

Университет ИТМО

Мегафакультет компьютерных технологий и управления

Факультет программной инженерии и компьютерной техники

ОТЧЕТ ПО ПРОЕКТИРОВАНИЮ И РАЗРАБОТКЕ ДИЗАЙН ПРОЕКТА

Курса «Рефакторинг баз данных и приложений»

Санкт-Петербург 2022

В данном спринте было проведено следующее:

1. Модульное тестирование сервисных модулей первого и второго приложения
2. Создание скрипта для создания БД.

### Изменения в коде.

Создан скрипт для создания необходимых таблиц БД:

```
CREATE TABLE treatment(id SERIAL PRIMARY KEY, name TEXT, type TEXT);

CREATE TABLE user(id SERIAL PRIMARY KEY, name TEXT, password TEXT, token TEXT, position TEXT);

CREATE TABLE doctor(id SERIAL PRIMARY KEY, FOREIGN KEY(user_id) REFERENCES user(id));

CREATE TABLE patient(id SERIAL PRIMARY KEY, name TEXT, diagnosis TEXT, insurance INT, status TEXT, email TEXT);

CREATE TABLE patient_doctor(FOREIGN KEY(patient_id) REFERENCES patient(id), FOREIGN KEY(doctor_id) REFERENCES doctor(id));

CREATE TABLE appointments(id SERIAL PRIMARY KEY, FOREIGN KEY(patient_id) REFERENCES patient(id), FOREIGN KEY(treatment_id) REFERENCES treatment(id), time_pattern TEXT, dose DOUBLE, start_date DATE, end_date DATE);

CREATE TABLE events(id SERIAL PRIMARY KEY, date DATE, status TEXT, reason TEXT, FOREIGN KEY(appointment_id) REFERENCES appointment(id));
```

### Модульное тестирование

Для примера приведу 2 сервиса, остальные протестированы аналогично.

Обращения в базу данных отмечены аннотацией @Mock, чтобы во время тестирования не происходило никаких изменений в бд. (проверяется только вызов метода).

```

@ExtendWith(MockitoExtension.class)
class PatientsServiceTest {

    @Mock
    PatientsDao patientsDao;

    @InjectMocks
    PatientsService patientsService;

    @Test
    @DisplayName("Test add patient with null")
    void addPatientNullTest() {
        PatientDto patientDto = new PatientDto();
        patientsService.addPatient(patientDto);
        verify(patientsDao, times(wantedNumberOfInvocations: 0)).findPatientByName(patientDto.getName(
    }

```

```

@ExtendWith(MockitoExtension.class)
public class DoctorsServiceTest {

    @Mock
    DoctorDao doctorDao;

    @InjectMocks
    DoctorsService doctorsService;

    @Test
    @DisplayName("Save doctor test")
    void addDoctorTest() {
        User user = new User();
        doctorsService.addDoctor(user);
        verify(doctorDao, times(wantedNumberOfInvocations: 1)).save(user);
    }
}

```