



Last indexed: 27 August 2025 (c12f7a)

PadelFlow Overview

Core Application Architecture

Server Setup and Configuration

User Roles and Authentication

Tournament Management  
Features

Real-time Features

Database Layer

SQLite Database Management

Database Extensions

Qt Framework Components

Image Format Support

Development Environment

IDE Configuration

Debugging Setup

Project Configuration

## Project Configuration

### Relevant source files

This document covers the project-level configuration files that control version control behavior, development environment setup, and build processes for the PadelFlow platform. The primary focus is on the Git ignore patterns and repository configuration that ensure a clean development workflow and prevent sensitive or generated files from being committed to version control.

For IDE-specific configurations, see [IDE Configuration](#). For debugging-related configurations, see [Debugging Setup](#).

## Git Ignore Configuration

The `.gitignore` file provides comprehensive exclusion patterns for Node.js development, covering multiple categories of files and directories that should not be tracked in version control.

## File Structure and Organization

The `.gitignore` configuration is organized into logical sections that address different aspects of Node.js development:



Sources:  .gitignore | 1-140

### Core Exclusion Categories

The Git ignore patterns are organized into several key categories:

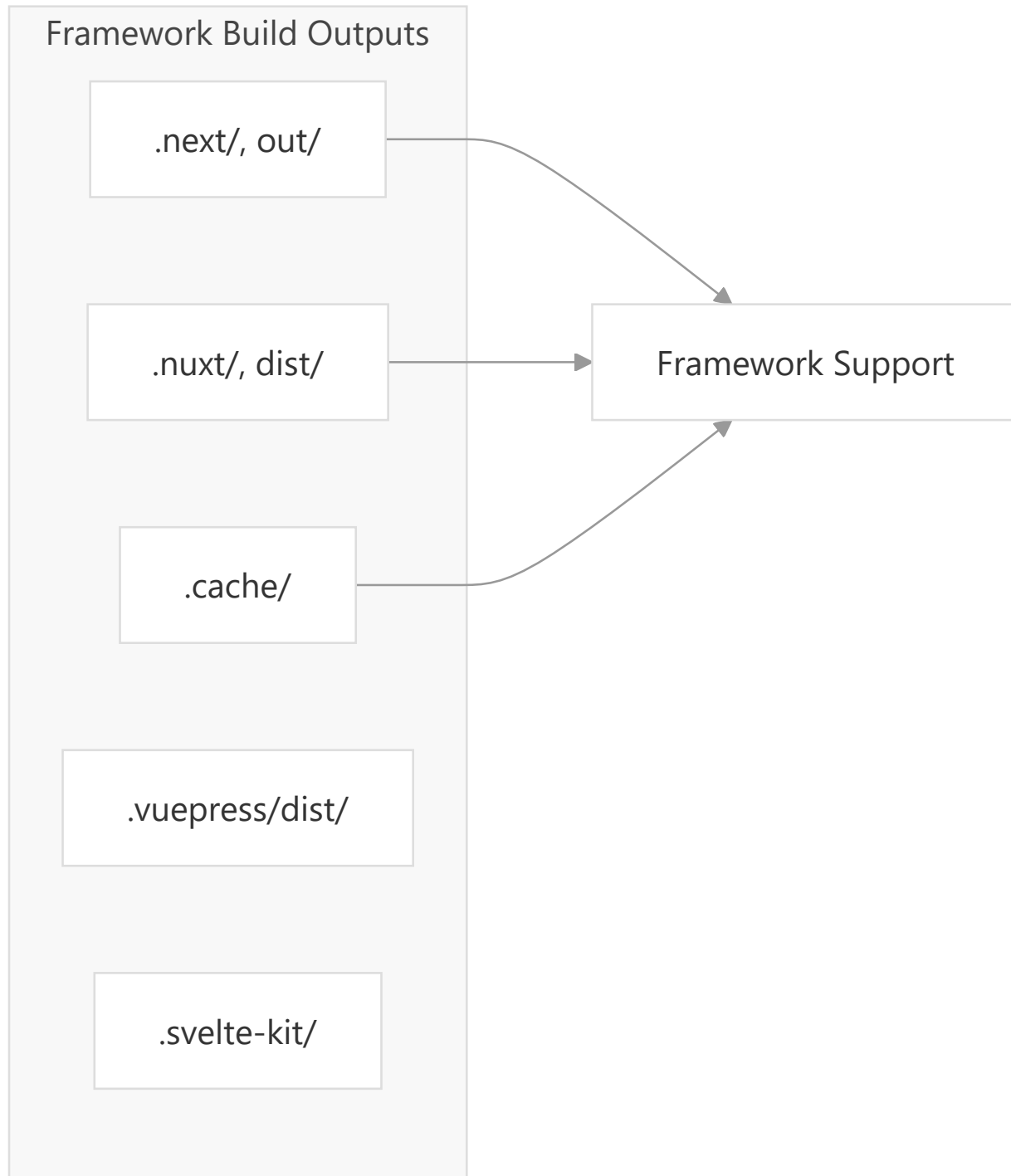
Category	Patterns	Purpose
Logs and Runtime	logs , *.log , pids , *.pid	Excludes runtime-generated log files and process identifiers
Dependencies	node_modules/ , jspm_packages/	Prevents dependency directories from being committed
Build Artifacts	build/Release , dist , .next , .nuxt	Excludes compiled and generated build outputs
Cache Directories	.cache , .npm , .parcel-cache	Prevents caching directories from being tracked
Environment Files	.env , .env.* (except .env.example )	Protects sensitive environment variables
Development Tools	.vscode-test , .eslintcache	Excludes IDE and tool-specific temporary files

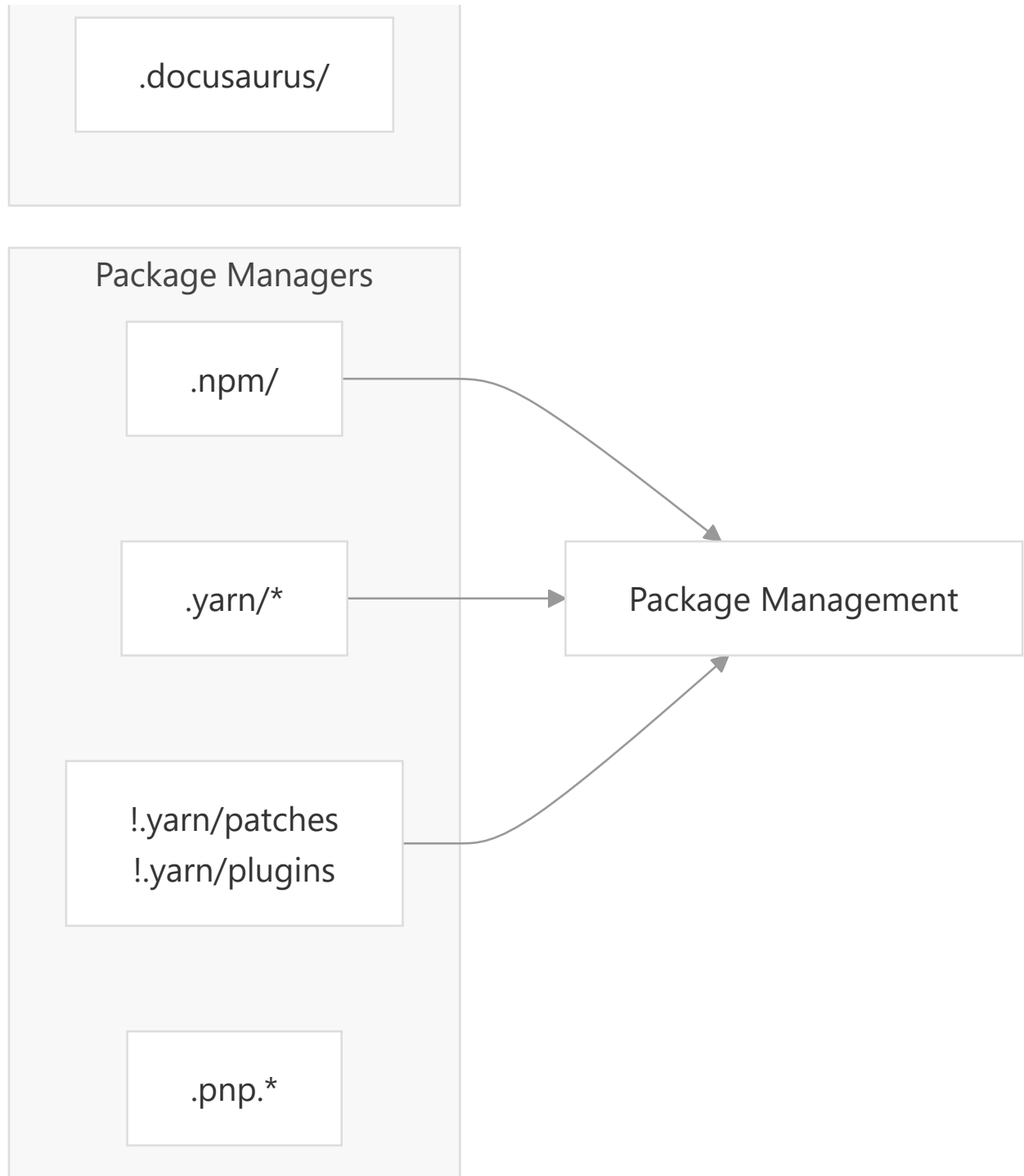
Sources:  .gitignore | 1-10  .gitignore | 40-42  .gitignore | 68-71  
 .gitignore | 125-126

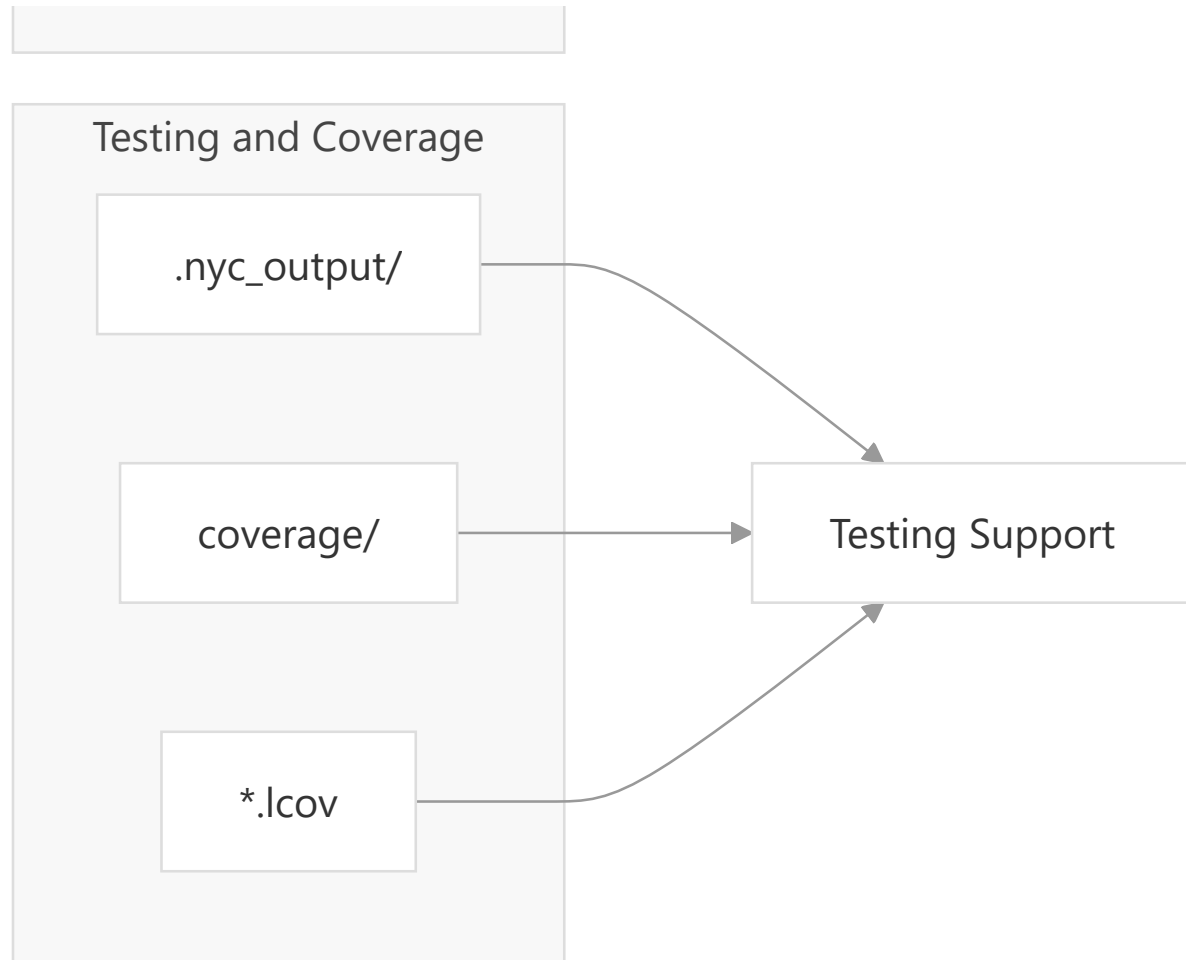
Framework-Specific Exclusions




The configuration includes patterns for multiple JavaScript frameworks and tools:











Sources:  .gitignore | 77-84  .gitignore | 120-133  .gitignore | 23-29

## Environment and Security Configuration

The `.gitignore` file includes specific patterns for environment variable management:

- **Environment Files:** `.env` and `.env.*` patterns exclude environment variable files

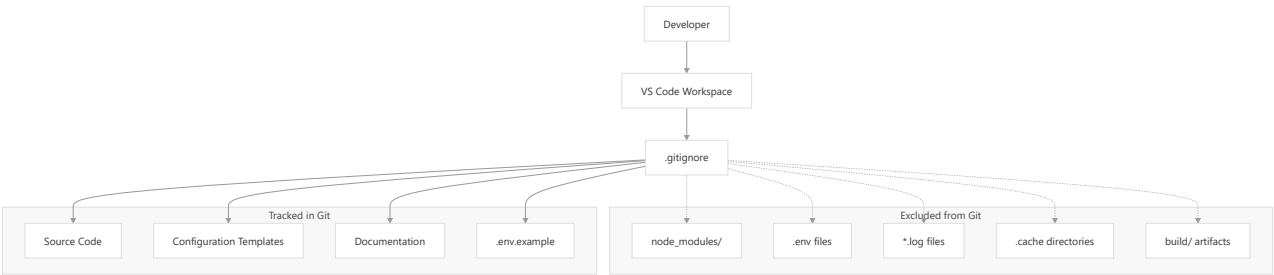
 .gitignore | 69-70



- **Environment Example Preservation:** `!.env.example` ensures example environment files are tracked  `.gitignore` | 71
- **Development Tool Exclusions:** `.vscode-test` prevents VSCode extension testing artifacts from being committed  `.gitignore` | 126

This configuration ensures sensitive configuration data remains local while maintaining example templates for other developers.

## Development Workflow Integration

The project configuration supports the PadelFlow development workflow by maintaining clean repository state:



Sources:  `.gitignore` | 41  `.gitignore` | 69-71  `.gitignore` | 2-7

## Configuration Management Best Practices

The `.gitignore` configuration implements several best practices:

1. **Comprehensive Coverage:** Includes patterns for multiple Node.js ecosystems and frameworks



2. **Security-First:** Excludes environment files while preserving examples
3. **Performance Optimization:** Prevents large dependency directories from being tracked
4. **Tool Integration:** Supports various development tools and their cache directories
5. **Selective Inclusion:** Uses negation patterns (e.g., `!.yarn/patches` ) to include specific necessary files within broader exclusions

The configuration supports both local development and CI/CD environments by ensuring only essential source code and configuration templates are version controlled.

**Sources:**  .gitignore | 131-135  .gitignore | 69-71  .gitignore | 40-42