**Министерство науки и высшего образования Российской Федерации**

**федеральное государственное автономное образовательное учреждение высшего образования**

**“НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО”  
(УНИВЕРСИТЕТ ИТМО)**

**ЦЕНТР АВТОРИЗОВАННОГО ОБУЧЕНИЯ ИНФОРМАЦИОННЫМ ТЕХНОЛОГИЯМ**

**ИТОГОВАЯ АТТЕСТАЦИОННАЯ РАБОТА**

**Разработка Web-приложения для автоматизации планирования посещений для клиентов фитнес-клуба**



Автор Симонова Маргарита Игоревна \_\_\_\_\_\_\_\_\_\_\_\_\_

(Фамилия Имя Отчество) (Подпись)

**Центр авторизованного обучения информационным технологиям**

Наименование программы **«Python-разработчик»**

Руководитель Кузьмин Константин Михайлович \_\_\_\_\_\_\_\_\_\_\_\_

(Подпись)

**К защите допустить**

|  |  |  |
| --- | --- | --- |
| *Заместитель директора ЦАО ИТ,*  *к.т.н.* |  | / *Т.В. Зудилова*/ |

Санкт-Петербург, 2024г.

Обучающийся Симонова Маргарита Игоревна Группа 124/12

(Фамилия, И. О.)

Работа принята «\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_ 2024г.

Работа выполнена с оценкой \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Дата защиты «\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_ 2024г.

Секретарь ИАК \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(Фамилия, И. О.) (подпись)

Листов хранения \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Демонстрационных материалов \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

СОДЕРЖАНИЕ

[ВВЕДЕНИЕ 4](#_Toc160398301)

[ВЫПОЛНЕНИЕ РАБОТЫ 6](#_Toc160398302)

[ЗАКЛЮЧЕНИЕ 19](#_Toc160398303)

[СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ 20](#_Toc160398304)

[ПРИЛОЖЕНИЕ 1 21](#_Toc160398305)

**ВВЕДЕНИЕ**

Основываясь на своем опыте посещения фитнес-клубов, в качестве клиента, хотелось бы отметить, что периодически я сталкивалась с проблемой слишком большого количества одновременно занимающихся. Перед тем как идти на занятие мне бы хотелось видеть примерное количество желающих прийти в это же время. Решением данной проблемы, на мой взгляд, могло бы стать создание автоматизированной системы планирования посещений зон клуба. Также, вероятно фитнес-клубу тоже бы хотелось иметь такие данные, ведь это бы позволило им распределить нагрузку равномернее, более гибко управлять тарифами. В связи с представленными выше доводами, считаю данную тему актуальной.

Целью дипломной работы является разработка системы для контроля загруженности фитнес-клуба и планирования клиентами времени посещения.

Для достижения поставленной цели необходимо выполнить следующие задачи:

1. Проанализировать предметную область.
2. Сформулировать функциональные требования к системе и задокументировать их с помощью модели прецедентов (диаграммы вариантов использования).
3. Разработать модель предметной области.
4. Спроектировать алгоритм решения задачи согласно процедурной парадигме.
5. Спроектировать диаграмму последовательностей.
6. Выбрать подход к разработке системы.
7. Создать иерархию классов системы.
8. Связать объекты с классами, сообщения на диаграммах взаимодействия – с операциями.
9. Каждый класс снабдить описанием, которое должно включать в себя краткое описание (ответственность класса), описание атрибутов в виде таблицы (имя, описание, тип), таблицу с описанием операций (имя, описание, сигнатура).
10. Построить диаграммы классов системы, отображающие связи между классами.
11. Для описания поведения экземпляров отдельных классов построить диаграммы состояний и деятельности.
12. Выбрать средства реализации.
13. Разработать и создать БД.
14. Разработать серверную часть приложения.
15. Протестировать.
16. Разработать способ развертывания системы.
17. Разработать инструкции.

**ВЫПОЛНЕНИЕ РАБОТЫ**

**Анализ предметной области**

*Концепция проекта*

Для администрации фитнес-клуба, а также для их клиентов, система «Training plan» станет информационной системой, которая позволит отслеживать количество посетителей, планирующих визит в определенное время. Для клиентов фитнес-клуба она поможет спланировать посещение в менее нагруженное время. Для фитнес-клуба введение данной системы в эксплуатацию позволит распределить нагрузку, уменьшить пиковое количество посетителей, что позволит увеличить удовлетворенность предоставляемыми услугами у клиентов, а также позволит разработать новые тарифные планы. В условиях пандемии, когда по санитарно-эпидемиологическим нормам посещение бассейна, а также групповых тренировок было ограничено в количестве занимающихся, данная система бы очень пригодилась. Также на некоторые классы занятий количество мест ограничено и запись обычно ведется вручную, что ведет к возможным ошибкам и нарушениям норм, автоматизация процесса позволит повысить качество предоставляемых услуг.

*Типы пользователей и роли*

Администратор – выполняет функции менеджера пользователей, имеет доступ к данным фитнес-клуба.

Менеджер – составляет расписание занятий, просматривает статистику планируемых посещений.

Клиент клуба – подает заявку на планируемое посещение, просматривает статистику планируемых посещений.

*Ограничения проекта*

Фитнес-клуб может быть сетью - иметь несколько площадок, а может иметь только одну площадку. Зоны групповых занятий постоянны, то есть в фитнес-клубе выделены залы только под групповые занятия, планировать визит в такие зоны без группового занятия нельзя.

**Функциональные требования к системе**

Представлены в таблице 1.

Таблица 1 – Функциональные требования

|  |  |
| --- | --- |
| **Функциональные требования** | **В рамках проекта** |
| 1. Система обеспечивает возможность пользователю заходить в систему со своими учетными данными | 1.1 Создание в базе данных таблицы, отражающей данные пользователя,  где указана роль (менеджер,  администратор, клиент) |
|  | 1.2 Реализация системы авторизации  и аутентификации пользователя |
| 2. Система обеспечивает возможность пользователю выбрать из главного меню необходимый функционал | 2.1 Создание на всех формах панели с главным меню |
| 3. Система обеспечивает возможность пользователям просматривать количество клиентов планирующих посетить каждую из зон клуба в течении выбранного дня | 3.1 Создание в базе данных таблицы, отражающей планируемые посещения |
|  | 3.2 Создание формы, отражающей количество планируемых посещений за выбранный день |
| 4. Система позволяет администратору  управлять учетными данными пользователей | 4.1 Создание формы работы с учетными данными (создание, изменение, удаление) |
|  | 4.2 Реализация для администратора  возможности создания, изменения, удаления учетных записей |
| 5. Система позволяет администратору менять данные о фитнес-клубе | 5.1 Создание в базе данных таблицы  с информацией о фитнес-клубе и таблицы о доступных зонах и групповых занятиях |
|  | 5.2 Создание формы работы с данными фитнес-клуба |
| 6. Система позволяет менеджеру  составлять расписание групповых занятий, указывая оптимальное количество присутствующих | 6.1 Создание в базе данных таблицы с информацией о расписании занятий с оптимальным количеством присутствующих |
|  | 6.2 Создание формы работы с расписанием |
| 7. Система позволяет клиенту заявлять о своем планируемом посещении | 7.1 Создание в базе данных таблицы, отражающей планируемые посещения клиентами |
|  | 7.2 Обеспечение возможности создания, удаления планируемых посещений |
|  | 7.3 Создание форм для создания, удаления планируемых посещений |

**Диаграмма вариантов использования Убрать ввод пользовательского пароля**

Представлена на рисунке 1.

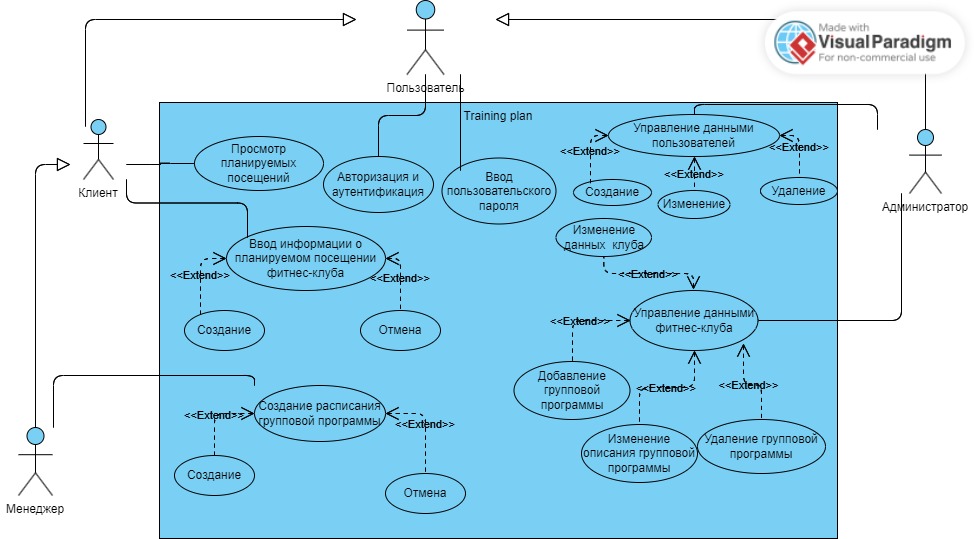


Рисунок 1 – Диаграмма вариантов использования

**Описание вариантов использования**

***Сценарий 1 – Авторизация и аутентификация***

Вариант использования: авторизация и аутентификация пользователя в системе

Участники: пользователь

Цель: авторизация и аутентификация

Краткое описание: пользователь вводит логин, пароль, открывается форма, соответствующая роли пользователя

Типичный ход событий:

1. Пользователь вводит логин, пароль

*Исключение 1 Введенный логин не существует*

*Исключение 2 Введенный пароль неверный*

Открывается форма, соответствующая роли пользователя

Раздел исключений:

*Исключение 1 Введенный логин не существует*

1. Пользователю выводится сообщение, что такого логина не существует, предлагается ввести данные снова
2. Пользователь вводит логин, пароль до тех пор, пока логин и пароль не будут легитимными или закрывает форму

*Исключение 2 Введенный пароль неверный*

1. Пользователю выводится сообщение, что пароль неверный, предлагается ввести пароль снова

2. Пользователь вводит пароль до тех пор, пока пароль не будет легитимным или закрывает форму

***Сценарий 2 – Управление данными пользователей***

Вариант использования: создание, изменение, удаление данных пользователя

Участники: администратор

Цель: настройка и создание учетных записей, поддержка актуальных данных о пользователях

Краткое описание: администратор вводит данные пользователя, присваивает ему роль, изменяет данные пользователя

Типичный ход событий:

1. Если администратору необходимо создать новую учетную запись, он вводит данные пользователя: ФИО, дата рождения, номер клубной карты (если это клиент или менеджер), логин, пароль, телефон, email, роль, дата создания учетной записи

*Исключение 1 Дублирование аккаунта*

*Исключение 2 Введенные данные некорректны*

1. Если данные пользователя изменились, администратор ищет учетную запись пользователя, меняет данные, сохраняет изменения

*Исключение 1 Дублирование аккаунта*

*Исключение 2 Введенные данные некорректны*

Раздел исключений:

*Исключение 1 Дублирование аккаунта*

1. Если введенные данные дублируют уже существующие данные администратору выводится об этом сообщение, запись не добавляется (не изменяется)
2. Администратор повторяет попытку ввода до тех пор, пока данные не станут уникальными или закрывает форму

*Исключение 2 Введенные данные некорректны*

1. Если введенные администратором данные не проходят проверки корректности, система выводит сообщение об этом, запись не создается (не изменяется)
2. Администратор повторяет попытку ввода данных до тех пор, пока данные не станут уникальными или закрывает форму

***Сценарий 3 – Управление данными фитнес-клуба***

Вариант использования: изменение данных фитнес клуба, изменение состава групповых программ

Участники: администратор

Цель: актуальное описание фитнес-клуба

Краткое описание: администратор изменяет данные фитнес-клуба, если это необходимо, изменяет состав групповых программ, их описание

Типичный ход событий:

1. В случае если данные клуба претерпели изменения, администратор изменяет данные

*Исключение 1 Введенные данные некорректны*

1. В случае если необходимо изменить описание групповой программы, администратор выбирает групповую программу из списка существующих, изменяет ее данные

*Исключение 1 Введенные данные некорректны*

1. В случае если необходимо добавить новую групповую программу, администратор создает новую запись

*Исключение 1 Введенные данные некорректны*

*Исключение 2 Дублирование данных*

1. В случае если групповая программа более не существует в данном клубе, администратор проставляет дату закрытия

Раздел исключений:

*Исключение 1* *Введенные данные некорректны*

Раздел исключений:

*Исключение 1* *Введенные данные некорректны*

1. Если администратор ввел некорректные данные, ему выводится сообщение, после чего данные не сохраняются
2. Администратор вводит данные до тех пор, пока они не станут корректными или закрывает форму

*Исключение 2 Дублирование данных*

1. В случае попытки повторного ввода групповой программы с тем же наименованием, администратору выводится сообщение, данные не сохраняются
2. Администратор вводит данные до тех пор, пока не введет уникальное наименование или не закроет форму

***Сценарий 4 – Создание расписания групповой программы***

Вариант использования: поддержка актуальности расписания групповой программы

Участники: менеджер

Цель: создание, удаление записи о предстоящем занятии групповой программы

Краткое описание: менеджер создает записи о предстоящем занятии групповой программы, указывает предполагаемое количество занимающихся, в случае отмены занятия менеджер помечает занятие как отмененное, сообщает клиентам об отмене занятий (по телефону), у клиентов появляется сообщение об отмененном занятии.

Типичный ход событий:

1. В случае если необходимо создать расписание предстоящего занятия, менеджер создает записи, содержащие данные о самом занятии, о дате и времени, о тренере
2. Если занятие отменяется по каким-то причинам, менеджер помечает занятие как отмененное, сообщает об этом клиентам, в планируемых посещениях клиента появляется отметка об отмене занятия

***Сценарий 5 – Ввод информации о планируемом посещении фитнес-клуба***

Вариант использования: запись данных о планируемом посещении клиента, удаление записи

Участники: клиент

Цель: создание записи о планируемом посещении, удаление

Краткое описание: клиент, ориентируясь на статистику планируемых посещений клиентов, вводит информацию о планируемом визите фитнес-клуба – выбирает дату, время занятия, зону посещения. В случае срыва планов, удаляет запись.

Типичный ход событий:

1. Клиент вводит данные о планируемом визите.

*Исключение 1 – На указанное время уже есть планируемое посещение*

1. В случае необходимости отмены планируемого посещения, удаляет его

Раздел исключений:

*Исключение 1 – На указанное время уже есть планируемое посещение*

1. Клиент вводит дату и время планируемого занятия, совпадающие с уже запланированными посещениями этого клиента
2. Клиенту выводится сообщение об ошибке, клиент либо отменяет неактуальное посещение, либо меняет дату и время текущего

***Сценарий 6 – Просмотр планируемых посещений на выбранную дату***

Вариант использования: просмотр планируемых посещений на выбранную дату

Участники: клиент, менеджер

Цель: оценка загруженности фитнес-клуба

Краткое описание: клиент или менеджер просматривает планируемые посещения

Типичный ход событий:

1. Клиент или менеджер выбирает дату, которая представляет интерес
2. Клиент или менеджер просматривает аналитику

**Модель предметной области**

После определения функциональных требований к приложению и разработки прецедентов требуется разработать **модель предметной области** – определить основные сущности в терминах предметной области.

**Алгоритм решения задачи согласно процедурной парадигме**

**Диаграмма последовательностей**

**Подход к разработке системы**

В связи с тем, что проект небольшой, с отсутствием четкой спецификации, задание и требования становятся более ясными в процессе разработки, полагаю, подход к разработке системы должен быть гибким. В связи с тем, что я разрабатываю систему одна, результат примерно ясен, задачи для меня проще всего разбить на подзадачи, поэтому, вероятно, ближе всего модель Kanban.

1. Создать иерархию классов системы.
2. Связать объекты с классами, сообщения на диаграммах взаимодействия – с операциями.
3. Каждый класс снабдить описанием, которое должно включать в себя краткое описание (ответственность класса), описание атрибутов в виде таблицы (имя, описание, тип), таблицу с описанием операций (имя, описание, сигнатура).
4. Построить диаграммы классов системы, отображающие связи между классами.
5. Для описания поведения экземпляров отдельных классов построить диаграммы состояний и деятельности.

**Средства реализации**

Django, MySQL

**Разработка и создание БД ИЗМЕНИТЬ**

На рисунке 2 представлена EER-диаграмма.

Рисунок 2 – EER-диаграмма

**Разработка серверной части приложения**

**Тестирование**

**Способ развертывания системы**

**Инструкции**

**ЗАКЛЮЧЕНИЕ**

Выводы о проделанной работе: решены ли все задачи, достигнута ли поставленная цель.

С какими трудностями пришлось столкнуться. Как они решались.

Какие дополнительные возможности можно реализовать на базе сделанной работы.

**СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ**

1. Приложение к решению Ученого совета Университета ИТМО от «29» ноября 2022 г. № 15 «ТРЕБОВАНИЯ К ВЫПУСКНЫМ КВАЛИФИКАЦИОННЫМ РАБОТАМ». – URL: https://student.itmo.ru/files/1314 (дата обращения 16.03.2023).

**ПРИЛОЖЕНИЕ 1**

Название (например, Листинг …)