

**Техническое задание**

**Выпускного проекта IT Academy 2022**

**Название команды: Mamacitas**

**ФИО и группы участников команды:**

**Керимкулова Айдай CS-W22, Салихова Ирада CS-W22**

## **Содержание:**

*Не редактировать, формируется автоматически*

[1. Общая информация о проекте](#_nd3tb1ioafjb)

[1.3 Цель проекта](#_cxtlvxqcobdb)

[1.4 Аудитория  
есть ли технические навыки, кто сможет пользоваться](#_e6e5b7jdkzxj)

[1.5 Ближайшие аналоги  
чем можно вдохновляться в процессе создания](#_c7og2h4kisg)

[1.6 Стек технологий  
перечислите какие технологии используются в проекте](#_eug9udyvjlwk)

[2. Структура проекта](#_7lcdna2o4uh3)

[2.1 Backend](#_qj5n2uzb4sae)

[2.2 Frontend](#_9uecc5jddj2x)

[2.3 Frontend + Backend: админка](#_v06rukjd7nbg)

[3. Количественные характеристики проекта](#_6wpxrame39yo)

**⚠️Внимание**

**Пожалуйста, заполните все поля. Отсутствие ответа будет засчитано как “0” и снизит ваши баллы, что может привести к недопуску к защитам**

## **Общая информация о проекте**

1.1 Ссылка на репозиторий: <https://github.com/AidaiKeri/CosmeticShop.git>

#### **1.3 Цель проекта**

Помочь клиенту выбрать нужную ему уходовую косметику и оплатить ее немедленно без дополнительных согласований с продавцом. Наш сайт позволяет не тратить свое время на хождение по магазинам, а зайти в онлайн магазин и совершить покупку у нас :)

#### **1.4 Аудитория**

У нас нет ограничений в использовании нашего сайта, он подходит для тех, кто имеет начальные навыки в использовании интернета (мы рассчитываем использовать легкий функционал для новичков в интернете)

**1.5 Ближайшие аналоги**

1) <https://hellobeauty.kg/>

2) <https://www.yves-rocher.ru/>

#### **1.6 Стек технологий**

**.NET** - это платформа от Microsoft, которая позволяет создавать программные приложения;  
**SQL Server** - система управления реляционными базами данных. Основной используемый язык запросов — Transact-SQL;  
**Logger** - unix-утилита, предоставляющая командный интерфейс для модуля системного журнала syslog;  
**GitHub** - веб-сервис для хостинга IT-проектов и их совместной разработки. Веб-сервис основан на системе контроля версий Git;  
**EF Core** - используется в качестве объектно-реляционного модуля сопоставления (O/RM), который:  
• Позволяет разработчикам .NET работать с базой данных с помощью объектов .NET.  
• Устраняет необходимость в большей части кода для доступа к данным, который обычно приходится писать.  
**C#** - современный объектно-ориентированный и типобезопасный язык программирования. C# позволяет разработчикам создавать разные типы безопасных и надежных приложений, выполняющихся в .NET.  
**OOP** - это подход, при котором программа рассматривается как набор объектов, взаимодействующих друг с другом.  
**SOLID** - пять принципов объектно-ориентированного программирования, которые задают архитектуру программы.

Top of Form

## **2. Структура проекта**

#### **2.1 Backend**

**2.1.1 Роли и доступы пользователей:**

|  |  |  |
| --- | --- | --- |
| **Название роли** | **Описание** | **Доступы** |
| Админ | Управляет сайтом | Изменение категорий, общее администрирование сайта |
| Пользователь | Пользуется сайтом | Возможность добавления заявок(запросов) |
|  |  |  |

**2.1.2 Механизмы регистрации, авторизации и восстановления пароля:**

Возможность регистрации, авторизации и восстановления пароля, предоставленные из коробки ASP.NET MVC

**2.1.3 Эндпоинты, содержащие вычисления:**

GET /Products/All выводит все продукты бренда

POST /Accounts/Login регистрация пользователя

#### **2.2 Frontend**

**2.2.1 Страницы**

**Страница 1: <<Cosmetic shop>>**

1. Описание: Главное меню. Общая информация о сайте. Авторизация и регистрация. Поиск. Тест опросник. Доставка. Каталог брендов.
2. Что отображается: Ссылки см.1 и изображения продукции.
3. Нужна ли пагинация: Нет
4. Виды запросов для формирования страницы --
5. По каким полям осуществляется поиск и сортировка, если она нужна:

Сортировка по брендам, поиску.

1. Какие виды валидации нужны, если используется AJAX обновление: --

**Страница 2: <<Личный кабинет>>**

1. Описание: Две функции “Войти” и “Зарегистрироваться”
2. Что отображается: Два поля см.1
3. Нужна ли пагинация: нет
4. Виды запросов для формирования страницы --
5. По каким полям осуществляется поиск и сортировка, если она нужна: нету
6. Какие виды валидации нужны, если используется AJAX обновление: --

**Страница 3: <<Бренды>>**

1. Описание: Список брендов, при нажатие переносит на определенный бренд.
2. Что отображается: Изображения брендов
3. Нужна ли пагинация: да
4. Виды запросов для формирования страницы --
5. По каким полям осуществляется поиск и сортировка, если она нужна: По брендам
6. Какие виды валидации нужны, если используется AJAX обновление: --

**Страница 4: <<Корзина>>**

1. Описание: При нажатие на эту кнопку, перебрасывает на страницу корзины пользователя
2. Что отображается: Сумма к оплате. Имеются ли там товары
3. Нужна ли пагинация: нет
4. Виды запросов для формирования страницы --
5. По каким полям осуществляется поиск и сортировка, если она нужна: --
6. Какие виды валидации нужны, если используется AJAX обновление: --

**Страница 5: <<Тест>>**

1. Описание: Панель “Начать тест”
2. Что отображается: Нажимая на эту кнопку, перекидывает на новую страницу с тестовыми вопросами
3. Нужна ли пагинация: нет
4. Виды запросов для формирования страницы --
5. По каким полям осуществляется поиск и сортировка, если она нужна: Один вопрос и несколько вариантов ответа
6. Какие виды валидации нужны, если используется AJAX обновление: --

**2.2.1 Вычисляемые страницы:**

|  |  |
| --- | --- |
| **Название страницы** | **Описание** |
| **Поиск по брендам** | **Найти нужный продукт** |
| **Личный кабинет** | **Посмотреть информацию о пользователе** |
| **Корзина** | **Посмотреть список добавленных товаров** |
|  |  |
|  |  |

#### **2.3 Frontend + Backend: админка**

|  |  |  |
| --- | --- | --- |
| **Страница** | **Роли с доступом** | **Сортировка** |
| Главная | Все | Yes |
| Личный кабинет | Все | Yes |
| Корзина | Все | Yes |
| Бренды | Все | Yes |
| Тест | Все | No |
|  |  | No |
|  |  | No |
|  |  | No |

#### 

## **3. Количественные характеристики проекта**

|  |  |
| --- | --- |
| **С#** | |
| Количество таблиц | 6 tables |
| Количество моделей | 5 models |
| Количество пользователей (ролей) | user types: 2 (admin, user) |
| Количество классов | 10 classes |
| Views counts | 10 |