Національний технічний університет України

«Київський політехнічний інститут імені Ігоря Сікорського»

Факультет інформатики та обчислювальної техніки

Кафедра автоматики та управління в технічних системах

Лабораторна робота №1

з дисципліни «Проектування розподілених систем»

за темою «Розгортання веб-порталу ввикористовуючи підхід IaaS»

**Виконав:**

Студент групи ІA-11мн

Новиков Данило Михайлович

**Перевірив:**

доц. Волокина Артем Миколайович

Київ 2022

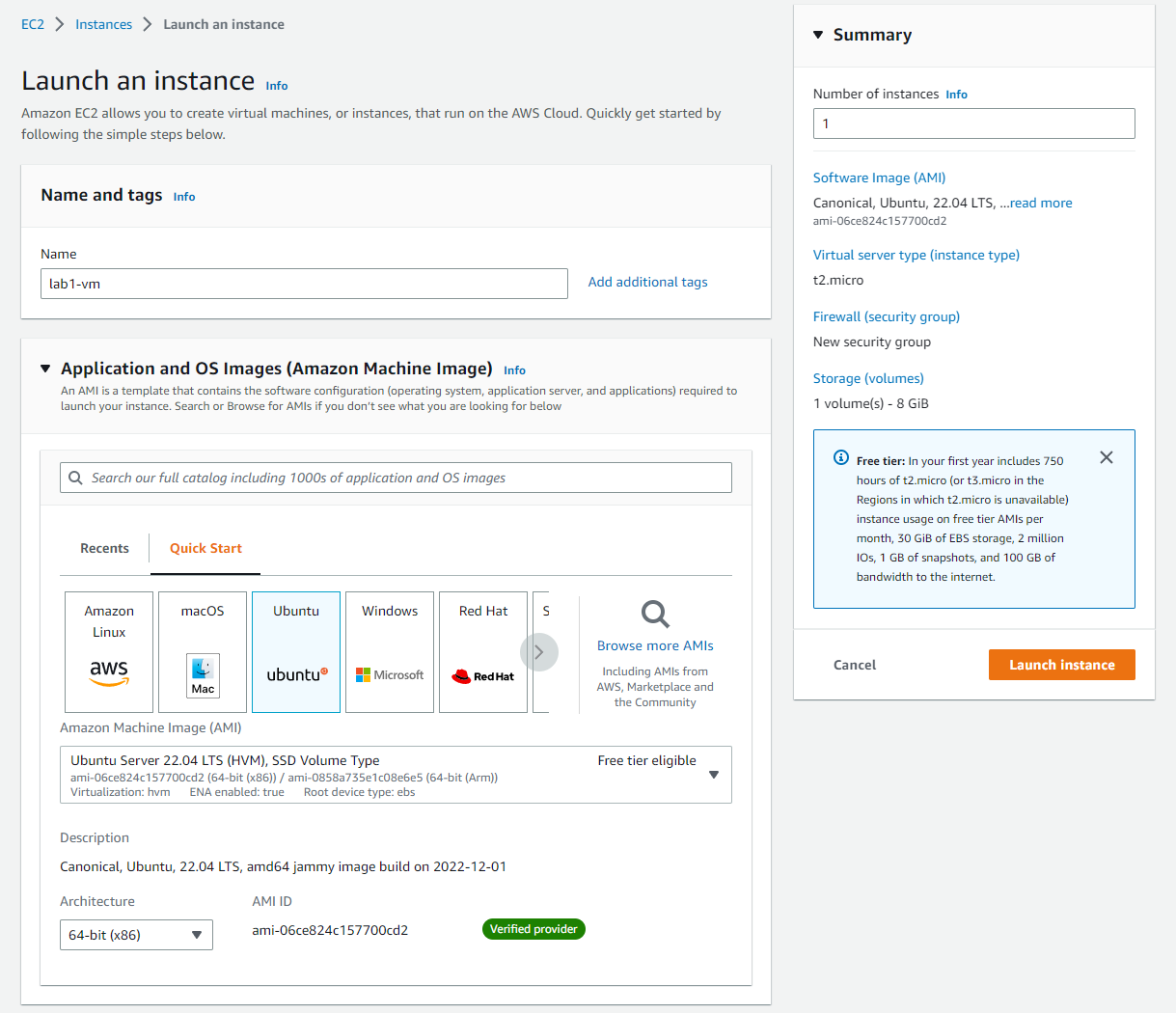
**Тема:** Розгортання веб-порталу ввикористовуючи підхід IaaS.

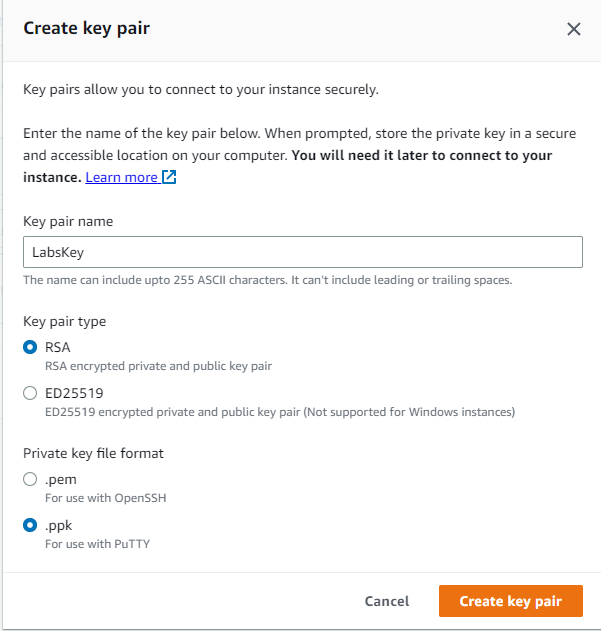
**Завдання:**

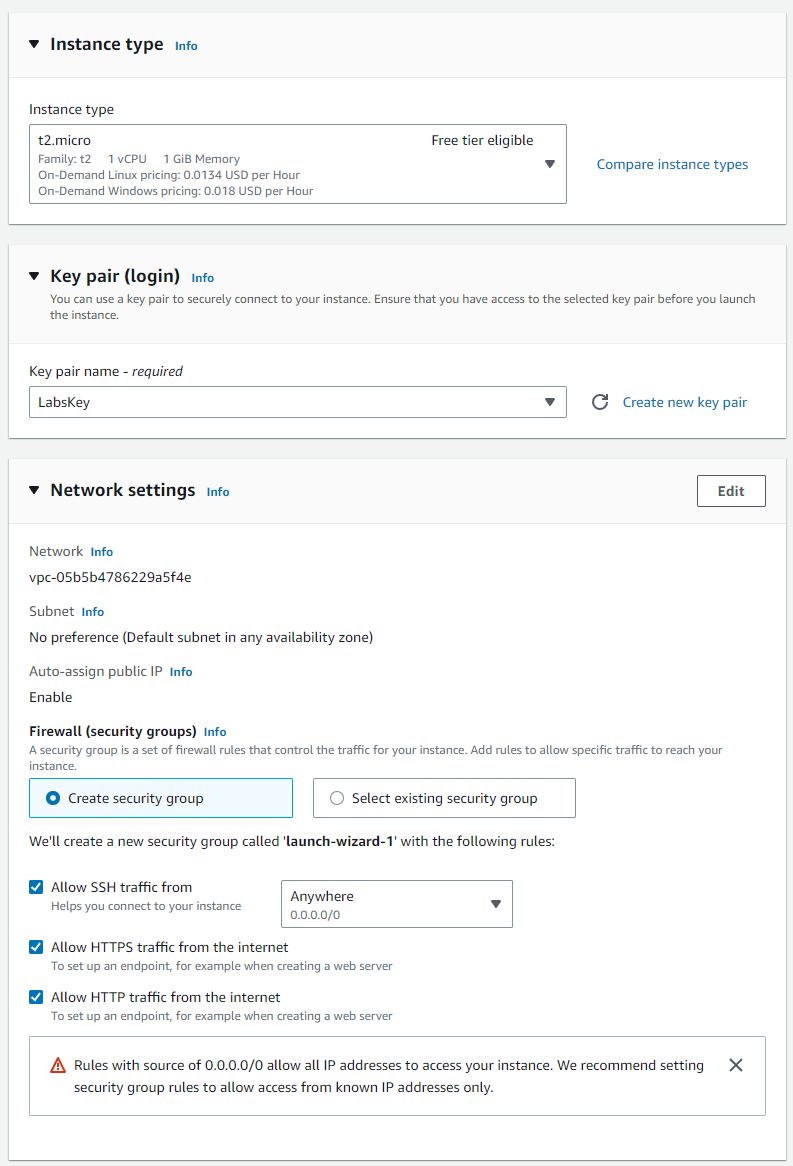
* Застосунок має бути розгорнутим в Azure Cloud або в Digital Ocean на віртуальному сервері (VPS)
* Застосунок має розгортатися за допомогою копіювання скомпільованого коду на сервер за допомогою SFTP або SSH

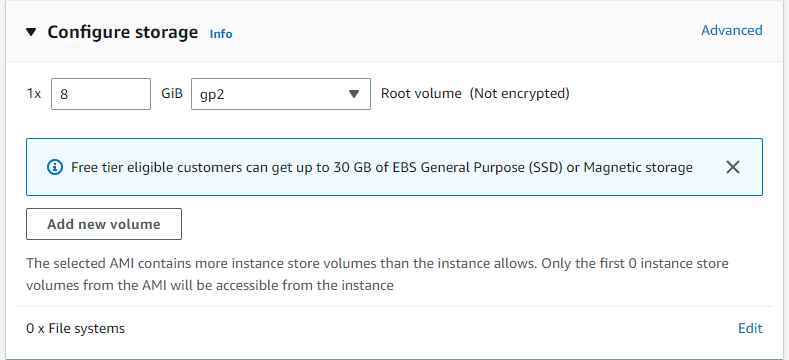
**Хід роботи**

Застосунок побудовано на .NET 6 з використанням бази даних PostgreSQL. Тому розгортання буде поділено на 2 етапа – сетап віртуальної машини («EC2») для API та налаштування БД в AWS («Amazon RDS»).

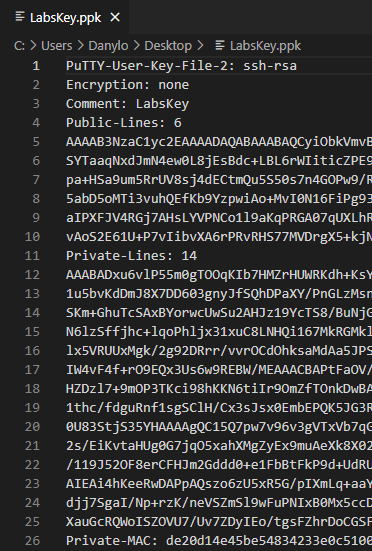
1. По-перше, була створена віртуальна машина зі стандартними налаштуваннями, використовуючи Ubuntu. Для неї був створений SSH ключ та нова Security Group з можливістю доступу по SSH: 



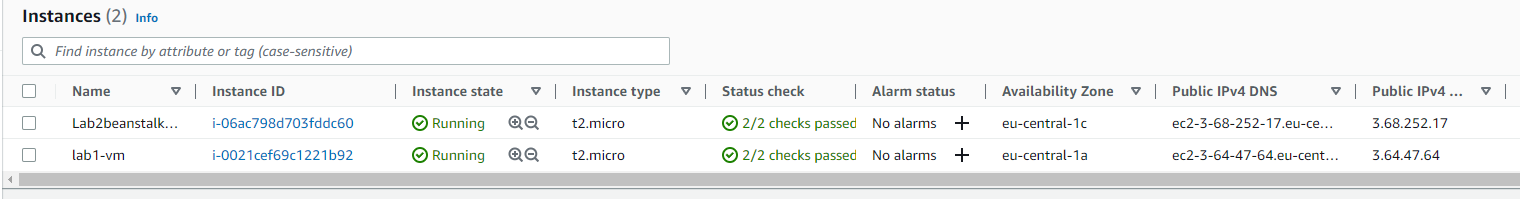


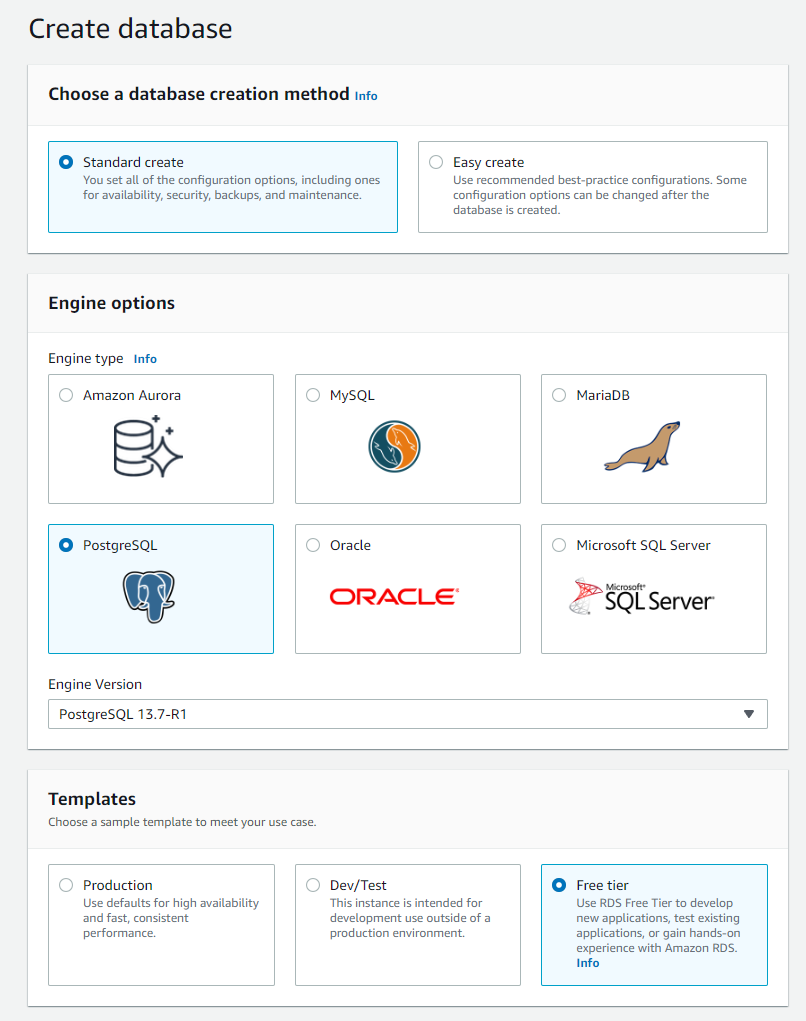
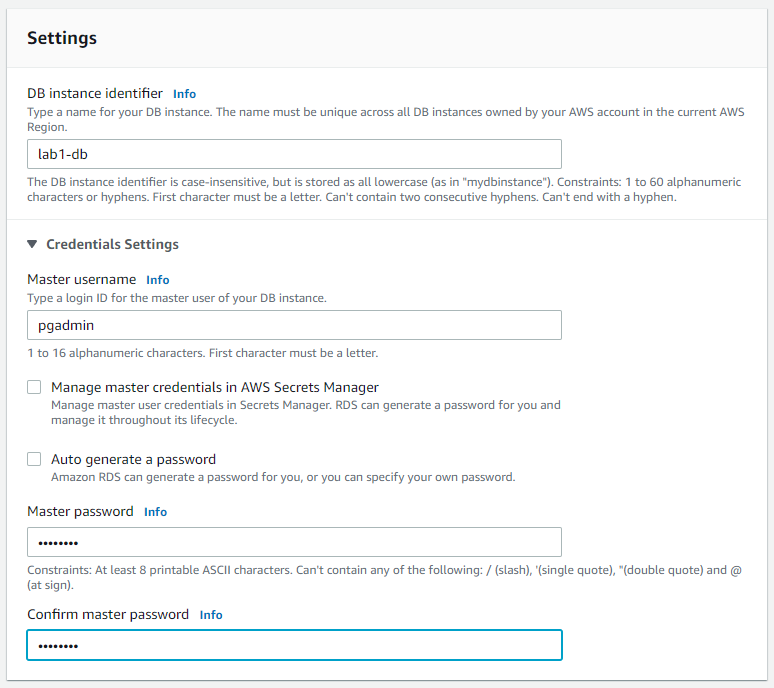
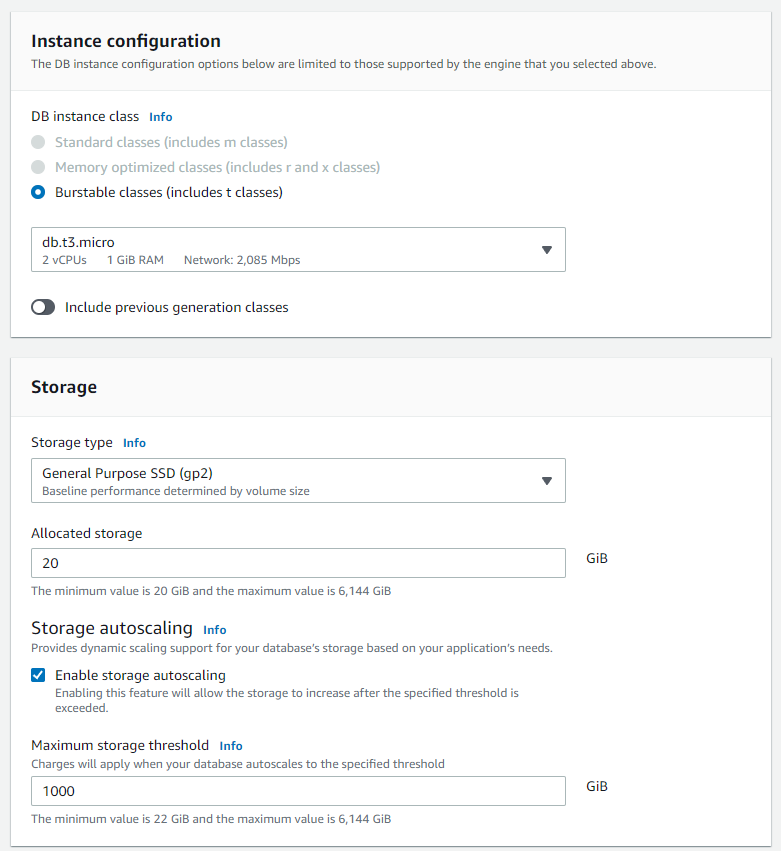
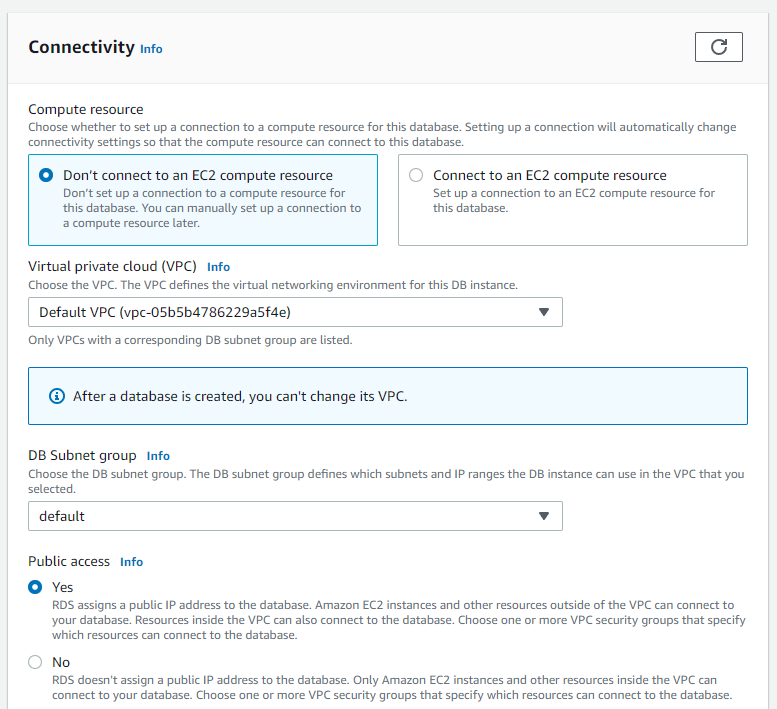
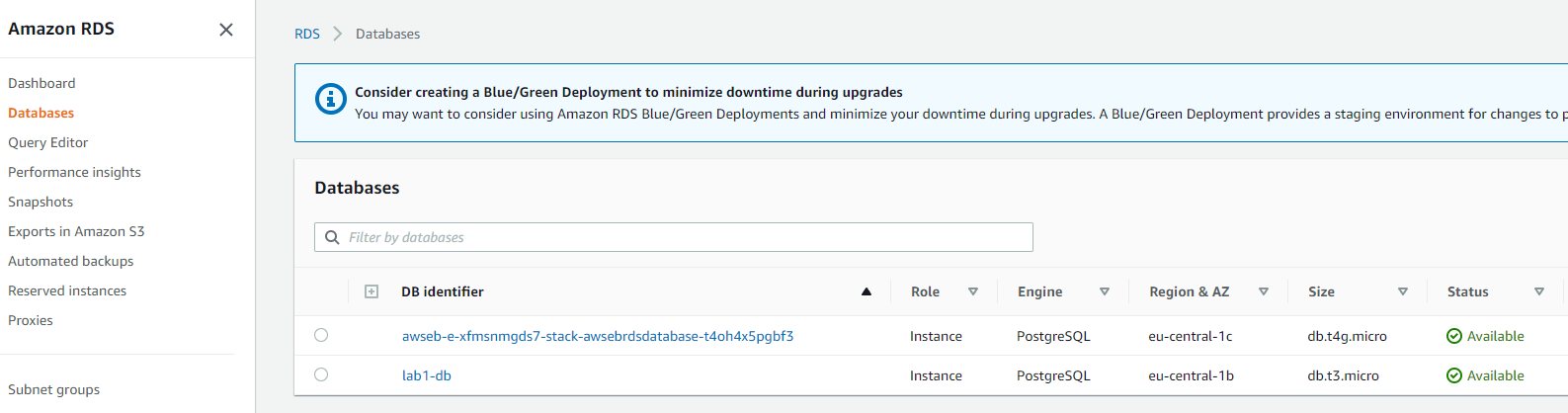
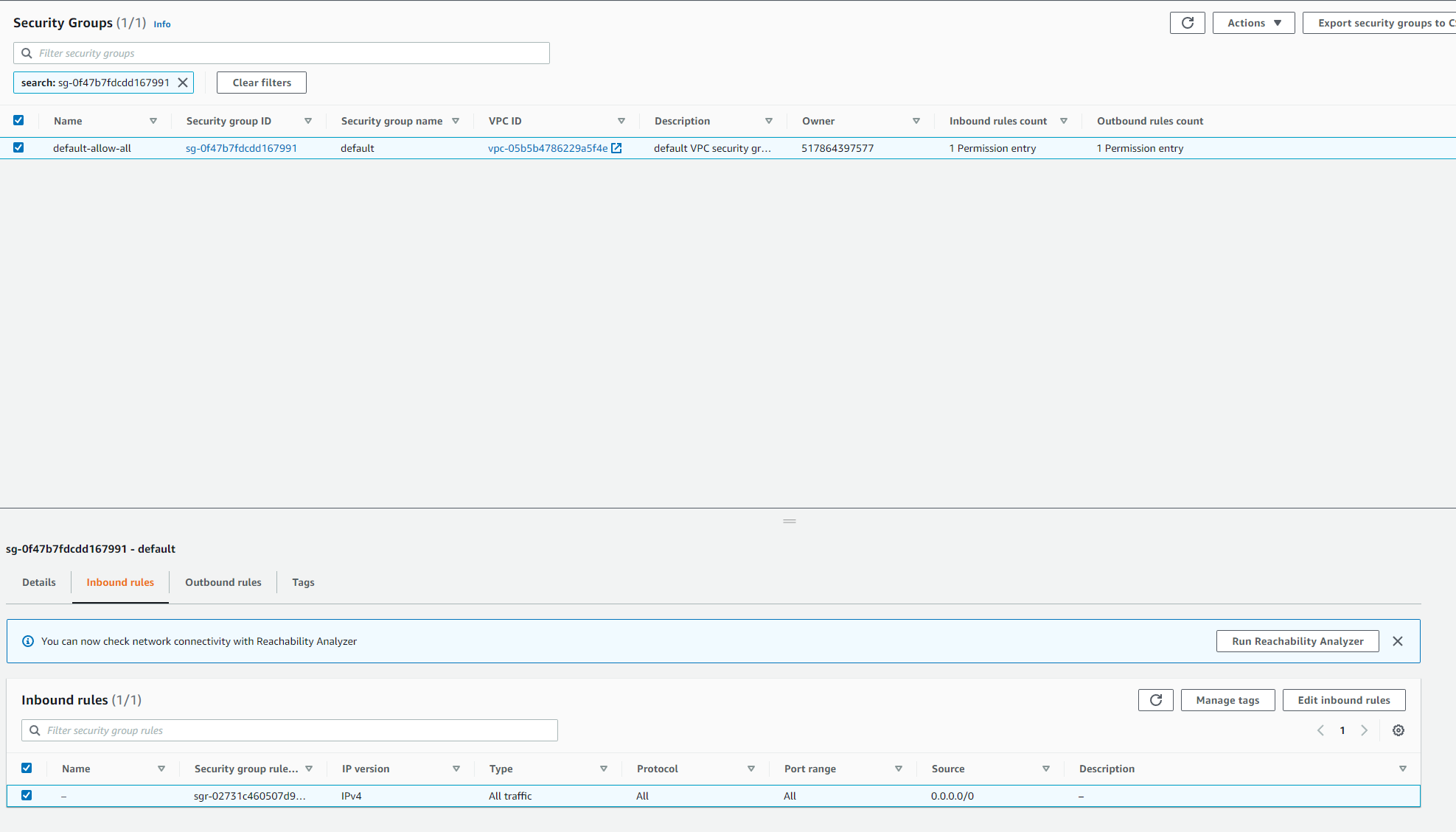
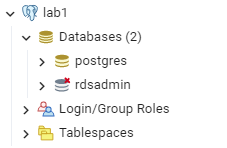
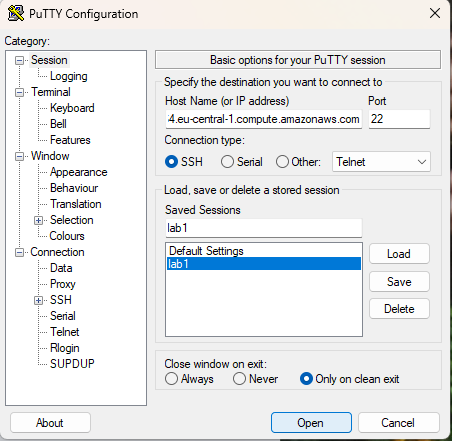
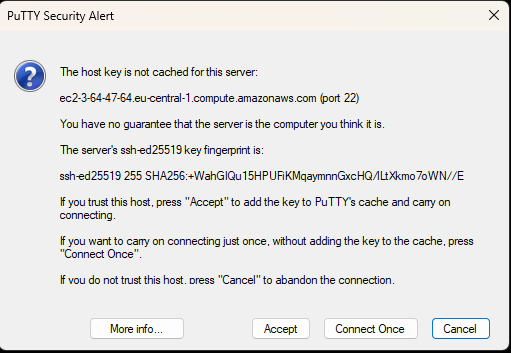
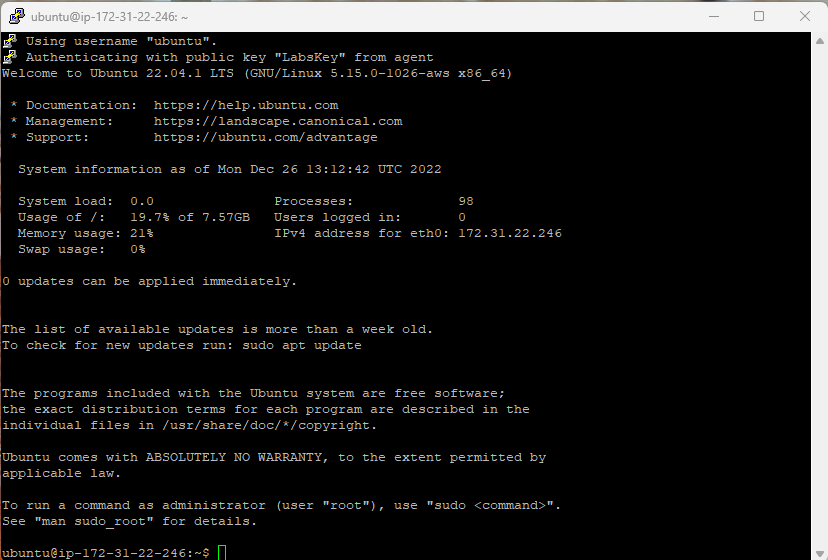


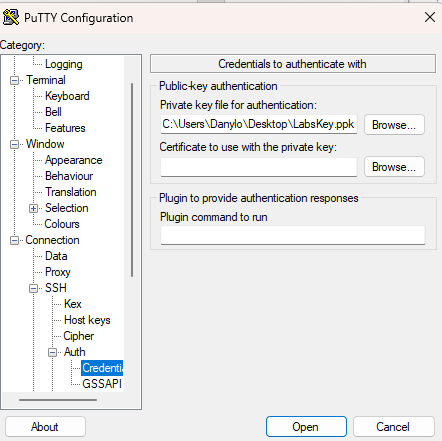
1. Дані файлу зі згенерованим SSH ключем:

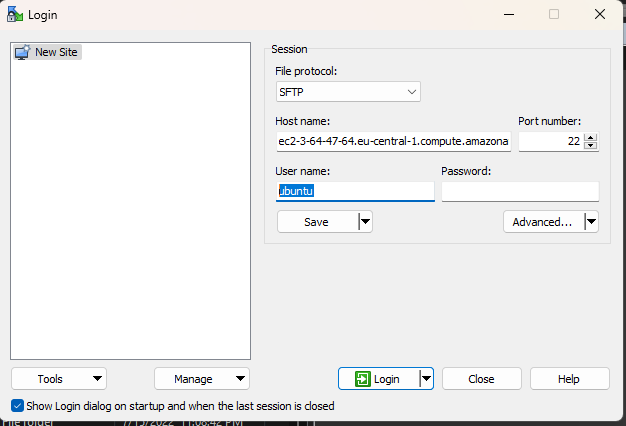


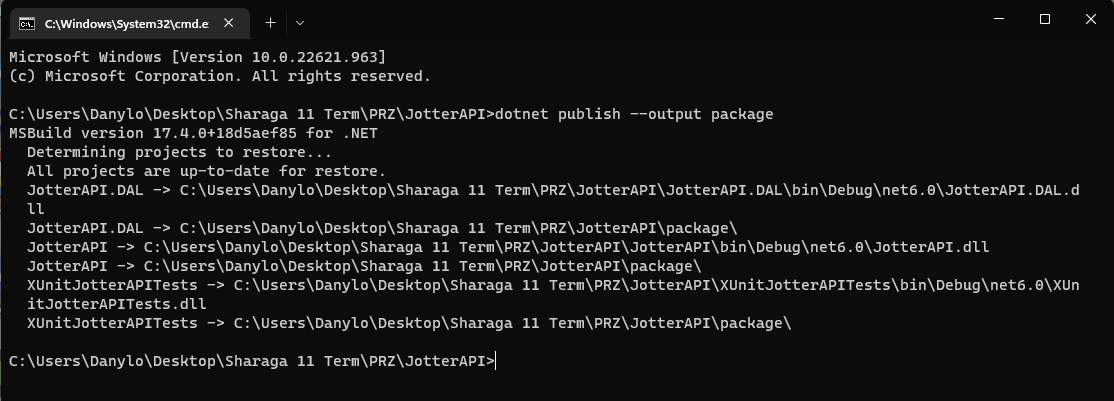
1. Ініціалізація створення віртуальної машини:

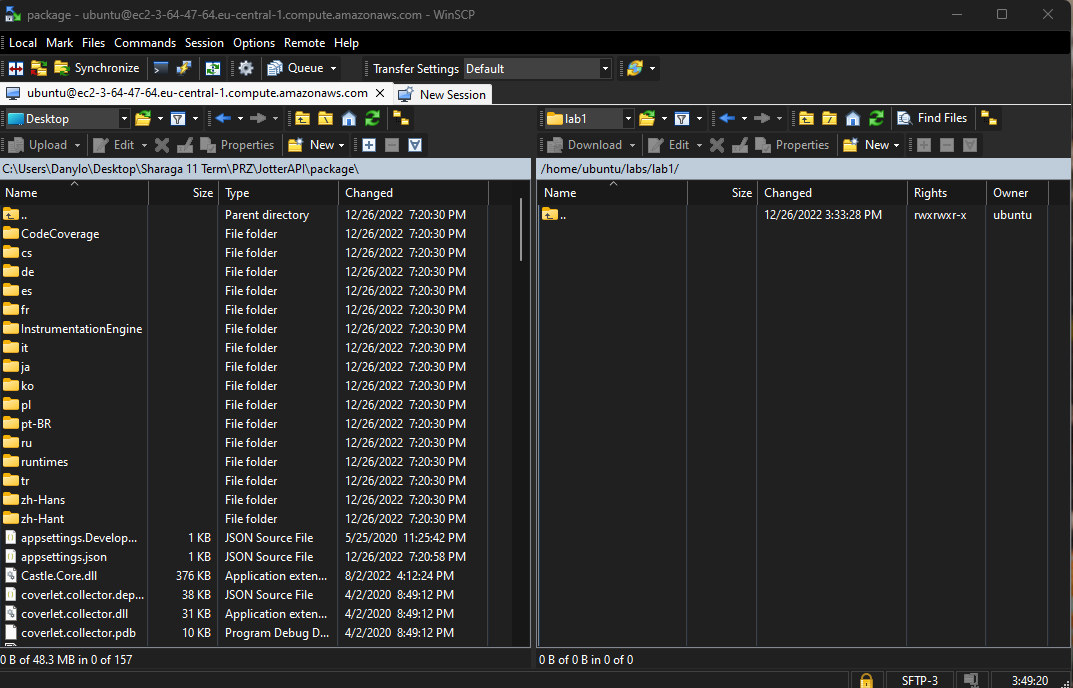


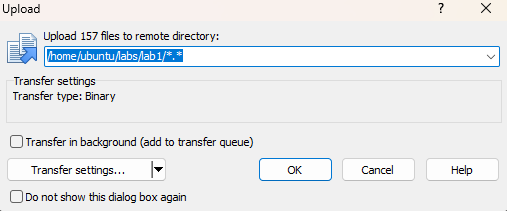
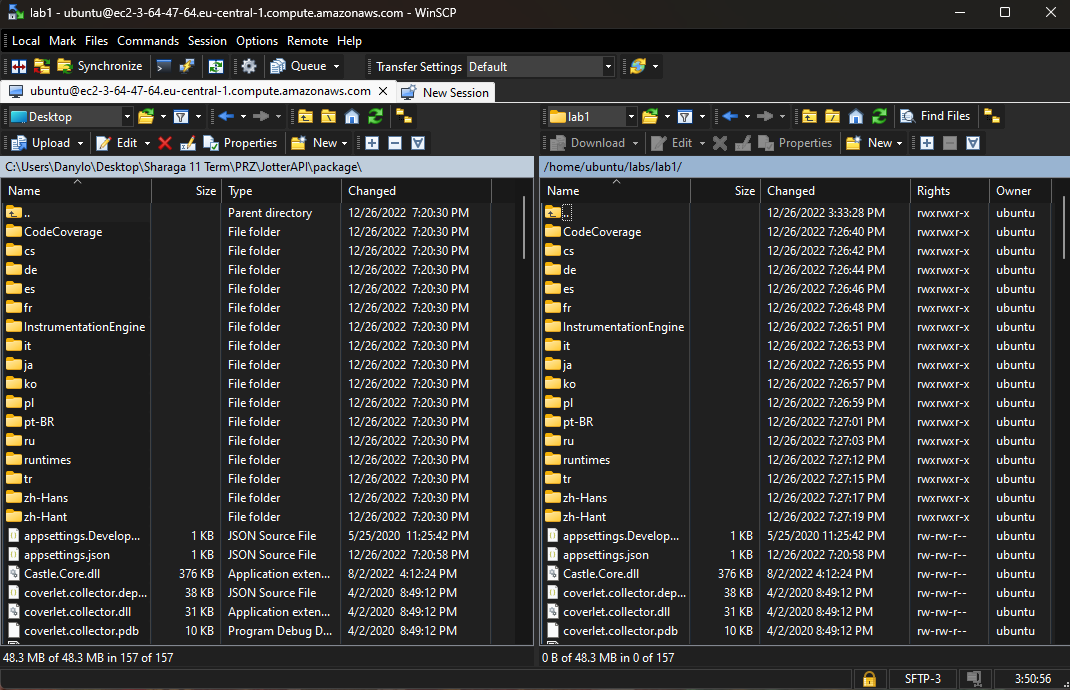
1. Створимо базу даних. Так як наш додаток використовує PostgreSQL, будемо використовувати Amazon RDS який має підтримку PostgreSQL:  
     
     
     
   
2. Перевіряємо що її успішно створено:  
   
3. Йдемо до «VPC security groups» за посиланням, налаштовуємо «Inbound rules» на весь IPv4 трафик:  
   
4. Перевіряємо що база жива та доступна:  
   
5. Використаємо Putty для підключення за SSH до EC2. Для цього, введемо IP адресу отриманої машини. Також, додамо наш отриманий файл з приватним ключем до авторизації:  
     
     
   

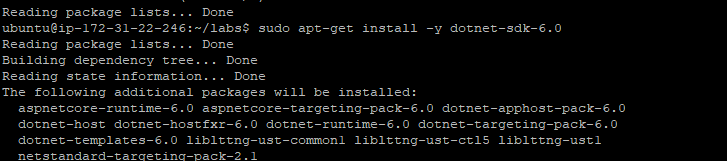


1. Також, використаємо WinSCP як SFTP клієнт:  
   
2. Після локального запуску проекту та перевірки, що все працює, запаблішимо проект у папку, в даному випадку, була використана папка package:

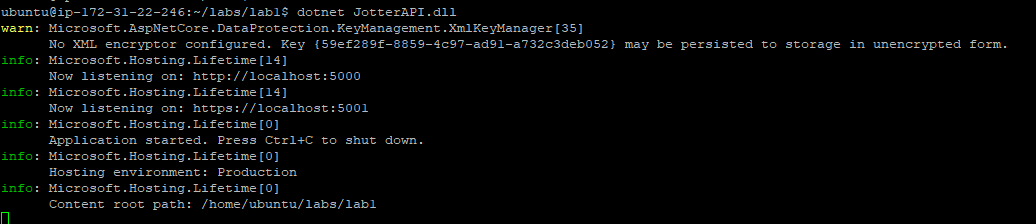
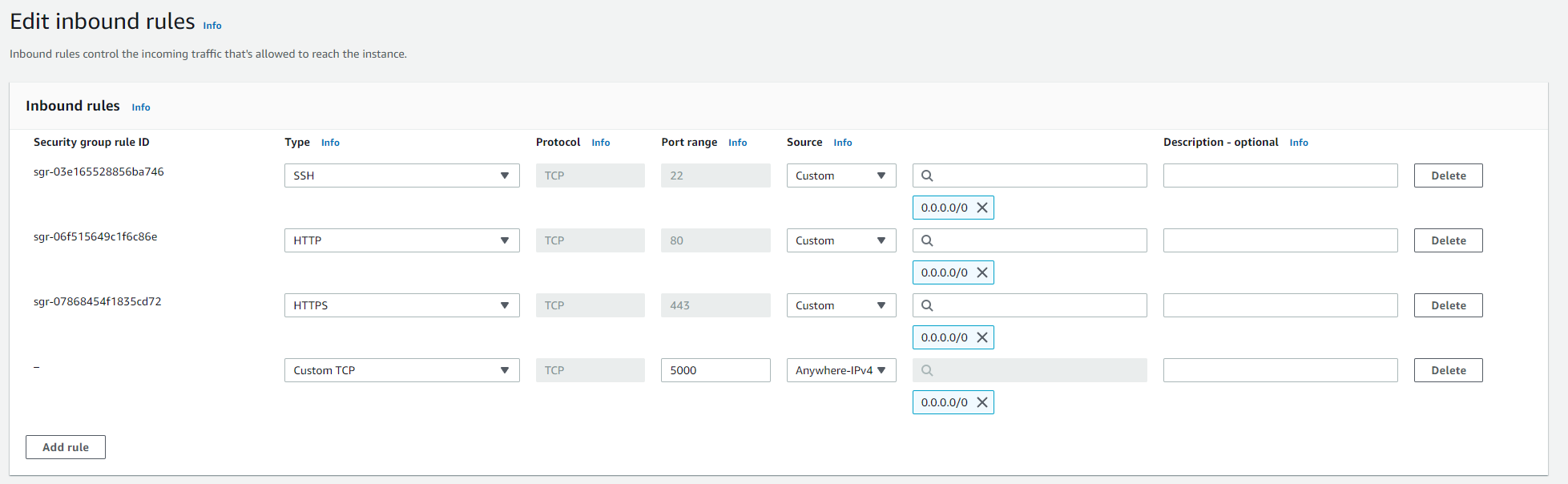
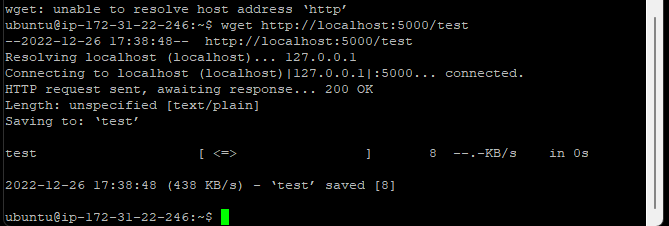
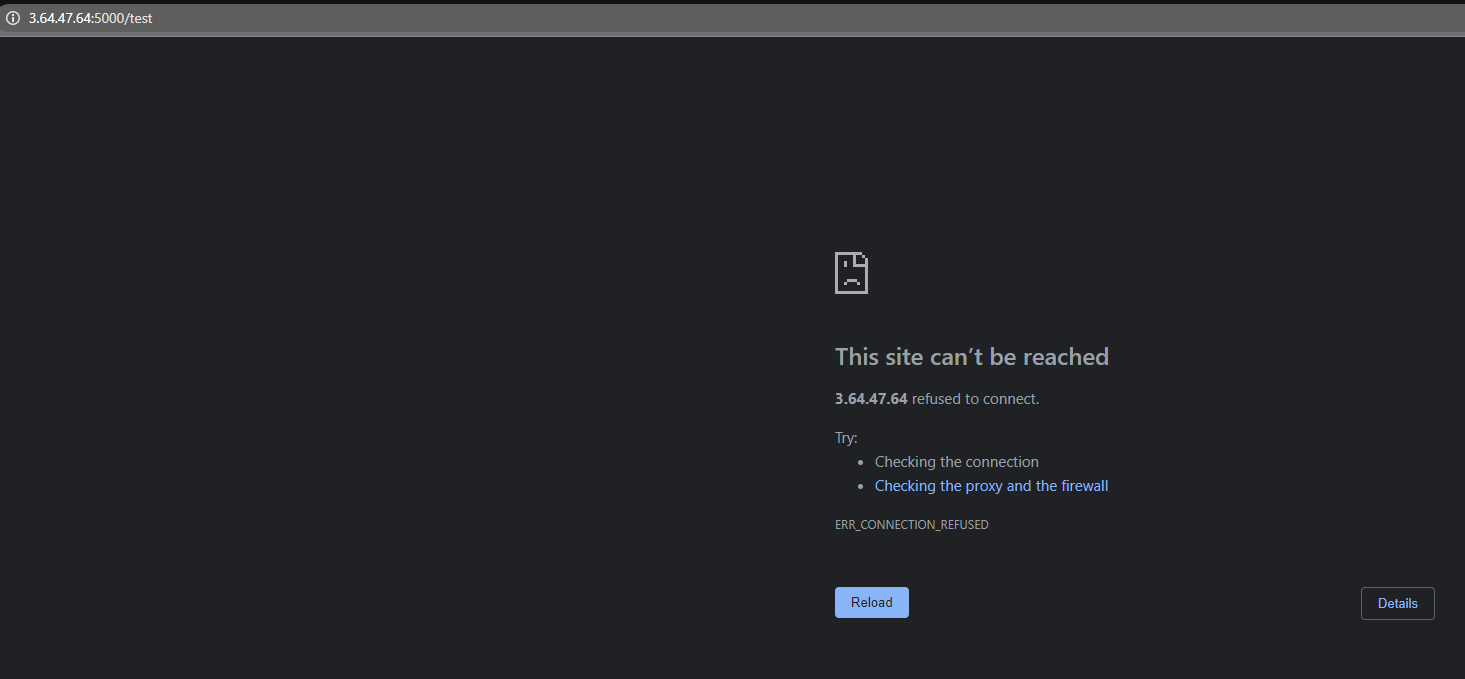


1. Використовуючи WinSCP клієнт, перемістимо отриманий пакет на віртуальну машину у папку Labs/lab1/:  
   

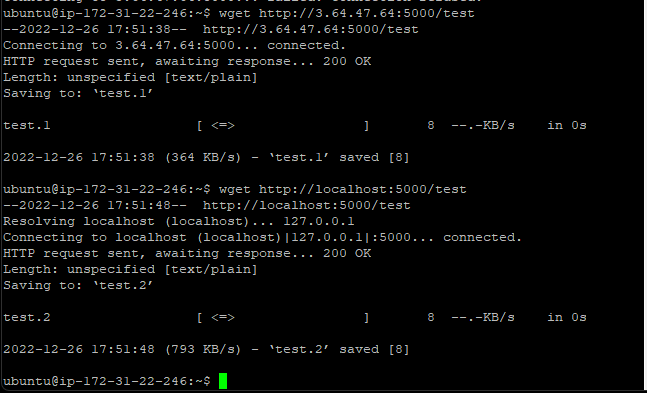
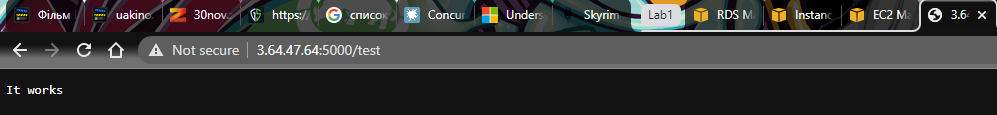
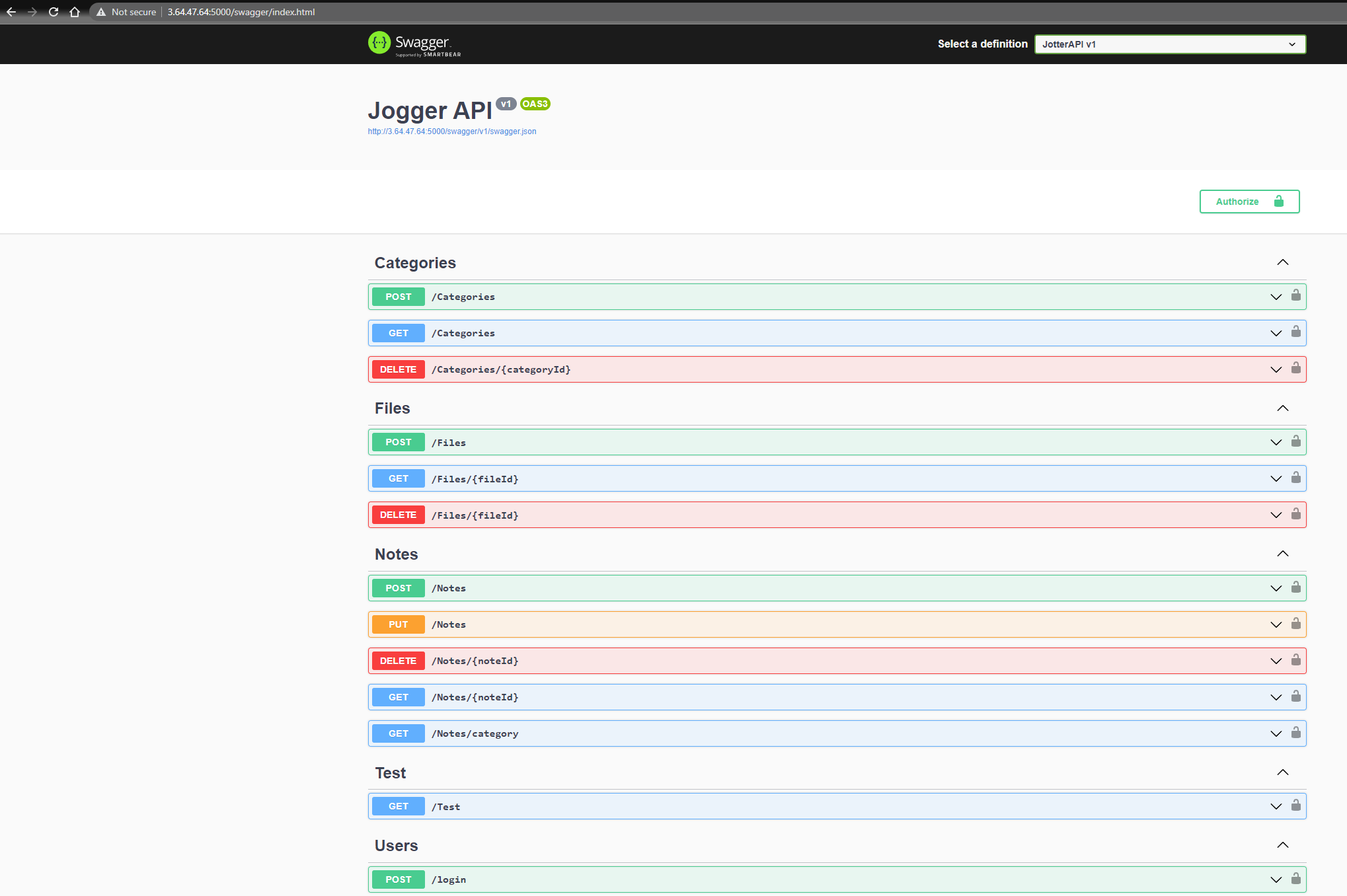
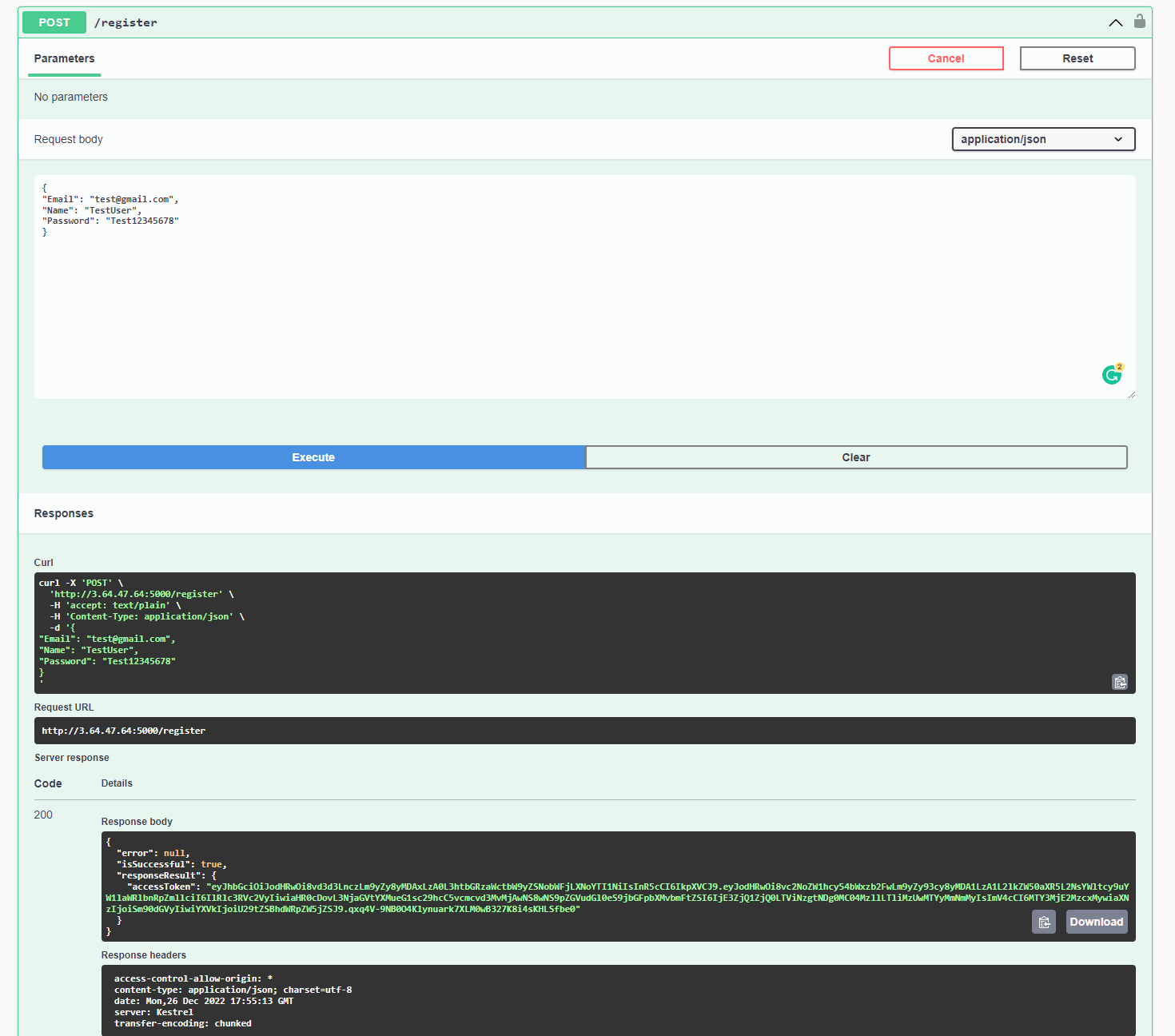
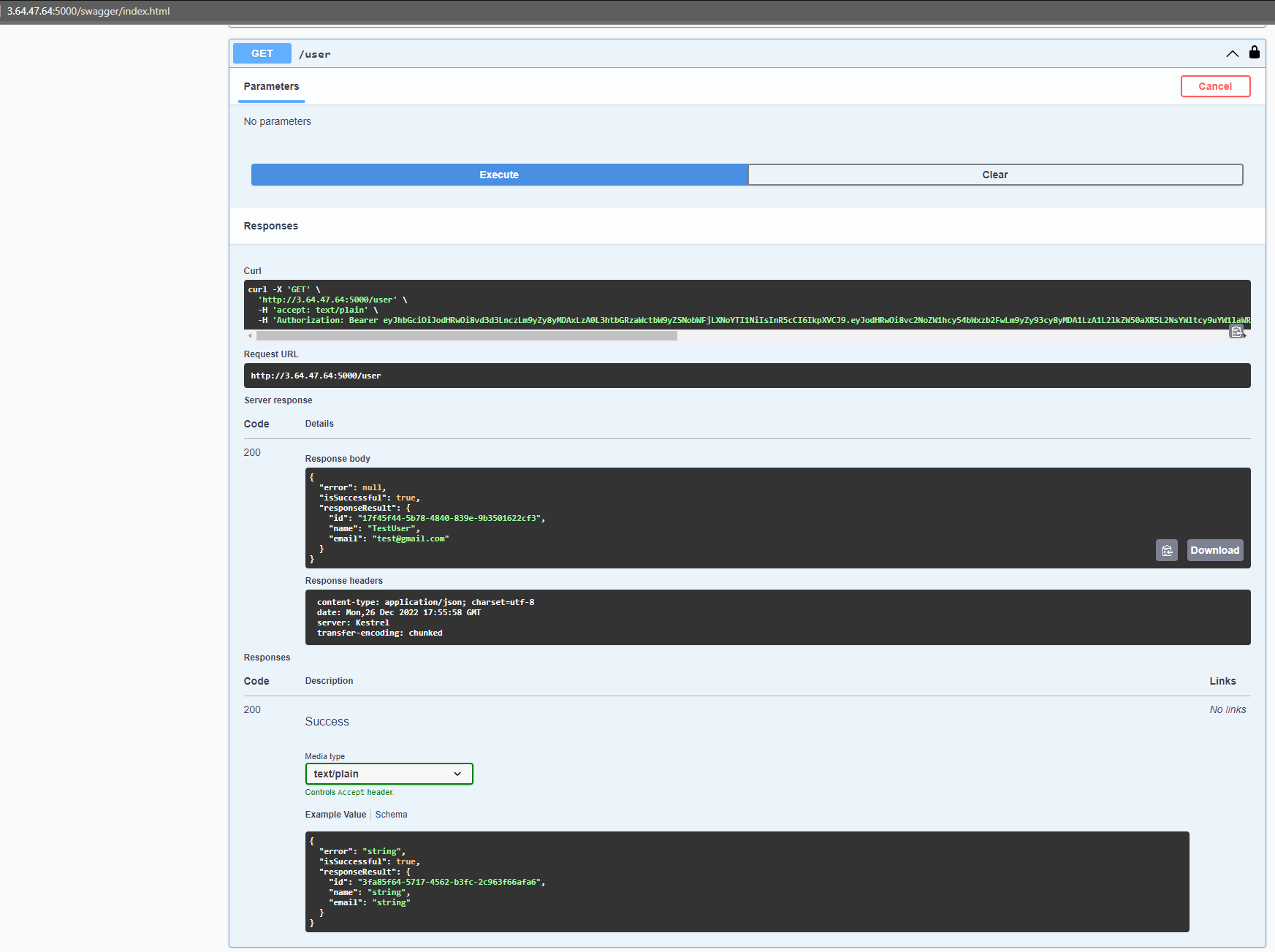
1. Перейдемо до налаштування EC2. Для цього, встановимо .NET6 SDK:  
   
2. Перевіримо коректність встановлення, виконавши наступну команду:



1. Запустимо застосунок, перейшовши у папку Labs/lab1/ та запустивши .dll файл:  
   
2. Додамо «Inbound Security Rule» для порту 5000 для доступу зовні до додатку на EC2:  
     
   
3. Перевіримо можливість отримання даних локально (на EC2) та за допомогою http запитів зі свого комп’ютера. Перший варіант у даному випадку буде працювати без проблем (використовуючи команду wget). А ось другий буде повертати помилки через неможливість з'єднання з сервером. Щоб прибрати цю помилку, додаємо строчку «.UseUrls(“http://0.0.0.0:5000”)» до Program.cs:  
     
   

Text

Description automatically generated

1. Перезапустимо застосунок та перевіримо його працездатність (http://3.64.47.64:5000/):  
     
     
     
     
   
2. Додаток задеплоєно за допомогою AWS EC2 використовуючи підхід IaaS.

**Висновок:**

В результаті виконання лабораторної роботи було розгорнуто застосунок на AWS EC2 за допомогою підключення через SSH для налаштування та підключення по SFTP для завантаження заздалегідь підготовленого застосунку + Amazon RDS для бази даних.