

Prerequisites

1. **Python 3.11+:** Check with `python --version`.
2. **Node.js 16+:** Check with `node --version`.
3. **Git:** Check with `git --version`.

Step-by-Step Setup

Step 1: Extract and Navigate

1. Extract the provided zip file to your desired location (e.g., `C:\Users\YourName\Desktop\integrations_technical_assessment`).
2. Open **Command Prompt** and navigate to the extracted directory.
Bash
`cd C:\Users\YourName\Desktop\integrations_technical_assessment`

Step 2: Set Up WSL and Redis

1. Open **WSL** (Windows Subsystem for Linux) by pressing Windows + R, typing `wsl`, and pressing Enter.
2. In the WSL terminal, run the following commands to install and start Redis:
Bash
`sudo apt update`
`sudo apt install -y redis-server`
`sudo service redis-server start`
`sudo systemctl enable redis-server`
3. Test Redis to ensure it's running:
Bash
`redis-cli ping`
Expected output: PONG

Step 3: Backend Setup

1. Open a **new Command Prompt** window and navigate to the backend directory.

Bash

```
cd C:\Users\YourName\Desktop\integrations_technical_assessment\backend
```

2. Create a Python virtual environment:

Bash

```
python -m venv .env
```

3. Activate the virtual environment:

Bash

```
.env\Scripts\activate
```

(You should see (.env) at the start of your command line.)

4. Upgrade pip and install the required dependencies:

Bash

```
python -m pip install --upgrade pip
```

```
pip install -r requirements.txt
```

Step 4: Create and Configure the .env File

1. Create a new .env file by copying the example file:

Bash

```
copy .env.example .env
```

2. Open the .env file with a text editor like Notepad:

Bash

```
notepad .env
```

3. Replace the placeholder values with your HubSpot credentials. Keep the Redis host and redirect URI as they are.

Code snippet

```
# HubSpot OAuth (REPLACE WITH YOUR VALUES)
```

```
HUBSPOT_CLIENT_ID=your_hubspot_client_id_here
```

```
HUBSPOT_CLIENT_SECRET=your_hubspot_client_secret_here  
HUBSPOT_REDIRECT_URI=http://localhost:8000/integrations/hubspot/oauth2callback
```

```
# Redis (keep as is)  
REDIS_HOST=localhost
```

Step 5: Start the Backend Server

1. Ensure your virtual environment is still activated ((.venv) is visible).
2. Load the environment variables.
Bash
`set -a; source .env; set +a`
3. Start the FastAPI server:
Bash
`uvicorn main:app --host 0.0.0.0 --port 8000 --reload`
4. The backend is running successfully when you see the following output:
Plaintext
INFO: Uvicorn running on http://0.0.0.0:8000
INFO: Application startup complete.

Step 6: Frontend Setup and Start

1. Open a **new Command Prompt** window. **Do not close the backend terminal.**
2. Navigate to the frontend directory:
Bash
`cd C:\Users\YourName\Desktop\integrations_technical_assessment\frontend`
3. Install the Node.js dependencies:
Bash
`npm install`
4. Start the React development server:
Bash
`npm start`

5. The frontend is running successfully when you see this output:

Plaintext

Local: http://localhost:3000

Get HubSpot Credentials

1. Go to the [HubSpot Developer Portal](#).
2. Click "Create app" and name it "My Integration App".
3. Navigate to the "Auth" tab.
4. Add the redirect URL: `http://localhost:8000/integrations/hubspot/oauth2callback`.
5. Enable the scope: `crm.objects.contacts.read`.
6. Click "Save changes".
7. Copy the "Client ID" and "Client Secret" and paste them into your backend/.env file.
8. Restart the backend server for the changes to take effect (press Ctrl+C and run the `uvicorn` command again).

Test the Application

1. Open your web browser and go to `http://localhost:3000`.
2. Select "HubSpot" from the dropdown and click "Connect to HubSpot".
3. Authorize the app in the pop-up window.
4. Click "Load Data" to view your contacts.

Troubleshooting

- **Redis Fails:** In your WSL terminal, run `sudo service redis-server start` and `redis-cli ping`.
- **Backend Dependencies Fail:** In the backend directory with the virtual environment activated, run `pip install -r requirements.txt --no-cache-dir`.
- **Frontend Fails:** In the frontend directory, run `npm cache clean --force`, delete `node_modules` and `package-lock.json`, then run `npm install`.
- **Port 8000 is Busy:** Use `netstat -ano | findstr :8000` to find the process ID and `taskkill /PID <PID_NUMBER> /F` to terminate it. Alternatively, start the backend on a different port:

```
uvicorn main:app --host 0.0.0.0 --port 8001 --reload.
```