

Dify zelf hosten: de Community Edition op je eigen computer of server



Bijlage bij de workshop AI Agents

LeX Consultancy B.V. · AI Netwerk Limburg · 20 september 2026

In de workshop en in alle handleidingen gebruiken we de online versie op cloud.dify.ai: aanmelden, bouwen, klaar. Dify is daarnaast open source. Je kunt dezelfde software zelf draaien: op je eigen laptop om te oefenen, of op een server van school zodat gegevens de organisatie niet verlaten. Deze bijlage beschrijft hoe je dat doet op Windows, Mac en Linux.

De stappen zijn voor iemand die wel eens een terminal heeft geopend maar geen beheerder is. Reken op een half uur voor de eerste keer.

Wanneer wel, wanneer niet. Voor de workshop en om te leren bouwen is de online versie de beste keuze: geen installatie, alles werkt meteen. Zelf hosten wordt interessant als je met echte leerlinggegevens wilt werken, als de school geen data buiten Europa wil hebben, of als je meer dan een paar mensen structureel met Dify wilt laten werken. Let op: zelf hosten neemt de verplichtingen niet weg. Het taalmodel (OpenAI, Anthropic, Mistral of een lokaal model) blijft de plek waar je tekst naartoe stuurt; dáár moet een verwerkersovereenkomst op zitten, niet alleen op Dify.

Wat je krijgt

De Community Edition is dezelfde Dify als online, met dezelfde Studio, workflows, chatflows, kennisbanken en tools. Een paar verschillen:

	Online (cloud.dify.ai)	Zelf gehost (Community Edition)
Kosten	Sandbox gratis met credits; daarna abonnement	Gratis software; je betaalt je eigen modelgebruik en je eigen computer of server
Taalmodel	Standaard OpenAI via Dify-credits	Je sluit zelf een aanbieder aan: OpenAI, Anthropic, Mistral, Azure, of lokaal via Ollama
Gegevens	Bij Dify (cloud) en bij de modelaanbieder	Op jouw computer of server, en bij de modelaanbieder
Delen	Webapp-link werkt overal	Webapp-link werkt alleen waar jouw computer of server bereikbaar is
Updates	Vanzelf	Zelf doen (paragraaf "Bijwerken")
Agent Console	Beschikbaar (bèta)	Volgt de releases; kan achterlopen op de cloud

Alle handleidingen in deze serie werken ook op een eigen installatie. Het enige verschil is het adres in de browser: `http://localhost` in plaats van `cloud.dify.ai`.

Wat je nodig hebt

- Een computer met minstens **2 processorkernen en 4 GB werkgeheugen** vrij voor Dify. In de praktijk: 8 GB in totaal is prettig, 16 GB werkt ontspannen.
- Ongeveer **10 GB vrije schijfruimte**.
- **Docker**: het programma dat Dify en zijn onderdelen (database, zoekindex, webserver) in aparte "containers" draait. Dify bestaat uit zo'n vijftien van die containers; Docker start ze in één keer.
- **Git** om de bestanden van Dify op te halen. Kan ook zonder: dan download je een zip.
- Een **API-sleutel van een modelaanbieder** (bijvoorbeeld OpenAI), of Ollama voor een lokaal model.

De installatie draait op Windows 10/11 (via WSL 2), macOS 10.14 of hoger (Intel en Apple Silicon) en elke gangbare Linux.

Stap 1. Docker installeren

Windows

1. Ga naar docker.com en download **Docker Desktop for Windows**.
2. Start het installatieprogramma. Laat **Use WSL 2** aangevinkt; dat is de Linux-laag waarin de containers draaien. Windows installeert WSL 2 zelf als het nog ontbreekt; soms is een herstart nodig.

3. Start Docker Desktop en wacht tot linksonder "Engine running" staat.
4. Open **Settings** → **Resources** en geef Docker minstens 8 GB geheugen als je computer dat heeft.

Bewaar de Dify-bestanden straks in de Linux-omgeving (bijvoorbeeld in `\\wsl$\\Ubuntu\\home\\jouwnaam`), niet op `C:\\Users\\...`. Dify draait vanaf een Windows-map merkbaar trager en geeft soms rechtenfouten.

Mac

1. Ga naar docker.com en download **Docker Desktop for Mac**. Kies de versie voor je processor (Apple Silicon of Intel; kijk onder → Over deze Mac).
2. Sleep Docker naar Programma's en start het. Geef de gevraagde toestemmingen.
3. Open **Settings** → **Resources** en zet het geheugen op minstens 8 GB en het aantal CPU's op minstens 2.

Linux

Op Ubuntu of Debian, in een terminal:

INTYPEN

```
sudo apt update
sudo apt install docker.io docker-compose-v2 git curl
sudo usermod -aG docker $USER
```

Meld je af en weer aan zodat de laatste regel werkt (jij mag dan Docker gebruiken zonder sudo). Op Fedora, Arch en andere distributies staat de installatie op docs.docker.com/engine/install. Dify vraagt Docker 19.03 of hoger en Docker Compose 2.24 of hoger. Controleer:

INTYPEN

```
docker --version
docker compose version
```

Stap 2. Dify ophalen

Open een terminal. Op Windows is dat de **Ubuntu**-app (WSL) of PowerShell; op Mac **Terminal**; op Linux je gewone terminal. Kies een map waar Dify mag staan en ga daarheen.

De ontwikkelaars raden aan de laatste **uitgebrachte versie** op te halen, niet de ontwikkelversie:

INTYPEN

```
git clone --branch "$(curl -s  
https://api.github.com/repos/langgenius/dify/releases/latest | jq -r  
.tag_name)" https://github.com/langgenius/dify.git
```

Die regel vraagt eerst aan GitHub welk versienummer de laatste release heeft en haalt precies die op. Hij heeft `git`, `curl` en `jq` nodig. Werkt hij niet (bijvoorbeeld omdat `jq` ontbreekt), doe het dan in twee stappen: kijk op github.com/langgenius/dify/releases welk versienummer bovenaan staat, en gebruik dat (op 20 september 2026 was dat 1.17.1):

INTYPEN

```
git clone --branch 1.17.1 https://github.com/langgenius/dify.git
```

Zonder git: download op dezelfde pagina onder **Assets** de zip van de nieuwste release en pak die uit.

Ga daarna naar de map met de Docker-bestanden:

INTYPEN

```
cd dify/docker
```

Stap 3. Instellingen klaarzetten

Dify leest zijn instellingen uit een bestand `.env`. Er staat een voorbeeld klaar; kopieer dat:

INTYPEN

```
cp .env.example .env
```

Op Windows in PowerShell: `copy .env.example .env`.

Voor een eerste installatie op je eigen computer hoeft je niets te wijzigen. Een paar regels die je later wel wilt weten:

Regel in <code>.env</code>	Wat het doet
<code>EXPOSE_NGINX_PORT=80</code>	De poort waarop Dify bereikbaar is. Staat 80 al in gebruik (bijvoorbeeld door een andere webserver), zet dan <code>8080</code> ; het adres wordt dan <code>http://localhost:8080</code> .
<code>SECRET_KEY=</code>	Sleutel waarmee Dify wachtwoorden en sessies beveiligt. Voor een server: vul hier een lange willekeurige tekenreeks in vóór de eerste start.
<code>CONSOLE_WEB_URL=</code> en <code>APP_WEB_URL=</code>	Het publieke adres van je installatie. Leeg is goed op je eigen computer; op een server vul je hier het adres in dat collega's gebruiken.
<code>UPLOAD_FILE_SIZE_LIMIT=15</code>	Maximale bestandsgrootte in MB voor uploads naar een kennisbank. Voor de transcriptietool (handleiding 05) met opnames van tientallen MB zet je dit hoger.

Het bestand `.env` bevat straks gevoelige gegevens. Zet het nooit in git of op een gedeelde schijf.

Stap 4. Starten

INTYPEN

```
docker compose up -d
```

De eerste keer haalt Docker alle onderdelen op; dat duurt een paar minuten, afhankelijk van je verbinding. `-d` betekent "op de achtergrond": je terminal komt terug en Dify blijft draaien.

Controleer of alles loopt:

INTYPEN

```
docker compose ps
```

Elke regel hoort **Up** of **healthy** te zeggen. Een container die **Restarting** blijft zeggen is meestal een geheugentekort: geef Docker meer geheugen (stap 1) en start opnieuw.

Stap 5. Beheerdersaccount aanmaken

Open in de browser:

INTYPEN

```
http://localhost/install
```

(Met een andere poort: `http://localhost:8080/install`. Op een server: `http://<adres-van-de-server>/install`.)

Je maakt hier het eerste account aan: dat is de beheerder van de werkruimte. Daarna kom je op `http://localhost` in dezelfde Studio als online.

Deze installatiepagina werkt maar één keer. Daarna log je gewoon in op `http://localhost`.

Stap 6. Een taalmodel aansluiten

Anders dan online heeft je eigen Dify geen credits en geen standaardmodel. Je sluit zelf een aanbieder aan.

1. Klik rechtsboven op je accountnaam en kies **Settings → Model Provider**.
2. Kies een aanbieder (OpenAI, Anthropic, Mistral, Azure OpenAI, Google) en klik **Install** als de plugin nog niet aanwezig is.
3. Klik **Setup** en plak je API-sleutel. Dify controleert de sleutel direct.
4. Kies onder **System Model Settings** welk model standaard wordt gebruikt voor nieuwe apps.

Vanaf hier zijn alle handleidingen van deze serie letterlijk te volgen. Waar een handleiding **gpt-5-mini** kiest, kies je hetzelfde model bij jouw aanbieder.

Zelf kiezen: een lokaal model. Wil je dat er helemaal niets naar buiten gaat, dan kun je een model op je eigen computer draaien met **Ollama** (ollama.com). Installeer Ollama, haal een model op (`ollama pull gemma3:12b` of een ander model dat op je computer past) en voeg in Dify de aanbieder **Ollama** toe met adres `http://host.docker.internal:11434`. Reken op minstens 16 GB werkgeheugen en veel geduld: de kwaliteit en snelheid liggen onder die van de grote aanbieders, maar voor oefenen en voor gevoelige teksten kan het genoeg zijn.

Stoppen, starten, bijwerken

Stoppen (gegevens blijven bewaard):

INTYPEN

```
docker compose down
```

Weer starten:

INTYPEN

```
docker compose up -d
```

Bijwerken naar een nieuwe versie. Lees eerst de release notes op github.com/langgenius/dify/releases: soms is er een extra stap. Daarna, in de map `dify/docker`:

INTYPEN

```
docker compose down  
git fetch --tags  
git checkout <nieuw versienummer>  
docker compose pull  
docker compose up -d
```

Vergelijk na een update `.env.example` met je eigen `.env`: nieuwe instellingen komen alleen in het voorbeeldbestand terecht. Heb je `docker-compose.yaml` zelf aangepast, zet die aanpassingen dan opnieuw.

Alles verwijderen, inclusief alle apps, kennisbanken en gebruikers:

INTYPEN

```
docker compose down -v
```

De `-v` gooit ook de opslag weg. Zonder `-v` blijft alles staan.

Waar je gegevens staan

Alles wat je in Dify maakt (apps, kennisbanken, gesprekken, uploads) staat in de map `dify/docker/volumes`. Een back-up is een kopie van die map terwijl Dify gestopt is. Apps kun je daarnaast los exporteren als DSL-bestand (in Studio: de drie puntjes bij een app → **Export DSL**). Zo'n bestand kun je in elke andere Dify importeren, ook online. Dat is ook de manier om een app die je thuis hebt gebouwd naar de schoolinstallatie te brengen.

Op een server voor de school

Op je eigen laptop werkt bovenstaande zonder meer. Voor gebruik door collega's komt er een laag bij, en dat is werk voor ICT:

- Een server (eigen hardware of een virtuele machine bij een Europese aanbieder) met 4 kernen, 8 tot 16 GB geheugen en een vaste naam zoals `dify.school.nl`.
- **HTTPS**: een certificaat en een reverse proxy (Nginx, Caddy, Traefik) ervoor. In `.env` zet je dan `CONSOLE_WEB_URL` en `APP_WEB_URL` op `https://dify.school.nl`.

- **Inloggen met schoolaccounts** (Microsoft Entra of Google Workspace) via de SSO-opties in `.env`; zonder SSO maakt de beheerder de accounts zelf aan.
- **Back-ups** van `volumes`, dagelijks, en één keer getest terugzetten.
- **Een verwerkersovereenkomst** met de modelaanbieder, en een besluit welke gegevens er wel en niet in mogen. Zelf hosten maakt dat gesprek niet overbodig; het maakt het wel eenvoudiger, omdat één partij minder in de keten zit.

De uitgebreide documentatie voor productiegebruik staat op docs.dify.ai onder **Self-Host**.

Als het niet werkt

Wat je ziet	Wat er aan de hand is
<code>docker: command not found</code>	Docker is niet geïnstalleerd of (Linux) je bent nog niet opnieuw aangemeld na <code>usermod</code>
<code>port is already allocated</code>	Poort 80 is bezet; zet <code>EXPOSE_NGINX_PORT=8080</code> in <code>.env</code> en start opnieuw
Pagina blijft laden, <code>docker compose ps</code> zegt Restarting	Te weinig geheugen voor Docker; verhoog het in Docker Desktop of sluit andere programma's
<code>http://localhost/install</code> geeft "already installed"	Er is al een account; ga naar <code>http://localhost</code> en log in
Modelsleutel wordt geweigerd	Sleutel verkeerd geplakt, of de aanbieder heeft geen tegoed op die sleutel
Kennisbank indexeert niet	De container <code>weaviate</code> (of de gekozen vectordatabase) draait niet; kijk met <code>docker compose logs weaviate</code>
Op Windows heel traag	De bestanden staan op <code>C:\</code> ; verplaats de map <code>dify</code> naar de Linux-kant van WSL

Logboek van een onderdeel bekijken, bijvoorbeeld de API:

INTYPEN

```
docker compose logs -f api
```

Licentie: CC BY-SA 4.0 (<https://creativecommons.org/licenses/by-sa/4.0/deed.nl>). Je mag dit document kopiëren, bewerken en hergebruiken, ook op je eigen school, als je de bron noemt (Guido van Dijk, LeX Consultancy B.V. en AI Netwerk Limburg, <https://git.lexconsultancy.nl/guido/dify-handleidingen>) en je bewerking onder dezelfde licentie deelt. Het logo van AI Netwerk Limburg valt buiten de licentie; namen van producten zijn merken van hun eigenaren. Scholen, personen en documenten in de voorbeelden zijn verzonnen.

Eigenaar en contact: Guido van Dijk, LeX Consultancy B.V. en AI Netwerk Limburg · guido@l-e-x.nl · guido@ai-netwerk-limburg.nl



AI Netwerk Limburg · LeX Consultancy B.V. · Dify zelf hosten: de Community Edition op je eigen computer of server