

# From Chaos to Clarity

How MCP Transforms Incident  
Response

PagerDuty



**Sebastián Villanelo**, Forward  
Deployed Engineer



**Rocío Bayon**, PM Forward  
Deployed Engineering



THIS IS FINE.



# The Universal Problem

*Context switching destroys effectiveness in high-stakes moments*



**+300%**

Error Rate



**-40%**

Focus Recovery



**-40%**

Productivity

“

**Bring operations into the conversation,  
not pull people out of their flow.**

# MCP Apps vs Basic Integration

## Tool Calls

- Text-only responses
- Raw JSON output
- No visual context
- Generic AI UX

VS

## Full UI Components

- ✓ Familiar PagerDuty UX
- ✓ Rich incident UI in IDE
- ✓ SRE Agent integration
- ✓ Audit trail on every action

# Architecture

## MCP server

Data + tools



## App protocol

Structured UI data



## UI Rendering

Rich components



## IDE Surfaces

VS Code, Claude  
Desktop, etc.

### 01

#### Reduce Cognitive Load

Not just fewer clicks, fewer mental shifts between tools and contexts.

- 
- Stay in flow state
  - Preserve mental model

### 02

#### Safety First

Confirm writes, auto-execute reads. Every AI action logged in the PagerDuty timeline.

- 
- Audit trail on every action
  - No silent writes

### 03

#### Human in the Loop

AI orchestrates, humans decide on critical actions. You stay in control when it counts.

- 
- Confirm before resolving
  - AI amplifies, not replaces

WATCH FOR THIS IN THE DEMO

# Three Things To Look For

01

## Still in VS Code

Not a single tab opens. The entire incident is handled without leaving the IDE.

→ **No context switch. Zero.**

02

## Safety Confirmation

Before any write operation, acknowledge, resolve, a prompt appears. You stay in control.

→ **Human in the loop.**

03

## SRE Agent joins

AI triage and a remediation plan surface directly inside the incident UI

→ **Agentic operations, live.**

# The Demo



*“I’m a developer. Shipping a feature. Deadline in 2 hours. And then...”*

# The Vision



TODAY



MCP APPS



FUTURE

## Human drive, AI assists

Tool calls return text. You open 5 tabs. Context switch kills your flow.

- Manual tool switching
- Raw JSON responses

## AI orchestrates, human supervises

Full PagerDuty UI inside VS Code. Rich incident context. SRE Agent integrated.

- Zero context switch
- Human confirms critical actions

## Multi-agent workflows

SRE Agent coordinates with security, deploy, and custom agents via MCP.

- SRE Agent ↔ external agents
- Build your own on our MCP

↓ **Burnout**

**Responder burnout**

No more juggling 5 tools at 3 AM. Context is already there.

↑ **Speed**

**Resolution speed**

AI triage in seconds. Human confirms. Incident resolved.

→ **Focus**

**Developer focus**

Handle the incident. Ship the feature. Mental model intact.

**This is what solving context switching for real looks like.**



PagerDuty MCP · [find us after the talk or scan the QR code](#)