

Project

August 19, 2026

Project

Kind: Class

Source: Pams/Logic/Pams.Business/BindingModels/Projects/Project.cs (line 6)

Part of: [Pams](#)

`Project` is a C# binding model that represents project details exchanged between the business layer and other application layers. It stores project identity, ownership, descriptive information, creation metadata, and the final product value.

Diagram

```
graph LR
    Project[Project] --> Identity[ProjectID<br/>ProjectCode]
    Project --> Details[ProjectName<br/>ProjectLocation<br/>ProjectDescription]
    Project --> Ownership[OwnerID<br/>OwnerName]
    Project --> Audit[CreatedById<br/>CreatedAt]
    Project --> Output[FinalProduct]
```

Usage

```
const projectPayload = {
  ProjectCode: "PAMS-ALPHA",
  ProjectName: "Alpha Project",
  ProjectLocation: "Main Office",
  ProjectDescription: "Tracks project planning and delivery.",
  OwnerID: null,
  OwnerName: "Project Owner",
  CreatedById: null,
  FinalProduct: "Completed project deliverable",
  CreatedAt: new Date().toISOString(),
};

const response = await fetch("/api/projects", {
  method: "POST",
```

```

headers: {
  "Content-Type": "application/json",
},
body: JSON.stringify(projectPayload),
});

if (!response.ok) {
  throw new Error("Unable to create project.");
}

const project = await response.json();
console.log(project.ProjectCode);

```

AI Coding Instructions

- Keep property names aligned with the C# model when mapping request and response payloads.
- Treat `OwnerID` , `CreatedById` , and `CreatedAt` as nullable values and handle `null` explicitly.
- Preserve project identifiers without assuming JavaScript `number` values can safely represent C# `long` values.
- Validate required project details, such as code, name, and location, at the API boundary before creating or updating a `Project` .

Relationships

- IMPORTS → `Project`
- IMPORTS → `ProjectRegistered`
- IMPORTS → `ProjectComment`
- IMPORTS → `CommentsView`
- IMPORTS → `Comments`
- IMPORTS → `CommentBindingModel`
- IMPORTS → `ContactPerson`
- IMPORTS → `ContactPersonBindingModel`
- IMPORTS → `Status`
- IMPORTS → `ProjectStatus`
- IMPORTS → `ShareholdersBindingModel`
- IMPORTS → `ProjectRegisteredBindingModel`
- IMPORTS → `UserLogResultBindingModel`

Used by

14 references from 14 files. Each is a place in this repository where the symbol is actually used — go read one rather than trusting an example.

Imported by (14)

- `FilesController` — `Pams/API/Pams.API/Controllers/Common/FilesController.cs` :27
- `InquiryController` — `Pams/API/Pams.API/Controllers/Sales/InquiryController.cs` :48
- `ProjectController` — `Pams/API/Pams.API/Controllers/Sales/ProjectController.cs` :28
- `ProjectResponsiblesController` —
`Pams/API/Pams.API/Controllers/Sales/ProjectResponsibleController.cs` :21
- `Inquiry` — `Pams/Logic/Pams.Business/BindingModels/Sales/InquiryBindingModel.cs` :12
- `Project` — `Pams/Logic/Pams.Business/BindingModels/Projects/Project.cs` :6
- `HasPermissionModel` —
`Pams/Logic/Pams.Business/Component/Security/RolePermissionLogic.Logic.cs` :11
- `EndUser` — `Pams/Logic/Pams.Business/Domain/Import/ImportDomain.cs` :3891

...and 6 more.