



Enov8 REST API's

Introduction .....	3
REST-API Prerequisites.....	3
Generating an API Key .....	3
Class Definitions.....	5
REST API End Points .....	5
Read REST API .....	7
Create REST API.....	10
Update REST API .....	12

## Introduction

Enov8's REST-API allows you to automatically Create, Update & Read information. Which in turn allows you to integrate other platforms and automation scripts, for example CICD scripts to capture "version", or test automation scripts to capture "health".

Currently the following three types of requests are supported.

1. POST (Create)
2. PUT (Update)
3. GET (Read)

## REST-API Prerequisites

The following will be required before using the enov8 REST-API

- The URL of your enov8 instance  
e.g. <https://company.enov8.com/ecosystem/api/>
- An "API KEY" comprising of the following parameters
  - app\_id (Application ID)
  - app\_key (Application Key)
- An enov8 platform user account with the required permissions
  - user\_id

Note: That a dedicated "Service User Account" is typically created.

URL Structure

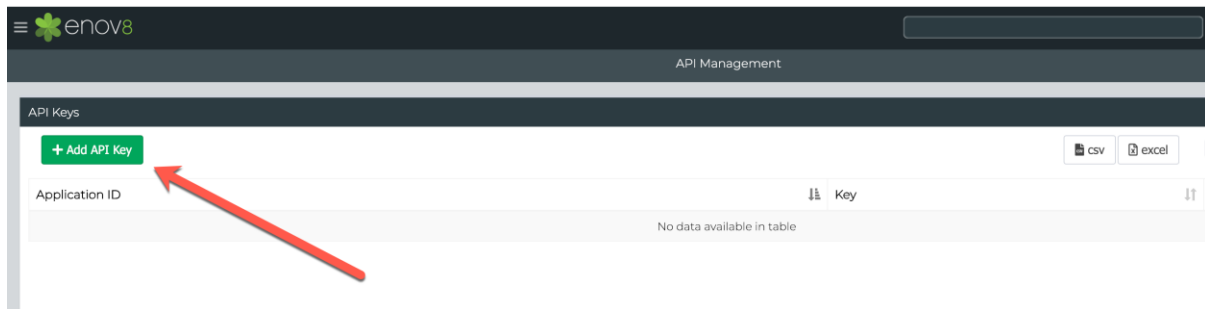
```
{ecosystem api url}/CLASSNAME
```

## Generating an API Key

The following snapshots have been taken from the EcoSystem support portal.

As a System Admin user login to the enov8 platform and navigate to **Configuration Management > System Administration > API Management**.

Click Add API Key

The image shows a 'New API Key' form. The form has a dark header with the title 'New API Key' and a close button (X). The form contains three main sections: 1. 'Application ID' with a text input field containing 'XYZ\_APP'. 2. 'Application Key' with a text input field containing 'demoapikey' and a 'Generate Random Key' link. 3. 'Permissions' with a list of selected permissions: 'Read', 'Create', and 'Update', each with a close button (X). At the bottom right of the form is a green 'Save' button.

Enter your Application ID and click Generate Random Key for your Application Key. Also select the permissions for the API Key (Read, Create, Update)

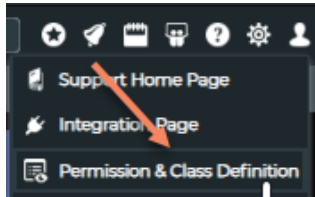
Note: Retain these details for future use.

Click Save.

You should receive a message saying API Key has been created successfully.

## Class Definitions

Permission & Class Definition module in ecosystem can be used to get details of the properties of a call for API calls and to under user permissions. Login to the enov8 platform and click on the help icon on top bar to open Permission & Class Definition



In Class/Property Definition widget select the class for which you want to see the property definition

Class Name	Property Name	Custom Display Name	Property Type	Mandatory	Max Allowed	Show in Quick Editor View	Edit Permission Roles*	Description
Booking	Icon		Icon	Yes	1	False	Ecosystem Administrator	The icon value of the object
Booking	System ID		SystemID	Yes	1	True	Ecosystem Administrator	This is the unique identifier for the object (Read Only)
Booking	System User		SystemUser	Yes	1	False	Ecosystem Administrator	The ecosystem user that created this object (Read Only)
Booking	System Time		SystemTime	Yes	1	False	Ecosystem Administrator	The time this object was created (Read Only)
Booking	Summary		String	Yes	1	True	Ecosystem Administrator System Admin Environment Manager Release Manager Data Manager Environment Coordinator Release Coordinator Data Coordinator Project Manager Config Manager	The unique summary of this Booking (Mandatory)
Booking	Description		Textarea	Yes	1	False	Ecosystem Administrator System Admin Environment Manager Release Manager Data Manager Environment Coordinator Release Coordinator Data Coordinator Project Manager Config Manager	The description of the Booking (eg. Additional details)
Booking	Project		Project	Yes	1	True	Ecosystem Administrator System Admin Environment Manager Release Manager Data Manager Environment Coordinator	The Project connected to this Booking (Mandatory)

## REST API End Points

The following is the list of API End Points available in ecosystem

Class Name	API End Point
Environment	{ecosystem api url}/Environment
System	{ecosystem api url}/System
SystemInstance	{ecosystem api url}/SystemInstance
SystemComponent	{ecosystem api url}/SystemComponent
User	{ecosystem api url}/User
LeanSR	{ecosystem api url}/LeanSR
EnvEvent	{ecosystem api url}/EnvEvent
Group	{ecosystem api url}/Group
Runsheet	{ecosystem api url}/Runsheet
Task	{ecosystem api url}/Task

Tag	{ecosystem api url}/Tag
BusinessUnit	{ecosystem api url}/BusinessUnit
BusinessTeam	{ecosystem api url}/BusinessTeam
BusinessProcess	{ecosystem api url}/BusinessProcess
Release	{ecosystem api url}/Release
Project	{ecosystem api url}/Project
Project<->System	{ecosystem api url}/Project<->System
Booking	{ecosystem api url}/Booking
Booking<->SystemInstance	{ecosystem api url}/Booking<->SystemInstance
Gate	{ecosystem api url}/Gate
Activity	{ecosystem api url}/Activity
PIR	{ecosystem api url}/PIR
TechnicalSpecification	{ecosystem api url}/TechnicalSpecification
TestAccount	{ecosystem api url}/TestAccount
ITService	{ecosystem api url}/ITService
System<->System	{ecosystem api url}/System<->System
SystemInstance<->SystemInstance	{ecosystem api url}/SystemInstance<->SystemInstance

## Read REST API

Data from ecosystem can be read from a particular class using GET REST API Call.

### URL Structure

```
{ecosystem api url}/ClassName&properties=PropertyNames&Conditions
```

### Header

The following parameters needs to be passed as part of request header to authenticate the request.

	KEY	VALUE
<input checked="" type="checkbox"/>	user-id	YOUR API USER ID
<input checked="" type="checkbox"/>	app-id	YOUR API ID
<input checked="" type="checkbox"/>	app-key	YOUR API KEY

PropertyNames : This fields can be used to specify the properties which are to be read as part of the GET Request

Conditions : Conditions can be used to specify any particular conditions for the read command.

### Example 1

```
{ecosystem api url}/SystemInstance?properties=Resource Name,Status&Status=InOperation&System=Tallyman
```

### Sample URL

```
http://companydemo.enov8.com/ecosystem/api/SystemInstance?properties=Resource Name,Status&Status=InOperation&System=Tallyman
```

In this example data is being read from System Instance class and properties returned are Resource Name and Status. A condition has been added to read data where status is equal to InOperation and System is equal to Tallyman.

## Example Output

```
Body Cookies Headers (8) Test Results
Pretty Raw Preview Visualize JSON
1 [
2   {
3     "Resource Name": "Tallyman (PROD)",
4     "Status": "InOperation"
5   },
6   {
7     "Resource Name": "Tallyman (DEV)",
8     "Status": "InOperation"
9   },
10  {
11   "Resource Name": "Tallyman (SIT)",
12   "Status": "InOperation"
13  },
14  {
15   "Resource Name": "Tallyman (SVP)",
16   "Status": "InOperation"
17  },
18  {
19   "Resource Name": "Tallyman (ST)",
20   "Status": "InOperation"
21  },
22  {
23   "Resource Name": "Tallyman (DR)",
24   "Status": "InOperation"
25  }
26 ]
```

## Example 2

```
{ecosystem api url}/SystemInstance?properties=Resource Name,Status&Status!=InOperation
```

## Sample URL

```
http://companydemo.enov8.com/ecosystem/api/SystemInstance?properties=Resource Name,Status&Status!=InOperation
```

In this example data is being read from System Instance class and properties returned are Resource Name and Status. A condition has been added to read data where status is not equal to InOperation.

## Example Output



Pretty Raw Preview Visualize JSON 

```
1  [
2  {
3    "Resource Name": "Tallyman (UAT)",
4    "Status": "UnplannedOutage"
5  },
6  {
7    "Resource Name": "SAS M0 (UAT)",
8    "Status": "PlannedOutage"
9  },
10 {
11  "Resource Name": "SAP (DEV)",
12  "Status": "UnplannedOutage"
13 },
14 {
15  "Resource Name": "MIS (SIT)",
16  "Status": "UnplannedOutage"
17 },
18 {
19  "Resource Name": "PDM (SVP)",
20  "Status": "Provisioning"
21 },
22 {
23  "Resource Name": "SAP (ST)",
24  "Status": "PlannedOutage"
25 },
26 {
27  "Resource Name": "GDW (DEV)",
28  "Status": "PlannedOutage"
29 }
```

## Create REST API

Data can be created in ecosystem for a particular class using POST REST API Call.

### URL Structure

```
{ecosystem api url}/ClassName
```

### Header

The following parameters needs to be passed as part of request header to authenticate the request.

KEY	VALUE
<input checked="" type="checkbox"/> user-id	YOUR API USER ID
<input checked="" type="checkbox"/> app-id	YOUR API ID
<input checked="" type="checkbox"/> app-key	YOUR API KEY

### Body

JSON Format data needs to be passed in Body for the data to be created in ecosystem for the particular class. The JSON should contain the property name and the value of the property. Please make sure all the mandatory properties are specified as part of the JSON Payload.

To understand the Property details, please refer to class definition module in ecosystem.

### Sample JSON Payload

```
1  {}
2  ... "Summary": "Demo - Event",
3  ... "Start Timestamp": "01-01-2022 00:00:00",
4  ... "End Timestamp": "15-01-2022 04:00:00",
5  ... "Environment": "ECO-000000004093",
6  ... "SystemInstance": "ECO-000000113980;ECO-000000114181",
7  ... "Type": "EnvDeployment",
8  ... "Status": "Planned",
9  ... "Assigned To": "ECO-000000004225",
10 ... "Organisation": "ECO-000000003945"
11 }
```

## Example 1

```
{ecosystem api url}/EnvEvent
```

## Sample URL

```
http://companydemo.enov8.com/ecosystem/api/EnvEvent
```

## Sample JSON

```
{
  "Summary": "Demo Event",
  "Start Timestamp": "01-01-2022 00:00:00",
  "End Timestamp": "15-01-2022 04:00:00",
  "Environment": "ECO-000000004093",
  "SystemInstance": "ECO-000000113980;ECO-000000114181",
  "Type": "EnvDeployment",
  "Status": "Planned",
  "Assigned To": "ECO-000000004225",
  "Organisation": "ECO-000000003945"
}
```

In this example data is being sent to ecosystem to create an environment event.

## Example Output

```
Body Cookies Headers (8) Test Results
Pretty Raw Preview Visualize JSON ↕
1  {
2    "success": true,
3    "total_attempted": 1,
4    "total_created": 1,
5    "result": [
6      {
7        "success": true,
8        "System ID": "ECO-000000240441"
9      }
10   ],
11   "value passed": [
12     {
13       "Summary": "Demo Event",
14       "Start Timestamp": "01-01-2022 00:00:00",
15       "End Timestamp": "15-01-2022 04:00:00",
16       "Environment": "ECO-000000004093",
17       "SystemInstance": "ECO-000000113980;ECO-000000114181",
18       "Type": "EnvDeployment",
19       "Status": "Planned",
20       "Assigned To": "ECO-000000004225",
21       "Organisation": "ECO-000000003945"
22     }
23   ]
24 }
```

## Update REST API

Data can be updated in ecosystem for a particular class using PUT REST API Call.

### URL Structure

```
{ecosystem api url}/ClassName
```

### Header

The following parameters needs to be passed as part of request header to authenticate the request.

KEY	VALUE
<input checked="" type="checkbox"/> user-id	YOUR API USER ID
<input checked="" type="checkbox"/> app-id	YOUR API ID
<input checked="" type="checkbox"/> app-key	YOUR API KEY

### Body

JSON Format data needs to be passed in Body for the data to be created in ecosystem for the particular class. The JSON should contain the property name and the value of the property which needs to be updated. Please make sure "System ID" is specified as part of the payload to specify the object being updated.

### Sample JSON Payload

```
1 {  
2   "System ID": "ECO-000000192052",  
3   "Status": "Passed"  
4 }
```

## Example 1

```
{ecosystem api url}/EnvEvent
```

### Sample URL

```
http://companydemo.enov8.com/ecosystem/api/EnvEvent
```

### Sample JSON

```
{  
  "System ID": "ECO-000000192052",  
  "Status": "Passed"  
}
```

In this example data is being sent to ecosystem to update the status an environment event with System ID "ECO-000000192052"

### Example Output

Body Cookies Headers (8) Test Results

```
Pretty Raw Preview Visualize JSON   
1 {  
2   "success": true,  
3   "total_attempted": 1,  
4   "total_updated": 1,  
5   "result": [  
6     {  
7       "success": true,  
8       "System ID": "ECO-000000192052"  
9     }  
10  ],  
11   "value passed": [  
12     {  
13       "System ID": "ECO-000000192052",  
14       "Status": "Passed"  
15     }  
16  ]  
17 }
```