

BRUNELLA AGENT SYSTEM (BAS) - Fejlesztési Állapotjelentés

Dátum: 2026. február 9.  
Verzió: v2.4+ (Production Ready Alpha)  
Git Branch: main-----1. Tesztelési és Build Állapot

Metrika	Eredmény
Build (TypeScript)	PASS
Tesztek	210/211 PASS
Tesztfájlok	29/30 PASS
Git Branches	5 db (main + 4 feature)

**Megjegyzés:** Az egyetlen hibás teszt a `test/rag.test.ts` fájlban található, a LanceDB vektor keresés üres eredményt ad vissza. Ez valószínűleg környezeti (environment-specifikus) probléma (pl. embedding modell hiánya vagy LanceDB verzió).2. Kész Funkciók (Production Ready)A) Core Backend (Node.js/TypeScript)

Komponens	Fájl(ok)	Állapot
Orchestrator Agent	<code>src/agents/OrchestratorAgent.ts</code>	KÉSZ ▾
Developer Agent	<code>src/agents/DeveloperAgent.ts</code>	KÉSZ ▾
Evaluator Agent	<code>src/agents/EvaluatorAgent.ts</code>	KÉSZ ▾
Researcher Agent	<code>src/agents/ResearcherAgent.ts</code>	KÉSZ ▾
DataScientist Agent	<code>src/agents/DataScientistAgent.ts</code>	KÉSZ ▾
EdgeProxy Agent	<code>src/agents/EdgeProxyAgent.ts</code>	KÉSZ ▾
ProjectConductor Agent	<code>src/agents/ProjectConductorAgent.ts</code>	KÉSZ ▾
Agent Architect	<code>src/agents/AgentArchitect.ts</code>	KÉSZ ▾
Robotkez Agent	<code>src/agents/RobotkezAgent.ts</code>	KÉSZ ▾

LintFixer Agent	<code>src/agents/LintFixerAgent.ts</code>	KÉSZ ▾
SpecWriter Agent	<code>src/agents/SpecWriterAgent.ts</code>	KÉSZ ▾
Voice Agent	<code>src/agents/VoiceAgent.ts</code>	KÉSZ ▾
Dynamic Agent Loader	<code>src/agents/DynamicAgentLoader.ts</code>	KÉSZ ▾
Agent Manager + Registry	<code>src/agents/AgentManager.ts</code>	KÉSZ ▾
Base Agent + Types	<code>src/agents/BaseAgent.ts, types.ts</code>	KÉSZ ▾
Permission System	<code>src/agents/permissions.ts</code>	KÉSZ ▾

#### B) Server & API (Express + Socket.IO)

Komponens	Fájl	Állapot
Web Server	<code>src/server/web.ts</code>	KÉSZ (Refaktorálva: 9... ▾)
Route modulok	<code>routes/agents, chat, health, tools, files, tasks, llm, external</code>	KÉSZ ▾
Socket.IO szolgáltatás	<code>src/server/SocketService.ts</code>	KÉSZ ▾
System Controller	<code>src/server/SystemController.ts</code>	KÉSZ ▾
Middleware & Error Handler	<code>src/server/middleware/errorHandler.ts</code>	KÉSZ (P3: Centralizált) ▾
Audit Routes	<code>src/server/auditRoutes.ts</code>	KÉSZ (P4: SQLite) ▾
Telemetry Routes	<code>src/server/telemetryRoutes.ts</code>	KÉSZ ▾

Phoenix Routes	<code>src/server/phoenixRoutes.ts</code>	KÉSZ ▾
Model Router Routes	<code>src/server/routerRoutes.ts</code>	KÉSZ ▾
Memory Routes	<code>src/server/memoryRoutes.ts</code>	KÉSZ ▾
Spec Routes	<code>src/server/specRoutes.ts</code>	KÉSZ ▾
Swagger/OpenAPI	<code>src/server/swagger.ts</code>	KÉSZ ▾
MCP Process Manager	<code>src/server/McpProcessManager.ts</code>	KÉSZ ▾
Tool Manager	<code>src/server/ToolManager.ts</code>	KÉSZ ▾

#### C) MCP Tool-ok (18 db)

Tool	Fájl	Cél
browser	<code>src/tools/browser.ts</code>	Böngésző vezérlés
browserBridge	<code>src/tools/browserBridge.ts</code>	Browser-use Python híd
ollamaTool	<code>src/tools/ollamaTool.ts</code>	Lokális LLM
geminiTool	<code>src/tools/geminiTool.ts</code>	Google Gemini
githubModelsTool	<code>src/tools/githubModelsTool.ts</code>	GitHub Models
knowledge	<code>src/tools/knowledge.ts</code>	RAG keresés
interpreter	<code>src/tools/interpreter.ts</code>	Python végrehajtás
workspace	<code>src/tools/workspace.ts</code>	Fájlkezelés

system	<code>src/tools/system.ts</code>	Rendszerparancsok
monitor	<code>src/tools/monitor.ts</code>	Monitorozás
n8n	<code>src/tools/n8n.ts</code>	n8n integráció
swarmTools	<code>src/tools/swarmTools.ts</code>	Cloudflare Workers
anythingllm	<code>src/tools/anythingllm.ts</code>	AnythingLLM
googleWorkspace	<code>src/tools/googleWorkspace.ts</code>	Google szolgáltatások
claudeTool	<code>src/tools/claudeTool.ts</code>	Claude integráció
copilotCliTool	<code>src/tools/copilotCliTool.ts</code>	Copilot CLI
julesCliTool	<code>src/tools/julesCliTool.ts</code>	Jules CLI
toolPermissions	<code>src/tools/toolPermissions.ts</code>	Jogosultságkezelés

#### D) Python Alrendszer (myai/)

Komponens	Fájl	Állapot
FastAPI Server	<code>myai/server.py</code>	KÉSZ (:8000) ▾
Browser Worker	<code>myai/browser_worker.py</code>	KÉSZ ▾
Browser Task Runner	<code>myai/browser_task_runner.py</code>	KÉSZ ▾
Refiner Logic	<code>myai/refiner_logic.py</code>	KÉSZ ▾
Pydantic Models	<code>myai/pydantic_models.py</code>	KÉSZ ▾
RAG	<code>myai/rag.py</code>	KÉSZ ▾
ChromaDB Adapter	<code>myai/chromadb_adapter</code>	KÉSZ ▾

	<code>.py</code>	
Config	<code>myai/config.py</code>	KÉSZ ▾
CLI	<code>myai/cli.py</code>	KÉSZ ▾
Vector DB Interface	<code>myai/vector_db_interface.py</code>	KÉSZ ▾
R2 Sync	<code>myai/sync_to_r2.py</code>	KÉSZ ▾
Dataset Manager	<code>myai/utils/dataset_manager.py</code>	KÉSZ ▾
PDF/Text parsers	<code>myai/utils/parser.py</code> , <code>pdfparser.py</code> , <code>textsplitter.py</code>	KÉSZ ▾

#### E) Dashboard (React + Vite + Shadcn/UI)

Komponens	Fájl	Állapot
Mission Control Layout	<code>MissionControlLayout.tsx</code>	KÉSZ ▾
Agent Management Panel	<code>AgentManagementPanel.tsx</code>	KÉSZ ▾
<i>... további 33+ panel</i>	<i>lásd a teljes táblázatot</i>	KÉSZ ▾
Auth	<code>auth/*.tsx</code>	KÉSZ ▾
Shadcn/UI komponensek	<code>ui/*.tsx</code> (50+ db)	KÉSZ ▾

#### F) Code Quality Sprints (Befejezve)

Sprint	Leírás	Állapot
P1	<code>web.ts</code> God File Refactoring (980 -> 200 LOC)	KÉSZ ▾
P2	<code>any</code> típusok eliminálása (15+ -> 0)	KÉSZ ▾
P3	Centralized Error Handling (AppError + middleware)	KÉSZ ▾
P4	Audit Log SQLite	KÉSZ ▾

	Persistence	
P5	Config Validation (Zod) - 211/211 teszt	KÉSZ ▾

#### G) Gold Protocol Sprints (Befejezve)

Sprint	Leírás	Állapot
G1	Spec Freeze	KÉSZ ▾
G2	Phoenix Protocol	KÉSZ ▾
G3	Model Router	KÉSZ ▾
G4	Cognitive Memory	KÉSZ ▾
G5	Glass Box Observability	KÉSZ ▾
G6	Audit	KÉSZ ▾
G7	Dashboard & CLI Integration	KÉSZ ▾

#### H) Adatréteg

Komponens	Fájl/Hely	Állapot
LanceDB (Node.js)	<code>data/brunella_lancedb /</code>	KÉSZ ▾
LanceDB (Python)	<code>data/brunella_lancedb _python/</code>	KÉSZ ▾
Task Queue SQLite	<code>data/tasks.db</code>	KÉSZ ▾
Audit SQLite	<code>data/audit.db</code>	KÉSZ ▾
Golden Dataset (JSONL)	<code>data/training/golden_ dataset.jsonl</code>	KÉSZ (struktúra kész) ▾

#### 3. Tesztelendő / Integrációra Váró Komponensek

Komponens	Állapot	Hiányzó / Következő Lépés
RAG Vector Search	1 FAIL teszt ▾	LanceDB embedding probléma ( <code>test/rag.test.ts:38</code> )
Incubator ( <code>train.py</code> )	Fájl KÉSZ ▾	Unsloth + CUDA tesztelés szükséges (RTX 3060)

Dataset Manager bekötés	Fájl KÉSZ ▾	Refiner-be való integráció tesztelése szükséges
n8n Workflow	Fájl NEM létezik ▾	<a href="#">n8n_workflows/</a> mappa üres - JSON generálandó
EV Hunter Task	Fájl NEM létezik ▾	<a href="#">myai/tasks/</a> mappa üres
EV Hunter Prompt	Fájl NEM létezik ▾	<a href="#">myai/prompts/</a> mappa üres
Cloudflare Worker	Fájl LÉTEZIK ▾	<a href="#">workers/</a> mappa üres
Dashboard Live Feed	Panel KÉSZ ▾	Valós Socket.IO screenshot stream tesztelés
Human-in-the-Loop	Panel KÉSZ ▾	Approve/Reject workflow tesztelés
Agent Genesis/Factory	UI KÉSZ ▾	Backend POST <a href="#">/api/factory/genesis</a> hiányzik
n8n/Langflow Bridge	Terv VAN ▾	Implementáció hiányzik

#### 4. Következő Lépések (Prioritások Szerint)Prioritás 1: "Zöld Villám" (EV Hunter) Élesítés

##### 1. Fájlok létrehozása:

- [myai/prompts/ev\\_hunter\\_prompt.md](#) (EV keresési prompt)
- [myai/tasks/ev\\_hunter.py](#) (Browser-use vezérlés)
- [n8n\\_workflows/ev\\_hunter.json](#) (Automatizálási workflow)
- [workers/scrapper-worker.js](#) (HTML tisztító)

##### 2. Python környezet beállítása: `pip install browser-use langchain-openai playwright`

##### 3. n8n integráció: Workflow importálás és `Execute Command` node tesztelése

##### 4. Élesítés: Szimulációs mód -> `browser-use` éles mód

#### Prioritás 2: Inkubátor (Data Flywheel)

1. [myai/utils/dataset\\_manager.py](#) bekötése a [refiner\\_logic.py](#)-ba
2. Golden Dataset gyűjtés automatizálása
3. [myai/incubator/train.py](#) tesztelése Unsloth + RTX 3060-on
4. GGUF export -> Ollama import pipeline kiépítése

#### Prioritás 3: Dashboard V3 "Mission Control"

1. Agent Swarm Commander - Execute Task modal + Socket.IO log stream

2. Robotkez Ops Center - Live screenshot feed + Start/Stop vezérlés
3. Neural Link Chat - Multi-provider selector (Ollama/GitHub/Gemini)
4. System Operations - Service toggles + Cloudflare Tunnel QR

#### Prioritás 4: Agent Genesis Module

1. POST `/api/factory/genesis` endpoint implementálása
2. Agent Architect prompt frissítése (n8n/Langflow JSON generálás támogatására)
3. Dinamikus gráf frissítés generálás után
4. Robotkez automatikus beállítása (opcionális)

#### Prioritás 5: Git & Dependency Cleanup

1. Branch takarítás (a megmaradt 4 feature branch elfogadható)
2. `npm audit fix` - dependency sebezhetőségek javítása
3. RAG teszt javítása (az 1 FAIL hiba elhárítása)

#### 5. Összegzés

Kategória	Szám
TypeScript ügynökök	17
MCP Tool-ok	18
Dashboard panelek	35+
Python modulok	13+
Tesztek	210 PASS / 1 FAIL
Aktív fejlesztési track-ek	7
Archivált track-ek	20+

A rendszer Core-ja **production-ready**. A következő nagy lépés a **"Zöld Villám" (EV Hunter) pipeline élesítése** és az **Inkubátor (Unsloth fine-tuning) beindítása**.