How Mendix is using AI to revolutionize application development

Ivan Zub / Team Lead & Software Engineer / Mendix
Yevgen Nerush / Lead Software Engineer / Mendix
Examples of intelligent assistants in the wild

**Office Assistant (aka Clippy)**
- An intelligent user interface that helped to interact with the Office help content

**Excel**
- Auto Fill feature to fill cells with data that follows a pattern or are based on data in other cells

**Visual Studio**
- Saves you time by putting what you’re most likely to use at the top of your completion list based on thousands of open source projects
Agenda

1. How Mendix Assist was born?
2. How can it help you be more productive?
3. How does it work?
Research projects in Mendix
Just a few of the features that came from the research projects:

- Nanoflows
- Native mobile
- Instant local deployments
- Alignment tools for microflows
Number of all patterns of all microflow objects: 38^8

Contain just a single activity: 13%

Microflows that are called from a page and can be converted to a nanoflow: 25%

Contain more than 100 activities: 0.5%

Can you guess what this number refers to? 2355
Mendix Assist in Studio Pro

DEMO
Mendix Assist in Studio Pro

1. Helps you to develop your logic
   - Available everywhere in the microflow
   - Shows the most relevant suggestions
   - Tries to prefill the object for you

2. Knows the context of your application
   - The context in which your microflow is called
   - Properties of the microflow, like parameters and return type
   - Entities and attributes that are relevant to this context

3. Evolves together with you, Mendix and the community
   - Regularly analyzes anonymized project data
   - Learns new features together with Mendix users
   - Evolves independently from Studio Pro releases
Next Steps

Patterns

Personalization

Error detection and prevention

Learning
So, how does it work?
The Future of Low-Code: AI-Assisted Development

- Assistive Intelligence
- Augmented Intelligence
- Autonomous Intelligence
- Anthropomorphic Intelligence
Machine Learning Pipeline

Data Ingestion

- Continuously collect anonymized microflow data to enable incremental learning

Data Preparation

- Clean, linearize, augment, and convert data to one-hot encoded vectors

Model Training

- Train Deep Recurrent Neural Network until we achieve good accuracy

Model Deployment

- Convert trained model to TensorFlow compatible format and serve it in a cluster of stateless microservices

Feedback Loop
Data Preparation

1. Anonymize microflows by filtering metadata

2. Perform data cleaning and filter out anti-patterns
MARS: Neural Network Architecture

Context aware RNN

Residual RNN

Context embedding

Residual embedding

concat

DNN

objective
Model Training

- 88 million samples to train, validate and test a model

- 3 days to train a model on a VM with GPU Tesla V100-SXM2-16GB and 488 GB of RAM

- 0.031 validation variance
  0.029 test variance

- 20 milliseconds inference time
+15% of Accuracy

![Bar Chart showing accuracy comparison between Prev Gen and Next Gen for Top 1 to Top 5.](chart.png)

- Top 1: Prev Gen 22%, Next Gen 37%
- Top 2: Prev Gen 37%, Next Gen 51%
- Top 3: Prev Gen 46%, Next Gen 60%
- Top 4: Prev Gen 52%, Next Gen 67%
- Top 5: Prev Gen 57%, Next Gen 72%
By becoming a more experienced Mendix Developer, you will create apps faster with better quality and can make Mendix smarter through a feedback loop.
We are extremely excited to see what you can build with Mendix Assist!
Go make it.