# Automation-as-Code with Python - Robocorp (Sema4.ai)

**Fernando Santana Pacheco**

October 2024

# Automation-as-Code

- **Definition**
  - Implementing automation using standard programming languages
  - Managing automation like regular software development
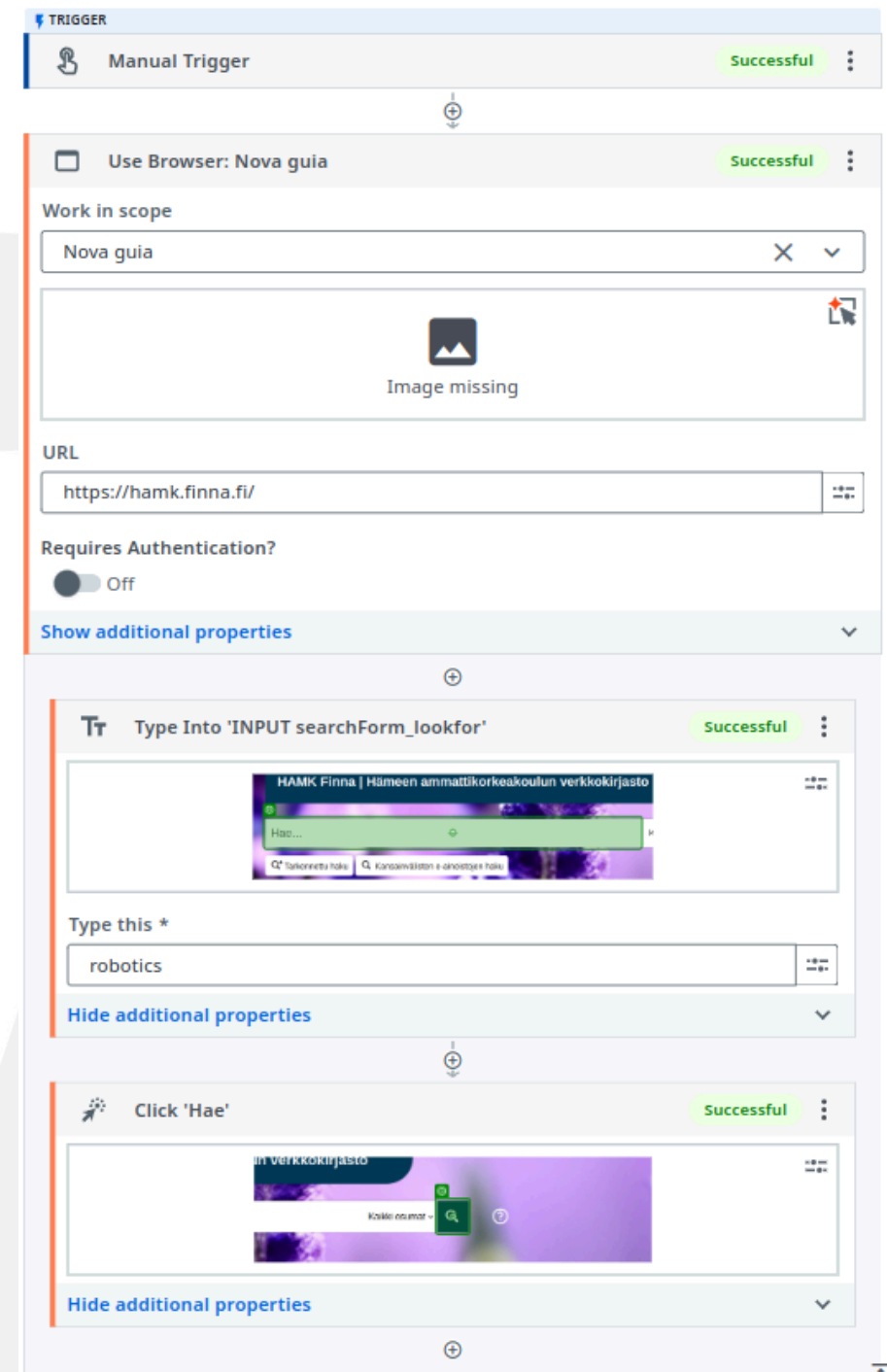  - Following DevOps and Software Engineering practices

# Key Principles of Automation-as-Code

- Version Control (Git)

- Code Review Process

- CI/CD (continuous integration/continuous delivery) Integration

- Infrastructure as Code (IaC)

- Test-Driven Development

- Modular Design

# Example: Robocorp vs UIPath

```python
#imports
@task
def open_page():
    page = browser.goto("https://hamk.finna.fi/")
    page.locator("#searchForm_lookfor").fill("robotics")
    page.locator(".btn-primary > .fi-basic-search").click()
    browser.screenshot()
```
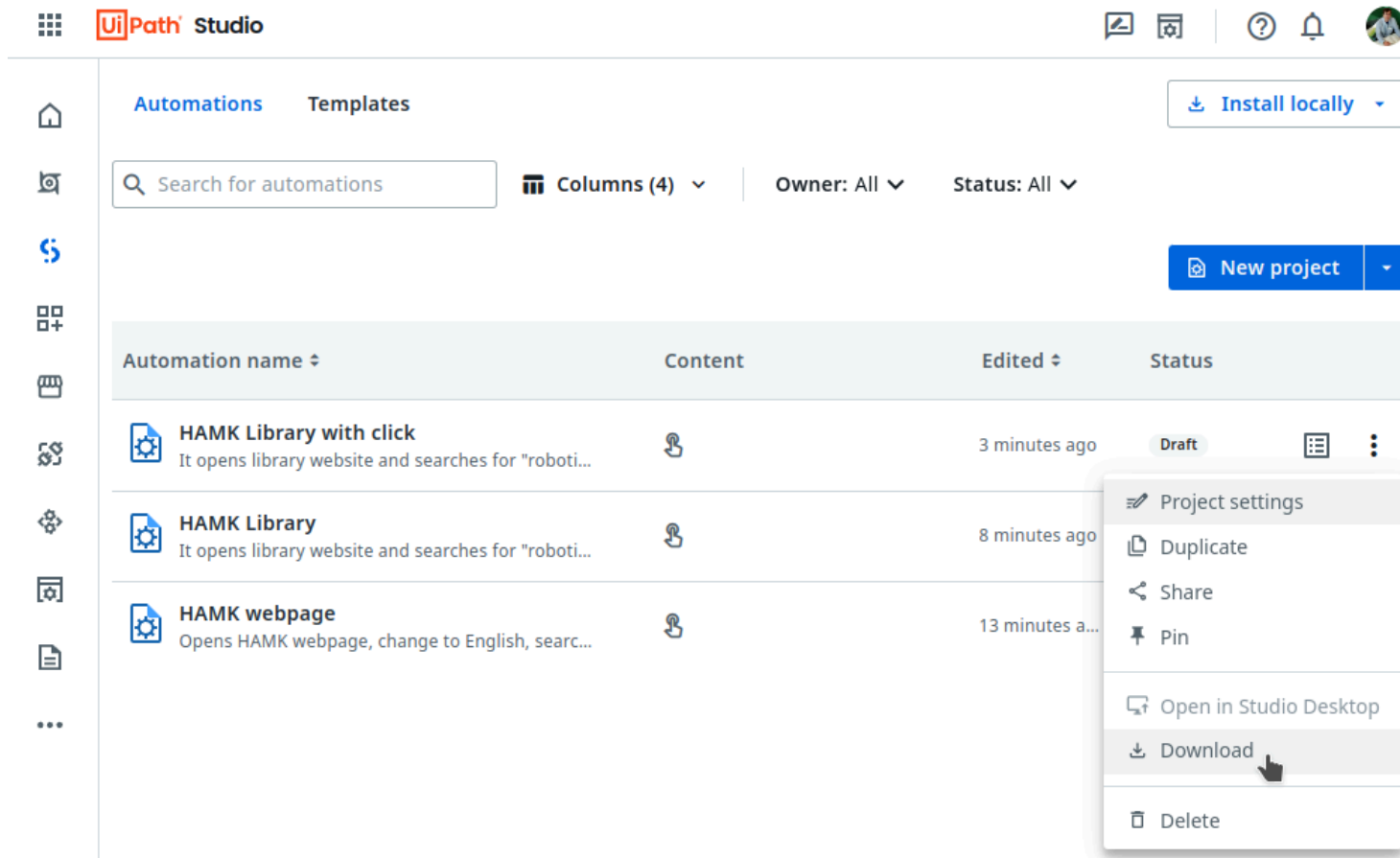
- Text- vs visual-based (block) programming

# Text-based in UIPath

To be precise, you can also use text-based programming in UIPath 🥳

# But...Text-based in UIPath is XML 🙁

```xml
66          <AssemblyReference>UiPath.UIAutomationNext.Activities.Design</AssemblyReference>
67        </scg:List>
68      </TextExpression.ReferencesForImplementation>
69      <Sequence sap:VirtualizedContainerService.HintSize="356,732" sap2010:WorkflowViewState.IdRef="Sequence_1">
70        <sap:WorkflowViewStateService.ViewState>
71          <scg:Dictionary x:TypeArguments="x:String, x:Object">
72            <x:Boolean x:Key="IsExpanded">True</x:Boolean>
73          </scg:Dictionary>
74        </sap:WorkflowViewStateService.ViewState>
75        <ui:ManualTrigger Result="{x:Null}" DisplayName="Manual Trigger" sap2010:WorkflowViewState.IdRef="ManualTrigger_1" />
76        <uix:NApplicationCard AttachMode="SingleWindow" CloseMode="IfOpenedByAppBrowser" DisplayName="Use Browser: Nova guia" sap2010:WorkflowViewState.IdRe
77          <uix:NApplicationCard.Body>
78            <ActivityAction x:TypeArguments="x:Object">
79              <ActivityAction.Argument>
80                <DelegateInArgument x:TypeArguments="x:Object" Name="WSSessionData" />
81              </ActivityAction.Argument>
82              <Sequence DisplayName="Do" sap2010:WorkflowViewState.IdRef="Sequence_2">
83                <uix:NTypeInto ActivateBefore="True" ClickBeforeMode="Single" DisplayName="Type Into 'INPUT searchForm_lookfor'" EmptyFieldMode="SingleLine"
84                  <uix:NTypeInto.Target>
85                    <uix:TargetAnchorable BrowserURL="hamk.finna.fi" CVScreenId="de37aa6a-0986-446c-8333-04d50db0e82a" CvElementArea="379, 198, 758, 50" CvT
86                  </uix:NTypeInto.Target>
87                </uix:NTypeInto>
88                <uix:NClick ActivateBefore="True" ClickType="Single" DelayAfter="5" DisplayName="Click 'Hae'" sap2010:WorkflowViewState.IdRef="NClick_1" Key
89                  <uix:NClick.Target>
90                    <uix:TargetAnchorable BrowserURL="hamk.finna.fi" CVScreenId="94532730-f1d2-4070-a20d-a8d6f86f96ec" CvElementArea="1131, 197, 50, 50" CvT
91                  </uix:NClick.Target>
92                </uix:NClick>
93              </Sequence>
94            </ActivityAction>
95          </uix:NApplicationCard.Body>
96          <uix:NApplicationCard.TargetApp>
97            <uix:TargetApp Area="0, 0, 0, 0" BrowserType="Chrome" Selector="&lt;html title='Nova guia' app='chrome.exe' /&gt;" Title="Nova guia" Url="https:
98          </uix:NApplicationCard.TargetApp>
99        </uix:NApplicationCard>
100     </Sequence>
101   </Activity>
```

# Automation-as-Code vs Traditional RPA

| Aspect | Automation-as-Code | Traditional Automation |
| --- | --- | --- |
| Development | Code-first approach | GUI-based development |
| Version Control | Git-based | Limited or proprietary |
| Reusability | Modular components | Limited reusability |
| Testing | Unit tests, integration tests | Manual testing |
| Deployment | CI/CD pipelines | Manual deployment |
| Maintenance | Standard code maintenance | Platform-dependent |
| Learning Curve | Steeper (requires coding) | Gentler (low-code) |

# Some Tools for Automation-as-Code RPA

- BotCity

- Python RPA

- Robocorp
  - Transitioning to Sema4.ai (29/Jan/2024)
  - Moving away from Robot Framework to full Python solution (12/Feb/2024)

# (Now Old) Robocorp with Robot Framework

- Example: search "robotics" in HAMK's library website

- Easy to read, but a different syntax/language

- Anyway, Robot Framework continues to be a great tool
  - Specially for test automation (more about it latter)

```
*** Tasks ***
Search For Robotics Books
    Open Browser       ${URL}      chrome
    Maximize Browser Window
    Wait Until Element Is Visible     ${SEARCH_BOX}      timeout=10s
    Input Text      ${SEARCH_BOX}      robotics
    Click Button     ${SEARCH_BUTTON}
    Wait Until Element Is Visible     ${RESULTS_PAGE}      timeout=10s
    Sleep     2s      # Wait for results to load completely
    Capture Page Screenshot     robotics_search_results.png
    [Teardown]     Close Browser
```

# Robocorp features

- Open-source RPA framework

- Built for developers and Python enthusiasts

- Alternative to traditional RPA platforms

- Focus on software development best practices

- Reviews in Gartner report
    - "(Robocorp) is doing away with the cumbersome drag-and-drop, rigid and inefficient graphical programming paradigm of the traditional RPA giants"
    - "by developers for developers"

# Key Advantages of Robocorp

- Native **Python Syntax**
  - No need to learn custom keywords
  - Full access to Python ecosystem (modules and packages)
  - Better IDE support and tooling
- Better Debugging
  - Standard Python debuggers work out of the box
  - Easy to set breakpoints and inspect variables
  - Clear stack traces

# Code Example: Browser Automation

```python
from robocorp.browser import Browser

def search_google():
    browser = Browser()
    page = browser.open_page("https://google.com")

    # Modern selector syntax
    search_box = page.get_element("input[name='q']")
    search_box.type("Robocorp Python")

    # Built-in waiting mechanisms
    page.click("input[name='btnK']")
```

# Cloud Features of Robocorp

- You can run locally in your computer or on the cloud

- Cloud solution
  - Control Room for orchestration
  - Built-in logging and monitoring
  - Process and task analytics

# RPA vs Test Automation: Key Differences

- **RPA** automates business processes end-to-end

  - Replaces manual repetitive tasks

  - Focuses on production workflows

  - Handles real business data

- **Test Automation** validates software functionality

  - Uses test data and environments

  - Focuses on finding defects

  - Typically runs in test environments

# RPA vs Test Automation: Technical Aspects

| Aspect | RPA | Test Automation |
|---|---|---|
| Error Handling | Business continuity focus | Bug detection focus |
| Data | Production data | Test data |
| Environment | Production systems | Test environments |
| Scheduling | Business hours/on demand | CI/CD pipeline |
| Success Metric | Process completion | Test coverage |
| Reporting | Business metrics | Test results |
| Security | Production credentials | Test credentials |

# Summary: RPA vs Test Automation

- RPA collects and fills real data
- Test automation finds bugs in software
    - Very important for developers
    - Robot Framework is one of the best tools

Robot Framework is an open source automation framework for test automation and robotic process automation (RPA). It is supported by the Robot Framework Foundation and widely used in the industry.

Its human-friendly and versalite syntax uses keywords and supports extending through libraries in Python, Java, and other languages.

It integrates with other tools for comprehensive automation without licensing fees, bolstered by a rich community with hundreds of 3rd party libraries.

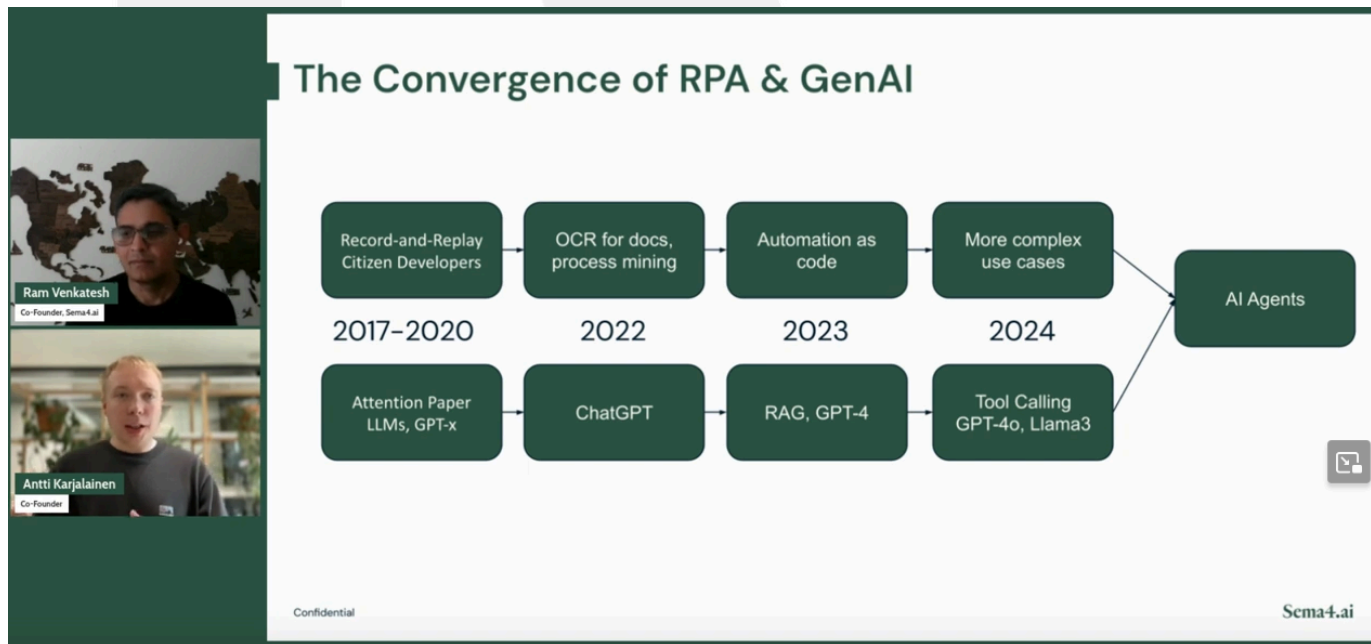# Getting started with Robocorp Sema4.ai

- Install VS Code

- Install Sema4.ai Extension for VS Code

- Keep in mind that many things are changing
  - Updates at https://updates.sema4.ai/

- Use the assistant ReMark

- Run the example for searching a keyword in HAMK's library website

# Next steps

- Practice at RPA Challenge

- Developer training at Sema4.ai

- Keep updated at Sema4.ai blog
  - Enterprise AI Agent – The Future of Knowledge Work (Oct 17, 2024)
  - Sema4.ai Recognized in the Gartner Hype Cycle... (Aug 6, 2024)
  - From RPA to Enterprise AI Agents (May 8, 2024)

# (Near) Future

Webinar: From RPA to AI Agents: The shift in automation every leader needs to know (published Sept 26, 2024)

# (Near) Future (cont.)

Webinar: From RPA to AI Agents: The shift in automation every leader needs to know (published Sept 26, 2024)