**Министерство образования Иркутской области**

Государственное бюджетное профессиональное

образовательное учреждение Иркутской области

«Иркутский авиационный техникум»

(ГБПОУИО «ИАТ»)

|  |  |
| --- | --- |
| КП.09.02.03.22.191.09 ПЗ |  |

ИНФОРМАЦИОННАЯ СИСТЕМА «ЙОГА КЛУБ»

|  |  |  |
| --- | --- | --- |
| Председатель ВЦК: | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  (подпись, дата) | (М.А. Кудрявцева) |
| Руководитель: | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  (подпись, дата) | (М.А. Кудрявцева) |
| Студент: | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  (подпись, дата) | (Д.А. Гусаков) |

Иркутск 2022

**Содержание**

[Введение 3](file:///G:\КП__Ильин%20Артур.docx#_Toc89852049)

[1.Описание предметной области 5](file:///G:\КП__Ильин%20Артур.docx#_Toc89852050)

[2.Анализ инструментальных средств разработки 7](file:///G:\КП__Ильин%20Артур.docx#_Toc89852051)

[3.Техническое задание 14](file:///G:\КП__Ильин%20Артур.docx#_Toc89852052)

[4.Проектирование ИС 15](file:///G:\КП__Ильин%20Артур.docx#_Toc89852053)

[4.1.Структурная схема ИС 15](file:///G:\КП__Ильин%20Артур.docx#_Toc89852054)

[4.2.Функциональная схема ИС 19](file:///G:\КП__Ильин%20Артур.docx#_Toc89852055)

[4.3.Проектирование базы данных 23](file:///G:\КП__Ильин%20Артур.docx#_Toc89852056)

[4.4.Проектирование интерфейса 26](file:///G:\КП__Ильин%20Артур.docx#_Toc89852057)

[5.Разработка ИС 29](file:///G:\КП__Ильин%20Артур.docx#_Toc89852058)

[5.1.Разработка интерфейса ИС 29](file:///G:\КП__Ильин%20Артур.docx#_Toc89852059)

[5.2.Разработка базы данных ИС 39](file:///G:\КП__Ильин%20Артур.docx#_Toc89852060)

[5.3.Разработка ИС 39](file:///G:\КП__Ильин%20Артур.docx#_Toc89852061)

[6.Документирование программного продукта 41](file:///G:\КП__Ильин%20Артур.docx#_Toc89852062)

[6.1.Руководство пользователя ИС 41](file:///G:\КП__Ильин%20Артур.docx#_Toc89852063)

[Заключение 50](file:///G:\КП__Ильин%20Артур.docx#_Toc89852064)

[Список используемых источников 51](file:///G:\КП__Ильин%20Артур.docx#_Toc89852065)

[Приложение А – Техническое задание 52](file:///G:\КП__Ильин%20Артур.docx#_Toc89852066)

[Приложение Б – Листинг 57](file:///G:\КП__Ильин%20Артур.docx#_Toc89852067)

**Введение**

Стресс, депрессия, кризис – прочно вошли в нашу жизнь. Из-за этих факторов человек находится всегда в напряжении, не умеет дать себе отдых, он спешит и суетится, мешает своей нервозностью жить и работать другим, и, в конце концов, становится обузой для себя и своего окружения. Рано или поздно такой образ жизни ведет к нервным расстройствам. Этот замкнутый круг невозможно разорвать приемом все новых и новых медикаментов. Занятие йогой откроют человеку двери в качественно новую жизнь, в особенности она, способствует поддержанию гармоничного душевного состояния, здоровья и энергичной активной жизни.

Для того, чтобы занятия йогой были структурированы и не несли вреда для людей, решивших заняться этим безусловно полезным занятием, существуют йога клубы, в них человек может заниматься йогой не опасаясь последствий, и, кроме того, иметь возможность находится в кругу единомышленников, заводить новые приятные знакомства, совершенствуя своё тело и разум.

Для того, чтобы в йога клубе царила атмосфера гармонии и порядка, необходимо обеспечить удобство клиентов, в том числе и в сфере информационных технологий, поэтому необходимо создать информационную систему, которая полностью покрывала бы потребности клиента, и исключала бы дискомфорт и неудобство с посещением клуба. Этим обусловлена актуальность данного курсового проекта.

Целью курсового проекта является создание информационной системы «Йога клуб».

Для достижения конечного результата, необходимо решить следующие задачи:

* исследовать выбранную предметную область ИС;
* выполнить анализ инструментальных средств разработки;
* подобрать литературу и интернет-источники по выбранной теме;
* разработать техническое задание;
* выполнить проектирование ИС;
* разработать интерфейс и структуру ИС;
* составить руководство пользователя.

1. **Описание предметной области ИС**

Предметной областью курсового проекта является йога клуб.

Йога клуб – это специально оборудованное помещение, в котором создается благоприятная атмосфера для практики йоги. Йога клуб имеет штат сотрудников: Администратор и тренеры, а также множество специализированных залов.

В йога клубе существует некоторое множество направлений йоги, которое может дополняться и корректироваться. Каждому направлению соответствует свой тренер, который является специалистом в данной области, имеет соответствующее образование и опыт, а также свой специализированный для этого направления зал. Тренер проводит занятия в соответствии с расписанием. Клиенты клуба разделяются по группам, группы составляются в соответствии с выбранным направлением, уровнем физической подготовленности и возрастом клиента.

Посещение клуба регулируется системой абонементов, существует три вида абонементов:

* Разовое посещение.
* Абонемент на месяц.
* Абонемент на год.

Стоимость абонементов не зависит от выбранного направления, когда клиент покупает абонемент, он имеет право посещать любое из представленных направлений, а также менять их по своему усмотрению на период действия абонемента.

Общие представление работы йога клуба выглядит следующим образом:

1. Клиент записывается на сайте выбирая нужный абонемент.
2. Администратор принимает и обрабатывает заявки с сайта, уточняет информацию с клиентом по телефонному разговору и в соответствии с полученной информацией определяет клиента в группу и регулирует расписание.
3. Тренеры проводят занятия.
4. **Анализ инструментальных средств разработки**

Для создания информационной системы «Йога клуб» были определены следующие инструментальные средства разработки:

1. HTML.
2. CSS.
3. JavaScript.
4. Bootstrap.
5. Python.
6. Django.
7. SQLite.
8. PyCharm.

HTML — это язык разметки, который представляет простые правила оформления и компактный набор структурных и семантических элементов разметки. HTML позволяет описывать способ представления логических частей документа (заголовки, абзацы, списки и т.д.) и создавать веб-страницы разной сложности.

CSS — технология описания внешнего вида документа, оформленного языком разметки. Преимущественно используется как средство оформления веб-страниц в формате HTML и XHTML, но может применяться с любыми видами документов в формате XML, включая SVG и XUL. Каскадные таблицы стилей используются создателями веб-страниц для задания цветов, шрифтов, расположения и других аспектов представления веб-документа. Основной целью разработки CSS являлось разделение содержимого (написанного на HTML или другом языке разметки) и оформления документа (написанного на CSS). Это разделение может увеличить доступность документа, предоставить большую гибкость и возможность управления его представлением, а также уменьшить сложность и повторяемость в структурном содержимом.

JavaScript — это язык программирования, как правило, выполняется на стороне клиента. Он используется для взаимодействия с пользователем. Он также используется в разработке игр, настольных и мобильных приложений, в создании документов pdf и виджетов рабочего стола. Веб-браузеры имеют встроенную поддержку для этого языка.

Особенности: Основная уникальность этого языка в том, что он имеет поддержку почти у всех браузеров и имеет полную интеграцию с ними. Все действия с помощью JavaScript являются очень простыми. Нет никакой другой технологии, которая бы вмещала в себя все подобные преимущества. Например, есть не кросс-браузеры, они имеют поддержку не у всех обозревателей (это XUL, ActiveX, VBScript). Также есть и такие, которые не в полной степени интегрированы с браузером. Это Silverlight, Flash, Java. На данный момент идет большое развитие подобной технологии и в разработке находятся элементы JavaScript 2.

Bootstrap — это бесплатный фреймворк с открытым исходным кодом для создания веб-сайтов и веб-приложений. Это самый популярный фреймворк HTML, CSS и JS для разработки адаптивных и мобильных проектов в Интернете. Одним из основных преимуществ фреймворков Bootstrap, является то, что они ускоряют время разработки, сохраняя при этом качество и согласованность всего сайта. Больше не нужно переделывать каждый элемент и тратить часы на то, чтобы всё выглядело и работало правильно на разных платформах, устройствах и в браузерах.

Python — это высокоуровневый язык программирования, который используется в различных сферах IT, таких как машинное обучение, разработка приложений, web, парсинг и другие.

Преимущества:

* Кроссплатформенность.
* Большое количество библиотек.
* динамическая типизация.
* Недостатки:
* Медлительность.
* Высокое потребление памяти.
* Строгая привязанность к системным библиотекам.

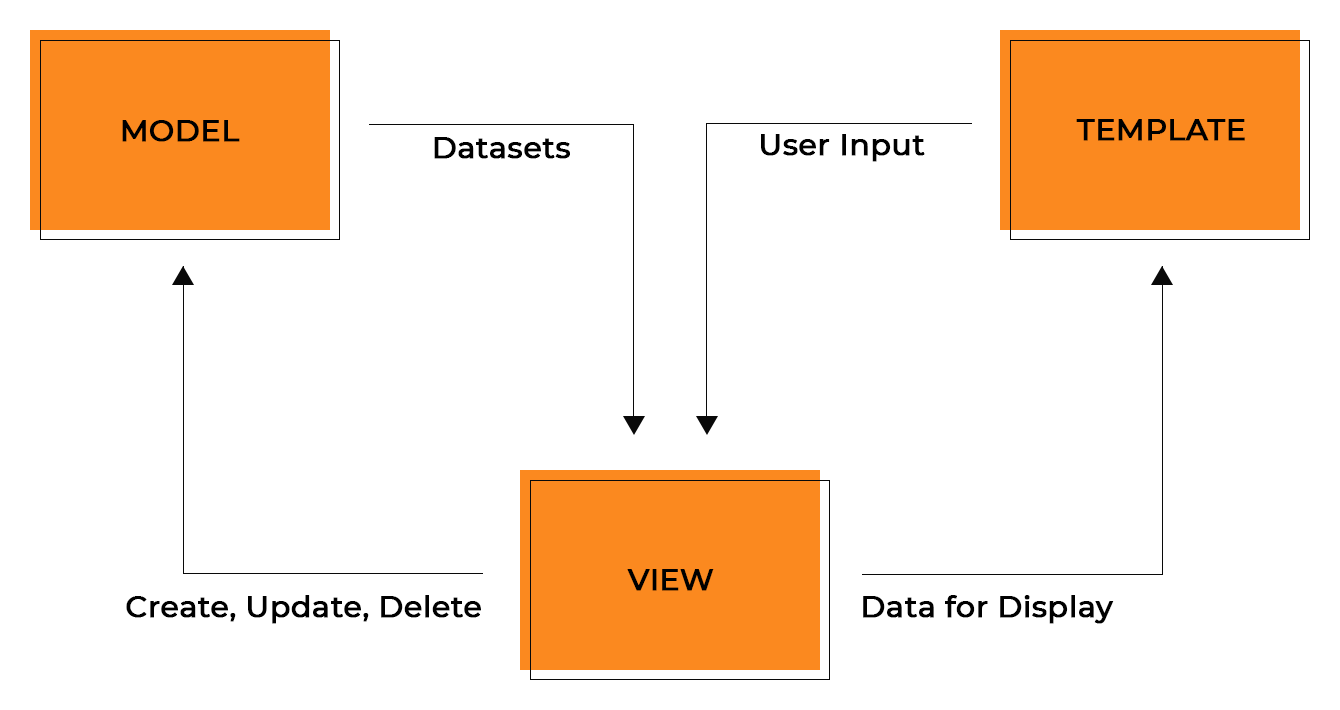
Django — это высокоуровневый Python веб-фреймворк, который позволяет быстро создавать безопасные и поддерживаемые веб-сайты.  
Создатели реализовали в Django паттерн MVT (модель-представление-шаблон), и он применяется в текущей версии фреймворка. Документация Django определяет модель (model) как «источник информации о данных, в которых содержатся ключевые поля и поведение данных». Обычно одна модель указывает на одну таблицу в базе данных. Django поддерживает базы данных PostgreSQL, MySQL, SQLite и Oracle. Представление решает три задачи: принимает HTTP-запросы, реализует бизнес-логику, определённую методами и свойствами, отправляет HTTP-ответ в ответ на запросы. То есть представление получает данные от модели и предоставляет шаблонам доступ к этим данным или предварительно обрабатывает данные и затем предоставляет к ним доступ шаблонам. Шаблоны представляют собой файлы с HTML-кодом, с помощью которого отображаются данные. Содержимое файлов может быть статическим или динамическим. Шаблоны не содержат бизнес-логики. Поэтому они только отображают данные. На рисунке 1 представлена схема, отображающая общее устройство паттерна MVT.

Рисунок 1 – Схема MVT

SQLite — компактная встраиваемая реляционная система баз данных. Является чисто реляционной системой баз данных. SQLite представляет библиотеку, которая написанна на языке C (ANSI-C) и которая реализует движок реляционных баз данных.На сегодняшний день SQLite, возможно, самая используемая система баз данных. Так, ее бд можно найти в каждом устройстве на Android, iOS, Mac, Windows 10/11, ее используются большинство распространенных браузеров - Firefox, Chrome, Safari и т.д. В отличие от других систем баз данных, как MS SQL Server, MySQL, Postgres и т.д., для SQLite не требуется сервер базы данных. SQLite представляет встраиваемый движок базы данных, который обращается напрямую к файлу базы данных на диске. Сооветственно для работы с базами данных не нужно явным образом устанавливать или как-то конфигурировать SQLite. SQLite имеет полноценную поддержку большинства возможностей, которыми обладают другие реляционные СУБД - таблицы, индексы, триггеры, представления.

PyCharm — это интегрированная среда разработки для Python, которая имеет полный комплект средств, необходимых для эффективного программирования на Python. Сейчас PyCharm распространяется в двух вариантах: платном (PyCharm Professional Edition) и бесплатном (PyCharm Community Edition). Бесплатная версия имеет открытый исходный код и распространяется под лицензией Apache 2. Это облегченная среда, которая подходит для разработки только на Python. Платный вариант представляет собой более расширенную и функциональную версию с возможностью разработки в том числе многоязычных веб-приложений, и дает возможность удаленной разработки, а также работы с базами данных.

Преимущества:

* PyCharm имеет удобный редактор кода со всеми полезными функциями: подсветкой синтаксиса, автоматическим форматированием, дополнением и отступами. PyCharm позволяет проверять версии интерпретатора языка на совместимость, а также использовать шаблоны кода.
* PyCharm позволяет быстро производить рефакторинг кода, а также использовать удобный графический отладчик.
* Утилита поддерживает все свежие версии Django, а также IronPython, Jython, Cython, PyPy wxPython, PyQt, PyGTK и многие другие инструменты.
* В PyCharm можно проводить интегрированное Unit тестирование, использовать интерактивные консоли для Python, Django, SSH, отладчика и баз данных.
* PyCharm имеет большую коллекцию плагинов, и его можно использовать в связке с разными трекерами вроде JIRA, Youtrack, Lighthouse, Redmine, Trac и так далее.
* PyCharm кросс-платформенная среда разработки: можно использовать на Linux, Windows и Mac OS.

1. **Техническое задание**

В начале разработки создавалось техническое задание, в котором указывались основные требования.

Согласно ГОСТ 19 техническое задание должно включать следующие разделы:

1. общие сведения;
2. назначение и цели создания системы;
3. требования к системе в целом;
   1. требования к структуре и функционированию системы;
   2. требования к надежности;
   3. требования к безопасности;
   4. требования к эксплуатации, техническому обслуживанию, ремонту и хранению компонентов системы;
4. требования к документированию;
5. состав и содержание работ по созданию системы.

Техническое задание на разработку приложения представлено в приложении А.

1. **Проектирование ИС**

Проектирование информационной системы позволяет разработчику получить структурированный чётки план действий, сокращает время разработки и повышает её качество, а также позволяет предусмотреть любые другие нюансы разработки.

Большая часть проектирования производится с помощью специальных схем и диаграмм, которые будут представлены далее.

* 1. **Структурная схема ИС**

Диаграмма прецедентов, представленная на рисунке 2, описывает какой функционал разрабатываемой информационной системы доступен каждой группе пользователей.

В данной информационной системе выделяется две группы пользователей: Клиенты и Персонал. Каждая из групп пользователей может пользоваться информационной системой по-своему.

****

Рисунок 2 – Диаграмма прецедентов

Диаграмма деятельности, изображённая на рисунке 3, представляет собой графическую схему рабочих процессов. Диаграмма деятельности используется для изучения бизнес-процессов с целью определения их потока и требований.

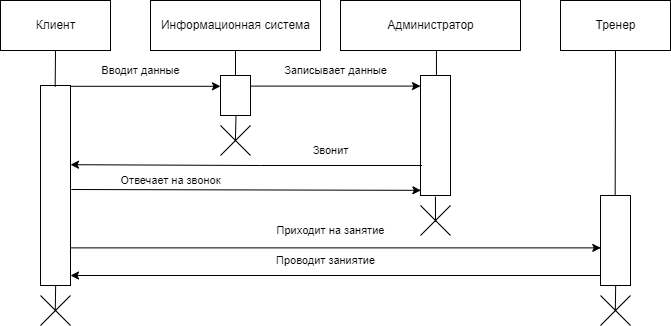


Рисунок 3 – Диаграмма деятельности

На рисунке 4 представлена диаграмма компонентов, которая визуализирует как компоненты информационной системой соединяются между собой для формирования более крупных компонентов или программных подсистем.

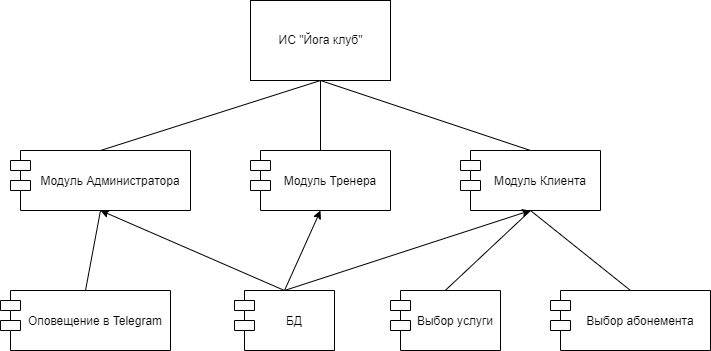


Рисунок 4 – Диаграмма компонентов

На рисунке 5 изображена диаграмма развёртывания, которая содержит графическое изображение устройств, процессов и связей между ними.

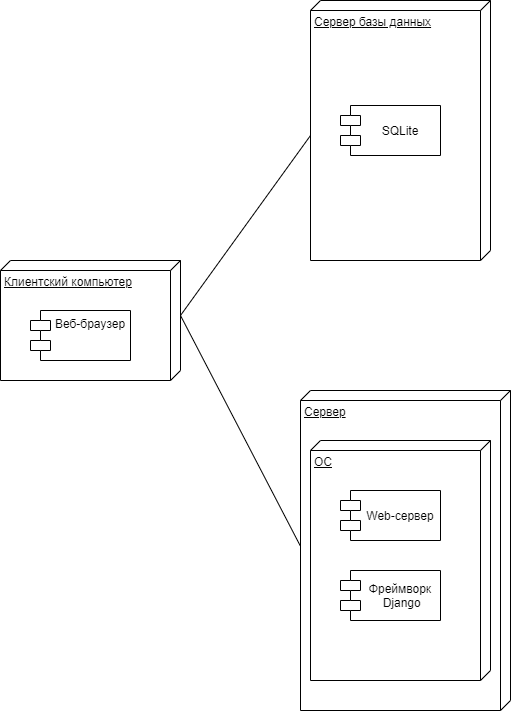


Рисунок 5 – Диаграмма развёртывания

* 1. **Функциональная схема ИС**

Контекстная диаграмма IDEF0, изображённая на рисунке 6, представляет систему как набор иерархических действий, в которой каждое действие преобразует некоторый объект или набор объектов.

Каждая IDEF0-диаграмма содержит блоки и дуги. Блоки изображают функции моделируемой системы. Дуги связывают блоки вместе и отобра­жают взаимодействия и взаимосвязи между ними.

На рисунке 7 представлена диаграмма декомпозиции, которая разбивает контекстную диаграмму на более детализированные компоненты более низкого уровня.

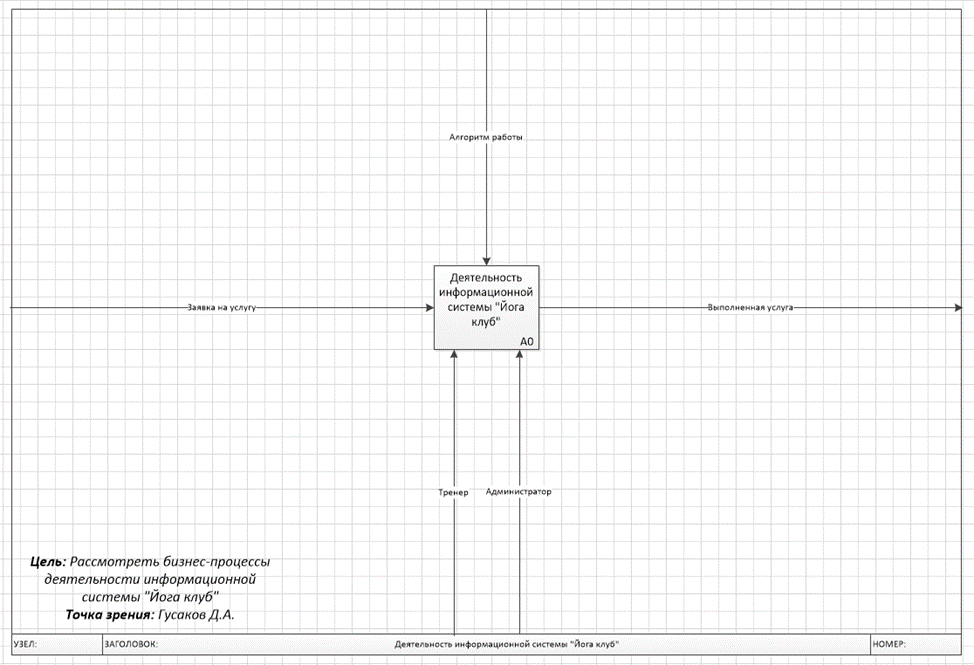
****

Рисунок 6 – Контекстная диаграмма IDEF0

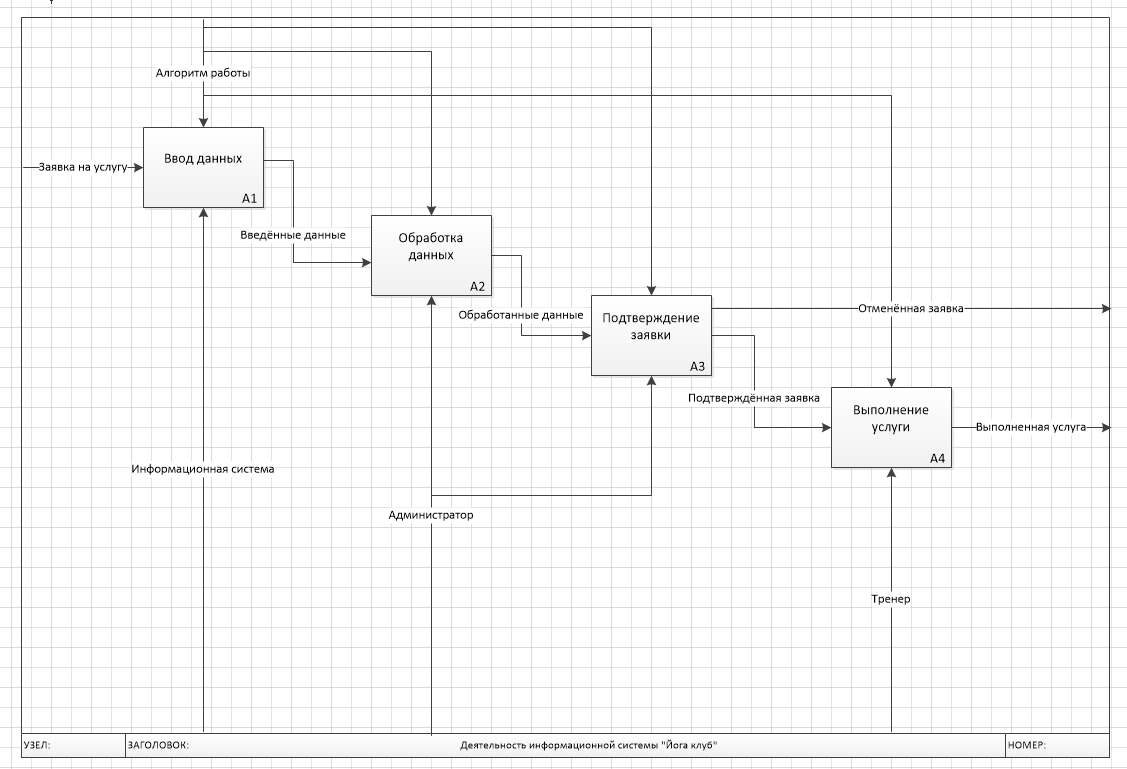


Рисунок 7 – Диаграмма декомпозиции

Диаграмма классов, изображённая на рисунке 8, предназначена для представления внутренней структуры программы в виде классов и связей между ними.

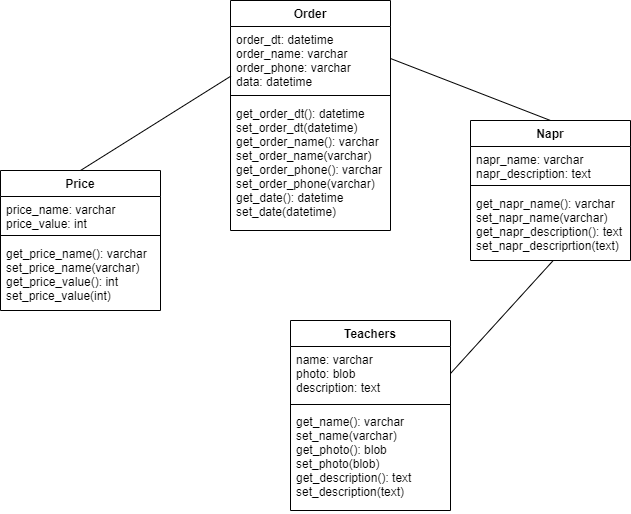


Рисунок 8 – Диаграмма классов

На рисунке 9 представлена диаграма потоков данных, предназначенная для моделированния информоционных систем с точки зрения действий над данными.

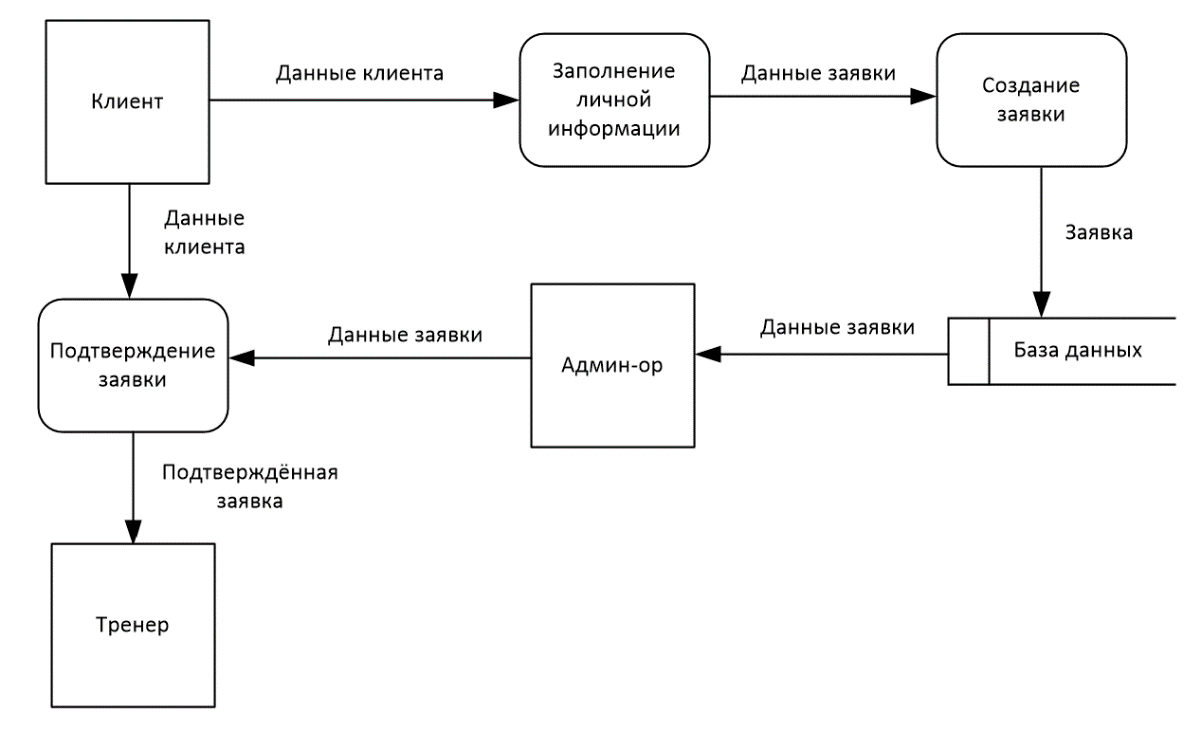


Рисунок 9 - Диаграмма потоков данных

* 1. **Проектирование базы данных**

Инфологическая модель, изображённая на рисунке 10, представляет собой отображение предметной области в виде совокупности информационных объектов и структурных связей

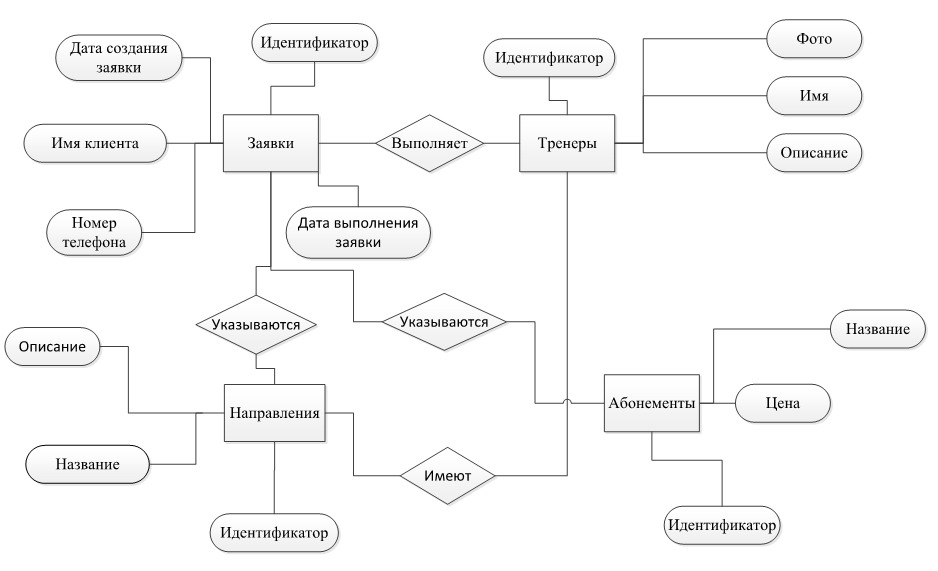


Рисунок 10 – Инфологическая модель

Даталогическая модель, изображённая на рисунке 11, представляет собой модель логического уровня системы, представляющая собой отображение логических связей между элементами базы данных в среде конкретной СУБД.

Исходными данными для даталогического проектирования является инфологическая модель предметной области.

Конечным результатом даталогического проектирования является логическая структура базы данных.

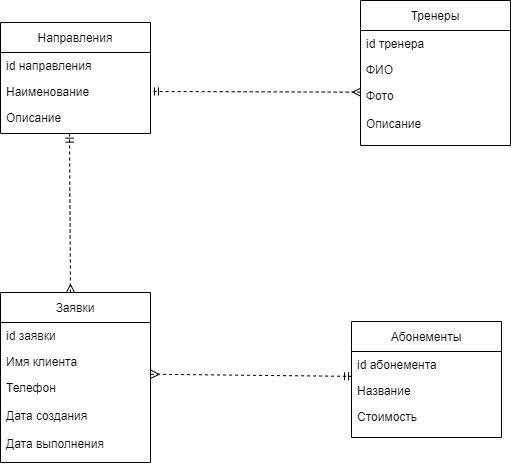


Рисунок 11 – Даталогическая модель

На рисунке 12 изображена ER-модель, позволяющая описывать концептуальные схемы предметной области.

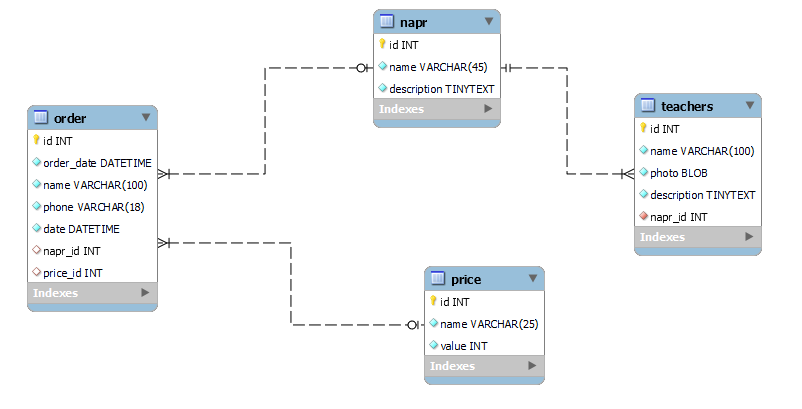


Рисунок 11 – ER-модель

На схеме базы данных представлены 4 таблицы: «order», «price», «napr» и «teachers». Поля id являются первичными ключами.

В таблицах 1, 2, 3, 4 представлены описания полей с типами данных для каждой таблицы данной ER-модели базы данных.

Таблица 1 – «order»

|  |  |  |
| --- | --- | --- |
| Поле | Тип данных | Описание |
| id | INT | Уникальный идентификатор заявки |
| order\_date | DATETIME | Дата создания заявки |
| name | VARCHAR(100) | Имя клиента |
| phone | VARCHAR(18) | Номер телефона |
| date | DATETIME | Дата выполнения заявки |
| napr\_id | INT | Идентификатор направления (внешний ключ) |
| price\_id | INT | Идентификатор абонемента (внешний ключ) |

Таблица 2 – «napr»

|  |  |  |
| --- | --- | --- |
| Поле | Тип данных | Описание |
| id | INT | Уникальный идентификатор направления |
| name | VARCHAR(45) | Наименование направления |
| description | TINYTEXT | Описание направления |

Таблица 3 – «price»

|  |  |  |
| --- | --- | --- |
| Поле | Тип данных | Описание |
| id | INT | Уникальный идентификатор абонемента |
| name | VARCHAR(25) | Наименование абонемента |
| value | INT | Стоимость |

Таблица 3 – «price»

|  |  |  |
| --- | --- | --- |
| Поле | Тип данных | Описание |
| id | INT | Уникальный идентификатор тренера |
| name | VARCHAR(100) | Имя тренера |
| photo | BLOB | Фотография тренера |
| description | TINYTEXT | Описание тренера |
| napr\_id | INT | Идентификатор направления (внешний ключ) |

База данных приведена к 3-ой нормальной форме так как все поля, принимаемые больше одного значения, декомпозированы, и каждый не ключевой атрибут не транзитивно зависит от первичного ключа.

* 1. **Проектирование интерфейса**

Для проектирования интерфейса был выбран инструмент Figma – бесплатный онлайн-сервис для создания прототипов сайтов или приложений.

В результате проектирования интерфейса были созданы прототипы трёх страниц: «Главная» (рисунок 13), «Тренеры» (рисунок 14), «Цены» (рисунок 15).

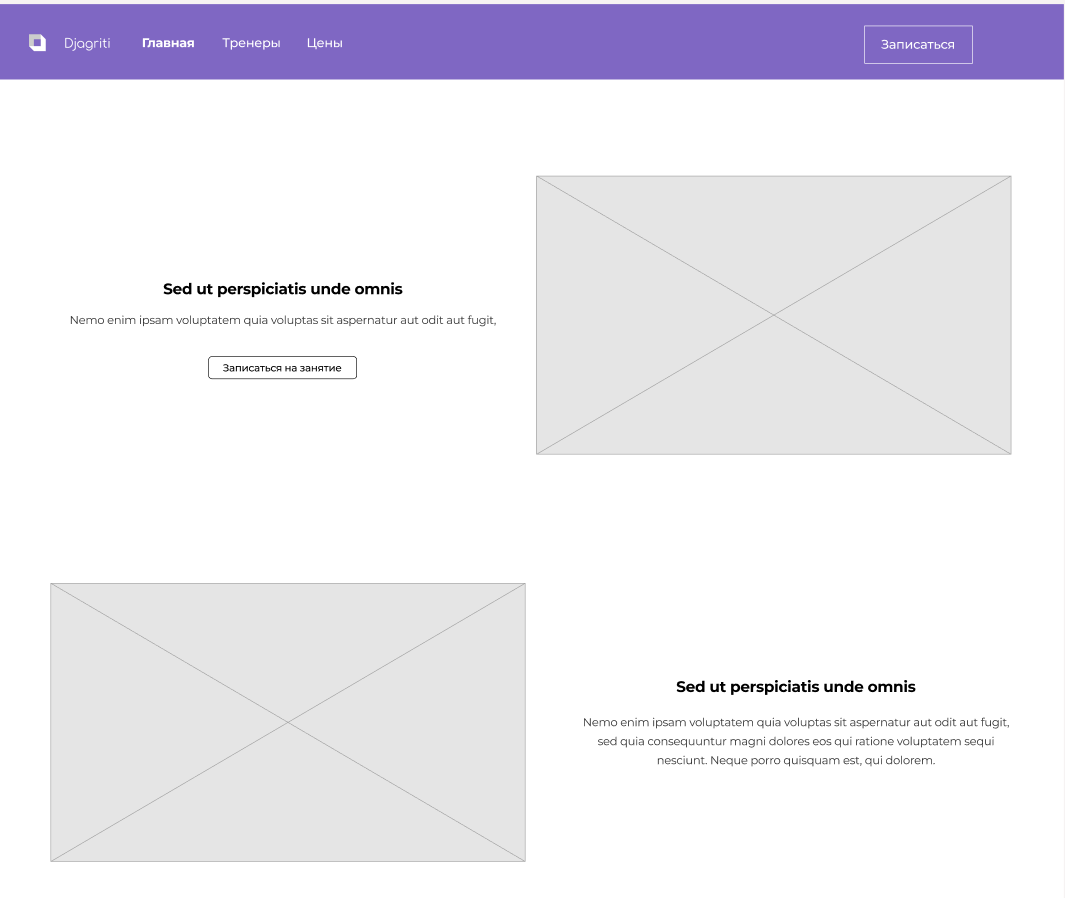


Рисунок 13 – Страница «Главная»

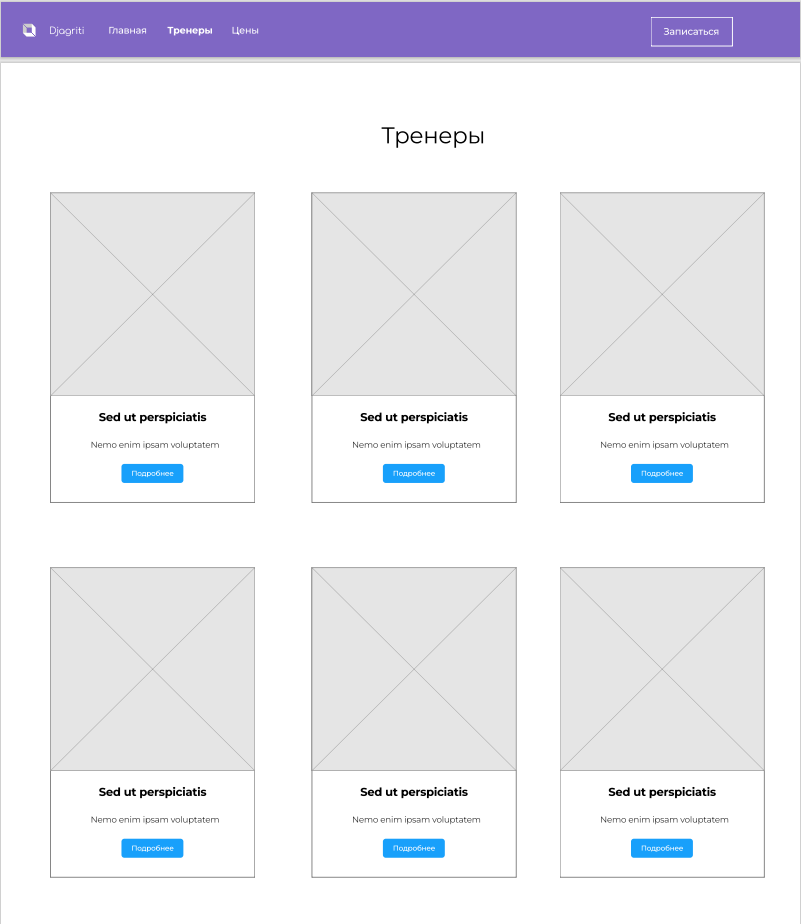


Рисунок 14 – Страница «Тренеры»

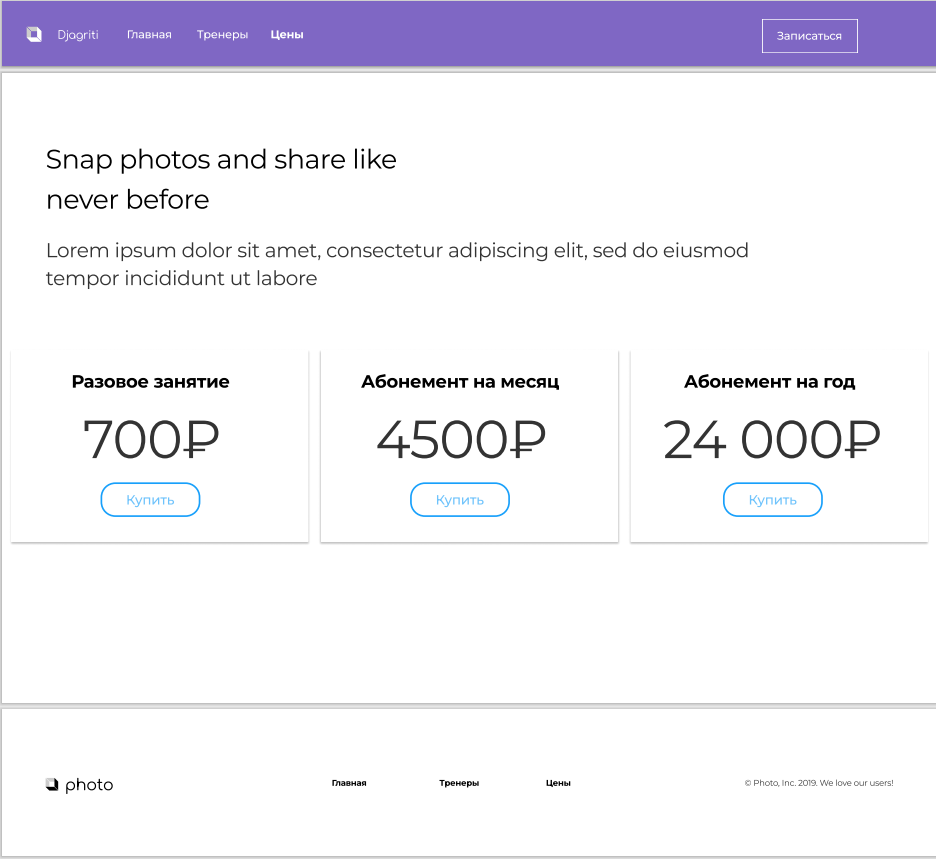


Рисунок 15 – Страница «Цены»

1. **Разработка ИС**
   1. **Разработка интерфейса ИС**

Для разработки пользовательского интерфейса были использованы такие инструменты как: HTML, CSS, Bootstrap.

В разрабатываемой информационной системе файлы, содержащие описание внешнего вида страниц, представлены в формате \*.html и \*.сss.

В фреймворке Django файлы формата \*.html называются «Шаблонами» и располагаются в каталоге «templates», а файлы формата \*.сss обычно располагаются в каталоге «static» и подключаются в html шаблонам посредством специального тега «{% load static %}»

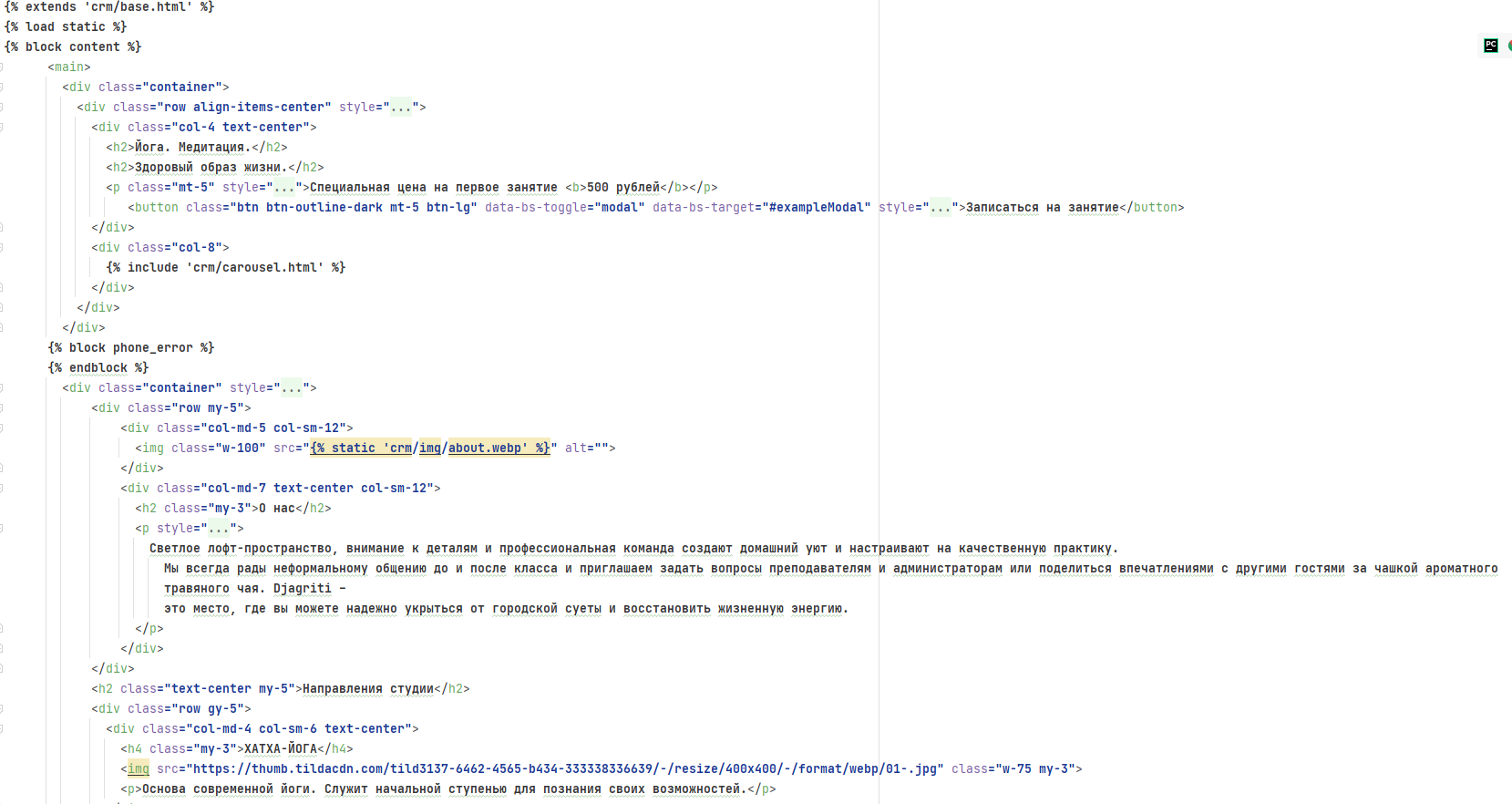


Рисунок 12 – HTML шаблон главной страницы

С помощью Bootstrap и CSS стилей были реализованы страницы: «Главная», «Тренеры», «Цены», «Авторизация», «Панель управления», которые представлены на рисунках 13, 14, 15, 16, 17, а также форма для записи (рисунок 18).

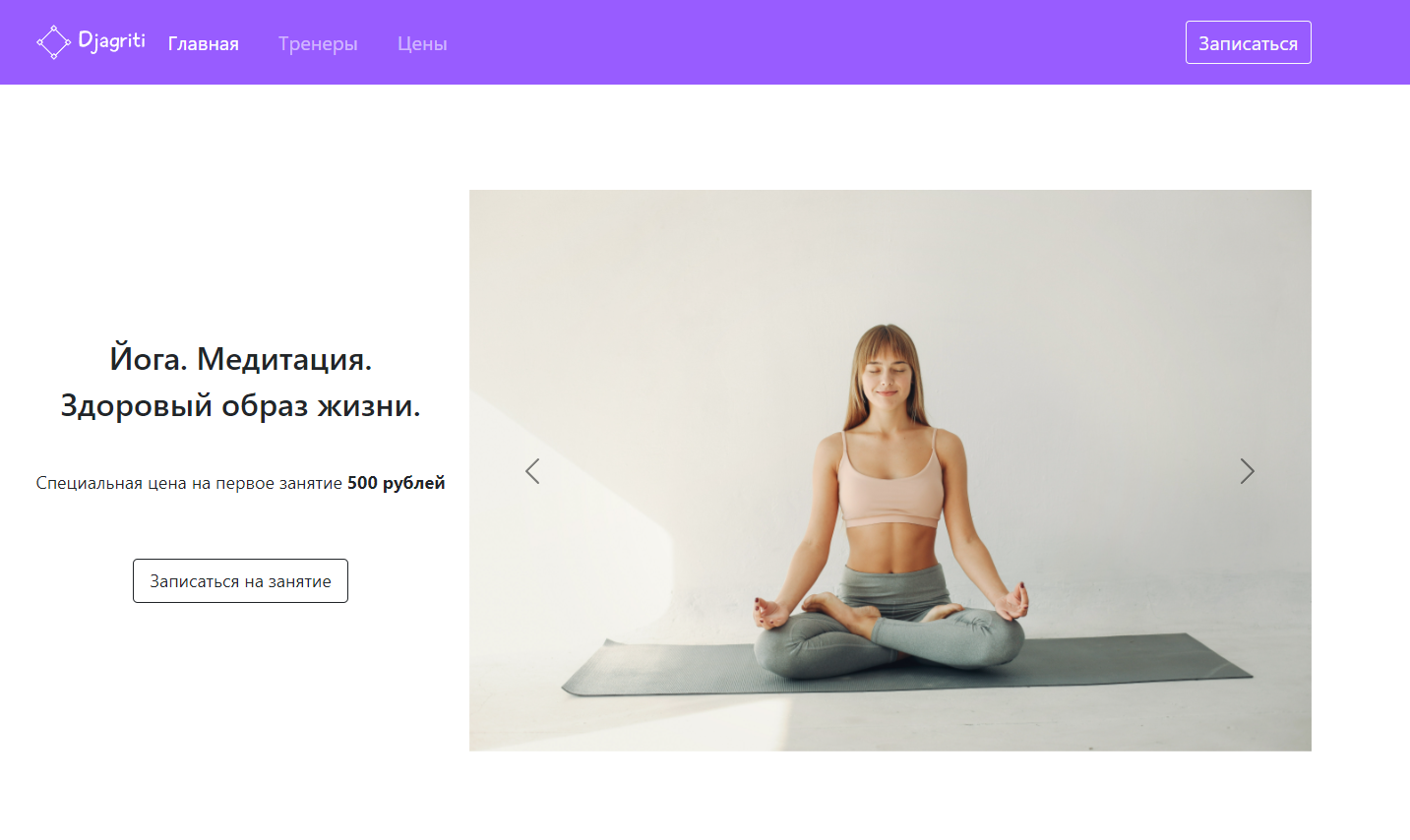


Рисунок 13 – Страница «Главная»

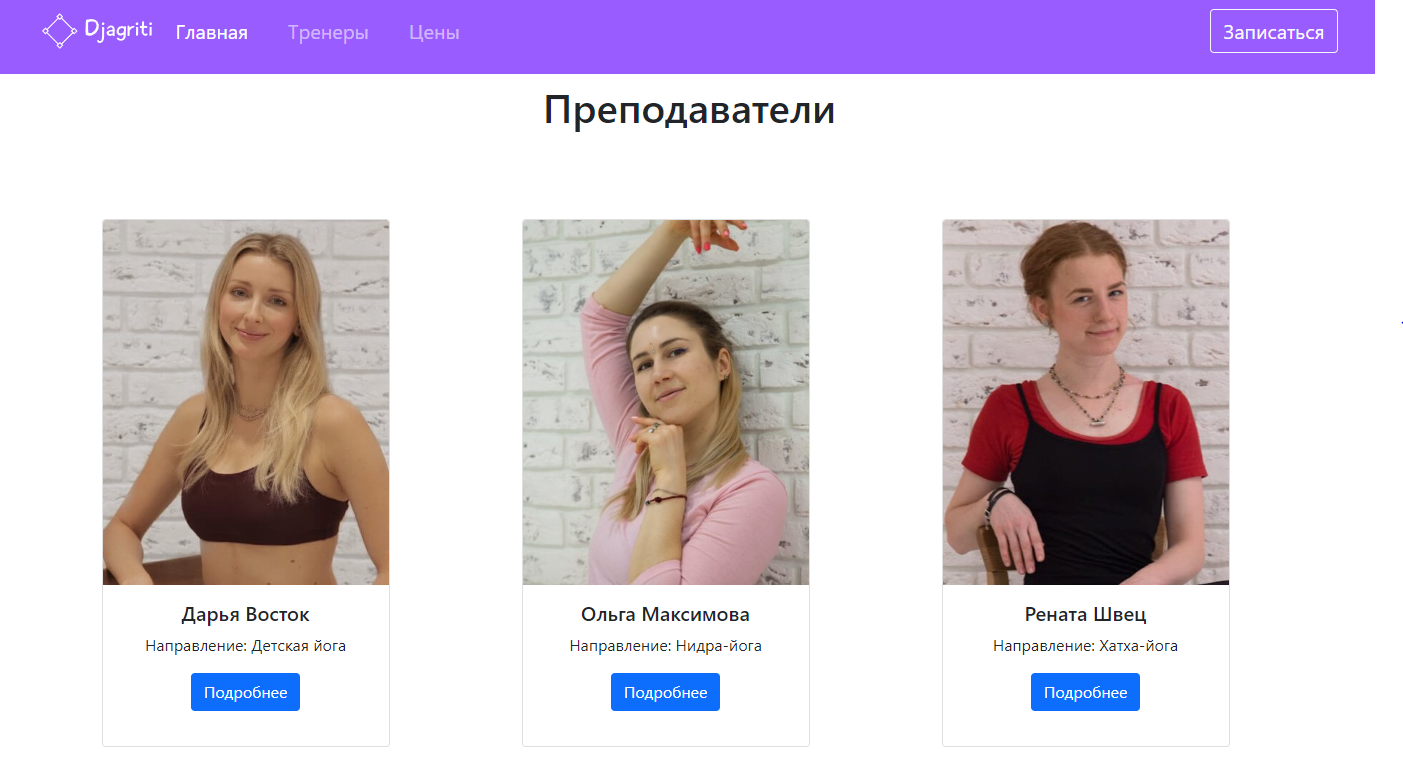


Рисунок 14 – Страница «Тренеры»

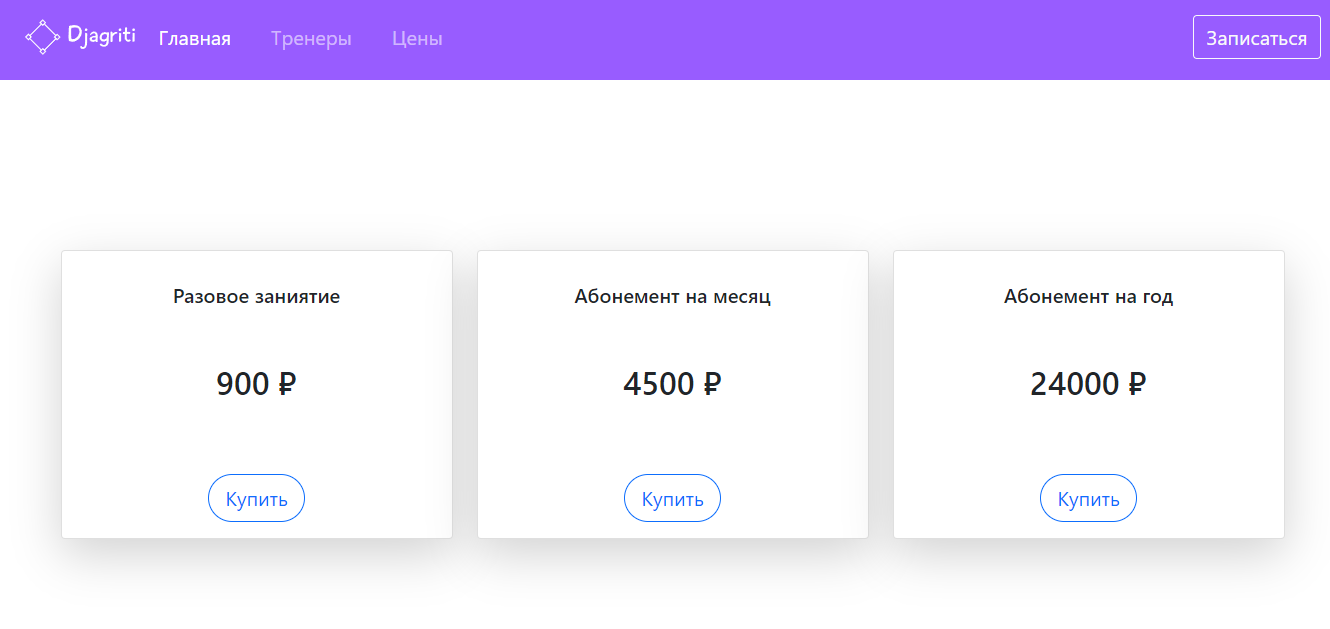


Рисунок 15 – Страница «Цены»

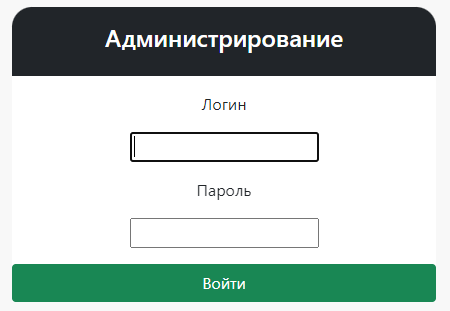


Рисунок 16 – Страница «Авторизации»

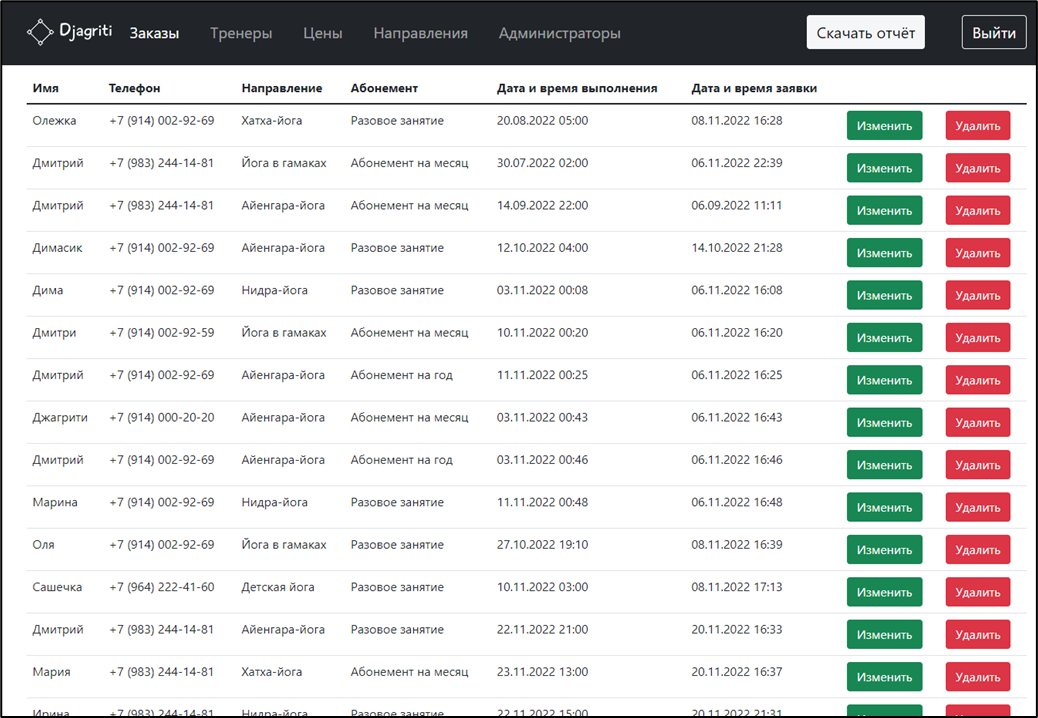


Рисунок 17 – Панель управления

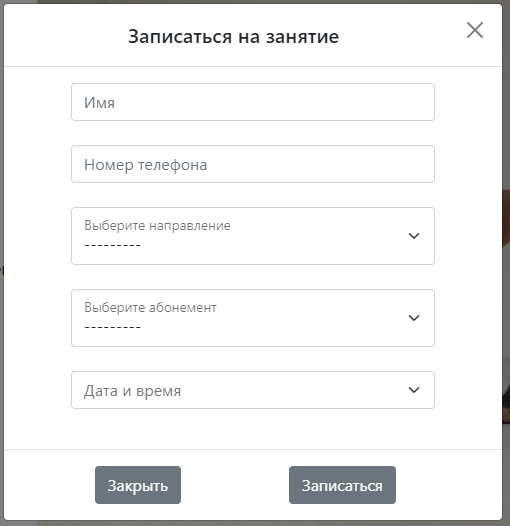


Рисунок 18 – Форма для записи

* 1. **Разработка базы данных ИС**

Разработка базы данных информационной системы «Йога клуб» реализовывалась в СУБД SQLite. База данных состоит из 4 таблиц.

В фреймворке Django используется ORM (объектно-реляционное отображение), которое позволяет представить описание таблиц базы данных в виде классов на языке Python. Все подобные классы, согласно архитектуре MVT, называются «моделями» и располагаются в файлах models.py и вместе образуют структуру базы данных.

На рисунках 19-22 изображены модели информационной системы.

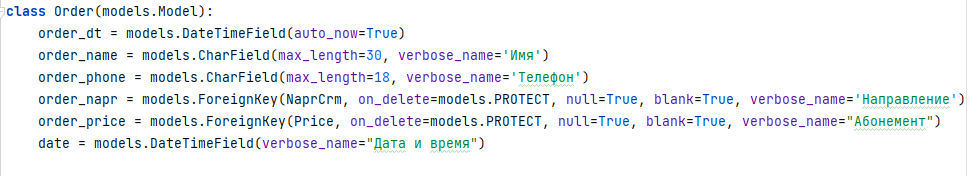


Рисунок 19 – Модель Order

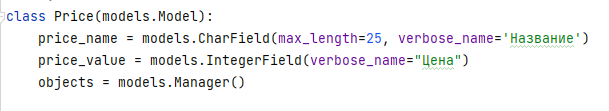


Рисунок 20 – Модель Price

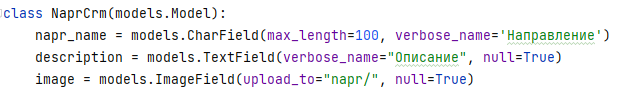


Рисунок 21 – Модель NaprCrm

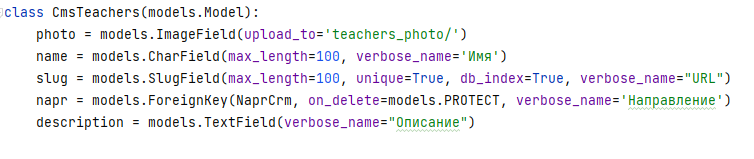


Рисунок 22 – Модель СmsTeacher

* 1. **Разработка ИС**

Подключение к базе данных SQLite осуществляется в конфигурационном файле settings.py (рисунок 23).

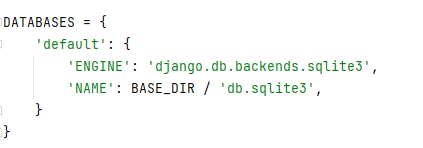


Рисунок 23 – Подключение к базе данных

Так как все таблицы базы данных в Django представлены в виде классов, то каждая отдельная запись в таблице является объектом, и поэтому запросы на выборку, добавление, удаление данных из базы осуществляются с помощью специальных методов, описанных в родительском классе Model.

В функции dashboard осуществлён запрос на выборку всех данных из таблицы Order c помощью метода all() (рисунок 24).

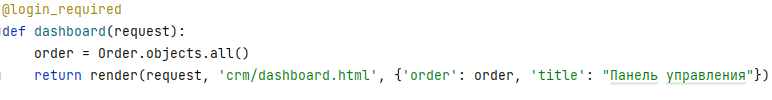


Рисунок 24 – Запрос к базе данных