МИНИСТЕРСТВО ОБРАЗОВАНИЯ РЕСПУБЛИКИ БЕЛАРУСЬ

Учреждения образования «БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ

ТЕХНОЛОГИЧЕСКИЙ УНИВЕРСИТЕТ»

Факультет Информационных Технологий

Кафедра Программной инженерии

Специальность 1-40 05 01 Информационные системы и технологии

Направление специальности 1-40 05 01 03 «Информационные системы и технологии (издательско-полиграфический комплекс)»

**ПОЯСНИТЕЛЬНАЯ ЗАПИСКА**

**КУРСОВОГО ПРОЕКТА:**

по дисциплине «Объектно-ориентированные технологии программирования и стандарты проектирования»

Тема Программное средство «Фитнес-центр»

Исполнитель

Студент 3 курса группы 1 Вакуленчик В.С.

(Ф. И. О.)

Руководитель работы ассистент Панченко О. Л.

(учен. степень, звание, должность, подпись, Ф.И.О.)

Курсовой проект защищен с оценкой

Председатель Смелов В.В.

(подпись)

# **Введение**

Сегодня активное отношение к своему здоровью и физическому развитию занимает ключевое место в повседневной жизни многих людей. Стремление к укреплению здоровья и поддержанию отличной физической формы повышает интерес к фитнес-заведениям.

Основной мотив курсовой работы - разработать программное приложение для спортивного клуба. Это приложение целью имеет оптимизацию взаимодействия с клиентами, обработку покупок абонементов и предоставление информации о тренерском составе.

Чтобы успешно реализовать этот проект, ключевым элементом будет являться база данных. Она будет содержать сведения о посетителях, их индивидуальных программах, графиках занятий, тренерах и специализированных залах. Через приложение пользователи смогут регистрироваться, приобретать абонементы и управлять своим профилем. Администраторы, в свою очередь, получат инструменты для удобного управления базой клиентов, а также организации и предоставления услуг.

Основной целью данного проекта является создание программного обеспечения, способного удовлетворить все вышеуказанные требования. В качестве основы для разработки выбран язык C#. Процесс создания программы включает в себя применение концепций ООП, использование паттернов проектирования, интеграцию с базой данных MS SQL Server и применение технологии Windows Presentation Foundation (WPF).

# **Анализ прототипов и формирование требований к проектируемому программному средству.**

Формулировка задачи и анализ существующих решений — ключевые этапы в процессе выполнения любой работы. Эти этапы способствуют: ясности в установке целей, минимизации повторения уже существующих инициатив и ошибок, а также повышению эффективности и качества конечного продукта.

## **Анализ прототипов**

**Cosmosfit** представляет собой веб-платформу с лаконичным и понятным дизайном, где доминируют черные оттенки, символизирующие фитнес и активный стиль жизни. В верхней части страницы расположено меню, включающее в себя такие важные категории как "Акции", "Клубы", "Партнерам". Дизайн сайта направлен на удобство пользователя, благодаря чему навигация происходит быстро и просто.

На платформе представлена обширная информация относительно фитнес-центра: от услуг и мероприятий до информации о команде, тренерах и способах связи. На стартовой странице выделяются разделы с актуальными новостями, предложениями по абонементам и опцией записи на вводное занятие. Посетители могут глубже погрузиться в содержание каждого фитнес-клуба, просматривая раздел "Клубы".

Cosmosfit предлагает возможность онлайн-регистрации на занятия или ознакомительные тренировки, а также предоставляет форму для связи с управлением центра. Для тех, кто ищет дополнительные советы или знания, на сайте функционирует блог, где регулярно публикуются статьи на тему здоровья, фитнеса и правильного образа жизни.

Визуальное представление интерфейса вы можете увидеть на рисунке 1.1.

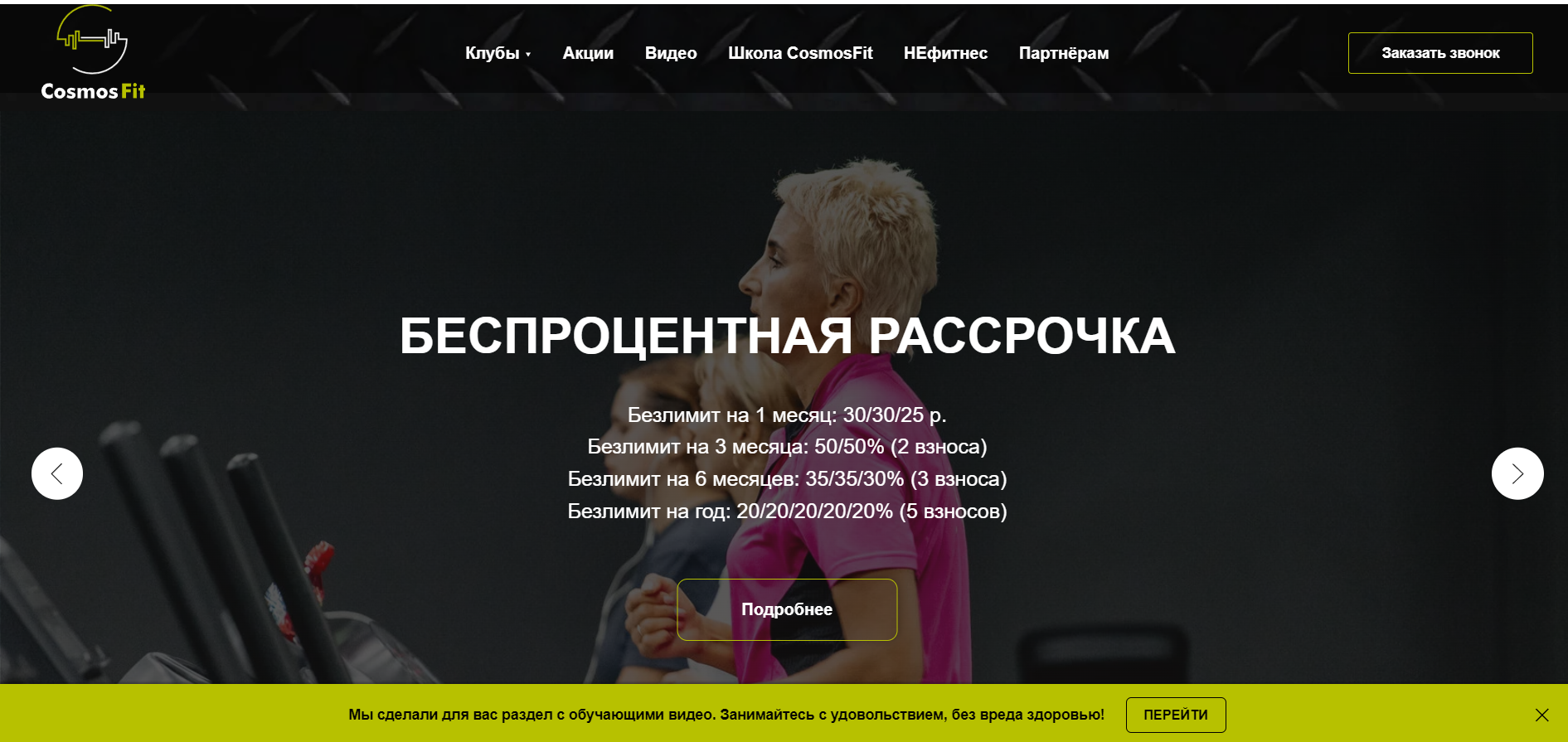


Рисунок 1.1 — Интерфейс приложение «Cosmosfit»

**Evo-club** предлагает своим посетителям стильный и современный дизайн, создающий приятное первое впечатление. Благодаря продуманной структуре, пользователи легко и быстро находят необходимую информацию на сайте. На главной странице вы можете узнать об основных преимуществах и особенностях клуба.

С помощью сайта можно легко подать заявку на участие в тренировках. SportZone предоставляет детальную информацию о предоставляемых услугах, включая актуальные стоимости абонементов и отдельных занятий. Посетители могут ознакомиться с расписанием, выбрать подходящую программу и забронировать место на занятии.

Для удобства пользователей на сайте реализованы формы для связи с администрацией и онлайн-бронирования. Также выделяется отдельный раздел "Контакты", в котором можно найти не только контактные данные, но и карту расположения клуба.

Визуальное оформление веб-платформы иллюстрируется на рисунке 1.2.

|  |
| --- |
|  |

Рисунок 1.2 — Интерфейс приложение «evo-club»

|  |
| --- |
| **MF** **(мир фитнеса)** отличается своим светлым и просторным дизайном, что подчеркивает легкость и вдохновляющую атмосферу клуба. Большие, яркие фотографии придают веб-сайту динамичность и делают его визуально привлекательным. Шрифты, выбранные для дизайна, удобны для чтения и гармонично вписываются в общий стиль.  Навигация по сайту продумана до мелочей, предоставляя пользователю возможность быстро найти интересующую его информацию. В главном меню легко заметны такие ключевые разделы как "Услуги", "О клубе" и "Направления", что облегчает ориентирование посетителей и сокращает время на поиск необходимой информации.  Веб-платформа MF предоставляет детальный обзор всех доступных услуг, их стоимости, а также расписание тренировок, что делает сайт не только информативным, но и функциональным для посетителей.  На сайте реализована удобная навигация, присутствует меню с категориями услуг и разделы с подробным описанием каждой из них. Присутствует возможность поиска информации на сайте. Некоторые разделы, такие как "Услуги", "О клубе" и "Направления", доступны с главной страницы, что позволяет быстро получить нужную информацию.  Сайт содержит подробную информацию об услугах фитнес-центра, их стоимости, расписании занятий и др.  Интерфейс интернет-ресурса представлен на рисунке 1.3. |

Рисунок 1.3 — Интерфейс приложение «mf»

## **1.2 Функциональные требования**

Исследование рассмотренных выше популярных аналогов дает возможность глубоко оценить сильные и слабые стороны существующих решений. Это, в свою очередь, позволяет определить ключевые требования к программному продукту, который разрабатывается в рамках курсовой работы.

Программное средство должно обеспечивать возможность выполнения перечисленных ниже функций:

Функции администратора сервиса:

* Поддержка работы c базой данных;
* отправка на почту клиента одобрение/отклонение абонемента;
* заморозка абонемента;
* составление расписания.

Функции клиента:

* Регистрация, авторизация и загрузка файлов;
* просмотр и бронирование абонемента;
* просмотр информации о тренерах и оставлять отзывы.

## **1.3 Вывод**

После изучения различных фитнес-клубов становится очевидно, что многие из них предлагают схожие услуги и обладают аналогичной концепцией, однако каждый клуб внедряет эти идеи со своими особенностями. Анализируя аналоги фитнес-центров, были выделены ключевые направления для нашего проекта: создание клуба, который предоставляет основные услуги для физического развития (групповые занятия, тренажерный зал, индивидуальные тренировки и т.д.). Должна быть предусмотрена система бронирования тренировок и записи на мероприятия. Помимо этого, клиентам будет доступна возможность сохранять свои достижения, фото до и после тренировок. Клуб также будет предоставлять статистику посещаемости и результатов тренировок за выбранный период. Интерфейс внутри клуба должен быть комфортным, понятным и максимально удовлетворять потребности клиентов.

# **2. Проектирование архитектуры проекта**

Архитектурное проектирование — это ключевой этап в создании любого программного продукта. В этом разделе приведены инструменты, выбранные для реализации проекта, представлена структура базы данных, перечислены интерфейсы, которые будут присутствовать в приложении, а также представлена диаграмма пользовательских сценариев.

## **2.1. Проектирование схемы БД**

Для оптимизированного сохранения и управления информацией приложения создана концептуальная модель базы данных. Эта модель устанавливает организацию данных, их взаимодействие и ключевые таблицы. Далее идет краткий обзор основных таблиц базы данных:

Packages - Информация о пакетах доступа в спортивный центр. Состоит из следующих столбцов:

* Id: Уникальный идентификатор пакета.
* Name: Наименование пакета.
* Duration: Продолжительность действия пакета (в днях).
* AccessHours: Часы доступа в спортивный центр.
* SessionsCount: Количество сессий в пакете.
* Image: Ссылка на изображение пакета.
* Cost: Стоимость пакета.

PackageFeatures - Связывает пакеты с их особенностями или функциями. Состоит из следующих столбцов:

* PackageId: Идентификатор пакета.
* FeatureId: Идентификатор функции или особенности.

Members - Информация о членах спортивного центра. Состоит из следующих столбцов:

* Id: Уникальный идентификатор члена.
* FirstName: Имя члена.
* LastName: Фамилия члена.
* Username: Логин члена.
* EmailAddress: Электронная почта члена.
* Mobile: Мобильный номер члена.
* UserType: Тип пользователя (например, обычный член, инструктор и т.д.).
* PasswordHash: Хэш пароля члена.
* ProfilePicture: Ссылка на профильное изображение члена.

Feedbacks - Отзывы членов о инструкторах. Состоит из следующих столбцов:

* Id: Уникальный идентификатор отзыва.
* FeedbackText: Текст отзыва.
* MemberId: Идентификатор члена, оставившего отзыв.
* InstructorId: Идентификатор инструктора, к которому относится отзыв.

Instructors - Информация о инструкторах спортивного центра. Состоит из следующих столбцов:

* Id: Уникальный идентификатор инструктора.
* FirstName: Имя инструктора.
* Biography: Биография или описание инструктора.
* Picture: Ссылка на изображение инструктора.

InstructorFeatures - Связывает инструкторов с их специализациями или функциями. Состоит из следующих столбцов:

* InstructorId: Идентификатор инструктора.
* FeatureId: Идентификатор функции или специализации.

Reservations - Записи на сессии или занятия. Состоит из следующих столбцов:

* Id: Уникальный идентификатор записи.
* PackageId: Идентификатор пакета, по которому произведена запись.
* MemberId: Идентификатор члена, сделавшего запись.
* BookingStatus: Статус записи (например, подтверждено, в ожидании и т.д.).

Features - Особенности или функции, доступные в спортивном центре. Состоит из следующих столбцов:

* Id: Уникальный идентификатор функции.
* Name: Название функции или особенности.
* Icon: Ссылка на иконку или изображение функции.

На основе этих таблиц и их столбцов была разработана логическая схема БД (Рисунок 2.1).

|  |
| --- |
|  |

Рисунок 2.1 — Логическая схема БД

## **2.2 Описание средств разработки**

Для разработки программного средства «Фитнес-центр» используются:

* среда разработки Microsoft Visual Studio 2022;
* программная платформа .NET;
* язык программирования C#;
* технология WPF;
* SQL Server Manager 2019;

Для создания приложения было решено использовать API Windows Presentation Foundation (WPF). WPF представляет себя как передовой инструмент для разработки клиентских решений под Windows, объединяющий в себе инновационные методы взаимодействия с пользователем на платформе .NET.

Так как основным языком программирования будет C#, необходим инструмент для связи с базой данных. В этом контексте был выбран Entity Framework. Этот инструмент предлагает объектный метод доступа к данным, предоставляя решение для связи объектов и реляционных баз данных (ORM) в рамках .NET Framework. С Entity Framework можно легко осуществлять запросы, используя LINQ to Entities, что значительно упрощает работу с базой данных.

В итоге, применение WPF в сочетании с Entity Framework дает возможность разработать эффективное приложение с современным дизайном интерфейса и быстрым доступом к информации в базе данных.

**2.3 Проектирование окон приложения**

При запуске программного средства «Фитнес-центр» посетитель направляется на экран входа, на котором представлены формы логина и регистрации.

На начальной странице зарегистрированный посетитель может войти в свой профиль, тогда как новые пользователи имеют возможность перейти к форме регистрации через кнопку "Регистрация" и создать свою учетную запись. После этого их автоматически направят обратно на экран входа для доступа к системе.

После успешного входа посетитель направляется на главный экран программы. Стоит упомянуть, что каждая страница включает универсальное меню для удобства перемещения и действий. В верхней части экрана отображаются логин и аватар пользователя, который можно легко обновить, кликнув по нему. Боковое меню служит для удобства навигации.

На вкладке "Абонементы" можно подобрать и приобрести подходящий пакет. Есть инструменты для сортировки и поиска.

В разделе "Инструкторы" предоставлена информация о специалистах и откликах на их работу, а также форма для оставления собственного мнения.

На странице пользователя можно обновить личные данные и просмотреть историю покупок.

Администратору предоставлены инструменты для модификации информации о абонементах и инструкторах. Также у администратора есть удобные средства для оперативной обработки заказов и сортировки.

**2.4 Проектирование User Case диаграммы**

Для визуального осмысления задач системы применяется схема использования, иллюстрирующая, какие возможности предоставляемой программной системы предоставлены для каждой категории пользователей. На схеме использования используются два главных типа элементов: сценарии применения и категории пользователей.

На схеме разные категории пользователей именуются как персонажи и представляют любые единицы, работающие с системой. Любая возможность системы определяется как сценарий применения. Каждый сценарий применения представляет собой комплекс операций, который может быть реализован персонажем для взаимодействия с системой и определяет последовательность действий, осуществляемых системой (Рисунок 2.2).

|  |
| --- |
|  |

Рисунок 2.2 — User Case диаграмма

**2.5 Вывод**

В данном разделе мы осуществили детальный анализ процесса формирования архитектуры нашего приложения, начиная с определения подходящего программного интерфейса и заканчивая конструкцией базы данных и детализацией пользовательского интерфейса. Эта часть проекта устанавливает критерии для коммуникации приложения с конечными пользователями и стратегии обработки информационных потоков. В роли основной платформы для клиентской части был выбран Windows Presentation Foundation (WPF). Для интеракции с базой данных был применён Entity Framework.

Мы сформировали концепцию базы данных, определив ключевые элементы и их связи. Это обеспечивает надежное и эффективное управление данными, что является основой стабильности приложения. Кроме того, был представлен набросок будущих экранов и функциональных разделов, с акцентом на удобство и соответствие ожиданиям пользователей. Сценарная диаграмма позволила нам глубже понять способы взаимодействия пользователей с платформой, предоставляемые инструменты и возможности для решения задач.