

Multi-Stage Approval Implementation Guide

Overview

This guide implements a multi-stage approval system where:

- Approvals must be sequential within a stage
- Only current approver can see and act on documents
- Template users can view approval status but cannot approve/reject
- Final approver triggers the external API

Step 1: Database Migration for Approval Tracking

Create a new migration to track approval progress:

```
php
```

```
// database/migrations/create_approval_progress_table.php
```

```
<?php
```

```
use Illuminate\Database\Migrations\Migration;
```

```
use Illuminate\Database\Schema\Blueprint;
```

```
use Illuminate\Support\Facades\Schema;
```

```
class CreateApprovalProgressTable extends Migration
```

```
{
```

```
    public function up()
```

```
    {
```

```
        Schema::create('approval_progress', function (Blueprint $table) {
```

```
            $table->id();
```

```
            $table->string('doc_num');
```

```
            $table->string('wdd_code');
```

```
            $table->string('draft_entry');
```

```
            $table->integer('stage_id');
```

```
            $table->integer('current_approver_user_id');
```

```
            $table->integer('current_approver_sequence')->default(1);
```

```
            $table->enum('status', ['pending', 'approved', 'rejected'])->default('pending');
```

```
            $table->text('remarks')->nullable();
```

```
            $table->integer('approved_by')->nullable();
```

```
            $table->timestamp('approved_at')->nullable();
```

```
            $table->json('approval_history')->nullable(); // Store approval chain
```

```
            $table->timestamps();
```

```
            $table->unique(['doc_num', 'wdd_code', 'draft_entry']);
```

```
            $table->foreign('stage_id')->references('id')->on('stages');
```

```
            $table->foreign('current_approver_user_id')->references('id')->on('user');
```

```
        });
```

```
    }
```

```
    public function down()
```

```
    {
```

```
        Schema::dropIfExists('approval_progress');
```

```
    }
```

```
}
```

Step 2: Create Approval Progress Model

```
php
```

```
// app/Models/ApprovalProgress.php
```

```
<?php
```

```
namespace App\Models;
```

```
use Illuminate\Database\Eloquent\Model;
```

```
class ApprovalProgress extends Model
```

```
{
```

```
    protected $table = 'approval_progress';
```

```
    protected $fillable = [
```

```
        'doc_num', 'wdd_code', 'draft_entry', 'stage_id',
```

```
        'current_approver_user_id', 'current_approver_sequence',
```

```
        'status', 'remarks', 'approved_by', 'approved_at', 'approval_history'
```

```
    ];
```

```
    protected $casts = [
```

```
        'approval_history' => 'array',
```

```
        'approved_at' => 'datetime'
```

```
    ];
```

```
    public function stage()
```

```
    {
```

```
        return $this->belongsTo(Stage::class, 'stage_id');
```

```
    }
```

```
    public function currentApprover()
```

```
    {
```

```
        return $this->belongsTo(User::class, 'current_approver_user_id');
```

```
    }
```

```
    public function approvedBy()
```

```
    {
```

```
        return $this->belongsTo(User::class, 'approved_by');
```

```
    }
```

```
}
```

Step 3: Create Required Models

```
php
```

```
// app/Models/Stage.php
```

```
<?php
```

```
namespace App\Models;
```

```
use Illuminate\Database\Eloquent\Model;
```

```
class Stage extends Model
```

```
{  
    protected $table = 'stages';  
  
    protected $fillable = [  
        'stage_name', 'stage_description', 'no_of_approval_required', 'no_of_rejection_required'  
    ];  
  
    public function authorizers()  
    {  
        return $this->hasMany(StageAuthorizer::class, 'stage_ID');  
    }  
}
```

```
// app/Models/StageAuthorizer.php
```

```
<?php
```

```
namespace App\Models;
```

```
use Illuminate\Database\Eloquent\Model;
```

```
class StageAuthorizer extends Model
```

```
{  
    protected $table = 'stages_authorizers';  
  
    protected $fillable = ['stage_ID', 'user_ID'];  
  
    public function stage()  
    {  
        return $this->belongsTo(Stage::class, 'stage_ID');  
    }  
  
    public function user()  
    {  
        return $this->belongsTo(User::class, 'user_ID');  
    }  
}
```

```
// app/Models/ApprovalTemplateUser.php
```

```
<?php
```

```
namespace App\Models;
```

```
use Illuminate\Database\Eloquent\Model;
```

```
class ApprovalTemplateUser extends Model
{
    protected $table = 'approval_template_users';

    protected $fillable = ['approval_template_ID', 'user_ID'];

    public function user()
    {
        return $this->belongsTo(User::class, 'user_ID');
    }
}
```

Step 4: Create Approval Service

```
php
```