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 работы с клиентами и исполнителями в сфере фотографии»

Выполнил

Студент 4 курса 7 группы

Говор Евгений Николаевич

Проверил

Профессор, д.т.н.

Жмайлов Борис Борисович

Ростов-на-Дону

2020

**Оглавление**

[Техническое задание 3](#_Toc59869117)

[Диаграмма вариантов использования 4](#_Toc59869118)

[Диаграмма классов 11](#_Toc59869119)

Техническое задание

Наименование проекта: Система хранения учетных записей пользователей с возможностью добавления портфолио и

Назначение: упрощенное создание учетных записей, как исполнителей так и заказчиков с помощью приложения. Отслеживание заказов.

Функции:

1. Создание, редактирование, удаление, чтение данных пользователей
2. Создание, редактирование, удаление, чтение данных о портфолио пользователей
3. Добавление, редактирование, удаление, чтение фотографий в портфолио пользователей
4. Просмотр фотографий по запросу

Диаграмма вариантов использования

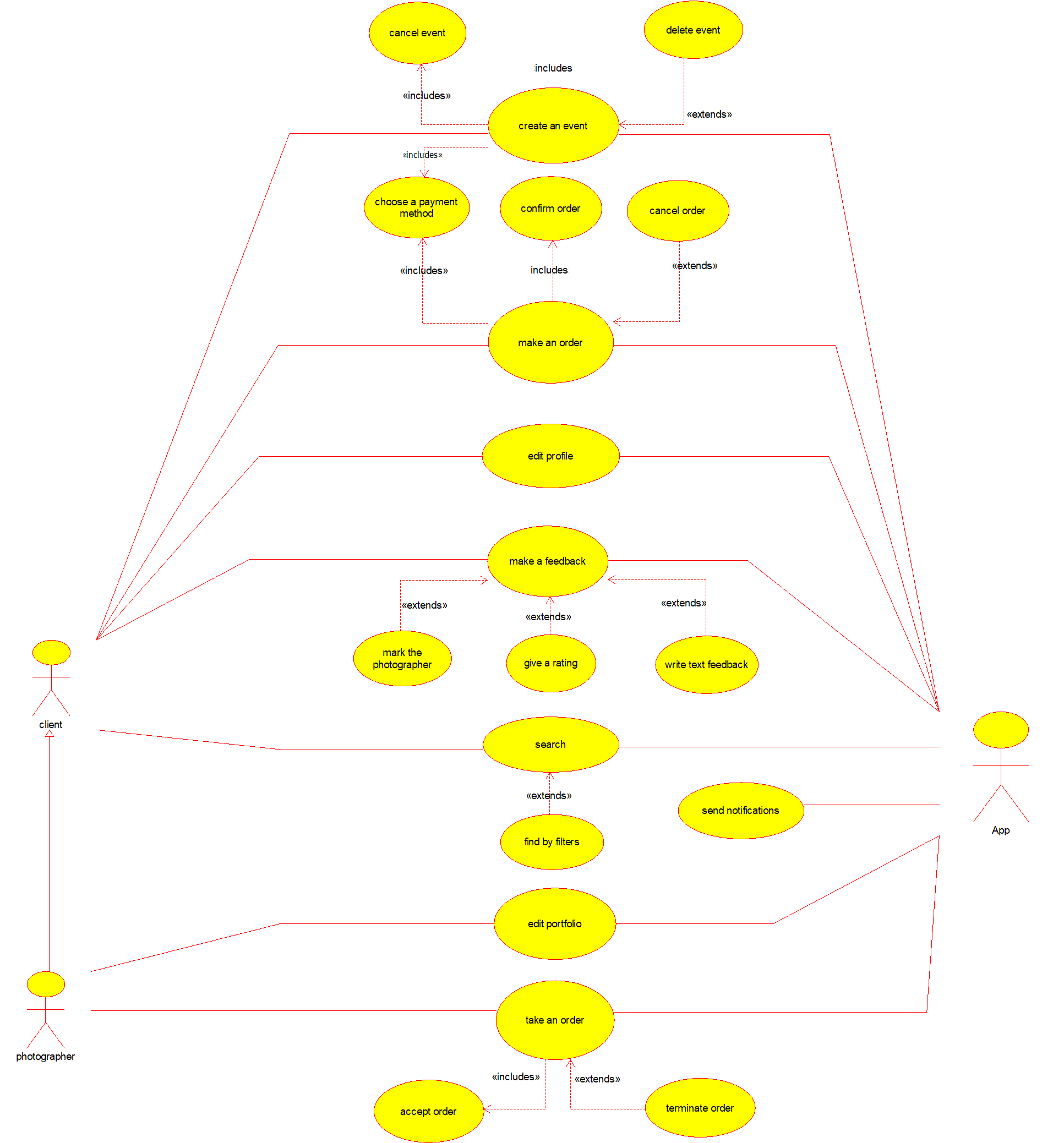


Рисунок 1. Диаграмма вариантов использования

|  |  |
| --- | --- |
| **Вариант использования** | **Детализация** |
| **CREATE\_NEW\_PROFILE** | Включает в себя использование CRUD функций для редактирования учетной записи:  **CREATE PROFILE:**  ***ВХОД:***  string name – имя пользователя  string lastname – фамилия пользователя  string birthday – дата рождения пользователя  string role – роль (фотограф/заказчик)  string profile\_name – имя профиля  string additional\_information – дополнительная информация профиля (контакты, описание)  ***ВЫХОД:***  int userID – ID только что созданного пользователя  **UPDATE PROFILE:**  ***ВХОД:***  int userID – ID пользователя  string field\_to\_change– поле учетной записи, которое требуется изменить  string value – значение, на которое требуется поменять выбранное поле  ***ВЫХОД:***  boolean status – статус изменения (True – успешно, False – неудачно)  **DELETE PROFILE:**  ***ВХОД:***  int userID – ID пользователя  ***ВЫХОД:***  boolean status – статус удаления (True – успешно, False – неудачно) |
| **GET PORTFOLIO** | Включает в себя использование CRUD функций для редактирования портфолио, если роль учетной записи выбрана как фотограф, т.е добавление, изменение, удаление фотографий.  **ADD PHOTO:**  ***ВХОД:***  int userID - ID пользователя, которому принадлежит портфолио  string filename – название файла  string caption – подпись к фотографии  int event – тип события  int mark\_users – ID других пользователей, которых можно отметить на фотографии  ***ВЫХОД:***  int postID – ID созданного поста  boolean status – статус добавления (True – успешно, False – неудачно)  **DELETE PHOTO:**  ***ВХОД:***  int postID – ID поста  ***ВЫХОД:***  boolean status – статус удаления (True – успешно, False – неудачно) |
| Действия, связанные с заказами: | |
| **CREATE EVENT** | Используется как исполнителем, так и заказчиком. Позволяет создать событие, которое может быть как предложением (со стороны фотографа), так и заказом (со стороны заказчика).  **ADD EVENT:**  ***ВХОД:***  int userID - ID пользователя, который создает событие  int eventType – услуга(0) либо заказ (1). Фотографы **по умолчанию** смогут создавать только услуги. Другие люди – заказы.  string event\_title – название события  string event\_description – подпись к фотографии  string photos\_filenames – названия файлов работ, чтобы прикрепить к событию  int event\_theme – тема события  int price – цена  ***ВЫХОД:***  int eventID – ID созданного события  boolean status – статус добавления (True – успешно, False – неудачно)  Например, если фотограф специализируется на детской фотографии, он может создать событие, дать название, выбрать тип события (детские праздники/портреты и тп), добавить примеры работ, указать цену.  Т.е фотограф создает событие-услугу. Заказчик сможет заказать эту услугу.  Также есть возможность редактировать\удалять событие с помощью CRUD – функций (DELETE EVENT, UDPATE EVENT)  **UPDATE EVENT:**  ***ВХОД:***  int eventID – ID события  string field\_to\_change– поле учетной записи, которое требуется изменить  string value – значение, на которое требуется поменять выбранное поле  ***ВЫХОД:***  boolean status – статус изменения (True – успешно, False – неудачно)  **DELETE EVENT:**  ***ВХОД:***  int eventID – ID события  ***ВЫХОД:***  boolean status – статус удаления (True – успешно, False – неудачно) |
| **CREATE EVENT** | Используется только заказчиком. Позволяет сделать заказ, т.е заказать услугу, предложенную фотографом.  **CREATE EVENT:**  ***ВХОД:***  int userID - ID пользователя, который делает заказ  int eventID – ID события  ***ВЫХОД:***  int orderID – ID созданного заказа (заявки)  boolean status – статус создания (True – успешно, False – неудачно)  После создания заказа система отправляет уведомление фотографу о новом заказе. Он может отклонить его, либо принять, тогда последуют уточнение заказа (выбор даты и времени, способа оплаты и тп.)  Это будет осуществляться с помощью методов **CHOOSE PAYMENT METHOD, CONFIRM ORDER**  **CHOOSE PAYMENT METHOD:**  ***ВХОД:***  int eventID – ID события  int payment\_method - тип оплаты  ***ВЫХОД:***  boolean status – статус изменения заказа (True – успешно, False – неудачно)  Подтвердить заказ должны обе стороны.  **CONFIRM ORDER:**  ***ВХОД:***  int eventID – ID события  int userID – ID пользователя  ***ВЫХОД:***  boolean status – статус подтверждения заказа (True – успешно, False – неудачно)  Обе стороны смогут отменить заказ с помощью метода **CANCEL ORDER**.  **CANCEL ORDER:**  ***ВХОД:***  int eventID – ID события  int userID – ID пользователя  ***ВЫХОД:***  boolean status – статус отмены заказа (True – успешно, False – неудачно) |
| **TAKE ORDER** | Используется только фотографом (исполнителем). Позволяет взять заказ, т.е выбрать событие, которое создал заказчик.  **TAKE ORDER:**  ***ВХОД:***  int userID - ID пользователя, который берет заказ  int eventID – ID события  ***ВЫХОД:***  int orderID – ID созданного заказа (заявки)  boolean status – статус создания (True – успешно, False – неудачно)  После создания заказа система отправляет уведомление заказчику о новом исполнителе. Он может отклонить его, либо принять, тогда последуют уточнение заказа (выбор даты и времени, способа оплаты и тп.)  Это будет осуществляться с помощью методов **CHOOSE PAYMENT METHOD, CONFIRM ORDER**  **CHOOSE PAYMENT METHOD:**  ***ВХОД:***  int eventID – ID события  int payment\_method - тип оплаты  ***ВЫХОД:***  boolean status – статус изменения заказа (True – успешно, False – неудачно)  Подтвердить заказ должны обе стороны.  **ACCEPT ORDER:**  ***ВХОД:***  int eventID – ID события  int userID – ID пользователя  ***ВЫХОД:***  boolean status – статус подтверждения заказа (True – успешно, False – неудачно)  Обе стороны смогут отменить заказ с помощью метода **CANCEL ORDER**.  **TERMINATE ORDER:**  ***ВХОД:***  int eventID – ID события  int userID – ID пользователя  ***ВЫХОД:***  boolean status – статус отмены заказа (True – успешно, False – неудачно) |
| **SEARCH** | Используется всеми пользователями для поиска событий. Предполагает поиск по фильтрам: город, тип события, рейтинг, цена  **SEARCH EVENTS:**  ***ВХОД:***  string query\_text – текст запроса, может быть названием искомого профиля, темой события и тп.  string filters – фильтры по которым будет вестись поиск  ***ВЫХОД:***  int eventID – ID события(ий)  **SEARCH USERS:**  ***ВХОД:***  string query\_text – текст запроса, может быть названием искомого профиля, темой события и тп.  string filters – фильтры по которым будет вестись поиск  ***ВЫХОД:***  int userID – ID пользователя(пользователей) |
| Действия, связанные с обратной связью | |
| **MAKE FEEDBACK** | Используется для создания отзывов, оценок.  **MAKE FEEDBACK:**  ***ВХОД:***  int eventID – ID события  int userID – ID пользователя  string feedback\_text – текст отзыва  int rating – оценка (от 1 до 10)  int mark\_user – отметить пользователей, участвующих в заказе  ***ВЫХОД:***  int feedbackID – ID отзыва  boolean status – статус создания отзыва (True – успешно, False – неудачно) |

Таблица 1. Детализация вариантов использования

|  |  |
| --- | --- |
| **Вариант использования** | **Детализация** |
| CREATE\_NEW\_PROFILE | Создание нового профиля пользователя для записи  Входные данные:  String id, String name, String last\_name, Integer age, String dateOfRegistration, Role role, String town, double rating  Выходные данные:  String id |
| EDIT\_PROFILE | Редактирование данных профиля пользователя  Входные данные: String id, String field, String value  Выходные данные: int result |
| DELETE\_PROFILE | Удаление профиля пользователя по его id  Входные данные: String id  Выходные данные: String msg |
| GET\_PROFILE | Получение информации о профиле пользователя по его id  Входные данные: String id  Выходные данные: String user |
| CREATE\_NEW\_EVENT | Создание нового события для записи  Указываются  Входные данные: long userId, long toyId, long materialId  Выходные данные: Boolean isDeleted |
| DELETE\_EVENT | Удаление события по его Id  Входные данные: String id  Выходные данные: String msg |
| EDIT\_EVENT | Редактирование события\заказа.  Входные данные: String id, String field, String value  Выходные данные: int result |
| GET\_EVENT | Получение информации о профиле пользователя по его id  Входные данные: String id  Выходные данные: String event |
| ADD\_PHOTO | Добавление фото в портфолио  Входные данные: String id, String path  Выходные данные: String msg |
| EDIT\_PHOTO | Редактирование фотографии по id  Входные данные: String id, String field, String value  Выходные данные: String msg |
| DELETE\_PHOTO | Удаление фотографии из портфолио по id  Входные данные: String id  Выходные данные: String msg |
| GET PHOTO | Получение информации о фотографии по ее id  Входные данные: String id  Возвращаемое значение: Photo photo |
| GET\_PORTFOLIO | Получение всех фотографий пользователя по его Id  Входные данные: String user\_id  Возвращаемое значение: String portfolio |
| SHOW\_PHOTO | Просмотр фотографии по его ID (изображение)  Входные данные: String user\_id  Возвращаемое значение: String path |

Диаграмма классов

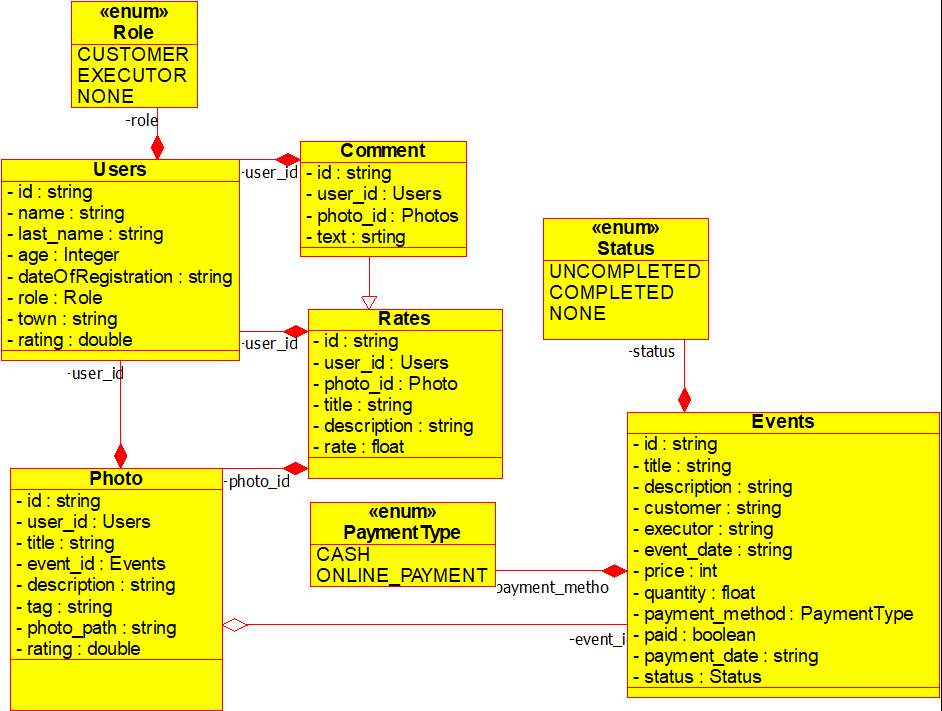


Рисунок 5. Диаграмма классов