Git Commit Description: Initial Dynamic Process Flow Module

This commit introduces the foundational architecture for a dynamic, model-agnostic process flow system integrated into a Laravel 8.75 application using Spatie Laravel Permissions.

# ✅ Key Features Implemented

- Polymorphic workflow system supporting multiple Eloquent models (Activity, PsipName).  
- Core tables and migrations for process flows, steps, actions, conditions, instances, and logs.  
- Eloquent models for all process-related entities, with complete relationships.  
- Service classes to manage:  
 • Step condition evaluation (ProcessConditionService)  
 • Step action execution (ProcessActionService)  
 • Step advancement and logging (ProcessEngineService)  
- Controller (`ProcessFlowController`) with `advance()` method to manually move flows to the next step.  
- Dynamic permission middleware integration via route names (Spatie).  
- Auto-completion logic ready for PsipName when all related Activity flows are complete.

# 🗃️ Directory Structure Updates

- Models: app/Models/ProcessFlow.php, ProcessStep.php, etc.  
- Services: app/Services/ProcessFlow/\*  
- Controller: app/Http/Controllers/ProcessFlowController.php

# 🔐 Routes

POST /process/advance/{instance} – protected with `permission` middleware; named `process.advance`

# 🔧 Developer Notes

- System is ready for dynamic UI implementation (admin/planning roles).  
- Next step includes CRUD UI for flows and step configuration.  
- Supports future expansion to SubActivity or other models without altering core logic.