

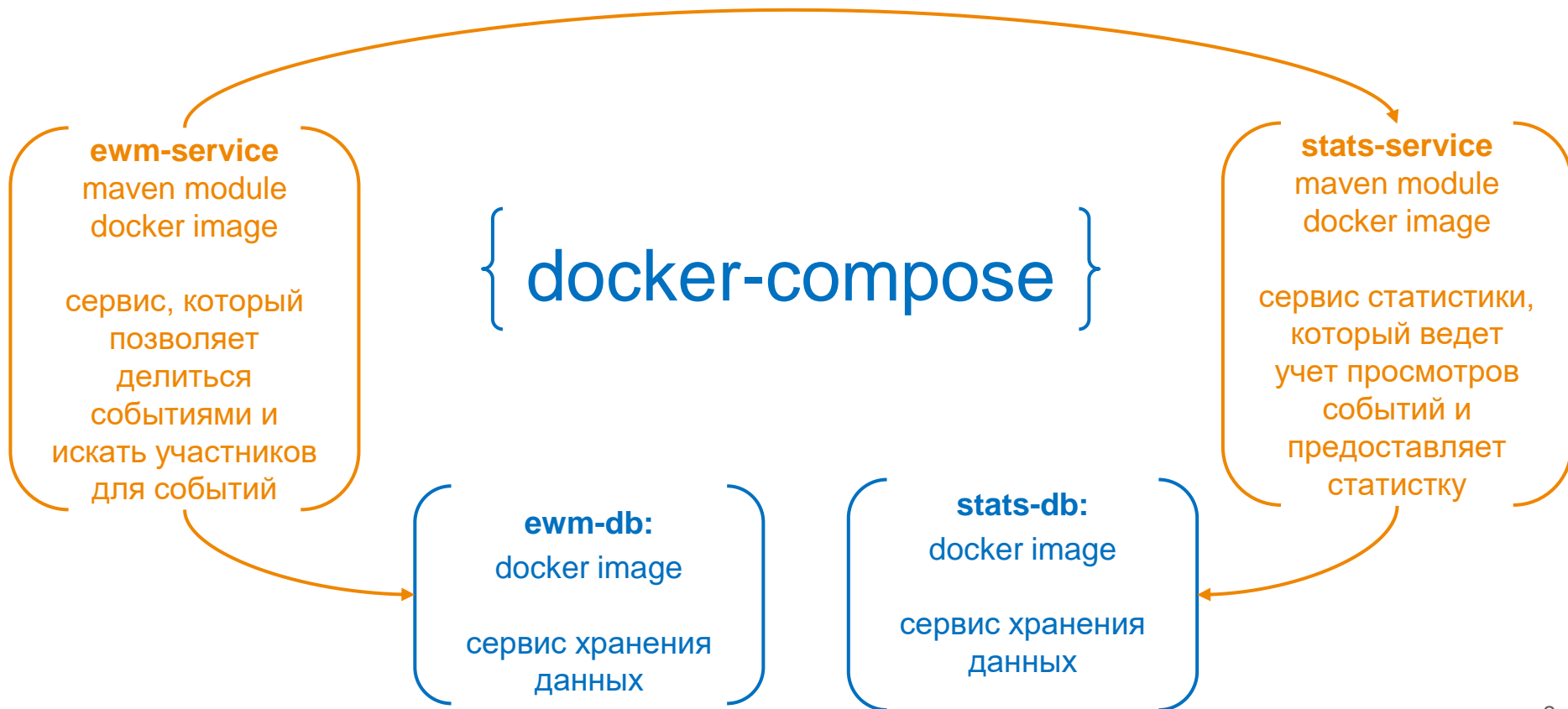
Александр Васильев

дипломный проект
«Explore with me»



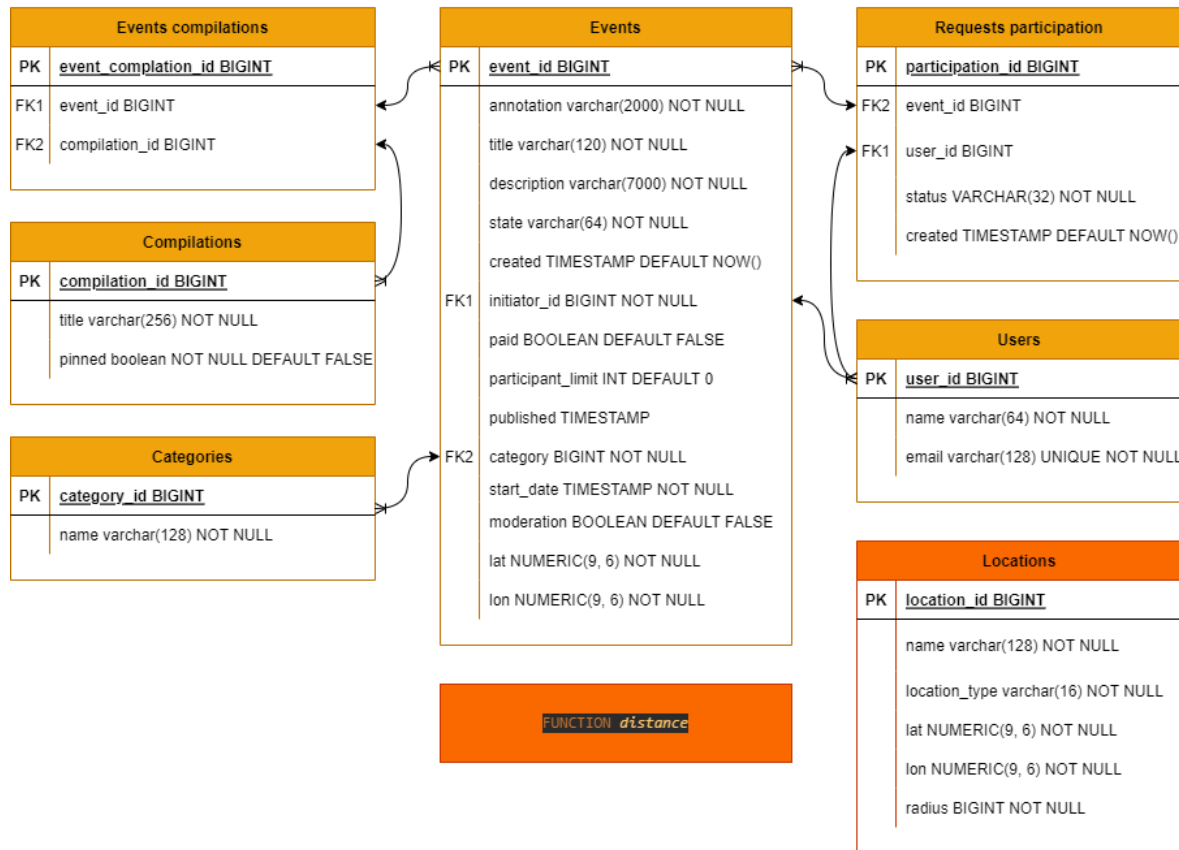
2022 год

Микросервисная архитектура проекта



Архитектура баз данных

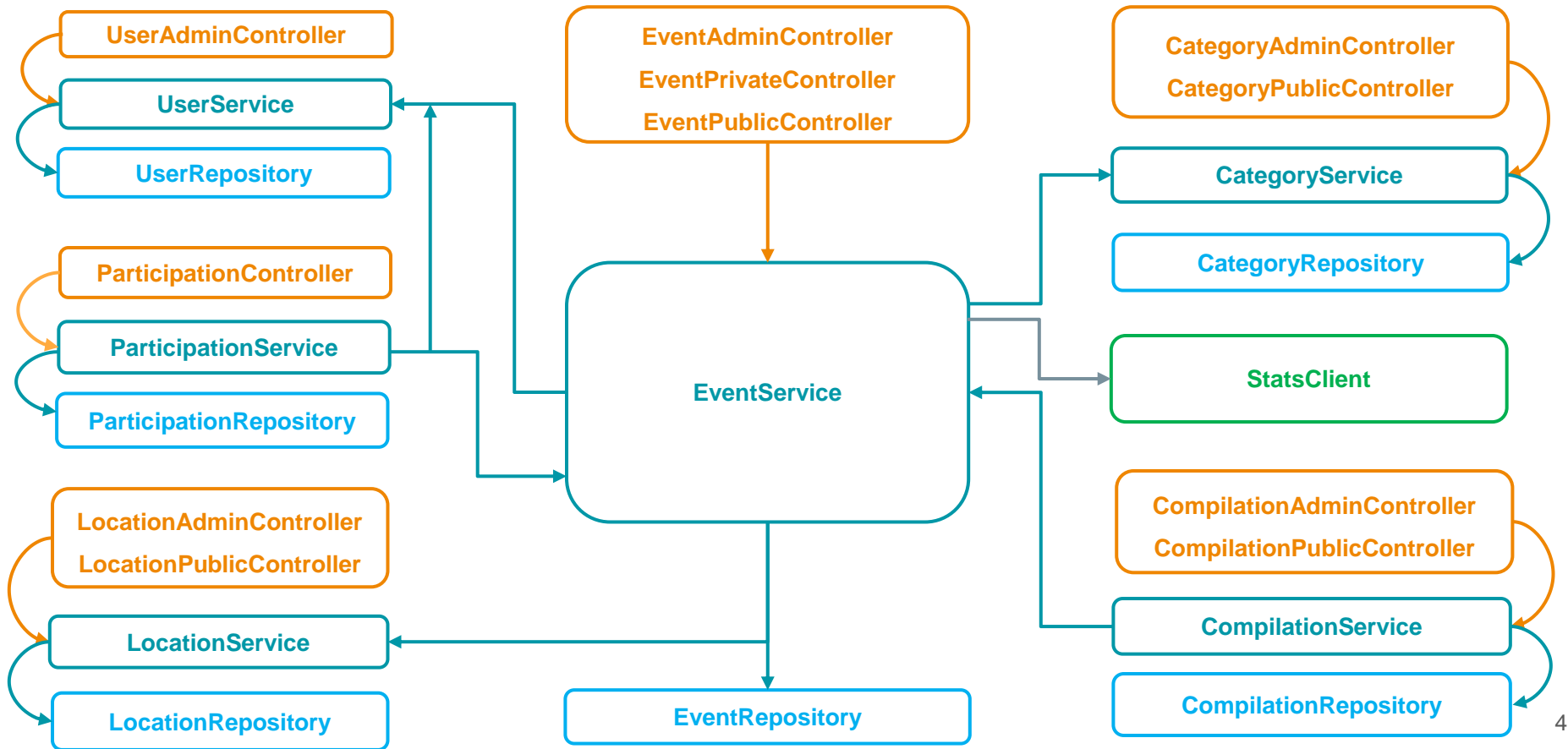
explore-with-me-server



explore-with-me-stats

Participation Requests	
PK	<u>hit_id BIGINT</u>
	app VARCHAR(32) NOT NULL
	uri VARCHAR(1000) NOT NULL
	ip VARCHAR(45) NOT NULL
	timestamp TIMESTAMP NOT NULL

Архитектура микросервиса ewm-service



Поккрытие тестами

Unit-тесты

16 тестов

пакет *event*

Интеграционные тесты

7 тестов

Мануальные тесты

пакет *location*

пакет *location*

explore-with-me-server

Element	Missed Instructions	Cov.	Missed Branches	Cov.	Missed	Cxty	Missed	Lines	Missed	Methods	Missed	Classes
com.notbadcode.explorewithme.participation	<div><div></div></div>	8 %	<div><div></div></div>	0 %	44	47	108	115	27	30	2	5
com.notbadcode.explorewithme.event	<div><div></div></div>	71 %	<div><div></div></div>	58 %	43	100	82	329	30	77	1	9
com.notbadcode.explorewithme.compilation	<div><div></div></div>	10 %	<div><div></div></div>	0 %	28	33	94	104	24	29	1	5
com.notbadcode.explorewithme.locations	<div><div></div></div>	27 %	<div><div></div></div>	0 %	23	29	64	85	21	27	1	6
com.notbadcode.explorewithme.error	<div><div></div></div>	8 %	<div><div></div></div>	n/a	10	15	59	67	10	15	1	5
com.notbadcode.explorewithme.category	<div><div></div></div>	26 %	<div><div></div></div>	0 %	16	23	42	55	15	22	1	5
com.notbadcode.explorewithme.user	<div><div></div></div>	33 %	<div><div></div></div>	n/a	12	19	32	49	12	19	0	4
com.notbadcode.explorewithme.stats	<div><div></div></div>	16 %	<div><div></div></div>	n/a	7	9	35	44	7	9	2	3
com.notbadcode.explorewithme.util	<div><div></div></div>	61 %	<div><div></div></div>	50 %	4	13	7	21	2	11	1	3
com.notbadcode.explorewithme	<div><div></div></div>	37 %	<div><div></div></div>	n/a	1	2	2	3	1	2	0	1
com.notbadcode.explorewithme.config	<div><div></div></div>	100 %	<div><div></div></div>	n/a	0	3	0	5	0	3	0	1
Total	2 545 of 4 268	40 %	69 of 98	29 %	188	293	525	877	149	244	10	47



Добавлена функциональность «Локации»

Admin: Локации

Администратор добавляет/редактирует локации по координатам и радиусу

Admin: Локации API для работы с локациями

POST /admin/locations Добавление новой локации

PATCH /admin/locations Редактирование локации

Public: События

Пользователи получают список опубликованных событий, входящих в локацию

Public: События Публичный API для работы с событиями

GET /events/locations/{locationId} Получение опубликованных событий по локации

Public: Локации

Пользователи получают список локаций, можно искать по названию и/или типу

Public: Локации API для работы с локациями

GET /locations Получение списка локаций

GET /locations/{locationId} Получение локации

Private: События

При добавлении события можно проверять, вхождение в локацию

Private: События Закрытый API для работы с событиями

POST /users/{userId}/events/locations/{locationId} Добавление нового события с проверкой попадания в локацию

Проблемы, возникшие при работе над дипломом

Проблема

Циклическая зависимость компонентов *EventService* и *CategoryService*

Первое решение

«Ленивая» загрузка *@Lazy* одного из компонентов

Финальное решение

Создание Query-запроса в *CategoryRepository* с разрывом зависимости от *EventService*

Проблема

Добавление функциональности «Локации» поломало интеграционные тесты

Решение

Использование библиотеки *Testcontainers*, позволяющей производить тесты на базах данных, развернутых в *docker*



Использованные технологии

- Java 11, Lombok
- Spring Boot
- JPA, Hibernate, QueryDSL
- PostgreSQL, SQL, FlyBase
- Swagger
- Junit, Mockito, Testcontainers
- Docker, docker-compose
- Maven (multi-module project)
- Postman

