Тестовый стенд – локальный сервер с kali linux. Подключение к серверу происходит через ssh по белому ip адресу.

Задачи:

1. Установка docker engine и docker compose
2. Написание yaml файла композа для запуска контейнеров с nginx любой версии и c couchdb версии 1.7.2

Оглавление

[1 Установка docker engine и docker compose 3](#_Toc204664973)

[2 Что такое couchdb 4](#_Toc204664974)

[3 Клонирование репозитория с github 4](#_Toc204664975)

[3 Написание нужного файла yaml compose и запуск контейнеров 4](#_Toc204664976)

## 1 Установка docker engine и docker compose

На моем локальном сервере уже установлены и функционируют docker engine и docker compose. Проверка установленных версий представлена на рисунке 1. Проверка запущенных docker контейнеров представлена на рисунке 2.

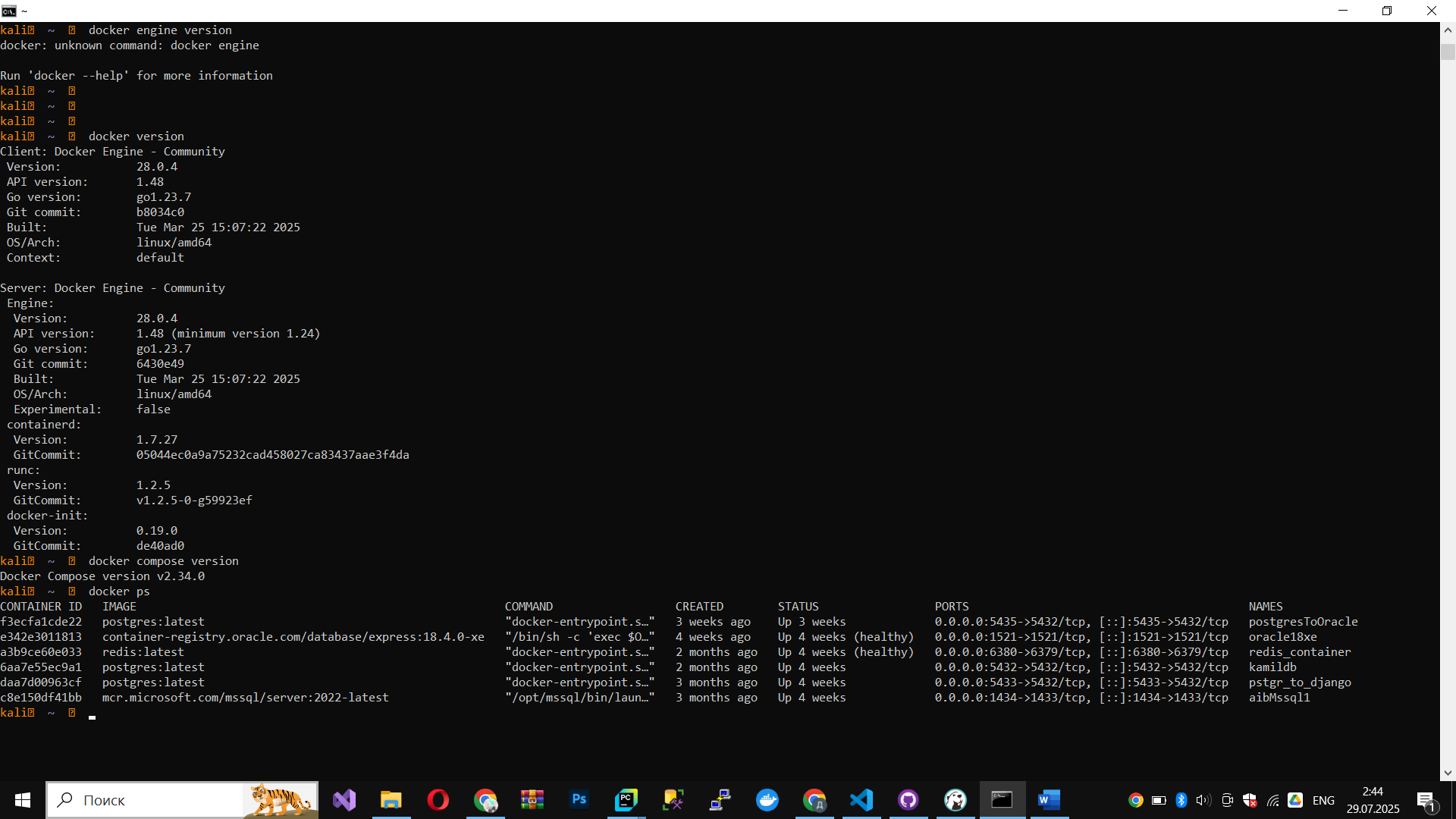


Рисунок 1 – Проверка установки docker engine и docker compose.

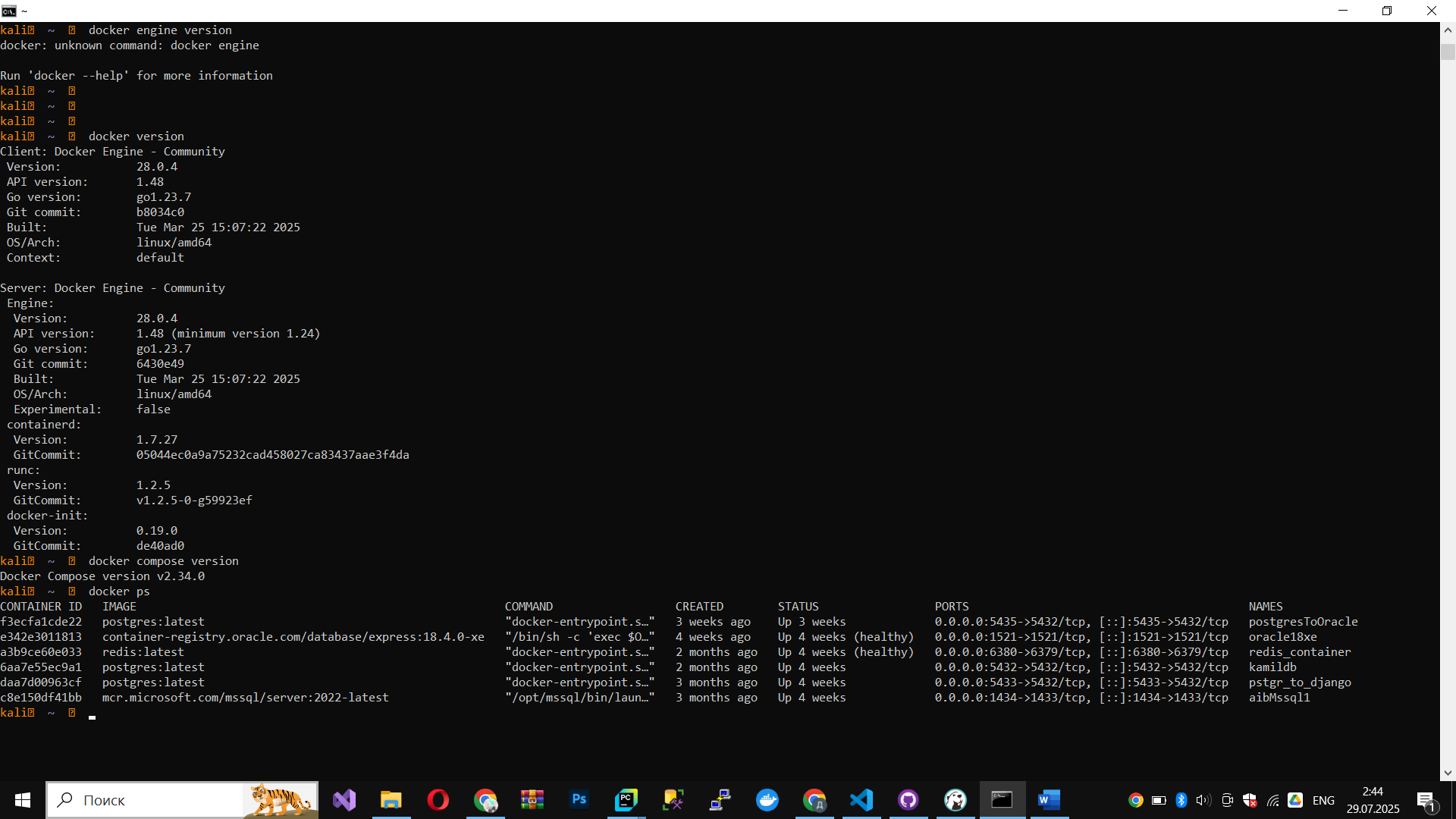


Рисунок 2 – Проверка запущенных docker контейнеров

В случае, если программы ранее не были установлены, их можно установить с помощью команд:

Для установки docker engine - «sudo apt-get update

sudo apt-get install docker.io

sudo systemctl enable --now docker»

Для установки docker compose: «sudo curl -L "https://github.com/docker/compose/releases/latest/download/docker-compose-$(uname -s)-$(uname -m)" -o /usr/local/bin/docker-compose

sudo chmod +x /usr/local/bin/docker-compose»

## 2 Что такое couchdb

Ранее я не имел опыт взаимодействия с этой программой, но судя по приписке “db”, это должна быть СУБД.

Apache CouchDB – свободно распространяемая СУБД с открытым исходным кодом. Тип – NoSql, документно-ориентированная. По описанию из сети, наводит на мысль о схожести с MongoDB. СУБД с низкой популярностью. Хранит данные в формате JSON. Не требует жесткой схемы данных – это позволяет работать с данными разнообразной структуры. Взаимодействие с данной СУБД происходит через Http запросы, из чего следует высокое удобство в веб-разработке.

Исходя из описания, можно выделить значительное преимущество данной СУБД – способность работать в условиях перебоев сети.

Вывод: СУБД, удобная в работе в тех приложений, в которых требуется периодическая с синхронизация данных, как к примеру мобильные приложения.

## 3 Клонирование репозитория с github

Для работы с этим проектом, я создал репозиторий github <https://github.com/Danil-512/Test-task-Devops>

Задача – клонировать его в директорию /home/danil/Desktop/

## 3 Написание нужного файла yaml compose и запуск контейнеров

Для хранения проекта и файла конфигурации yaml, создам директорию на рабочем столе kali. В нее клонирую репозиторий с Github. Клонирование репозитория представлено на рисунке 3.

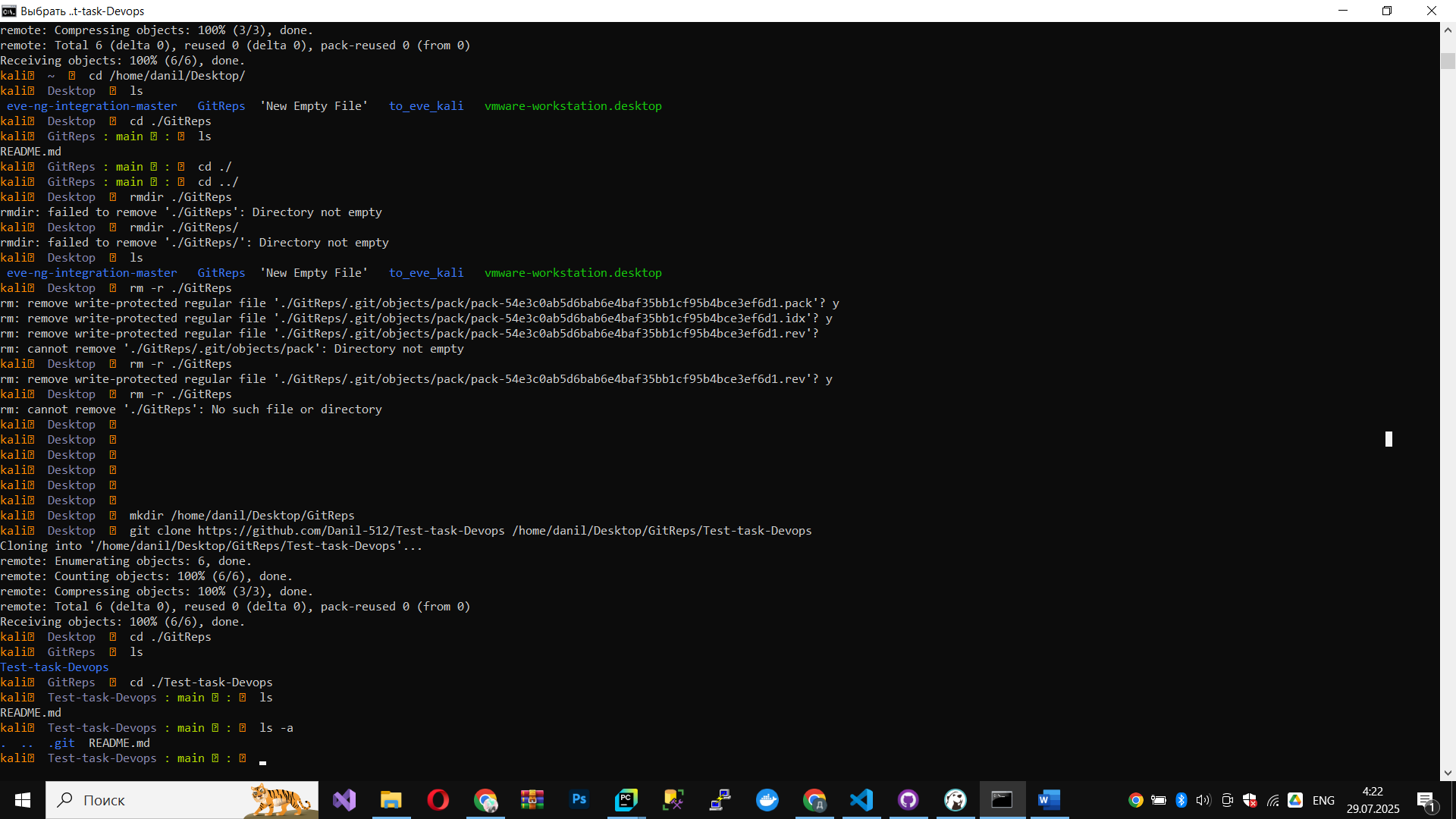


Рисунок 3 – Клонирование репозитория с Github

Далее необходимо создать файл конфигурации docker compose в формате yaml. Имя файла - docker-compose.yaml. Создание файла представлено на рисунке 4.

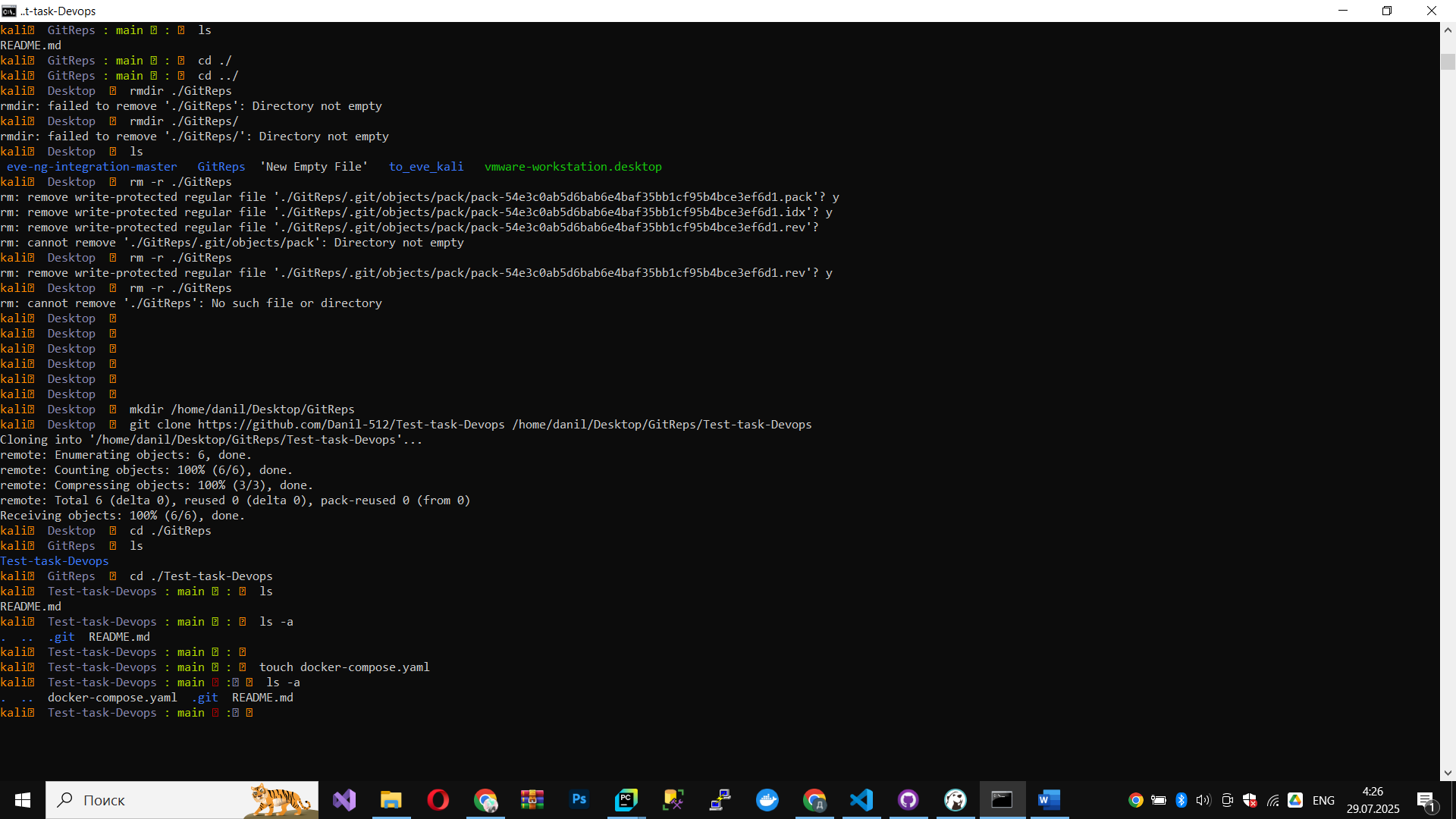


Рисунок 4 – Создание файла docker-compose.yaml

Файл конфигурации docker-compose представлен на рисунке 5 с подробным описанием каждой строки.

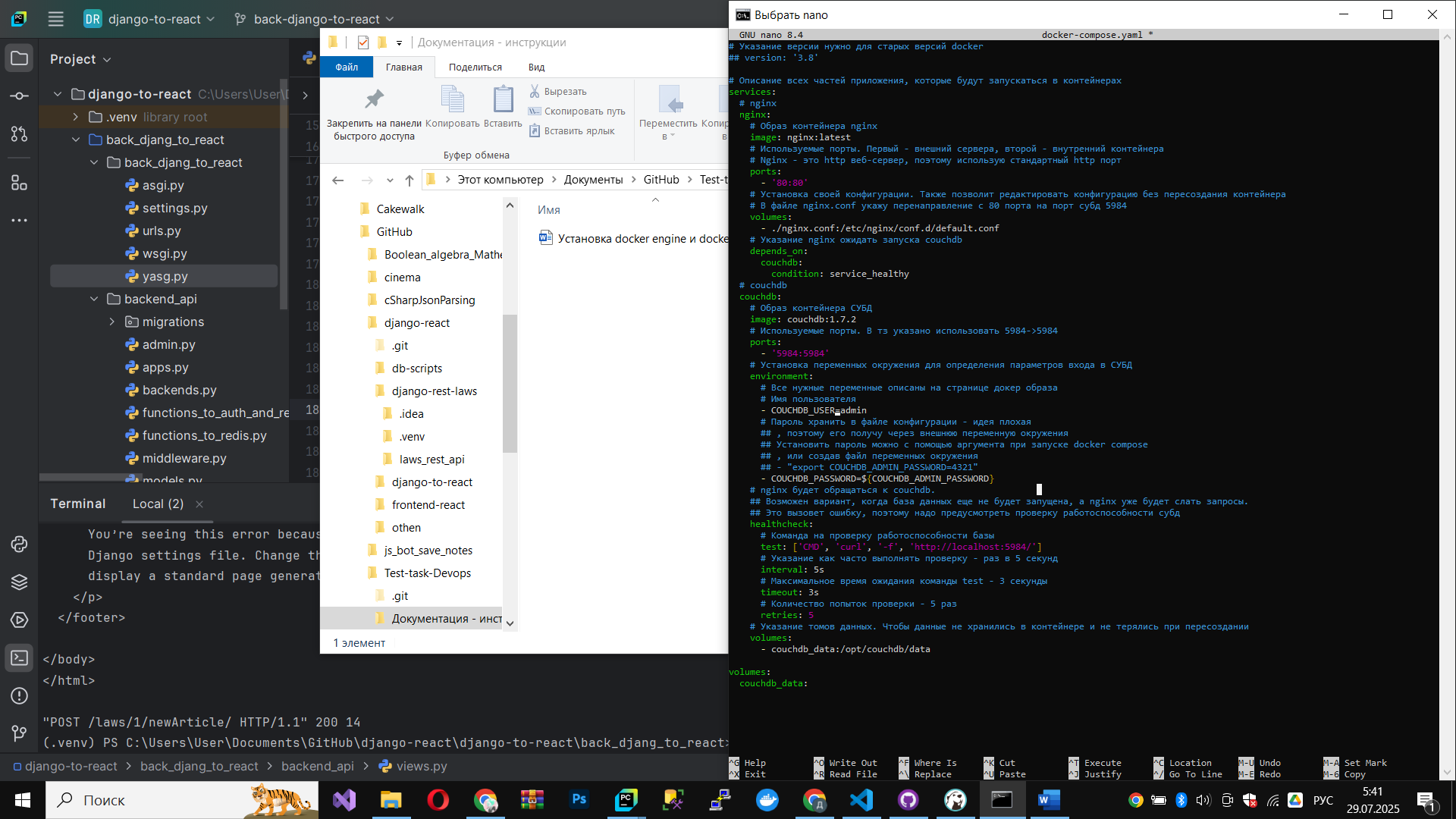


Рисунок 5 – файл docker-compose

Создание переменной окружения для пароля, проверка файла конфигурации compose и создание файла конфигурации nginx.conf представлено на рисунке 6

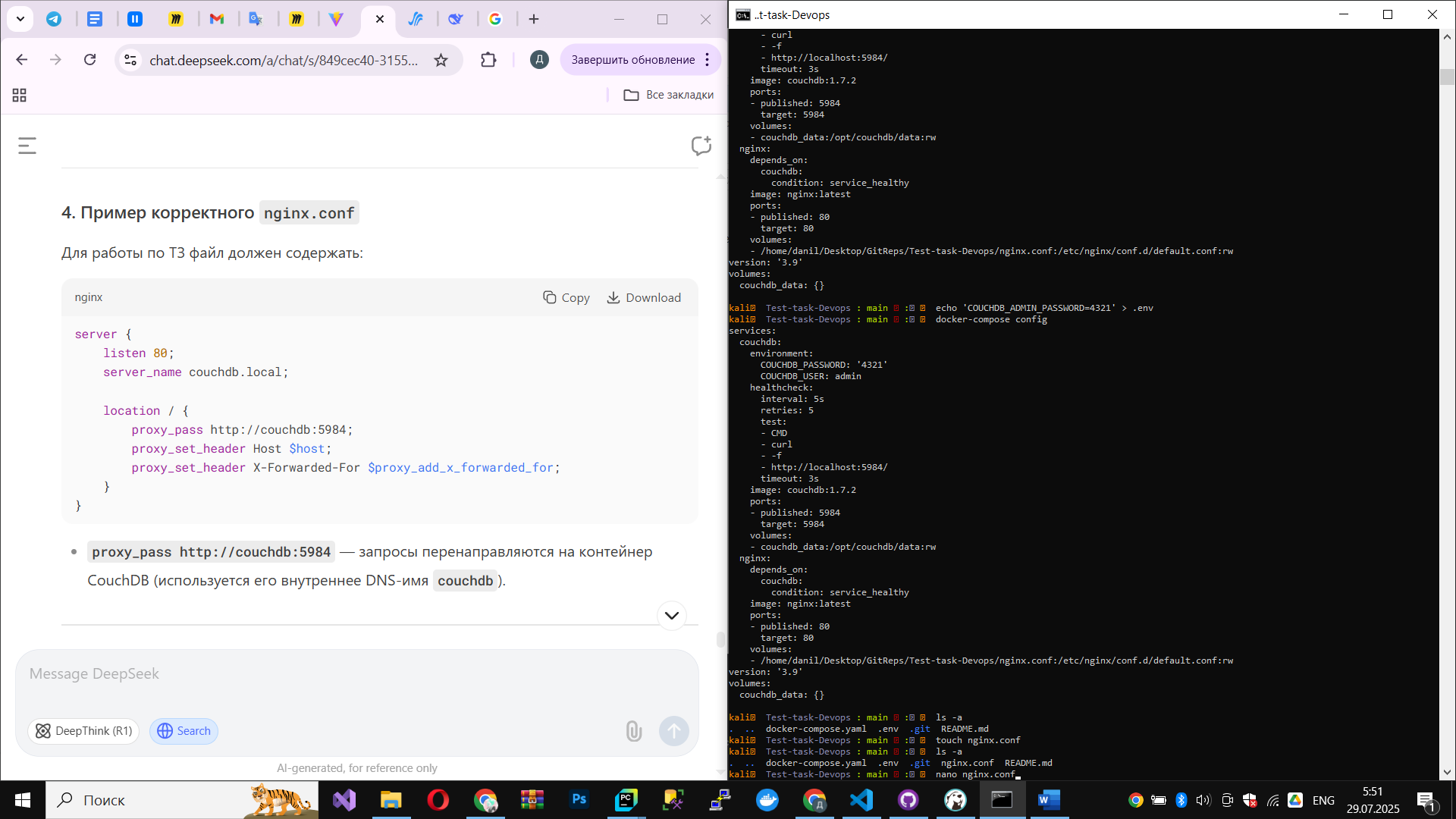


Рисунок 6 – Проверка конфигурации, создание переменной окружения

Далее необходимо настроить конфигурацию nginx в файле nginx.conf для перенаправления от порта 80 к порту 5984. Настройка nginx.conf представлена на рисунке 7.

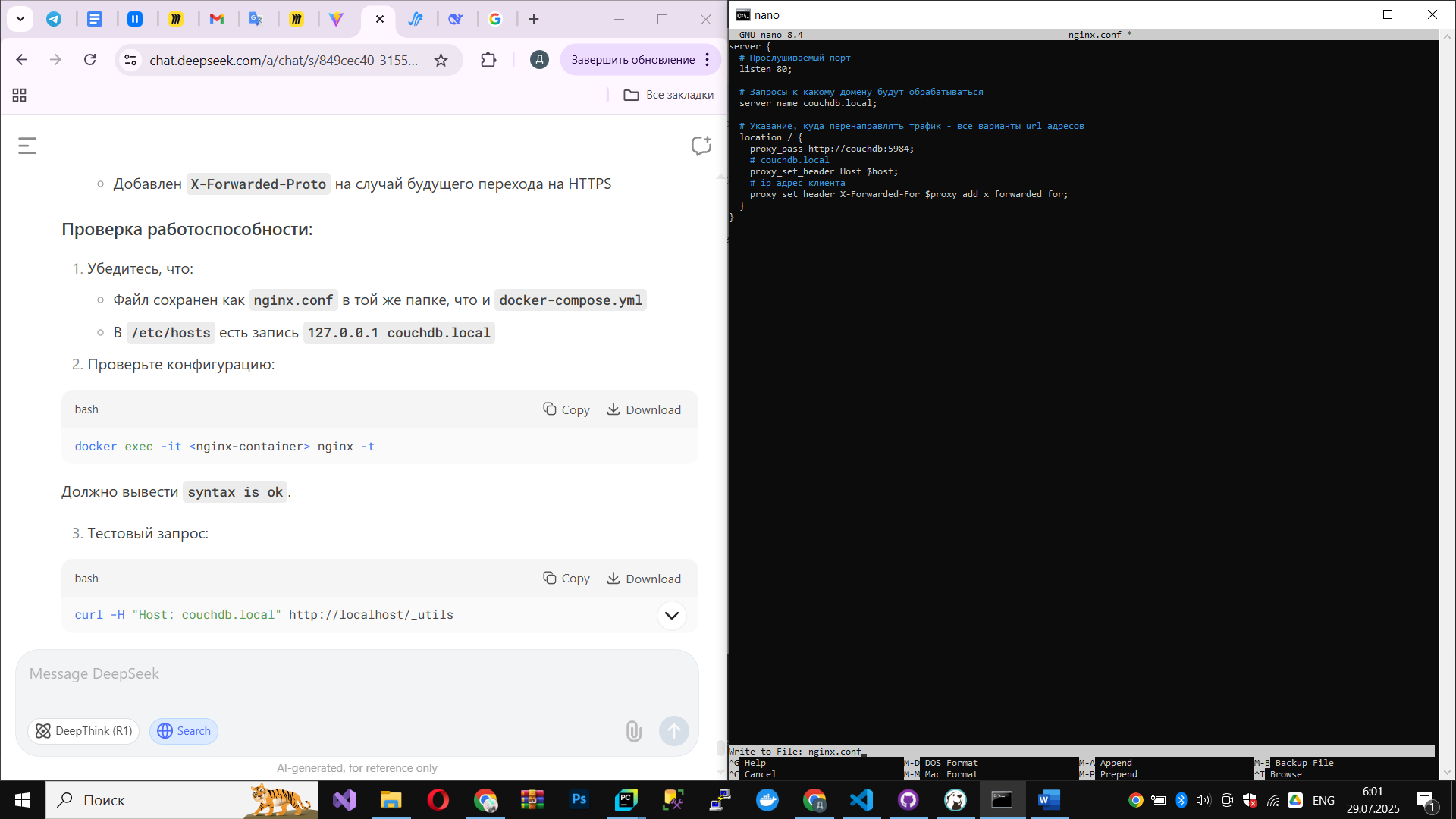


Рисунок 7 – Файл конфигурации nginx.conf

Далее на сервере необходимо прописать в системный файл hosts доменное имя couchdb.local. Это требуется по тз и без этой настройки, адрес couchdb.local будет запрашиваться из внешней сети. Данная настройка представлена на рисунке 8.

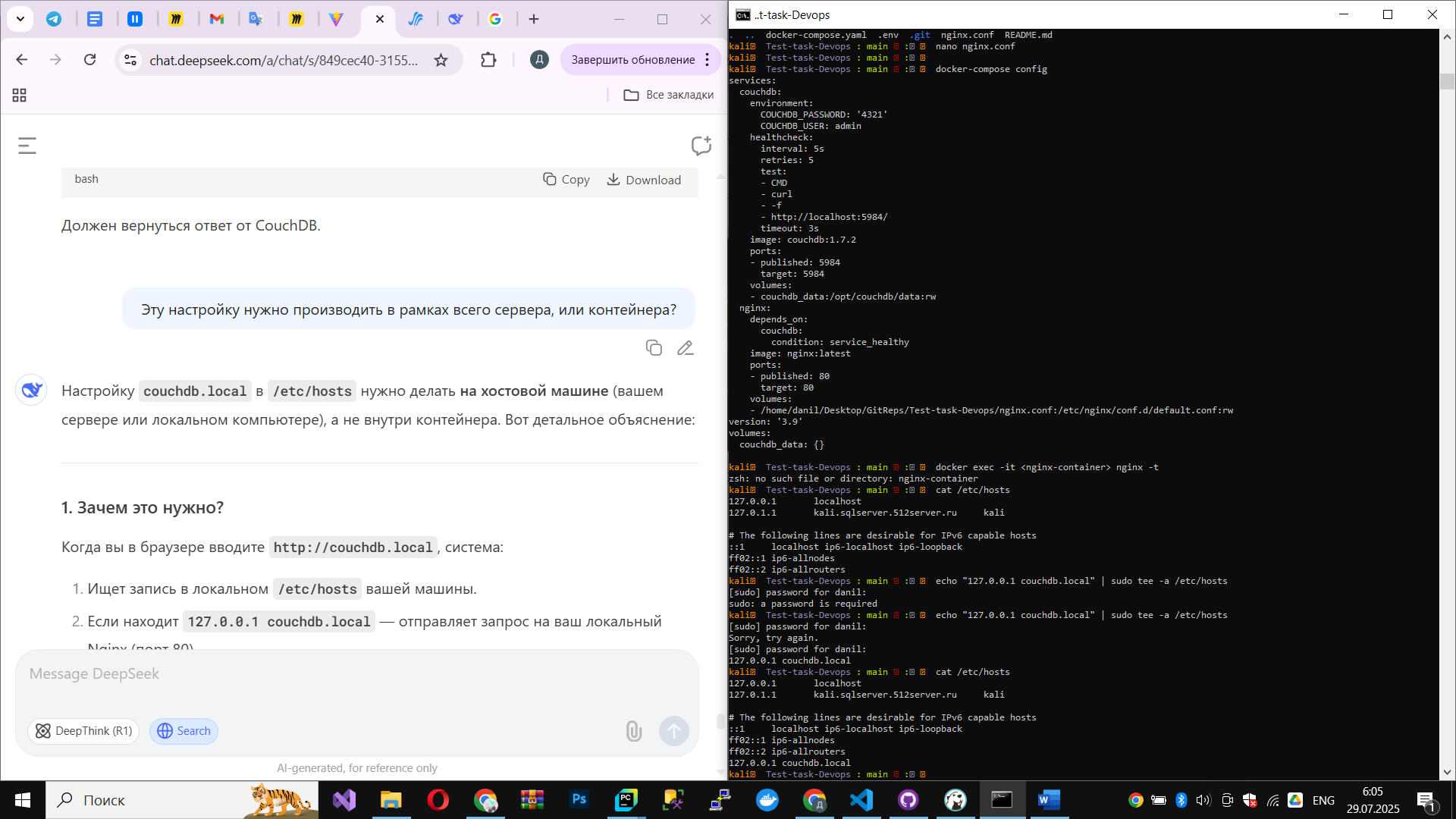


Рисунок 8 – Добавление доменного имени в hosts

Все настройки произведены, можно запускать контейнеры. Запуск происходит с помощью команды «docker compose up --build». Запуск представлен на рисунке 9.

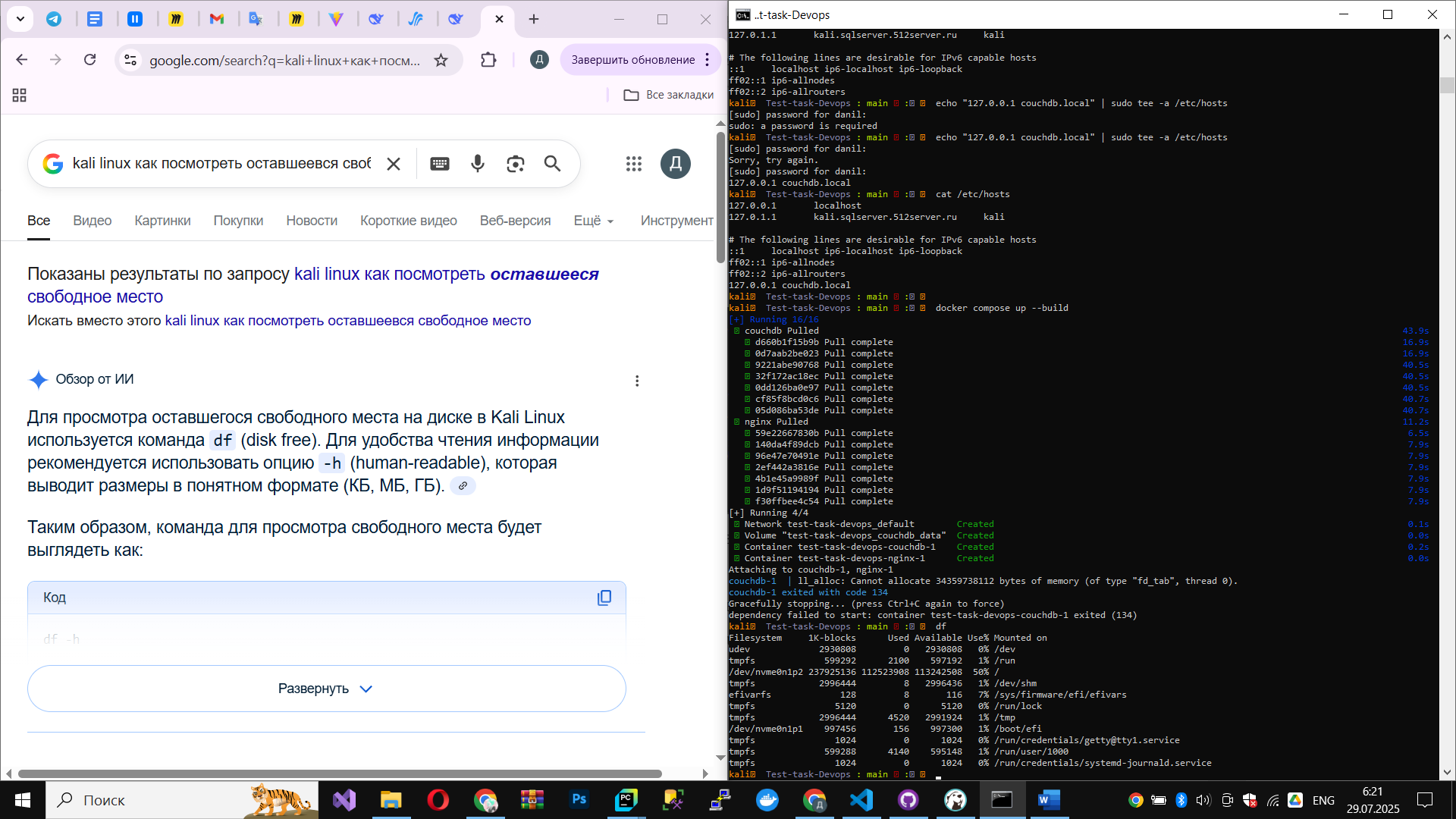


Рисунок 9 – Неудачный запуск контейнеров

Запуск не удался. Была получена ошибка о недостатке места. CouchDb запрашивает 32 гб свободного места, сервер такое пространство предоставить не может. Выход – указать ограничение места в файле конфигурации docker compose. Измененный файл конфигурации представлен на рисунке 10.

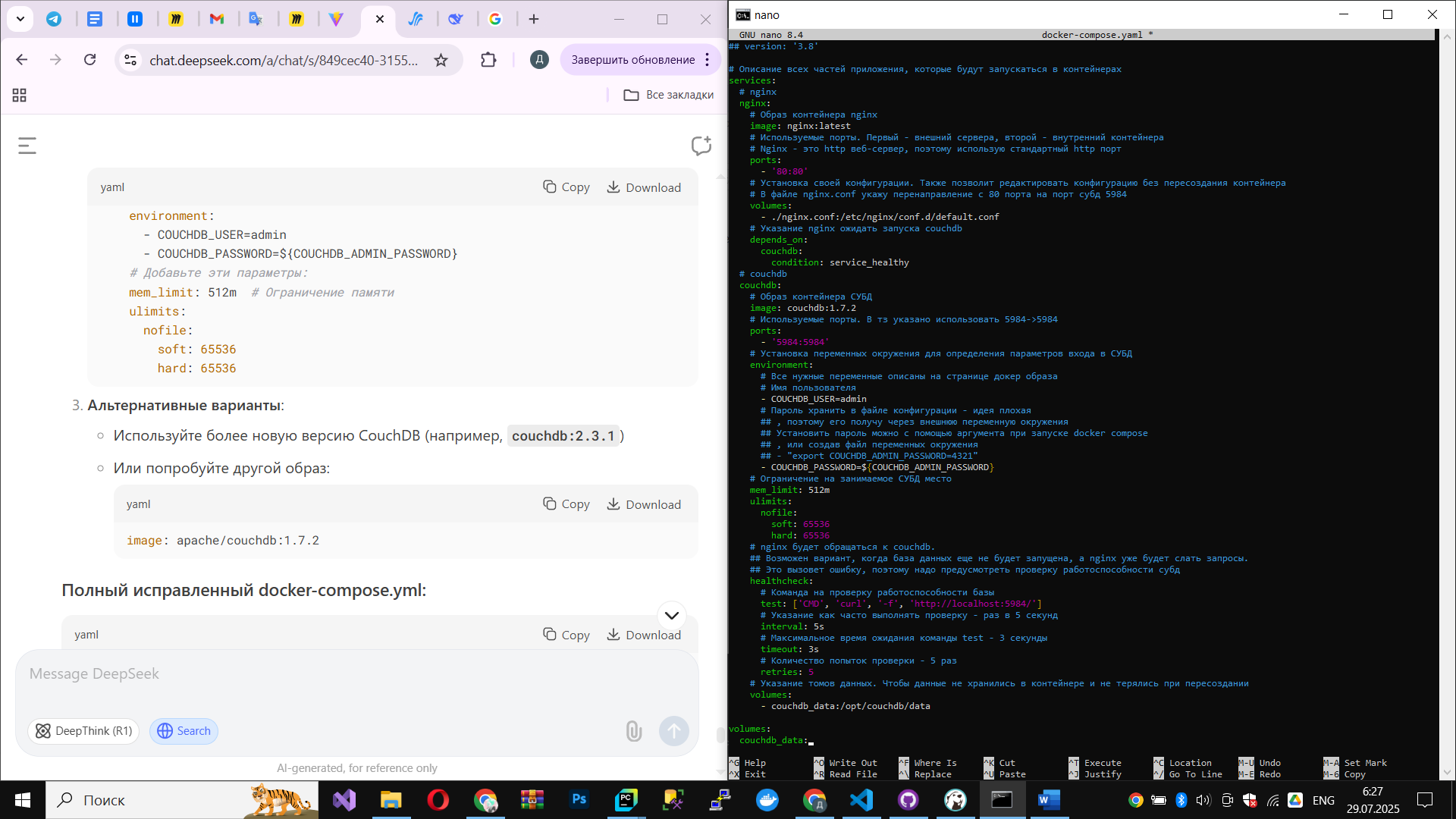


Рисунок 10 – Измененный файл конфигурации с ограничением места для СУБД

Далее с помощью запуска сборки контейнеров через «docker-compose down -v && docker-compose up --build», получилось успешно запустить docker-compose. Работа контейнеров представлена на рисунке 11.

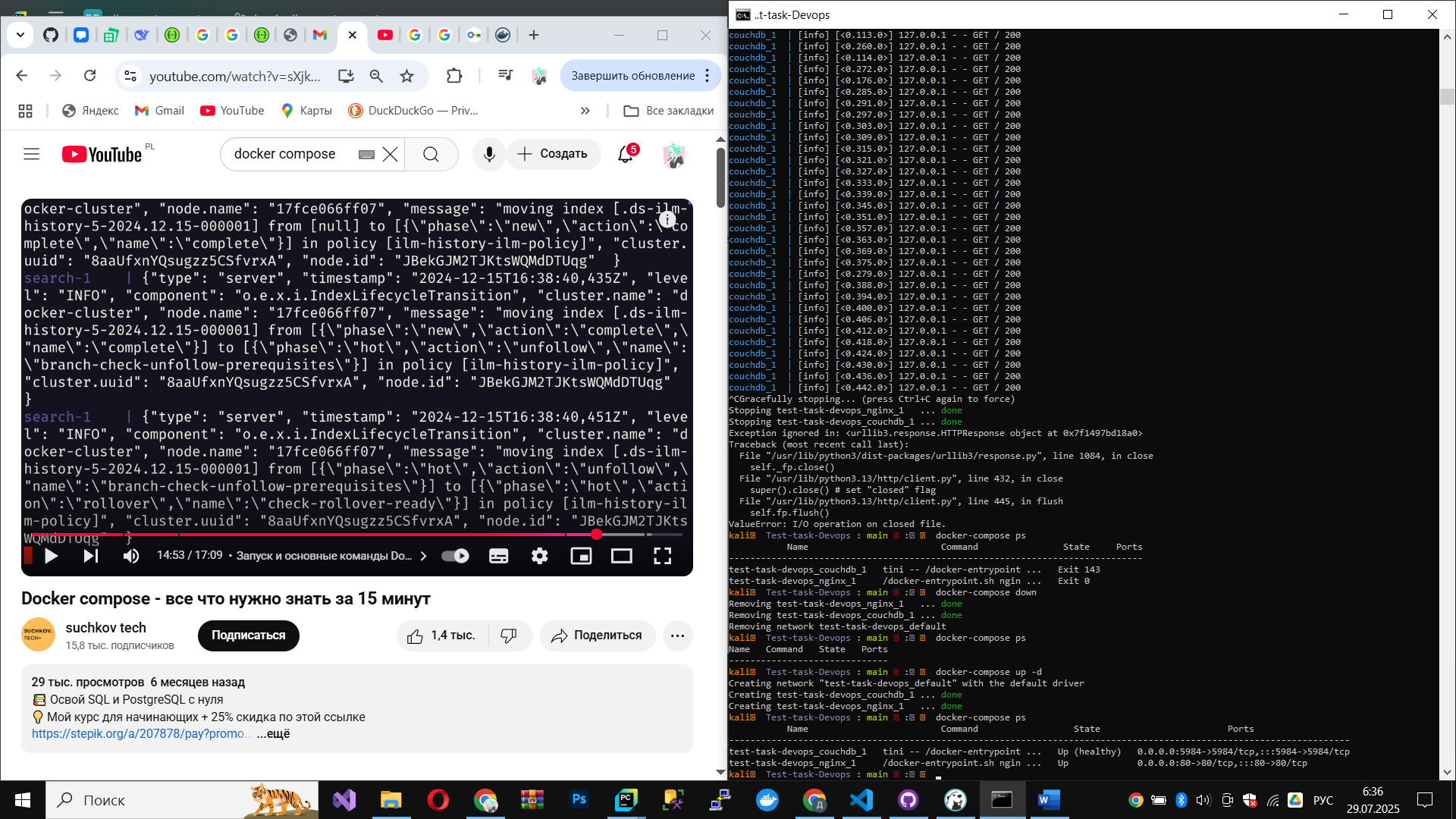


Рисунок 11 – Успешно запущенные контейнеры.

Проверка обращения по адресу couchdb.local представлена на рисунке 12.

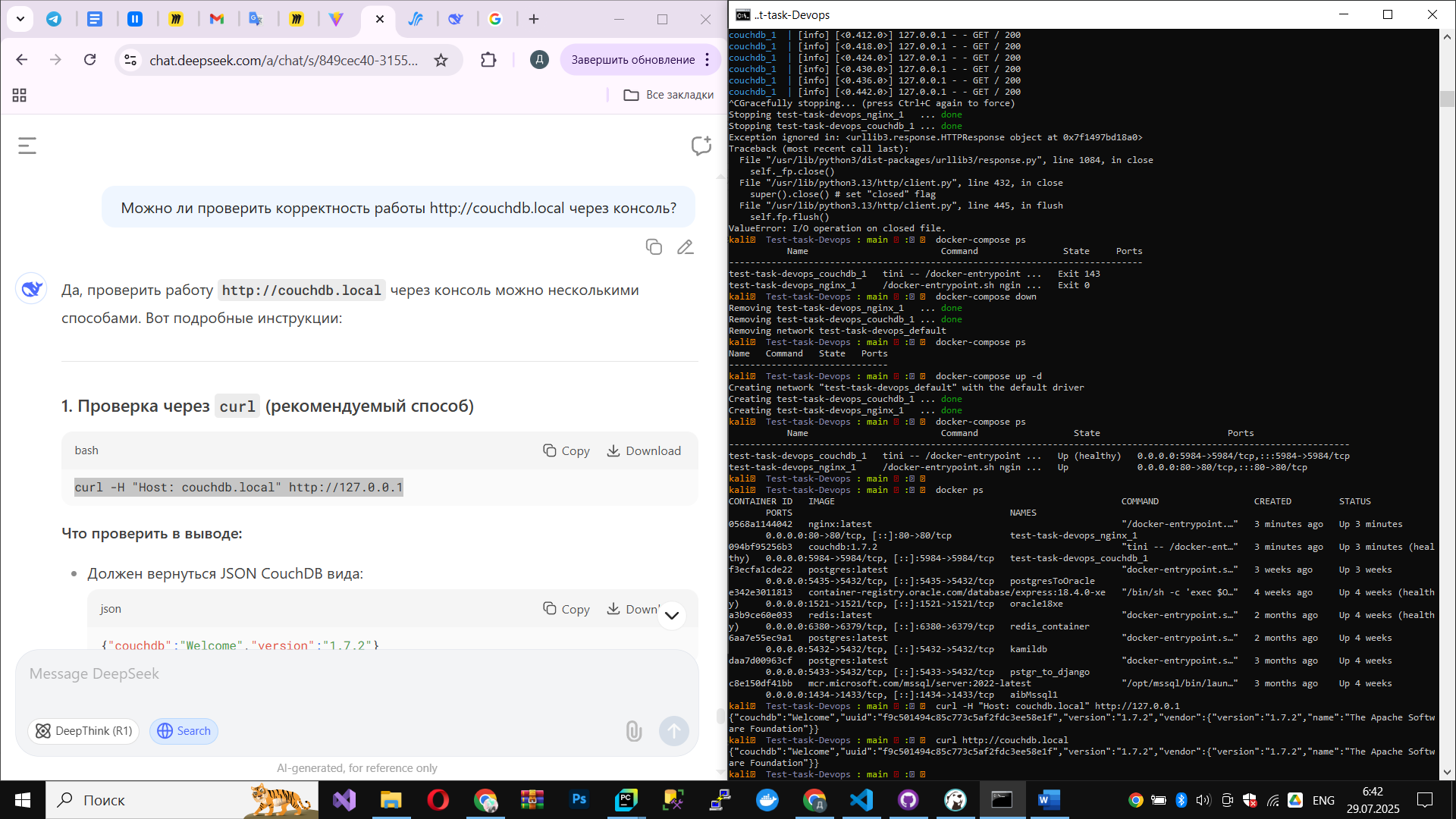


Рисунок 12 – Проверка работы СУБД и nginx

Исходя из консольного вывода на рисунке 12, можно сделать вывод, что контейнеры с couchdb и nginx успешно запущены и работают, nginx перенаправляет трафик с порта 80 на порт 5984 СУБД, СУБД дает корректный ответ.