**МИНОБРНАУКИ РОССИИ**

**САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ**

**ЭЛЕКТРОТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ**

**«ЛЭТИ» ИМ. В.И. УЛЬЯНОВА (ЛЕНИНА)**

**Кафедра ИС**

**ЛАБОРАТОРНАЯ РАБОТА**

**по дисциплине «Архитектура информационных систем»**

**Тема: Танцевальная студия**

Студентка гр. 3395:

Преподаватель:

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Разгулова Е.Ю.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Ананьева В.Я.

Санкт-Петербург

2024

**ЗАДАНИЕ**

**на лабораторную работу**

|  |  |  |
| --- | --- | --- |
| Студентка Разгулова Е.Ю. | | |
| Группа 3395 | | |
| Тема работы: Танцевальная студия | | |
| Исходные данные: | | |
| Содержание пояснительной записки:  Пояснительная записка состоит из спецификации разработанной системы, UX моделей и UML диаграмм. | | |
| Предполагаемый объем пояснительной записки:  Не менее 15 страниц. | | |
| Дата выдачи задания: 10.02.2025 | | |
| Дата сдачи реферата: 31.03.2025 | | |
| Дата защиты реферата: 31.03.2025 | | |
| Студентка |  | Разгулова Е.Ю. |
| Преподаватель |  | Ананьева В.Я. |

**содержание**

|  |  |  |
| --- | --- | --- |
| 1. | Требования к системе | 4 |
| 1.1.  1.2.  1.3.  1.4.  1.5.  1.6.  1.7.  1.8.  1.9.  1.10. | Преамбула  Введение  Глоссарий  Пользовательские требования  Описание системной архитектуры  Системные требования  Системные модели  Возможные направления эволюции системы  Приложение  Индексы | 4  4  5  6  7  7  10  12  13  13 |
| 2.  3.  4. | Архитектурные обоснования  Модели нулевого уровня  Архитектурное описание на UML | 14  15  28 |
| 4.1. | Usecase диаграмма | 28 |
| 4.2.  4.3.  4.4. | Диаграмма классов  Диаграмма Activity  Диаграмма State machine | 39  40  41 |
| 5. | Тесты | 43 |
|  | Заключение  Список использованных источников | 44  45 |
|  |  |  |

1. **ТРЕБОВАНИЕ К СИСТЕМЕ**

**1.1. ПРЕАМБУЛА**

Информационная система для записи в танцевальную студию и просмотра расписания. Версия 1.0.

**1.2. ВВЕДЕНИЕ**

Информационная система, разрабатываемая для танцевальной студии, представляет собой комплексное программное решение, направленное на автоматизацию и оптимизацию ключевых бизнес-процессов организации. Система предназначена для обеспечения эффективного управления ресурсами, улучшения взаимодействия с клиентами и повышения рентабельности деятельности студии. Система разрабатывается как web-сайт. Пользователями системы будут сотрудники студии (администраторы и преподаватели) и клиенты (авторизованные и неавторизованные). Неавторизованный клиент может только регистрироваться на занятия и покупать абонементы, остальные функции ему не доступны.

Основные цели системы:

* Оптимизация операционной деятельности танцевальной студии посредством автоматизации рутинных задач и процессов.
* Повышение эффективности управления ресурсами, включая расписание занятий, преподавательский состав и материально-техническую базу.
* Улучшение качества обслуживания клиентов посредством предоставления удобного доступа к информации и упрощения процессов регистрации и оплаты.

Функциональные возможности системы охватывают следующие области:

* Управление клиентской базой данных (CRM).
* Управление расписанием занятий и мероприятий.
* Управление финансами и платежами.
* Управление преподавательским составом.
* Управление уведомлениями.
* Администрирование системы.

Ключевые характеристики системы:

* Система автоматизирует процессы управления расписанием, учета посещаемости, выставления счетов, рассылки уведомлений и другие рутинные задачи, что позволяет сократить трудозатраты и минимизировать вероятность ошибок.
* Система предоставляет клиентам удобные инструменты для просмотра расписания, регистрации на занятия, оплаты услуг и получения информации о студии.
* Система обеспечивает сбор, хранение и обработку данных о клиентах, занятиях, финансах и других аспектах деятельности студии.
* Система может быть адаптирована к изменяющимся потребностям студии и масштабирована по мере ее роста.

**1.3. ГЛОССАРИЙ**

Информационная система далее будет сокращаться до системы.

Абонемент – право клиента на посещение определенного количества занятий в течение установленного периода времени.

Администратор (Админ) – пользователь информационной системы с полными правами доступа, осуществляющий настройку системы, управление пользователями, расписанием, финансами и другими аспектами деятельности студии.

Архив – раздел системы, содержащий устаревшие или неактивные данные, например, расписание прошедших занятий или профили бывших клиентов..

Занятие – урок танцев, проводимый преподавателем в определенное время и в определенном зале.

Запись – процесс регистрации клиента на определенное занятие.

История посещений – список занятий, которые посетил клиент, с указанием дат и времени.

Клиент – человек, посещающий занятия в танцевальной студии.

Личный кабинет (ЛК) – Персонализированный раздел системы, доступный каждому зарегистрированному пользователю (Клиент, преподавателю, администратору), где он может просматривать и управлять своими данными, настройками и функциями, доступными в соответствии с его ролью.

Направление – конкретное танцевальное направление.

Оплата – внесение денежных средств за посещение занятий или приобретение абонемента.

Преподаватель – пользователь информационной системы, сотрудник студии, проводящий занятия, который использует ее для общения с Клиентами и составления своего рабочего расписания.

Расписание – совокупность всех занятий всех преподавателей всех направлений на неделе.

Регистрация – процесс создания личного кабинета для пользователя.

Роль – статус пользователя (администратор, Клиент, преподаватель).

Способ оплаты – вариант оплаты услуг студии (например, наличные, банковская карта).

Перемена в расписании – ситуация, когда студия вынуждена отказаться провести занятие во время расписания в связи с непредвиденными ситуациями.

Тип абонемента – разновидность абонемента, отличающаяся по цене, количеству занятий и сроку действия. Абонементы могут быть на 4/8/12 занятий в месяц и без лимитный на полгода.

Финансы – поступающие на счет студии денежные средства.

Учет – контроль количества и/или качества каких-либо характеристик.

Учетная запись – это основной набор данных, идентифицирующих пользователя в системе. Она предоставляет доступ к системе в целом.

**1.4. ПОЛЬЗОВАТЕЛЬСКИЕ ТРЕБОВАНИЯ**

Администратор должен иметь возможность:

1. Зарегистрироваться в системе.
2. Зайти в личный кабинет.
3. Редактировать и отправлять в архив учетные записи.
4. Раздавать роли новым пользователям.
5. Подтвердить запись Клиента.
6. Отслеживать оплату Клиента.
7. Просматривать и редактировать расписание.
8. Подтверждать расписание.
9. Отправлять уведомления.
10. Просматривать статистику проведенных занятий.

Клиент должен иметь возможность:

1. Просмотреть информацию о студии.
2. Зарегистрироваться в системе.
3. Зайти в личный кабинет.
4. Изменять контактную информацию о себе.
5. Записаться на нужное направление или несколько.
6. Просматривать данные о своих записях и абонементах.
7. Выбрать преподавателя.
8. Оплатить абонемент или занятие.
9. Просматривать свое расписание.
10. Отменить занятие.
11. Смотреть уведомления о перемене в расписании.

Преподаватель должен иметь возможность:

1. Зарегистрироваться в системе.
2. Зайти в личный кабинет.
3. Составить свое расписание.
4. Корректировать свое расписание.
5. Просматривать информации о клиентах.
6. Отправлять уведомления клиентам.
7. Оставлять отметки о проведенном занятии.

**1.5. ОПИСАНИЕ СИСТЕМНОЙ АРХИТЕКТУРЫ**

Система включает следящие подсистемы:

1. Подсистема управления расписанием.

Включает в себя создание и редактирование расписания, управление помещениями, просмотр расписания.

1. Подсистема управления финансами.

Включает в себя учет поступлений, подтверждение оплаты, сам процесс оплаты.

1. Подсистема управления клиентами.

Включает в себя управление информацией о клиентах (ФИО, контактный email, контактный телефон, дата рождения, возраст, уровень подготовки), управление абонементами и разовыми посещениями, история взаимодействия с клиентами (звонки и сообщения).

1. Подсистема управления преподавателями.

Включает в себя учет данных преподавателей (ФИО, контактные данные, специализация, опыт работы), учет проведенных занятий, взаимодействие с преподавателями (обмен сообщениями, уведомлениями).

1. Подсистема управления уведомлениями.

Включает в себя отправку и получение уведомлений.

Эти подсистемы должны быть интегрированы между собой для обмена данными и обеспечения целостности информации. Подсистемы управления расписанием, преподавателями, клиентами и финансами интегрируются с подсистемой уведомлений. Подсистема управления финансами использует данные из подсистемы управления с клиентами. Подсистема управления расписанием использует данные из подсистемы управления преподавателями.

В системе предусмотрены следующие интеграции:

1. Интеграция с социальными сетями (Telegram, ВКонтакте);
2. Интеграция с онлайн-календарями (Google Calendar, Outlook Calendar).

**1.6. СИСТЕМНЫЕ ТРЕБОВАНИЯ**

Функциональные требования:

1. Главная страница пользовательского интерфейса содержит информацию о студии (контактная информация, адрес и общая информация) и информацию о преподавателях и направлениях.
2. В пользовательском интерфейсе на главной странице должна быть кнопка регистрации.
3. В пользовательском интерфейсе на главной странице должна быть кнопка авторизации.
4. В пользовательском интерфейсе на главной странице должна быть кнопка записи на занятие.
5. В пользовательском интерфейсе на главной странице должна быть кнопка кнопки покупки абонемента.
6. При нажатии кнопки регистрации всплывает окно регистрации.
7. В окне регистрации должны быть поля: «Имя», «Фамилия», «Дата рождения», «Телефон», «Пароль», «Повторить пароль», - и кнопка «Зарегистрироваться».
8. В поле «Имя», «Фамилия» необходимо вводить только русские буквы и их количество может быть до 20.
9. В поле «Дата рождения» необходимо выбрать число: от 0 до 31, если месяц 1,3,5,7,8,10,12; от 0 до 30, если месяц 4,6,9,11; от 0 до 29, если месяц 2; месяц от 1 до 12, год от 1900 до 2025.
10. В поле «Телефон» необходимо ввести 11 цифр.
11. В поле «Пароль» и «Повторить пароль» нужно ввести не менее 8, но не более 20 символов, среди которых необходимо наличие цифр и латинских букв.
12. Пользователь может нажать кнопку «Зарегистрироваться», только если строки «Пароль» и «Повторить пароль» совпадают.
13. При нажатии кнопки «Зарегистрироваться» всплывает окно авторизации.
14. При нажатии кнопки авторизация всплывает окно авторизации.
15. В окне авторизации должны быть поля «Почта» и «Пароль», а также кнопка «Зайти в личный кабинет».
16. При нажатии кнопки «Зайти в личный кабинет» пользователь переходит по ссылке в личный кабинет.
17. При нажатии кнопки записи на занятии всплывает окно записи на занятие.
18. В окне записи на занятие должны быть поля «Направление», «Преподаватель» и кнопка «Записаться».
19. Поле «Направление» – это элемент управления выбором из 15 направлений студии.
20. Поле «Преподаватель» – это элемент управления выбором из 20 Преподавателей студии.
21. Нажать на кнопку «Записаться» можно, только если оба поля заполнены.
22. После записи всплывает окно оплаты.
23. При нажатии кнопки «Купить абонемент» всплывает окно выбора абонемента.
24. В окне выбора должен быть элемент управления выбором абонемента и кнопка «Оплатить».
25. При нажатии кнопки «Оплатить» всплывает окно оплаты.
26. Окно оплаты состоит из выбора способа оплаты (Наличные, Карта).
27. В личном кабинете администратора должны быть ссылки на следующие страницы: «Расписание», «Уведомления», «Отправить уведомления», «Клиенты», «Преподаватели», «Посмотреть статистику».
28. Личный кабинет преподавателя содержит ссылки на следующие страницы: «Личная информация», «Расписание», «Проведение занятий», «Клиенты», «Отправить уведомление», «Уведомления».
29. В личном кабинете у Клиента должны быть ссылки на следующие страницы: «Личная информация», «Расписание», «Уведомления», «Чат с поддержкой» и главная страница.
30. Страница «Личная информация» клиента содержит информацию, которую пользователь ввел при регистрации (Имя, Фамилия, Дата рождения, Телефон) и еще два поля Почта и Уровень подготовки.
31. Страница «Личная информация» преподавателя содержит информацию: Имя, Фамилия, Дата рождения, Телефон, Почта, Дата начала работы, Должность.
32. На странице «Личная информация» есть кнопка «Редактировать».
33. При нажатии кнопки «Редактировать» можно изменить данные.
34. В поле «Почта» можно вводить любые символы до 20.
35. В поле «Уровень подготовки» можно выбрать уровень от I до VI.
36. В поле «Должность» можно вводить только русские буквы.
37. В поле «Дата начала работы» необходимо выбрать число: от 0 до 31, если месяц 1,3,5,7,8,10,12; от 0 до 30, если месяц 4,6,9,11; от 0 до 29, если месяц 2; месяц от 1 до 12, год от 1900 до 2025.
38. Страница «Уведомления» показывает последние уведомления за неделю.
39. Страница «Проведение занятий» содержит расписание занятий на день, где к каждому занятию есть возможность поставить отметку «Проведено».
40. Страница «Клиенты» в личном кабинете преподавателя содержит информацию из личного кабинета клиентов, записавшихся на занятия к преподавателю.
41. Страница «Клиенты» в личном кабинете админа содержит информацию из личного кабинета клиентов.
42. Страница «Преподаватели» в личном кабинете админа содержит информацию из личного кабинета преподавателей.
43. Страница «Посмотреть статистику» содержит график проведенных занятий за неделю по каждому преподавателю.
44. Страница «Расписание» в личном кабинете клиента содержит готовое расписание на неделю.
45. Страница «Расписание» в личном кабинете админа содержит расписание на неделю и кнопки «Редактировать» и «Согласовать».
46. Страница «Расписание» в личном кабинете преподавателя содержит расписание на неделю и кнопки «Создать» и «Редактировать».
47. При выделении занятия администратор добавляет номер зала, в котором проходит занятие.
48. При нажатии кнопки «Согласовать» расписание публикуется в течение 30 минут.
49. При нажатии кнопки «Создать» открывается окно создания занятия.
50. Окно создания состоят из полей «Направление», «Дата», «Время».
51. Поле «Направление» – это элемент управления выбором из 15 направлений студии.
52. Поле «Время» состоит из выбора двух чисел: первое от 10 до 20, второе 00 или 30.
53. Поле «Дата» необходимо выбрать число: от 0 до 31, если месяц 1,3,5,7,8,10,12; от 0 до 30, если месяц 4,6,9,11; от 0 до 29, если месяц 2; месяц от 1 до 12, год от 1900 до 2025.
54. При нажатии кнопки «Редактировать» всплывает окно редактирования.
55. Окно редакции состоит из выбора: «Перенести занятие» и «Отменить занятие».
56. При нажатии кнопки «Перенести занятие» появляется выбор «Дата», «Временя».
57. При нажатии кнопки «Отменить занятие» занятие удаляется из расписания и переносится в архив.
58. Страница «Отправить уведомления» имеет поле для записи строки и кнопку отправить.
59. Страницу «Чат с поддержкой» имеет поле для записи строки и кнопку отправить.
60. При регистрации нового человека в системе у администратора появляется уведомление.
61. При записи на занятие у администратора появляется уведомление.
62. При покупке абонемента у администратора появляется уведомление.
63. На страницах «Клиенты» и «Преподаватели» в ЛК админа есть кнопки «Редактировать» и «Архивировать».
64. При нажатии кнопки «Архивировать» учетная запись архивируется.
65. При нажатии кнопки «Редактировать» всплывает окно редакции.
66. Окно редакции аналогично странице «Личная информация» у соответствующего пользователя.

Нефункциональные требования:

1. Неправильный пароль можно ввести не больше трёх раз, иначе вход в кабинет блокируется на минуту.
2. При взаимодействии с системой время ответа системы не больше 5 секунд.
3. Резервное копирование данных должно выполняться раз в сутки.
4. Система должна быть способна выдерживать растущую нагрузку с увеличением количества пользователей до 500 человек ежеминутно.
5. Система должна соответствовать требованиям законодательства о защите персональных данных GDPR.
6. Система должна быть адаптируема к таким устройствам, как компьютер, планшет, смартфон.

**1.7. СИСТЕМНЫЕ МОДЕЛИ**

Основные объекты и их атрибуты:

1. Клиент (Client):

* ClientID (уникальный идентификатор)
* FirstName (имя)
* LastName (фамилия)
* DateOfBirth (дата рождения)
* Email (электронная почта)
* PhoneNumber (номер телефона)
* RegistrationDate (дата регистрации)
* Level (уровень подготовки)
* MedicalRestrictions (медицинские ограничения)

1. Преподаватель (Teacher):

* TeacherID (уникальный идентификатор)
* FirstName (имя)
* LastName (фамилия)
* Email (электронная почта)
* PhoneNumber (номер телефона)
* HireDate (дата приема на работу)
* SalaryRate (почасовая ставка)

1. Занятие (Class):

* ClassID (уникальный идентификатор)
* TeacherID (идентификатор преподавателя)
* RoomID (идентификатор зала)
* DateTime (дата и время начала занятия)
* Duration (продолжительность занятия)
* Level (уровень подготовки)

1. Оплата (Payment):

* PaymentID (уникальный идентификатор)
* ClientID (идентификатор клиента)
* Date (дата оплаты)
* Amount (сумма оплаты)
* PaymentMethod (способ оплаты)
* Description (описание платежа, например, "оплата абонемента", "разовое посещение")

1. Зал (Room):

* RoomID (идентификатор зала)
* Capacity (вместимость)

1. Абонемент (Subscription):

* SubscriptionID (уникальный идентификатор)
* ClientID (идентификатор клиента)
* SubscriptionTypeID (идентификатор типа абонемента)
* PurchaseDate (дата покупки)
* ExpirationDate (дата окончания срока действия)
* NumberOfClasses (количество занятий)
* RemainingClasses (количество оставшихся занятий)
* Price (цена)

1. Тип абонемента(Subscription Type):
   * SubscriptionTypeID (идентификатор типа абонемента)
   * Type (тип абонемента)

Взаимосвязи между объектами:

* Один Клиент может иметь несколько Абонементов.
* Один Абонемент принадлежит одному Клиенту.
* Один Преподаватель может вести несколько Занятий.
* Одно Занятие ведется одним Преподавателем.
* Одно Занятие проводится в одном Зале.
* Один Зал может использоваться для проведения нескольких Занятий.
* Один Клиент может совершить несколько Оплат.
* Один Тип абонемента может быть связан с несколькими Абонементами.

Бизнес-правила:

* Способ оплаты может принимать значения: «наличные», «банковская карта».
* Клиент не может записаться на занятие, если у него нет действующего абонемента или если он не оплатил разовое посещение.
* Количество участников на занятии не должно превышать вместимость зала.
* При продаже абонемента должно быть проверено, не истек ли срок действия предыдущего абонемента клиента.
* Оплата должна быть зарегистрирована в системе с указанием клиента, даты, суммы и способа оплаты.

Семантические отношения:

1. "Клиент участвует в занятии" – это отношение между объектами Клиент и Занятие. Оно описывает факт посещения клиентом определенного занятия.
2. "Преподаватель ведет занятие" – это отношение между объектами Преподаватель и Занятие. Оно описывает ответственность преподавателя за проведение занятия.
3. "Абонемент дает право на посещение занятий" – это отношение между объектами Абонемент и Занятие. Оно описывает, что наличие абонемента позволяет клиенту посещать занятия.
4. "Клиент оплачивает занятие" – отношение между объектами Клиент и Оплата. Оно описывает факт оплаты клиентом определенного занятия.
5. "Клиент оплачивает абонемент" – отношение между объектами Клиент и Абонемент. Оно описывает факт оплаты абонемента клиентом.
6. "Занятие проходит в зале N" – отношение между объектами Занятие и Зал. Оно описывает наличие определенного зала, в котором проходит занятие.

**1.8. ВОЗМОЖНЫЕ НАПРАВЛЕНИЯ ЭВОЛЮЦИИ СИСТЕМЫ**

Направления эволюции:

* Автоматизация подтверждения расписания.
* Возможность восстановления пароля через систему.
* Отслеживание оборудования через систему.
* Формирование отчетов по финансам.
* Формирование отчетов для аналитики.
* Интеграция с платежными системами для оплаты онлайн.
* Интеграция с картами.

**1.9. ПРИЛОЖЕНИЕ**

Система двухуровневая. Состоит из пользовательского интерфейса и серверной части. Пользовательский интерфейс представлен в виде web-приложения.

**1.10. ИНДЕКСЫ**

**Бэкенд –** серверная часть системы, отвечающая за обработку данных, логику приложения и взаимодействие с базой данных.  
**Бизнес-правило –** правило или ограничение, определяющее поведение системы и обеспечивающее соответствие бизнес-процессам студии.

**Интеграция –** взаимодействие между различными подсистемами или внешними системами для обмена данными и выполнения общих задач.

**Резервное копирование –** процесс создания копии данных системы для восстановления в случае сбоя или потери данных.

Элемент управления выбором – элементы пользовательского интерфейса, которые позволяют пользователю выбрать из группы допустимых объектов тот, с которым будет совершено действие.

Атрибут (Поле) – элемент данных, предназначенный для компьютерного описания какой-либо характеристики, соотношения или класса.

Окно – основное понятие оконного интерфейса, интерфейсный контейнер.

Ссылка - это объект, указывающий на определённые данные, но не хранящий их.

Персональные данные – любая информация, прямо или косвенно относящаяся к физическому лицу (субъекту персональных данных), и позволяющая его определить.

**2. АРХИТЕКТУРНОЕ ОБОСНОВАНИЕ**

Данные о системе, ее клиентах, ее сотрудниках будет храниться в реляционной базе данных на сервере PostgreSQL, и будет использоваться СУБД DBeaver, поскольку:

* И PostgreSQL и DBeaver поддерживаеют язык запросов SQL;
* PostgreSQL бесплатный и в нем легко разобраться;
* DBeaver также бесплатный сервис;
* DBeaver имеет приятный, понятный интерфейс;
* В PostgreSQL можно хранить данные и управлять ими в форматах JSON и XML;
* DBeaver поддерживает подключение расширений, позволяя настраивать программу для конкретных задач;
* DBeaver можно подключить к таким средам, как Eclipse, IntelliJ IDEA, Visual Studio Code;
* DBeaver поддерживает языки программирования;
* В PostgreSQL пользователи могут создавать функции, операторы, типы данных и индексные методы.

Fronted и Backend разработка будут осуществлены на языке программирования JavaScript.

Преимущества:

* Использование JavaScript как на frontend, так и на backend позволяет использовать один и тот же язык программирования для всей системы, что упрощает разработку и поддержку.
* Высокая производительность и масштабируемость благодаря асинхронной обработке запросов.
* Большое количество npm-пакетов (библиотек).

Недостатки:

* Может быть сложнее в отладке, чем некоторые другие языки.
* Требует хорошего знания асинхронного программирования.

1. **МОДЕЛИ НУЛЕВОГО УРОВНЯ**

Модели 0 уровня системы представлены ниже:

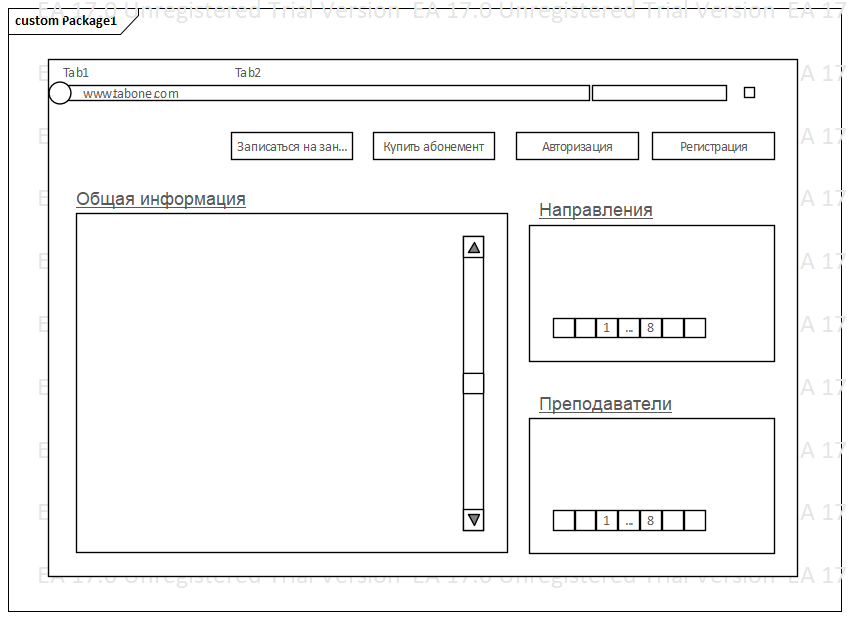


Рис. 3.1. Главная страница

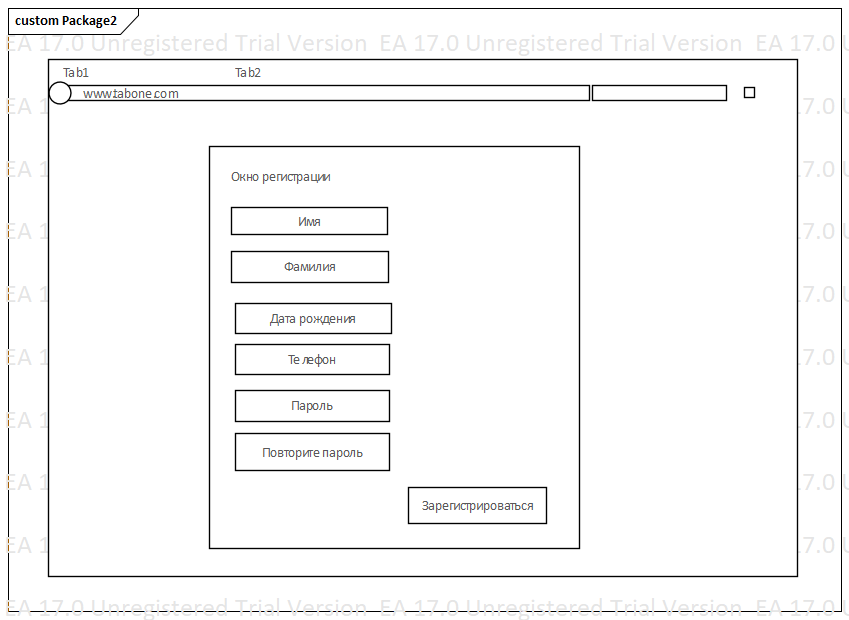


Рис. 3.2. Окно регистрации

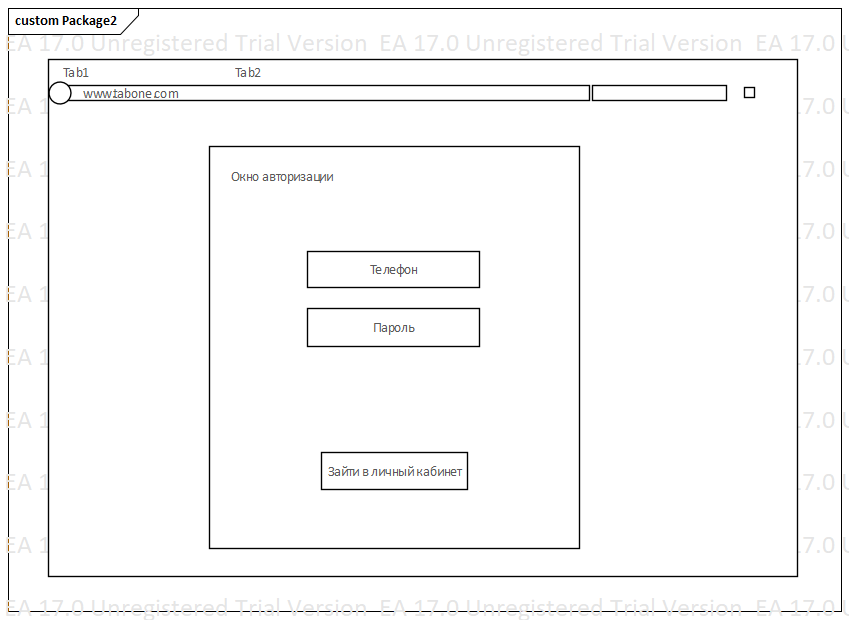


Рис. 3.3. Окно авторизации

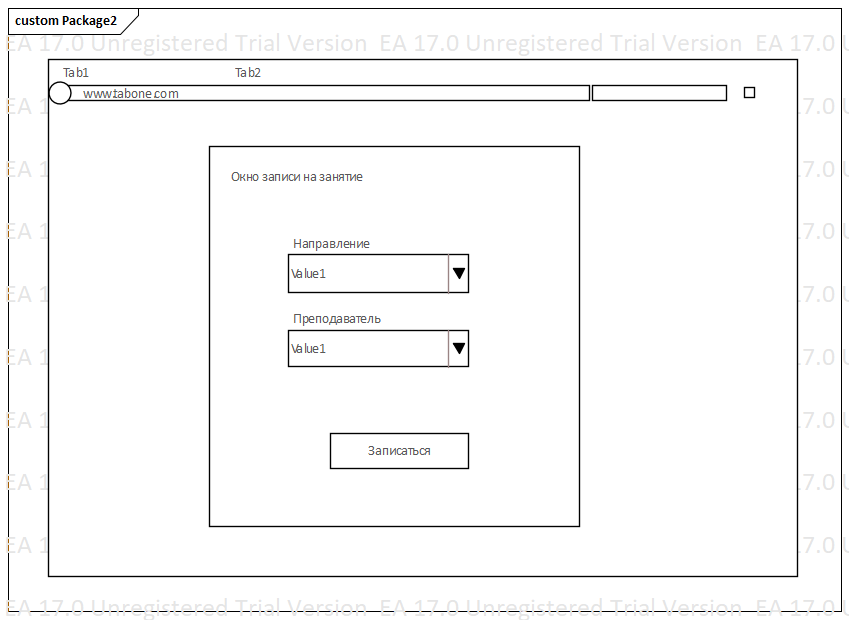


Рис. 3.4. Окно записи на занятие

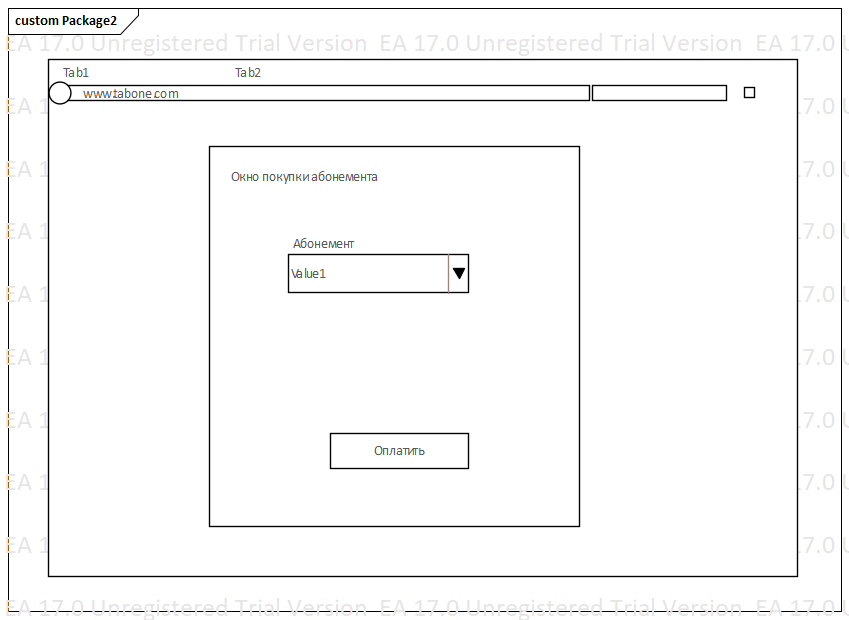


Рис. 3.5. Окно покупки абонемента

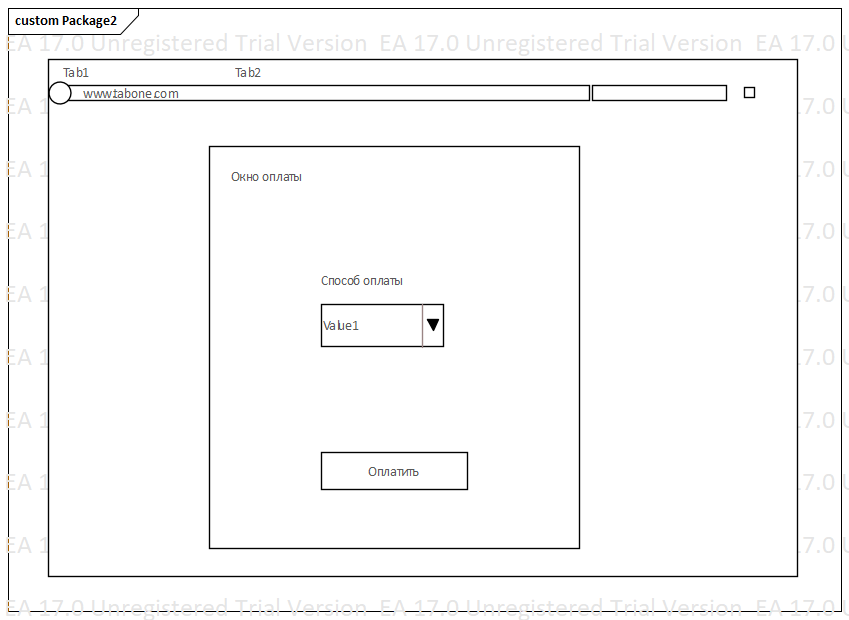


Рис. 3.6. Окно оплаты

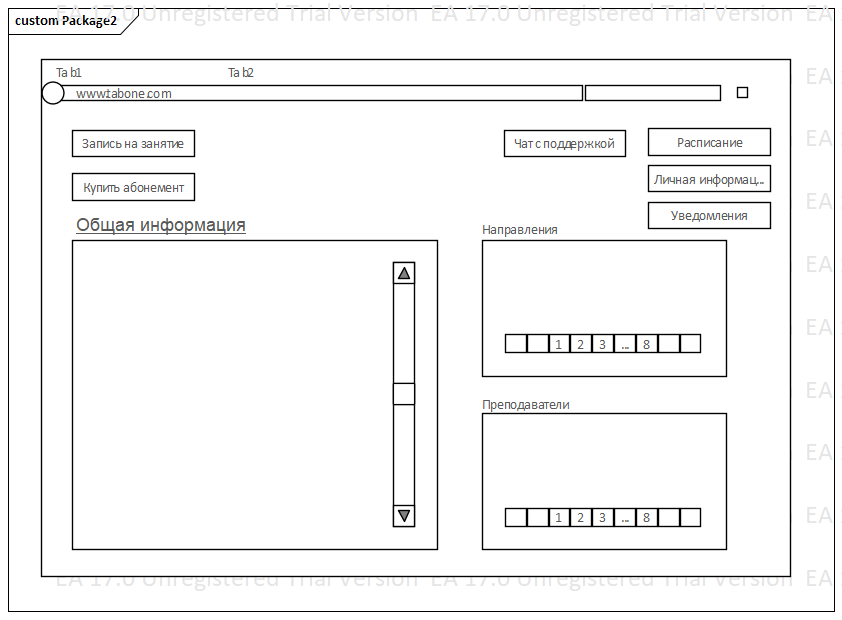


Рис. 3.7. Личный кабинет клиента

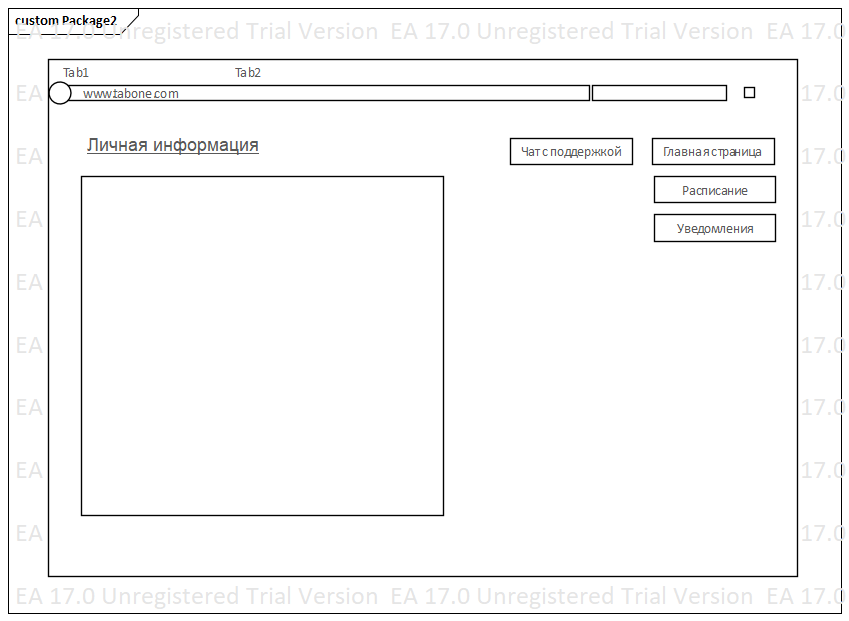


Рис. 3.8. Личный кабинет клиента

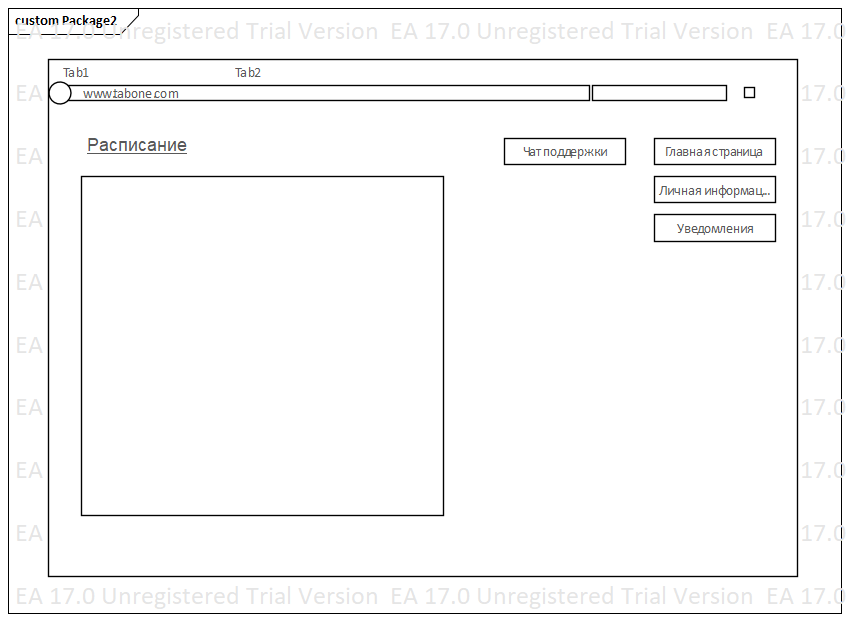


Рис. 3.9. Личный кабинет клиента

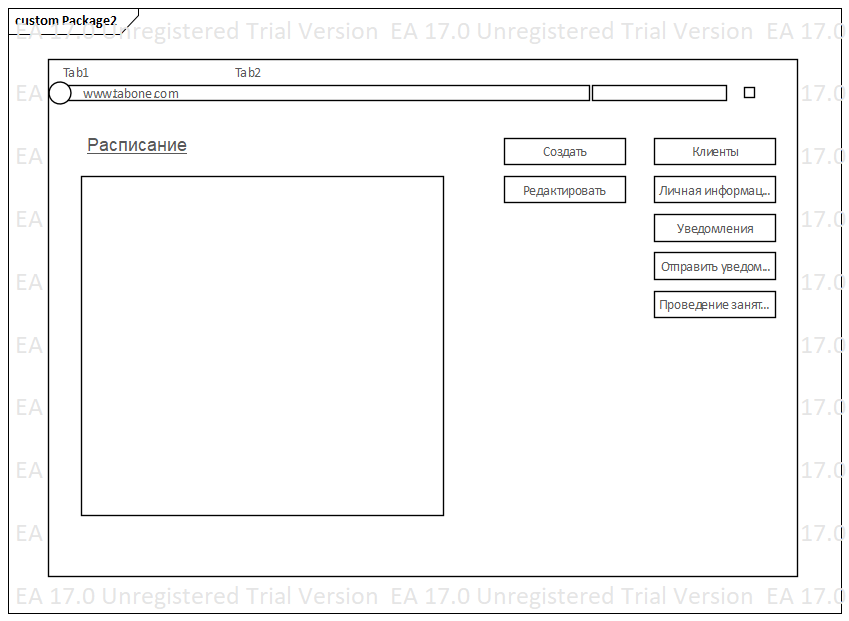


Рис.3.10. Личный кабинет преподавателя

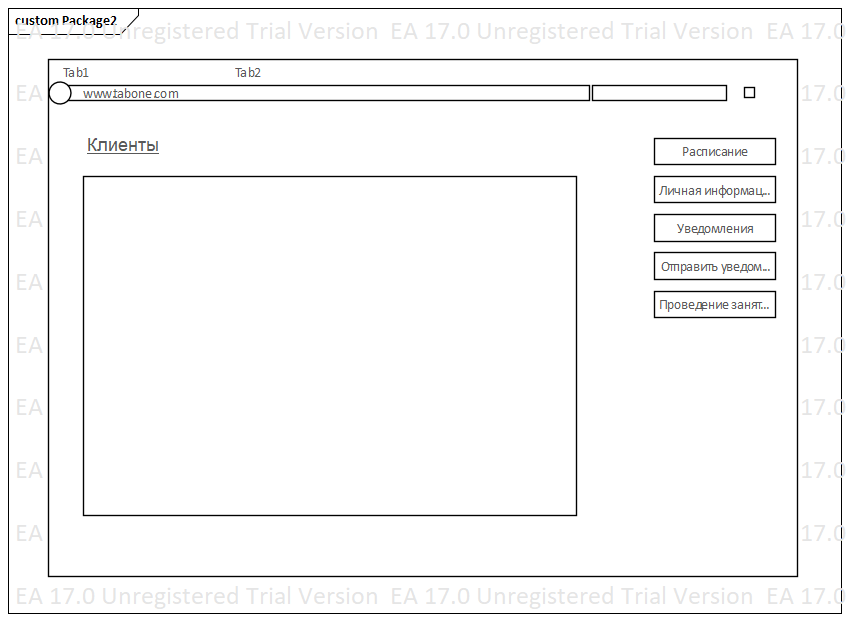


Рис.3.11. Личный кабинет преподавателя

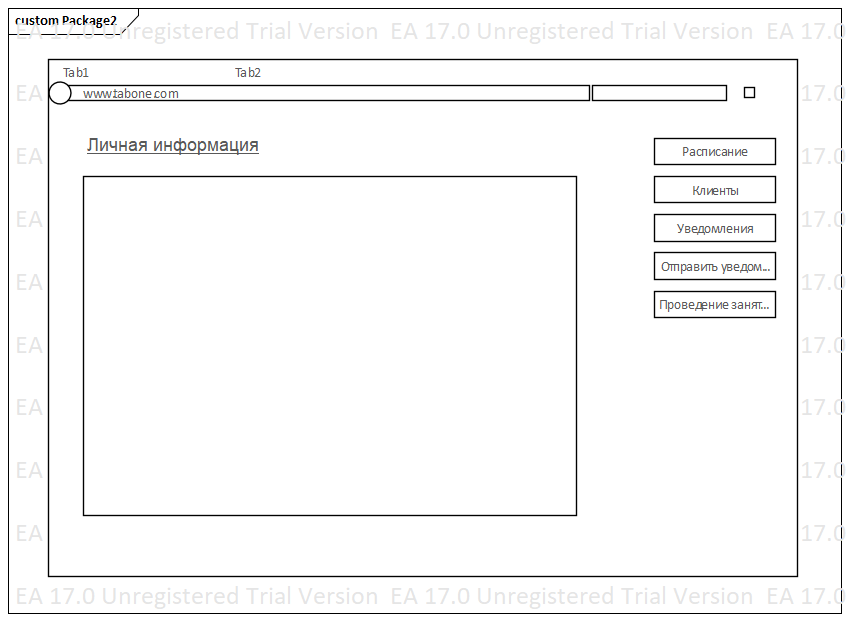


Рис.3.12. Личный кабинет преподавателя

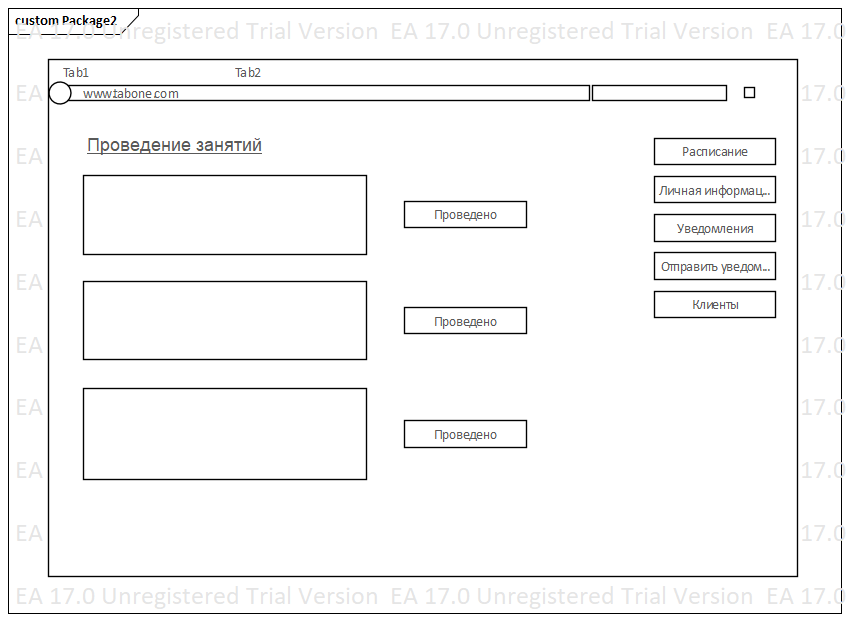


Рис.3.13. Личный кабинет преподавателя

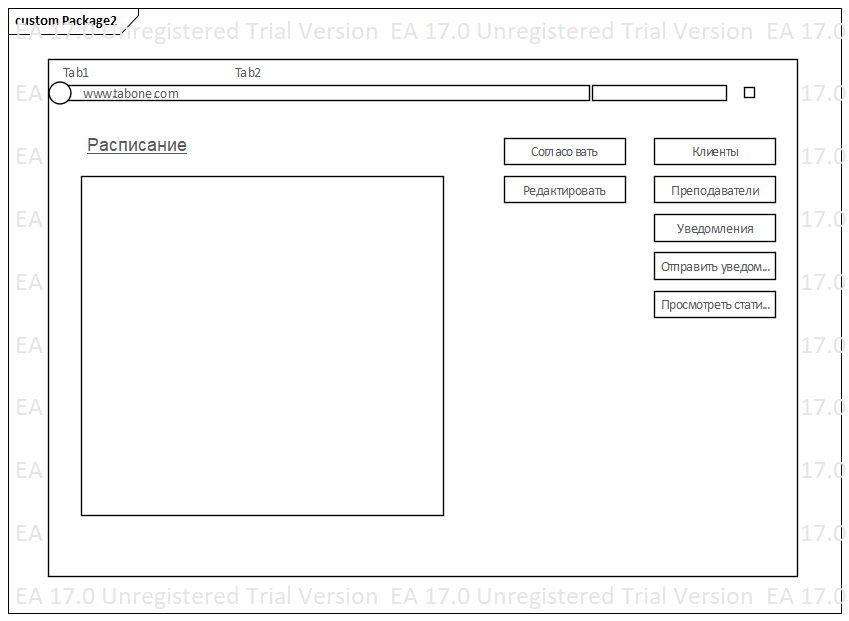


Рис. 3.14. Личный кабинет администратора

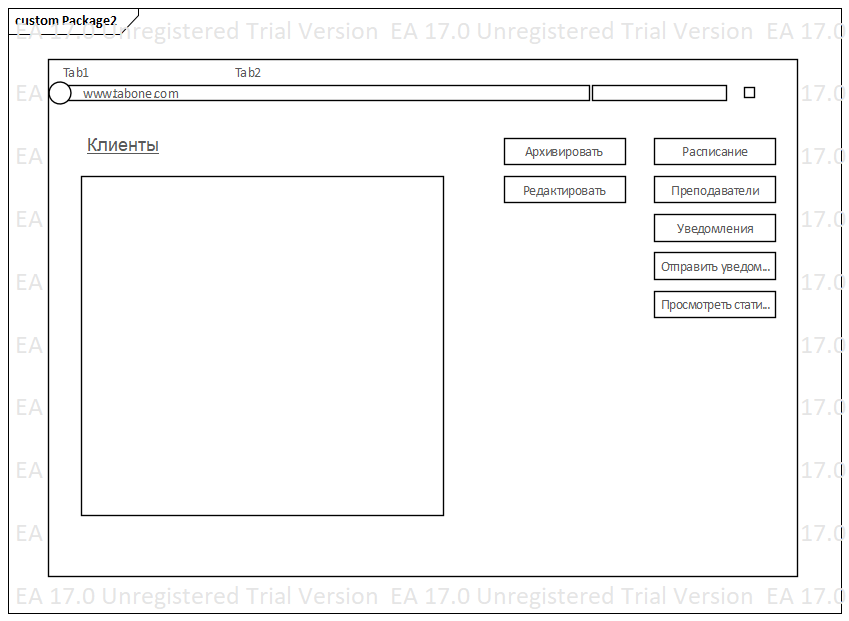


Рис. 3.15. Личный кабинет администратора

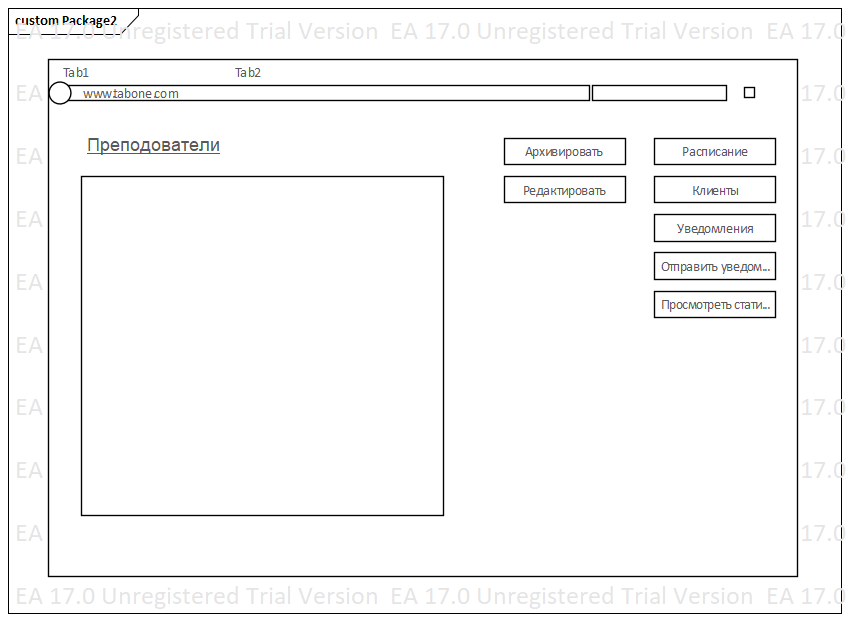


Рис. 3.16. Личный кабинет администратора

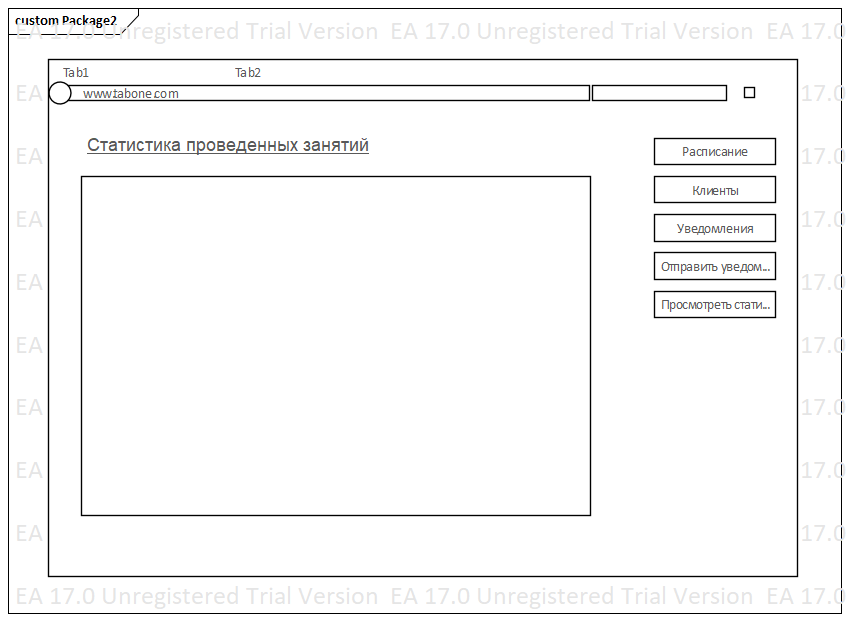


Рис. 3.17. Личный кабинет администратора

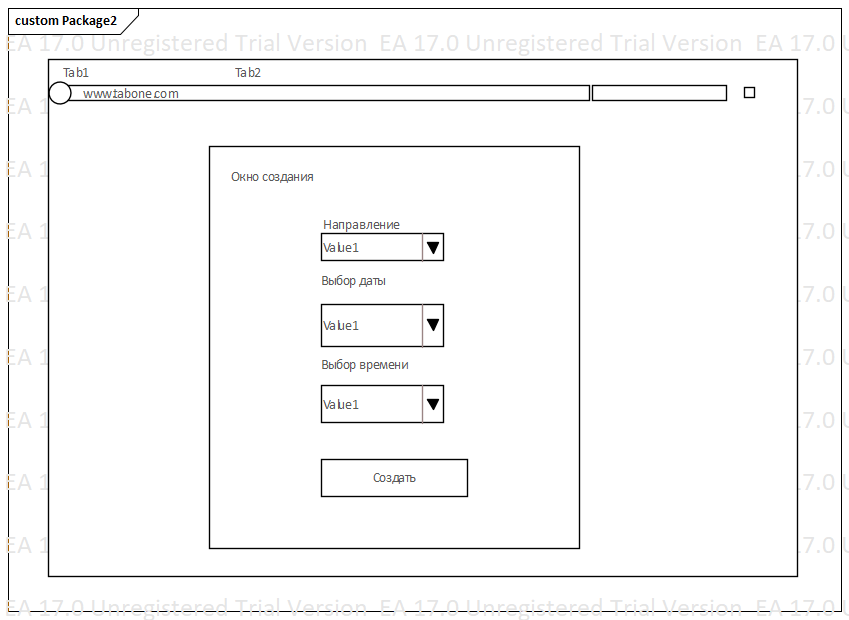


Рис.3.18. Окно создания расписания

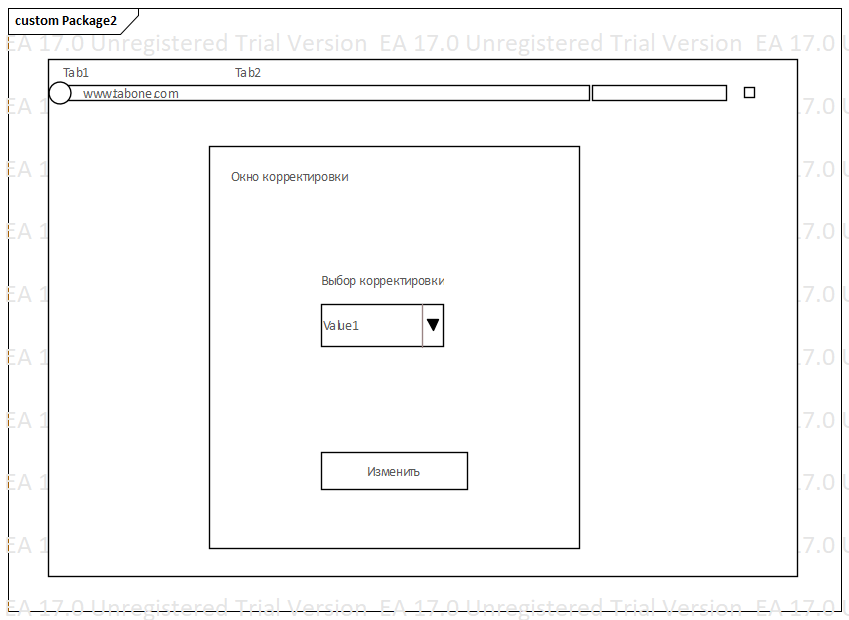


Рис.3.19. Окно корректировки расписания

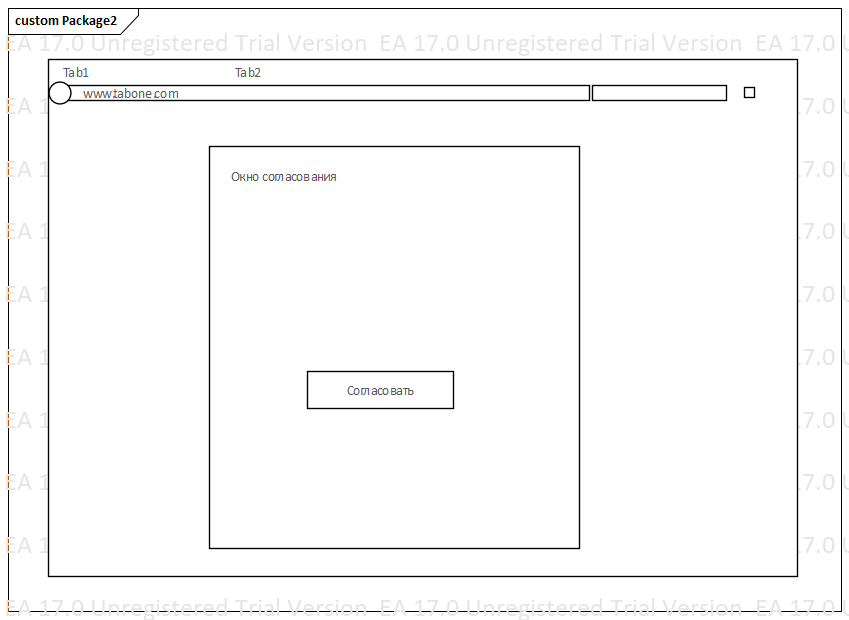


Рис.3.20. Окно согласования расписания

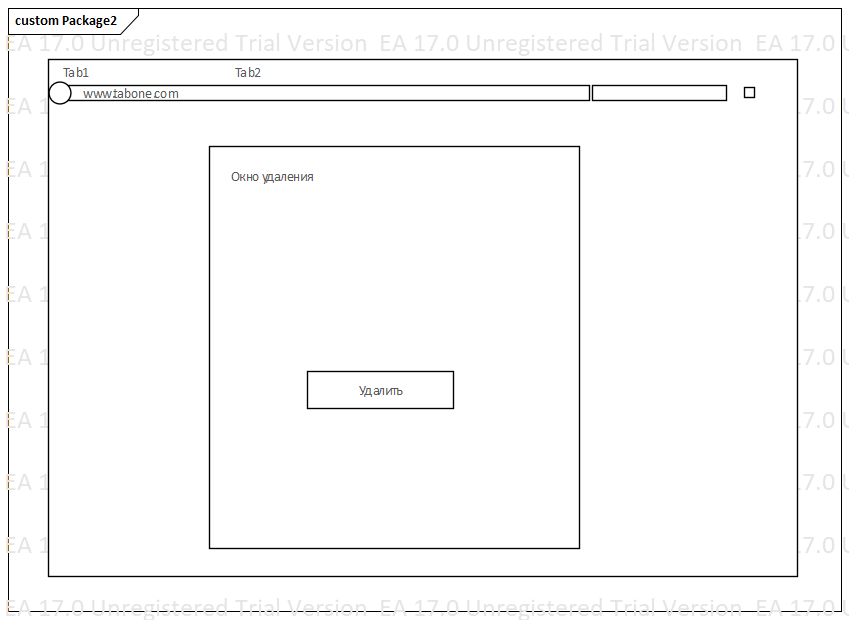


Рис.3.21. Окно удаления занятия

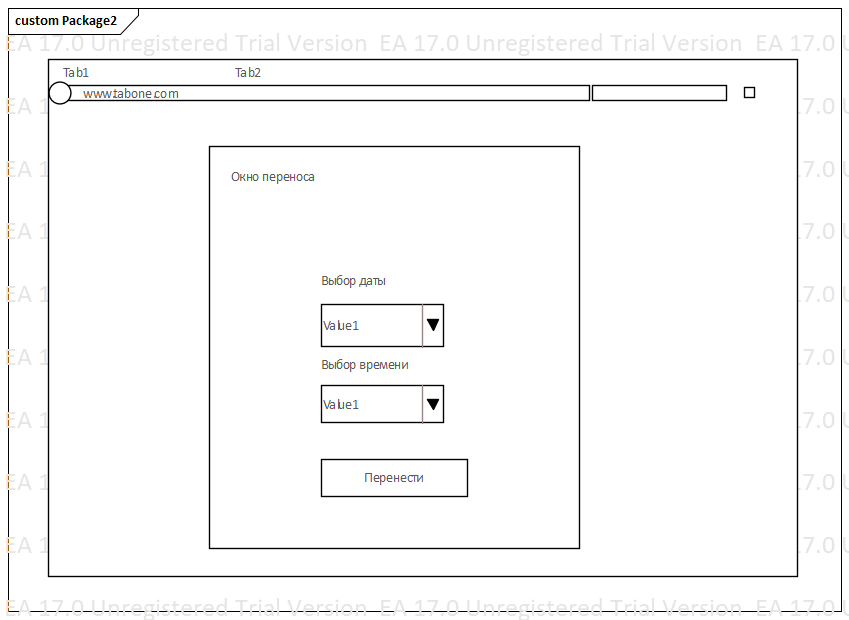


Рис.3.22. Окно переноса занятия

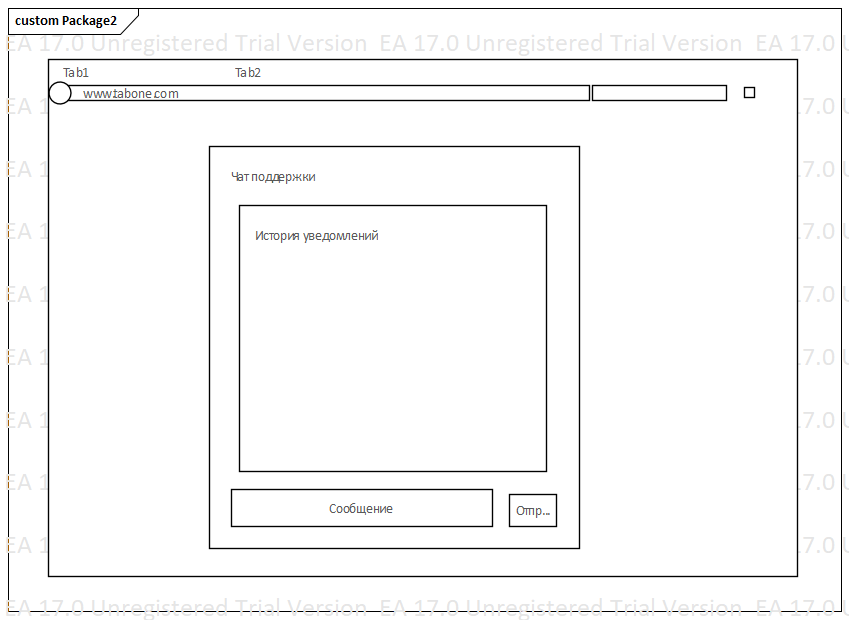


Рис. 3.23. Окно чата поддержки

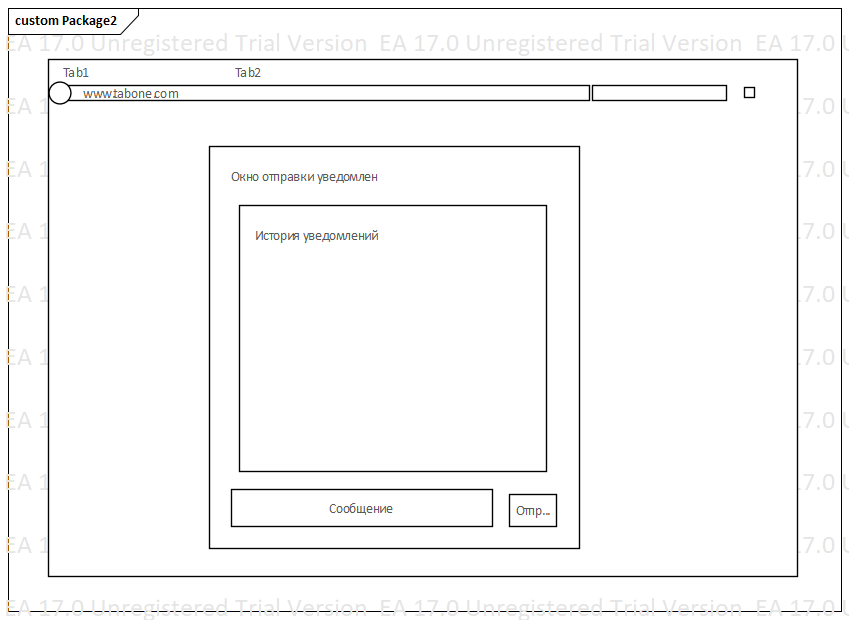


Рис. 3.24. Окно отправки уведомлений

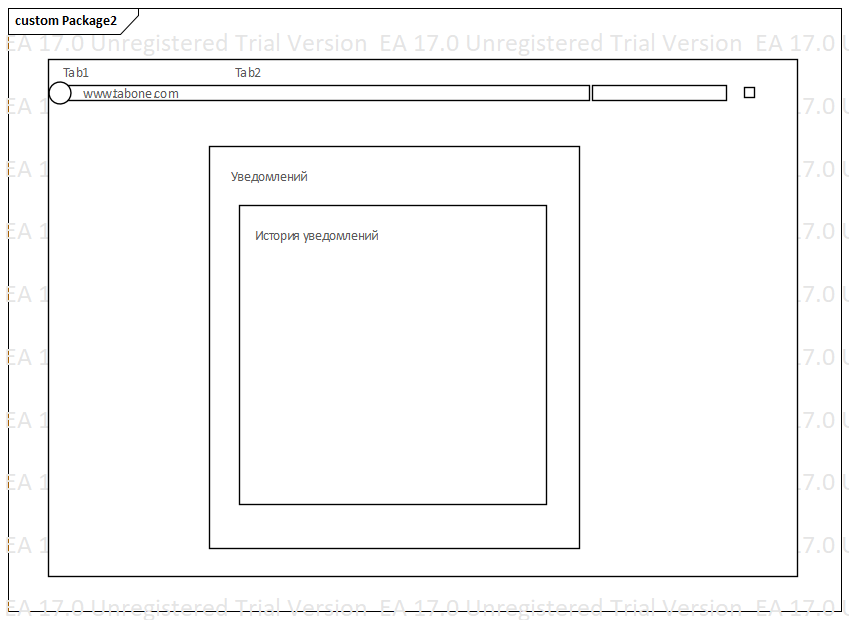


Рис. 3.25. Окно уведомлений

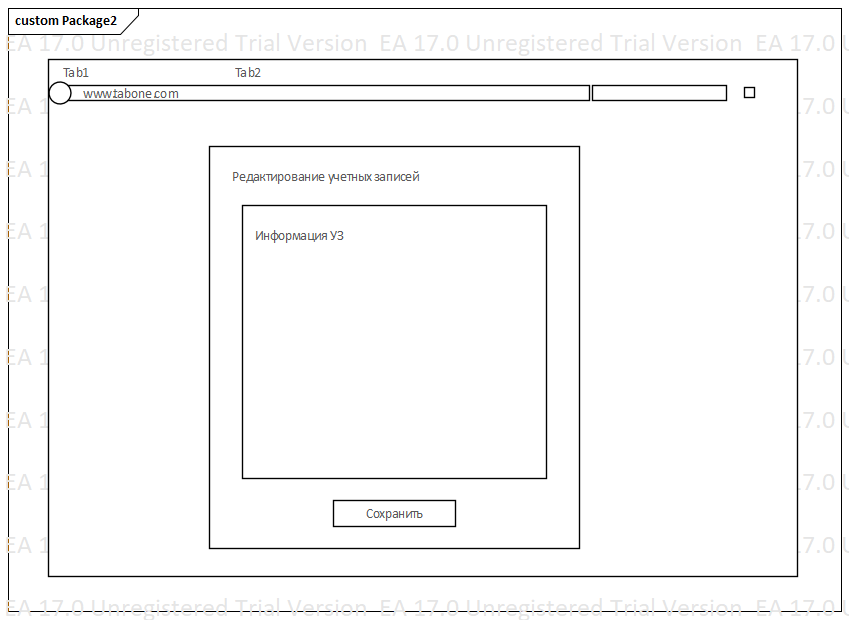


Рис. 3.26. Окно редактирования учетной записи

1. **АРХИТЕКТУРНОЕ ОПИСАНИЕ НА UML**

**4.1.USECASE ДИАГРАММА**

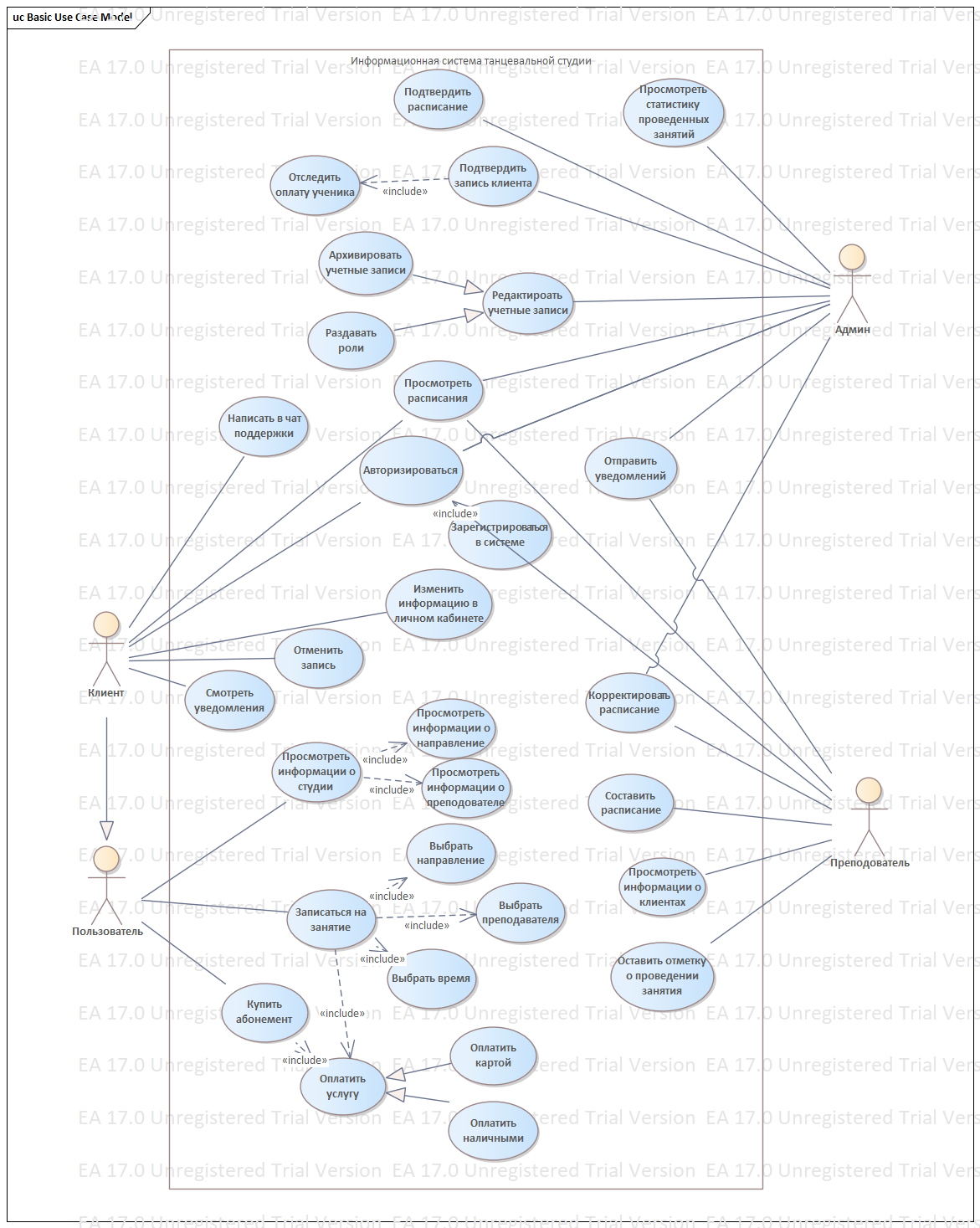


Рис. 4.1. Usecase диаграмма

Таблица 1.1. Описание case «Авторизоваться»:

|  |  |
| --- | --- |
| Вариант использования | Авторизоваться |
| Актеры | Клиент, Преподаватель, Администратор |
| Краткое описание | Пользователь заходит в свой личный кабинет |
| Цель | Идентифицировать пользователя |
| Ссылки на другие варианты использования | Include: «Авторизоваться» включает в себя «Зарегистрироваться» |

Таблица 1.2. Ход выполнения:

|  |  |
| --- | --- |
| Действия актера | Отклик системы |
| 1) Актер загружает исходную страницу. | 2) Система отображает страницу. |
| 3) Актер нажимает кнопку «Авторизация». | 4) Система открывает окно авторизации. |
| 5) Актер вводит свой логин и пароль. | 6) Система принимает логин и пароль. |
| 7) Актер нажимает кнопку «Зайти в личный кабинет». | 8) Система ищет пользователя по логину и проверяет пароль. Если такого логина нет, система просит пользователя зарегистрироваться. Если неверный пароль, система сообщает, что пароль введен некорректно. Если пользователь найден, то система отображает личный кабинет пользователя. |

Таблица 2.1. Описание case «Зарегистрироваться»:

|  |  |
| --- | --- |
| Вариант использования | Зарегистрироваться |
| Актеры | Клиент, Преподаватель, Администратор |
| Краткое описание | Создание учетной записи и личного кабинета пользователя |
| Цель | Идентифицировать пользователя |
| Ссылки на другие варианты использования | Include: «Авторизоваться» включает в себя «Зарегистрироваться» |

Таблица 2.2. Ход выполнения:

|  |  |
| --- | --- |
| Действия актера | Отклик системы |
| 1) Актер загружает исходную страницу. | 2) Система отображает страницу. |
| 3) Актер нажимает кнопку «Зарегистрироваться». | 4) Система открывает окно регистрации. |
| 5) Актер вводит свое имя, фамилию, дату рождения, телефон и пароль. | 6) Система проверяет имя, фамилию, дату рождения, телефон на корректность. |
| 7) Актер нажимает кнопку «Зайти в личный кабинет». | 8) Если все данные введены корректно, система добавляет новую учетную запись пользователя, отправляет уведомление администратору и открывает окно авторизации. Если данные некорректны, система сообщает, что данные введены некорректно. |

Таблица 3.1. Описание case «Просмотреть расписание»:

|  |  |
| --- | --- |
| Вариант использования | Просмотреть расписание |
| Актеры | Клиент, Преподаватель, Администратор |
| Краткое описание | Открывается страница с расписанием на неделю |
| Цель | Узнать составленное расписание |
| Ссылки на другие варианты использования | Ссылки отсутствуют |

Таблица 3.2. Ход выполнения:

|  |  |
| --- | --- |
| Действия актера | Отклик системы |
| 1) Актер открывает личный кабинет | 2) Система отображает личный кабинет актера |
| 3) Актер нажимает кнопку «Расписание» | 3) Система отображает страницу с расписанием на неделю |

Таблица 4.1. Описание case «Создать расписание»:

|  |  |
| --- | --- |
| Вариант использования | Корректировать расписание |
| Актеры | Преподаватель |
| Краткое описание | Добавление занятий в определенное время |
| Цель | Создание расписания |
| Ссылки на другие варианты использования | Ссылки отсутствуют |

Таблица 4.2. Ход выполнения:

|  |  |
| --- | --- |
| Действия актера | Отклик системы |
| 1) Преподаватель открывает личный кабинет | 2) Система отображает личный кабинет преподавателя |
| 3) Преподаватель нажимает кнопку «Расписание» | 4) Система открывает страницу с расписанием |
| 5) Преподаватель нажимает кнопку «Создать» | 6) Система открывает окно создания занятия |
| 7) Преподаватель выбирает направление и время | 8) Система отображает выбор |
| 9) Преподаватель нажимает кнопку «Создать занятие» | 10) Система записывает занятие, направление, время, преподавателя и отправляет уведомление администратору для подтверждения. |

Таблица 5.1. Описание case «Корректировать расписание»:

|  |  |
| --- | --- |
| Вариант использования | Корректировать расписание |
| Актеры | Преподаватель, Администратор |
| Краткое описание | Отмена или добавление занятия в расписание |
| Цель | Внесение изменений в расписание |
| Ссылки на другие варианты использования | Ссылки отсутствуют |

Таблица 5.2. Ход выполнения:

|  |  |
| --- | --- |
| Действия актера | Отклик системы |
| 1) Актер открывает личный кабинет | 2) Система отображает личный кабинет |
| 3) Актер нажимает кнопку «Расписание» | 4) Система открывает страницу с расписанием |
| 5) Актер выбирает занятие | 6) Система выделяет занятие |
| 7) Актер нажимает кнопку «Изменить» | 8) Система открывает окно корректирования |
| 9) Препод выбирает отменить занятие | 10) Система архивирует занятие и отправляет уведомление администратору для подтверждения |
| 11) Админ выбирает отменить занятие | 12) Система архивирует занятие |
| 13) Препод выбирает перенести занятие | 14) Система отображает дополнительный выбор времени переноса |
| 15) Препод выбирает новое время и нажимает кнопку сохранить | 16) Система меняет время занятия и отправляет админу уведомление для подтверждения |
| 17) Админ выбирает перенести занятие | 18) Система отображает дополнительный выбор времени переноса и зала |
| 19) Админ выбирает новое время и зал и нажимает кнопку сохранить | 20) Система меняет время занятия и зал |

Таблица 6.1. Описание case «Подтвердить расписание»:

|  |  |
| --- | --- |
| Вариант использования | Подтвердить расписание |
| Актеры | Администратор |
| Краткое описание | Проверка и подтверждение созданного расписания |
| Цель | Публикация готового расписания |
| Ссылки на другие варианты использования | Ссылки отсутствуют |

Таблица 6.2. Ход выполнения:

|  |  |
| --- | --- |
| Действия актера | Отклик системы |
| 1) Админ открывает личный кабинет | 2) Система отображает страницу личного кабинета |
| 3) Админ нажимает кнопку «Расписание» | 4) Система отображает страницу расписания |
| 5) Админ выбирает занятие | 6) Система выделяет выбранное занятие |
| 7) Админ добавляет зал | 8) Система записывает зал, в котором будет проходить занятие |
| 9) Админ проверяет расписание и нажимает кнопку «Согласовать» | 10) Система публикует расписание |

Таблица 7.1. Описание case «Подтвердить запись клиента»:

|  |  |
| --- | --- |
| Вариант использования | Подтвердить запись клиента |
| Актеры | Администратор |
| Краткое описание | Админ утверждает запись клиента на занятие |
| Цель | Контроль количества клиентов на занятии |
| Ссылки на другие варианты использования | Include: «Подтвердить запись клиента» включает в себя «Отследить оплату ученика» |

Таблица 7.2. Ход выполнения:

|  |  |
| --- | --- |
| Действия актера | Отклик системы |
| 1) Админ заходит в уведомления | 2) Система показывает окно уведомлений |
| 3) Админ видит новую запись и переходит по ней | 4) Система показывает окно записи |
| 5) Админ подтверждает запись | 6) Система добавляет запись в БД |

Таблица 8.1. Описание case «Отследить оплату ученика»:

|  |  |
| --- | --- |
| Вариант использования | Отследить оплату ученика |
| Актеры | Администратор |
| Краткое описание | Админ понимает как клиент будет платить за занятие |
| Цель | Подготовка к способу оплаты |
| Ссылки на другие варианты использования | Include: «Подтвердить запись клиента» включает в себя «Отследить оплату ученика» |

Таблица 8.2. Ход выполнения:

|  |  |
| --- | --- |
| Действия актера | Отклик системы |
| 1) Админ заходит в уведомления | 2) Система показывает окно уведомлений |
| 3) Админ видит новую запись и переходит по ней | 4) Система показывает окно записи |

Таблица 9.1. Описание case «Редактировать учетные записи»:

|  |  |
| --- | --- |
| Вариант использования | Редактировать учетные записи |
| Актеры | Администратор |
| Краткое описание | Возможность изменения личной информации клиентов и сотрудников |
| Цель | Обновление данных |
| Ссылки на другие варианты использования | Generalize: «Раздавать роли» частный случай «Редактировать учетные записи»  Generalize: «Архивировать учетные записи» частный случай «Редактировать учетные записи» |

Таблица 9.2. Ход выполнения:

|  |  |
| --- | --- |
| Действия актера | Отклик системы |
| 1)Админ загружает страницу «Клиенты» или «Преподаватели» | 2) Система отображает страницу |
| 3) Админ нажимает кнопку «Редактировать» | 4) Система открывает окно редакции |
| 5) Админ изменяет данные | 6) Система записывает новые данные |

Таблица 10.1. Описание case «Раздавать роли»:

|  |  |
| --- | --- |
| Вариант использования | Раздавать роли |
| Актеры | Администратор |
| Краткое описание | Администратор присваивает роли |
| Цель | Дать права доступа |
| Ссылки на другие варианты использования | Generalize: «Раздавать роли» частный случай «Редактировать учетные записи» |

Таблица 10.2. Ход выполнения:

|  |  |
| --- | --- |
| Действия актера | Отклик системы |
| 1)Админ загружает страницу «Клиенты» или «Преподаватели» | 2) Система отображает страницу |
| 3) Админ нажимает кнопку «Редактировать» | 4) Система открывает окно редакции |
| 5) Админ изменяет роль | 6) Система записывает новую роль |

Таблица 11.1. Описание case «Архивировать учетные записи»:

|  |  |
| --- | --- |
| Вариант использования | Архивировать учетные записи |
| Актеры | Администратор |
| Краткое описание | Администратор архивирует данные |
| Цель | Убрать неактивные учетные записи |
| Ссылки на другие варианты использования | Generalize: «Архивировать учетные записи» частный случай «Редактировать учетные записи» |

Таблица 11.2. Ход выполнения:

|  |  |
| --- | --- |
| Действия актера | Отклик системы |
| 1)Админ загружает страницу «Клиенты» или «Преподаватели» | 2) Система отображает страницу |
| 3) Админ нажимает кнопку «Редактировать» | 4) Система открывает окно редакции |
| 5) Админ нажимает кнопку «Архивировать» | 6) Система отправляет данные в архив и удаляет их |

Таблица 12.1. Описание case «Просмотреть статистику проведенных занятий»:

|  |  |
| --- | --- |
| Вариант использования | Просмотреть статистику проведенных занятий |
| Актеры | Администратор |
| Краткое описание | Администратор смотрит, проведены ли занятия |
| Цель | Узнать статистику проведения занятий |
| Ссылки на другие варианты использования | Ссылки отсутствуют |

Таблица 12.2. Ход выполнения:

|  |  |
| --- | --- |
| Действия актера | Отклик системы |
| 1) Админ открывает личный кабинет | 2) Система отображает личный кабинет |
| 3) Админ загружает страницу «Статистика» | 4) Система отображает страницу |

Таблица 13.1. Описание case «Отправить уведомление»:

|  |  |
| --- | --- |
| Вариант использования | Отправить уведомление |
| Актеры | Преподаватель, Администратор |
| Краткое описание | Отправить необходимую информацию клиентам |
| Цель | Извещение клиентов |
| Ссылки на другие варианты использования | Ссылки отсутствуют |

Таблица 13.2. Ход выполнения:

|  |  |
| --- | --- |
| Действия актера | Отклик системы |
| 1) Актер открывает личный кабинет | 2) Система отображает личный кабинет |
| 3) Актер загружает страницу «Отправить уведомления» | 4) Система отображает страницу |
| 5) Актер набирает необходимое сообщение | 6) Система записывает сообщение |
| 7) Актер нажимает «Отправить» | 8) Система отправляет уведомления о новом сообщении клиентам |

Таблица 14.1. Описание case «Оставить отметку о проведении занятия»:

|  |  |
| --- | --- |
| Вариант использования | Оставить отметку о проведении занятия |
| Актеры | Преподаватель |
| Краткое описание | Преподаватель оставляет отметку, что провел занятие |
| Цель | Узнать статистику проведения занятий |
| Ссылки на другие варианты использования | Ссылки отсутствуют |

Таблица 14.2. Ход выполнения:

|  |  |
| --- | --- |
| Действия актера | Отклик системы |
| 1) Преподаватель открывает личный кабинет | 2) Система отображает личный кабинет |
| 3) Преподаватель загружает страницу «Отметки» | 4) Система отображает страницу |
| 5) Преподаватель нажимает кнопку «Провел» у нужного занятия | 6) Система записывает, что занятие проведено |

Таблица 15.1. Описание case «Просмотреть информацию о клиентах»:

|  |  |
| --- | --- |
| Вариант использования | Просмотреть информацию о клиентах |
| Актеры | Преподаватель |
| Краткое описание | Просмотреть информацию о клиентах, узнать уровень их подготовки и возраст |
| Цель | Узнать целевую аудиторию |
| Ссылки на другие варианты использования | Ссылки отсутствуют |

Таблица 15.2. Ход выполнения:

|  |  |
| --- | --- |
| Действия актера | Отклик системы |
| 1) Преподаватель открывает личный кабинет | 2) Система отображает личный кабинет |
| 3) Преподаватель загружает страницу «Клиенты» | 4) Система отображает страницу |

Таблица 16.1. Описание case «Смотреть уведомления»:

|  |  |
| --- | --- |
| Вариант использования | Смотреть уведомления |
| Актеры | Клиент |
| Краткое описание | Клиент просматривает пришедшие ему уведомления |
| Цель | Извещение клиентов |
| Ссылки на другие варианты использования | Ссылки отсутствуют |

Таблица 16.2. Ход выполнения:

|  |  |
| --- | --- |
| Действия актера | Отклик системы |
| 1) Клиент открывает личный кабинет | 2) Система отображает личный кабинет |
| 3) Клиент загружает страницу «Уведомления» | 4) Система отображает страницу |

Таблица 17.1. Описание case «Изменить информацию в личном кабинете»:

|  |  |
| --- | --- |
| Вариант использования | Изменить информацию в личном кабинете |
| Актеры | Клиент |
| Краткое описание | Клиент меняет свои личные данные |
| Цель | Обновление данных |
| Ссылки на другие варианты использования | Ссылки отсутствуют |

Таблица 17.2. Ход выполнения:

|  |  |
| --- | --- |
| Действия актера | Отклик системы |
| 1) Клиент открывает личный кабинет | 2) Система отображает личный кабинет |
| 3) Клиент загружает страницу «Личная информация» | 4) Система отображает страницу |
| 5) Клиент нажимает «Редактировать» | 6) Система открывает окно редакции |
| 7) Клиент меняет данные и нажимает «Сохранить» | 8) Система сохраняет данные и отправляет уведомление админу |

Таблица 18.1. Описание case «Просмотреть информацию о студии»:

|  |  |
| --- | --- |
| Вариант использования | Просмотреть информацию о студии |
| Актеры | Клиент |
| Краткое описание | Клиент читает информацию о студии |
| Цель | Информирование клиента |
| Ссылки на другие варианты использования | Include: «Просмотреть информацию о студии» включает в себя «Просмотреть информацию о направлении»  Include: «Просмотреть информацию о студии» включает в себя «Просмотреть информацию о преподавателе» |

Таблица 18.2. Ход выполнения:

|  |  |
| --- | --- |
| Действия актера | Отклик системы |
| 1) Клиент загружает главную страницу | 2) Система отображает страницу |

Таблица 19.1. Описание case «Просмотреть информацию о направлении»:

|  |  |
| --- | --- |
| Вариант использования | Просмотреть информацию о направлении |
| Актеры | Клиент |
| Краткое описание | Клиент читает информацию о направлениях |
| Цель | Информирование клиента |
| Ссылки на другие варианты использования | Include: «Просмотреть информацию о студии» включает в себя «Просмотреть информацию о направлении» |

Таблица 19.2. Ход выполнения:

|  |  |
| --- | --- |
| Действия актера | Отклик системы |
| 1) Клиент загружает главную страницу | 2) Система отображает страницу |

Таблица 20.1. Описание case «Просмотреть информацию о преподавателе»:

|  |  |
| --- | --- |
| Вариант использования | Просмотреть информацию о преподавателе |
| Актеры | Клиент |
| Краткое описание | Клиент читает информацию о преподавателях |
| Цель | Информирование клиента |
| Ссылки на другие варианты использования | Include: «Просмотреть информацию о студии» включает в себя «Просмотреть информацию о преподавателе» |

Таблица 20.2. Ход выполнения:

|  |  |
| --- | --- |
| Действия актера | Отклик системы |
| 1) Клиент загружает главную страницу | 2) Система отображает страницу |

Таблица 21.1. Описание case «Записаться на занятие»:

|  |  |
| --- | --- |
| Вариант использования | Записаться на занятие |
| Актеры | Клиент |
| Краткое описание | Клиент записывается на занятие |
| Цель | Принять клиента на занятие |
| Ссылки на другие варианты использования | Include: «Записаться на занятие» включает в себя «Выбрать время»  Include: «Записаться на занятие» включает в себя «Выбрать направление»  Include: «Записаться на занятие» включает в себя «Выбрать преподавателя»  Include: «Записаться на занятие» включает в себя «Оплатить услугу» |

Таблица 21.2. Ход выполнения:

|  |  |
| --- | --- |
| Действия актера | Отклик системы |
| 1) Клиент загружает главную страницу | 2) Система отображает страницу |
| 3) Клиент нажимает «Записаться на занятие» | 4) Система открывает окно записи |
| 5) Клиент выбирает время, направление, преподавателя и нажимает «Записаться» | 6) Система записывает клиента, открывает окно оплаты и отправляет уведомление администратору |

Таблица 22.1. Описание case «Выбрать время»:

|  |  |
| --- | --- |
| Вариант использования | Выбрать время |
| Актеры | Клиент |
| Краткое описание | Клиент выбирает время занятия |
| Цель | Принять клиента на занятие |
| Ссылки на другие варианты использования | Include: «Записаться на занятие» включает в себя «Выбрать время» |

Таблица 22.2. Ход выполнения:

|  |  |
| --- | --- |
| Действия актера | Отклик системы |
| 1)Клиент загружает окно записи | 2) Система отображает окно |
| 3) Клиент выбирает время из списка | 4) Система записывает время |

Таблица 23.1. Описание case «Выбрать направление»:

|  |  |
| --- | --- |
| Вариант использования | Выбрать направление |
| Актеры | Клиент |
| Краткое описание | Клиент выбирает направление танцев |
| Цель | Принять клиента на занятие |
| Ссылки на другие варианты использования | Include: «Записаться на занятие» включает в себя «Выбрать направление» |

Таблица 23.2. Ход выполнения:

|  |  |
| --- | --- |
| Действия актера | Отклик системы |
| 1)Клиент загружает окно записи | 2) Система отображает окно |
| 3) Клиент выбирает направление из списка | 4) Система записывает направление |

Таблица 24.1. Описание case «Выбрать преподавателя»:

|  |  |
| --- | --- |
| Вариант использования | Выбрать преподавателя |
| Актеры | Клиент |
| Краткое описание | Клиент выбирает пеподавателя |
| Цель | Принять клиента на занятие |
| Ссылки на другие варианты использования | Include: «Записаться на занятие» включает в себя «Выбрать преподавателя» |

Таблица 24.2. Ход выполнения:

|  |  |
| --- | --- |
| Действия актера | Отклик системы |
| 1)Клиент загружает окно записи | 2) Система отображает окно |
| 3) Клиент выбирает преподавателя из списка | 4) Система записывает преподавателя |

Таблица 25.1. Описание case «Купить абонемент»:

|  |  |
| --- | --- |
| Вариант использования | Купить абонемент |
| Актеры | Клиент |
| Краткое описание | Клиент приобретает абонемент |
| Цель | Принять клиента |
| Ссылки на другие варианты использования | Include: «Купить абонемент» включает в себя «Оплатить услугу» |

Таблица 25.2. Ход выполнения:

|  |  |
| --- | --- |
| Действия актера | Отклик системы |
| 1) Клиент загружает главную страницу | 2) Система отображает страницу |
| 3) Клиент нажимает «Купить абонемент» | 4) Система открывает окно покупки абонемента |
| 5) Клиент выбирает вид абонемента и нажимает «Оплатить» | 6) Система записывает данные, открывает окно оплаты и отправляет уведомление админу |

Таблица 26.1. Описание case «Оплатить услугу»:

|  |  |
| --- | --- |
| Вариант использования | Оплатить услугу |
| Актеры | Клиент |
| Краткое описание | Клиент производит оплату |
| Цель | Получить оплату |
| Ссылки на другие варианты использования | Generalize: «Оплатить картой» частный случай «Оплатить услугу»  Generalize: «Оплатить наличными» частный случай «оплатить» |

Таблица 26.2. Ход выполнения:

|  |  |
| --- | --- |
| Действия актера | Отклик системы |
| 1)Клиент нажимает оплатить | 2) Система отображает окно оплаты |
| 3) Клиент выбирает способ оплаты из списка | 4) Система записывает способ оплаты |

Таблица 27.1. Описание case «Оплатить наличными»:

|  |  |
| --- | --- |
| Вариант использования | Оплатить наличными |
| Актеры | Клиент |
| Краткое описание | Клиент производит оплату |
| Цель | Получить оплату |
| Ссылки на другие варианты использования | Generalize: «Оплатить наличными» частный случай «оплатить» |

Таблица 27.2. Ход выполнения:

|  |  |
| --- | --- |
| Действия актера | Отклик системы |
| 1)Клиент нажимает оплатить | 2) Система отображает окно оплаты |
| 3) Клиент выбирает оплатить наличными | 4) Система записывает способ оплаты |

Таблица 28.1. Описание case «Оплатить картой»:

|  |  |
| --- | --- |
| Вариант использования | Оплатить картой |
| Актеры | Клиент |
| Краткое описание | Клиент производит оплату |
| Цель | Получить оплату |
| Ссылки на другие варианты использования | Generalize: «Оплатить картой» частный случай «Оплатить услугу» |

Таблица 28.2. Ход выполнения:

|  |  |
| --- | --- |
| Действия актера | Отклик системы |
| 1)Клиент нажимает оплатить | 2) Система отображает окно оплаты |
| 3) Клиент выбирает оплатить картой | 4) Система записывает способ оплаты |

Таблица 29.1. Описание case «Написать в чат поддержки»:

|  |  |
| --- | --- |
| Вариант использования | Написать в чат поддержки |
| Актеры | Клиент |
| Краткое описание | Клиент отправляет уведомление сотрудникам |
| Цель | Извещение сотрудников |
| Ссылки на другие варианты использования | Ссылки отсутствуют |

Таблица 29.2. Ход выполнения:

|  |  |
| --- | --- |
| Действия актера | Отклик системы |
| 1) Клиент открывает личный кабинет | 2) Система отображает личный кабинет |
| 3) Клиент загружает страницу «Чат поддержки» | 4) Система отображает страницу |
| 5) Клиент набирает необходимое сообщение | 6) Система записывает сообщение |
| 7) Клиент нажимает «Отправить» | 8) Система отправляет уведомления о новом сообщении сотрудникам |

**4.2. ДИАГРАММА КЛАССОВ**

Классы системы и их свойства были определены в разделе «Системные модели», а конкретно: Абонемент, Тип абонемента, Оплата, Клиент, Занятие, Зал, Преподаватель.

Связи между объектами:

* Абонемент имеет Тип абонемента (Абонемент может иметь только один тип абонемента, но тип абонемента может принадлежать нескольким абонементам или не принадлежать никому)
* Клиент покупает Абонемент (Клиент может купить сколько угодно абонементов, абонемент должен принадлежать одному клиенту)
* Клиент производит Оплату (Клиент может произвести сколько угодно оплат, но оплата обязательно принадлежит только одному клиенту)
* Клиент записывается на Занятие (Клиент может записаться на сколько угодно занятий от одного, на занятие могут записаться от 1 до 30 человек, поскольку 30 – максимальная вместимость залов)
* 1 Занятие проводится в 1 Зале
* Занятие проводит Преподаватель (Занятие проводит только 1 преподаватель, но Преподаватель может вести сколько угодно Занятий)

Методы, которые используют классы:

* Класс Абонемент использует метод «узнать тип абонемента», который производится по id типа абонемента
* Класс Клиент использует методы: «узнать дату рождения», «узнать медицинские показания», «узнать уровень»
* Класс Оплата использует методы: «узнать сумму», «узнать дату»
* Класс Преподаватель использует методы: «узнать телефон», «узнать ставку»
* Класс Занятие использует метод «узнать преподавателя» по id преподователя

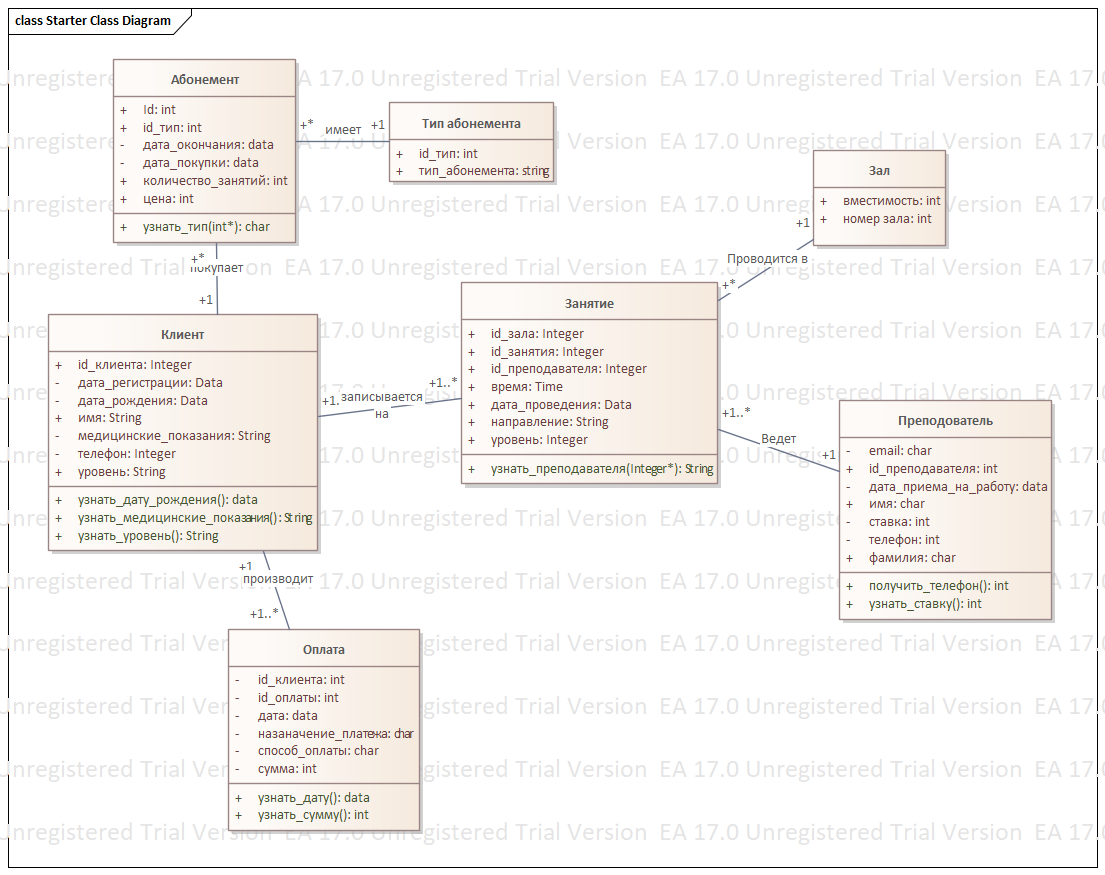
****

Рис. 4.2. Диаграмма классов

**4.3. ДИАГРАММА ACTIVITY**

На диаграмме представлен процесс публикации расписания.

Сначала преподавателям необходимо в личном кабинете создать занятия себе на неделю. При создании нового занятие появляется уведомление для администратора, которое он получает. После этого администратор назначает в ново-созданному занятию зал проведения. Все созданные занятия и присвоенные им залы и есть расписание. Администратор проверяет расписание на наличие не состыковок, если таких нет, то администратор согласовывает расписание и отправляет уведомление об этом. Если есть недочеты, то администратор редактирует расписание, после чего так же отправляет уведомление о принятии расписания. После всех этих действий расписание публикуется.

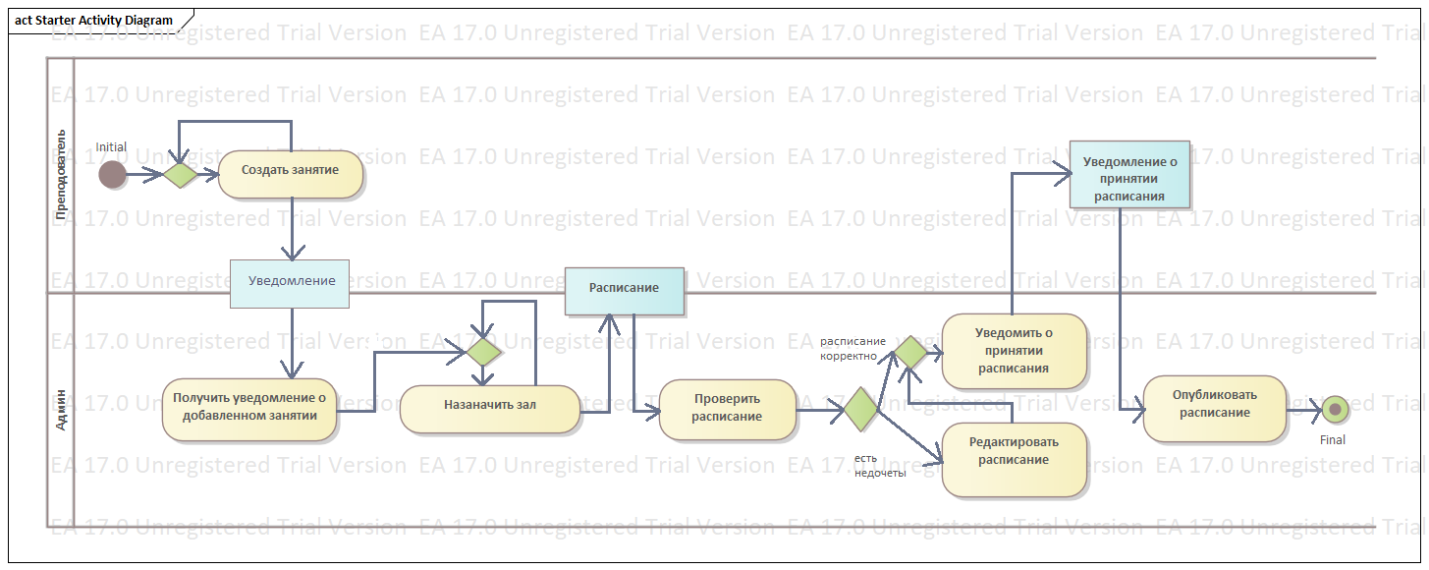


Рис. 4.3. Диаграмма activity

**4.4. ДИАГРАММА STATE MASHINE**

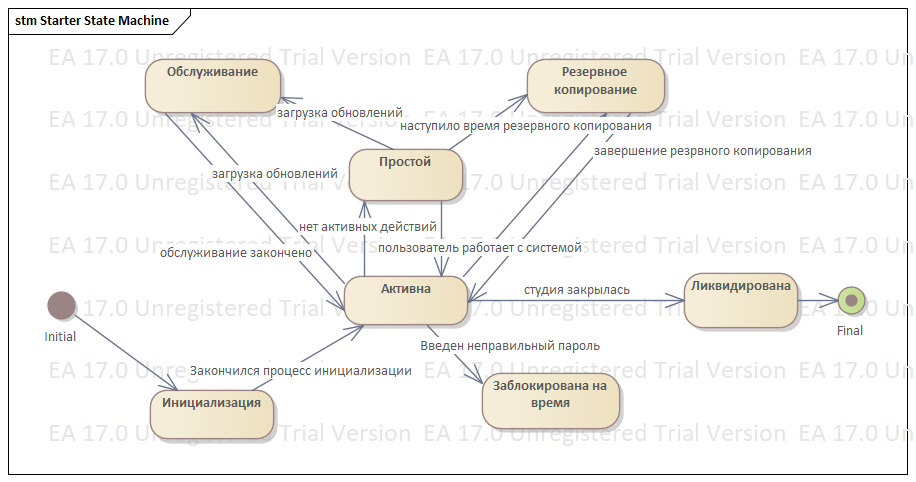


Рис. 4.4. Диаграмма State mashine

Инициализация – состояние при запуске системы, когда происходит загрузка необходимых компонентов и настроек.

Активна – основное рабочее состояние, когда система готова к выполнению запросов пользователей и обработке данных.

Простой – состояние, когда система работает, но не выполняет никаких активных задач. Она ожидает запросов от пользователей.

Обслуживание – состояние, когда система находится в режиме обслуживания (например, для обновления программного обеспечения или устранения неполадок). Во время обслуживания доступ к системе может быть ограничен или полностью недоступен.

Заблокирована на время – состояние, когда система заблокирована (например, из-за превышения числа неудачных попыток входа) и недоступна для пользователей.

Резервное копирование – состояние, когда система сохраняет собственные данные.

Ликвидирована – состояние, когда система завершила свою работу.

1. **ТЕСТЫ**

Тесты на нефункциональные требования:

1. Создать Личный кабинет. Входя в личный кабинет, ввести неправильный пароль. Проверишь, что система не пускает пользователя с неправильным паролем. Вести правильный пароль, проверить, что система пускает пользователя. Ввести неправильный пароль четыре раза, проверить реакцию системы.
2. Нажать на кнопку зарегистрироваться, измерить, сколько секунд отвечает система. Проверить так все кнопки системы (авторизация, зайти в личный кабинет, зарегистрироваться, записаться на занятие, и т.д.).
3. Понаблюдать за системой 30 суток и проверить, что резервное копирование данных выполняется каждые сутки.
4. Одновременно в систему должны зайти 1000 пользователей. Проверить, что система работает корректно.
5. Зайти в систему с компьютера, проверить, что данные отображаются корректно. Зайти в систему с планшета, проверить, что данные отображаются корректно. Зайти в систему со смартфона, проверить, что данные отображаются корректно.

Тесты на функциональные тесты:

1. Зайти в интерфейс системы и проверить наличие всех функций.
2. Протестировать возможность зарегистрироваться в системе.
3. Протестировать возможность авторизоваться.
4. Проверить в личном кабинете разных пользователей наличие всех функций.
5. Протестировать возможность просмотра расписания.
6. Протестировать возможность редакции и создания расписания.
7. Протестировать возможность согласования расписания.
8. Протестировать возможность записи на занятие.
9. Протестировать возможность покупки абонемента.
10. Протестировать возможность отправки и получения уведомления.
11. Протестировать наличие информации о клиентах в ЛК преподавателей и админов.
12. Протестировать возможность отмечать проведение занятий.
13. Протестировать возможность архивировать и редактировать информацию учетных записей.
14. Протестировать возможность редактировать личную информацию в ЛК.
15. Протестировать наличие чата поддержки у клиентов.

**ЗАКЛЮЧЕНИЕ**

В лабораторной работе была успешно спроектирована информационная система для танцевальной студии. Создание информационной системы для танцевальной студии сильно упрощает некоторые бизнес-процессы, помогает привлекать новых клиентов. С помощью такой системы пользователи смогут не забывать расписание, автоматически вовремя получать уведомления об изменениях. Руководству проще следить за работой своих сотрудников. В перспективе такая система поможет автоматизировать процесс отчетности.

**СПИСОК ИСПОЛЬЗОВАННОЙ ЛИТЕРАТУРЫ**

1. Водяхо А.И., Выговский Л.С., Дубенецкий В.А.Цехановский В.В., Архитектурные решения информационных систем. - СПб.: Издательство «Лань», 20 с.
2. Леоненков А. В. Самоучитель UML 2 – СПб.: БХВ-Петербург, 2007.– 576 с.
3. Рамбо, Якобсон, Буч: Введение в UML от создателей языка.