

PROGRAM ●
Genie

Ver3.3

Training Handbook

(ver1.10)



1	<i>ProgramGenie Introduction & Overview</i>	10
1.1	What is ProgramGenie?	10
1.2	How Does ProgramGenie Accelerate Delivery Across the SDLC?	10
2	<i>Introduction to Projects</i>	11
2.1	Create a New Project	11
2.1.1	+ (New Project)	11
2.1.2	Image	12
2.1.3	Name	12
2.1.4	Lead	12
2.1.5	Description	12
2.1.6	Users	12
2.1.7	New Connector	12
2.1.8	Exit New Project	12
2.1.9	Save	12
2.2	Search for a Project	13
2.3	Show a Project as List or Tiles	13
2.4	Deactivate & Activate a Project	14
2.4.1	Show Inactive	14
2.4.2	Deactivate (dark standby icon)	14
2.4.3	Activate (light standby icon)	14
3	<i>Dashboard, Settings & Widgets</i>	15
3.1	Dashboard Screen	15
3.1.1	Basic Information	15
3.1.2	Artefacts Information	15
3.1.3	Active Connectors	16
3.2	Settings Screen	16
3.2.1	Project Settings	16
3.2.2	ProgramGenie Process Workflow	16



3.3	Widgets Screen	17
3.3.1	Customise	18
3.3.2	Add Card	18
3.4	Widgets actions	21
3.4.1	Minimize & Maximize Widget	21
3.4.2	Edit Widget	21
3.4.3	Check for new data	22
3.4.4	Width	22
3.4.5	Height	23
3.4.6	Hide Widget	23
3.4.7	Delete Widget	24
3.5	Search and sort Widget	24
3.6	Refresh Ingestions	25
3.7	Sorting	26
3.8	Edit Custom Widget	27
3.9	Widget Artefact Action Menu	27
3.10	Widget Artefact Function Menu	28
3.10.1	Export	28
3.10.2	Download (Word/PDF)	28
3.10.3	View Integrations	29
3.10.4	Delete	29
3.11	Default Widgets	30
3.11.1	Collaterals Widget	30
3.11.2	Documents Widget	30
3.11.3	Epic Widget	30
3.11.4	Story Widget	30
3.11.5	SubTask Widget	30
3.11.6	Test Widget	30
3.11.7	Wireframes Widget	30



3.11.8	Code Repository Widget	31
3.11.9	Validate Widget	31
4	<i>Uploading Collateral</i>	33
4.1	Insert Document	Error! Bookmark not defined.
4.2	Public URL	35
4.3	Create now document	36
5	<i>Generating Artefacts</i>	38
5.1	Generation Screen – Sources Area	38
5.1.1	Edit	38
5.1.2	Sources Toggle	39
5.1.3	Target	39
5.1.4	Add Source (+)	40
5.1.5	Delete Source (trash)	40
5.1.6	Generate	40
5.2	Generation Screen – Output Area	40
5.2.1	n Out Of n (next/previous)	41
5.2.2	Edit	41
5.2.3	Save Artefact	41
5.3	Generate Multiple Versions	41
5.4	Notifications	42
5.4.1	Delete	42
5.4.2	Open	42
5.4.3	Stop	42
5.4.4	Blue Dot	42
5.5	Generate a Project Brief	43
5.6	Generate Epics	45
5.7	Generate User Stories	48
5.8	Generate Test Cases	51
5.9	Generate SubTasks	53



5.10	AI Supported Content Editing	55
5.10.1	Simplify	55
5.10.2	Shorten	56
5.10.3	Lengthen	56
5.10.4	Fix	56
5.10.5	Prompt	56
6	<i>Connector Configuration</i>	57
6.1	Intro to Connectors	57
6.2	Jira Connector	57
6.2.1	Integration	57
6.2.2	Name	57
6.2.3	Description	57
6.2.4	Domain	57
6.2.5	Token	57
6.2.6	Email	59
6.2.7	Project Key	59
6.3	Confluence Connector	60
6.3.1	Integration	60
6.3.2	Name	60
6.3.3	Description	60
6.3.4	Domain	60
6.3.5	Token	60
6.3.6	Email	60
6.3.7	Project Key	60
6.4	Zephyr Connector	61
6.4.1	Integration	61
6.4.2	Name	61
6.4.3	Description	61
6.4.4	Domain	61



6.4.5	Token	61
6.4.6	Email	62
6.4.7	Project Key	62
6.5	Azure DevOps (ADO) Connector	63
6.5.1	Integration	63
6.5.2	Name	63
6.5.3	Description	63
6.5.4	Organization	63
6.5.5	Token	63
6.5.6	Email	64
6.5.7	Project Key	64
6.6	GitLab Connector	65
6.6.1	Integration	65
6.6.2	Name	65
6.6.3	Description	65
6.6.4	Domain	65
6.6.5	Token	65
6.6.6	Project	67
6.6.7	Branch	67
6.6.8	Email	67
6.6.9	Project Key	67
6.7	BitBucket Connector	68
6.7.1	Integration	68
6.7.2	Name	68
6.7.3	Description	68
6.7.4	Domain	68
6.7.5	Token	68
6.7.6	Project	68
6.7.7	Branch	68



6.7.8	Email	68
6.7.9	Project Key	68
6.8	GitHub Connector	69
6.8.1	Integration	69
6.8.2	Name	69
6.8.3	Description	69
6.8.4	Domain	69
6.8.5	Token	69
6.8.6	Project	69
6.8.7	Branch	69
6.8.8	Email	69
6.8.9	Project Key	69
6.9	Test-Rail Connector	70
6.9.1	Integration	70
6.9.2	Name	70
6.9.3	Description	70
6.9.4	Domain	70
6.9.5	Token	70
6.9.6	Email	70
6.9.7	Project Key	70
6.10	Xray Connector	71
6.10.1	Integration	71
6.10.2	Name	71
6.10.3	Description	71
6.10.4	Domain	71
6.10.5	Token	71
6.10.6	Email	71
6.10.7	Project Key	71
7	<i>Importing & Exporting Artefacts via Connectors</i>	74



7.1	Export to Confluence	74
7.1.1	Download (Word/PDF)	76
7.1.2	Edit	77
7.2	Export artefacts to Jira	79
7.4	Export a Test Case to Zephyr	82
7.4.1	Enable Zephyr Scale	84
7.5	Repositories Widget	84
7.5.1	Repository View	85
7.5.2	Open Repository Description	85
7.5.3	Open Repository File Summary	86
7.5.4	Generate Project Brief	86
7.5.5	Generate Epic from Repository	86
8	<i>Generating More Artefacts</i>	87
8.1	Generate a Wireframe Description	87
8.2	Generate Wireframes	89
8.3	Edit a Wireframe	91
8.4	Generate API Documentation	94
8.5	Generate Database Documentation	96
8.6	Generate Individual Artefacts	97
8.6.1	Generate a manual artifact from the artifact Widget	97
8.6.2	Generate an artefact from the artefact Widget using ProgramGenie AI	99
9	<i>ProgramGenie Chat</i>	102
9.1	ProgramGenie Chat Area	102
9.1.1	Toggle Chat	102
9.1.2	Prompt Field	102
9.1.3	Add File	103
9.1.4	Output Screen	103
9.1.5	New Chat	104
9.1.6	Chat History	105



9.1.7	Delete Chats	105
9.1.8	Copy	105
9.1.9	Duplicate Message	105
9.1.10	Human in the Loop	105
9.1.11	Web Search	106
9.2	Basic Prompting	107
9.2.1	Descriptive Prompting	107
9.2.2	Instructional Prompting	107
9.2.3	Comparative Prompting	107
9.2.4	Creative Prompting	108
9.2.5	Analytical Prompting	108
9.3	Onboarding Prompting	108
9.4	Adding Projects	108
10	Genie Vision	109
10.1	Download the Browser Extension	109
10.2	Install for Microsoft Edge	109
10.3	Install for Google Chrome	110
10.4	Basic Operation	111
11	Identity	112
11.1	Personal Access Tokens	112
11.1.1	Add Token	112
11.1.2	Filter Tokens	113
11.1.3	Active Token	114
11.1.4	Revoke Token	114



1 ProgramGenie Introduction & Overview

1.1 What is ProgramGenie?

ProgramGenie is revolutionary AI-powered Software Development Lifecycle (SDLC) Assistant, designed to accelerate software delivery. It acts as a team of virtual subject matter experts, including, solution and enterprise architects, product owners, business and data analysts, software engineers, manual testers, and SDETs.

ProgramGenie transforms software development by automating program artifact creation, performing impact and source code analysis, enabling teams to **build faster, smarter, and more efficiently.**

1.2 How Does ProgramGenie Accelerate Delivery Across the SDLC?

ProgramGenie’s intuitive UI interface enables rapid ingestion and analysis of various data sources, including Word documents, Excel spreadsheets, PowerPoint presentations, images, videos, source code, URLs, Confluence and Wiki pages, and audio files. It then uses this information to generate essential program artefacts, such as:

- **Project Briefs:** Defining the goals and objectives of a project.
- **Epics:** High level representations of key features, functionality, or business requirements.
- **Weighted User Stories:** Structured in the persona, “as a (type of user), I want (feature or action), so that (benefit or value)” with corresponding acceptance criteria.
- **Wireframes:** Providing a visual outline of the structured layout of the app without focusing on the design details.
- **Tasks:** What needs to be done by whom and in what order.
- **API and Database:** Documentation of API’s and database models.
- **Gherkin-style tests:** Created in Zephyr, Zephyr Enterprise, Octane, etc. in the format “Given (precondition or context), When (action or event), Then (expected outcome)”.
- **Automated Test Scripts:** If UI HTML pages already exist, ProgramGenie can automatically generate test scripts using Selenium, Cypress, or Playwright.

“Utilizing these capabilities, ProgramGenie reduces workloads from months or weeks to just hours or minutes.”

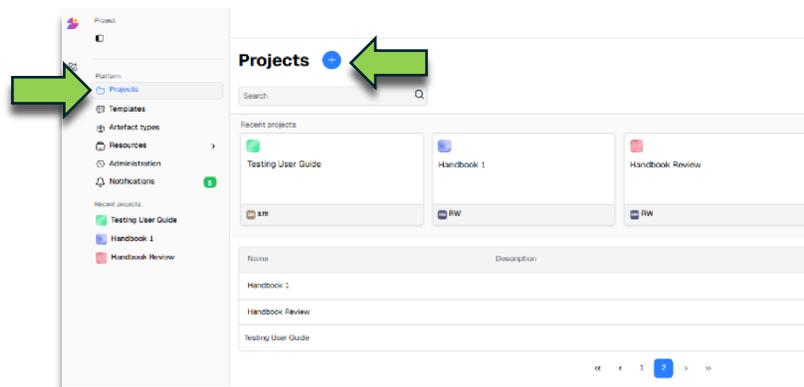


2 Introduction to Projects

2.1 Create a New Project

ProgramGenie uses 'Projects' to segregate work within an environment. To create a new project:

1. Click on 'Projects' within the 'ProgramGenie' sidebar on the left side of the screen.
2. The 'Projects' screen is displayed.
3. Click on the '+' icon, to add a new project.



4. The 'New Project' screen is displayed.
5. Enter the values (described below) in the 'New Project' screen and press 'Save'.

6. The new project is created and the new projects 'Dashboard' screen is displayed.

2.1.1 + (New Project)

Press the Project '+' icon on the ProgramGenie 'Projects' screen to commence the process of creating a new project.



2.1.2 Image

Click on the 'Image' area on the 'New Project' screen to add an optional image to the project. Clicking on this will open a standard windows file search window. Select an image and press open, the image will be scaled to fit in the image box.

2.1.3 Name

Enter a name for the project in the 'Name' field on the 'New Project' screen. This field is mandatory, and cannot be a name already in use.

2.1.4 Lead

Select a name for the owner of the project from the 'Lead' dropdown list on the 'New Project' screen. Use the filter field on the dropdown to type a user name and filter the list. This field is mandatory.

2.1.5 Description

Enter an optional description of the project in the 'Description' field on the 'New Project' screen.

2.1.6 Users

Use the 'User dropdown list to select users who will have access to the project. The project Lead will have been added already and cannot be removed. Use the filter field on the dropdown to type a user name and filter the list.

2.1.7 New Connector

Press the 'New Connector' button on the 'New Project' Screen to configure a connector to integrate with external tools such as Jira, Confluence and various code repositories. The configuration of each connector will be described in the Connector Configuration section below. It is not mandatory to configure any connectors when creating a new project.

2.1.8 Exit New Project

Press the '<' icon to exit the 'New Project' screen without saving any of the entered project setup.

2.1.9 Save

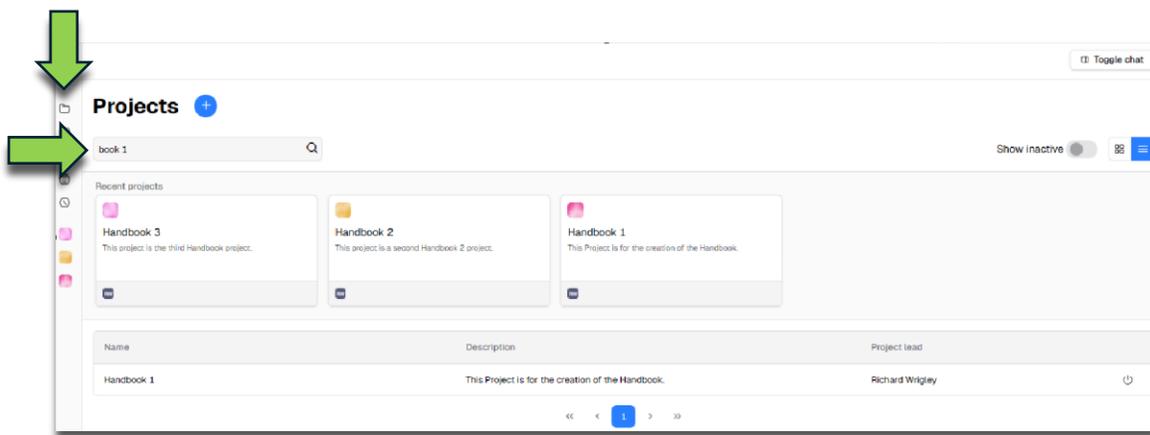
Press the 'Save' button to create the new project. The project will then open in the 'Dashboard' screen.



2.2 Search for a Project

A ProgramGenie environment can have a number of Projects created within it, allowing users to segregate work. To find a project:

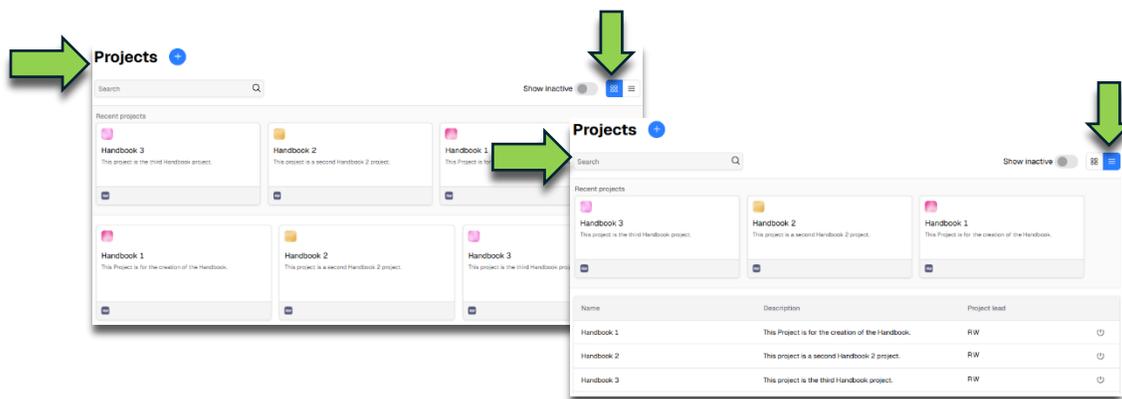
1. Click on 'Projects' within the 'ProgramGenie' sidebar on the left side of the screen.
2. The 'Projects' screen is displayed.
3. Enter the name of the project you want to find in the search field denoted by the spyglass icon.



4. The list of projects in the table below is filtered using the search text entered.
5. Click on the desired project name to open the project.

2.3 Show a Project as List or Tiles

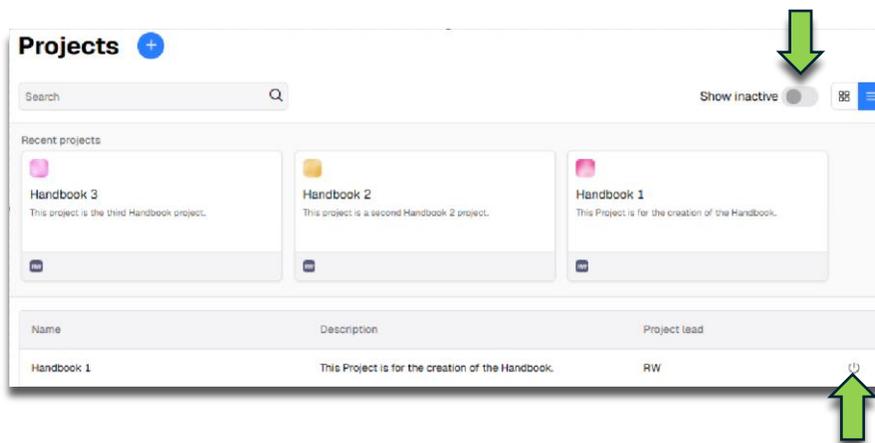
A ProgramGenie The list of available projects can be viewed in two ways, either as a list or as a number of tiles. Click on the list/tile icon on the top right to toggle the view.





2.4 Deactivate & Activate a Project

Projects are used in ProgramGenie to segregate work within an environment. When a project is no longer required it can be 'Deactivated'. A deactivated project is not shown within the projects table and cannot be searched for. Enabling the 'show inactive' switch allows 'Deactivated' projects to be seen and searched for in the projects table. A 'Deactivated' project can be 'Activated' again.



2.4.1 Show Inactive

Turn on the 'Show inactive' switch to allow 'Deactivated' projects to be displayed in the projects table. Turn off the 'Show inactive' switch to hide 'Deactivated' projects, displaying only 'Activated' projects.

2.4.2 Deactivate (dark standby icon)

Press the dark standby icon at the right hand end of an active project row in the project table to 'Deactivate' a project.

2.4.3 Activate (light standby icon)

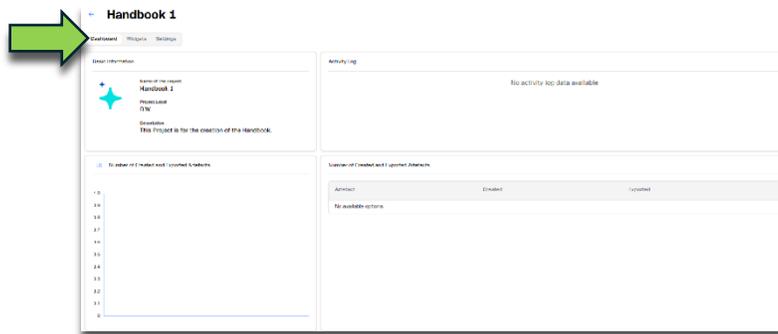
Press the light standby icon at the right hand end of a deactivated project row in the project table (Show inactive enabled) to 'Activate' a project.



3 Dashboard, Settings & Widgets

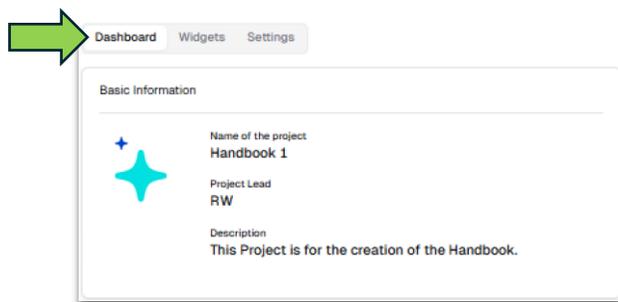
3.1 Dashboard Screen

The 'Dashboard' screen displays an overview of the current project. Several informational widgets are used and are described below.



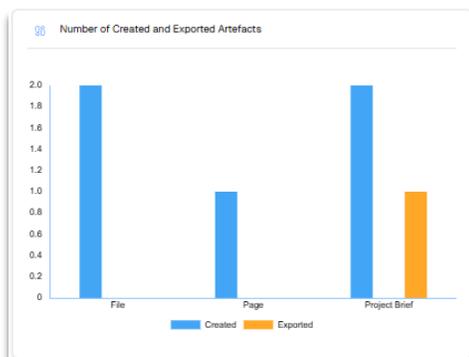
3.1.1 Basic Information

The 'Basic Information' widget displays the core description of the project as set when the project was created or updated in the settings screen.



3.1.2 Artefacts Information

There are two widgets displaying the amount of artefacts created and exported. The first displays information in a bar chart and the second in a table.



Artefact	Created	Exported
File	2	0
Page	1	0
Project Brief	2	1



3.1.3 Active Connectors

The 'Active Connectors' widget shows the current integration with external tools.

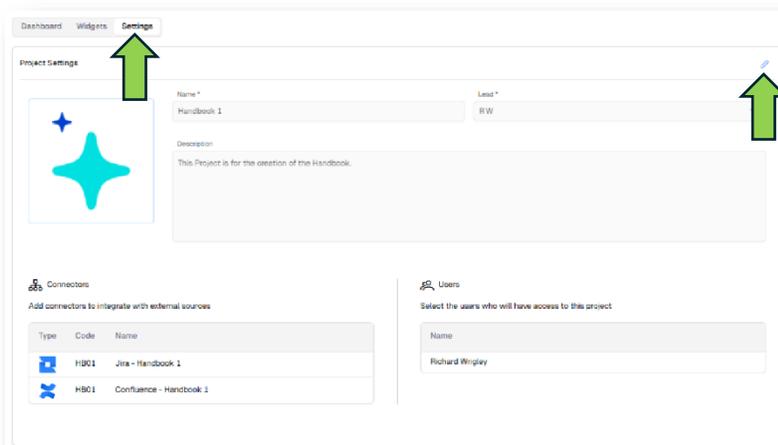
Type	Code	Name
	HB01	Jira - Handbook 1
	HB01	Confluence - Handbook 1

3.2 Settings Screen

The ProgramGenie 'Settings' screen shows information regarding the settings of the current project. The sections below describe each area.

3.2.1 Project Settings

The ProgramGenie 'Project Settings' area allows the user to edit the current project settings. Click on the 'Edit' (pen) icon to edit the values then press 'Save' to commit any changes.

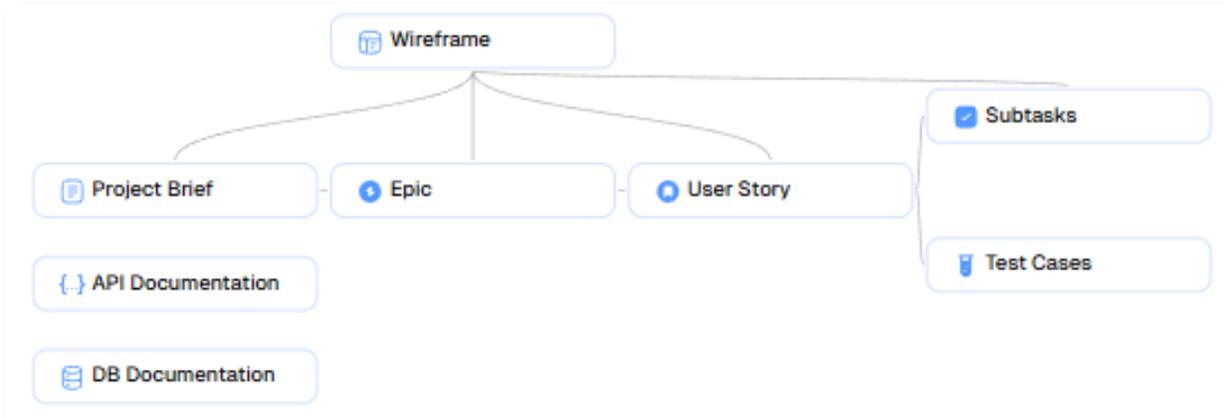


3.2.2 ProgramGenie Process Workflow

ProgramGenie uses a predefined 'Process Workflow' which defines how SDLC artefacts are derived and generated. Starting with uploaded 'Collateral' the user can generate a Project Brief, then from the project brief generate epics, from each epic generate user stories and from each user stories generate Subtasks & Test Cases. API & DB Documentation can also

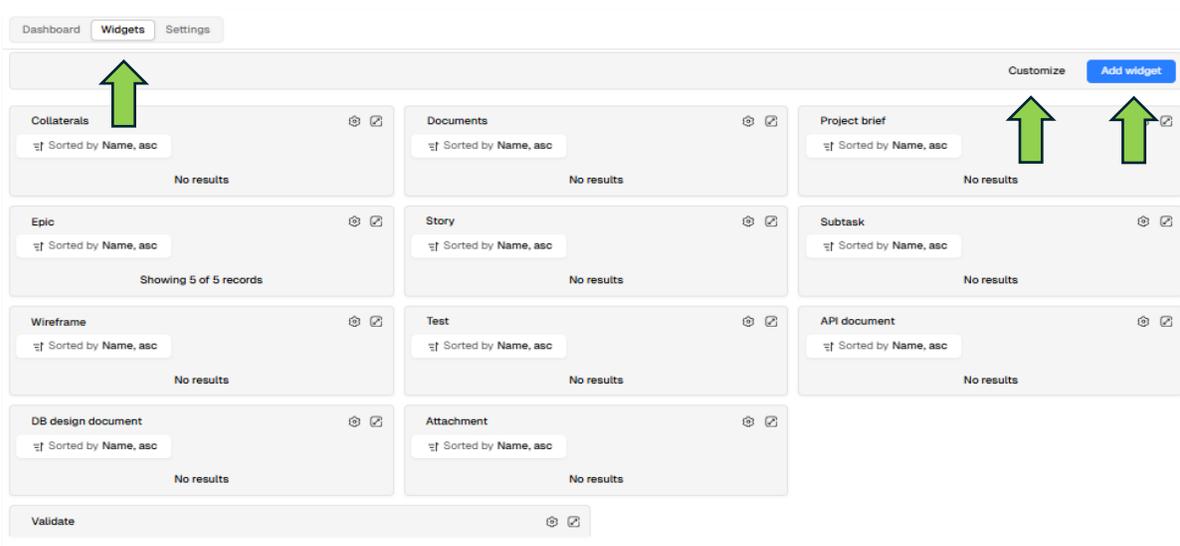


be generated from a Project Brief, also Wireframes can be generated from Epics, User Stories, Subtasks and Test Cases.



3.3 Widgets Screen

The Widgets screen is the key screen where users interact with their ingested documents, ProgramGenie generated artefacts for example, Project Briefs, Epics, Stories, Test Cases and integrated repositories etc. The screen consists of a number of display Widgets. Each display Widget can be minimized or maximized to help manage screen space, the order of the display Widgets can also be rearranged. A user can also create their own display Widgets selected content using the Add Widget button.





3.3.1 Customise

Press the customize button to display the View Settings menu. This menu displays all the widgets available to be displayed in the Widget area.

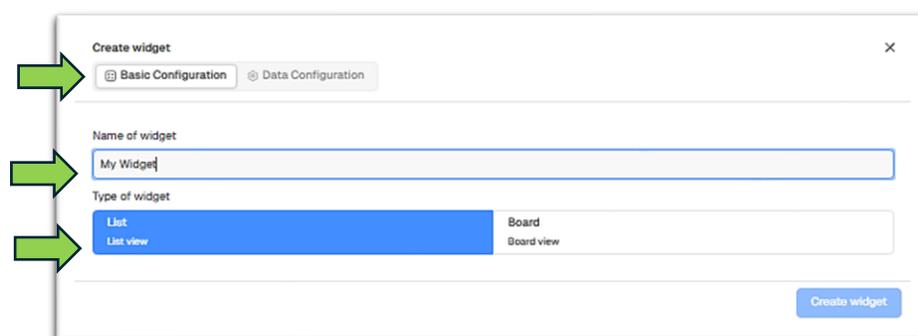


The Eye icon next to the Widget name show the display status for the Widget, click on the icon toggles the on/off status. Click on the Trash icon to permanently delete a custom widget, note Default Widgets cannot be deleted. Drop and drag menu items to change the display order of them in the Widget tab

3.3.2 Add Card

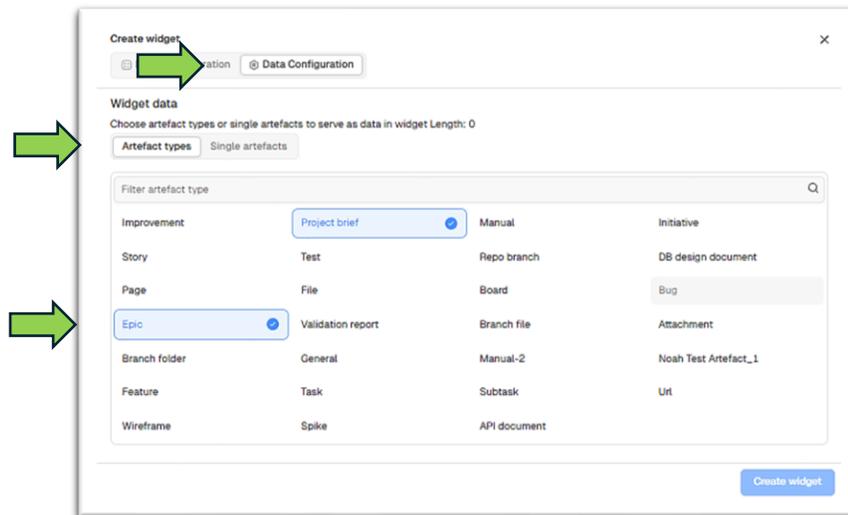
Click on the Add Card widget on the Widget tab to create your own custom widget.

1. Goto the 'Widgets' screen.
2. Click on the 'Add Card' button.
3. Observe that the Create Widget screen is displayed, defaulted to the Basic Configuration screen

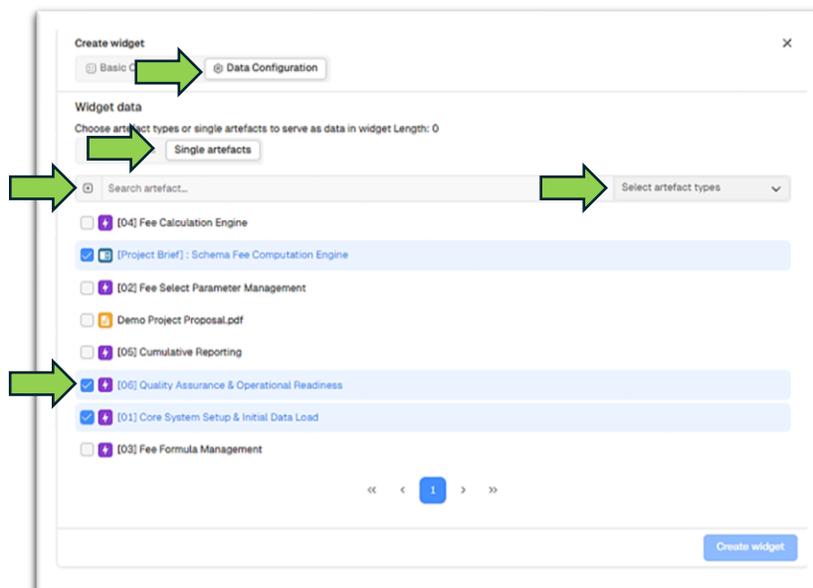




4. On the Basic Configuration screen enter a name for the widget.
5. Select the Type of Widget, either List or Board (hierarchy of linked items).
6. Select the Data Configuration Tab.
7. There are two options available either to Select 'Artefact Types', where groups of types of artefacts can be shown in the custom widget.

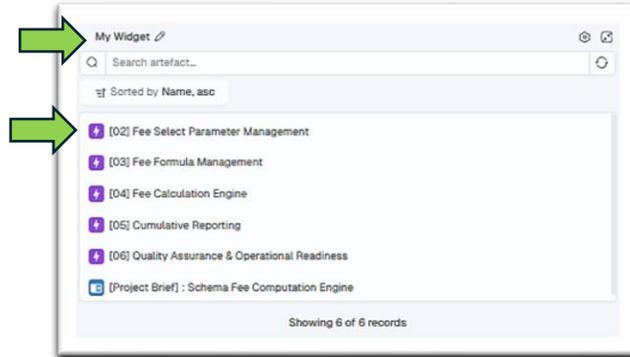


8. Or to select 'Single Artefacts' where individual artefacts can be selected for display in the custom widget.
9. There is a search function and an artefact filter to help in finding the desired artefacts.





10. Press the Create Widget button to create the new custom widget in the Widget tab



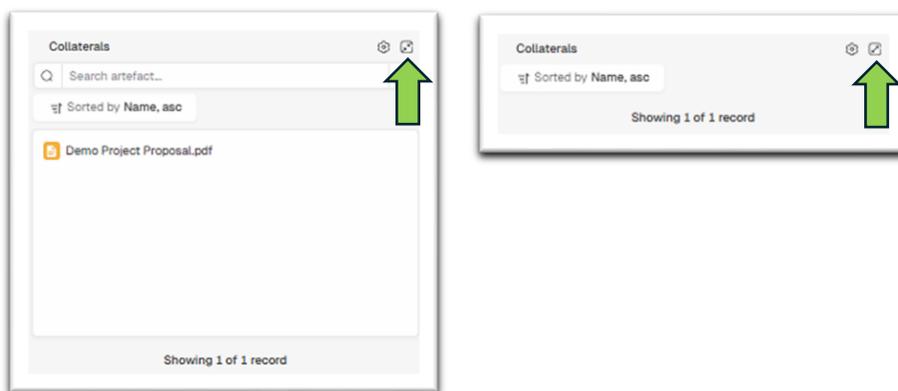
11. Click on the pen icon to edit the configuration of the custom widget.



3.4 Widgets actions

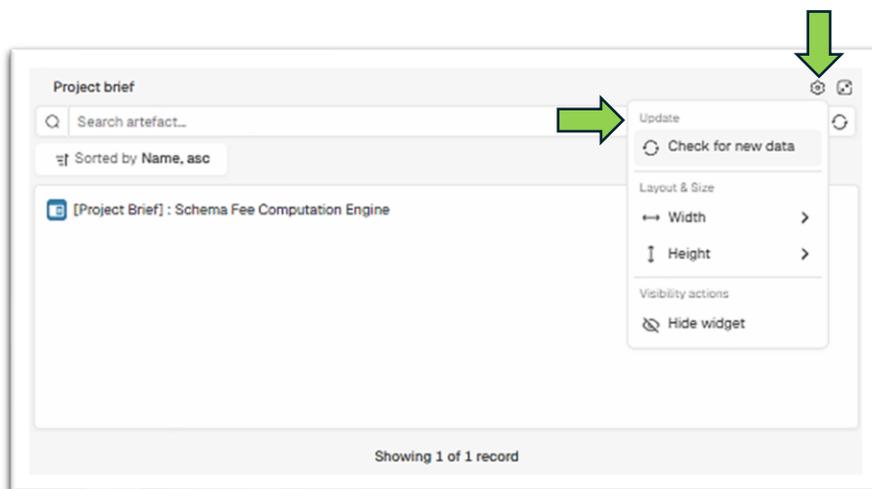
3.4.1 Minimize & Maximize Widget

Click on the Minimize/Maximize icon, in the top right corner of a display Widget to minimize or maximize the widget



3.4.2 Edit Widget

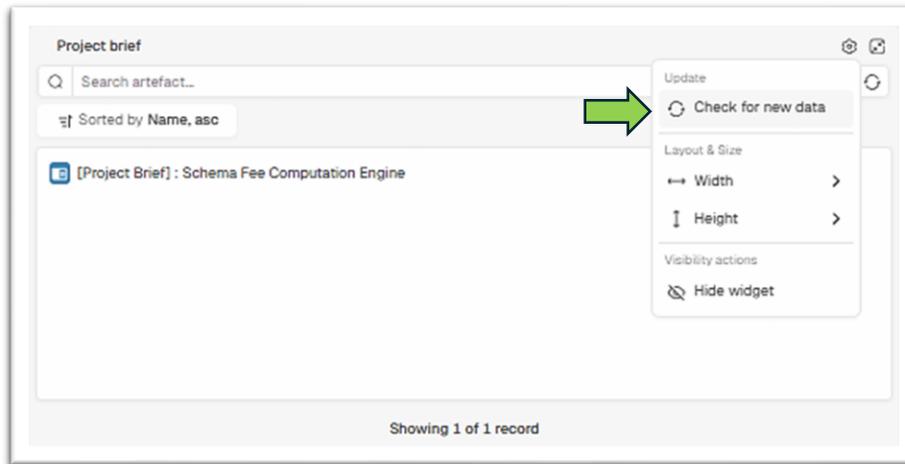
Click on the 'Edit' (cog) icon in the top right corner of the widgets screen to display the Edit Widget menu





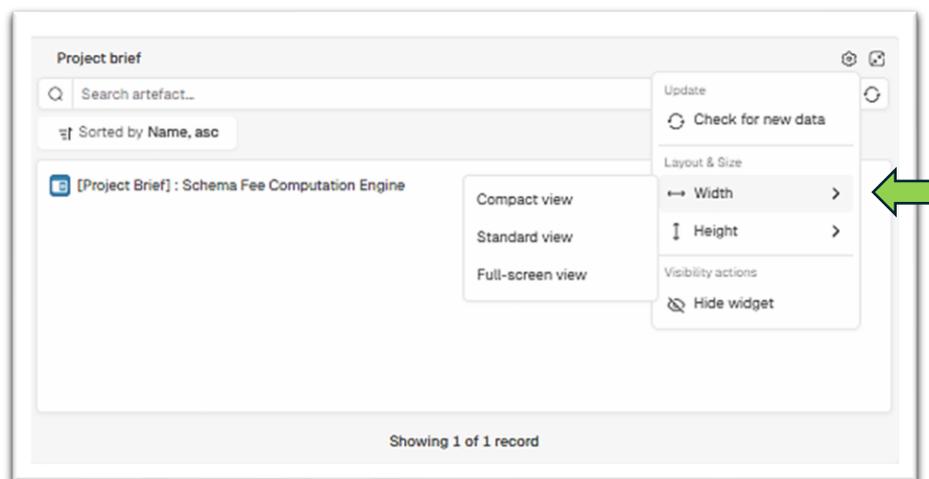
3.4.3 Check for new data

Check for new data functionality refreshes data on the widget



3.4.4 Width

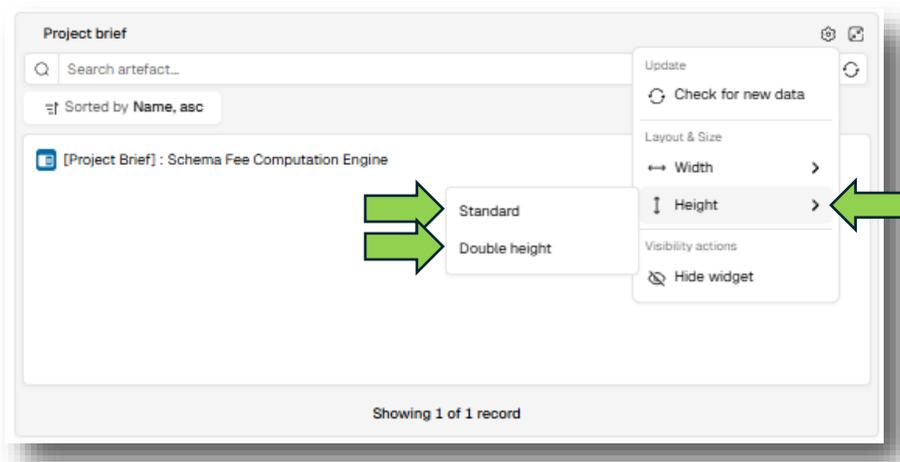
Select the Width menu item to change the horizontal size of the Widget. There are three options, Compact view which uses 1/3 of the horizontal screen size, Standard view which takes up 1/2 the horizontal screen size and finally Full-Screen View which takes the total width of the screen size.





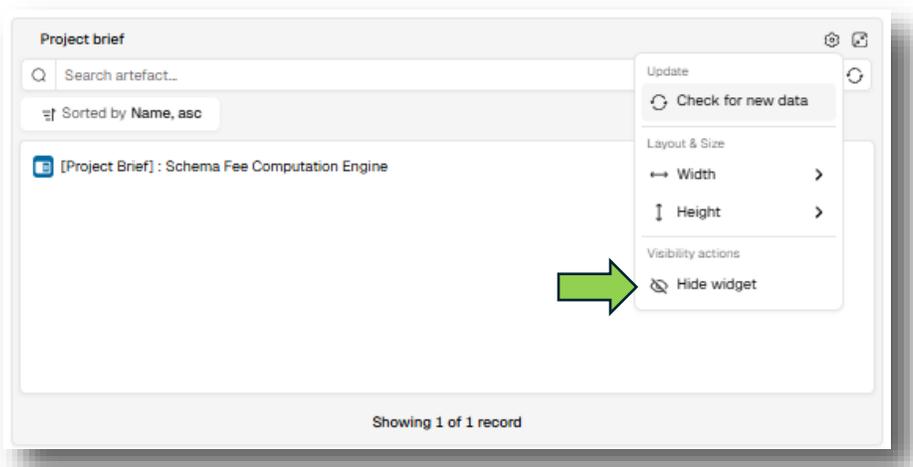
3.4.5 Height

Select the Height menu to change the vertical size of the Widget. There are two options, Standard shows 6 rows of artefacts (before a scroll bar is required) and Double Height shows 12 rows of artefacts (before a scroll bar is required).



3.4.6 Hide Widget

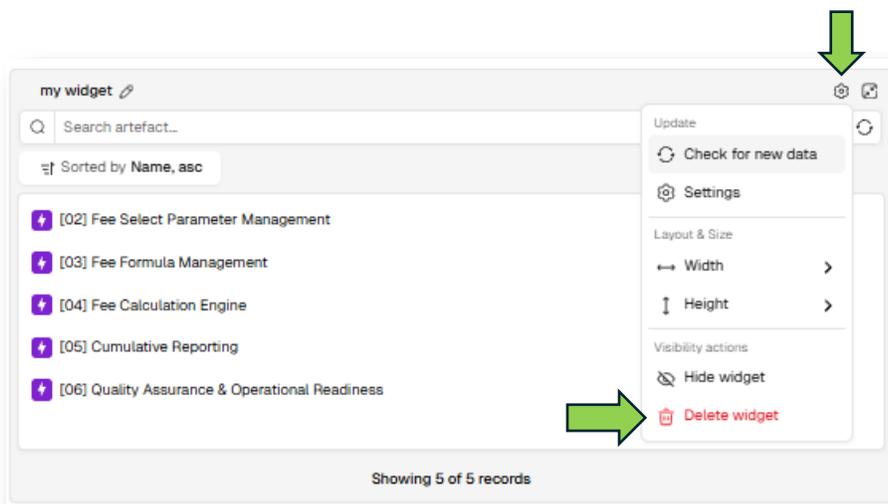
Click on Hide widget to collapse the widget. All widgets on that horizontal row of the dashboard must be hidden to recover the horizontal space from the dashboard.





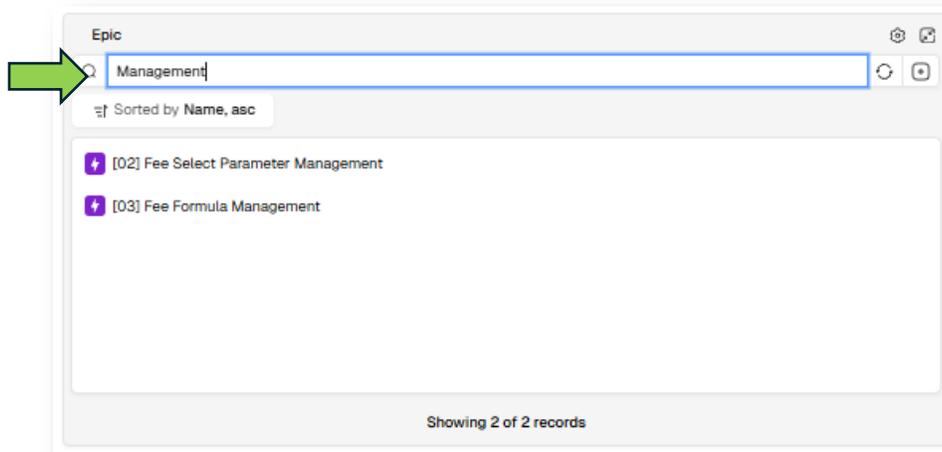
3.4.7 Delete Widget

The Delete Widget function is only available on users own custom Widgets. Select Delete and the widget will be permanently removed from the dashboard. Remember this is only a view and does not effect any of the artefacts.



3.5 Search and sort Widget

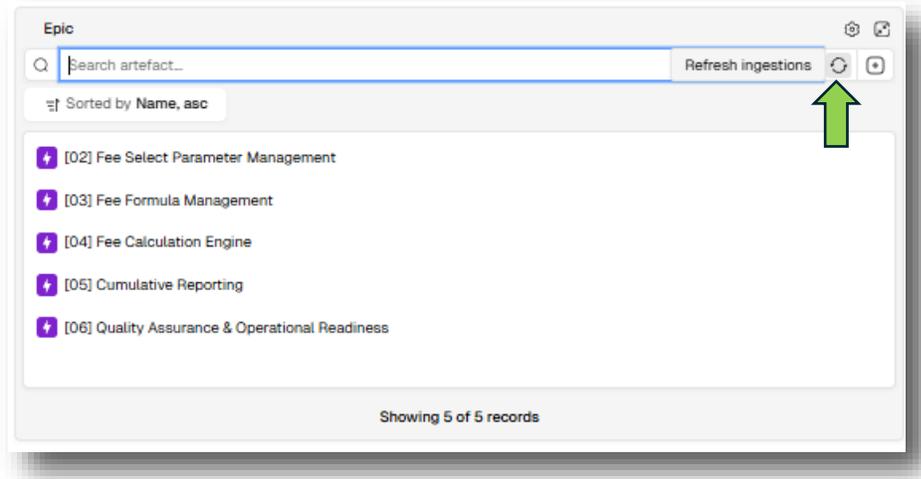
Enter a search string into the search field in the list options area of the widget. Entering text in this area filters the artefacts displayed on the widget. Clear the text in the search field to clear the filter.



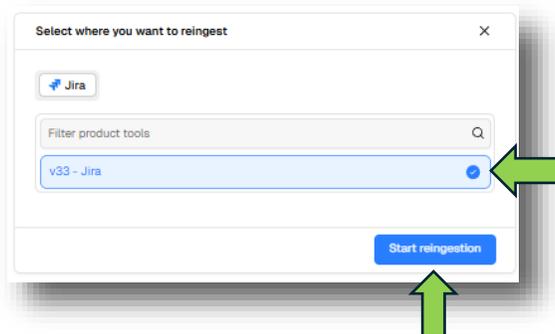


3.6 Refresh Ingestions

Click the refresh icon to ensure that exported artefacts displayed in the widget are synchronised with the integrated external repository (for example, Jira).



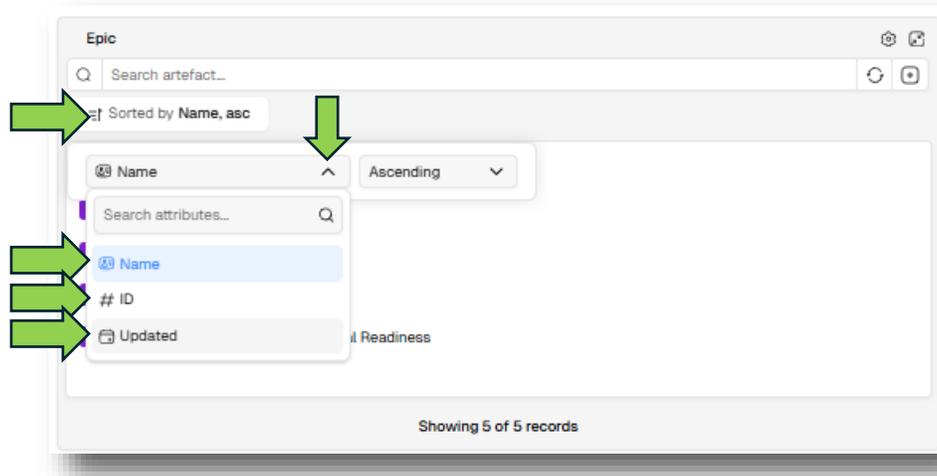
Clicking on the Refresh Icon displays the 'Reingest Selection' screen. Select the repository to synchronise with and press Start Reingestion to synchronise the widget.



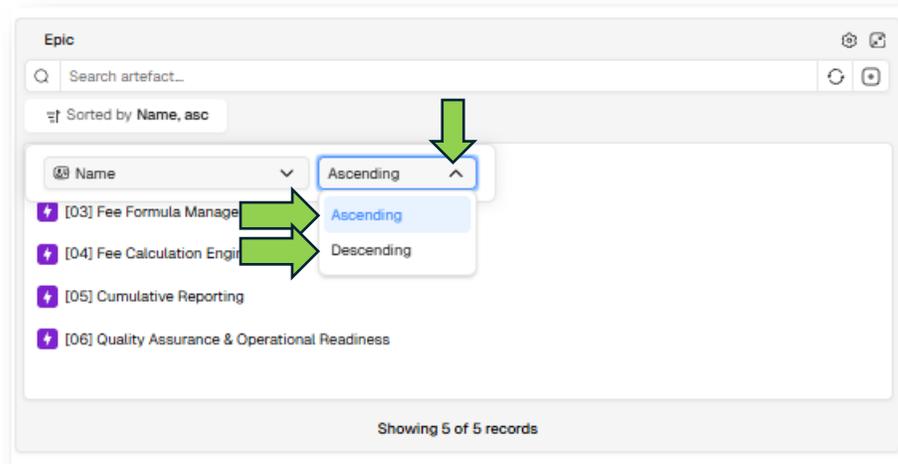


3.7 Sorting

Click on the Sorted by icon to open the Sort menu. The Sort Menu has two sub menus, Name and Ascending/Descending. Click on the Name menu to display a menu with three options. Name sorts the widget by artefact name, ID sorts the widget by the underlying ID for the artefact and Updated sorts the widget based on the updated date/time.



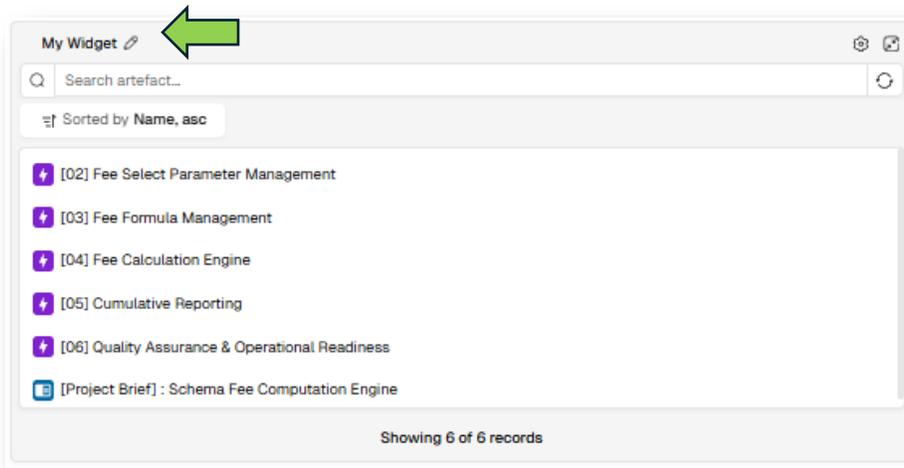
Click on Ascending/Descending to show a menu to select if the sort should be performed in Ascending or Descending order.





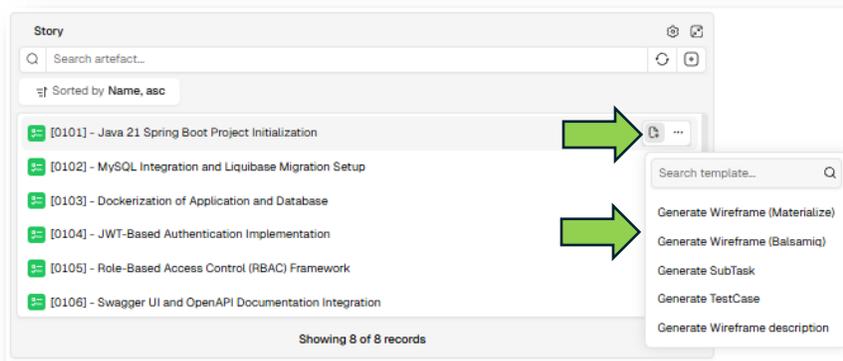
3.8 Edit Custom Widget

The Edit Widget (pen) icon is only available on user created Custom Widgets, click on this icon to change the preference created for the widget when it was created.



3.9 Widget Artefact Action Menu

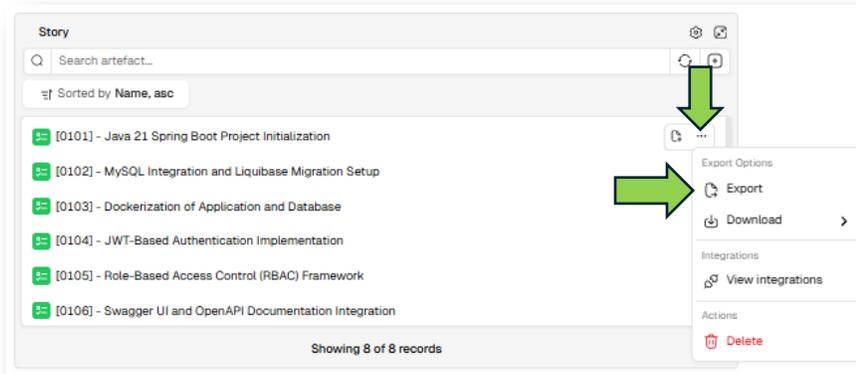
Click on the Artefact Action Menu icon next to an artefact to activate an action on that artefact, these are specific to the type of artefact and are described later in this document.





3.10 Widget Artefact Function Menu

Click on the Artefact Function Menu icon (...) next to an artefact to display a menu displaying the artefact options.

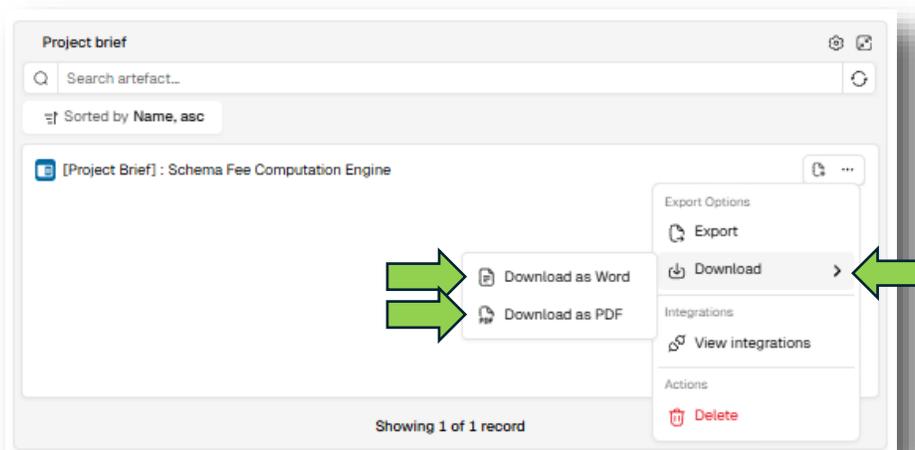


3.10.1 Export

Click on the Export menu item to export the artefact to a pre-configured integration such as confluence or jira, this is covered later in this document.

3.10.2 Download (Word/PDF)

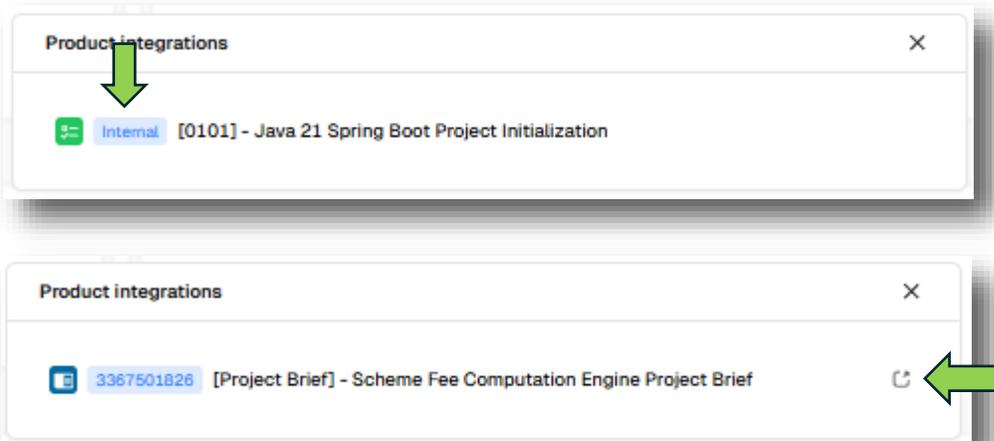
Click on the Download the artefact in either Word or PDF format. This function can also be invoked from by clicking on an artefact and selecting download on the screen displayed.





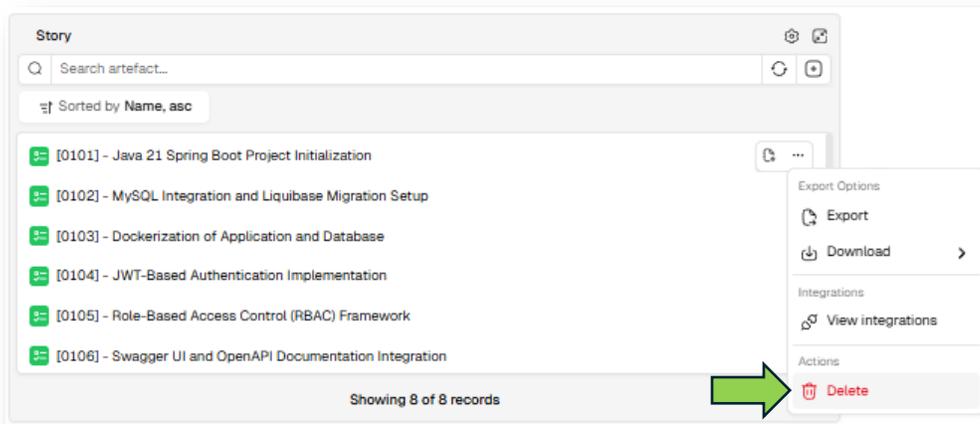
3.10.3 View Integrations

Click on the View Integrations menu item to display the Integrations window. If the Artefact has not been exported then the artefact will be marked as Internal. If the artefact has been exported, the window will display the Goto icon. Click on the Goto Icon to open the artefact with the exported application.



3.10.4 Delete

Click on the Delete Icon to permanently delete the artefact.





3.11 Default Widgets

The Widgets dashboard comes preconfigured with a number of useful display widgets, see below. There are a number of custom bespoke widgets that Clients can request their environment to be configured with.

3.11.1 Collaterals Widget

The Collaterals widget shows files and other types of artefact that have been uploaded, for example, Word Documents, PDF's, Images, etc.

3.11.2 Documents Widget

The Documents widget displays all 'documents' format artefacts that have been generated by ProgramGenie. For example Project Briefs, Wireframe Description Documents, etc.

3.11.3 Epic Widget

The Epic widget displays Epics that have been generated by ProgramGenie or any external Epics that originate in an external Epic Management tool that has been integrated with ProgramGenie via a connector, for example Jira or ADO.

3.11.4 Story Widget

The Story widget displays Stories that have been generated by ProgramGenie or any external Stories that originate in an external Story Management tool that has been integrated with ProgramGenie via a connector, for example Jira or ADO.

3.11.5 SubTask Widget

The SubTask widget displays Sub-Tasks that have been generated by ProgramGenie or any external Sub-Tasks that originate in an external Sub-Task Management tool that has been integrated with ProgramGenie via a connector, for example, Jira or ADO.

3.11.6 Test Widget

The Test widget displays Test Cases that have been generated by ProgramGenie.

3.11.7 Wireframes Widget

The Wireframes widget displays Wireframes that have been generated by ProgramGenie, these can be in a number of formats, for example Balsamiq or Materialize.



3.11.8 Code Repository Widget

Code Repository widget displays a code repository that has been ingested in ProgramGenie via connectors for GitHub/GitLab/BitBucket or a similar code repository tool.

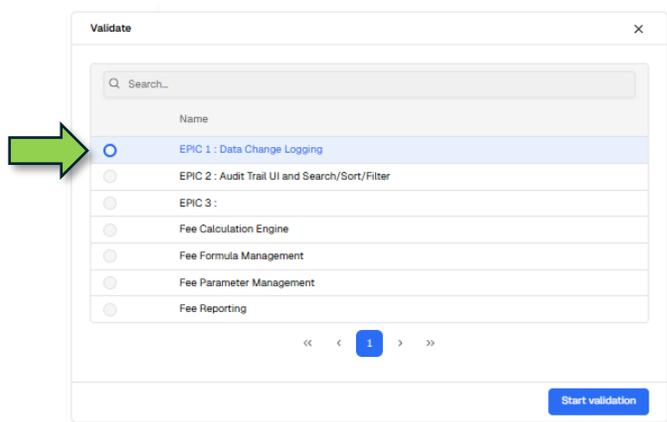
3.11.9 Validate Widget

The 'Validate' widget is used for analyzing existing artefacts versus a set of collateral, with the objective of creating an assessment of completeness. In practice this is best used for legacy or enhancement projects where many of the artefacts were not directly generated from the collateral using ProgramGenie. Using the Validate widget on a new build project where all artefacts were generated by ProgramGenie will result in an outcome that gives an alternative view to the currently generated set of artefacts, rather than a true validation. There are four validation activities available, Validate Project Brief, Validate Epic, Validate Story and Validate Test Cases.

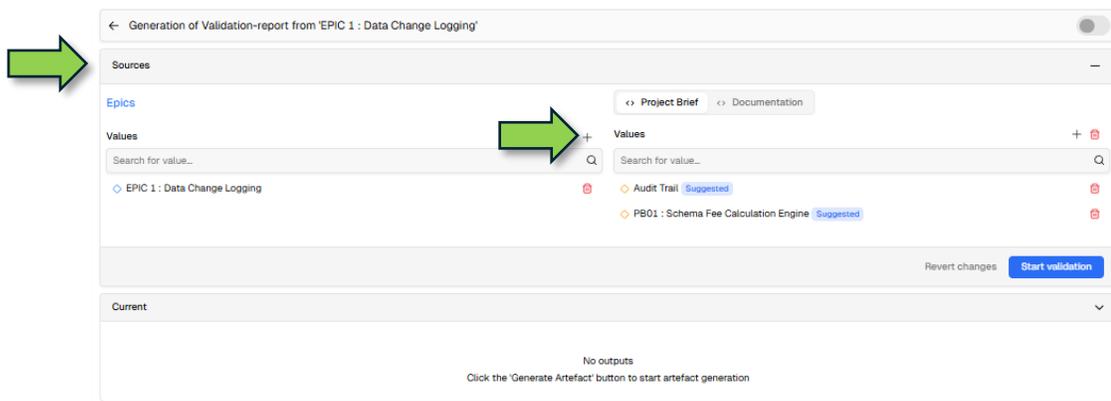
1. Goto the 'Widgets' screen.
2. Click on 'Validate Epic' in the 'Validate' widget.



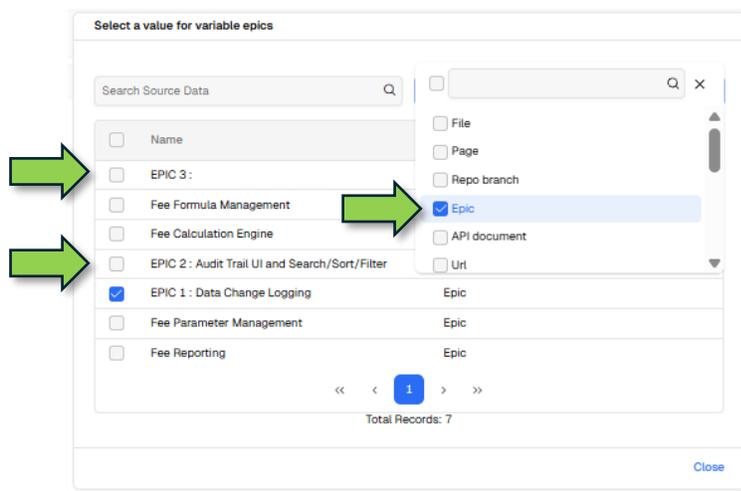
3. Observe the validation window appears.



4. Click on an epic and press start validation
5. Observe the Generation of Validation Report window appears
6. Click on the add values icon to add more Epics to the validation

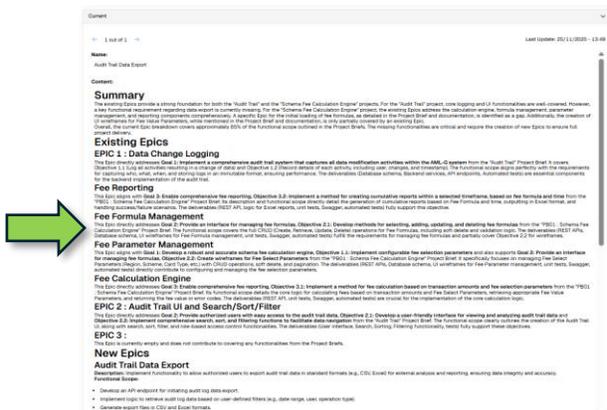


7. Select Epic from the Artifact Type dropdown



8. Select all the Epics

9. Press Start Validation and confirm



10. After a short time the Validation report is output.

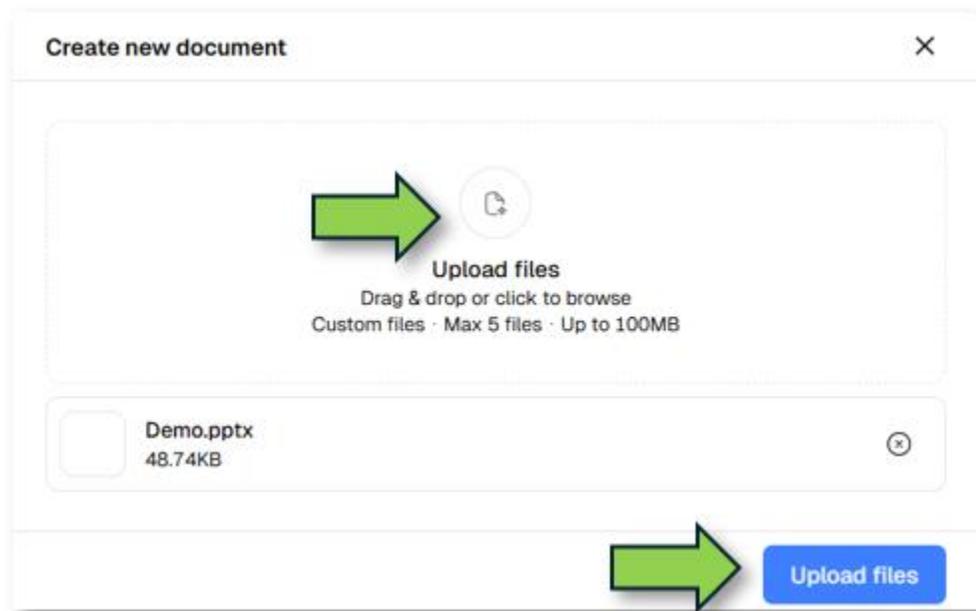
11. Press the save button

12. The document is saved in the documents widget



4 Uploading Collateral

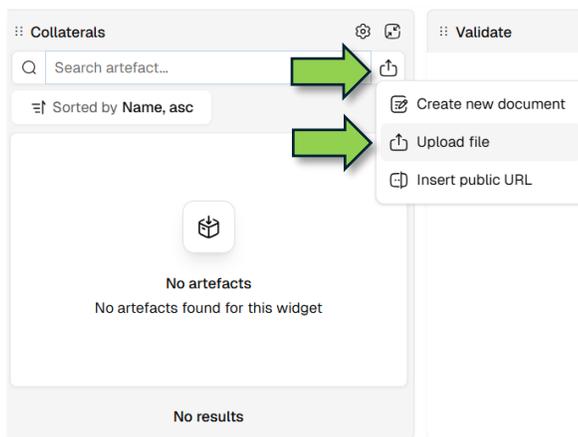
Collateral, in ProgramGenie terms, are the foundational input artefacts that seed the generation of new ProgramGenie artefacts. There are a three core input artefact mechanisms, 'Documents' covering files of various formats, 'Public URL' covering internet content and 'Write Text' allowing users to input some text to ProgramGenie and upload it as a text file input.



The 'Insert

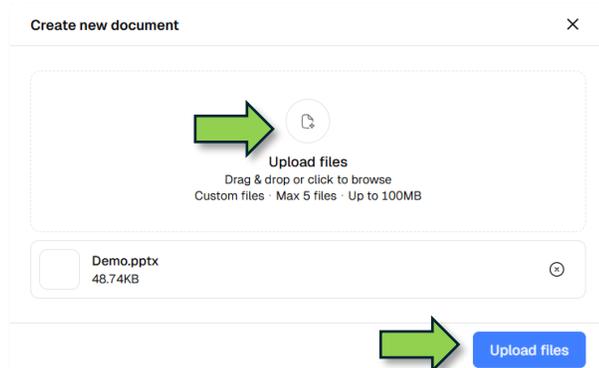
Documents' function allows a user to upload a file containing project information to be used as a source of data when generating artefacts.

13. Goto the 'Widgets' screen.
14. Click on 'Upload' icon on the 'Collaterals' widget.
15. Select 'Upload file' from the dropdown.





16. The 'Upload files' screen is displayed.



17. Drag and drop a file into the documents area

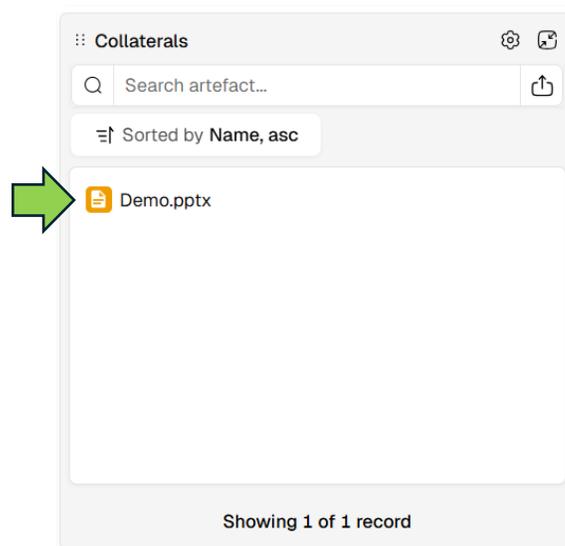
18. Or click on the file icon to open the standard windows file selection screen.

19. Added files are displayed in the file area.

20. Multiple files can be added, the trash icon can be used to remove all selected files, or the 'X' icon to remove individual files.

21. Press the 'Upload files' button.

22. The uploaded documents are displayed in the 'Collaterals' widget (this may take several minutes for large documents or multiple files).





23. If the uploads do not appear once the processing has completed, press the page refresh for the browser.

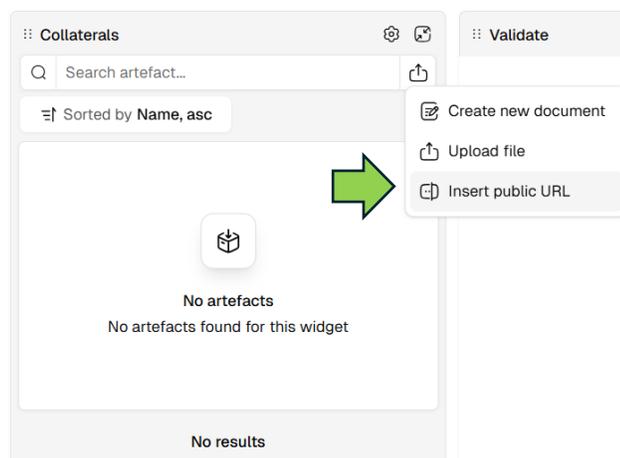
The 'Upload file' function can upload office suite documents, simple text files, e-mails, text files, Images, PDFs, audio and video recordings.

Important: Restrictions are set to 5 files up to 100MB each in one go.

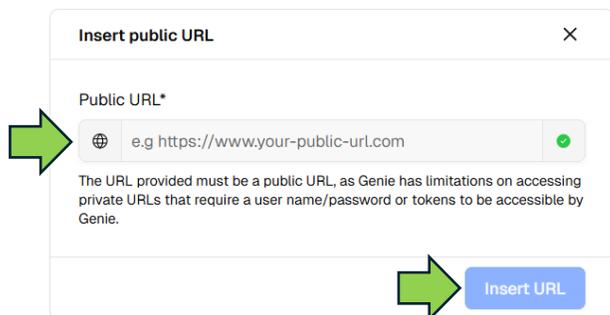
4.1 Public URL

ProgramGenie can upload information from the internet contained in 'Public URLs'.

1. Goto the 'Widgets' screen.
2. Click on 'Upload' icon on the 'Collaterals' widget.
3. Select 'Insert public URL' from the dropdown.



4. The 'Insert Public URL' screen is displayed.
5. Enter a valid URL (including the https://).

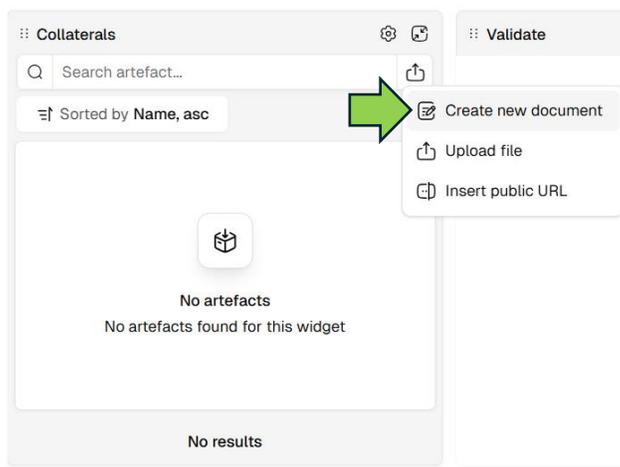


6. Press the 'Insert public URL' button.
7. The uploaded documents are displayed in the 'Collaterals' widget.

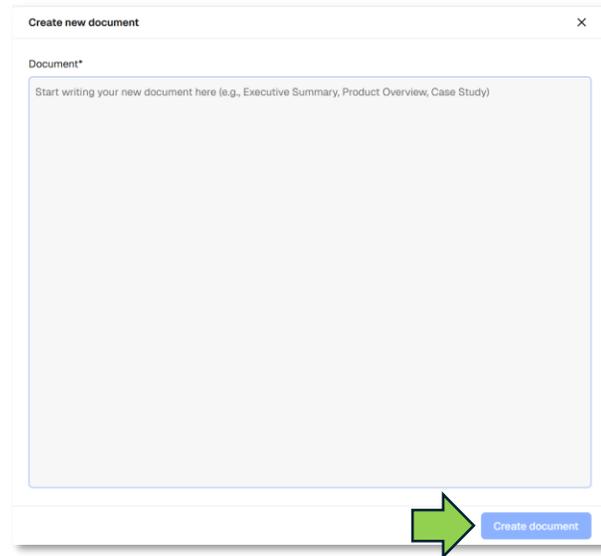
4.2 Create new document

ProgramGenie can upload information typed directly into ProgramGenie.

1. Goto the 'Widgets' screen.
2. Click on 'Upload Collateral' upload icon on the 'Collateral' widget.
3. Select 'Create new document' from the dropdown.



4. The 'Create new document' screen is displayed.
5. Enter some extra project information in the text area.



6. Press the 'Create new document' button.
7. The created document is displayed in the 'Collaterals' widget. If the uploads do not appear once the processing has completed, press the page refresh for the browser.



5 Generating Artefacts

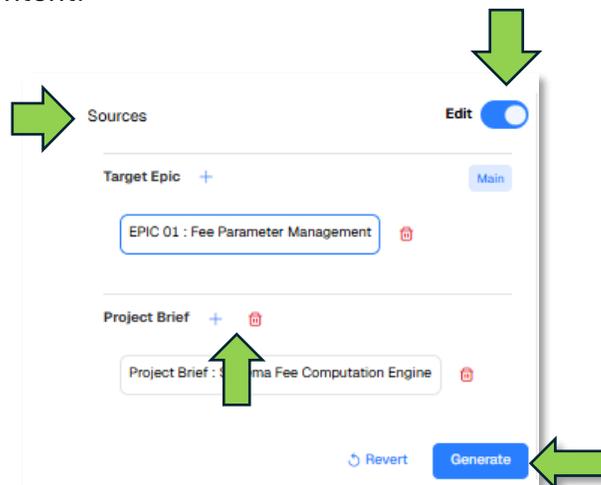
The 'Generation' screen provides a key function within ProgramGenie, it is displayed whenever a generation action is invoked. The 'Generation' screen has two areas, the left side of the screen is the 'Sources' area where seed documents and artefacts used in the ProgramGenie AI based artefact generation are edited and displayed. The right side of the screen displays the generated output, allowing the user to browse, edit and save generated artefacts.

5.1 Generation Screen – Sources Area

The left side of the 'Generation' screen displays a list of the source documents and artefacts that will be used to generate artefacts. When a generation request is invoked from an artefact for the first time, ProgramGenie AI suggests a list of potential sources for the generation. The user can then edit this list removing or adding sources as required.

5.1.1 Edit

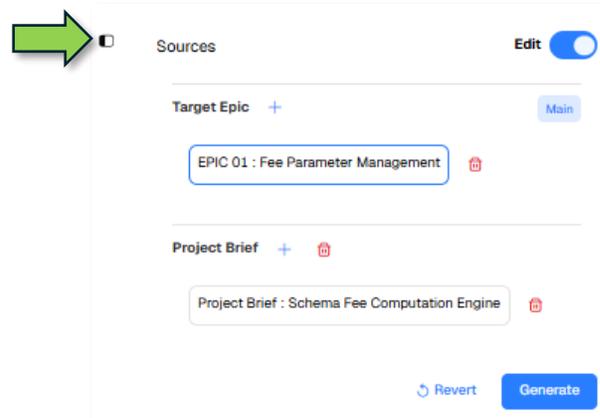
Enable the edit button to enter edit mode. Use the Add (+) and the Delete (trash) icons to change the source content.





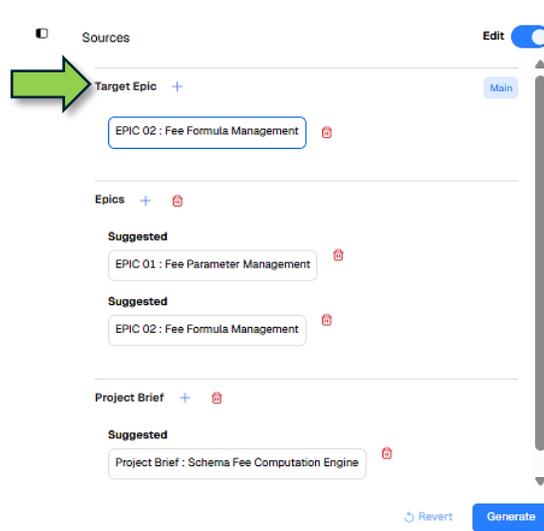
5.1.2 Sources Toggle

Press Toggle button on the upper left of the sources area to hide the whole sources area.



5.1.3 Target

The 'Target' source, is a special feature that is available on certain generation actions. It allows the user to change the 'Target' parent artefact for the generation. When you invoke a generate action from an artefact, it will default the 'Target' parent artefact to the artefact that the action was started from, however, this can be changed to point at a different artefact of the same type. For example, you can start the generation of stories from Epic 1 but then change the Target to Epic 2, and when the stories are saved they will have Epic 2 as their parent.





5.1.4 Add Source (+)

Press the 'Add' (+) icon to open the source selection window. Select new sources from the screen and press the 'Ok' button. This will add the selected sources to the sources list displayed.

5.1.5 Delete Source (trash)

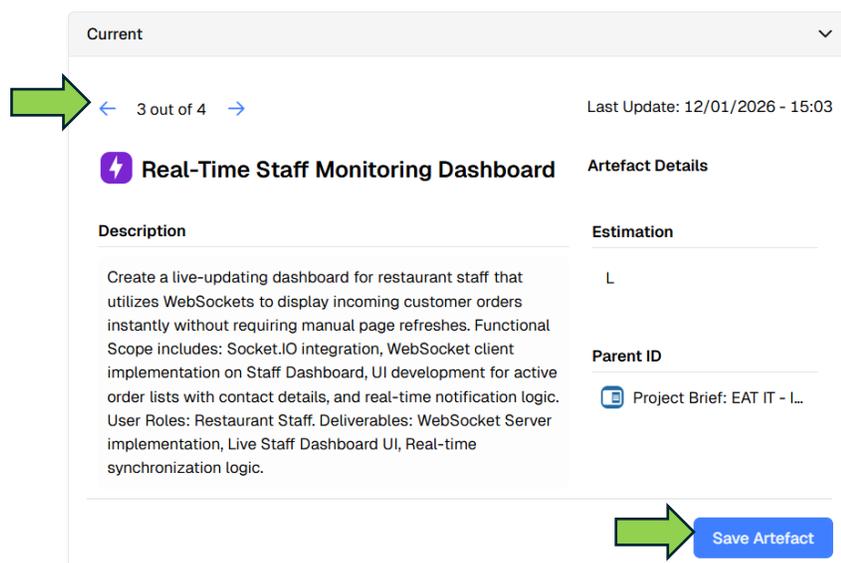
Press the 'Delete' (trash) icon next to a source to remove it from the target and sources lists.

5.1.6 Generate

Press the 'Generate' button to start the generation process, based on the list of sources.

5.2 Generation Screen – Output Area

When the generate process completes it displays the output on the right of the 'Generation' screen. Some generation actions result in one output for example creating a Project Brief' from an uploaded Project Proposal, whereas others will result in multiple outputs, for example generating User Stories from an Epic. Multiple outputs can be stepped through with the output displayed one at a time on display area, the content of each can be edited, reviewed and edits saved. At this point the generated output is only in the generate window, the user can then select which of the output artefacts they want to 'Save Artefact' to the relevant ProgramGenie widget (ie. Save in ProgramGenie).





5.2.1 n Out Of n (next/previous)

The 'n Out Of n' indicator shows how many outputs have been generated and the related icons allow the user to move to the next or previous output.

5.2.2 Edit

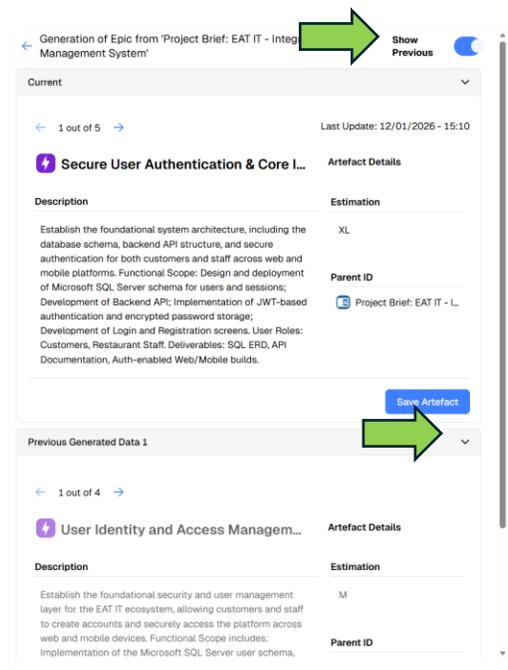
Click on artefact text to edit it. The edit mechanism will be dependent on the type of output produced for example Epics are edited in a text editor whereas Wireframes are edited using an Wireframe editor.

5.2.3 Save Artefact

Press 'Save Artefact' to save the generated output to ProgramGenie as an artefact on the 'Widgets' screen.

5.3 Generate Multiple Versions

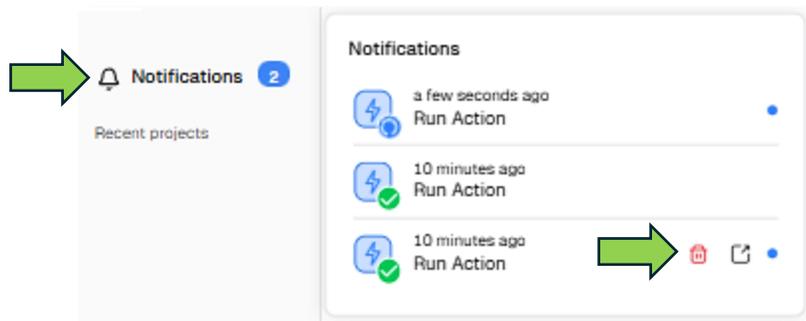
Each time the same generation action is run on the same artefact a new version of the output is created. Enabling the 'Show Previous' switch on the top right of the 'Generation' screen shows all previous generation runs and related outputs and can be scrolled down to and interacted with to create new artefacts.





5.4 Notifications

The 'Notifications' popup window displays the status of actions in ProgramGenie, for example the request to generate a wireframe. The screen is accessed from the ProgramGenie sidebar. Clicking on the Notification item in the sidebar opens the Notifications popup showing a list of colour coded recent tasks (Green=complete, Blue=running & Red=Error).



5.4.1 Delete

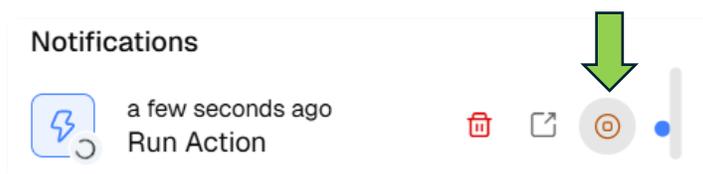
Click on the 'Delete' (trash) icon to remove the action from the list.

5.4.2 Open

Click on the 'Open' icon to open the artefact that was created as a result of the action. This action is only available to (green) completed actions.

5.4.3 Stop

When some action is in process state, stop option will be available. This option will stop process that is currently running.



5.4.4 Blue Dot

The 'Blue Dot' is a simple indicator that a user can use to recognize that they have dealt with a notification, which is especially useful when a number of actions have been invoked. All Notifications are given a blue dot, the user can click on the dot to acknowledge and remove it.



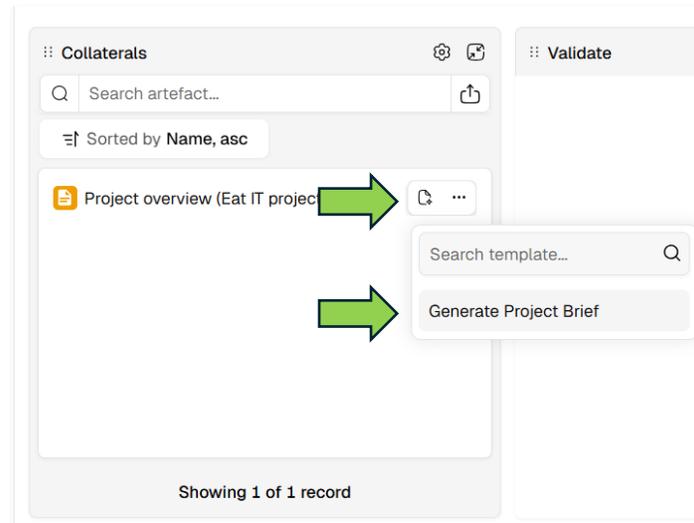
5.5 Generate a Project Brief

A key output of the ProgramGenie process is the production of a 'Project Brief' document, which provides a detailed overview of the project with respect to some of the following:

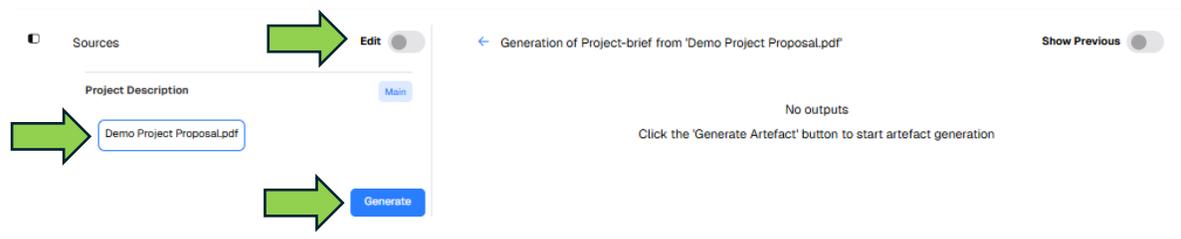
- Project Overview
- Background Information
- Goals and Objectives
- Functional & Non-Functional Requirements and Specifications
- Constraints and Assumptions
- Scope and Out of Scope
- Deliverables (Software, Documentation, Timeline and Key Milestones)
- Budget & Resource Requirements (Development Team & Infrastructure)
- Quality Standards and Metrics
- Risk Assessment & Mitigation
- Project Team & Project Governance
- Communication and Reporting
- KPI's
- Sustainability
- Innovation and Creativity
- Regulatory Compliance
- Training and Knowledge Transfer
- Closure and Handover

A brief document is generated using proprietary AI mechanisms from uploaded 'Collateral' source data.

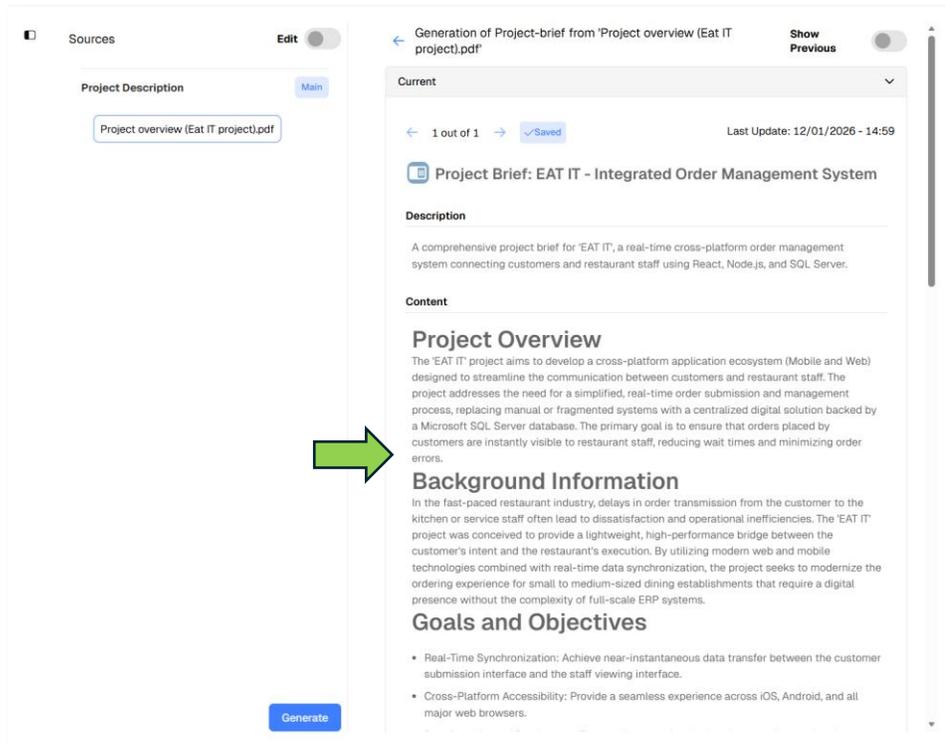
1. Go to the 'Widgets' screen.
2. Go to the 'Collaterals' widget.
3. Select the 'Generate Artefact' icon to the right of an uploaded project proposal.
4. Select 'Generate Project Brief' from the dropdown.



5. The 'Generate' screen is displayed.
6. Observe that the name of the uploaded Collateral document that the generation was activated from is shown as the 'Project Description'. This is the source document that will be used to seed the AI generation. More source documents can be added by enabling the 'edit' switch and pressing the '+' icon and selecting additional source documents (from the Collateral widget).



7. Press the 'Generate' Button.
8. Once the Project Brief has been generated it is displayed in the right hand side of the 'Generation Screen'.



9. At this point there is an option to edit the 'Project Brief' if required by clicking on text
10. Once any edits have been committed press the 'Save' button.
11. Press the '<' icon to return back to the widgets screen. Observe that the newly generated and saved 'Project Brief' is displayed in the 'Documents' widget.
12. Click on the 'Project Brief' in the 'Documents' widget.
13. The 'Project Brief' is displayed in a text window.

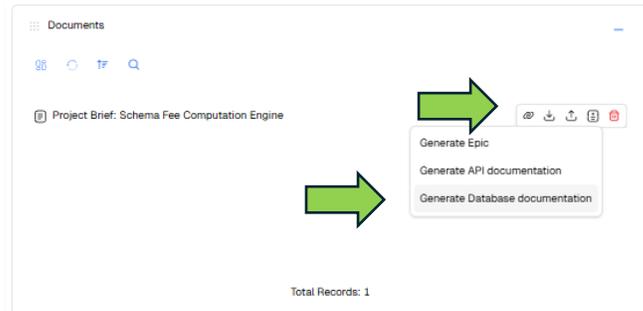
5.6 Generate Epics

A 'Project Brief' in the 'Documents' widget can be used to generate 'Epics' using proprietary AI mechanisms. An Epic is a large body of work that encompasses a collection of smaller, related user stories, bugs, or tasks, allowing teams to manage and track significant initiatives over multiple sprints or releases. Epics provide a high-level view of a project's objectives, breaking down a large goal into more manageable components that can be completed by the team, ultimately contributing to the epic's overall completion.

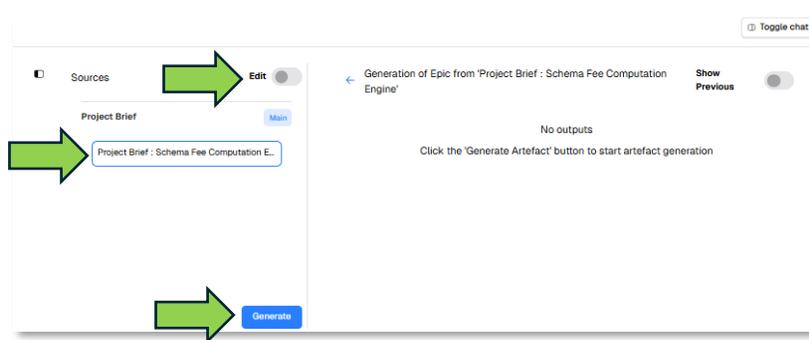
1. Go to the 'Widgets' screen.
2. Go to the 'Documents' widget.
3. Select the 'Generate Artefact' icon to the right of an generated 'Project Brief'.



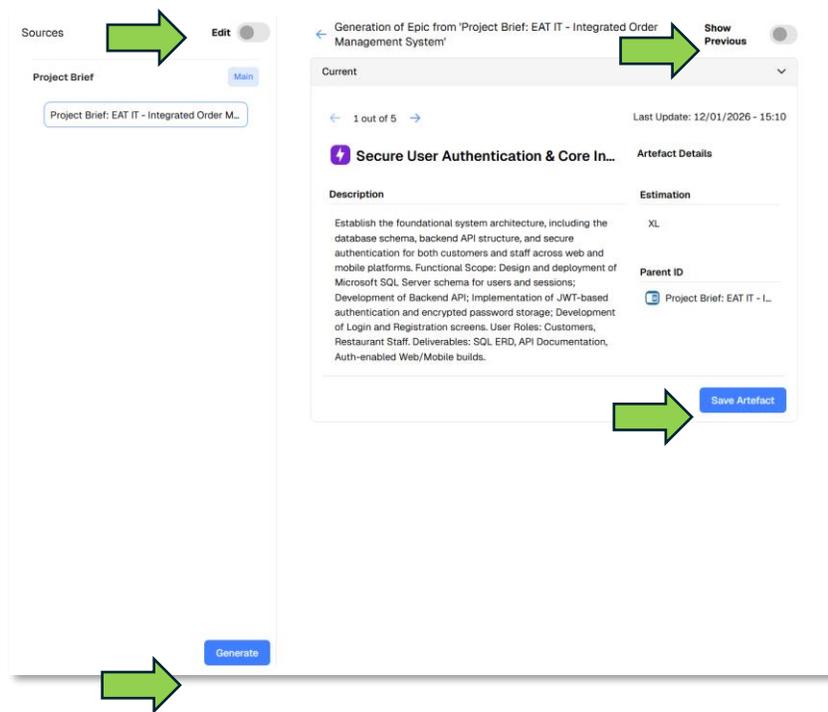
4. Select 'Generate Epics' from the dropdown.



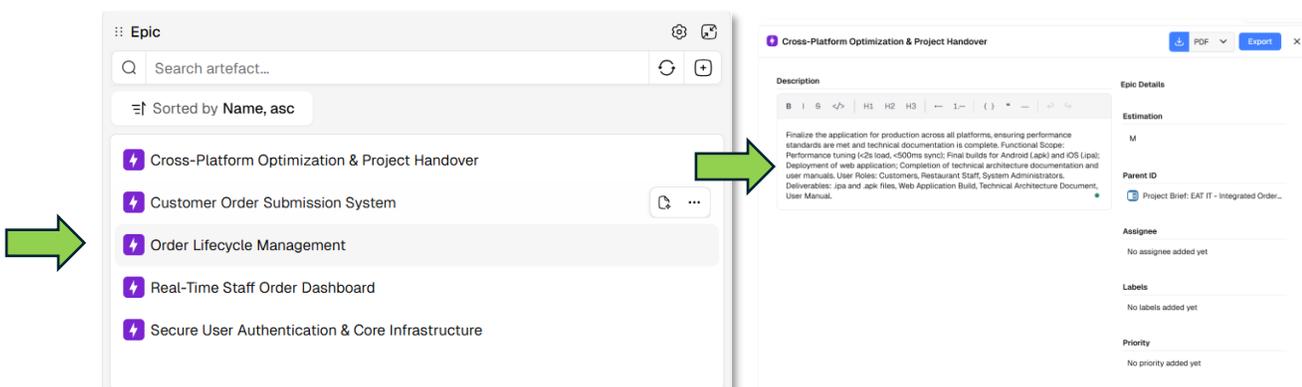
5. The 'Generate' screen is displayed.



6. Observe that the name of the Project Brief that the generation was started from is shown as the 'Project Brief'. This is the source that will be used to seed the AI generation. More source documents can be added by enabling the 'edit' switch and pressing the '+' icon and selecting additional source documents (from the Collateral widget).
7. Press the 'Generate' Button. It may take a short time for the Epics to be generated.
8. Once the Epics have been generated they are displayed in the right hand side of the 'Generation Screen'.



9. The 'n out of n' indicator at the top of the page displays how many Epics have been generated and which Epic is being displayed.
10. At this point there is an option to edit the 'Epic' if required by clicking on text
11. Once any edits have been committed press the 'Save' button.
12. Press the forward and backward arrows to move through the Epics displayed.
13. Press the 'Save' button to commit each generated Epic to the 'Epics' widget.
14. Press the '<' icon to return back to the widgets screen.
15. Observe that the saved 'Epics' are displayed in the 'Epics' widget.
16. Click on a 'Epic' in the 'Epics' widget.
17. The 'Epic' is displayed in a text window.

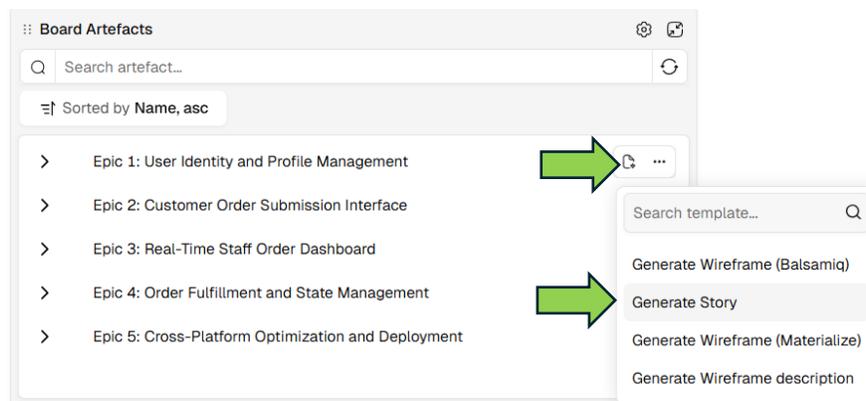




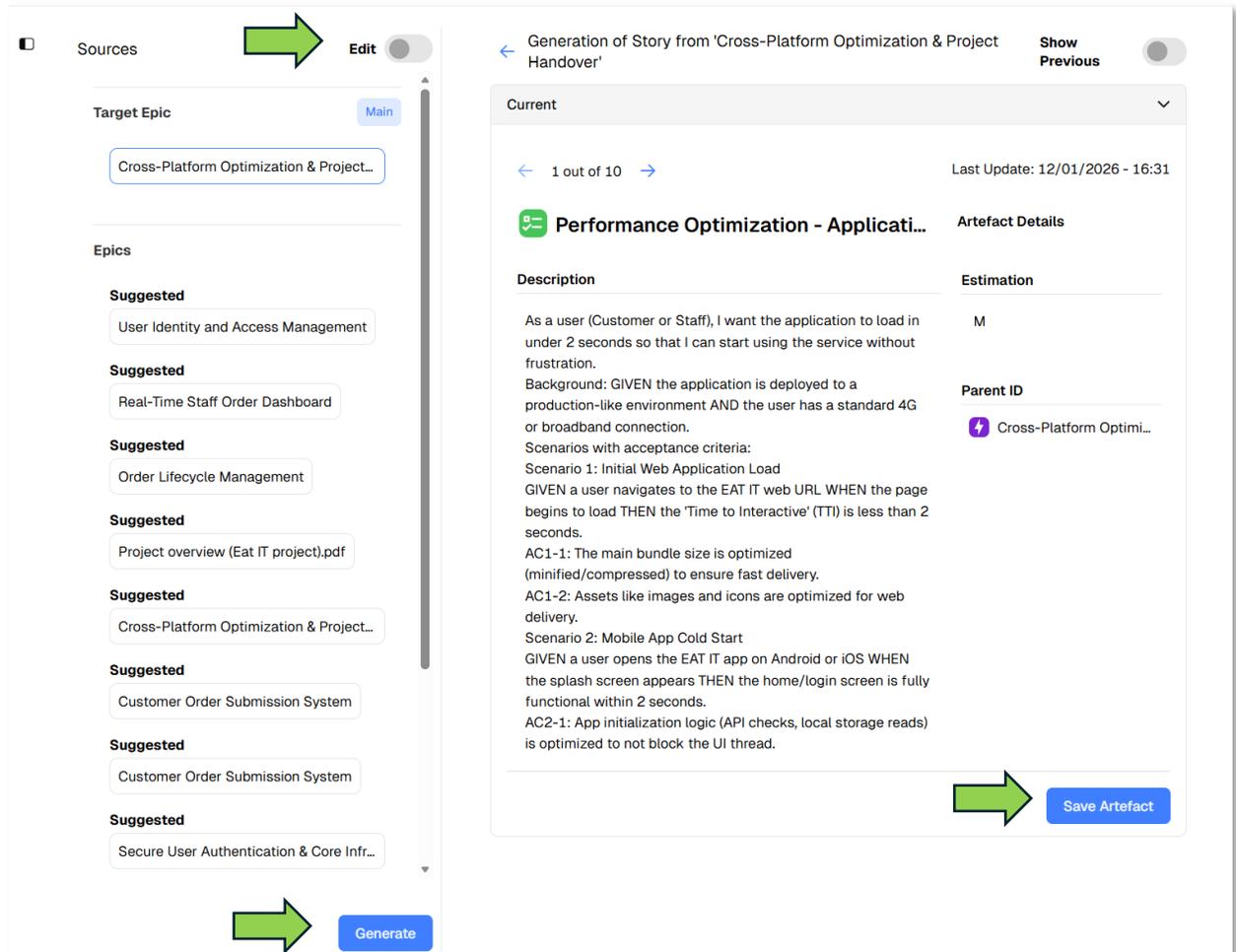
5.7 Generate User Stories

A 'Epic' in the 'Epics' or 'Board Artefacts' widget can be used to generate 'User Stories' using proprietary AI mechanisms. A 'Story' (or 'User Story') is a concise description of a product feature written from the perspective of an end-user, detailing what they need and why. Stories are small, user-centric requirements that can be completed within a single sprint and are organized under larger work items called Epics to provide a complete view of the overall project goal.

1. Go to the 'Widgets' screen.
2. Go to the 'Epics' widget.
3. Select the 'Generate Artefact' icon to the right of an 'Epic'.
4. Select 'Generate Story' from the dropdown.



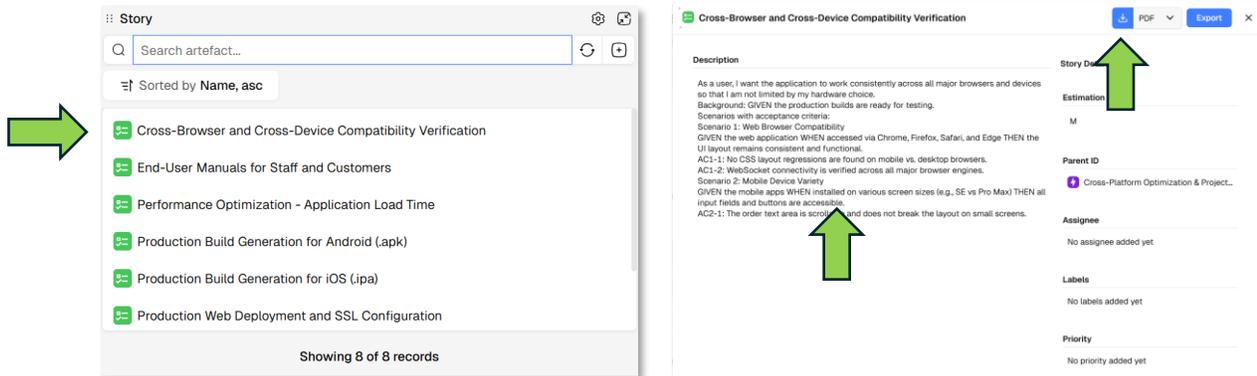
5. The 'Generate' screen is displayed.
6. Observe that the name of the Epic that the generation was invoked from is shown as the 'Target Epic'. This is the source that will be used to seed the AI generation. More source documents can be added by enabling the 'edit' switch and pressing the '+' icon and selecting additional source documents (from the Collateral widget).
7. Press the 'Generate' Button.



8. Once the Stories have been generated they are displayed in the right hand side of the 'Generation Screen'.
9. The 'n out of n' indicator at the top of the page displays how many Stories have been generated and which Story is being displayed.
10. At this point there is an option to edit the 'User Story' if required by clicking on text
11. Once any edits have been committed press the 'Save' button.
12. Press the forward and backward arrows to move through the Stories displayed.
13. Press the 'Save' button to commit each generated Story to the 'User Stories' widget.
14. Press the '<' icon to return back to the widgets screen.



15. Observe that the saved 'Stories' are displayed in the 'User Stories' widget.



16. Click on a 'Story' in the 'User Stories' widget.

17. The 'Story' is displayed in a text window.

18. Click on text in User Story the make further changes.

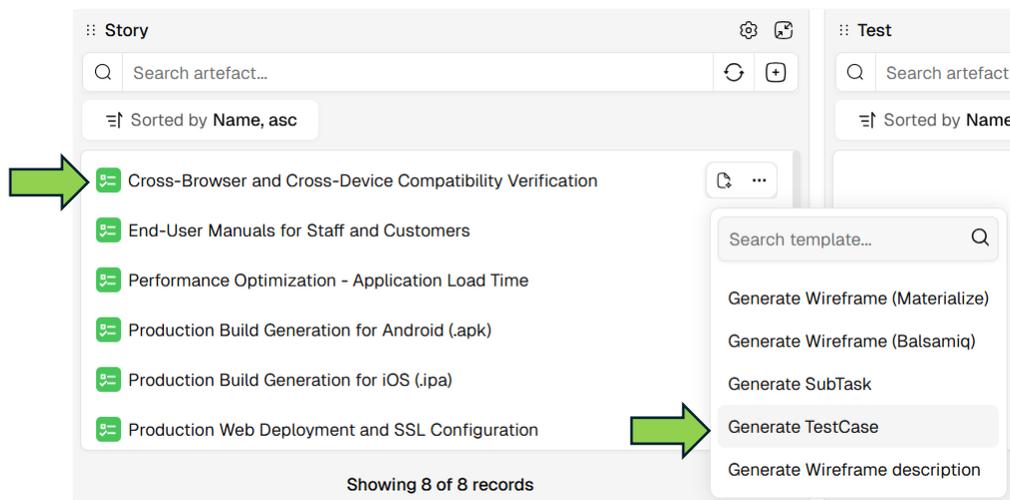
19. Use the download Icon to download in PDF or Word format.



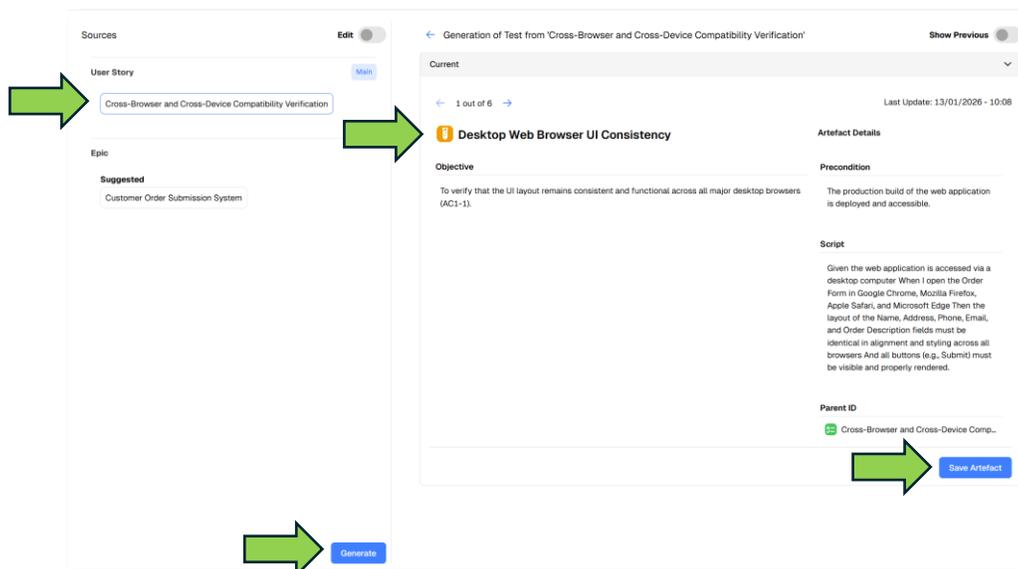
5.8 Generate Test Cases

A 'Story' in the 'User Stories' or 'Board Artefacts' widget can be used to generate 'Test Cases' using proprietary AI mechanisms. Test Cases are a set of conditions that help testers determine whether a product works correctly or satisfies the requirements.

1. Go to the 'Widgets' screen.
2. Go to the 'User Stories' widget.
3. Select the 'Generate Artefact' icon to the right of a 'Story'.
4. Select 'Generate TestCase' from the dropdown.

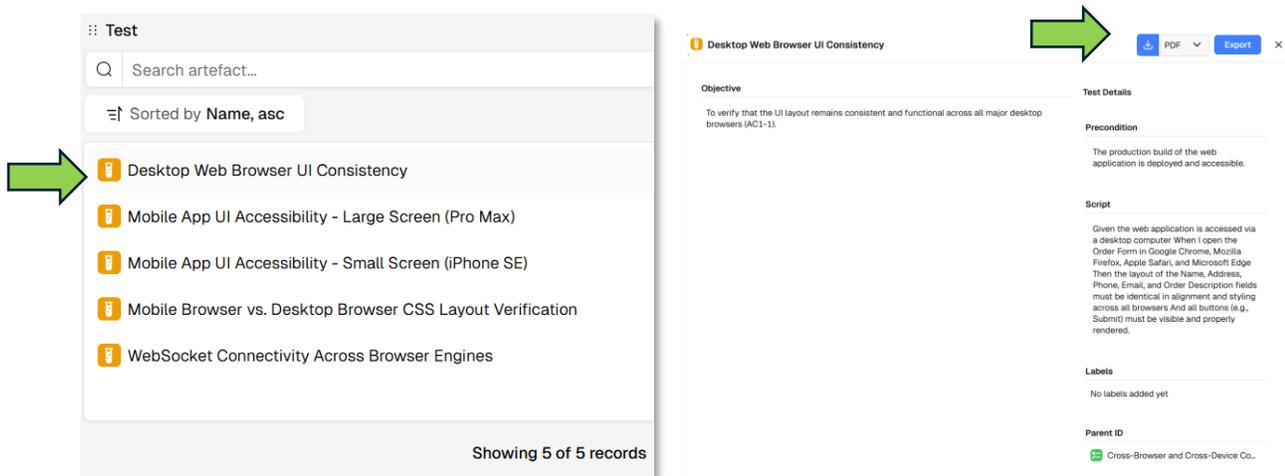


5. The 'Generate' screen is displayed.





6. Observe that the name of the Story that the generation was invoked from is shown as the 'Target Story'. This is the source that will be used to seed the AI generation. More source documents can be added by enabling the 'edit' switch and pressing the '+' icon and selecting additional source documents (from the Collateral and other widgets).
7. Press the 'Generate' Button.
8. Once the Test Cases have been generated they are displayed in the right hand side of the 'Generation Screen'.
9. The 'n out of n' indicator at the top of the page displays how many Test Cases have been generated and which Test Case is being displayed.
10. At this point there is an option to edit by clicking on generated text in 'Test Case'
11. Once any edits have been committed press the 'Save Artefact' button.
12. Press the forward and backward arrows to move through the Test Cases displayed.
13. Press the 'Save' button to commit each generated Test Case to the 'Test Cases' widget.
14. Press the '<- ' icon to return back to the widgets screen.
15. Observe that the saved 'Test Cases' are displayed in the 'Test Cases' widget.



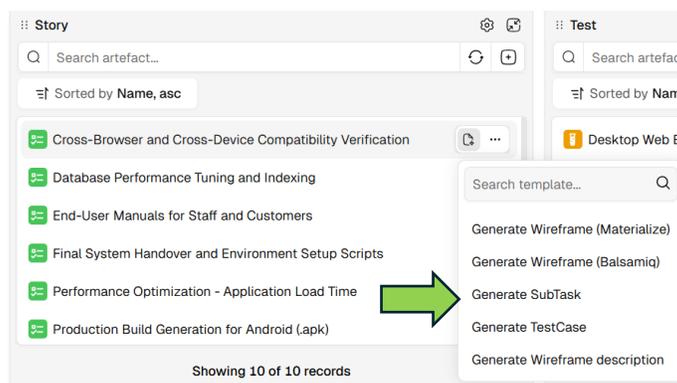
16. Click on a 'Test Case' in the 'Test Cases' widget.
17. The 'Test Case' is displayed in a text window.
18. Click on text to edit and make further changes.
19. Use the download Icon to download in PDF or Word format.



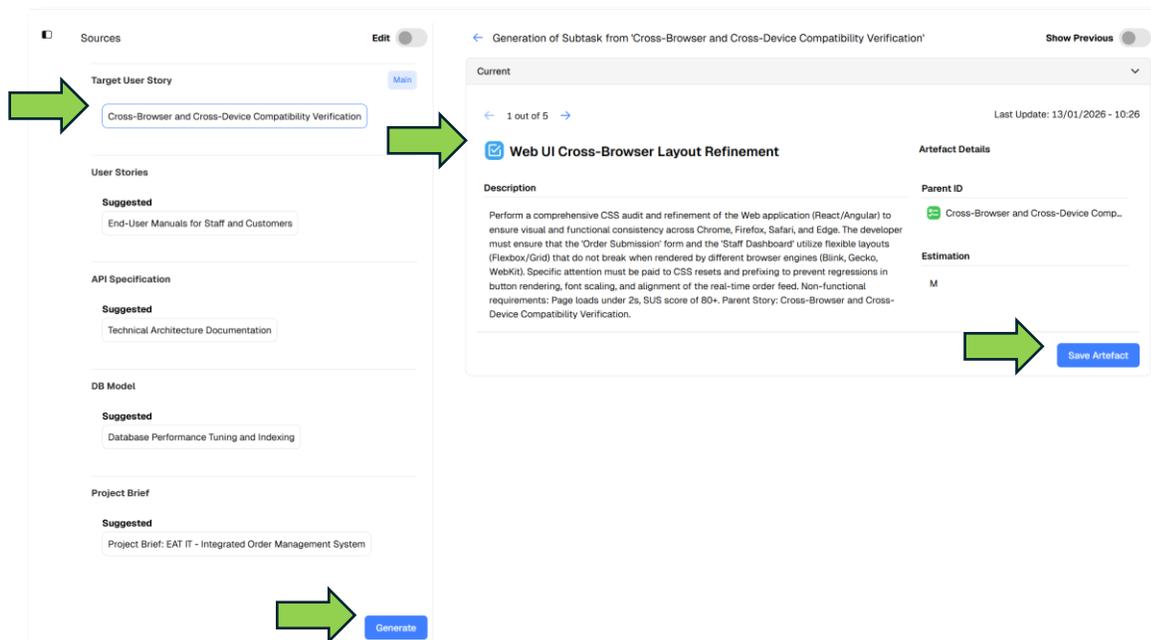
5.9 Generate SubTasks

A 'Story' in the 'User Stories' or 'Board Artefacts' widget can be used to generate 'SubTasks' using proprietary AI mechanisms. A 'SubTask' is a smaller, more manageable piece of work nested under a larger parent 'Story'. Subtasks help break down complex projects into smaller components that can be assigned to different team members, tracked individually, and contribute to the overall progress of the parent issue. They provide enhanced visibility into work and a more granular way to manage workflows and deadlines for a larger project.

1. Go to the 'Widgets' screen.
2. Go to the 'User Stories' widget.
3. Select the 'Generate Artefact' icon to the right of a 'Story'.
4. Select 'Generate SubTasks' from the dropdown.

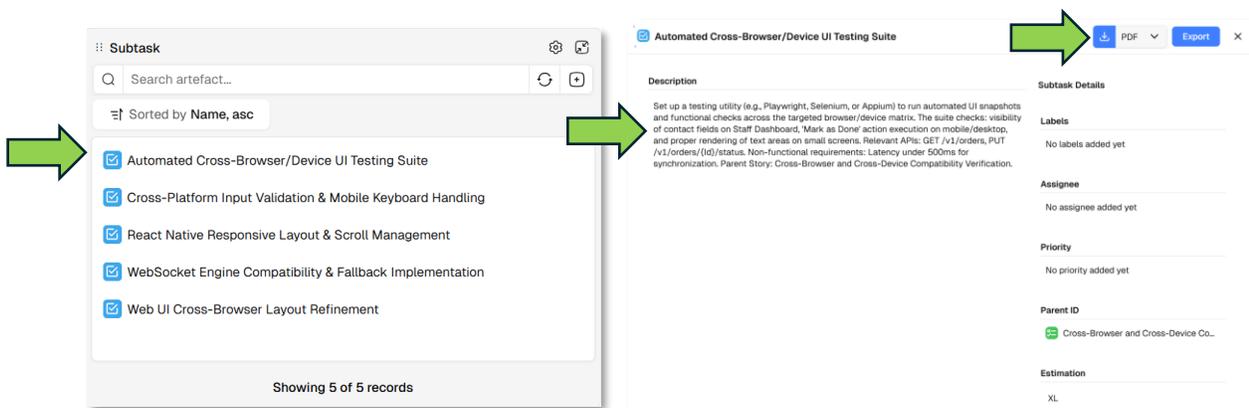


5. The 'Generate' screen is displayed.





6. Observe that the name of the Story that the generation was invoked from is shown as the 'Target Story'. This is the source that will be used to seed the AI generation. More source documents can be added by enabling the 'edit' switch and pressing the '+' icon and selecting additional source documents (from the Collateral and other widgets).
7. Press the 'Generate' Button.
8. Once the SubTasks have been generated they are displayed in the right hand side of the 'Generation Screen'.
9. The 'n out of n' indicator at the top of the page displays how many SubTasks have been generated and which SubTask is being displayed.
10. At this point there is an option to edit text by clicking on text.
11. Press the forward and backward arrows to move through the SubTasks displayed.
12. Press the 'Save' button to commit each generated SubTask to the 'Sub-Tasks' widget.
13. Press the '<' icon to return back to the widgets screen.
14. Observe that the saved 'SubTasks' are displayed in the 'Sub-Tasks' widget.

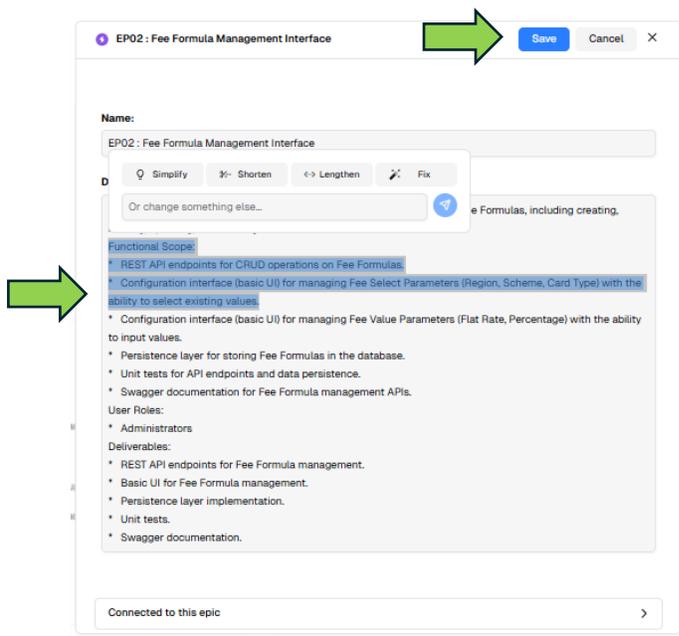


15. Click on a 'SubTask' in the 'Sub-Tasks' widget.
16. The 'SubTask' is displayed in a text window.
17. Use the Edit button to make further changes.
18. Use the download Icon to download in PDF or Word format.

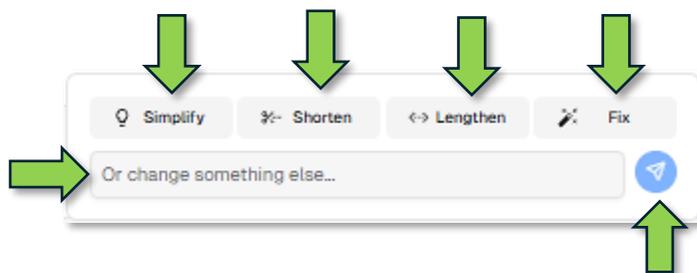


5.10 AI Supported Content Editing

Editing content has AI supported capabilities. To display the AI functionality, in edit mode, simply select and highlight the text you wish to change and the AI edit options will appear in a popup window.



Pressing one of the AI options (Simplify, Shorten, Lengthen & Fix) will apply the AI generated change to the text in the text window. Pressing tick icon will permanently apply the change whilst pressing the X icon will revert the change. Remember that you are still in Edit mode, press Save to keep and Cancel to revert overall changes. AI supported content editing is available for all text based artefact when in edit mode.



5.10.1 Simplify

Press the 'Simplify' button to reduce the complexity of the language in the highlighted text. Press the green tick to commit change, cross to cancel.



5.10.2 Shorten

Press the 'Shorten' button to replace the highlighted text with more concise version. Press the green tick to commit change, cross to cancel.

5.10.3 Lengthen

Press the 'Lengthen' button to replace the highlighted text with lengthened and elaborated version. Press the green tick to commit change, cross to revert.

5.10.4 Fix

Press the 'Fix' button to correct any typographical errors in the highlighted text. Press the green tick to commit change, cross to revert.

5.10.5 Prompt

Enter a prompt style request into the "or change something else" text field and press the arrow button to replace the highlighted text with the outcome of the prompt. For example, try generating a completely different description of an epic by highlighting all the text and then entering a prompt such as *"create the content of an Epic using collateral my-design.pdf as source and restricting the output to the subject of authentication"*. Press the green tick to commit change, cross to revert.



6 Connector Configuration

6.1 Intro to Connectors

ProgramGenie can integrate with a number of external products. The sections below describe the configuration required for each connector. Connectors can be configured when a project is created (optional) or added at a later point via the ‘Settings’ screen. It is possible to add multiple connectors of the same type, for example to allow two Jira repositories to be available in the same ProgramGenie project. For Jira and Confluence connectors, if the repository defined in the configuration does not exist, ProgramGenie will automatically create it for the user.

6.2 Jira Connector

The ‘Jira’ connector allow ProgramGenie to be integrated with Jira, a powerful project management and issue-tracking software developed by Atlassian.

6.2.1 Integration

Select the product that ProgramGenie is to be integrated with, in this case Jira.

6.2.2 Name

Create a unique name of the instance of the connector.

6.2.3 Description

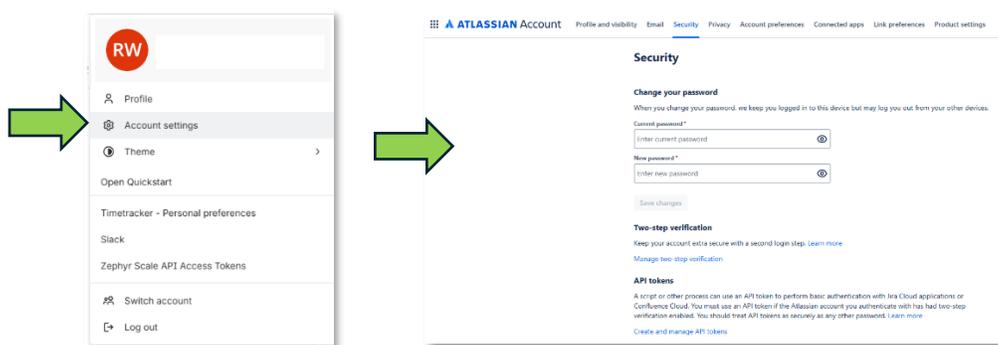
An optional description of the use of the instance of the connector.

6.2.4 Domain

Your Atlassian domain, for example “https://my-domain.atlassian.net”.

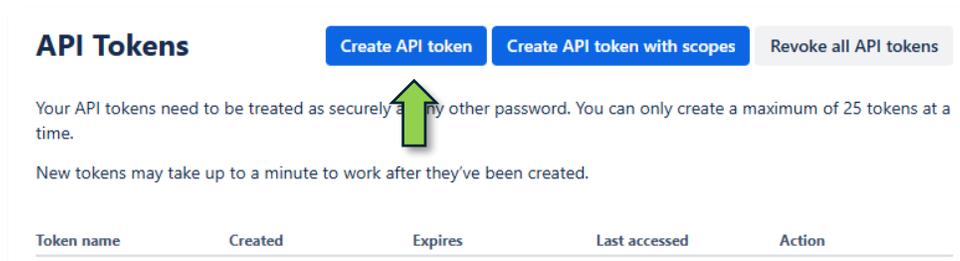
6.2.5 Token

The Jira authentication token used by the project to connect, this is used in conjunction with the Email address field. The token must be generated by the (email) user, to obtain the token:

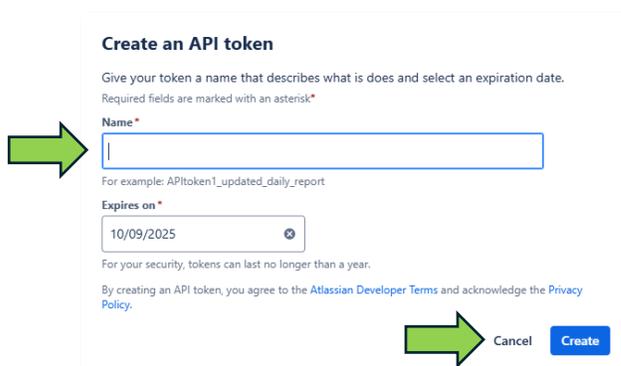




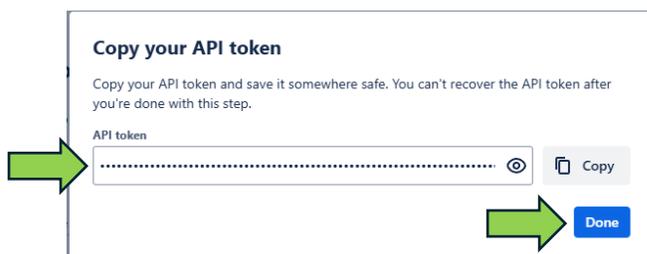
1. In Jira, select profile -> Account Settings
2. The Accounts settings screen is displayed, select the Security tab.
3. In the 'API Tokens' section, click on 'Create and Manage API tokens'.
4. The API Tokens screen is displayed.



5. Click on Create API Token.



6. Give the connector a name and set the expiry date, ProgramGenie will stop being able to interact with Jira when the expiry date is reached.
7. Press create.



8. Copy the generated token and paste it in the Token field in the ProgramGenie connector screen.



6.2.6 Email

The email address of the user (generally the project lead) whose connection and Jira access rights will be used by the project to authenticate. This must be the same user that generated the Token. The user must have appropriate Jira write access to create, edit and delete Epics & Stories, etc.

6.2.7 Project Key

A maximum ten character alphanumeric key (with first character capitalized) for the Project, this is the code that Jira uses as the prefix for Issue Key values i.e. 'ABCD as in story ABCD-23'. If connecting to an existing Jira project use the key for that project. If creating a new project use a unique (unused) key and ProgramGenie will automatically create a new project with this key.



6.3 Confluence Connector

The 'Confluence' connector allow ProgramGenie to be integrated with Confluence. Confluence is a team collaboration and knowledge management tool developed by Atlassian, it serves as a centralized workspace where teams can create, organize, and collaborate.

6.3.1 Integration

Select the product that ProgramGenie is to be integrated with, in this case Confluence.

6.3.2 Name

Create a unique name of the instance of the connector.

6.3.3 Description

An optional description of the use of the instance of the connector.

6.3.4 Domain

Your Atlassian domain, for example "https://my-domain.atlassian.net".

6.3.5 Token

See token creation in the Jira connector, confluence uses the same token generation mechanism. Jira and Confluence can use the same email & token combination for both.

6.3.6 Email

See email setup in the Jira connector, confluence uses the same email-token generation mechanism. Jira and Confluence can use the same email & token combination for both connectors.

6.3.7 Project Key

An alphanumeric name key for the Confluence project. ProgramGenie will automatically create a new confluence project from this key for use with this ProgramGenie project.



6.4 Zephyr Connector

The 'Zephyr' connector allows ProgramGenie to be integrated with Zephyr, a test management and execution tool for Jira.

6.4.1 Integration

Select the product that ProgramGenie is to be integrated with, in this case Zephyr.

6.4.2 Name

Create a unique name of the instance of the connector.

6.4.3 Description

An optional description of the use of the instance of the connector.

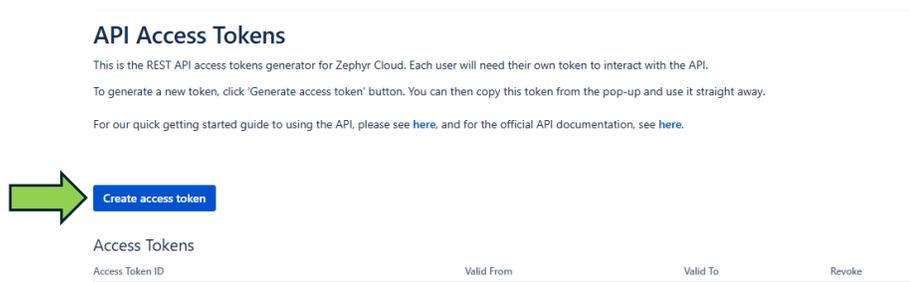
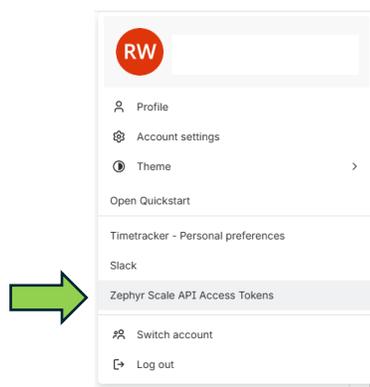
6.4.4 Domain

Your Atlassian domain, for example "https://my-domain.atlassian.net".

6.4.5 Token

Zephyr requires a different Token to be generated, different to the Jira/Confluence Token. The token must be generated by the same user set in the 'email' field, to obtain the token:

1. In Jira, select profile -> Zephyr Scale API Access Tokens





2. The Zephyr API Access Tokens screen is displayed
3. Press Create Access Token
4. The Access Token screen is displayed, copy the token and paste into the token field on the zephyr connector.



6.4.6 Email

The email address of the user (generally the project lead) whose connection and access rights will be used by Zephyr to authenticate. This must be the same user that generated the Zephyr Token. The user must have appropriate write access rights to interact with Zephyr & Jira.

6.4.7 Project Key

The four character alphanumeric name key for the Jira project that Zephyr will integrate with.



6.5 Azure DevOps (ADO) Connector

The 'Azure DevOps' connector allows ProgramGenie to be integrated with Azure DevOps (ADO), a SaaS platform provided by Microsoft that offers a comprehensive suite of tools for software development and collaboration. The ADO repository needs to have been created already in order to connect to it.

6.5.1 Integration

Select the product that ProgramGenie is to be integrated with, in this case Azure DevOps.

6.5.2 Name

Create a unique name of the instance of the connector.

6.5.3 Description

An optional description of the use of the instance of the connector.

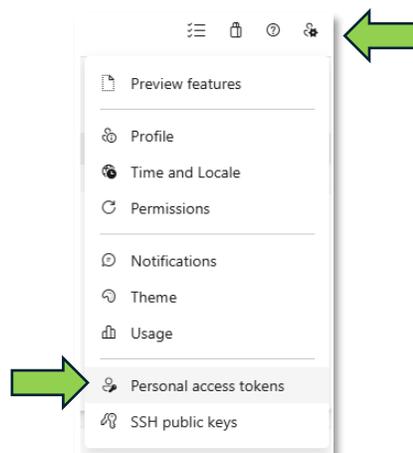
6.5.4 Organization

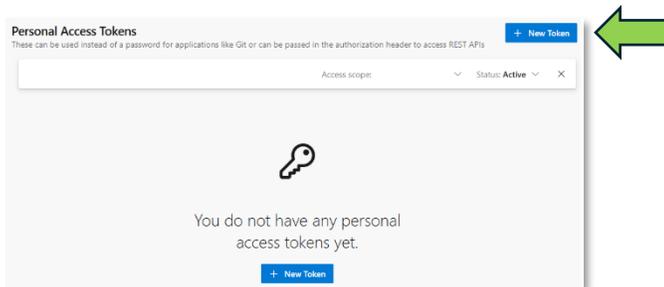
Your organization, as described in ADO for example "my-company-com".

6.5.5 Token

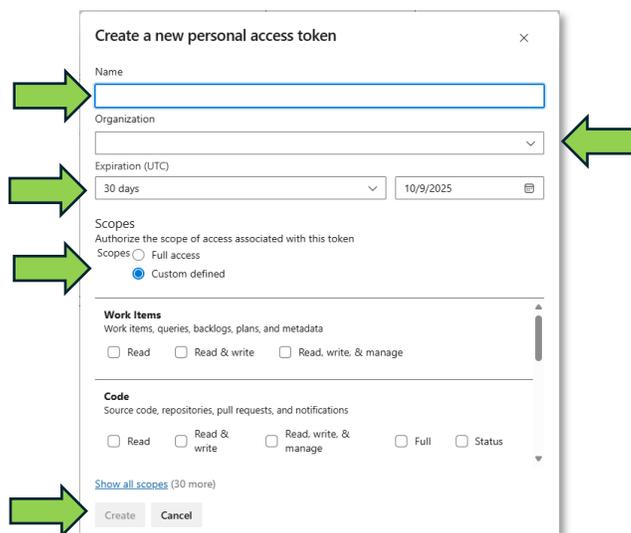
ADO requires a Token to be generated, to obtain the token:

1. In ADO select 'user settings -> Personal access tokens'
2. The 'Personal Access Tokens' screen is displayed.
3. Press the 'New Token' button.





4. The 'Create a new personal access token' screen is displayed.



5. Enter a Name for the token.
6. Select your Organization.
7. Select Expiry.
8. Select Access Scope required.
9. Press the 'Create' button.
10. The Success! Screen is displayed.
11. Copy the token and paste it into the token field on the ADO screen.

6.5.6 Email

Enter the email address that corresponds with the user used to create the authentication token used in the Token field above.

6.5.7 Project Key

A four character alphanumeric name key for the ADO project.



6.6 GitLab Connector

The 'GitLab' connector allows ProgramGenie to be integrated with GitLab, an all-in-one DevSecOps platform for software development.

6.6.1 Integration

Select the product that ProgramGenie is to be integrated with, in this case GitLab.

6.6.2 Name

Create a unique name of the instance of the connector.

6.6.3 Description

An optional description of the use of the instance of the connector.

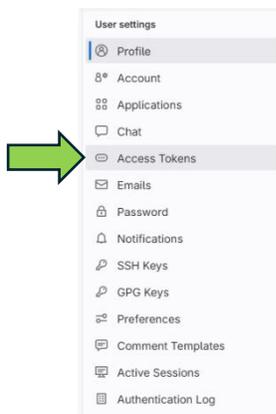
6.6.4 Domain

Your GitLab domain including the path to the project, for example “https://gitlab.my-company.com/examples/my-example”

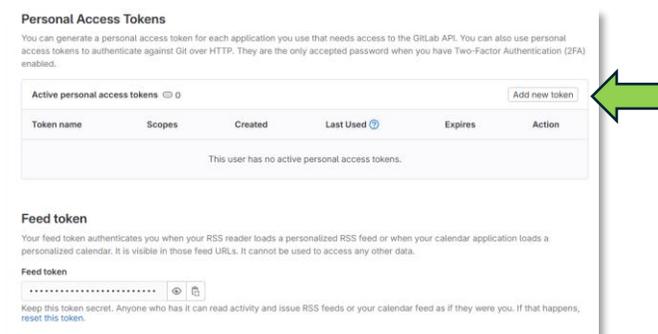
6.6.5 Token

GitLab requires a Token to be generated, to obtain the token:

1. In the GitLab User Settings sidebar, click on 'Access Tokens'.



2. The Personal Access Tokens screen is displayed.





3. Click on 'Add new token'.
4. The 'Add a personal access token' is displayed.

Active personal access tokens 0

Add a personal access token

Token name
Handbook[GitLab]

For example, the application using the token or the purpose of the token.

Expiration date
2025-10-31

Select scopes
Scopes set the permission levels granted to the token. [Learn more.](#)

- api**
Grants complete read/write access to the API, including all groups and projects, the container registry, the dependency proxy, and the package registry.
- read_api**
Grants read access to the API, including all groups and projects, the container registry, and the package registry.
- read_user**
Grants read-only access to the authenticated user's profile through the Juser API endpoint, which includes username, public email, and full name. Also grants access to read-only API endpoints under Juser.
- create_runner**
Grants create access to the runners.
- k8s_proxy**
Grants permission to perform Kubernetes API calls using the agent for Kubernetes.
- read_repository**
Grants read-only access to repositories on private projects using Git-over-HTTP or the Repository Files API.
- write_repository**
Grants read-write access to repositories on private projects using Git-over-HTTP (not using the API).
- read_registry**
Grants read-only access to container registry images on private projects.
- write_registry**
Grants write access to container registry images on private projects. You need both read and write access to push images.
- ai_features**
Grants access to GitLab Duo related API endpoints.

[Create personal access token](#) [Cancel](#)

Token name	Scopes	Created	Last Used	Expires	Action
This user has no active personal access tokens.					

Feed token
Your feed token authenticates you when your RSS reader loads a personalized RSS feed or when your calendar application loads a personalized calendar. It is visible in those feed URLs, it cannot be used to access any other data.

Feed token
.....

Keep this token secret. Anyone who has it can read activity and issue RSS feeds or your calendar feed as if they were you. If that happens, [reset this token.](#)

5. Add a Name for the Token.
6. Set the expiry date.
7. Select the access rights.
8. Click on 'Create personal access token'.
9. The 'Personal Access Tokens' screen is displayed.

Personal Access Tokens
You can generate a personal access token for each application you use that needs access to the GitLab API. You can also use personal access tokens to authenticate against Git over HTTP. They are the only accepted password when you have Two-Factor Authentication (2FA) enabled.

✔ **Your new personal access token**
.....
Make sure you save it - you won't be able to access it again.

Active personal access tokens 1 [Add new token](#)

Token name	Scopes	Created	Last Used	Expires	Action
Handbook GitLab	api_read_api, read_user, create_runner, k8s_proxy, read_repository, write_repository, read_registry, write_registry, ai_features	Sep 09, 2025	Never	in 1 month	

10. The token is shown (obfuscated). Click on the 'Copy Personal Access Token' icon to make a copy of the token.
11. Paste the copied token into the Token field in the ProgramGenie GitLab connector screen.



6.6.6 Project

Select the Project from the project dropdown, this should be the same project in the referenced in the Domain.

6.6.7 Branch

Select the code Branch you are interested in.

6.6.8 Email

Enter the email address that corresponds with the user used to create the authentication token used in the Token field above.

6.6.9 Project Key

The Project Key will be set automatically when you select the Project & Branch. The field is not editable.



6.7 BitBucket Connector

The 'BitBucket' connector allows ProgramGenie to be integrated with BitBucket, a Git-based code hosting and collaboration platform designed for professional teams.

6.7.1 Integration

Select the product that ProgramGenie is to be integrated with, in this case BitBucket.

6.7.2 Name

Create a unique name of the instance of the connector.

6.7.3 Description

An optional description of the use of the instance of the connector.

6.7.4 Domain

Your BitBucket domain including the path to the project, for example "https://bitbucket.org/my-example"

6.7.5 Token

BitBucket requires a Token to be generated.

6.7.6 Project

The name of the BitBucket project.

6.7.7 Branch

The BitBucket project branch.

6.7.8 Email

The user name for the project, for example "myproject-admin".

6.7.9 Project Key

The Project Key will be automatically set once the Project and Branch are selected. The field is not editable.



6.8 GitHub Connector

The 'GitHub' connector allows ProgramGenie to be integrated with GitHub, a cloud-based platform for hosting and collaborating on software projects, built on the Git version control system.

6.8.1 Integration

Select the product that ProgramGenie is to be integrated with, in this case GitHub.

6.8.2 Name

Create a unique name of the instance of the connector.

6.8.3 Description

An optional description of the use of the instance of the connector.

6.8.4 Domain

Your GitHub domain including the path to the project.

6.8.5 Token

GitHub requires a Token to be generated, you can do this within your GitHub user account "Github -> Settings -> Personal access tokens -> Generate new".

6.8.6 Project

The name of the GitHub project.

6.8.7 Branch

The GitHub project branch.

6.8.8 Email

The user name for the project that the token was created for.

6.8.9 Project Key

The Project Key will be automatically set once the Project and Branch are selected. The field is not editable.



6.9 Test-Rail Connector

The 'Test-Rail' connector allows ProgramGenie to be integrated with Test-Rail, a web-based test case management tool.

6.9.1 Integration

Select the product that ProgramGenie is to be integrated with, in this case Test-Rail.

6.9.2 Name

Create a unique name of the instance of the connector.

6.9.3 Description

An optional description of the use of the instance of the connector.

6.9.4 Domain

Your Test-Rail domain.

6.9.5 Token

Test-Rail requires a Token to be generated, you can do this within your Test-Rail user account.

6.9.6 Email

The user name for the project that the token was created for.

6.9.7 Project Key

The Project Key for the project.



6.10 Xray Connector

The 'Xray' connector allows ProgramGenie to be integrated with Xray, a test management solution for Jira.

6.10.1 Integration

Select the product that ProgramGenie is to be integrated with, in this case Xray.

6.10.2 Name

Create a unique name of the instance of the connector.

6.10.3 Description

An optional description of the use of the instance of the connector.

6.10.4 Domain

Your Xray domain.

6.10.5 Token

Xray requires a Token to be generated, you can do this within your Xray user account.

6.10.6 Email

The user name for the project that the token was created for.

6.10.7 Project Key

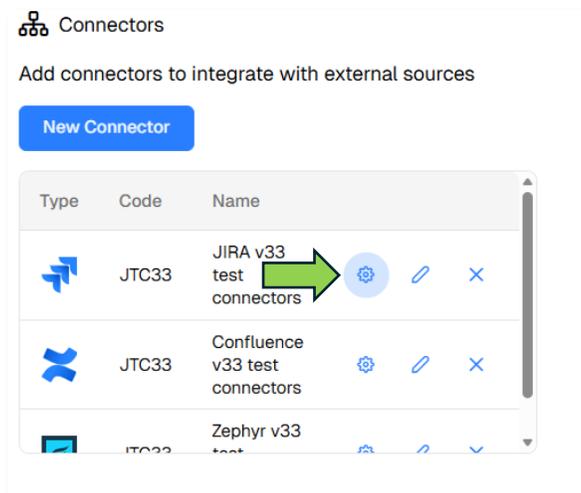
The Project Key for the project.



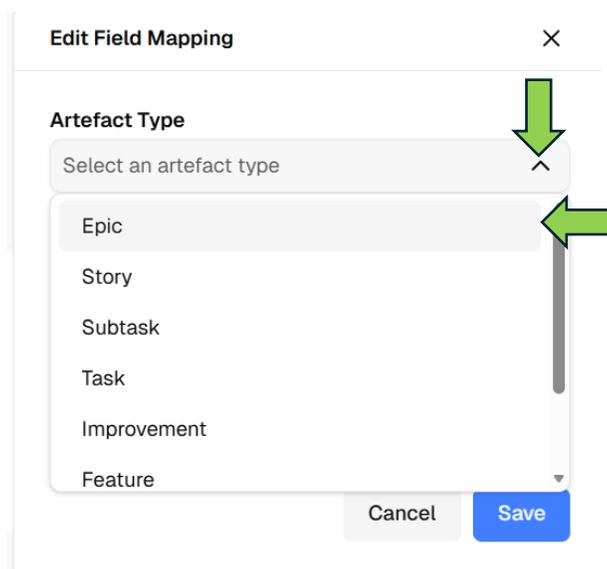
6.11 Edit Connectors fields options

Connector fields can be edited if users wish to change fields specific to their JIRA instance to ensure connectors work properly. In this example, there are default fields for JIRA for Epics:

1. Go to the 'Connectors' part of the setup screen.
2. Click on the Cog icon.



3. Select an artefact type





4. Change the value according to the field name in the JIRA mapping. User must be aware of the fields that exist in their version of JIRA; if the wrong name of a field is added, the connector will not work properly.
5. After the change, press the “Save” button.

Edit Field Mapping ×

Artefact Type
Epic ▾

Name
summary

Description
description

Estimation
timetracking.originalEstimate 

Parent ID
parent.key

Labels
labels

Priority
priority.name

Assignee
assignee.emailAddress

Cancel Save 

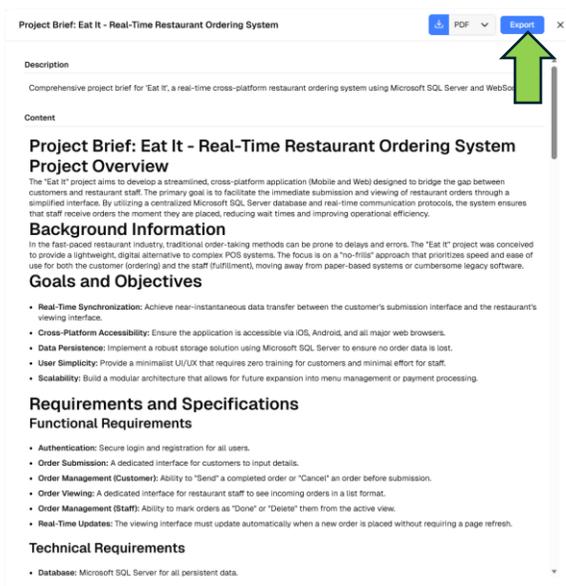


7 Importing & Exporting Artefacts via Connectors

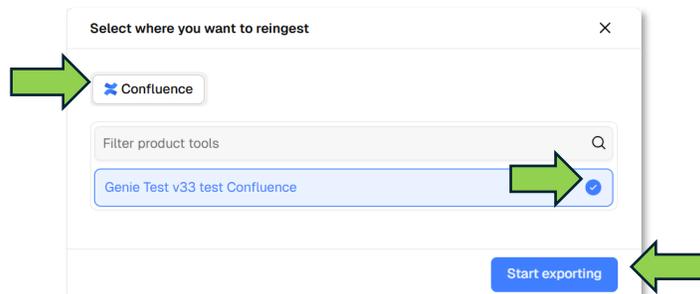
Artefacts generated in ProgramGenie can be exported to external repositories such as Jira or Confluence via ProgramGenie Connectors (described in Connector Configuration section). Likewise, existing data in external repositories such as Jira and various code repositories can be imported as artefacts

7.1 Export to Confluence

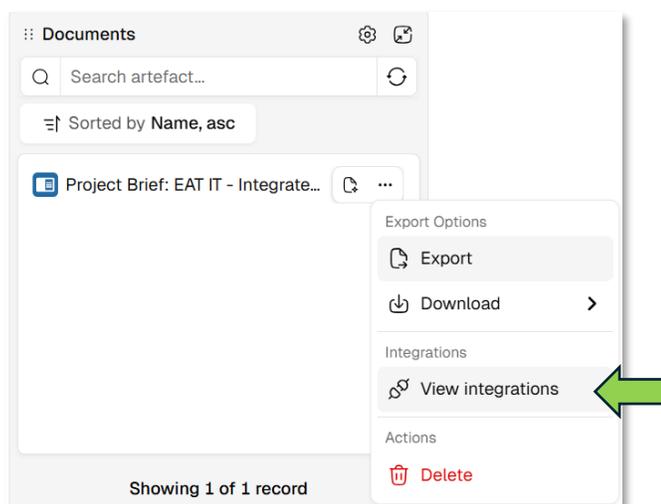
A documents can be exported to Confluence, this can be done from the text view by pressing the 'Export' button at the top of the page. To export to Confluence, the integration between ProgramGenie and Confluence must be configured either when the project was created or later via the 'Settings' screen (this is described in the Connectors section). Example of exporting Project Brief to Confluence:



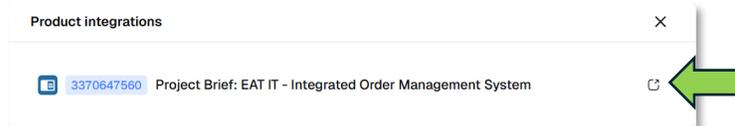
6. Go to the 'Widgets' screen.
7. Go to the 'Documents' widget.
8. Click on the project brief file in widget.
9. Press the 'Export' button on an expanded Project Brief.
10. The 'Export' window is displayed.
11. Note: if the Confluence Connector is not configured a connector not configured warning will be displayed.



12. Click on the 'Confluence' icon.
13. The configured confluence connectors are displayed.
14. It is possible to have more than one Confluence connector configured (see Connectors section for more information).
15. Select the Confluence connector you wish to export to (this can be more than one).
16. Press the 'Export' button.
17. Go to the 'Documents' widget.



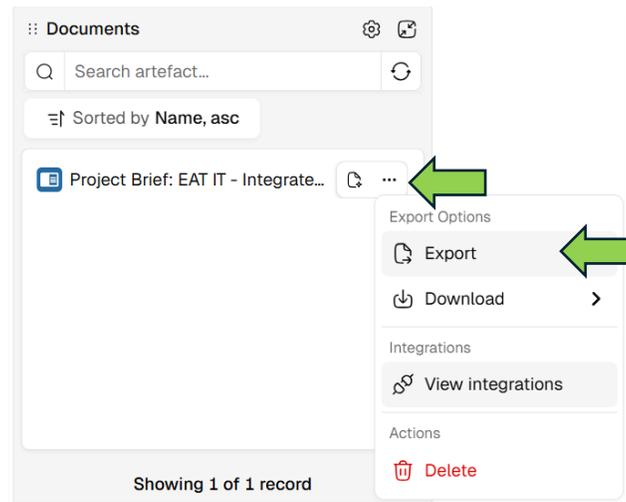
18. Click on the 'Link' icon.
19. The 'Product Integrations' window is displayed.



20. Click on the External Link icon to the right of the name of the Project Brief.
21. A Confluence page is displayed containing the Project Brief information.



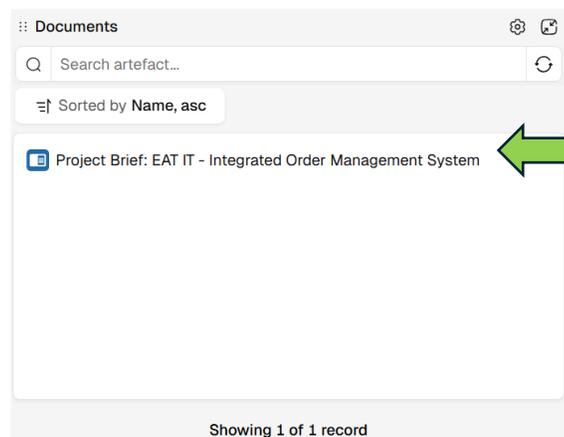
Alternative option to export is to use “Export Options” menu:



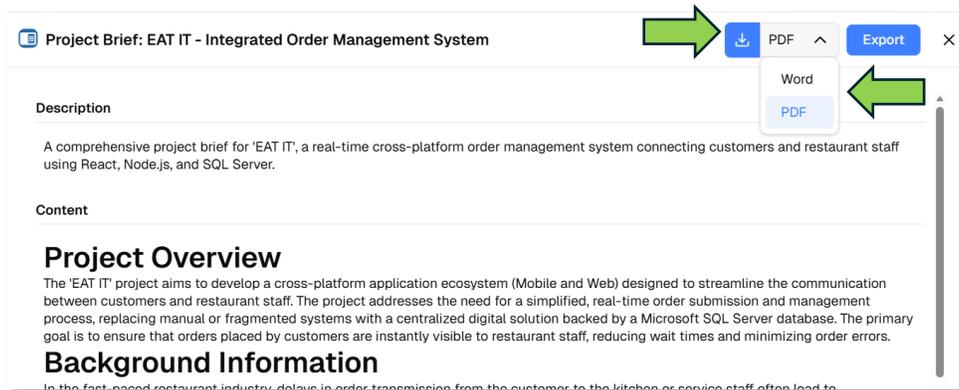
7.1.1 Download (Word/PDF)

Once a Project Brief has been opened in the text viewer, it may be downloaded, in either Word or PDF format.

1. Go to the ‘Widgets’ screen.
2. Go to the ‘Documents’ widget.

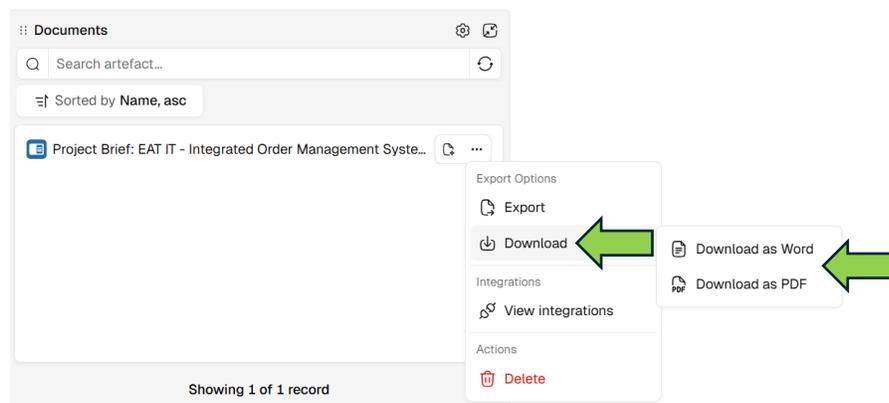


3. Click on the document name to open the text viewer.
4. In the document viewer click on the dropdown arrow and select the output format you want the download to be created in, either Word or PDF.
5. Press the download icon.



6. The document is saved to your browser download folder.

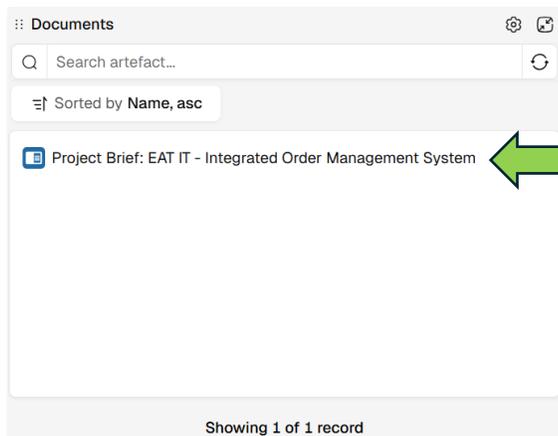
Alternative option to export is to use “Export Options” menu:



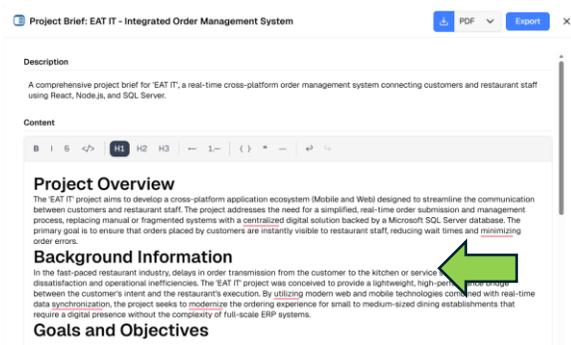
7.1.2 Edit

Documents can be edited once they have been generated. Remember, exported documents need to be re-exported to resynchronize your local changes.

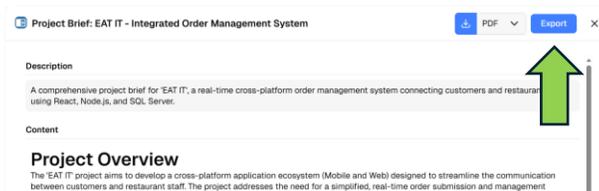
1. Go to the ‘Widgets’ screen.
2. Go to the ‘Documents’ widget.



3. Click on the document name to open the text viewer.
4. Click on text in the document to enter edit mode.



5. Once the editing is complete press outside text field to save changes.
6. If the document has been previously exported, it must be re-exported to synchronize the changes made with the external version.



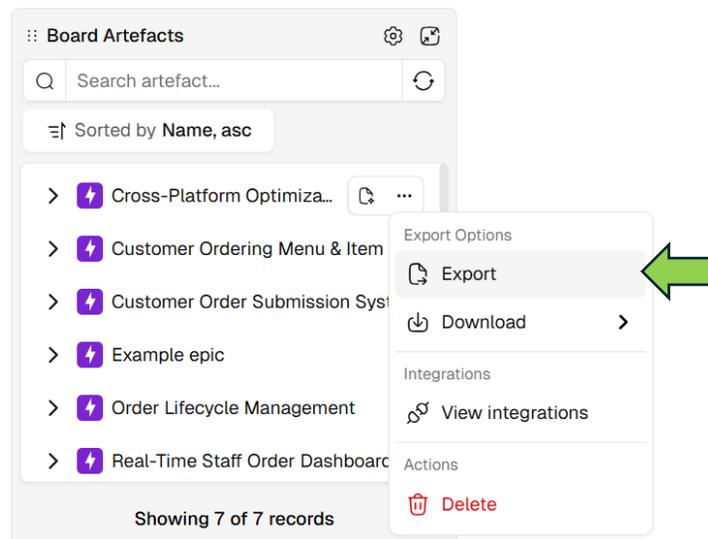


7.2 Export artefacts to Jira

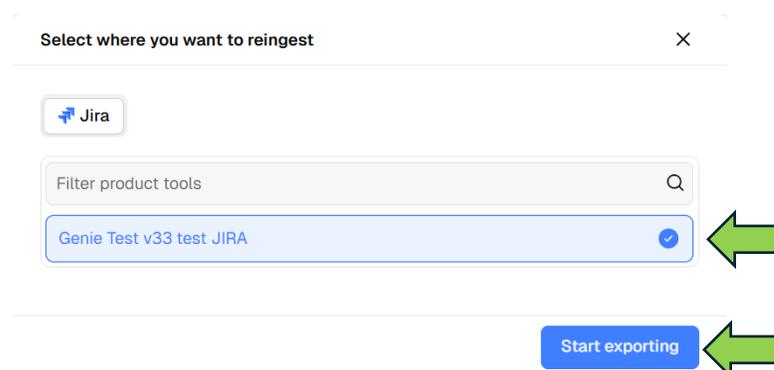
An artefact can be exported to Jira; this can be done from the text view of the board artefacts by pressing the ‘Export ‘ button at the top of the page or by pressing the ‘Export’ Icon in the “Export Options” menu linked to artefacts in the widget. To export to Jira, the integration between ProgramGenie and Jira must be configured either when the project was created or later via the ‘Settings’ screen (this is described in the Connectors section).

Exporting of Epic example:

1. Go to the ‘Widgets’ screen.
2. Go to the ‘Epics’ widget.
3. Press the ‘Export’ icon to the right of an ‘Epic’.
4. Or Click on the ‘Epic’ then press the ‘Export’ button on the text view displayed.

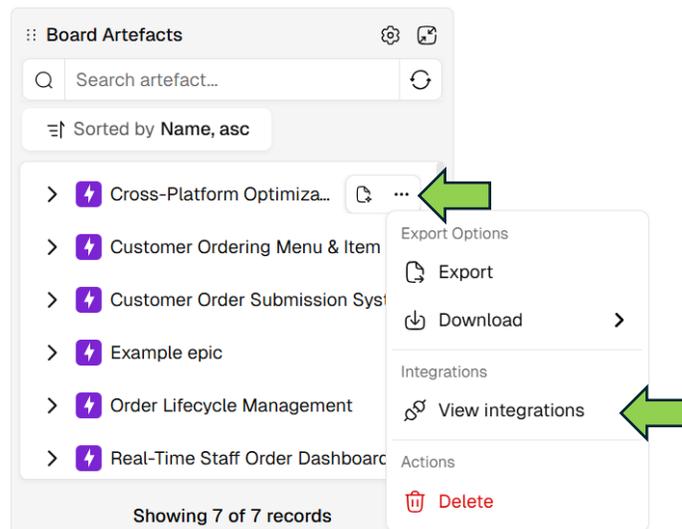


5. The ‘Export’ window is displayed.
6. Or if the Jira Connector is not configured a ‘connector not configured’ warning will be displayed.

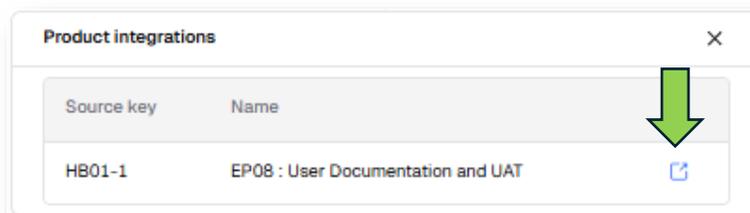




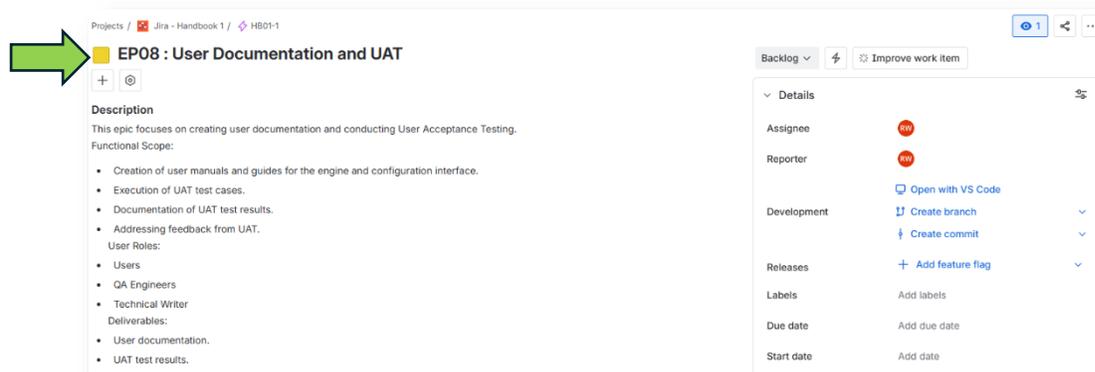
7. Click on the 'Jira' icon.
8. The configured Jira connectors are displayed.
9. It is possible to have more than one Jira connector configured (see Connectors section for more information).
10. Select the Jira connector you wish to export to (this can be more than one).
11. Press the 'Export' button.
12. Go to the 'Epics' widget.
13. Click on the 'Link' icon by pressing "Export Options" icon (...) and clicking on View integrations icon



14. The 'Product Integrations' window is displayed.



15. Click on the External Link icon to the right of the name of the Epic.
16. A Jira Epic page is displayed containing the Epic information.



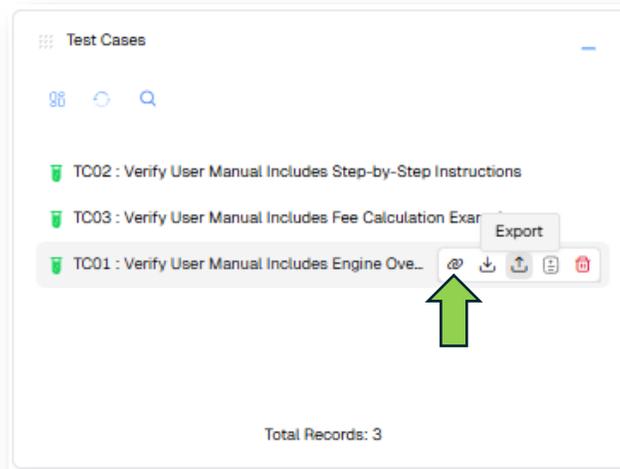




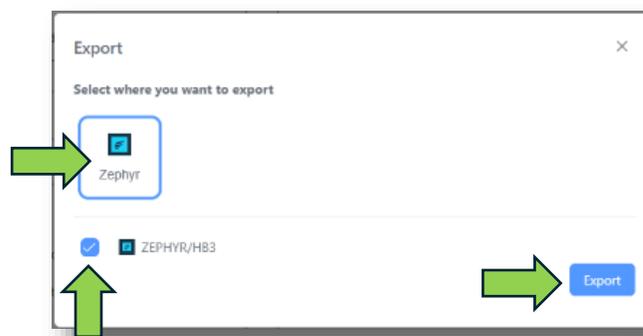
7.4 Export a Test Case to Zephyr

A 'Test Case' can be exported to Zephyr, this can be done from the text view of the Test Case by pressing the 'Export' button at the top of the page or by pressing the 'Export' Icon next to the, 'Test Case' in the 'Test Cases' or 'Board Artefacts' widget. To export to Jira, the integration between ProgramGenie and Zephyr must be configured either when the project was created or later via the 'Settings' screen (this is described in the Connectors section). Also, the 'Epic' and 'Story' that the 'Test Case' is derived from must be exported to Jira to allow a 'Test Case' to be exported to Zephyr.

1. Go to the 'Widgets' screen.
2. Go to the 'Test Cases' widget.
3. Press the 'Export' icon to the right of a 'Test Case'.
4. Or Click on the 'Test Case' then press the 'Export' button on the text view displayed.



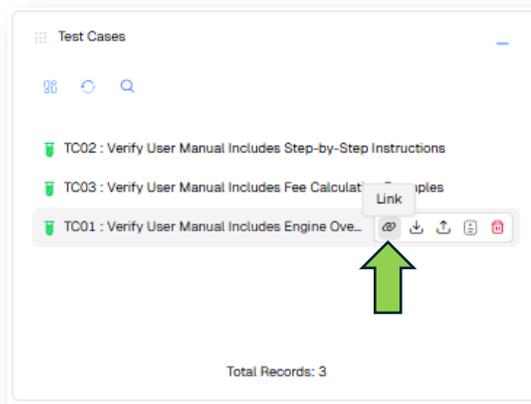
5. The 'Export' window is displayed.
6. Or if the Jira Connector is not configured a 'connector not configured' warning will be displayed.



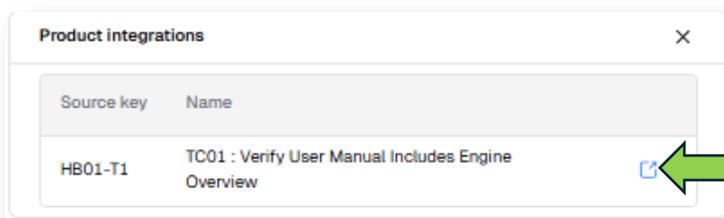
7. Click on the 'Zephyr' icon.



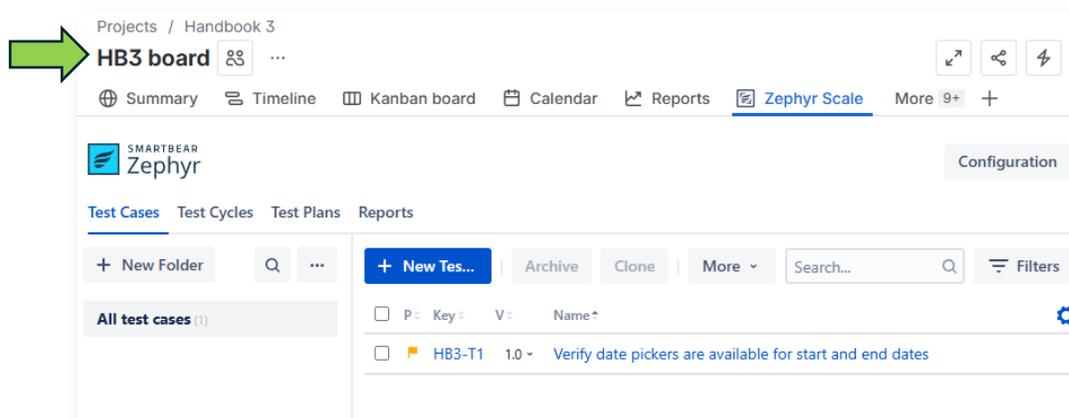
- The configured Zephyr connectors are displayed.
- Select the Zephyr connector you wish to export to.
- Press the 'Export' button.
- Go to the 'Test Cases' widget.



- Click on the 'Link' icon.
- The 'Product Integrations' window is displayed.



- Click on the External Link icon to the right of the name of the Test Case.
- A Zephyr page is displayed containing the Test Case information.

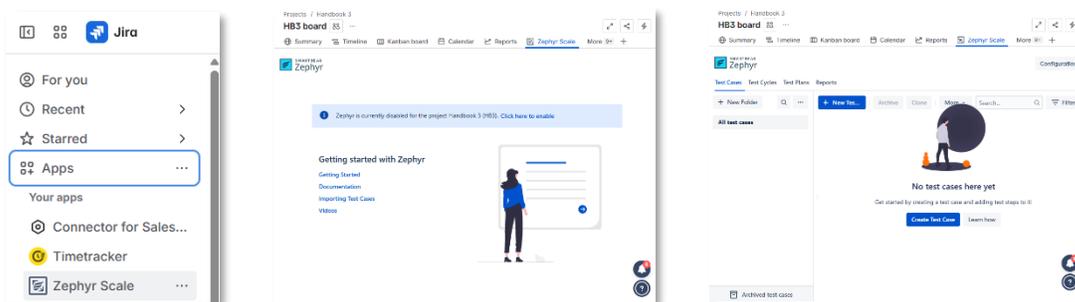




7.4.1 Enable Zephyr Scale

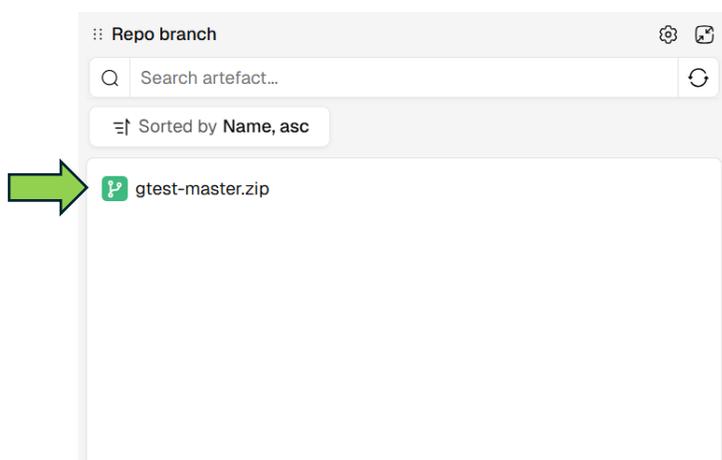
Test Cases can be exported to Zephyr. Zephyr for Jira is a test management tool that integrates directly with Jira to provide features for creating, organizing, executing, and tracking software tests within the Jira platform. In order to export Zephyr must be enabled using the following steps.

1. Go to the Jira project associated with the ProgramGenie project.
2. Select Apps -> Zephyr Scale from the Jira sidebar.
3. Zephyr is displayed showing a Zephyr is disabled warning.
4. Click on the 'Click Here To Enable' message.
5. Zephyr is enabled and displayed.



7.5 Repositories Widget

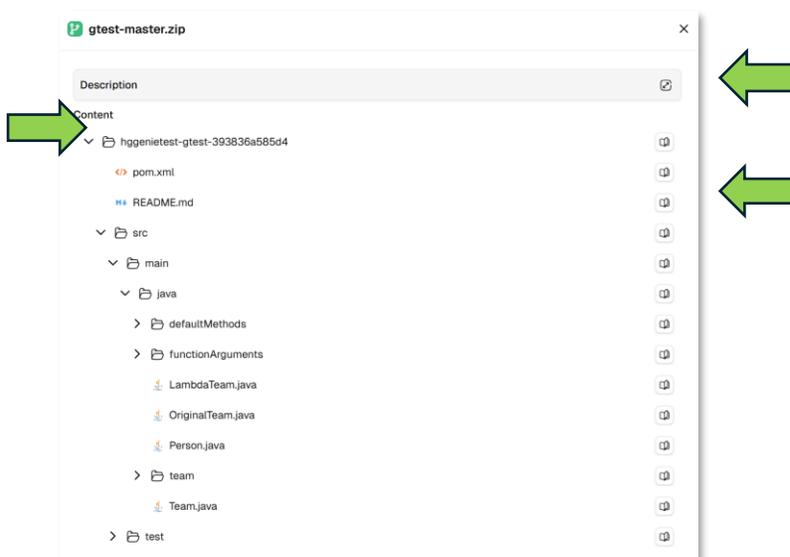
The 'Repositories' widget displays the contents of an integrated code repository (see the connectors section for more information about which repository connectors are available)





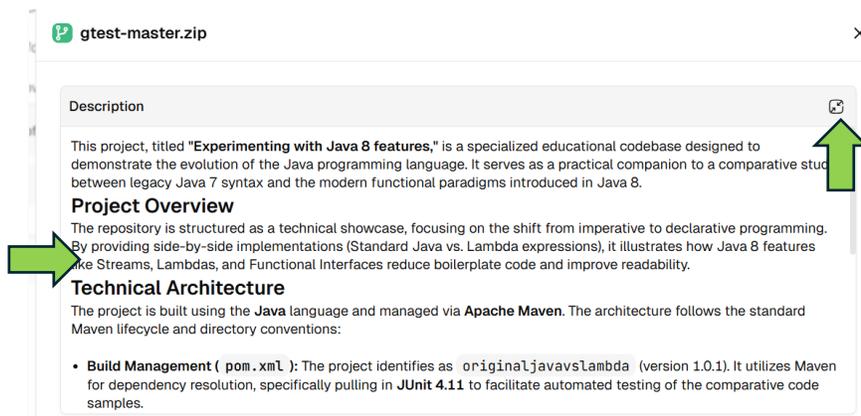
7.5.1 Repository View

Clicking on a repository in the 'Repositories' widget opens a hierarchic view of the repository.



7.5.2 Open Repository Description

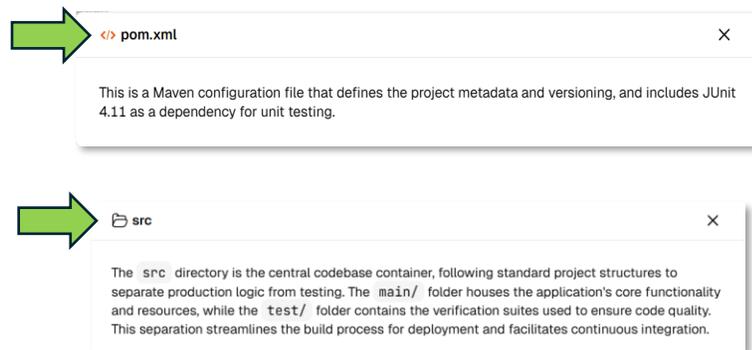
Click on the 'Description' expand button to expand the Description window, showing a 'Description' of the repository.





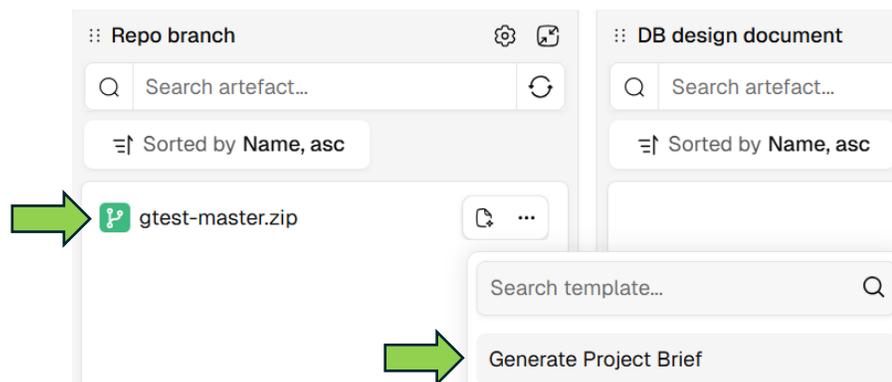
7.5.3 Open Repository File Summary

Click on the 'book' icon next to the repository contents to display a Summary of the item. If the item is a Directory then the 'Directory Summary' is displayed, else if the item is a file then the 'Files Summary' is displayed.



7.5.4 Generate Project Brief

Click on The Generate Actions icon next to a repository and select 'Generate Project Brief' to generate a project brief using the contents of the repository as input.



7.5.5 Generate Epic from Repository

Click on The Generate Actions icon next to a repository and select 'Generate Epic' to generate Epics using the contents of the repository as input.

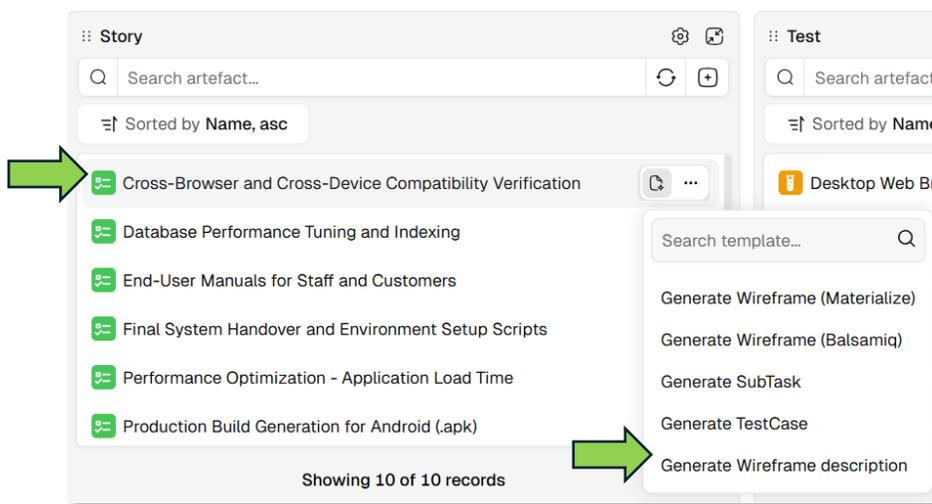


8 Generating More Artefacts

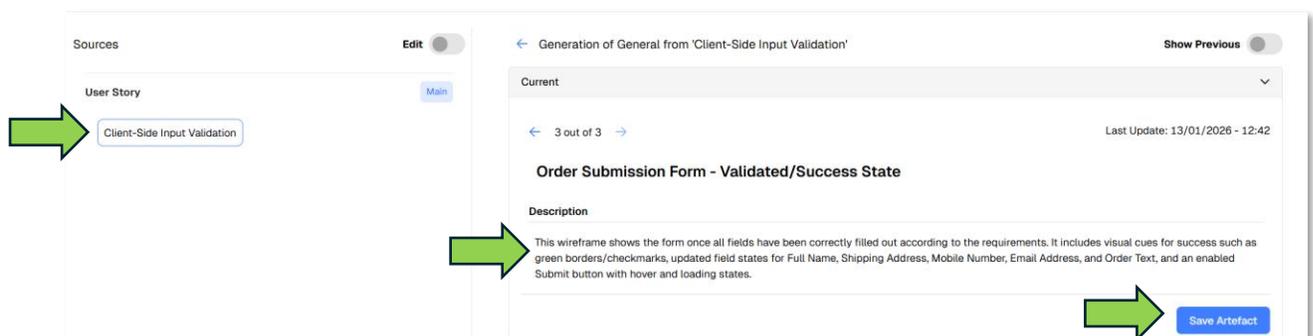
8.1 Generate a Wireframe Description

An 'Epic' in the 'Epics' widget, or a 'Story' in the 'User Stories' widget can be used to generate a 'Wireframe Description' using proprietary AI mechanisms.

1. Go to the 'Widgets' screen.
2. Go to the 'User Stories' widget.
3. Select the 'Generate Artefact' icon to the right of a 'Story'.
4. Select 'Generate Wireframe Description' from the dropdown.

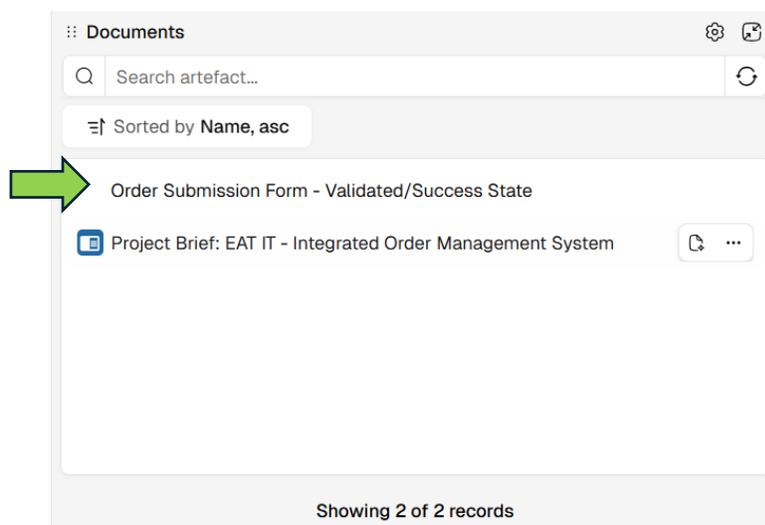


5. The 'Generate' screen is displayed.





6. Observe that the name of the Story that the generation was invoked from is shown as the 'User Story'. This is the source that will be used to seed the AI generation. More source documents can be added by enabling the 'edit' switch and pressing the '+' icon and selecting additional source documents (from the Collateral widget).
7. Press the 'Generate' Button.
8. Once the Wireframe Description has been generated, it is displayed in the right-hand side of the 'Generation Screen'.
9. At this point, there is an option to edit by clicking on the generated text.
10. Once any edits have been committed, press the 'Save Artefact' button.
11. Press the '<-' icon to return back to the widgets screen.
12. Observe that a new document is created in the 'Documents' Widget containing the 'Wireframe Description'.



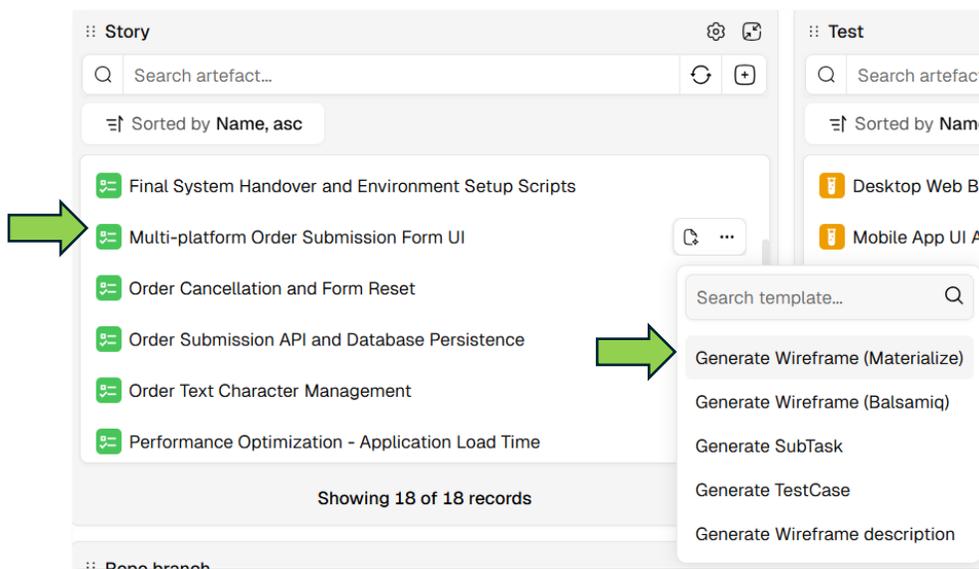
13. Click on the new 'Wireframe Description' document in the 'Documents' widget.
14. The 'Wireframe Description' is displayed in a text window.



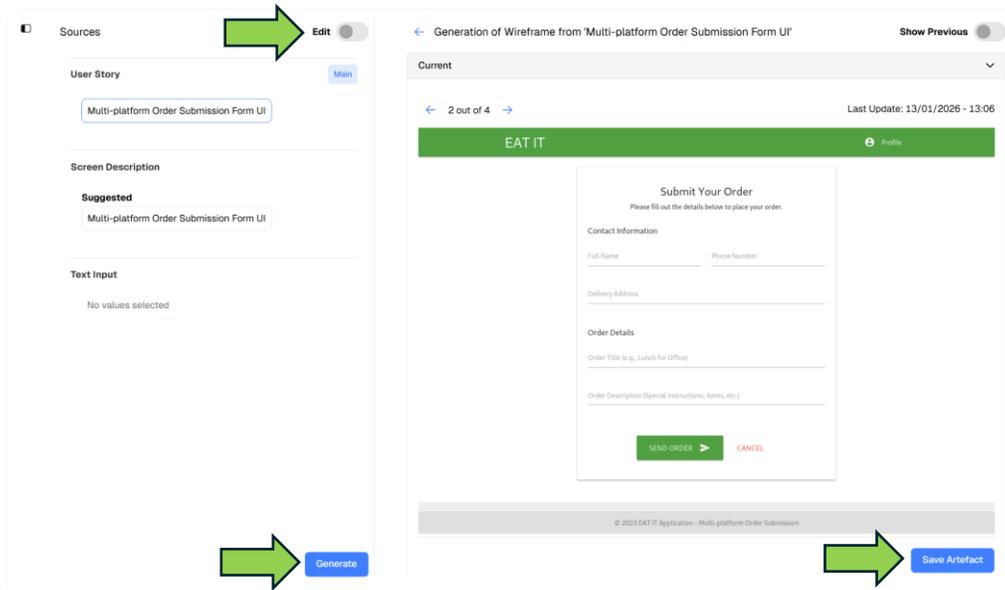
8.2 Generate Wireframes

An 'Epic' in the 'Epics' widget, a 'Story' in the 'User Stories' widget or a 'SubTask' in the 'Sub-Tasks' widget can be used to generate a 'Wireframe' in the various formats (for example Materialize or Balsamiq) using proprietary AI mechanisms.

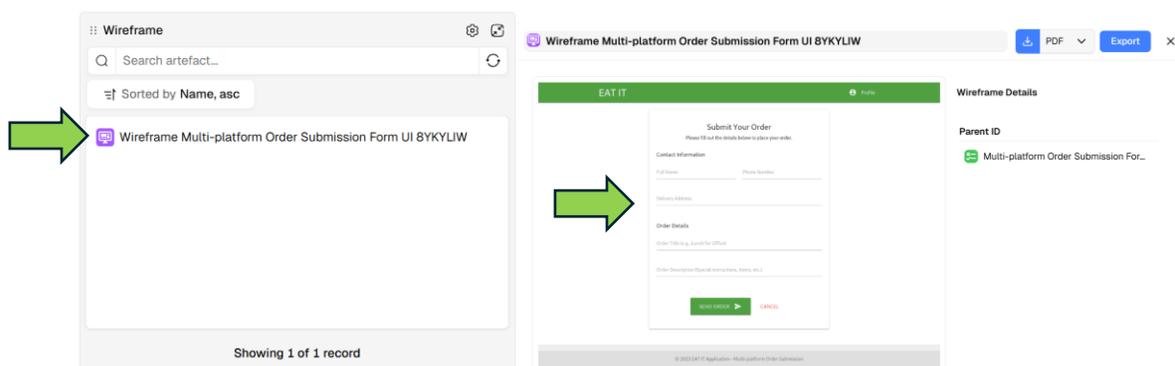
1. Go to the 'Widgets' screen.
2. Go to the 'User Stories' widget.
3. Select the 'Generate Artefact' icon to the right of a 'Story'.
4. Select 'Generate Wireframe (Materialize)' or 'Generate Wireframe (Balsamiq)' from the dropdown.



5. The 'Generate' screen is displayed.
6. Observe that the name of the Story that the generation was invoked from is shown as the 'User Story'. This is the source that will be used to seed the AI generation.
7. More source documents can be added by enabling the 'edit' switch and pressing the '+' icon and selecting additional source documents (from the Collateral widget).



8. Press the 'Generate' Button.
9. ProgramGenie will generate four suggested versions of the Wireframe, once they have been generated they are displayed in the right hand side of the 'Generation Screen'.
10. Use the 'n out of n' indicator at the top of the page to move through the generated suggested Wireframes.
11. At this point there is an option to edit the 'Wireframe' if required by clicking on wireframe itself.
12. Once any edits have been committed press the 'Save' button.
13. Press the 'Save' button to commit each generated Wireframe to the 'Wireframes' widget.
14. Press the '<' icon to return back to the widgets screen.

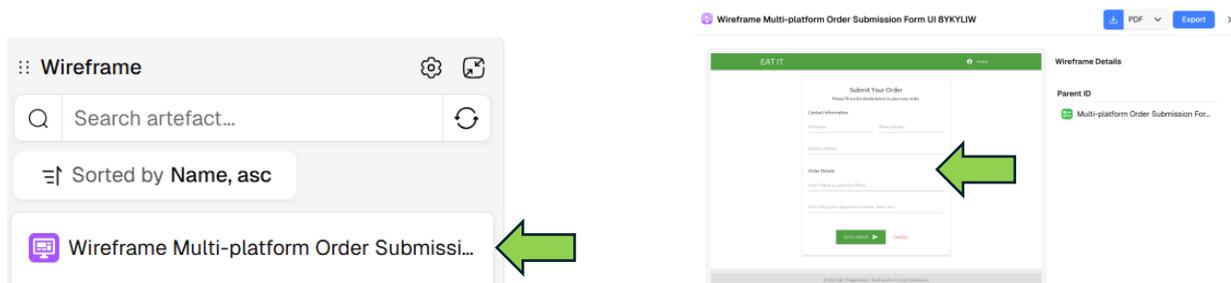


15. Observe that a new Wireframe is created in the 'Wireframes' widget.
16. Click on the new 'Wireframe' in the 'Wireframes' widget.
17. The 'Wireframe' is displayed in a window.

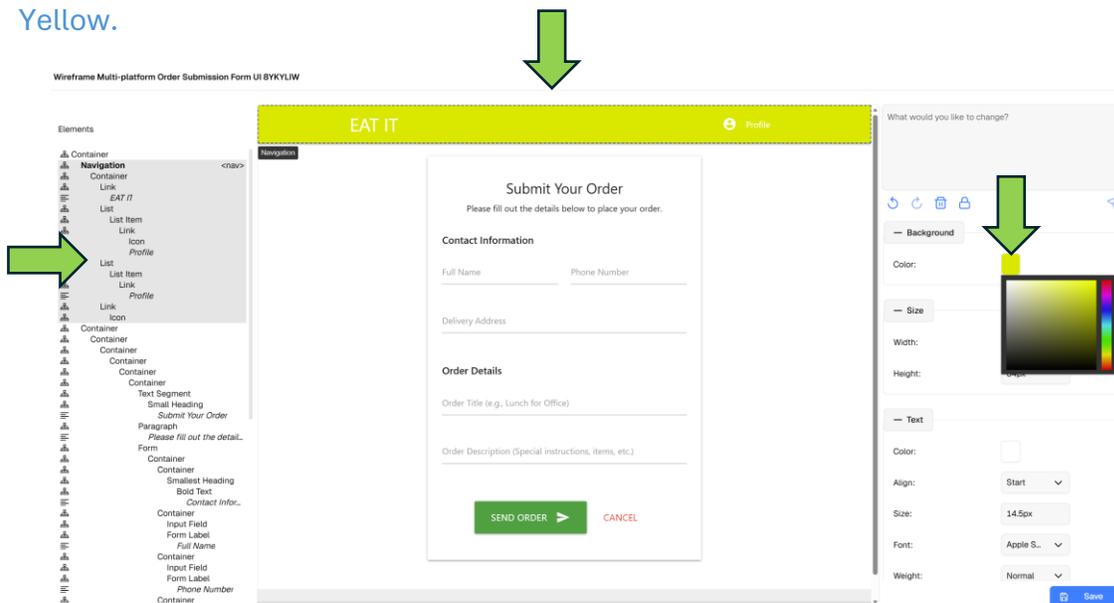


8.3 Edit a Wireframe

A Wireframe can be edited, either from the generate screen or after it has been saved by clicking on it in the Wireframes widget.



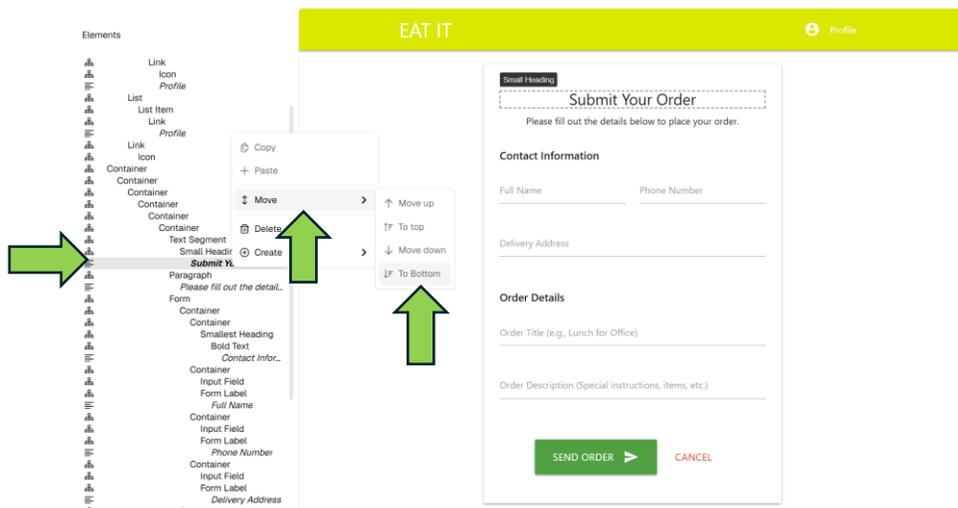
1. Open the wireframe editor (from Generate screen or Wireframes widget)
2. On the left side is a list of all elements
3. On the right side is a list of all attributes for any selected element
4. Click on one of the elements in the Elements list and set the Background Colour to Yellow.



5. Observe that the background color of the selected element is set to yellow and the Save button has been enabled.
6. Press Save to commit the change and observe the change has been saved.
7. Select Edit to re-enter edit mode.

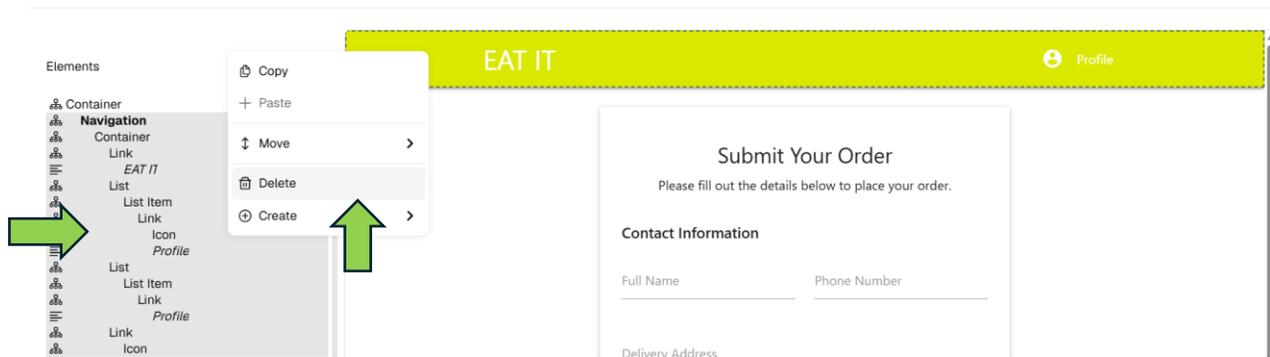


8. Right click on the same yellow element.
9. Click on 'Move' and 'Move to the Bottom'.

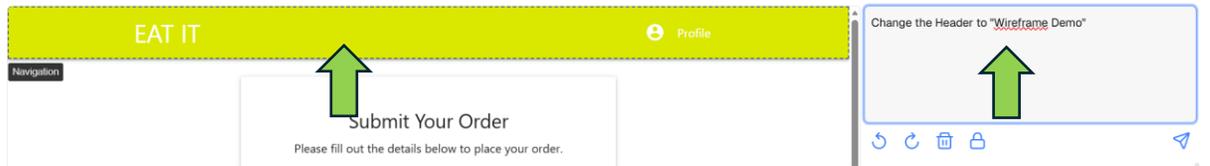


10. Observe that the selected item moves to the end (remember this functionality will depend on the type of elements you are manipulating).
11. Right-click on the same yellow element and select delete.

Wireframe Multi-platform Order Submission Form UI 8YKYLIIW

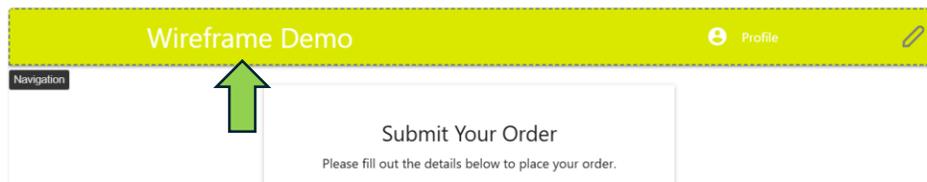


12. Observe that the Element is deleted.
13. Press Save to commit the change and observe the change has been saved.
14. Select Edit to re-enter edit mode.
15. Select the Header element
16. Enter a prompt to edit the header value. For example, "Change the Header to WIREFRAME DEMO"



17. Press the execute prompt arrow icon

18. Observe that the header title has been changed



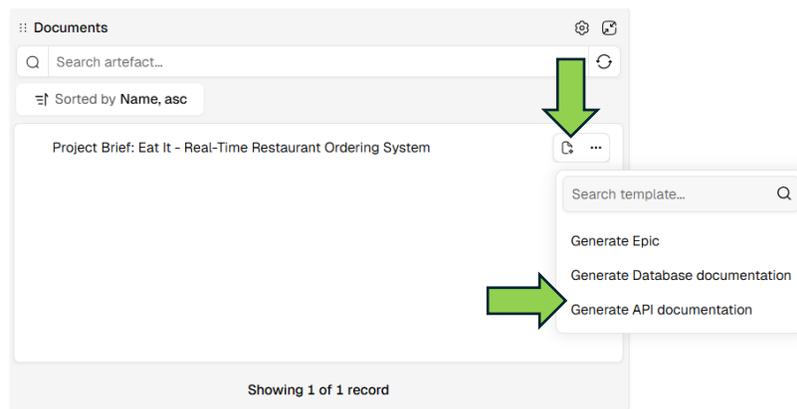
19. Press Save to commit the change and observe the change has been saved.



8.4 Generate API Documentation

A 'Project Brief' in the 'Documents' widget can be used to generate 'API Documentation' using proprietary AI mechanisms. An 'API Document' is a set of human-readable instructions for using and integrating with an API, including detailed information about an API's available endpoints, methods, resources, authentication protocols, parameters, and headers, as well as examples of common requests and responses.

1. Go to the 'Widgets' screen.
2. Go to the 'Documents' widget.
3. Select the 'Generate Artefact' icon to the right of a generated 'Project Brief'.
4. Select 'Generate API Documentation' from the dropdown.



5. The 'Generate' screen is displayed.
6. Observe that the name of the Project Brief that the generation was invoked from is shown as the 'Project Brief'. This is the source that will be used to seed the AI generation. More source documents can be added by enabling the 'edit' switch and pressing the '+' icon and selecting additional source documents (from the Collateral widget).
7. Press the 'Generate' Button.
8. It may take a short period of time for the document to be created.
9. Once the API Document has been generated it is displayed in the right hand side of the 'Generation Screen'.
10. There is an option to edit the currently displayed 'API Document' if required, by clicking on text.



Sources

Project Brief

Project Brief: Eat It - Real-Time Restaurant Ordering System

User Stories

Suggested

Real-time Sync Trigger (WebSocket Emission)

Suggested

Character Count & Length Constraints

Suggested

Order Submission API & SQL Persistence

Suggested

Order Cancellation Logic

Suggested

Customer Dashboard Layout & Profile Display

Suggested

Free-form Order Text Input

Suggested

Order Submission Action (Send)

Suggested

Post-Submission Success Feedback

Suggested

Order Validation & Error Handling

Generate

Generation of Api-document from 'Project Brief: Eat It - Real-Time Restaurant Ordering System'

Current

1 out of 1

Last Update: 14/01/2026 - 16:12

API Documentation: Eat It - Real-Time Restaurant Ordering System

Description

API documentation for the Eat It restaurant ordering system, covering authentication, user profile management, and order lifecycle operations.

Content

API Documentation: Eat It - Real-Time Restaurant Ordering System

This document outlines the API endpoints for the "Eat It" system, designed to facilitate real-time order submission and management.

Table of Contents

- 1. Authentication & User Profile
 - [POST /api/auth/register](#)
 - [POST /api/auth/login](#)
 - [GET /api/users/me](#)
- 2. Orders
 - [POST /api/orders](#)
 - [GET /api/orders](#)
 - [PATCH /api/orders/{id}/status](#)
 - [DELETE /api/orders/{id}](#)

Authentication & User Profile

Handles user registration, secure login, and retrieval of profile information for the order submission screen.

POST /api/auth/register

Registers a new user (Customer or Staff) in the system.

Request Contract (JSON Body):

Name	Type	Description
name	string	Full name of the user.

11. Once any edits have been committed press the 'Save' button.
12. Press the '<' icon to return back to the widgets screen.
13. Observe that the saved 'API Document' is displayed in the 'Documents' widget.
14. Click on a 'API Document' in the 'Documents' widget. The 'API Document' is displayed in a text window.
15. Use the Edit button to make further changes.
16. Use the download Icon to download in PDF or Word format.

Documents

Search artefact...

Sorted by Name, asc

API Documentation: Eat It - Real-Time Restaurant Ordering System

Project Brief: Eat It - Real-Time Restaurant Ordering System

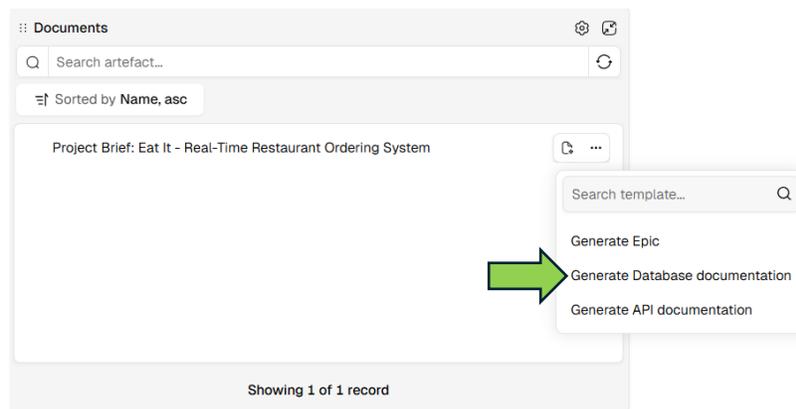
Showing 2 of 2 records



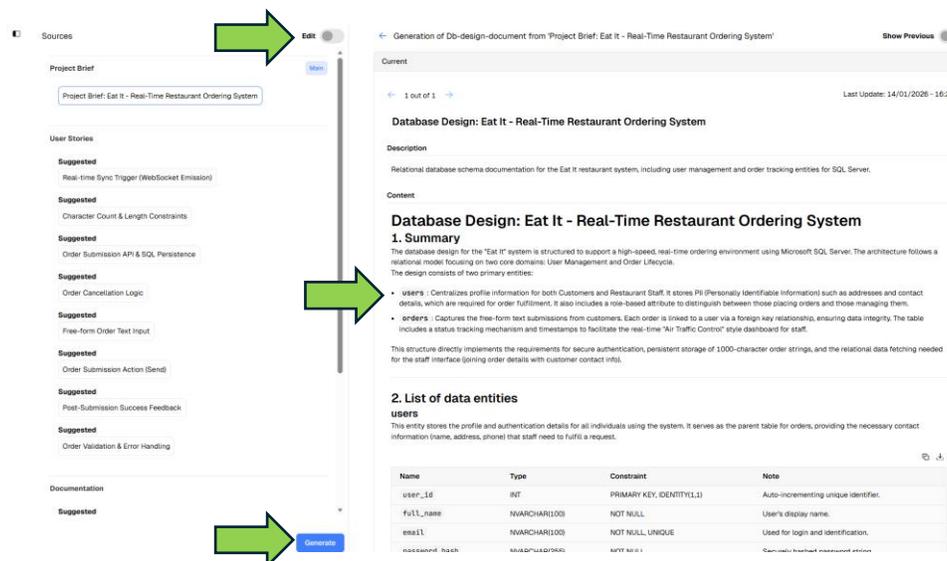
8.5 Generate Database Documentation

A 'Project Brief' in the 'Documents' widget can be used to generate 'Database Documentation' using proprietary AI mechanisms.

1. Go to the 'Widgets' screen.
2. Go to the 'Documents' widget.
3. Select the 'Generate Artefact' icon to the right of a generated 'Project Brief'.
4. Select 'Generate Database Documentation' from the dropdown.

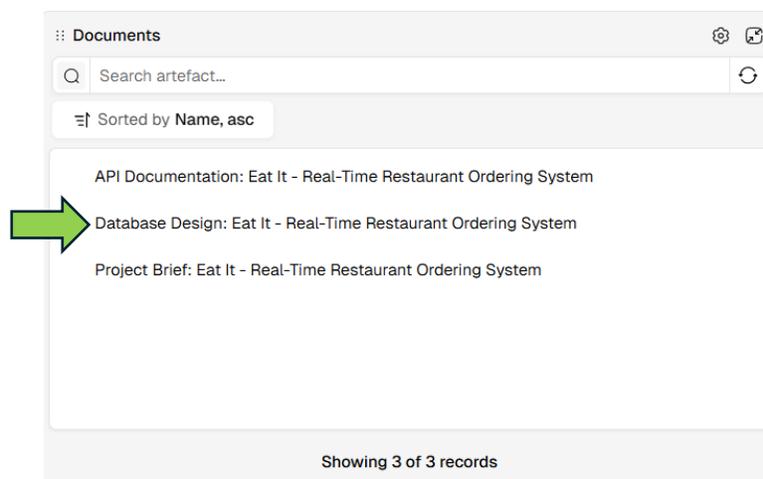


5. The 'Generate' screen is displayed.
6. Observe that the name of the Project Brief that the generation was invoked from is shown as the 'Project Brief'. This is the source that will be used to seed the AI generation. More source documents can be added by enabling the 'edit' switch and pressing the '+' icon and selecting additional source documents (from the Collateral widget).
7. Press the 'Generate' Button.
8. It make take a short period of time for the document to be created.





9. Once the Database Document has been generated it is displayed in the right hand side of the 'Generation Screen'.
10. There is an option to 'Edit' the currently displayed 'Database Document' if required.
11. Once any edits have been committed press the 'Save' button.
12. Press the '<-' icon to return back to the widgets screen.
13. Observe that the saved 'Database Document' is displayed in the 'Documents' widget.
14. Click on a 'Database Document' in the 'Documents' widget.
15. The 'Database Document' is displayed in a text window.
16. Use the Edit button to make further changes.
17. Use the download Icon to download in PDF or Word format.



8.6 Generate Individual Artefacts

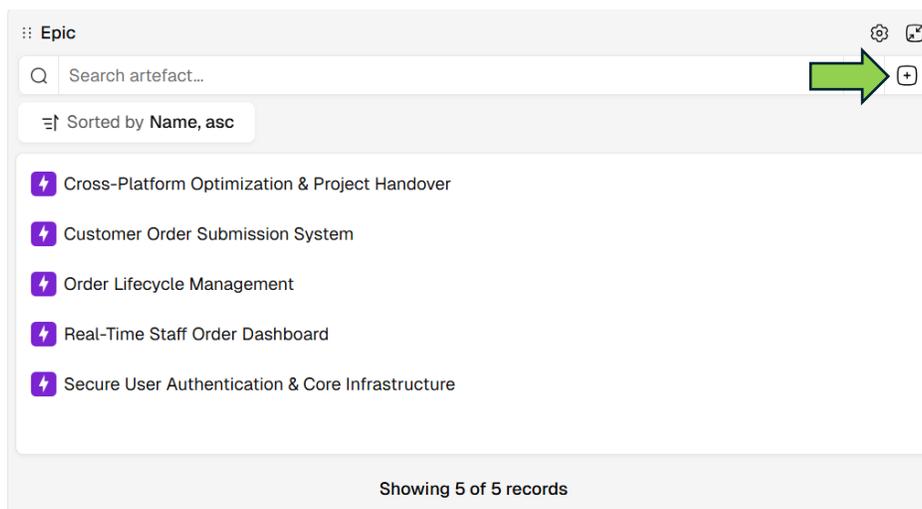
Individual artefacts can be created from widgets, by clicking on the + icon (where available) in a widget on the dashboard.

8.6.1 Generate a manual artifact from the artifact Widget

To generate an individual manual artefact (where the user enters the content), carry out the following example steps. In this example, we will create a new Epic

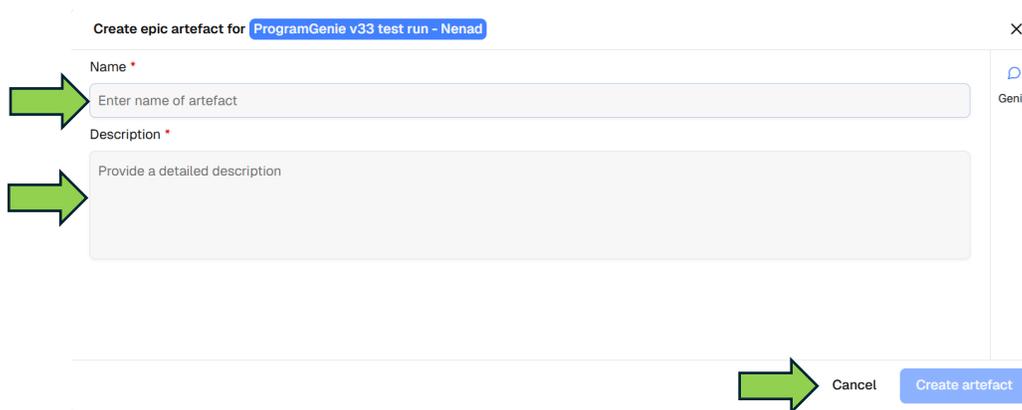


1. Go to the 'Epics' widget.



2. Click on the + icon.

3. The create Epic Artefact window appears

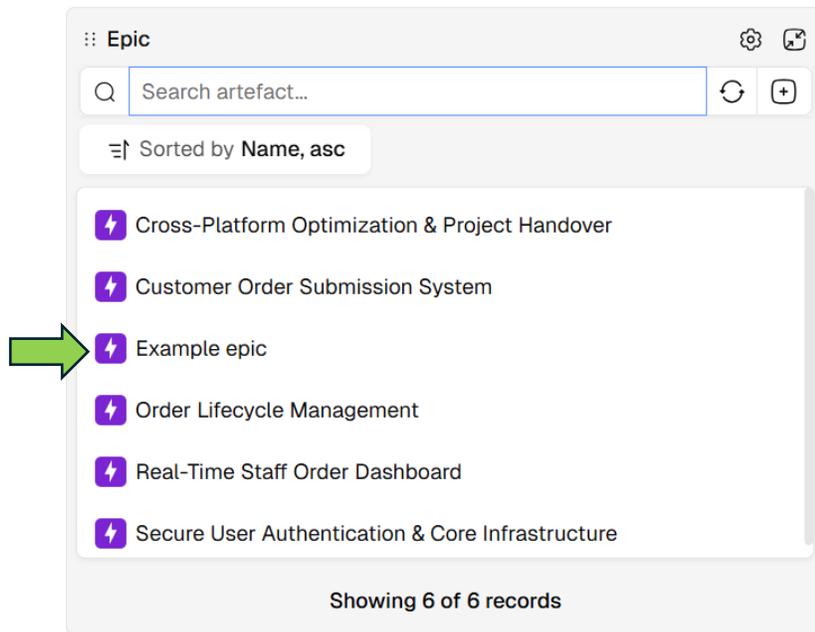


4. Enter a name for the epic that will be created

5. Enter the epic content

6. Press create artefact button

7. The new Epic is created and displayed in the Epic widget

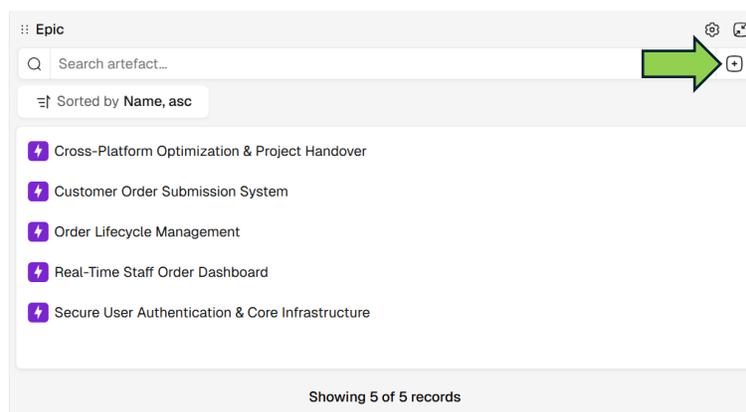


In the same way you can create any other type of artefacts.

8.6.2 Generate an artefact from the artefact Widget using ProgamGenie AI

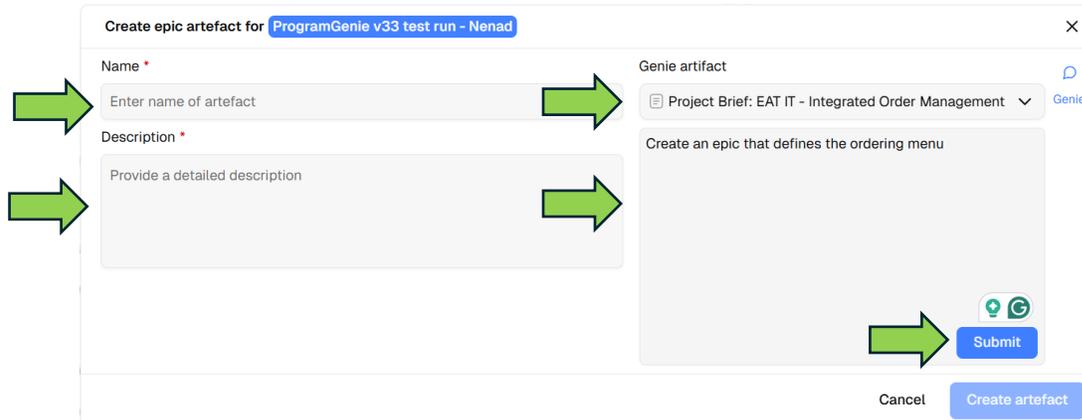
To generate an individual artifact using ProgramGenie AI to generate the content of the artefact, carry out the following steps. In this example we will create a new Epic, but this identical steps are used for any artifact type.

1. Go to the 'Epics' widget.
2. Click on the + icon.

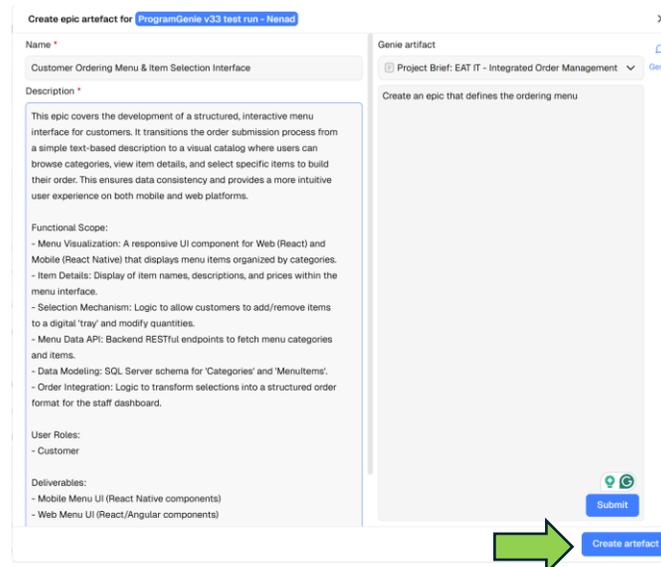




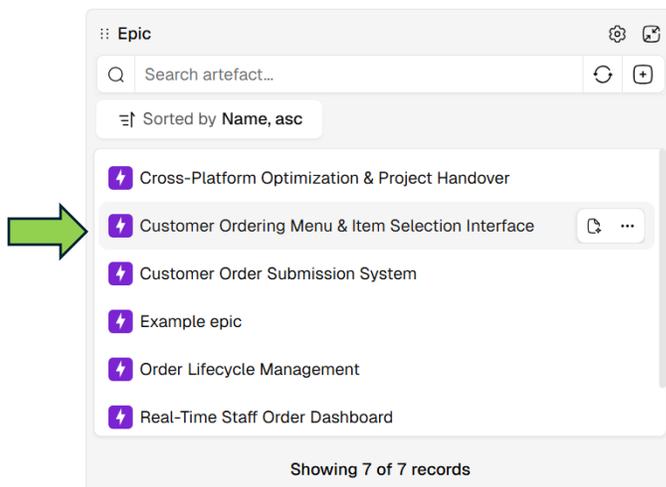
3. The create Epic Artefact window appears
4. Click on the Genie icon on the right hand side to open the prompt area



5. Select the Genie artefact you wish to use as the source for the epic generation
6. Enter a prompt to generate the content of the Epic, for example, “Create an epic that defines the requirements for an Audit Trail”
7. Press submit



8. The generated Epic content is displayed in the description area
9. The title has also been generated
10. Both the title and the content can be edited
11. Press the Create Artefact



12. The new Epic is created and displayed in the Epic widget



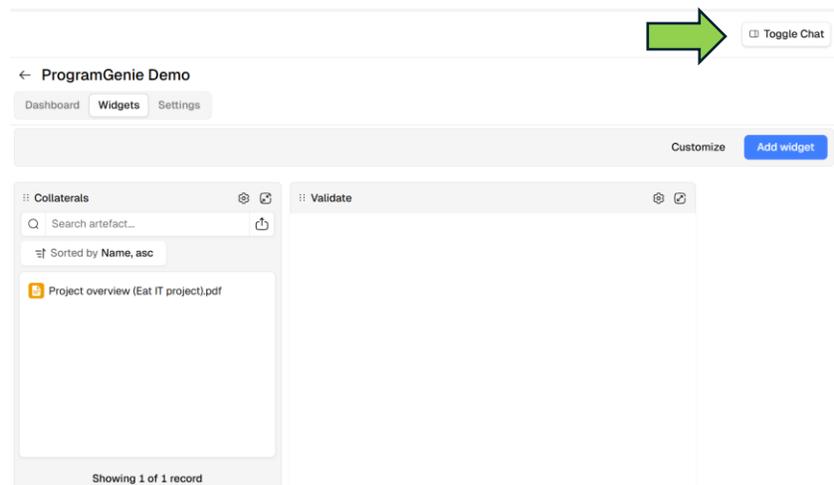
9 ProgramGenie Chat

9.1 ProgramGenie Chat Area

The ProgramGenie Chat area is an important ProgramGenie function that can be opened as a sidebar at the right of the screen. It gives the user similar functionality found in the most popular Large Language Model (LLM) based AI tools, but with the added bonus of having access and understanding of all the artefacts added to a project.

9.1.1 Toggle Chat

The ‘Toggle chat’ button is displayed in the right hand corner of the ProgramGenie Application. Pressing this button opens the ProgramGenie Chat sidebar window. Clicking again on the ‘Toggle chat’ button hides the sidebar.



9.1.2 Prompt Field

Enter a prompt (question/statement) in the Prompt Field and press return (or the adjacent arrow icon), for example “What are the most important Epics in my project”. A busy icon will be shown and output will start to appear in the Output Screen.

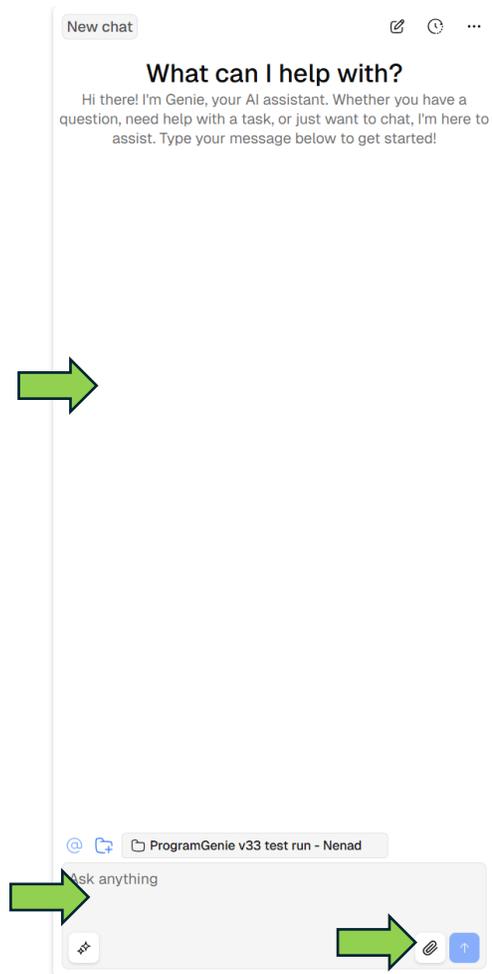


9.1.3 Add File

Click on the (paperclip) Add File icon to ingest a file into the current Chat session. This file can then be referred to as part of a prompt. For example “List the top 5 functions in the added document”.

9.1.4 Output Screen

The Output Screen displays the results of the prompt. A scrollbar is available to the right of the text to allow the results to be scrolled through. Standard windows select and copy functions are available. Remember to start a new chat if you don't want to continue the context that you have created for the next prompt.





9.1.5 New Chat

Click the New Chat (Pen) icon to start a new chat session. This will clear the Chat Output Window and add the new session to the Chat History.

The screenshot shows a chat interface with a chat history item and a new chat input field. The chat history item is titled "EAT IT Project Overview" and contains a message from "11 minutes ago" asking "can you tell me something about this project?". The response is a detailed overview of the "EAT IT" system, including its core features and technical stack. The new chat input field is at the bottom, with a placeholder text "Ask a follow up question" and a "New Chat" icon (pen) on the left. The chat history item has a "New Chat" icon (pen) on the right, which is highlighted with a green arrow. The chat history item also has a "Copy" icon and a "Share" icon on the right, which are also highlighted with green arrows. The chat history item has a "More" icon (three dots) on the right, which is also highlighted with a green arrow. The chat history item has a "Close" icon (X) on the right, which is also highlighted with a green arrow.

EAT IT Project Overview

can you tell me something about this project?

11 minutes ago

Based on the project details for **EAT IT (ProgramGenie Demo)**, here is an overview of the system:

Project Overview: EAT IT

EAT IT is an integrated order management system designed to modernize how restaurants handle customer orders. It replaces manual processes with a real-time digital bridge between customers and kitchen staff.

Core Features

The system is divided into two main interfaces:

- 1. Customer Order Submission (Web & Mobile)**
 - Allows customers to register and log in securely.
 - Provides a form to enter contact details (Name, Address, Phone) and specific order instructions.
 - Includes features like profile auto-fill and immediate submission feedback.
- 2. Real-Time Staff Dashboard**
 - A live screen for kitchen/service staff that updates instantly (sub-500ms latency) when a new order arrives.
 - Uses **WebSockets** so staff don't need to refresh the page.
 - Allows staff to mark orders as "Done" or delete them.

Technical Stack

@ ProgramGenie Demo

Ask a follow up question



9.1.6 Chat History

Click the Chat History (Clock) icon to view a list of previous chats. 

9.1.7 Delete Chats

Click the 3 dots icon and then select Delete Chats to delete the chat history 

9.1.8 Copy

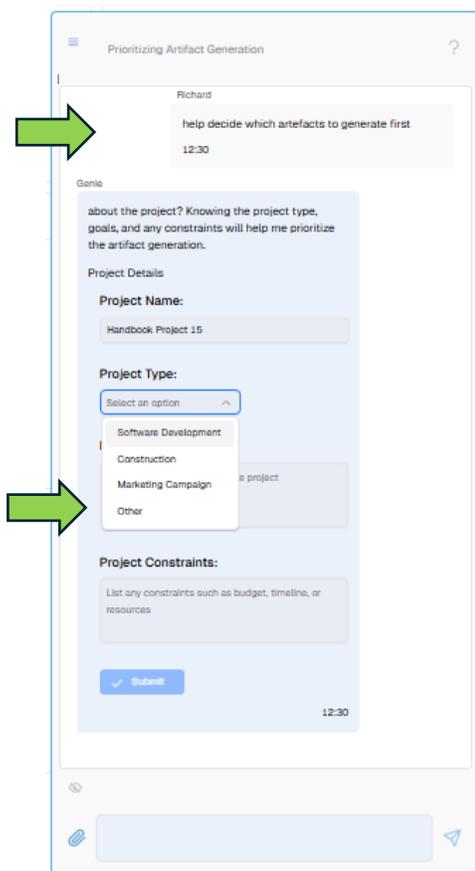
Click the 2 squares icon to copy the content of the output screen to the clipboard 

9.1.9 Duplicate Message

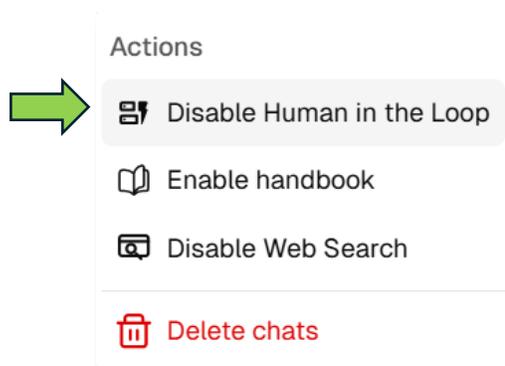
Click on the duplicate message (+ in a square) icon, to copy the last prompt back to the prompt area. 

9.1.10 Human in the Loop

The 'Human In The Loop' ProgramGenie Chat feature, helps the user refine their prompt journey by asking the user to enter data and make decisions in order to refine and focus a prompts response. Questions are triggered when the user writes prompts that request chat to 'Help' or 'decide' something that has a potentially broad response that could be refined if the prompt had more context. For example, in the diagram below, the user has entered (in a new empty project) the prompt "Help decide which artefacts to generate first". ProgramGenie chat responded with a structured response requesting that the user provides more context entering data into fields and selecting from dropdowns menus.



The Human in the Loop function can be disabled by clicking on the three dots icon and then selecting Disable Human in the Loop.

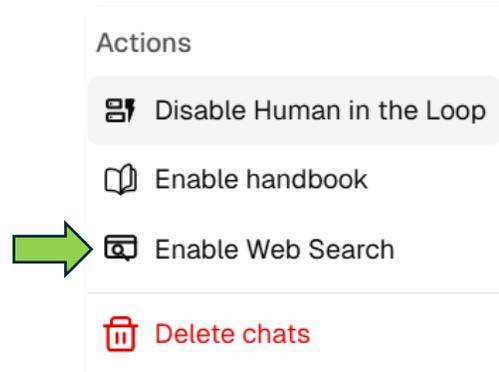


9.1.11 Web Search

"The 'Web Search' Genie Chat feature extends the user's informational horizon by allowing Genie to access the web and search for answers, utilizing a broader context than the



information kept strictly within Genie. This functionality is triggered when the user enables the feature to answer queries that require real-time data, external documentation, or global market insights.



9.2 Basic Prompting

Prompting in ProgramGenie chat works in exactly the same fashion as you may have come across in other commercial AI systems. The user types in a statement and the AI constructs a response which is output as text. However, one key feature of using ProgramGenie Chat is the fact that all artefacts and collateral in ProgramGenie are in the context of the AI and thus can be directly referenced in a prompt. For example “Summarize all the epics in this project”.

9.2.1 Descriptive Prompting

Descriptive prompts help create expanded content from a simple concept, you can direct the content by defining what to include. For example ***"Describe a Project Overview for launching a new mobile app, including its business goals and scope."***

9.2.2 Instructional Prompting

Instructional prompts are useful when breaking down work into manageable pieces. For example, ***"Create a list of the stories needed for implementing user login with email and password."***

9.2.3 Comparative Prompting

Comparative prompts help evaluate differences across artefacts and collateral in ProgramGenie. For example, ***"Compare stories with test cases and identify gaps in test coverage"***.



9.2.4 Creative Prompting

Creative prompts direct the AI to generate structured, reusable content. These prompts are perfect for building out QA documentation and automation scripts. For example, ***"Write a test case for verifying password reset functionality in Jira."***

9.2.5 Analytical Prompting

Analytical prompts are useful to gain clarity and structure across artefacts. For example, ***"Summarize the key differences between epics and stories in this project."***

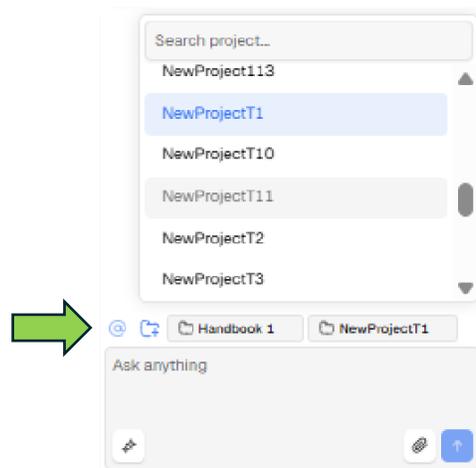
9.3 Onboarding Prompting

The ProgramGenie Chat can help when with analysis, for example when onboarding a new or existing project. Below are some ideas for prompts, remember virtually anything is possible (as long as it outputs as text).

1. ***"Create a data model for this project"***
2. ***"Create an ASCII diagram of the data model for this project"***
3. ***"Create a DDL script needed for creating this model"***
4. ***"I what to change the patient booking system what stories and test are impacted"***

9.4 Adding Projects

The context for the AI Chat function is restricted to the current project, however, more projects can be added by clicking on the 'Add projects to context' icon and then select a project from the displayed list. Added projects are displayed in the horizontal project list and can be removed by clicking on the X icon next to them.



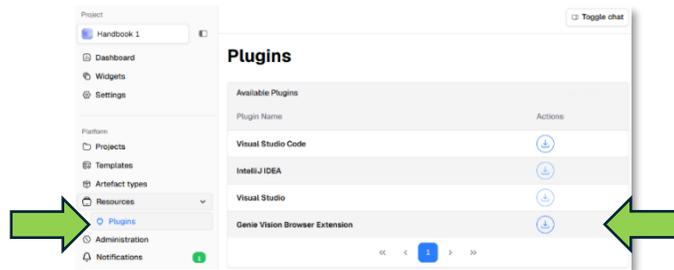


10 Genie Vision

The 'Genie Vision' is a special chat function that allows the ProgramGenie chat window to interact directly with live web pages. In order to use the functionality the user must install the 'Genie Vision Browse Extension' into their browser.

10.1 Download the Browser Extension

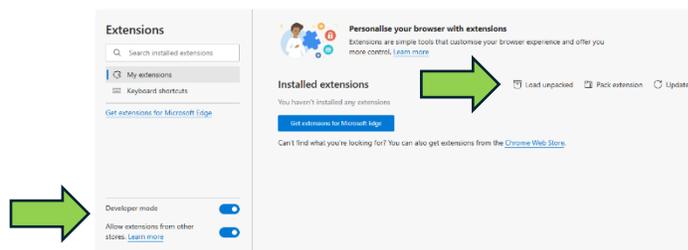
To download the 'Genie Vision Browser Extension', on the left hand sidebar select Resources then Plugins.



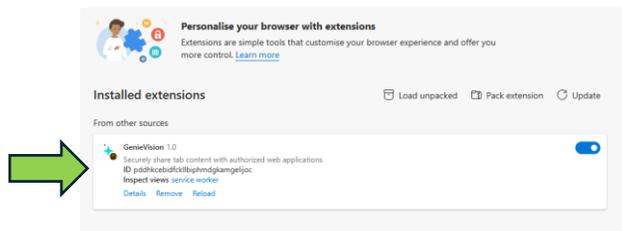
10.2 Install for Microsoft Edge

Follow the instructions below to install the 'Eye of ProgramGenie' for Microsoft Edge.

- 1) In Microsoft Edge enter "Edge: //extensions".
- 2) The Installed extensions screen appears.



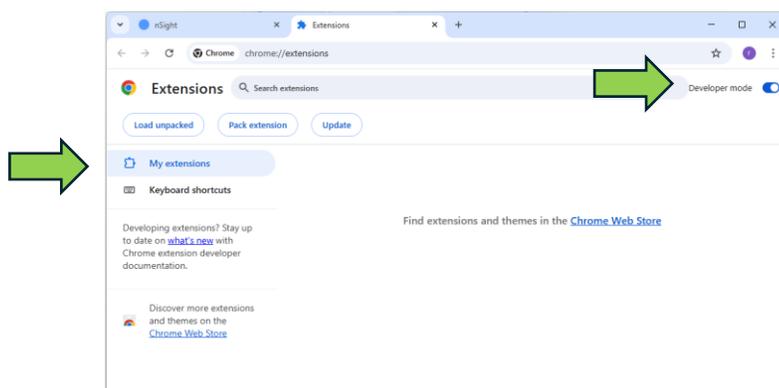
- 3) Select 'Developer mode on' in the 'Extensions' sidebar.
- 4) Click on the 'Load unpacked', and observe the File selection window is displayed.
- 5) Navigate to the 'GenieVision.zip' distribution folder downloaded from the plugins screen.
- 6) Select at the folder level (not its contents). Observe that the 'ProgramGenieVision' distribution has been installed.



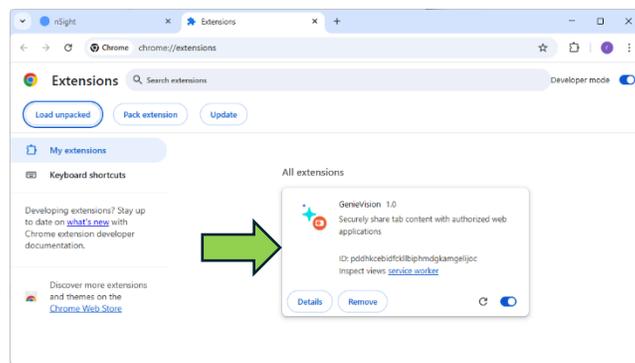
10.3 Install for Google Chrome

The Output Screen displays the results

- 1) Select 3 dots top right of the browser.
- 2) Select extensions -> Manage extensions.
- 3) The extensions screen is displayed.



- 4) Turn on developer mode (right corner).
- 5) Select 'load unpacked'.
- 6) File selection window is displayed.
- 7) Navigate to the GenieVision distribution folder that was downloaded from the plugins screen, and select the Dist folder. (note: you may need to unzip the GenieVision file first)
- 8) Select at the folder level (not its contents).
- 9) Observe that the 'GenieVision' distribution has been installed.

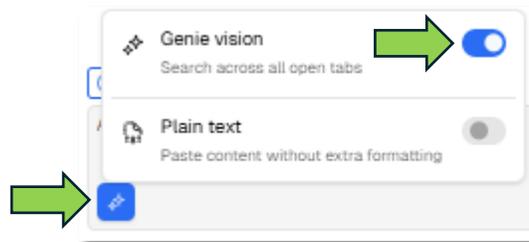




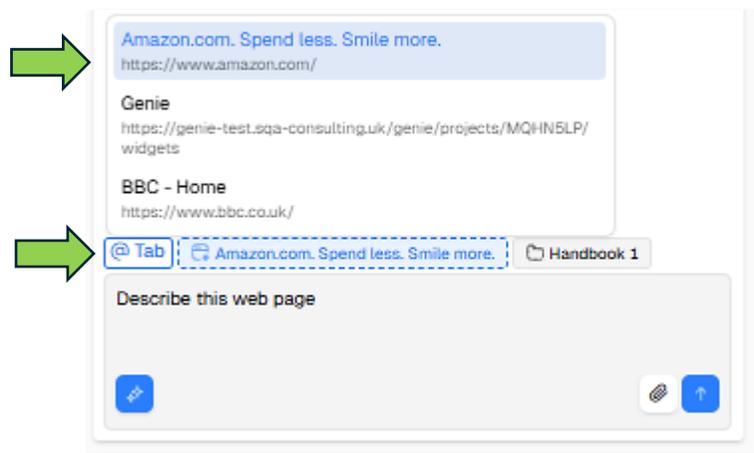
10.4 Basic Operation

ProgramGenie Vision enables ProgramGenie chat to interact with other open web pages in the same browser.

- 1) Click on the GenieVision icon
- 2) Enable GenieVision



- 3) Press the '@Tab' button.

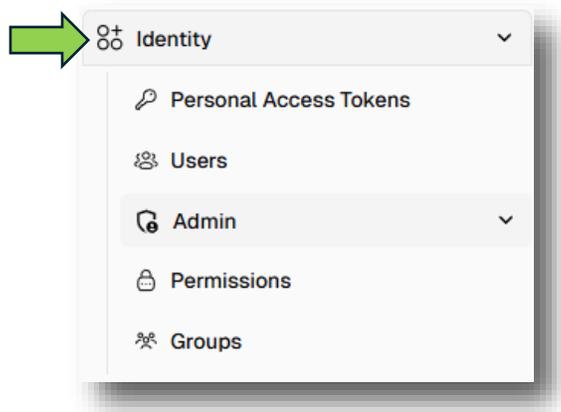


- 4) Observe the menu displays the addresses of the pages in each of the tabs in the browser.
- 5) Select the page you are interested in.
- 6) The page is now available content for use in chat.
- 7) Try entering the prompt “Create a list of the test cases required to test this”.



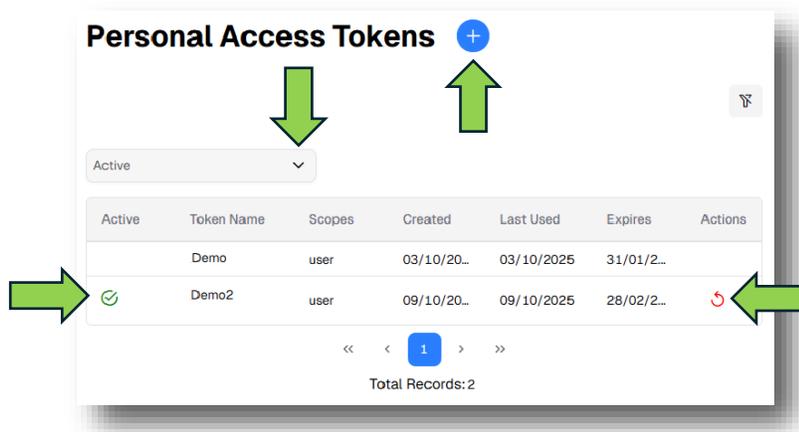
11 Identity

The 'Genie The ProgramGenie Identity function is used to manage users, groups & permissions and to generate Access Tokens. The Identity function is found on the far left of the ProgramGenie sidebar. User will need Administrator privileges.



11.1 Personal Access Tokens

Click on the Personal Access Token menu to open the Personal Access Token screen.



11.1.1 Add Token

Use the Add token button to create a new token for use with Program Genie plugins.

- 1) Press the Add Token (+) icon, to open the 'Add New Token' popup.
- 2) Enter a Name for the token in the Token Name field
- 3) Enter an expiration date for the token.



Add New Token [X]

Token Name *

Expiration Date * [Calendar Icon]

Cancel Save

- 4) Press Save
- 5) Press the Copy Token to copy your token, then save it in a safe/secure place, you will not be able to see this token again.
- 6) Observe the new token in the Personal Access Tokens screen.

Personal Access Tokens + [Filter Icon]

Active [Dropdown Arrow]

Active	Token Name	Scopes	Created	Last Used	Expires	Actions
	MyToken	user	20/10/2025		31/10/2025	
	Demo	user	03/10/2025	03/10/2025	31/01/2026	
	Demo_2	user	09/10/2025	09/10/2025	28/02/2026	

<< < 1 > >>

Total Records:3

11.1.2 Filter Tokens

Click on the Filter area to display a menu that allows you to select whether you see All, Active or Inactive tokens in the Personal Access Tokens area. Select All to clear filter.

Active [X] [Up Arrow]

[Search Bar]

All

Active

Inactive



11.1.3 Active Token

Click on the Active Token (green tick) icon to see detailed information.

The screenshot shows the 'Personal Access Tokens' interface. On the left, there is a table with columns: Active, Token Name, Scopes, Created, and Last Used. The first row has a green tick icon, 'MyToken', 'user', '20/10/2025', and is empty. On the right, a detailed view for 'MyToken' is shown with fields: Token Name (MyToken), Scopes (user), Created (2025-10-20T16:03:11.444+00:00), Last Used, and Expires (2025-10-31T00:00:00.000+00:00). A green arrow points from the green tick icon in the table to the detailed view panel.

11.1.4 Revoke Token

Click the Revoke Token (red arrow) to remove the tokens access. The active green tick icon will be removed and the Token will no longer work (permanently).

The screenshot shows the 'Personal Access Tokens' interface with a table containing three rows. The columns are: Active, Token Name, Scopes, Created, Last Used, Expires, and Actions. The first row has a green tick icon, 'MyToken', 'user', '20/10/2025', '03/10/2025', '31/10/2025', and a red revoke icon. The second row has a green tick icon, 'Demo', 'user', '03/10/2025', '03/10/2025', '31/01/2026', and a red revoke icon. The third row has a green tick icon, 'Demo_2', 'user', '09/10/2025', '09/10/2025', '28/02/2026', and a red revoke icon. A green arrow points to the red revoke icon in the 'Actions' column for the 'MyToken' row.

Distribution

Name	Role	Date	Version
RW	Author	17-Sep-2025	1.00



Version History

Name	Summary of Changes (List the reason for each version of the document)	Revision Date	Version
RW	V2.9 Initial version	17-Sep-2025	1.00
RW	V3.0 Jira Id changes	25-Sep-2025	1.01
RW	V3.0 Remove progress window references	01-Oct-2025	1.02
RW	V3.1 Add AI Edit	03-Oct-2025	1.03
RW	V3.1 Reformat as Training Handbook	07-Oct-2025	1.04
RW	V3.1 Reformat colors	28-Oct-2025	1.05
RW	V3.2 Review Comments	29-Oct-2025	1.06
RW	V3.2 More Review Comments	10-Nov-2025	1.07
RW	V3.2.2 Add Validation Widget	25-Nov-2025	1.08
RW	V3.2.2 typos	12-Dec-2025	1.09
NJ	V3.3 New screens and features added	16-Jan-2026	1.10