Asynchronous Web Chat Project

Ayman Makki

1 Introduction

1.1 Description

The goal of this project is the automatization of the installation process of a complex web server application with the knowledge you have acquired during the labs. The web application includes multiple features : login, public chat room, notifications, friend system and private chats for friends. It is implemented in Python using the Django framework and derived from an implementation on the Github of "mitchtabian" in the "Codingwithmitch-Chat" repository.

In order to do so, there is no need to be aware of the python implementation of the web application. However, you should know that since it is using asynchronous communications, it handles differently basic HTTP requests and Web Sockets requests. This allows chats and notifications to be pushed to the user in an efficient manner without requiring nor a POST request or a refresh from the browser.

Therefore, the web application has two possibilities to handle requests from the user. By using the WSGI server (gunicorn), it answers to usual HTTP requests such as a GET request for a new page or a POST request for logging in. But it can also use the ASGI server (daphne) which handles requests and answers asynchronously to allow sudden push of informations to the user without a request from the user for new data (useful for notifications).

The forwarding of requests to both servers is ensured by Nginx. In a nutshell, here are the different services that we will need to set up at the start of our servers :

1. Nginx to forward requests to servers
2. PostGreSQL as the database software
3. The WSGI server
    - Django as the central web application
    - Gunicorn as the WSGI server to handle HTTP requests

4. The ASGI server

   - Daphne as the ASGI server to handle Web Sockets communications
   - Redis and channels to hangle Web Sockets communications with Django

1.2 Representation

To facilitate a global perspective upon the architecture of the web application, you can find in Figure 1 a visual representation.

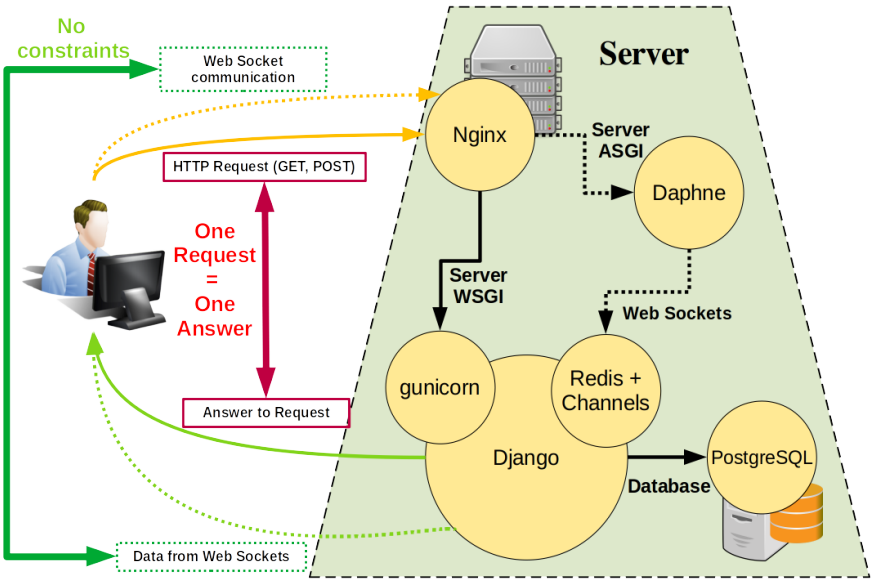

Figure 1: Representation of the web application