<?php

namespace App\Http\Controllers\Admin;

use App\Http\Controllers\Controller;

use App\Models\AdditionalDocument;

use App\Models\Application;

use App\Models\ApplicationFormImage;

use App\Models\{ApplicationImage,

CoApplicant,

CustomDetail,

DocumentType,

FormUpload,

LoanType,

Brand,

ApplicationAdditional,

ApplicationAdditionFormUploads,

ApplicationAdditionFormDocument

};

use Carbon\Carbon;use Illuminate\Http\Request;

use Illuminate\Support\Facades\DB;use Illuminate\Support\Facades\File;

use Illuminate\Support\Facades\Redirect;

use Illuminate\Support\Facades\Session; // use App\Http\Controllers\AllDataController;

use Illuminate\Support\Facades\Validator;

use Illuminate\Validation\Rule;

use Symfony\Component\HttpFoundation\Response;

use Yajra\DataTables\Facades\DataTables;

use Illuminate\Support\Facades\Auth;

class ApplicationController extends Controller

{

// protected $allDataController;

// public function \_\_construct()

// {

// $this->allDataController = new AllDataController();

// }

public function index(Request $request)

{

$isEditable = false;

$loan\_types = LoanType::pluck('title', 'id')->prepend('Select LoanType', '');

return view('admin.application.index', compact('isEditable', 'loan\_types'));

}

public function view(Request $request)

{

if (isset($request->id)) {

$data = Brand::where(['id' => $request->id])->select('id', 'brand')->first();

return response()->json(['status' => true, 'data' => $data]);

} else {

return response()->json(['status' => false, 'data' => 'Required Details Not Found']);

}

}

public function store(Request $request)

{

// dd($request->all());

$customer\_id = $request->client\_id;

// $customer\_id = $request->client\_id ;

$co\_applicants = $request->co\_applicants\_name;

$co\_applicants\_emails = $request->co\_applicants\_email;

$co\_applicants\_phones = $request->co\_applicants\_phone;

$co\_applicants\_addresss = $request->co\_applicants\_address;

$loan\_type\_id = $request->loan\_type\_id;

$document\_features = $request->document\_features;

$document\_type\_ids = $request->document\_id;

$form\_upload\_ids = $request->form\_upload\_id;

$uploadForm\_features = $request->uploadForm\_features;

$addition\_documents = $request->addition\_document;

// Application::where(['customer\_id'=> $customer\_id]) ;

$id = $request->case\_id;

if ($id) {

$applicant = Application::findOrFail($id);

$isUpdating = true;

} else {

$applicant = new Application();

$isUpdating = false;

}

$applicant->customer\_id = $request->client\_id;

$applicant->loan\_type\_id = $request->loan\_type\_id;

$applicant->loan\_amount = $request->loan\_amount;

$applicant->term = $request->term;

$applicant->pp\_ev = $request->pp\_ev;

$applicant->rate = $request->rate;

$applicant->proc\_fee = $request->proc\_fee;

$applicant->save();

if ($isUpdating) {

// Clear existing related records for updates

$applicant->applicationLoanDocument2()->detach();

$applicant->applicantDocument1()->detach();

$applicant->applicationLoanFormUpload2()->detach();

$applicant->applicantformUpload2()->detach();

$applicant->additionalDocument2()->detach();

$applicant->co\_applicant1()->detach();

}

if ($applicant) {

$id = $applicant->id;

for ($si = 0; $si < count($co\_applicants); $si++) {

$check\_co\_applicant = CoApplicant::where(['name' => $co\_applicants[$si], 'email' => $co\_applicants\_emails[$si]])->first();

if (!$check\_co\_applicant) {

$check\_co\_applicant = CoApplicant::firstOrCreate(

[

'name' => $co\_applicants[$si],

'email' => $co\_applicants\_emails[$si],

'phone' => $co\_applicants\_phones[$si],

'address' => trim($co\_applicants\_addresss[$si]),

]

);

} else {

CoApplicant::where(['name' => $co\_applicants[$si], 'email' => $co\_applicants\_emails[$si]])->update([

'phone' => $co\_applicants\_phones[$si],

'address' => trim($co\_applicants\_addresss[$si]),

]);

}

$applicationDetail = Application::find($id);

// Attach the co-applicant if it doesn't already exist in the pivot table

if (!$applicationDetail->co\_applicant1()->where('co\_applicant\_id', $check\_co\_applicant->id)->exists()) {

$applicationDetail->co\_applicant1()->attach($check\_co\_applicant->id);

}

}

// with loan\_type\_id

if (isset($document\_features) && count($document\_features) > 0) {

$applicant->applicationLoanDocument2()->detach($document\_features);

$applicant->applicationLoanDocument2()->attach($document\_features);

}

// without loan\_type\_id

if (isset($document\_type\_ids) && count($document\_type\_ids) > 0) {

$applicant->applicantDocument1()->detach($document\_type\_ids);

$applicant->applicantDocument1()->attach($document\_type\_ids);

}

if (isset($uploadForm\_features) && count($uploadForm\_features) > 0) {

$applicant->applicationLoanFormUpload2()->detach($uploadForm\_features);

$applicant->applicationLoanFormUpload2()->attach($uploadForm\_features);

}

// without loan\_type\_id

if (isset($form\_upload\_ids) && count($form\_upload\_ids) > 0) {

$applicant->applicantformUpload2()->detach($form\_upload\_ids);

$applicant->applicantformUpload2()->attach($form\_upload\_ids);

}

// AdditionalDocument

if (isset($addition\_documents) && count($addition\_documents) > 0) {

foreach ($addition\_documents as $addition\_document) {

// Check if the additional document exists or create it

$check\_co\_additional\_doucment = AdditionalDocument::firstOrCreate(['title' => $addition\_document]);

// Attach the additional document to the application if it doesn't already exist in the pivot table

$applicationDetail = Application::find($id);

if (!$applicationDetail->additionalDocument2()->where('additional\_id', $check\_co\_additional\_doucment->id)->exists()) {

$applicationDetail->additionalDocument2()->attach($check\_co\_additional\_doucment->id);

}

}

}

if (!$isUpdating) {

$application\_id = 'APP' . sprintf('%05d', $applicant->id);

$applicant->ref\_no = $application\_id;

$applicant->save();

}

// new list add 31-07-2024

$applicant-> loan\_category = $request->loan\_category ?? null;

$applicant->client\_propertity\_value = $request->client\_propertity\_value ?? null;

$applicant->client\_extra\_comment = $request->client\_extra\_comment ?? null;

$applicant->loan\_amount = $request->loan\_amount ?? null;

$applicant->term\_year = $request->term\_year ?? null;

$applicant->term\_month = $request->term\_month ?? null;

if($request->loan\_category == 'broker'){

$applicant->mortgage\_status = $request->mortgage\_status ?? null;

$applicant->purpose\_loan= $request->purpose\_loan ?? null;

$applicant->application\_made= $request->application\_made ?? null;

$applicant->live\_or\_intent\_property = $request->live\_or\_intent\_property ?? null;

}else{

$applicant->mortgage\_status = null;

$applicant->purpose\_loan= null;

$applicant->application\_made= null;

$applicant->live\_or\_intent\_property = null;

}

$applicant->save();

//new list end 31-07-2024

$titles = [];

$addition\_form\_document\_titles = $request->addition\_form\_document;

if ( isset($addition\_form\_document\_titles) && count($addition\_form\_document\_titles) > 0) {

foreach ($addition\_form\_document\_titles as $index => $addition\_form\_document\_title) {

// Sanitize and format the title

$addition\_form\_document = ucwords($addition\_form\_document\_title);

// Check if a file is present for the current title

$addition\_form\_document\_file = $request->file('addition\_form\_document\_file')[$index] ?? null;

// Check if a record already exists

$check\_data = ApplicationAdditionFormUploads::where([

'title' => $addition\_form\_document,

'application\_id' => $id

])->first();

if (!$check\_data && $addition\_form\_document\_file) {

// Save the new file to the public/formupload directory

$fileExtension = $addition\_form\_document\_file->getClientOriginalExtension();

$fileName = $id . '\_' . str\_replace(" ", '\_', $addition\_form\_document) . '\_' . time() . '.' . $fileExtension;

$addition\_form\_document\_file->move(public\_path('formupload'), $fileName);

// Save the file information in the database

$check\_data = new ApplicationAdditionFormUploads();

$check\_data->application\_id = $id;

$check\_data->title = $addition\_form\_document;

$check\_data->file\_path = $fileName;

$check\_data->save();

} elseif ($check\_data && $addition\_form\_document\_file) {

// File exists; Handle file update

$fileExtension = $addition\_form\_document\_file->getClientOriginalExtension();

$fileName = $id . '\_' . str\_replace(" ", '\_', $addition\_form\_document) . '\_' . time() . '.' . $fileExtension;

$addition\_form\_document\_file->move(public\_path('formupload'), $fileName);

// Update the existing file information in the database

$oldFilePath = public\_path('formupload') . '/' . $check\_data->file\_path;

if (File::exists($oldFilePath)) {

File::delete($oldFilePath);

}

$check\_data->file\_path = $fileName;

$check\_data->save();

}

// Add title to the list

$titles[] = $addition\_form\_document;

}

}

// Delete old files that are not in the current request

$delete\_form\_additionals = ApplicationAdditionFormUploads::where('application\_id', $id)

->whereNotIn('title', $titles)

->get();

foreach ($delete\_form\_additionals as $formUpload) {

$imagePath = public\_path('formupload/' . $formUpload->file\_path);

if (File::exists($imagePath)) {

File::delete($imagePath);

}

$formUpload->forceDelete();

}

$url = route('admin.document.index', $applicant->id);

$message = $isUpdating ? "Case Updated Successfully and application id is " . $applicant->ref\_no : "Case Created Successfully and application id is " . $application\_id;

return response()->json(['status' => true, 'data' => $message, 'url' => $url]);

}

}

public function edit(Request $request)

{

if (isset($request->id)) {

$data = Brand::where(['id' => $request->id])->select('id', 'brand')->first();

return response()->json(['status' => true, 'data' => $data]);

} else {

return response()->json(['status' => false, 'data' => 'Required Details Not Found']);

}

}

public function destroy(Request $request)

{

if (isset($request->id)) {

$delete = Brand::where(['id' => $request->id])->update([

'deleted\_at' => Carbon::now(),

]);

return response()->json(['status' => 'success', 'data' => 'Brand Deleted Successfully']);

} else {

return response()->json(['status' => 'error', 'data' => 'Technical Error']);

}

}

public function massDestroy()

{

$academicYears = Brand::whereIn('id', request('ids'))->delete();

// find(request('ids'));

// foreach ($academicYears as $academicYear) {

// $academicYear->delete();

// }

return response(null, Response::HTTP\_NO\_CONTENT);

}

public function create()

{

LoanType::whereNotNull('title')->pluck('title', 'id');

}

public function GetClients(Request $request)

{

$search = $request->search;

if ($search == '') {

$clients = CustomDetail::orderby('name', 'asc')->select('id', 'name', 'email')->distinct()->limit(6)->get();

} else {

$clients = CustomDetail::orderby('name', 'asc')->select('id', 'name', 'email')->where('name', 'like', '%' . $search . '%')->distinct()->limit(6)->get();

}

$response = [];

foreach ($clients as $client) {

$response[] = array(

"id" => $client->id,

"text" => $client->name . '(' . $client->email . ')',

);

}

return response()->json($response);

}

public function GetClientInfo(Request $request)

{

$client\_details = CustomDetail::select('\*')

->where('id', '=', $request->id)

->get();

return response()->json($client\_details);

}

public function customerStore(Request $request)

{

$customer\_id = intVal($request->customer\_id);

$validator = Validator::make($request->all(), [

"name" => ["required"],

"last\_name" => ["nullable"],

'email' => ["required", "email", Rule::unique('customer\_details')->ignore($customer\_id)],

"phone" => ["required", "numeric"],

"address1" => ["nullable"],

"address2" => ["nullable"],

]);

if ($validator->fails()) {

return response()->json([

'status' => 400,

'errors' => $validator->errors(),

]);

} else {

$data = CustomDetail::updateOrCreate(

['id' => $request->customer\_id],

[

'name' => $request->name,

'phone' => $request->phone,

'email' => $request->email,

'address1' => $request->address1,

'address2' => $request->address2,

'last\_name' => $request->last\_name

]

);

$isNew = $data->wasRecentlyCreated;

return response()->json($data->id ? ['status' => 200, 'data' => $data, 'errors' => null, 'is\_new' => $isNew] : ['status' => 400, 'data' => null, 'errors' => null, 'is\_new' => '']);

}

}

public function getLoanTypeDetails(Request $request)

{

$id = $request->loan\_type\_id;

$loanType = LoanType::find($id);

if ($loanType) {

$documentTypes = $loanType->documents;

$document\_id = $loanType->documents->pluck('id') ?? [];

$document\_content = '';

foreach ($documentTypes as $documentType) {

$document\_content .= '<div class="form-group col-md-3">';

$document\_content .= '<input type="checkbox" name="document\_features[]" value="' . $documentType->id . '">';

$document\_content .= '&nbsp;&nbsp;' . $documentType->title . '<br>';

$document\_content .= '</div>';

}

$formUploads = $loanType->formuploads;

$formuploads\_id = $loanType->formuploads->pluck('id') ?? [];

$formUpload\_content = '';

foreach ($formUploads as $formUpload) {

$formUpload\_content .= '<div class="form-group col-md-3">';

$formUpload\_content .= '<input type="checkbox" name="uploadForm\_features[]" value="' . $formUpload->id . '">';

$formUpload\_content .= '&nbsp;&nbsp;' . $formUpload->title . '<br>';

$formUpload\_content .= '</div>';

}

$document\_typs = DocumentType::whereNotIN('id', $document\_id)->pluck('title', 'id');

$form\_uploads\_typs = FormUpload::whereNotIN('id', $formuploads\_id)->pluck('title', 'id');

return response()->json(['status' => true, 'document\_content' => $document\_content, 'formUpload\_content' => $formUpload\_content, 'document\_typs' => $document\_typs, 'form\_uploads\_typs' => $form\_uploads\_typs]);

} else {

return response()->json(['status' => false, 'document\_content' => [], 'formUpload\_content' => [], 'document\_typs' => '', 'form\_uploads\_typs' => '']);

}

}

public function document\_index($id)

{

$id = intval($id);

$data = Application::find($id);

$isdocument = false;

if (!$data || $data == null) {

return view('admin.application.document\_index', compact('isdocument'));

}

$documentImages = ApplicationImage::where('application\_id', $id)->whereIn('document\_id', $data->applicationLoanDocument2->pluck('id'))->get();

$documentImages2 = ApplicationImage::where('application\_id', $id)->whereIn('document\_id', $data->applicantDocument1->pluck('id'))->get();

$documentImages3 = ApplicationFormImage::where('application\_id', $id)->whereIn('form\_id', $data->applicantformUpload2->pluck('id'))->get();

$documentImages4 = ApplicationFormImage::where('application\_id', $id)->whereIn('form\_id', $data->applicationLoanFormUpload2->pluck('id'))->get();

$documentImages5 = ApplicationAdditional::where('application\_id', $id)->whereIn('additional\_id', $data->additionalDocument2->pluck('id'))->get();

// $documentImages5 = ApplicationAdditionFormDocument::where('application\_id', $id)->whereIn('additional\_form\_id', $data->ApplicationAdditionFormUploads->pluck('id'))->get();

// dd( $documentImages5);

$isdocument = true;

return view('admin.application.document\_index', compact('data', 'isdocument', 'documentImages', 'documentImages2', 'documentImages3', 'documentImages4','documentImages5'));

}

public function uploadImageGet(Request $request)

{

$user = Auth::user() ?? '' ;

$role = $user ? ($user->roles[0]->id ?? '' ): '';

$status\_image\_path = '';

if (isset($request->file) && isset($request->applicationId) && isset($request->documentId) && isset($request->imageId)) {

$status = false;

$message = '';

$applicationImage = '';

if ($request->hasFile('file')) {

$applicationImage = ApplicationImage::where([

'application\_id' => $request->applicationId,

'document\_id' => $request->documentId,

'id' => $request->imageId,

])->first();

$index = intval($request->index);

$index++;

if ($applicationImage) {

$oldFilePath = $applicationImage->file\_path;

$status = $applicationImage->status == "Rejected" ? 'Reupload' : $applicationImage->status ;

$image = $request->file('file');

$imageName = pathinfo($oldFilePath, PATHINFO\_FILENAME) . '.' . $image->getClientOriginalExtension();

$destinationPath = public\_path('/formupload');

$newFilePath = $imageName;

if ($image->move($destinationPath, $imageName)) {

// Update the file path in the database

$applicationImage->file\_path = $newFilePath;

$applicationImage->status = $status;

$applicationImage->save();

$status\_image\_path = $this ->statusImage($status);

// if($status == 'Rejected'){

// $status\_image\_path = '<img src="' . asset('formupload/file\_cancel.png') . '" class="document\_status\_img">';

// }elseif($status == 'Reupload'){

// $status\_image\_path = '<img src="' . asset('formupload/stage\_reupload.png') . '" class="document\_status\_img">';

// }elseif($status == 'Accepted'){

// $status\_image\_path = '<img src="' . asset('formupload/file\_success.png') . '" class="document\_status\_img">';

// }elseif($status == 'Upload'){

// $status\_image\_path = '<img src="' . asset('formupload/file\_uploaded.png') . '" class="document\_status\_img">';

// }

$message = 'Successful image updated';

$status = true;

$fileExtension = pathinfo('formupload/' .$applicationImage->file\_path, PATHINFO\_EXTENSION);

$doc = ['doc', 'docx'];

if(in\_array($fileExtension ,$doc)){

$ImagePath = asset('formupload/word\_preview.png') ;

}else

if($fileExtension == 'pdf'){

$ImagePath = asset('formupload/pdf\_preview.png') ;

}else{

$ImagePath = asset('formupload/'. $applicationImage->file\_path) ;

}

$html = '' ;

$html .= '<input type="checkbox" class="delete-checkbox" data-image-id="'.$applicationImage->id.'" data-file-id="uploaded\_file\_'.$index.'">';

$html .= '<input type="hidden" name="uploadform" value="uploaded\_file\_'.$index.'">';

$html .= '<img src="'.$ImagePath.'" style="max-width: 100%; height: 100%;">';

$html .= '<div id="upload\_file\_button\_'.$index.'" class="d-flex justify-content-center upload\_file\_button">';

$html .= '<button id="upload\_preview\_'.$index.'" class="btn btn-outline-primary btn-sm" title="Preview" data-app-id ="' .$request->applicationId .'" data-document-id ="' .$request->documentId .'" data-file-id="uploaded\_file\_'.$index.'" data-image-id="'.$applicationImage->id.'" data-file-type="'.$applicationImage->fileExtension.'"><i class="fas fa-eye"></i></button>';

$html .= '<button id="upload\_change\_'.$index.'" class="btn btn-outline-warning btn-sm" title="Change" data-app-id ="' .$request->applicationId .'" data-document-id ="' .$request->documentId .'" data-file-id="uploaded\_file\_'.$index.'" data-image-id="'.$applicationImage->id.'" data-file-type="'.$applicationImage->fileExtension.'"><i class="far fa-edit"></i></button>';

$html .= '<button id="upload\_delete\_'.$index.'" class="btn btn-outline-danger btn-sm" title="Delete" data-app-id ="' .$request->applicationId .'" data-document-id ="' .$request->documentId .'" data-file-id="uploaded\_file\_'.$index.'" data-image-id="'.$applicationImage->id.'" data-file-type="'.$applicationImage->fileExtension.'"><i class="fas fa-trash"></i></button>';

$html .= '</div>';

$html .= '<div id="upload\_remark\_'.$index.'" class="upload\_remark">';

$html .= '<div class="form-group">';

// if($role != 1){

// $html .= '<div><input type="text" name="input\_remark" class="form-control form-control-sm" id="input\_remark\_'.$index.'" placeholder="Remark" value="'.$applicationImage->remark.'">';

// }

// $html .= '<input type="hidden" name="imageData" value="'.$applicationImage->id.'">';

// $html .= '</div></div></div>';

$hide = '';

if($role == 1){

$hide = 'readonly';

}

$html .= '<div>';

$html .= '<input type="text" name="input\_remark" class="form-control form-control-sm" id="input\_remark\_' . $index . '" placeholder="Remark" value="' .$applicationImage->remark. '" ' . $hide . '>';

$html .= '<input type="hidden" name="imageData" value="'.$applicationImage->id .'">';

$html .= '</div>';

$html .= '</div></div></div>';

} else {

$message = 'Failed to move the uploaded file';

}

} else {

return response()->json(['status' => false, 'message' => 'Image path not found']);

}

return response()->json(['status' => $status, 'message' => $message, 'data' => $html, 'image\_path'=> $status\_image\_path]);

}

}

if (isset($request->applicationId) && isset($request->documentId) && isset($request->file)) {

if ($request->hasFile('file')) {

$image = $request->file('file');

$imageName = intval($user ->id). '\_'.intval($request->currentImageCount).'\_'.time() . '.' . $image->getClientOriginalExtension();

$destinationPath = public\_path('/formupload');

$image->move($destinationPath, $imageName);

$ApplicationImage = ApplicationImage::create(

['application\_id' => $request->applicationId, 'document\_id' => $request->documentId, 'file\_path' => $imageName, 'status' => 'Upload']

);

$data = ApplicationImage::find($ApplicationImage->id) ?? '';

//start

$html = '';

$fileExtension = pathinfo('formupload/' .$ApplicationImage ->file\_path, PATHINFO\_EXTENSION);

$doc = ['doc', 'docx'];

if(in\_array($fileExtension ,$doc)){

$ImagePath = asset('formupload/word\_preview.png') ;

}else

if($fileExtension == 'pdf'){

$ImagePath = asset('formupload/pdf\_preview.png') ;

}else{

$ImagePath = asset('formupload/'. $ApplicationImage ->file\_path) ;

}

$status = $ApplicationImage->status ?? null ;

$status\_image\_path = $this ->statusImage($status);

// if($status == 'Rejected'){

// $status\_image\_path = '<img src="' . asset('formupload/file\_cancel.png') . '" class="document\_status\_img">';

// }elseif($status == 'Reupload'){

// $status\_image\_path = '<img src="' . asset('formupload/stage\_reupload.png') . '" class="document\_status\_img">';

// }elseif($status == 'Accepted'){

// $status\_image\_path = '<img src="' . asset('formupload/file\_success.png') . '" class="document\_status\_img">';

// }elseif($status == 'Upload'){

// $status\_image\_path = '<img src="' . asset('formupload/file\_uploaded.png') . '" class="document\_status\_img">';

// }

$index = intval($request->currentImageCount) ?? 0 ;

$index ++ ;

$html .= '<div class="col-6 my-2 all\_file\_data">';

$html .= '<form>';

$html .= '<div id="uploaded\_file\_'.$index.'" class="uploaded\_file position-relative">';

$html .= '<input type="checkbox" class="delete-checkbox" data-image-id="'.$ApplicationImage ->id.'" data-file-id="uploaded\_file\_'.$index.'">';

$html .= '<input type="hidden" name="uploadform" value="uploaded\_file\_'.$index.'">';

$html .= '<img src="'.$ImagePath.'" style="max-width: 100%; height: 100%;">';

$html .= '<div id="upload\_file\_button\_'.$index.'" class="d-flex justify-content-center upload\_file\_button">';

$html .= '<button id="upload\_preview\_'.$index.'" class="btn btn-outline-primary btn-sm" title="Preview" data-app-id ="' .$request->applicationId .'" data-document-id ="' .$request->documentId .'" data-file-id="uploaded\_file\_'.$index.'" data-image-id="'.$ApplicationImage ->id.'" data-file-type="'.$ApplicationImage ->fileExtension.'"><i class="fas fa-eye"></i></button>';

$html .= '<button id="upload\_change\_'.$index.'" class="btn btn-outline-warning btn-sm" title="Change" data-app-id ="' .$request->applicationId .'" data-document-id ="' .$request->documentId .'" data-file-id="uploaded\_file\_'.$index.'" data-image-id="'.$ApplicationImage ->id.'" data-file-type="'.$ApplicationImage ->fileExtension.'"><i class="far fa-edit"></i></button>';

$html .= '<button id="upload\_delete\_'.$index.'" class="btn btn-outline-danger btn-sm" title="Delete" data-app-id ="' .$request->applicationId .'" data-document-id ="' .$request->documentId .'" data-file-id="uploaded\_file\_'.$index.'" data-image-id="'.$ApplicationImage ->id.'" data-file-type="'.$ApplicationImage ->fileExtension.'"><i class="fas fa-trash"></i></button>';

$html .= '</div>';

$html .= '<div id="upload\_remark\_'.$index.'" class="upload\_remark">';

$html .= '<div class="form-group">';

$hide = '' ;

if($role == 1){

$hide = 'readonly';

}

$html .= '<div>';

$html .= '<input type="text" name="input\_remark" class="form-control form-control-sm" id="input\_remark\_' . $index . '" placeholder="Remark" value="' . $ApplicationImage ->remark . '" ' . $hide . '>';

$html .= '<input type="hidden" name="imageData" value="' . $ApplicationImage ->id . '">';

$html .= '</div>';

$html .= '</div></div></div></form></div>';

//end

$data = $html;

return response()->json(['status' => true, 'data' => $data, 'image\_path' => $status\_image\_path]);

}

}

$data = ApplicationImage::where(['application\_id' => $request->applicationId, 'document\_id' => $request->documentId]) ;

if($role != 1){

$data->where('status', '!=','Verified');

}

$data = $data ->get();

$html = '';

// Loop through each item in $dynamicData

foreach ($data as $index => $item) {

$index++;

$fileExtension = pathinfo('formupload/' .$item['file\_path'], PATHINFO\_EXTENSION);

$doc = ['doc', 'docx'];

if(in\_array($fileExtension ,$doc)){

$ImagePath = asset('formupload/word\_preview.png') ;

}else

if($fileExtension == 'pdf'){

$ImagePath = asset('formupload/pdf\_preview.png') ;

}else{

$ImagePath = asset('formupload/'. $item['file\_path']) ;

}

$reject\_class = '' ;

if($item['status'] == "Rejected"){

$reject\_class= 'border\_red' ;

}

$html .= '<div class="col-6 my-2 all\_file\_data">';

$html .= '<form>';

$html .= '<div id="uploaded\_file\_'.$index.'" class="uploaded\_file position-relative '.$reject\_class.'">';

$html .= '<input type="checkbox" class="delete-checkbox" data-image-id="'.$item['id'].'" data-file-id="uploaded\_file\_'.$index.'">';

$html .= '<input type="hidden" name="uploadform" value="uploaded\_file\_'.$index.'">';

$html .= '<img src="'.$ImagePath.'" style="max-width: 100%; height: 100%;">';

$html .= '<div id="upload\_file\_button\_'.$index.'" class="d-flex justify-content-center upload\_file\_button">';

$html .= '<button id="upload\_preview\_'.$index.'" class="btn btn-outline-primary btn-sm" title="Preview" data-app-id ="' .$request->applicationId .'" data-document-id ="' .$request->documentId .'" data-file-id="uploaded\_file\_'.$index.'" data-image-id="'.$item['id'].'" data-file-type="'.$item['fileExtension'].'"><i class="fas fa-eye"></i></button>';

$html .= '<button id="upload\_change\_'.$index.'" class="btn btn-outline-warning btn-sm" title="Change" data-app-id ="' .$request->applicationId .'" data-document-id ="' .$request->documentId .'" data-file-id="uploaded\_file\_'.$index.'" data-image-id="'.$item['id'].'" data-file-type="'.$item['fileExtension'].'"><i class="far fa-edit"></i></button>';

$html .= '<button id="upload\_delete\_'.$index.'" class="btn btn-outline-danger btn-sm" title="Delete" data-app-id ="' .$request->applicationId .'" data-document-id ="' .$request->documentId .'" data-file-id="uploaded\_file\_'.$index.'" data-image-id="'.$item['id'].'" data-file-type="'.$item['fileExtension'].'"><i class="fas fa-trash"></i></button>';

$html .= '</div>';

$html .= '<div id="upload\_remark\_'.$index.'" class="upload\_remark">';

$html .= '<div class="form-group">';

$hide = '' ;

if($role == 1){

$hide = 'readonly';

}

$html .= '<div>';

$html .= '<input type="text" name="input\_remark" class="form-control form-control-sm" id="input\_remark\_' . $index . '" placeholder="Remark" value="' . $item['remark'] . '" ' . $hide . '>';

$html .= '<input type="hidden" name="imageData" value="' . $item['id'] . '">';

$html .= '</div>';

$html .= '</div></div></div></form></div>';

}

$data = $html ;

return response()->json(['status' => true, 'data' => $data]);

}

//22-07-2024

public function uploadAdditionImageGet(Request $request)

{

$user = Auth::user() ?? '' ;

$role = $user ? ($user->roles[0]->id ?? '' ): '';

$image\_path = '';

if (isset($request->file) && isset($request->applicationId) && isset($request->documentId) && isset($request->imageId)) {

$status = false;

$message = '';

$applicationImage = '';

if ($request->hasFile('file')) {

$applicationImage = ApplicationAdditional::where([

'application\_id' => $request->applicationId,

'additional\_id' => $request->documentId,

'id' => $request->imageId,

])->first();

$index = intval($request->index);

$index++;

if ($applicationImage) {

$oldFilePath = $applicationImage->file\_path;

$status = $applicationImage->status == "Rejected" ? 'Reupload' : $applicationImage->status ;

$image = $request->file('file');

$imageName = pathinfo($oldFilePath, PATHINFO\_FILENAME) . '.' . $image->getClientOriginalExtension();

$destinationPath = public\_path('/formupload');

$newFilePath = $imageName;

if ($image->move($destinationPath, $imageName)) {

// Update the file path in the database

$applicationImage->file\_path = $newFilePath;

$applicationImage->status = $status;

$applicationImage->save();

$status\_image\_path = $this ->statusImage($status);

// if($status == 'Rejected'){

// $image\_path = '<img src="' . asset('formupload/file\_cancel.png') . '" class="document\_status\_img">';

// }elseif($status == 'Reupload'){

// $image\_path = '<img src="' . asset('formupload/stage\_reupload.png') . '" class="document\_status\_img">';

// }elseif($status == 'Accepted'){

// $image\_path = '<img src="' . asset('formupload/file\_success.png') . '" class="document\_status\_img">';

// }elseif($status == 'Upload'){

// $image\_path = '<img src="' . asset('formupload/file\_uploaded.png') . '" class="document\_status\_img">';

// }

$message = 'Successful image updated';

$status = true;

$fileExtension = pathinfo('formupload/' .$applicationImage->file\_path, PATHINFO\_EXTENSION);

$doc = ['doc', 'docx'];

if(in\_array($fileExtension ,$doc)){

$ImagePath = asset('formupload/word\_preview.png') ;

}else

if($fileExtension == 'pdf'){

$ImagePath = asset('formupload/pdf\_preview.png') ;

}else{

$ImagePath = asset('formupload/'. $applicationImage->file\_path) ;

}

$html = '' ;

$html .= '<input type="checkbox" class="delete-checkbox addition\_delete\_checkbox" data-image-id="'.$applicationImage->id.'" data-file-id="uploaded\_file\_'.$index.'">';

$html .= '<input type="hidden" name="uploadform" value="uploaded\_file\_'.$index.'">';

$html .= '<img src="'.$ImagePath.'" style="max-width: 100%; height: 100%;">';

$html .= '<div id="upload\_file\_button\_'.$index.'" class="d-flex justify-content-center upload\_file\_button">';

$html .= '<button id="upload2\_preview\_'.$index.'" class="btn btn-outline-primary btn-sm" title="Preview" data-app-id ="' .$request->applicationId .'" data-document-id ="' .$request->documentId .'" data-file-id="uploaded\_file\_'.$index.'" data-image-id="'.$applicationImage->id.'" data-file-type="'.$applicationImage->fileExtension.'"><i class="fas fa-eye"></i></button>';

$html .= '<button id="upload2\_change\_'.$index.'" class="btn btn-outline-warning btn-sm" title="Change" data-app-id ="' .$request->applicationId .'" data-document-id ="' .$request->documentId .'" data-file-id="uploaded\_file\_'.$index.'" data-image-id="'.$applicationImage->id.'" data-file-type="'.$applicationImage->fileExtension.'"><i class="far fa-edit"></i></button>';

$html .= '<button id="upload2\_delete\_'.$index.'" class="btn btn-outline-danger btn-sm" title="Delete" data-app-id ="' .$request->applicationId .'" data-document-id ="' .$request->documentId .'" data-file-id="uploaded\_file\_'.$index.'" data-image-id="'.$applicationImage->id.'" data-file-type="'.$applicationImage->fileExtension.'"><i class="fas fa-trash"></i></button>';

$html .= '</div>';

$html .= '<div id="upload\_remark\_'.$index.'" class="upload\_remark">';

$html .= '<div class="form-group">';

// if($role != 1){

// $html .= '<div><input type="text" name="input\_remark" class="form-control form-control-sm" id="input\_remark\_'.$index.'" placeholder="Remark" value="'.$applicationImage->remark.'">';

// }

// $html .= '<input type="hidden" name="imageData" value="'.$applicationImage->id.'">';

// $html .= '</div></div></div>';

if($role == 1){

$hide = 'readonly';

}

$html .= '<div>';

$html .= '<input type="text" name="input\_remark" class="form-control form-control-sm" id="input\_remark\_' . $index . '" placeholder="Remark" value="' .$applicationImage->remark. '" ' . $hide . '>';

$html .= '<input type="hidden" name="imageData" value="'.$applicationImage->id .'">';

$html .= '</div>';

$html .= '</div></div></div>';

} else {

$message = 'Failed to move the uploaded file';

}

} else {

return response()->json(['status' => false, 'message' => 'Image path not found']);

}

return response()->json(['status' => $status, 'message' => $message, 'data' => $html, 'image\_path'=> $image\_path ]);

}

}

if (isset($request->applicationId) && isset($request->documentId) && isset($request->file)) {

if ($request->hasFile('file')) {

$image = $request->file('file');

$imageName = intval($user ->id). '\_'.intval($request->currentImageCount).'\_'. time() . '.' . $image->getClientOriginalExtension();

$destinationPath = public\_path('/formupload');

$image->move($destinationPath, $imageName);

$ApplicationImage = ApplicationAdditional::create(

['application\_id' => $request->applicationId, 'additional\_id' => $request->documentId, 'file\_path' => $imageName, 'status' => 'Upload']

);

$data = ApplicationAdditional::find($ApplicationImage->id) ?? '';

//start

$html = '';

$fileExtension = pathinfo('formupload/' .$ApplicationImage ->file\_path, PATHINFO\_EXTENSION);

$doc = ['doc', 'docx'];

if(in\_array($fileExtension ,$doc)){

$ImagePath = asset('formupload/word\_preview.png') ;

}else

if($fileExtension == 'pdf'){

$ImagePath = asset('formupload/pdf\_preview.png') ;

}else{

$ImagePath = asset('formupload/'. $ApplicationImage ->file\_path) ;

}

$index = intval($request->currentImageCount) ?? 0 ;

$index ++ ;

$html .= '<div class="col-6 my-2 all\_file\_data">';

$html .= '<form>';

$html .= '<div id="uploaded\_file\_'.$index.'" class="uploaded\_file position-relative">';

$html .= '<input type="checkbox" class="delete-checkbox addition\_delete\_checkbox" data-image-id="'.$ApplicationImage ->id.'" data-file-id="uploaded\_file\_'.$index.'">';

$html .= '<input type="hidden" name="uploadform" value="uploaded\_file\_'.$index.'">';

$html .= '<img src="'.$ImagePath.'" style="max-width: 100%; height: 100%;">';

$html .= '<div id="upload\_file\_button\_'.$index.'" class="d-flex justify-content-center upload\_file\_button">';

$html .= '<button id="upload2\_preview\_'.$index.'" class="btn btn-outline-primary btn-sm" title="Preview" data-app-id ="' .$request->applicationId .'" data-document-id ="' .$request->documentId .'" data-file-id="uploaded\_file\_'.$index.'" data-image-id="'.$ApplicationImage ->id.'" data-file-type="'.$ApplicationImage ->fileExtension.'"><i class="fas fa-eye"></i></button>';

$html .= '<button id="upload2\_change\_'.$index.'" class="btn btn-outline-warning btn-sm" title="Change" data-app-id ="' .$request->applicationId .'" data-document-id ="' .$request->documentId .'" data-file-id="uploaded\_file\_'.$index.'" data-image-id="'.$ApplicationImage ->id.'" data-file-type="'.$ApplicationImage ->fileExtension.'"><i class="far fa-edit"></i></button>';

$html .= '<button id="upload2\_delete\_'.$index.'" class="btn btn-outline-danger btn-sm" title="Delete" data-app-id ="' .$request->applicationId .'" data-document-id ="' .$request->documentId .'" data-file-id="uploaded\_file\_'.$index.'" data-image-id="'.$ApplicationImage ->id.'" data-file-type="'.$ApplicationImage ->fileExtension.'"><i class="fas fa-trash"></i></button>';

$html .= '</div>';

$html .= '<div id="upload\_remark\_'.$index.'" class="upload\_remark">';

$html .= '<div class="form-group">';

$hide = '' ;

if($role == 1){

$hide = 'readonly';

}

$html .= '<div>';

$html .= '<input type="text" name="input\_remark" class="form-control form-control-sm" id="input\_remark\_' . $index . '" placeholder="Remark" value="' . $ApplicationImage ->remark . '" ' . $hide . '>';

$html .= '<input type="hidden" name="imageData" value="' . $ApplicationImage ->id . '">';

$html .= '</div>';

$html .= '</div></div></div></form></div>';

//end

$data = $html;

return response()->json(['status' => true, 'data' => $data]);

}

}

$data = ApplicationAdditional::where(['application\_id' => $request->applicationId, 'additional\_id' => $request->documentId])->get();

$html = '';

// Loop through each item in $dynamicData

// foreach ($data as $index => $item) {

// $index++;

// $fileExtension = pathinfo('formupload/' .$item['file\_path'], PATHINFO\_EXTENSION);

// if($fileExtension == 'pdf'){

// $ImagePath = asset('formupload/pdf\_preview.png') ;

// }else{

// $ImagePath = asset('formupload/'. $item['file\_path']) ;

// }

// $reject\_class = '' ;

// if($item['status']== "Rejected"){

// $reject\_class= 'border\_red' ;

// }

// // $html .= '<div class="col-6 my-2 all\_file\_data 1\_">';

// // $html .= '<form>';

// // $html .= '<div id="uploaded\_file\_'.$index.'" class="uploaded\_file position-relative ' .$reject\_class. '">';

// // $html .= '<input type="checkbox" class="delete-checkbox addition\_delete\_checkbox" data-image-id="'.$item['id'].'" data-file-id="uploaded\_file\_'.$index.'">';

// // $html .= '<input type="hidden" name="uploadform" value="uploaded\_file\_'.$index.'">';

// // $html .= '<img src="'.$ImagePath.'" style="max-width: 100%; height: 100%;">';

// // $html .= '<div id="upload\_file\_button\_'.$index.'" class="d-flex justify-content-center upload\_file\_button">';

// // $html .= '<button id="upload2\_preview\_'.$index.'" class="btn btn-outline-primary btn-sm" title="Preview" data-app-id ="' .$request->applicationId .'" data-document-id ="' .$request->documentId .'" data-file-id="uploaded\_file\_'.$index.'" data-image-id="'.$item['id'].'" data-file-type="'.$item['fileExtension'].'"><i class="fas fa-eye"></i></button>';

// // $html .= '<button id="upload2\_change\_'.$index.'" class="btn btn-outline-warning btn-sm" title="Change" data-app-id ="' .$request->applicationId .'" data-document-id ="' .$request->documentId .'" data-file-id="uploaded\_file\_'.$index.'" data-image-id="'.$item['id'].'" data-file-type="'.$item['fileExtension'].'"><i class="far fa-edit"></i></button>';

// // $html .= '<button id="upload2\_delete\_'.$index.'" class="btn btn-outline-danger btn-sm" title="Delete" data-app-id ="' .$request->applicationId .'" data-document-id ="' .$request->documentId .'" data-file-id="uploaded\_file\_'.$index.'" data-image-id="'.$item['id'].'" data-file-type="'.$item['fileExtension'].'"><i class="fas fa-trash"></i></button>';

// // $html .= '</div>';

// // $html .= '<div id="upload\_remark\_'.$index.'" class="upload\_remark">';

// // $html .= '<div class="form-group">';

// // $hide = '' ;

// // if($role == 1){

// // $hide = 'readonly';

// // }

// // $html .= '<div>';

// // $html .= '<input type="text" name="input\_remark" class="form-control form-control-sm" id="input\_remark\_' . $index . '" placeholder="Remark" value="' . $item['remark'] . '" ' . $hide . '>';

// // $html .= '<input type="hidden" name="imageData" value="' . $item['id'] . '">';

// // $html .= '</div>';

// // $html .= '</div></div></div></form></div>';

// $html .= '<div class="col-6 my-2 all\_file\_data 1\_">';

// $html .= '<form>';

// $html .= '<div id="uploaded\_file\_'.$index.'" class="uploaded\_file position-relative ' .$reject\_class. '">';

// $html .= '<input type="checkbox" class="delete-checkbox addition\_delete\_checkbox" data-image-id="'.$item['id'].'" data-file-id="uploaded\_file\_'.$index.'">';

// $html .= '<input type="hidden" name="uploadform" value="uploaded\_file\_'.$index.'">';

// $html .= '<img src="'.$ImagePath.'" style="max-width: 100%; height: 100%;">';

// $html .= '<div id="upload\_file\_button\_'.$index.'" class="d-flex justify-content-center upload\_file\_button">';

// $html .= '<button id="upload2\_preview\_'.$index.'" class="btn btn-outline-primary btn-sm" title="Preview" data-app-id ="' .$request->applicationId .'" data-document-id ="' .$request->documentId .'" data-file-id="uploaded\_file\_'.$index.'" data-image-id="'.$item['id'].'" data-file-type="'.$item['fileExtension'].'"><i class="fas fa-eye"></i></button>';

// $html .= '<button id="upload2\_change\_'.$index.'" class="btn btn-outline-warning btn-sm" title="Change" data-app-id ="' .$request->applicationId .'" data-document-id ="' .$request->documentId .'" data-file-id="uploaded\_file\_'.$index.'" data-image-id="'.$item['id'].'" data-file-type="'.$item['fileExtension'].'"><i class="far fa-edit"></i></button>';

// $html .= '<button id="upload2\_delete\_'.$index.'" class="btn btn-outline-danger btn-sm" title="Delete" data-app-id ="' .$request->applicationId .'" data-document-id ="' .$request->documentId .'" data-file-id="uploaded\_file\_'.$index.'" data-image-id="'.$item['id'].'" data-file-type="'.$item['fileExtension'].'"><i class="fas fa-trash"></i></button>';

// $html .= '</div>';

// $html .= '<div id="upload\_remark\_'.$index.'" class="upload\_remark">';

// $html .= '<div class="form-group">';

// $hide = '';

// if($role == 1){

// $hide = 'readonly';

// }

// $html .= '<div>';

// $html .= '<input type="text" name="input\_remark" class="form-control form-control-sm" id="input\_remark\_' . $index . '" placeholder="Remark" value="' . $item['remark'] . '" ' . $hide . '>';

// $html .= '<input type="hidden" name="imageData" value="' . $item['id'] . '">';

// $html .= '</div>';

// $html .= '</div>';

// $html .= '</div>';

// $html .= '</div>';

// $html .= '</form>';

// $html .= '</div>';

// }

foreach ($data as $index => $item) {

$index++;

$fileExtension = pathinfo('formupload/' .$item['file\_path'], PATHINFO\_EXTENSION);

$doc = ['doc', 'docx'];

if(in\_array($fileExtension ,$doc)){

$ImagePath = asset('formupload/word\_preview.png') ;

}else

if ($fileExtension == 'pdf') {

$ImagePath = asset('formupload/pdf\_preview.png');

} else {

$ImagePath = asset('formupload/'. $item['file\_path']);

}

$reject\_class = '';

if ($item['status'] == "Rejected") {

$reject\_class = 'border\_red';

}

$html .= '<div class="col-6 my-2 all\_file\_data 1\_">';

$html .= '<form>';

$html .= '<div id="uploaded\_file\_'.$index.'" class="uploaded\_file position-relative ' .$reject\_class. '">';

$html .= '<input type="checkbox" class="delete-checkbox addition\_delete\_checkbox" data-image-id="'.$item['id'].'" data-file-id="uploaded\_file\_'.$index.'">';

$html .= '<input type="hidden" name="uploadform" value="uploaded\_file\_'.$index.'">';

$html .= '<img src="'.$ImagePath.'" style="max-width: 100%; height: 100%;">';

$html .= '<div id="upload\_file\_button\_'.$index.'" class="d-flex justify-content-center upload\_file\_button">';

$html .= '<button id="upload2\_preview\_'.$index.'" class="btn btn-outline-primary btn-sm" title="Preview" data-app-id ="' .$request->applicationId .'" data-document-id ="' .$request->documentId .'" data-file-id="uploaded\_file\_'.$index.'" data-image-id="'.$item['id'].'" data-file-type="'.$item['fileExtension'].'"><i class="fas fa-eye"></i></button>';

$html .= '<button id="upload2\_change\_'.$index.'" class="btn btn-outline-warning btn-sm" title="Change" data-app-id ="' .$request->applicationId .'" data-document-id ="' .$request->documentId .'" data-file-id="uploaded\_file\_'.$index.'" data-image-id="'.$item['id'].'" data-file-type="'.$item['fileExtension'].'"><i class="far fa-edit"></i></button>';

$html .= '<button id="upload2\_delete\_'.$index.'" class="btn btn-outline-danger btn-sm" title="Delete" data-app-id ="' .$request->applicationId .'" data-document-id ="' .$request->documentId .'" data-file-id="uploaded\_file\_'.$index.'" data-image-id="'.$item['id'].'" data-file-type="'.$item['fileExtension'].'"><i class="fas fa-trash"></i></button>';

$html .= '</div>';

$html .= '<div id="upload\_remark\_'.$index.'" class="upload\_remark">';

$html .= '<div class="form-group">';

$hide = '';

if ($role == 1) {

$hide = 'readonly';

}

$html .= '<div>';

$html .= '<input type="text" name="input\_remark" class="form-control form-control-sm" id="input\_remark\_' . $index . '" placeholder="Remark" value="' . $item['remark'] . '" ' . $hide . '>';

$html .= '<input type="hidden" name="imageData" value="' . $item['id'] . '">';

$html .= '</div>';

$html .= '</div>';

$html .= '</div>';

$html .= '</div>';

$html .= '</form>';

$html .= '</div>';

}

$data = $html ;

// dd( $data );

return response()->json(['status' => true, 'data' => $data]);

}

// 22-07-2024

public function uploadFormImageGet(Request $request)

{

$user = Auth::user() ?? '' ;

$role = $user ? ($user->roles[0]->id ?? '' ): '';

$status\_image\_path = '' ;

if (isset($request->file) && isset($request->applicationId) && isset($request->documentId) && isset($request->imageId)) {

$status = false;

$message = '';

$applicationImage = '';

if ($request->hasFile('file')) {

$applicationImage = ApplicationFormImage::where([

'application\_id' => $request->applicationId,

'form\_id' => $request->documentId,

'id' => $request->imageId,

])->first();

$index = intval($request->index);

$index++;

if ($applicationImage) {

$oldFilePath = $applicationImage->file\_path;

$image = $request->file('file');

$imageName = pathinfo($oldFilePath, PATHINFO\_FILENAME) . '.' . $image->getClientOriginalExtension();

$destinationPath = public\_path('/formupload');

$newFilePath = $imageName;

$status\_ = $applicationImage->status == "Rejected" ? "Reupload" : $applicationImage->status ;

// if($role==1){

// $status\_ = "Upload" ;

// }else{

// $status\_ = "Upload" ;

// }

// dd ($status\_);

if ($image->move($destinationPath, $imageName)) {

// Update the file path in the database

$applicationImage->file\_path = $newFilePath;

$applicationImage->status = $status\_ ;

$applicationImage->save();

$status\_image\_path = $this ->statusImage($status\_);

$message = 'Successful image updated';

$status = true;

$fileExtension = pathinfo('formupload/' .$applicationImage->file\_path, PATHINFO\_EXTENSION);

// dd($fileExtension);

$doc = ['doc', 'docx'];

if(in\_array($fileExtension ,$doc)){

$ImagePath = asset('formupload/word\_preview.png') ;

}else

if($fileExtension == 'pdf'){

$ImagePath = asset('formupload/pdf\_preview.png') ;

}else{

$ImagePath = asset('formupload/'. $applicationImage->file\_path) ;

}

// $reject\_class = '' ;

$html = '' ;

// if($applicationImage->status == "Rejected"){

// $reject\_class= 'border\_red' ;

// }

// $reject\_class = 'border\_red' ;

$html .= '<input type="checkbox" class="delete-checkbox form\_delete-checkbox" data-image-id="'.$applicationImage->id.'" data-file-id="uploaded\_file\_'.$index.'">';

$html .= '<input type="hidden" name="uploadform" value="uploaded\_file\_'.$index.'">';

$html .= '<img src="'.$ImagePath.'" style="max-width: 100%; height: 100%;">';

$html .= '<div id="upload\_file\_button\_'.$index.'" class="d-flex justify-content-center upload\_file\_button">';

$html .= '<button id="upload1\_preview\_'.$index.'" class="btn btn-outline-primary btn-sm" title="Preview" data-app-id ="' .$request->applicationId .'" data-document-id ="' .$request->documentId .'" data-file-id="uploaded\_file\_'.$index.'" data-image-id="'.$applicationImage->id.'" data-file-type="'.$applicationImage->fileExtension.'"><i class="fas fa-eye"></i></button>';

$html .= '<button id="upload1\_change\_'.$index.'" class="btn btn-outline-warning btn-sm" title="Change" data-app-id ="' .$request->applicationId .'" data-document-id ="' .$request->documentId .'" data-file-id="uploaded\_file\_'.$index.'" data-image-id="'.$applicationImage->id.'" data-file-type="'.$applicationImage->fileExtension.'"><i class="far fa-edit"></i></button>';

$html .= '<button id="upload1\_delete\_'.$index.'" class="btn btn-outline-danger btn-sm" title="Delete" data-app-id ="' .$request->applicationId .'" data-document-id ="' .$request->documentId .'" data-file-id="uploaded\_file\_'.$index.'" data-image-id="'.$applicationImage->id.'" data-file-type="'.$applicationImage->fileExtension.'"><i class="fas fa-trash"></i></button>';

$html .= '</div>';

$html .= '<div id="upload\_remark\_'.$index.'" class="upload\_remark">';

$html .= '<div class="form-group">';

// if($role != 1){

// $html .= '<div><input type="text" name="input\_remark" class="form-control form-control-sm" id="input\_remark\_'.$index.'" placeholder="Remark" value="'.$applicationImage->remark.'">';

// }

// $html .= '<input type="hidden" name="imageData" value="'.$applicationImage->id.'">';

// $html .= '</div></div></div>';

$hide = '';

if($role == 1){

$hide = 'readonly';

}

$html .= '<div>';

$html .= '<input type="text" name="input\_remark" class="form-control form-control-sm" id="input\_remark\_' . $index . '" placeholder="Remark" value="' .$applicationImage->remark. '" ' . $hide . '>';

$html .= '<input type="hidden" name="imageData" value="'.$applicationImage->id .'">';

$html .= '</div>';

$html .= '</div></div></div>';

} else {

$message = 'Failed to move the uploaded file';

}

} else {

return response()->json(['status' => false, 'message' => 'Image path not found']);

}

// if ($applicationImage) {

// $oldFilePath = $applicationImage->file\_path;

// $image = $request->file('file');

// $imageName = pathinfo($oldFilePath, PATHINFO\_FILENAME) . '.' . $image->getClientOriginalExtension();

// $destinationPath = public\_path('/formupload');

// $newFilePath = $imageName;

// if ($image->move($destinationPath, $imageName)) {

// // Update the file path in the database

// $applicationImage->file\_path = $newFilePath;

// $applicationImage->save();

// $message = 'Successful image updated';

// $status = true;

// } else {

// $message = 'Failed to move the uploaded file';

// }

// } else {

// return response()->json(['status' => false, 'message' => 'Image path not found']);

// }

return response()->json(['status' => $status, 'message' => $message, 'data' => $html, 'image\_path'=> $status\_image\_path ]);

}

}

if (isset($request->applicationId) && isset($request->documentId) && isset($request->file)) {

if ($request->hasFile('file')) {

$image = $request->file('file');

$imageName = intval($user ->id). '\_'.intval($request->currentImageCount).'\_'.time() . '.' . $image->getClientOriginalExtension();

$destinationPath = public\_path('/formupload');

$image->move($destinationPath, $imageName);

$ApplicationImage = ApplicationFormImage::create(

['application\_id' => $request->applicationId, 'form\_id' => $request->documentId, 'file\_path' => $imageName, 'status' => 'Upload']

);

$data = ApplicationFormImage::find($ApplicationImage->id) ?? '';

$data\_id = $data->id ;

//start

$html = '';

$fileExtension = pathinfo('formupload/' .$ApplicationImage ->file\_path, PATHINFO\_EXTENSION);

$doc = ['doc', 'docx'];

if(in\_array($fileExtension ,$doc)){

$ImagePath = asset('formupload/word\_preview.png') ;

}else

if($fileExtension == 'pdf'){

$ImagePath = asset('formupload/pdf\_preview.png') ;

}else{

$ImagePath = asset('formupload/'. $ApplicationImage ->file\_path) ;

}

$status = $ApplicationImage ->status;

$status\_image\_path = $this ->statusImage($status);

$index = intval($request->currentImageCount) ?? 0 ;

$index ++ ;

$html .= '<div class="col-6 my-2 all\_file\_data">';

$html .= '<form>';

$html .= '<div id="uploaded\_file\_'.$index.'" class="uploaded\_file position-relative">';

$html .= '<input type="checkbox" class="delete-checkbox form\_delete-checkbox" data-image-id="'.$ApplicationImage ->id.'" data-file-id="uploaded\_file\_'.$index.'">';

$html .= '<input type="hidden" name="uploadform" value="uploaded\_file\_'.$index.'">';

$html .= '<img src="'.$ImagePath.'" style="max-width: 100%; height: 100%;">';

$html .= '<div id="upload\_file\_button\_'.$index.'" class="d-flex justify-content-center upload\_file\_button">';

$html .= '<button id="upload1\_preview\_'.$index.'" class="btn btn-outline-primary btn-sm" title="Preview" data-app-id ="' .$request->applicationId .'" data-document-id ="' .$request->documentId .'" data-file-id="uploaded\_file\_'.$index.'" data-image-id="'.$data ->id.'" data-file-type="'.$ApplicationImage ->fileExtension.'"><i class="fas fa-eye"></i></button>';

$html .= '<button id="upload1\_change\_'.$index.'" class="btn btn-outline-warning btn-sm" title="Change" data-app-id ="' .$request->applicationId .'" data-document-id ="' .$request->documentId .'" data-file-id="uploaded\_file\_'.$index.'" data-image-id="'.$data ->id.'" data-file-type="'.$ApplicationImage ->fileExtension.'"><i class="far fa-edit"></i></button>';

$html .= '<button id="upload1\_delete\_'.$index.'" class="btn btn-outline-danger btn-sm" title="Delete" data-app-id ="' .$request->applicationId .'" data-document-id ="' .$request->documentId .'" data-file-id="uploaded\_file\_'.$index.'" data-image-id="'.$data ->id.'" data-file-type="'.$ApplicationImage ->fileExtension.'"><i class="fas fa-trash"></i></button>';

$html .= '</div>';

$html .= '<div id="upload\_remark\_'.$index.'" class="upload\_remark">';

$html .= '<div class="form-group">';

$hide = '' ;

if($role == 1){

$hide = 'readonly';

}

$html .= '<div>';

$html .= '<input type="text" name="input\_remark" class="form-control form-control-sm" id="input\_remark\_' . $index . '" placeholder="Remark" value="' . $ApplicationImage ->remark . '" ' . $hide . '>';

$html .= '<input type="hidden" name="imageData" value="' . $ApplicationImage ->id . '">';

$html .= '</div>';

$html .= '</div></div></div></form></div>';

//end

$data = $html;

// Save image path in the database if needed

// Example: Image::create(['path' => $imageName]);

return response()->json(['status' => true, 'data' => $data , 'image\_path'=> $status\_image\_path ]);

}

}

$data = ApplicationFormImage::where(['application\_id' => $request->applicationId, 'form\_id' => $request->documentId, ])->get();

}

public function UploadAdditionalFormImageGet2(Request $request)

{

$user = Auth::user() ?? '' ;

$role = $user ? ($user->roles[0]->id ?? '' ): '';

$status\_image\_path = '' ;

if (isset($request->file) && isset($request->applicationId) && isset($request->documentId) && isset($request->imageId)) {

$status = false;

$message = '';

$applicationImage = '';

if ($request->hasFile('file')) {

$applicationImage = ApplicationAdditionFormDocument::where([

'application\_id' => $request->applicationId,

'form\_id' => $request->documentId,

'id' => $request->imageId,

])->first();

$index = intval($request->index);

$index++;

if ($applicationImage) {

$oldFilePath = $applicationImage->file\_path;

$image = $request->file('file');

$imageName = pathinfo($oldFilePath, PATHINFO\_FILENAME) . '.' . $image->getClientOriginalExtension();

$destinationPath = public\_path('/formupload');

$newFilePath = $imageName;

$status\_ = $applicationImage->status == "Rejected" ? "Reupload" : $applicationImage->status ;

// if($role==1){

// $status\_ = "Upload" ;

// }else{

// $status\_ = "Upload" ;

// }

// dd ($status\_);

if ($image->move($destinationPath, $imageName)) {

// Update the file path in the database

$applicationImage->file\_path = $newFilePath;

$applicationImage->status = $status\_ ;

$applicationImage->save();

$status\_image\_path = $this ->statusImage($status\_);

$message = 'Successful image updated';

$status = true;

$fileExtension = pathinfo('formupload/' .$applicationImage->file\_path, PATHINFO\_EXTENSION);

// dd($fileExtension);

$doc = ['doc', 'docx'];

if(in\_array($fileExtension ,$doc)){

$ImagePath = asset('formupload/word\_preview.png') ;

}else

if($fileExtension == 'pdf'){

$ImagePath = asset('formupload/pdf\_preview.png') ;

}else{

$ImagePath = asset('formupload/'. $applicationImage->file\_path) ;

}

// $reject\_class = '' ;

$html = '' ;

// if($applicationImage->status == "Rejected"){

// $reject\_class= 'border\_red' ;

// }

// $reject\_class = 'border\_red' ;

$html .= '<input type="checkbox" class="delete-checkbox form\_delete-checkbox" data-image-id="'.$applicationImage->id.'" data-file-id="uploaded\_file\_'.$index.'">';

$html .= '<input type="hidden" name="uploadform" value="uploaded\_file\_'.$index.'">';

$html .= '<img src="'.$ImagePath.'" style="max-width: 100%; height: 100%;">';

$html .= '<div id="upload\_file\_button\_'.$index.'" class="d-flex justify-content-center upload\_file\_button">';

$html .= '<button id="upload1\_preview\_'.$index.'" class="btn btn-outline-primary btn-sm" title="Preview" data-app-id ="' .$request->applicationId .'" data-document-id ="' .$request->documentId .'" data-file-id="uploaded\_file\_'.$index.'" data-image-id="'.$applicationImage->id.'" data-file-type="'.$applicationImage->fileExtension.'"><i class="fas fa-eye"></i></button>';

$html .= '<button id="upload1\_change\_'.$index.'" class="btn btn-outline-warning btn-sm" title="Change" data-app-id ="' .$request->applicationId .'" data-document-id ="' .$request->documentId .'" data-file-id="uploaded\_file\_'.$index.'" data-image-id="'.$applicationImage->id.'" data-file-type="'.$applicationImage->fileExtension.'"><i class="far fa-edit"></i></button>';

$html .= '<button id="upload1\_delete\_'.$index.'" class="btn btn-outline-danger btn-sm" title="Delete" data-app-id ="' .$request->applicationId .'" data-document-id ="' .$request->documentId .'" data-file-id="uploaded\_file\_'.$index.'" data-image-id="'.$applicationImage->id.'" data-file-type="'.$applicationImage->fileExtension.'"><i class="fas fa-trash"></i></button>';

$html .= '</div>';

$html .= '<div id="upload\_remark\_'.$index.'" class="upload\_remark">';

$html .= '<div class="form-group">';

// if($role != 1){

// $html .= '<div><input type="text" name="input\_remark" class="form-control form-control-sm" id="input\_remark\_'.$index.'" placeholder="Remark" value="'.$applicationImage->remark.'">';

// }

// $html .= '<input type="hidden" name="imageData" value="'.$applicationImage->id.'">';

// $html .= '</div></div></div>';

$hide = '';

if($role == 1){

$hide = 'readonly';

}

$html .= '<div>';

$html .= '<input type="text" name="input\_remark" class="form-control form-control-sm" id="input\_remark\_' . $index . '" placeholder="Remark" value="' .$applicationImage->remark. '" ' . $hide . '>';

$html .= '<input type="hidden" name="imageData" value="'.$applicationImage->id .'">';

$html .= '</div>';

$html .= '</div></div></div>';

} else {

$message = 'Failed to move the uploaded file';

}

} else {

return response()->json(['status' => false, 'message' => 'Image path not found']);

}

// if ($applicationImage) {

// $oldFilePath = $applicationImage->file\_path;

// $image = $request->file('file');

// $imageName = pathinfo($oldFilePath, PATHINFO\_FILENAME) . '.' . $image->getClientOriginalExtension();

// $destinationPath = public\_path('/formupload');

// $newFilePath = $imageName;

// if ($image->move($destinationPath, $imageName)) {

// // Update the file path in the database

// $applicationImage->file\_path = $newFilePath;

// $applicationImage->save();

// $message = 'Successful image updated';

// $status = true;

// } else {

// $message = 'Failed to move the uploaded file';

// }

// } else {

// return response()->json(['status' => false, 'message' => 'Image path not found']);

// }

return response()->json(['status' => $status, 'message' => $message, 'data' => $html, 'image\_path'=> $status\_image\_path ]);

}

}

if (isset($request->applicationId) && isset($request->documentId) && isset($request->file)) {

if ($request->hasFile('file')) {

$image = $request->file('file');

$imageName = intval($user ->id). '\_'.intval($request->currentImageCount).'\_'.time() . '.' . $image->getClientOriginalExtension();

$destinationPath = public\_path('/formupload');

$image->move($destinationPath, $imageName);

$ApplicationImage = ApplicationFormImage::create(

['application\_id' => $request->applicationId, 'form\_id' => $request->documentId, 'file\_path' => $imageName, 'status' => 'Upload']

);

$data = ApplicationAdditionFormDocument::find($ApplicationImage->id) ?? '';

$data\_id = $data->id ;

//start

$html = '';

$fileExtension = pathinfo('formupload/' .$ApplicationImage ->file\_path, PATHINFO\_EXTENSION);

$doc = ['doc', 'docx'];

if(in\_array($fileExtension ,$doc)){

$ImagePath = asset('formupload/word\_preview.png') ;

}else

if($fileExtension == 'pdf'){

$ImagePath = asset('formupload/pdf\_preview.png') ;

}else{

$ImagePath = asset('formupload/'. $ApplicationImage ->file\_path) ;

}

$status = $ApplicationImage ->status;

$status\_image\_path = $this ->statusImage($status);

$index = intval($request->currentImageCount) ?? 0 ;

$index ++ ;

$html .= '<div class="col-6 my-2 all\_file\_data">';

$html .= '<form>';

$html .= '<div id="uploaded\_file\_'.$index.'" class="uploaded\_file position-relative">';

$html .= '<input type="checkbox" class="delete-checkbox form\_delete-checkbox" data-image-id="'.$ApplicationImage ->id.'" data-file-id="uploaded\_file\_'.$index.'">';

$html .= '<input type="hidden" name="uploadform" value="uploaded\_file\_'.$index.'">';

$html .= '<img src="'.$ImagePath.'" style="max-width: 100%; height: 100%;">';

$html .= '<div id="upload\_file\_button\_'.$index.'" class="d-flex justify-content-center upload\_file\_button">';

$html .= '<button id="upload1\_preview\_'.$index.'" class="btn btn-outline-primary btn-sm" title="Preview" data-app-id ="' .$request->applicationId .'" data-document-id ="' .$request->documentId .'" data-file-id="uploaded\_file\_'.$index.'" data-image-id="'.$data ->id.'" data-file-type="'.$ApplicationImage ->fileExtension.'"><i class="fas fa-eye"></i></button>';

$html .= '<button id="upload1\_change\_'.$index.'" class="btn btn-outline-warning btn-sm" title="Change" data-app-id ="' .$request->applicationId .'" data-document-id ="' .$request->documentId .'" data-file-id="uploaded\_file\_'.$index.'" data-image-id="'.$data ->id.'" data-file-type="'.$ApplicationImage ->fileExtension.'"><i class="far fa-edit"></i></button>';

$html .= '<button id="upload1\_delete\_'.$index.'" class="btn btn-outline-danger btn-sm" title="Delete" data-app-id ="' .$request->applicationId .'" data-document-id ="' .$request->documentId .'" data-file-id="uploaded\_file\_'.$index.'" data-image-id="'.$data ->id.'" data-file-type="'.$ApplicationImage ->fileExtension.'"><i class="fas fa-trash"></i></button>';

$html .= '</div>';

$html .= '<div id="upload\_remark\_'.$index.'" class="upload\_remark">';

$html .= '<div class="form-group">';

$hide = '' ;

if($role == 1){

$hide = 'readonly';

}

$html .= '<div>';

$html .= '<input type="text" name="input\_remark" class="form-control form-control-sm" id="input\_remark\_' . $index . '" placeholder="Remark" value="' . $ApplicationImage ->remark . '" ' . $hide . '>';

$html .= '<input type="hidden" name="imageData" value="' . $ApplicationImage ->id . '">';

$html .= '</div>';

$html .= '</div></div></div></form></div>';

//end

$data = $html;

// Save image path in the database if needed

// Example: Image::create(['path' => $imageName]);

return response()->json(['status' => true, 'data' => $data , 'image\_path'=> $status\_image\_path ]);

}

}

$data = ApplicationFormImage::where(['application\_id' => $request->applicationId, 'form\_id' => $request->documentId, ])->get();

}

public function uploadAdditionalImageGet(Request $request)

{

$user = Auth::user() ?? '' ;

$role = $user ? ($user->roles[0]->id ?? '' ): '';

$status\_image\_path = '';

if (isset($request->file) && isset($request->applicationId) && isset($request->documentId) && isset($request->imageId)) {

$status = false;

$message = '';

$applicationImage = '';

if ($request->hasFile('file')) {

$applicationImage = ApplicationAdditional::where([

'application\_id' => $request->applicationId,

'additional\_id' => $request->documentId,

'id' => $request->imageId,

])->first();

$index = $request->index;

if ($applicationImage) {

$oldFilePath = $applicationImage->file\_path;

$status = $applicationImage->status == "Rejected" ? 'Reupload' : $applicationImage->status;

$image = $request->file('file');

$imageName = pathinfo($oldFilePath, PATHINFO\_FILENAME) . '.' . $image->getClientOriginalExtension();

$destinationPath = public\_path('/formupload');

$newFilePath = $imageName;

if ($image->move($destinationPath, $imageName)) {

// Update the file path in the database

$applicationImage->file\_path = $newFilePath; //$status

$applicationImage->status = $status;

$applicationImage->save();

$status\_image\_path = $this ->statusImage($status);

// if($status == 'Rejected'){

// $status\_image\_path = '<img src="' . asset('formupload/file\_cancel.png') . '" class="document\_status\_img">';

// }elseif($status == 'Reupload'){

// $status\_image\_path = '<img src="' . asset('formupload/stage\_reupload.png') . '" class="document\_status\_img">';

// }elseif($status == 'Accepted'){

// $status\_image\_path = '<img src="' . asset('formupload/file\_success.png') . '" class="document\_status\_img">';

// }elseif($status == 'Upload'){

// $status\_image\_path = '<img src="' . asset('formupload/file\_uploaded.png') . '" class="document\_status\_img">';

// }

$message = 'Successful image updated';

$status = true;

$fileExtension = pathinfo('formupload/' .$applicationImage->file\_path, PATHINFO\_EXTENSION);

$doc = ['doc', 'docx'];

if(in\_array($fileExtension ,$doc)){

$ImagePath = asset('formupload/word\_preview.png') ;

}else

if($fileExtension == 'pdf'){

$ImagePath = asset('formupload/pdf\_preview.png') ;

}else{

$ImagePath = asset('formupload/'. $applicationImage->file\_path) ;

}

$html = '' ;

// $reject\_class = '' ;

// if($applicationImage->status == "Rejected"){

// $reject\_class= 'border\_red' ;

// }

$html .= '<input type="checkbox" class="delete-checkbox addition\_delete\_checkbox" data-image-id="'.$applicationImage->id.'" data-file-id="uploaded\_file\_'.$index.'">';

$html .= '<input type="hidden" name="uploadform" value="uploaded\_file\_'.$index.'">';

$html .= '<img src="'.$ImagePath.'" style="max-width: 100%; height: 100%;">';

$html .= '<div id="upload\_file\_button\_'.$index.'" class="d-flex justify-content-center upload\_file\_button">';

$html .= '<button id="upload2\_preview\_'.$index.'" class="btn btn-outline-primary btn-sm" title="Preview" data-app-id ="' .$request->applicationId .'" data-document-id ="' .$request->documentId .'" data-file-id="uploaded\_file\_'.$index.'" data-image-id="'.$applicationImage->id.'" data-file-type="'.$applicationImage->fileExtension.'"><i class="fas fa-eye"></i></button>';

$html .= '<button id="upload2\_change\_'.$index.'" class="btn btn-outline-warning btn-sm" title="Change" data-app-id ="' .$request->applicationId .'" data-document-id ="' .$request->documentId .'" data-file-id="uploaded\_file\_'.$index.'" data-image-id="'.$applicationImage->id.'" data-file-type="'.$applicationImage->fileExtension.'"><i class="far fa-edit"></i></button>';

$html .= '<button id="upload2\_delete\_'.$index.'" class="btn btn-outline-danger btn-sm" title="Delete" data-app-id ="' .$request->applicationId .'" data-document-id ="' .$request->documentId .'" data-file-id="uploaded\_file\_'.$index.'" data-image-id="'.$applicationImage->id.'" data-file-type="'.$applicationImage->fileExtension.'"><i class="fas fa-trash"></i></button>';

$html .= '</div>';

$html .= '<div id="upload\_remark\_'.$index.'" class="upload\_remark">';

$html .= '<div class="form-group">';

// if($role != 1){

// $html .= '<div><input type="text" name="input\_remark" class="form-control form-control-sm" id="input\_remark\_'.$index.'" placeholder="Remark" value="'.$applicationImage->remark.'">';

// }

// $html .= '<input type="hidden" name="imageData" value="'.$applicationImage->id.'">';

// $html .= '</div></div></div>';

$hide = '';

if($role == 1){

$hide = 'readonly';

}

$html .= '<div>';

$html .= '<input type="text" name="input\_remark" class="form-control form-control-sm" id="input\_remark\_' . $index . '" placeholder="Remark" value="' .$applicationImage->remark. '" ' . $hide . '>';

$html .= '<input type="hidden" name="imageData" value="'.$applicationImage->id .'">';

$html .= '</div>';

$html .= '</div></div></div>';

} else {

$message = 'Failed to move the uploaded file';

}

} else {

return response()->json(['status' => false, 'message' => 'Image path not found']);

}

// if ($applicationImage) {

// $oldFilePath = $applicationImage->file\_path;

// $image = $request->file('file');

// $imageName = pathinfo($oldFilePath, PATHINFO\_FILENAME) . '.' . $image->getClientOriginalExtension();

// $destinationPath = public\_path('/formupload');

// $newFilePath = $imageName;

// if ($image->move($destinationPath, $imageName)) {

// // Update the file path in the database

// $applicationImage->file\_path = $newFilePath;

// $applicationImage->save();

// $message = 'Successful image updated';

// $status = true;

// } else {

// $message = 'Failed to move the uploaded file';

// }

// } else {

// return response()->json(['status' => false, 'message' => 'Image path not found']);

// }

return response()->json(['status' => $status, 'message' => $message, 'data' => $html, 'image\_path'=> $status\_image\_path]);

}

}

if (isset($request->applicationId) && isset($request->documentId) && isset($request->file)) {

if ($request->hasFile('file')) {

$image = $request->file('file');

$imageName = intval($user ->id). '\_'.intval($request->currentImageCount).'\_'.time() . '.' . $image->getClientOriginalExtension();

$destinationPath = public\_path('/formupload');

$image->move($destinationPath, $imageName);

$ApplicationImage = ApplicationAdditional::create(

['application\_id' => $request->applicationId, 'additional\_id' => $request->documentId, 'file\_path' => $imageName, 'status' => 'Upload']

);

$data = ApplicationAdditional::find($ApplicationImage->id) ?? '';

//start

$html = '';

$fileExtension = pathinfo('formupload/' .$ApplicationImage ->file\_path, PATHINFO\_EXTENSION);

$doc = ['doc', 'docx'];

if(in\_array($fileExtension ,$doc)){

$ImagePath = asset('formupload/word\_preview.png') ;

}else

if($fileExtension == 'pdf'){

$ImagePath = asset('formupload/pdf\_preview.png') ;

}else{

$ImagePath = asset('formupload/'. $ApplicationImage ->file\_path) ;

}

$status = $ApplicationImage->status ?? null ;

$status\_image\_path = $this ->statusImage($status);

// if($status == 'Rejected'){

// $status\_image\_path = '<img src="' . asset('formupload/file\_cancel.png') . '" class="document\_status\_img">';

// }elseif($status == 'Reupload'){

// $status\_image\_path = '<img src="' . asset('formupload/stage\_reupload.png') . '" class="document\_status\_img">';

// }elseif($status == 'Accepted'){

// $status\_image\_path == '<img src="' . asset('formupload/file\_success.png') . '" class="document\_status\_img">';

// }elseif($status == 'Upload'){

// $status\_image\_path == '<img src="' . asset('formupload/file\_uploaded.png') . '" class="document\_status\_img">';

// }

$index = intval($request->currentImageCount)?? 0 + 1 ;

$html .= '<div class="col-6 my-2 all\_file\_data">';

$html .= '<form>';

$html .= '<div id="uploaded\_file\_'.$index.'" class="uploaded\_file position-relative">';

$html .= '<input type="checkbox" class="delete-checkbox addition\_delete\_checkbox" data-image-id="'.$ApplicationImage ->id.'" data-file-id="uploaded\_file\_'.$index.'">';

$html .= '<input type="hidden" name="uploadform" value="uploaded\_file\_'.$index.'">';

$html .= '<img src="'.$ImagePath.'" style="max-width: 100%; height: 100%;">';

$html .= '<div id="upload\_file\_button\_'.$index.'" class="d-flex justify-content-center upload\_file\_button">';

$html .= '<button id="upload2\_preview\_'.$index.'" class="btn btn-outline-primary btn-sm" title="Preview" data-app-id ="' .$request->applicationId .'" data-document-id ="' .$request->documentId .'" data-file-id="uploaded\_file\_'.$index.'" data-image-id="'.$ApplicationImage ->id.'" data-file-type="'.$ApplicationImage ->fileExtension.'"><i class="fas fa-eye"></i></button>';

$html .= '<button id="upload2\_change\_'.$index.'" class="btn btn-outline-warning btn-sm" title="Change" data-app-id ="' .$request->applicationId .'" data-document-id ="' .$request->documentId .'" data-file-id="uploaded\_file\_'.$index.'" data-image-id="'.$ApplicationImage ->id.'" data-file-type="'.$ApplicationImage ->fileExtension.'"><i class="far fa-edit"></i></button>';

$html .= '<button id="upload2\_delete\_'.$index.'" class="btn btn-outline-danger btn-sm" title="Delete" data-app-id ="' .$request->applicationId .'" data-document-id ="' .$request->documentId .'" data-file-id="uploaded\_file\_'.$index.'" data-image-id="'.$ApplicationImage ->id.'" data-file-type="'.$ApplicationImage ->fileExtension.'"><i class="fas fa-trash"></i></button>';

$html .= '</div>';

$html .= '<div id="upload\_remark\_'.$index.'" class="upload\_remark">';

$html .= '<div class="form-group">';

$hide = '' ;

if($role == 1){

$hide = 'readonly';

}

$html .= '<div>';

$html .= '<input type="text" name="input\_remark" class="form-control form-control-sm" id="input\_remark\_' . $index . '" placeholder="Remark" value="' . $ApplicationImage ->remark . '" ' . $hide . '>';

$html .= '<input type="hidden" name="imageData" value="' . $ApplicationImage ->id . '">';

$html .= '</div>';

$html .= '</div></div></div></form></div>';

//end

$data = $html;

// Save image path in the database if needed

// Example: Image::create(['path' => $imageName]);

return response()->json(['status' => true, 'data' => $data, 'image\_path' => $status\_image\_path]);

}

}

// $data = ApplicationFormImage::where(['application\_id' => $request->applicationId, 'form\_id' => $request->documentId])->get();

}

public function uploadImageGet1(Request $request)

{

$user = Auth::user() ?? '' ;

$role = $user ? ($user->roles[0]->id ?? '' ): '';

$status\_image\_path = '';

if (isset($request->file) && isset($request->applicationId) && isset($request->documentId) && isset($request->imageId)) {

$status = false;

$message = '';

$applicationImage = '';

if ($request->hasFile('file')) {

$applicationImage = ApplicationFormImage::where([

'application\_id' => $request->applicationId,

'form\_id' => $request->documentId,

'id' => $request->imageId,

])->first();

$index = $request->index;

if ($applicationImage) {

$oldFilePath = $applicationImage->file\_path;

$status = $applicationImage->status == "Rejected" ? 'Reupload' : $applicationImage->status;

// dd($applicationImage->status);

$image = $request->file('file');

$imageName = pathinfo($oldFilePath, PATHINFO\_FILENAME) . '.' . $image->getClientOriginalExtension();

$destinationPath = public\_path('/formupload');

$newFilePath = $imageName;

if ($image->move($destinationPath, $imageName)) {

// Update the file path in the database

$applicationImage->file\_path = $newFilePath; //$status

$applicationImage->status = $status;

$applicationImage->save();

$status\_image\_path = $this ->statusImage($status);

// if($status == 'Rejected'){

// $status\_image\_path = '<img src="' . asset('formupload/file\_cancel.png') . '" class="document\_status\_img">';

// }elseif($status == 'Reupload'){

// $status\_image\_path = '<img src="' . asset('formupload/stage\_reupload.png') . '" class="document\_status\_img">';

// }elseif($status == 'Accepted'){

// $status\_image\_path = '<img src="' . asset('formupload/file\_success.png') . '" class="document\_status\_img">';

// }elseif($status == 'Upload'){

// $status\_image\_path = '<img src="' . asset('formupload/file\_uploaded.png') . '" class="document\_status\_img">';

// }

$message = 'Successful image updated';

$status = true;

$fileExtension = pathinfo('formupload/' .$applicationImage->file\_path, PATHINFO\_EXTENSION);

$doc = ['doc', 'docx'];

if(in\_array($fileExtension ,$doc)){

$ImagePath = asset('formupload/word\_preview.png') ;

}else

if($fileExtension == 'pdf'){

$ImagePath = asset('formupload/pdf\_preview.png') ;

}else{

$ImagePath = asset('formupload/'. $applicationImage->file\_path) ;

}

$html = '' ;

$html .= '<input type="checkbox" class="delete-checkbox form\_delete-checkbox" data-image-id="'.$applicationImage->id.'" data-file-id="uploaded\_file\_'.$index.'">';

$html .= '<input type="hidden" name="uploadform" value="uploaded\_file\_'.$index.'">';

$html .= '<img src="'.$ImagePath.'" style="max-width: 100%; height: 100%;">';

$html .= '<div id="upload\_file\_button\_'.$index.'" class="d-flex justify-content-center upload\_file\_button">';

$html .= '<button id="upload1\_preview\_'.$index.'" class="btn btn-outline-primary btn-sm" title="Preview" data-app-id ="' .$request->applicationId .'" data-document-id ="' .$request->documentId .'" data-file-id="uploaded\_file\_'.$index.'" data-image-id="'.$applicationImage->id.'" data-file-type="'.$applicationImage->fileExtension.'"><i class="fas fa-eye"></i></button>';

$html .= '<button id="upload1\_change\_'.$index.'" class="btn btn-outline-warning btn-sm" title="Change" data-app-id ="' .$request->applicationId .'" data-document-id ="' .$request->documentId .'" data-file-id="uploaded\_file\_'.$index.'" data-image-id="'.$applicationImage->id.'" data-file-type="'.$applicationImage->fileExtension.'"><i class="far fa-edit"></i></button>';

$html .= '<button id="upload1\_delete\_'.$index.'" class="btn btn-outline-danger btn-sm" title="Delete" data-app-id ="' .$request->applicationId .'" data-document-id ="' .$request->documentId .'" data-file-id="uploaded\_file\_'.$index.'" data-image-id="'.$applicationImage->id.'" data-file-type="'.$applicationImage->fileExtension.'"><i class="fas fa-trash"></i></button>';

$html .= '</div>';

$html .= '<div id="upload\_remark\_'.$index.'" class="upload\_remark">';

$html .= '<div class="form-group">';

// if($role != 1){

// $html .= '<div><input type="text" name="input\_remark" class="form-control form-control-sm" id="input\_remark\_'.$index.'" placeholder="Remark" value="'.$applicationImage->remark.'">';

// }

// $html .= '<input type="hidden" name="imageData" value="'.$applicationImage->id.'">';

// $html .= '</div></div></div>';

if($role == 1){

$hide = 'readonly';

}

$html .= '<div>';

$html .= '<input type="text" name="input\_remark" class="form-control form-control-sm" id="input\_remark\_' . $index . '" placeholder="Remark" value="' .$applicationImage->remark. '" ' . $hide . '>';

$html .= '<input type="hidden" name="imageData" value="'.$applicationImage->id .'">';

$html .= '</div>';

$html .= '</div></div></div>';

} else {

$message = 'Failed to move the uploaded file';

}

} else {

return response()->json(['status' => false, 'message' => 'Image path not found']);

}

return response()->json(['status' => $status, 'message' => $message, 'data' => $html, 'imaget\_path' =>$status\_image\_path ]);

}

}

if (isset($request->applicationId) && isset($request->documentId) && isset($request->file)) {

if ($request->hasFile('file')) {

$image = $request->file('file');

$imageName = intval($user ->id). '\_'.intval($request->currentImageCount).'\_'. time() . '.' . $image->getClientOriginalExtension();

$destinationPath = public\_path('/formupload');

$image->move($destinationPath, $imageName);

$ApplicationImage = ApplicationFormImage::create(

['application\_id' => $request->applicationId, 'form\_id' => $request->documentId, 'file\_path' => $imageName, 'status' => 'Upload']

);

$data = ApplicationFormImage::find($ApplicationImage->id) ?? '';

//start

$html = '';

$fileExtension = pathinfo('formupload/' .$ApplicationImage ->file\_path, PATHINFO\_EXTENSION);

$doc = ['doc', 'docx'];

if(in\_array($fileExtension ,$doc)){

$ImagePath = asset('formupload/word\_preview.png') ;

}else

if($fileExtension == 'pdf'){

$ImagePath = asset('formupload/pdf\_preview.png') ;

}else{

$ImagePath = asset('formupload/'. $ApplicationImage ->file\_path) ;

}

$status = $ApplicationImage->status ?? null ;

$status\_image\_path = $this ->statusImage($status);

// if($status == 'Rejected'){

// $status\_image\_path = '<img src="' . asset('formupload/file\_cancel.png') . '" class="document\_status\_img">';

// }elseif($status == 'Reupload'){

// $status\_image\_path = '<img src="' . asset('formupload/stage\_reupload.png') . '" class="document\_status\_img">';

// }elseif($status == 'Accepted'){

// $status\_image\_path = '<img src="' . asset('formupload/file\_success.png') . '" class="document\_status\_img">';

// }elseif($status == 'Upload'){

// $status\_image\_path = '<img src="' . asset('formupload/file\_uploaded.png') . '" class="document\_status\_img">';

// }

$index = intval($request->currentImageCount)?? 0 + 1 ;

$html .= '<div class="col-6 my-2 all\_file\_data">';

$html .= '<form>';

$html .= '<div id="uploaded\_file\_'.$index.'" class="uploaded\_file position-relative">';

$html .= '<input type="checkbox" class="delete-checkbox form\_delete-checkbox" data-image-id="'.$ApplicationImage ->id.'" data-file-id="uploaded\_file\_'.$index.'">';

$html .= '<input type="hidden" name="uploadform" value="uploaded\_file\_'.$index.'">';

$html .= '<img src="'.$ImagePath.'" style="max-width: 100%; height: 100%;">';

$html .= '<div id="upload\_file\_button\_'.$index.'" class="d-flex justify-content-center upload\_file\_button">';

$html .= '<button id="upload1\_preview\_'.$index.'" class="btn btn-outline-primary btn-sm" title="Preview" data-app-id ="' .$request->applicationId .'" data-document-id ="' .$request->documentId .'" data-file-id="uploaded\_file\_'.$index.'" data-image-id="'.$ApplicationImage ->id.'" data-file-type="'.$ApplicationImage ->fileExtension.'"><i class="fas fa-eye"></i></button>';

$html .= '<button id="upload1\_change\_'.$index.'" class="btn btn-outline-warning btn-sm" title="Change" data-app-id ="' .$request->applicationId .'" data-document-id ="' .$request->documentId .'" data-file-id="uploaded\_file\_'.$index.'" data-image-id="'.$ApplicationImage ->id.'" data-file-type="'.$ApplicationImage ->fileExtension.'"><i class="far fa-edit"></i></button>';

$html .= '<button id="upload1\_delete\_'.$index.'" class="btn btn-outline-danger btn-sm" title="Delete" data-app-id ="' .$request->applicationId .'" data-document-id ="' .$request->documentId .'" data-file-id="uploaded\_file\_'.$index.'" data-image-id="'.$ApplicationImage ->id.'" data-file-type="'.$ApplicationImage ->fileExtension.'"><i class="fas fa-trash"></i></button>';

$html .= '</div>';

$html .= '<div id="upload\_remark\_'.$index.'" class="upload\_remark">';

$html .= '<div class="form-group">';

$hide = '' ;

if($role == 1){

$hide = 'readonly';

}

$html .= '<div>';

$html .= '<input type="text" name="input\_remark" class="form-control form-control-sm" id="input\_remark\_' . $index . '" placeholder="Remark" value="' . $ApplicationImage ->remark . '" ' . $hide . '>';

$html .= '<input type="hidden" name="imageData" value="' . $ApplicationImage ->id . '">';

$html .= '</div>';

$html .= '</div></div></div></form></div>';

//end

$data = $html;

return response()->json(['status' => true, 'data' => $data, 'image\_path' => $status\_image\_path]);

}

}

$data = ApplicationFormImage::where(['application\_id' => $request->applicationId, 'form\_id' => $request->documentId])->get();

$html = '';

$reject\_class = '' ;

// if($applicationImage->status == "Rejected"){

// $reject\_class= 'border\_red' ;

// }

// $reject\_class = 'border\_red' ;

// Loop through each item in $dynamicData

foreach ($data as $index => $item) {

$index++;

$fileExtension = pathinfo('formupload/' .$item['file\_path'], PATHINFO\_EXTENSION);

$doc = ['doc', 'docx'];

if(in\_array($fileExtension ,$doc)){

$ImagePath = asset('formupload/word\_preview.png') ;

}else

if($fileExtension == 'pdf'){

$ImagePath = asset('formupload/pdf\_preview.png') ;

}else{

$ImagePath = asset('formupload/'. $item['file\_path']) ;

}

$reject\_class = '' ;

if($item['status'] == "Rejected"){

$reject\_class= 'border\_red' ;

}

$html .= '<div class="col-6 my-2 all\_file\_data">';

$html .= '<form>';

$html .= '<div id="uploaded\_file\_'.$index.'" class="uploaded\_file position-relative '.$reject\_class .'">';

$html .= '<input type="checkbox" class="delete-checkbox form\_delete-checkbox" data-image-id="'.$item['id'].'" data-file-id="uploaded\_file\_'.$index.'">';

$html .= '<input type="hidden" name="uploadform" value="uploaded\_file\_'.$index.'">';

$html .= '<img src="'.$ImagePath.'" style="max-width: 100%; height: 100%;">';

$html .= '<div id="upload\_file\_button\_'.$index.'" class="d-flex justify-content-center upload\_file\_button">';

$html .= '<button id="upload1\_preview\_'.$index.'" class="btn btn-outline-primary btn-sm" title="Preview" data-app-id ="' .$request->applicationId .'" data-document-id ="' .$request->documentId .'" data-file-id="uploaded\_file\_'.$index.'" data-image-id="'.$item['id'].'" data-file-type="'.$item['fileExtension'].'"><i class="fas fa-eye"></i></button>';

$html .= '<button id="upload1\_change\_'.$index.'" class="btn btn-outline-warning btn-sm" title="Change" data-app-id ="' .$request->applicationId .'" data-document-id ="' .$request->documentId .'" data-file-id="uploaded\_file\_'.$index.'" data-image-id="'.$item['id'].'" data-file-type="'.$item['fileExtension'].'"><i class="far fa-edit"></i></button>';

$html .= '<button id="upload1\_delete\_'.$index.'" class="btn btn-outline-danger btn-sm" title="Delete" data-app-id ="' .$request->applicationId .'" data-document-id ="' .$request->documentId .'" data-file-id="uploaded\_file\_'.$index.'" data-image-id="'.$item['id'].'" data-file-type="'.$item['fileExtension'].'"><i class="fas fa-trash"></i></button>';

$html .= '</div>';

$html .= '<div id="upload\_remark\_'.$index.'" class="upload\_remark">';

$html .= '<div class="form-group">';

$hide = '' ;

if($role == 1){

$hide = 'readonly';

}

$html .= '<div>';

$html .= '<input type="text" name="input\_remark" class="form-control form-control-sm" id="input\_remark\_' . $index . '" placeholder="Remark" value="' . $item['remark'] . '" ' . $hide . '>';

$html .= '<input type="hidden" name="imageData" value="' . $item['id'] . '">';

$html .= '</div>';

$html .= '</div></div></div></form></div>';

}

$data = $html ;

// $data = ApplicationFormImage::where(['application\_id' => $request->applicationId, 'form\_id' => $request->documentId])->get();

// dd($data );

return response()->json(['status' => true, 'data' => $data]);

}

public function UploadAdditionalFormImageGet(Request $request)

{

$user = Auth::user() ?? '' ;

$role = $user ? ($user->roles[0]->id ?? '' ): '';

$status\_image\_path = '';

if (isset($request->file) && isset($request->applicationId) && isset($request->documentId) && isset($request->imageId)) {

$status = false;

$message = '';

$applicationImage = '';

if ($request->hasFile('file')) {

$applicationImage = ApplicationAdditionFormDocument::where([

'application\_id' => $request->applicationId,

'additional\_form\_id' => $request->documentId,

'id' => $request->imageId,

])->first();

$index = $request->index;

if ($applicationImage) {

$oldFilePath = $applicationImage->file\_path;

$status = $applicationImage->status == "Rejected" ? 'Reupload' : $applicationImage->status;

// dd($applicationImage->status);

$image = $request->file('file');

$imageName = pathinfo($oldFilePath, PATHINFO\_FILENAME) . '.' . $image->getClientOriginalExtension();

$destinationPath = public\_path('/formupload');

$newFilePath = $imageName;

if ($image->move($destinationPath, $imageName)) {

// Update the file path in the database

$applicationImage->file\_path = $newFilePath; //$status

$applicationImage->status = $status;

$applicationImage->save();

$status\_image\_path = $this ->statusImage($status);

// if($status == 'Rejected'){

// $status\_image\_path = '<img src="' . asset('formupload/file\_cancel.png') . '" class="document\_status\_img">';

// }elseif($status == 'Reupload'){

// $status\_image\_path = '<img src="' . asset('formupload/stage\_reupload.png') . '" class="document\_status\_img">';

// }elseif($status == 'Accepted'){

// $status\_image\_path = '<img src="' . asset('formupload/file\_success.png') . '" class="document\_status\_img">';

// }elseif($status == 'Upload'){

// $status\_image\_path = '<img src="' . asset('formupload/file\_uploaded.png') . '" class="document\_status\_img">';

// }

$message = 'Successful image updated';

$status = true;

$fileExtension = pathinfo('formupload/' .$applicationImage->file\_path, PATHINFO\_EXTENSION);

$doc = ['doc', 'docx'];

if(in\_array($fileExtension ,$doc)){

$ImagePath = asset('formupload/word\_preview.png') ;

}else

if($fileExtension == 'pdf'){

$ImagePath = asset('formupload/pdf\_preview.png') ;

}else{

$ImagePath = asset('formupload/'. $applicationImage->file\_path) ;

}

$html = '' ;

$html .= '<input type="checkbox" class="delete-checkbox form\_additional\_delete-checkbox" data-image-id="'.$applicationImage->id.'" data-file-id="uploaded\_file\_'.$index.'">';

$html .= '<input type="hidden" name="uploadform" value="uploaded\_file\_'.$index.'">';

$html .= '<img src="'.$ImagePath.'" style="max-width: 100%; height: 100%;">';

$html .= '<div id="upload\_file\_button\_'.$index.'" class="d-flex justify-content-center upload\_file\_button">';

$html .= '<button id="upload3\_preview\_'.$index.'" class="btn btn-outline-primary btn-sm" title="Preview" data-app-id ="' .$request->applicationId .'" data-document-id ="' .$request->documentId .'" data-file-id="uploaded\_file\_'.$index.'" data-image-id="'.$applicationImage->id.'" data-file-type="'.$applicationImage->fileExtension.'"><i class="fas fa-eye"></i></button>';

$html .= '<button id="upload3\_change\_'.$index.'" class="btn btn-outline-warning btn-sm" title="Change" data-app-id ="' .$request->applicationId .'" data-document-id ="' .$request->documentId .'" data-file-id="uploaded\_file\_'.$index.'" data-image-id="'.$applicationImage->id.'" data-file-type="'.$applicationImage->fileExtension.'"><i class="far fa-edit"></i></button>';

$html .= '<button id="upload3\_delete\_'.$index.'" class="btn btn-outline-danger btn-sm" title="Delete" data-app-id ="' .$request->applicationId .'" data-document-id ="' .$request->documentId .'" data-file-id="uploaded\_file\_'.$index.'" data-image-id="'.$applicationImage->id.'" data-file-type="'.$applicationImage->fileExtension.'"><i class="fas fa-trash"></i></button>';

$html .= '</div>';

$html .= '<div id="upload\_remark\_'.$index.'" class="upload\_remark">';

$html .= '<div class="form-group">';

// if($role != 1){

// $html .= '<div><input type="text" name="input\_remark" class="form-control form-control-sm" id="input\_remark\_'.$index.'" placeholder="Remark" value="'.$applicationImage->remark.'">';

// }

// $html .= '<input type="hidden" name="imageData" value="'.$applicationImage->id.'">';

// $html .= '</div></div></div>';

if($role == 1){

$hide = 'readonly';

}

$html .= '<div>';

$html .= '<input type="text" name="input\_remark" class="form-control form-control-sm" id="input\_remark\_' . $index . '" placeholder="Remark" value="' .$applicationImage->remark. '" ' . $hide . '>';

$html .= '<input type="hidden" name="imageData" value="'.$applicationImage->id .'">';

$html .= '</div>';

$html .= '</div></div></div>';

} else {

$message = 'Failed to move the uploaded file';

}

} else {

return response()->json(['status' => false, 'message' => 'Image path not found']);

}

return response()->json(['status' => $status, 'message' => $message, 'data' => $html, 'imaget\_path' =>$status\_image\_path ]);

}

}

if (isset($request->applicationId) && isset($request->documentId) && isset($request->file)) {

if ($request->hasFile('file')) {

$image = $request->file('file');

$imageName = intval($user ->id). '\_'.intval($request->currentImageCount).'\_'. time() . '.' . $image->getClientOriginalExtension();

$destinationPath = public\_path('/formupload');

$image->move($destinationPath, $imageName);

$ApplicationImage = ApplicationAdditionFormDocument::create(

['application\_id' => $request->applicationId, 'additional\_form\_id' => $request->documentId, 'file\_path' => $imageName, 'status' => 'Upload']

);

$data = ApplicationAdditionFormDocument::find($ApplicationImage->id) ?? '';

//start

$html = '';

$fileExtension = pathinfo('formupload/' .$ApplicationImage ->file\_path, PATHINFO\_EXTENSION);

$doc = ['doc', 'docx'];

if(in\_array($fileExtension ,$doc)){

$ImagePath = asset('formupload/word\_preview.png') ;

}else

if($fileExtension == 'pdf'){

$ImagePath = asset('formupload/pdf\_preview.png') ;

}else{

$ImagePath = asset('formupload/'. $ApplicationImage ->file\_path) ;

}

$status = $ApplicationImage->status ?? null ;

$status\_image\_path = $this ->statusImage($status);

// if($status == 'Rejected'){

// $status\_image\_path = '<img src="' . asset('formupload/file\_cancel.png') . '" class="document\_status\_img">';

// }elseif($status == 'Reupload'){

// $status\_image\_path = '<img src="' . asset('formupload/stage\_reupload.png') . '" class="document\_status\_img">';

// }elseif($status == 'Accepted'){

// $status\_image\_path = '<img src="' . asset('formupload/file\_success.png') . '" class="document\_status\_img">';

// }elseif($status == 'Upload'){

// $status\_image\_path = '<img src="' . asset('formupload/file\_uploaded.png') . '" class="document\_status\_img">';

// }

$index = intval($request->currentImageCount)?? 0 + 1 ;

$html .= '<div class="col-6 my-2 all\_file\_data">';

$html .= '<form>';

$html .= '<div id="uploaded\_file\_'.$index.'" class="uploaded\_file position-relative">';

$html .= '<input type="checkbox" class="delete-checkbox form\_additional\_delete-checkbox" data-image-id="'.$ApplicationImage ->id.'" data-file-id="uploaded\_file\_'.$index.'">';

$html .= '<input type="hidden" name="uploadform" value="uploaded\_file\_'.$index.'">';

$html .= '<img src="'.$ImagePath.'" style="max-width: 100%; height: 100%;">';

$html .= '<div id="upload\_file\_button\_'.$index.'" class="d-flex justify-content-center upload\_file\_button">';

$html .= '<button id="upload3\_preview\_'.$index.'" class="btn btn-outline-primary btn-sm" title="Preview" data-app-id ="' .$request->applicationId .'" data-document-id ="' .$request->documentId .'" data-file-id="uploaded\_file\_'.$index.'" data-image-id="'.$ApplicationImage ->id.'" data-file-type="'.$ApplicationImage ->fileExtension.'"><i class="fas fa-eye"></i></button>';

$html .= '<button id="upload3\_change\_'.$index.'" class="btn btn-outline-warning btn-sm" title="Change" data-app-id ="' .$request->applicationId .'" data-document-id ="' .$request->documentId .'" data-file-id="uploaded\_file\_'.$index.'" data-image-id="'.$ApplicationImage ->id.'" data-file-type="'.$ApplicationImage ->fileExtension.'"><i class="far fa-edit"></i></button>';

$html .= '<button id="upload3\_delete\_'.$index.'" class="btn btn-outline-danger btn-sm" title="Delete" data-app-id ="' .$request->applicationId .'" data-document-id ="' .$request->documentId .'" data-file-id="uploaded\_file\_'.$index.'" data-image-id="'.$ApplicationImage ->id.'" data-file-type="'.$ApplicationImage ->fileExtension.'"><i class="fas fa-trash"></i></button>';

$html .= '</div>';

$html .= '<div id="upload\_remark\_'.$index.'" class="upload\_remark">';

$html .= '<div class="form-group">';

$hide = '' ;

if($role == 1){

$hide = 'readonly';

}

$html .= '<div>';

$html .= '<input type="text" name="input\_remark" class="form-control form-control-sm" id="input\_remark\_' . $index . '" placeholder="Remark" value="' . $ApplicationImage ->remark . '" ' . $hide . '>';

$html .= '<input type="hidden" name="imageData" value="' . $ApplicationImage ->id . '">';

$html .= '</div>';

$html .= '</div></div></div></form></div>';

//end

$data = $html;

return response()->json(['status' => true, 'data' => $data, 'image\_path' => $status\_image\_path]);

}

}

$data = ApplicationAdditionFormDocument::where(['application\_id' => $request->applicationId, 'additional\_form\_id' => $request->documentId])->get();

$html = '';

$reject\_class = '' ;

// if($applicationImage->status == "Rejected"){

// $reject\_class= 'border\_red' ;

// }

// $reject\_class = 'border\_red' ;

// Loop through each item in $dynamicData

foreach ($data as $index => $item) {

$index++;

$fileExtension = pathinfo('formupload/' .$item['file\_path'], PATHINFO\_EXTENSION);

$doc = ['doc', 'docx'];

if(in\_array($fileExtension ,$doc)){

$ImagePath = asset('formupload/word\_preview.png') ;

}else

if($fileExtension == 'pdf'){

$ImagePath = asset('formupload/pdf\_preview.png') ;

}else{

$ImagePath = asset('formupload/'. $item['file\_path']) ;

}

$reject\_class = '' ;

if($item['status'] == "Rejected"){

$reject\_class= 'border\_red' ;

}

$html .= '<div class="col-6 my-2 all\_file\_data">';

$html .= '<form>';

$html .= '<div id="uploaded\_file\_'.$index.'" class="uploaded\_file position-relative '.$reject\_class .'">';

$html .= '<input type="checkbox" class="delete-checkbox form\_additional\_delete-checkbox" data-image-id="'.$item['id'].'" data-file-id="uploaded\_file\_'.$index.'">';

$html .= '<input type="hidden" name="uploadform" value="uploaded\_file\_'.$index.'">';

$html .= '<img src="'.$ImagePath.'" style="max-width: 100%; height: 100%;">';

$html .= '<div id="upload\_file\_button\_'.$index.'" class="d-flex justify-content-center upload\_file\_button">';

$html .= '<button id="upload3\_preview\_'.$index.'" class="btn btn-outline-primary btn-sm" title="Preview" data-app-id ="' .$request->applicationId .'" data-document-id ="' .$request->documentId .'" data-file-id="uploaded\_file\_'.$index.'" data-image-id="'.$item['id'].'" data-file-type="'.$item['fileExtension'].'"><i class="fas fa-eye"></i></button>';

$html .= '<button id="upload3\_change\_'.$index.'" class="btn btn-outline-warning btn-sm" title="Change" data-app-id ="' .$request->applicationId .'" data-document-id ="' .$request->documentId .'" data-file-id="uploaded\_file\_'.$index.'" data-image-id="'.$item['id'].'" data-file-type="'.$item['fileExtension'].'"><i class="far fa-edit"></i></button>';

$html .= '<button id="upload3\_delete\_'.$index.'" class="btn btn-outline-danger btn-sm" title="Delete" data-app-id ="' .$request->applicationId .'" data-document-id ="' .$request->documentId .'" data-file-id="uploaded\_file\_'.$index.'" data-image-id="'.$item['id'].'" data-file-type="'.$item['fileExtension'].'"><i class="fas fa-trash"></i></button>';

$html .= '</div>';

$html .= '<div id="upload\_remark\_'.$index.'" class="upload\_remark">';

$html .= '<div class="form-group">';

$hide = '' ;

if($role == 1){

$hide = 'readonly';

}

$html .= '<div>';

$html .= '<input type="text" name="input\_remark" class="form-control form-control-sm" id="input\_remark\_' . $index . '" placeholder="Remark" value="' . $item['remark'] . '" ' . $hide . '>';

$html .= '<input type="hidden" name="imageData" value="' . $item['id'] . '">';

$html .= '</div>';

$html .= '</div></div></div></form></div>';

}

$data = $html ;

// $data = ApplicationFormImage::where(['application\_id' => $request->applicationId, 'form\_id' => $request->documentId])->get();

// dd($data );

return response()->json(['status' => true, 'data' => $data]);

}

public function uploadImageDelete(Request $request)

{

// $data = ApplicationImage::where(['application\_id'=> $request->applicationId,'document\_id' => $request->documentId , 'id' =>$request->image\_id ])->delete();

$image = ApplicationImage::findOrFail($request->image\_id);

// Determine the path to the image file

$imagePath = public\_path('formupload/' . $image->path); // Adjust the path based on your structure

// Delete the image file from the public directory

if (File::exists($imagePath)) {

File::delete($imagePath);

}

// Delete the image record from the database

$image->delete();

// Delete the image record from the database

// $image->delete();

return response()->json(['status' => true, 'data' => 'true']);

}

public function uploadImageDelete1(Request $request)

{

// $data = ApplicationImage::where(['application\_id'=> $request->applicationId,'document\_id' => $request->documentId , 'id' =>$request->image\_id ])->delete();

$image = ApplicationFormImage::findOrFail($request->image\_id);

// Determine the path to the image file

$imagePath = public\_path('formupload/' . $image->path); // Adjust the path based on your structure

// Delete the image file from the public directory

if (File::exists($imagePath)) {

File::delete($imagePath);

}

// Delete the image record from the database

$image->delete();

// Delete the image record from the database

// $image->delete();

return response()->json(['status' => true, 'data' => 'true']);

}

public function uploadImageDelete2(Request $request)

{

// $data = ApplicationImage::where(['application\_id'=> $request->applicationId,'document\_id' => $request->documentId , 'id' =>$request->image\_id ])->delete();

$image = ApplicationAdditional::findOrFail($request->image\_id);

// Determine the path to the image file

$imagePath = public\_path('formupload/' . $image->path); // Adjust the path based on your structure

// Delete the image file from the public directory

if (File::exists($imagePath)) {

File::delete($imagePath);

}

// Delete the image record from the database

$image->delete();

// Delete the image record from the database

// $image->delete();

return response()->json(['status' => true, 'data' => 'true']);

}

public function uploadImageDelete3(Request $request)

{

// $data = ApplicationImage::where(['application\_id'=> $request->applicationId,'document\_id' => $request->documentId , 'id' =>$request->image\_id ])->delete();

$image = ApplicationAdditionFormDocument::findOrFail($request->image\_id);

// Determine the path to the image file

$imagePath = public\_path('formupload/' . $image->file\_path); // Adjust the path based on your structure

// Delete the image file from the public directory

if (File::exists($imagePath)) {

File::delete($imagePath);

}

// Delete the image record from the database

$image->delete();

// Delete the image record from the database

// $image->delete();

return response()->json(['status' => true, 'data' => 'true']);

}

public function uploadImageDeletes(Request $request)

{

// Fetch image records based on provided IDs

$imageIds = $request->input('image\_ids', []);

// Ensure image IDs are provided

if (empty($imageIds)) {

return response()->json(['status' => false, 'message' => 'No image IDs provided.']);

}

// Retrieve the images from the database

$images = ApplicationImage::whereIn('id', $imageIds)->get();

// Iterate through each image record

foreach ($images as $image) {

// Determine the path to the image file

$imagePath = public\_path('formupload/' . $image->path);

// Delete the image file from the public directory if it exists

if (File::exists($imagePath)) {

File::delete($imagePath);

}

// Delete the image record from the database

$image->delete();

}

return response()->json(['status' => true, 'message' => 'Images deleted successfully.']);

}

public function uploadFormImageGetDeletes(Request $request)

{

// Fetch image records based on provided IDs

$imageIds = $request->input('image\_ids', []);

// Ensure image IDs are provided

if (empty($imageIds)) {

return response()->json(['status' => false, 'message' => 'No image IDs provided.']);

}

// Retrieve the images from the database

$images = ApplicationFormImage::whereIn('id', $imageIds)->get();

// Iterate through each image record

foreach ($images as $image) {

// Determine the path to the image file

$imagePath = public\_path('formupload/' . $image->path);

// Delete the image file from the public directory if it exists

if (File::exists($imagePath)) {

File::delete($imagePath);

}

// Delete the image record from the database

$image->delete();

}

return response()->json(['status' => true, 'message' => 'Images deleted successfully.']);

}

public function uploadAdditionalImageGetDeletes(Request $request)

{

// Fetch image records based on provided IDs

$imageIds = $request->input('image\_ids', []);

// Ensure image IDs are provided

if (empty($imageIds)) {

return response()->json(['status' => false, 'message' => 'No image IDs provided.']);

}

// Retrieve the images from the database

$images = ApplicationAdditional::whereIn('id', $imageIds)->get();

// Iterate through each image record

foreach ($images as $image) {

// Determine the path to the image file

$imagePath = public\_path('formupload/' . $image->path);

// Delete the image file from the public directory if it exists

if (File::exists($imagePath)) {

File::delete($imagePath);

}

// Delete the image record from the database

$image->delete();

}

return response()->json(['status' => true, 'message' => 'Images deleted successfully.']);

}

public function uploadAdditionalFormImageGetDeletes(Request $request)

{

// Fetch image records based on provided IDs

$imageIds = $request->input('image\_ids', []);

// Ensure image IDs are provided

if (empty($imageIds)) {

return response()->json(['status' => false, 'message' => 'No image IDs provided.']);

}

// Retrieve the images from the database

$images = ApplicationAdditionFormDocument::whereIn('id', $imageIds)->get();

// Iterate through each image record

foreach ($images as $image) {

// Determine the path to the image file

$imagePath = public\_path('formupload/' . $image->path);

// Delete the image file from the public directory if it exists

if (File::exists($imagePath)) {

File::delete($imagePath);

}

// Delete the image record from the database

$image->delete();

}

return response()->json(['status' => true, 'message' => 'Images deleted successfully.']);

}

public function remarkUpdated(Request $request)

{

$applicationId = $request->input('applicationId');

$documentId = $request->input('documentId');

$inputNames = json\_decode($request->input('inputNames'), true);

if (!$applicationId || !$documentId || !$inputNames) {

return response()->json(['status' => true, 'message' => 'Data Not found.', 'data' => false]);

}

// Initialize an array to hold parsed data

$parsedData = [];

// Iterate through each serialized string and parse it

foreach ($inputNames as $serializedString) {

parse\_str($serializedString, $parsedArray);

$parsedData[] = $parsedArray;

}

$user = Auth::user() ?? '' ;

$role = $user ? ($user->roles[0]->id ?? '' ): '';

if($role != 1){

// Process each parsed form data

foreach ($parsedData as $formData) {

if(isset($formData['imageData']) && $formData['imageData']!= '' ){

// Example: Save each form data to the database

$model = ApplicationImage::where([

'application\_id' => $applicationId,

'document\_id' => $documentId,

'id' => $formData['imageData'],

])->update([

'remark' => $formData['input\_remark'],

]);

}

}

}

return response()->json(['status' => true, 'message' => 'successfully updated stauts.', 'data' => true]);

$inputNames = $request->input('inputNames', []); // Get the inputNames array from the request

}

public function formRemarkUpdated(Request $request)

{

$applicationId = $request->input('applicationId');

$documentId = $request->input('documentId');

$inputNames = json\_decode($request->input('inputNames'), true);

if (!$applicationId || !$documentId || !$inputNames) {

return response()->json(['status' => true, 'message' => 'Data Not found.', 'data' => false]);

}

// Initialize an array to hold parsed data

$parsedData = [];

// Iterate through each serialized string and parse it

foreach ($inputNames as $serializedString) {

parse\_str($serializedString, $parsedArray);

$parsedData[] = $parsedArray;

}

// Process each parsed form data

$user = Auth::user() ?? '' ;

$role = $user ? ($user->roles[0]->id ?? '' ): '';

if($role != 1){

foreach ($parsedData as $formData) {

// Example: Save each form data to the database

$model = ApplicationFormImage::where([

'application\_id' => $applicationId,

'form\_id' => $documentId,

'id' => $formData['imageData'],

])->update([

'remark' => $formData['input\_remark'],

]);

}

}

return response()->json(['status' => true, 'message' => 'successfully updated stauts.', 'data' => true]);

$inputNames = $request->input('inputNames', []); // Get the inputNames array from the request

}

public function additionalremarkUpdated(Request $request)

{

$applicationId = $request->input('applicationId');

$documentId = $request->input('documentId');

$inputNames = json\_decode($request->input('inputNames'), true);

// dd($inputNames);

if (!$applicationId || !$documentId || !$inputNames) {

return response()->json(['status' => true, 'message' => 'Data Not found.', 'data' => false]);

}

// Initialize an array to hold parsed data

$parsedData = [];

// Iterate through each serialized string and parse it

foreach ($inputNames as $serializedString) {

parse\_str($serializedString, $parsedArray);

$parsedData[] = $parsedArray;

}

// dd($parsedData);

// Process each parsed form data

$user = Auth::user() ?? '' ;

$role = $user ? ($user->roles[0]->id ?? '' ): '';

if($role != 1){

// Process each parsed form data

foreach ($parsedData as $formData) {

// dd($formData);

if(isset($formData['imageData']) && $formData['imageData']!= '' ){

// Example: Save each form data to the database

$model = ApplicationAdditional::where([

'application\_id' => $applicationId,

'additional\_id' => $documentId,

'id' => $formData['imageData'],

])->update([

'remark' => $formData['input\_remark'],

]);

}

}

}

return response()->json(['status' => true, 'message' => 'successfully updated stauts.', 'data' => true]);

$inputNames = $request->input('inputNames', []); // Get the inputNames array from the request

}

public function additionalFormRemarkUpdated(Request $request)

{

$applicationId = $request->input('applicationId');

$documentId = $request->input('documentId');

$inputNames = json\_decode($request->input('inputNames'), true);

// dd($inputNames);

if (!$applicationId || !$documentId || !$inputNames) {

return response()->json(['status' => true, 'message' => 'Data Not found.', 'data' => false]);

}

// Initialize an array to hold parsed data

$parsedData = [];

// Iterate through each serialized string and parse it

foreach ($inputNames as $serializedString) {

parse\_str($serializedString, $parsedArray);

$parsedData[] = $parsedArray;

}

// dd($parsedData);

// Process each parsed form data

$user = Auth::user() ?? '' ;

$role = $user ? ($user->roles[0]->id ?? '' ): '';

if($role != 1){

// Process each parsed form data

foreach ($parsedData as $formData) {

// dd($formData);

if(isset($formData['imageData']) && $formData['imageData']!= '' ){

// Example: Save each form data to the database

$model = ApplicationAdditionFormDocument::where([

'application\_id' => $applicationId,

'additional\_form\_id' => $documentId,

'id' => $formData['imageData'],

])->update([

'remark' => $formData['input\_remark'],

]);

}

}

}

return response()->json(['status' => true, 'message' => 'successfully updated stauts.', 'data' => true]);

$inputNames = $request->input('inputNames', []); // Get the inputNames array from the request

}

public function remarkUpdatedadmin(Request $request)

{

$applicationId = $request->input('applicationId');

$documentId = $request->input('documentId');

$imageId = $request->input('imageId');

$radio = $request->input('radio');

$admin\_remark = $request->input('admin\_remark');

$image\_path = '';

$data = [

'admin\_remark' => $admin\_remark,

];

// Check if 'radio' input exists and is not empty

if ($request->has('radio') && !empty($request->input('radio'))) {

$radio = $request->input('radio');

$data['status'] = $radio;

$image\_path = $this->statusImage($data['status'] );

}

if (!$applicationId || !$documentId || !$imageId ) {

return response()->json(['status' => true, 'message' => 'Data Not found.', 'data' => false]);

}

// Example: Save each form data to the database

$model = ApplicationImage::where([

'application\_id' => $applicationId,

'document\_id' => $documentId,

'id' => $imageId,

])->update($data);

return response()->json(['status' => true, 'message' => 'successfully updated stauts.', 'data' => true, 'image\_path'=> $image\_path]);

$inputNames = $request->input('inputNames', []); // Get the inputNames array from the request

}

public function remarkUpdatedadmin2(Request $request)

{

$applicationId = $request->input('applicationId');

$documentId = $request->input('documentId');

$imageId = $request->input('imageId');

$radio = $request->input('radio');

$admin\_remark = $request->input('admin\_remark');

$image\_path = '';

$data = [

'admin\_remark' => $admin\_remark,

];

// Check if 'radio' input exists and is not empty

if ($request->has('radio') && !empty($request->input('radio'))) {

$radio = $request->input('radio');

$data['status'] = $radio;

$image\_path = $this->statusImage($data['status'] );

}

if (!$applicationId || !$documentId || !$imageId ) {

return response()->json(['status' => true, 'message' => 'Data Not found.', 'data' => false]);

}

// Example: Save each form data to the database

$model = ApplicationAdditional::where([

'application\_id' => $applicationId,

'additional\_id' => $documentId,

'id' => $imageId,

])->update($data);

return response()->json(['status' => true, 'message' => 'successfully updated stauts.', 'data' => true, 'image\_path'=> $image\_path]);

$inputNames = $request->input('inputNames', []); // Get the inputNames array from the request

}

public function formRemarkUpdatedAdmin(Request $request)

{

$applicationId = $request->input('applicationId');

$documentId = $request->input('documentId');

$imageId = $request->input('imageId');

$radio = $request->input('radio');

$admin\_remark = $request->input('admin\_remark');

$image\_path = '';

if (!$applicationId || !$documentId || !$imageId) {

return response()->json(['status' => true, 'message' => 'Data Not found.', 'data' => false]);

}

$data = [

'admin\_remark' => $admin\_remark,

];

// Check if 'radio' input exists and is not empty

if ($request->has('radio') && !empty($request->input('radio'))) {

$radio = $request->input('radio');

$data['status'] = $radio;

$image\_path = $this->statusImage($data['status'] );

}

if (!$applicationId || !$documentId || !$imageId ) {

return response()->json(['status' => true, 'message' => 'Data Not found.', 'data' => false]);

}

// Example: Save each form data to the database

$model = ApplicationFormImage::where([

'application\_id' => $applicationId,

'form\_id' => $documentId,

'id' => $imageId,

])->update($data);

return response()->json(['status' => true, 'message' => 'successfully updated stauts.', 'data' => true, 'image\_path' => $image\_path]);

$inputNames = $request->input('inputNames', []); // Get the inputNames array from the request

}

public function additionalFormRemarkUpdatedAdmin(Request $request)

{

$applicationId = $request->input('applicationId');

$documentId = $request->input('documentId');

$imageId = $request->input('imageId');

$radio = $request->input('radio');

$admin\_remark = $request->input('admin\_remark');

$image\_path = '';

if (!$applicationId || !$documentId || !$imageId) {

return response()->json(['status' => true, 'message' => 'Data Not found.', 'data' => false]);

}

$data = [

'admin\_remark' => $admin\_remark,

];

// Check if 'radio' input exists and is not empty

if ($request->has('radio') && !empty($request->input('radio'))) {

$radio = $request->input('radio');

$data['status'] = $radio;

$image\_path = $this->statusImage($data['status'] );

}

if (!$applicationId || !$documentId || !$imageId ) {

return response()->json(['status' => true, 'message' => 'Data Not found.', 'data' => false]);

}

// Example: Save each form data to the database

$model = ApplicationAdditionFormDocument::where([

'application\_id' => $applicationId,

'additional\_form\_id' => $documentId,

'id' => $imageId,

])->update($data);

return response()->json(['status' => true, 'message' => 'successfully updated stauts.', 'data' => true, 'image\_path' => $image\_path]);

$inputNames = $request->input('inputNames', []); // Get the inputNames array from the request

}

public function documenUploadPreview(Request $request){

$data = ApplicationImage::where(['application\_id' => $request->appId, 'document\_id' => $request->documentId, 'id'=>$request->imageId])->first();

if(!$data){

return response()->json(['status' => false, 'data' => $data]) ;

}

$user = Auth::user() ?? '' ;

$role = $user ? ($user->roles[0]->id ?? '' ): '';

$html = '';

$file\_path = explode('.',$data->file\_path)[1] ?? '' ;

$doc = ['doc', 'docx'];

if(in\_array($file\_path ,$doc)){

$ImagePath = asset('formupload/' . $data->file\_path);

$html .= '<iframe id="docViewer" src="https://docs.google.com/gview?url=' . $ImagePath . '&embedded=true" width="100%" height="100%"></iframe>';

}else

if($file\_path == 'pdf'){

$ImagePath = asset('formupload/'. $data->file\_path) ;

$html .= ' <iframe id="pdfViewer" src="' . $ImagePath . '" width="100%" height="100%"></iframe>' ;

}else{

$ImagePath = asset('formupload/'. $data->file\_path) ;

$html .= '<img src="' . $ImagePath . '" style="width: 100%; height: auto;">';

}

$html .= '<div class="container-fluid" id="new\_added\_content\_' . $data->id . '">';

if( $role == 1){

// Pane section with radio buttons

$html .= '<div class="pane m-3">';

// $html .= '<label class="label"><span>None</span><input id="idk\_' . $data->id . '" class="input idk" name="radio" type="radio"' . ($data->status == '' ? 'checked="checked"' : '') . ' value=""></label>';

$html .= '<label class="label"><span>Accept</span><input id="yes\_' . $data->id . '" class="input yes" name="radio" type="radio"' . ($data->status == 'Verified' ? 'checked="checked"' : '') . ' value="Verified"></label>';

$html .= '<label class="label"><span>Reject</span><input id="no\_' . $data->id . '" class="input no" name="radio" type="radio" ' . ($data->status == 'Rejected' ? 'checked="checked"' : '') . ' value="Rejected"></label>';

$html .= '<span class="preview\_selection"></span>';

$html .= '</div>';

// Admin remark input section

$html .= '<div class="extra-input m-3" id="extra-input\_' . $data->id . '">';

$html .= '<input type="text" name="admin\_remark" id="extra-text\_' . $data->id . '" class="form-control extra\_input" placeholder="Reason" value="' . $data->admin\_remark . '">';

$html .= '</div>';

}else if($role != 1 && $data->status == "Rejected" ){

// Admin remark input section

$html .= '<div class="extra-input m-3" id="extra-input\_' . $data->id . '">';

$html .= '<input type="text" name="admin\_remark" id="extra-text\_' . $data->id . '" class="form-control extra\_input" placeholder="Reason" value="' . $data->admin\_remark . '" readonly>';

$html .= '</div>';

}

$html .= '</div>'; // Closing container-fluid

$data = $html ;

return response()->json(['status' => true, 'data' => $data]);

}

public function formUploadPreview(Request $request){

$data = ApplicationFormImage::where(['application\_id' => $request->appId, 'form\_id' => $request->documentId , 'id' => $request->imageId ])->first();

if(!$data){

return response()->json(['status' => false, 'data' => $data]) ;

}

$user = Auth::user() ?? '' ;

$role = $user ? ($user->roles[0]->id ?? '' ): '';

$html = '';

$file\_path = explode('.',$data->file\_path)[1] ?? '' ;

if($file\_path == 'pdf'){

$ImagePath = asset('formupload/'. $data->file\_path) ;

$html .= ' <iframe id="pdfViewer" src="' . $ImagePath . '" width="100%" height="100%"></iframe>' ;

}else{

$ImagePath = asset('formupload/'. $data->file\_path) ;

$html .= '<img src="' . $ImagePath . '" style="width: 100%; height: auto;">';

}

$html .= '<div class="container-fluid" id="new\_added\_content\_' . $data->id . '">';

if( $role == 1){

// Pane section with radio buttons

$html .= '<div class="pane m-3">';

// $html .= '<label class="label"><span>Nonek</span><input id="idk\_' . $data->id . '" class="input idk" name="radio" type="radio"' . ($data->status == '' ? 'checked="checked"' : '') . ' value=""></label>';

$html .= '<label class="label"><span>Accept</span><input id="yes\_' . $data->id . '" class="input yes" name="radio" type="radio"' . ($data->status == 'Verified' ? 'checked="checked"' : '') . ' value="Verified"></label>';

$html .= '<label class="label"><span>Reject</span><input id="no\_' . $data->id . '" class="input no" name="radio" type="radio" ' . ($data->status == 'Rejected' ? 'checked="checked"' : '') . ' value="Rejected"></label>';

$html .= '<span class="preview\_selection"></span>';

$html .= '</div>';

// Admin remark input section

$html .= '<div class="extra-input m-3" id="extra-input\_' . $data->id . '">';

$html .= '<input type="text" name="admin\_remark" id="extra-text\_' . $data->id . '" class="form-control extra\_input" placeholder="Reason" value="' . $data->admin\_remark . '">';

$html .= '</div>';

}

$html .= '</div>'; // Closing container-fluid

$data = $html ;

return response()->json(['status' => true, 'data' => $data]);

}

public function AdditioanlUploadPreview(Request $request){

$data = ApplicationAdditional::where(['application\_id' => $request->appId, 'additional\_id' => $request->documentId, 'id' => $request->imageId])->first();

if(!$data){

return response()->json(['status' => false, 'data' => $data]) ;

}

$user = Auth::user() ?? '' ;

$role = $user ? ($user->roles[0]->id ?? '' ): '';

$html = '';

$file\_path = explode('.',$data->file\_path)[1] ?? '' ;

if($file\_path == 'pdf'){

$ImagePath = asset('formupload/'. $data->file\_path) ;

$html .= ' <iframe id="pdfViewer" src="' . $ImagePath . '" width="100%" height="100%"></iframe>' ;

}else{

$ImagePath = asset('formupload/'. $data->file\_path) ;

$html .= '<img src="' . $ImagePath . '" style="width: 100%; height: auto;">';

}

$html .= '<div class="container-fluid" id="new\_added\_content\_' . $data->id . '">';

if( $role == 1){

// Pane section with radio buttons

$html .= '<div class="pane m-3">';

// $html .= '<label class="label"><span>None</span><input id="idk\_' . $data->id . '" class="input idk" name="radio" type="radio"' . ($data->status == '' ? 'checked="checked"' : '') . ' value=""></label>';

$html .= '<label class="label"><span>Accept</span><input id="yes\_' . $data->id . '" class="input yes" name="radio" type="radio"' . ($data->status == 'Verified' ? 'checked="checked"' : '') . ' value="Verified"></label>';

$html .= '<label class="label"><span>Reject</span><input id="no\_' . $data->id . '" class="input no" name="radio" type="radio" ' . ($data->status == 'Rejected' ? 'checked="checked"' : '') . ' value="Rejected"></label>';

$html .= '<span class="preview\_selection"></span>';

$html .= '</div>';

// Admin remark input section

$html .= '<div class="extra-input m-3" id="extra-input\_' . $data->id . '">';

$html .= '<input type="text" name="admin\_remark" id="extra-text\_' . $data->id . '" class="form-control extra\_input" placeholder="Reason" value="' . $data->admin\_remark . '">';

$html .= '</div>';

}

$html .= '</div>'; // Closing container-fluid

$data = $html ;

return response()->json(['status' => true, 'data' => $data]);

}

public function AddtioanlFormUploadPreview(Request $request){

$data = ApplicationAdditionFormDocument::where(['application\_id' => $request->appId, 'additional\_form\_id' => $request->documentId, 'id' => $request->imageId])->first();

if(!$data){

return response()->json(['status' => false, 'data' => $data]) ;

}

$user = Auth::user() ?? '' ;

$role = $user ? ($user->roles[0]->id ?? '' ): '';

$html = '';

$file\_path = explode('.',$data->file\_path)[1] ?? '' ;

if($file\_path == 'pdf'){

$ImagePath = asset('formupload/'. $data->file\_path) ;

$html .= ' <iframe id="pdfViewer" src="' . $ImagePath . '" width="100%" height="100%"></iframe>' ;

}else{

$ImagePath = asset('formupload/'. $data->file\_path) ;

$html .= '<img src="' . $ImagePath . '" style="width: 100%; height: auto;">';

}

$html .= '<div class="container-fluid" id="new\_added\_content\_' . $data->id . '">';

if( $role == 1){

// Pane section with radio buttons

$html .= '<div class="pane m-3">';

// $html .= '<label class="label"><span>None</span><input id="idk\_' . $data->id . '" class="input idk" name="radio" type="radio"' . ($data->status == '' ? 'checked="checked"' : '') . ' value=""></label>';

$html .= '<label class="label"><span>Accept</span><input id="yes\_' . $data->id . '" class="input yes" name="radio" type="radio"' . ($data->status == 'Verified' ? 'checked="checked"' : '') . ' value="Verified"></label>';

$html .= '<label class="label"><span>Reject</span><input id="no\_' . $data->id . '" class="input no" name="radio" type="radio" ' . ($data->status == 'Rejected' ? 'checked="checked"' : '') . ' value="Rejected"></label>';

$html .= '<span class="preview\_selection"></span>';

$html .= '</div>';

// Admin remark input section

$html .= '<div class="extra-input m-3" id="extra-input\_' . $data->id . '">';

$html .= '<input type="text" name="admin\_remark" id="extra-text\_' . $data->id . '" class="form-control extra\_input" placeholder="Reason" value="' . $data->admin\_remark . '">';

$html .= '</div>';

}

$html .= '</div>'; // Closing container-fluid

$data = $html ;

return response()->json(['status' => true, 'data' => $data]);

}

public function uploadDocumentDelete(Request $request)

{

// Validate the request

$request->validate([

'applicationId' => 'required|integer',

'documentId' => 'required|integer',

]);

$applicationId = $request->applicationId;

$documentId = $request->documentId;

// Find the application record

$application = Application::find($applicationId);

if (!$application) {

return response()->json(['status' => false, 'message' => 'Application not found']);

}

// Detach related records with the specific applicationId and documentId

$application->applicationLoanDocument2()->detach($documentId);

$application->applicantDocument1()->detach($documentId);

// Find all images associated with the given applicationId and documentId

$images = ApplicationImage::where('application\_id', $applicationId)

->where('document\_id', $documentId)

->get();

if(count($images)> 0){

foreach ($images as $image) {

// Determine the path to the image file

$imagePath = public\_path('formupload/' . $image->path); // Adjust the path based on your structure

// Delete the image file from the public directory

if (File::exists($imagePath)) {

File::delete($imagePath);

}

// Delete the image record from the database

$image->delete();

}

}

return response()->json(['status' => true, 'message' => 'All images and related records deleted successfully']);

}

public function uploadFormDelete(Request $request)

{

// Validate the request

$request->validate([

'applicationId' => 'required|integer',

'documentId' => 'required|integer',

]);

$applicationId = $request->applicationId;

$documentId = $request->documentId;

// Find the application record

$application = Application::find($applicationId);

if (!$application) {

return response()->json(['status' => false, 'message' => 'Application not found']);

}

// Detach related records with the specific applicationId and documentId

$application->applicationLoanFormUpload2()->detach($documentId);

$application->applicantformUpload2()->detach($documentId);

// Find all images associated with the given applicationId and documentId

$images = ApplicationFormImage::where('application\_id', $applicationId)

->where('form\_id', $documentId)

->get();

if(count($images)> 0){

foreach ($images as $image) {

// Determine the path to the image file

$imagePath = public\_path('formupload/' . $image->path); // Adjust the path based on your structure

// Delete the image file from the public directory

if (File::exists($imagePath)) {

File::delete($imagePath);

}

// Delete the image record from the database

$image->delete();

}

}

return response()->json(['status' => true, 'message' => 'All images and related records deleted successfully']);

}

public function uploadAdditionalDocumentDelete(Request $request){

// Validate the request

$request->validate([

'applicationId' => 'required|integer',

'documentId' => 'required|integer',

]);

$applicationId = $request->applicationId;

$documentId = $request->documentId;

// Find the application record

$application = Application::find($applicationId);

if (!$application) {

return response()->json(['status' => false, 'message' => 'Application not found']);

}

// Detach related records with the specific applicationId and documentId

$application->additionalDocument2()->detach($documentId);

// Find all images associated with the given applicationId and documentId

$images = ApplicationAdditional::where('application\_id', $applicationId)

->where('additional\_id', $documentId)

->get();

if(count($images)> 0){

foreach ($images as $image) {

// Determine the path to the image file

$imagePath = public\_path('formupload/' . $image->path); // Adjust the path based on your structure

// Delete the image file from the public directory

if (File::exists($imagePath)) {

File::delete($imagePath);

}

// Delete the image record from the database

$image->delete();

}

}

return response()->json(['status' => true, 'message' => 'All images and related records deleted successfully']);

}

public function uploadFormAdditionalDocumentDelete(Request $request){

// Validate the request

$request->validate([

'applicationId' => 'required|integer',

'documentId' => 'required|integer',

]);

$applicationId = $request->applicationId;

$documentId = $request->documentId;

// Find the application record

$application = Application::find($applicationId);

if (!$application) {

return response()->json(['status' => false, 'message' => 'Application not found']);

}

$images = ApplicationAdditionFormDocument::where(['additional\_form\_id'=>$documentId , application\_id => $applicationId ])->get();

if(count($images)> 0){

foreach ($images as $image) {

// Determine the path to the image file

$imagePath = public\_path('formupload/' . $image->file\_path); // Adjust the path based on your structure

// Delete the image file from the public directory

if (File::exists($imagePath)) {

File::delete($imagePath);

}

// Delete the image record from the database

$image->delete();

}

}

$formFileData = ApplicationAdditionFormUploads::find($documentId);

if($formFileData){

$imagePath = public\_path('formupload/' . $formFileData->file\_path); // Adjust the path based on your structure

// Delete the image file from the public directory

if (File::exists($imagePath)) {

File::delete($imagePath);

}

// Delete the image record from the database

$formFileData->delete();

}

return response()->json(['status' => true, 'message' => 'All images and related records deleted successfully']);

}

private function statusImage($status){

$image\_path = '';

if($status == 'Rejected'){

$image\_path = '<img title="Rejected" src="' . asset('formupload/file\_cancel.png') . '" class="document\_status\_img">';

}elseif($status == 'Reupload'){

$image\_path = '<img title="Reuploaded" src="' . asset('formupload/stage\_reupload.png') . '" class="document\_status\_img">';

}elseif($status == 'Accepted'){

$image\_path = '<img title="Accepted" src="' . asset('formupload/file\_success.png') . '" class="document\_status\_img">';

}elseif($status == 'Upload'){

$image\_path = '<img title="Uploaded" src="' . asset('formupload/file\_uploaded.png') . '" class="document\_status\_img">';

}

return $image\_path ;

}

public function documentCaseStatusChange(Request $request){

$id = $request -> id ?? '';

$status = $request -> status ?? '';

if( $id && $status){

$application = Application::find($id) ;

if(!$application){

return response()->json(['status' => false, 'message' => 'Case Not found']);

}

$application ->status = $status ;

$application ->status\_updated\_by = Auth::user()->id ?? null ;

$application -> save();

return response()->json(['status' => true, 'message' => 'Case Stage Changed Successful']);

}else{

return response()->json(['status' => false, 'message' => 'Given details Invalid']);

}

}

// uploadFormDelete

}