

PromanX REST-API

Übersicht der Schnittstelle (Zusammenfassung der Swagger/OpenAPI-Dokumentation)

Version 2026.1.10.22 · OpenAPI 3.0 · Stand: 30.07.2026

1. Einleitung

Die PromanX REST-API stellt einen Datenzugriff auf die PromanX-Datenbank bereit. Sie ermöglicht externen Systemen, Projekt-, Firmen-, Mitarbeiter- und Berichtsdaten strukturiert abzufragen sowie Stammdaten zu pflegen.

Anbieter: PROMAN Software GmbH · Kontakt: support@proman.at · www.proman.at

Die API folgt der OpenAPI-3.0-Spezifikation und ist über eine Basis-URL je Umgebung (z. B. Demo-/Testsystem oder produktives Kundensystem) erreichbar. Alle Endpunkte liegen unter dem Pfad-Präfix /api.

2. Authentifizierung

Der Zugriff erfolgt über einen Login-Endpunkt, der im Austausch gegen Benutzername/E-Mail und Passwort ein Bearer-Token ausstellt. Dieses Token wird anschließend bei allen weiteren Aufrufen im HTTP-Header 'Authorization: Bearer <token>' mitgeschickt.

Methode	Pfad	Beschreibung
POST	/api/Login	Anmeldung mit E-Mail und Passwort, liefert Zugriffstoken

3. Funktionsbereiche

3.1 Zentrale Stammdaten (unternehmensweit)

Firmen, Mitarbeiter und Funktionen, die nicht an ein einzelnes Projekt gebunden sind. Verknüpfung/Entkopplung zu Projekten ist jeweils über eigene Endpunkte möglich, sowohl über interne IDs als auch über kundenseitige externe IDs (ExternalId).

Methode	Pfad	Beschreibung
GET	/api/Central/Companies	Liste aller zentralen Firmen
POST	/api/Central/Companies/AddOrUpdateCompany	Firma anlegen/aktualisieren
GET	/api/Central/Companies/LinkCompanyToProject/{id}	Firma einem Projekt zuordnen
GET	/api/Central/Companies/UnlinkCompanyFromProject/{id}	Firmenzuordnung zu Projekt aufheben
DELETE	/api/Central/Companies/DeleteCompany/{id}	Firma löschen
GET	/api/Central/Employees	Liste aller zentralen Mitarbeiter
POST	/api/Central/Employees/AddOrUpdateEmployee	Mitarbeiter anlegen/aktualisieren
GET	/api/Central/Employees/LinkEmployeeToProject/{id}	Mitarbeiter einem Projekt zuordnen
DELETE	/api/Central/Employees/DeleteEmployee/{id}	Mitarbeiter löschen
GET	/api/Central/Functions	Liste aller zentralen Funktionen/Rollen
POST	/api/Central/Functions/AddFunction	Funktion anlegen
DELETE	/api/Central/Functions/DeleteFunction	Funktion löschen

3.2 Projektbezogene Stammdaten

Analoge Verwaltung von Firmen, Mitarbeitern und Funktionen, jedoch mit direktem Projektbezug (Stammdaten je Projekt statt unternehmensweit).

Methode	Pfad	Beschreibung
GET	/api/Projects/{aProjectId}/Companies	Firmen eines Projekts
POST	/api/Projects/{aProjectId}/Companies/AddOrUpdateCompany	Firma im Projekt anlegen/aktualisieren
DELETE	/api/Projects/{aProjectId}/Companies/DeleteCompany/{id}	Firma aus Projekt löschen
GET	/api/Projects/{aProjectId}/Employees	Mitarbeiter eines Projekts
POST	/api/Projects/{aProjectId}/Employees/AddOrUpdateEmployee	Mitarbeiter im Projekt anlegen/aktualisieren
DELETE	/api/Projects/{aProjectId}/Employees/DeleteEmployee/{id}	Mitarbeiter aus Projekt löschen
GET	/api/Projects/{aProjectId}/Functions	Funktionen eines Projekts
POST	/api/Projects/{aProjectId}/Functions/AddFunction	Funktion im Projekt anlegen
DELETE	/api/Projects/{aProjectId}/Functions/DeleteFunction	Funktion im Projekt löschen

3.3 Projekte

Zugriff auf Projektstammdaten sowie spezielle Endpunkte zur Änderungsverfolgung, mit denen effizient nur die seit einem Stichtag geänderten Datensätze abgefragt werden können (statt aller Daten).

Methode	Pfad	Beschreibung
GET	/api/Projects	Liste aller Projekte
GET	/api/Projects/{aProjectId}	Projekt inkl. Attribute nach ID
POST	/api/Projects/Add	Neues Projekt anlegen
POST	/api/Projects/Update/{aProjectId}	Projekt aktualisieren
GET	/api/Projects/changedSince/{date}	Projekte mit Änderungen seit Datum
GET	/api/Projects/changedProjectItemsSince/{date}	Projekt-Positionen mit Änderungen seit Datum
GET	/api/Projects/changedCollectionsSince/{date}	Berichtssammlungen mit Änderungen seit Datum
GET	/api/Projects/changedReportsSince/{date}	Berichte mit Änderungen seit Datum

3.4 Berichtswesen (Report Collections & Reports)

Hierarchischer Zugriff auf Besprechungs-/Baustellenberichte: Projekt → Berichtssammlung → Bericht → Berichtspunkt. Zusätzlich History-Funktion je Berichtspunkt sowie Versand eines Berichts per E-Mail.

Methode	Pfad	Beschreibung
GET	/api/Projects/{aProjectId}/ReportCollections	Berichtssammlungen eines Projekts
GET	/api/Projects/{...}/ReportCollections/{id}	Berichtssammlung nach ID
GET	/api/Projects/{...}/ReportCollections/changedItemsSince/{datetime}	Geänderte Positionen seit Zeitpunkt
GET	/api/Projects/{...}/Reports	Berichte einer Sammlung
GET	/api/Projects/{...}/Reports/{aReportId}	Bericht nach ID (inkl. Berichtspunkte)
GET	/api/Projects/{...}/Reports/{id}/history/{itemId}	Historie eines Berichtspunkts
GET	/api/Projects/{...}/Reports/{id}/getReportMail	Bericht als E-Mail-Inhalt abrufen

3.5 Dokumente

Methode	Pfad	Beschreibung
GET	/api/Documents/{aProjectId}	Liste aller Dokumente eines Projekts
GET	/api/Download/Document/{aDocumentId}	Dokument herunterladen

3.6 System / Sonstiges

Methode	Pfad	Beschreibung
---------	------	--------------

Methode	Pfad	Beschreibung
GET	/api/Information	API-/Systeminformationen
GET	/api/Mandate	Mandanteninformationen
POST	/api/Mandate/SetMandate	Aktiven Mandanten setzen

4. Datenmodell (Auszug)

Die API definiert rund 50 strukturierte Datentypen (OpenAPI-Schemas). Die wichtigsten Objektgruppen:

- Firmen & Mitarbeiter: Company, Employee, CompanyProjectData, EmployeeMailAddress, EmployeeTelephone, Gender, PhoneNumber, EmailAddress u. a.
- Projekte: ProjectCoreData, AddProjectRequest, UpdateProjectRequest
- Berichtswesen: ApiResultReport, ApiResultReportItem, ApiResultReportCollection(s), ApiResultReportMail, ApiResultReportsAndItems
- Dokumente: StoredDocument, StoredDocumentType, Folder, PrintDocumentType
- Sonstige: Mandate, FunctionType, ParticipantMailingItem, UserLoginRequest

Antworten folgen durchgängig einem einheitlichen Wrapper-Muster mit den Feldern Success, ErrorMessage sowie dem eigentlichen, typisierten Datenfeld (z. B. ApiResultProject, ApiResultReports). Datensätze führen zusätzlich Metadaten wie LastModifiedDate, DeletionDate und – wo zutreffend – ExternalId zur Verknüpfung mit kundenseitigen Systemen.

5. Hinweise zur Nutzung

- Change-Tracking: Über die changedSince-/changedItemsSince-Endpunkte lassen sich Synchronisationen effizient auf tatsächlich geänderte Datensätze beschränken, statt jeweils den vollen Datenbestand zu übertragen.
- Externe IDs: Viele Objekte unterstützen parallel zur internen ID eine ExternalId, um Datensätze mit einem kundenseitigen führenden System zu verknüpfen.
- Löschungen: Lösch-Endpunkte sind sowohl über die interne ID als auch über die ExternalId verfügbar.
- Umgebungen: Es existiert ein Demo-/Testsystem für die Anbindungsentwicklung; produktive Zugangsdaten sowie IP-Freischaltung werden im Rahmen der Beauftragung bereitgestellt.

Diese Übersicht basiert auf der OpenAPI-Spezifikation Version 2026.1.10.22 und dient als erste Orientierung. Für die vollständige, technische Detaildokumentation (Request-/Response-Schemas, Pflichtfelder, Fehlercodes) erhalten Kunden nach Vertragsabschluss Zugriff auf die interaktive Swagger-Dokumentation.