

IBM BOB

**Az AI fejlesztő Agent,
aki alkalmazást
fejleszt és tesztel**

Tamás Székely

IBM Data & AI Presales Engineer

tamas.szekely@arrow.com

ARROW





Mi az AI fejlesztő agent



Hagyományos eszközök	AI Fejlesztő Agent
Kód kiegészítés (autocomplete)	Teljes funkciók implementálása
Statikus elemzés	Kontextus-alapú döntéshozatal
Sablon alapú generálás	Kreatív problémamegoldás
Egy fájl szerkesztése	Multi-fájl koordináció
Manuális tesztelés	Automatikus teszt generálás

Miért most?

- Technológiai érettség
- Gyorsulás igénye
- Költséghatékonyság
- Minőség javítás



Képességek - Kódgenerálás, tesztelés, bug javítás, dokumentáció

Mód	Cél	Mikor használjuk
Code (Kód)	Kód írása és módosítása	Funkciók implementálásához, hibák javításához vagy kód fejlesztésekhez
Advanced (Haladó)	Komplex fejlesztési feladatok	Amikor teljes hozzáférésre van szükség minden eszközhöz, beleértve az MCP és böngésző képességeket
Plan (Tervezés)	Tervezés és dizájn	Implementálás előtt, amikor architektúrát kell tervezni vagy technikai specifikációt készíteni
Ask (Kérdez)	Információszerzés	Amikor magyarázatokra vagy információkra van szükség fájlok módosítása nélkül



Képességek - Kódgenerálás, tesztelés, hibajavítás, dokumentáció

Ask Mód



- Kód magyarázat
- Dokumentáció olvasása
- Tanácsadás
- Információ keresés

Code Mód



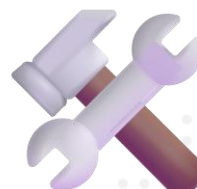
- Kód írása, refraktorálása
- Új funkciók implementálása
- Bug-ok javítása
- Kód optimalizálás

Plan Mód



- Architektúra tervezés
- Technikai specifikáció készítése
- Feladatok lebontása
- Stratégia kidolgozása

Advanced Mód



- Teljes eszköztár elérése
- MCP szerverek használata
- Böngésző automatizálás
- Komplex, többlépéses feladatok

Nyelvi támogatás

50+ programozási nyelvet támogatása

Általános Célú Nyelvek

- Python - AI/ML, backend, scripting
- JavaScript - Frontend, backend (Node.js)
- TypeScript - Type-safe JavaScript
- Java - Enterprise alkalmazások
- C# - .NET, Unity, enterprise
- C++ - Rendszerprogramozás, játékfejlesztés
- Swift - iOS/macOS fejlesztés

Markup & Config Nyelvek (8)

- JSON - Adatcsere
- YAML - Konfiguráció
- XML - Strukturált adat
- Markdown - Dokumentáció
- Dockerfile - Container definíció
- Terraform - Infrastructure as Code

Képesség	Leírás
Kódgenerálás	Teljes alkalmazások, funkciók, osztályok
Refraktorálás	Kód átírás, optimalizálás, modernizálás
Hiba javítás	Hibakeresés, javítás minden nyelven
Tesztelés	Unit, integration tesztek generálása
Dokumentáció	Kommentek, docstring-ek, README
Konverzió	Nyelv közötti átírás (pl. Python → TypeScript)
Best practices	Idiomatikus kód, design pattern-ek

Web Fejlesztés

- HTML5 - Markup
- CSS3/SCSS/SASS - Stíluslapok
- PHP - Backend scripting
- Ruby - Ruby on Rails

R - Statisztikai elemzés

- MATLAB - Numerikus számítások
- Julia - Tudományos számítások
- Perl - Text processing
- Bash/Shell - Scripting
- PowerShell - Windows automation

Adatbázis & Query Nyelvek

- SQL (MySQL, PostgreSQL, SQLite)
- T-SQL (SQL Server)
- PL/SQL (Oracle)
- MongoDB Query (NoSQL)
- GraphQL - API query nyelv

+ Funkcionális & Speciális Nyelvek + Mobile & Embedded



Fejlesztési folyamat

Bemenetek:

- Felhasználói kérés (természetes nyelv)
- Projekt struktúra (fájlok, mappák)
- Meglévő kód (releváns fájlok)
- Korábbi interakciók (history)
- Környezeti információk (OS, shell, tools)

Reasoning (Gondolkodás)

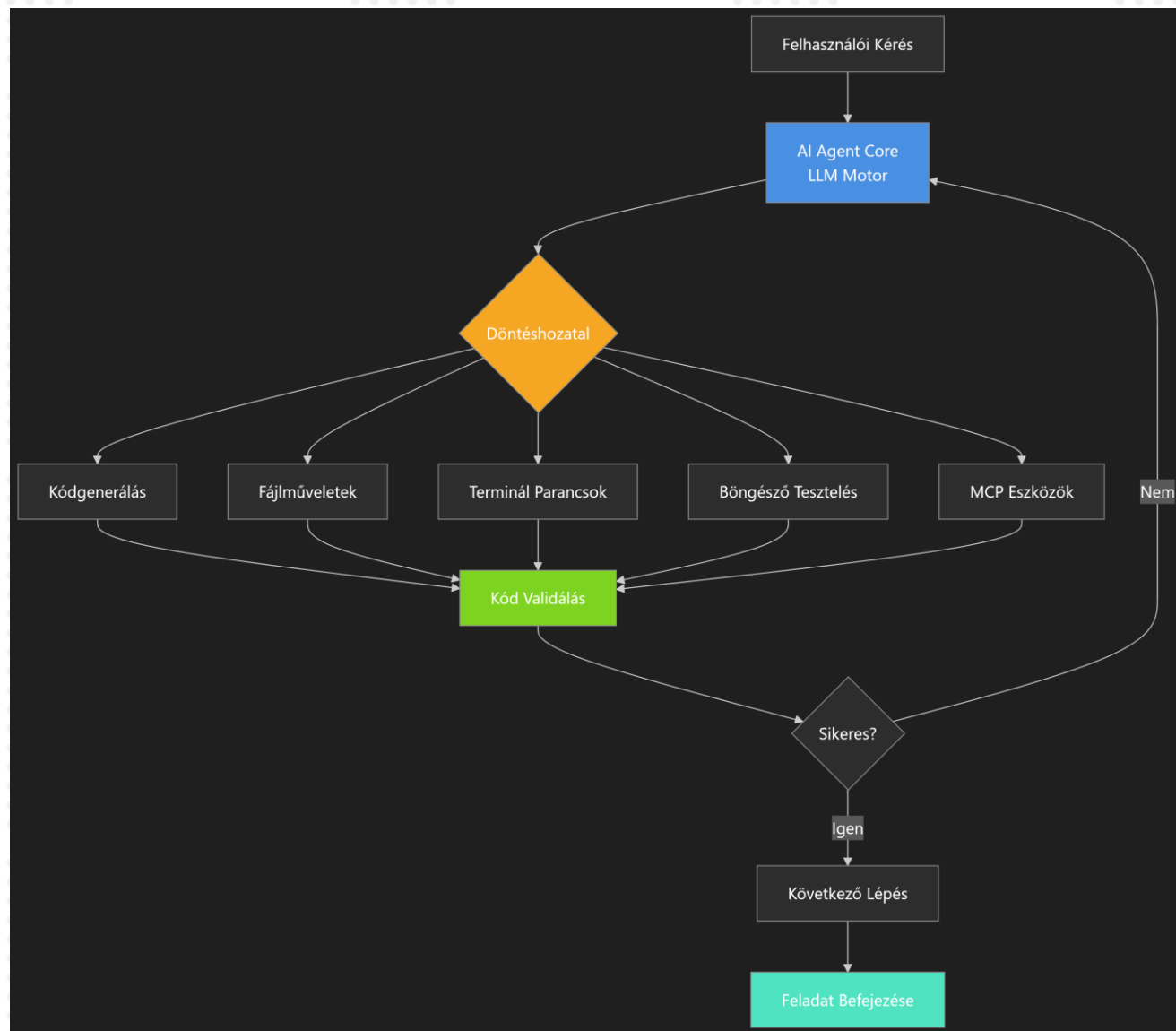
- **Feladat elemzés:** Mit kell elérni?
- **Stratégia tervezés:** Hogyan érjük el?
- **Kockázat értékelés:** Mi mehet rosszul?
- **Eszköz kiválasztás:** Melyik tool a legjobb?

Példa döntési folyamat

```
if task == "új_funkció":  
    1. read_file() → meglévő kód megértése  
    2. write_to_file() → új kód generálása  
    3. execute_command("npm test") → tesztelés  
    4. browser_action() → UI validálás
```

Visszacsatolás & Adaptáció

- Sikeres művelet → következő lépés
- Hiba történt → újrapróbálkozás más stratégiával
- Felhasználói feedback → finomhangolás



Fejlesztési folyamat

Bemenetek:

- Felhasználói kérés (természetes nyelv)
- Projekt struktúra (fájlok, mappák)
- Meglévő kód (releváns fájlok)
- Korábbi interakciók (history)
- Környezeti információk (OS, shell, tools)

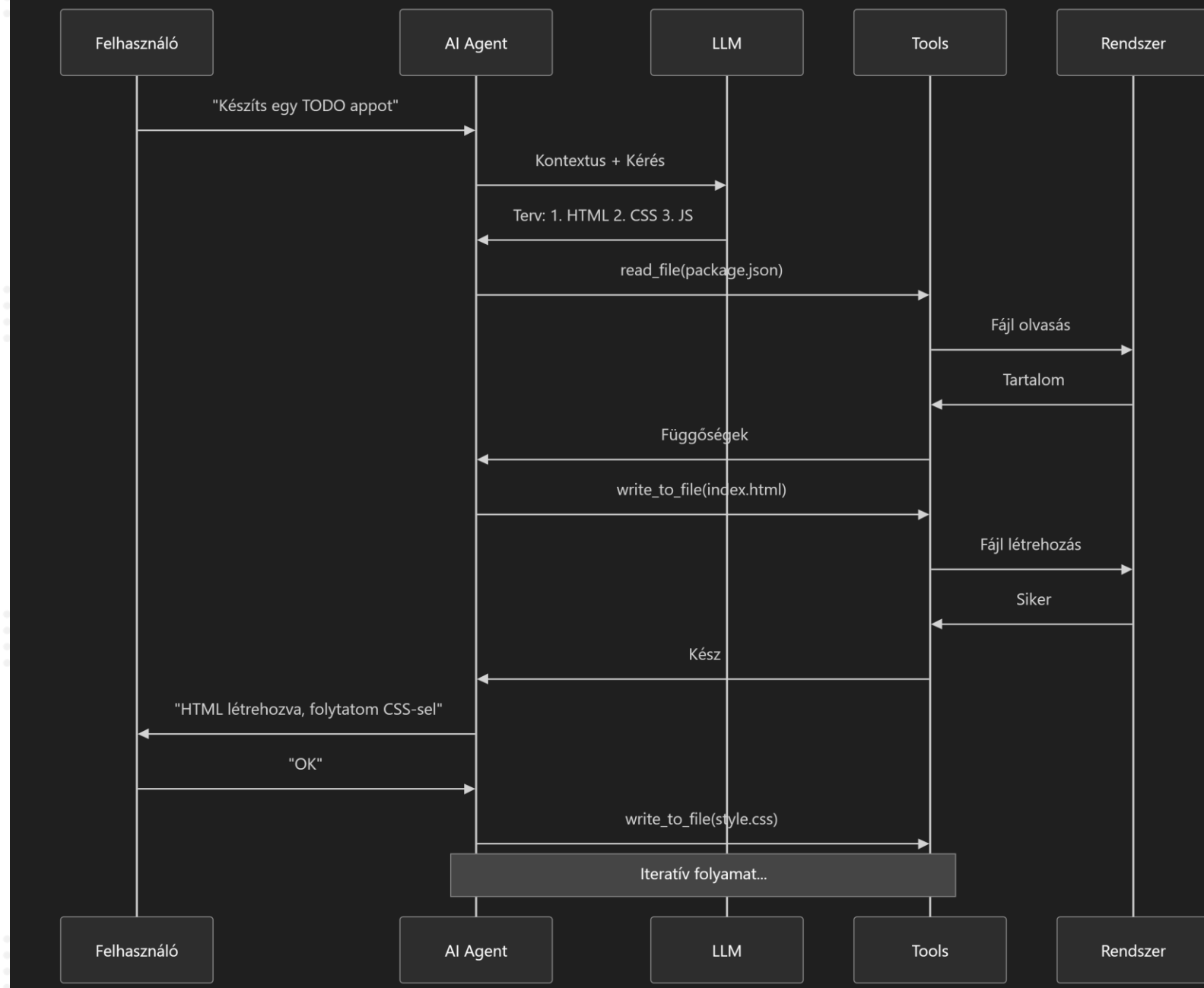
Reasoning (Gondolkodás)

- **Feladat elemzés:** Mit kell elérni?
- **Stratégia tervezés:** Hogyan érjük el?
- **Kockázat értékelés:** Mi mehet rosszul?
- **Eszköz kiválasztás:** Melyik tool a legjobb?

```
# Példa döntési folyamat
if task == "új_funkció":
    1. read_file() → meglévő kód megértése
    2. write_to_file() → új kód generálása
    3. execute_command("npm test") → tesztelés
    4. browser_action() → UI validálás
```

Visszacsatolás & Adaptáció

- Sikeres művelet → következő lépés
- Hiba történt → újrapróbálkozás más stratégiával
- Felhasználói feedback → finomhangolás



AI Agent Tesztelési Képességek



Unit Tesztek

Automatikus generálás:

- Minden függvényhez, metódushoz test case-ek
- Edge case-ek lefedése (null, empty, boundary values)
- Mock objektumok és stub-ok automatikus létrehozása
- Assertion-ök intelligens generálása

E2E (End-to-End) Tesztek

Teljes felhasználói folyamatok szimulálása:

- Regisztráció → Bejelentkezés → Funkció használat → Kijelentkezés
- Vásárlási folyamat (e-commerce)
- Form kitöltés és validáció
- Multi-step wizard-ok

Integration Tesztek

Komponensek közötti interakciók tesztelése:

- API endpoint-ok validálása
- Adatbázis műveletek tesztelése
- Külső szolgáltatások mock-olása
- Middleware és authentication flow-k

Böngésző Alapú Tesztelés

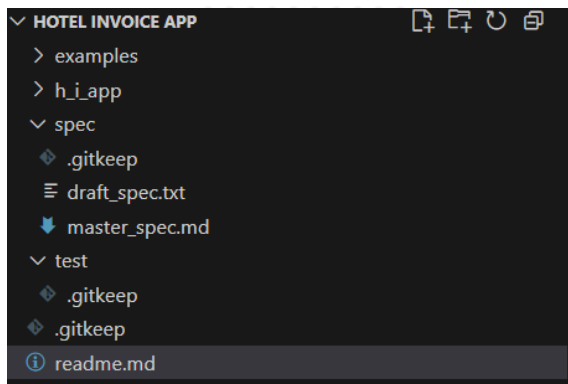
Automatizált UI interakciók:

- Kattintások, scroll, form kitöltés
- Screenshot és video felvétel
- Network request monitoring
- Console log elemzés
- Performance metrics



Hotel Invoice App

Alkalmazás funkciói



Szállodai Számla Feldolgozás

Ebben a projektben spec coding segítségével fejlesztünk egy Python-alapú alkalmazást szállodai számla PDF-ek feldolgozására, költségadatok kinyerésére AI használatával, és az eredmények megjelenítésére egy interaktív webes felületen keresztül.

Funkciók

- **PDF Feldolgozás**: 1-5 szállodai számla PDF egyidejű feltöltése és feldolgozása
- **AI-alapú Adatkinyerés**: IBM watsonx.ai használata az intelligens számlaadat-kinyeréshez
- **Interaktív Vizualizációk**: Költségek megtekintése szállodánként és kategóriánként interaktív diagramokkal
- **Adatexportálás**: Kinyert adatok letöltése CSV formátumban
- **Modern Felhasználói Felület**: Tiszta, reszponzív Streamlit felület

Technológiai Stack

- **UI Keretrendszer**: Streamlit
- **Dokumentumfeldolgozás**: Docling (IBM)
- **AI/LLM**: IBM watsonx.ai
- **Adatkezelés**: Pandas
- **Vizualizáció**: Plotly

Előfeltételek

- Python 3.11 vagy újabb
- IBM watsonx.ai fiók API hozzáféréssel



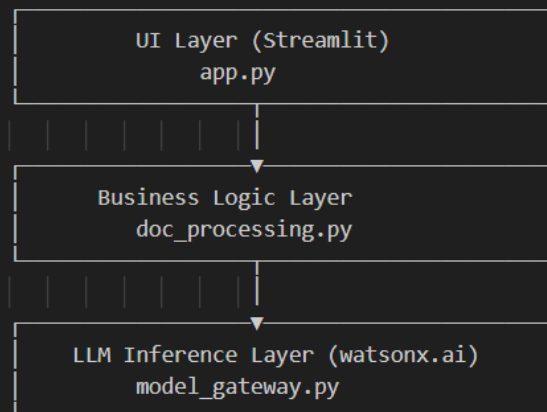
Master_spec.md

2. Architecture

2.1 Component Overview

The application follows a three-tier architecture:

...



...

2.2 Component Responsibilities

2.2.1 UI Layer (`app.py`)

- File upload interface
- User interaction controls
- Data presentation (tables and graphs)
- Session state management

2.2.2 Business Logic Layer (`doc\_processing.py`)

- PDF to Markdown conversion using Docling
- Data orchestration between UI and LLM
- Data transformation and aggregation

✓ ## 3. Detailed Specifications

✓ ### 3.1 UI Specifications (`app.py`)

✓ ##### 3.1.1 Design Requirements

- **Style**: Modern, elegant HTML/CSS styling
- **Responsiveness**: Clean layout with proper spacing
- **Color Scheme**: Professional color palette
- **Typography**: Clear, readable fonts

✓ ##### 3.1.2 Components

✓ ##### File Upload Section

- **Type**: Multi-file uploader
- **Accepted Format**: PDF only
- **Maximum Files**: 5
- **Validation**: File type and count validation
- **Feedback**: Display uploaded file names and sizes

✓ ##### Control Buttons

✓ 1. **Submit Button**

- **Label**: "Submit"
- **Function**: Trigger invoice processing
- **State**: Disabled when no files uploaded
- **Feedback**: Show processing spinner during execution

✓ 2. **Analyze Button**

- **Label**: "Analyze"
- **Function**: Generate and display graphs
- **State**: Disabled until data is processed
- **Feedback**: Show loading indicator during graph generation

File Edit Selection View Go Run Terminal Help

hotel invoice app

IBM BOB

EXPLORER

HOTEL INVOICE APP

examples

.gitkeep

example1.pdf

example2.pdf

example3.pdf

spec

.gitkeep

draft_spec.txt

test

.gitkeep

.gitkeep

readme.md

OUTLINE

TIMELINE

readme.md

readme.md > # Szállodai Számla Feldolgozás > ## Funkciók

28

29

30

31

32

33

34

35

36

37

38

39

40

41

42

43

44

45

46

47

48

49

50

51

52

53

Szállodai Számla Feldolgozás

Ebben a projektben spec coding segítségével fejlesztünk egy Python-alapú alkalmazást szállodai számla PDF-ek feldolgozására, költségadatok kinyerésére AI használatával, és az eredmények megjelenítésére egy interaktív webes felületen keresztül.

Funkciók

- **PDF Feldolgozás**: 1-5 szállodai számla PDF egyidejű feltöltése és feldolgozása

- **AI-alapú Adatkinyerés**: IBM watsonx.ai használata az intelligens számlaadat-kinyeréshez

- **Interaktív Vizualizációk**: Költségek megtekintése szállodánként és kategóriánként interaktív diagramokkal

- **Adatexportálás**: Kinyert adatok letöltése CSV formátumban

- **Modern Felhasználói Felület**: Tiszta, responszív Streamlit felület

Technológiai Stack

- **UI Keretrendszer**: Streamlit

- **Dokumentumfeldolgozás**: Docling (IBM)

- **AI/LLM**: IBM watsonx.ai

- **Adatkezelés**: Pandas

- **Vizualizáció**: Plotly

Előfeltételek

- Python 3.11 vagy újabb

- IBM watsonx.ai fiók API hozzáféréssel



Hi, I'm IBM Bob

Ask me questions or let me code for you.

Recent Tasks

See all

a readme.md file-ba tedd bele a magyar fordítását is

Just now - 0.03

What would you like me to help with?

(@ to add context, / for commands, hold shift to drag in files/images)

Code

ADK: 2.6.0

Ln 39, Col 74 (36 selected)

Spaces: 4

UTF-8

LF

{ } Markdown

Bob - Settings

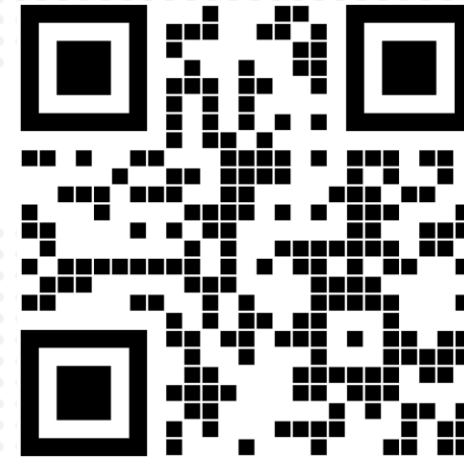
Bob már elérhető!

A dark-themed banner for IBM Bob. At the top, there are several icons: a code editor window with a terminal prompt, a toggle switch, a robot character wearing a blue hard hat and holding a code icon, a server rack, a pie chart, and a network diagram. Below these icons, the text reads: "Welcome to IBM Bob: Your AI-Powered Development Partner". Underneath this, a smaller line of text says: "Hi, I'm Bob! I'm here to work right alongside you in your codebase, and help you build quality software faster." At the bottom, there are two buttons: "Get free trial" and "Download".

Welcome to IBM Bob:
Your AI-Powered Development Partner

Hi, I'm Bob! I'm here to work right alongside you in your codebase, and help you build quality software faster.

[Get free trial](#) [Download](#)



<https://bob.ibm.com>

Köszönöm a figyelmet!

Tamás Székely

IBM Data & AI Presales Engineer
tamas.szekely@arrow.com



watsonx | AI Lab

ARROW

