Purgo ai

User document

Introduction

Purgo AI is an advanced AI platform designed to help you build, test, and deploy data applications quickly and efficiently into production. It streamlines the process of creating data applications, providing a seamless way to tackle complex tasks such as migrations, ETL workflows, and more.

Problem

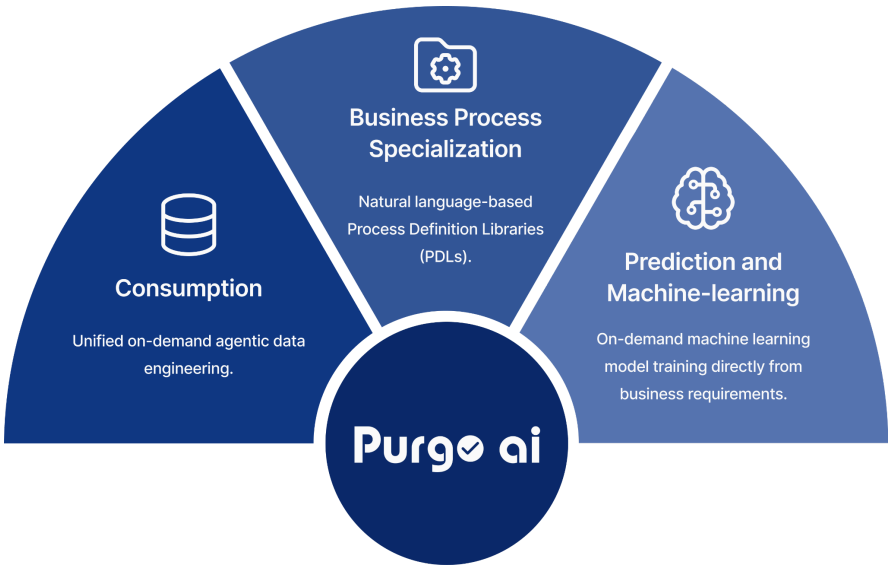
Building data applications for use cases like migration and ETL can be both time-consuming and costly. Engineers face numerous challenges, including:

- Understanding all the systems involved
- Writing code to execute the task
- Generating sample data for testing
- Creating tests to validate the code
- Ensuring the tests run successfully
- Finally, deploying the code into production

Although leveraging LLMs and AI code generators can help, it remains a challenge if engineers don't know the right questions to ask or the context to provide. This often results in several iterations across different models, extending the time to completion.

Solution

With **Purgo AI**, all these complexities are built into a unified framework, drastically accelerating your production process from weeks to mere minutes. Purgo AI understands your systems, identifies the relevant context, generates the right prompts, and produces the assets you need to complete your workload. Let Purgo AI accelerate your data app development today.



Getting Started

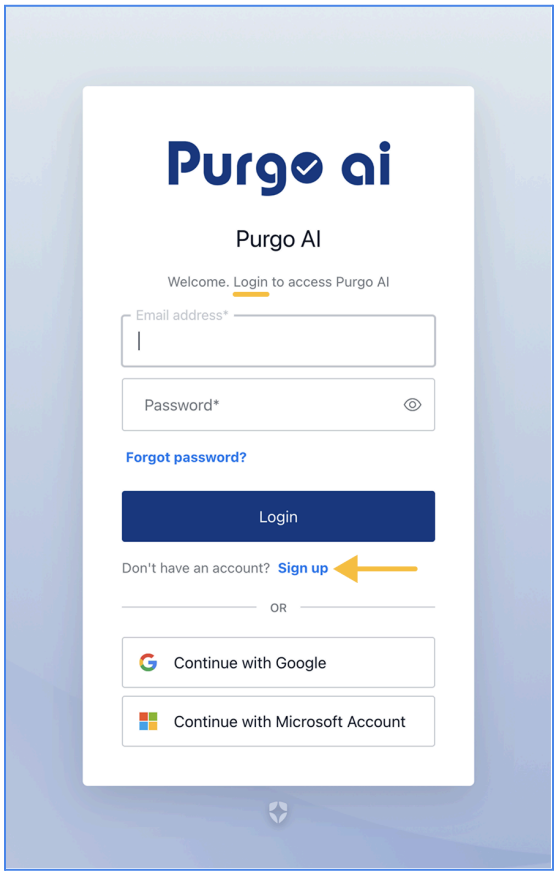
Welcome to Purgo AI! Below, you will find step-by-step instructions on how to create your Purgo AI account and log in to the platform.

How to Sign Up for Purgo AI

Signing up for Purgo AI is the first step to start building, testing, and deploying your data applications. Follow these simple steps to create your account.

Step 1: Navigate to the Purgo AI Sign-Up Page

- Open your web browser and type in the following URL: https://app.purgo.ai/
- This will take you to the main Purgo AI page, where you can sign up for a new account.



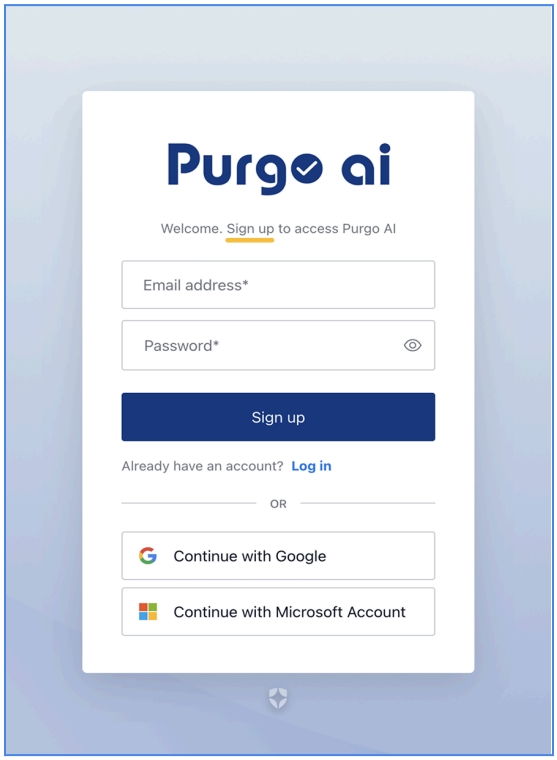
Step 2: Click on the "Sign Up" Button

- On the main page, look for the **Sign Up** button. This is usually located in the right below the login button.

- Click the **Sign Up** button to begin the registration process.

Step 3: Enter Your Email Address and Password

- You will now be prompted to provide some basic information:
    - **Email Address**: Enter a valid email address that you have access to. This email will be used for future communications from Purgo AI, and you will need it for logging in.
    - **Password**: Choose a strong password that you can remember. It's important to create a password that is secure, so be sure to include a combination of letters, numbers, and special characters for extra security.

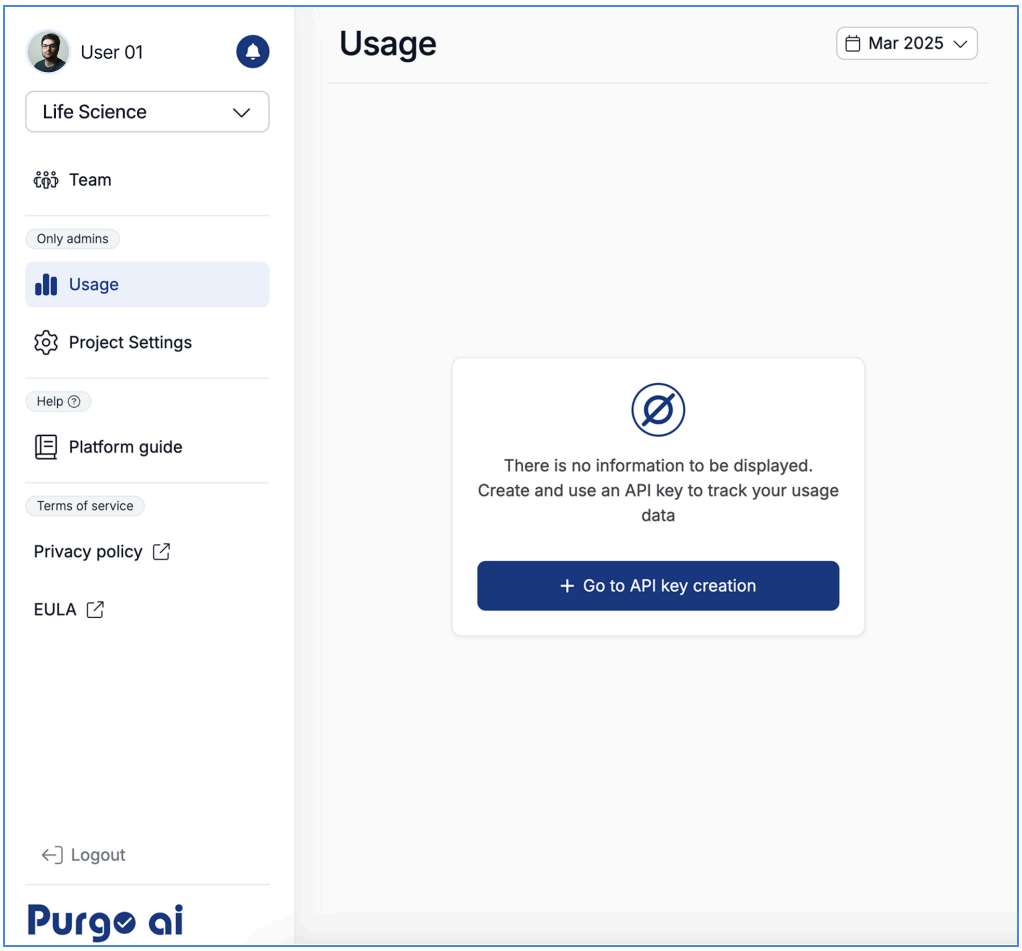


Step 4: Click on "Sign Up"

- After entering your email address and password, click the **Sign Up** button to submit your information.
- If everything is entered correctly, you will be successfully registered.

Step 5: Access the Usage Page

- Once your registration is complete, you will automatically be taken to the **Usage** page. This is the dashboard where you will track your activity and use Purgo AI to manage your data applications.
- Congratulations! You now have an account and can begin exploring the platform.



ETL Pipelines

Purgo AI simplifies the process of building and managing data applications and pipelines by integrating with popular platforms such as Jira, Confluence, GitHub, and Databricks. It uses pre-trained domain knowledge of enterprise and cloud data warehouses to automatically generate optimized code (e.g., Python, Spark, SQL) for data applications. In this guide, we'll walk through how to build and manage data applications within Jira using Purgo AI.

Pre-requisites

Before getting started, make sure the following are in place:

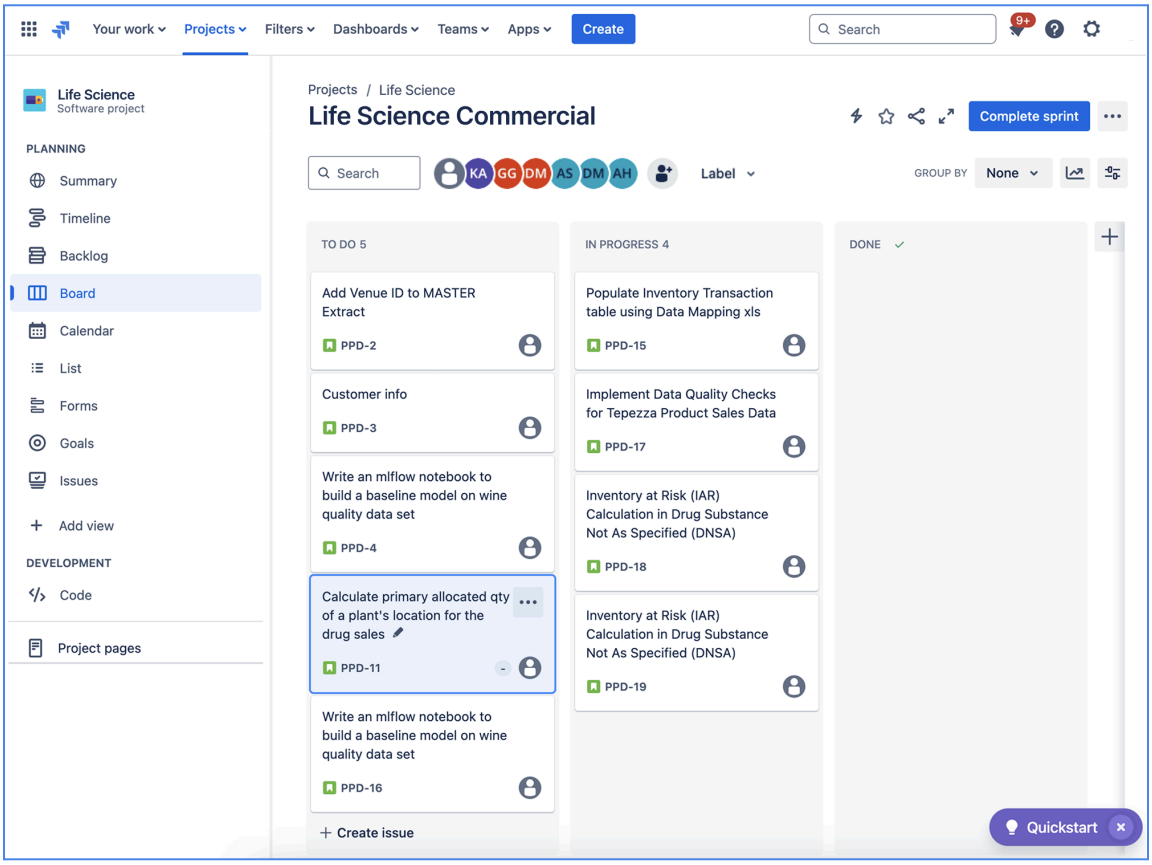
1. **Jira Integration**: Ensure the **Purgo AI app** is installed and configured in your Jira project by an admin.
2. **Linked Platforms**: Purgo AI should be connected to any relevant platforms, such as Confluence for documentation, GitHub for version control, and Databricks for executing the generated code.

Once the setup is complete, you can begin developing data apps and pipelines.

Step 1: Pick a Jira User Story

In your Jira project, Purgo AI allows you to develop data applications by starting with user stories. User stories define the business and technical requirements needed to build the app.

1. **Access the Project**:
   The project within Jira contains **sample user stories** that will guide you through developing a data app.
2. **Select a User Story**:
   From the Jira board, pick a user story that outlines a particular requirement for the data app. This user story will serve as the foundation for your project.



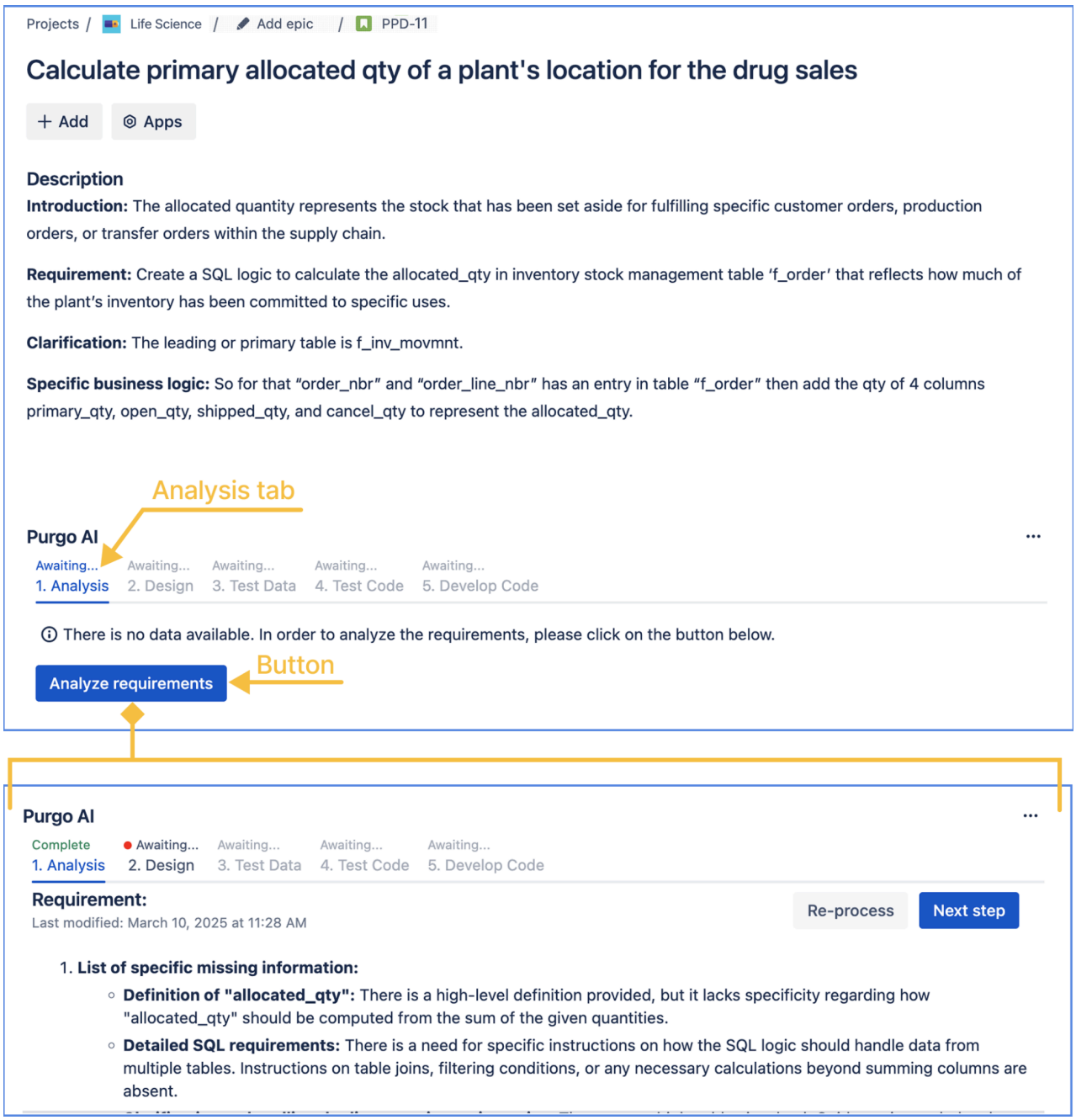
Step 2: Requirements Analysis

Once a user story is selected, the next step is analyzing the requirements. Purgo AI assists in this phase by identifying potential gaps, inconsistencies, and conflicting requirements.

1. **Open the User Story**:
   Click on the selected Jira user story ticket.

2. **Initiate Requirements Analysis**:
    - In the first and only active tab “1. Analysis” click on the button **“Analyze requirements”,** This will initiate the analysis process, Purgo AI will now analyze the requirements of the user story. Once finished processing, it will provide feedback, and retrieve relevant context that is crucial for the development phase.

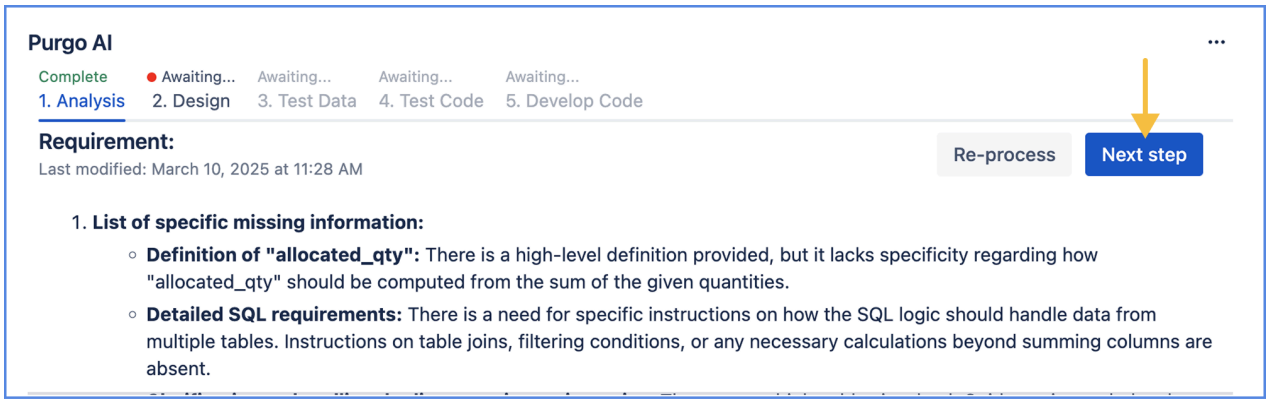
Note: *If there is not enough detail in the user story, Purgo will leave comments on the Jira content on the additional details needed.*



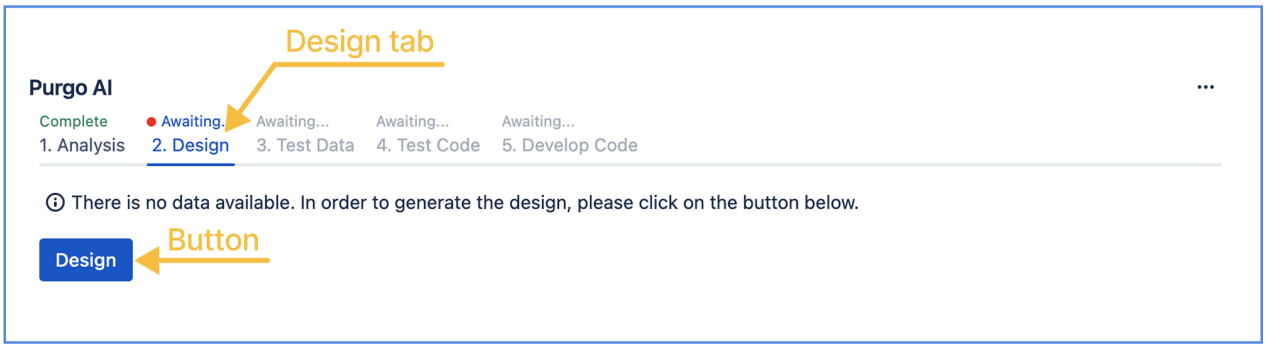
Step 3: Design

After analyzing the requirements, Purgo AI moves on to the design phase, and the design tab will become available. In this phase, a **Gherkins document** is automatically generated, which describes the app's features, scenarios, processing logic, and actions.

1. **Initiate Design**:
    - After reviewing the Requirement Analysis, you can either click on the **"Next step"** button or move into the "2. Design" tab and click on the **"Design"** button. Either way, this will trigger the processing of the design. Once the design is processed the next tab, "3. Test Data" will become available.
2. **Gherkins Document**:
   This document will be stored in the associated repository and outlines all necessary specifications in a **YAML format**. It serves as the foundation for test and development code generation.



OR

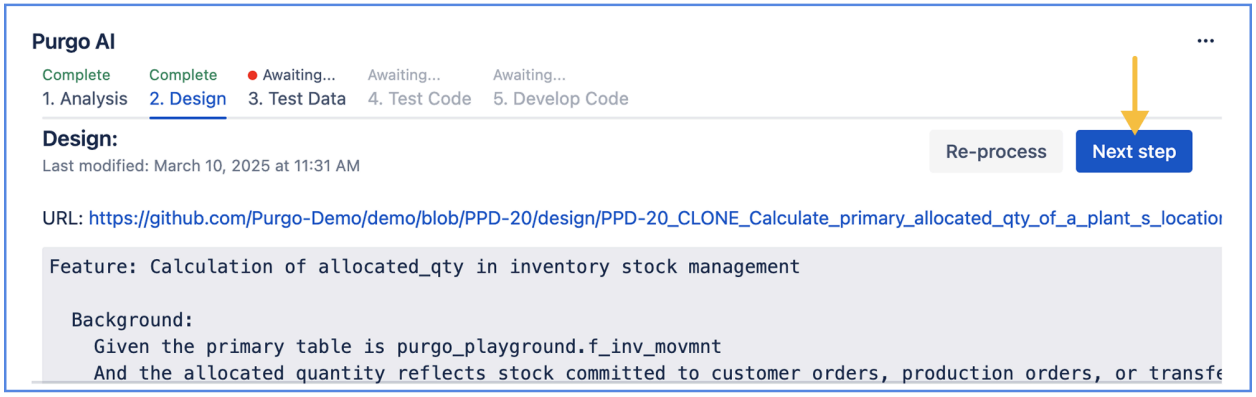


Step 4: Test Data and Code

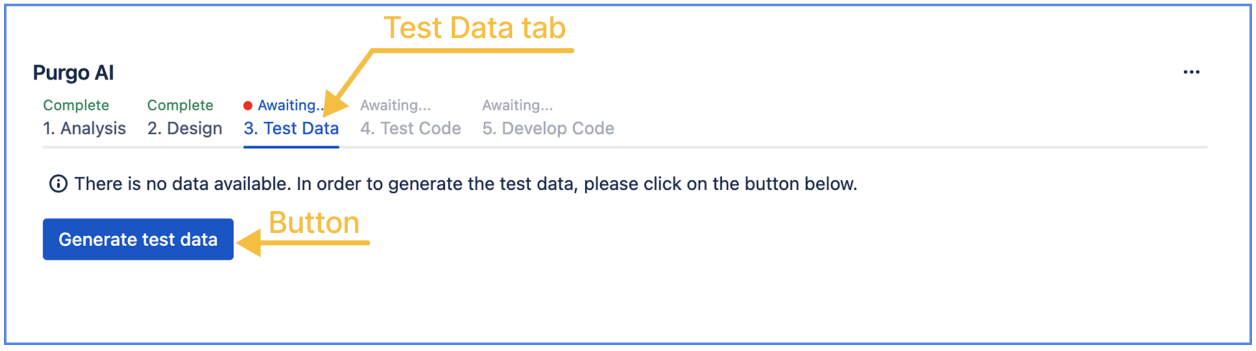
Purgo AI follows a **Test-Driven Development (TDD)** approach. This means that before the actual application code is generated, test harnesses are created based on the specifications from the design phase.

1. **Generate Test Data and Code**:
    - **3- Test Data:** After reviewing the Design output, click on the **“Next step”** button or move into the “3. Test Data” tab and click on the **“Generate test data”** button. Either one will trigger the generation of sample data for testing the code. After the sample data is generated, the next tab “Test Code” will become available.
    - **4- Test Code:** After reviewing the Test Data,click on the **“Next step”** button or move into the “4. Test Code” tab and click on the **"Generate test code”** button. Either one will trigger the generation of test scripts that will validate the application code.

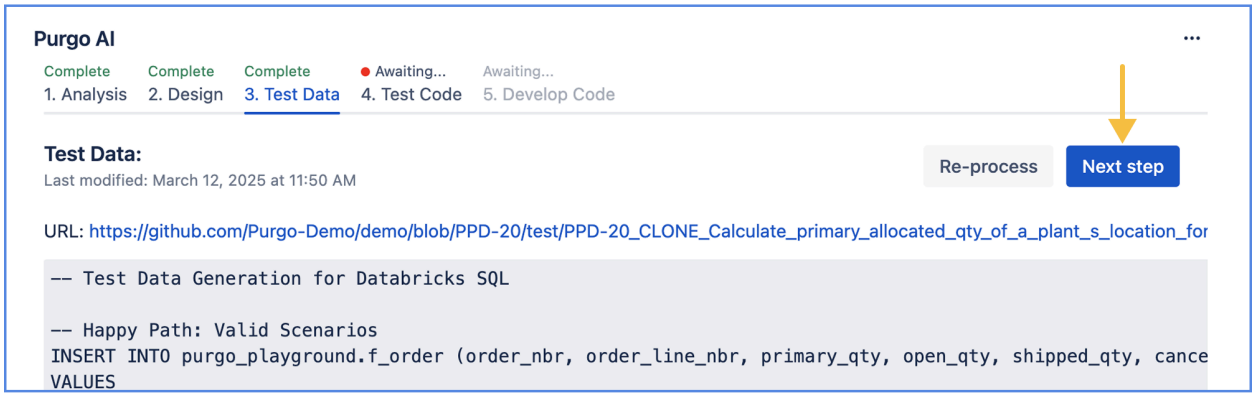
3- Test Data



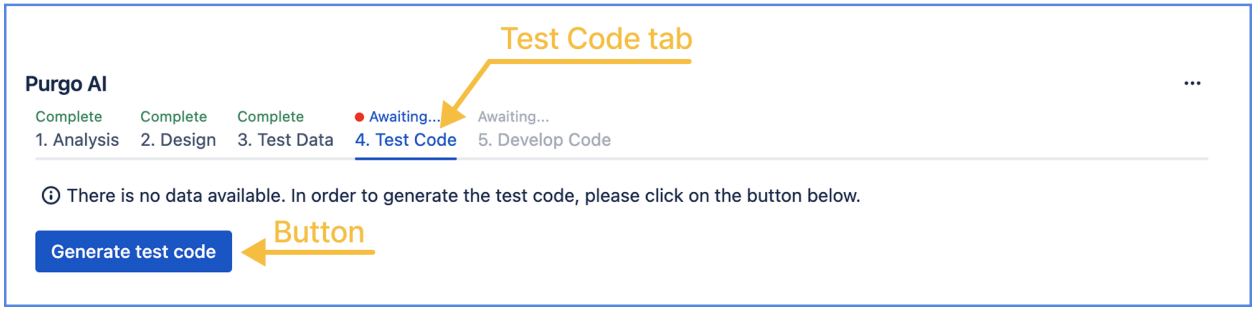
OR



4- Test Code



OR

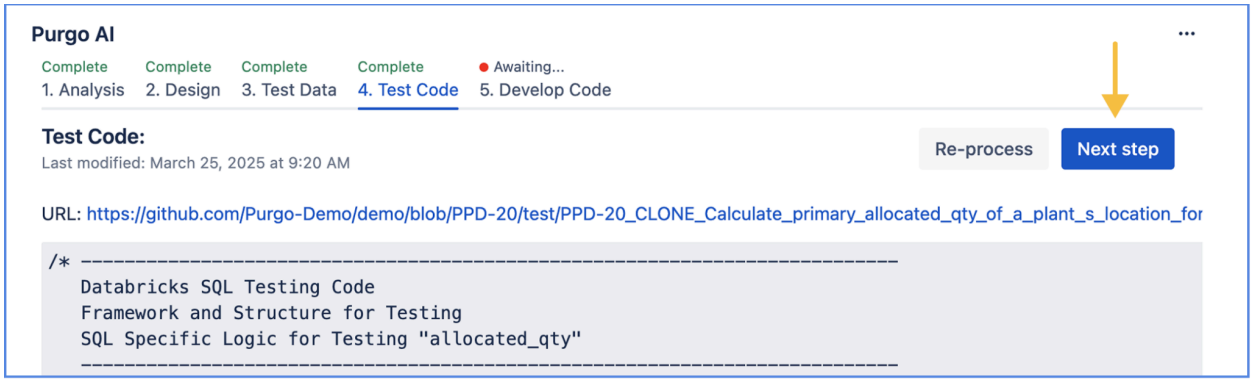


Step 5: Develop Code

Once the test data and scripts are in place, the next and last tab, “5. Develop Code” will become available and now Purgo AI can generate the final code for your data application.

1. **Initiate Code Generation**:
    - After reviewing the Test artifacts (test data and test code), click on the **“Next step”** button or move into the “5. Develop Code” tab and click on the **“Develop code”** button.

Purgo AI will then generate **execution-ready code** based on the earlier analysis, design, and test steps.



OR

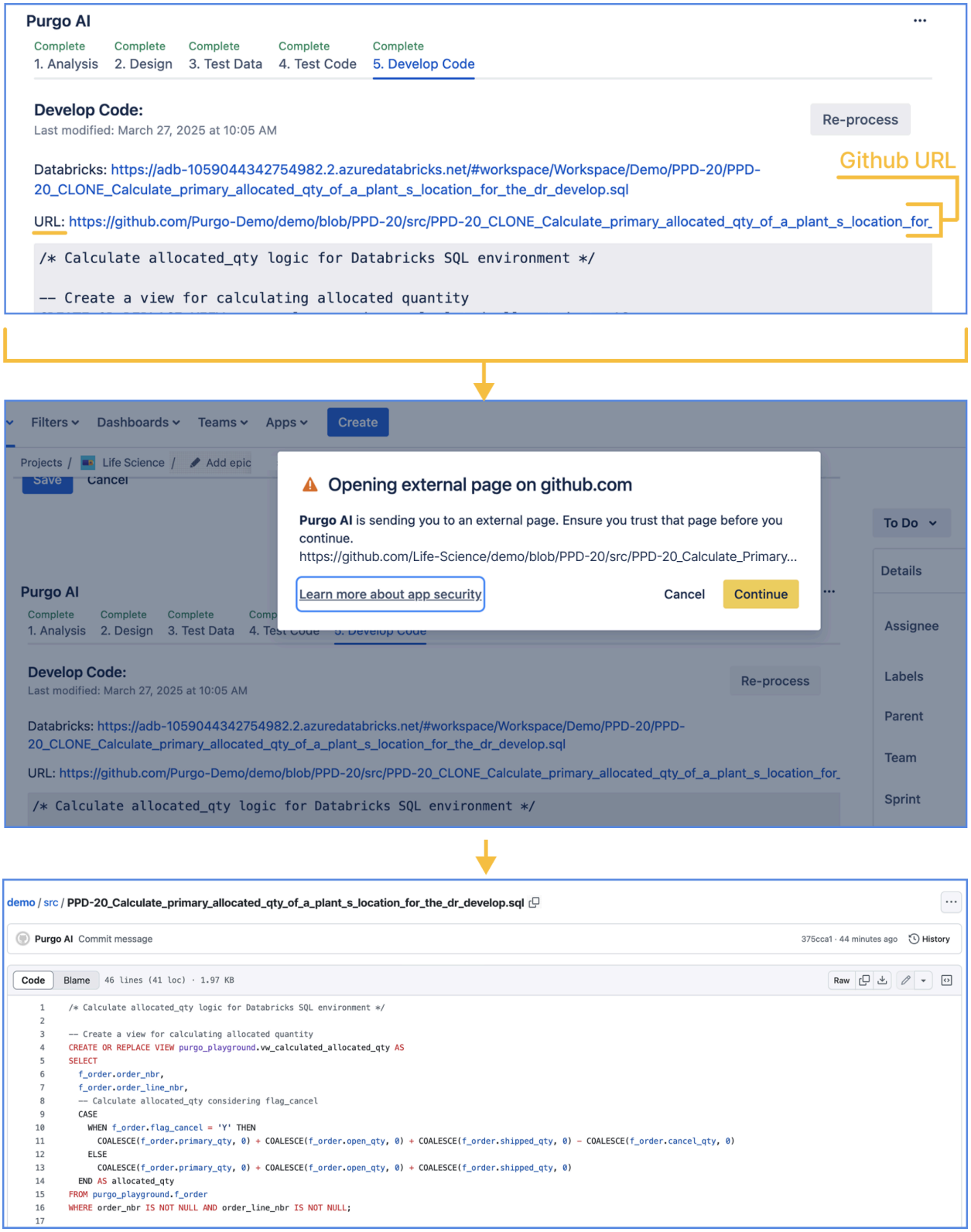
Note: *Any of the five steps (analysis, design, test data, test code, develop code) can be re-processed, but when one step is reprocessed, all subsequent steps will become outdated and will have to be re-processed as well.*

Note: *If the Jira ticket is updated all the steps will become outdated and will have to be re-processed.*

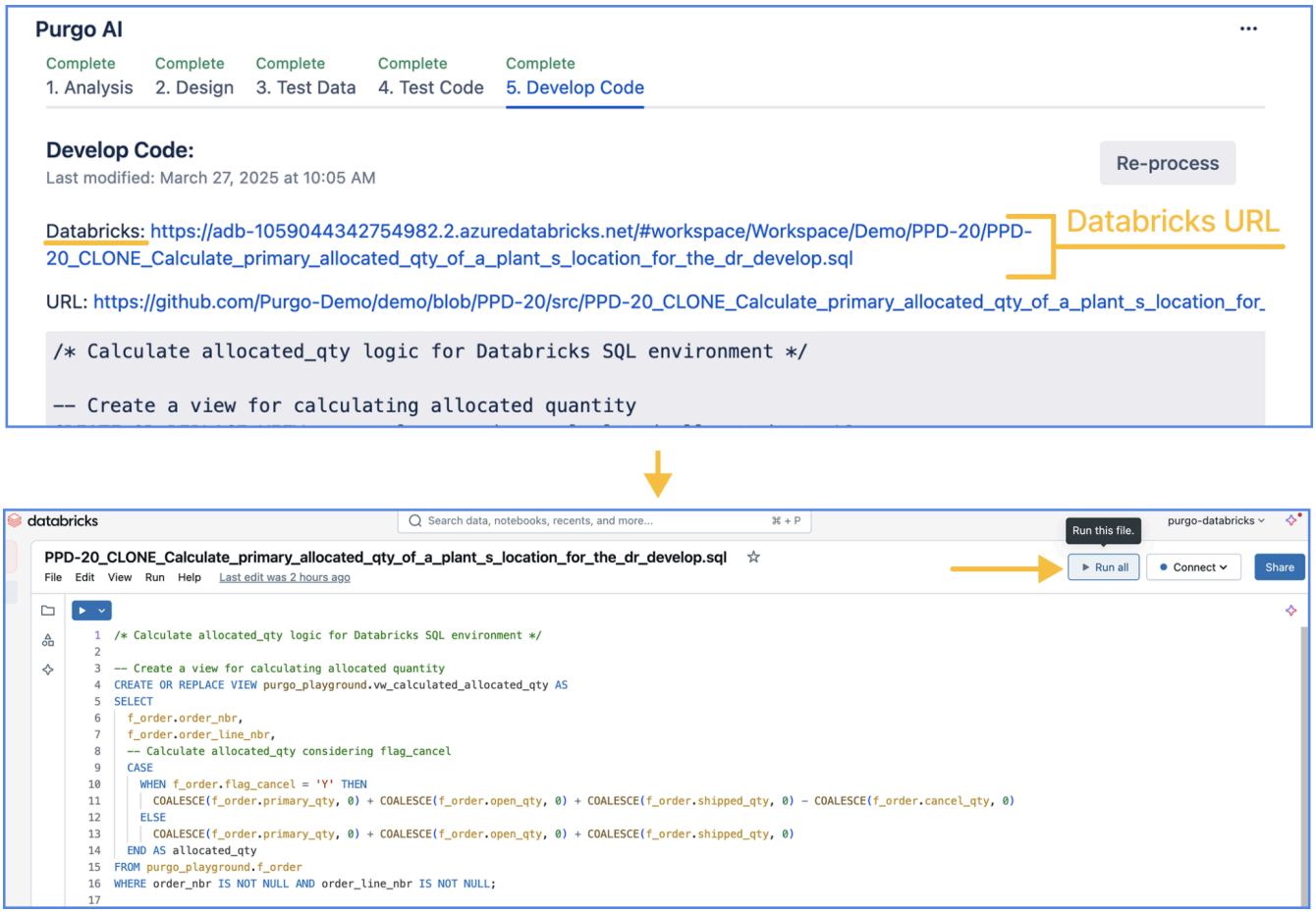
Step 6: Review and Execute Code

After Purgo AI generates the application code, it is placed in both your **GitHub repository** and your **Databricks Workspace** (if Databricks is configured for your project). The code can be reviewed and executed in these environments.

1. **Open the Code in GitHub**:
    - A URL link will be provided by Purgo AI under the “5.Develop Code” tab, which will direct you to the **GitHub repository** where the code is stored.
    - Click on the URL to review the code and any associated artifacts.


Purgo AI
Complete 1. Analysis
Complete 2. Design
Complete 3. Test Data
Complete 4. Test Code
Complete 5. Develop Code
Develop Code:
Last modified: March 27, 2025 at 10:05 AM
Re-process
Github URL
Databricks: https://adb-1059044342754982.2.azuredatabricks.net/#workspace/Workspace/Demo/PPD-20/PPD-20_CLONE_Calculate_primary_allocated_qty_of_a_plant_s_location_for_the_dr_develop.sql
URL: https://github.com/Purgo-Demo/demo/blob/PPD-20/src/PPD-20_CLONE_Calculate_primary_allocated_qty_of_a_plant_s_location_for_
/* Calculate allocated_qty logic for Databricks SQL environment */
-- Create a view for calculating allocated quantity
Filters
Dashboards
Teams
Apps
Create
Projects / Life Science / Add epic
Save
Cancel
Opening external page on github.com
Purgo AI is sending you to an external page. Ensure you trust that page before you continue.
https://github.com/Life-Science/demo/blob/PPD-20/src/PPD-20_Calculate_Primary...
Learn more about app security
Cancel
Continue
To Do
Details
Assignee
Labels
Parent
Team
Sprint
demo / src / PPD-20_Calculate_primary_allocated_qty_of_a_plant_s_location_for_the_dr_develop.sql
Purgo AI Commit message
375cca1 · 44 minutes ago
History
Code
Blame
46 lines (41 loc) · 1.97 KB
Raw
/* Calculate allocated_qty logic for Databricks SQL environment */
-- Create a view for calculating allocated quantity
CREATE OR REPLACE VIEW purgo_playground.vw_calculated_allocated_qty AS
SELECT
f_order.order_nbr,
f_order.order_line_nbr,
-- Calculate allocated_qty considering flag_cancel
CASE
WHEN f_order.flag_cancel = 'Y' THEN
COALESCE(f_order.primary_qty, 0) + COALESCE(f_order.open_qty, 0) + COALESCE(f_order.shipped_qty, 0) - COALESCE(f_order.cancel_qty, 0)
ELSE
COALESCE(f_order.primary_qty, 0) + COALESCE(f_order.open_qty, 0) + COALESCE(f_order.shipped_qty, 0)
END AS allocated_qty
FROM purgo_playground.f_order
WHERE order_nbr IS NOT NULL AND order_line_nbr IS NOT NULL;

2. **Execute the Code in Databricks**:
    - If Databricks is configured for your project, you can also open the corresponding code in **Databricks**.
    - In the Databricks Workspace, simply run the code to execute the data app and pipelines.



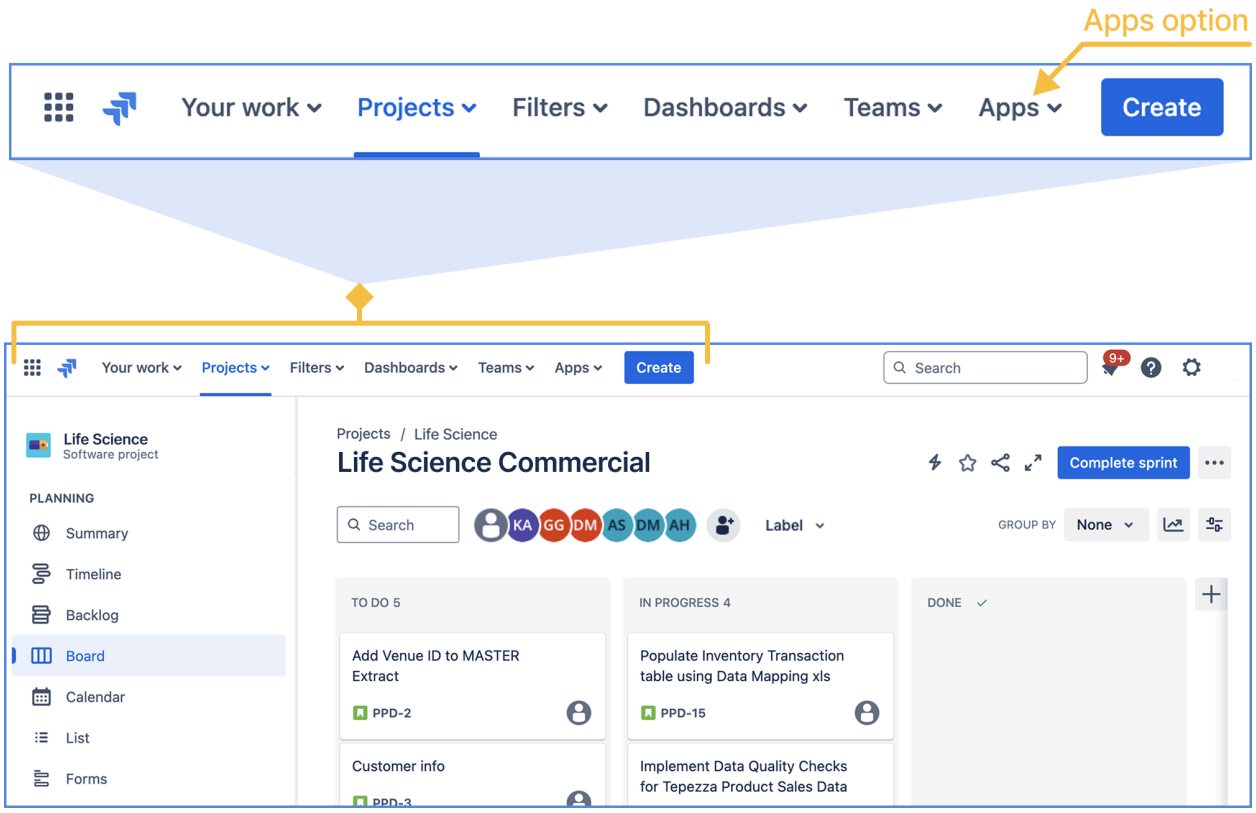
Connecting Purgo to Jira

To fully utilize **Purgo AI** within Jira, you need to install the Purgo AI plugin and configure it using an API key. This process allows Jira to integrate with Purgo AI for seamless project management and automation. In this guide, we will walk through the steps required to install the Purgo AI plugin in Jira and set up the necessary API keys and integrations.

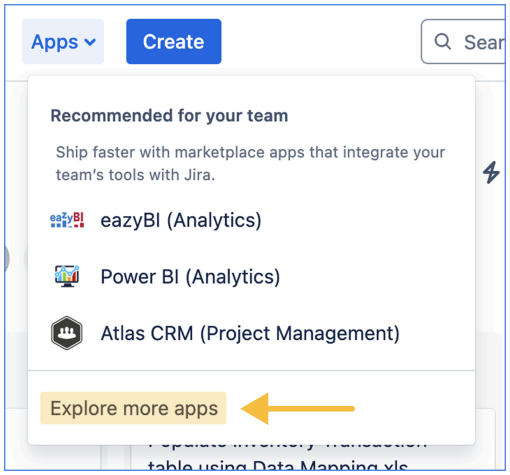
Step 1: Install the Purgo AI Plugin in Jira

Before you can configure the Purgo AI plugin, it must first be installed in your Jira instance.

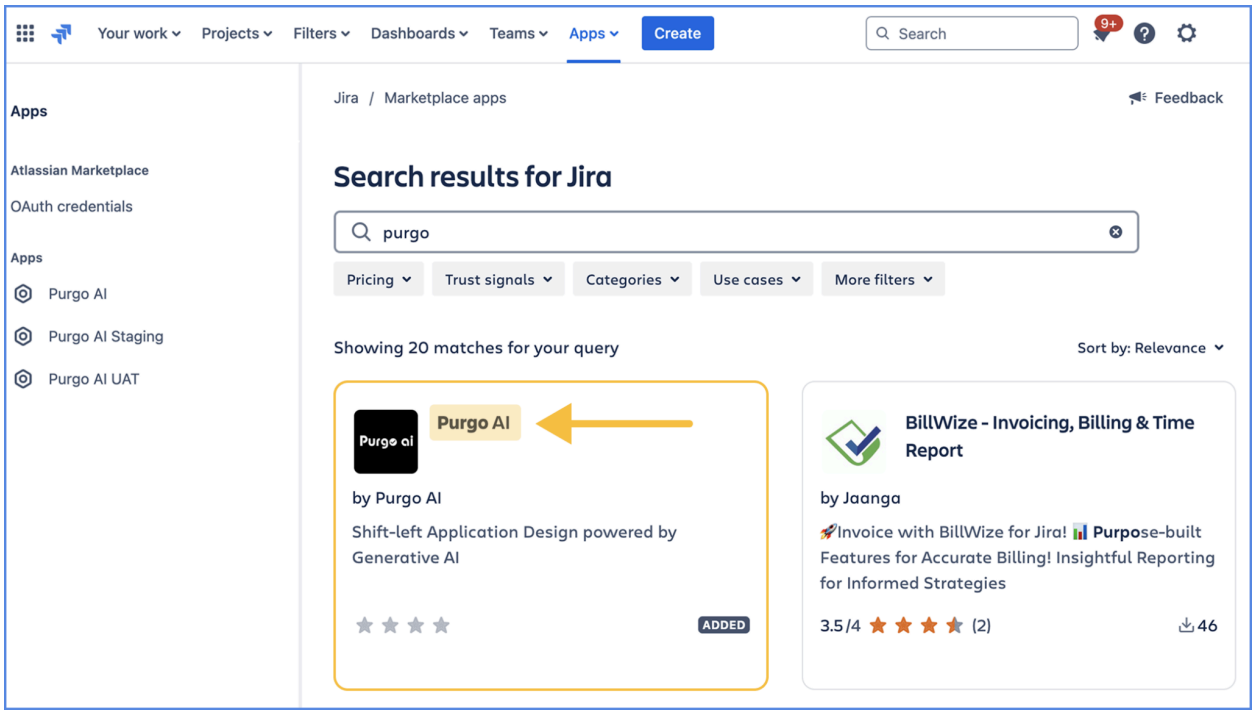
1. **Navigate to the Apps Section in Jira**:
    - Open Jira and log in with your account.
    - In the **navigation menu**, click on the **Apps** option. This is usually located at the top of the Jira interface.



2. **Explore More Apps**:
    - After clicking on **Apps**, a dropdown menu will appear. Select **Explore More Apps** to browse for additional Jira plugins.



3. **Search for Purgo AI**:
    - In the **search bar** at the top of the page, type the keyword **Purgo** and press enter.
    - Look for the **Purgo AI** app in the search results.

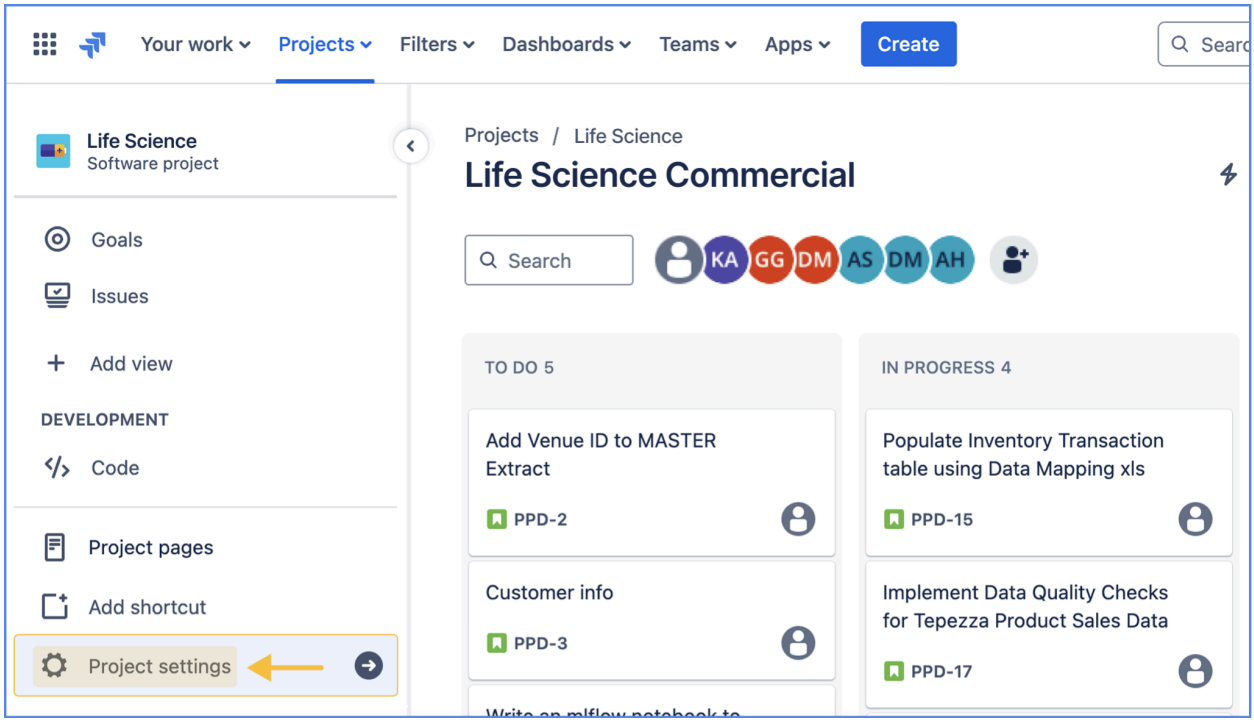


4. **Install the Purgo AI Plugin**:
    - Once you find the **Purgo AI** plugin, click on it to open the app details page.
    - Click the **Install** button to add Purgo AI to your Jira environment.
    - The installation process may take a few moments. Once complete, the Purgo AI app will be available in Jira.

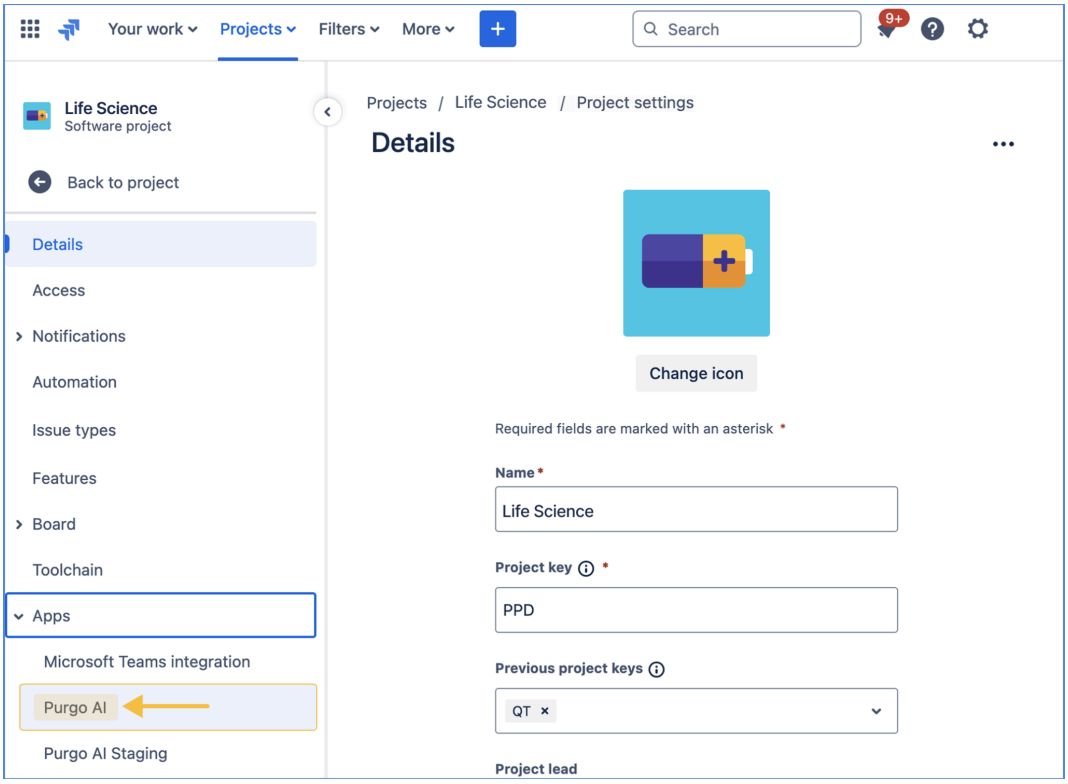
Step 2: Configure the Purgo AI Plugin in Jira

With the Purgo AI app installed, you now need to configure it using the API keys generated from Purgo AI and GitHub. These keys will authenticate Jira to use Purgo AI's features.

1. **Go to Project Settings**:
    - In Jira, open the **Project** where you want to use Purgo AI.
    - Click on the **Project Settings** option, typically found in the left-side navigation panel.



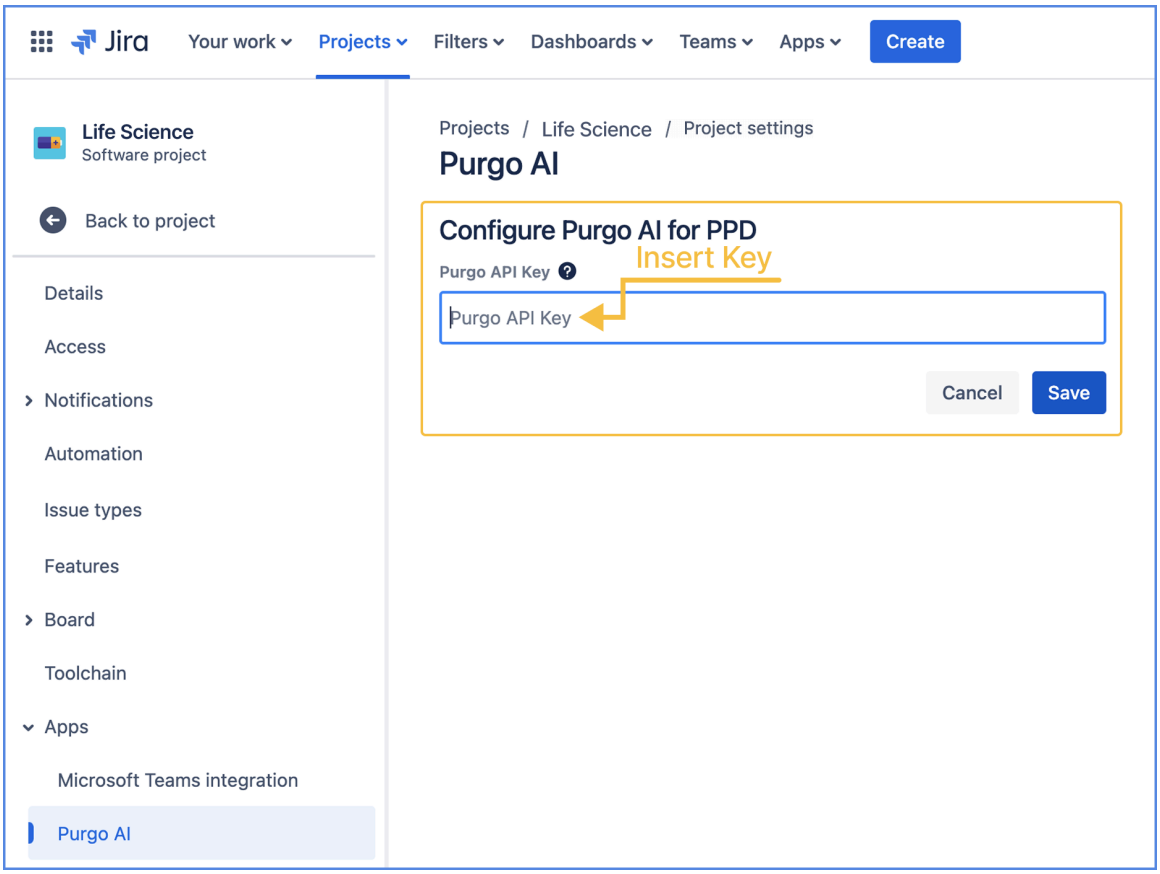
2. **Access the Purgo AI Configuration**:
   - In the **Apps** section of the Project Settings, click on **Purgo AI** to access the configuration page.
   - This is where you will set up the API integration.



Step 3: Configure GitHub and Purgo AI API Keys

Now that you're on the Purgo AI configuration page, follow these steps to set up the necessary authentication tokens.

1. **Enter Your Purgo API Key**:
    - You will need to enter the **API Key** generated from your Purgo AI account.
        - If you haven't generated an API key yet, follow the instructions in the Purgo AI Admin Panel to create one.
        - Once you have the key, paste it into the **Purgo API Key** field in Jira.



Step 4: Complete the Configuration

Once you have entered the required information (Purgo API Key):

1. **Save the Configuration**:
    - After inputting all the necessary key, click **Save** to finalize the setup.
2. **Verify the Integration**:
    - Once saved, Jira will now be connected to Purgo AI and GitHub. You can start using Purgo AI features within Jira to automate workflows, manage data apps, and more.

Summary

By following these steps, you've successfully installed and configured Purgo AI within Jira. This integration allows you to streamline data application development, automate processes, and connect Jira with GitHub for efficient project management. Be sure to safeguard your API keys and tokens as they provide access to your systems.

Usage Panel

Once Purgo AI is configured in Jira and actively used within your projects, you can track and monitor valuable insights using the built-in visual data tools provided by Purgo AI. These graphs allow you to monitor API usage, track project activity, and gain a better understanding of how Purgo AI is being utilized across your organization. This guide will walk you through the two key visual tools—**API Calls Graph** and **Stories Touched Graph**—and explain how to use them.

Prerequisites

Before you can visualize data in the graphs, make sure:

- **Purgo AI is fully configured in Jira** with an active API key.
- Your team is actively using Purgo AI within Jira for managing tasks, automating workflows, and interacting with user stories.

API Calls Graph

The **API Calls Graph** provides insights into how often the Purgo AI API is being used within your Jira projects. This graph allows you to monitor API call frequency and identify trends in usage over time.

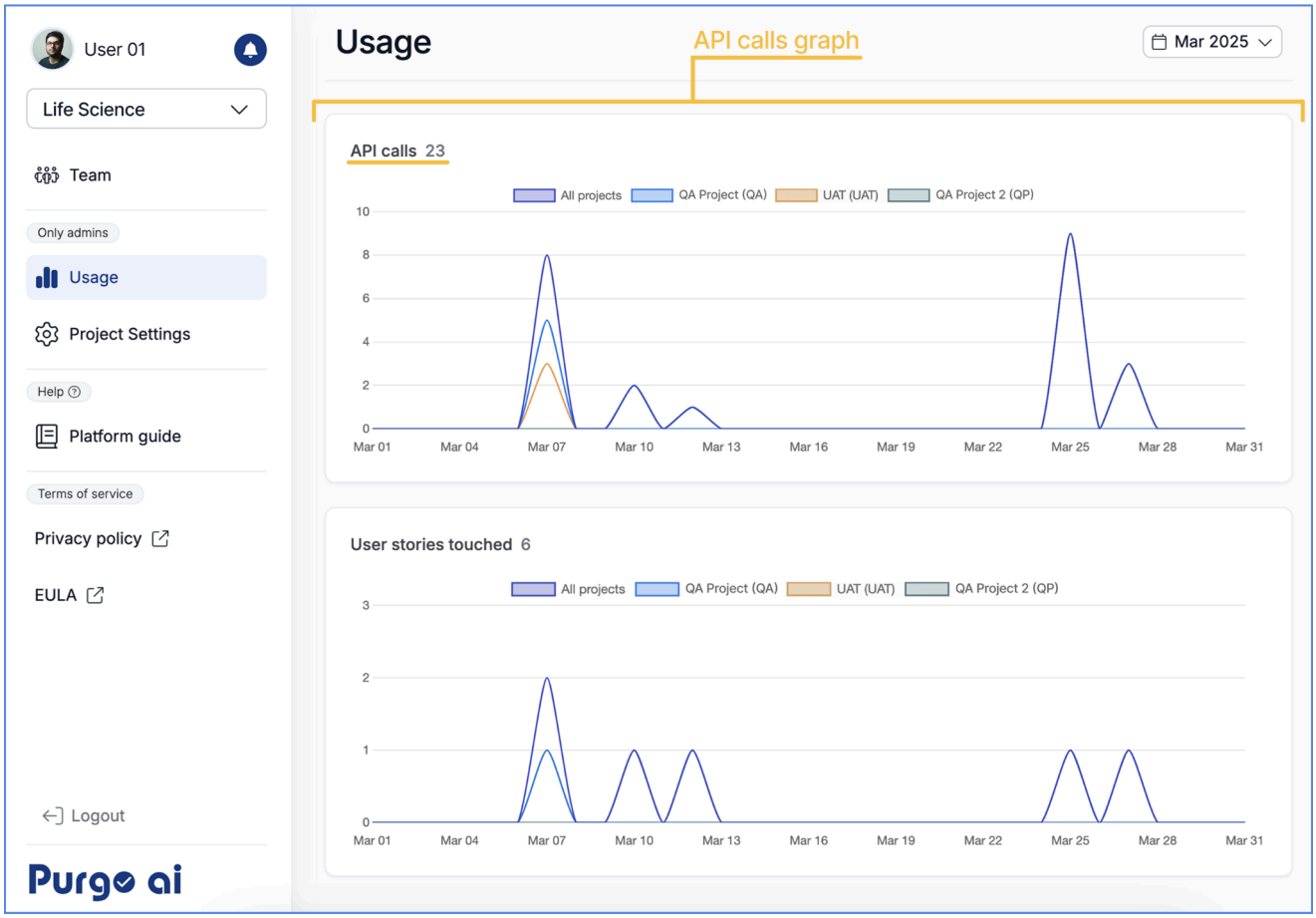
Key Features:

1. **Track API Usage**:
    - The graph displays the total number of times the Purgo AI API has been called within a specific project or across multiple projects.
    - It helps you measure how frequently Purgo AI is being used to automate workflows and manage data applications.
2. **Click to View Total API Calls**:
    - You can interact with the timeline on the graph by clicking on specific points.
    - This allows you to drill down into the data and view the total API calls made on specific dates for each project, as well as the overall total across all projects.
3. **Supports Multiple Projects**:

- The API Calls Graph can track activity for multiple projects using the same API key.
- This is useful for teams working on different projects, as it provides a centralized view of Purgo AI usage across the entire organization.

How to Use:

- Monitor this graph regularly to track how often Purgo AI is being used within your projects.
- Use the timeline to investigate spikes in API usage, which may indicate high activity periods or increased automation needs.



Stories Touched Graph

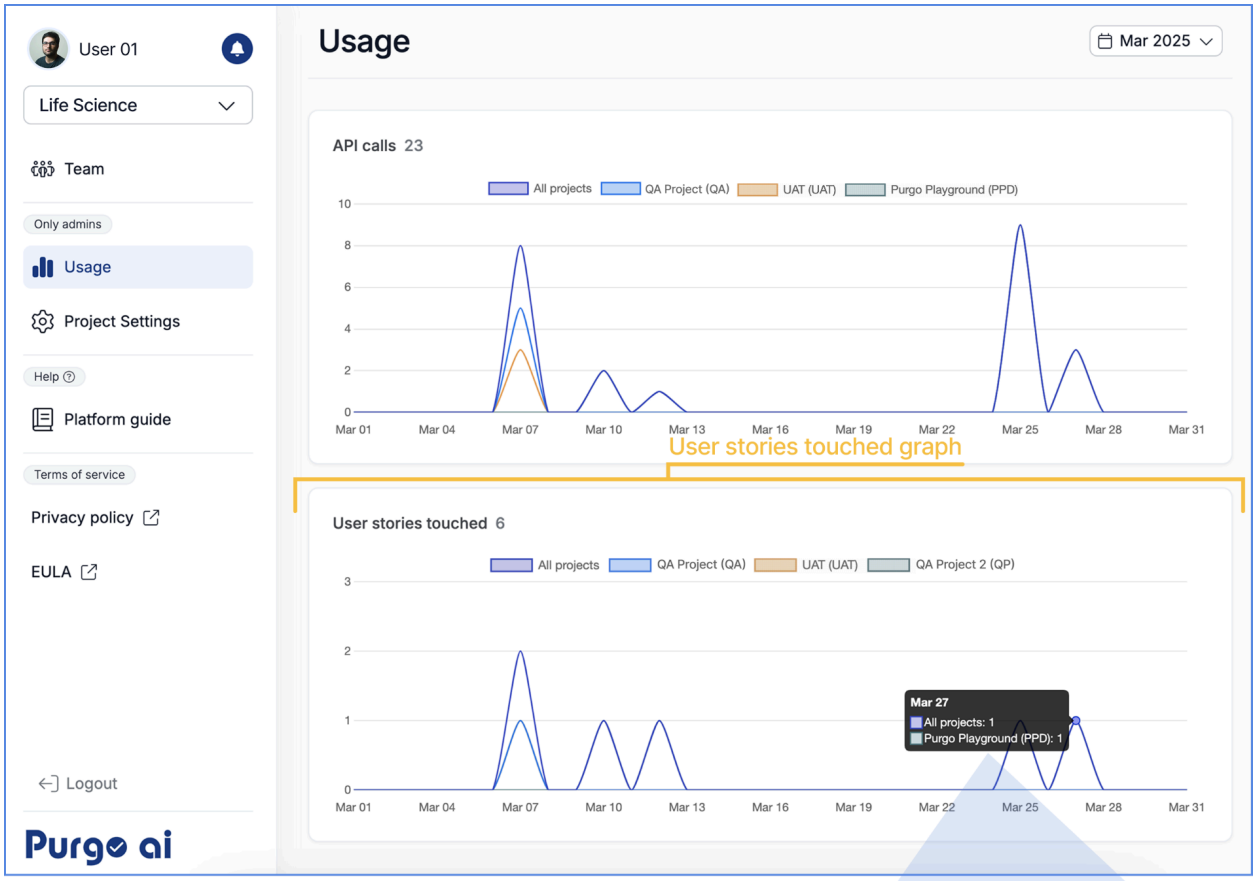
The **Stories Touched Graph** is designed to show how many user stories in Jira have been worked on using Purgo AI. This graph helps project managers and team members understand the progress being made on various tasks and how actively Purgo AI is contributing to the project workflows.

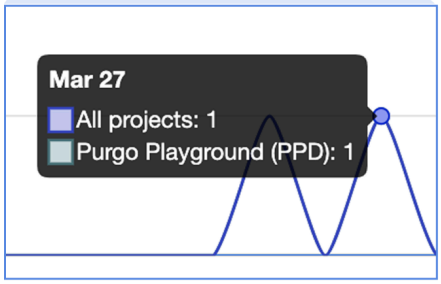
Key Features:

1. **Track User Stories**:
    - This graph displays the number of Jira stories that have been worked on within a specific project or across multiple projects.
    - It shows the impact of Purgo AI in managing tasks and automating work across different story tickets.
2. **Filter by Month**:
    - You can apply a filter to group data by month, allowing you to track how much work has been done over time.
    - This feature helps you see trends in activity, such as whether your team is working more or less frequently on user stories with the assistance of Purgo AI.
3. **Multiple Project Support**:
    - Like the API Calls Graph, the Stories Touched Graph supports multiple projects, giving you the ability to see the total number of stories worked on across all Jira projects configured with the Purgo AI API.

How to Use:

- Use the filter option to group data by month and assess the number of stories worked on over time.
- This is especially useful for tracking project progress and ensuring that Purgo AI is effectively helping your team manage tasks.
- Check this graph frequently to identify patterns in story activity and to adjust workflows as necessary.





You can interact with the elements of any of the two graphs

Insights and Monitoring Progress

Both the **API Calls Graph** and **Stories Touched Graph** offer valuable insights into how Purgo AI is being used in your Jira projects. Monitoring these graphs allows you to:

- Track overall **usage trends** of Purgo AI across your team or organization.
- Ensure that **API calls** and automated workflows are being effectively leveraged.
- Gain insights into the **volume of work** Purgo AI is handling through Jira story management.

- Use the data to **optimize team performance** by understanding when and where Purgo AI adds the most value to your projects.

Conclusion

By visualizing data through the **API Calls Graph** and **Stories Touched Graph**, Purgo AI provides an intuitive way to monitor the effectiveness of its integration with Jira. Regularly reviewing these graphs will help you identify trends, improve team workflows, and ensure you're getting the most out of Purgo AI's automation and project management features.

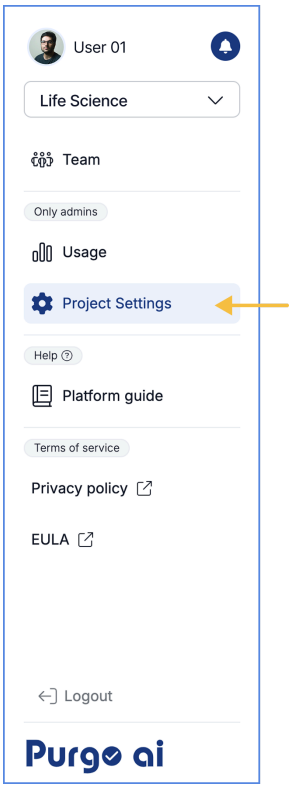
Connecting Purgo AI to Databricks

With the integration of **Purgo AI** and **Databricks**, you can seamlessly transfer generated information from Jira to Databricks using the **Develop Code** option in the Purgo AI plugin. This configuration ensures that all code generated in Purgo AI for Jira issues is automatically stored and available within your Databricks workspace.

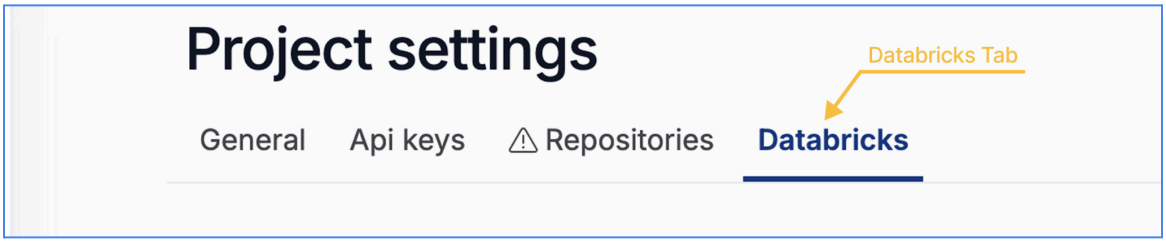
Below is a step-by-step guide to configuring this integration:

Step 1: Access Project Settings in Purgo AI

1. **Navigate to the "Project settings" screen**:
    - In Purgo AI, start by navigating to the **"Project settings"** option in the "only admins" section of the navigation menu (nav bar).



2. **Click on Databricks**:
    - Once in the Project settings section, click on the **“Databricks”** tab. This is where you’ll configure your Databricks account to work with Purgo AI.

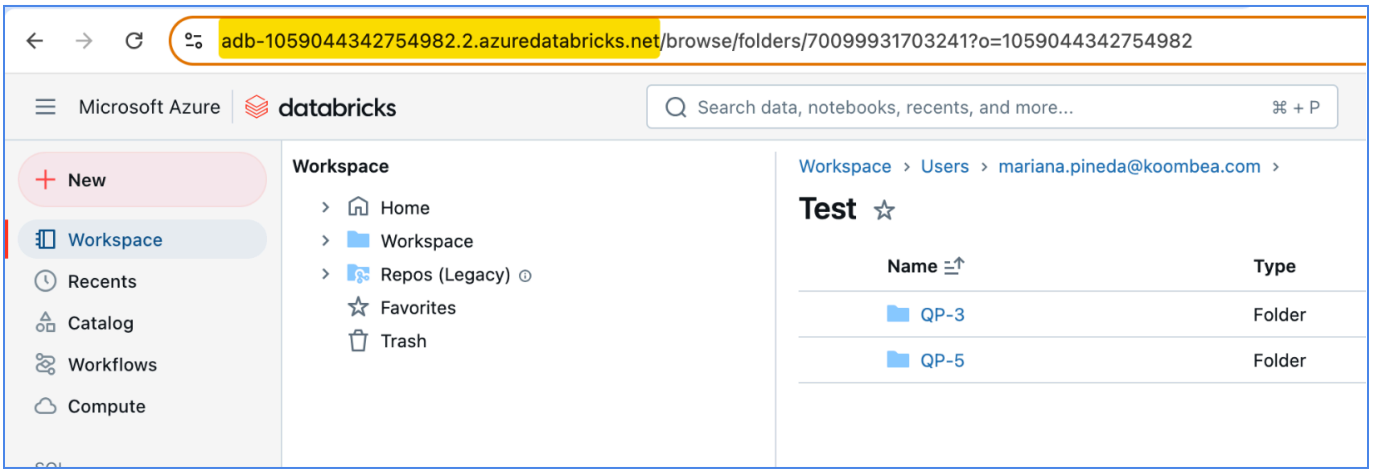


Step 2: Fill in Databricks Configuration Details

In the **Databricks** section of the Project settings, you’ll need to enter several key details from your Databricks account, such as the Databricks host configuration, the authentication method selection and the data configuration. These details allow Purgo AI to securely send generated data to your Databricks environment.

Here are the fields you need to complete for the **Databricks host configuration:**

1. **Databricks Host:**
    - This field requires the **URL of your Databricks workspace**.
    - To find it:
        - Log in to your Databricks account.
        - Navigate to your Databricks workspace and copy the URL from your browser’s address bar.
        - Paste this URL into the **Databricks Host** field in Purgo AI.



Here are the fields you need to complete for the **authentication method selection:**

In this section you will first have to pick your authentication method between “Databricks OAuth” and “Databricks token”. Depending on the method you select, the info you have to provide will vary.

If you select **Databricks OAuth** you must provide the following information:

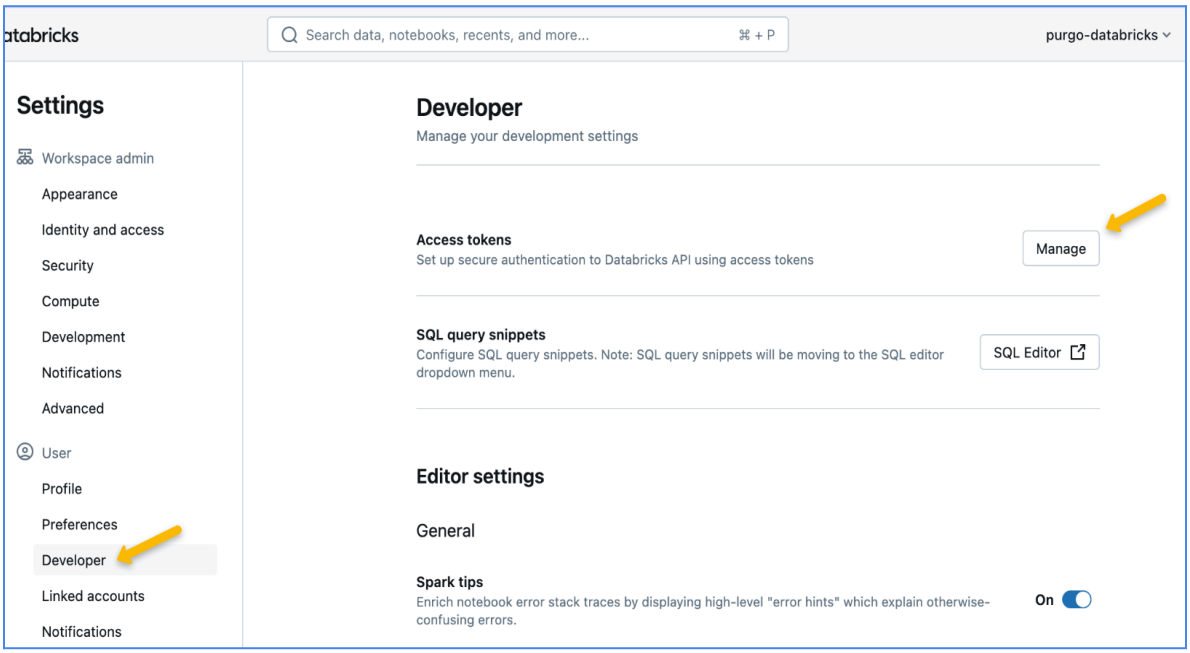
1. **Client ID**
2. **Client secret**

To find these information, both Client ID and Client secret you must:

- Log in to your Databricks account console as an account admin.
- In the left sidebar click on **“User management”**.
- Then Click on the **"Service principals"** tab.
- Under **"OAuth secrets"**, click **"Generate secret".**
- The generated secret and client ID will be displayed, copy them and paste them on Purgo AI under the respective inputs.

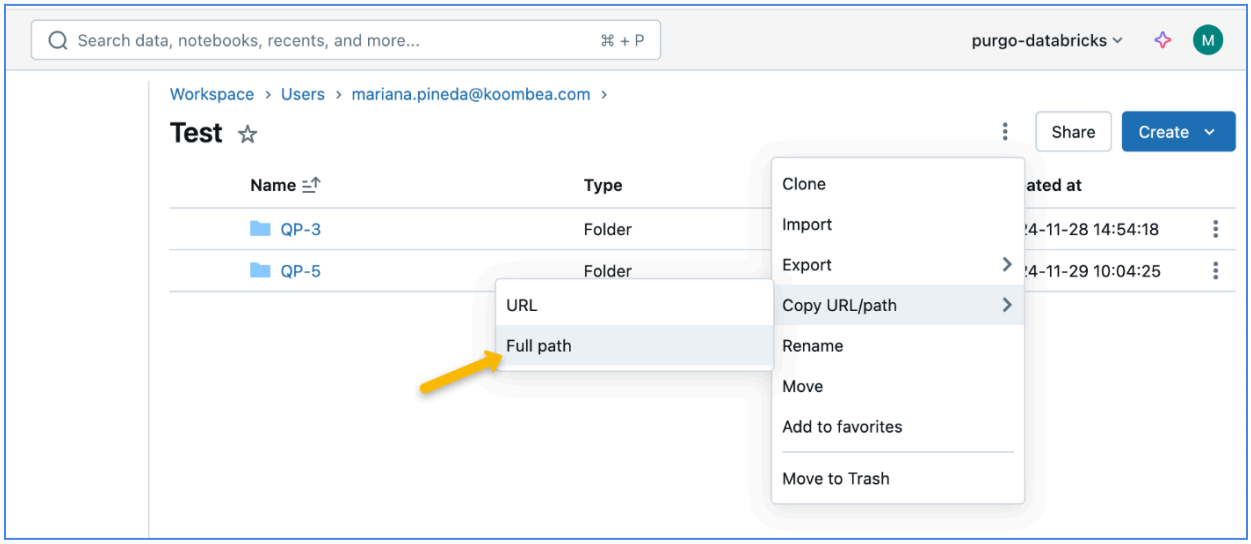
If you select **Databricks token** you must provide the following information:

1. **Databricks Token:**
    - The **Databricks Token** is an authentication token required to allow Purgo AI to communicate with your Databricks environment.
    - To generate this token:
        - Log in to your Databricks account.
        - Click on your profile icon and select **User Settings**.
        - Under the **Access Tokens** section, click on **Generate New Token**.
        - Copy the newly generated token and paste it into the **Databricks Token** field in Purgo AI.



Here are the fields you need to complete for the **Data configuration:**

1. **Shared Directory:**
    - The **Shared Directory** is the location within your Databricks workspace where Purgo AI will store generated data.
    - To configure this:
        - In Databricks, navigate through the workspace to the directory where you want the data to be stored.
        - You can either create a new directory or choose an existing one.
        - Copy the directory path and paste it into the **Shared Directory** field in Purgo AI.



2. **Additional Fields (Optional)**:
    - **Catalog Name**: If your Databricks environment uses Unity Catalog, enter the catalog name in the appropriate field.
    - **Schemas**: Define any schemas needed for organizing the data in your Databricks workspace.
    - **Cluster ID**: Include the cluster ID if necessary

Step 3: Finalize the Configuration

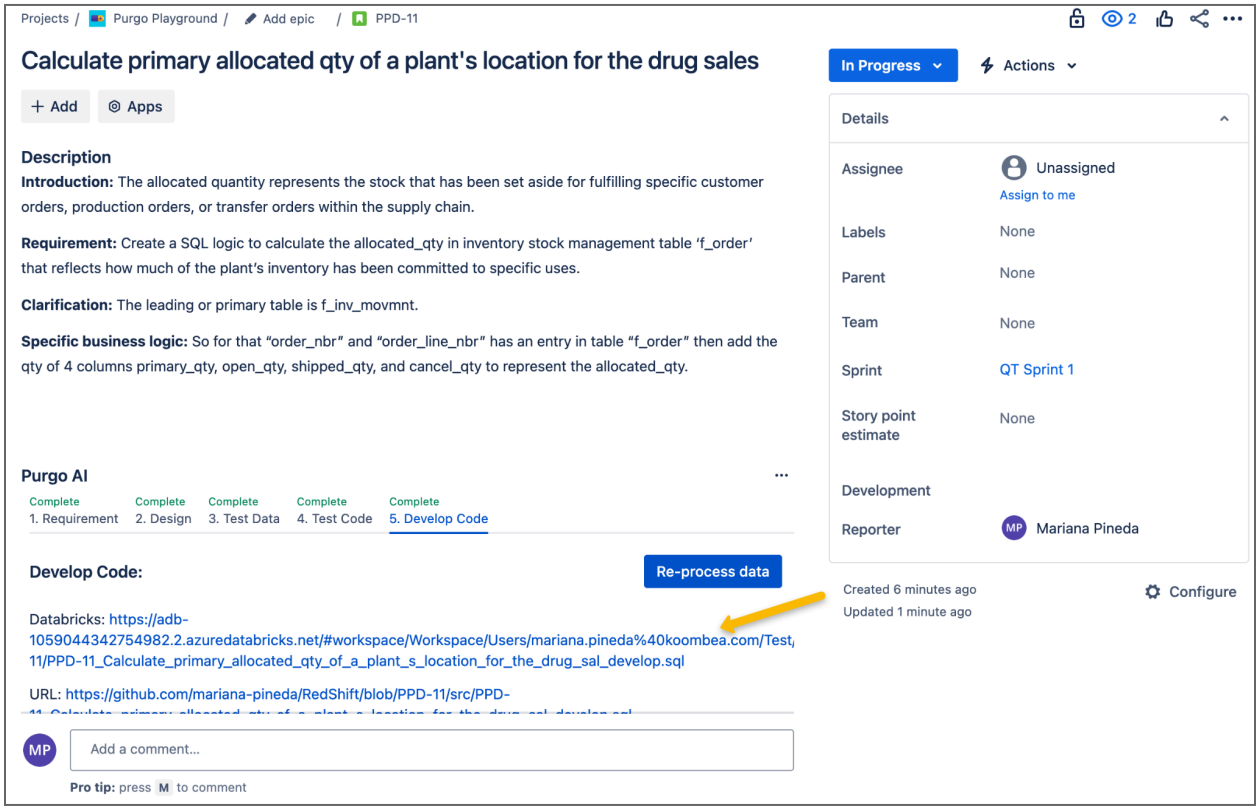
Once you've entered all the required information (Databricks host configuration, Authentication method selection, and Data configuration):

1. **Save Your Settings**:
    - Ensure all fields are completed correctly, and then click **Save** to apply your configuration.
2. **Verify the Integration**:
    - Your Jira instance is now connected to Databricks through Purgo AI. Any code or data generated via Jira will now be sent directly to Databricks.

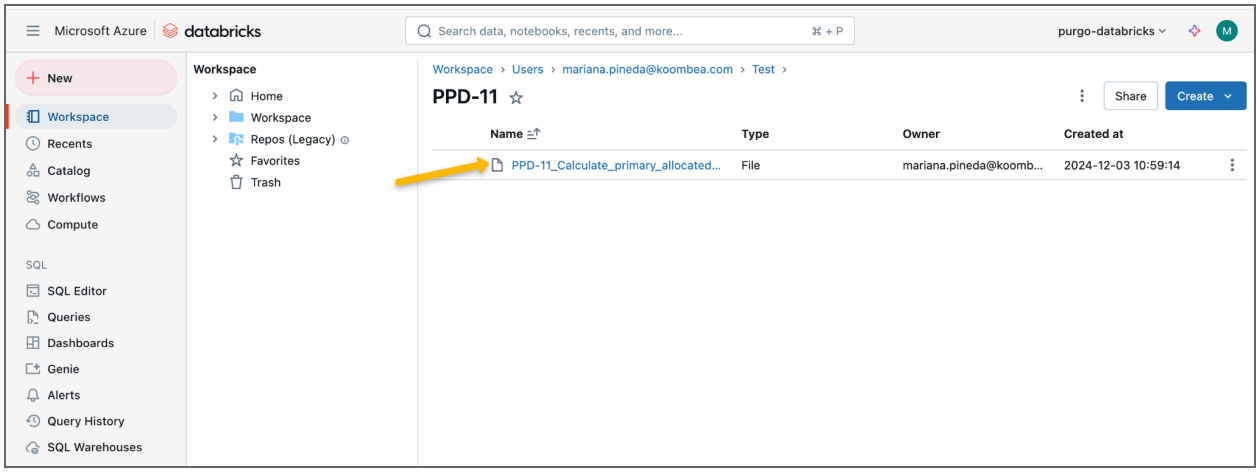
Step 4: Using the Develop Code Option in Jira

Once the configuration is complete, Purgo AI will automatically send information generated from Jira issues to Databricks. Here's how it works:

1. **Develop Code in Purgo AI**:
    - In Jira, when you use the **Develop Code** option in the Purgo AI plugin, the system will automatically generate code based on the requirements outlined in your Jira issue.
2. **Data Transfer to Databricks**:
    - The generated code and data will be automatically sent to the **Shared Directory** you specified during configuration.



3. **Access the Data in Databricks**:
    - In your Databricks workspace, navigate to the **Shared Directory** you specified in Step 2.
    - You will find the generated data and code stored there, ready for you to run, review, or modify directly in Databricks.

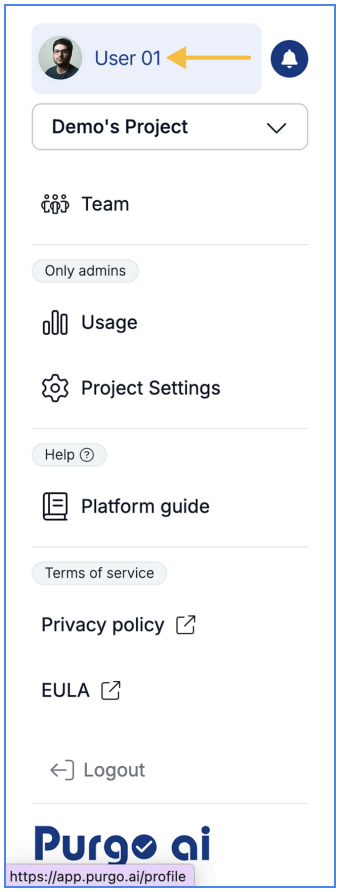


Summary

By following these steps, you have successfully integrated **Purgo AI** with **Databricks**. This integration allows for seamless transfer of data and code generated from Jira directly into your Databricks environment, enhancing your workflow and ensuring that all generated assets are available in your preferred cloud workspace.

Account Panel

The **Account Panel** in Purgo AI is where you can configure and manage settings related to the account itself. To access this panel you must click on your icon or name, located at the top of the navigation menu (nav bar). Below, we will guide you through the different sections of the Account Panel, Profile, Security and Notification settings.



Profile tab

The **Profile** section lets you update personal and company information to ensure your account reflects the correct details.

1. **Personal Information**

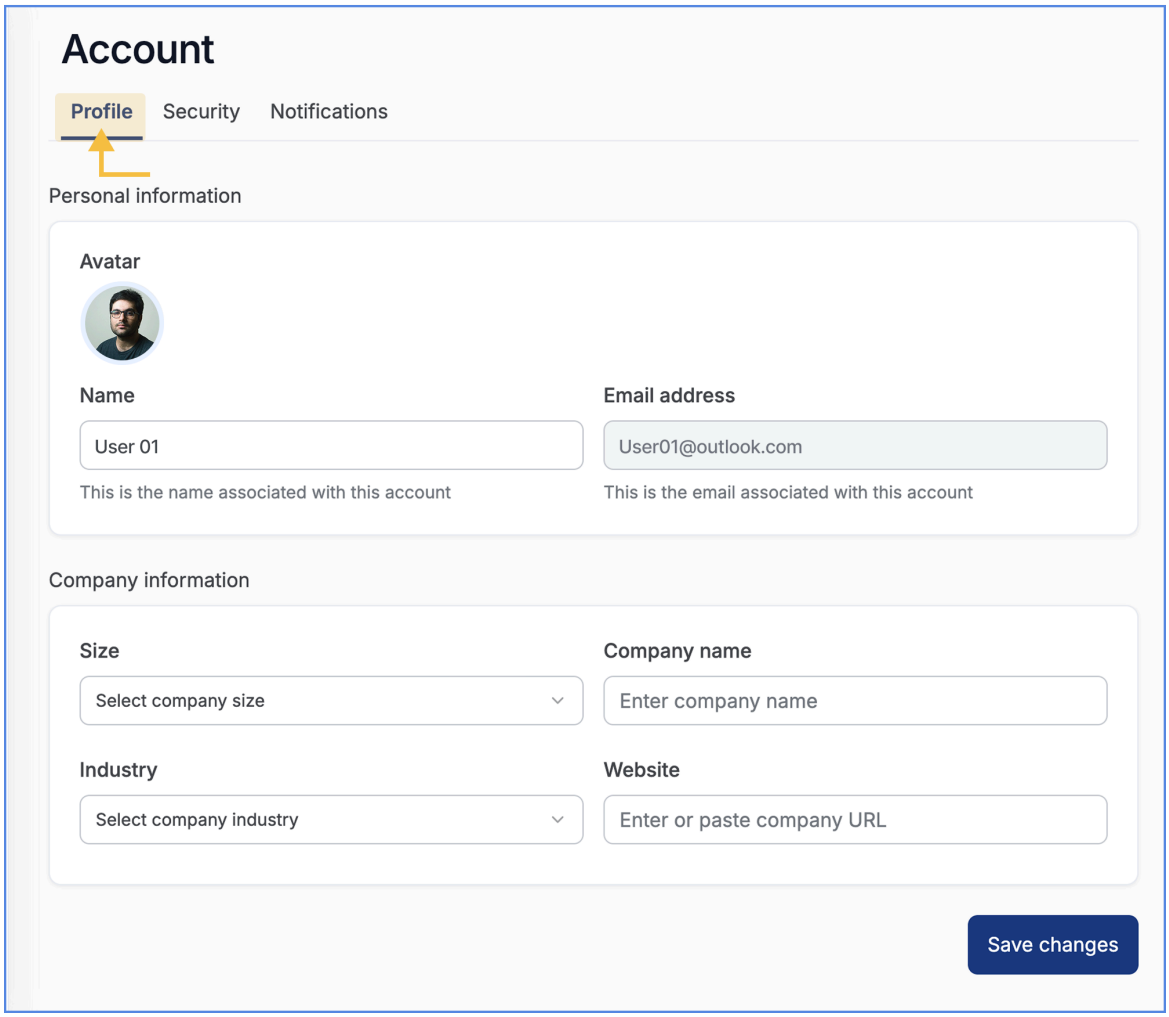
- **Name**:
  - You can update your name in the text box provided. Any changes will reflect in your account profile.
- **Email**:
  - This is a read-only field that shows the email associated with your Purgo AI account. If you need to change your email, you may need to contact support.

2. **Company Information**

- **Company Size**:
  - Use the dropdown menu to select the size of your company. This is useful for tailoring Purgo AI features and resources to better suit your organization's scale.
- **Company Name**:

    - Enter your company's name here for accurate representation in your profile.
- **Industry**:
    - Select your industry from the dropdown menu. This helps in aligning Purgo AI's features to your industry's specific needs.
- **Website**:
    - Provide the URL of your company's website in this field

Profile Tab



Security tab

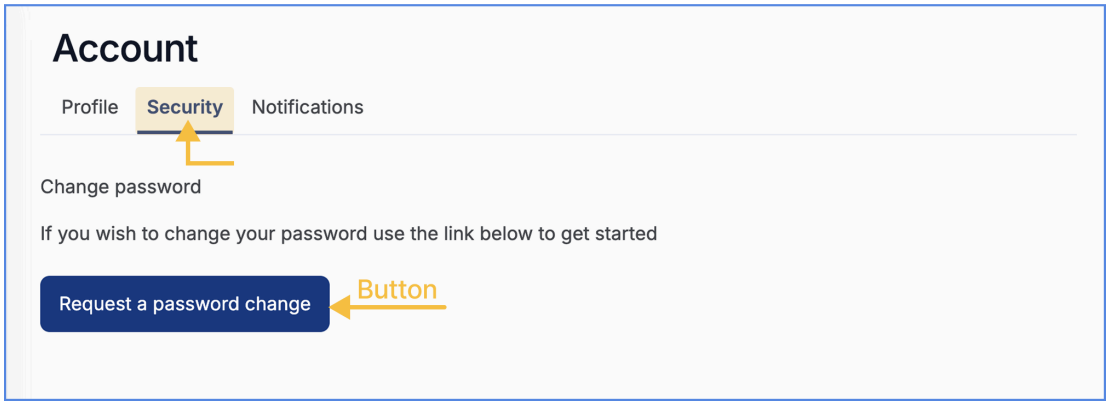
The **Security** section in the Account Panel allows you to manage security aspects of your Purgo AI account.

- **Change Password Button**:
    - To change your account password, simply click on the **Trigger Password Change** button. You will be prompted to follow the necessary steps to update

your password securely. It's recommended to periodically change your password to maintain account security.

***Note:** This tab (Security tab) will only be available if you didn't sign in directly with your google or microsoft account.*

Security Tab



Notification Tab

In the Notification section of the Account Panel, you can manage the notifications for projects, and teams. Here, you can decide whether you want to receive in app notifications, email notifications or both. It's as easy as just turning on or off the toggle buttons for the respective notifications.

Project settings

In the **Project settings** section you can manage various settings related to your projects. Within this section you can configure the general aspects of your project, as well as the API keys, Repositories, Databricks, Enterprise and LLMs settings. You can access the project settings from the navigation menu (nav bar), under the "Only admin" section clicking on **"Project settings".** Below, we will guide you through the different sections of the project settings

The **General Settings** section allows you to manage your Purgo AI project details, set defaults, and configure integrations with external platforms like Databricks.

1. **General settings**

- **Project Name**:
    - This is where you can set the name of your Purgo project. Simply enter your preferred name in the text box provided.
- **Project ID**:
    - You will see a read-only field that displays the unique Project ID assigned to your Purgo AI project. This ID is essential for internal tracking and integration with external systems.

2. **Repositories settings**

- **Enter Your GitHub Repository**:
    - Below the GitHub Auth Type field, you'll see a field to input your **GitHub Repository**.
    - Enter the URL or name of the GitHub repository where you want Purgo AI to store generated data and code.
- **Generate and Enter GitHub API Token**:
    - To connect Jira and Purgo AI to GitHub, you'll need to generate a **GitHub API Token**.
        - In GitHub, go to your **Profile Settings**.
        - Navigate to **Developer Settings** > **Personal Access Tokens** and click on **Generate New Token**.
        - Copy the token and return to Jira. Paste the token into the appropriate field under GitHub Authentication.

3. **Databricks Settings**

Purgo AI provides seamless integration with Databricks. In this section, you can configure your connection details to ensure smooth data migration and analytics.

- **Databricks Host**:
    - Enter the URL of your Databricks workspace here.
- **Databricks Token**:
    - This field is for your Databricks access token. You can generate the token from your Databricks account under the User Settings. Paste the token here to enable secure integration.
- **Shared Directory**:
    - Specify the directory within Databricks where your generated data and scripts will be stored.
- **Catalog Name**:
    - If you are using Databricks' Unity Catalog, enter the **Catalog Name** associated with your workspace.
- **Schemas**:

- Define the schemas for your data. You can manage different schema names that will help organize data within your Databricks environment.

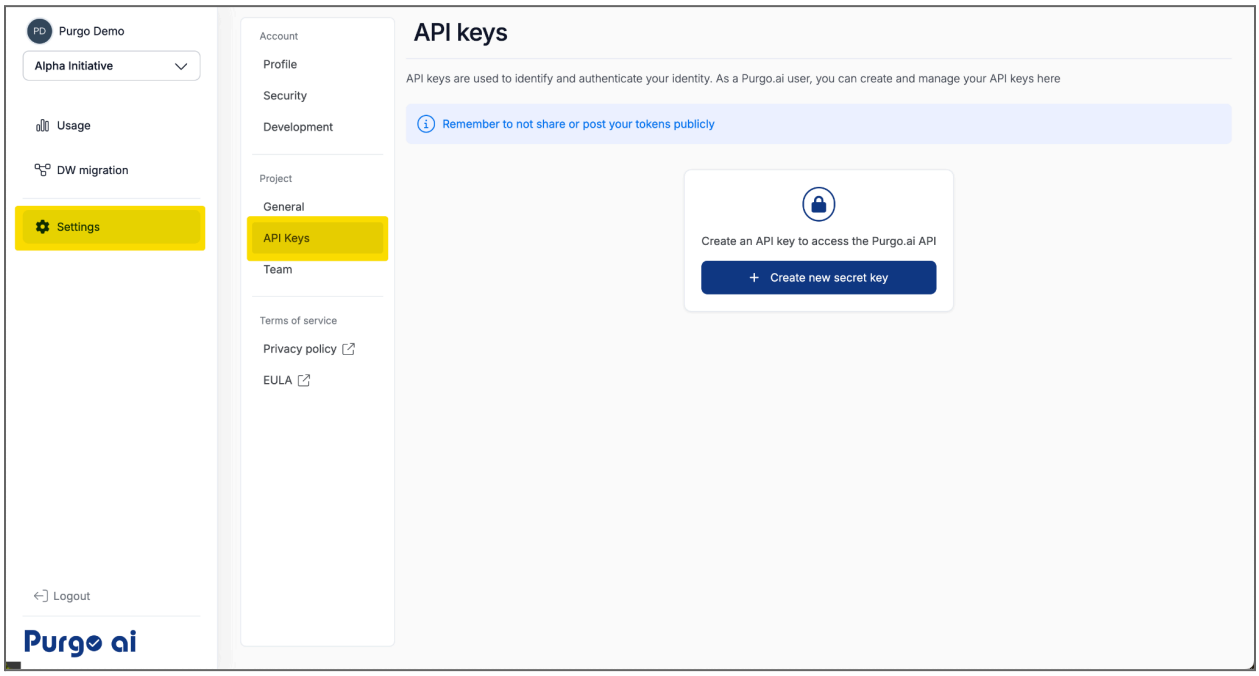
4. API keys settings

The **Admin Panel** in Purgo AI allows users to manage API keys, which are essential for integrating Purgo AI with other platforms like Jira, GitHub, and Databricks. API keys serve as authentication tokens, enabling secure communication between Purgo AI and external services. In this guide, we'll walk you through the process of creating, copying, and managing API keys using the admin panel.

Step 1: Access the Admin Panel

To create or manage API keys, you first need to access the admin settings.

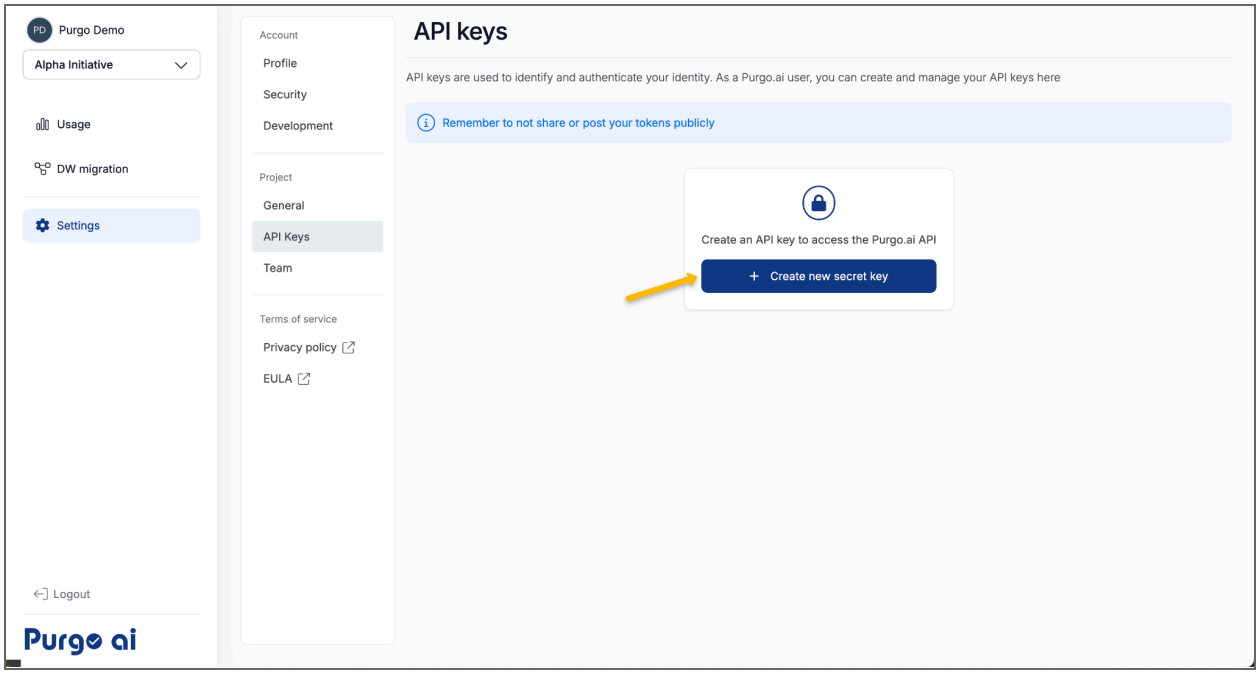
1. **Login to Purgo AI**:
   - Once you are logged into Purgo AI, navigate to the **Admin Panel** by clicking on the settings icon (located at the bottom of the left menu bar).



2. **Settings and API Key Button**:
   - If you are a new user, you will find a button labeled "Create new secret key" under **Settings >> API Key** in the console. Click on this button to proceed.

Alternatively:

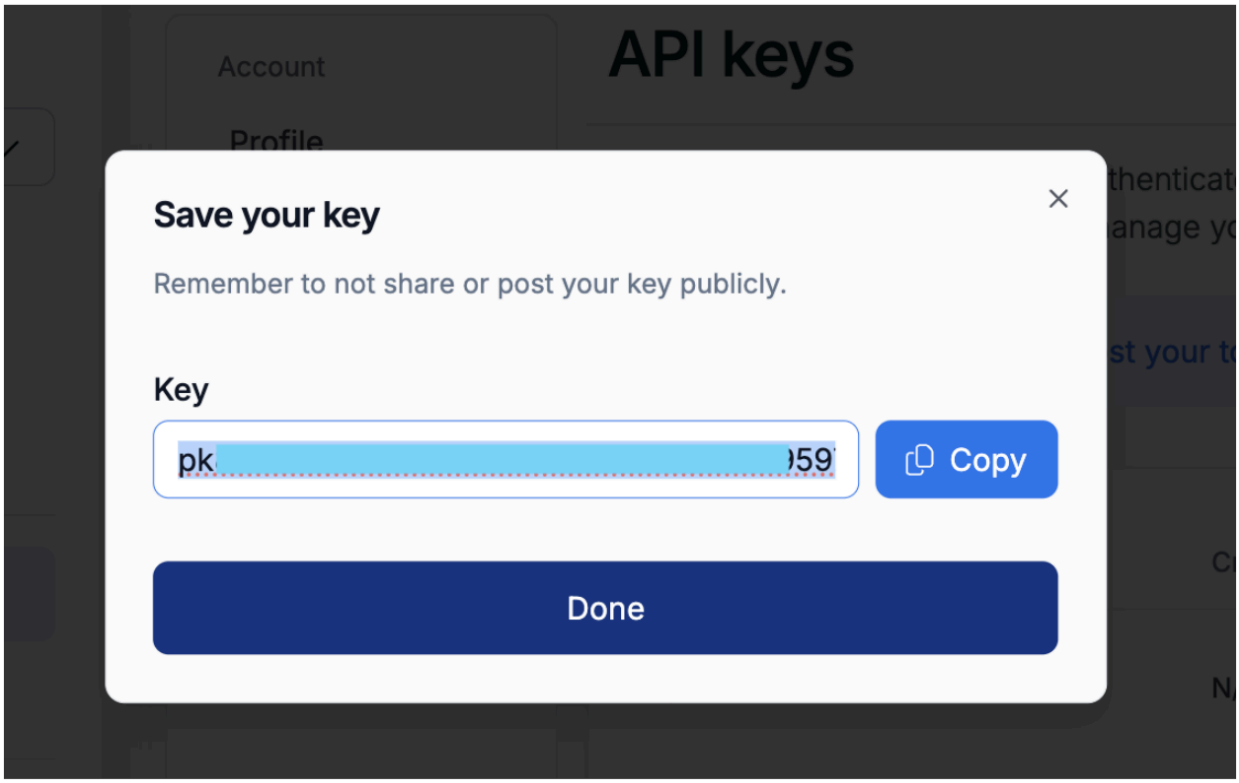
- You can access the API Key management by going to the **API Keys** option in the side menu of the Admin Panel.



Step 2: Create a New API Key

Once you're in the API Keys section of the Admin Panel, follow these steps to create a new API key:

1. **Click the Create New Secret Key Button**:
    - Look for the **Create new secret key** button. Click on this button to generate a new API key.
2. **Generated Key Modal**:
    - A modal (popup window) will appear displaying your newly generated API key. This key is unique and will be required for secure integration with other platforms.

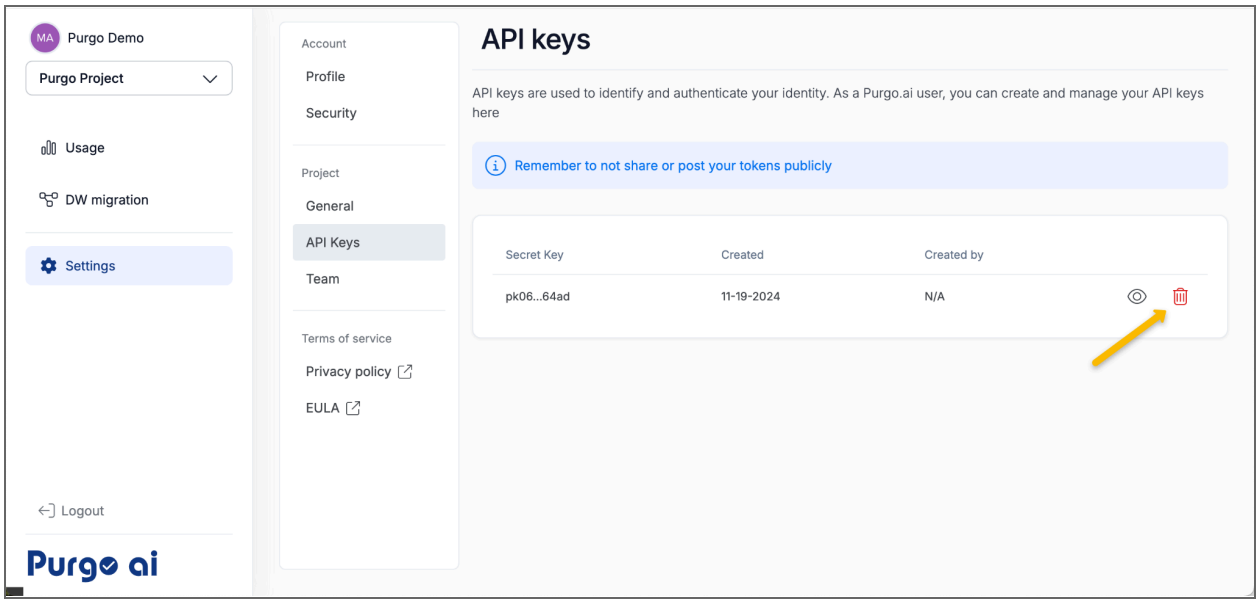


Step 3: Copy and Save Your API Key

1. **Copy the API Key**:
    - In the modal, you'll see the generated API key. Copy the key to a secure location.

Step 4: Managing API Keys

1. **View Existing API Keys**:
    - After you close the modal, the newly created API key will appear in the API Keys section of the Admin Panel.
2. **Delete an API Key**:
    - If you no longer need an API key, you can easily delete it by clicking on the **Delete** button next to the key.



Troubleshooting

Jira User Stories

Problem: Code developed by Purgo AI does not contain a Github Repository link

Solution: If the Github Repo address and Github API Token are not configured for the project on the Purgo AI Console, the Github URL will not be populated for Purgo AI-generated output. Admin should configure these to solve this problem

—

Problem: Code developed by Purgo AI does not contain a Databricks link

Solution: If the Databricks host, token and shared directory are not configured, the Databricks URL will not be generated when running the "Develop Code" step.

—

Problem: The "Requirement" step is executed with a file type other than those supported, and the error message "File type not supported" is shown.

Solution: Supported file extensions for Jira attachments include ['csv', 'xls', 'xlsx', 'xml', 'txt', 'dat', 'json']

—

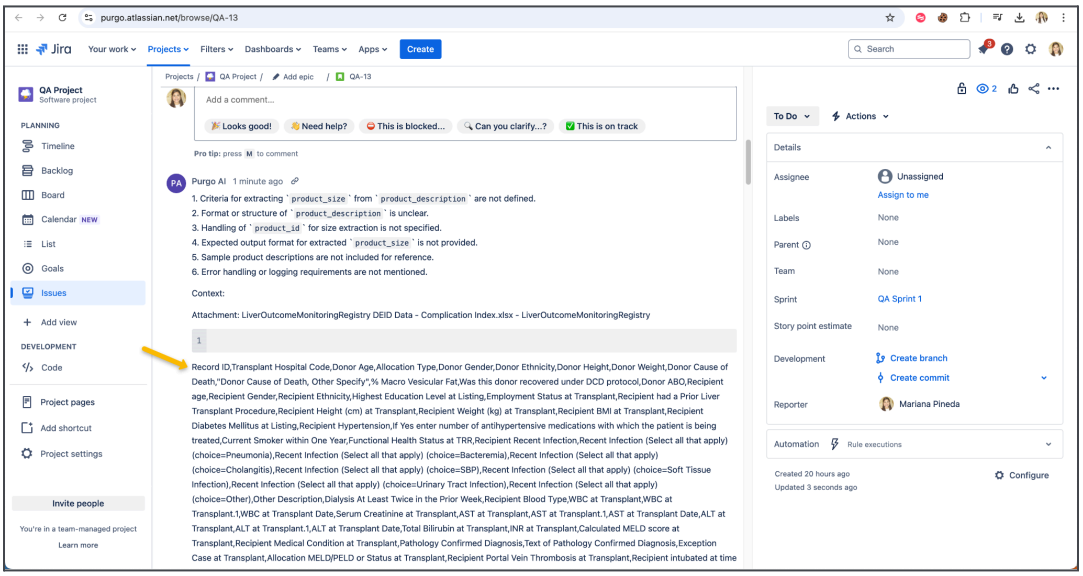
Problem: When running the "Requirement" step with a corrupt file, the error message "File to process attachment" is generated.

Solution: Check the attachment files to ensure the file is not corrupt and can be opened using Excel and/or a text editor, depending on the file type.

—

Problem: When processing a file (.xls, .xlsx), the context result appears outside the "Code snippet" element because the number of columns is too large. The header does not fit within the 32K buffer limit.

Solution: Reduce the number of columns in the file header to fit within the 32K buffer limit, or split the data across multiple sheets/files to ensure compatibility with the "Code snippet" element display constraints.



—

Problem: When running the "Requirement" step with a file larger than 5MB, the error message "Attachment is too large to process" is shown.

Solution: The current attachment file size limit is 5MB per file. Reduce the file size to meet this limit, or split the data across multiple smaller files to proceed with processing.

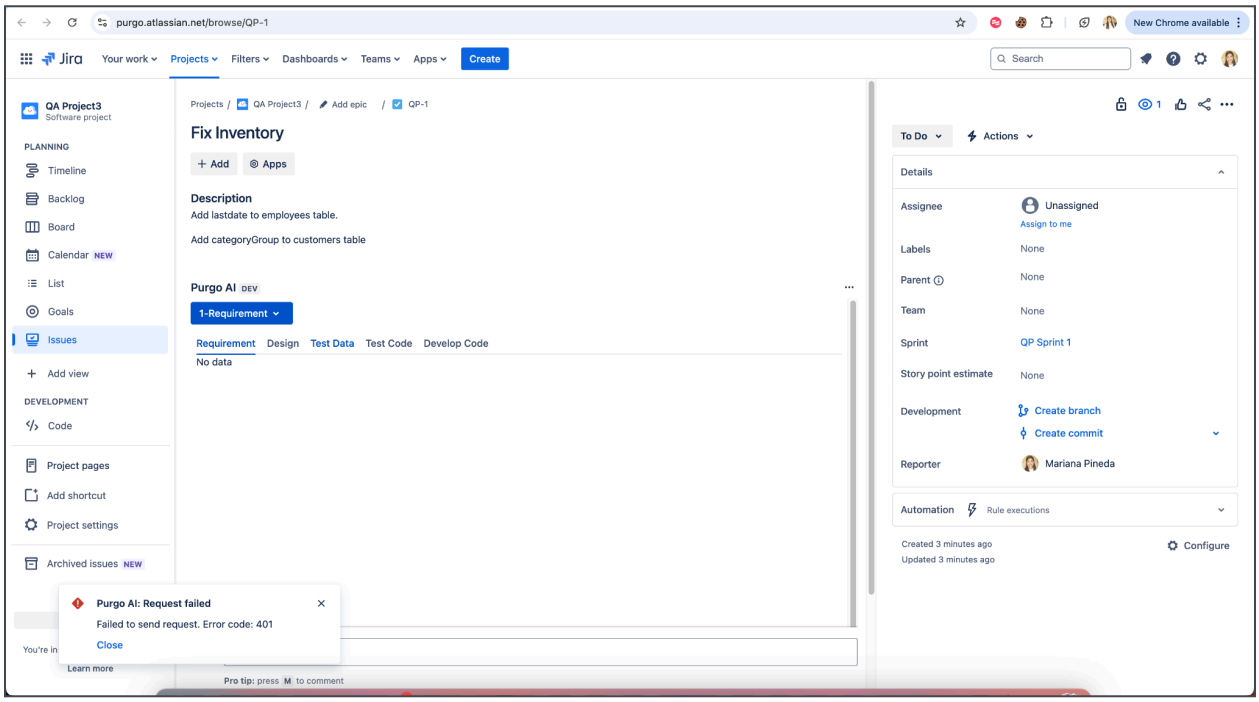
—

Problem: When accessing the Purgo AI app in Jira, the error message "Purgo AI: Request failed" appears.

Solution: This issue often arises from a mismatch between the API Key configured in Jira and the one generated in the Purgo AI Admin Console. To resolve this:

1. Log in to the Purgo AI Admin Console at https://app.purgo.ai.
2. Navigate to the Project API Key section.
3. Verify that the API Key matches exactly with the one configured in your Jira Project.

Ensuring both keys are identical should resolve the "Request failed" error.



Jira Admin Configuration

Problem: During Purgo AI configuration, the console displays the error: "Unauthorized: Invalid Purgo API Key."

Solution: This error typically indicates a mismatch between the API key configured in your Jira project and the one generated in the Purgo AI Admin Console. To resolve this issue:

1. Log in to the Purgo AI Admin Console at https://app.purgo.ai.
2. Navigate to the Settings -> Project -> API Key section.
3. Verify that the API key matches exactly with the one configured in your Jira project.

Ensuring both keys are identical should resolve the "Unauthorized: Invalid Purgo API Key" error.

provides users with an intuitive interface to manage various settings related to your projects, personal information, and security. Below, we will guide you through the different sections of the Admin Panel, focusing on **General Settings**, **Security Settings**, **API Keys** and **Profile Settings** to help you configure your Purgo AI experience efficiently.