From Your Repository to the RAD Platform
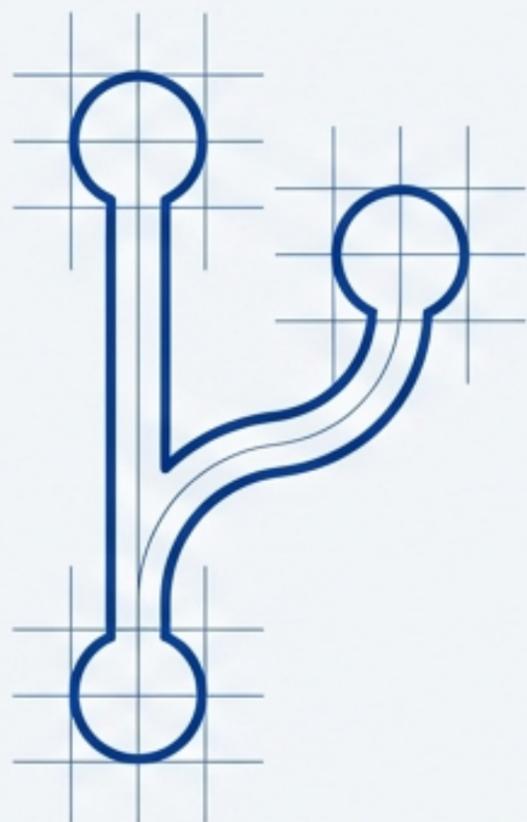
A Partner's Guide to Publishing and Deploying Private Modules

1. CONNECT   2. PUBLISH   3. DEPLOY   4. MANAGE

NotebookLM

Transform Your Code into a Deployable Asset

This guide walks you through the simple, secure workflow for bringing your private Git repositories into the RAD Platform. By following these steps, you will publish your custom solutions as first-class, private modules that only your team can deploy and manage.

Direct Integration

Connect directly to your existing GitHub repositories.

Total Privacy

Your modules are visible only to you and your organization's Administrators.

Seamless Deployment

Deploy your private modules using the same familiar RAD Platform interface.

First, let's securely link your GitHub repository to your RAD Platform profile.

This one-time setup authorizes the RAD Platform to see your repositories. You will need a GitHub Personal Access Token with `repo` scope to complete this step.

1 CONNECT — **2.** PUBLISH — **3.** DEPLOY — **4.** MANAGE
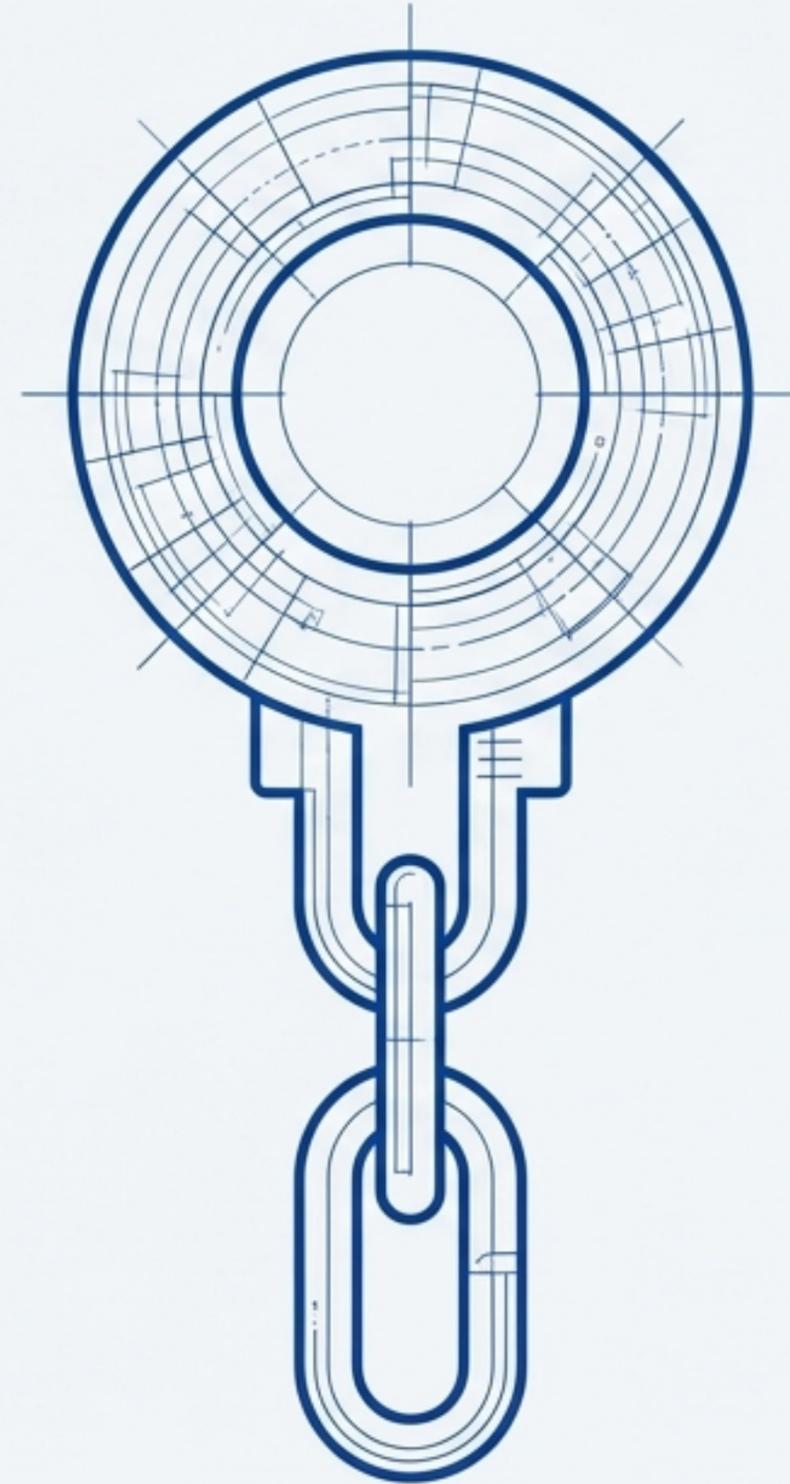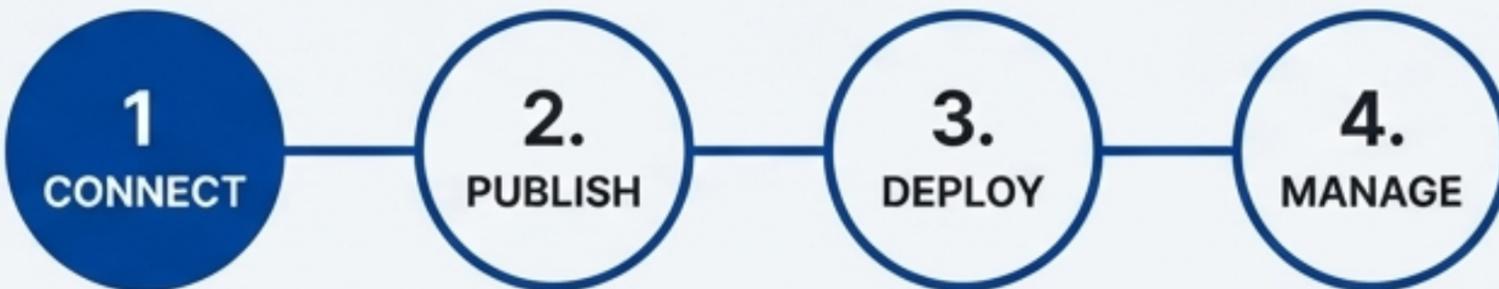
Connect Your GitHub Account

1. Navigate to your **Profile Avatar** and select **Profile**.

2. Scroll down to the **Partner Settings** section.

3. Enter your **GitHub Access Token**. (Important: Ensure the token has `repo` scope.)

4. Click the **Fetch Repos** button to populate the list.

5. Select your target **Repository** from the dropdown menu.

6. Click **Save Github Settings** to confirm.

NotebookLM
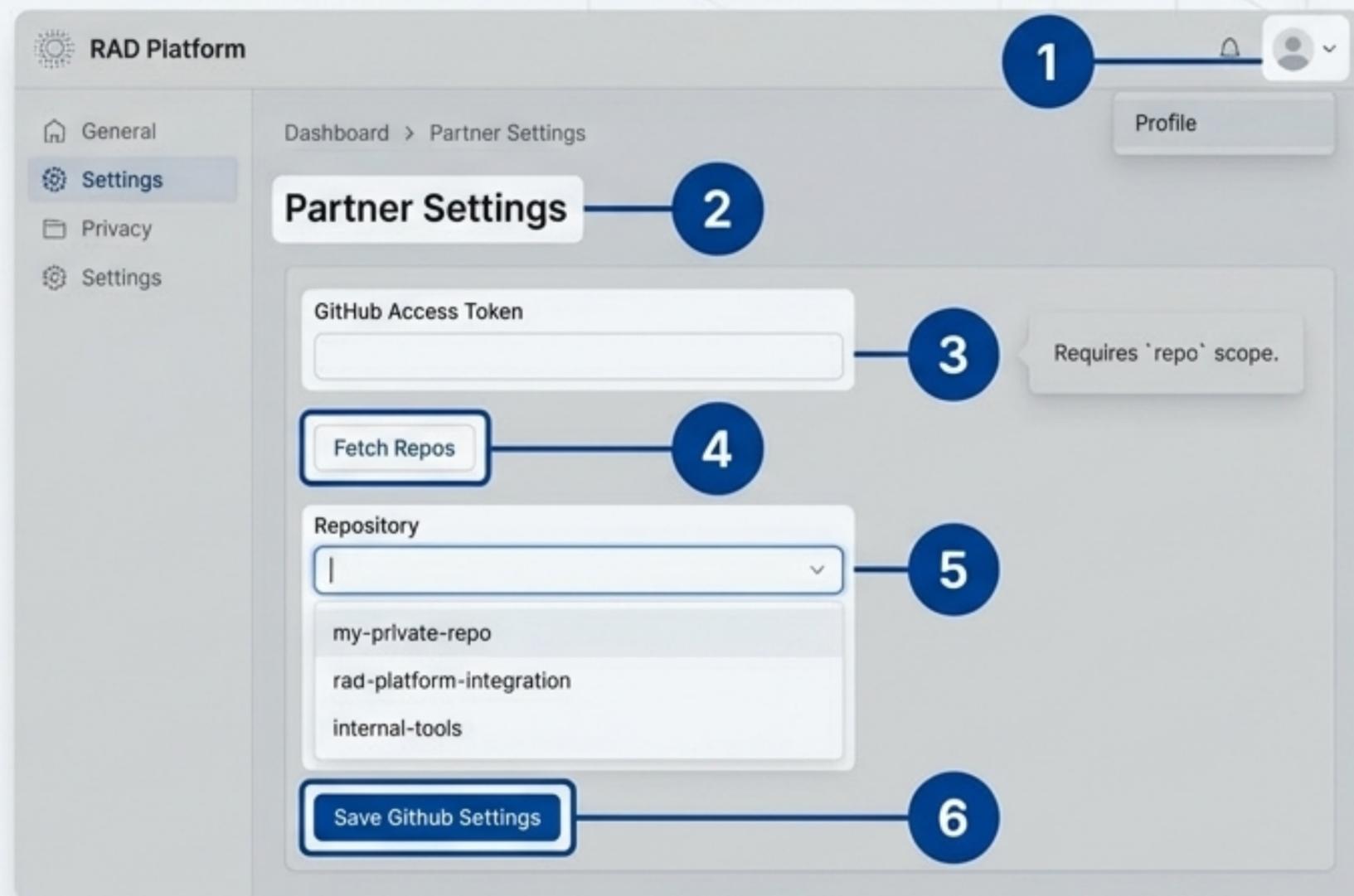
Optional Power-Up: Enhance Your Module with Jules AI

If you use our AI refinement agent, Jules, you can connect it here to work with your private modules. This step is not required to publish or deploy.

Action Steps

1. In the same **Partner Settings** section, locate the Jules API Key field.

2. Enter your **Jules API Key**.

3. Click **Save API Settings**.

1 CONNECT ——— **2** PUBLISH ——— **3** DEPLOY ——— **4** MANAGE
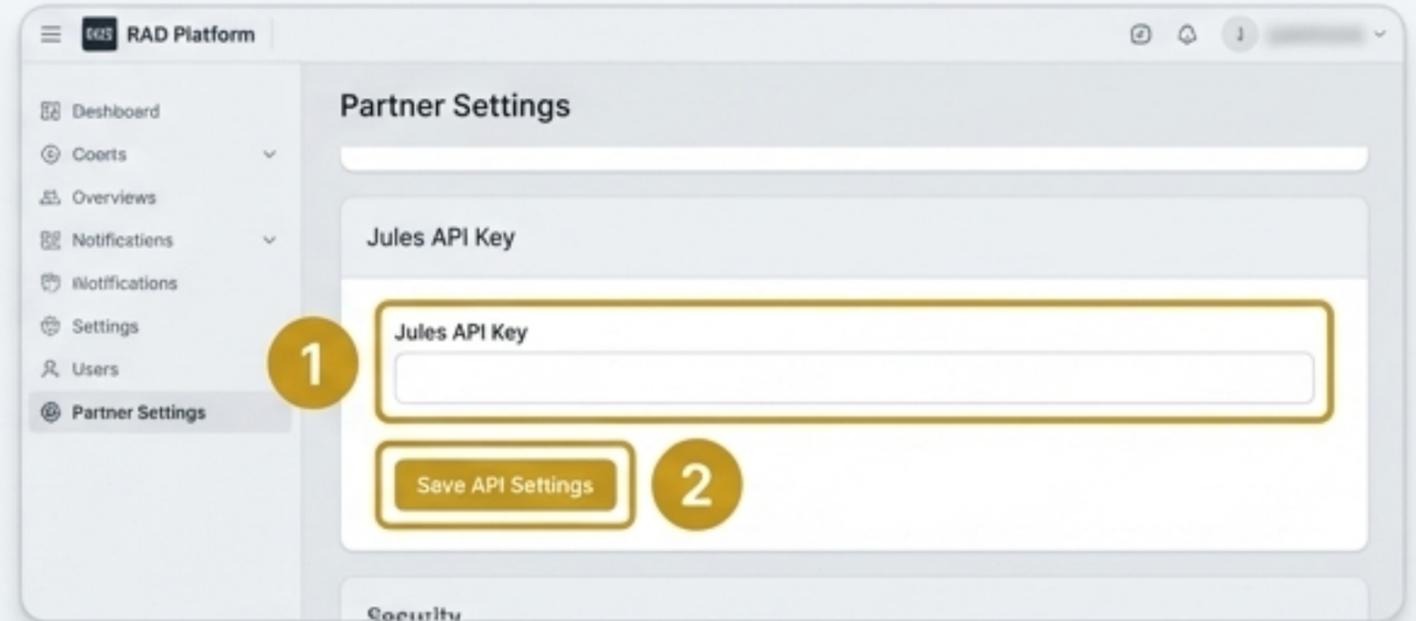
# With your repository connected, you can now publish your modules.

Publishing makes your module visible within the RAD Platform's deployment interface, ready for you and your team to use. The platform will scan your selected repository and automatically identify all valid modules.

1. CONNECT ——— **2. PUBLISH** ——— 3. DEPLOY ——— 4. MANAGE

NotebookLM
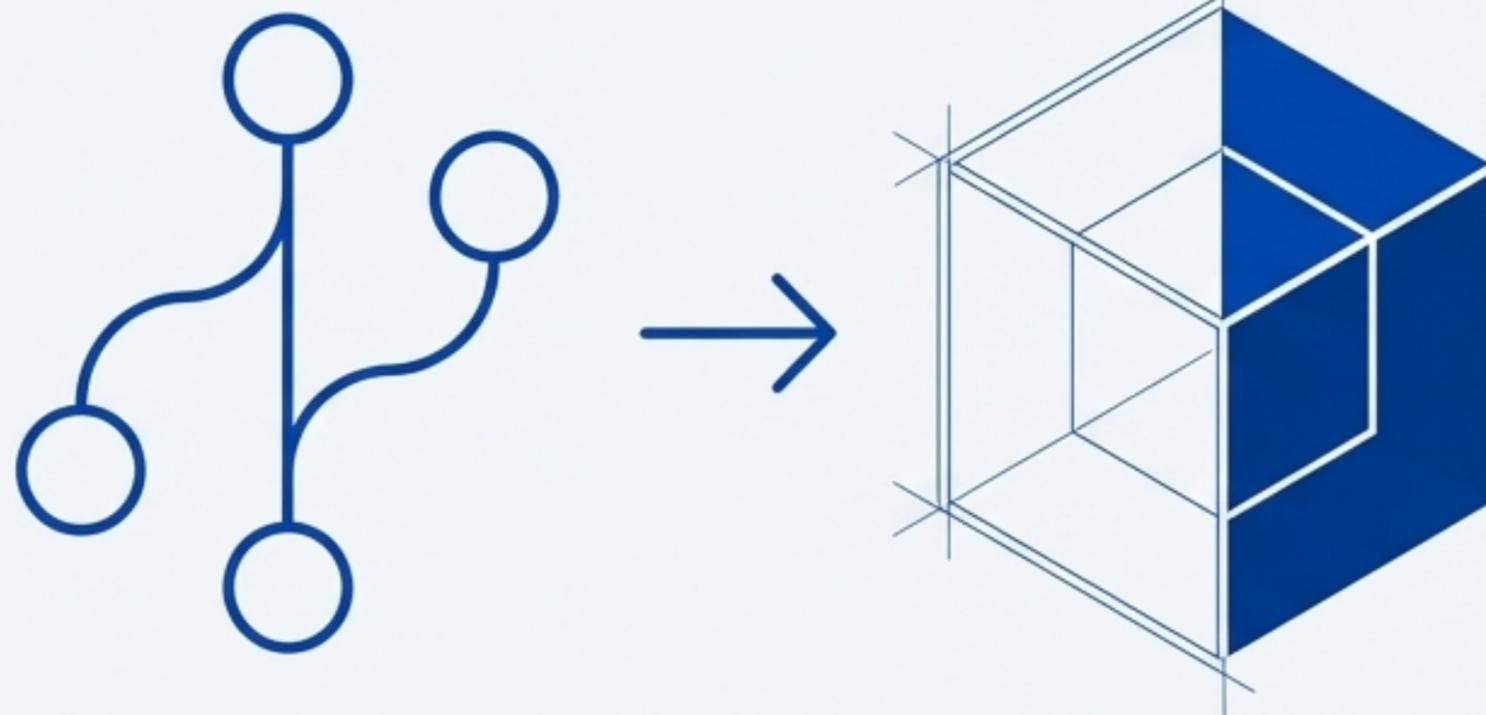
Select and Publish Your Module

1. Navigate to the **Publish** page from the main navigation bar.

2. The platform will automatically scan your connected repo and display all valid modules as cards.

3. Find the module you wish to publish (e.g., `my-custom-app`) and **click its card** to select it.

4. Click the **Publish** button. (This button will show **Update** for modules you've previously published).

Success! After publishing, you will be automatically redirected to the Deploy page.
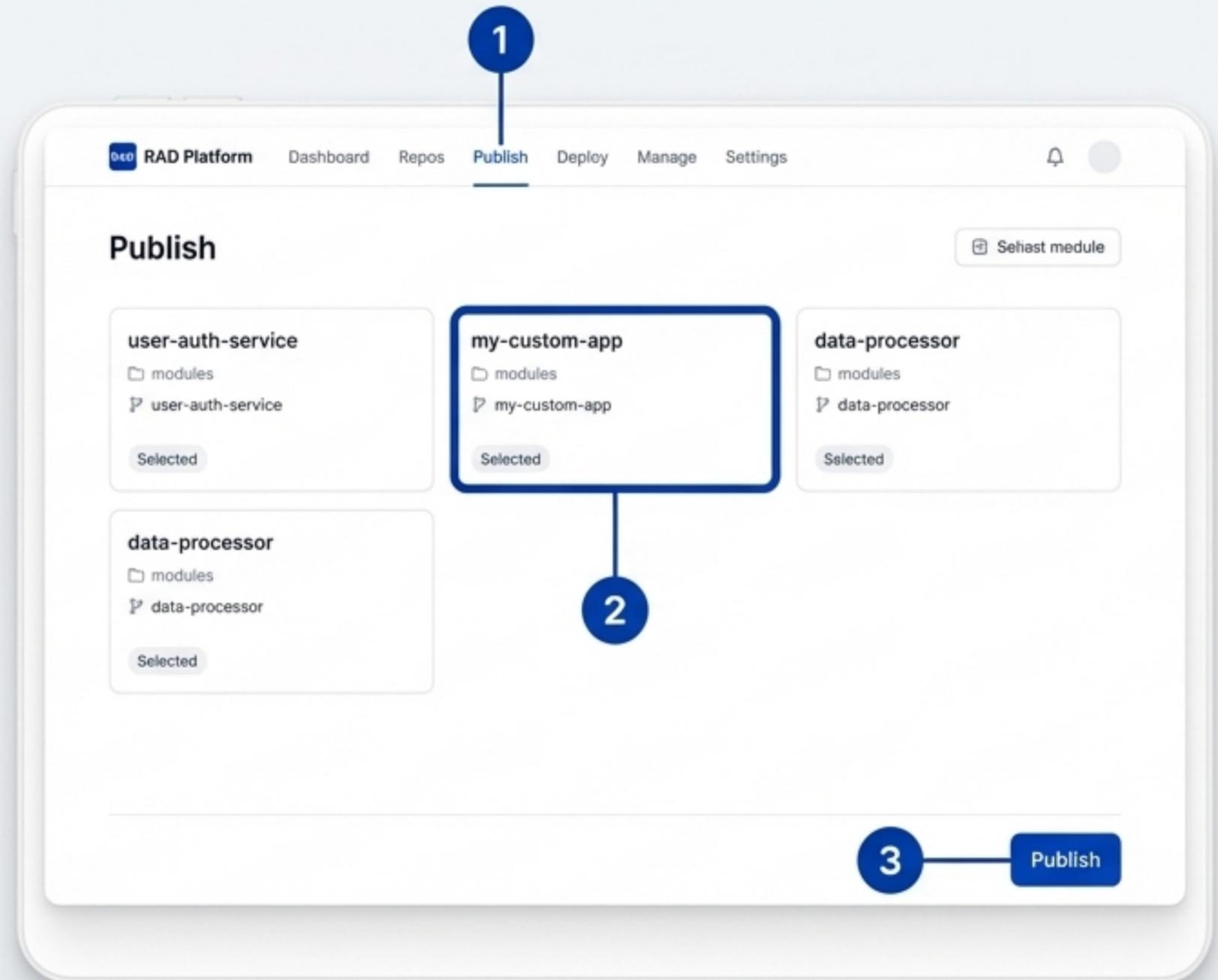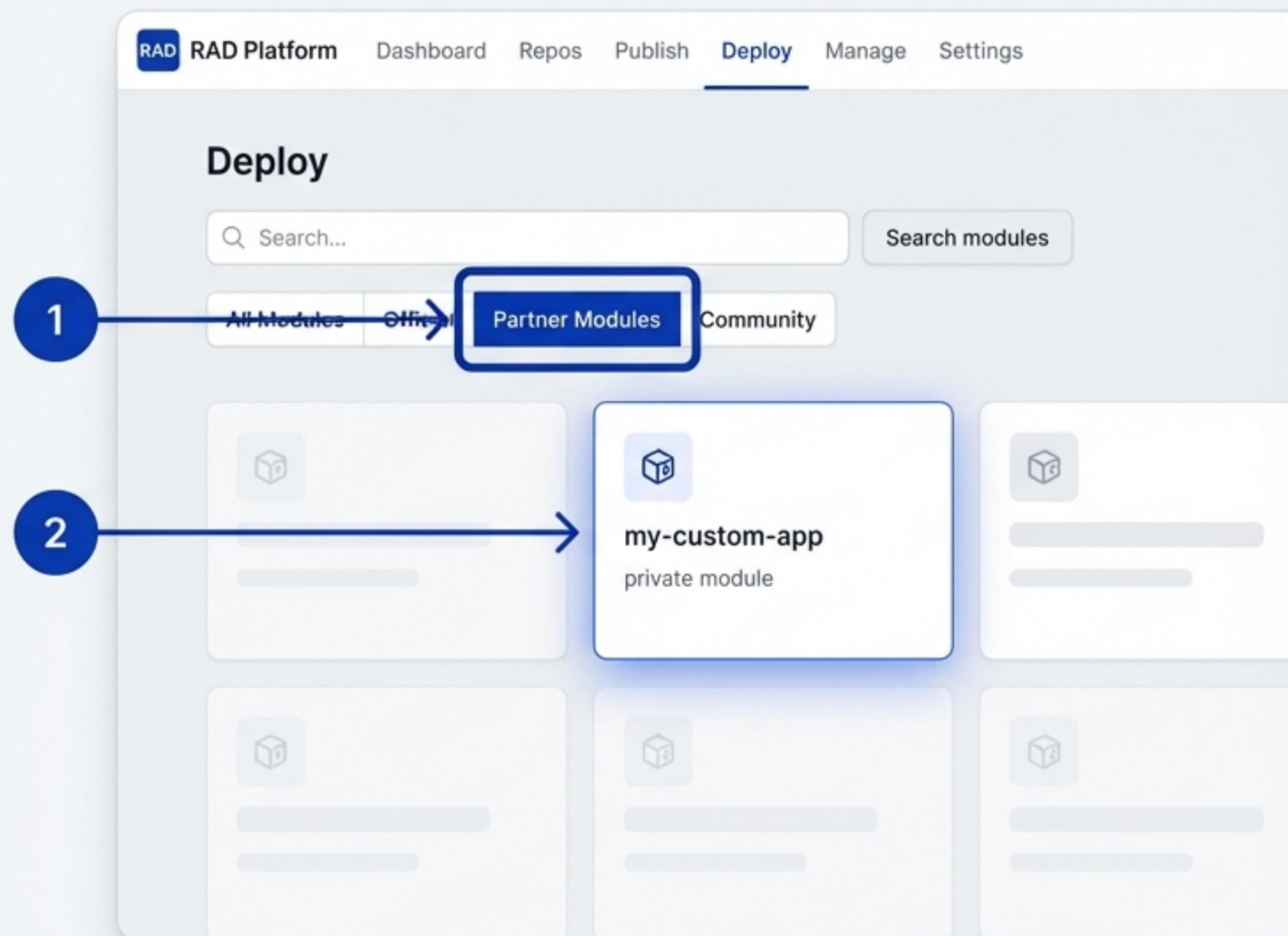
Your private module is published. It's time to bring it to life.

Deployment works just like any standard RAD Platform module. Your private module now appears in a dedicated tab on the Deploy page, ensuring it's easy to find and completely separate from public modules.

1. CONNECT   2. PUBLISH   3. DEPLOY   4. MANAGE
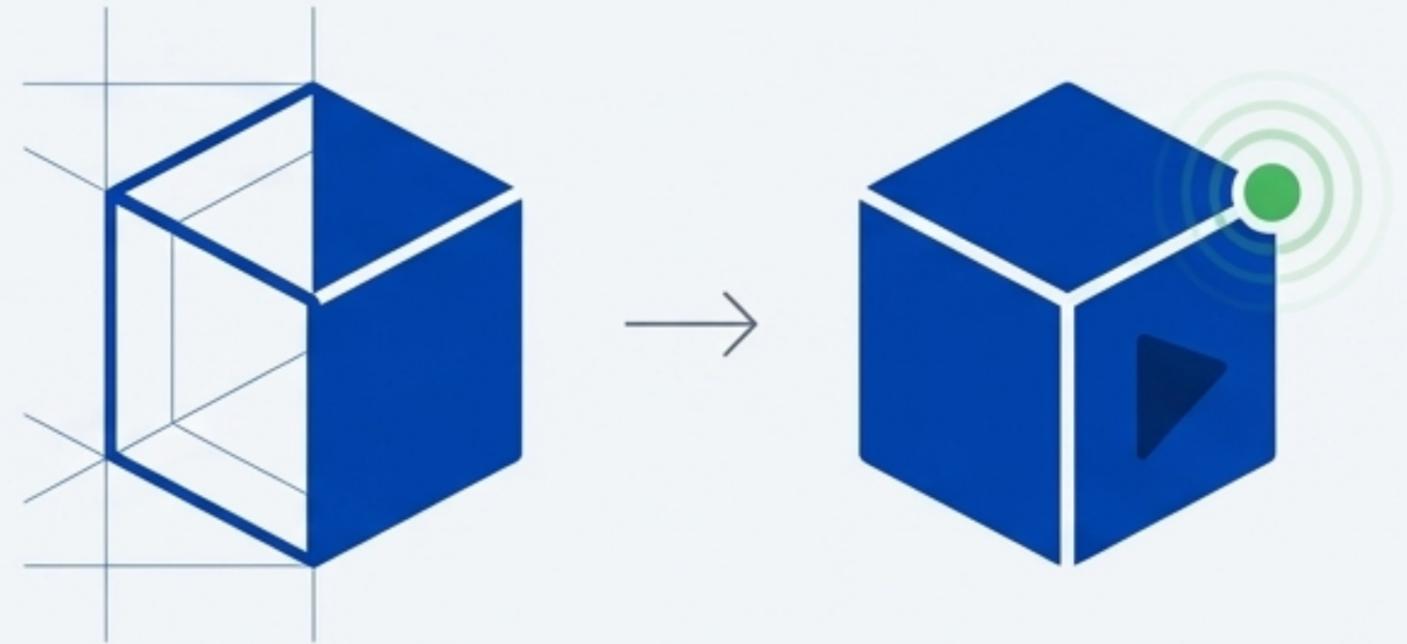
Deploy Your Private Module

(1) On the **Deploy page**, select the **Partner Modules** tab.

(2) You will see **your published module** (my-custom-app) listed.

(3) **Click the module** to open the configuration screen.

(4) **Configure and deploy it** just as you would with any standard module.

RAD **RAD Platform**   Dashboard   Repos   Publish   **Deploy**   Manage   Settings

Deploy

🔍 Search...   Search modules

1 → ~~All Modules~~   ~~Official~~   **Partner Modules**   Community

2 →

my-custom-app
private module

Your Module, Your Control.

Remember: Only you and Administrators in your organization can see this module.

Your intellectual property is protected. Private modules never appear in the public catalog or for other partners. They are scoped exclusively to yr organization, providing a secure way to deploy proprietary technology on the RAD Platform.

The Private Module Workflow at a Glance

1. CONNECT

Add GitHub Token & select repo in Profile -> Partner Settings.

2. PUBLISH

Go to Publish, select your module card, and click Publish.

3. DEPLOY

Find your module in the Deploy -> Partner Modules tab and launch.

4. MANAGE

Update your module by re-publishing from the same branch.

Optional: Add Jules AI Key

Your Module is Live.

You have successfully connected your repository, published your code, and deployed it as a private module. It is now a secure, first-class citizen within the RAD Platform, fully integrated into your workflow and visible only to your team.

1. CONNECT   2. PUBLISH   3. DEPLOY   4. MANAGE