|  |  |  |
| --- | --- | --- |
|  |  |  |
| МИНОБРНАУКИ РОССИИ | | |
| Федеральное государственное бюджетное образовательное учреждение  высшего образования  **«МИРЭА – Российский технологический университет»**  **РТУ МИРЭА** | | |

Институт Информационных технологий

Кафедра Инструментального и прикладного программного обеспечения

**ПРАКТИЧЕСКАЯ РАБОТА №1**

по дисциплине «Разработка серверных частей интернет-ресурсов»

**Тема практической работы:**

**Студент группы** ИКБО-30-20 Максим Шевченко Алекссандрович

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(подпись студента)

**Руководитель практической работы**

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(подпись руководителя)

Работа представлена «\_\_\_\_»\_\_\_\_\_\_\_\_\_\_\_\_ 2022 г.

Допущен к работе «\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_ 2022 г.

Москва 2022

# **Цель работы:**

Создать свою конфигурацию серверного программного обеспечения, в которой должны присутствовать веб-сервер, операционная система, язык программирования и база данных.

# **Постановка задачи:**

Для проверки работоспособности конфигурации требуется инициализировать базу данных: создать отдельного пользователя для работы с ней, создать базу данных, в которой создать таблицу пользователи с полями: идентификационный номер, имя, фамилия. Также для проверки конфигурации требуется сгенерировать тестовую страничку, содержащую выборку из созданной таблицы и информационное сообщение о версии языка программирования, его настройках и конфигурации.

# **Ход работы:**

Мною было принято решение использовать технологии LAMP — Linux, Apache, MySQL, PHP. Для этого необходимо было создать файл docker-compose.yml , в котором будет прописаны инструкции для запуска и настройки необходимых нам сервисов. Структура проекта будет выглядеть следующим образом: директория database, в котором будет хранится скрипт для создания и инициализирования базы данных, который был предоставлен в качестве проверки работоспособности конфигурации, директория server, в котором хранится index.php и style.css (Рисунок 1)

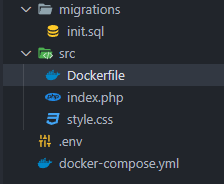


Рисунок 1 – Структура проекта.

Также в директории server присутствует Dockerfile, необходимый для закачки образа php apache-buster и установки нужных расширений (Рисунок 2).

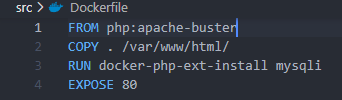


Рисунок 2 – Содержимое Dockerfile.

В файле docker-compose.yml прописаны инструкции для запуска необходимых нам технологий (Рисунок 3).

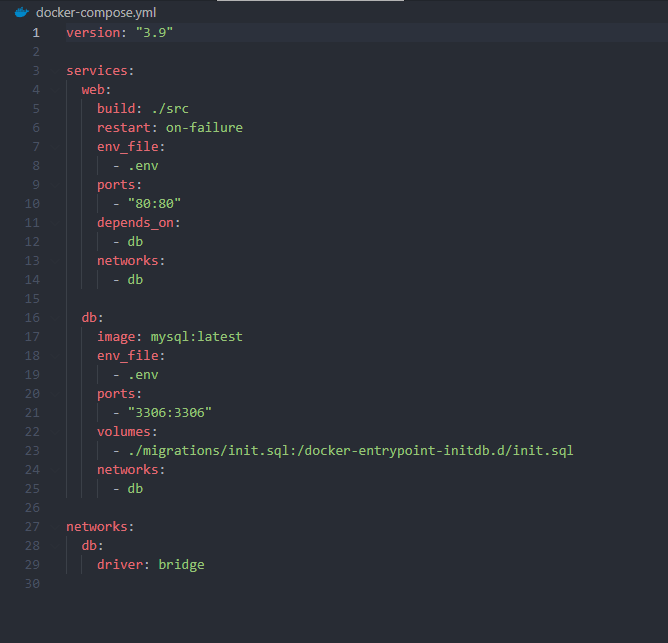


Рисунок 3 – Инструкции для базы данных.

Через консоль осуществим запуск с помощью команды docker-compose up (Рисунок 4).

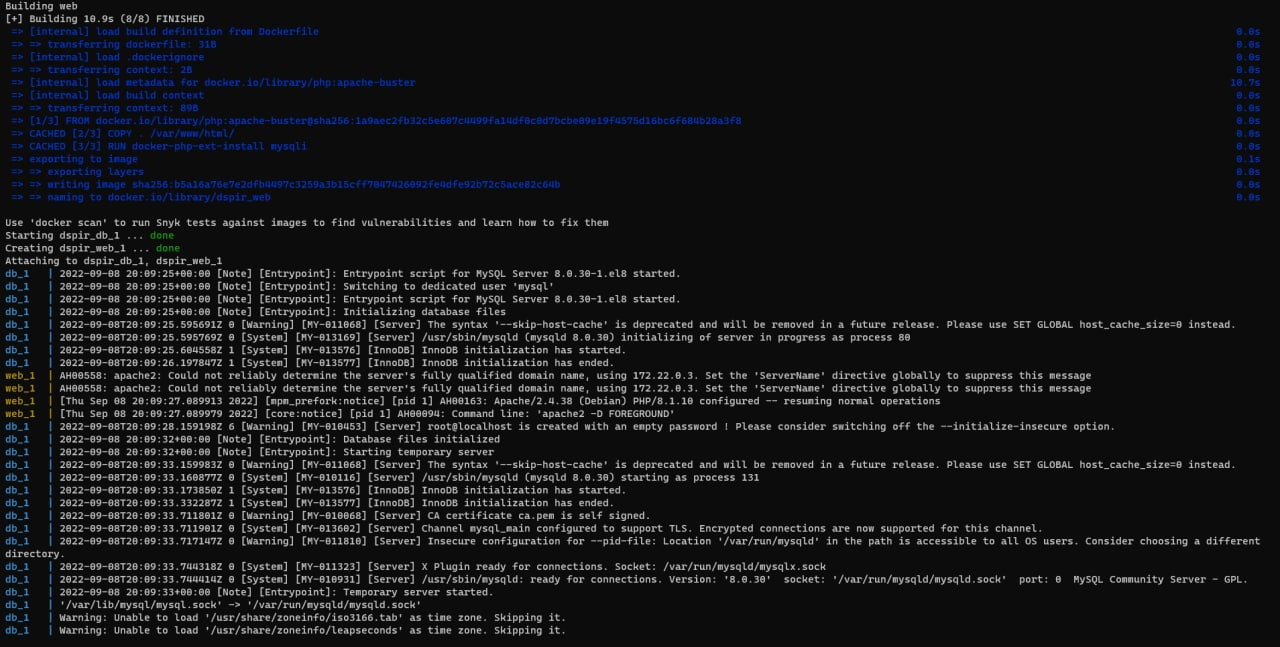


Рисунок 4 – Запуск docker-compose.

Переходим по localhost:8000 и видим наш готовый результат (Рисунок 5).

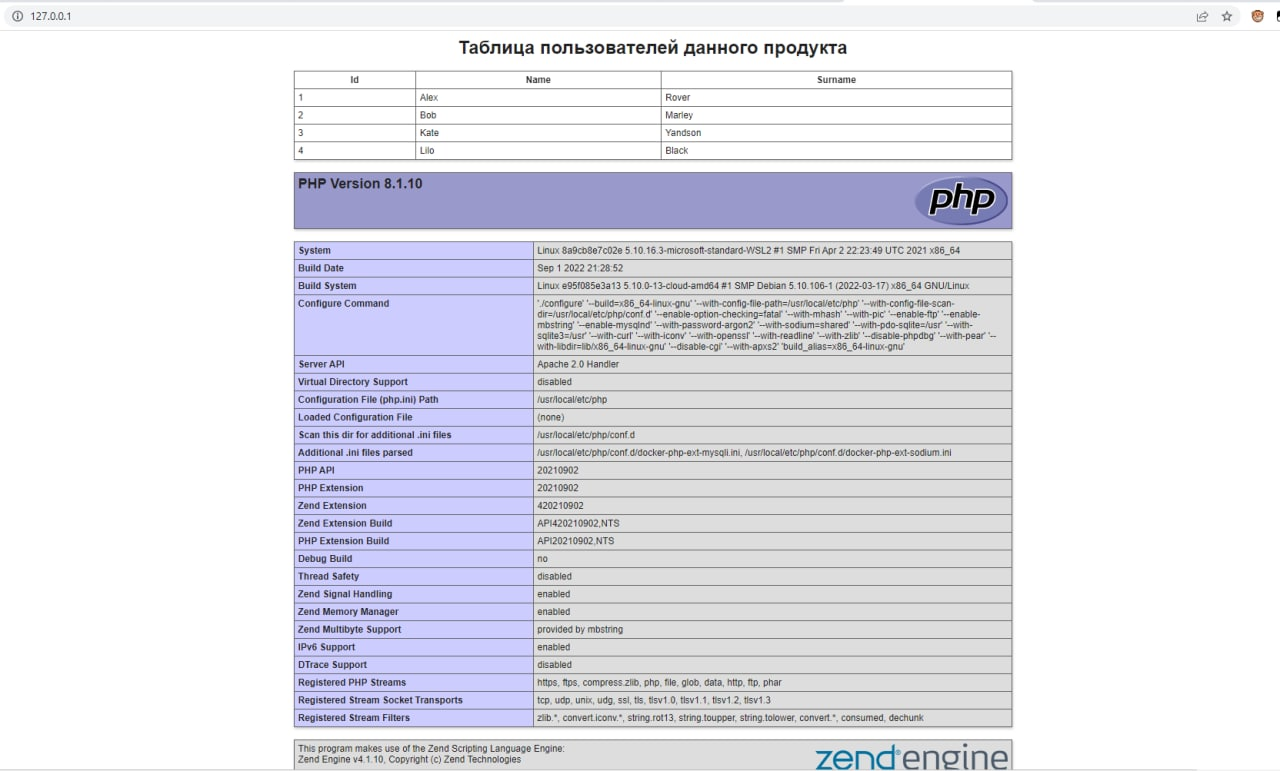


Рисунок 5 – Результат.

# **Выводы:**

В ходе выполнения практической работы была успешна установлена конфигурация LAMP и выполнены поставленные задачи.

# **Ответы на вопросы:**

1. Сервер и клиент - вычислительная или сетевая архитектура, в которой задания или сетевая нагрузка распределены между поставщиками услуг, называемыми серверами, и заказчиками услуг, называемыми клиентами. Фактически клиент и сервер — это программное обеспечение.

2. База данных - совокупность данных, хранимых в соответствии со схемой данных

3. API - описание способов, которыми одна компьютерная программа может взаимодействовать с другой программой.

4. Сервис, отличия от сервера - веб-сервер - это сервер реализующий http протокол. веб-сервис - это технология для взаимодействия между системами

5. Архитектура клиент-сервер - вычислительная или сетевая архитектура, в которой задания или сетевая нагрузка распределены между поставщиками услуг, называемыми серверами, и заказчиками услуг, называемыми клиентами.

6. Виды сервисов - сервисы делятся на два вида по способу использования: Started и Bound, и на два вида по способу взаимодействия с пользователем: Background и Foreground.

7. Масштабируемость - в электронике и информатике означает способность системы, сети или процесса справляться с увеличением рабочей нагрузки при добавлении ресурсов

8. Протоколы передачи данных - это стек протоколов TCP и IP. Первый обеспечивает и контролирует надёжную передачу данных и следит за её целостностью. Второй же отвечает за маршрутизацию для отправки данных.

9. Толстый и тонкий клиенты - Толстые клиенты работают с информацией на основе собственных аппаратных и программных возможностей, в то же время тонкие применяют ПО центрального сервера только чтобы обработать данные, предоставляя системе лишь требуемый графический интерфейс для выполнения работы пользователем.

10. Паттерн MVC: общие тезисы - MVC расшифровывается как модель-представление-контроллер (от англ. model-view-controller). Это способ организации кода, который предполагает выделение блоков, отвечающих за решение разных задач. Один блок отвечает за данные приложения, другой отвечает за внешний вид, а третий контролирует работу приложения. Компоненты: модель, представление и контроллер.

11. Паттерн MVC: Model-View-Presenter - MVP (Model-View-Presenter) — паттерн разработки пользовательского интерфейса. Шаблон MVP является производным от MVC, но при этом имеет несколько иной подход. Основное отличие — представление (presenter) не так сильно связано моделью (model). Модель представляет данные для отображения пользователю.

12. Паттерн MVC: Model-View-View Model - Model-View-ViewModel (MVVM) — шаблон проектирования архитектуры приложения. Представлен как модификация шаблона Presentation Model. Ориентирован на современные платформы разработки, такие как Windows Presentation Foundation, Silverlight от компании Microsoft, ZK framework.

13. Паттерн MVC: Model-View-Controller - Model-View-Controller — схема разделения данных приложения и управляющей логики на три отдельных компонента: модель, представление и контроллер — таким образом, что модификация каждого компонента может осуществляться независимо. Модель предоставляет данные и реагирует на команды контроллера, изменяя своё состояние

14. Docker: общие тезисы и определения - программное обеспечение для автоматизации развёртывания и управления приложениями в средах с поддержкой контейнеризации, контейнеризатор приложений

15. Dockerfile - Dockerfile — это текстовый сценарий инструкций, который используется для создания образа контейнера.

16. Docker Compose — это средство для определения и запуска приложений Docker с несколькими контейнерами.

17. LAMP — это набор ПО для запуска веб-сервера на VPS. В него входит стабильная версия Linux — Ubuntu 20.04 LTS, веб-сервер Apache, база данных MySQL, PHP как модуль Apache, а также Python и Perl.