editor2">

</div>

<input class="form-control" type="text" id="reference" name="reference" style="display: none;">

</div>

</div>

</div>

<div class="tab-pane fade" id="file-tab-pane" role="tabpanel" aria-labelledby="file-tab" tabindex="0">

<div class="table-input">

<h5 class="title6 mt-5" id="title-6">Upload Files</h5>

<h6 class="sub13 mt-3" id="sub-13">Provide any files our editorial team may need to evaluate your submission. In addition to the main work, you may wish to submit data sets, conflict of interest statements, or other supplementary files if these will be helpful for our editors.</h6>

<button type="button" class="btn btn-primary btn-sm mt-5" onclick="openFileModal()" id="upload-btn">Upload File</button>

<table class="table table-hover" id="table-file">

<thead>

<tr>

<th scope="col" style="background-color: #0858a4; color: white; font-weight: normal;">File Name</th>

<th scope="col" style="background-color: #0858a4; color: white; font-weight: normal;">Type</th>

<th scope="col" style="background-color: #0858a4; color: white; font-weight: normal;">Action</th>

</tr>

</thead>

<tbody id="fileList">

<tr>

<td>

<input type="file" class="form-control" name="file\_name" id="file\_name" accept=".docx">

</td>

<td></td>

<td><button type="button" class="btn btn-danger btn-sm">Delete</button></td>

</tr>

<tr>

<td>

<input type="file" class="form-control" name="file\_name2" id="file\_name2" accept=".docx">

</td>

<td></td>

<td><button type="button" class="btn btn-danger btn-sm">Delete</button></td>

</tr>

<tr>

<td>

<input type="file" class="form-control" name="file\_name3" id="file\_name3" accept=".docx">

</td>

<td></td>

<td><button type="button" class="btn btn-danger btn-sm">Delete</button></td>

</tr>

</tbody>

</table>

</div>

</div>

<div class="tab-pane fade" id="contributors-tab-pane" role="tabpanel" aria-labelledby="contributors-tab" tabindex="0">

<div class="contributors-container">

<?php

if (isset($\_SESSION['LOGGED\_IN']) && $\_SESSION['LOGGED\_IN'] === true) {

$first\_name = $\_SESSION['first\_name'];

$firstName = isset($\_SESSION['first\_name']) ? ucfirst($\_SESSION['first\_name']) : '';

$middleName = isset($\_SESSION['middle\_name']) ? ' ' . ucfirst($\_SESSION['middle\_name']) : '';

$lastName = isset($\_SESSION['last\_name']) ? ' ' . ucfirst($\_SESSION['last\_name']) : '';

$contributor = $firstName . $middleName . $lastName;

echo "<p id='contributor' style='display: none'>$contributor

</p>";

}

?>

<h5 class="title7 mt-5" id="title-7">Add Contributors</h5>

<h6 class="sub14 mt-3" id="sub-14">Add details for all of the contributors to this submission. Contributors added here will be sent an email confirmation of the submission, as well as a copy of all editorial decisions recorded against this submission.</h6>

<div class="cont">

<!-- <h5 class="title8" id="title-8">Contributors</h5> -->

<button type="button" class="btn btn-primary btn-sm"id="contributor-btn">Add Contributors</button>

</div>

<table class="table table-striped" id="table-contributor">

<thead>

<tr >

<th scope="col" style="background-color: #0858a4; color: white; font-weight: normal;">Contributors</th>

<th scope="col" style="background-color: #0858a4; color: white; font-weight: normal;">Action</th>

</tr>

</thead>

<tbody>

<tr>

<td name="contributor" id="contributor"><?php echo $contributor; ?></td>

<td></td>

</tr>

</tbody>

</table>

</div>

</div>

<div class="tab-pane fade" id="comment-tab-pane" role="tabpanel" aria-labelledby="comment-tab" tabindex="0">

<div class="comment-container mt-5">

<h5 class="title8" id="title-8">Author Notes</h5>

<h6 class="sub15" id="sub-15">Please provide the following details to help our editorial team manage your submission.</h6>

<div id="editor3"></div>

</div>

<input class="form-control" type="text" id="notes" name="notes" style="display: none;">

</div>

<div class="tab-pane fade" id="review-tab-pane" role="tabpanel" aria-labelledby="review-tab" tabindex="0">

<h5 class="title9" id="title-9f">Review and Submit</h5>

<h6 class="sub16" id="sub-16f">Review the information you have entered before you complete your submission. You can change any of the details displayed here by clicking the edit button at the top of each section.</h6>

<h6 class="sub17" id="sub-17f">

Once you complete your submission, a member of our editorial team will be assigned to review it. Please ensure the details you have entered here are as accurate as possible.</h6>

<div class="article-info-container">

<div class="article-header">

<h5 class="title10" id="title-10f">Details</h5>

<button type="button" class="btn btn-outline-light btn-sm" id="update-cont-2">Update</button>

</div>

<div class="editable-content mt-5" id="editable-content">

<label id="sub-26">Title: </label><br>

<input type="text" class="form-control" id="input5f1" readonly><br>

<label id="sub-27">Keywords: </label><br>

<input type="text" class="form-control" id="input6" readonly><br>

<label id="sub-28">Abstract: </label><br>

<input type="text" class="form-control" id="input7" readonly><br>

<label id="sub-29">Reference: </label><br>

<input type="text" class="form-control" id="input8" readonly><br>

</div>

<div class="file-container">

<h5 class="title11" id="title-11">Files: </h5>

<div class="file-header-container">

<h5 class="title1f" id="title1f">File Name</h5>

<button type="button" class="btn btn-outline-light btn-sm" id="update-cont-3">Update</button>

</div>

<div class="file-content-container mt-3">

<input type="text" class="form-control" id="input9" readonly>

</div>

</div>

<div class="cont-container">

<h5 class="title12" id="title-12">Contributors: </h5>

<div class="cont-header-container">

<h5 class="title13" id="title-13">Contributors Name</h5>

<button type="button" class="btn btn-outline-light btn-sm" id="update-cont-4">Update</button>

</div>

<div class="file-content-container mt-3">

<input type="text" class="form-control" id="input10" readonly>

</div>

</div>

</div>

</div>

</div>

<div id="btn-action">

<button type="submit" class="btn btn-success btn-sm" id="submit">Submit</button>

<button type="button" class="btn btn-primary btn-sm" id="next">Next</button>

<button type="button" class="btn btn-secondary btn-sm" id="prev">Prev</button>

</div>

</form>

<script src="https://cdn.jsdelivr.net/npm/@popperjs/core@2.11.8/dist/umd/popper.min.js" ></script>

<script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.2/dist/js/bootstrap.min.js"></script>

<script src="https://cdnjs.cloudflare.com/ajax/libs/sweetalert/2.1.2/sweetalert.min.js"></script>

<script src="https://cdn.jsdelivr.net/npm/sweetalert2@11"></script>

<script src="https://cdn.quilljs.com/1.3.6/quill.js"></script>

<script src="../JS/reusable-header.js"></script>

<script src="../JS/ex\_submit.js"></script>

<script src="../JS/ex\_submit\_duplicate\_article.js"></script>

<script src="../JS/ex\_submit\_journal\_type.js"></script>

</body>

</html>

submit.php

<?php

require 'dbcon.php';

session\_start();

if (isset($\_SESSION['LOGGED\_IN']) && $\_SESSION['LOGGED\_IN'] === true) {

$id = $\_SESSION['id'];

echo "<p id='author\_id' style='display: none'>$id</p>";

$firstName = isset($\_SESSION['first\_name']) ? ucfirst($\_SESSION['first\_name']) : '';

$middleName = isset($\_SESSION['middle\_name']) ? ' ' . ucfirst($\_SESSION['middle\_name']) : '';

$lastName = isset($\_SESSION['last\_name']) ? ' ' . ucfirst($\_SESSION['last\_name']) : '';

$contributor = $firstName . $middleName . $lastName;

echo "<p id='contributor' style='display: none'>$contributor</p>";

}

if ($\_SERVER['REQUEST\_METHOD'] === "POST") {

$privacy = $\_POST['privacy'];

$title = $\_POST['title'];

$abstract = $\_POST['abstract'];

$keyword = $\_POST['keyword'];

$reference = $\_POST['reference'];

$category = $\_POST['category'];

$comment = $\_POST['comment'];

$author\_id = (isset($\_SESSION['id'])) ? $\_SESSION['id'] : null;

$contributor = (isset($\_SESSION['first\_name'])) ? $\_SESSION['first\_name'] : null;

if ($author\_id === null && $contributor){

echo 'Can\'t find information about this user.';

exit();

}

// Insert into article table

$sql = "INSERT INTO article (author, privacy, title, journal\_id, author\_id, abstract, keyword, `references`, comment)

VALUES (:author, :privacy, :title, :journal\_id, :author\_id, :abstract, :keyword, :references, :comment)";

$params = array(

'author' => $contributor,

':privacy' => $privacy,

'title' => $title,

':journal\_id' => $category,

':author\_id' => $author\_id,

':abstract' => $abstract,

':keyword' => $keyword,

':references' => $reference,

'comment' => $comment

);

$lastInsertedArticleId = database\_run($sql, $params, true);

// Check if files were uploaded

for ($i = 1; $i <= 3; $i++) {

$fileInputName = "file\_name$i";

if (!empty($\_FILES[$fileInputName]['name'])) {

$articleId = $lastInsertedArticleId;

$fileNames = $\_FILES[$fileInputName]['name'];

foreach ($fileNames as $index => $fileName) {

// Insert file names into the example\_files table

$fileInsertSql = "INSERT INTO example\_files (article\_id, file\_name) VALUES (:article\_id, :file\_name)";

$fileInsertParams = array(

':article\_id' => $articleId,

':file\_name' => $fileName,

);

database\_run($fileInsertSql, $fileInsertParams);

}

}

}

Header("Location: timeline.php");

exit();

} else {

echo 'Error: Invalid request method.';

exit();

}

?>