

PromanX REST API

Interface Overview (Summary of the Swagger/OpenAPI documentation)

Version 2026.1.10.22 · OpenAPI 3.0 · As of: 30 July 2026

1. Introduction

The PromanX REST API provides data access to the PromanX database. It allows external systems to query project, company, employee, and reporting data in a structured way, and to maintain master data.

Provider: PROMAN Software GmbH · Contact: support@proman.at · www.proman.at

The API follows the OpenAPI 3.0 specification and is reachable via an environment-specific base URL (e.g. a demo/test system or a customer's production system). All endpoints are located under the /api path prefix.

2. Authentication

Access is handled via a login endpoint that exchanges an email address and password for a bearer token. This token is then included in the 'Authorization: Bearer <token>' header on all subsequent calls.

Method	Path	Description
POST	/api/Login	Sign in with email and password, returns an access token

3. Functional Areas

3.1 Central Master Data (company-wide)

Companies, employees, and functions that are not tied to a single project. Linking/unlinking to projects is available through dedicated endpoints, both via internal IDs and via customer-side external IDs (ExternalId).

Method	Path	Description
GET	/api/Central/Companies	List of all central companies
POST	/api/Central/Companies/AddOrUpdateCompany	Create/update a company
GET	/api/Central/Companies/LinkCompanyToProject/{id}	Link a company to a project
GET	/api/Central/Companies/UnlinkCompanyFromProject/{id}	Remove a company's link to a project
DELETE	/api/Central/Companies/DeleteCompany/{id}	Delete a company
GET	/api/Central/Employees	List of all central employees
POST	/api/Central/Employees/AddOrUpdateEmployee	Create/update an employee
GET	/api/Central/Employees/LinkEmployeeToProject/{id}	Link an employee to a project
DELETE	/api/Central/Employees/DeleteEmployee/{id}	Delete an employee
GET	/api/Central/Functions	List of all central functions/roles
POST	/api/Central/Functions/AddFunction	Create a function
DELETE	/api/Central/Functions/DeleteFunction	Delete a function

3.2 Project-Level Master Data

Analogous management of companies, employees, and functions, but scoped directly to a project (master data per project rather than company-wide).

Method	Path	Description
GET	/api/Projects/{aProjectId}/Companies	Companies in a project
POST	/api/Projects/{aProjectId}/Companies/AddOrUpdateCompany	Create/update a company within a project
DELETE	/api/Projects/{aProjectId}/Companies/DeleteCompany/{id}	Delete a company from a project
GET	/api/Projects/{aProjectId}/Employees	Employees in a project

Method	Path	Description
POST	/api/Projects/{aProjectId}/Employees/AddOrUpdateEmployee	Create/update an employee within a project
DELETE	/api/Projects/{aProjectId}/Employees/DeleteEmployee/{id}	Delete an employee from a project
GET	/api/Projects/{aProjectId}/Functions	Functions in a project
POST	/api/Projects/{aProjectId}/Functions/AddFunction	Create a function within a project
DELETE	/api/Projects/{aProjectId}/Functions/DeleteFunction	Delete a function within a project

3.3 Projects

Access to project master data as well as dedicated change-tracking endpoints, which let you efficiently query only the records changed since a given date (instead of the full dataset).

Method	Path	Description
GET	/api/Projects	List of all projects
GET	/api/Projects/{aProjectId}	Project by ID, including attributes
POST	/api/Projects/Add	Create a new project
POST	/api/Projects/Update/{aProjectId}	Update a project
GET	/api/Projects/changedSince/{date}	Projects changed since a date
GET	/api/Projects/changedProjectItemsSince/{date}	Project items changed since a date
GET	/api/Projects/changedCollectionsSince/{date}	Report collections changed since a date
GET	/api/Projects/changedReportsSince/{date}	Reports changed since a date

3.4 Reporting (Report Collections & Reports)

Hierarchical access to meeting/site reports: Project → Report Collection → Report → Report Item. Also includes a history function per report item and the ability to retrieve a report as email content.

Method	Path	Description
GET	/api/Projects/{aProjectId}/ReportCollections	Report collections of a project
GET	/api/Projects/{...}/ReportCollections/{id}	Report collection by ID
GET	/api/Projects/{...}/ReportCollections/changedItemsSince/{datetime}	Items changed since a given point in time
GET	/api/Projects/{...}/Reports	Reports in a collection
GET	/api/Projects/{...}/Reports/{aReportId}	Report by ID (including report items)
GET	/api/Projects/{...}/Reports/{id}/history/{itemId}	History of a report item
GET	/api/Projects/{...}/Reports/{id}/getReportMail	Retrieve a report as email content

3.5 Documents

Method	Path	Description
GET	/api/Documents/{aProjectId}	List of all documents in a project
GET	/api/Download/Document/{aDocumentId}	Download a document

3.6 System / Other

Method	Path	Description
GET	/api/Information	API/system information
GET	/api/Mandate	Mandate (tenant) information
POST	/api/Mandate/SetMandate	Set the active mandate (tenant)

4. Data Model (Excerpt)

The API defines roughly 50 structured data types (OpenAPI schemas). The main object groups:

- Companies & employees: Company, Employee, CompanyProjectData, EmployeeMailAddress, EmployeeTelephone, Gender, PhoneNumber, EmailAddress, and others
- Projects: ProjectCoreData, AddProjectRequest, UpdateProjectRequest
- Reporting: ApiResultReport, ApiResultReportItem, ApiResultReportCollection(s), ApiResultReportMail, ApiResultReportsAndItems
- Documents: StoredDocument, StoredDocumentType, Folder, PrintDocumentType
- Other: Mandate, FunctionType, ParticipantMailingItem, UserLoginRequest

Responses consistently follow a uniform wrapper pattern with the fields Success, ErrorMessage, and the actual typed data field (e.g. ApiResultProject, ApiResultReports). Records also carry metadata such as LastModifiedDate, DeletionDate, and — where applicable — an ExternalId for linking to customer-side systems.

5. Usage Notes

- Change tracking: The changedSince/changedItemsSince endpoints let you limit synchronization to records that actually changed, instead of transferring the full dataset each time.
- External IDs: Many objects support an ExternalId alongside the internal ID, to link records with a customer-side system of record.
- Deletions: Delete endpoints are available both via the internal ID and via the ExternalId.
- Environments: A demo/test system is available for integration development; production credentials and IP allowlisting are provided as part of the engagement.

This overview is based on OpenAPI specification version 2026.1.10.22 and is intended as an initial orientation. Full technical detail documentation (request/response schemas, required fields, error codes) is made available to customers after contract signing via the interactive Swagger documentation.