**Weather App**

**Описание проекта**

Weather App – это приложение для получения актуальной погоды с использованием OpenWeatherMap API. Оно поддерживает хранение истории запросов в базе данных и предоставляет удобный интерфейс для просмотра данных.

**Инструкция по установке**

**Запуск без Docker**

1. Установите JDK 21 и Maven.
2. Склонируйте репозиторий:
3. git clone <репозиторий>
4. cd weather
5. Соберите проект:
6. mvn clean package
7. Запустите приложение:
8. java -jar target/weather-0.0.1-SNAPSHOT.jar

**Запуск с Docker**

1. Убедитесь, что установлен Docker и Docker Compose.
2. Запустите контейнеры:
3. docker-compose up -d
4. Откройте браузер и перейдите по адресу:
5. http://localhost:8080

**API-ключ OpenWeatherMap**

1. Зарегистрируйтесь на <https://openweathermap.org/> .
2. Получите API-ключ в разделе **API keys**.
3. Укажите ключ в **application.properties**:
4. weather.api.key=4d611e05d71e73675cae9320120b474c (пример)

**Описание работы базы данных**

Используется MySQL, данные хранятся в следующих таблицах:

* **weather\_request**: история запросов (город, дата, температура и т. д.).
* **users**: пользователи (если реализована авторизация).

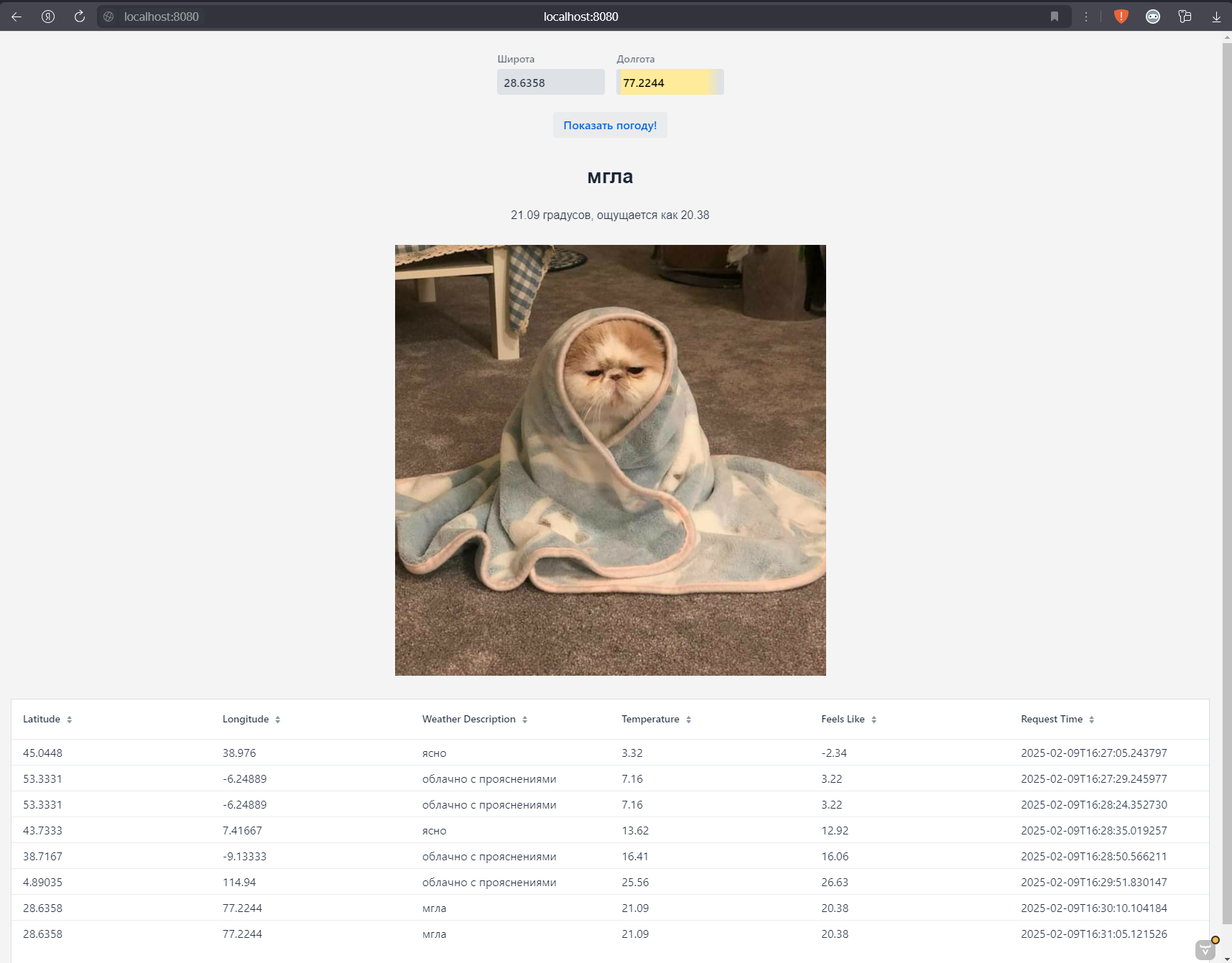
**Настройки подключения к БД**

Если используется локальная БД, убедитесь, что MySQL работает, и укажите корректные параметры в **application.properties**:

spring.datasource.url=jdbc:mysql://localhost:3306/weather

spring.datasource.username=root

spring.datasource.password=ghj100ghj100 (пример)



**Разработка и развертывание**

**Сборка JAR-файла**

mvn clean package

**Загрузка в репозиторий**

git add .

git commit -m "Initial commit"

git push origin main

**Финальная проверка**

1. Очистите проект и разверните заново, следуя инструкциям в **README.md**.
2. Проверьте запуск Docker-контейнеров:
3. docker-compose down && docker-compose up -d
4. Убедитесь, что история запросов сохраняется и корректно отображается.