федеральное государственное автономное образовательное учреждение высшего образования

Санкт-Петербургский национальный исследовательский УНИВЕРСИТЕТ информационных технологий, механики и оптики

**Факультет программной инженерии и компьютерной техники**

**ЛАБОРАТОРНАЯ РАБОТА №2**

**ПО ДИСЦИПЛИНЕ «ТЕСТИРОВАНИЕ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ»**

Выполнил: Чудаков М.И.

Группа: P3319

Преподаватель: Исаев И.В.

Санкт-Петербург

2019

**Цель**

Необходимо разработать консольное приложение в соответствие с методологий TDD.

**Задачи**

1. Необходимо из краткого описания сформировать функциональные требования к программному продукту (не менее 20, т.е. дополнить);
2. Необходимо полностью покрыть тестами данные требования используя разработку через тестирование;
3. Для хранения данных использовать PostgreSQL или MongoDB;
4. Реализовать продукт 100% покрытый тестами.

**Вариант 19 (4)**

Разработка консольного приложения “Музыкальный каталог”.

При запуске программы пользователю предлагается справочная информация по использованию. Далее предлагается ввести команду, позволяющую выполнить одно из действий:

* осуществить поиск музыкальной композиции в каталоге по определенному критерию;
* вывести информацию о всех существующих в каталоге композициях;
* добавить информацию о композиции в каталог;
* удалить существующую в каталоге запись;
* выйти из программы.

Критериями поиска могут служить: имя (название) автора/исполнителя или название композиции. В качестве результата поиска в консоль должен выводиться список композиций в виде «исполнитель – название». Удаление или добавление записи осуществляется после ввода всей информации о композиции.

**Сформулированные функциональные требования**

1. Музыкальная композиция содержит следующие критерии: название, автор, жанр. Каждый критерий является отдельной сущностью БД: Композиция (название, автор, жанр); Автор (имя); Жанр (название).
2. Все данные хранятся в БД под управлением СУБД postgresql
3. Сочетание названия и автора композиции должно быть уникальным. Имя автора должно быть уникальным. Название жанра должно быть уникальным.
4. Добавление новой композиции в каталог. При добавлении новой композиции пользователь передает название композиции, автора и жанр. В случае если автор не существует данные заносятся в соответствующую таблицу, в случае если существует, данные в таблице “автор”, не дублируются, при создании композиции, в поле “автор” заносится ссылка на существующая запись. Запись о жанре должна содержаться в соответствующей таблице, в случае если запись не существует, новая композиция не создается.
5. Добавление нового жанра. Пользователь передает название жанра, если данный жанр уже существует, новая запись не создается.
6. Вывести информацию о всех существующих композициях.
7. Вывести все композиции одного автора, имя автора передается пользователем.
8. Найти композицию по названию, пользователь передает название композиции и имя автора.
9. Удалить существующую в каталоге запись о композиции, пользователь передает название композиции и имя автора.
10. Поиск композиций по жанру, пользователь передает название жанра.
11. Вывести информацию о всех существующих авторах.
12. Вывести информацию о всех существующих жанров.
13. Создание плейлиста, плейлист является сущностью БД: Плейлист (название). Связан с сущностью “Композиция” связью многие ко многим. Пользователь передает название плейлиста.
14. Добавление существующей композиции в плейлист. Пользователь передает название плейлиста, название композиции и имя автора.
15. Удаление композиции из плейлиста, пользователь передает название плейлиста, название композиции и имя автора.
16. Возможность удалить плейлист. Пользователь передает название плейлиста.
17. Просмотр названий всех плейлистов.
18. Просмотр всех композиций в плейлисте, пользователь передает название плейлиста.
19. При запуске приложение пользователь получает справочную информацию по использованию.
20. При переходе в любой блок меню пользователю выводится информация в виде списка названий опций, каждая опция – переход в другой блок меню. Для перехода пользователю необходимо ввести название соответствующей опции.