



# EXCELLATOR

User Manual – v1.0

## Document Control

This document has been developed by the EnoviQ team to explain some of the commonly used features in Excellator.

This document is not an exhaustive list of features and functions, but it does provide a general overview of the user flows and typical features that are used in day-to-day use of Excellator.

For more detailed insight into a particular function or feature, please contact the EnoviQ team to arrange a walk-through.

## Version Control

Version Number	Author	Date	Comments
1.0	Manish Sharma	April 2025	Initial Version
1.1			

## Table of Contents

1. Business Case .....	3
Executive Summary.....	3
Problem Statement.....	3
Proposed Solution.....	3
Benefits .....	3
Financial Impact.....	3
Conclusion.....	3
2. Login.....	4
3. Home.....	5
4. Modeling (Logic Development).....	6
4.1 Ist keyword.....	6
4.2 Xinput keyword.....	6
4.3 Xoutput keyword.....	6
4.4 Print keyword .....	7
5. Model upload to Excellator.....	8
6. API Details .....	10
6.1 Service Version.....	10
6.2 Endpoint URL .....	11
6.2.1 Input (Request Payload) .....	11
6.2.2 Output (Response Payload).....	12
6.3 Service Parameters .....	12
7. Version Control .....	14

# 1. Business Case

## Executive Summary

In today's fast-paced business environment, organizations face the challenge of rapidly implementing new functionalities while minimizing coding efforts. Our new robust product addresses this challenge by enabling users to write logic directly within Excel, which is then seamlessly converted into an API. This eliminates the need for complex programming and accelerates integration with existing systems.

## Problem Statement

Developing new functionalities and integrating them into existing systems often requires significant coding efforts, time, and expertise. This complexity can delay project timelines and increase costs. Additionally, maintaining and updating such code becomes cumbersome as business requirements evolve.

## Proposed Solution

Our product revolutionizes the development process by allowing users to write logic directly within Excel spreadsheets. This logic is then automatically converted into APIs that can be easily integrated with existing systems. This approach significantly reduces development time and complexity while empowering non-programmers to build functional solutions.

## Benefits

- **Reduced Development Time:** Rapidly develop and deploy new features with minimal coding, boosting speed by 5-10 times.
- **Easy Integration:** Seamlessly connect with existing systems without major architectural changes.
- **Cost Efficiency:** Cut down the requirement for specialized programming resources by 40-50%.
- **Enhanced Flexibility:** Rapidly adapt to changing business requirements by updating Excel logic.
- **Enabling Non-Technical Users:** Allows users with limited coding skills to participate actively.

## Financial Impact

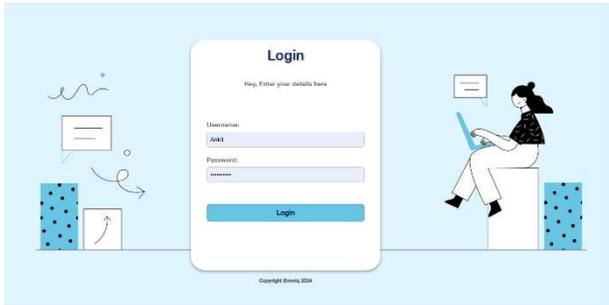
By minimizing coding requirements and leveraging familiar tools like Excel, our product significantly reduces development costs and project timelines. The reduced dependency on specialized programming resources also translates into cost savings and operational efficiency.

## Conclusion

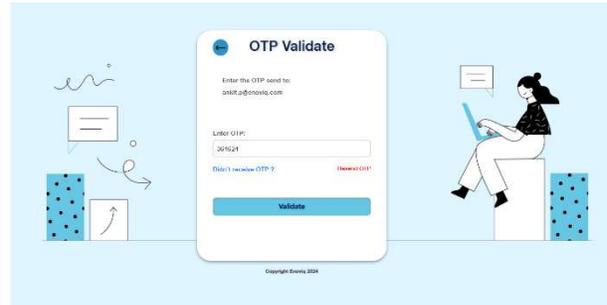
Excellator offers a groundbreaking solution for businesses seeking agility, cost reduction, and seamless integration. By adopting this product, organizations can rapidly build and deploy new functionalities without the usual complexities associated with traditional software development.

## 2. Login

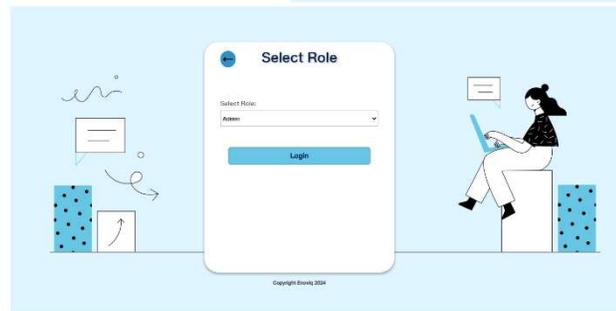
- Login with UserName and Password which is assigned to user and click on Login button
- An OTP will be sent to the linked Email/Mobile. Enter that OTP and select user role to proceed further.



Login Page



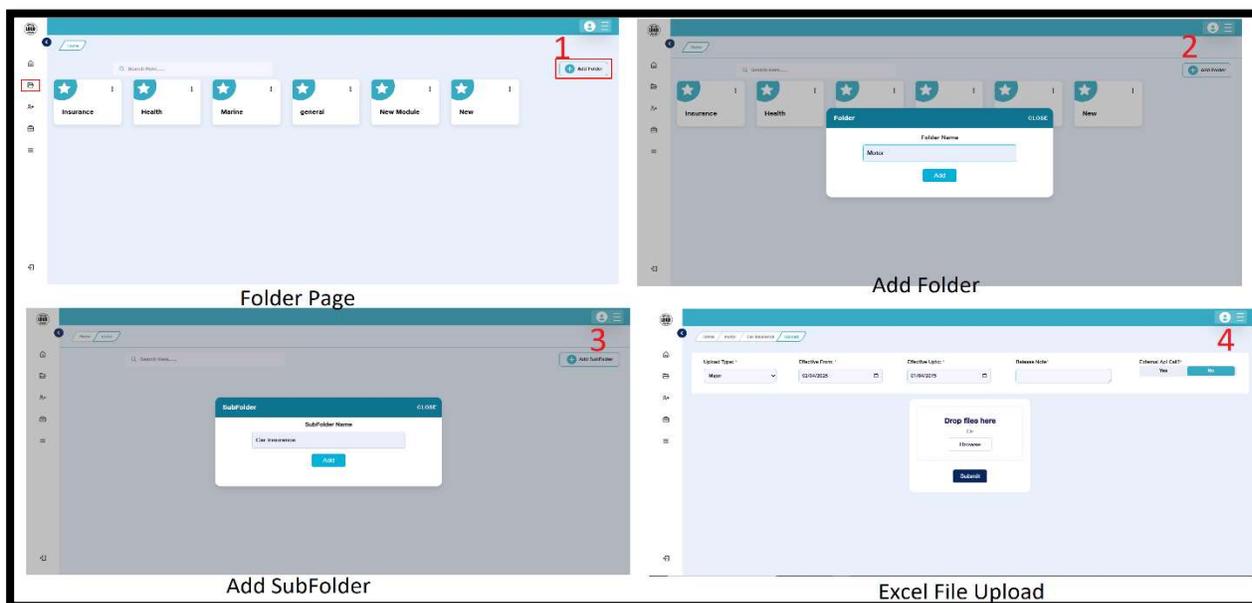
OTP Page



Select Role Page

### 3. Home

- After validation, the user lands on the **Home Page**
- Users can view existing folders and also can create new ones via **"Add Folder"**. (Ref Image 1,2)
- A pop-up allows entering the folder name; clicking **"Add"** will creates it with a success message.
- Clicking on folder will provide the access to add subfolders. (Ref Image 3)
- Users can create multiple folders and subfolders, as needed.
- Click on the subfolder. After clicking on the subfolder, a page will open where the user can see the following fields: Upload Type (dropdown with Major, Minor, Patch), Effective From, Effective Up To, Release Note, and an input field to upload your Excel file. Select the upload type (Major), choose the "Effective From" date, select the "Effective Up To" date, enter the release note, and then either drag and drop the Excel file or browse to upload it. (Ref Image 4)



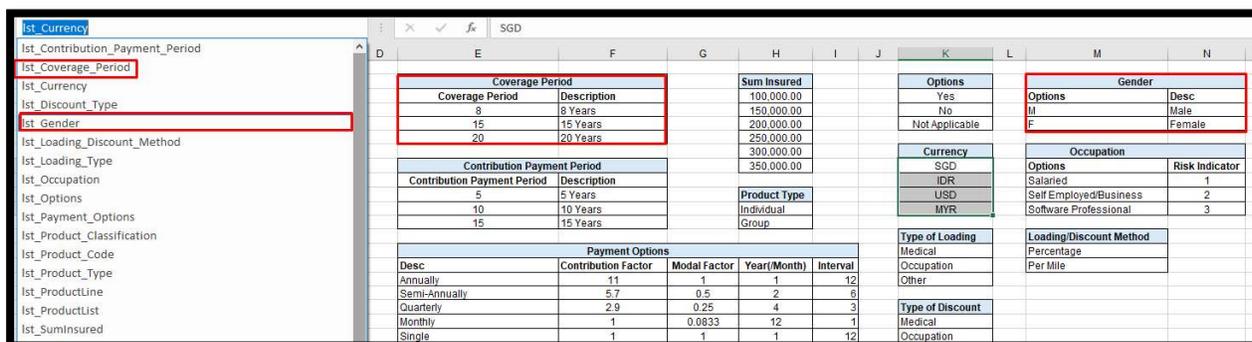
## 4. Modeling (Logic Development)

- Business Users (Product Team, Loan Department, Inventory Manager, Rate Designer) can also create their own excel by applying logic inside excel as per your need
- There are certain keyword definitions which are applied to develop the logic in excel spreadsheets
  - Ist** – List Definitions for in-cell Dropdowns,
  - Xinput** – Input Parameters part of API Request Payload,
  - Xoutput** – Output Parameters which are part of Response
  - PrintDocument**- convert pdf from excel sheet
- Ist is for creating master data whereas Xinput is for input data and Xoutput is for getting the output from master data and input data
- There is also one keyword **Print** for converting Excel sheet into Pdf

Below is the excel where user can see logic. In this Excel user can create Master Sheet, input sheet, output sheet.

### 4.1 Ist keyword

- In the master sheet, user can create master data using Ist keyword
- Select the cells for which the user wants to create master data and change the name of the selected cell in 'Name Box' by **prefixing** the Ist keyword. (*i.e. Ist\_Gender*)
- This Ist\_Gender user can now be used in any input field wherever required.



Coverage Period		Sum Insured
Coverage Period	Description	
8	8 Years	100,000.00
15	15 Years	150,000.00
20	20 Years	200,000.00
		250,000.00
		300,000.00
		350,000.00

Options		Gender	
Options	Yes	Options	Desc
	No	M	Male
	Not Applicable	F	Female

Contribution Payment Period		Product Type
Contribution Payment Period	Description	
5	5 Years	Individual
10	10 Years	Group
15	15 Years	

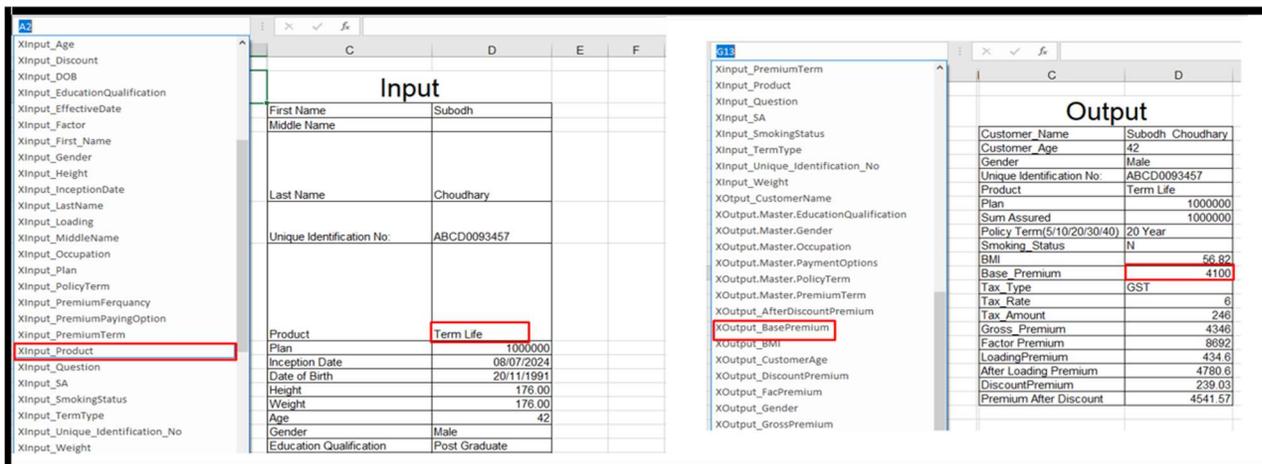
Payment Options			
Desc	Contribution Factor	Modal Factor	Year(Month) Interval
Annually	11	1	1 12
Semi-Annually	5.7	0.5	2 6
Quarterly	2.9	0.25	4 3
Monthly	1	0.0833	12 1
Single	1	1	1 12

### 4.2 Xinput keyword

- In Input sheet, user can create input field using Xinput keyword.
- Select the cell for which user wants to create input field and change the name of that cell by **prefixing** the Xinput keyword (*i.e. Xinput\_Product\_name*)

### 4.3 Xoutput keyword

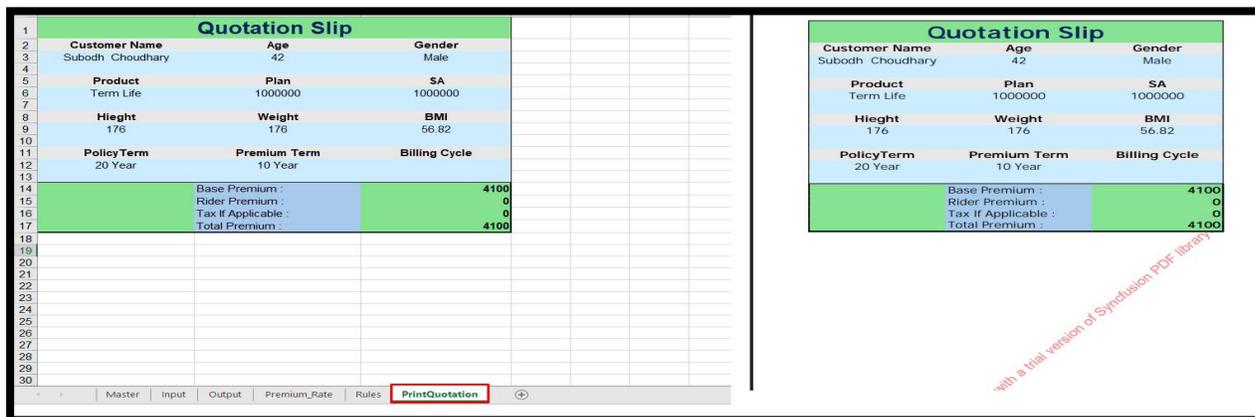
- In output sheet, user can create output field using Xoutput keyword.
- Select the cell for which user wants to create output field and change the name of that cell by **prefixing** the Xoutput keyword (*i.e. Xoutput\_BasePremium*)



#### 4.4 Print keyword

- If user wants to create a pdf of any excel sheet, then user needs to add **Print** keyword before the sheet name & the same will be converted to PDF (as shown in the image below)

Note: *This step is optional, but it will be used if the user wants to generate the PDF*



Once step 4.1, 4.2, 4.3 & 4.4 is done then logic can be applied into the excel file(model) & is ready to upload

Once the excel file(model) is uploaded successfully then the API end point is generated along with the version number which will start with 1.0.0



## 5. Model upload to Excellator

Conditions for preparing the model (Excel file).

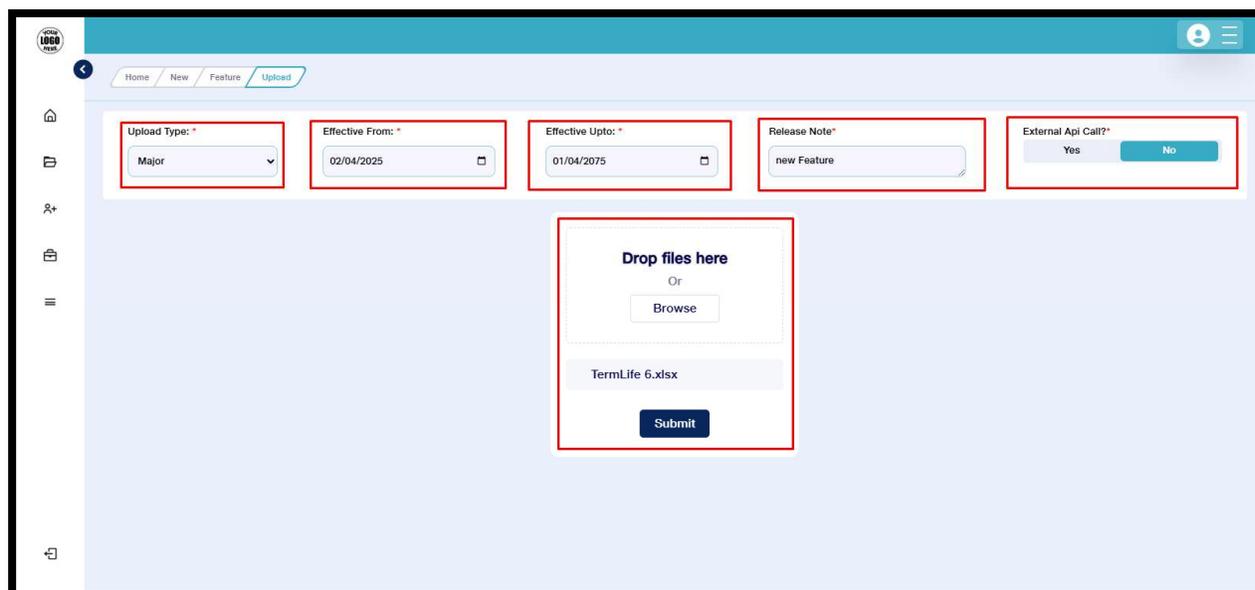
- Prepare Excel Model according to business logic and calculation.
- Make sure there are no spaces in Filename
- Engine name will be same as Upload file name- so make sure to give a relevant name that should match with the use case.

There are multiple modes for uploading a Model to Excellator:

- New Model: This is considered an Initial version under Major category
- Revised Model: This is controlled via
  - Major: Complex features are changed, and engine version will be x+1
  - Minor: Small changes related to calculations or payload/response changes and engine version will be x+.1
  - Patch: This is for cosmetic change and the engine version will be x+.01

In future, if user wants to change something in excel than user can change and upload the updated excel in the same subfolder by selecting uploaded type minor, patch based on your change so it will manage version

Select upload type, choose effective from and effective up to date, add release not, if any and then drop or browse the file and click on submit button



The screenshot shows the 'Upload' page in the EnoviQ application. The page has a teal header with the EnoviQ logo and a user profile icon. Below the header is a navigation bar with 'Home', 'New', 'Feature', and 'Upload' tabs. The main content area contains several form fields, each highlighted with a red border:

- Upload Type:** A dropdown menu with 'Major' selected.
- Effective From:** A date input field with '02/04/2025' and a calendar icon.
- Effective Upto:** A date input field with '01/04/2075' and a calendar icon.
- Release Note:** A text input field containing 'new Feature'.
- External Api Call?:** A toggle switch with 'Yes' and 'No' options, currently set to 'No'.

Below these fields is a large central area for file upload, also highlighted with a red border. It contains the text 'Drop files here' and 'Or' above a 'Browse' button. Below the 'Browse' button is a file list showing 'TermLife 6.xlsx' and a 'Submit' button.

**In case, if the excel file(model) is uploaded with the upload type as minor**

- New version of the file will be created with new version level and second position will be increased by 1 in version level. For e.g. If the version before uploading the minor changes in excel was 1.0.0 then after uploading the excel the new version will be generated as 1.1.0.

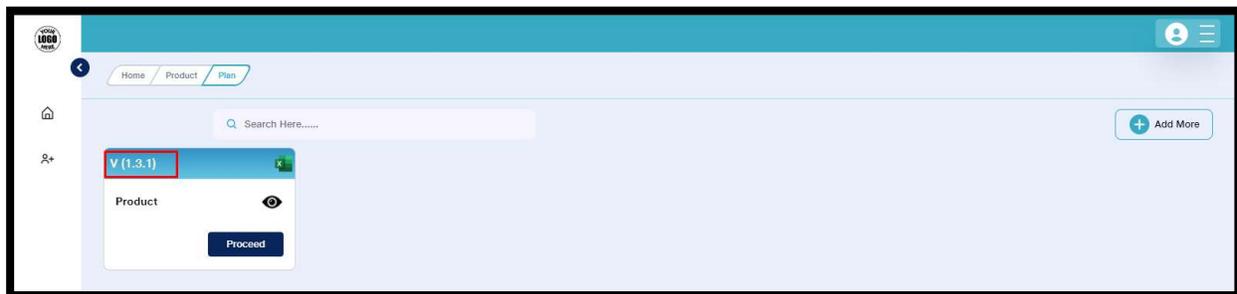
**In case, if the excel file(model) is uploaded with the upload type as Patch**

- New version of the file will be created with new version level and third position will be increased by 1 in version level. For e.g. If the version before uploading the patch in excel was 1.0.0 then after uploading the excel the new version will be generated as 1.0.1

Version Level	Active Period From	Active Period To	Unpublished Date	Release Note	Action
V1.0.1	2024-09-20	2024-09-22	2024-09-22T06:46:48Z	nothing	
V1.0.0	2024-09-20	2024-09-22	2024-09-22T06:46:14Z	minor change	
V1.0.0	2024-09-20	2024-09-19	2024-09-20T07:55:02Z	minor changes	
V1.0.0	2024-09-20	2024-09-19	2024-09-20T06:56:02Z	changes	
V1.0.0	2024-09-20	2024-09-19	2024-09-20T05:31:28Z	Plan 1	

Version Level	Active Period From	Active Period To	Unpublished Date	Release Note	Action
V1.1.0	2024-09-20	2024-09-22	2024-09-22T06:46:48Z	nothing	
V1.0.0	2024-09-20	2024-09-22	2024-09-22T06:46:14Z	minor change	
V1.0.0	2024-09-20	2024-09-19	2024-09-20T07:55:02Z	minor changes	
V1.0.0	2024-09-20	2024-09-19	2024-09-20T06:56:02Z	changes	
V1.0.0	2024-09-20	2024-09-19	2024-09-20T05:31:28Z	Plan 1	

The latest version found in the subfolder will be shown here.



## 6. API Details

Once the Model is uploaded, Excellator provides an API endpoint with complete details.

- Service Parameters: (marked as 1)
- Service Version: (marked as 2)
- Endpoint: (marked as 3)

Range	Range values
Service Name	TermLife6
Version	V1.0.0
Correlation_Id	0d8b8716-4d5e-c880-d0ef-779c198ddeb1
Size of original Excel file	23.42 KB
Number of sheets	6 (0 hidden)

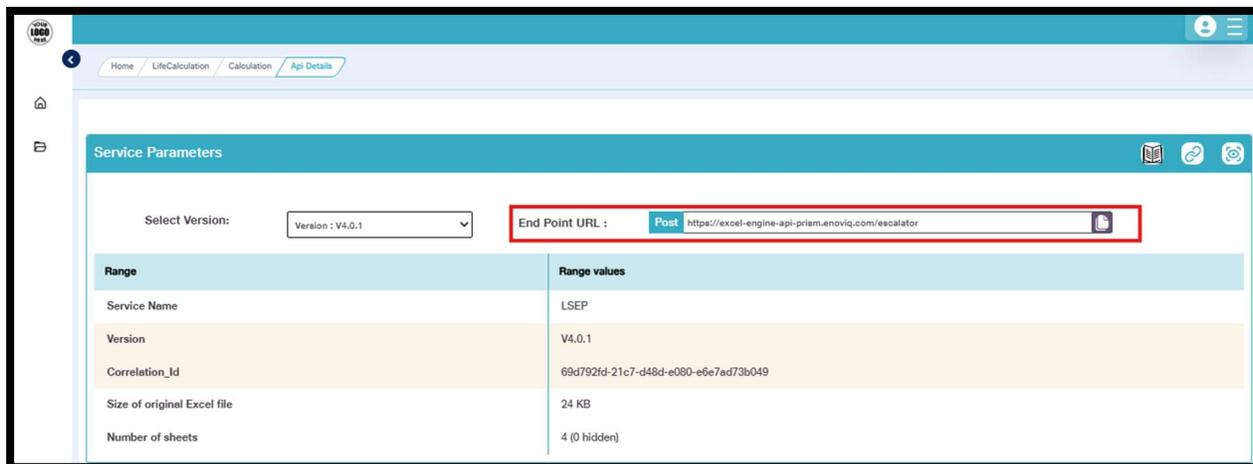
### 6.1 Service Version

All the available versions of Excel file (Model) appears here from which the user can select the version, as required.

Range	Range values
Service Name	LSEP
Version	V4.0.1
Correlation_Id	69d792fd-21c7-d48d-e080-e6e7ad73b049
Size of original Excel file	24 KB
Number of sheets	4 (0 hidden)

## 6.2 Endpoint URL

Endpoint URL will appear here which can be used to call out the API

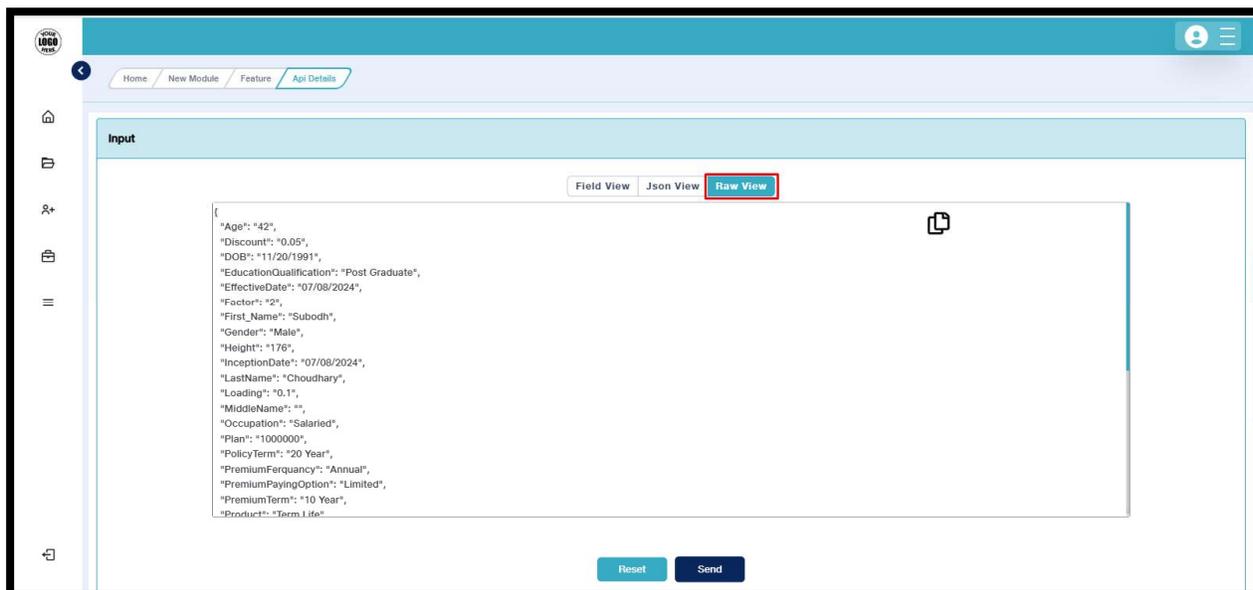


The screenshot shows the 'Service Parameters' section of a web application. At the top, there is a 'Select Version:' dropdown menu set to 'Version : V4.0.1'. To its right, the 'End Point URL :' field is highlighted with a red border and contains the text 'Post https://excel-engine-api-priam.enoviq.com/escalator'. Below this, there is a table with two columns: 'Range' and 'Range values'.

Range	Range values
Service Name	LSEP
Version	V4.0.1
Correlation_Id	69d792fd-21c7-d48d-e080-e6e7ed73b049
Size of original Excel file	24 KB
Number of sheets	4 (0 hidden)

### 6.2.1 Input (Request Payload)

All the input parameters from the Excel file (model) are converted into JSON format and included in the request payload.



The screenshot shows the 'Input' section of the web application. At the top, there are three tabs: 'Field View', 'Json View', and 'Raw View', with 'Raw View' selected and highlighted in red. Below the tabs is a large text area containing a JSON object representing the request payload. At the bottom of the text area, there are two buttons: 'Reset' and 'Send'.

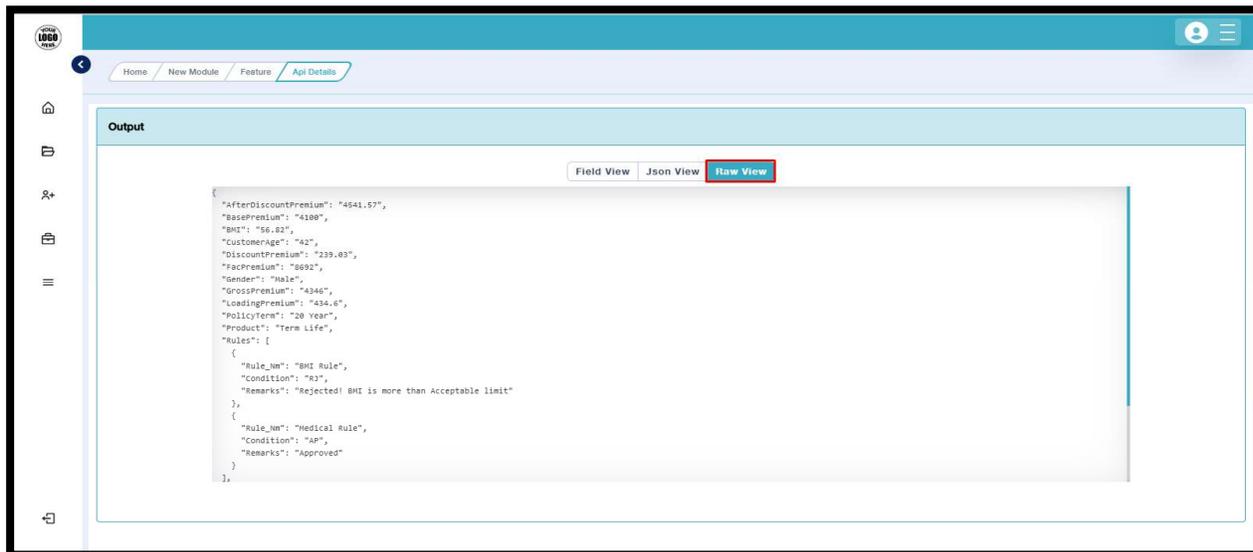
```

{
  "Age": "42",
  "Discount": "0.05",
  "DOB": "11/20/1991",
  "EducationQualification": "Post Graduate",
  "EffectiveDate": "07/08/2024",
  "Factor": "2",
  "First_Name": "Subodh",
  "Gender": "Male",
  "Height": "176",
  "InceptionDate": "07/08/2024",
  "LastName": "Choudhary",
  "Loading": "0.1",
  "MiddleName": "",
  "Occupation": "Salaried",
  "Plan": "1000000",
  "PolicyTerm": "20 Year",
  "PremiumFrequency": "Annual",
  "PremiumPayingOption": "Limited",
  "PremiumTerm": "10 Year",
  "Product": "Term Life"
}

```

## 6.2.2 Output (Response Payload)

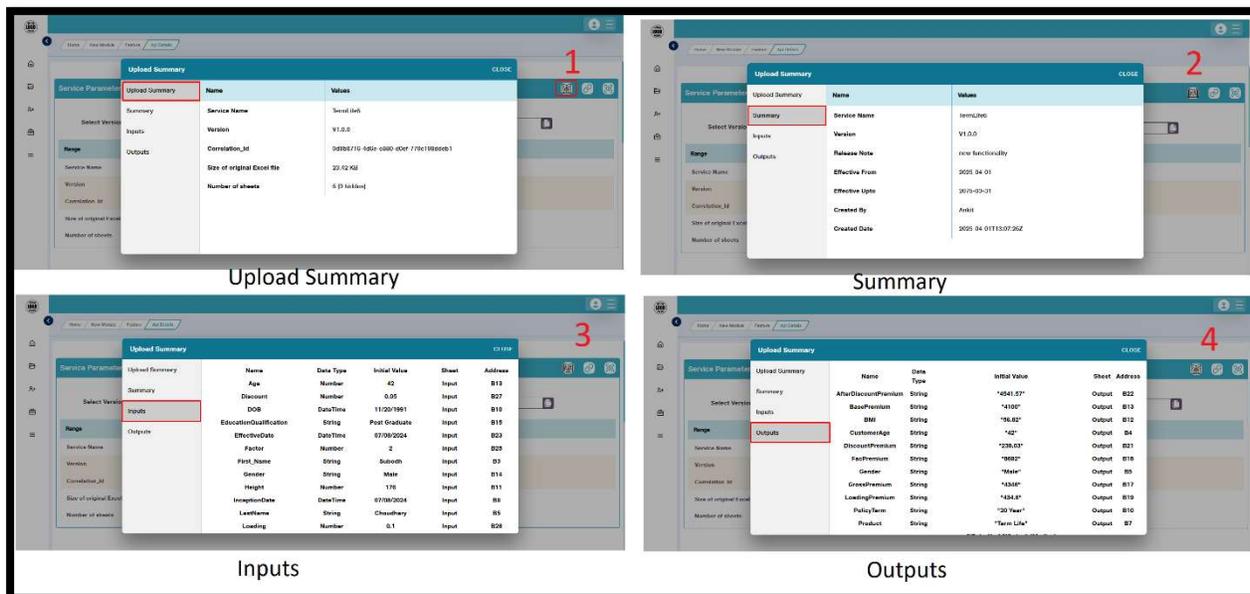
Now based on the payload request, the response will be generated accordingly



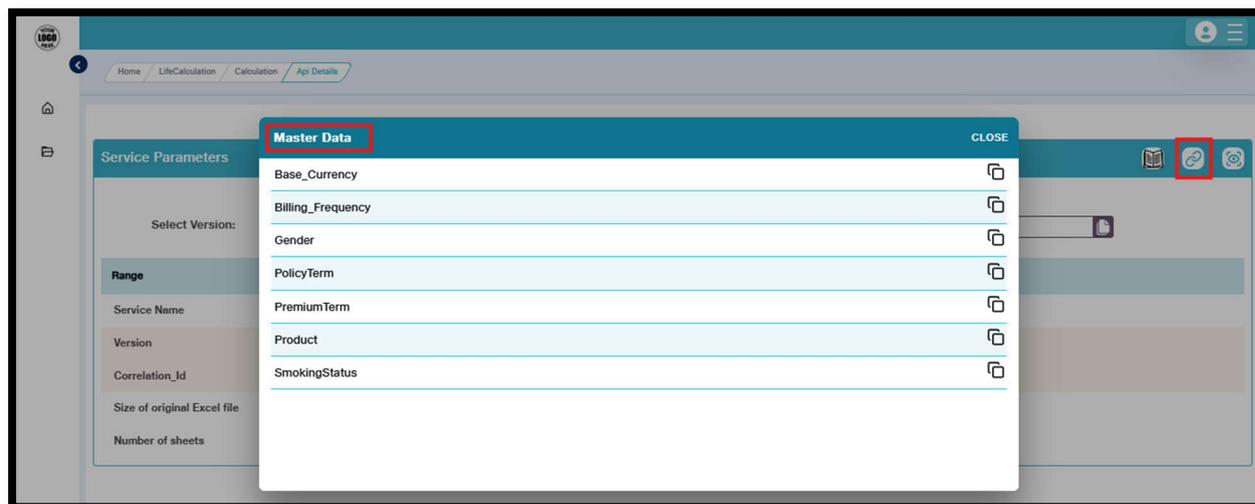
## 6.3 Service Parameters

**Book Symbol:** Click to view uploaded Excel details. (Refer Image 1)

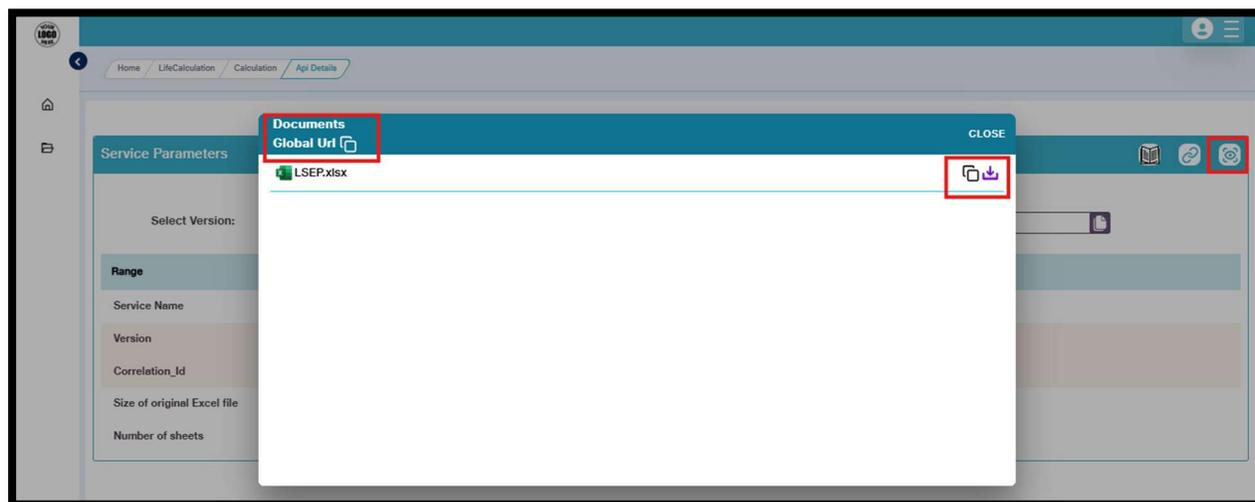
- **Upload Summary:** Shows service name, version and Correlation Id. (Refer Image 1)
- **Summary:** Model upload details like release notes, effective dates, creator & creation date. (Refer Image 2)
- **Inputs:** All input parameters details with cell address. (Refer Image 3)
- **Outputs:** All output parameters with cell address. (Refer Image 4)



**Master data symbol:** Click on the master data symbol to view the summary of Masters available



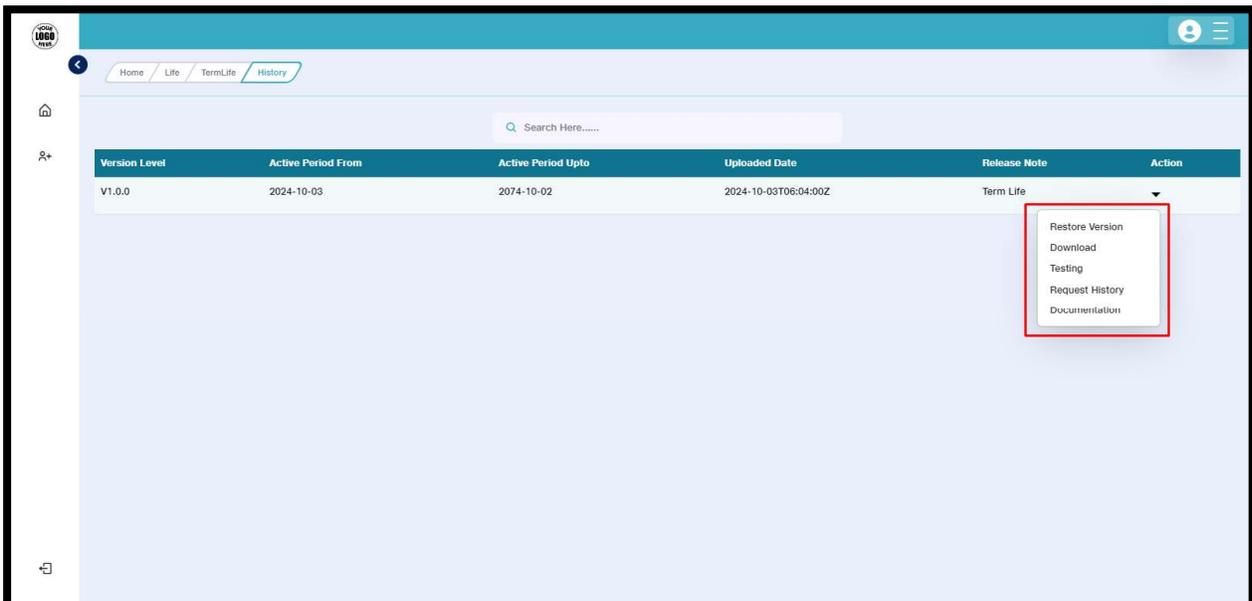
**Documents symbol:** Clicking on the document symbol User can download the uploaded excel file (model) & PDF version, if available



## 7. Version Control

Here the user can have access to all the available versions against the Uploaded Excel File (Model) with the details like Version Level, Effective Period, Uploaded Date & Release Note. In addition to this some actions can be performed which are appended below

- **Restore Version** – Any version can be restored & can be used from the list of available versions
- **Download** – The Excel file (Model) can be downloaded in either its encrypted or original form.
- **Testing** – User can test the model's logic through the user interface (UI).
- **Request History** – User can view the history of the requested APIs
- **Documentation** – View the documentation preview as a URL or through Swagger.



The screenshot displays the 'History' tab of the EnoviQ interface. It features a table with the following columns: Version Level, Active Period From, Active Period Upto, Uploaded Date, Release Note, and Action. A dropdown menu is open under the 'Action' column for the first row, showing options: Restore Version, Download, Testing, Request History, and Documentation. The 'Restore Version' option is highlighted with a red box.

Version Level	Active Period From	Active Period Upto	Uploaded Date	Release Note	Action
V1.0.0	2024-10-03	2074-10-02	2024-10-03T06:04:00Z	Term Life	<ul style="list-style-type: none"><li>Restore Version</li><li>Download</li><li>Testing</li><li>Request History</li><li>Documentation</li></ul>