

1 ПРАКТИЧЕСКОЕ ЗАДАНИЕ SE-19

1.1 ПЕРЕВЕСТИ ПРИЛОЖЕНИЕ НА SPRING

1.1.1 ЗАМЕНИТЬ ЗАВИСИМОСТИ И АННОТАЦИИ WELD CDI НА SPRING CONTEXT

1.1.2 ЗАМЕНИТЬ РЕПОЗИТОРИИ APACHE DELTA SPIKE НА SPRING DATA JPA

1.1.3 ЗАМЕНИТЬ ТЕСТЫ WELD - APACHE DELTA SPIKE НА SPRING TESTING

1.2 РЕКОМЕНДАЦИИ

1.2.1 УДАЛИТЬ ВСЕ ЗАВИСИМОСТИ WELD И APACHE DELTA SPIKE ИЗ MAVEN-ПРОЕКТА

1.2.2 РЕАЛИЗОВАТЬ SPRING CONFIG С ИСПОЛЬЗОВАНИЕМ JAVA BASED ПОДХОДА И АННОТАЦИЙ

1.3 ПРОГРАММНЫЙ КОД

```
<dependency>
  <groupId>org.springframework</groupId>
  <artifactId>spring-context</artifactId>
  <version>5.0.8.RELEASE</version>
</dependency>
```

Рис. 1: MAVEN-ЗАВИСИМОСТЬ SPRING CONTEXT

```
<dependency>
  <groupId>org.springframework.data</groupId>
  <artifactId>spring-data-jpa</artifactId>
  <version>2.0.7.RELEASE</version>
</dependency>
```

Рис. 2: MAVEN-ЗАВИСИМОСТЬ SPRING DATA JPA

```
@Bean
public DataSource dataSource() {
    final DriverManagerDataSource dataSource =
        new DriverManagerDataSource();
    dataSource.setDriverClassName("com.mysql.jdbc.Driver");
    dataSource.setUrl("jdbc:mysql://127.0.0.1:3306/example");
    dataSource.setUsername("root");
    dataSource.setPassword("");
    return dataSource;
}

@Bean
public LocalContainerEntityManagerFactoryBean entityManagerFactory(
    final DataSource dataSource
) {
    final LocalContainerEntityManagerFactoryBean factoryBean;
    factoryBean = new LocalContainerEntityManagerFactoryBean();
    factoryBean.setDataSource(dataSource);
    factoryBean.setJpaVendorAdapter(new HibernateJpaVendorAdapter());
    factoryBean.setPackagesToScan("ru.volnenko.spring.model");
    final Properties properties = new Properties();
    properties.put("hibernate.show_sql", "true");
    properties.put("hibernate.hbm2ddl.auto", "update");
    properties.put("hibernate.dialect",
        "org.hibernate.dialect.MySQL5Dialect");
    factoryBean.setJpaProperties(properties);
    return factoryBean;
}

@Bean
public PlatformTransactionManager transactionManager(
    final LocalContainerEntityManagerFactoryBean emf
) {
    final JpaTransactionManager transactionManager =
        new JpaTransactionManager();
    transactionManager.setEntityManagerFactory(emf.getObject());
    return transactionManager;
}
```

Рис. 3: JAVA BASED КОНФИГУРАЦИЯ SPRING JPA