

1 ПРАКТИЧЕСКОЕ ЗАДАНИЕ SE-14

1.1 ВНЕДРИТЬ В ПРИЛОЖЕНИЕ HIBERNATE

- 1.1.1 ТЕКУЩИЕ СУЩНОСТИ ПЕРЕВЕСТИ В ОБЪЕКТЫ DTO ДЛЯ СОХРАНЕНИЯ ТРАНСПОРТА ДАННЫХ ПО SOAP
- 1.1.2 СОЗДАТЬ ГРАФЫ СУЩНОСТЕЙ С ИСПОЛЬЗОВАНИЕМ JPA
- 1.1.3 СОЗДАТЬ JAVA BASED КОНФИГУРАЦИЮ СЕССИИ HIBERNATE
- 1.1.4 РЕАЛИЗОВАТЬ КАСКАДНОЕ УДАЛЕНИЕ ГРАФА СУЩНОСТИ "ПРОЕКТ"
- 1.1.5 РЕАЛИЗОВАТЬ УПРАВЛЕНИЕ ТРАНЗАКЦИЕЙ HIBERNATE НА УРОВНЕ СЕРВИСОВ
- 1.1.6 РЕАЛИЗОВАТЬ ПРЕОБРАЗОВАНИЕ СУЩНОСТЕЙ В КЛАССЫ DTO НА СЛОЕ ПРЕЗЕНТАЦИИ ДАННЫХ

1.2 РЕКОМЕНДАЦИИ

- 1.2.1 ИСПОЛЬЗОВАТЬ ИНТЕРФЕЙСЫ JPA ВМЕСТО РЕАЛИЗАЦИИ HIBERNATE
- 1.2.2 ИСПОЛЬЗОВАТЬ ИНТЕРФЕЙС "EntityManagerFactory" ВМЕСТО "SessionFactory"
- 1.2.3 ИСПОЛЬЗОВАТЬ ИНТЕРФЕЙС "EntityManager" ВМЕСТО "Session"
- 1.2.4 РЕАЛИЗОВАТЬ ХРАНЕНИЕ ПЕРЕЧИСЛЕНИЙ В ВИДЕ СТРОКИ С ИСПОЛЬЗОВАНИЕМ АННОТАЦИИ "@Enumerated"
- 1.2.5 РЕАЛИЗОВАТЬ ПРОВЕРКУ НА УНИКАЛЬНОСТЬ ЛОГИНА С ИСПОЛЬЗОВАНИЕМ АННОТАЦИИ "@Column"
- 1.2.6 ВЫНЕСТИ ОБЩИЕ ПОЛЯ СУЩНОСТЕЙ В АБСТРАКТНЫЙ КЛАСС СУЩНОСТИ С ИСПОЛЬЗОВАНИЕМ "@MappedSuperclass"
- 1.2.7 ИСПОЛЬЗОВАТЬ СРЕДСТВА АВТОМАТИЧЕСКОЙ ГЕНЕРАЦИИ СХЕМЫ СУБД ПО СТРАТЕГИИ ОБНОВЛЕНИЯ
- 1.2.8 ИСПОЛЬЗОВАТЬ ОТОБРАЖЕНИЕ SQL-КОНСТРУКЦИЙ ПРИ ВЫЗОВЕ HIBERNATE НА ЧТЕНИЕ / ЗАПИСЬ
- 1.2.9 СОЗДАВАТЬ РЕПОЗИТОРИЙ И ЭКЗЕМПЛЯР ENTITY MANAGER НА КАЖДЫЙ ВЫЗОВ СЕРВИСА

1.3 ПРОГРАММНЫЙ КОД

```
<dependency>
  <groupId>org.hibernate</groupId>
  <artifactId>hibernate-core</artifactId>
  <version>5.4.0.Final</version>
  <scope>compile</scope>
</dependency>
```

Рис. 1: MAVEN-ЗАВИСИМОСТЬ HIBERNATE CORE

```
private EntityManagerFactory factory() {
    final Map<String, String> settings = new HashMap<>();
    settings.put(Environment.DRIVER, propertyService.getJdbcDriver());
    settings.put(Environment.URL, propertyService.getJdbcUrl());
    settings.put(Environment.USER, propertyService.getJdbcUsername());
    settings.put(Environment.PASS, propertyService.getJdbcPassword());
    settings.put(Environment.DIALECT,
        "org.hibernate.dialect.MySQL5InnoDBDialect");
    settings.put(Environment.HBM2DDL_AUTO, "update");
    settings.put(Environment.SHOW_SQL, "true");

    final StandardServiceRegistryBuilder registryBuilder
        = new StandardServiceRegistryBuilder();
    registryBuilder.applySettings(settings);
    final StandardServiceRegistry registry = registryBuilder.build();
    final MetadataSources sources = new MetadataSources(registry);
    sources.addAnnotatedClass(Task.class);
    sources.addAnnotatedClass(Project.class);
    sources.addAnnotatedClass(User.class);
    sources.addAnnotatedClass(Session.class);
    sources.addAnnotatedClass(Cat.class);
    final Metadata metadata = sources.getMetadataBuilder().build();
    return metadata.getSessionFactoryBuilder().build();
}
```

Рис. 2: JAVA BASED КОНФИГУРАЦИЯ HIBERNATE

```
<?xml version="1.0" encoding="UTF-8"?>
<persistence xmlns="http://xmlns.jcp.org/xml/ns/persistence"
             xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
             xsi:schemaLocation="http://xmlns.jcp.org/xml/ns/persistence
http://xmlns.jcp.org/xml/ns/persistence/persistence_2_1.xsd"
             version="2.1">
  <persistence-unit name="ENTERPRISE"
    transaction-type="RESOURCE_LOCAL">
    <provider>
      org.hibernate.jpa.HibernatePersistenceProvider
    </provider>

    <class>ru.volnenko.tm.model.User</class>
    <class>ru.volnenko.tm.model.Task</class>
    <class>ru.volnenko.tm.model.Project</class>
    <class>ru.volnenko.tm.model.Session</class>
    <properties>
      <property name="javax.persistence.jdbc.driver"
        value="com.mysql.jdbc.Driver"/>
      <property name="javax.persistence.jdbc.url"
        value="jdbc:mysql://localhost:3306/task-manager"/>
      <property name="javax.persistence.jdbc.user" value="root"/>
      <property name="javax.persistence.jdbc.password" value=""/>
    </properties>
  </persistence-unit>
</persistence>
```

Рис. 3: КОНФИГУРАЦИЯ PERSISTENCE UNIT

```
<dependency>
  <groupId>org.hibernate</groupId>
  <artifactId>hibernate-c3p0</artifactId>
  <version>5.4.0.Final</version>
  <scope>compile</scope>
</dependency>
```

Рис. 4: MAVEN-ЗАВИСИМОСТЬ HIBERNATE C3P0