



# Building APIs Together: How to Embrace a Multiplayer Strategy

**Alex Barnett,**  
Head of Developer Relations, Postman

 [@alexbarnett](#)   [@getposman](#)





# Building APIs Together: How to Embrace a Multiplayer Strategy

**Alex Barnett,**  
Head of Developer Relations, Postman

 [@alexbarnett](#) [@getposman](#)





# Every day, millions of developers depend on Postman to bring their APIs to life

**35M**

Developers

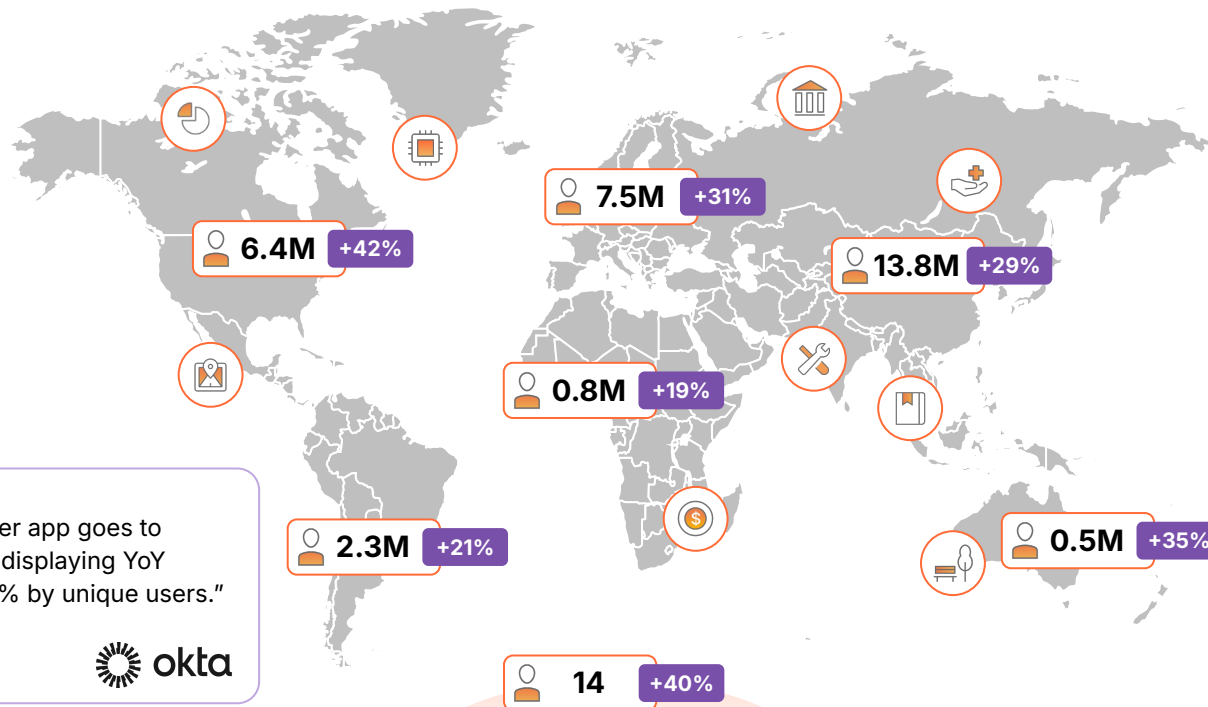
**98%**

Of the Fortune 500

## Developers' Best of Show

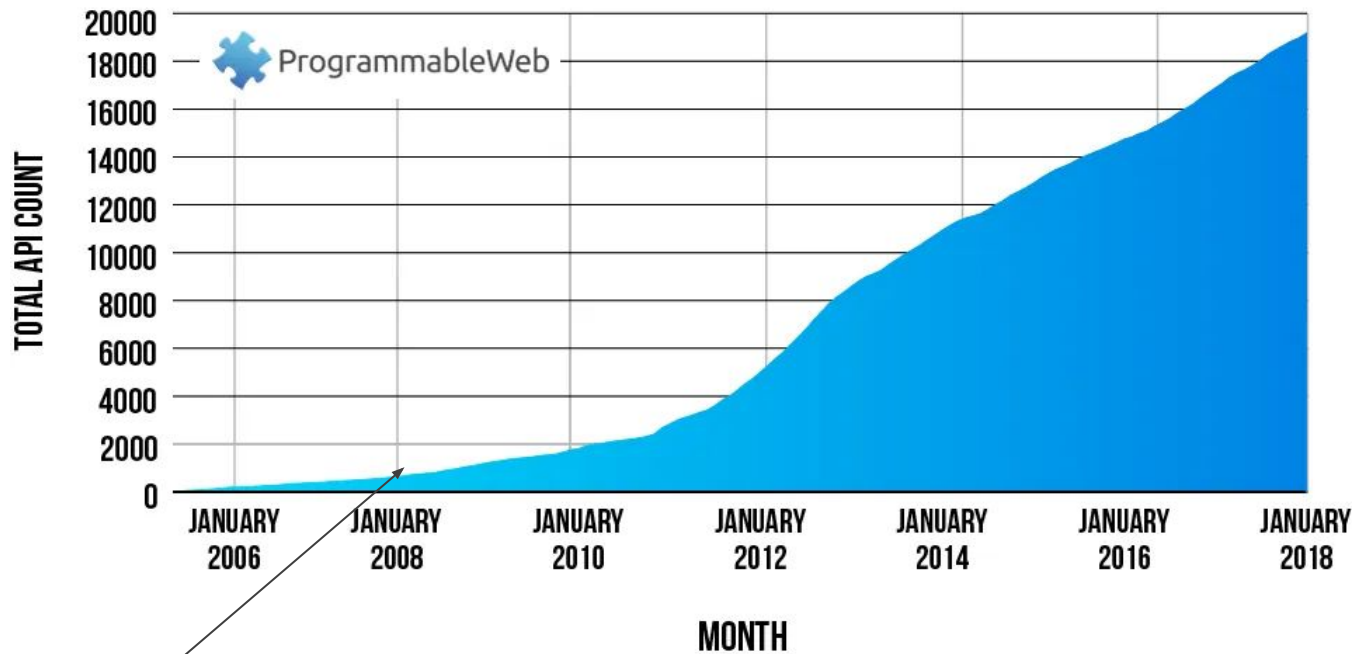
“The prize for fastest growing developer app goes to Postman...as the clear growth leader, displaying YoY growth of 78% by customers and 168% by unique users.”

Businesses at Work 2023





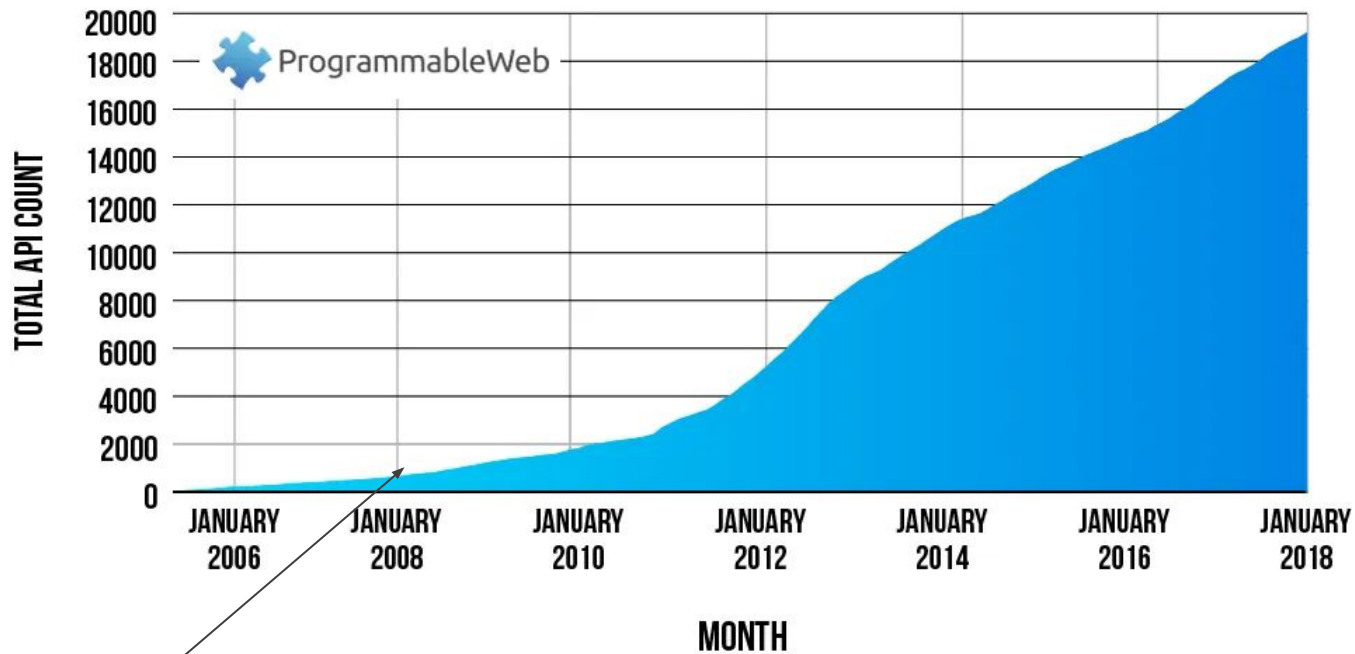
## GROWTH IN WEB APIS SINCE 2005



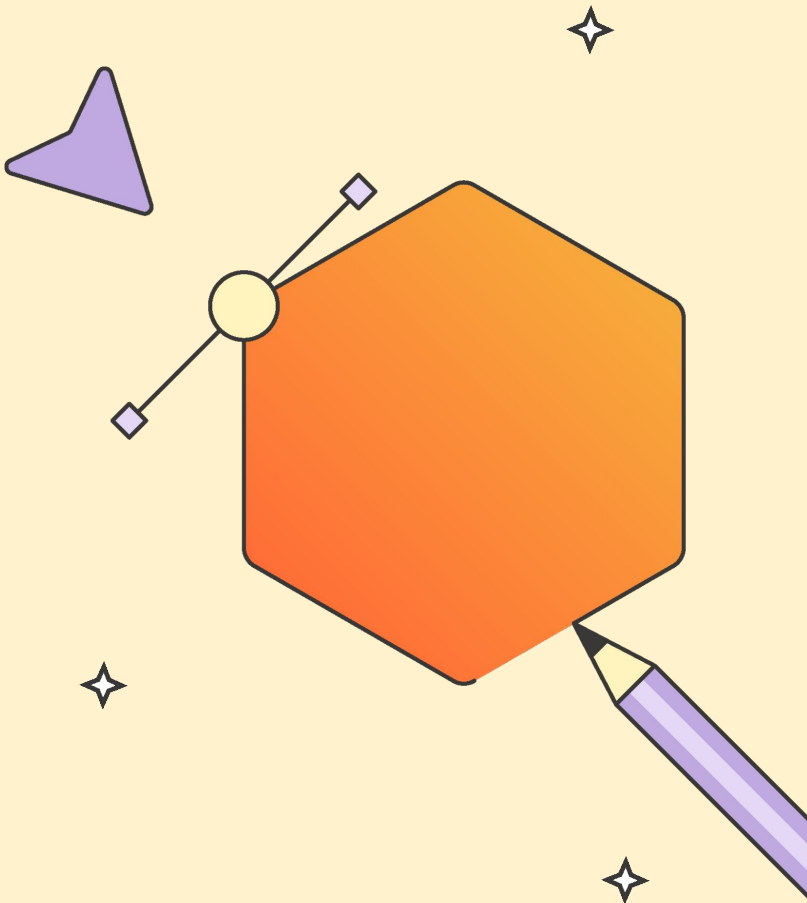
**Wow!!!  
(lol.)**



## GROWTH IN WEB APIS SINCE 2005



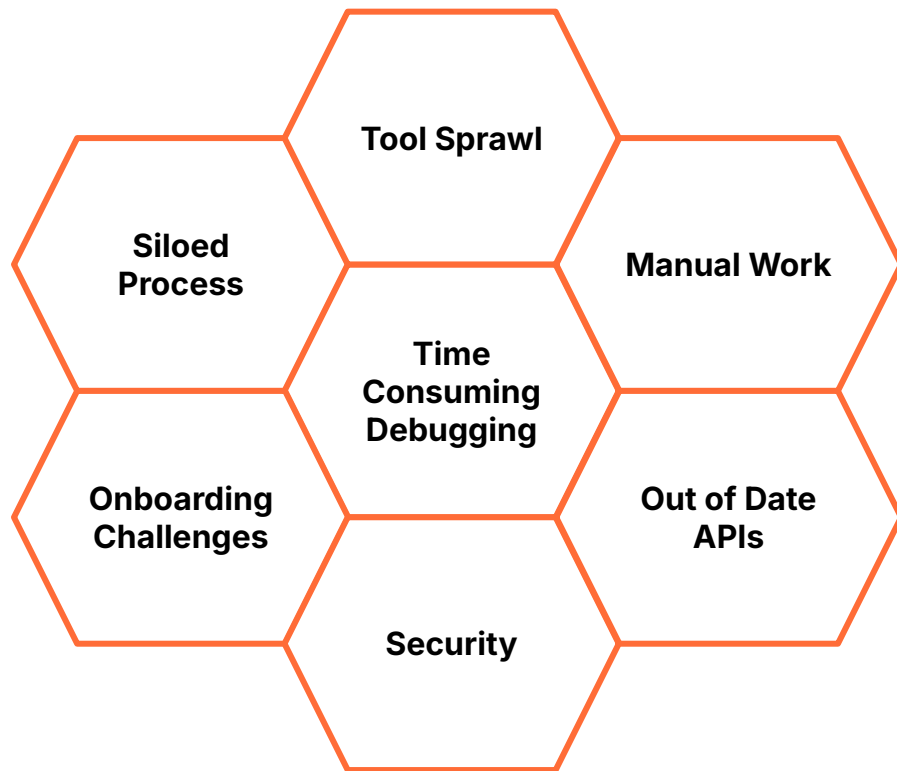
**Wow!!!  
(lol.)**



# Complexities of **API** Development



# Pitfalls of Traditional, Siloed Processes





# Limitations of the 'Single Player' Approach

**DUPLICATE WORK**

**VERSIONING ISSUES**

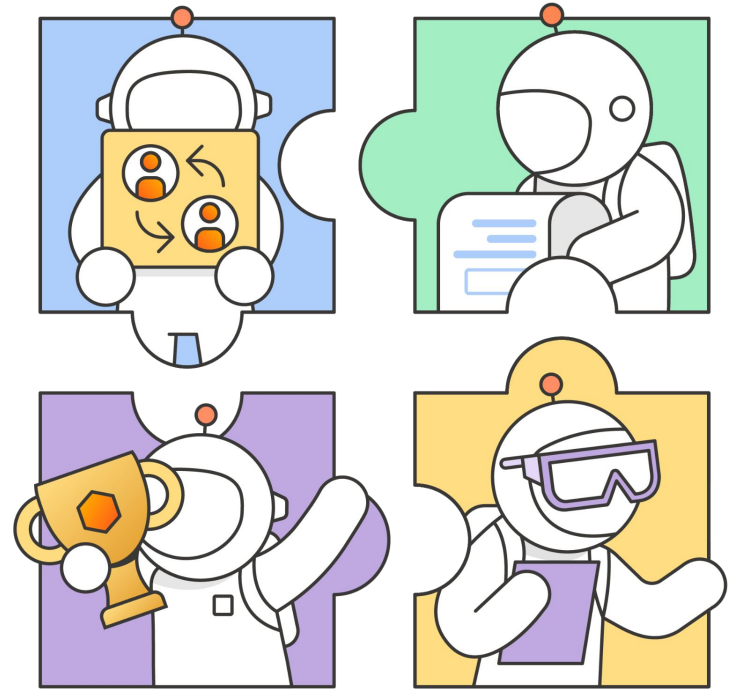
**INEFFICIENT COLLABORATION**







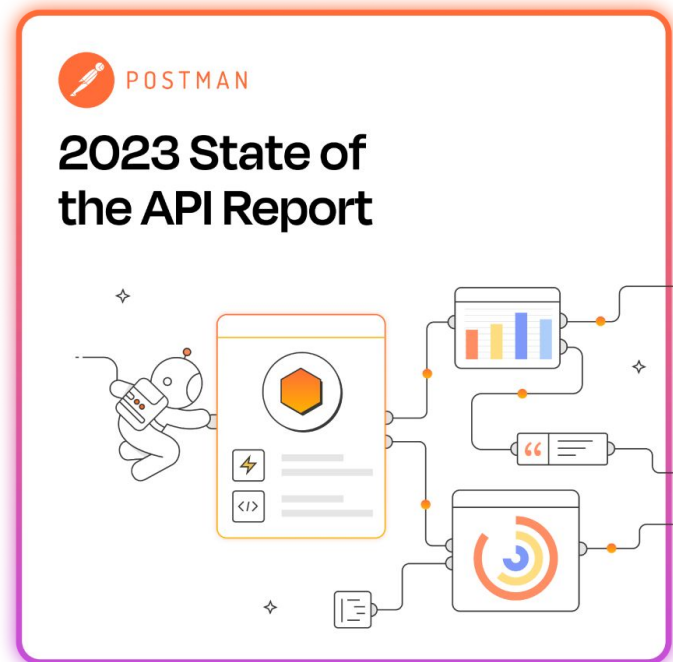
# The Need for a **Collaborative** Approach





# Why **Collaborating** on APIs Matters

What % of their APIs were public, private, or partner APIs?





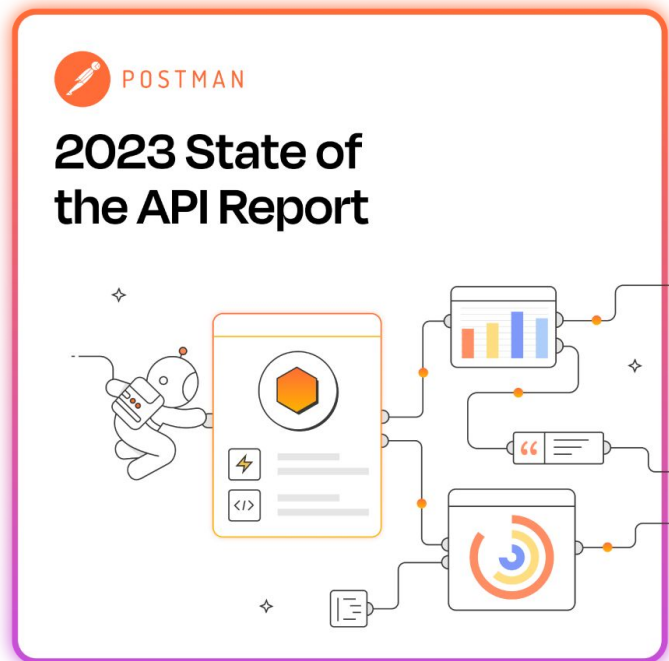
# Why **Collaborating** on APIs Matters



What % of their APIs were public, private, or partner APIs?

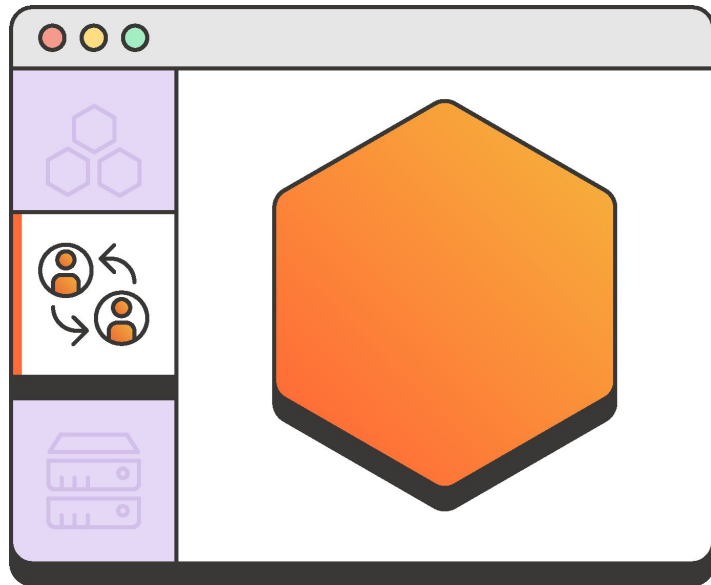
—

61% percent of businesses' APIs are for internal use only





# A Single Platform for **Collaborative** API Development





# A Single Platform for **Collaborative** API Development

Accelerate API Development

Discover, share, build, and run  
tests **together**



# A Single Platform for **Collaborative** API Development

**Accelerate API Development**

Discover, share, build, and run tests **together**

**Improve API Quality**

Continuous testing, mocking, feedback & debugging **together**



# A Single Platform for **Collaborative** API Development

## Accelerate API Development

Discover, share, build, and run tests **together**

## Improve API Quality

Continuous testing, mocking, feedback & debugging **together**

## Enhance Real-Time Collaboration

Organize API collections within Workspaces, contributing **together**



# A Single Platform for **Collaborative** API Development

## Accelerate API Development

Discover, share, build, and run tests **together**

## Improve API Quality

Continuous testing, mocking, feedback & debugging **together**

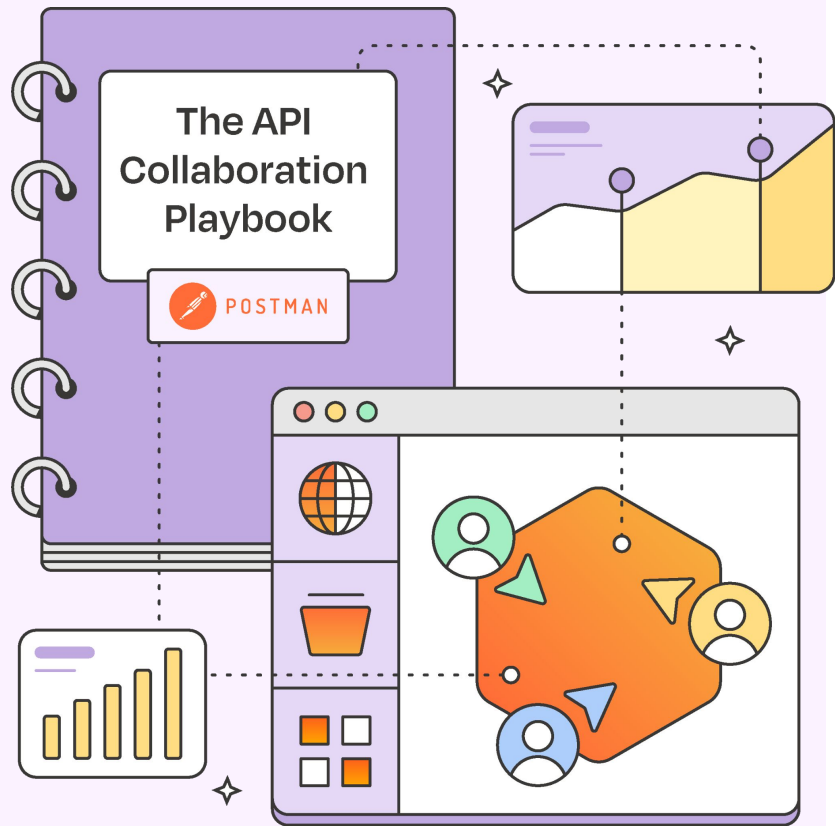
## Enhance Real-Time Collaboration

Organize API collections within Workspaces, contributing **together**

## Scale Securely

With robust governance and compliance, safeguarding security & privacy with local data residency

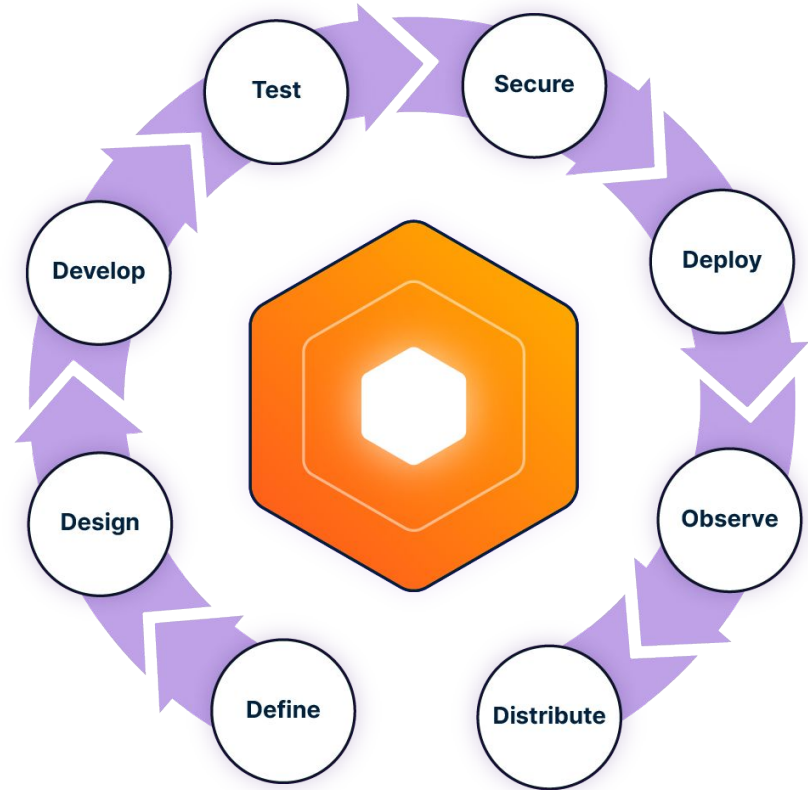


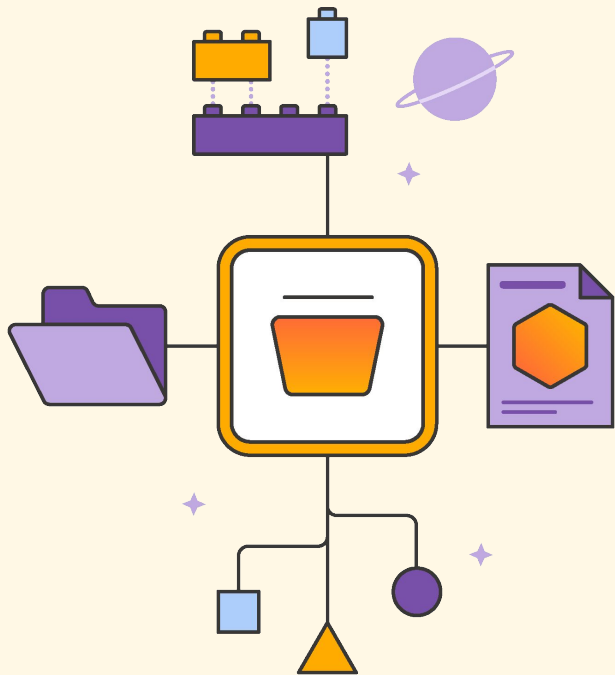


# Key Features For a Multiplayer Platform



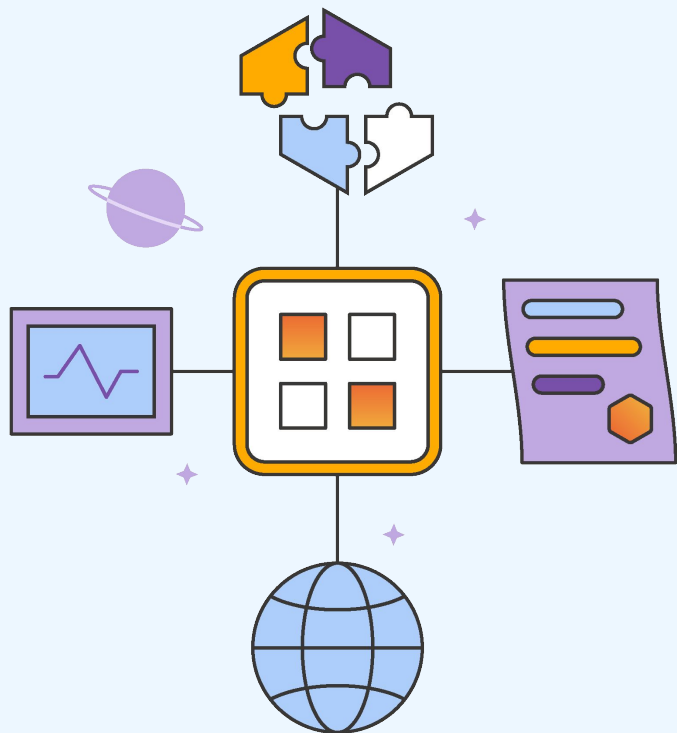
# API Development Landscape





# Collections

the building blocks of  
API development



# Workspaces

the shared canvas for  
executing & collaborating  
on API assets



# Workspace Types: Tailored to Different **Collaboration** Scenarios



**Public**

Publish APIs in the Public Network, making assets in workspaces, API collections for **anyone on the internet.**



**Private**

Private API Network offers a central directory of workspaces, collections, and APIs for **internal team use.**



**Partner**

Partner Workspaces allow **collaboration with external partners**, enabling them to consume your team's APIs and collaborate on API projects

# Post Updates



The screenshot shows the Postman API workspace interface. At the top, there are navigation tabs for 'Home', 'Workspaces', and 'API Network'. A search bar labeled 'Search Postman' is located in the top right. Below the navigation, the workspace name 'Postman API' is displayed, along with 'New' and 'Import' buttons. The main content area is titled 'Overview' and includes sub-tabs for 'Overview', 'Updates', 'BETA', and 'Settings'. The 'Updates' tab is currently selected. The main content area is divided into three sections: 'Workspace description', 'About', and 'Pinned collections'. The 'Workspace description' section contains a heading, a paragraph, and a bulleted list. The 'About' section contains a heading, a paragraph, and a 'Contributors' list. The 'Pinned collections' section contains a heading and a paragraph. A footer bar at the bottom of the interface lists various tools and services: Postbot, Runner, Auto-select agent, Cookies, Vault, Bifrost, br-feature-workspace-updater, Trash, and a refresh icon.

Home Workspaces API Network Search Postman

Postman API New Import Overview + No Environment

Postman API Overview Updates BETA Settings

**Workspace description**

Add a description to accelerate team collaboration. Here's what you can include:

- Introduction
- How-to-guides
- Best practices
- FAQs

**About**

This is a dummy security workspace

**Contributors**

- You

**Pinned collections**

Make it easy to discover notable collections in this workspace by pinning them here.

[Explore workspace templates](#)

Online Console Postbot Runner Auto-select agent Cookies Vault Bifrost br-feature-workspace-updater Trash

# Add Comments



Home Workspaces ▾ API Network ▾ Search Postman

API Doc Reference New Import

Overview **POST Create a collection** +

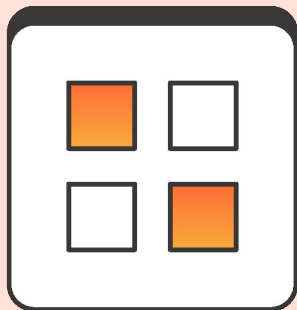
API Documentation #reference / Collections / **Create a collection**

**POST** ▾ `https://api.getpostman.com/collections`

Params • Authorization Headers (9) Body • Scripts Settings

Query Params

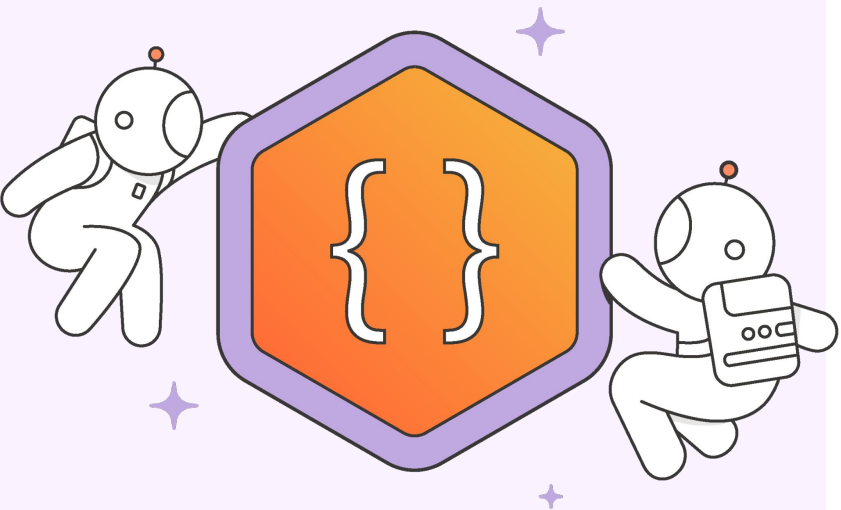
<input type="checkbox"/>	Key	Value
<input type="checkbox"/>	workspace	<code>{{workspaceid}}</code>
<input type="checkbox"/>	Key	Value



# Collections + Workspaces

Changes API development from  
a single player activity to  
multiplayer enterprise.





# Scripts

the logic provided to drive  
dynamic workflows

# Package Library

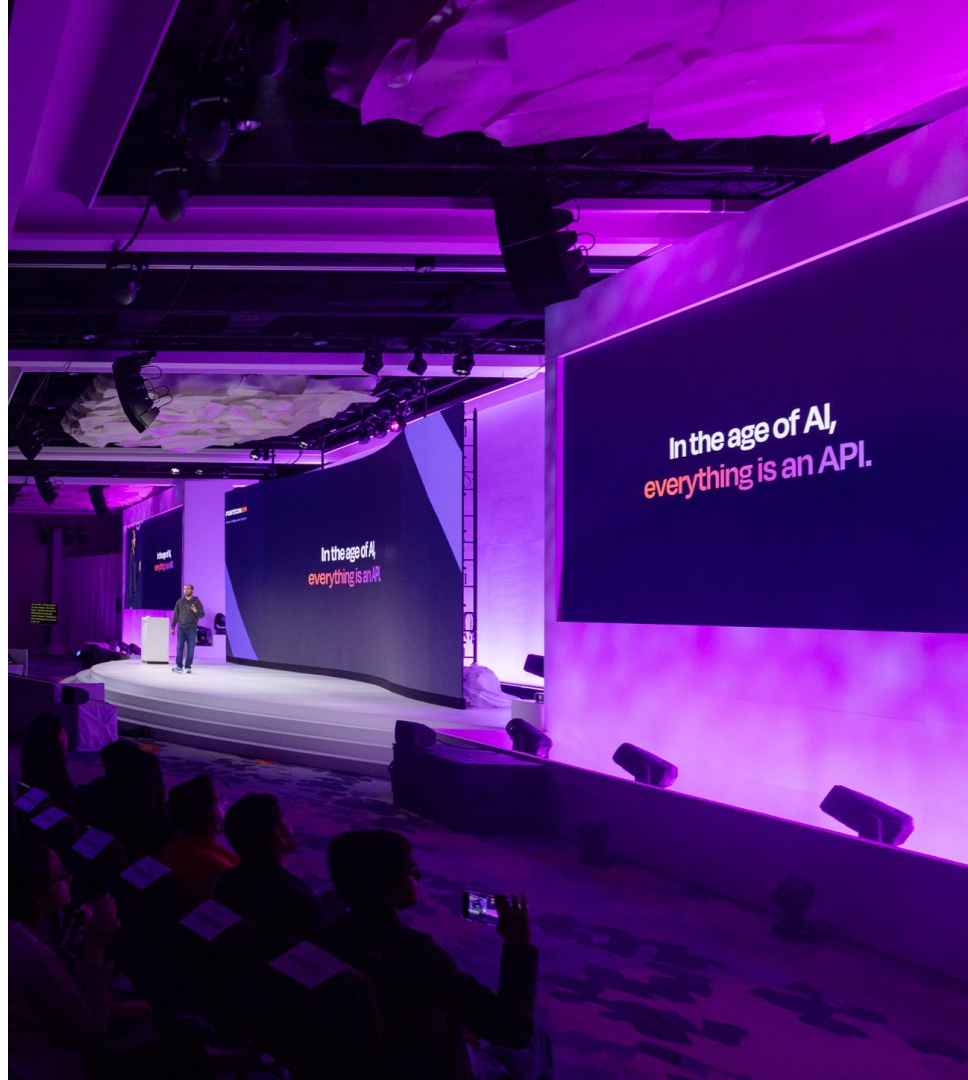


The screenshot shows the Postman Package Library interface. On the left, there's a sidebar with 'Collections' and 'Environments'. The main area displays a list of packages under the 'Package Library' tab. A search bar is present at the top of the list. The package 'sanity-tests' is highlighted, and its details are shown in a modal window. The modal displays the package name 'sanity-tests', the description 'Common Sanity tests', and the author 'Shobhit Katikia' with a profile picture and the text 'Last updated Apr...'. Below the modal, a list of other packages is visible, including 'stringutils', 'test-json-response-schema', and 'endpoint-and-regex-id-check'. On the right side, a code editor shows the JavaScript code for the 'sanity-tests' package, including a 'sanityCheck' function and a 'checkString' function. A callout box highlights the 'module.exports' line in the code.

```
module.exports = {sanityCheck,checkString}
```

# In the age of AI everything is an API.

**Abhinav Asthana**  
CEO - Postman





# Postbot is your AI Assistant

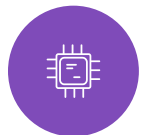
(and collaborator) for all things API testing and documentation

The screenshot shows the Postman web interface. At the top, there's a search bar and user avatars. The main area displays a REST client for the 'ExchangeRate-API' with a 'GET' request to 'Exchange Rate - USD'. The URL is {{baseUri}}/v6/{{api-key}}/latest/USD. Below the URL bar are tabs for Params, Authorization, Headers (5), Body, Pre-request Script, Tests, and Settings. The 'Tests' tab is active, showing a single test script. At the bottom, the 'Body' tab is selected, displaying the JSON response in 'Pretty' format. The response includes fields for 'result', 'documentation', 'terms\_of\_use', 'time\_last\_update\_unix', 'time\_last\_update\_utc', 'time\_next\_update\_unix', 'time\_next\_update\_utc', 'base\_code', and 'conversion\_rates'. A 'Postbot' AI assistant window is overlaid on the right side of the interface. It features the Postbot logo, a greeting 'Hello there!', and the question 'Need any help with this request?'. Below the text are four suggested actions: 'Add tests to this request', 'Test for response...', 'Visualize response...', and 'Save a field from response'. At the bottom of the window, it says 'Hi! How can I help?' with a play button icon.

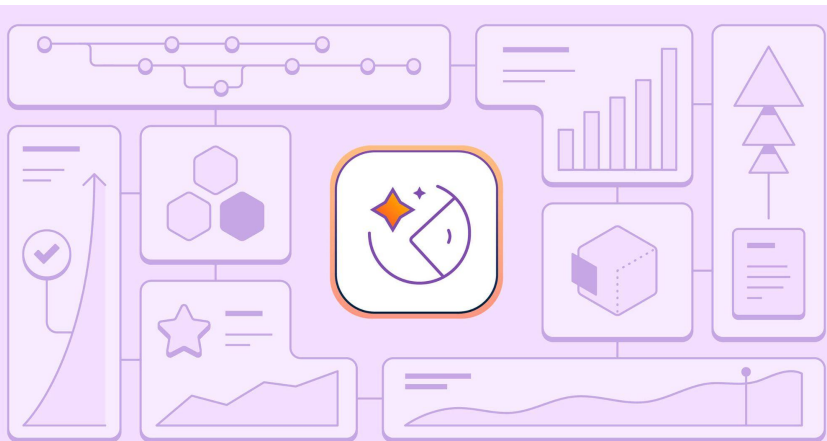
```
1 {
2   "result": "success",
3   "documentation": "https://www.exchangerate-api.com/docs",
4   "terms_of_use": "https://www.exchangerate-api.com/terms",
5   "time_last_update_unix": 1709078401,
6   "time_last_update_utc": "Wed, 28 Feb 2024 00:00:01 +0000",
7   "time_next_update_unix": 1709164801,
8   "time_next_update_utc": "Thu, 29 Feb 2024 00:00:01 +0000",
9   "base_code": "USD",
10  "conversion_rates": {
11    "USD": 1,
12    "AED": 3.6725,
13    "AFN": 73.3267,
```



# How can Postbot help you



**Postbot is your AI assistant  
for API workflows in Postman**



**Postbot can help you:**

- Write tests
- Visualize responses
- Get help using AI Expert Curated Support
- Write documentation
- Debug



# Benefits of Collaborative API Development

- **Accelerated development** and deployment
- **Enhanced onboarding** for new team members
- **Improved Quality** and reduced errors
- **Better communication** and change management
- **Scalability** for growing teams and organizations





2024 Survey



Thank you



@alexbarrett @getposman

# Prototyping

# Documenting

# Testing

# Demo-ing

The screenshot shows the Postman interface with a REST client request configured. The top navigation bar includes 'Home', 'Workspaces', and 'API Network'. The left sidebar shows a tree view of collections, with 'NOVA Security Tests' expanded to show 'admin/{admin id}' and 'users'. The main panel displays a 'POST' request to the endpoint `{{baseUrl}}/admin/adminid/user/new`. The 'Params' tab is active, showing 'Query Params' and 'Path Variables' tables. The 'Body' tab is selected, showing a JSON response in 'Pretty' format. The status bar indicates a '200 OK' response.

NOVA Security Tests / admin/{admin id} / Create network user / Create network user

POST `{{baseUrl}}/admin/adminid/user/new` Try

Params Headers Body

Query Params

Key	Value	Description	Bulk Edit
Key	Value	Description	

Path Variables

Key	Value	Description	Bulk Edit
adminid	Value	Description	

Body Headers (1) Status Code 200 OK

Pretty Raw Preview Text

```
1 {
2   "user": {
3     "name": "Dominique Destine",
4     "id": "{{userId}}",
5     "email": "dominique@vandelayindustries.com"
6   }
7 }
```

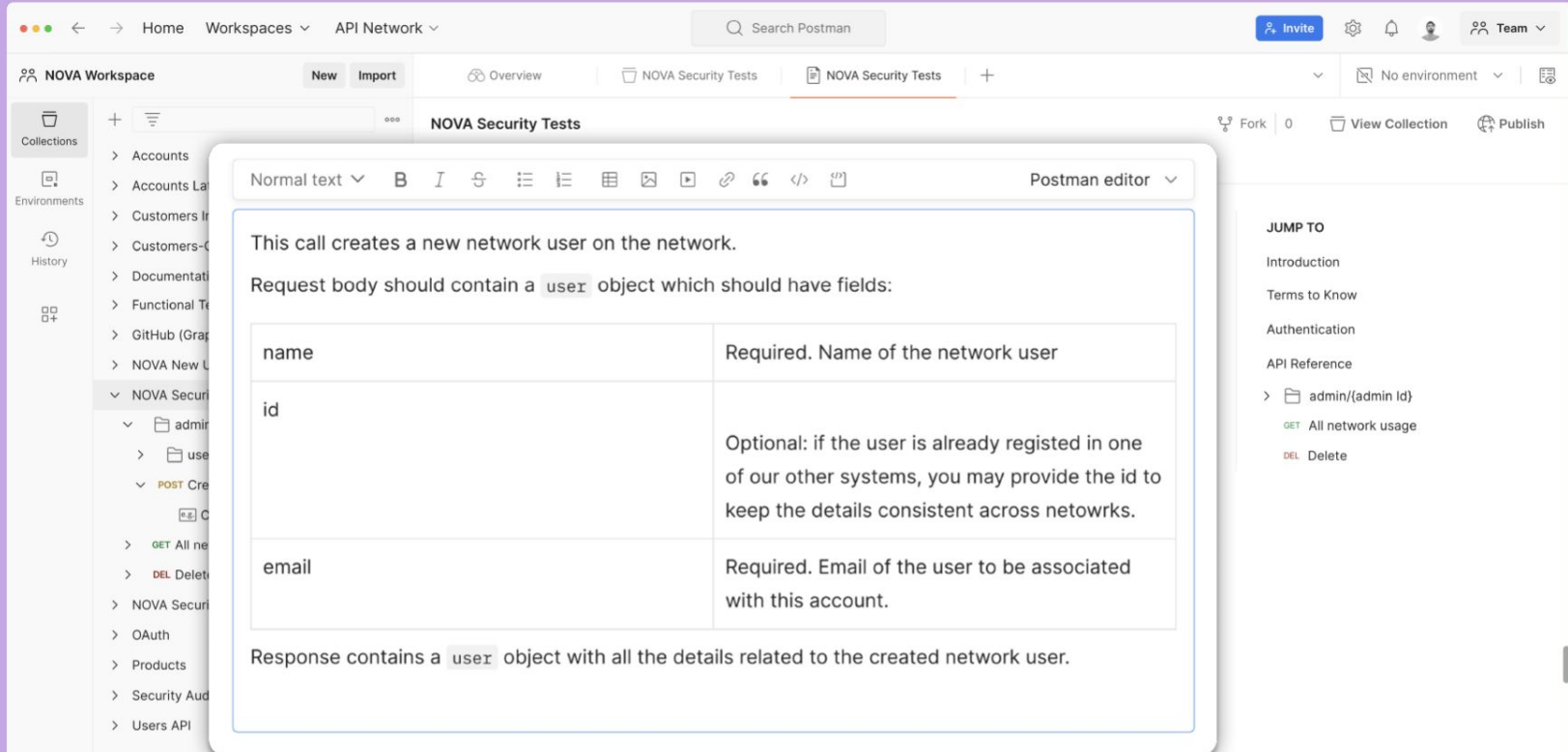


# Prototyping

# Documenting

# Testing

# Demo-ing



## Path Variables

# Prototyping

# Documenting

# Testing

# Demo-ing

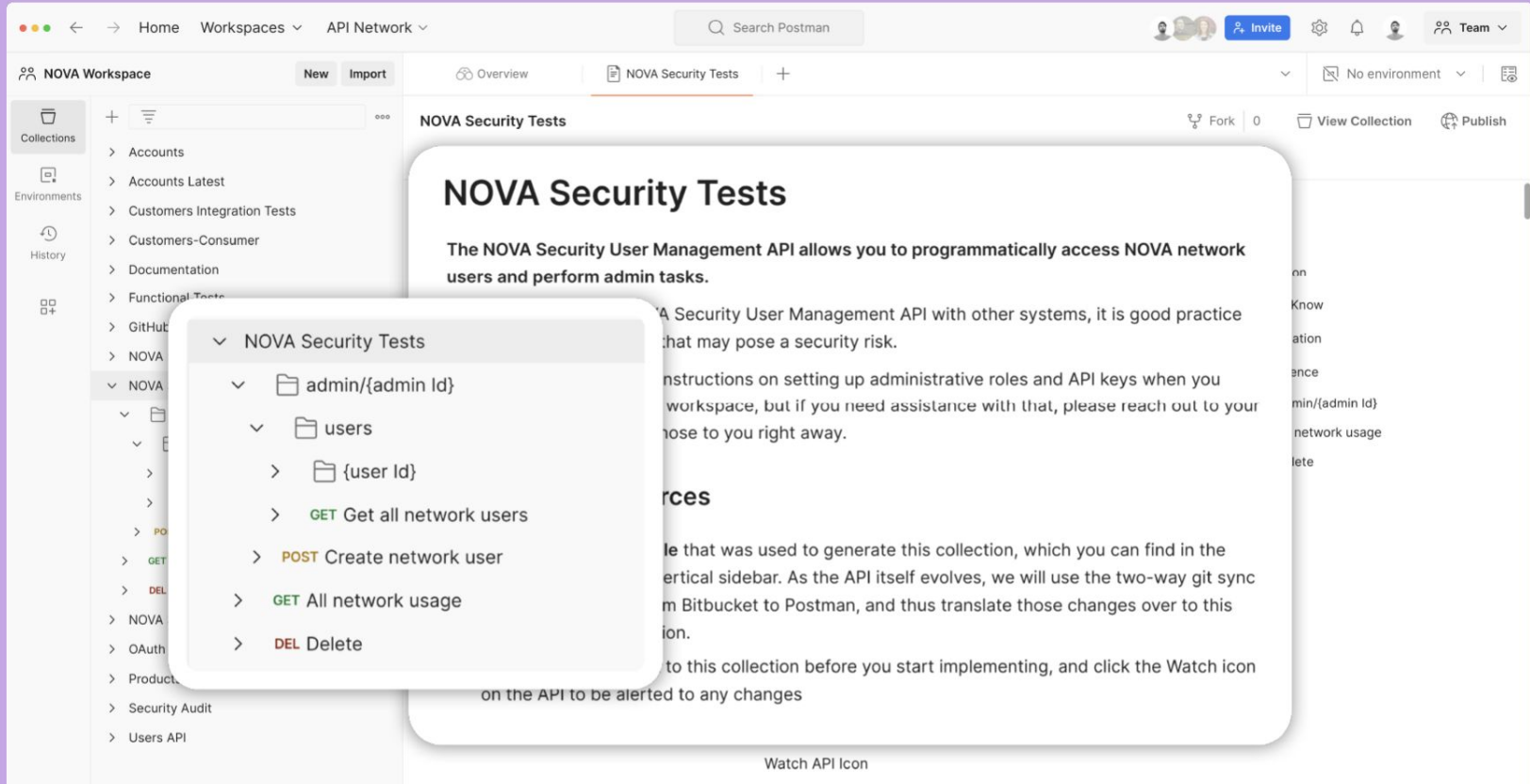
The screenshot shows the Postman interface with a test run summary overlay. The overlay displays the following information:

- NOVA Security User Management - Run results** (Run Again button)
- Automate Run ▾ + New Run Export Results
- Ran today at 13:19:26 · [View all runs](#)

Source	Environment	Iterations	Duration	All tests	Avg. Resp. Time
Runner	none	1	2s 331ms	16	305 ms

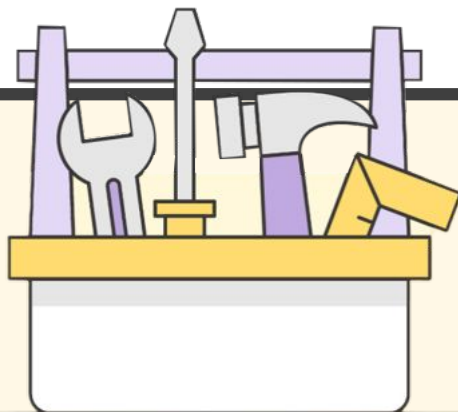
Iteration 1

  - DELETE Delete a network user**  
https://ba06aa48-0211-4a09-b4f3-8cd2573c9306.mock.pstmn.io/admin/adminid/users/userid 200 OK 713 ms 464 B
    - PASS Status code is 200
  - GET Get a network user**  
https://ba06aa48-0211-4a09-b4f3-8cd2573c9306.mock.pstmn.io/admin/elit cillum minim/users/elit cillum minim 200 OK 219 ms 743 B
    - PASS Status code is 200
    - PASS Status code is 200
    - PASS Response time is less than 2000ms
    - PASS Schema is not empty
  - PUT Update a network user**  
https://ba06aa48-0211-4a09-b4f3-8cd2573c9306.mock.pstmn.io/admin/9038759/users/3409860 200 OK 240 ms 677 B
    - PASS Status code is 200
    - PASS Status code is 200
    - PASS Response time is less than 2000ms



“The benefits of Postman are objectively clear. The collaboration tools, the testing tools, the mocking tools really speed up and standardize the entire API lifecycle.”

**Michael Carey,**  
Sr. Product Manager, API,  
Lightspeed





# 2023 State of the API Report



May hide this slide -

## This workflows chaos has real business implications

### **SLOWER DEVELOPMENT CYCLES**

Fragmented API systems cause communication and execution bottlenecks, delaying development

### **WASTED DEVELOPMENT EFFORT**

Ad hoc discovery and distribution practices lead to duplication of assets and poor APIs that don't get used

### **LOST OPPORTUNITIES**

Inability to share collections with partners and external developers for testing and integration means lost opportunities



# Collaborate Publicly and Privately

The screenshot displays the Postman workspace interface for "Acme <> Starlight". The interface is divided into several sections:

- Header:** Includes navigation links for Home, Workspaces, and API Network, a search bar for Postman, and user management options like "Invite" and "Team".
- Workspace Overview:** Shows the workspace name "Acme <> Starlight" and a status bar with "Unwatch" and "0" items.
- Left Sidebar:** Contains a "Collections" list with "SpaceCash Payments v2" (vStarlight) expanded to show "Auth", "Transactions", and "Accounts". It also includes "Environments" and "History" sections.
- Main Content Area:**
  - Welcome:** "Welcome to the Acme <> Starlight partner workspace!"
  - Getting started with this workspace:** A section with a rocket icon providing instructions on using API keys from the "Space Cash Partner Developer Portal" and selecting the "Partner Testing" environment.
  - Help and Support:** A section with a megaphone icon directing users to the "Partner Onboarding Team" and mentioning the ability to add comments.
  - Contacts:** A section with a thumbs-up icon listing "Ramji Enamuthu (ramji@spacecash.com)" and "Mary Jane (mary.jane@cosmiccabs.com)".
  - Pinned collections:** A section with a plus icon showing "SpaceCash Payments v2" and "[beta] Partne...".
- Right Sidebar:**
  - About:** "A single place for Acme and Starlight teams to collaborate on APIs"
  - Tags:** "partner-api"
  - Contributors:** "You" (with profile icon)
  - Partners:** "Dan Perkins" (with profile icon and email "ramji.enamuthu+beta43@postman.com") and a "View all partners" link.
- Footer:** Includes status indicators for "Online", "Find and replace", "Console", "Runner", "Start Proxy", "Cookies", "Vault", "Trash", and a refresh icon.