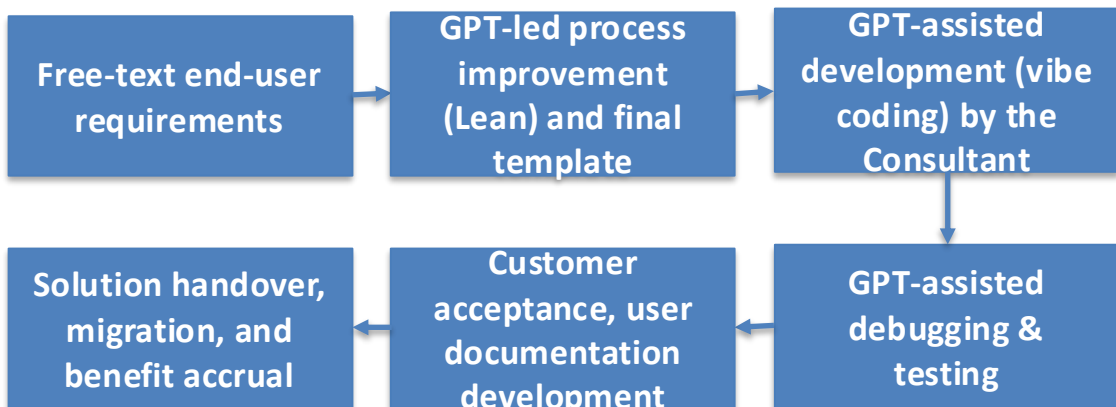


# End user-defined GPT-shaped automated workflows that make practical sense.

## CHALLENGE

- A customer spots that any business process around them requires optimisation and automation.
- Software vendors cannot afford to solve customers' niche problems other than by including them in the next edition of the mass software release (not guaranteed).
- A customer doesn't want to wait for the corporate digital transformation project – no time, budget, resources or all of those. They want simple, quick, inexpensive, and plug-and-play solutions to improve everyday business process efficiency.
- Any such solution must be secure and acceptable to the IT department and easy to use.
- Such a solution must demonstrate its benefits to any concerned party.



## SOLUTION

- A customer briefly describes a legacy process and formulates a free-text automation requirement for the custom GPT.
- A custom GPT prepares a standardised lean-optimised solution description to be approved by the client before automation. IT may review and accept/edit it, as well.
- A solution is based on a familiar and readily-available Microsoft 365 ecosystem (Power Apps, SharePoint, Outlook, Teams) with its embedded AI tools (AI Builder).
- Data security and governance are not compromised, as no third party enters the corporate IT environment.
- Based on the customer's internally approved solution, the Consultant prepares the automation-first Power App solution, with AI functionality, only if required and acceptable for a customer.
- The Consultant hands over the Power App portable “managed” solution (zip file with the solution description and user instructions), which the end-user can implement or ask the IT to, and start using in minutes.
- **Power Apps naturally track time and cost savings,** which the end user can report as additional benefits.
- Each workflow includes an audit trail of critical artefacts and evidence.

**2-3 weeks  
from an idea to  
the benefits**

# Proposed commercial model (with demo rates)

## Scenario 1: Cash payment

### Typical estimate

1. GPT-assisted process analysis, improvement, and standardized definition: \$500 fixed fee (GPT execution)
2. Process analysis, end-user interview and final process sign-off: 4 hours x \$100 = \$400
3. Solution development: 20 hours x \$100 = \$2,000
4. Debugging and testing: 10 hours x \$100 = \$1,000
5. Acceptance and user documentation: 4 hours x \$100 = \$400

**TOTAL: \$4,300 (one-off) + 20% annually** (bugfixes and updates/improvements).

## Scenario 2: Savings share\*

### Typical estimate

1. Time savings per flow run (agreed with the Client) = 10 min
2. Flow runs per day = 5
3. Flow runs per annum = 250 w.d. x 5 = 1,250
4. Total time savings = 12,500 min = 208,33 hours
5. Average hourly rate of back-office staff = \$30
6. Savings share = 50%/50%

**TOTAL: (\$3,125 + 20%) annually** (bugfixes and updates/ improvements, billed quarterly, based on Power App flow stats)

\* Savings assumptions and measurement methodology to be agreed and documented prior to delivery.

**Client is responsible for providing Microsoft 365 licenses, including Outlook, Teams, Power Apps, SharePoint, AI Builder/Copilot Studio**

The screenshot shows the Microsoft Power Automate web interface. On the left is a navigation pane with options like Home, Create, Templates, Learn, My flows, Approvals, Solutions, Automation center, Process mining, AI hub, and Desktop flow activity. The main area displays details for a flow named 'II - Customer Onboarding'. The 'Details' tab shows the flow name, primary owner (Sergii Dovgalenko), creation date (Dec 30, 2025 at 10:58 AM), modification date (Dec 31, 2025 at 08:05 AM), type (Automated), and plan (This flow runs on owner's plan). The 'Savings' tab is active, showing a 'Start tracking your automation impact' button and a simulated time saved of 20 minutes over the last 7 days. A note states: 'Assuming a 10-minutes saving per successful run, you'd have saved 20 minutes from 2 run in the past week. Learn more'. An 'Enable savings' button is at the bottom.

The screenshot shows the 'Savings' settings panel. It includes a header 'Savings' and a sub-header 'With savings rules, measure the impact of your automation in terms of time saved and money saved. Learn more'. There are two main sections: 'Enable time-saving rule' and 'Enable money-saving rule'. The 'Enable time-saving rule' section has a toggle switch set to 'On' and a field to define manual processing time per successful run, currently set to 10 minutes. The 'Enable money-saving rule' section has a toggle switch set to 'On' and a field to define the hourly rate, currently set to 10, which equates to UAH 1.67 per successful run. There is also a section for 'Calculate money savings with user-defined baseline' which is currently disabled. At the bottom, it says 'Last rule update: N/A'.