

Sub7 ForkU GO Platform API Reference

Overview

This document provides a reference for the GO Platform API endpoints available for the Sub7 ForkU Mobile MVP. The API follows a consistent pattern for CRUD operations across all entities.

Base URL

`https://godev.collectiveintelligence.com.au/forkuapp-de6b98b5-4402-4a8f-891b-70b3591df162`

Authentication

Authentication is handled through the GOSecurityProvider API. The endpoints support token-based authentication:

Method	Endpoint	Description
POST	<code>/api/gosecurityprovider/authenticate</code>	Authenticate user and get token
POST	<code>/api/gosecurityprovider/register</code>	Register new user (basic)
POST	<code>/api/gosecurityprovider/registerfull</code>	Register new user (full details)
POST	<code>/api/gosecurityprovider/logout</code>	Logout user
GET	<code>/api/gosecurityprovider/keepalive</code>	Keep session alive
POST	<code>/api/gosecurityprovider/changepassword</code>	Change user password

User Management

GOUser API

Method	Endpoint	Description
GET	<code>/api/gouser/byid/{id}</code>	Get user by ID
GET	<code>/api/gouser/list</code>	Get list of users
POST	<code>/api/gouser</code>	Create or update user
DELETE	<code>/dataset/api/gouser/{id}</code>	Delete user

User Roles & Groups

Method	Endpoint	Description
GET	/api/gorole/list	Get all roles
GET	/api/gogroup/list	Get all groups
GET	/api/gouserrole/list	Get user-role assignments
GET	/api/gousergroup/list	Get user-group assignments
POST	/api/gouserrole	Assign role to user
POST	/api/gousergroup	Assign user to group

Business & User Association

Method	Endpoint	Description
GET	/api/userbusiness/list	Get user-business associations
POST	/api/userbusiness	Associate user with business
DELETE	/dataset/api/userbusiness/{businessId}/{g0UserId}	Remove association

Core MVP Entities

Vehicle Management

Vehicle

Method	Endpoint	Description
GET	/api/vehicle/byid/{id}	Get vehicle by ID
GET	/api/vehicle/list	Get all vehicles
POST	/api/vehicle	Create or update vehicle
DELETE	/dataset/api/vehicle/{id}	Delete vehicle
GET	/api/vehicle/file/{id}/PhotoModel	Get vehicle photo

Vehicle Type

Method	Endpoint	Description
GET	/api/vehicletype/list	Get all vehicle types
POST	/api/vehicletype	Create or update vehicle type
DELETE	/dataset/api/vehicletype/{id}	Delete vehicle type

Vehicle Category

Method	Endpoint	Description
GET	/api/vehiclecategory/list	Get all vehicle categories
POST	/api/vehiclecategory	Create or update vehicle category
DELETE	/dataset/api/vehiclecategory/{id}	Delete vehicle category

Pre-Shift Checklist

Checklist

Method	Endpoint	Description
GET	/api/checklist/byid/{id}	Get checklist by ID
GET	/api/checklist/list	Get all checklists
POST	/api/checklist	Create or update checklist
DELETE	/dataset/api/checklist/{id}	Delete checklist

Checklist Items

Method	Endpoint	Description
GET	/api/checklistitem/list	Get all checklist items
POST	/api/checklistitem	Create or update checklist item
DELETE	/dataset/api/checklistitem/{id}	Delete checklist item

Checklist Answers

Method	Endpoint	Description
GET	/api/checklistanswer/list	Get all checklist answers
POST	/api/checklistanswer	Submit checklist answer
DELETE	/dataset/api/checklistanswer/{id}	Delete checklist answer

Answered Checklist Items

Method	Endpoint	Description
GET	/api/answeredchecklistitem/list	Get all answered items
POST	/api/answeredchecklistitem	Create answered item
DELETE	/dataset/api/answeredchecklistitem/{id}	Delete answered item

Vehicle Session

Method	Endpoint	Description
GET	/api/vehiclesession/byid/{id}	Get session by ID
GET	/api/vehiclesession/list	Get all sessions
POST	/api/vehiclesession	Create or update session
DELETE	/dataset/api/vehiclesession/{id}	Delete session

Incident Reporting

Incident Base

Method	Endpoint	Description
GET	/api/incident/byid/{id}	Get incident by ID
GET	/api/incident/list	Get all incidents
POST	/api/incident	Create or update incident
DELETE	/dataset/api/incident/{id}	Delete incident

Collision Incident

Method	Endpoint	Description
GET	/api/collisionincident/list	Get all collision incidents
POST	/api/collisionincident	Create or update collision incident
DELETE	/dataset/api/collisionincident/{id}	Delete collision incident

Near Miss Incident

Method	Endpoint	Description
GET	/api/nearmissincident/list	Get all near miss incidents
POST	/api/nearmissincident	Create or update near miss incident
DELETE	/dataset/api/nearmissincident/{id}	Delete near miss incident

Hazard Incident

Method	Endpoint	Description
GET	/api/hazardincident/list	Get all hazard incidents
POST	/api/hazardincident	Create or update hazard incident
DELETE	/dataset/api/hazardincident/{id}	Delete hazard incident

Vehicle Fail Incident

Method	Endpoint	Description
GET	/api/vehiclefailincident/list	Get all vehicle fail incidents
POST	/api/vehiclefailincident	Create or update vehicle fail incident
DELETE	/dataset/api/vehiclefailincident/{id}	Delete vehicle fail incident

Location & Business

Business

Method	Endpoint	Description
GET	/api/business/byid/{id}	Get business by ID
GET	/api/business/list	Get all businesses
POST	/api/business	Create or update business
DELETE	/dataset/api/business/{id}	Delete business

Business Configuration

Method	Endpoint	Description
GET	/api/businessconfiguration/list	Get all business configurations
POST	/api/businessconfiguration	Create or update business configuration
DELETE	/dataset/api/businessconfiguration/{id}	Delete business configuration

Site

Method	Endpoint	Description
GET	/api/site/byid/{id}	Get site by ID
GET	/api/site/list	Get all sites
POST	/api/site	Create or update site
DELETE	/dataset/api/site/{id}	Delete site

Country & State

Method	Endpoint	Description
GET	/api/country/list	Get all countries
GET	/api/countrystate/list	Get all states/provinces
POST	/api/country	Create or update country
POST	/api/countrystate	Create or update state/province

Certification Management

Method	Endpoint	Description
GET	/api/certification/byid/{id}	Get certification by ID
GET	/api/certification/list	Get all certifications
POST	/api/certification	Create or update certification
DELETE	/dataset/api/certification/{id}	Delete certification

Media Management

Method	Endpoint	Description
GET	/api/multimedia/byid/{id}	Get multimedia by ID
GET	/api/multimedia/list	Get all multimedia
POST	/api/multimedia	Create or update multimedia
DELETE	/dataset/api/multimedia/{id}	Delete multimedia
GET	/api/multimedia/file/{id}/Image	Get multimedia image

Media Associations

Method	Endpoint	Description
GET	/api/usermultimedia/list	Get user-media associations
GET	/api/vehiclemultimedia/list	Get vehicle-media associations
GET	/api/incidentmultimedia/list	Get incident-media associations

File Uploads

Method	Endpoint	Description
POST	/api/gofileuploader/uploadfile	Upload file

API Format Options

The API supports two formats for requests and responses:

1. JSON format: /api/... endpoints
2. Dataset format: /dataset/api/... endpoints (legacy format)

For MVP development, use the JSON format endpoints.

Error Handling

All endpoints return standard HTTP status codes:

- 200: Success
- 400: Bad request
- 401: Unauthorized
- 403: Forbidden
- 404: Not found
- 500: Server error

Implementation Notes

1. Authentication token should be included in the header of all requests
2. For creating or updating entities, use the POST method with the full entity data
3. When retrieving lists, use paging parameters to limit result size
4. File uploads require multipart/form-data content type
5. When working with relationships, ensure all required IDs are valid

GO Platform Integration Strategy

For the Android MVP implementation:

- 1. Start with authentication flow using gosecurityprovider endpoints
- 2. Implement vehicle check feature with checklist-related endpoints
- 3. Build session management using vehiclesession endpoints
- 4. Create incident reporting with the incident endpoints

Example Authentication Flow

```
kotlin

// Login example
val loginRequest = LoginRequest(username, password)
api.authenticate(loginRequest).enqueue(object : Callback<AuthResponse> {
    override fun onResponse(call: Call<AuthResponse>, response: Response<AuthResponse>) {
        if (response.isSuccessful) {
            val token = response.body()?.token
            // Store token securely
            // Navigate to main screen
        } else {
            // Handle error
        }
    }

    override fun onFailure(call: Call<AuthResponse>, t: Throwable) {
        // Handle network error
    }
})
```

