

PrismaService

September 4, 2026

PrismaService

Kind: Service**Source:** [atloria-monorepo/libs/database/src/lib/prisma.service.ts](https://github.com/atloria-monorepo/libs/database/src/lib/prisma.service.ts)

`PrismaService` is a NestJS service that manages the application's Prisma database client lifecycle. It connects to the database when the module initializes, disconnects during shutdown, and can enable query logging for debugging and observability.

Methods

Method	Signature	Returns	Description
<code>onModuleInit</code>	<code>onModuleInit()</code>	unknown	
<code>onModuleDestroy</code>	<code>onModuleDestroy()</code>	unknown	
<code>enableQueryLogging</code>	<code>enableQueryLogging()</code>	unknown	Enable query logging for development

Dependencies

- `ConfigService`

When something fails

- `PrismaService` handles failure in 1 place: it lets it reach the caller in all 1.

Diagram

```
sequenceDiagram
    participant Nest as NestJS Application
    participant Service as PrismaService
    participant Prisma as Prisma Client
    participant DB as Database
```

```
Nest->>Service: onModuleInit()
Service->>Prisma: $connect()
Prisma->>DB: Open database connection
DB-->>Prisma: Connection established

Note over Service,Prisma: Optional development/debug setup
Nest->>Service: enableQueryLogging()
Service->>Prisma: Subscribe to query events

Nest->>Service: onModuleDestroy()
Service->>Prisma: $disconnect()
Prisma->>DB: Close database connection
```

Usage

```
import { Injectable } from '@nestjs/common';
import { PrismaService } from '@atloria/database';

@Injectable()
export class UserService {
  constructor(private readonly prisma: PrismaService) {}

  async findByEmail(email: string) {
    return this.prisma.user.findUnique({
      where: { email },
    });
  }

  async createUser(email: string, name: string) {
    return this.prisma.user.create({
      data: {
        email,
        name,
      },
    });
  }
}
```

AI Coding Instructions

- Inject `PrismaService` through NestJS dependency injection instead of creating `PrismaClient` instances directly.
- Use the inherited Prisma client delegates, such as `prisma.user`, `prisma.post`, and `prisma.$transaction`, for database access.

- Do not manually call `$connect()` or `$disconnect()` from feature services; lifecycle hooks handle connection management.
- Enable query logging only when needed for debugging, as verbose query output may expose sensitive data or affect log volume.
- Ensure modules that use this service import the database module that provides and exports `PrismaService` .

Relationships

- `DEPENDS_ON` → `configservice`

Referenced By

- `ActivityService` (`DEPENDS_ON`)
- `AIUsageService` (`DEPENDS_ON`)
- `AIService` (`DEPENDS_ON`)
- `UIAnalyzerService` (`DEPENDS_ON`)
- `AnalyticsService` (`DEPENDS_ON`)
- `AnalyticsExportService` (`DEPENDS_ON`)
- `ScheduledReportsService` (`DEPENDS_ON`)
- `FeedbackService` (`DEPENDS_ON`)
- `AppService` (`DEPENDS_ON`)
- `AssetsService` (`DEPENDS_ON`)
- `AudienceService` (`DEPENDS_ON`)
- `AuditService` (`DEPENDS_ON`)
- `AuthService` (`DEPENDS_ON`)
- `PlatformAdminGuard` (`DEPENDS_ON`)
- `ProjectOrgGuard` (`DEPENDS_ON`)
- `ResourceOrgGuard` (`DEPENDS_ON`)
- `SamlController` (`DEPENDS_ON`)
- `SamlService` (`DEPENDS_ON`)
- `BillingService` (`DEPENDS_ON`)
- `PlanService` (`DEPENDS_ON`)
- `BitbucketService` (`DEPENDS_ON`)
- `BoardsService` (`DEPENDS_ON`)

- BrandingService (DEPENDS_ON)
- CaptureService (DEPENDS_ON)
- FlowDraftStitcherService (DEPENDS_ON)
- ChangeRequestService (DEPENDS_ON)
- DocsConfigService (DEPENDS_ON)
- PreviewService (DEPENDS_ON)
- ChangelogDraftsService (DEPENDS_ON)
- ChangelogPublicService (DEPENDS_ON)
- ChangelogSubscriptionsService (DEPENDS_ON)
- CollaborationService (DEPENDS_ON)
- CommentService (DEPENDS_ON)
- ContentTasksService (DEPENDS_ON)
- DashboardService (DEPENDS_ON)
- VersionCarryForwardService (DEPENDS_ON)
- ChangelogService (DEPENDS_ON)
- DocVersionService (DEPENDS_ON)
- DocVersionExportService (DEPENDS_ON)
- DocVersionRetentionService (DEPENDS_ON)
- ScheduledPublishService (DEPENDS_ON)
- VersionAccessGuard (DEPENDS_ON)
- VersionDiffService (DEPENDS_ON)
- DocsPrController (DEPENDS_ON)
- DocsPrService (DEPENDS_ON)
- PageRegenService (DEPENDS_ON)
- OutboxService (DEPENDS_ON)
- ReconcilerService (DEPENDS_ON)
- CategoryService (DEPENDS_ON)
- DocumentService (DEPENDS_ON)
- DocumentRevisionService (DEPENDS_ON)
- DocumentVersionService (DEPENDS_ON)
- DraftDocumentService (DEPENDS_ON)
- UrlService (DEPENDS_ON)
- ManualVideoService (DEPENDS_ON)

- AzureSearchService (DEPENDS_ON)
- DocAutomationSchedulerService (DEPENDS_ON)
- DocAutomationService (DEPENDS_ON)
- DocumentIndexingService (DEPENDS_ON)
- JobReconcileService (DEPENDS_ON)
- Neo4jDocsOrchestratorService (DEPENDS_ON)
- AgentAnalyticsService (DEPENDS_ON)
- EventsController (DEPENDS_ON)
- EventsService (DEPENDS_ON)
- GitSyncIngestService (DEPENDS_ON)
- GitSyncService (DEPENDS_ON)
- GitSyncMirrorService (DEPENDS_ON)
- GitHubService (DEPENDS_ON)
- GithubAppService (DEPENDS_ON)
- GitLabService (DEPENDS_ON)
- EntityExtractorService (DEPENDS_ON)
- KgSyncService (DEPENDS_ON)
- InvitationService (DEPENDS_ON)
- DeskService (DEPENDS_ON)
- IssuesService (DEPENDS_ON)
- TranslationsService (DEPENDS_ON)
- ManualsInsightsService (DEPENDS_ON)
- NotificationService (DEPENDS_ON)
- OpsAlertsService (DEPENDS_ON)
- OpsService (DEPENDS_ON)
- OrganizationService (DEPENDS_ON)
- ParserService (DEPENDS_ON)
- SymbolIndexService (DEPENDS_ON)
- ParsingService (DEPENDS_ON)
- PlatformAdminBootstrapService (DEPENDS_ON)
- PlatformService (DEPENDS_ON)
- PmeService (DEPENDS_ON)
- ContentVisibilityService (DEPENDS_ON)

- ProjectDomainsService (DEPENDS_ON)
- ProjectService (DEPENDS_ON)
- PublicProjectController (DEPENDS_ON)
- SyncJobService (DEPENDS_ON)
- SyncController (DEPENDS_ON)
- SyncService (DEPENDS_ON)
- DocsAccessService (DEPENDS_ON)
- DocsVisibilityService (DEPENDS_ON)
- ProjectAccessService (DEPENDS_ON)
- ReaderClaimsExchangeService (DEPENDS_ON)
- ReaderTokenService (DEPENDS_ON)
- DevhubService (DEPENDS_ON)
- ReaderAccountService (DEPENDS_ON)
- ReaderKeyService (DEPENDS_ON)
- ReaderLogRetentionService (DEPENDS_ON)
- ReaderMagicLinkService (DEPENDS_ON)
- ScimAdminService (DEPENDS_ON)
- ScimAuthGuard (DEPENDS_ON)
- ScimService (DEPENDS_ON)
- ScreenshotService (DEPENDS_ON)
- PublicSdkController (DEPENDS_ON)
- SdkArtifactsService (DEPENDS_ON)
- SnippetsService (DEPENDS_ON)
- SuggestionService (DEPENDS_ON)
- AgentAdapterService (DEPENDS_ON)
- SupportAgentService (DEPENDS_ON)
- DocsTrueupService (DEPENDS_ON)
- FreshnessService (DEPENDS_ON)
- TechDocsChatService (DEPENDS_ON)
- TechDocsRagService (DEPENDS_ON)
- StaleAlertService (DEPENDS_ON)
- TechDocsConnectorsService (DEPENDS_ON)
- TechDocsExportService (DEPENDS_ON)

- TechDocsNotionExportService (DEPENDS_ON)
- TechDocsSourcesService (DEPENDS_ON)
- TechnicalDocsChatController (DEPENDS_ON)
- TechnicalDocsEnrichmentService (DEPENDS_ON)
- TechnicalDocsGenerationService (DEPENDS_ON)
- TechnicalDocsMaterializerService (DEPENDS_ON)
- TechnicalDocsMcpController (DEPENDS_ON)
- TechnicalDocsParserService (DEPENDS_ON)
- TechnicalDocsQueue (DEPENDS_ON)
- TechnicalDocsController (DEPENDS_ON)
- TechnicalDocsService (DEPENDS_ON)
- TemplateService (DEPENDS_ON)
- ToursHealingService (DEPENDS_ON)
- ToursService (DEPENDS_ON)
- TrialService (DEPENDS_ON)
- WebhookService (DEPENDS_ON)
- AnalysisService (DEPENDS_ON)
- UsersService (DEPENDS_ON)
- ScreenshotMetadataService (DEPENDS_ON)
- BatchScreenshotService (DEPENDS_ON)
- FlowReplayService (DEPENDS_ON)
- DatabaseModule (MODULE_PROVIDES)
- DatabaseModule (MODULE_EXPORTS)