МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное автономное образовательное

учреждение высшего образования

«ЮЖНЫЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

ИНСТИТУТ ВЫСОКИХ ТЕХНОЛОГИЙ И ПЬЕЗОТЕХНИКИ

Кафедра прикладной информатики и инноватики

КУРСОВАЯ РАБОТА

по дисциплине

«Архитектура информационных систем»

на тему

«Сервис для поиска клиентов\исполнителей в сфере фотографии»

Выполнил

Студент 4 курса 7 группы

Говор Евгений Николаевич

Проверил

Профессор, д.т.н.

Жмайлов Борис Борисович

Ростов-на-Дону

2021

Оглавление

[Техническое задание 3](#_Toc66358828)

[Диаграмма вариантов использования 4](#_Toc66358829)

[Диаграмма компонент 16](#_Toc66358830)

[Тестирование 16](#_Toc66358831)

[Скриншоты тестов: 30](#_Toc66358832)

# Техническое задание

Наименование проекта: Сервис для поиска клиентов\исполнителей в сфере фотографии

Назначение: создание учетных записей, пользователей и фотографов. Получение объектов из различных источников хранения данных.

Функции:

1. Создание, редактирование, удаление, чтение данных пользователей, фотографов
2. Поиск пользователей или фотографов, событий
3. Создание, редактирование, удаление, чтение данных событий, создаваемых пользователями
4. Добавление, редактирование, удаление, чтение фотографий в портфолио пользователей, а также вывод портфолио
5. Просмотр фотографий по запросу
6. Добавление комментариев, оценок, отзывов и их получение

# Диаграмма вариантов использования

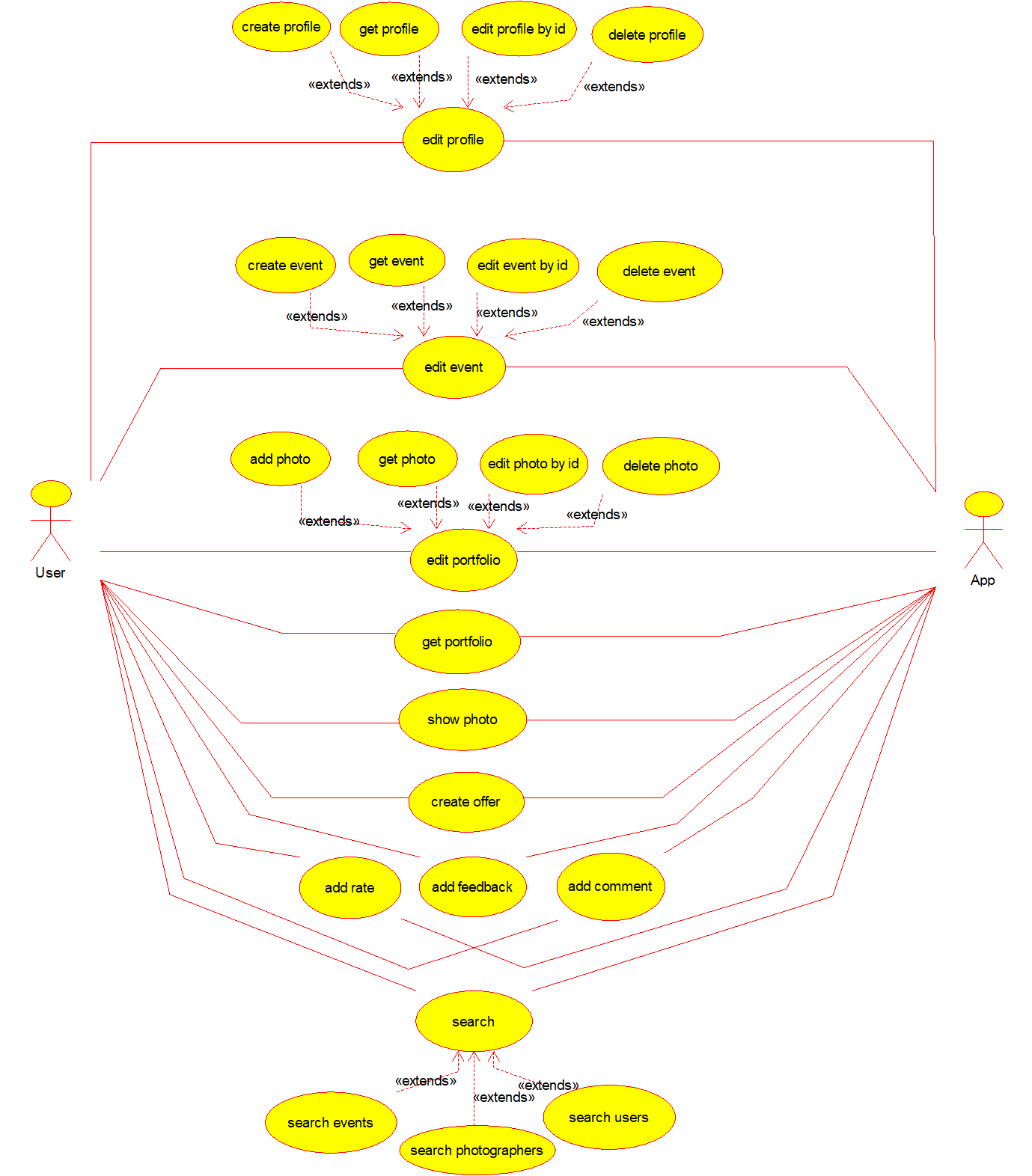


Рисунок 1. Диаграмма вариантов использования

Таблица 1. Детализация вариантов использования

|  |  |
| --- | --- |
| **Вариант использования** | **Детализация** |
| **CREATE\_NEW\_PROFILE** | Используется для создания профилей пользователей  ***ВХОД:***  String name - имя пользователя String lastName - фамилия пользователя Date birthday - дата рождения пользователя Date dateOfRegistration - дата регистрации пользователя Role role – роль пользователя String town – город проживания  ***ВЫХОД:***  Boolean - true - метод выполнен успешно  false - метод выполнен с ошибками |
| **GET\_PROFILE** | Используется для получения объекта пользователя User  ***ВХОД:***  String id – ID профиля  ***ВЫХОД:***  Optional<User>, если данный ID существует |
| **EDIT\_PROFILE** | Используется для чтения и изменения объекта пользователя User  ***ВХОД:***  String id - ID профиля  String field – изменяемое поле  String value – новое значение для поля  ***ВЫХОД:***  Boolean - true - метод выполнен успешно  false - метод выполнен с ошибками |
| **DELETE\_PROFILE** | Используется для удаления объекта пользователя User  ***ВХОД:***  String id - ID профиля  ***ВЫХОД:***  Boolean - true - метод выполнен успешно  false - метод выполнен с ошибками |
| **CREATE\_NEW\_EVENT** | Создание нового события  ***ВХОД:***  String title - название события  String description - описание события  String customer - пользователь, создавший событие  Date eventDate - дата начала события  Date creationDate - дата создания события  Integer price - цена события  Float quantity - кол-во часов  EventTypr type - тип события  ***ВЫХОД:***  Boolean: true - метод выполнен успешно  false - метод выполнен с ошибками |
| **GET\_EVENT** | Создание события по ID  ***ВХОД:***  String id - ID события  ***ВЫХОД:***  Optional<Event>, если данный ID существует |
| **EDIT\_EVENT** | Используется для чтения и изменения объекта пользователя Event  ***ВХОД:***  String id – ID события  String field – изменяемое поле  String value – новое значение для поля  ***ВЫХОД:***  Boolean - true - метод выполнен успешно  false - метод выполнен с ошибками |
| **DELETE\_EVENT** | Используется для удаления объекта события Event  ***ВХОД:***  String id - ID события  ***ВЫХОД:***  Boolean - true - метод выполнен успешно  false - метод выполнен с ошибками |
| **ADD\_PHOTO** | Используется для добавления объекта фотографий Photo  ***ВХОД:***  String id - ID пользователя  String path – путь к файлу  ***ВЫХОД:***  Boolean - true - метод выполнен успешно  false - метод выполнен с ошибками |
| **GET\_PHOTO** | Используется для добавления объекта фотографий Photo  ***ВХОД:***  String id - ID пользователя  ***ВЫХОД:***  Optional<Photo>, если данный ID существует |
| **EDIT\_PHOTO** | Используется для чтения и изменения объектов фотографий Photo  ***ВХОД:***  String id – ID фото  String field – изменяемое поле  String value – новое значение для поля  ***ВЫХОД:***  Boolean - true - метод выполнен успешно  false - метод выполнен с ошибками |
| **DELETE\_PHOTO** | Используется для удаления объектов фотографий Photo  ***ВХОД:***  String id - ID события  ***ВЫХОД:***  Boolean - true - метод выполнен успешно  false - метод выполнен с ошибками |
| **GET\_PORTFOLIO** | Используется для получения всех объектов фотографий Photo конкретного пользователя  ***ВХОД:***  String id - ID пользователя  ***ВЫХОД:***  Optional<ArrayList<Photo>> - все фотографии пользователя, если данный ID существует |
| **SHOW\_PHOTO** | Используется для просмотра фотографий  ***ВХОД:***  String path - путь фотографии  (path берется из метода getPhotoPathById, куда подается ID фотографии, а на выходе получаем путь к файлу)  ***ВЫХОД:***  Показ фотографии в окне |
| **CREATE\_OFFER** | Используется для создания заявки на выполнение событий  ***ВХОД:***  String userId - ID пользователя (заказчика)  String eventide - ID события, на который идет заявка  String photographerId - ID фотографа, которой откликается на заявку  Boolean isActive - состояние заявки  Date creationDate - дата создания заявки  ***ВЫХОД:***  Boolean - true - метод выполнен успешно  false - метод выполнен с ошибками |
| **ADD\_FEEDBACK** | Используется для создания отзыва о фотографе  ***ВХОД:***  String userId - ID пользователя, который оставляет отзыв  String photographerId - ID фотографа, к которой адресован отзыв  Integer rate - оценка  String text - текст отзыва  Date creationDate - дата создания отзыва  ***ВЫХОД:***  Boolean - true - метод выполнен успешно  false - метод выполнен с ошибками |
| **ADD\_COMMENT** | Используется для создания комментария о фотографии  ***ВХОД:***  String userId - ID пользователя, который оставляет комментарий  String photoId- ID фотографии, к которой адресован комментарий  String comment- текст комментария  Date date- дата создания комментария  ***ВЫХОД:***  Boolean - true - метод выполнен успешно  false - метод выполнен с ошибками |
| **SEARCH\_USERS** | Используется для поиска пользователей по какому-либо полю  ***ВХОД:***  String field - искомое поле  String value - искомое значение  ***ВЫХОД:***  Optional<ArrayList<User>> - все объекты пользоватей, соответствующие критерию поиска |
| **SEARCH\_PHOTOGRAPHERS** | Используется для поиска фотографов по какому-либо полю  ***ВХОД:***  String field - искомое поле  String value - искомое значение  ***ВЫХОД:***  Optional<ArrayList<Photographer>> - все объекты фотографов, соответствующие критерию поиска |
| **SEARCH\_EVENTS** | Используется для поиска событий по какому-либо полю  ***ВХОД:***  String field - искомое поле  String value - искомое значение  ***ВЫХОД:***  Optional<ArrayList<Event>> - все объекты событий, соответствующие критерию поиска |

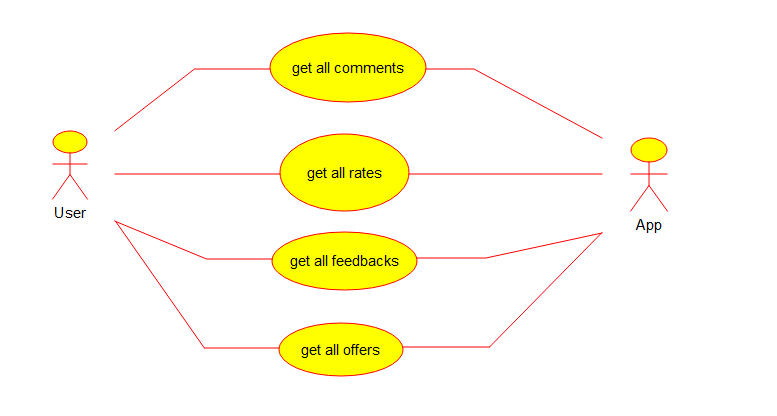


Рисунок 2. Диаграмма вариантов использования

Таблица 2. Детализация вариантов использования

|  |  |
| --- | --- |
| **Вариант использования** | **Детализация** |
| **GET\_ALL\_OFFERS** | Используется для получения всех заявок по ID события  ***ВХОД:***  String eventId - ID события  ***ВЫХОД:***  Optional<ArrayList<Offer>> - все объекты заявок данного события |
| **GET\_ALL\_COMMENTS** | Используется для получения всех комментариев по ID фотографии  ***ВХОД:***  String photoId - ID фотографии  ***ВЫХОД:***  Optional<ArrayList<Comment>> - все объекты комментариев к данному фото |
| **GET\_ALL\_RATES** | Используется для получения всех оценок по ID фотографии  ***ВХОД:***  String photoId - ID фотографии  ***ВЫХОД:***  Optional<ArrayList<Rate>> - все объекты оценок к данному фото |
| **GET\_ALL\_FEEDBACKS** | Используется для получения всех отзывов по ID фотографа  ***ВХОД:***  String photographerId - ID фотографа  ***ВЫХОД:***  Optional<ArrayList<Feedback>> - все объекты отзывов данного фотографа |

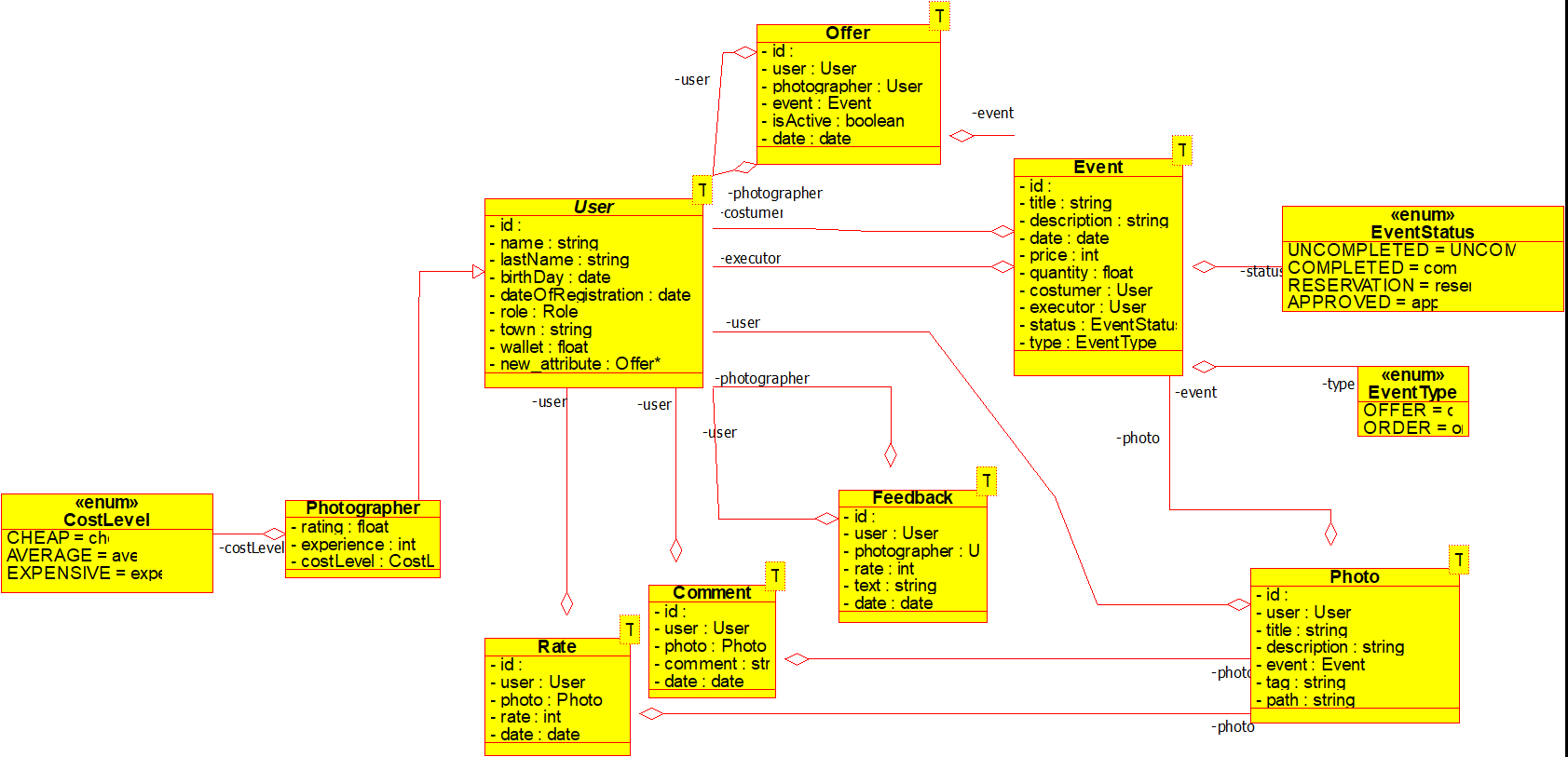
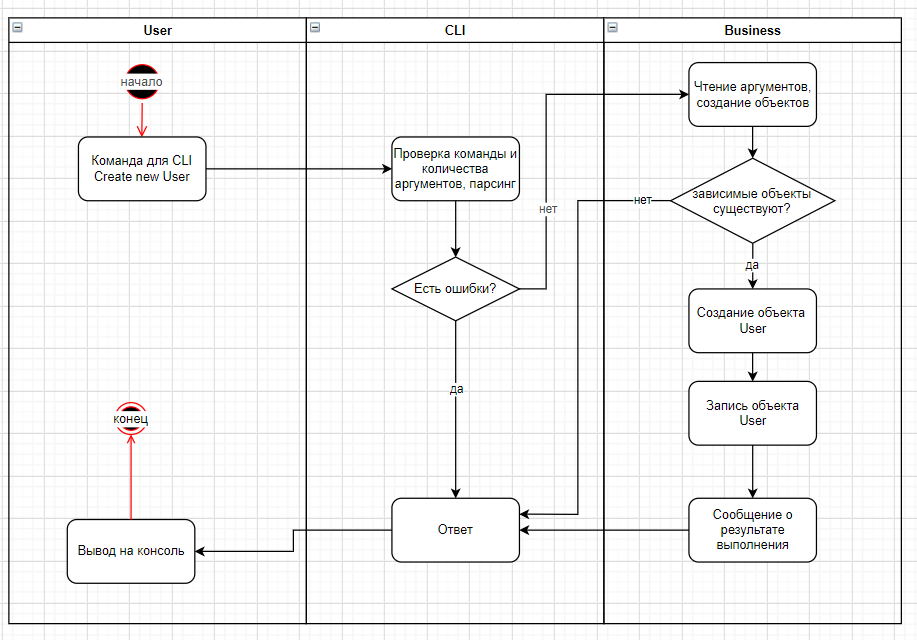
Диаграмма классов

Рисунок 3. Диаграмма классов

Диаграмма деятельности

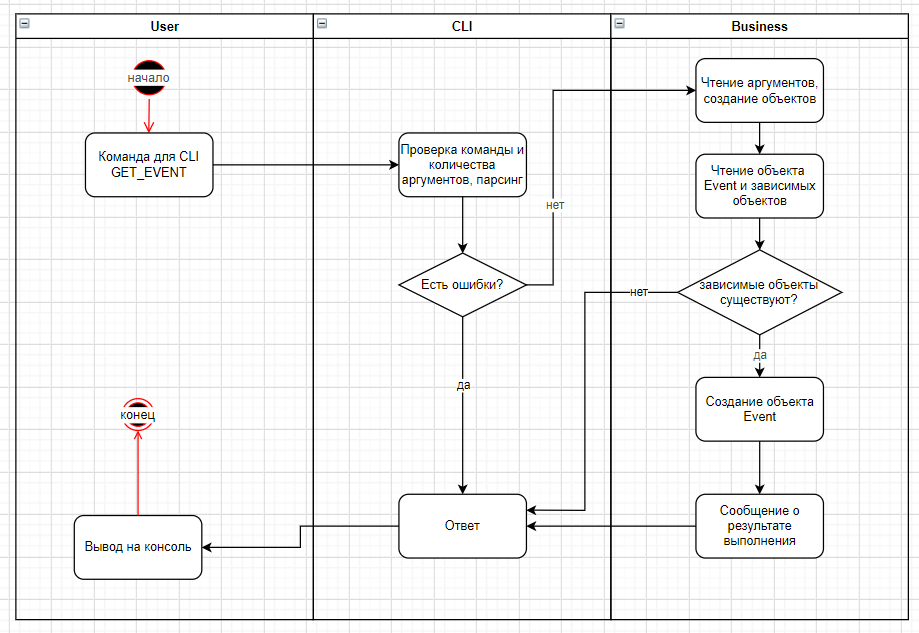
Рисунок 4. Диаграмма деятельности метода CREATE\_NEW\_USER

Рисунок 5. Диаграмма деятельности метода GET\_EVENT

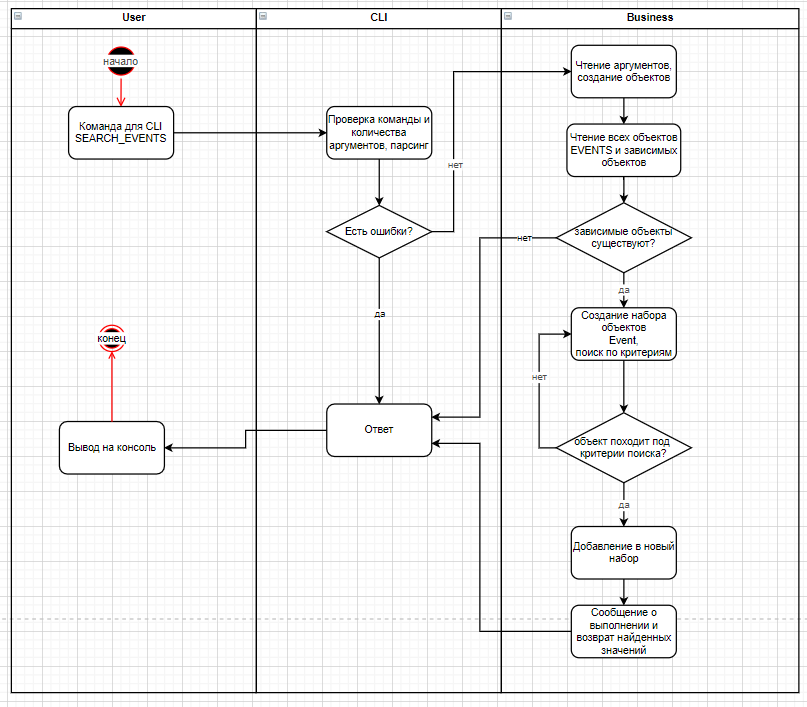


Рисунок 6. Диаграмма деятельности метода SEARCH\_EVENTS

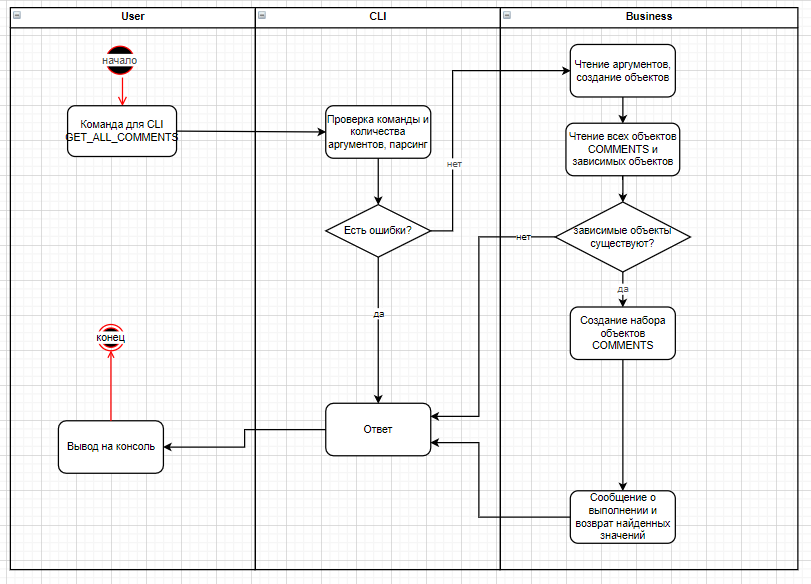
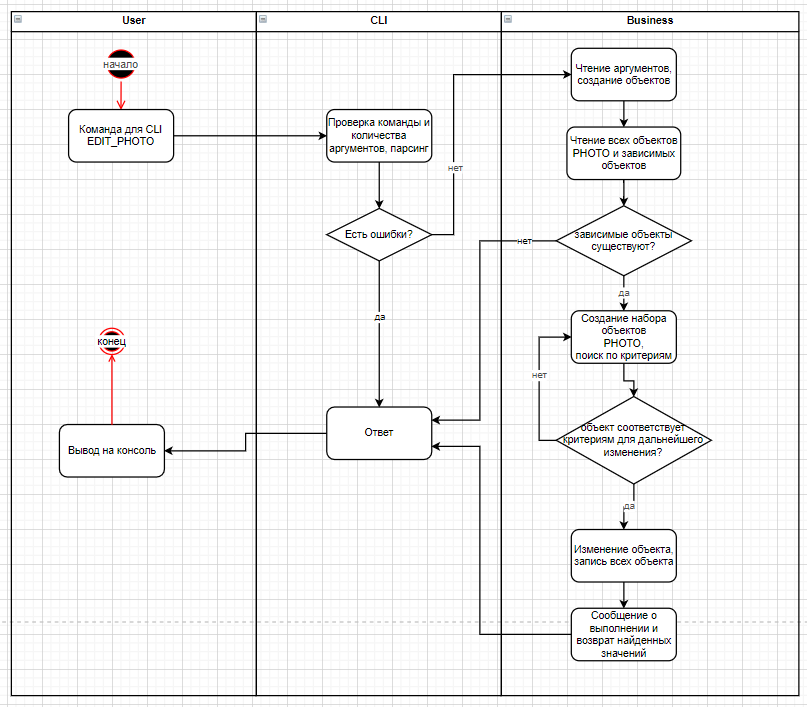
Рисунок 7. Диаграмма деятельности метода GET\_ALL\_COMMENTS

Рисунок 8. Диаграмма деятельности метода EDIT\_PHOTOS

# Диаграмма компонент

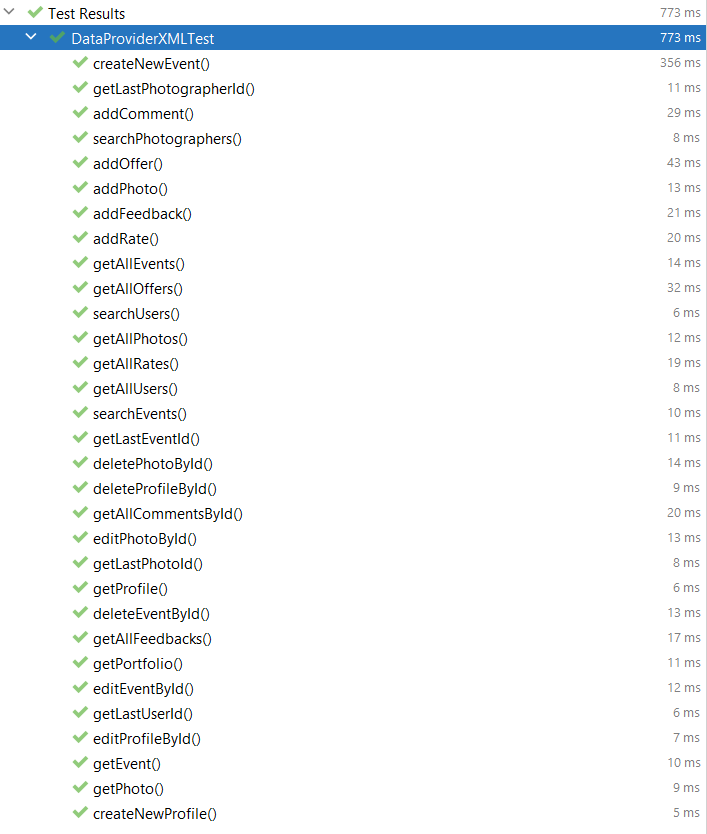
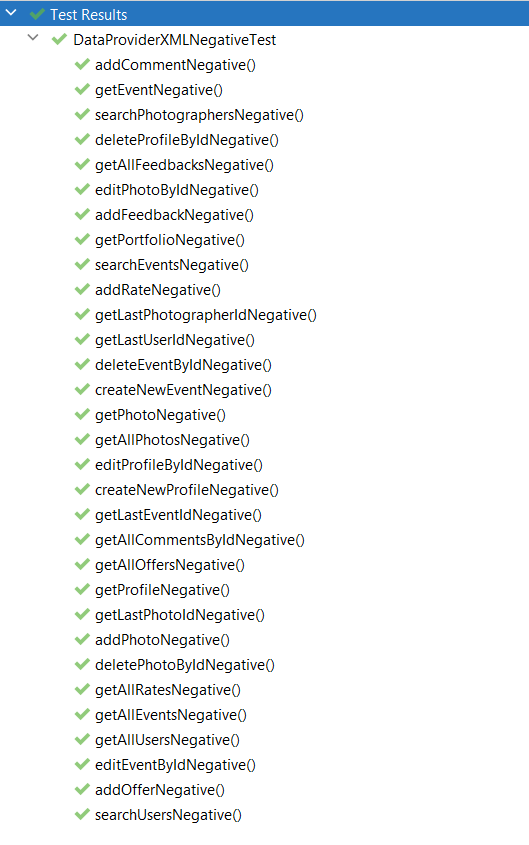
Рисунок 9. Диаграмма компонент

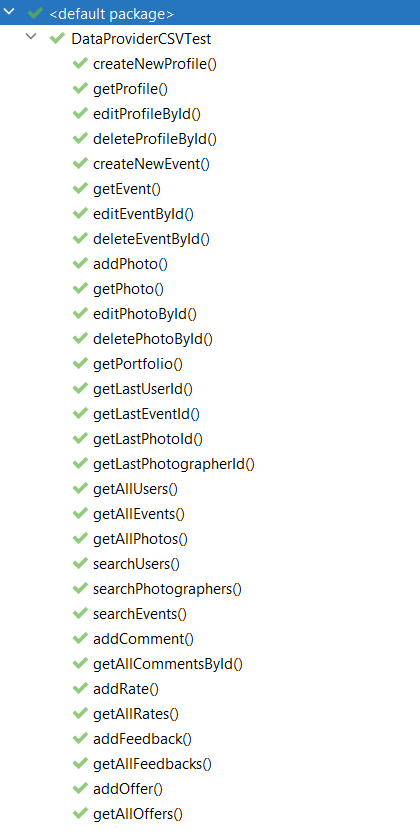
# Тестирование

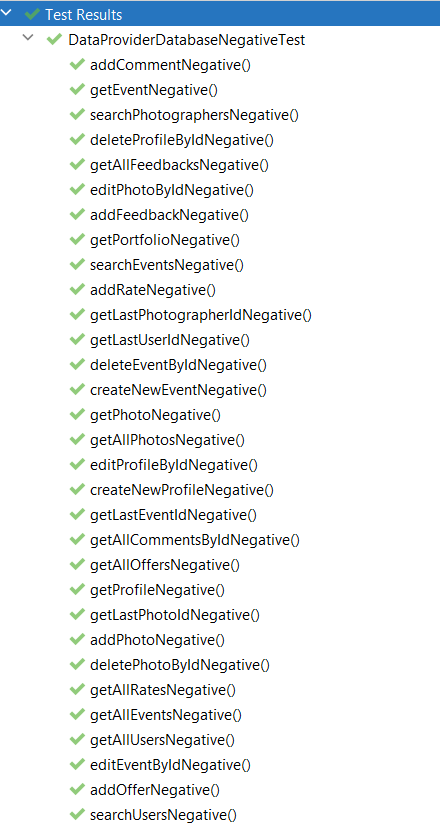
Таблица 3. Тестирование

|  |  |  |  |
| --- | --- | --- | --- |
| **Тестовый метод** | **Тип сценария** | **Тестируемый метод** | **Что контролируется?** |
| createNewProfile  Создание пользователя | Позитивный | Boolean **createNewProfile**(String name,String lastName,Date birthDay,Date dateOfRegistration,Role role,String town); | 1. Создание объекта и запись   CLI: CREATE\_NEW\_PROFILE Sergey Esenin 12-10-1999 customer Moscow  assertTrue(result) |
| createNewProfileNegative  Создание пользователя | Негативный | Boolean **createNewProfile**(String name,String lastName,Date birthDay,Date dateOfRegistration,Role role,String town); | 1. Создание объекта и запись (ошибка в указании роли)   CLI: CREATE\_NEW\_PROFI Sergey Esenin 12-10-1999 cola Moscow  **assertFalse**(result)   1. Создание объекта и запись (ошибка в методе)   CLI: CREATE\_NEW\_PROFI Sergey Esenin 12-10-1999 customer Moscow  **assertFalse**(result) |
| getProfile  Получение объекта пользователя | Позитивный | Optional<User> getProfile(String id) | 1. Получение объекта по cуществующему Id   GET\_PROFILE " + provider.getLastUserId() (вернет последний существующий ID)  **assertTrue**(result) |
| getProfileNegative  Получение объекта пользователя | Негативный | Optional<User> getProfile(String id) | 1. Получение объекта по несуществующему ID   GET\_PROFILE 5000  **assertFalse**(result) |
| editProfileById  Изменение объекта пользователя | Позитивный | Boolean editProfileById(String id, String field, String value); | 1. Получение объекта по cуществующему Id, изменение существующего поля   EDIT\_PROFILE " + provider.getLastUserId() + " name Alexey (вернет последний существующий ID)  **assertTrue**(result) |
| editProfileById  Изменение объекта пользователя | Негативный | Boolean editProfileById(String id, String field, String value); | 1. Получение объекта по неcуществующему Id, изменение существующего поля   EDIT\_PROFILE 5000 name Alexey  **assertFalse**(result) |
| deleteProfileById  Удаление объекта пользователя | Позитивный | Boolean deleteProfileById(String id); | 1. Удаление объекта по cуществующему Id   DELETE\_PROFILE " + provider.getLastUserId() (вернет последний существующий ID)  **assertTrue**(result) |
| deleteProfileByIdNegative  Удаление объекта пользователя | Негативный | Boolean deleteProfileById(String id); | 1. Получение объекта по неcуществующему Id, изменение существующего поля   DELETE\_PROFILE 5000  **assertFalse**(result) |
| createNewEvent  Создание нового события | Позитивный | Boolean createNewEvent(String title,String description,String customer,Date eventDate,Date creationDate,Integer price,Float quantity,EventType type); | 1. Создание объекта по неcуществующему Id пользователя   CREATE\_NEW\_EVENT auto\_race competition " + provider.getLastUserId() + " 14-02-2021 150 2.5 (вернет последний существующий ID пользователя)  **assertTrue**(result) |
| createNewEventNegative  Создание нового события | Негативный | Boolean createNewEvent(String title,String description,String customer,Date eventDate,Date creationDate,Integer price,Float quantity,EventType type); | 1. Создание объекта по неcуществующему Id пользователя   CREATE\_NEW\_EVENT auto\_race competition 5000 14-02-2021 150 2.5  **assertFalse**(result) |
| getEvent  Получение объекта события | Позитивный | Optional<Event> getEvent(String id); | 1. Получение объекта по cуществующему Id события   GET\_EVENT " + provider.getLastEventId() (вернет последний существующий ID события)  **assertTrue**(result) |
| getEventNegative  Получение объекта события | Негативный | Optional<Event> getEvent(String id);); | 1. Создание объекта по неcуществующему Id события   GET\_EVENT 5000 **assertFalse**(result) |
| editEventById  Получение объекта события | Позитивный | Boolean editEventById(String id, String field, String value); | 1. Изменение объекта по cуществующему Id события   EDIT\_EVENT " + provider.getLastEventId() + " title racing (вернет последний существующий ID события)  **assertTrue**(result) |
| editEventByIdNegative  Получение объекта события | Негативный | Boolean editEventById(String id, String field, String value); | 1. Изменение объекта по неcуществующему Id события   EDIT\_EVENT 5000 title racing **assertFalse**(result) |
| deleteEventById  Получение объекта события | Позитивный | Boolean deleteEventById(String id); | 1. Удаление объекта по cуществующему Id события   DELETE\_EVENT " + provider.getLastEventId(); (вернет последний существующий ID события)  **assertTrue**(result) |
| deleteEventByIdNegative  Получение объекта события | Негативный | Boolean deleteEventById(String id); | 1. Изменение объекта по неcуществующему Id события   DELETE\_EVENT 5000 **assertFalse**(result) |
| addPhoto  Создание и запись объекта фотографии | Позитивный | Boolean addPhoto(String id, String path); | 1. Создание объекта по cуществующему Id пользователя   ADD\_PHOTO " + provider.getLastUserId() + " ./testPhotos/test2.jpg" (вернет последний существующий ID пользователя)  **assertTrue**(result) |
| addPhotoNegative  Создание и запись объекта фотографии | Негативный | Boolean addPhoto(String id, String path); | 1. Изменение объекта по неcуществующему Id пользователя   ADD\_PHOTO 5000 ./testPhotos/test2.jpg **assertFalse**(result) |
| getPhoto  Получение объекта фотографии | Позитивный | Optional<Photo> getPhoto(String id); | 1. Получение объекта по cуществующему Id фото   GET\_PHOTO " + provider.getLastPhotoId() (вернет последний существующий ID фотографии)  **assertTrue**(result) |
| getPhotoNegative  Получение объекта фотографии | Негативный | Optional<Photo> getPhoto(String id); | 1. Получение объекта по неcуществующему Id фотографии   GET\_PHOTO 5000 **assertFalse**(result) |
| editPhotoById  Изменение объекта фотографии | Позитивный | Boolean editPhotoById(String id, String field, String value); | 1. Изменение объекта по cуществующему Id фото   EDIT\_PHOTO " + provider.getLastPhotoId() + " title test\_photo" (вернет последний существующий ID фотографии)  **assertTrue**(result) |
| editPhotoByIdNegative  Изменение объекта фотографии | Негативный | Boolean editPhotoById(String id, String field, String value); | 1. Изменение объекта по неcуществующему Id фотографии   EDIT\_PHOTO 5000 title test\_photo" **assertFalse**(result) |
| deletePhotoById  Удаление объекта фотографии | Позитивный | Boolean deletePhotoById(String id); | 1. Удаление объекта по cуществующему Id фото   DELETE\_PHOTO " + provider.getLastPhotoId(); " (вернет последний существующий ID фотографии)  **assertTrue**(result) |
| deletePhotoByIdNegative  Удаление объекта фотографии | Негативный | Boolean deletePhotoById(String id); | 1. Получение объекта по неcуществующему Id фотографии   DELETE\_PHOTO 5000 **assertFalse**(result) |
| getPortfolio  Получение портфолио | Позитивный | Optional<ArrayList<Photo>> getPortfolio(String userId); | 1. Получение портфолио по cуществующему Id пользователя   GET\_PORTFOLIO " + provider.getLastPhotoId(); " (вернет последний существующий ID фотографии)  **assertTrue**(result) |
| getPortfolioNegative  Получение портфолио | Негативный | Optional<ArrayList<Photo>> getPortfolio(String userId); | 1. Получение портфолио по cуществующему Id пользователя   **assertFalse(result)** |
| getLastUserId  Получение последнего ID пользователя | Позитивный | String getLastUserId(); | 1. Получение ID по cуществующему Id пользователя   **assertNotNull** (result) |
| getLastUserIdNegative  Получение последнего ID пользователя | Негативный | String getLastUserId(); | 1. Получение ID по неcуществующему Id пользователя, так как файлы будут пусты   **assertNull** (result) |
| getLastEventId  Получение последнего ID события | Позитивный | String getLastEventId(); | 1. Получение ID по cуществующему Id события   **assertNotNull** (result) |
| getLastEventIdNegative  Получение последнего ID события | Негативный | String getLastEventId(); | 1. Получение ID по неcуществующему Id события, так как файлы будут пусты   **assertNull** (result) |
| getLastEventId  Получение последнего ID фото | Позитивный | String getLastPhotoId(); | 1. Получение ID по cуществующему Id фото   **assertNotNull** (result) |
| getLastEventIdNegative  Получение последнего ID фото | Негативный | String getLastPhotoId(); | 1. Получение ID по неcуществующему Id фото, так как файлы будут пусты   **assertNull** (result) |
| getLastPhotographerId  Получение последнего ID фото | Позитивный | String getLastPhotographerId(); | 1. Получение ID по cуществующему Id фотографу   **assertNotNull** (result) |
| getLastPhotographerIdNegative  Получение последнего ID фото | Негативный | String getLastPhotographerId(); | 1. Получение ID по неcуществующему Id фотографу, так как файлы будут пусты   **assertNull** (result) |
| getAllUsers  Получение всех пользователей | Позитивный | Optional<ArrayList<User>> getAllUsers(); | 1. Получение объектов по cуществующим Id пользователей   **assertNotNull** (result) |
| getAllUsersNegative  Получение всех пользователей | Негативный | Optional<ArrayList<User>> getAllUsers(); | 1. Получение объектов по неcуществующим Id пользователей, так как файлы будут пусты   **assertNull** (result) |
| getAllEvents  Получение всех событий | Позитивный | Optional<ArrayList<Event>> getAllEvents(); | 1. Получение объектов по cуществующим Id событий   **assertNotNull** (result) |
| getAllEventsNegative  Получение всех событий | Негативный | Optional<ArrayList<Event>> getAllEvents(); | 1. Получение объектов по неcуществующим Id событий, так как файлы будут пусты   **assertNull** (result) |
| getAllPhotos  Получение всех фотографий | Позитивный | Optional<ArrayList<Photo>> getAllPhotos(); | 1. Получение объектов по cуществующим Id фотографий   **assertNotNull** (result) |
| getAllPhotosNegative  Получение всех фотографий | Негативный | Optional<ArrayList<Photo>> getAllPhotos(); | 1. Получение объектов по неcуществующим Id фотографий, так как файлы будут пусты   **assertNull** (result) |
| addComment  Создание комментария | Позитивный | Boolean addComment(String userId, String photoId, String comment, Date date); | 1. Создание объектов по cуществующим Id пользователя   ADD\_COMMENT " + provider.getLastUserId() + " " + provider.getLastPhotoId() + " beautiful\_photo"  **assertTrue**(result) |
| addCommentNegative  Создание комментария | Негативный | Boolean addComment(String userId, String photoId, String comment, Date date); | 1. Создание объектов по неcуществующим Id пользователя   ADD\_COMMENT 5000" " + provider.getLastPhotoId() + " beautiful\_photo"  **assertFalse** (result) |
| addRate  Создание оценки | Позитивный | Boolean addRate(String userId, String photoId, Integer rate, Date date); | 1. Создание объектов по cуществующим Id пользователя   ADD\_RATE " + provider.getLastUserId() + " " + provider.getLastPhotoId() + " 5"  **assertTrue**(result) |
| addRateNegative  Создание оценки | Негативный | Boolean addRate(String userId, String photoId, Integer rate, Date date); | 1. Создание объектов по неcуществующим Id пользователя   ADD\_RATE 5000" + provider.getLastPhotoId() + " 5"  **assertFalse** (result) |
| addFeedback  Создание отзыва | Позитивный | Boolean addFeedback(String userId,String photographerId,Integer rate,String text,Date creationDate); | 1. Создание объектов по cуществующим Id пользователя   ADD\_FEEDBACK " + userId + " " + photographerId + " 5 thank\_you"  **assertTrue**(result) |
| addFeedbackNegative  Создание отзыва | Негативный | Boolean addFeedback(String userId,String photographerId,Integer rate,String text,Date creationDate); | 1. Создание объектов по неcуществующим Id пользователя   ADD\_FEEDBACK 5000" + " " + photographerId + " 5 thank\_you"  **assertFalse** (result) |
| addOffer  Создание заявки | Позитивный | Boolean createOffer(String userId,String eventId,String photographerId,Boolean isActive,Date creationDate); | 1. Создание объектов по cуществующим Id пользователя   CREATE\_OFFER " + userId + " " + eventId + " " + photographerId  **assertTrue**(result) |
| addOfferNegative  Создание заявки | Негативный | Boolean createOffer(String userId,String eventId,String photographerId,Boolean isActive,Date creationDate); | 1. Создание объектов по неcуществующим Id пользователя   CREATE\_OFFER 5000" + " " + eventId + " " + photographerId  **assertFalse** (result) |
| getAllOffers  Получение всех заявок | Позитивный | Optional<ArrayList<Offer>> getAllOffersById(String eventId); | 1. Получение объектов по cуществующему Id пользователя   GET\_ALL\_OFFERS" + provider.getLastEventId()  **assertTrue**(result) |
| getAllOffersNegative  Получение всех заявок | Негативный | Optional<ArrayList<Offer>> getAllOffersById(String eventId); | 1. Получение объектов по неcуществующему Id пользователя   GET\_ALL\_OFFERS 5000  **assertFalse** (result) |
| getAllFeedbacks  Получение всех отзывов | Позитивный | Optional<ArrayList<Feedback>> getAllFeedbacksById(String photographerId); | 1. Получение объектов по cуществующему Id фотографа   GET\_ALL\_FEEDBACKS " + provider.getLastPhotographerId()  **assertTrue**(result) |
| getAllFeedbacksNegative  Получение всех отзывов | Негативный | Optional<ArrayList<Offer>> getAllOffersById(String eventId); | 1. Получение объектов по неcуществующему Id фотографа   GET\_ALL\_FEEDBACKS 5000  **assertFalse** (result) |
| getAllRates Получение всех оценок фотографии | Позитивный | Optional<ArrayList<Rate>> getAllRatesById(String photoId); | 1. Получение объектов по cуществующему Id фотографии   GET\_ALL\_RATES " + provider.getLastPhotographerId()  **assertTrue**(result) |
| getAllRatesNegative  Получение всех заявок фотографии | Негативный | Optional<ArrayList<Rate>> getAllRatesById(String photoId); | 1. Получение объектов по неcуществующему Id фотографии   GET\_ALL\_RATES 5000  **assertFalse** (result) |
| getAllCommentsByIdПолучение всех комментариев фотографии | Позитивный | Optional<ArrayList<Comment>> getAllCommentsById(String photoId); | 1. Получение объектов по cуществующему Id фотографии   GET\_ALL\_COMMENTS " + provider.getLastPhotographerId()  **assertTrue**(result) |
| getAllCommentsByIdNegative  Получение всех заявок комментариев | Негативный | Optional<ArrayList<Comment>> getAllCommentsById(String photoId); | 1. Получение объектов по неcуществующему Id фотографии   GET\_ALL\_COMMENTS 5000  **assertFalse** (result) |
| searchUsers  Поиск пользователей | Позитивный | Optional<ArrayList<User>> searchUsers(String field, String value); | 1. Получение объектов по cуществующим Id пользователей   SEARCH\_USERS town Moscow  **assertTrue**(result) |
| searchUsersNegative  Поиск пользователей | Негативный | Optional<ArrayList<User>> searchUsers(String field, String value); | 1. Получение объектов по неcуществующим Id пользователей, так как файлы будут пусты   SEARCH\_USERS town Moscow  **assertFalse** (result) |
| searchPhotographers  Поиск фотографов | Позитивный | Optional<ArrayList<User>> searchPhotographers(String field, String value); | 1. Получение объектов по cуществующим Id фотографов   SEARCH\_PHOTOGRAPHERS town Moscow  **assertTrue**(result) |
| searchPhotographersNegative  Поиск фотографов | Негативный | Optional<ArrayList<User>> searchPhotographers(String field, String value); | 1. Получение объектов по неcуществующим Id фотографов, так как файлы будут пусты   SEARCH\_PHOTOGRAPHERS town Moscow  **assertFalse** (result) |
| searchEvents  Поиск событий | Позитивный | Optional<ArrayList<Event>> searchEvents(String field, String value); | 1. Получение объектов по cуществующим Id событий   SEARCH\_EVENTS description competition **assertTrue**(result) |
| searchEventsNegative  Поиск событий | Негативный | Optional<ArrayList<Event>> searchEvents(String field, String value); | 1. Получение объектов по неcуществующим Id событий, так как файлы будут пусты   SEARCH\_EVENTS description competition  **assertFalse** (result) |

# Скриншоты тестов:

XML

CSV

Database

