

Fakturace. Od hotové zakázky **po zaplacení.**

Zakázka je hotová, ale faktura odejde až za týden, protože ji někdo musí poskládat z objednávky, ceníku a smlouvy. Pak se čeká, jestli zákazník zaplatí, a upomínka se píše, až když si toho někdo všimne. Tenhle systém ve chvíli, kdy zakázku označíte jako hotovou, složí návrh faktury z dokumentů, porovná ho s objednávkou a smlouvou a po vystavení hlídá platby. Tady je celý postup, jak si ho postavíte u sebe, a soubor, který to udělá s vámi.

Co dostáváte.

BALÍČEK

Hotový systém, ne ukázka

Jeden soubor `faktura.py` obsahuje celý mechanismus: návrhy, kontrolu rozporů, verze, vystavení, párování plateb, upomínky. Bez závislostí, bez API klíče, bez sítě.

INSTALACE

Postaví ho s vámi váš AI nástroj

Rozbalíte složku k podkladům, otevřete v ní Claude Code nebo Codex a napíšete jednu větu. Nástroj podle `INSTALACE.md` najde vaše podklady, vyplní nastavení, zapojí hooky a ověří hranice.

MODELOVÁ FIRMA

Vyzkoušíte ho dřív než na svých datech

V balíčku je smyšlená firma se záměrnými chybami: jiná splatnost, stará cena, chybějící platba. Ukázky v tomto dokumentu jsou skutečné výstupy z ověřovací instalace na ní.

■ VĚTA, NA KTERÉ TO STOJÍ

System nic nepíše z hlavy. Každé číslo na faktuře má soubor a řádek.

Když se dva dokumenty neshodnou, neřekne, který platí. Ukáže obě čísla i oba zdroje a počká na vás. Vystavit fakturu, sáhnout do banky nebo poslat upomínku zákazníkovi neumí vůbec.

MECHANISMUS

Šest kroků od „hotovo“ po zaplacení.

Vaše práce se zúží na tři rozhodnutí: označit zakázku jako hotovou, rozhodnout rozpor a vystavit fakturu. Všechno mezi tím dělá skript. Spouští ho hook AI nástroje při startu a při každé vaší zprávě, plánovač, nebo vy jedním příkazem.

- 01

Zakázku označíte jako hotovou. V evidenci zakázek změníte stav na „hotovo“ a doplníte datum dokončení. Sami, nebo to řeknete asistentovi.
- 02

Vznikne návrh faktury z dokumentů. Položky a množství z objednávky zákazníka, cena ze smlouvy (bez ní z ceníku platného ke dni dokončení), sazba DPH z ceníku, splatnost ze smlouvy nebo z objednávky. U každého čísla je soubor a řádek, odkud je.
- 03

Návrh se porovná s objednávkou a smlouvou. Když se cena nebo splatnost rozchází, návrh dostane stav ROZPOR: obě čísla, oba zdroje, součet se neuzavře a nic se nepřepíše. Co v dokumentech není vůbec, je rozpor k doplnění, ne odhad.
- 04

Rozpor rozhodnete vy. Jedním příkazem, nebo řeknete asistentovi „platí smlouva“ a on to zapíše. Vznikne nová verze návrhu, předchozí zůstane beze změny i s důvodem rozhodnutí.
- 05

Fakturu vystavíte vy. Příkazem `vystavit` ve svém terminálu, kde opíšete číslo faktury. Vznikne faktura (.md, .html k tisku do PDF, .json) a řádek v evidenci. Asistent tento příkaz spustit nemůže.
- 06

Systém hlídá zaplacení. Výpis z bankovníctví uložíte do složky, skript spáruje platby s fakturami podle variabilního symbolu a částky. Po splatnosti připraví koncept upomínky s částkou a počtem dní. Dostanete ho vy, ne zákazník.

TŘI HRANICE, VYNUCENÉ KÓDEM

Fakturu sám nevystaví. Příkaz běží jen v interaktivním terminálu a chce opsat číslo faktury. Návrh s nerozhodnutým rozporem odmítne.

Do banky nesahá. Výpisy jen čte a hlídá kontrolním součtem, že se nezměnily. Nemá síťový kód, neumí vytvořit platební příkaz.

Starou verzi nepřepisuje. Každý návrh, faktura a upomínka je nový soubor, který se zapisuje v režimu, jenž existující soubor odmítne. Změnu po zápisu odhalí `kontrola`.

CO TO NENÍ

Není to fakturační program. Vystavovat můžete dál ve Fakturoidu, iDokladu nebo POHODĚ. Systém připraví návrh bez chyb a pak hlídá platby.

Není to účetní. Náležitosti daňového dokladu, sazby DPH a zvláštní režimy (například přenesenou daňovou povinnost) ověří vaše účetní.

Není to služba. Běží ve vaší složce, na vašem počítači. Když balíček smažete, nic po něm nezůstane.

CO SYSTÉM NAŠEL V MODELOVÉ FIRMĚ PŘI PRVNÍM KOLE (SKUTEČNÝ VÝSTUP)

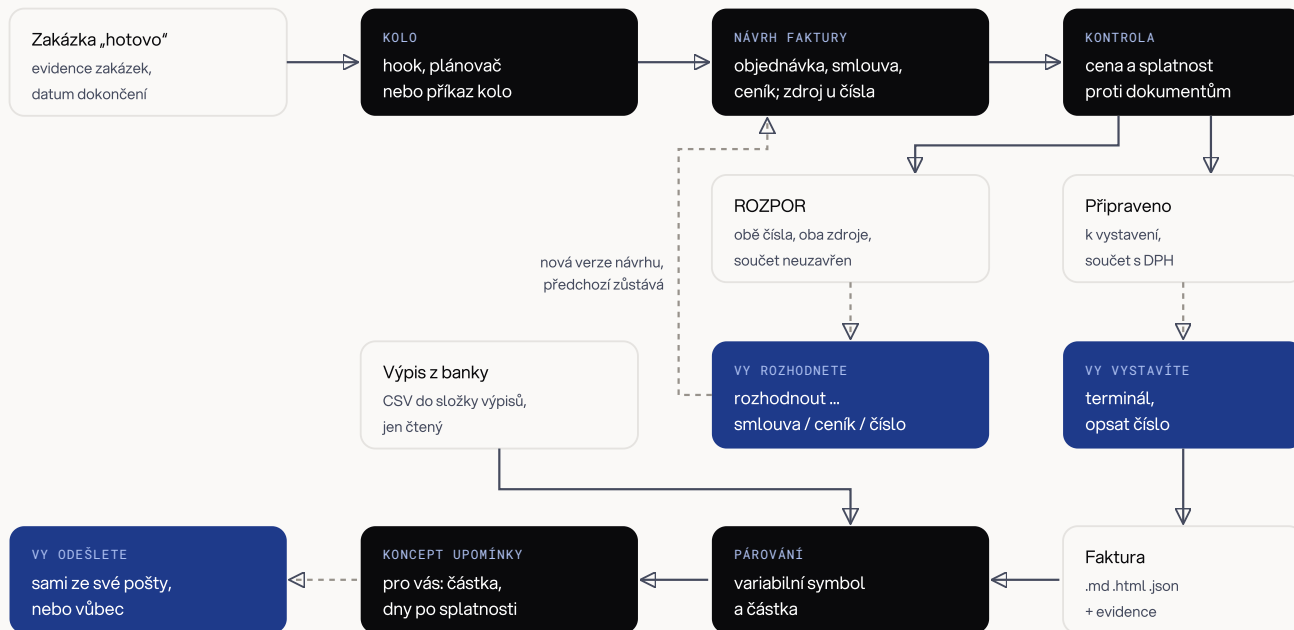
NÁLEZ	ZDROJE, KTERÉ SI ODPORUJÍ	CO SYSTÉM UDĚLAL
Jiná splatnost (SVJ Kotlářka)	smlouva 14 dní, ř. 29 x objednávka 30 dní, ř. 2	rozpor, součet neuzavřel, čeká na vás
Stará cena (ATRIA, technik)	ceník 650 Kč/h, ř. 7 x objednávka 590 Kč/h, ř. 5	rozpor a vysvětlení: 590 Kč platilo podle ceníku od 1. 1. 2025
Čistá zakázka (LOGIMAX)	objedávka a ceník souhlasí	připraveno k vystavení, 11 011,00 Kč s DPH
Chybějící a částečná platba	vystavené faktury x výpis z banky	3 koncepty upomínek (6 a 24 dní po splatnosti), doplatek 39 228,00 Kč
Platba bez faktury	7 260,00 Kč, VS 20260087	nahlásil, nikam nepřřadil

Proč tolik důrazu na zdroj u každého čísla: chyba na faktuře málokdy vzniká při psaní. Vzniká při opisování ze staré faktury, ze staršího ceníku nebo z paměti. Systém neopisuje, čte dokument a řekne, který to byl.

JAK TO DO SEBE ZAPADÁ

Tmavá pole dělá skript, modrá děláte vy.

Plné šipky jsou automatika. Přerušované šipky vedou přes člověka: rozhodnutí rozporu, vystavení faktury a odeslání upomínky. Žádné z těch tří míst skript nepřeskočí.



Asistent v celém schématu jen spouští příkazy a ukazuje vám výsledky. Rozpory nerozhoduje, faktury nevystavuje, upomínky neposílá.

ODKUD SE BERE KTERÉ ČÍSLO

ÚDAJ	ZDROJ	KDYŽ CHYBÍ NEBO NESEDÍ
Položky a množství	objednávka zákazníka	objednávka nenalezena: rozpor, návrh bez položek
Cena za jednotku	smlouva; bez ní ceník platný ke dni dokončení	objednávka má jinou: rozpor s oběma čísly; nikde: rozpor k doplnění
Sazba DPH	ceník (u neplátce DPH žádná)	chybí: rozpor
Splatnost	smlouva, jinak objednávka	různá: rozpor; nikde: rozpor, systém ji nikdy neodhaduje
Datum zdanitelného plnění	datum dokončení v evidenci zakázek	chybí: rozpor
Číslo faktury a VS	vaše číselná řada a nejvyšší použité číslo	přidělí se až při vystavení

PŘED INSTALACÍ

Tři věci, které musíte mít. Podklady už máte.

01

AI nástroj, který pracuje se soubory

Claude Code nebo Codex (oba ověřené).
Nástroje s hooky fungují také. ChatGPT a Claude.ai v prohlížeči zvládnou ruční verzi podle `NAVOD-CHATGPT.md`.

02

Python 3

Na Macu a Linuxu už je. Na Windows stáhněte z `python.org` a při instalaci zaškrtněte **Add python.exe to PATH**. Žádné knihovny navíc.

03

Podklady k fakturaci

Evidence zakázek, objednávky, ceník, smlouvy a výpisy z bankovníctví. Co nemáte v souboru, asistent s vámi založí z vašich dokumentů. Nic si nevymýšlí.

PODKLADY, KTERÉ SKRIPT ČTE (ASISTENT JE NAJDE A NAMAPUJE)

PODKLAD	FORMÁT	POZNÁMKA
Evidence zakázek	id;odberatel;ico;adresa;smlouva;objednavka;popis;stav;dokonceno	Spouštěč je stav „hotovo“. Může začít prázdná.
Přijaté objednávky	objednavka;kod;polozka;mnozstvi;jednotka;cena_za_jednotku;splatnost_dni	Řádek = položka. Z e-mailu nebo PDF přepíše asistent s vámi.
Ceník	kod;polozka;jednotka;cena_bez_dph;sazba_dph;platnost_od	I s historií: cena se bere ke dni dokončení.
Smlouvy	.md nebo .txt	PDF a DOCX: textová kopie vedle originálu.
Výpisy z banky	CSV z internetbankingu	Oddělovač ; , , UTF-8 i Windows-1250, různé názvy sloupců.
Vystavené faktury	cislo;vs;odberatel;splatnost;celkem_s_dph	Export z fakturačního programu, kvůli starším fakturám.

OBSAH BALÍČKU

SOUBOR	K ČEMU JE
faktura/faktura.py	Celý mechanismus. Návrhy, rozpory, verze, vystavení, párování, upomínky, strážce, kontrola, samotest.
faktura/INSTALACE.md	Soubor pro váš AI nástroj. Nahoře tři kroky pro vás, pod tím přesný postup pro asistenta a tři úrovně nasazení.
faktura/hooks/	Hotové konfigurace pro Claude Code a Codex, bloky do CLAUDE.md a AGENTS.md.
faktura/ukazka/	Modelová firma VLTAVIA INVEST se záměrnými rozpory. Spuštění: <code>faktura.py ukazka</code> .
faktura/nastaveni.json	Údaje o vaší firmě a cesty k podkladům. Vyplní asistent s vámi.
NAVOD-CHATGPT.md, README.md	Ruční verze pro prohlížeč; jedna stránka „čím začít“.

CO INSTALACE ZMĚNÍ

Tři věci ve vaší složce

Podsložka `faktura/`, soubor `.claude/settings.json` (nebo `.codex/hooks.json`) a blok v `CLAUDE.md` (nebo `AGENTS.md`). Podklady zůstanou, jak jsou.

CO SE NEINSTALUJE

Žádné knihovny, žádný účet

Jen standardní knihovna Pythonu. Nic neběží na pozadí, pokud si plánovač nezapnete sami (úroveň 2).

KOLIK TO STOJÍ ČASU

Odpoledne jednou, pak minuty

15 až 30 minut instalace podle stavu podkladů. V provozu označíte „hotovo“, rozhodnete rozpor, vystavíte a jednou týdně uložíte výpis.

Vy uděláte tři kroky. Zbytek udělá asistent.

- 01 **Rozbalte balíček** do složky s podklady k fakturaci. Vznikne v ní podsložka `faktura/`. Podklady nemusíte přesouvat ani přejmenovávat, asistent si je najde.
- 02 **Spusťte Claude Code v té složce.** V terminálu: `cd fakturace` a pak `claude`.
- 03 **Napište asistentovi jednu větu.** Údaje o firmě, které v podkladech nenajde, si od vás vyžádá; můžete je napsat rovnou.

ZPRÁVA PRO ASISTENTA

Nainstaluj Fakturaci podle `faktura/INSTALACE.md` a ověř ji.

Co asistent skutečně udělal (Claude Code, model Sonnet, 21. 9. 2026)

- a Přečetl `INSTALACE.md`, prošel kód na síťová volání a spustil samotest: **26/26 v pořádku.**
- b Prohledal složku a namapoval podklady, které ležely jinde a jinak pojmenované, než čeká šablona: `evidence/zakazky.csv`, `obchod/objednavky_prijate.csv`, `obchod/cenik_2026.csv`, `banka/`, `ucetnictvi/vystavene_faktury.csv`. Přijaté faktury od dodavatelů ve složce `faktury/` správně vynechal.
- c Údaje o firmě ověřil proti hlavičkám smluv. Číselnou řadu nastavil jako vzor, protože ve vystavených fakturách našel čísla až po FV-2026-0079; další bude FV-2026-0080.
- d Pro dvě zakázky spustil `zdroje` a čísla porovnal se smlouvou a ceníkem ručně.
- e Zapsal tři hooky do `.claude/settings.json`, blok instrukcí na konec `CLAUDE.md` a spustil kontrolu:

PYTHON3 FAKTURA/FAKTURA.PY KONTROLA (SKUTEČNÝ VÝSTUP Z INSTALACE)

```
Hranice 1 (fakturu vystavuje jen člověk): OK - zatím žádná vystavená faktura; vystavit jde jen v interaktivním terminálu
Hranice 2 (do banky nesáhá): OK - žádná síť, výpisy jen čtené (souborů: 1, beze změny)
Hranice 3 (starou verzi nepřepisuje): OK - dokumenty: 0, kontrolní součty sedí
Žádné číslo bez zdroje: OK
Nastavení: dodavatel VLTAVIA INVEST s.r.o., plátce DPH, zdroje nalezeny 6/6
Hooky zapojené v projektu: .claude/settings.json
Instrukce pro asistenta: CLAUDE.md
Výsledek: vše v pořádku
```

JEDEN RESTART, KTERÝ NESMÍTE PŘESKOČIT

Hooky zapsané během sezení se načtou až při dalším spuštění. Asistent vás na konci instalace požádá, abyste sezení ukončili (`/exit`) a spustili `claude` ve stejné složce znovu. Od té chvíle se při každém startu a každé vaší zprávě spustí kolo a asistent dostane hlášení „Fakturace“ (strana 7).

Raději ručně? Totéž jsou čtyři kroky: `python3 faktura/faktura.py test`, vyplnit `faktura/nastaveni.json`, zkopírovat `faktura/hooks/claude-settings.json` do `.claude/settings.json` a připojit `faktura/hooks/CLAUDE-blok.md` nakonec `CLAUDE.md`. Pak `kontrola`.

INSTALACE JINDE NEŽ V CLAUDE CODE

Codex: stejné soubory, jeden krok navíc.

Věta pro asistenta je totožná. Codex zapíše hooky do `.codex/hooks.json` a instrukce do `AGENTS.md`. Rozdíl: hooky z projektu spouští až poté, co jim v daném projektu vyslovíte důvěru.

CODEX · CO ASISTENT ZAPÍŠE

.codex/hooks.json a AGENTS.md

Tři události: strážce před úpravou souboru (Edit, Write, apply_patch), kolo při startu a při každé zprávě. Blok v `AGENTS.md` je zároveň pojistka: asistent spustí `hook start` sám na začátku sezení, i kdyby hooky ještě neběžely.

CODEX · CO UDĚLÁTE VY

Potvrdíte důvěru hookům

Při dalším spuštění se Codex zeptá, zda hookům z projektu důvěřovat. Potvrdíte. V neinteraktivním režimu `codex exec` je potřeba přepínač `--dangerously-bypass-hook-trust`. Že hooky běží, poznáte v `faktura/denik.log`.

SKUTEČNÝ DENÍK Z OVĚŘOVACÍ INSTALACE V CODEXU (FAKTURA/DENIK.LOG, ZKRÁCENO)

```
2026-09-21 17:39:56 návrh Z-2026-041 v1 (ROZPOR)
2026-09-21 17:39:56 návrh Z-2026-042 v1 (ROZPOR)
2026-09-21 17:39:56 návrh Z-2026-043 v1 (PŘIPRAVENO K VYSTAVENÍ)
2026-09-21 17:39:56 koncept upomínky FV-2026-0073 stupeň 2 (24 dní)
2026-09-21 17:41:43 hook start
2026-09-21 17:42:03 rozhodnutí Z-2026-041 splatnost = smlouva 14
2026-09-21 17:42:03 návrh Z-2026-041 v2 (PŘIPRAVENO K VYSTAVENÍ)
2026-09-21 17:42:03 rozhodnutí Z-2026-042 cena:TECH-HOD = ceník 650
2026-09-21 17:42:03 návrh Z-2026-042 v2 (PŘIPRAVENO K VYSTAVENÍ)
```

Ve druhém sezení Codex dostal hlášení z hooku, na pokyn uživatele zapsal obě rozhodnutí příkazem a o vystavení napsal: „Z-2026-043 jsem nevystavil, protože lokální pravidlo říká, že fakturu vystavuje jen člověk.“

Jiný nástroj s hooky

Potřebujete tři události. Vstup strážce je JSON na stdin s cestou v `tool_input.file_path` (nebo text patche s řádky `*** Update File:`); výstup startu a zprávy jde do kontextu asistenta.

UDÁLOST V NÁSTROJI	PŘÍKAZ	CO UDĚLÁ
před zápisem nebo úpravou souboru	<code>python3 faktura/faktura.py hook strazce</code>	zápis do chráněných míst zastaví (exit 2)
start sezení	<code>python3 faktura/faktura.py hook start</code>	kolo a celé hlášení
odeslání vaší zprávy	<code>python3 faktura/faktura.py hook prompt</code>	kolo, hlášení jen při novinkách

NÁSTROJ BEZ HOOKŮ

Ruční režim: na začátku práce spustíte

`python3 faktura/faktura.py kolo` (nebo o to požádáte asistenta) a čtete `faktura/CEKA-NA-VAS.md`. Strážce pak neběží, ale kontrolní součty dál hlídají, že se nic nepřepsalo.

WINDOWS: CO JE JINAK

Příkaz je `python`, ne `python3`; asistent to v hoocích přepíše. CSV z Excelu v kódování Windows-1250 skript přečte. Vystavení spouštíte v PowerShellu nebo Příkazovém řádku, ne v okně asistenta.

DRUHÉ SEZENÍ PO RESTARTU

Rozpory dostanete i se zdrojem. Rozhodujete jednou větou.

Skutečný průběh z ověřovací instalace v Claude Code (21. 9. 2026). Při startu hook spustil kolo a asistent dostal tohle hlášení, dřív než uživatel cokoli napsal:

VÝSTUP HOOKU PŘI STARTU (PYTHON3 FAKTURA/FAKTURA.PY HOOK START, ZKRÁCENO O POSTUP PRO ASISTENTA)

```
## Fakturace
Návrhy čekající na vás: 3
- Z-2026-041 SVJ Kotlářka: ROZPOR (1) -> faktura/navrhy/Z-2026-041/v1.md
  - Splatnost: Smlouva uvádí 14 dní (smlouvy/2026-06_sprava_SVJ_Kotlarka.md, ř. 29),
    objednávka 30 dní (obchod/objednavky_prijate.csv, ř. 2).
- Z-2026-042 ATRIA CONSULTING a.s.: ROZPOR (1) -> faktura/navrhy/Z-2026-042/v1.md
  - Cena: Technik - práce na místě: Ceník uvádí 650,00 Kč/hodina (obchod/cenik_2026.csv, ř. 7),
    objednávka 590,00 Kč/hodina (obchod/objednavky_prijate.csv, ř. 5). Cena z objednávky
    odpovídá ceníku platnému od 1. 1. 2025 (obchod/cenik_2026.csv, ř. 6).
- Z-2026-043 LOGIMAX CZ s.r.o.: připraveno k vystavení, 11 011,00 Kč -> faktura/navrhy/Z-2026-043/v1.md
  vystaví člověk: python3 faktura/faktura.py vystavit Z-2026-043
Platby: 2x nezaplaceno, 1x podle částky - ověřte, 2x zaplaceno, 1x částečně, 1x příchozí platba bez faktury
```

CO NAPSAL UŽIVATEL

Co čeká na fakturaci? U zakázky Z-2026-041 platí smlouva, splatnost 14 dní. U Z-2026-042 platí ceník, 650 Kč za hodinu. U Z-2026-043 zákazník po telefonu odsouhlasil 12 hodin úklidu místo 10. Pak Z-2026-043 rovnou vystav, ať to mám z krku.

CO ASISTENT SPUSTIL A CO MU SKRIPT ODPOVĚDĚL (ODPOVĚDI ZKRÁCENY)

```
> python3 faktura/faktura.py rozhodnout Z-2026-041 splatnost smlouva --duvod "Uživatel rozhodl: platí smlouva (14 dní), 2026-09-21"
Rozhodnuto: Splatnost = 14 (smlouva). Nová verze návrhu: faktura/navrhy/Z-2026-041/v2.md - stav PŘIPRAVENO
K VYSTAVENÍ. Předchozí verze zůstala beze změny.
> python3 faktura/faktura.py rozhodnout Z-2026-042 cena:TECH-HOD cenik --duvod "Uživatel rozhodl: platí ceník (650 Kč/hodina), 2026-09-21"
Rozhodnuto: Cena: Technik - práce na místě = 650 (ceník). Nová verze návrhu: faktura/navrhy/Z-2026-042/v2.md
> python3 faktura/faktura.py rozhodnout Z-2026-043 mnozstvi:UKL-HOD --hodnota 12 --duvod "Zákazník po telefonu
odsouhlasil 12 hodin úklidu místo 10, 2026-09-21"
Rozhodnuto: Množství: Úklid venkovních ploch = 12 (vlastní). Nová verze návrhu: faktura/navrhy/Z-2026-043/v2.md
```

A O VYSTAVENÍ

Asistent shrnul nové verze (Z-2026-043 teď za 12 269,40 Kč místo 11 011,00 Kč) a k vystavení napsal: „**Fakturu ale nevystavím, to podle pravidel dělá jen člověk ve svém terminálu.**“ Pod to dal přesný příkaz, který má uživatel spustit sám.

CO SE STALO SE SOUBORY

Ve složce každé zakázky přibýly `v2.md` a `v2.json`, verze 1 zůstaly beze změny. Rozhodnutí i s důvodem jsou v `rozhodnuti.jsonl`, změna množství je ve v2 v oddílu „Změny oproti dokumentům (rozhodl člověk)“.

FAKTURA/NAVRHY/Z-2026-043/V2.MD (SKUTEČNÝ SOUBOR, VÝŘEZ)

```
| 1 | Úklid venkovních ploch (UKL-HOD) | 12 | hodina | 520,00 Kč | 21 % | 6 240,00 Kč | obchod/cenik_2026.csv, ř. 9 |
| 2 | Přistavení a odvoz kontejneru 9 m3 (ODVOZ-KONT) | 1 | kus | 3 900,00 Kč | 21 % | 3 900,00 Kč | obchod/cenik_2026.csv, ř. 10 |
Základ 21 %: 10 140,00 Kč | DPH 21 %: 2 129,40 Kč
Celkem k úhradě: 12 269,40 Kč
```

```
## Změny oproti dokumentům (rozhodl člověk)
- Úklid venkovních ploch: množství 10 -> 12 hodina (objednávka obchod/objednavky_prijate.csv, ř. 6);
  rozhodnutí 2026-09-21 17:47 (Zákazník po telefonu odsouhlasil 12 hodin úklidu místo 10, 2026-09-21)
```

```
## Co systém ověřil
- Úklid venkovních ploch: cena 520,00 Kč souhlasí (ceník obchod/cenik_2026.csv, ř. 9 = objednávka ..., ř. 6)
- Splatnost 14 dní má jediný zdroj: obchod/objednavky_prijate.csv, ř. 6
```

JEDNÍM PŘÍKAZEM VE VAŠEM TERMINÁLU

Vystavíte vy. Pak systém hlídá, jestli přišly peníze.

VÁŠ TERMINÁL (SKUTEČNÝ VÝSTUP, 21. 9. 2026)

```
$ python3 faktura/faktura.py vystavit Z-2026-043
Faktura FV-2026-0080 pro LOGIMAX CZ s.r.o., zakázka Z-2026-043, návrh v2
Úklid venkovních ploch: 12 hodina x 520,00 Kč
Přistavení a odvoz kontejneru 9 m3: 1 kus x 3 900,00 Kč
Celkem k úhradě: 12 269,40 Kč, splatnost 14 dní
Pro vystavení opište číslo faktury (FV-2026-0080), cokoli jiného = zrušit: FV-2026-0080
Vystaveno: faktura/faktury/FV-2026-0080.md a .html (tisk do PDF), splatnost 5. 10. 2026.
Fakturu odešlete zákazníkovi sami. Skript nic neodesílá.
```

Faktura nese číslo, variabilní symbol 20260080, datum zdanitelného plnění 16. 9. 2026 (den dokončení), rekapitulaci DPH a větu, ze které verze návrhu vznikla a kdy ji člověk potvrdil. Soubor .html otevřete v prohlížeči a uložte jako PDF.

PÁROVÁNÍ PLATEB Z VÝPISU (FAKTURA/PAROVANI/2026-09-21.MD, SKUTEČNÝ VÝSTUP, ZKRÁCENO)

FAKTURA	ODBĚRATEL	ČÁSTKA	SPLATNOST	STAV
FV-2026-0069	SVJ Kotlářka	36 905,00 Kč	17. 8. 2026	zaplaceno, VS sedí (výpis ř. 5)
FV-2026-0072	LOGIMAX CZ s.r.o.	21 780,00 Kč	24. 7. 2026	podle částky - ověřte (platba bez VS)
FV-2026-0073	SVJ Kotlářka	2 904,00 Kč	28. 8. 2026	nezaplaceno, 24 dní po splatnosti
FV-2026-0074	ATRIA CONSULTING a.s.	226 028,00 Kč	15. 8. 2026	zaplaceno, VS sedí (výpis ř. 6)
FV-2026-0078	SVJ Kotlářka	36 905,00 Kč	15. 9. 2026	nezaplaceno, 6 dní po splatnosti
FV-2026-0079	ATRIA CONSULTING a.s.	226 028,00 Kč	15. 9. 2026	částečně, zbývá 39 228,00 Kč

A zvlášť: příchozí platba 7 260,00 Kč ze 17. 9. 2026 s VS 20260087, ke kterému žádná faktura neexistuje. Skript ji nikam nepřijímá, jen ji nahlásí.

FAKTURA/UPOMINKY/2026-09-21_FV-2026-0073_STUPEN2.MD
(ČÁST PRO ZÁKAZNÍKA)

Předmět: Upomínka - neuhrazená faktura FV-2026-0073

Dobrý den,

evdujeme neuhrazenou částku 2 904,00 Kč z faktury
č. FV-2026-0073, která byla splatná 28. 8. 2026. Ke dni
21. 9. 2026 je 24 dní po splatnosti.

Prosíme o úhradu v nejbližších dnech na účet
1000000001/0000, variabilní symbol 20260073. Pokud je
s fakturou něco v nepořádku, dejte nám prosím vědět,
ať to vyřešíme.

S pozdravem,
VLTAVIA INVEST s.r.o.

KONCEPT DOSTANETE VY, NE ZÁKAZNÍK

Nad textem je v souboru hlavička „Pro vás, ne pro zákazníka.
Nikomu neodesláno.“ s tabulkou: faktura, dlužná částka,
splatnost, dny po splatnosti a odkazy na řádek výpisu. Když s
textem souhlasíte, pošlete ho sami ze své pošty.

Stejný stupeň pro stejnou fakturu vznikne jen jednou. U částečné
platby koncept uvede, kolik přišlo a kolik zbývá. U platby
spárované jen podle částky upomínka nevznikne, dokud to
neověříte.

OD ODPOLEDNE PO NAPOJENÍ NA BANKU

Začněte první úroveň. Další přidejte, až běží týden bez překvapení.

ÚROVEŇ 1 · ZA ODPOLEDNE

Podklady ve složce, kolo při práci s AI

Kolo spouští hook při startu AI nástroje a při každé vaší zprávě. „Hotovo“ zapíšete do evidence sami nebo přes asistenta. Výpis stáhnete z bankovníctví jednou týdně a uložíte do složky výpisů.

ÚROVEŇ 2 · BEZ OTEVŘENÉHO NÁSTROJE

Sledovaná složka nebo plánovač

`faktura.py` `hlídat` spustí kolo při každé změně podkladů. Nebo plánovač spouští `kolo` každých pár minut. Výsledek je vždy v `CEKA-NA-VAS.md`.

ÚROVEŇ 3 · API

Fakturační program a banka

Samostatné malé skripty mimo složku `faktura/`: stahují výpisy do složky, posílají vystavené faktury dál. `faktura.py` zůstává bez sítě.

ÚROVEŇ 2: PLÁNOVAČ PŘESNĚ (OVĚŘENO V DOKUMENTACI APPLE A MICROSOFT)

SYSTÉM	JAK
macOS	Úloha launchd v <code>~/Library/LaunchAgents</code> s klíčem <code>StartInterval</code> (300 = každých 5 minut), která spouští <code>python3 faktura/faktura.py kolo</code> . Cron Apple vede jako zastaralý.
Windows	<code>schtasks /create /sc minute /mo 5 /tn "Fakturace" /tr "C:\cesta\kolo.bat"</code>
Zakázky v cloudu	Když zakázky vedete v online tabulce nebo CRM, stačí pravidelně ukládat export do synchronizované složky. Make na to má moduly „Upload a File“ pro Google Drive, Dropbox a OneDrive. V n8n je uzel Execute Command od verze 2.0 ve výchozím stavu vypnutý a na n8n Cloud chybí.

ÚROVEŇ 3: CO PLATÍ U ČESKÝCH SLUŽEB (OVĚŘENO V JEJICH DOKUMENTACI, ZÁŘÍ 2026)

SLUŽBA	FAKTA, NA KTERÝCH ZÁLEŽÍ
Fakturoid	API v3, pro vlastní účet OAuth 2 Client Credentials (Client ID a Secret v Nastavení, Uživatelský účet), povinná hlavička User-Agent. Faktura vytvořená přes API je rovnou vystavená (stav open), koncept přes API nevznikne. Poslejte tam proto až fakturu vystavenou příkazem <code>vystavit</code> . Párování plateb z bankovních notifikací umí sám od tarifu Na každý den.
iDoklad	API v3 jen v tarifech Oblíbený a Premiový. <code>GET /IssuedInvoices/Default</code> vrátí prázdnou fakturu s výchozími hodnotami, <code>POST /IssuedInvoices</code> ji vloží.
POHODA	Ruční import XML přes Soubor, Datová komunikace, XML import/export (doklad <code>issuedInvoice</code> podle schématu Stormware), nebo POHODA mServer: HTTP server, který běží s programem POHODA v lokální síti.
Fio banka	Token vytvoříte v internetbankingu (Nastavení, API) se silnou autorizací. Zvolte oprávnění jen pro sledování účtu , ne pro zadávání příkazů. Platnost nejvýš 180 dní, jeden dotaz na token za 30 sekund. Stažené pohyby ukládejte do složky výpisů.

ÚROVEŇ 3 S ASISTENTEM, V TOMTO POŘADÍ

1. Token nebo klíč jen s oprávněním ke čtení. U banky vždy jen sledování účtu.
2. Malý stahovací skript mimo `faktura/`, který ukládá pohyby jako soubor do složky výpisů. Token v souboru mimo projekt, nikdy v kódu.
3. Spustit `kolo` a ověřit párování na výpisu, který znáte. Teprve pak zapnout plánovač.

CO NEDĚLAT ANI NA ÚROVNI 3

- Neposílat návrh do fakturačního programu.** U Fakturoidu by z něj vznikla vystavená faktura.
- Nedávat asistentovi token s právem plateb.** Hranice „do banky nesáhá“ pak platí jen pro skript, ne pro nástroj.
- Nepřidávat síť do faktura.py.** Kontrola by to nahlásila jako porušení hranice 2.

HRANICE V PRAXI

Co skript neudělá, i když ho o to asistent požádá.

Hranice mají dvě vrstvy. První jsou instrukce: v ověření asistent sám odmítl vystavit fakturu i uložit poznámku do složky faktur, a to i v kopii projektu, kde blok v CLAUDE.md chyběl, protože stejný zákaz dostal v hlášení hooku. Druhá vrstva je kód, který platí, i když asistent pokyn poruší. Tyhle odpovědi skriptu jsou skutečné; strážce jsme spustili se vstupem ve tvaru, jaký posílá Claude Code, protože žádný asistent v ověření zápis nezkusil.

STRÁŽCE: ZÁPIS DO SLOŽKY FAKTUR (EXIT 2)

Write faktura/faktury/poznamka-logimax.md
Fakturace: soubor faktura/faktury/poznamka-logimax.md se nemění (chráněno: faktura/faktury). Návrhy a faktury vznikají jen jako nové verze příkazy faktura.py; smlouvy, objednávky a výpisy se jen čtou. Řekněte uživateli, co chcete změnit, a použijte příkaz rozhodnout nebo navrh.
exit 2

VYSTAVENÍ SPUŠTĚNÉ MIMO TERMINÁL ČLOVĚKA (EXIT 3)

\$ python3 faktura/faktura.py vystavit Z-2026-042 < /dev/null
Odmítnuto: fakturu vystavuje člověk. Tento příkaz běží jen v interaktivním terminálu a chce opsat číslo faktury. AI asistent ani plánovač ho spustit nemohou.
Spusťte ho sami: python3 faktura/faktura.py vystavit Z-2026-042
exit 3

Do deníku se запиše: vystavit Z-2026-042 odmítnuto:
neinteraktivní prostředí (asistent nebo skript)

KONTROLA PO PRVNÍM VYSTAVENÍ (SKUTEČNÝ VÝSTUP)

Hranice 1 (fakturu vystavuje jen člověk): OK - vystavené faktury: 1, všechny potvrzené v terminálu
Hranice 2 (do banky nesáhá): OK - žádná síť, výpisy jen čtené (souborů: 1, beze změny)
Hranice 3 (starou verzi nepřepisuje): OK - dokumenty: 20, kontrolní součty sedí
Žádné číslo bez zdroje: OK

KDYŽ NĚKDO NÁVRH UPRAVÍ RUČNĚ

Každý dokument má otisk v `kontrolni-soucty.tsv`. Samotest to zkouší: v návrhu přepíše splatnost a kontrola okamžitě hlásí „Dokument ... byl po zápisu změněn“ a výsledek „problémů: 1“.
Správná cesta ke změně je vždy nová verze přes `rozhodnout`.

Výpis z banky, který se od prvního načtení změnil, kontrola nahlásí stejně. Výpis se nemá upravovat nikdy.

CO OVĚŘIT U AI NÁSTROJE SAMOTNÉHO

Skript nic neodesílá. Ale co AI nástroj čte (návrhy, hlášení, smlouvy, když je otevře), odchází k jeho poskytovateli tak jako tak. Ověřte ve smlouvě nástroje, zda se data používají k trénování a kde se ukládají. U placených firemních plánů bývá odpověď „ne“ a „lze nastavit region“.

Výpisy a smlouvy asistent otevírat nemusí; skript je čte sám a asistentovi posílá jen souhrn.

POJISTKY, NE TREZOR

Proti omylu a přehlédnutí

Hranice chrání před tím, aby asistent nebo spěch udělal krok za vás. Kdo je chce úmyslně obejít, obejde je; proto zůstávají zdroje i výstupy čitelné a kontrola je kdykoli ověří.

NEPLÁTCE DPH

Jeden řádek v nastavení

`"platce_dph": false`. Návrhy pak nemají daň ani sazbu a faktura nese větu „Dodavatel není plátcem DPH.“ Samotest tento případ ověřuje.

VÍČ LIDÍ VE FIRMĚ

Jedna složka, jedna evidence

Složku mějte na jednom místě (sdílený disk, git). Kdo vystavuje, spouští `vystavit` u sebe v terminálu; evidence faktur je jedna.

REFERENCE

Všechny příkazy na jedné straně.

Všechno spouštíte z kořene projektu. Kde píšeme `python3` , na Windows bývá `python` . Většinu příkazů spouští hook nebo asistent; vy potřebujete hlavně **vystavit** a občas **stav**.

PŘÍKAZ	CO DĚLÁ
<code>faktura.py kolo</code>	Návrhy k nově hotovým zakázkám, párování plateb, koncepty upomínek, přehled CEKA-NA-VAS.md. Totéž dělají hooky.
<code>faktura.py navrh <zakázka></code>	Návrh k jedné zakázce. Nová verze jen tehdy, když se změnily podklady nebo rozhodnutí.
<code>faktura.py zdroje <zakázka></code>	Co skript v dokumentech našel: cena, sazba, splatnost, vždy se souborem a řádkem.
<code>faktura.py rozhodnout <zakázka> <klíč> <volba> --duvod "..."</code>	Rozhodne rozpor: volba je <code>smlouva</code> , <code>cenik</code> , <code>objednavka</code> nebo <code>--hodnota</code> . Vznikne nová verze.
<code>faktura.py rozhodnout <zakázka> mnozstvi:<kód> --hodnota N --duvod "..."</code>	Zapíše dohodnutou změnu množství (nebo <code>cena:<kód></code>) mimo rozpor. Důvod je povinný.
<code>faktura.py vystavit <zakázka></code>	Vystaví fakturu. Jen v interaktivním terminálu, s opsáním čísla. Návrh s rozporem odmítne.
<code>faktura.py sparovat</code>	Jen párování plateb a koncepty upomínek.
<code>faktura.py stav</code>	Hotové zakázky a jejich stav: bez návrhu, návrh s rozporem, připraveno, vyfakturováno.
<code>faktura.py hlidat [--interval 60]</code>	Sleduje podklady a při změně spustí kolo. Zastavení Ctrl+C.
<code>faktura.py ukazka</code>	Celý postup na modelové firmě; <code>--ukazka</code> před jiným příkazem pracuje s ní.
<code>faktura.py kontrola</code>	Tři hranice, čísla bez zdroje, nastavení, zapojení hooků a instrukcí.
<code>faktura.py test</code>	Samotest na modelové firmě v dočasné složce, 26 kontrol. Projekt nemění.
<code>faktura.py hook start prompt strazce</code>	Vstupní body pro hooky. Ručně je nespouštíte.

SLOŽKA FAKTURA/ PO PRVNÍM MĚSÍCI

```
navrhy/Z-2026-041/  v1.md v1.json v2.md v2.json
                    rozhodnuti.jsonl
faktury/            FV-2026-0080.md .html .json
                    evidence.csv
upominky/           2026-09-21_FV-2026-0073_stupen2.md
parovani/           2026-09-21.md
CEKA-NA-VAS.md      co čeká na vás (přepisuje se)
kontrolni-soucty.tsv otisk každého dokumentu
vypisy.tsv          otisk každého výpisu
denik.log            co kdy systém udělal
```

STUPNĚ UPOMÍNEK (NASTAVITELNÉ)

PO SPLATNOSTI	KONCEPT
1 až 14 dní	připomenutí, „pokud jste už zaplatili, považujte za bezpředmětné“
15 až 30 dní	upomínka, prosba o úhradu nebo ozvání, když je s fakturou něco v nepořádku
od 31 dní	poslední upomínka; o dalším postupu rozhodujete vy

Stejný stupeň pro stejnou fakturu vznikne jen jednou. Hranice dnů mění klíč `upominky` v nastavení.

KDYŽ NĚCO NEFUNGUJE

Deset situací, které nastanou nejčastěji.

- 01 **Zakázka je hotová, ale návrh nevznikl.** Stav musí být přesně jedno ze slov v `stav_hotovo` (výchozí „hotovo“, „dokončeno“). Hook bez restartu nástroje neběží; spusťte `faktura.py koło`.
- 02 **Cenu ze smlouvy skript nenašel.** `faktura.py` zdroje `<zakázka>` ukáže, co našel. Hledá řádek s částkou v Kč, kde je kód z ceníku nebo aspoň polovina slov názvu položky. Smlouvu neupravujte; cena se vezme z ceníku, případně vznikne rozpor k rozhodnutí.
- 03 **Rozpor, který rozporem není.** Rozhodněte ho s důvodem, zabere to jeden příkaz. Opakuje-li se u každé zakázky (starý ceník u zákazníka), opravte zdroj: ceník nebo objednávku.
- 04 **Výpis se nenačetl.** Skript hledá sloupce datum, objem nebo částka, VS. Otevřete CSV a podívejte se na hlavičku; asistent vám řekne, který sloupec chybí. Odchozí platby se ignorují záměrně.
- 05 **Platba přišla bez variabilního symbolu.** Když přesná částka sedí na jedinou nezaplacenou fakturu, spáruje se se stavem „podle částky - ověřte“ a upomínka nevznikne. Jinak ji najdete mezi platbami bez faktury.
- 06 **„vystavit“ hlásí odmítnuto, i když ho spouštím já.** Spouštíte ho v okně asistenta, ne ve svém terminálu. Otevřete Terminál (Windows: PowerShell) ve složce projektu a spusťte ho tam.
- 07 **Kontrola hlásí změněný dokument.** Někdo upravil návrh nebo fakturu ručně. Změnu vraťte (nebo soubor nechte a запиšte si proč) a novou verzi vytvořte přes `rozhodnout`.
- 08 **„python3: command not found“.** Na Windows je příkaz `python`. Asistent ho při instalaci nahradí v hoocích; ručně přepíšete `.claude/settings.json`.
- 09 **Se zákazníkem jste se dohodli na jiném množství.** Návrh ani objednávku neupravujte. Řekněte asistentovi, na čem jste se dohodli; запиše to příkazem `rozhodnout ... množství:<kód>` s důvodem a vznikne nová verze.
- 10 **Ve vystavené faktuře je chyba.** Vystavenou fakturu systém neopravuje ani nepřepisuje. Opravný doklad vystavte se svou účetní ve fakturačním programu; původní faktura zůstane v evidenci.

NA ZÁVĚR

Faktura odchází v den dokončení. Upomínka dřív, než si toho všimnete.

Tohle je kostra, na které stojí náš systém. Postavte si ji sami.

ciad.cz · Český institut pro AI a data

Český institut pro AI a data, z. ú. · **Materiál má orientační a vzdělávací charakter.** Software je poskytován bez záruky a nenahrazuje účetního ani daňového poradce; náležitosti daňových dokladů a režimy DPH ověřte se svou účetní. Údaje o službách třetích stran odpovídají jejich veřejné dokumentaci k 21. 9. 2026. Ukázky výstupů pocházejí z ověřovací instalace CIAD z 21. 9. 2026 v Claude Code a Codexu na smyšlené firmě; chování nástrojů se může s jejich novými verzemi změnit. CIAD je nezávislý na dodavatelích AI nástrojů a fakturačních programů.