

# **Training Suite**

---

Modules, labs, and onboarding paths

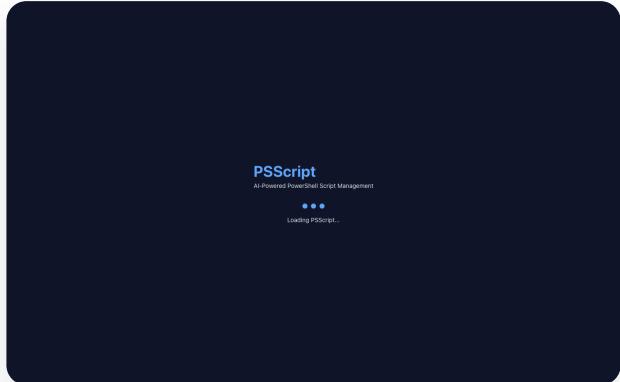
PSScript Manager Training

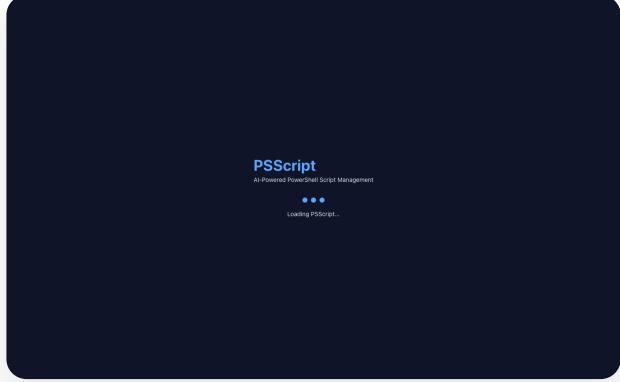
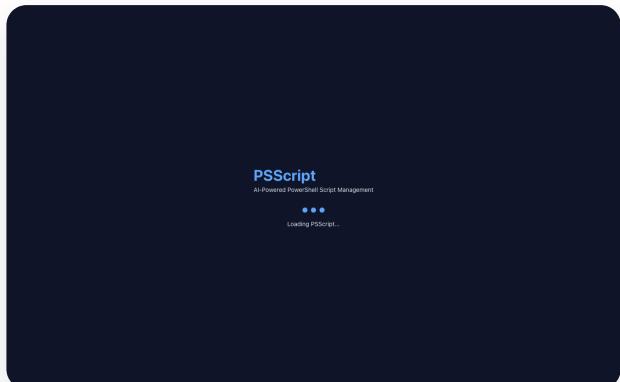
# PSScript Training Suite

---

AI-powered PowerShell operations training, packaged for onboarding, compliance, and day-2 operations. Built in the style of polished GitHub READMEs (à la Supabase/PostHog) but tailored to this platform.

# What you'll learn (at a glance)

Capability	What it teaches	Key screens/assets
Script intake	Uploads, tagging, versioning, hash-based dedup	
AI analysis	Security scoring, fix recommendations, audit notes	
Discovery	Keyword + vector search, documentation explorer, crawl	

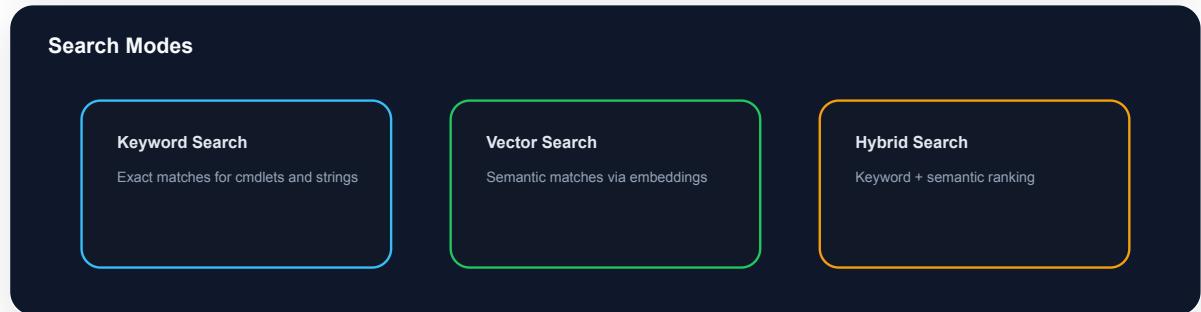
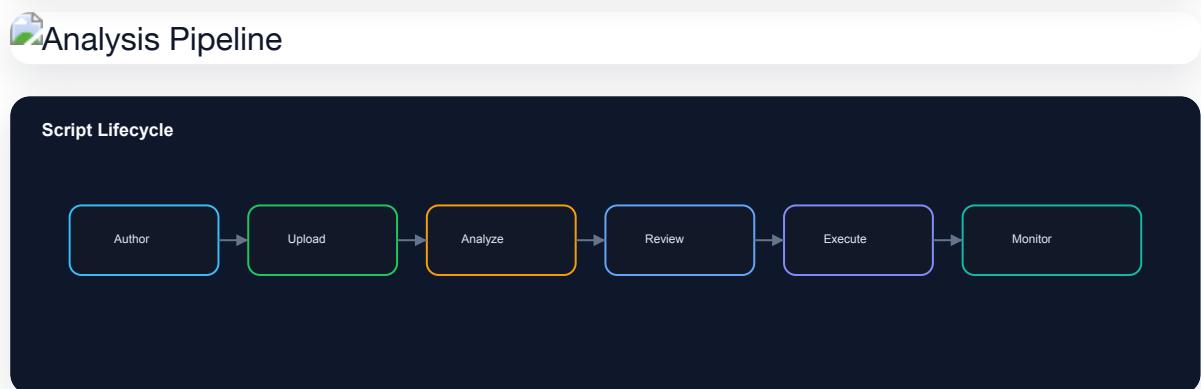
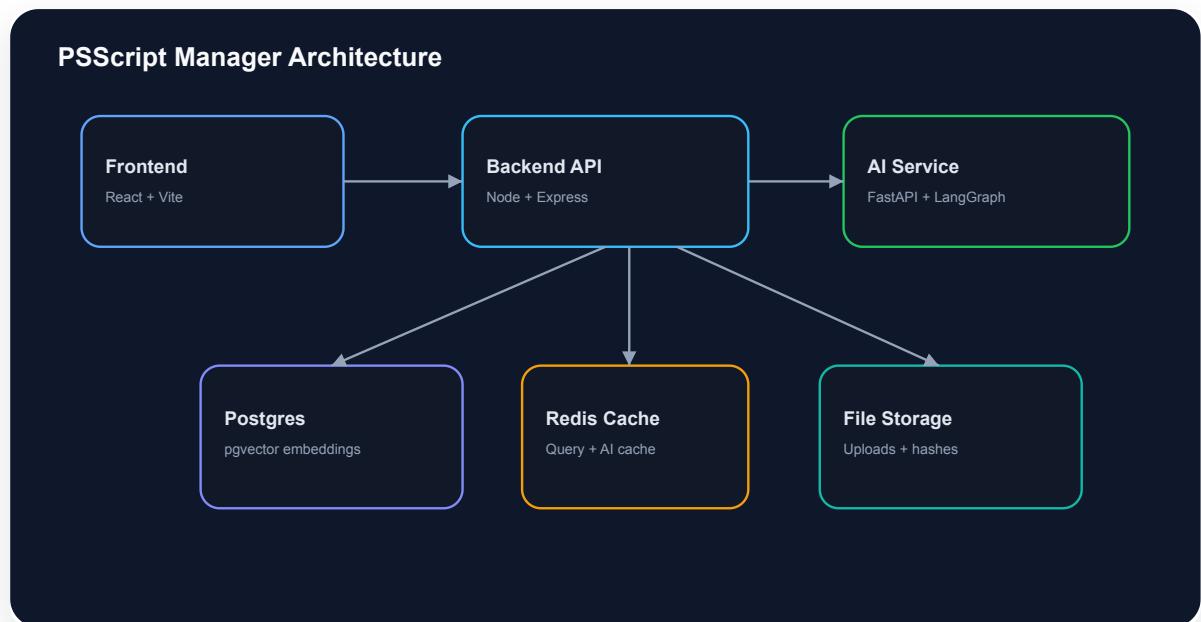
Capability	What it teaches	Key screens/assets
AI copilots	Chat assistant, agentic workflows, history	
Operations	Analytics, API usage, exports, training links	

## Feature tour (README-style checklist)

---

- [x] Script library with categories, owners, version history, and file-hash dedup
- [x] AI analysis pipeline (scorecards, recommendations, remediation notes)
- [x] Documentation explorer + crawler for command references and context storage
- [x] Chat/agent experiences for “how to fix” and “explain this script” prompts
- [x] Analytics dashboards for adoption, security posture, and training coverage
- [x] Export kit (HTML, PDF, DOCX) plus Playwright screenshot automation

# Architecture and flows

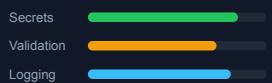


## Security Scorecard

### Overall score

**8.1** / 10

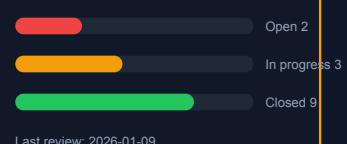
Low risk



### Top findings

- 2 credential hardcodes
- 4 missing parameter guards
- 3 verbose logging gaps
- 1 high-risk cmdlet chain
- 6 scripts need owner tags

### Remediation status

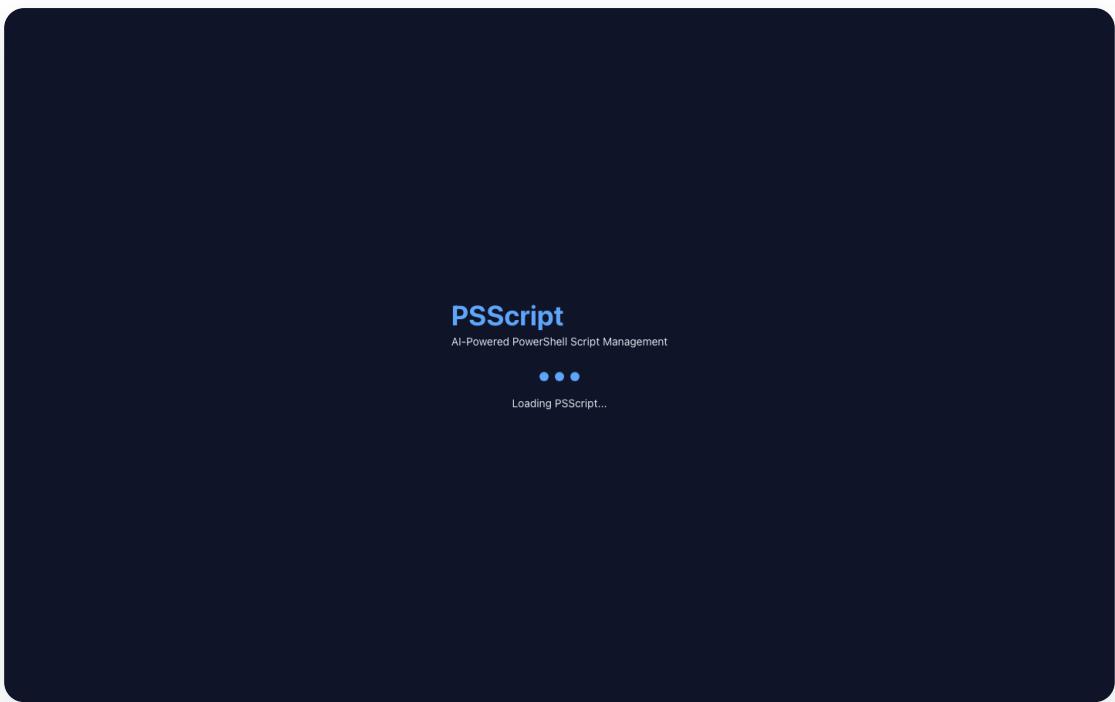


Last review: 2026-01-09

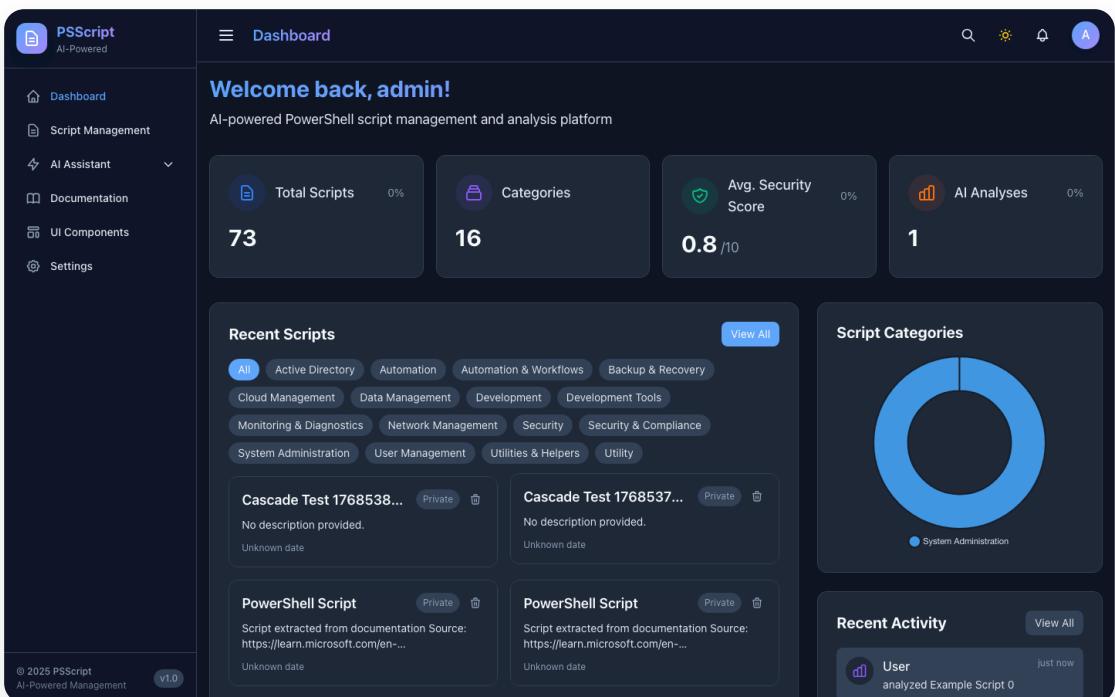
Next review: 2026-02-01

# Screenshot gallery

- Login + default access:



- Dashboard with stats and activity:



- Script detail with metadata:

PSscript  
AI-Powered

Dashboard Script Management AI Assistant Documentation UI Components Settings

**Upload Script**

Title: test-script

Description: What does this script do?

Category: -- Select Category --

Tags: Add custom tag Add

Make script public

Analyze with AI

Preview Analysis

Cancel Uploading...

Script Content Preview: 224 lines

```
# Enhanced PowerShell script for AI Analysis Testing
# Author: Claude
# Version: 2.0
# Date: 2025-03-11

# Import necessary modules
Import-Module -Name Microsoft.PowerShell.Security
Import-Module -Name Microsoft.PowerShell.Utility

function Get-SystemInfo {
    <#>
```

© 2025 PSscript  
AI-Powered Management v1.0

- Upload with live preview:

PSscript  
AI-Powered

Dashboard Script Management AI Assistant Documentation UI Components Settings

**Upload Script**

Title: test-script

Description: What does this script do?

Category: -- Select Category --

Tags: Add custom tag Add

Make script public

Analyze with AI

Preview Analysis

Cancel Upload Script

Script Content Preview: 224 lines

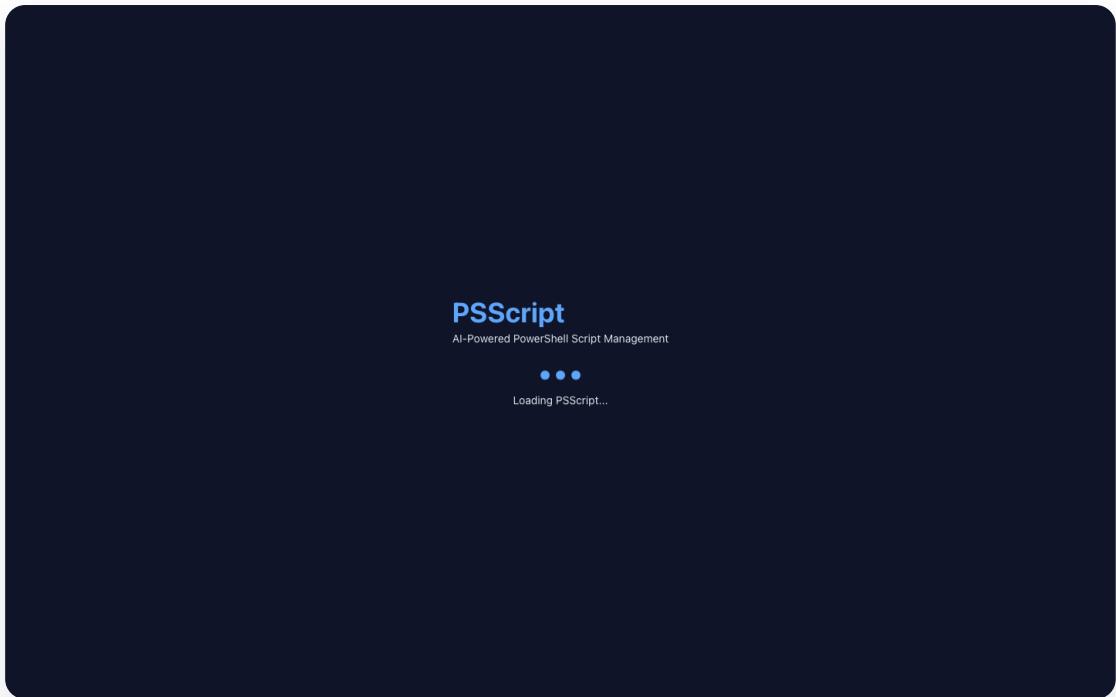
```
# Enhanced PowerShell script for AI Analysis Testing
# Author: Claude
# Version: 2.0
# Date: 2025-03-11

# Import necessary modules
Import-Module -Name Microsoft.PowerShell.Security
Import-Module -Name Microsoft.PowerShell.Utility

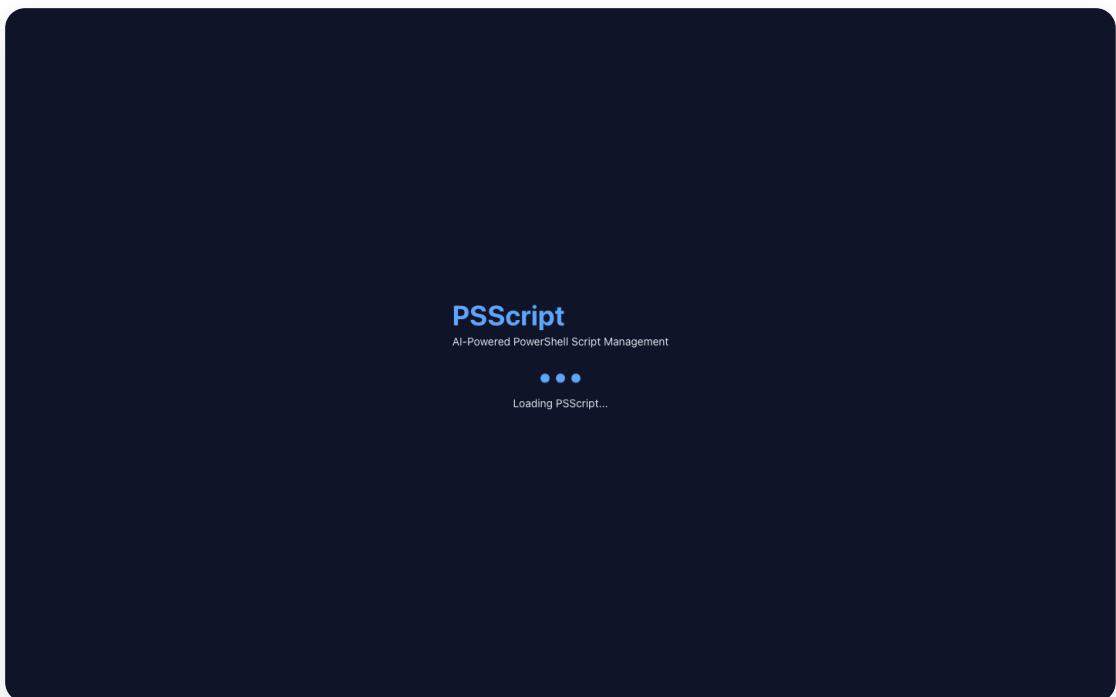
function Get-SystemInfo {
    <#>
```

© 2025 PSscript  
AI-Powered Management v1.0

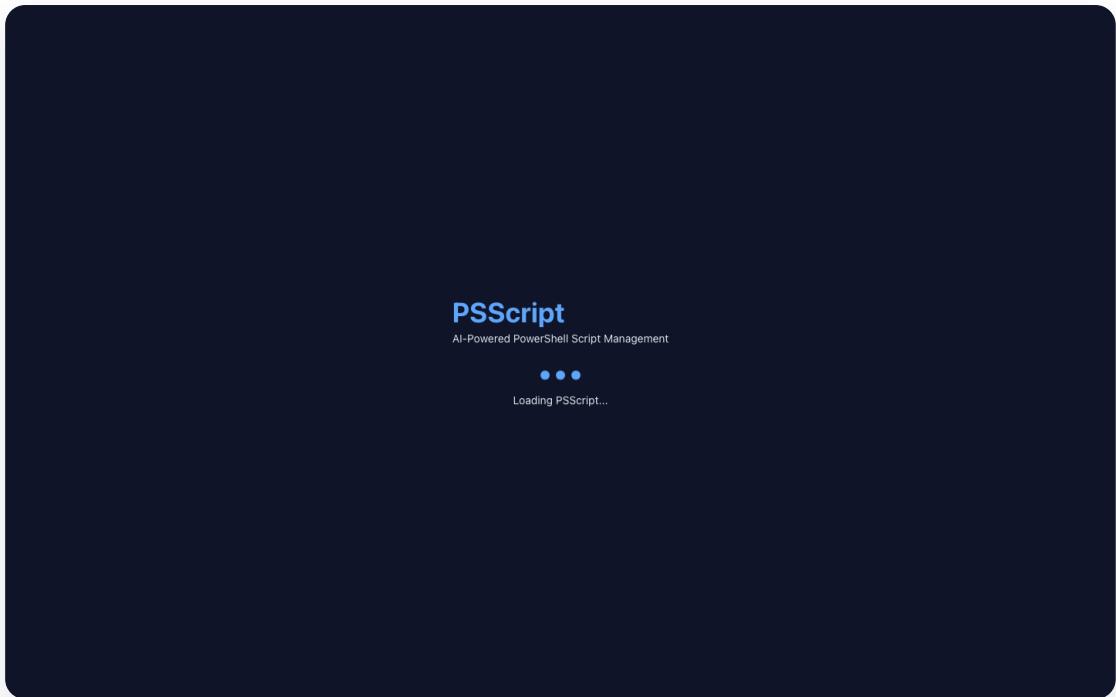
- AI analysis + recommendations:



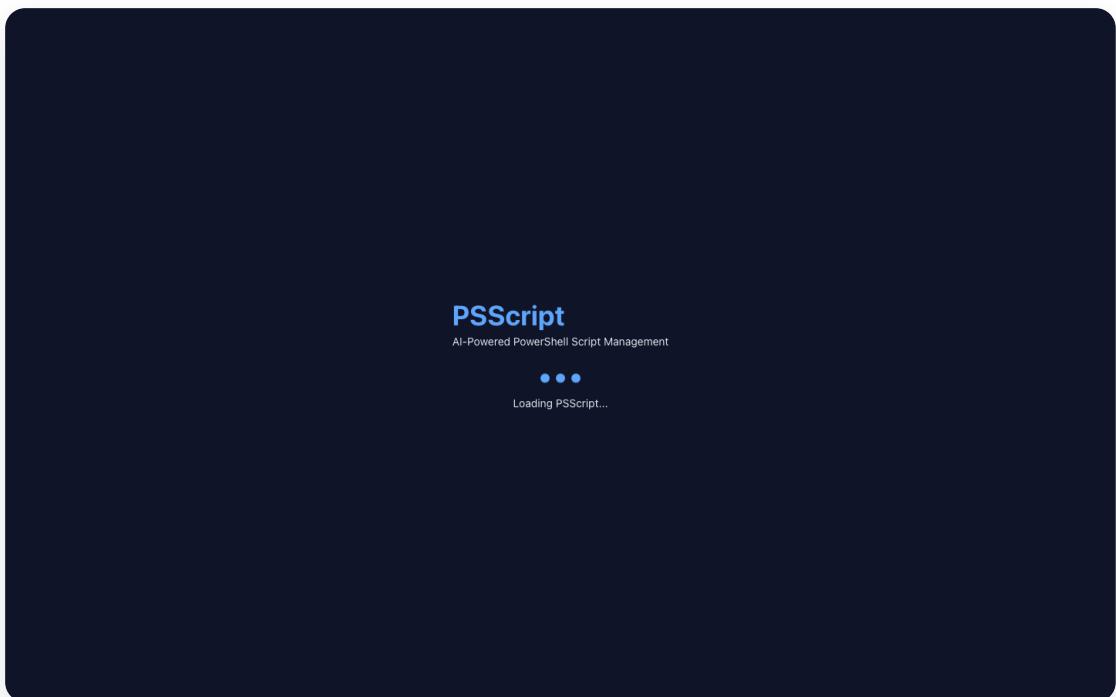
- Documentation explorer:



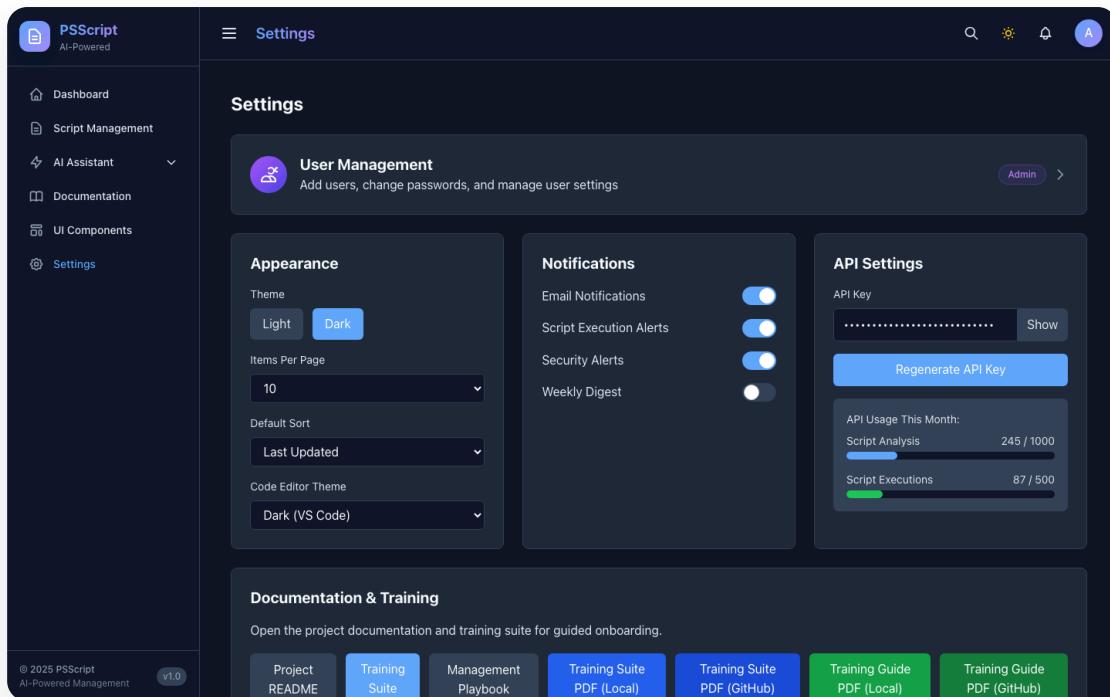
- Chat assistant:



- Analytics + ops KPIs:



- Settings with training links:



## Audience and outcomes

---

Audience	Goals	Outcome artifacts
Script authors	Upload, categorize, analyze, and improve scripts	Curated script library with analysis reports
Security reviewers	Evaluate risk, audit findings, track fixes	Security scorecards and remediation notes
Platform admins	Operate services, manage users, tune data	Environment runbook and health checklist

# Training roadmap



```
flowchart LR
Start[Orientation] --> Foundations[Module 01: Platform Foundations]
Foundations --> Lifecycle[Module 02: Script Lifecycle]
Lifecycle --> Analysis[Module 03: AI Analysis]
Analysis --> Search[Module 04: Search and Discovery]
Search --> Ops[Module 05: Operations]
Ops --> Labs[Hands-on Labs]
```

# Curriculum map

---

Module	Description	Duration	File
Module 01	Platform Foundations	45 min	modules/module-01-foundations.md
Module 02	Script Lifecycle	60 min	modules/module-02-lifecycle.md
Module 03	AI Analysis and Security	75 min	modules/module-03-analysis.md
Module 04	Search and Discovery	45 min	modules/module-04-search.md
Module 05	Operations and Governance	60 min	modules/module-05-operations.md

# Labs

---

Lab	Goal	File
Lab 01	Upload and analyze a script	<code>labs/lab-01-upload-analyze.md</code>
Lab 02	Vector search and similarity	<code>labs/lab-02-vector-search.md</code>
Lab 03	Documentation and AI chat workflows	<code>labs/lab-03-docs-chat.md</code>
Lab 04	Analytics and governance checks	<code>labs/lab-04-analytics.md</code>
Lab 05	Settings and training resources	<code>TRAINING-GUIDE.md</code>

## Suggested schedules

---

Option	Duration	Focus
Fast start	1 day	Core workflows and 2 labs
Standard	2 days	All modules and all labs
Deep dive	3 days	Modules, labs, and ops review

## Environment setup (mock-friendly)

---

1) Start everything with sample data:

```
./start-all-mock.sh
```

2) Ports: - Frontend: <http://localhost:3002> - Backend API: <http://localhost:4000/api>  
- AI service: <http://localhost:8000>

3) Capture screenshots (Playwright):

```
./scripts/capture-readme-screenshots.sh
```

Screens land in `docs/screenshots/`.

## Exported formats (print kit)

---

Generate HTML, PDF, and DOCX:

```
scripts/export-docs.sh --all
```

Key outputs (after export): - docs/exports/pdf/Training-Suite.pdf -  
docs/exports/pdf/Training-Guide.pdf - docs/exports/docx/Training-  
Suite.docx - docs/exports/docx/Training-Guide.docx -  
docs/exports/pdf/README.pdf (project README)

## Assessment rubric

---

Skill area	Basic	Proficient	Advanced
Script hygiene	Upload and tag scripts	Adds categories and metadata	Enforces naming and dedup rules
AI analysis	Runs analysis	Applies recommendations	Creates reusable fixes and notes
Search	Keyword search	Uses filters	Uses vector similarity effectively
Operations	Starts services	Reads logs	Diagnoses and remediates issues

# Support and troubleshooting

---

- Demo access: `../LOGIN-CREDENTIALS.md`
- Support + operational playbooks: `../SUPPORT.md`
- Infrastructure setup: `.../../DOCKER-QUICKSTART.md`
- References/inspirations: `../REFERENCE-SOURCES.md` (GitHub README patterns)

Generated 2026-01-16 23:34 UTC