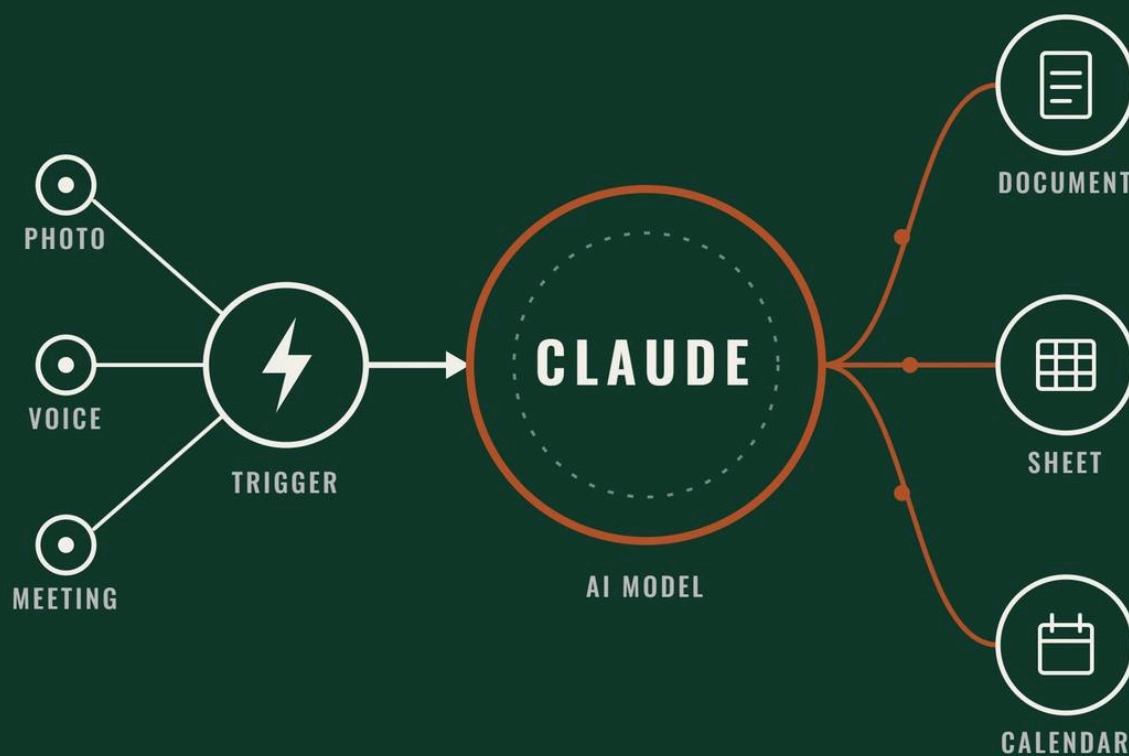




AI

FOR WORK AND LIFE

A digest of automations built without a
programmer

Contents

00

Introduction:

why this digest exists and what it is built from

01

Work:

4 automations for Company #1, #2 and #3

02

Family:

school, parent chats, small-business bookkeeping

03

Other:

short summaries of long articles

04

In the queue:

5 tasks waiting in line

05

What to take away:

4 techniques and a checklist for your own task

Quick navigation

If you...

...are drowning in minutes and tasks after meetings

01

...want to lighten home life and a family business

02

If you...

...retype data from documents by hand

01 02

...are ready to start but don't know where

05

AI without a programmer: what already works

Over a few months I built 13 AI automations. Eight already run every day; five more are in development or planned.

All of them were built without an in-house programmer. I describe the task, and Claude itself helps build and debug the scenario on a no-code platform.

This is not theory but real processes: meeting minutes, transport documents, a school schedule, bookkeeping in a family business.

What you will get:

The problem, solution and tools for each case.

A “BEFORE” and “AFTER” comparison for each automation.

Four techniques you can bring to your own department, and a checklist for your first task.

What it is built from

Make.com — the “conveyor”: connects services and runs steps on an event or a schedule.

Claude (Anthropic) — reads text, photos and scans, writes summaries and returns structured data.

OpenAI Whisper — turns voice messages into text.

Telegram bots — the interface: send a photo, voice note or file, get the result.

Google Docs, Sheets, Drive, Calendar, Gmail — templates, records, archive, calendar and mail.

Claude Code — helps build and debug the scenarios themselves.

8

running daily or in pilot

5

in development, planned or on hold

3

areas: work, family and other

Status as of 1 October 2026

RUNNING

PILOT

PARTIAL

IN DEVELOPMENT

PLANNED

ON HOLD

Minutes, tasks and documents without manual entry

Case 1: Meeting summaries

RUNNING

Problem: after a meeting, minutes are written by hand or not at all. Agreements and deadlines get lost.

Solution: Sembly records the meeting and sends the transcript to Make. Claude writes a summary: topics, decisions, tasks with owners and deadlines. The summary goes into a template, is saved to a shared folder and arrives by email as a PDF.

BEFORE → AFTER

MINUTES

BEFORE: written by hand or not at all

AFTER: arrive by email after the meeting

TASKS

BEFORE: in participants' heads

AFTER: a list with owners and deadlines

ARCHIVE

BEFORE: scattered files

AFTER: one folder, one template

Tools: Sembly · Make · Claude · Google Docs · Drive · Gmail

Case 2: Action items → Claude dashboard

RUNNING

Problem: tasks from dozens of online-meeting minutes are scattered across documents. There is no overall picture of who does what and what is overdue.

Solution: twice a day the scenario collects new summaries, Claude extracts the tasks, and they land in a Google Sheet without duplicates. Statuses and comments are set by hand; the Claude dashboard shows the overall picture.

BEFORE → AFTER

TASKS

BEFORE: in dozens of online-meeting minutes

AFTER: in one spreadsheet

STATUS

BEFORE: unknown

AFTER: from “Not started” to “Done”

UPDATES

BEFORE: by hand, when someone remembers

AFTER: twice a day, autonomously

Tools: Make · Claude · Google Sheets · Drive · Claude dashboard

Case 3: Powers of attorney from a B/L

RUNNING

Problem: the power of attorney for cargo release is filled in by hand, retyping data from the bill of lading. Slow, with a risk of typos.

Solution: a photo or scan of the B/L is sent to a Telegram bot. Claude reads the document, the bot picks the template for the shipping line and asks who the power of attorney is for. A ready PDF comes back. Access is whitelist-only.

BEFORE → AFTER

DATA

BEFORE: retyped from the B/L

AFTER: read from a photo

TEMPLATE

BEFORE: chosen by hand

AFTER: matched to the line

RESULT

BEFORE: editing a Word document

AFTER: a ready PDF in the chat

Tools: Telegram · Make · Claude · Google Docs · Drive

Case 4: CMRs for LCL shipments

PILOT

Problem: the operator manually transfers data from the B/L, invoice and packing list into a CMR — for every shipment. One container = 10+ CMRs typed by hand.

Solution: the Telegram bot takes an Excel file with a batch of shipments (one row = one CMR) or a package of source documents, which it reads itself. The warehouse is picked with a button from a directory. The output is a CMR in PDF and Word, with a copy sent to the archive.

BEFORE → AFTER

INPUT

BEFORE: three documents by hand

AFTER: Excel or a package of files

VOLUME

BEFORE: one CMR at a time

AFTER: a batch from one Excel file

ARCHIVE

BEFORE: in the operator's folders

AFTER: a shared folder on Drive

Tools: Telegram · Make · Claude · Google Sheets · Docs · Drive

Key takeaway

All four automations solve the same problem: they remove the manual transfer of data from one place to another. That is where time is lost and errors appear most often.

The same techniques — for school and a family business

Case 1: Homework planner

RUNNING

Problem: it is hard for a fifth-grader to organise his own day, combining afternoon-shift school (5 days a week) with football training (4 days a week). Fitting in homework so everything is done before bed and before the next school day is a task in itself.

Solution: the child photographs the assignments and sends them to the bot. Claude recognises the subjects, and the scenario places them in the calendar: heavy ones in the morning, light ones the evening before, in the gaps between school and training.

Tools: Telegram · Make · Claude · Google Calendar

BEFORE → AFTER

PLANNING

BEFORE: by hand every evening

AFTER: from a single photo

TIME

BEFORE: “I’ll do it later”

AFTER: specific slots in the calendar

ROUTINE

BEFORE: easy to break

AFTER: school, football and homework in one place

Case 2: Parent-chat agent

RUNNING

Problem: school and football Viber chats bring hundreds of messages. The important things — games, meet-ups, changes — are easy to miss.

Solution: Viber notifications on the phone are captured and sent to Make. Claude picks out what matters: games with meet-up and start times go straight into the child’s calendar, and the key points from school chats arrive in Telegram as one summary of what needs attention.

BEFORE → AFTER

READING

BEFORE: every chat, end to end

AFTER: only what matters, in a summary

GAMES

BEFORE: copied into the calendar by hand

AFTER: event created automatically

HISTORY

BEFORE: lost in the feed

AFTER: a log in a spreadsheet

Tools: Viber · MacroDroid · Make · Claude · Google Calendar · Sheets · Telegram

Case 3: Bookkeeping bot for a tourism sole proprietor

PARTIAL

Problem: client invoices, their numbering and bookkeeping are handled by hand, in different spreadsheets.

Solution: by voice or text: “invoice so-and-so for such-and-such amount”. Whisper turns the voice into text, Claude structures the data and translates it into Ukrainian. The bot creates a PDF invoice with the next number and records it in the summary spreadsheet.

BEFORE → AFTER

INVOICE

BEFORE: template and number by hand

AFTER: by voice — a numbered PDF

RECORDS

BEFORE: several spreadsheets

AFTER: one summary spreadsheet

NEXT

BEFORE: —

AFTER: supplier invoices and bank statements

Tools: Telegram · Make · Whisper · Claude · Google Docs · Sheets · Drive

Other

03

Small helpers for every day

Case: Article summaries

RUNNING

Problem: there is no time to read long articles and news in full.

Solution: a link is sent to a Telegram bot. The scenario downloads the page, cleans up the text, and an AI model returns a short summary right in the chat.

BEFORE → AFTER

READING

BEFORE: the whole article

AFTER: a summary in the chat

DECISION

BEFORE: “I’ll read it later”

AFTER: instantly clear whether it's worth reading

Tools: Telegram · Make · OpenAI

What is being prepared and discussed

Client bankruptcy monitoring WORK

UNDER DISCUSSION

Problem: we learn about a client's bankruptcy case late, when the debt is already hard to recover.

Idea: regular checks of clients from Company #1's accounting system via the YouControl API; the result is written to the client card, with an alert on any new notice.

Overseas invoice recognition WORK

PLANNED

Problem: every month, invoices from cloud storage are compiled into a spreadsheet by hand.

Idea: on the 5th of each month the scenario collects invoices from cloud storage, AI reads the details and amounts and builds a summary spreadsheet.

Carb and calorie counting bot FAMILY

IN DEVELOPMENT

Problem: counting carbs and calories for every meal by hand is slow and imprecise.

Idea: the bot estimates portion weight, carbs and calories from a photo of the meal and keeps a food diary; entries can also be added by voice.

Candidate sourcing for vacancies WORK

ON HOLD

Problem: applications are scattered across several job sites.

Idea: a single summary of candidates from several platforms for a specific vacancy.

Training report OTHER

IN DEVELOPMENT

Problem: there is training data, but no conclusions drawn from it.

Idea: once a week, Strava data plus sleep and heart-rate data from Garmin go to Claude; a progress report and next week's plan arrive in Telegram.

Progress on these tasks — in future issues.

4 techniques that recur in every case

1

Document → data → document

A photo, scan or file is read by AI and dropped into a ready template.

Powers of attorney, CMRs, invoices

2

Message stream → summary

A transcript, chat or article becomes a short summary with tasks.

Meetings, Viber chats, articles

3

Voice → bookkeeping entry

Say a phrase — get a row in a spreadsheet or a ready invoice.

Business invoices, bookkeeping entries

4

Data → calendar and tasks

AI arranges things by time and by the rules you set.

Homework, games, action items

What to keep in mind

Start small: one narrow task that repeats every day or week.

Add error handling: Make switches a scenario off after a failure, and you only notice when results stop arriving.

Lock down bots with a user whitelist, especially if they handle documents.

Keep a human in the loop where mistakes are costly: AI is rarely wrong, but when it is, it is confidently wrong.

Checklist: is your task ready for automation?

- ☐ The task repeats every day or every week.
- ☐ The data arrives in a clear form: a document, email, message or file.
- ☐ The rules can be put into words: “if this — then do that”.
- ☐ It is clear where the result should go: a spreadsheet, calendar or mailbox.
- ☐ It is clear who checks the result if a mistake is costly.

Glossary

TERMS

No-code — building automations from ready-made blocks, without programming.

Scenario — a chain of steps in Make: “photo arrives → read it → fill the template → send”.

Trigger — the event that starts a scenario: a message to a bot, the end of a meeting, a schedule.

LLM — large language model (Claude, GPT): reads and writes text and understands photos of documents.

Recognition (OCR) — turning a photo or scan into data you can work with.

Whisper — a model that turns a voice message into text.

Whitelist — the list of users a bot responds to. Everyone else is refused.

Human-in-the-loop — a person checks the AI's result where a mistake is costly.

TOOLS IN THIS DIGEST

Platform: Make.com

AI models: Claude (Anthropic), OpenAI, Whisper

Interface: Telegram bots

Data and documents: Google Docs, Sheets, Drive, Calendar, Gmail

Sources: Sembly, Viber, Strava

Building scenarios: Claude Code

Digest prepared by:

MP@MYKOLAI.COM.UA

Let's talk

Have a routine task you'd like to hand off? Write to me — we'll work out how to automate it.