Sisukord

AI-lahendus Ask Directo: Directo MCP serveri seadistamine - Claude Desktop ja Gemini CLI 3
1. Sissejuhatus - miks MCP? 3
Mis on MCP? 3
Mis kasu see annab? 3
Arhitektuur 3
2. Eelduste kontroll 4
2.1 Kõigile kohustuslikud eeldused 4
2.2 Claude Desktop kasutajale lisaks 4
2.3 Gemini CLI kasutajale lisaks 5
2.4 Admin õigused — millal on vaja? 5
3. Ühised sammud (tee üks kord) 6
Samm 3.1 — Node.js paigaldamine (kohustuslik) 6
Vali oma tee 7
TEE A — Kiirpaigaldus skriptidega (soovitatud) 7
TEE B — Käsitsi, samm-sammult (alternatiiv Tee A-le) 8
Samm B1 — PowerShell execution policy seadistamine 9
Samm B2 — CredentialManager moodul 9
Samm B3 — MCP serveri kausta loomine 10
Samm B4 — directo-mcp.js paigutamine 10
Samm B5 — package.json loomine 11
Samm B6 — Sõltuvuste installimine 11
Samm B7 — API võtme turvaline salvestamine 11
4. Claude Desktop seadistamine 12
Samm 4.1 — Konfiguratsioonifaili asukoht 12
Samm 4.2 — Konfiguratsioonifail 13
Samm 4.3 — Claude Desktop taaskäivitamine 14
Samm 4.4 — Testimine 14
5. Gemini CLI seadistamine 15
Samm 5.1 — Gemini CLI paigaldamine 15
Samm 5.2 — Google API võtme seadistamine 16
Samm 5.3 — Konfiguratsioonifaili asukoht 16
Samm 5.4 — MCP serveri lisamine 16
Samm 5.5 — Kausta usaldamine (Trusted Folders) 17
Samm 5.6 — Testimine 18
6. Tõrkeotsing 18
Üldised vead (mõlemale kliendile) 18
Claude Desktop spetsiifilised vead 19
Gemini CLI spetsiifilised vead 19
7. MCP tööriistad 19
8. Turvalisus 20
9. Viited 20

AI-lahendus Ask Directo: Directo MCP serveri seadistamine - Claude Desktop ja Gemini CLI

See juhend selgitab kuidas ühendada Directo ERP süsteem AI assistendiga (Claude või Gemini) läbi MCP (Model Context Protocol) lokaalse serveri. Juhend on mõeldud nii Claude Desktop kui Gemini CLI kasutajatele - ühised sammud pead tegema üks kord ja seejärel vali oma AI kliendile vastav sektsioon.

1. Sissejuhatus - miks MCP?

Mis on MCP?

MCP (Model Context Protocol) on avatud standard, mis võimaldab AI assistentidel otse suhelda väliste süsteemidega — andmebaasidega, API-dega ja rakendustega.

Ilma MCP-ta töötab AI assistendiga nii:

1. Sina kopeerid andmeid Directost
2. kleebid need AI vestlusesse
3. AI vastab
4. Sina kopeerid vastuse tagasi Directosse

MCP-ga töötab see nii:

1. Sina küsid loomulikus keeles: „Mis artikleid on laos alla miinimumtaseme?“
2. AI küsib ise Directo API-st andmed
3. AI analüüsib ja vastab kohe

Mis kasu see annab?

Olukord	Ilma MCP-ta	MCP-ga
Laoseisu kontroll	Ava Directo → otsi → kopeeri → kleebi AI-sse	„Mis artikleid on laos alla 10 ühiku?“
Arve analüüs	Ekspordi CSV → laadi AI-sse → küsi	„Millised kliendid pole viimase 90 päeva jooksul maksnud?“
Müügiaruanne	Käsitsi koosta → kopeeri → analüüsi	„Võrdle selle kuu müüki eelmise aastaga“
Tellimuste jälgimine	Ava mitu vaadet → kopeeri infot	„Millised tellimused on üle 5 päeva täitmata?“

Arhitektuur

```
Sina (loomulikus keeles)
        ↕
AI klient (Claude Desktop või Gemini CLI)
        ↕  MCP protokoll (lokaalne)
```

```
MCP Server (sinu arvutis, Node.js)
        ↕  HTTPS
Directo API (pilves)
```

MCP server jookseb **sinu arvutis** - Directo andmed ei liigu läbi AI pilveteenuse serveri.



Lokaalne paigaldus - oluline mõista:\

- MCP server paigaldatakse **ainult sinu isiklikku arvutisse**
- See **ei ole jagatud** - teised kolleegid ei pääse sinu MCP serverile ligi
- Iga kasutaja peab **ise** kõik sammud tegema oma arvutis
- MCP server töötab ainult siis, kui **sinu arvuti on sisse lülitatud** ja AI klient on käivitatud
- Kui soovid MCP serverit jagada mitme kasutajaga, on vaja eraldi avaliku serveri lahendust (ei kuulu selle juhendi skoopi)

2. Eelduste kontroll



Kontrolli kõik eeldused enne paigaldamist. Puuduv eeldus on kõige levinum ebaõnnestumise põhjus.

2.1 Kõigile kohustuslikud eeldused

Eeldus	Kontrollimine	Parandus
Windows 10 või 11 (64-bit)	Start → Settings → System → About	—
Node.js 22 või uuem (LTS)	`node –version`	nodejs.org → LTS
npm	`npm –version` (töötab alles pärast execution policy seadistamist, vt allpool)	Tuleb Node.js-ga automaatselt kaasa
PowerShell execution policy	`Get-ExecutionPolicy` → peab vastama RemoteSigned	Vt allpool
CredentialManager moodul	`Get-Module -ListAvailable CredentialManager`	Vt allpool
Directo API võti	Directo → Admin → Seaded → API	Küsi Directo administraatorilt

2.2 Claude Desktop kasutajale lisaks

Eeldus	Kontrollimine	Link
Claude Desktop installitud	Start menüüs on Claude olemas	claude.ai/download
Anthropic konto	Sisselogimine töötab	claude.ai

Ametlik dokumentatsioon: docs.claude.ai

2.3 Gemini CLI kasutajale lisaks



⚠ **Gemini CLI tasuta tier on endiselt olemas, aga muutunud.** Autentimisviisist sõltuvalt:\

- **Google kontoga (Gemini Code Assist üksikkasutajale):** ~60 päringut/min ja ~1000 päringut/päevas\
- **Tasuta Gemini API võtmega (unpaid):** kitsam — ~10 päringut/min, ~250 päringut/päevas, ainult Flash-mudelid\

\ 18. juunil 2026 asendatakse unpaid tieri ja Google One kasutajate jaoks Gemini CLI **Antigravity CLI-ga** — see puudutab konkreetset autentimisteed, mitte tasuta juurdepääsu tervikuna. Suurema mahu või Pro-mudelite jaoks on vaja Google AI Pro/Ultra tellimust või Gemini Code Assist litsentsi. Kontrolli oma projekti tegelikke limiite AI Studios, sest need muutuvad sageli ja sõltuvad regioonist ja konto seisust.

Eeldus	Kontrollimine	Link
Google konto	gmail.com ligipääs töötab	accounts.google.com
Google AI Studio API võti	aistudio.google.com/apikey	Loo API võti
Gemini CLI installitud	`gemini –version`	`npm install -g @google/gemini-cli`

Ametlik dokumentatsioon: geminicli.com/docs

2.4 Admin õigused — millal on vaja?



Hea uudis: peaaegu kõik sammud saab teha **ilma** administraatoriõigusteta. Tõelist admin-õigust läheb vaja ainult ühel hetkel - Node.js installimisel.

Samm	Admin õigused?	Põhjus
Node.js installimine	**Jah** - installer nõuab	Süsteemitaseme paigaldus (Program Files)
PowerShell execution policy	Ei	`-Scope CurrentUser` kirjutab HKCU registrisse
CredentialManager moodul	Ei	`-Scope CurrentUser` paigaldab kasutaja profiili alla
npm install (MCP server)	Ei	Kasutajataseme paigaldus
Claude Desktop config	Ei	Kasutaja kaustad
Gemini CLI config	Ei	Kasutaja kaustad
API võtme salvestamine	Ei	Credential Manager kasutajataseme võti

Riskid ja oluline hoiatus:

- Node.js installer — madal risk, ametlik tarkvara
- **Ära ava ühtegi ülejäänud sammu administraatorina**, isegi kui varasem juhendi versioon või internetist leitud näited seda soovitavad - `-Scope CurrentUser` on mõeldud just admin-õiguste vältimiseks

- Kui sinu igapäevane konto ja admin-õigused tulevad **eraldi Windowsi kontodelt** (levinud ettevõtte/domeeni keskkonnas), tekitab administraatorina avamine tegevuse teise konto profiili all — moodul, seadistus või võti „kaob“ sinu tavakasutaja vaatevinklist. Vaata täpsemalt tõrkeotsingu peatükki.
- Ära käivita `npm install` administraatorina - see tekitab õiguste konflikte
- `npm install -g` (nt Gemini CLI, Samm 5.1) vajab admin-õigusi ainult siis, kui npm'i globaalne prefix osutab `Program Files` kausta — Samm 5.1 sisaldab parandust, mis seda vältib

3. Ühised sammud (tee üks kord)



Need sammud teed **oma arvutis**. Kui kolleeg soovib sama lahendust kasutada, peab ta need sammud tegema **oma arvutis** iseseisvalt.

Samm 3.1 — Node.js paigaldamine (kohustuslik)



See on ainus samm, mis on **kohustuslik mõlemal teel** allpool ja mida ei saa automatiseerida (vajab admin-õigusi). Tee see enne edasi liikumist, olenemata sellest, kumma tee valid.

1. Mine https://nodejs.org ja laadi alla **LTS** versioon (22 või uuem)
2. Käivita allalaaditud `.msi` fail administraatorina
3. Installi vaikimisi seadetega — ära muuda paigalduskohta

Kontrolli paigaldust (tavaline PowerShell, mitte admin):

```
node --version
```

Oodatav tulemus:

```
v22.11.0 või uuem
```



`npm –version` kontrolli siin veel tahtlikult ei tee — npm on PowerShelli skript (`npm.ps1`) ja annab enne execution policy seadistamist (Tee A skript või Samm B1) vea `running scripts is disabled`, isegi kui Node.js ja npm ise on korrektselt paigaldatud. See ei ole viga, vaid oodatav käitumine selles etapis. `npm –version` töötab pärast Tee A/B läbimist.



Kui PowerShell ütleb „not recognized“ — sulge aken ja ava uuesti. Node.js lisab end PATH-i ainult uutes PowerShell sessioonides.

Vali oma tee



Allpool on **kaks alternatiivset teed** sama tulemuse saavutamiseks — MCP server töökorda seatud, koos API võtme ja AI kliendi seadistusega. **Tee ainult ÜKS neist, mitte mõlemat.**

	Tee A — Skriptid	Tee B — Käsitsi
Aeg	~5 minutit	~20 minutit
Sobib	Enamikule kasutajatele	Soovid ise igast sammust aru saada, või Tee A ei tööta sinu keskkonnas
Sisaldab samme	A1—A2	B1—B7

Kui valid Tee A ja see õnnestub, **jäta Tee B (B1—B7) täielikult vahele** — need kirjeldavad sama asja, mida skriptid juba sinu eest tegid, ainult käsitsi. Tee B on abimaterjal tõrkeotsinguks, mitte täiendav kohustuslik samm.

TEE A — Kiirpaigaldus skriptidega (soovitatud)

Samm A1 — failide allalaadimine

Lae alla `directo-mcp-failid.zip` ja paki lahti:

directo-mcp-failid.zip



Failid on wikisse lisatud **ZIP-arhiivina**, mitte eraldi linkidena — DokuWiki blokeerib vaikimisi `.js` ja `.ps1` failide üleslaadimise (turvakaalutlusel, vt `conf/mime.conf`), aga `.zip` on vaikimisi lubatud. Ära nimeta faile ümber teiseks laiendiks (nt `.txt`) „lahenduseks“ — vale laiend põhjustab hiljem `MODULE_NOT_FOUND` tüüpi vigu.

Paki lahti (parem klõps ZIP-il → **Extract All**, või PowerShellis):

```
Expand-Archive "$env:USERPROFILE\Downloads\directo-mcp-failid.zip" -
DestinationPath "$env:USERPROFILE\Downloads\directo-mcp-failid"
cd "$env:USERPROFILE\Downloads\directo-mcp-failid"
```

Enne skriptide käivitamist tuleb kõrvaldada **kaks** Windowsi vaikimisi turvapiirangut — mõlemad on kohustuslikud, mitte valikulised:

1) Luba PowerShellis skriptide käivitamine üldse:

```
Set-ExecutionPolicy -ExecutionPolicy RemoteSigned -Scope CurrentUser
```



Ilma selleta annab iga järgnev skript vea `running scripts is disabled on`

this system / UnauthorizedAccess. -Scope CurrentUser ei vaja admin-õigusi.

2) Eemalda Windowsi „kaugelt pärit“ (Mark of the Web) märgistus:

```
Get-ChildItem "$env:USERPROFILE\Downloads\directo-mcp-failid" -Recurse |
Unblock-File
```

Ka see on kohustuslik, mitte lisavõimalus — brauseri kaudu alla laaditud failid on vaikimisi blokeeritud **isegi kui execution policy on juba õige**. Windows märgistab internetist alla laaditud failid eraldi turvasildiga, mis kandub üle ka ZIP-i seest lahti pakitud failidele — RemoteSigned nõuab selliste failide puhul digiallkirja, mida meie skriptidel pole. Ilma selle sammuta on veateade `is not digitally signed` / `UnauthorizedAccess`, mis on eksitavalt sarnane 1. sammu veaga, aga põhjus ja lahendus on erinevad.

Arhiiv sisaldab nelja faili: `directo-mcp.js` (MCP server ise), `set-directo-apikey.ps1`, `install-claude-mcp.ps1`, `install-gemini-mcp.ps1`.

Samm A2 — käivitamine

Samas PowerShelli aknas (mitte administraatorina), juba lahtipakitud kaustas: A2.1 API võti (ühine mõlemale, tee see esimesena)

```
.\set-directo-apikey.ps1
```

A2.2 Vali oma AI klient (või mõlemad):
Claude

```
.\install-claude-mcp.ps1
```

Gemini

```
.\install-gemini-mcp.ps1
```

Iga skript näitab lõpus kokkuvõtte — mis on tehtud (✅) ja mis vajab veel käsitsi tegevust (⚠).

Kui skriptid õnnestusid, oled Peatüki 3 osas valmis. Mine otse Peatükki 4.3 (Claude Desktop) ja/või Peatükki 5 (Gemini CLI) — Tee B (allpool) ei ole vajalik.

TEE B — Käsitsi, samm-sammult (alternatiiv Tee A-le)

See on **Tee A alternatiiv**, mitte täiendus. Tee seda ainult siis, kui Tee A skriptid ei õnnestunud, või soovid iga sammu ise läbi teha. Kui Tee A juba õnnestus, jäta see sektsioon vahele.

Samm B1 — PowerShell execution policy seadistamine

Kontrolli olemasolevat seadistust:

```
Get-ExecutionPolicy
```

Kui vastus ei ole `RemoteSigned`, paranda seda **tavalises** PowerShellis (mitte administraatorina):

```
Set-ExecutionPolicy -ExecutionPolicy RemoteSigned -Scope CurrentUser
```



`-Scope CurrentUser` kirjutab seadistuse ainult `HKEY_CURRENT_USER` registrisse, mitte masina tasemel kohta — seetõttu ei vaja see admin-õigusi. Admin-õigusi läheks vaja ainult `-Scope LocalMachine` korral, mida see juhend ei kasuta.

Küsib kinnitust → vajuta Y → Enter.

Samm B2 — CredentialManager moodul

Ava **tavaline** PowerShell (mitte administraatorina) — täpselt sama kasutajakonto all, mille all hiljem ka MCP server töötab:

```
Install-Module -Name CredentialManager -Force -Scope CurrentUser
```



`-Scope CurrentUser` on mõeldud just selleks, et vältida admin-õiguste vajadust — moodul paigaldub sinu enda profiili alla (`Documents\WindowsPowerShell\Modules`). Kui käsk küsib kinnitust NuGet provideri või „untrusted repository“ jaoks, vasta Y — ka need sammud ei vaja admin-õigusi.

Kontrolli et installimine õnnestus (samas tavalises PowerShellis):

```
Get-Module -ListAvailable CredentialManager
```

Peaks kuvama mooduli nime ja versiooni.



⚠ Kui sinu igapäevane töökonto ja admin-õigused tulevad **eraldi Windowsi kontodelt** (levinud ettevõtte/domeeni seadistus), veendu et see samm ja hilisem Samm B7 toimuvad **sama konto** all, mille alt hiljem käivitub ka `node directo-mcp.js`. Elevatsiooniga avatud PowerShell võib tegelikult töötada teise konto identiteedi all, mistõttu moodul või API võti jääks vale profiili alla ja hilisem samm ebaõnnestub nähtamatult.

Samm B3 — MCP serveri kausta loomine

Ava tavaline PowerShell (mitte admin) ja käivita:

```
New-Item -ItemType Directory -Force -Path "$env:USERPROFILE\directo-mcp"
```



Kasuta alati `$env:USERPROFILE` muutujat, mitte käsitsi kirjutatud `C:\Users\SINUNIMI`. Domeeniga liidetud arvutites (ettevõtte/AD keskkond) võib tegelik profiilikaust erineda sinu sisselogimisnimest — nt sisaldada domeenisufiksit (`SINUNIMI.DOMEEN`). `$env:USERPROFILE` annab alati tegeliku, õige tee, sõltumata sellest.

Samm B4 — directo-mcp.js paigutamine



See fail sisaldab kõiki 32 Directo API tööriista ja on **täielikult skeemipõhiselt kontrollitud** — iga parameeter on üle vaadatud vastu Directo API tegelikke ws_*.json skeeme. Ära kopeeri seda faili käsitsi Notepadist ega mõnest muust allikast — kasuta alati värsket, ametlikku versiooni, et vältida vananenud/valede parameetritega koopiaid.

Lae alla `directo-mcp.js` fail — see sisaldub Tee A juures viidatud `directo-mcp-failid.zip` arhiivis (paki lahti, vt Samm A1) — ja aseta see kausta:

```
$env:USERPROFILE\directo-mcp\directo-mcp.js
```

Kõige lihtsam viis: kui fail on juba lahtipakitud Downloads alamkaustas, kopeeri see PowerShellis:

```
Copy-Item "$env:USERPROFILE\Downloads\directo-mcp-failid\directo-mcp.js"
"$env:USERPROFILE\directo-mcp\directo-mcp.js"
```



Veendu et failinimi on täpselt `directo-mcp.js` — mitte `directo-mcp.js.txt`. Kontrolli:

```
Test-Path "$env:USERPROFILE\directo-mcp\directo-mcp.js"
```

Peaks vastama `True`. Kui `False`, kontrolli täpset failinime:

```
Get-ChildItem "$env:USERPROFILE\directo-mcp"
```

Samm B5 — package.json loomine

Ava uus Notepad aken ja kopeeri:

```
{
  "name": "directo-mcp",
  "version": "1.0.0",
  "type": "module",
  "description": "Directo ERP MCP Server",
  "main": "directo-mcp.js",
  "scripts": {
    "start": "node directo-mcp.js"
  },
  "dependencies": {
    "@modelcontextprotocol/sdk": "^1.0.0",
    "zod": "^3.22.0"
  }
}
```

Salvesta sama kausta nimega `package.json` (failitüüp: All Files).

Samm B6 — Sõltuvuste installimine

Tavaline PowerShell (mitte admin):

```
cd "$env:USERPROFILE\directo-mcp"
npm install
```

Edukal installimisel tekib kaustas `node_modules` kaust ja `package-lock.json` fail.



See on **lokaalne** npm install (paigaldub `directo-mcp` kausta enda `node_modules` alla), mitte globaalne — ei vaja kunagi admin-õigusi ega erilist npm-prefixi seadistust, sõltumata sellest, kuidas Node.js algselt paigaldati.

Samm B7 — API võtme turvaline salvestamine

Directo API võti salvestatakse Windows Credential Managerisse — krüpteeritult sinu kasutajakonto alla. Keegi teine arvutis ei näe seda võtit.

```
New-StoredCredential -Target DIRECTO_API_KEY -UserName directo -Password
SINU_PÄRIS_VÕTI_SIIA -Persist LocalMachine
```

Kontrolli et salvestus õnnestus:

```
(Get-StoredCredential -Target
```

```
DIRECTO_API_KEY).GetNetworkCredential().Password
```

Peaks kuvama sinu API võtme.

API võtme uuendamine tulevikus (rotatsioon):

```
Remove-StoredCredential -Target DIRECTO_API_KEY
New-StoredCredential -Target DIRECTO_API_KEY -UserName directo -Password
UUS_VÕTI -Persist LocalMachine
```



Käivita see samm **täpselt samas kasutajakontos**, mille all hiljem ka node `directo-mcp.js` töötab. Vaata Sammu B2 hoiatust eraldi admin-konto riski kohta.



Tee A skript (`set-directo-apikey.ps1`) teeb sama asja turvalisemalt (küsib võtme varjatult, ei kuva ekraanil) ja sobib ka rotatsiooniks — kasuta seda, kui saad.

4. Claude Desktop seadistamine



Tee kõigepealt Peatükk 3 (Samm 3.1 + kas Tee A või Tee B). Claude Desktop seadistamine eeldab et MCP server on juba loodud.



MCP server töötab ainult **sinu arvutis** ja ainult **Claude Desktop rakenduses** — mitte claude.ai veebiversioonis ega teiste kasutajate arvutites.



Soovitatud: kui käivitasid Tee A (Peatükk 3) `install-claude-mcp.ps1` skripti, on Sammud 4.1—4.2 juba tehtud — skript tuvastab, patchib ja varundab konfiguratsioonifaili automaatselt. Mine otse Sammu 4.3 juurde.

Samm 4.1 — Konfiguratsioonifaili asukoht

Claude Desktopil on Windowsis **kaks võimalikku** konfiguratsiooni asukohta, sõltuvalt installimisviisist:

Installimisviis	Asukoht
Otsene `.exe` installer (tavaline)	`%APPDATA%\Claude\`
Microsoft Store / MSIX	`%LOCALAPPDATA%\Packages\Claude_pzs8sxrjxfjjc\LocalCache\Roaming\Claude\`



See on dokumenteeritud, aktiivne Claude Desktopi tõrge: MSIX-installi puhul Windows **virtualiseerib** `%APPDATA%\Claude\` asukoha läbipaistmatult teisele teele. Kui muudad vale (standard) faili, ei jõua muudatus kunagi rakenduseni ja MCP server jääb vaikselt laadimata, ilma selgitava veata.

Kontrolli, kumb sinu masinas kehtib — ava File Explorer ja proovi mõlemat aadressiribale:

```
%APPDATA%\Claude
```

Kui seal on olemas `claude_desktop_config.json` fail (või Claude Desktop on seal midagi juba loonud), kasuta seda teed. Kui mitte, proovi:

```
%LOCALAPPDATA%\Packages\Claude_pzs8sxrjxfjjc\LocalCache\Roaming\Claude
```



Kui kumbki kaust ei eksisteeri, käivita Claude Desktop vähemalt korra — see loob vajalikud kaustad automaatselt (tavaliselt standardteele).

Samm 4.2 — Konfiguratsioonifail

Loo (õiges, Sammus 4.1 tuvastatud) kaustas fail nimega `claude_desktop_config.json`.



Faili loomisel Notepadiga: File → Save As → failitüüp **All Files (*.*)** → nimi: `claude_desktop_config.json`

Faili sisu:

```
{
  "mcpServers": {
    "directo": {
      "command": "node",
      "args": [
        "C:\\Users\\SINUNIMI\\directo-mcp\\directo-mcp.js"
      ]
    }
  }
}
```



Asenda `SINUNIMI` oma **tegeliku** profiilikausta nimega — mitte oletatava kasutajanimega. Domeeniga (AD) liidetud arvutites võib tegelik profiilikaust erineda sisselogimisnimest (nt sisaldada domeenisufiksit). Kontrolli täpset väärtust:

```
$env:USERPROFILE
```

Kasuta JSON-is täpselt seda, mida käsk tagastab (topelt-kaldkriipsudega). Näide

illustratsiooniks — kui `$env:USERPROFILE` annab tulemuseks `C:\Users\jaan.kasutaja.FIRMA`, peaks tee JSON-is olema `C:\\Users\\jaan.kasutaja.FIRMA\\directo-mcp\\directo-mcp.js`. Domeenisufiksi (`.FIRMA`) olemasolu või puudumine sõltub sinu enda arvuti seadistusest — kasuta alati oma käsu tegelikku väljundit, mitte seda näidet.

Kui fail juba sisaldab muud sisu (Claude Desktopi enda seaded, nt `preferences` plokk) — **ära kirjuta seda üle**. `mcpServers` peab olema faili **juuretasemel**, samal tasemel nagu `preferences`, mitte selle sees. Kui pole kindel sulgude õigest asukohast, kasuta pigem Sammu 3.0 skripti, mis muudab faili struktuurina (`ConvertFrom-Json`/`ConvertTo-Json`), mitte käsitsi tekstina — see välistab sulgude-vea täielikult.

API võtit **ei lisata** konfiguratsioonifaili — server loeb selle automaatselt Credential Managerist (Tee A skript või Samm B7).

Samm 4.3 — Claude Desktop taaskäivitamine

1. Paremklikk Claude ikoonil **süsteemisalves** (tegumiriba kella kõrval) → **Quit** / **Exit**
2. Pelgalt akna sulgemisest ei piisa — Claude Desktop jääb taustal tööle
3. Kui Quit puudub — ava Task Manager → leia Claude protsess → End Task
4. Ava Claude Desktop uuesti ja oota mõni sekund

Samm 4.4 — Testimine

Kirjuta Claude Desktop vestluses:

> Anna mulle kõik laoseisud

Claude peaks automaatselt kasutama `get_stocklevels` tööriista ja tagastama Directo andmed.

Logide kontroll kui midagi ei tööta:

Logi asukoht järgib sama kahte võimalust, mis Sammus 4.1:

```
# Otsene .exe installer:
Get-Content "$env:APPDATA\Claude\logs\mcp-server-directo.log" -Tail 30

# Microsoft Store / MSIX:
Get-Content
"$env:LOCALAPPDATA\Packages\Claude_pzs8sxrjxfjjc\LocalCache\Roaming\Claude\l
ogs\mcp-server-directo.log" -Tail 30
```

Ametlik Claude Desktop MCP dokumentatsioon: modelcontextprotocol.io — Connect to local MCP servers

5. Gemini CLI seadistamine

Tee kõigepealt Peatükk 3 (Samm 3.1 + kas Tee A või Tee B). Gemini CLI seadistamine eeldab et MCP server on juba loodud.

MCP server töötab ainult **sinu arvutis** ja ainult terminalipõhises Gemini CLI rakenduses — mitte Gemini veebiversioonis ega teiste kasutajate arvutites.

⚠ Enne jätkamist veendu, et sul on kehtiv autentimine — kas Google konto (Gemini Code Assist), tasuta Gemini API võti (Flash-mudelid) või tasuline tellimus. Limiidid ja mudelivalik sõltuvad autentimisviisist (vt Samm 2.3).

Soovitatud: kui käivitasid Tee A (Peatükk 3) `install-gemini-mcp.ps1` skripti, on Sammud 5.1 ja 5.3—5.4 juba tehtud. Käi läbi ainult Samm 5.2 (Google API võti — skript seda ei tee, see on eraldiseisev Google'i, mitte Directo, autentimine) ja Sammud 5.5—5.6.

Samm 5.1 — Gemini CLI paigaldamine

See samm ei vaja admin-õigusi — tingimusel, et npm'i globaalne install-asukoht on suunatud sinu enda profiili sisse. Kui pole veel seadistatud, tee see korra läbi (tavaline PowerShell, mitte administraatorina):

```
mkdir "$env:APPDATA\npm-global" -Force
npm config set prefix "$env:APPDATA\npm-global"
[Environment]::SetEnvironmentVariable("Path", $env:Path +
";$env:APPDATA\npm-global", "User")
```

Sulge PowerShell täielikult ja ava **uus** aken (PATH värskendub alles uues sessioonis), seejärel paigalda Gemini CLI:

```
npm install -g @google/gemini-cli
```

Kontrolli:

```
gemini --version
```

⚠ Kui käsk ikkagi küsib admin-õigusi, on tõenäoline põhjus, et npm'i prefix on juba varem (nt teise tarkvara installi käigus) seadistatud osutama `C:\Program Files\…` peale. Kontrolli praegust seadistust käsuga `npm config get prefix` — see peaks

pärast ülaltoodud sammu näitama `%APPDATA%\npm-global`, mitte Program Files kausta.

Samm 5.2 — Google API võtme seadistamine

Ava aistudio.google.com/apikey ja loo API võti.

Salvesta võti keskkonnmuutujana püsivalt:

```
[System.Environment]::SetEnvironmentVariable("GEMINI_API_KEY",
"SINU_GOOGLE_API_VÕTI", "User")
```

Sulge PowerShell ja ava uuesti. Kontrolli:

```
$env:GEMINI_API_KEY
```

Samm 5.3 — Konfiguratsioonifaili asukoht

Gemini CLI konfiguratsioon asub Windowsis:

```
$env:USERPROFILE\.gemini\settings.json
```

Kontrolli kas kaust on olemas:

```
Test-Path "$env:USERPROFILE\.gemini"
```

Kui vastus on `False`, loo kaust:

```
New-Item -ItemType Directory -Force -Path "$env:USERPROFILE\.gemini"
```

Samm 5.4 — MCP serveri lisamine

Loo või muuda faili `$env:USERPROFILE\.gemini\settings.json`:

```
{
  "mcpServers": {
    "directo": {
      "command": "node",
      "args": [
        "C:\\Users\\SINUNIMI\\directo-mcp\\directo-mcp.js"
      ],
      "env": {
        "DIRECTO_API_KEY": "$DIRECTO_API_KEY"
      }
    }
  }
```

```
}
```



„trust“: true lisandub Sammus 5.5 — see puudutab kausta usaldamist, mitte API võtit, seega on eraldi teema.



Asenda SINUNIMI oma **tegeliku** profiilikausta nimega — kontrolli täpset väärtust käsuga $env:USERPROFILE (vt Sammu 4.2 hoiatust sama teema kohta).



API võtme loogika täpsustus: directo-mcp.js loeb võtme kõigepealt Credential Managerist (Tee A skript või Samm B7) — see töötab **täpselt samamoodi Gemini puhul kui Claude Desktopi puhul**, kuna mõlemad käivitavad sama serverifaili. env plokk siin on **varuvõimalus**, mitte ainuke tee — kui Credential Manager on juba seadistatud, ei ole see plokk rangelt vajalik, aga ei tee ka kahju. Kui soovid seda siiski kasutada, salvesta väärtus keskkonnamuutujana:

```
[System.Environment]::SetEnvironmentVariable("DIRECTO_API_KEY", "SINU_DIRECTO_API_VÕTI", "User")
```

Gemini CLI toetab env väärtustes keskkonnamuutuja laiendust süntaksiga $VAR_NAME — sellepärast on JSON-is kirjas „$DIRECTO_API_KEY“, mitte tegelik võti.

Samm 5.5 — Kausta usaldamine (Trusted Folders)

Gemini CLI-l on turvafunktsioon **Trusted Folders**, mis vaikimisi ei luba usaldamata kaustast MCP servereid ühendada. Kui see funktsioon on sisse lülitatud, tuleb kaust usaldada.

Kõige lihtsam viis — usalda server otse settings.json-is, lisades directo plokki „trust“: true:

```
{
  "mcpServers": {
    "directo": {
      "command": "node",
      "args": [
        "C:\\Users\\SINUNIMI\\directo-mcp\\directo-mcp.js"
      ],
      "env": {
        "DIRECTO_API_KEY": "$DIRECTO_API_KEY"
      },
      "trust": true
    }
  }
}
```



„trust“: true möödub tööriistakutsete kinnitusdialoogidest. Kasuta seda **ainult** serverite puhul, mida sa ise täielikult haldad — see Directo server on selline.

Alternatiiv — kui Trusted Folders on sisse lülitatud, kuvab Gemini CLI kausta esmakordsel avamisel usaldusdialoogi. Vali seal **Trust folder**. Valik salvestatakse faili `~/.gemini/trustedFolders.json` ja seda küsitakse iga kausta kohta ainult üks kord. Usaldustaseme hilisemaks muutmiseks kasuta CLI sees käsku `/permissions`.

Samm 5.6 — Testimine

Käivita Gemini CLI:

```
gemini
```

Kontrolli MCP ühendust:

```
/mcp list
```

Peaks kuvama:

```
 directo - Ready (32 tools)
```

Testi päring:

> Anna mulle kõik laoseisud

Ametlik Gemini CLI MCP dokumentatsioon: geminicli.com/docs/tools/mcp-server

6. Tõrkeotsing

Üldised vead (mõlemale kliendile)

Veateade	Põhjus	Lahendus
`DIRECTO_API_KEY puudub`	Credential Manager salvestus ebaõnnestus	Korda Samm B7 (või Tee A: `.\set-directo-apikey.ps1`)
`Cannot find module`	npm install ebaõnnestus VÕI directo-mcp.js on vales kaustas/nimega	Korda Samm B6; kontrolli `Test-Path „$env:USERPROFILE\directo-mcp\directo-mcp.js“`
`SyntaxError`	directo-mcp.js fail on vigane või poolik	Lae fail uuesti alla, vt Samm B4 (või Tee A: A1)
`Incorrect filter id`	Vale API filtrinimi	Teadaolev piirang — kasuta ilma filtrita

Veateade	Põhjus	Lahendus
`401 / 403`	Directo API võti on vale või aegunud	Kontrolli võtit Directo adminis
`running scripts is disabled`	PowerShell execution policy	`Set-ExecutionPolicy RemoteSigned -Scope CurrentUser` (Samm A1 või B1)
`is not digitally signed` / `UnauthorizedAccess`	Windowsi Mark-of-the-Web (fail alla laaditud brauseriga)	`Get-ChildItem <kaust> -Recurse | Unblock-File` (vt Samm A1)
`npm not recognized`	Node.js pole PATH-is	Sulge ja ava PowerShell uuesti

Claude Desktop spetsiifilised vead

Probleem	Lahendus
Haamriikoon puudub	Kontrolli logifaili (vt allpool)
MCP server ei käivitu	Kontrolli et config faili nimi pole `.txt` laiendiga
Config kaust puudub	Käivita Claude Desktop korra — kaustad luuakse automaatselt
Config muudetud, aga midagi ei muutu	Kontrolli, et muudad ÕIGET faili — vt Samm 4.1 (kaks võimalikku asukohta, MSIX vs otsene install)

Logifail (proovi mõlemat, vt Samm 4.1):

```
Get-Content "$env:APPDATA\Claude\logs\mcp-server-directo.log" -Tail 30
Get-Content "$env:LOCALAPPDATA\Packages\Claude_pzs8sxrjxfjjc\LocalCache\Roaming\Claude\logs\mcp-server-directo.log" -Tail 30
```

Gemini CLI spetsiifilised vead

Probleem	Lahendus
`/mcp list` näitab Disconnected	Lisa `„trust“: true` serveri konfiguratsiooni (Samm 5.4) või usalda kaust dialoogis / käsuga `/permissions`
`GEMINI_API_KEY` viga	Korda Samm 5.2, ava uus PowerShell aken
Rate limit / 429 viga	Tasuta tieri päringulimiit ületatud — oota limiidi lähtestumist või kaalu tasulist tellimust (vt Samm 2.3)

Gemini CLI diagnostika:

```
/mcp list
```

7. MCP tööriistad

Mõlemad kliendid (Claude Desktop ja Gemini CLI) kasutavad samu tööriistu:

Grupp	Tööriistad
Müük	get_invoices, get_orders, get_sales_quotations, get_sales_contracts, get_receipts

Grupp	Tööriistad
Kliendid	get_customers, get_suppliers, get_contacts
Tooted / ladu	get_items, get_itemclasses, get_stocklevels, get_stocklevels_sn, get_allocations, get_priceformulas
Ost	get_purchase_orders, get_purchase_invoices, get_purchase_payments, get_stock_receipts
Raamatupidamine	get_transactions, get_accounts, get_financial_budgets, get_nettings
Logistika	get_deliveries, get_movements
Muu	get_projects, get_objects, get_resources, get_events, get_users, get_recipes, get_replacementlogs
Sünkroonimine	get_deleted

8. Turvalisus

Meede	Selgitus
Credential Manager (mõlemad kliendid)	API võti krüpteeritud, nähtav ainult sinu Windowsi kontole — Claude ja Gemini loevad sama kirjet
Keskkonnamuutuja (Gemini varuvõimalus)	Kui Credential Manager on seadistatud, pole `env` plokk rangelt vajalik — vt Sammu 5.4 täpsustust
Lokaalne server	MCP server töötab ainult sinu arvutis, ei ole avalikult ligipääsetav
Minimaalsed õigused	Piirita Directo API võtme õigused — ainult vajalikud ressursid
OneDrive/Dropbox	Välista `directo-mcp` kaust pilvesünkroonist
Admin õigused	Ainult Node.js installimisel (Samm 3.1) — jagatud eelsamm mõlemale teele — kõik muu, sh CredentialManager moodul, töötab `-Scope CurrentUser` all ilma admin-õigusteta (vt 2.4)

9. Viited

Claude Desktop:

- Claude Desktop allalaadimine
- Claude Desktop MCP seadistamine
- Claude Desktop lokaalsete MCP serverite juhend
- MCP ametlik dokumentatsioon

Gemini CLI:

- Gemini CLI dokumentatsioon
- Gemini CLI MCP seadistamine
- Google AI Studio API võtmed
- Gemini Code Assist dokumentatsioon

Directo:

- Directo API dokumentatsioon

Üldine:

- Node.js allalaadimine
- CredentialManager PowerShell moodul

Kiirpaigalduse skriptid (vt Tee A, Peatükk 3):

- `directo-mcp.js` — MCP server (32 tööriista, skeemipõhiselt kontrollitud)
- `set-directo-apikey.ps1` — API võtme haldus (algseadistus + rotatsioon)
- `install-claude-mcp.ps1` — Claude Desktopi automaatpaigaldus
- `install-gemini-mcp.ps1` — Gemini CLI automaatpaigaldus

From:
https://wiki.directo.ee/ - **Directo Help**

Permanent link:
https://wiki.directo.ee/et/ask_directo?rev=1790846140

Last update: **2026/10/01 12:15**