Министерство науки и высшего образования Российской Федерации

Федеральное государственное бюджетное образовательное учреждение высшего образования

«Московский политехнический университет»

Кафедра «Инфокогнитивные технологии»

Образовательная программа «Веб-технологии»

ЛАБОРАТОРНАЯ РАБОТА №4   
ПО ДИСЦИПЛИНЕ

«Программная инженерия»

на тему:

 «Объектно-ориентированный подход к моделированию бизнес-процессов.

Модель классов и модель прецедентов»

**Выполнил:**

Студент группы 181-321

                                      Малькина А. А.

Москва, 2020

1. Назначение программы и цели ее создания

Косметологические клиники, оказывающие услуги по поддержанию и усовершенствованию естественной красоты человека, в настоящее время пользуются огромной популярностью. Каждый день они предоставляют свои услуги десяткам посетителей. Рабочие часы специалистов расписаны на несколько месяцев вперед. Однако запись на прием остается устаревшей и неудобной.

Данное программное обеспечение разрабатывается с целью упрощения создания записей на прием к косметологу, а также их учета и хранения. К тому же, данный программный продукт позволит клиентам в любое удобное время отменить запись и посмотреть историю своих посещений. Также возможность просматривать записи должна присутствовать у косметологов и администратора.

1. Перечень задач
   1. Требования к задаче «Регистрация»

Незарегистрированный пользователь может перейти на страницу регистрации по ссылке Зарегистрироваться. Указав свои данные в форме, расположенной на странице регистрации, и нажав на кнопку Зарегистрироваться, пользователь попадет на страницу авторизации при этом его данные будут сохранены в БД и в личном кабинете.

* 1. Требования к задаче «Авторизация»

Данные, введенные пользователем, сравниваются с данными, хранящимися в привязанной БД. Если введены некорректные данные (пустые поля, неправильный логин и/или пароль), пользователю будет демонстрироваться сообщение с типом ошибки. Если введены корректные данные и они существуют в БД, пользователь попадет в личный кабинет, на странице которого будет отображаться его имя и роль.

* 1. Требования к задаче «Выход»

При нажатии кнопки Выйти пользователь завершает сессию для своей учетной записи и возвращается на главную страницу.

* 1. Требования к задаче «Запись на прием»

Авторизованный клиент может записаться на прием при помощи формы, которая находится в личном кабинете. Клиент выбирает мастера, услугу, а также дату и время. После этого он получается уведомление, подтверждающее успешность создания записи. Также добавить запись о посещении может администратор по заявке клиента. Все данные вносятся в БД, косметолог становится недоступным на записанное время, другой пользователь не может забронировать этого косметолога на это же время.

* 1. Требования к задаче «Просмотр истории своих посещений клиентом»

Авторизованный клиент может посмотреть все записи своих предстоящих и прошедших посещений в личном кабинете.

* 1. Требования к задаче «Отмена записи клиентом»

Авторизованный клиент может отменить запись в личном кабинете. Запись удаляется из БД.

* 1. Требования к задаче «Просмотр списка записей косметологом»

Сотрудник может посмотреть все записи, предназначенные для него.

* 1. Требования к задаче «Просмотр списка записей администратором»

Администратор может посмотреть все записи, а также отфильтровать их для удобства по сотрудникам или по дате.

* 1. Требования к задаче «Добавление косметолога администратором»

Администратор может добавить нового сотрудника, указав его ФИО, роль в системе, список услуг, контактные данные, логин и пароль. Запись о новом сотруднике сохраняется в БД.

* 1. Требования к задаче «Редактирование информации о косметологе администратором»

Администратор может редактировать информацию о косметологе. Информация в БД обновится.

* 1. Требования к задаче «Удаление косметолога администратором»

Администратор может просматривать список косметологов и удалять сотрудников из системы. Данные о об удаленном сотруднике будут удалены из БД.

* 1. Требования к задаче «Добавление услуги администратором»

Администратор может добавить услугу, указав ее длительность и стоимость. Информация в БД обновится.

* 1. Требования к задаче «Редактирование услуги администратором»

Администратор может редактировать услугу, изменяя ее название, длительность и стоимость. Информация в БД обновится.

* 1. Требования к задаче «Удаление услуги администратором»

Администратор может просматривать список оказываемых услуг и удалять неактуальные на данный период времени. Данные об удаленной услуге будут удалены из БД.

* 1. Требования к задаче «Отмена записи администратором»

Администратор может отменить запись, указав причину. При этом информация об отмене записи будет доступна клиенту. Информация в БД обновится.

На основе вышеописанных требований составлена модель прецедентов (Рис.1) для графического описания функциональных требований к программе. Выбранная среда моделирования: draw.io.

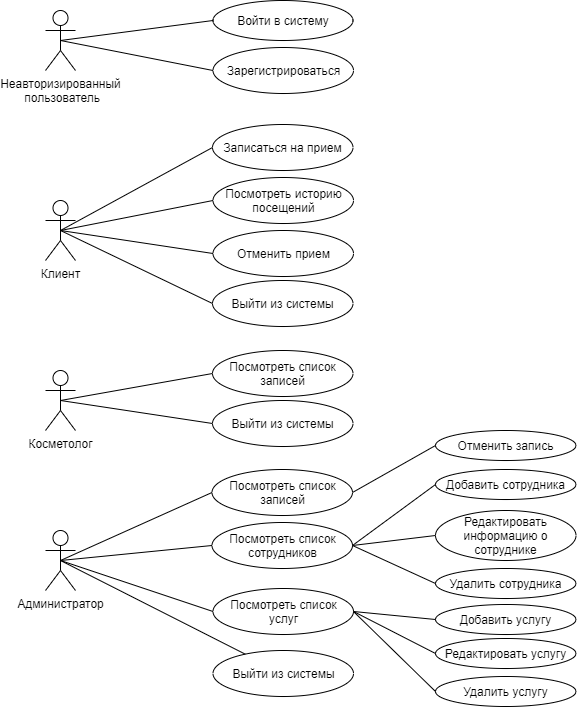


Рис.1

В данной модели присутствуют следующие сущности:

1. Клиенты (clients) – хранит id, ФИО, его контактные данные, а также логин и пароль.
2. Косметологи (beauticians) – хранит id, ФИО, должность, список услуг, его контактные данные, а также логин и пароль.
3. Услуги (services) – хранит в себе общий список всех услуг, которые оказываются в данной косметологии. Содержит поля id, название услуги, описание, длительность и стоимость.
4. Записи (appointments) – содержит поля id, дату и время, статус, а также id пользователя, id косметолога и id услуги.
5. Расписание сотрудника (timetable) – содержит дату и время, статус заявки и id записи.

На основе выделенных сущностей создана модель сущность-связь (ERD) (Рис. 2). Выбранная среда моделирования: draw.io.

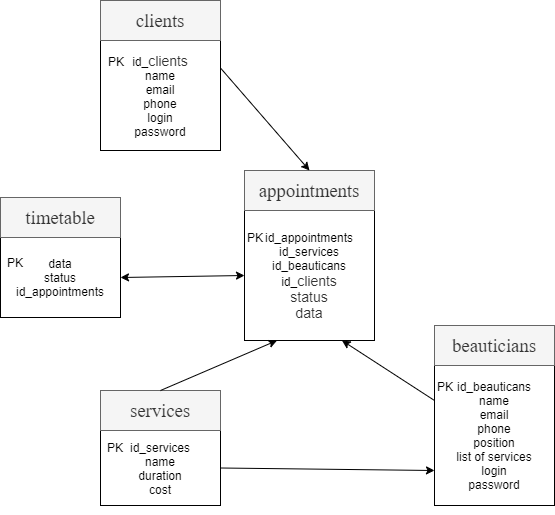


Рис.2