MCP SERVER | NO CODE | CLOUD HOSTED

Dixa MCP for AI Agents

Manage customer support queues and agent availability metrics

Dixa MCP gives your AI agent full visibility into customer support operations. It lets you monitor live conversations across multiple channels, check real-time agent availability, and track which service queues need attention. You get a single pane of glass for managing entire support teams and understanding where customer issues are getting stuck.

A+ Quality Score 100/100

omnichannel-support | customer-experience | agent-performance | service-queues | conversation-tracking | helpdesk

MAINTAINED BY VINKIUS CLOUD FOR AI AGENTS · JUNE 2026
vinkius.com/mcp/dixa

SCAN TO VIEW

The connectivity layer between AI and the world's software.

Vinkius sits between AI and every application. All communication passes through Vinkius Cloud via the Model Context Protocol (MCP) — with governance, observability, and security at every layer.

Your AI Connections Run Through Vinkius Cloud

The world's largest managed MCP catalog

Vinkius is the connectivity layer where AI connects to the software your business already runs. We handle the hosting, the security, the credentials, the uptime — you get agents that actually do things.

We operate the world's largest managed MCP catalog. Major SaaS platforms, CRMs, databases, and cloud providers — running, monitored, production-ready. This MCP server is hosted and maintained by the Vinkius Cloud for AI Agents.

> *The agent doesn't manage credentials, doesn't manage uptime, doesn't manage security. Vinkius does.*
>
> — Architecture principle

Four Pillars of the Vinkius Runtime

01 — Security by design

Credentials stay encrypted at rest via AES-256. The AI agent never touches raw keys — they're injected into a sandboxed V8 isolate at runtime. Actions are logged, and connections have an emergency kill switch.

02 — Built on MCP Fusion

This MCP server was built with **MCP Fusion**, the open-source framework (Apache 2.0) that powers the entire Vinkius catalog. Schema-as-firewall strips undeclared fields, compiled PII redaction runs at zero overhead, and cryptographic lockfiles produce git-diffable audit trails.

03 — Deterministic observability

Eight immutable metrics per endpoint: request volume, p95 latency, error rate, active connections, cost attribution. A live payload feed logs every tool call with mutation detection.

04 — Autonomous operations

Servers are deployed, monitored, and patched autonomously. New capabilities and security patches ship weekly. Zero-downtime deployments ensure continuous availability across all managed MCP servers.

AES-256	Ed25519	24h TTL	V8 Isolate
Encryption at rest	PKI vault signatures	Ephemeral session keys	Sandboxed execution

One Token. Instant Access.

Every MCP server on Vinkius is accessed through a **Connection Token**. Tokens are generated in the cloud dashboard and produce a unique MCP endpoint URL. Paste this URL into any MCP-compatible client — no SDK required.

A single token can serve **multiple AI clients simultaneously**, or you can issue separate tokens per client for granular access control. Each token tracks its own request count, last activity timestamp, and can be individually enabled or revoked.

MCP ENDPOINT

```
https://edge.vinkius.com/{token}/mcp
```

Claude · Cursor · VS Code · Windsurf · Grok · Gemini

Security Is the Architecture

Security in Vinkius is not a feature — it's the foundation of the runtime. The gateway enforces multiple independent protection layers between AI agents and third-party APIs.

01 — Ed25519 PKI Vault

Every workspace has an Ed25519 Master Key. Session keys are generated ephemerally (24h TTL) and signed by the Master Key. Credentials never leave the vault boundary.

02 — V8 Isolate Sandboxing

Tool code runs inside isolated-vm V8 isolates with 64 MB memory caps and per-request timeouts. No filesystem access, no network access except through the SSRF-guarded fetch bridge.

03 — SSRF Guard

All outbound HTTP requests are DNS-resolved and validated before execution. Private IP ranges (10.x, 172.16-31.x, 192.168.x, AWS metadata 169.254.x) are blocked at the network layer.

04 — DLP & PII Redaction

A ResponseGuard pipeline intercepts every tool response. Configurable redaction patterns strip sensitive fields (emails, SSNs, card numbers) before data reaches the AI agent.

05 — Cryptographic Audit Trail

Every request is signed into a SHA-256 hash chain with Ed25519 signatures. Events form a tamper-proof, SIEM-exportable forensic record.

06 — Honeytoken Trap System

Phantom credentials are injected into isolated environments. If a honeytoken is used outside Vinkius infrastructure, the server is quarantined instantly.

Emergency Kill Switch

EU AI Act Art. 14(1) Compliant

The kill switch is an **emergency halt** mechanism — not a simple toggle. When triggered, it executes three actions atomically:

01 — Server deactivated

The MCP server is immediately taken offline across the entire cluster.

02 — All tokens revoked

Every connection token is invalidated. Total lockout — reconnection blocked until new tokens are issued.

03 — WebSocket connections killed

Active connections terminated via Redis pubsub broadcast. Propagates to every runtime node in the cluster.

Full Visibility. Zero Guesswork.

The Vinkius cloud dashboard includes a full MCP Governance suite — real-time analytics and security controls for production AI operations.

Control Plane

KPI dashboard with request volume, latency, success rate, token consumption, and AI-generated operational briefings.

Agent Activity

Which AI clients are connecting, how often, and what they're doing — real-time session tracking.

FinOps

Cost tracking per tool, payload compression savings, budget optimization signals, and consumption trends.

Tool Health

Slowest and most error-prone tools, with actionable root-cause insights and performance baselines.

Firewall & DLP

PII redaction activity, sensitive data protection counters, and security event timeline.

Incident Log

Error trends, failure rates, status-code breakdowns, and forensic audit trail access.

> Get started at **cloud.vinkius.com** — connect your AI agent in under 60 seconds.

Dixa MCP

10 tools available
Cloud-hosted on Vinkius

You can connect the Dixa platform directly to your AI workflow using this MCP. Instead of logging into separate dashboards to check on staff or tickets, your agent handles it all in one conversation. It lets you pull detailed data on every customer interaction—from initial contact through resolution—and know exactly who is working on what and how fast. You can monitor the health of your entire support operation: checking queue volumes, verifying which agents are available right now, and seeing team assignments across the whole company.

This capability lets you move beyond simple ticketing. It gives your agent operational insight into your customer experience. Through Vinkius, this MCP acts as a central hub, making complex tasks like auditing agent performance or listing every open ticket something that happens instantly in chat, not over three different tabs.

Core Capabilities

01 — Monitor live conversations

Retrieve full details and current statuses for specific customer service chats.

02 — Check agent availability

Get a high-level summary of which support agents are currently online or away.

03 — Identify open tickets

Find all conversations that are stuck in an 'Open' or unassigned status, signaling immediate action is needed.

04 — Audit support queues and teams

List all configured service queues and map out the full structure of your internal support teams and their members.

05 — Review agent performance data

Fetch comprehensive profile details and performance metrics for any specific team member.

One Click on Vinkius — From Prompt to Execution

Available at **vinkius.com/mcp/dixa** — connect your AI agent in three steps.

01 First, connect the Dixa MCP to your AI client and authorize it using your API token.

02 Next, tell your agent exactly what you need—for example, 'What's going on with our open support queues?'

03 Your agent executes the necessary calls, gathering real-time data on tickets, agents, or teams and presenting a concise summary back to you.

The bottom line is your AI client treats Dixa operational data like native knowledge, letting you query complex support metrics using simple conversation prompts.

Built For

This MCP is for Support Managers and Operations Leads who are tired of juggling multiple dashboards to understand the true state of their customer service team. If your job requires knowing agent availability, ticket backlog, or team structure instantly, this connector saves you time.

Support Manager

You use it to quickly check queue volumes and determine if enough agents are available across different service teams.

Customer Success Lead

You rely on it to research conversation history and audit agent performance metrics without leaving the chat window.

Operations Analyst

You use it to monitor support team structures, list all configured queues, and verify organizational metadata instantly.

What Changes When You Connect

01 Instantly assess team capacity. Instead of manually checking dashboards, use the `quick_agent_presence_audit` tool to get a real-time status of every available agent.

02 Reduce response time by finding stuck tickets. The ability to call `list_open_support_tickets` lets your agent pinpoint exactly which conversations need immediate human attention.

03 Understand team structure instantly. Use `list_support_teams` and `list_service_agents` together to map out who belongs where, making shift management simple.

04 Deep conversation analysis. By calling `get_conversation_details`, you get the full story on any customer chat without needing to navigate through old email threads or ticket views.

05 Operational oversight. You can check all active service queues using `list_service_queues` and understand exactly how your support workload is routed.

Real-World Applications

A critical outage requires immediate staffing

The Ops Lead asks their agent: 'Who's available in the billing queue right now?' The agent runs `quick_agent_presence_audit` and reports back that three agents are available, allowing the lead to immediately redirect work. This prevents service slowdowns when you need it most.

Auditing a recent support issue

A Support Manager needs background on customer 'X'. They prompt their agent with: 'Show me everything about conversation ID 9876.' The agent uses `get_conversation_details` and summarizes the entire chat history, identifying key moments that led to the current problem.

Restructuring a support department

The Director needs to verify team membership. They prompt: 'List all teams and who is assigned.' The agent executes `list_support_teams` and provides an accurate breakdown of every member in the organization, saving hours of manual spreadsheet cross-referencing.

Checking for neglected tickets

At the end of the day, a team lead asks: 'What conversations haven't been touched today?' The agent runs `list_open_support_tickets` and gives a list of high-priority cases that are currently unassigned or stalled.

Patterns to Avoid

Treating the MCP like a generic database lookup

✗ AVOID

Just asking: 'Give me all conversation data.' This results in a massive, unusable dump of raw text with no context or summary.

✓ INSTEAD

Instead, ask for something specific and actionable. Try: 'List the top 5 open support tickets that are unassigned,' which uses `list_open_support_tickets` to give you immediate action items.

Ignoring agent performance context

✗ AVOID

Running a list of all agents (`list_service_agents`) and then getting nothing useful. You just get names, but no idea if they are actually working.

✓ INSTEAD

Always follow up the list with an action-oriented query, like: 'Check the profile for Agent Jane Doe.' This uses `get_agent_profile` to give you actionable performance metrics.

Focusing only on conversations

✗ AVOID

Only running a search by subject keyword (`search_conversations_by_subject`) when the issue is actually about team structure.

✓ INSTEAD

Remember that support operations are about people and processes too. Run `list_support_teams` to get the organizational context needed before you diagnose conversations.

The Right Fit

Use this MCP if your daily work requires knowing the operational status of a helpdesk or customer service team, whether that means monitoring live queue volumes, auditing agent load, or tracking ticket handoffs. If your goal is purely data archiving—just needing to pull historical records without understanding current state—you might only need basic database access. However, if you are building an AI workflow that needs to *act* on the status of a support ticket (e.g., reassign it, notify a manager), this MCP's ability to read real-time queue and agent data is critical. Don't use it just because it handles conversations; use it because it gives you operational intelligence.

Dixa MCP: Managing Customer Support Queues with AI

Today, managing support operations means clicking through a dozen different dashboards. You check the queue widget to see volume, then switch tabs to verify agent status, and finally jump into a separate reporting tool just to find out who owns a specific ticket. It's slow, it’s prone to human error, and you spend half your time aggregating data instead of solving problems.

With Dixa MCP, all that operational intelligence is available in one conversation. You simply ask your agent, 'What support queues are backed up today?' and get an immediate, actionable summary of the workload distribution—no clicking required.

Dixa MCP: Tracking Agent Performance Using AI

Tracking agent performance is historically a manual nightmare. Managers have to pull reports on every individual, cross-reference their ticket volume against their response time metrics, and then manually flag underperformers or top stars for meetings.

Now, you can ask your agent about an agent's profile directly. You get instant access to specific performance data points that were previously locked away in complex reporting tools. It makes coaching and resource allocation immediate.

Dixa: 10 Tools for Customer Support Metrics

Use these tools to gather all operational metrics related to your customer service team, including conversation history, agent profiles, and queue statuses.

#	TOOL	DESCRIPTION
01	`get_service_account_metadata`	Retrieves usage limits and general metadata for your Dixa account.
02	`get_agent_profile`	Gets detailed performance data and profile information for a single support agent.
03	`get_conversation_details`	Retrieves comprehensive details about one specific customer conversation, including history and status.
04	`quick_agent_presence_audit`	Provides a quick, high-level snapshot of the current availability status for all agents.
05	`list_service_agents`	Generates a list of every support agent registered within your Dixa organization.
06	`list_customer_conversations`	Lists all customer service conversations recorded in your account over time.
07	`list_open_support_tickets`	Identifies and lists specific conversations that are currently marked as 'Open' or 'Unassigned'.
08	`list_service_queues`	Lists all the active support queues set up within your Dixa system.
09	`list_support_teams`	Provides a list of every configured support team and which members belong to it.
10	`search_conversations_by_subject`	Searches for relevant conversations by using keywords found in the subject line.

See It in Action

Real prompts you can use once this MCP is connected to your AI agent through Vinkius Cloud.

U I need a full list of all agents in the billing team, and I want to know who is available right now.

🔍 Agent Status Report: Billing Team

The following 4 members are currently 'Available' for chats:
- Alice Johnson (ID: AJ-901)
- Marcus Chen (ID: MC-332)
- Sarah Lee (ID: SL-550)
- Raj Patel (ID: RP-119)

Team Breakdown:
- Total Team Members: 12
- Available: 4
- Away/Offline: 8

You can use this data to route new high-priority tickets instantly.

U Show me the details on conversation '9003'. It's about a payment failure.

💬 Conversation Details: ID 9003

- **Subject:** Payment Failure - Card Declined
- **Status:** Open / Needs Review
- **Assigned Agent:** Jane Doe (ID: JD-901)
- **Creation Date:** 2 hours ago via Web Chat
- **Last Update:** 5 minutes ago.

The chat history shows three attempts to resolve the issue. The last message was from the customer confirming they would try a different card tomorrow.

U What are all the active support queues and which ones have tickets waiting?

Service Queue Audit

QUEUE NAME	PURPOSE	OPEN TICKETS	STATUS
Billing Inquiries	Card issues, charges	14	**High**
Feature Request	Suggestions/Feedback	3	Low
General Support	Initial contact triage	29	Critical

Please prioritize the 'General Support' queue. It has the highest volume.

Frequently Asked Questions

01 **How does Dixa MCP help me manage my support team's daily workload?**

Dixa MCP gives you a centralized view of your entire operation. You can check agent availability and monitor queue backlogs in real time, ensuring that work is always routed to an available person quickly.

02 **Can I use Dixa MCP to find conversations from last month?**

Yes. The MCP allows you to list all customer conversations across various dates and channels. You can even search for specific issues by keywords in the subject line, making old data easy to retrieve.

03 **Is Dixa MCP good for finding tickets that are stuck or forgotten?**

Absolutely. It lets you run a dedicated check for open or unassigned support tickets. This is perfect for proactively flagging cases that have been sitting in the queue too long.

04 **How do I use Dixa MCP to see my agents' performance?**

You can pull detailed profiles and metrics for specific agents. This helps you track everything from their response time to overall ticket volume, making team coaching much more data-driven.

05 **What kind of reporting can I get with Dixa MCP?**

The system handles complex operational reporting by listing all configured support teams and the specific members assigned to them. It gives you a complete structural map of your entire department.

Go Live in 60 Seconds

Get your connection token from **cloud.vinkius.com**, then paste the endpoint URL into any MCP-compatible client.

YOUR MCP ENDPOINT

```
https://edge.vinkius.com/[TOKEN]/mcp
```

CLIENT	WHERE TO CONFIGURE
Claude AI	Profile → Customize → Connectors → "+" → Add custom connector → Paste endpoint
Cursor	Settings → Features → MCP Servers → "+ Add New MCP Server" → Type: SSE → Paste endpoint
VS Code	Ctrl/Cmd+Shift+P → "MCP: Add Server" → add `"dixa": { "url": "…" }`
Windsurf	MCP Settings → `mcp_settings.json` → Add endpoint URL
ChatGPT	Settings → Tools & plugins → Add MCP server → Paste endpoint
Gemini	Extensions → Add MCP Server → Paste endpoint URL

ASK AN AI ABOUT THIS

Let your preferred AI explain this MCP server

- Ask ChatGPT
- Ask Claude
- Ask Perplexity
- Ask Gemini
- Ask Grok

READY TO CONNECT

Dixa is live on Vinkius Cloud.

Get your connection token, paste it into your AI agent, and start building. No SDK. No deployment. Just results.

Start at cloud.vinkius.com →

vinkius.com · support@vinkius.com

INDEPENDENT PLATFORM DISCLAIMER

Vinkius is an independent platform and is not affiliated with, endorsed by, sponsored by, verified by, or otherwise authorized by Dixa. All third-party trademarks, logos, and brand names are the property of their respective owners. Their use in this document is strictly for informational purposes to identify service compatibility and interoperability.

DOCUMENT INFORMATION

Generated	June 2026
MCP Server	Dixa MCP
Server ID	`019d7586-f2d3-7205-9871-4cacf7935dee`
Platform	Vinkius Cloud for AI Agents
Endpoint	`https://edge.vinkius.com/{token}/mcp`

LICENSE & USAGE

This document is generated automatically by the Vinkius PDF Engine. Content reflects the MCP server configuration at the time of generation and may change as updates are deployed. For the most current information, visit vinkius.com/mcp/dixa.